infra MAGAZINE

Edição 01 :: Ano I

DevMedia

ESPECIAL: Virtualização com o Hyper-V
Entenda os conceitos e adote a virtualização em seu ambiente de TI

Configurando o Servidor DHCP3
Simplifique a administração de redes Windows e Linux

Monitoramento de Redes de Computadores
Trabalhando com a ferramenta Nagios



Segurança

Utilizando Hardening e Sistemas de Detecção de Intrusão

Configurações básicas no Apache2
Conheça o mais famoso e bem sucedido servidor Web

Gateway com FreeBSD
Compartilhando a internet com um SO robusto e de baixo custo

RENOVE JÁ!

Sua assinatura pode estar acabando



Renovando a assinatura de sua revista favorita você ganha brindes e descontos exclusivos.

Faça a renovação de sua assinatura agora mesmo.



www.devmedia.com.br/renovacao ou (21)3382-5038

DevMedia

Sumário

Olá, eu sou o **DevMan**! Desta página em diante, eu estarei lhe ajudando a compreender com ainda mais facilidade o conteúdo desta edição. Será um prazer contar com sua companhia! Confira abaixo o que teremos nesta revista:



Easy Infra, Servidores

06 – Windows Server 2008 R2
Instalação e configurações básicas
[*Fabiano de Santana*]

Easy Infra, Servidores

13 – Configurações básicas no Apache2
Instalação, configurações, desempenho e segurança em Sistemas Linux
[*Ricardo Franzen*]

Segurança, Servidores

18 – Hardening
Blindando um Sistema GNU/Linux
[*Flávio Alexandre dos Reis, Marcos Fabiano Verbena e Eduardo Pagani Julio*]

Segurança, Redes

26 – Sistema de Detecção de Intrusão
Uma abordagem da segurança em redes utilizando o Prelude-IDS
[*Edelberto Franco Silva e Eduardo Pagani Julio*]

Sistemas Operacionais, Redes, Web

36 – Gateway com FreeBSD
Compartilhe a internet com FreeBSD
[*Rogério Tomassoni de A. Junior*]

Servidores, Core, Tutorial

48 – Configurando o Servidor DHCP3
DHCP no Ubuntu como ferramenta de administração de redes transparente
[*Racy Rassilan*]

Desempenho, Tutorial

54 – Virtualização com o Hyper-V
Entenda os conceitos e adote a virtualização em seu ambiente de TI
[*Jorge Barata*]

Boas Práticas, Segurança, Redes

64 – Monitoramento de Redes de Computadores
Trabalhando com a ferramenta Nagios
[*Renata Benini e Marcelo Daibert*]

[Easy Infra] Artigos introdutórios sobre infraestrutura: redes, servidores, sistemas operacionais, ferramentas, etc.

[Servidores] Destaque para itens como instalação, configuração e gerência, de servidores DHCP, DNS, Web, etc.

[Sistemas Operacionais] Foco em características de sistemas operacionais como Windows, Linux, FreeBSD, entre outros.

[Redes] Artigo dentro do contexto de Redes: configuração, ferramentas de monitoração/gerenciamento, entre outros.

[Segurança] Artigo dentro do contexto de Segurança: redes, servidores, sistemas operacionais, infraestrutura em geral.

[Core] Técnicas, ferramentas, e outros assuntos que fogem ás demais categorias.

[Tutorial] Artigo no estilo tutorial passo a passo.

[Desempenho] Matérias que abordam questões relacionadas a desempenho.

infra MAGAZINE

Ano I • Edição 01 • 2011

Edição

Editor

Eduardo Spínola (*eduspinola@gmail.com*)

Arte

Capa e Diagramação Romulo Araujo (romulo@devmedia.com.br)

Produção

Gerência de Marketing Kaline Dolabella (kalined@terra.com.br)

Jornalista Responsável Kaline Dolabella - JP24185

Revisão e Supervisão Thiago Vincenzo (thiago.v.ciancio@devmedia.com.br)

Coordenação Geral Daniella Costa (daniella@devmedia.com.br)

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

Cristiany Queiroz

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 Easy Java Magazine, entre em contato com o editor, informando o título e mini-resumo do tema que você gostaria de publicar:

Eduardo Spínola - Editor da Revista
eduspinola@gmail.com

DevMedia

Editorial

A DevMedia sempre se mostrou aberta às novidades do mercado. Se analisarmos os últimos anos, veremos que nosso leque não se manteve fechado a revistas. Realizamos grandes investimentos em eventos, cursos, vídeo aulas, consultorias, e em novos formatos para as nossas revistas. Hoje, além da edição impressa, também disponibilizamos um grande diferencial para nossos leitores, a revista digital. Isso confirma que estamos sempre em busca de oferecer os produtos que você precisa e que se adequem às suas necessidades.

Neste contexto, recentemente começamos a projetar uma nova revista, sobre uma área que até encontramos muito conteúdo na web, mas não com a qualidade que estamos acostumados a proporcionar. Pensando nisso, decidimos lançar a Infra Magazine, uma revista totalmente digital voltada para profissionais que lidam com questões de infraestrutura, como administração de redes, o que envolve segurança e desempenho, Sistemas Operacionais, novidades da área de TI, instalação, configuração e manutenção de servidores, entre tantos outros temas.

O nosso objetivo não é prolongar a "briga" entre Sistemas Operacionais ou qualquer outro confronto que possa existir na área, mas sim apresentar o que há de melhor e pior em cada tecnologia, para que você, leitor, possa analisar e identificar qual das opções se encaixa melhor às suas necessidades.

Agora, vamos conhecer os artigos desta primeira edição! O destaque de capa está relacionado com um tema que preocupa qualquer administrador, a segurança. Por isso, planejamos dois artigos nesta área. **Hardening – Blindando um Sistema GNU/Linux**, tem como objetivo explicitar de forma prática algumas técnicas que consistem na implementação de diretivas de segurança que devem ser seguidas antes, durante e após a instalação e configuração de servidores GNU/Linux. Na vídeo-aula elaborada para esta matéria, saiba como proteger o ambiente de TI através de técnicas de Hardening. No segundo, **Sistema de Detecção de Intrusão**, serão abordadas definições e modelos de IDS, e uma ferramenta disponível para o usuário. Esta ferramenta é o Prelude-IDS, que servirá como demonstrativo da teoria comentada em um ambiente real.

Outro tema de grande importância e bastante ponderado nos últimos meses é a virtualização. Em **Virtualização com o Hyper-V**, conheceremos o seu conceito, as principais ferramentas e, por fim, faremos uma introdução ao Hyper-V, quando exibiremos, na prática, o seu uso. No vídeo desta matéria, veja como adotar virtualização no seu ambiente de trabalho.

No artigo **Gateway com FreeBSD**, mostraremos como instalar e configurar o FreeBSD – um sistema operacional robusto e de baixo custo – para atuar como gateway de uma rede local de computadores. Ao final, o leitor estará apto a colocar em produção o FreeBSD para realizar o compartilhamento de internet com a rede, independente dos Sistemas Operacionais que os computadores desta utilizem.

O que você acha de aprender a configurar um servidor Web Apache? **Configurações básicas no Apache2** ensina como instalar esta ferramenta de duas maneiras: compilando o código fonte e através dos próprios pacotes da distribuição. Além disso, serão apresentadas dicas de desempenho e segurança, recomendáveis para qualquer servidor. No vídeo deste artigo destacaremos algumas opções para aumentar a segurança do seu servidor web.

Monitoramento de Redes de Computadores apresenta uma definição sobre o uso de servidores e ativos de redes e a importância de monitorar uma rede de computadores. Para isso, analisaremos algumas ferramentas de monitoramento e um estudo de caso com o Nagios, uma aplicação de código aberto que permite monitorar tanto hosts quanto serviços, alertando o administrador quando ocorrerem problemas.

Ainda nesta edição, aprenderemos a **Configurar o Servidor DHCP3**. Esta matéria explica como funciona o DHCP, e como implementá-lo para garantir uma melhor administração de redes Windows e Linux em uma organização. Para enriquecer o conteúdo, disponibilizamos um vídeo que expõe os conceitos e demonstra o que é preciso para preparar o servidor DHCP.

Para finalizar, a instalação e configuração de um Sistema Operacional que atue como servidor é uma das tarefas básicas e primordiais executadas em qualquer organização. Nesse sentido, esta matéria apresenta um tutorial explicando como realizar esta atividade considerando o **Windows Server 2008 R2**. A instalação de um sistema operacional é avaliada como uma tarefa simples, no entanto, seu planejamento é fundamental.

Com isso, concluímos a quarta edição da Easy Java.

Boa leitura e até a próxima!



Eduardo Oliveira Spínola
eduspinola@gmail.com
twitter.com/Java_Magazine

Dificuldades em encontrar conteúdo sobre .Net para iniciantes?



A DevMedia simplifica pra você!

easy .net magazine

Easy .Net Magazine:

A primeira revista digital feita especialmente para quem está dando os primeiros passos em .Net!

Você encontrará exemplos e modelos fáceis e práticos, noções introdutórias e conceitos básicos.

Conheça as vantagens:

- Vídeos e apresentações "embutidos" nas páginas;
- Ilustrações, diagramas e modelos apresentados em PowerPoint;
- Entrevistas em vídeo;
- Códigos-fonte que podem ser copiados com CTRL+C/CTRL+V.

A Easy.Net não é feita em PDF ou Flash!
Ela utiliza um aplicativo 100% online.



DevMedia group

Acesse agora a edição Nº 1 totalmente grátis e comprove

www.devmedia.com.br/easynet

Easy.Net Magazine, seu curso inicial de .Net!

Windows Server 2008 R2

Instalação e configurações básicas

Fabiano de Santana

O objetivo deste artigo é apresentar os passos que devemos seguir para efetuar com sucesso a instalação do Windows Server 2008 R2, bem como sua configuração básica. É importante ressaltar que a instalação de um sistema operacional é extremamente simples, porém, seu planejamento é algo fundamental. Quando falamos em planejamento nos referimos a responder perguntas como: qual será a função do servidor? Que aplicações serão instaladas nesse servidor? Quantos usuários acessarão esse servidor simultaneamente? Esse servidor será acessado através da Internet? E assim por diante.

Com essas perguntas básicas já podemos ter uma visão um pouco mais ampliada do quanto o planejamento é importante para o negócio de uma empresa. Qualquer falha ou erro no dimensionamento inicial do servidor pode custar muito caro para a empresa, devido a indisponibilidades e necessidades de atualizações de hardware.

Portanto, recomendamos fortemente que um planejamento prévio seja feito juntamente com as áreas de negócio da empresa. Para mais informações sobre esse planejamento, indicamos os links abaixo:

- Microsoft Assessment and Planning (MAP) Toolkit for Windows Server 2008 R2: *http://technet.microsoft.com/en-us/solutionaccelerators/dd537573*;
- Upgrade to Windows Server 2008 R2 with SP1: *http://www.microsoft.com/windowsserver2008/en/us/why-upgrade.aspx*.

Neste artigo partiremos do princípio de que você já tenha definido toda a estratégia e planejou durante alguns dias a implementação do Windows Server 2008 R2. Além disso, antes de iniciarmos a instalação de nosso servidor, é importante estarmos atentos a dois conjuntos de informações fundamentais:

- Os pré-requisitos de instalação do sistema operacional;
- Os caminhos possíveis para atualização do sistema operacional.

Iremos agora conhecer os requisitos mínimos para instalação do Windows Server 2008 R2.

Resumo DevMan

De que se trata o artigo:

Este artigo apresenta, passo a passo, como devemos proceder para realizar a instalação e configuração inicial do Windows Server 2008 R2.

Para que serve:

A instalação e configuração de um sistema operacional que atue como servidor é uma das tarefas básicas e primordiais executadas em qualquer organização. Nesse sentido, este artigo apresenta um tutorial explicando como realizar esta tarefa considerando o Windows Server 2008 R2.

Em que situação o tema é útil:

No dia a dia dos analistas de suporte que possuem como uma de suas tarefas a instalação e configuração do Windows Server 2008 R2.

Componente	Requisito Mínimo
Processador	Mínimo: 1.4 GHz (processador x64) Observação: Caso esteja utilizando o Windows Server 2008 R2 for Itanium, será necessário um processador Intel Itanium 2.
Memória	Mínimo: 512 MB RAM Máximo: 8 GB (Foundation Edition) ou 32 GB (Standard Edition) ou 2 TB (Enterprise, Datacenter e Itanium-Based Systems).
Requisitos de Espaço em Disco	Mínimo: 32 GB ou mais Observação: Para computadores que possuam mais de 16 GB de RAM, será necessário mais espaço em disco para paginação, hibernação e armazenamento de arquivos.
Monitor	Super VGA (800×600) ou monitor de maior resolução.
Outros	Unidade de DVD, Teclado e Mouse.

Tabela 1. Requisitos de Instalação

Requisitos de instalação

Podemos observar na **Tabela 1** os requisitos mínimos para instalação do Windows Server 2008 R2. É importante lembrar que cada versão deste S.O. pode ter suas particularidades com relação a requisitos mínimos de hardware. Para analisar estes requisitos, você deve também levar em conta os dados levantados na fase de planejamento.

Conhecidos os pré-requisitos, vamos conhecer agora os possíveis caminhos de atualização caso você esteja migrando de um servidor mais antigo.

Caminhos de atualização suportados

Esse também é um item fundamental e que deve obrigatoriamente fazer parte do planejamento da instalação do Windows

Sistema Operacional de Origem	Sistema Operacional de Destino
Windows Server 2003 Standard Edition com Service Pack 2 (SP2) ou Windows Server 2003 R2 Standard Edition	Windows Server 2008 R2 Standard, Windows Server 2008 R2 Enterprise
Windows Server 2003 Enterprise Edition com SP2 ou Windows Server 2003 R2 Enterprise Edition	Windows Server 2008 R2 Enterprise, Windows Server 2008 R2 Datacenter
Windows Server 2003 Datacenter Edition com SP2 ou Windows Server 2003 R2 Datacenter Edition	Windows Server 2008 R2 Datacenter
Instalação Server Core do Windows Server 2008 Standard com ou sem SP2	Instalação Server Core do Windows Server 2008 R2 Standard ou Windows Server 2008 R2 Enterprise
Instalação Server Core do Windows Server 2008 Enterprise com ou sem SP2	Instalação Server Core do Windows Server 2008 R2 Enterprise ou Windows Server 2008 R2 Datacenter
Instalação Server Core do Windows Server 2008 Datacenter	Instalação Server Core do Windows Server 2008 R2 Datacenter
Instalação Server Core do Windows Web Server 2008 com ou sem SP2	Instalação Server Core do Windows Server 2008 R2 Standard ou Windows Web Server 2008 R2
Instalação completa do Windows Server 2008 Standard com ou sem SP2	Instalação completa do Windows Server 2008 R2 Standard ou Windows Server 2008 R2 Enterprise
Instalação completa do Windows Server 2008 Enterprise com ou sem SP2	Instalação completa do Windows Server 2008 R2 Enterprise ou Windows Server 2008 R2 Datacenter
Instalação completa do Windows Server 2008 Datacenter com ou sem SP2	Instalação completa do Windows Server 2008 R2 Datacenter
Instalação completa do Windows Web Server 2008 com ou sem SP2	Instalação completa do Windows Server 2008 R2 Standard ou Windows Web Server 2008 R2
Instalação Server Core do Windows Server 2008 R2 Standard	Instalação Server Core do Windows Server 2008 R2 Standard (reparação in-loco) ou Windows Server 2008 R2 Enterprise
Instalação Server Core do Windows Server 2008 R2 Enterprise	Instalação Server Core do Windows Server 2008 R2 Enterprise (reparação in-loco) ou Windows Server 2008 R2 Datacenter
Instalação Server Core do Windows Server 2008 R2 Datacenter	Instalação Server Core do Windows Server 2008 R2 Datacenter (reparação in-loco)
Instalação Server Core do Windows Web Server 2008 R2	Instalação Server Core do Windows Web Server 2008 R2 (reparação in-loco) ou Windows Server 2008 R2 Standard
Instalação completa do Windows Server 2008 R2 Standard	Instalação completa do Windows Server 2008 R2 Standard (reparação in-loco) ou Windows Server 2008 R2 Enterprise
Instalação completa do Windows Server 2008 R2 Enterprise	Instalação completa do Windows Server 2008 R2 Enterprise (reparação in-loco) ou Windows Server 2008 R2 Datacenter
Instalação completa do Windows Server 2008 R2 Datacenter	Instalação completa do Windows Server 2008 R2 Datacenter (reparação in-loco)
Instalação completa do Windows Web Server 2008 R2	Instalação completa do Windows Web Server 2008 R2 (reparação in-loco) ou Windows Server 2008 R2 Standard

Tabela 2. Caminhos de migração

Server 2008 R2. Na **Tabela 2** podemos observar a relação de caminhos de atualização suportados por este S.O.

Além das restrições apresentadas na **Tabela 2**, algumas considerações importantes quanto a instalação do Windows Server 2008 R2 partir de um outro sistema operacional são:

- Não é possível migrar um sistema operacional 32 bits para 64 bits. Aqui é importante salientar que o Windows Server 2008 R2 existe apenas na versão 64 bits;
- Não é possível migrar um sistema operacional de um idioma para o outro.

Neste momento, tendo o planejamento de instalação definido e conhecidos os pré-requisitos e caminhos possíveis de atualização, partiremos para a instalação e configuração do Windows Server 2008 R2.

Instalação e configuração do Windows Server 2008 R2

Agora que já vimos os requisitos mínimos para instalação do Windows Server 2008 R2, vamos para a parte prática. Inicialmente faremos a instalação da versão Enterprise (ler **Nota DevMan 1**).

Para fazer o download do Windows Server 2008 R2 para testes, visite o link *http://msdn.microsoft.com/pt-br/evalcenter/ee175713*.

A partir de agora, veremos passo a passo o processo completo de instalação do Windows Server 2008 R2:

1. Insira o DVD do Windows Server em seu computador. Automaticamente a tela da **Figura 1** será exibida. Defina o idioma do sistema operacional, o formato de data e hora e a configuração do teclado. Após isso clique em *Avançar*;
2. Na próxima tela você tem a opção de iniciar a instalação ou entrar no modo de *Reparar o Computador*. A título de curiosidade, escolheremos a opção *Reparar o Computador* para ver as opções disponíveis;
3. Na tela que será exibida clique em *Avançar* (ver **Figura 2**);
4. Na próxima tela serão exibidas as opções disponíveis para reparação do sistema (ver **Figura 3**):

– **Recuperação da Imagem do Sistema**: com essa opção você pode fazer uma



Nota do DevMan 1

Server Core

Não veremos neste artigo o processo de instalação do Windows Server 2008 R2 Server Core, que é o novo tipo de instalação do Windows Server que não possui interface gráfica, ou seja, nada de telas quando executado localmente. Caso a instalação seja remota, podemos trabalhar usando os consoles de administração do MMC. O processo é extremamente simples e praticamente o mesmo quando utilizamos o assistente de instalação (apresentado neste artigo).



Figura 1. Tela Inicial da Instalação do Windows Server 2008 R2.

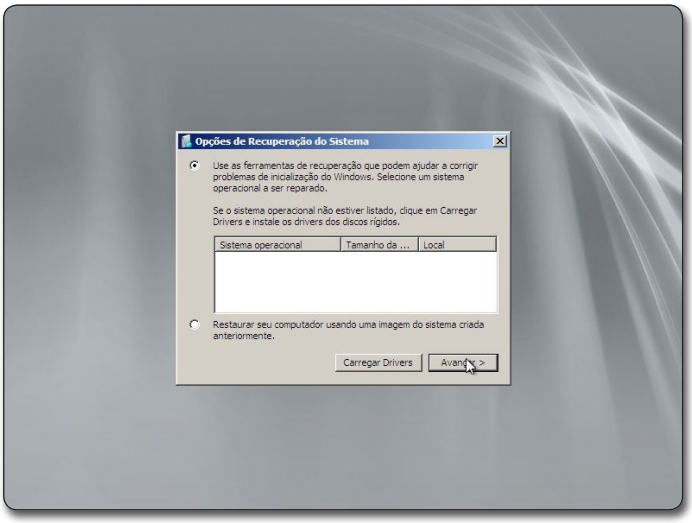


Figura 2. Opções de Recuperação do Sistema

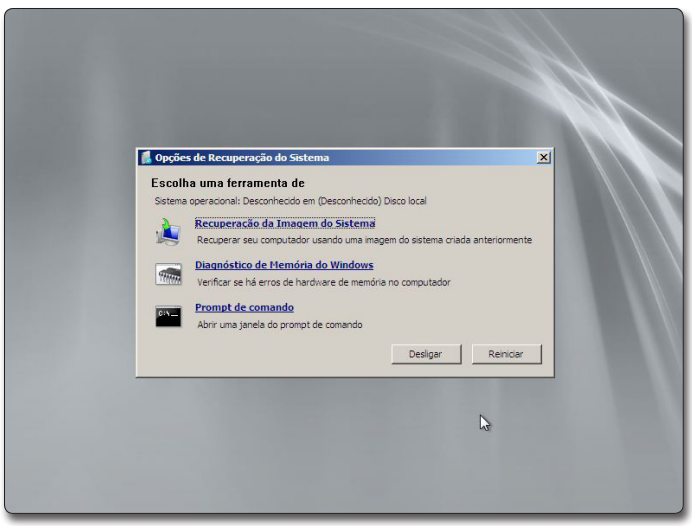


Figura 3. Opções de Recuperação do Sistema.

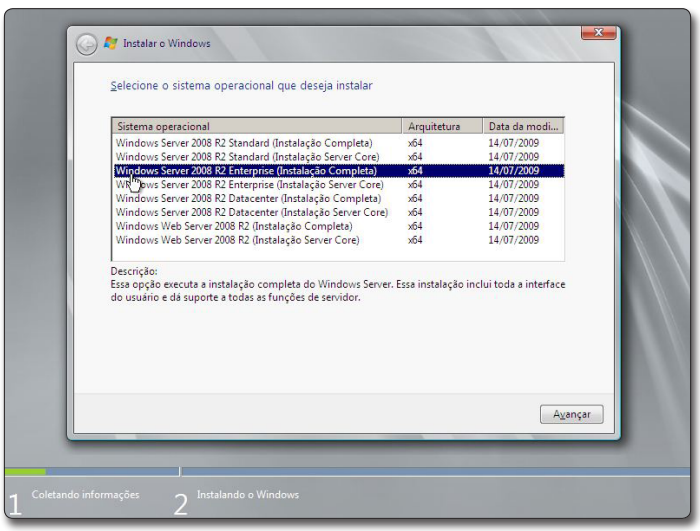


Figura 4. Seleção do Sistema Operacional que será instalado

restauração do Windows a partir de um backup completo existente. Esta ação formatará o disco rígido do computador e fará a restauração a partir do backup existente;

– **Ferramenta de Diagnóstico de Memória do Windows:** essa ferramenta nos permite fazer um diagnóstico na memória RAM do computador. Deste modo podemos detectar se o computador está com problemas relacionados com a memória física;

– **Prompt de Comando:** essa opção abre o prompt de comando do Windows para que você possa executar os comandos disponíveis.

5. Bom, agora que você já conheceu as opções de reparação do sistema, vamos voltar para a tela inicial de instalação e escolher a opção *Instalar Agora*. Feito isso, será dado início ao processo de instalação;

6. Na próxima tela, você deverá selecionar a edição do sistema operacional que deseja instalar. Nesse caso vamos selecionar a edição Enterprise Completa e clicar em *Avançar* (ver **Figura 4**);

7. Na próxima tela, leia e aceite os termos de licença. Após isso clique em *Avançar*;

8. Em seguida você tem a opção de Atualização, que estará disponível somente quando você já tiver uma versão do Windows instalada que suporte a atualização para o Windows Server 2008 R2. Essa opção mantém todos os arquivos, configurações e programas existentes. Mas no nosso caso, vamos fazer uma nova instalação, e para isso clicaremos em *Personalizada* (ver **Figura 5**);

9. Na próxima tela você pode selecionar o disco no qual o sistema operacional será instalado. Nessa tela você também pode exibir algumas opções adicionais. Para isso clique em *Opções de Unidade* (ver **Figura 6**);

10. Veja que na tela da **Figura 7** novas opções são exibidas, como *Formatar*, *Excluir*, *Novo*, etc. Para nosso exemplo, vamos selecionar o disco/partição existente e clicar em *Avançar*;

11. Agora a instalação será realmente iniciada. Assim, basta aguardar até que seja concluída (ver **Figura 8**);

12. Após a instalação ser concluída, você deverá alterar a senha da conta de usuário Administrador;

13. Dessa forma, na próxima tela, digite duas vezes a nova senha do Administrador e clique na seta azul. Após a senha ser trocada, clique em *OK* (ver **Figura 9**);
14. Pronto. Aqui finalizamos a instalação do Windows Server 2008 R2. Um processo extremamente simples e rápido.

Após a instalação ser concluída, automaticamente o assistente *Tarefas de Configuração Iniciais* será aberto, como pode ser visto na **Figura 10**. Esse é um assistente muito interessante que nos permite fazer todas as configurações básicas do nosso sistema, como por exemplo, ativar a instalação do Windows, definir um fuso horário, fazer as configurações de rede, alterar o nome do servidor, e assim por diante.

A primeira configuração que poderemos fazer é a ativação do Windows, desde que esse esteja conectado com a Internet. Para isso, clique em *Ativação do Windows*. Na tela apresentada na **Figura 11** digite o Product Key do Windows e clique em *Avançar*.

Após o Windows ser ativado, será exibida uma tela de sucesso informando que você está utilizando um produto Original. Clique em *Fechar*.

Agora, voltando ao assistente *Tarefas de Configuração Iniciais*, clique em *Fornecer o nome e o domínio do computador*, para que possamos configurar o nome do nosso servidor. Na tela apresentada na **Figura 12** digite o nome para o servidor e clique em *Ok* até que todas as telas sejam fechadas. Nessa etapa será solicitado um restart no servidor para que o novo nome seja configurado.

Aqui é importante citar que é fundamental que você tenha um padrão para nomes de servidores. De preferência, devem ser utilizados nomes intuitivos, fáceis de ser lembrados e que identifiquem a função de um servidor. Por exemplo, o nome SRV-Files01 significa que é um servidor de arquivos. O nome SRV-AD significa que é um servidor com o Active Directory. Enfim, são apenas alguns exemplos para ilustrar a importância de se ter um padrão para definição de nomes.

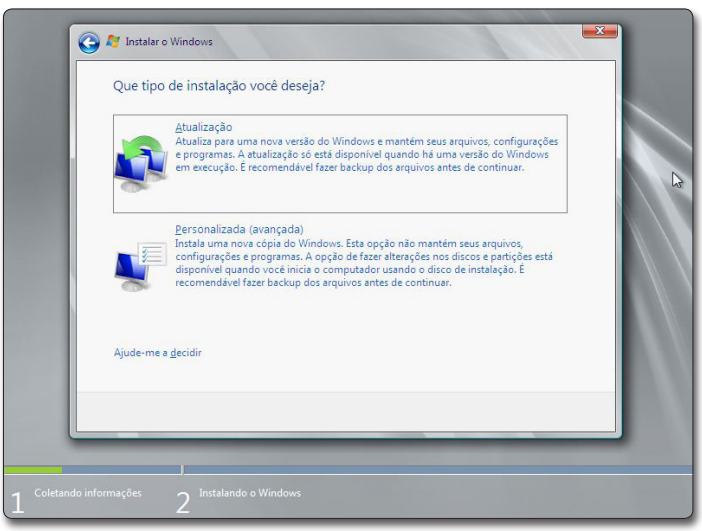


Figura 5. Tipo de Instalação.

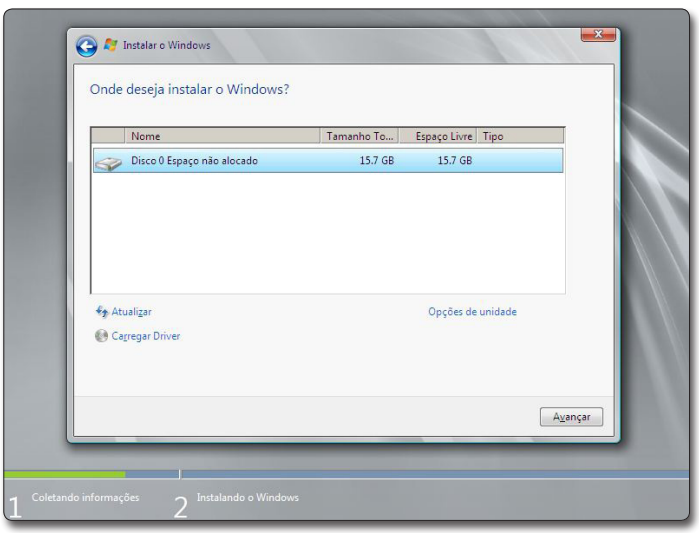


Figura 6. Seleção da partição / disco

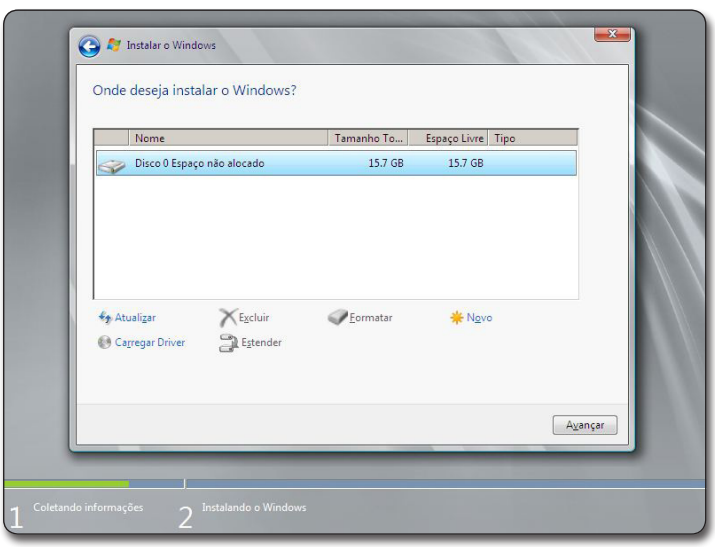


Figura 7. Seleção da partição/disco.

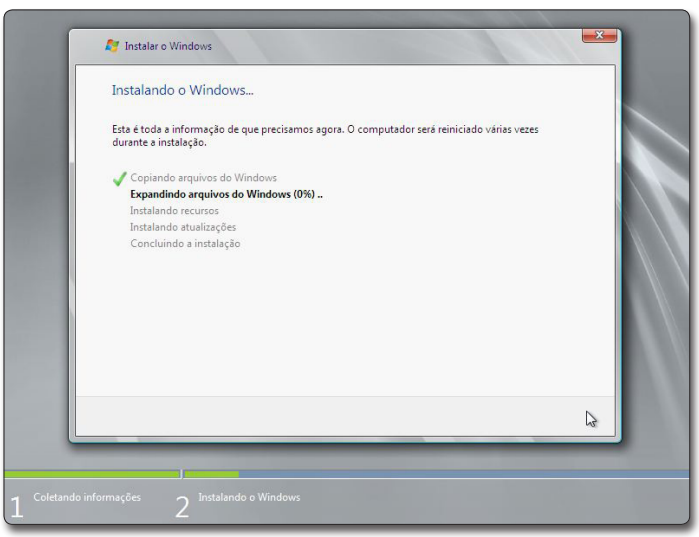


Figura 8. Instalando o Windows.

Neste momento, voltando ao assistente *Tarefas de Configuração Iniciais*, clique em *Configurar Rede*. Essa configuração também é fundamental para um servidor, pois na grande maioria dos casos todos os nossos funcionários, ou clientes, acessarão esse servidor. Deste modo, essa é uma configuração primordial que permitirá ao servidor ser acessado através da rede da empresa. Outro ponto importante é sempre configurar os servidores com IPs estáticos. Na tela da **Figura 13** faça as configurações desejadas e clique em *OK* até que todas as telas sejam fechadas.

Outra configuração extremamente importante são as atualizações do sistema operacional, ou seja, as correções liberadas mensalmente pela Microsoft.

Uma das tarefas essenciais hoje em dia é manter o sistema operacional e suas aplicações atualizadas, com todas as atualizações de segurança instaladas. Para quem não sabe, todos os meses a Microsoft lança uma série de atualizações de segurança para o Windows e outros softwares (na segunda terça-feira de todos os meses). Esporadicamente, atualizações são lançadas mais de uma vez por mês. É fundamental que você instale essas atualizações, pois com isso seu sistema estará protegido das vulnerabilidades já conhecidas no mercado. Instalar as atualizações significa aumentar o nível de segurança do seu computador.

Com a ferramenta Windows Update do Windows Server 2008 R2 podemos manter nosso sistema operacional e alguns aplicativos como o Microsoft Office, Internet Explorer e Windows Defender, atualizados. Através do Painel de Controle podemos realizar todas essas tarefas rapidamente, além de definir se a verificação e download de atualizações será automática ou não. É importante salientar que para fazer alterações no Windows Update do Windows Server 2008 você precisará possuir privilégios administrativos. Na tela da **Figura 14** você



Figura 9. Trocar a senha do Administrador

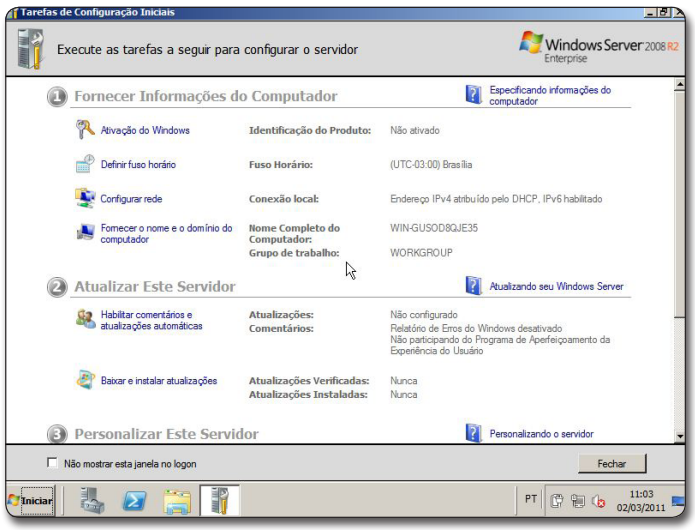


Figura 10. Tarefas de Configuração Iniciais

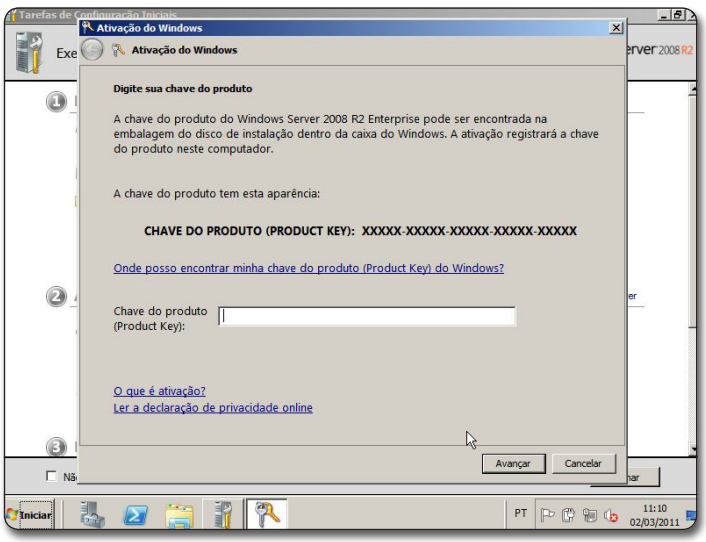


Figura 11. Ativação do Windows

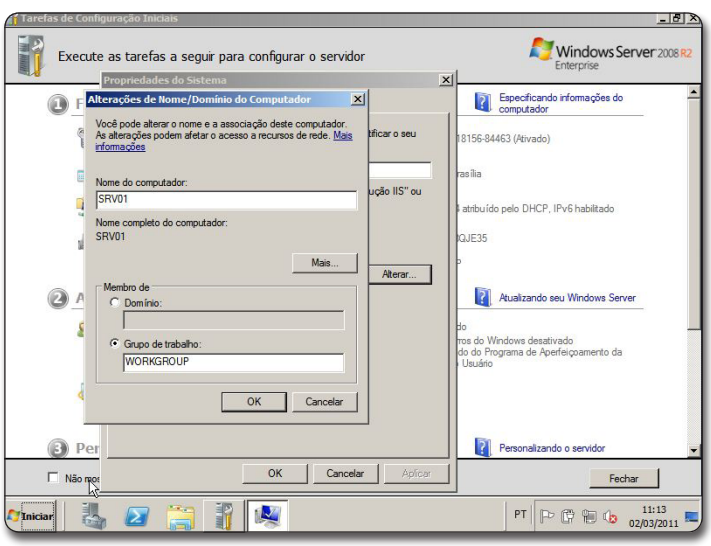


Figura 12. Alterar o nome do servidor (hostname).

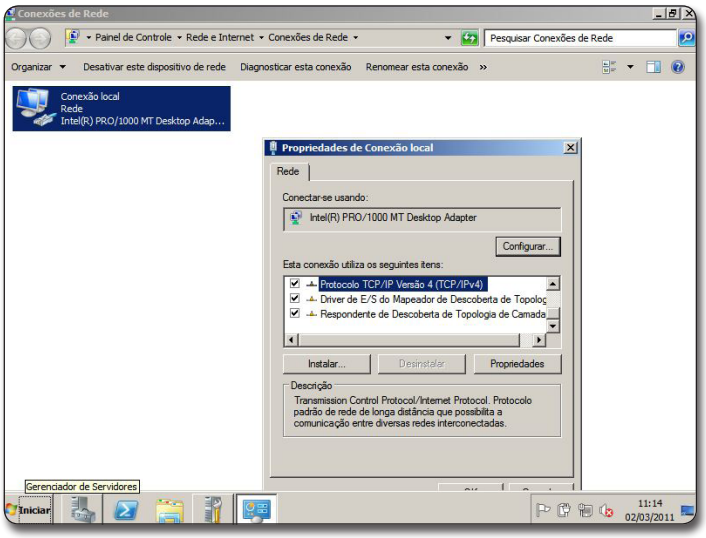


Figura 13. Configurações TCP/IP.

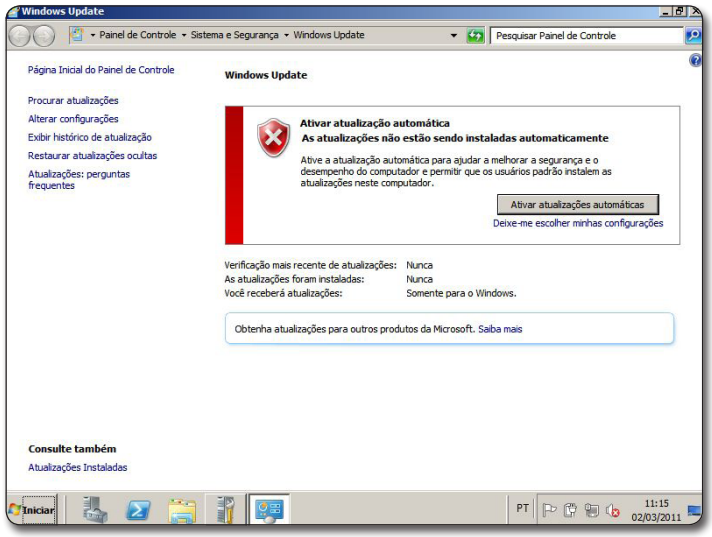


Figura 14. Windows Update

pode clicar em *Deixe-me escolher minhas configurações*.

Para fazer as configurações do Windows Update basta clicar em *Alterar Configurações* no canto superior esquerdo. Na tela que será aberta podemos definir se as atualizações serão instaladas automaticamente – observe a **Figura 15**. Caso a opção *Instalar Atualizações Automaticamente* seja selecionada, podemos definir os dias da semana e horários que o Windows Update fará a busca por novas atualizações e o download automaticamente.

Temos também a opção *Baixar as atualizações, mas deixar-me optar por instalá-las*. Ao selecionar essa opção, o Windows Update fará apenas o download das atualizações, e você precisará instalá-las manualmente.

A outra opção disponível é *Procurar atualizações, mas deixar-me optar por baixá-las e instalá-las*. Nesse caso, o Windows Update não fará o download das atualizações e você precisará baixar e instalar as atualizações manualmente.

Por fim, temos a opção *Nunca verificar se há atualizações*, que simplesmente desabilita o Windows Update, ou seja, as atualizações não serão verificadas pelo Windows. Vale lembrar que não é uma boa prática de segurança escolher essa opção.

Para servidores, recomendamos que você selecione a opção *Baixar as atualizações, mas deixar-me optar por instalá-las*.

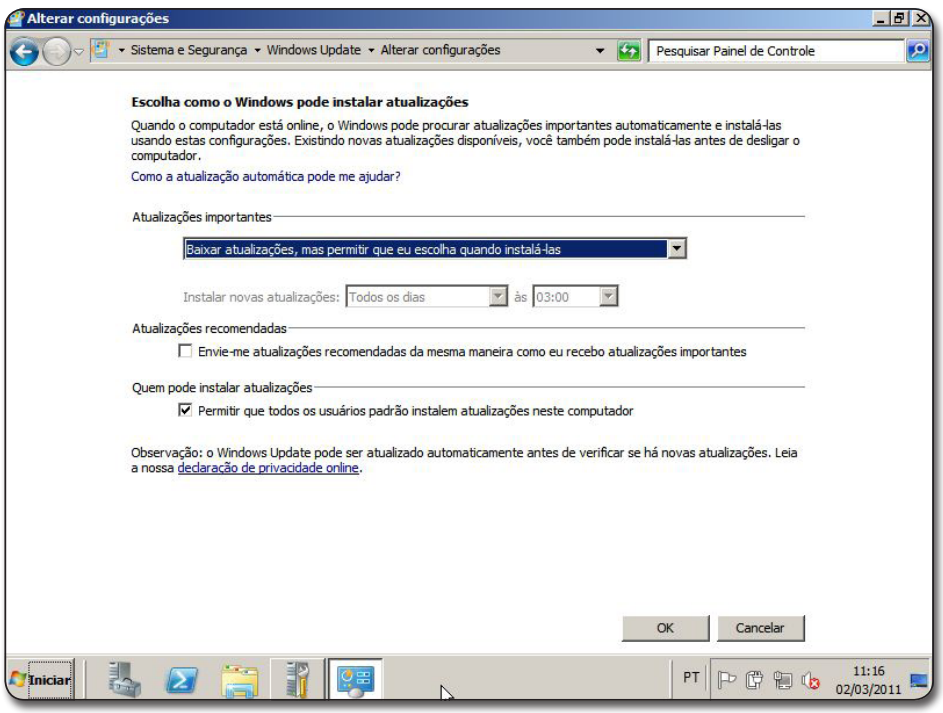


Figura 15. Windows Update

Conforme dissemos anteriormente, grande parte das atualizações pede a reinicialização do sistema operacional após a instalação. E um servidor não pode ser reinicializado a qualquer momento. Isto é, você deverá, com antecedência, programar essa manutenção no servidor.

Na tela principal do Windows Update temos ainda a opção *Exibir histórico de atualização*. Aqui você pode visualizar todas as atualizações que foram instaladas no servidor, as datas em que isso aconteceu e se as instalações ocorreram com sucesso ou não. Você pode também clicar duas vezes sobre uma atualização para obter mais informações, como por exemplo, uma descrição e o site da Microsoft com mais detalhes.

Por último temos a opção *Restaurar as atualizações ocultas*. Caso necessário, você poderá ocultar alguma atualização que não seja interessante para seu servidor.

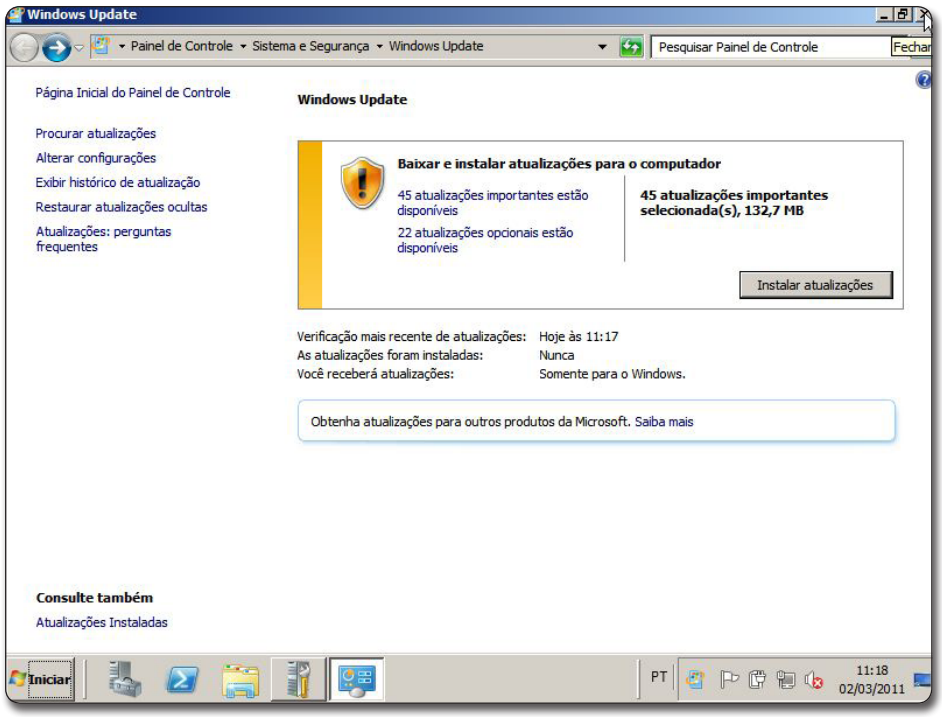


Figura 16. Windows Update - Atualizações Pendentes.

Posteriormente, você poderá restaurar essas atualizações de forma que possa instalá-las. Para restaurar uma atualização oculta, basta selecionar as atualizações desejadas e clique em *Restaurar*.

Na **Figura 16** temos uma demonstração de atualizações disponíveis para instalação no servidor que acabamos de instalar. Para instalar essas atualizações basta clicar em *Instalar atualizações*. Após isso, é só reiniciar o servidor.

Conclusão

Com isso concluímos nosso processo de instalação do mais novo sistema operacional da Microsoft voltado para servidores, o Windows Server 2008 R2. Conforme vocês puderam observar, é um processo rápido e simples, porém, destacamos novamente que a fase de planejamento deve ser muito bem realizada. Caso contrário, você poderá ter muita dor de cabeça no futuro, sem contar os prejuízos que poderão ocorrer.



Fabiano de Santana
www.fabianosantana.com.br
contato@fabianosantana.com.br
Mais de 12 anos de experiência na área de TI, onde atuou como Analista de Suporte/Administrador de Redes Microsoft e Lotus Notes/Domino. Atuou em projetos de implementação e administração de infraestrutura de redes Microsoft e segurança. Formado em Análise de Sistemas e atualmente cursando Pós-Graduação em Segurança da Informação. Aprovado em mais de 20 exames, obteve as certificações MCP, MCDST, MCSA 2000/2003 Security, MCSE 2000/2003 Security, IBM Certified Associate System Administrator Lotus Notes/Domino 7 e ITIL Foundation. Nomeado MVP em 2008, 2009, 2010 e 2011. Desenvolve e-books, livros e vídeo-aulas sobre produtos Microsoft. Atualmente são mais de 25 e-books publicados em cinco anos de trabalho. Trabalha atualmente com Team Leader na IBM Brasil.

Conheça o Windows Server 2008 R2
www.microsoft.com/brasil/servidores/windowsserver2008/r2/default.mspx

Página principal do Windows Server 2008 R2
www.microsoft.com/windowsserver2008/pt/br/default.aspx

Edições do Windows Server 2008 R2
http://www.microsoft.com/windowsserver2008/pt/br/r2-editions-overview.aspx

Virtual Labs
http://www.microsoft.com/windowsserver2008/pt/br/virtual-lab.aspx

Documentações do Sistema Operacional
http://www.microsoft.com/windowsserver2008/pt/br/product-documentation.aspx

Discos Virtuais/Imagens
http://www.microsoft.com/windowsserver2008/pt/br/virtual-hard-drive.aspx

Configurações básicas no Apache2

Instalação, configurações, desempenho e segurança em Sistemas Linux

Ricardo Franzen

Resumo DevMan

De que se trata o artigo:

Neste artigo será realizada a instalação do Apache de duas formas distintas. Uma delas é compilando através do código-fonte e a outra é através de pacotes próprios da distribuição. Também serão apresentadas configurações básicas de desempenho e segurança, recomendáveis em qualquer servidor Web Apache.

Para que serve:

As configurações sugeridas são aplicáveis em servidores de pequeno, médio e grande porte, com poucas ou milhares de requisições por minuto.

Em que situação o tema é útil:

Este artigo é ideal para aqueles que estão iniciando na área de infraestrutura e servidores, e até mesmo para os leitores com um pouco mais de experiência, que estão sempre em busca de novos conhecimentos, permitindo uma melhor configuração de servidores.

O Apache é um software multiplataforma e pode ser instalado nos principais sistemas operacionais disponíveis no mercado, como Linux, BSDs, Windows, entre outros menos populares. Por ser um software robusto, logo se pensa que exige um hardware muito poderoso para funcionar adequadamente. É aí que os inexperientes se enganam.

O Apache é um servidor web *open source* extremamente confiável e que tem suas funcionalidades expansíveis através de módulos que podem ser adicionados na medida em que se tornam necessários. Seu nome na verdade é Apache HTTP Server (veja seção **Links**), mas é popularmente conhecido como Apache.

O principal objetivo deste artigo é apresentar um conjunto de operações básicas no Apache, especialmente no que diz respeito à instalação, configuração, desempenho e segurança.

Servidor Web Apache

O Apache é hoje um dos mais respeitados softwares Open Source existentes no mercado, sendo um concorrente direto do IIS (veja seção Links), desenvolvido pela Microsoft.

Sua qualidade fica claramente estampada quando analisamos sua utilização a nível global na Internet. Segundo o site Netcraft (veja seção Links), em Janeiro de 2010, 53,84% dos sites existentes na internet rodavam sob o Apache, contra 24,08% do IIS.

Uma vantagem, que é maior que o dobro do segundo colocado, prova a excelência desta aplicação e a qualidade que um software *open source* pode obter quando possui um desenvolvimento maduro e sério.

O principal fator relacionado à performance é o número de instâncias do servidor httpd. O Apache suporta um grande número de instâncias simultâneas, onde cada uma é relacionada a um cliente acessando o servidor.

Na verdade, o gargalo acaba se localizando na velocidade do link e, dependendo das páginas servidas, no hardware. Códigos PHP mal escritos e scripts CGI podem elevar a carga do processador e, multiplicando pelo número de instâncias simultâneas fazendo essas requisições, tem-se um verdadeiro problema. Por este motivo é possível afirmar que um servidor bem configurado não é sinônimo de desempenho superior.

Como já informado, o Apache oferece suporte a módulos que, quando instalados, adicionam funcionalidades ao servidor. Como exemplos, pode-se citar o *mod_php*, *mod_python*, *mod_ftp*, entre outros. A variedade de módulos é muito grande e pode ser consultada diretamente no site do projeto (veja seção **Links**).

Serão apresentadas duas formas de instalação do Apache no Linux. A primeira delas é compilando o servidor a partir do código fonte, e a segunda é através de um gerenciador de pacotes. Optou-se por utilizar o Ubuntu (veja seção **Links**) por ser a distribuição Linux mais popular hoje em dia.



Compilando os fontes

Este artigo parte do princípio que o leitor está familiarizado com a compilação de algum software ou, ao menos detém as noções básicas, além de possuir os pacotes para realizar tal procedimento.

Em caso de dúvidas, no Ubuntu, é necessário instalar o pacote *build-essential* para que seja possível compilar os arquivos do Apache. Para isto, execute o comando:

```
# apt-get install build-essential
```

Quando este artigo foi escrito, a versão estável era a 2.2.16. No entanto, as configurações realizadas são válidas para qualquer versão do Apache2.

É possível baixar o arquivo pelo browser ou pelo terminal. Lembre-se, geralmente servidores não possuem interface gráfica instalada. Dessa forma, seguiremos a segunda opção.

Para realizar o download do pacote, execute o comando:

```
# wget http://linorg.usp.br/apache/httpd/httpd-2
.2.14.tar.gz
```

Agora é necessário efetuar a transformação em Super Usuário, através do comando:

```
# sudo su
```

Assim será possível executar todos os comandos sem nenhum tipo de problema. Porém, muito cuidado com os comandos que serão utilizados: como *root* nada impedirá você de fazer alterações indevidas no sistema.

Neste momento é necessário descompactar o arquivo baixado. Para isso, execute o comando abaixo para armazenar os arquivos no diretório */usr/src:*

```
# tar -zxvf httpd-2.2.14.tar.gz -C /usr/src
```

Observe que foi utilizado o parâmetro "-C" para especificar o destino dos arquivos que estão sendo extraídos do pacote.

Feito isso, deve-se acessar esse diretório:

```
# cd /usr/src/httpd-2.2.14/
```

Seguindo a proposta do artigo, esta instalação será básica, ou seja, não serão instalados módulos e/ou adicionadas funções extras ao Apache. Entretanto, caso tenha interesse de habilitar ou desabilitar algumas opções já na instalação, pode fazê-lo com o comando:

```
# ./configure --help
```

Agora, devem ser executados os três comandos básicos necessários para a compilação:

```
# ./configure
# make
# make install
```

Aguarde, dependendo do seu computador, isto pode levar alguns minutos. Concluído este processo, o servidor será instalado no diretório padrão que é */usr/local/apache2*.

Para levantar o serviço com as configurações padrão, execute o comando:

```
# /usr/local/apache2/bin/apachectl start
```

Após estes passos o Apache deve ter sido compilado com sucesso. Agora, ao acessar o endereço 127.0.0.1 com um navegador será exibida uma mensagem de confirmação, mostrando que o servidor está funcionando.

Instalando através do Gerenciador de Pacotes Synaptics

Este artigo foi escrito utilizando como base de testes a distribuição Linux Ubuntu, porém os mesmos passos podem ser aplicados a outras distribuições derivadas do Debian.

Para realizar a instalação é necessário acessar o menu *Sistema* > *Administração* > *Gerenciador de Pacotes Synaptic*, pesquisar por "Apache" e depois selecionar o pacote "Apache2". Em seguida é só aplicar e esperar o sistema realizar todo o trabalho.

Caso esteja utilizando o terminal, este processo pode ser feito com o comando:

```
# apt-get install apache2
```

A instalação via gerenciador de pacotes é visivelmente mais simples e torna todo gerenciamento mais fácil, visto que atualizações e *patches* de segurança são aplicadas e instaladas automaticamente. A compilação (apresentada no tópico anterior) só se faz necessária quando o administrador deseja customizar ao máximo o servidor.

Configurações Básicas

Depois da instalação padrão, o Apache já pode ser utilizado normalmente, eliminando a necessidade de inúmeras configurações adicionais. Para testá-lo, basta abrir o seu browser e acessar o endereço *http://localhost*. Neste momento você verá uma mensagem de confirmação do Apache.

O próximo passo será realizar algumas alterações no *httpd.conf* e demonstrar como habilitar o suporte a módulos, que estendem as possibilidades de utilização do servidor. Como exemplo, será instalado o módulo do PHP5, por este ser um dos recursos mais utilizados no Apache e pelo PHP ser uma das principais linguagens de programação voltadas à internet atualmente.

Quando o Apache é compilado, o arquivo de configuração é um só, o *httpd.conf*. Na instalação via gerenciador de pacotes no Ubuntu, esse arquivo é fragmentado para facilitar sua utilização. Todas as configurações citadas neste artigo podem ser realizadas em qualquer um dos ambientes, porém, a base será na configuração fragmentada que, de certa forma, acaba sendo um pouco mais complexa.

Os arquivos de configuração encontram-se no diretório */etc/apache2*.

ServerName

Nossa primeira alteração no *httpd.conf* será o parâmetro *ServerName*. Este parâmetro é utilizado quando o servidor Apache irá responder por um endereço do tipo *www.meusite.com.br*. Na instalação padrão, o *ServerName* não é especificado, ficando a critério do administrador utilizá-lo ou não. Este parâmetro pode ser inserido no início do arquivo de configuração *apache2.conf* da seguinte forma:

```
ServerName "www.meusite.com" (caso seja uma
URL) ou
ServerName "111.11.111.11" (caso seja um IP)
```

Virtual Hosts

Mas o que fazer se o servidor for responder por mais de um site ou possuir subdomínios? Neste caso deve-se editar

outro arquivo, responsável pelos *Virtual Hosts* dentro do Apache, que pode ser encontrado no diretório */etc/apache2/sites-available/default*.

Na configuração padrão, existe um Virtual Host configurado que é responsável pelo domínio padrão do seu servidor. Um novo Virtual Host pode ser configurado apenas copiando as configurações de um já existente e alterando o parâmetro *Document Root*, que indica o diretório onde ficarão armazenados os respectivos arquivos.

Existe também a possibilidade de criar diferentes arquivos ao invés de utilizar o arquivo *default*. Se isso for feito, para que o Apache reconheça este arquivo, precisa-se adicioná-lo à lista de sites habilitados com o comando:

```
# a2ensite nome-do-arquivo
```

Depois é necessário fazer com que o Apache recarregue as configurações e reconheça o novo arquivo com os novos Virtual Hosts. Isto pode ser feito através do comando:

```
# /etc/init.d/apache reload
```

Dentro de cada Virtual Host é possível configurar diversos "subdiretórios" entre os parâmetros *<Directory></Directory>*. Dessa forma pode-se criar configurações personalizadas para determinados diretórios ou até apontá-los para outros lugares.

Um exemplo de configuração de um Virtual Host no Apache pode ser visto na **Listagem 1**.

Na **Listagem 1** é possível observar a configuração para o domínio *www.meusite.com.br*. Nele foram adicionados alguns parâmetros para melhorar o acesso ao site, são eles:

• *ServerAlias:* Parâmetro que permite que o site responda em mais de um endereço. Configura-se esse parâmetro para que o site responda também sem o *www* na frente do endereço. Muitos administradores se esquecem de fazer isto e quando o usuário tenta acessar o site sem utilizar o *www* o Apache retorna um erro;

• *ExpiresActive:* Esta funcionalidade do Apache é um recurso muito interessante e pouco utilizado. Com ele, pode-se determinar o tempo que o browser do cliente fará cache de determinados tipos de arquivo, reduzindo assim a quantidade de requisições feitas ao servidor, sem falar na velocidade de carregamento do site, que será visivelmente incrementada. Pode não parecer, mas pequenas configurações como esta muitas vezes não são feitas pelos administradores e acabam gerando um elevado número de requisições repetidas e desnecessárias ao servidor Web.

Hardening

Segundo a Wikipédia, *hardening* é um processo de mapeamento das ameaças, mitigação dos riscos e execução das atividades corretivas com foco na infraestrutura com o objetivo principal de torná-la preparada para enfrentar tentativas de ataque.

Se o servidor vai estar aberto na Internet, tem-se obrigação de mantê-lo seguro. Sem entrar em detalhes de configuração de *firewall* e de sistema, que seria assunto para outro artigo, deve-se atentar para alguns pequenos detalhes e configurações que podem fazer a diferença no servidor.

Por padrão, em diretórios que não possuem nenhum arquivo *index*, o Apache lista todos os arquivos existentes nos mesmos. Isto em alguns casos não é de interesse do administrador, seja porque o diretório contém arquivos privados, ou apenas para não permitir que os arquivos que ali se encontram sejam vistos. Esta configuração pode ser realizada em diretórios específicos mas, em algumas situações, a intenção pode ser a de bloquear a listagem de arquivos em todos os diretórios. Para isso, é necessário editar o arquivo */etc/apache2/sites-available/default* e alterar os seguintes parâmetros:

```
Options FollowSymLinks
AllowOverride None
```

para:

```
Oder Deny,Allow
Deny from all
Options None
AllowOverride None
```

Feitas estas alterações, o Apache deve ser reiniciado com o seguinte comando:

```
# /etc/init.d/apache2 reload
```

Algumas alterações relativas à segurança ainda podem ser feitas no arquivo *security*, localizado no diretório */etc/apache2/conf.d/security*.

Não é interessante que alguma pessoa mal intencionada consiga informações sobre o servidor facilmente. Para evitar que isso aconteça, alguns parâmetros podem ser alterados para restringir o nível de informação que é passado para os clientes. São eles:

• **ServerTokens OS:** Esta diretiva é responsável pelo nível de informação sobre o servidor que será retornado no cabeçalho HTTP. Os parâmetros existentes são: Full, OS, Minimal, Minor, Major e Prod. Onde Full retorna muitas informações como o Sistema Operacional e os módulos instalados e Prod retorna o mínimo de informações;

• **ServerSignature On:** Adiciona em páginas geradas pelo servidor (páginas de erro, listagem de arquivos, etc.) uma linha contendo a versão do servidor e o nome do Virtual Host;

• **TraceEnable On:** Este parâmetro é apenas utilizado para testes e diagnóstico do servidor.

Listagem 1. Exemplo de Configuração de um Virtual Host.

```
<VirtualHost *:80>
    ServerAdmin seu-email@meusite.com.br

    ServerName www.meusite.com.br
    ServerAlias meusite.com.br www.meusite.com.br
    DocumentRoot /var/www/meusite.com.br

    ExpiresActive On
    <FilesMatch "\.(gif|jpg|jpeg|png|css|js|swf|GIF|JPG|
     JPEG|PNG|txt|TXT)$">
        ExpiresDefault "access plus 1 year"
    </FilesMatch>

    <Directory /var/www/meusite.com.br>
        Options -FollowSymLinks +SymLinksIfOwnerMatch
        AllowOverride None
        Deny from all
    </Directory>

    CustomLog /var/log/apache2/meusite.com.br.log
    combined
</VirtualHost>
```

Altere estes três parâmetros para:

```
ServerTokens Prod
ServerSignature Off
TraceEnable Off
```

Como de costume, sempre que alguma configuração é alterada o Apache precisa ser reiniciado para carregar os novos parâmetros.

Veja na vídeo aula deste artigo algumas dicas sobre como aumentar a segurança do seu servidor web.

Utilizando Módulos

Na sua instalação padrão, o Apache traz habilitado muitos módulos que, algumas vezes, não são utilizados. Os módulos disponíveis podem ser vistos em */etc/apache2/mods-available*. Já os módulos que estão habilitados ficam em */etc/apache2/mods-enabled*.

Como esta questão de módulos a serem utilizados é muito particular e varia em cada caso, será explicado apenas como habilitar e desabilitar os módulos no Apache.

Para habilitar um módulo, deve-se executar o comando:

```
# a2enmod modulo
```

Por exemplo:

```
# a2enmod php5
```

Agora, para desabilitar um módulo, pode-se executar:

```
# a2dismod modulo
```

Para listar todos os módulos do Apache que estão disponíveis nos repositórios do Ubuntu, execute:

```
# apt-cache search apache2-mod
```



Nota do DevMan

Safe Mode: Safe Mode, ou Modo Seguro, é uma tentativa de resolver um problema de segurança em servidores compartilhados. Com ele, o PHP faz verificações quando trabalha com manipulação de arquivos, verificando a propriedade do script atual e do arquivo a ser alterado. A partir da versão 5.3.0 do PHP este recurso se tornou obsoleto e seu uso desencorajado no próprio manual, na versão 6.0 será completamente removido.

Habilitando o suporte ao PHP

A instalação padrão do Apache não vem com suporte ao PHP ou a qualquer outra linguagem interpretada. Para habilitar o suporte ao PHP, é necessário realizar a instalação do pacote *libapache2-mod_php5* para depois configurá-lo no Apache.

O primeiro passo para isso será abrir um console e digitar o comando:

```
# sudo su
```

Como já informado, o código acima faz com que todos os comandos sejam realizados com os privilégios do usuário *root*. Com isso é possível chamar o gerenciador de pacotes e instalar o módulo do PHP:

```
# apt-get install libapache2-mod-php5
```

Para verificar se o módulo foi realmente instalado, é necessário acessar o diretório */etc/apache2/mods-available*. Nele, devem ser listados, entre outros, dois arquivos: *php5.conf* e *php5.load*.

Agora que o módulo PHP está instalado no sistema, é possível habilitá-lo no servidor Apache. Para isso, o seguinte comando deve ser executado:

```
# a2enmod php5
```

Depois de habilitado, os dois arquivos anteriormente descritos (*php5.conf* e *php5.load*) receberam links simbólicos no diretório */etc/apache2/mods-enabled*. Isto significa que o módulo do PHP5 será carregado na próxima inicialização do Apache, que precisa ser reiniciado com o comando:

```
# /etc/init.d/apache2 restart
```

Dependendo dos serviços que serão disponibilizados no seu servidor, pode ser necessário alterar algum parâmetro de funcionamento do PHP. Estas alterações são feitas no arquivo *php.ini* que fica localizado no diretório */etc/php5/apache2*.

Para exemplificar, vejamos algumas configurações importantes que podem ser realizadas no *php.ini*.

Desabilitar ou habilitar o *safe mode*:

```
safe_mode = Off
```

Para determinar a memória que um script PHP pode alocar, utiliza-se o comando *memory_limit*, prevenindo que algum *script* mal escrito ou mal intencionado consuma toda a memória do servidor:

```
memory_limit = 16M
```

Caso o servidor não seja de testes, mas sim de produção, talvez seja interessante desabilitar a exibição de erros e alertas nas páginas geradas:

```
display_errors = Off
```

Ao fazer isto, pode-se querer registrar estes erros em algum lugar e, para isso, é importante habilitar o *log* de erros:

```
log_errors = On
```

Por medida de segurança, também é interessante desabilitar variáveis globais. Isso evita que alguma pessoa mal intencionada, em alguns casos, consiga alterar valores de variáveis, fazendo com que o sistema se comporte de forma anormal:

```
register_globals = Off
```

Mais informações sobre o parâmetro *register_globals* podem ser encontradas no endereço (veja seção **Links**).

Em sua configuração padrão, o PHP vem com o suporte a *upload* de arquivos habilitado. Se esta função não for utilizada, pode-se desabilitá-la e evitar um possível problema gerado por alguma pessoa mal intencionada que consiga se aproveitar desse serviço. Para esse ajuste, altere a configuração para:

```
file_uploads = Off
```

Mas, caso seja necessário receber arquivos, e mais, arquivos grandes, é necessário alterar o tamanho máximo dos arquivos que podem ser recebidos, por exemplo:

```
upload_max_filesize = 5M
```

Caso a aplicação utilize sessões, pode ser interessante aumentar o tamanho máximo do tempo de vida de uma sessão que, por padrão, é de 24 minutos. Esta opção é recomendada para sistemas que possuem

uma utilização frequente e contínua, evitando assim que o usuário tenha que se autenticar a todo o momento:

```
session.gc.maxlifetime = 1440 (em segundos)
```

Enfim, o *php.ini* é bastante extenso, sendo recomendada a leitura do arquivo. Os comandos e os comentários que se encontram nele podem ser bastante úteis, além de ser uma excelente forma de estudo das possibilidades que as configurações do PHP oferecem.

Companheiros para o Apache

Claro que o Apache não é único no mundo dos servidores Web. Existem outros softwares que seguem a mesma proposta. Um bom exemplo é o Lighttpd, que pode ser utilizado individualmente ou em conjunto com o Apache.

O Lighttpd (veja seção **Links**) é um servidor desenvolvido para ambientes de alta performance. Seu consumo de memória é muito baixo, além de possuir um bom gerenciamento de carga. Entretanto, seu desempenho não é muito bom quando se trata de páginas dinâmicas, como páginas em PHP. Já quando utilizado para prover conteúdo estático, como arquivos .html, .css, .js e imagens, seu desempenho é consideravelmente superior ao Apache. Uma boa combinação é utilizar o Apache para prover conteúdo dinâmico e o Lighttpd para conteúdo estático.

Outro que não é exatamente um Web Server, mas que merece ser lembrado, é o novíssimo HipHop (veja seção **Links**), desenvolvido pelo Facebook. Ele traduz os códigos PHP para C++ e, depois disso, o compila, aumentando assim a velocidade na execução.

Conclusão

Uma boa forma de aprender é através da tentativa e erro, pois assim aprende-se o que pode e o que não pode ser feito. Por isso, não se deve ter medo de arriscar e alterar algum parâmetro de configuração. Caso erre em algum ponto, sempre haverá a possibilidade de voltar atrás e recuperar o estado em que tudo estava funcionando corretamente. No entanto, é importante informar que os testes devem ser realizados em ambientes específicos para tal, nunca em servidores de produção, onde um erro de configuração pode gerar um problema mais sério.



Ricardo Franzen
rfranzen@gmail.com
Estudante do curso de Sistemas de Informação, atua na área de TI há 7 anos e com infraestrutura há 4 anos. Atualmente trabalha como Sys Admin em uma administradora de hotéis com atuação em todo território Nacional.

Apache HTTP Server:
http://httpd.apache.org

Módulos Apache:
modules.apache.org

IIS:
www.iis.net

Netcraft:
www.netcraft.com

Ubuntu:
www.ubuntu.com

Parâmetro register_globals do PHP:
www.php.net/manual/pt_BR/security.globals.php

Lighttpd:
www.lighttpd.net

HipHop:
http://developers.facebook.com/news.php?story=358&blog=1

Securing Apache: Step-by-Step:
http://www.symantec.com/connect/articles/securing-apache-step-step

Otimizando seu web server com Apache2 + Lighttpd:
http://www.vivaolinux.com.br/artigo/Otimizando-seu-web-server-com-Apache2-+-Lighttpd

Documentação do Apache em Português do Brasil:
http://httpd.apache.org/docs/2.2/pt-br/

Tutorial do Nikto, um scanner de vulnerabilidades para WebServers:
http://blog.alexos.com.br/?p=297&lang=pt-br

Framework para auditoria de vulnerabilidades em servidores Web:
http://w3af.sourceforge.net/

Hardening

Blindando um Sistema GNU/Linux

Flávio dos Reis, Marcos Verbena e Eduardo Pagani Julio

Devido ao crescente número de ameaças existentes na Internet e dentro dos ambientes corporativos, se faz necessária a utilização de técnicas capazes de proporcionar maior segurança, estabilidade e tranquilidade para os administradores de redes. Pensando nisso, este artigo tem como objetivo apresentar o conceito e técnicas disponíveis para a implementação de *hardening* em servidores GNU/Linux.

Hardening, ou blindagem de sistemas, consiste na utilização de técnicas para prover mais segurança a servidores que disponibilizam serviços externos, como servidores Web, ou até mesmo serviços internos, como servidores de banco de dados, de arquivos, entre outros.

Neste artigo são analisadas e discutidas técnicas para a aplicação de *hardening* desde a instalação do sistema operacional, o processo de particionamento de discos, análise de serviços desnecessários e inseguros, localização de senhas fracas, verificação de usuários inválidos, desconexão de usuários não autorizados, implementação de políticas de utilização de serviços de rede, gerenciamento de privilégios e aplicação de segurança no terminal.

Para a realização da parte prática do artigo, foram utilizados três computadores com as seguintes configurações: Cliente 1: Ubuntu 9.10; Cliente 2: Ubuntu 9.10; Servidor: Debian Lenny.

Para o servidor, o sistema operacional escolhido foi o GNU/Debian Lenny, por ser um Linux reconhecidamente pela comunidade como estável e robusto, voltado para servidores. As técnicas aqui demonstradas foram implementadas no servidor e as máquinas clientes foram utilizadas apenas para a realização de acessos e monitoramento, conforme a **Figura 1**.

Resumo DevMan

De que se trata o artigo:

Este artigo tem como objetivo demonstrar de forma prática algumas técnicas de blindagem de sistema também conhecidas como Hardening. Esta técnica consiste na implementação de diretivas de segurança que devem ser seguidas antes, durante e após a instalação e configuração de servidores GNU/Linux.

Para que serve:

A técnica de Hardening pode ser utilizada em qualquer sistema operacional. Com o grande aumento no número de ameaças existentes na Internet é fundamental que o sistema de um servidor esteja preparado para superar todas as tentativas de invasão. Esta técnica não deve ser implementada somente em servidores que ficam conectados diretamente a Internet, muitas vezes fornecendo serviços como, por exemplo servidores web, mas também em máquinas que provêm serviços internos de rede como servidores de arquivos e de impressão. Com a blindagem de sistemas é possível aumentar o desempenho do hardware, liberando recursos que estão sendo utilizados por aplicativos desnecessários, implementando configurações específicas em alguns serviços, além de gerar um ambiente mais seguro.

Em que situação o tema é útil:

Hardening pode ser utilizado para evitar que usuários mal intencionados aproveitem da ausência do administrador e implantem scripts maliciosos em servidores infectando toda a rede, bloquear que o usuário administrador faça login diretamente no terminal, efetuar logout por tempo de inatividade, remover pacotes que não são utilizados, remover permissões especiais de binários executáveis, dentre outras técnicas que serão apresentadas posteriormente.

Hardening

O *hardening* consiste na realização de alguns ajustes finos para o fortalecimento da segurança de um sistema. Muitos administradores sem experiência em segurança preparam seus servidores com uma instalação básica e depois que suas aplicações estão disponíveis nenhum procedimento é feito para manter a integridade do sistema.

Em um sistema GNU/Linux é possível atingir um alto nível de segurança implementando configurações que permitam o aperfeiçoamento da segurança aplicada ao sistema. Quando se deseja aplicar a técnica de *hardening* há três grandezas que devem ser consideradas: segurança, risco e flexibilidade.



Ponto de Montagem	nosuid	noexec	noatime
/boot	X	-	-
/	-	-	-
/home	X	x	-
/usr	X	-	-
/tmp	X	x	-
/var	X	x	-
/var/log	X	x	x

Tabela 1. Opções da tabela de partições



Nota do DevMan

O administrador deverá ter atenção ao implementar as técnicas. Caso não tenha experiência nos exemplos aqui demonstrados, faça os testes em máquinas virtuais a fim de entender e evitar possíveis falhas na implementação



Nota do DevMan

nosuid é o parâmetro usado para inibir a execução de binários com permissão de suid bit; noexec é o parâmetro usado para inibir a execução de um binário na partição; e noatime é o parâmetro responsável por eliminar a necessidade de escrita no disco para arquivos que precisam ser somente lidos.



Nota do DevMan

Backdoor: são programas que instalam um ambiente de serviço em um computador, tornando-o acessível à distância, permitindo o controle remoto da máquina sem que o usuário saiba, como uma porta dos fundos não autorizada.

Listagem 1. Listando pacotes instalados no Debian.

```
# dpkg -l | awk '{print $2,$3}' | sed '1,5d' >/root/
  auditoria/pacotes.txt
```

Listagem 2. Listando pacotes instalados no Red Hat.

```
#rpm –qa > /root/auditoria/pacotes.txt
```

O administrador de redes deve analisar muito bem essas grandezas e encontrar um estado de harmonia entre elas, levando o sistema a uma alta produtividade e segurança, pois quanto maior a segurança menor o risco e também a flexibilidade.

É importante ressaltar que as técnicas aqui apresentadas podem não ser adequadas para todas as situações. Por isso, antes de implantar efetivamente as técnicas de *hardening*, é fundamental que haja um estudo completo do cenário e serviços em questão. Inicialmente recomenda-se sempre instalar versões atuais dos sistemas operacionais, que contenham correções e *patches* de segurança, pois pode ser problemático utilizar uma versão antiga sem atualizações, deixando o sistema temporariamente vulnerável no caso da existência de pacotes com falhas. Serviços críticos como web, email e DNS devem estar sempre nas versões mais atuais. Softwares desnecessários devem ser desinstalados e pacotes inseguros devem ser substituídos por alternativas mais confiáveis.

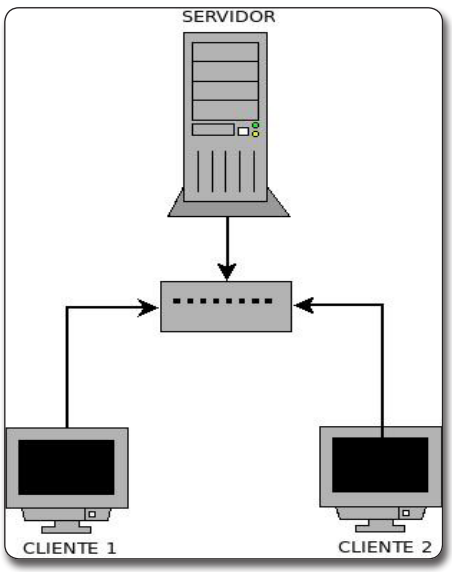


Figura 1. Estrutura utilizada para o artigo

A seguir são apresentadas algumas utilizações de *hardening* muito úteis na configuração de sistemas operacionais.

Particionamento de discos

O particionamento de discos é um ponto importante quando se pensa em segurança. Ao particionar o disco, é inserida no sistema uma maior segurança, pois cada partição tem sua tabela de alocação de arquivos separada. Um exemplo de como pode ser configurada a tabela de partições é apresentado na **Tabela 1.**

O comando *mount* (comando UNIX usado para montar partições) permite utilizar algumas opções para aumentar a segurança nas partições. Crackers podem aproveitar do diretório */tmp*, onde por padrão, qualquer usuário pode gravar dados no sistema, para introduzir um *backdoor* ou qualquer outro programa malicioso para ter um acesso completo ao sistema.

Serviços desnecessários e inseguros

Depois do sistema instalado, deve ser realizada uma verificação minuciosa de todos os programas instalados e se são realmente necessários, mesmo sendo uma instalação básica. Um servidor nunca deve conter programas "clientes". Serviços como telnet, rshd, rlogind, rwhod, ftpd, sendmail, identd, wget, dentre outros, deverão ser removidos. Estes serviços podem ser desinstalados usando o gerenciador de pacotes do sistema operacional, ou desativando-os em todos os níveis de inicialização. Além disso, podem-se remover entradas específicas dos programas no *boot* do sistema operacional.

Neste exemplo são apresentados os passos necessários para a remoção de pacotes que não são utilizados no sistema. A remoção de pacotes obsoletos deverá ser executada, evitando assim que vulnerabilidades sejam exploradas. Para seguir o exemplo, crie um diretório em */root* chamado auditoria, onde será gerado um arquivo com todos os pacotes que estão instalados, podendo assim analisar quais serão removidos.

O comando *dpkg -l*, utilizado em distribuições GNU/Linux Debian e derivados, faz uma pesquisa no sistema e lista todos os pacotes instalados. Um filtro com o comando *awk* é utilizado para formatar a saída do comando, mostrando assim somente a segunda e terceira colunas, e o comando *sed*, nesse exemplo, retira as cinco primeiras linhas. Esse resultado é gravado

em um arquivo texto chamado *pacotes.txt*, como pode ser visto na **Listagem 1**.

Em distribuições Red Hat e derivados utiliza-se o comando *rpm –qa*, fazendo assim uma pesquisa em todos os pacotes instalados, como pode ser visto na **Listagem 2.**

A análise desse arquivo pode ser um pouco demorada, ainda mais se o administrador estiver analisando a saída gerada por um servidor Red Hat, uma vez que essa distribuição traz um número maior de pacotes instalados que um GNU/LINUX Debian. Um pacote interessante para remover é o *wget*. Com esse comando, um *cracker* pode fazer com que o servidor alvo execute *downloads* de arquivos, através de um servidor Web forjado, por exemplo. Assim, o *cracker* pode jogar qualquer *script* que possa danificar o sistema ou até mesmo abrir uma porta para novas invasões.

O comando *aptitude* remove o aplicativo *wget*, e a opção *purge* faz com que os arquivos de configuração do aplicativo sejam removidos, caso existam, não deixando vestígios de sua instalação, como pode ser visto na **Listagem 3**.

Outro exemplo é o pacote *wireless-tools*. Caso o servidor em questão não tenha nenhuma placa de rede sem fio, não há necessidade de tal aplicativo.

Procura de senhas fracas

A senha deve ser única, intransferível e de propriedade de um único usuário. O administrador não deve saber essas senhas. Um procedimento muito comum para isso consiste na alteração da senha no momento do primeiro login do usuário. Dessa forma o administrador não terá acesso à senha escolhida pelo usuário. Mas, na maioria das vezes, as senhas escolhidas são fáceis de serem descobertas, pois são usadas sequências simples, como 123456, ou informações pessoais como datas de aniversário, nome próprio, entre outras.

Nestes casos, é possível utilizar ferramentas capazes de avaliar se a senha do usuário é fraca ou não. O utilitário *John the Ripper* pode ser utilizado com a finalidade de encontrar senhas fracas escolhidas por usuários de um sistema. Encontrando senhas fracas, o administrador poderá solicitar que o usuário efetue a substituição da senha por outra que atenda a política de segurança, tornando assim o sistema mais robusto e tolerante à exploração de vulnerabilidades.

```
debian:~# time john /etc/shadow
Loaded 3 password hashes with 3 different salts (FreeBSD MD5 [32/32])
123456           (root)
123456           (eduardo)
123456           (flavio)
guesses: 3  time: 0:00:00:01 100% (2)  c/s: 5574  trying: 123456

real    0m1.479s
user    0m1.412s
sys     0m0.052s
debian:~# _
```

Figura 2. Execução do John the Ripper procurando por senhas fracas

Se uma auditoria está sendo executada em um servidor GNU/Linux de uma empresa, será necessário descobrir se os usuários estão usando senhas fracas. No que diz respeito às senhas, a norma ISO 27002 diz nos itens 11.2.3 e 11.3.1 que devem ser controladas por meio de um processo de gerenciamento formal e que os usuários sejam solicitados a seguir boas práticas de segurança da informação na seleção e uso de senhas.

No exemplo, o aplicativo *John the Ripper* é utilizado para teste de senhas fracas. Ele é uma ferramenta de *Brute Force* (força bruta, que testa combinações de senha para encontrar a resposta), que pode ajudar a descobrir senhas fracas de usuários, checando diretamente o arquivo */etc/shadow* dos servidores. O aplicativo pode ser instalado com o *aptitude*, como pode ser visto na **Listagem** 4.

O *John the Ripper* é executado passando como parâmetro o arquivo */etc/shadow*. O comando *time* foi acrescentado para que possa contar o tempo que o aplicativo levou para descobrir a senha dos usuários e também do *root*, conforme mostra a **Figura 2.**

Regras para definir senhas fortes devem ser levadas em consideração como, por exemplo, estipular um número mínimo de caracteres (por exemplo, 10 caracteres), utilizar letras minúsculas, maiúsculas, números e caracteres especiais. Faça com que as senhas sejam alteradas em intervalos de tempo curtos, que podem ser definidos pelo administrador. Um valor

Listagem 3. Removendo aplicativos.

```
# aptitude purge wget
```

Listagem 4. Instalando o John the Ripper.

```
#aptitude install john
```

que pode ser adotado é a cada 20 dias, por exemplo. Defina a quantidade de senhas já utilizadas que não poderão ser reaproveitadas. Deve-se também educar os usuários para que não divulguem suas senhas para terceiros.

Tomando essas providências simples, pode-se impedir que algum atacante consiga quebrar a senha utilizando força bruta. Mesmo que a senha seja quebrada em algum momento, com a troca de senhas periódica, esta poderá já não ser mais válida.

O PAM (*Pluggable Authentication Module*) é um conjunto de bibliotecas compartilhadas que permitem ao administrador do sistema local definir como determinadas aplicações autenticam os usuários, sem a necessidade de modificar e recompilar programas. O PAM é um recurso que auxilia muito quando se pensa em segurança. Existem vários módulos que podem ser implementados, aumentando assim o controle de criação e troca de senhas. A **Figura 3** mostra que o usuário não conseguiu mudar a senha na primeira tentativa, recebendo a mensagem "Escolha uma senha mais longa". Isso ocorre devido a uma alteração no módulo do PAM. A opção min=10 foi adicionada, limitando assim um tamanho mínimo para a senha escolhida, como pode ser visto na **Listagem 5**.

Usuários Inválidos

Nos Sistemas GNU/Linux há três tipos de usuários, usuário *root*, que é o admi-

nistrador do sistema, usuários comuns, os quais possuem uma senha para logar no sistema e acesso a um diretório *home* onde os mesmos poderão ter privacidade com seus arquivos pessoais e, por último, os usuários de sistema, responsáveis por controlar requisições de serviços. O *shell* é a interface entre usuário e sistema. Sem um *shell* válido, não é possível digitar comandos e interagir com o sistema. O usuário de sistema *www-data*, responsável por receber requisições do servidor Web, que esteja com um *shell* válido, poderá introduzir vulnerabilidades ao seu sistema.

Desconexão de usuários não autorizados

Inicialmente é importante ter o conhecimento de que usuários não autorizados podem estar dentro ou fora da empresa.

O acesso não autorizado por sistemas externos deve ser cancelado com extrema urgência, em especial se o usuário estiver ocultando sua identidade. Para os casos de acesso por usuários internos não autorizados, podem ser necessárias ações disciplinares dependendo da natureza do acesso.

É importante salientar que, quando ocorre uma invasão de sistema, é fundamental que as evidências relacionadas ao acesso indevido sejam registradas antes da desativação da conta não autorizada, tendo cuidado para não destruir provas relacionadas ao crime.

Colocar senha criptografada no GRUB

Muitos administradores não estão preparados para lidar com estruturas críticas. O simples acesso físico de um usuário a sala de servidores pode representar uma violação de segurança grave, pois este poderá conseguir acesso de *root* se reiniciar o servidor e alterar a senha do *root* através do gerenciador de *boot* (*grub*). Esse processo poderá ser evitado se uma senha criptografada for adicionada ao gerenciador, não permitindo que um usuário qualquer inicie o sistema no modo de segurança, por exemplo.

Política de utilização de serviços de Rede

TCP wrappers oferecem controle de acesso a vários serviços. A maioria dos serviços de rede modernos, como SSH, Telnet e FTP, utilizam os *TCP wrappers* que ficam monitorando a entrada de um pedido e o serviço requisitado.

O uso do *TCP wrappers* é uma boa prática na implementação de segurança em redes, limitando o uso dos serviços de rede. Liberar acesso somente a IPs desejados, configurar as restrições do *ssh* não permitindo *login* como *root* e configurar os módulos do *pam* para restringir acesso ao servidor em determinado horário, são boas práticas que devem ser adotadas.

Gerenciamento de privilégios

O usuário *root* é o mais visado por *crackers* ou usuários mal intencionados. Seu foco é conseguir a senha *root* e obter acesso total ao sistema.

Para dificultar a ação destas ameaças, desativar o *login* como usuário *root* nos terminais modo texto torna-se fundamental. Dessa forma, o administrador deverá efetuar o *login* como usuário comum e quando for necessário executar uma tarefa administrativa tornar-se *root* com o comando *su*. Determinar a data de validade para a senha dos usuários e, com auxílio do comando *usermod*, remover *shells* válidos de usuários que não estão em uso também são ações importantes a serem tomadas para garantir a robustez do sistema.

Segurança no Terminal

Quando é citado o assunto segurança, logo se imagina a exploração de uma vulnerabilidade por uma ameaça remota e, na maioria das vezes, as ameaças internas são esquecidas ou até mesmo subestimadas. Usuários internos mal intencionados podem causar grandes prejuízos e, se o usuário em questão tiver acesso físico aos servidores, a situação fica ainda mais grave.

Um usuário mal intencionado com acesso físico a sala de servidores pode usufruir de uma estação logada como usuário *root* e assim danificar o sistema. A variável TMOUT tem a função de executar um *logout* automático após determinado tempo de inatividade do terminal. Seu valor pode ser configurado no arquivo */etc/profile*. O valor a ser adicionado deverá ser analisado com cuidado, evitando assim acessos indevidos. Valores muito altos podem dar espaço para que o usuário tenha acesso à estação logado como root, valores muito baixos podem interferir em tarefas onde é necessário uma pesquisa, por exemplo. O arquivo */etc/profile* será editado e adicionado a variável com valor de 60 segundos, como pode ser visto na **Listagem 6**.

O arquivo */etc/profile* só é lido durante o boot do sistema, dessa forma utiliza-se o comando *source* para que o arquivo possa ser lido novamente atribuindo ao sistema as alterações aplicadas, como pode ser visto na **Listagem 7**.

Após 60 segundos de inatividade, o *shell* fará *logout* automático, conforme mostra a **Figura 4.**

Outro ponto importante a ser levado em consideração quando se pensa em controle de acesso em uma organização, é avaliar quem tem acesso aos servidores. Usuários

Listagem 5. Limitando o tamanho mínimo de uma senha.

```
password required pam_unix.so nullok obscure
min=10 md5
```

Listagem 6. Inserindo valores no /etc/profile.

```
#vim /etc/profile

TMOUT=60
export PATH TMOUT
```

Listagem 7. Comando source.

```
#source /etc/profile
```

```
flavio@debian:~$ passwd
Mudando senha para flavio.
Senha UNIX (atual):
Digite a nova senha UNIX:
Redigite a nova senha UNIX:
Escolha uma senha mais longa
Digite a nova senha UNIX:
Redigite a nova senha UNIX:
passwd: senha atualizada com sucesso
flavio@debian:~$ _
```

Figura 3. Alterando as senha dos usuários

```
root@debian:/# timed out waiting for input: auto-logout
flavio@debian:~$ _
```

Figura 4. Terminal efetuando logout automático

mal intencionados podem simplesmente usar o CTRL+ALT+DEL para reiniciar o servidor, parando assim todos os serviços disponíveis em uma rede. Isso pode ocorrer em empresas que não têm uma política de acesso aos seus servidores. Segundo a norma ISO 27002, devem-se tratar as questões de acesso físico à sala de servidores. No entanto, independentemente desta norma, pode-se inibir a função CTRL+ALT+DEL editando o arquivo */etc/inittab*, conforme a **Listagem 8.**

SSH

O SSH (*Secure Shell*) é um programa usado para acessar remotamente outro computador usando uma rede, executar comandos em uma máquina remota e copiar arquivos de um computador para outro. Fornece autenticação forte e comunicação segura sobre canais inseguros (como a Internet, por exemplo). É muito usado para logar em um sistema GNU/Linux através de uma máquina Windows, Mac ou mesmo outro GNU/Linux, na qual os tradicionais *telnet* e *rlogin* não podem fornecer criptografia da senha e da sessão. Detalhes importantes precisam ser levados em consideração quando um servidor *ssh* é configurado. Para isso, algumas configurações importantes devem ser feitas no arquivo de configuração, conforme apresenta a **Listagem 9**.

Após efetuar as alterações, o serviço *ssh* precisa ser reiniciado (ver **Listagem 10**).

Portas Abertas

Quando o sistema novo é instalado, alguns aplicativos (serviços) podem abrir portas introduzindo assim vulnerabilidades no sistema. Com o aplicativo *nmap*, pode-se fazer uma busca por todas as portas abertas no sistema e, em seguida, podem ser criadas regras no *firewall* para bloquear as que não devem estar disponíveis. Uma listagem de portas feita utilizando o aplicativo *nmap* pode ser visto na **Listagem 11.**

Analisando o resultado do *nmap* pode-se observar que a porta 80 está aberta, normalmente usada por um servidor Web. Se esse serviço não for utilizado, a porta deverá ser fechada, evitando que vulnerabilidades sejam exploradas. Uma regra do *firewall iptables* poderá ser utilizada para bloquear tentativas de acesso a essa porta, como exibe a **Listagem 12**.

Após adicionar a regra no *firewall*, faça novamente a pesquisa com o *nmap* e analise o resultado. Observe que a porta 80 está sendo filtrada, conforme a **Listagem 13**.

SUID BIT

O SUID BIT é uma das permissões especiais disponíveis no GNU/Linux. Quando está atribuída a um binário, é possível que um usuário execute o mesmo com os privilégios de seu dono. Se o dono do binário for o usuário *root*, o usuário vai executar o binário como *root*.

Podemos encontrar problemas de segurança ao ter binários com essa permissão especial configurada. Segundo a norma ISO 27002, no item 11.6.1, o acesso à informação e às funções dos sistemas de aplicações por usuário e pessoal do suporte devem ser restritas, de acordo com o definido na política de controle de acesso. Esse ponto deve estar muito bem determinado e esclarecido para todos os funcionários da organização. Pois caso algo seja violado, existe um documento para comprovar que aquilo não é certo.

Através de um *script* é possível localizar e alterar essa permissão, como apresenta a **Listagem 14**.

Depois de criado o script, é necessário alterar suas permissões, deixando que apenas o usuário *root* possa executá-lo, e retirando qualquer permissão dos demais usuários e grupos. Para essa função é usado o comando *chmod* (**Listagem 15**).

O primeiro passo é executar o script, e como parâmetro deverá ser utilizado "n" ou "N". Nesse modo ele apenas irá gerar uma lista de binários que possuem a permissão de SUID-BIT no sistema, de acordo com a **Listagem 16.**

Observe no diretório */root/auditoria* que o arquivo *list.suid* foi criado (**Listagem 17**).

Como se trata de um sistema recém-instalado, o administrador não deve

Listagem 8. Editando o arquivo /etc/ssh/inittab.

```
#vim /etc/inittab

# Antes
# What to do when CTRL-ALT-DEL is pressed.
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now

# Depois
# What to do when CTRL-ALT-DEL is pressed.
#ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now

#ou

# Depois
# What to do when CTRL-ALT-DEL is pressed.
ca:12345:ctrlaltdel:/bin/echo "Opção desativada !"
```

Listagem 9. Editando o arquivo /etc/ssh/sshd_config.

```
#vim /etc/ssh/sshd_config

# Altere a porta padrão
Port 42129
# Protocolo 2 (anteriores possuem falhas de segurança)
Protocol 2
# Tempo ativado para digitar a senha
LoginGraceTime 45
# Não aceitar login como root
PermitRootLogin no
# Não aceitar login sem senha
PermitEmptyPasswords no
# Usar o modulo do pam para se autenticar
UsePAM yes
# Definir usuários que tem permissão de fazer login
AllowUsers "flavio marcos eduardo"
```

Listagem 10. Reiniciando o serviço ssh

```
$invoke-rc.d ssh restart
```

Listagem 11. Listagem de portas utilizando nmap.

```
#nmap -A -p 1-65535 localhost
PORT    STATE SERVICE VERSION

22/tcp    open    ssh        protocol 2.0
25/tcp    open    smtp       Postfix smtpd
53/tcp    open    domain     dnsmasq 2.47
80/tcp    open    http       httpd 2.2.11
631/tcp   open    ipp        CUPS 1.3.9
3306/tcp  open    mysql      MySQL 5.0.75-0
```

Listagem 12. Bloqueando a porta 80 com iptables.

```
#iptables –A INPUT –i eth0 –p tcp –dport 80 –j DROP
```

Listagem 13. Conferindo portas com nmap.

```
#nmap -A -p 80 localhost
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     protocol 2.0
25/tcp  open  smtp    Postfix smtpd
53/tcp  open  domain  dnsmasq 2.47
80/tcp  filtered  http
631/tcp open  ipp     CUPS 1.3.9
3306/tcp open  mysql  MySQL 5.0.75-0
```

encontrar nenhum problema. Mas é necessária a atenção do mesmo, pois se estiver analisando um servidor de uma empresa, onde este se encontra em produção, devem-se remover as permissões *suid*.

Agora execute o *script* utilizando o parâmetro "s" ou "S", removendo assim a permissão SUID BIT de todos os binários, como visto na **Listagem 18**. Deve-se observar quais binários serão necessários ficar com a permissão ativa. Considerando que o *firewall* está sendo desenvolvido, o comando *su* será necessário para que o usuário comum possa virar *root* e assim executar as políticas implementadas no *firewall*. Utilizando o comando *chmod*, atribui-se a permissão *suid* somente ao comando *su*, como visto na **Listagem 19**.

Outro binário que precisamos deixar com a permissão SUID BIT ativa é o passwd, para que os usuários consigam trocar suas próprias senhas. Com o comando chmod atribui-se a permissão SUID BIT ao binário */usr/bin/passwd*, como visto na **Listagem 20**.

Com essa modificação, um dos problemas será resolvido. Mas apenas retirar a permissão SUID BIT dos comandos não evitará problemas no sistema de arquivos. Segundo a norma ISO 27002 nos itens 10.4 e 10.4.1, deve-se proteger a integridade do software e da informação e ter um controle contra códigos maliciosos.

Uma *backdoor* é um código malicioso. Um *cracker* pode instalar uma *backdoor* no sistema utilizando uma técnica como, por exemplo, o *PHP Injection* (inserção de script malicioso através de páginas vulneráveis), a colocando dentro do */tmp* com permissões de SUID BIT e executá-la remotamente.

Para exemplificar um ataque ao diretório */tmp*, é utilizado o comando *adduser* para adicionar um usuário flavio, como visto na **Listagem 21**.

Seguindo o exemplo, será utilizado o comando *cp* para copiar todos os *shells* do sistema para a partição */tmp* e, logo após, será atribuída a permissão de SUID BIT, como visto na **Listagem 22**.

A seguir é apresentada uma sequência de comandos, onde o usuário *flavio* está logado no terminal *tty1*, o que pode ser verificado com o comando *w*. Em seguida é realizado o acesso ao diretório */tmp*, onde é possível executar o *script sh* de acordo com a **Listagem 23**.

Pode-se observar com o comando *id* que o usuário *flavio* ganhou acesso de *root "euid=0(root)"*. Sendo o usuário *root* o dono desse *shell* com a permissão de SUID BIT ativa, o usuário consegue executar o *shell* sh como *root*, conforme mostra a **Figura 5**.

Para evitar esse tipo de problema, alguns parâmetros do comando *mount* podem ser utilizados. O parâmetro apresentado aqui é o *nosuid*, que inibe a execução de binários na partição indicada. Logado como usuário *root*, aplica-se essa opção à partição montada em */tmp*, como visto na **Listagem 24**.



Nota do DevMan

nmap: Network Mapper, é um aplicativo livre e de código aberto, sobre licença GPL, utilizado para explorar uma rede afim de efetuar uma auditoria de segurança.

Listagem 14. Script sugerido para remoção de permissões SUID-BIT.

```
#vim /root/auditoria/localiza_suid.sh
#!/bin/bash
# Envia uma mensagem na saída padrão de vídeo
echo "Verificando arquivos com permissão de
SUID BIT.."
# faz uma busca em todo sistema por arquivos que
contenham a permissão de SUID BIT e salva em um
arquivo texto
find / -perm -4000 > /root/auditoria/list.suid
# Envia mensagem na tela orientando o que deve
ser feito
echo -n "Deseja remover o SUID BIT dos
arquivos?(S/N):"
# Recebe a opção escolhida
read acao

# Executa a ação
case $acao in
 S|s)
  chmod -Rv -s /
  echo " Permissões de SUID BIT Removidas!"
  sleep 3
  exit
 ;;
 N|n)
  exit
 ;;
 *)
  echo "Opção Inválida!!"
  sleep 3
  exit
 ;;
esac
```

Listagem 15. Alterando as permissões do script.

```
#chmod 700 /root/auditoria/localiza_suid.sh
```

Listagem 16. Executando o script com o parâmetro n ou N.

```
#./localiza_suid.sh n
```

Listagem 17. Listando o diretório auditoria.

```
# ls /root/auditoria/list.suid
```

Listagem 18. Executando o script com o parâmetro (s ou S).

```
#./localiza_suid.sh s
```

Listagem 19. Adicionando permissão de SUID-BIT ao binário su.

```
# chmod +s /bin/su
```

Listagem 20. Adicionando permissão de SUID-BIT ao passwd.

```
# chmod +s /usr/bin/passwd
```

Listagem 21. Adicionando o usuário flavio.

```
#adduser flavio
```

Listagem 22. Teste com binários que tenham permissão SUID-BIT.

```
# cp /bin/*sh* /tmp
# chmod 4755 /tmp/*sh*
```

Listagem 23. Usuário comum com permissão de root.

```
$ w
22:31:21 up  3:46,  1 users,  load average: 0,07, 0,07,
0,08
USER    TTY     FROM       LOGIN@   IDLE   JCPU
PCPU WHAT
flavio  tty1    -          22:20    0.00s  0.26s  0.01s w
$ cd /tmp
$ ./sh
```

```
flavio@debian:/tmp$ ./sh
flavio@debian:/tmp# id
uid=1000(flavio) gid=1000(flavio) euid=0(root) grupos=20(dialout),24(cdrom),25(f
loppy),29(audio),44(video),46(plugdev),1000(flavio)
flavio@debian:/tmp# _
```

Figura 5. Usuário comum recebendo direitos de root

```
flavio@debian:/tmp$ ./sh
flavio@debian:/tmp$ id
uid=1000(flavio) gid=1000(flavio) grupos=20(dialout),24(cdrom),25(floppy),29(aud
io),44(video),46(plugdev),1000(flavio)
flavio@debian:/tmp$ _
```

Figura 6. Usuário comum sem direitos de root

```
debian:/# mount
/dev/hda2 on / type ext3 (rw,errors=remount-ro)
tmpfs on /lib/init/rw type tmpfs (rw,nosuid,mode=0755)
proc on /proc type proc (rw,noexec,nosuid,nodev)
sysfs on /sys type sysfs (rw,noexec,nosuid,nodev)
procbususb on /proc/bus/usb type usbfs (rw)
udev on /dev type tmpfs (rw,mode=0755)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
devpts on /dev/pts type devpts (rw,noexec,nosuid,gid=5,mode=620)
/dev/hda1 on /boot type ext3 (rw)
/dev/hda9 on /home type ext3 (rw)
/dev/hda5 on /tmp type ext3 (rw,noexec)
/dev/hda8 on /usr type ext3 (rw)
/dev/hda6 on /var type ext3 (rw)
/dev/hda7 on /var/log type ext3 (rw)
debian:/# _
```

Figura 7. Comando mount, em destaque a partição /tmp com opção noexec

```
debian:/tmp# ./sh
bash: ./sh: Permissão negada
debian:/tmp# _
```

Figura 8. Tentando executar um binário em uma partição com a opção noexec aplicada

```
debian:/# cat /etc/fstab
# /etc/fstab: static file system information.
#
# <file system> <mount point>   <type>  <options>          <dump>     <pass>
proc            /proc           proc    defaults           0          0
/dev/hda2       /               ext3    errors=remount-ro  0          1
/dev/hda1       /boot           ext3    defaults           0          2
/dev/hda9       /home           ext3    defaults           0          2
/dev/hda5       /tmp            ext3    nouser,noexec,rw,auto,dev,nosuid  0  2
/dev/hda8       /usr            ext3    defaults           0          2
/dev/hda6       /var            ext3    defaults           0          2
/dev/hda7       /var/log        ext3    defaults           0          2
/dev/hda10      none            swap    sw                 0          0
/dev/hdc        /media/cdrom0   udf,iso9660 user,noauto     0          0
/dev/fd0        /media/floppy0  auto    rw,user,noauto     0          0
debian:/# _
```

Figura 9. Arquivo /etc/fstab alterado com as opções noexec

É importante lembrar que o diretório */tmp* necessariamente deverá estar em uma partição separada. Isso segue as boas práticas de particionamento.

Após ter aplicado a correção ao diretório */tmp*, faça novamente o *login* com o usuário flavio e tente executar a *shell sh* novamente. Observe que a diretiva *"euid=0(root)"* não mais é listada. Dessa forma o usuário já não tem mais permissões de *root*, conforme mostra a **Figura 6**.

NOEXEC

Um *script* malicioso pode ser inserido em uma partição do sistema causando danos ao servidor. O comando *mount* tem um parâmetro chamado *noexec*, seu uso pode evitar que um *script* seja executado dentro dessa partição. Assim, para solucionar este problema, pode-se aplicar a opção *noexec* e remontar a partição.

Utilizando o comando *mount* com o parâmetro *remount* a partição */tmp* será remontada com a opção de leitura e escrita e acrescentando o *noexec*, de acordo com a **Listagem 25**.

Para conferir se a alteração foi efetivada, utilizamos o comando *mount* novamente (ver **Figura 7**).

Após remontar a partição, foi realizada mais uma tentativa de executar o binário *sh*. No entanto, agora sua execução está impedida, independente do usuário logado. No exemplo foi executado como *root*, como mostra a **Figura 8**.

Realizando a montagem das partições manualmente, os parâmetros não serão aplicados de forma fixa às partições, ou seja, ao reiniciar o sistema, essas configurações serão perdidas. Para que as configurações sejam mantidas mesmo após um restart, basta editar o arquivo */etc/fstab* (responsável por armazenar a tabela de partições do GNU/LINUX, que é lido durante o boot do sistema). Dessa forma, quando a opção *noexec* é inserida diretamente no arquivo, já será implementado o bloqueio ao iniciar a montagem da tabela de partições do sistema. Um exemplo desse arquivo pode ser visto na **Figura 9**.

O administrador poderá encontrar um problema ao executar o aplicativo *aptitude* se as partições */var* e */tmp* estiverem com o *noexec* e *nosuid* ativos, pois esse aplicativo precisa executar *scripts* dentro de tais partições. O *script* da **Listagem 26** pode ser utilizado para contornar tal problema. Dessa forma pode-se remover a proteção ao executar o aplicativo e, em seguida, reativá-las.

Listagem 24. Remontando o diretório /tmp com a opção nosuid.

```
# mount -o remount,rw,nosuid /tmp
# mount
```

Listagem 25. Remontando o diretório /tmp com a opção noexec.

```
# mount -o remount,rw,noexec /tmp
```

Listagem 26. Script para alterar opção noexec e nosuid em partições desejadas.

```
#!/bin/bash
case $1 in
 start)
  # monta as partições listadas
  mount -o remount,rw,noexec /var
  mount -o remount,rw,noexec /tmp
  mount
  echo "Partições SEM permissão de execução"
 ;;
 stop)
  # monsta as partições listadas
  mount -o remount,rw,exec /var
  mount -o remount,rw,exec /tmp
  mount
  echo " Partições COM permissão de execução "
 ;;
 *) echo "erro use $0 {start|stop}"
  exit 0
 ;;
esac
exit 1
```

Conclusão

Neste artigo foram tratados os conceitos de *hardening*, além de demonstrar algumas técnicas para endurecer o acesso ao sistema. Contudo, o assunto é muito extenso e importante, devendo ser tratado como fator fundamental em um projeto de implementação de um servidor, antes mesmo de entrar em produção, desde a instalação até a disponibilização de serviços na rede.

Sistemas como o *AppArmor* e o *SELinux* estão sendo utilizados para aumentar ainda mais a segurança em sistemas Linux, criando uma camada a mais de segurança, e devem ser considerados para também endurecer o sistema.

É importante salientar que existem técnicas de *hardening* mais específicas, como o *hardening* em *kernel*, em serviços como *Apache* e *MySql*, entre outros, que serão temas de novos artigos.



Flávio Alexandre dos Reis
reis.falexandre@gmail.com reisfa.blogspot.com
Tecnólogo em Sistemas de Informação, Pós Graduando em Segurança da Informação - FMG/JF, Certificado LPIC-1.



Marcos Fabiano Verbena
marcosverbena@gmail.com
Bacharel em Sistemas de Informação CES/JF, Pós Graduando em Segurança da Informação - FMG/ JF



Eduardo Pagani Julio
epagani@gmail.com
Mestre em Computação pela Universidade Federal Fluminense. Atualmente professor em Instituições de Ensino Superior da Região de Juiz de Fora em disciplinas ligadas a redes, e consultor atuando principalmente nas áreas de redes de computadores e segurança, com enfoque em Linux.

Site do nmap
http://nmap.org/

Hardening
http://www.ufpa.br/dicas/vir/inv-back.htm

Linux Hardening
http://www.csirt.pop-mg.rnp.br/docs/hardening/linux.html

Cursos Online

WebMobile PLUS



A Revista **WebMobile** oferece para seus assinantes uma série de Cursos Online de alto padrão de qualidade .

Conheça abaixo os cursos já disponíveis:
www.devmedia.com.br/curso/webmobile

Curso em .NET: **Aplicação completa de orçamento Doméstico no Visual Studio 2005**

Curso em Java: **Introdução ao uso de Web Services em Java ME**

A sua melhor opção de aprendizagem!
Assine a **WebMobile** e
Comece já seu treinamento!
www.devmedia.com.br/assine

DevMedia Mais Informações: www.devmedia.com.br/central - Tel.: 21 3382-5038

Sistema de Detecção de Intrusão

Uma abordagem da segurança em redes utilizando o Prelude-IDS

Edelberto Franco Silva e Eduardo Pagani Julio

Este artigo abordará os conceitos e aplicações do Sistema de Detecção de Intrusão (IDS) a fim de que, ao final, o leitor tenha a capacidade de visualizar como este mecanismo de segurança funciona, além de como preparar um ambiente integrado de detecção de intrusão. Mecanismos de segurança são cada vez mais necessários na busca para que se tenha confiabilidade, disponibilidade e integridade em ambientes computacionais e é exatamente este o ponto em que o IDS vem a colaborar. Serão apresentados os modelos de IDS, suas principais funcionalidades e sua finalidade. Ao final será abordada uma ferramenta de IDS assim como sua aplicação em um ambiente real. Para a instalação do IDS, o sistema operacional escolhido é o Linux, porém não se faz necessário que sejam monitorados somente ambientes com este mesmo sistema operacional, uma vez que será demonstrada a interoperabilidade do IDS escolhido por meio de um padrão proposto de formato de mensagens e alertas. Os principais fatores que levaram à escolha deste Sistema Operacional para testes partem desde sua atenção à segurança do ambiente operacional como um todo, assim como a existência de inúmeras ferramentas voltadas a este fim; além de apresentar várias distribuições gratuitas o Linux é *open source*, ou seja, seu código-fonte é aberto, acessível e pode ser modificado pelo usuário (experiente). A ferramenta de IDS a ser apresentada neste artigo é o Prelude-IDS. Cada componente será detalhado, assim como os sensores de terceiros disponíveis e suas maiores virtudes.

Resumo DevMan

De que se trata o artigo:

O presente artigo tem como objetivo principal a apresentação do conceito de Sistema de Detecção de Intrusão (Intrusion Detection System - IDS). IDSs têm em sua essência o auxílio à segurança do ambiente de computação em uma rede de computadores. Neste cenário, a cooperação deve estar presente em vários níveis. No artigo serão expostos conceitos, modelos e uma ferramenta disponível ao usuário. Esta ferramenta é o Prelude-IDS, que servirá como demonstrativo da aplicação em um ambiente real do conceito apresentado.

Para que serve:

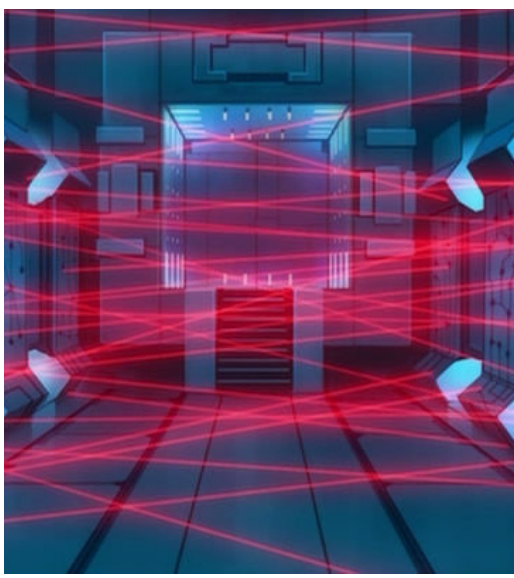
O Sistema de Detecção de Intrusão pode ser utilizado em ambientes corporativos, assim como em redes locais, dependendo de seu modelo empregado. Basicamente tem-se um modelo baseado em host e outro baseado em rede. Sua funcionalidade principal é servir como uma segunda linha de defesa à invasão, agindo posteriormente à constatação da intrusão no sistema. Sistemas neste caso pode ser a rede, um servidor ou um computador pessoal analisando as atividades da rede ou do computador em questão.

Em que situação o tema é útil:

Registros de invasões são notados diariamente aos milhares por toda a Internet, assim como em redes locais cabeadas ou sem fio. Com a constância no crescimento e na dependência dos ambientes computacionais de produção, das mais diversas áreas, pela interligação de sistemas e acesso/armazenamento de dados de forma segura, é necessário agir de forma a evitar a invasão de intrusos à rede. Porém, esta premissa pode nem sempre ser verdadeira, o que faz do Sistema de Detecção de Intrusão uma importante ferramenta no auxílio ao escopo de segurança da rede como um todo, uma vez que age após a constatação de uma provável invasão. É preciso pensar na segurança.

Sistema de Detecção de Intrusão

Normalmente o que encontramos são mecanismos que tentam impedir o acesso indevido ao sistema. Mas quando o sistema se encontra comprometido por algum motivo, seja uma sessão ativa esquecida por um usuário, uma falha de segurança de um software desatualizado instalado no servidor que gerencia toda a rede, ou em um dos computadores da rede participantes da rede local. Neste momento nos deparamos com uma intrusão, e é aí que se faz necessária a utilização de um mecanismo que identifique e alerte ou responda à atividade maliciosa em questão.



O Sistema de Detecção de Intrusão (*Intrusion Detection System* - IDS) pode ser definido como um sistema automatizado de segurança e defesa detectando atividades hostis em uma rede ou em um computador (*host* ou nó). Além disso, o IDS tenta impe-

dir tais atividades maliciosas ou reporta ao administrador de redes responsável pelo ambiente. Trata-se de um mecanismo de segunda linha de defesa. Isto quer dizer que, somente quando há evidências de uma intrusão/ataque é que seus mecanismos são utilizados. A primeira linha defensiva é aquela que tentará limitar ou impedir o acesso ao ambiente, o que pode ser, por exemplo, um *firewall*. O IDS pode apresentar uma forma de resposta a algum tipo de ataque, trabalhando em conjunto com a primeira linha de defesa, por exemplo, incluindo regras no *firewall* ou bloqueando a sessão em questão. Pode ainda reportar as atividades maliciosas constatadas aos outros nós da rede.

O que é IDS e como funciona

Conforme os conceitos descritos em um dos artigos-base sobre IDS (ver referências ao final), podemos conceituar a detecção de intrusão como um processo de monitoramento de eventos que ocorrem em um sistema de computação ou em uma rede e tem o intuito de analisar possíveis incidentes, possíveis violações ou iminências de violações às regras de segurança deste ambiente. Incidentes podem ter várias causas, desde a ação de *malwares* (worms, spywares etc.) até ataques que visam o ganho não autorizado do ambiente em questão.

A utilização de IDS como sistema de prevenção pode envolver desde alertas ao administrador da rede e exames preventivos até a obstrução de uma conexão suspeita. Ou seja, o processo de detecção de intrusão é o de identificar e responder de maneira preventiva atividades suspeitas que possam interferir nos princípios da integridade, confiabilidade e disponibilidade. Além disso, as ferramentas de IDS são capazes de distinguir de onde se originaram os ataques, de dentro ou fora da rede em questão. Os IDS geralmente analisam arquivos locais em busca de rastros de tentativas mal-sucedidas de conexão à máquina, ou até mesmo nas camadas do modelo de pilha TCP/IP abaixo da camada de aplicação, como por exemplo, alterações nos campos do cabeçalho do protocolo IP.

Considerações sobre IDS:

- Não é um software antivírus projetado para detectar softwares maliciosos tais como vírus, *trojans*, e outros;
- Não é usado como um sistema de registro de rede, por exemplo, para detectar total vulnerabilidade gerada por ataques DoS (*Denial-of-Service*) que venham a congestionar a rede. Para isso são usados sistemas de monitoramento de tráfego de rede;
- Não é uma ferramenta de avaliação de vulnerabilidades, verificando erros e falhas de sistema operacional e serviços de rede. Tal atividade é de ordem dos *scanners* de segurança que varrem a rede em busca destas mesmas falhas.

Em um modelo básico de IDS é possível incluir alguns elementos. Primeiramente, as decisões provenientes do IDS são baseadas sob a coleta de dados realizada. As fontes de dados podem incluir desde entradas pelo teclado, registros de comandos básicos a registros de aplicações. As decisões somente são tomadas quando se tem uma quantidade significativa de dados em sua base que confirmam a maliciosidade daquele computador. Os dados são armazenados por tempo indefinido (que podem ser apagados posteriormente), para mais tarde servirem de referência, ou então temporariamente, esperando o processamento. Os dados coletados com informações iguais (considerados, portanto, elementos homogêneos), são cruciais para o trabalho do IDS. Um ou mais algoritmos são executados, procurando evidências para que se tomem rapidamente decisões contra as atividades suspeitas.

Geralmente os IDS são controlados por configurações que especificam todas suas ações. Estas configurações ditam onde os dados serão coletados para análise, assim como qual resposta será resultado para cada tipo de intrusão. O melhor ajuste de configurações ajuda a definir uma maior proteção ao ambiente, porém, o contrário, provavelmente, será prejudicial. O IDS gera um alarme, é ele o responsável por todo o tipo de saída, desde respostas automáticas, alerta de atividades suspeitas ao administrador e notificação ao usuário.

É interessante que se tenha em mente o fato dos alertas não serem conclusivos ou que possa haver existência de erros tanto de análise como de configuração, o que pode gerar os chamados falsos positivos, que são alertas, ou ações, em resposta a evidências encontradas pelo IDS, porém de forma equivocada. A mesma frágil configuração pode gerar falsos negativos, que se conceitua pela falta de alerta ou decisão para um ataque real. Busca-se sempre que o IDS tenha o menor número de falsos positivos e falsos negativos quanto possível.

Tipos de IDS

A detecção de intrusão pode ser classificada em três categorias: detecção por assinatura ou mau uso (*Misuse Detection*), detecção por anomalia (*Anomaly Detection*) e detecção baseada em especificação (*Specification-based Detection*). A seguir serão explicadas de forma mais detalhada cada uma destas categorias.

Detecção por assinatura

Detectores deste tipo analisam as atividades do sistema procurando por eventos ou conjuntos de eventos que correspondam a padrões pré-definidos de ataques e outras atividades maliciosas. Estes padrões são conhecidos como assinaturas. Geralmente cada assinatura corresponde a um ataque ou outra atividade específica. É natural e simples pensar em um exemplo típico deste tipo de evidência, onde o atacante tenta se logar no sistema por meio de um acesso remoto, por exemplo o SSH (*Secure Shell*), e erra a senha por mais de três vezes. Por meio de uma assinatura encontrada nos *logs* (registros) do sistema, ou seja, a linha correspondente ao erro de autenticação, é então emitido ao administrador um alerta e bloqueado o acesso do atacante, conforme a configuração do IDS realizada pelo administrador.

A seguir listamos as vantagens e desvantagens da utilização deste tipo de detecção.

Vantagens da detecção baseada em assinaturas:

- São muito eficientes na detecção (comparando-se com a detecção baseada em

anomalias) sem gerar grande número de alarmes falsos;
• Podem diagnosticar o uso de uma ferramenta ou técnica específica de ataque.

Desvantagens da detecção baseada em assinaturas:
• Estes tipos de detectores somente podem detectar ataques conhecidos, ou seja, que estão incluídos no conjunto de assinaturas que o IDS possui, necessitando-se assim de constante atualização deste conjunto (assim como os antivírus);
• A maioria destes detectores possui assinaturas muito específicas, não detectando dessa forma as variantes de um mesmo ataque.

Detecção por anomalia

Detectores baseados em anomalias identificam comportamentos não usuais (anomalias) em um computador ou na rede. Eles funcionam a partir do pressuposto que ataques são diferentes da atividade normal e assim podem ser detectados por sistemas que identificam estas diferenças. Este tipo de detecção constrói um perfil que representa o comportamento normal de usuário, nós e conexões de rede. Este perfil é construído a partir de dados coletados em um período de operação considerado normal, geralmente sob a supervisão do administrador da rede em questão. Estes detectores monitoram a rede e usam uma variedade de medidas para determinar quando os dados monitorados estão fora do normal, ou seja, desviando do perfil.

Um exemplo clássico a este tipo de detecção é quando um usuário específico utiliza sempre o acesso à Internet durante certo período do dia, no horário comercial, por exemplo. Imaginemos que este usuário é um gerente da empresa que está sendo monitorada pelo IDS baseado em anomalias. Este IDS passou toda uma semana criando o perfil deste usuário e, a partir do último dia daquela semana, emprega seu perfil como mandatário ao horário permitido de utilização da Internet. Certo dia, após a detecção estar ativa, o gerente deseja utilizar o acesso à Internet durante a madrugada para entregar um relatório de última hora, nada usual conforme o perfil criado. A resposta a esta detecção realizada pelo IDS baseado em anomalia é o bloqueio do acesso à Internet para aquele usuário/terminal, que poderia ser uma verdade se não houvesse esta exceção, porém foi tratada na verdade como um falso positivo.

Infelizmente, este tipo de detecção geralmente produzirá um grande número de alarmes falsos, pois o comportamento de usuários e sistemas pode variar amplamente. Apesar desta desvantagem, pesquisadores afirmam que a detecção baseada em anomalias pode identificar novas formas de ataques, coisa que a detecção baseada em assinatura não pode fazer. Além disso, algumas formas de detecção baseadas em anomalias produzem uma saída que pode ser usada como fonte de informações para detectores baseados em assinaturas. Por exemplo, um detector baseado em anomalias pode gerar um número que representa a quantidade normal de arquivos acessados por um usuário particular, com isso um detector baseado em assinaturas pode possuir uma assinatura que gera um alarme quando esse número excede 10%, por exemplo.

Ainda que alguns IDSs comerciais incluam formas limitadas de detecção de anomalias, poucos, se nenhum, confiam somente nesta tecnologia. As detecções de anomalias existentes em sistemas comerciais geralmente giram em torno da detecção de *scans* de rede ou portas. Entretanto, a detecção por anomalias continua na área de pesquisa e pode se tornar muito importante para a o desenvolvimento de IDSs no futuro de forma cada vez mais robusta.

A seguir são listadas as vantagens e desvantagens da utilização da detecção por anomalia.

Vantagens da detecção por anomalias:
• Detecta comportamentos não usuais, logo possui a capacidade de detectar sintomas de ataques sem um conhecimento prévio deles;
• Produz informações que podem ser usadas na definição de assinaturas para detectores baseados em assinaturas.

Desvantagens da detecção por anomalias:
• Geralmente produz um grande número de alarmes falsos devido ao comportamento imprevisível de usuários e sistemas;
• Requer muitas sessões para coleta de amostra de dados do sistema, de modo a caracterizar os padrões de comportamento normais.

Detecção baseada em especificação

Define um modelo muito mais complexo que os anteriores, já que sua análise pode ser realizada nas camadas abaixo da camada de aplicação da pilha de protocolos da Internet ou no nível de controle do sistema operacional. Ela se restringe à operação correta de um programa ou protocolo, e monitora a execução do programa com respeito à definição estipulada. Essa técnica pode fornecer a descoberta de ataques previamente desconhecidos, com isso potencializando sua atividade, além de apresentar taxas muito baixas de falsos positivos. Este modelo de detecção não é tão amplamente divulgado como os demais citados neste artigo especialmente por sua maior complexidade de desenvolvimento e restrição à aplicação que se destina, uma vez que ele visa, por exemplo, uma única aplicação.

Pode-se pensar em um exemplo para este modelo de detecção onde será realizada uma pré-análise das chamadas de um servidor de FTP ao núcleo do sistema operacional. Todas as chamadas e principalmente as consideradas críticas ao sistema são analisadas a fim de se observar possíveis alterações posteriores com objetivos maliciosos. Este tipo de detecção é considerada de mais baixo nível do que as demais citadas neste artigo, o que também demonstra um conhecimento muito mais profundo das ações realizadas pelo programa a ser analisado assim como do sistema operacional e das funções de rede.

IDS baseado em rede

Este tipo de IDS tem por objetivo detectar ataques pela análise dos pacotes que trafegam pela rede através de uma escuta em um segmento de rede (como um *sniffer*-farejador de pacotes). Com isso, um IDS tem a capacidade de monitorar o tráfego

de todos os nós que estão conectados neste segmento, protegendo-os.

IDSs baseados em rede (*Network Intrusion Detection System* - NIDS) geralmente consistem de um conjunto de sensores colocados em vários pontos da rede que monitoram o tráfego, realizando uma análise local do mesmo e relatando ataques a um console central de gerenciamento. Como os sensores são limitados a executarem somente o IDS, eles podem ser mais facilmente protegidos contra ataques.

Este tipo de IDS garante que, com poucos IDSs instalados, mas bem posicionados, pode se monitorar uma grande rede. É geralmente simples adicionar esse tipo de IDS a uma rede e são considerados bem seguros a contra ataques. Porém, apresentam algumas desvantagens, como a dificuldade em processar todos os pacotes em uma rede grande e sobrecarregada. Assim, podem falhar no reconhecimento de um ataque lançado durante períodos de tráfego intenso, além de muitas das vantagens dos IDSs baseados em rede não se aplicarem às redes mais modernas baseadas em *switches*, pois estes segmentam a rede, necessitando assim de ativar portas de monitoramento nesses equipamentos para que o sensor funcione corretamente. Outra grande desvantagem do IDS baseado em rede é o de não poder analisar informações criptografadas, já que essas somente serão visualizadas na máquina de destino.

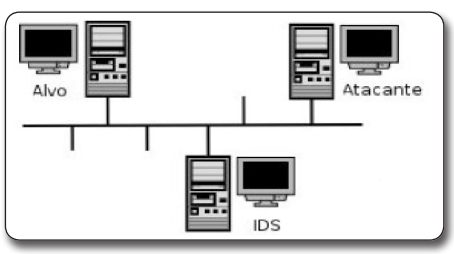


Figura 1. IDS baseado em Rede

Um exemplo de colocação em uma rede de um IDS baseado em rede pode ser visto na **Figura 1**.

IDS baseado em host

Este tipo de IDS é instalado em um host que será alertado sobre ataques ocorridos contra a própria máquina. Este IDS avalia a segurança deste *host* como base em arquivos de *log* do sistema operacional, *log* de acesso e *log* de aplicação, por exemplo. Tem grande importância, pois fornece segurança a tipos de ataques em que o *firewall* e um IDS baseado em rede não detectam, como os baseados em protocolos criptografados, já que estão localizados no destino da informação.

Um exemplo de sua utilização pode ser visto na **Figura 2**, onde um IDS baseado em *host* avisa aos demais sistemas de IDSs dos outros nós sobre a presença de um intruso.

O IDS baseado em host monitora as conexões de entrada no *host* e tenta determinar se alguma destas conexões pode ser uma ameaça. Monitora também arquivos, sistema de arquivos, *logs*, ou outras partes do *host* em particular, que podem ter atividades suspeitas representando uma tentativa de intrusão ou até mesmo uma invasão bem sucedida.

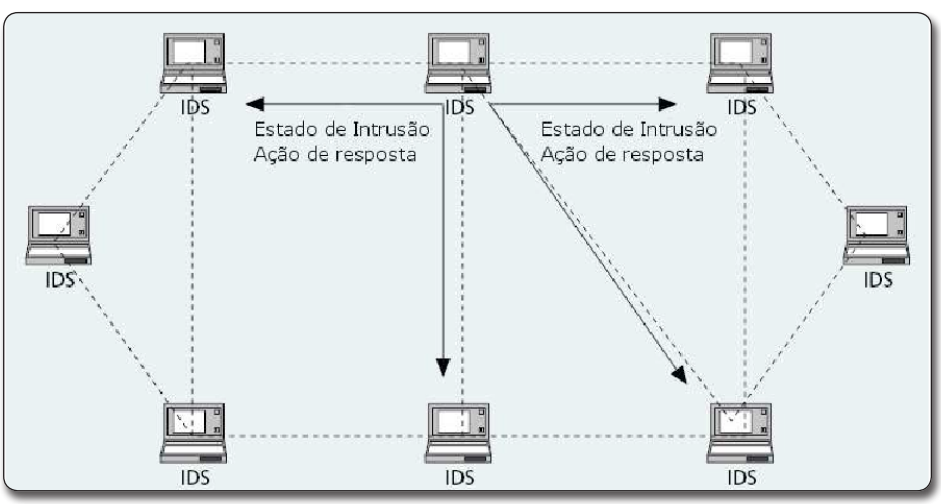


Figura 2. IDS baseado em Host.

Alguns HIDS possuem a capacidade de interpretar a atividade da rede e detectar ataques em todas as camadas do protocolo, aumentando assim a sua capacidade de bloqueio a determinados ataques que não seriam notados pelo *firewall* ou pelo IDS de rede, tais como pacotes criptografados. Esta análise é restrita a pacotes direcionados ao *host* protegido pelo IDS.

Um exemplo de tentativa suspeita que é detectada pelo IDS baseado em *host* é o *login* sem sucesso em aplicações que utilizam autenticação de rede. Desta forma, o sistema IDS informará ao administrador de rede que existe um usuário tentando utilizar uma aplicação que ele não tem permissão.

Para ilustrar os conceitos, é apresentado o IDS Prelude, abordando seus componentes e método de funcionamento. Ao final estaremos aptos, de forma conceitual, a seguir o passo a passo de instalação deste IDS no Linux.

O Prelude-IDS

O Prelude-IDS é um IDS híbrido, ou seja, é um produto que permite a unificação em um sistema centralizado a vários tipos de aplicações, sejam elas de código-fonte aberto ou proprietário, além de poder interoperabilizar as funções de um NIDS com as de um HIDS, sendo em sua essência um NIDS. A fim de implementar tal tarefa, o Prelude-IDS utiliza o padrão IDMEF (*Intrusion Detection Message Exchange Format*) que permite que diferentes tipos de sensores gerem eventos utilizando uma mesma linguagem. Este modelo é baseado na premissa de que não se deve confiar em uma única fonte de informação para uma análise de segurança, já que métodos diferentes de análise têm vantagens diferentes. Assim, o Prelude-IDS decidiu unificar estes métodos em um único produto, produzindo uma ferramenta bastante completa de segurança.

Possui suporte às linguagens de programação C, C++, Python, Ruby, Lua e Perl, de modo que é possível converter aplicações de segurança existentes para a estrutura

do Prelude-IDS. O framework do Prelude é descrito pela **Figura 3**. Ele fornece também sensores que funcionam como analisadores do registro, como é o caso do Prelude-LML. Um sensor de Prelude-IDS pode ser um programa que é capaz de trabalhar em conjunto com o *framework* Prelude-IDS.

O Prelude tem a habilidade de encontrar atividades maliciosas em conjunto com outros tipos de sensores, listados a seguir, além de mais de cem tipos de registros de sistema (*syslog*), entre outros, a fim de melhorar sua evidência ao apontar um ataque:

• **AuditD:** fornece um espaço para o usuário criar suas regras de auditoria, bem como armazenar e pesquisar registros nos arquivos de auditoria a partir do *kernel* 2.6. Possui um *plugin* de detecção de intrusão que analisa os eventos em tempo real e envia por meio de alertas IDMEF ao Prelude Manager;

• **Nepenthes:** é uma ferramenta de emulação de atividades de *malwares*, atuando passivamente neste tipo de emulação com a intenção de tentar explorar tais falhas;

• **NuFW:** é uma ferramenta que funciona como uma extensão para o *Netfilter* (Firewall Linux), que substitui as regras comuns baseadas no gerenciamento de IPs, por informações mais detalhadas sobre os dados dos computadores e dos usuários em si. Sua intenção é agregar maior fidelidade às fontes, já que endereços IPs são trivialmente forjados;

• **OSSEC:** é um HIDS *open source* desenvolvido por um brasileiro. Ele executa a análise de *logs*, verificação de integridade, monitoramente de registro (no caso do Windows) e detecção de *rootkit*, além de prover resposta ativa e alerta em tempo real;

• **PAM:** Linux-PAM é um sistema de bibliotecas que lida com as tarefas de autenticação de aplicações no sistema. A biblioteca oferece uma interface para a concessão de privilégios de programas (como o *login* e *su*);

• **Samhain:** é um sistema *open source* multiplataforma de detecção de intrusão baseado em *host* (HIDS) para POSIX (Unix, Linux, Cygwin/Windows) que fornece verificação de integridade de arquivos, bem como detecção de *rootkits*, monitoramento de portas, detecção de arquivos executáveis SUID desonestos e processos ocultos;

• **SanCP:** é uma ferramenta de segurança de rede projetada para coletar informações estatísticas sobre o tráfego de rede, bem como, gravá-lo em um arquivo utilizando o formato *pcap* para fins de auditoria, análise histórica e descoberta de atividades da rede;

• **Snort:** é um IDS de rede (NIDS) capaz de desenvolver análise de tráfego em tempo real e registro de pacote em redes IP. Executa análise de protocolos e analisa padrões de conteúdo, podendo ser considerado um IDS que trabalha tanto com o método de detecção por anomalia quanto por assinatura, por exemplo, para a detecção de ataques do tipo :

  • *buffer overflows*: sucintamente pode ser descrito como o estouro do limite de espaço de memória para certa aplicação, o que pode gerar desde problemas na execução da aplicação até a exploração de dados por meio de acesso a endereços de memória utilizados por outras aplicações;

  • *stealth port scans*: baseado no escaneamento de portas com a finalidade de encontrar aplicações rodando no computador de destino, e pode ter como objetivo explorar falhas e conseguir acesso por meio destas portas. Várias técnicas são utilizadas para esta verificação, sendo a baseada no three-way Handshake do TCP uma das mais eficientes;

  • ataques CGI: visam a exploração de falhas que possam existir entre a aplicação e o servidor controlados pelo suporte CGI;

  • *SMB probes*: exploração da checagem de compartilhamento que utiliza o SMB;

  • *OS fingerprinting*: tem o intuito de descobrir o Sistema Operacional utilizado pela possível vítima;

  • Além de outras.

Por padrão, oferece em torno de 20 mil regras (somando-se as regras da *Vulnerability Research Team* e do *Emerging Threads*).

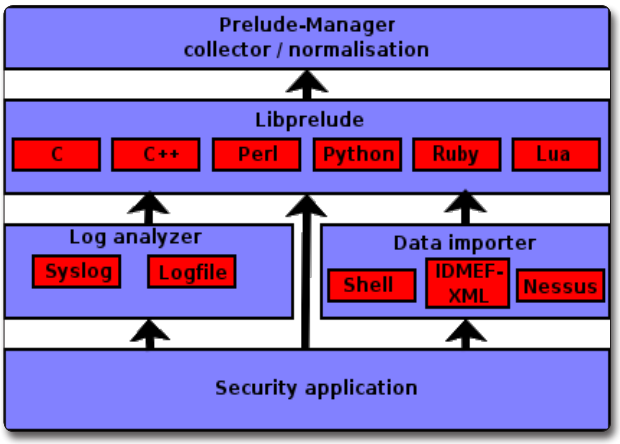


Figura 3. Framework do Prelude-IDS.



Nota do DevMan

IDMEF é um protocolo experimental proposto para trocas de mensagens entre sistemas de detecção de intrusão automatizados. Os sistemas de detecção podem usar o relatório de alerta sobre eventos que considerem suspeitos. O desenvolvimento deste formato padrão tem como objetivo permitir a interoperabilidade entre fontes, sejam elas comerciais ou abertas.

Componentes do Prelude-IDS

Serão apresentados os componentes que formam o *framework* do Prelude-IDS, desde seu sensor de coleta até sua biblioteca de conexão e armazenagem de dados.

• **Prelude Manager:** é um servidor de alta disponibilidade que aceita conexões a partir de sensores distribuídos e/ou outros gerentes, e salva os eventos recebidos para uma mídia especificada pelo usuário (banco de dados, arquivo de *log*, e-mail, etc.). É um concentrador capaz de lidar com um grande número de conexões e processamento de grandes quantidades de eventos. O Prelude Manager vem com vários *plugins* como os de filtragem (por critérios de impacto reportados pela mensagem do IDMEF, limiares de consulta, etc.) e envio de alertas ao administrador por SMTP;

• **LibPrelude:** é uma biblioteca que permite conexões seguras entre os sensores e o Prelude Manager, fornece, ainda, uma API (*Application Programming Interface*) para comunicação com os subsistemas do Prelude, que fornece as funcionalidades

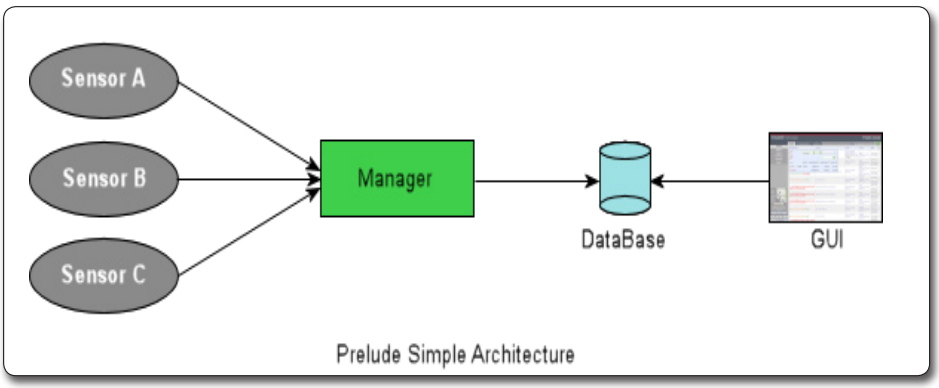


Figura 4. Arquitetura simples de um ambiente com Prelude-IDS



Nota do DevMan

Lua é uma linguagem de programação poderosa, rápida e leve, projetada para estender aplicações, sejam elas em C, C++, Java, C#, Smalltalk, Fortran, Ada, Erlang, e outras linguagens de script, como Perl e Ruby. Suas características a fazem uma linguagem ideal para configuração, automação (scripting) e prototipagem rápida. Lua está presente em muitos jogos (e.g. World of Warcraft) e o middleware para TV Digital Ginga. Essa linguagem foi criada na PUC-Rio.

necessárias para a geração e emissão de alertas em IDMEF, além de garantir retransmissão de alertas para momentos de interrupção temporária. Portanto, é a biblioteca utilizada por terceiros para integração ao Prelude-IDS;

• **LibPreludeDB:** a biblioteca em questão é uma camada de abstração sobre o tipo e o formato do banco de dados para armazenamento das mensagens, permitindo ao desenvolvedor utilizar sua base de armazenamento de mensagens IDMEF sem a necessidade de SQL, independentemente do tipo ou formato da base de dados;

• **Prelude-LML:** é um analisador de *log* que permite ao Prelude coletar e analisar *logs* do sistema a fim de encontrar evidências de ações maliciosas; transforma-o em um alerta IDMEF e então envia ao Prelude Manager. É possível customizar a análise de *logs* por meio de criação de regras (assinaturas). Hoje já avalia mais de 400 regras, dentre elas aplicações conhecidas, como Squid, Sudo, Cacti, Asterisk, OpenSSH, Apache, etc.;

• **Prelude-Correlator:** permite, por meio de criação de regras utilizando a linguagem Lua, conectar e buscar alertas em um servidor remoto do Prelude Manager e ativar correções localmente, conforme suas regras para aquele incidente. Mensagens IDMEF são geradas como alerta de correção;

• **Prewikka:** é a interface Web com o usuário (GUI – *Graphical User Interface*) que provê várias facilidades na análise e administração do Prelude. Além da visualização de alertas, é possível saber o *status* tanto do Prelude Manager como dos sensores.

Na **Figura 4** é possível visualizar uma arquitetura simples do Prelude-IDS, onde temos vários sensores, por exemplo, o Prelude-LML, se conectando a um Prelude Manager que, por sua vez, armazena os dados coletados em uma base de dados (podemos pensar aqui em um banco de dados MySQL) e as exibe ao administrador por meio da GUI, o Prewikka.

Após a demonstração de toda a estrutura do Prelude-IDS, é possível iniciar a parte prática deste artigo, mostrando como instalar o Prelude IDS em uma estação com o sistema operacional Linux.

Instalação do Prelude-IDS

Para a instalação do Prelude-IDS utilizamos a versão disponível no momento da escrita do artigo, o Linux Ubuntu 9.10 de codinome Karmic, com o *kernel* 2.6.31 instalado por padrão. As versões dos componentes do Prelude-IDS foram instaladas de modo a facilitar este guia, todas elas estavam disponíveis no repositório *universe* do Ubuntu. Apesar da versão corrente do Prelude-IDS ser a 1.0 e ser suportada pela versão 10.04 do Ubuntu Linux (codinome Lucid), utilizaremos as versões a seguir, uma vez que adotamos a versão 9.10 do Ubuntu em toda a confecção deste artigo. As versões dos componentes de Prelude-IDS foram:

- Prelude Manager: 0.9.14.2-2;
- LibPrelude: 0.9.24.1-1;
- LibPreludeDB: 0.9.15.3-1;
- Prelude-LML: 0.9.14-2;
- Prelude-Correlator: 0.9.0~beta5-1;
- Prewikka: 0.9.17.1.

Baixando os pacotes necessários

A seguir, mostraremos a etapa em que se faz necessário o *download* dos pacotes do Prelude-IDS, assim como a de alguns outros pacotes. Utilizaremos o comando *apt-get* em todo este processo. Pressupomos que o usuário logado no momento é o *root*. É possível assumir este controle com o comando *sudo –s*.

Na **Listagem 1** pode-se observar a instalação da base de dados MySQL, que armazenará os alertas, e o Apache, servidor web que servirá para a visualização das alertas.

Conforme pode-se notar na **Figura 5**, será necessária a criação de uma senha para o acesso do usuário *root* à base de dados do MySQL. Essa senha será utilizada novamente durante a instalação do Prelude-IDS.

O próximo passo que temos a seguir é a instalação do Prelude Manager, o gerente, que é feito com o comando da **Listagem 2**, assim como o resultado esperado. Após a instalação do prelude-manager é necessário alterar o conteúdo do arquivo */etc/default/prelude-manager* de RUN para yes, conforme exibido na última linha da **Listagem 2**, e rodar o comando */etc/init.d/prelude-manager start*. As telas de configuração com o banco de dados são representadas pelas **Figuras 6**, **7** e **8**.

Figura 6. Tela de inicialização da configuração do Prelude-Manager com o MySQL.

Feito isto, instalaremos o Prelude-LML executando o comando da **Listagem 3**. Porém, atenção ao erro reportado aqui. Este erro existe porque é necessário que o processo do passo da **Listagem 4** seja

executado a fim de registrar o sensor do Prelude-LML junto ao prelude-manager. Somente desta forma o Prelude-LML poderá se comunicar com o prelude-manager e enviar seus alertas a ele.

Ao final desses passos, para que o ambiente do Prelude-IDS esteja completo, precisamos apenas do Prelude-Correlator e da interface Web de administração, o Prewikka. A instalação dos pacotes é realizada pelo comando da **Listagem 5**.

A **Listagem 5**, assim como a **Listagem 2**, pede a alteração do conteúdo de um arquivo cuja opção RUN deverá estar como 'yes'. Esse arquivo está localizado em */etc/default/prelude-correlator* e é necessária a execução do comando */etc/init.d/prelude-correlator start*. Assim que a instalação do Prewikka iniciar, tem-se as telas de

Listagem 1. Instalando pacotes do MySQL e Apache.

```
apt-get install mysql-server-5.1 apache2
```

Listagem 2. Instalação e resultado do Prelude Manager.

```
apt-get install prelude-manager

dbconfig-common: writing config to /etc/dbconfig-common/prelude-manager.conf
granting access to database prelude for prelude@localhost: success.
verifying access for prelude@localhost: success.
creating database prelude: success.
verifying database prelude exists: success.
populating database via sql... done.
dbconfig-common: flushing administrative password
Generating 1024 bits RSA private key... This might take a very long time.
[Increasing system activity will speed-up the process].
Generation in progress... X
+++++O.+++++O

Created profile 'prelude-manager' with analyzerID '1559276264123084'.
Changed 'prelude-manager' ownership to UID:113 GID:122.
 * prelude-manager disabled, please adjust the configuration to your needs
 * and then set RUN to 'yes' in /etc/default/prelude-manager to enable it.
```

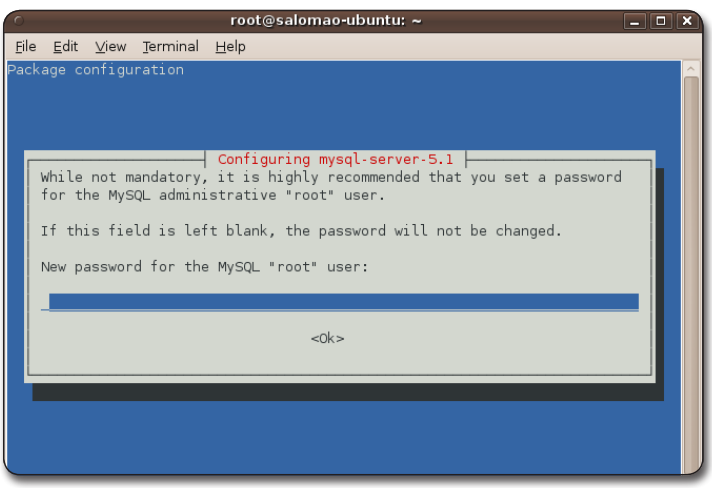


Figura 5. Tela de instalação do MySQL.

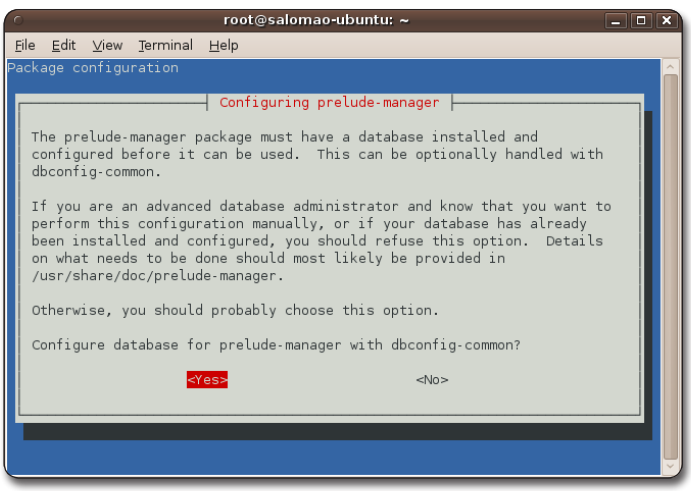


Figura 6. Tela de inicialização da configuração do Prelude-Manager com o MySQL.

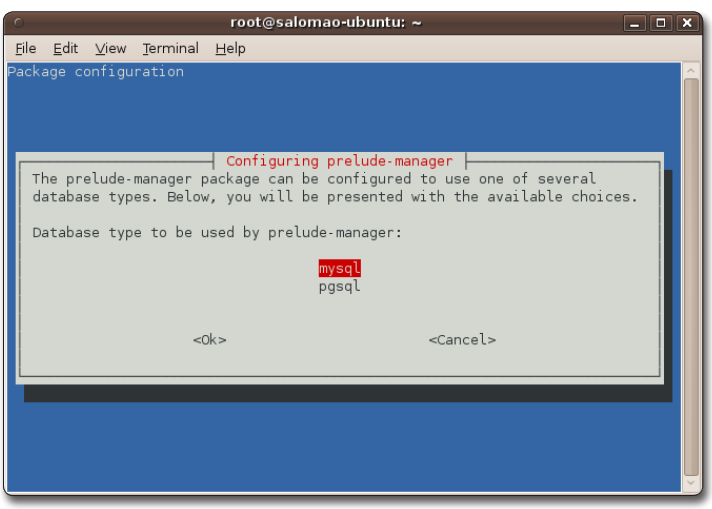


Figura 7. Escolha do Banco de Dados MySQL

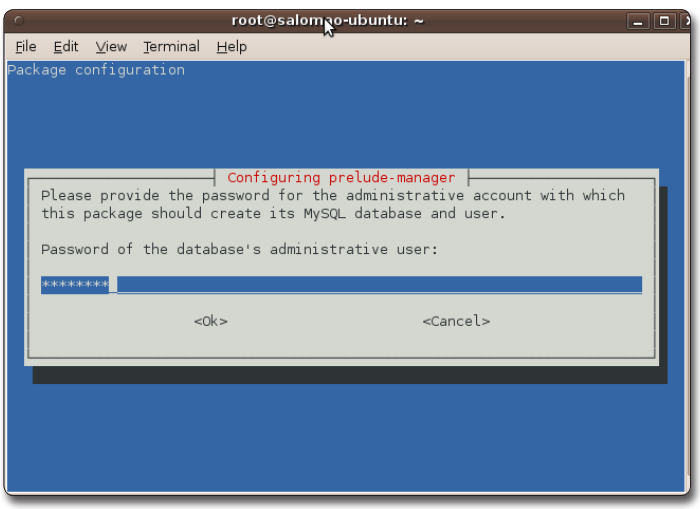


Figura 8. Senha do usuário root criada na instalação do MySQL

Listagem 3. Instalação e resultado do Prelude –LML.

```
apt-get install prelude-lml
* erro na hora de inicializar, isto porque e necessario o comando:
prelude-admin register "prelude-lml" "idmef:w" <manager address> --uid 0 --gid 0
```

Listagem 4. Passos para o registro do Prelude –LML.

```
// Abra outro terminal
---------
TERMINAL 2
root@salomao-ubuntu:~# prelude-admin registration-server prelude-manager
The "ne5mpjua" password will be requested by "prelude-admin register"
in order to connect. Please remove the quotes before using it.

Generating 1024 bits Diffie-Hellman key for anonymous authentication...
Waiting for peers install request on 0.0.0.0:5553...
-----------
E insira no terminal anterior a senha gerada: ne5mpjua

TERMINAL 1
terá como resultado as linhas:
Enter the one-shot password provided on 127.0.0.1:
Confirm the one-shot password provided on 127.0.0.1:

Connecting to registration server (127.0.0.1:5553)... Authentication succeeded.
-----------
TERMINAL 2
e deverá confirmar "Y" na tela anterior o
Connection from 127.0.0.1:40864...
Registration request for analyzerID="276627230846594" permission="idmef:w".
Approve registration? [y/n]:  y
127.0.0.1:40864 successfully registered.
------------

TERMINAL 1
Successful registration to 127.0.0.1:5553.
------------
```

Listagem 5. Instalação do Prelude-Correlator e Prewikka.

```
apt-get install prelude-correlator prewikka

Setting up prelude-correlator (0.9.0~beta5-1) ...
 * prelude-correlator disabled, please adjust the configuration to your needs
 * and then set RUN to 'yes' in /etc/default/prelude-correlator to enable it.
```

configuração do banco de dados para sua futura conexão e exibição na página Web, conforme as **Figuras 9**, **10** e **11.**

Para que o servidor Prewikka seja executado, é necessário rodar o seguinte comando: *prewikka-httpd*. Talvez seja mais interessante rodar esse comando em um *script* de inicialização. Pode ser criado na pasta */etc/init.d/seuscript.sh* e utilizado o comando *update-rc.d* para adicioná-lo à inicialização. O Prewikka é acessado via *browser* pelo endereço, em nosso caso: HTTP://localhost:8000.

A **Figura 12** mostra sua interface Web onde a aba ativa é a de alertas. Nela podemos ver os alertas gerados pelos sensores conectados ao prelude-manager. É possível visualizar alertas de sensores diferentes, como snort e prelude-lml (sshd, netfilter – firewall do Linux). A interface web dispõe de mais possibilidades para a administração do ambiente de rede. É possível, por exemplo, visualizar o status dos sensores, assim como estatísticas de todos os alertas e atividades do IDS. Fica claro que se trata de um ambiente de boa integração com as ferramentas e facilidade de visualização do ambiente de produção onde o Prelude-IDS se encontra em atividade.

Ainda é possível a integração de maneira simples do IDS Snort, citado anteriormente. A única necessidade que se tem é a de registrá-lo como um sensor, após sua instalação e configuração padrão, assim como realizado nos passos da **Listagem 4** para o Prelude-LML.

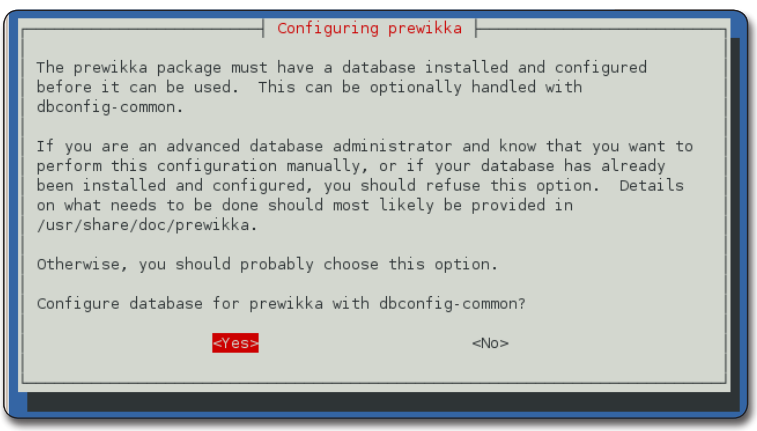


Figura 9. Início da configuração do banco de dados para Prewikka

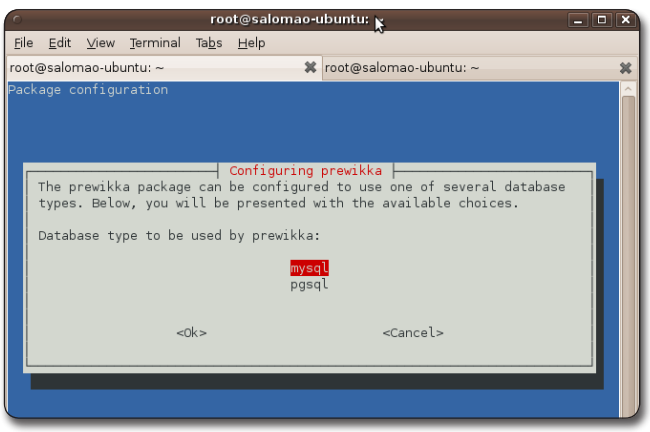


Figura 10. Seleção do banco de dados utilizado

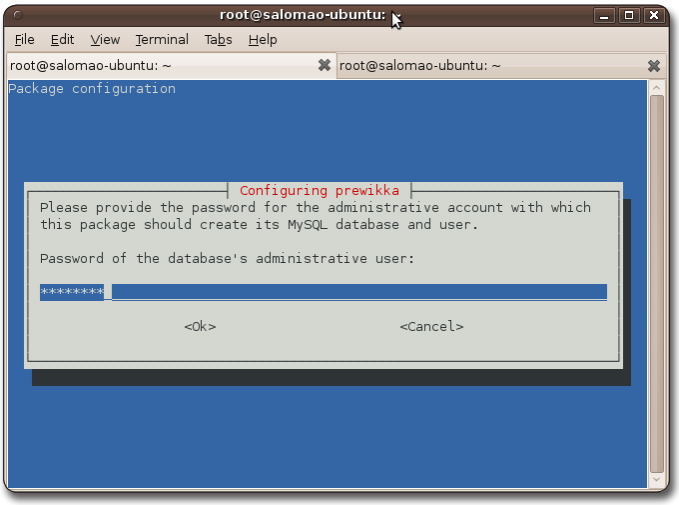


Figura 11. Senha do usuário root criada na instalação do MySQL

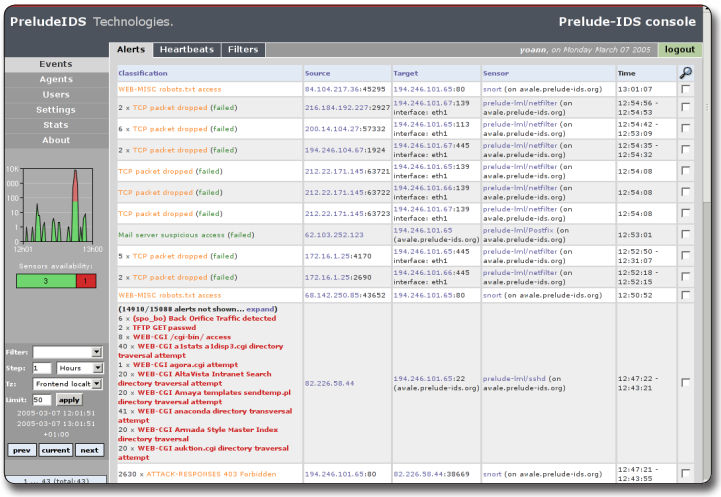


Figura 12. Tela de alerta de eventos da interface web Prewikka

Conclusão

Neste artigo foram apresentados os conceitos de IDS, assim como seus modelos básicos, baseados em rede e *host*, seus tipos de detecção e ainda a ferramenta Prelude-IDS, esta última sendo abordada com a finalidade de apresentar uma aplicação de código-aberto real onde é possível perceber os conceitos de IDS. O Prelude-IDS foi abordado exatamente por ser uma ferramenta integradora de IDS e não somente um único e isolado sistema.

É interessante que se procure analisar os sensores que hoje se integram a este IDS, a fim de torná-lo o mais completo possível. O OSSEC e o Snort merecem destaque neste ponto. Também é necessário dar atenção à correção de eventos por meio do Prelude-Correlator, além da estruturação e criação de bases de alertas replicados para que não haja somente um ponto de armazenamento de dados e falhas. Replicações são feitas de maneira simples, alterando apenas uma diretiva em alguns poucos arquivos de configuração do Prelude-IDS.



Edelberto Franco Silva
esilva@ic.uff.br
Mestrando pela Universidade Federal Fluminense (UFF) em Ciência da Computação e Bacharel em Sistemas de Informação pela Faculdade Metodista Granbery (FMG). Tem como interesse a área de redes sem fio e segurança em redes. Atualmente trabalha em projetos de redes mesh sem fio e de inclusão digital de escolas no âmbito federal, estadual e municipal.



Eduardo Pagani Julio
epagani@gmail.com
Mestre em Computação pela Universidade Federal Fluminense. Atualmente professor em Instituições de Ensino Superior da Região de Juiz de Fora em disciplinas ligadas a redes, e consultor atuando principalmente nas áreas de redes de computadores e segurança, com enfoque em Linux.

Prelude-IDS – IDS Híbrido
http://www.prelude-technologies.com

Snort - NIDS
http://www.snort.org

Snort BR – Comunidade brasileira do Snort
http://www.snort.org.br

OSSEC - HIDS
http://www.ossec.net

Lua – A linguagem de programação
http://www.lua.org

Survey sobre IDS
http://cseweb.ucsd.edu/classes/fa01/cse221/projects/group10.pdf

Pacotes do Prelude-IDS para Ubuntu
http://packages.ubuntu.com/search?keywords=prelude

Stealth TCP Port Scanning
http://hatsecurity.com/2008/05/21/stealth-tcp-port-scanning/

Guide to Intrusion Detection and Prevention Systems
http://csrc.ncsl.nist.gov/publications/nistpubs/800-94/SP800-94.pdf

Cursos Online

Assinatura .net magazine PLUS

Mais conteúdo .NET por muito menos!

A revista **.net Magazine** oferece para seus assinantes uma série de **Cursos Online** de alto padrão de qualidade .
Conheça abaixo os cursos já disponíveis.



Curso em destaque

Crie uma loja virtual completa

Confira neste curso Online como construir uma loja virtual completa no Visual studio 2005. Aprenda como criar o banco de dados, carrinho de compras, profile, página de erros e muito mais .

Confira o plano de aula completo:
www.devmedia.com.br/lojavirtual

A sua melhor opção de aprendizagem!
Assine a **.net Magazine** e comece já seu treinamento!
www.devmedia.com.br/assine

Outros cursos disponíveis: **www.devmedia.com.br/curso**

- Construindo relatórios com Crystal Reports e Visual Studio 2005
- Criando uma aplicação Web Completa
- Criando uma aplicação client/server no Visual Studio 2005
- Aprenda a criar um blog com ASP.NET
- Curso de C#

DevMedia Mais Informações: www.devmedia.com.br/central - Tel.: 21 3382-5038

Gateway com FreeBSD

Compartilhe a internet com FreeBSD

Rogério Tomassoni de A. Junior

Resumo DevMan

De que se trata o artigo:

Mostraremos neste artigo como instalar e configurar o FreeBSD (um sistema operacional robusto e de baixo custo) para atuar como gateway de uma rede local de computadores. Isto permitirá que todos os computadores acessem a internet.

Para que serve:

Fazer com que o leitor seja capaz de entender e aprender como funciona a instalação e a configuração do FreeBSD para atuar como gateway, baseando-se em um cenário real. Ao final o leitor estará apto a colocar em produção o sistema aqui instalado e configurado.

Em que situação o tema é útil:

O conteúdo apresentado neste artigo apóia profissionais e empresas que tenham interesse em saber como é possível utilizar um sistema de baixo custo para realizar o compartilhamento de internet com a rede local, independente dos sistemas operacionais que os demais computadores da rede utilizem.

O FreeBSD é um sistema operacional gratuito, estável e respeitado. Ele está disponível para download no site oficial, sendo compatível com arquiteturas Pentium, Athlon, AMD64, Opteron, UltraSPARC, dentre outras. A maior incidência de uso do FreeBSD é em servidores, porém nada impede de o utilizarmos como usuários domésticos, apesar de ser um sistema não tão simples de configurar para se trabalhar como uma estação de trabalho (onde normalmente encontramos o Windows).

Nesse artigo, focaremos na instalação, configuração e alguns comandos básicos característicos do sistema, de forma que ao final o FreeBSD esteja apto a ser adicionado em uma rede como um gateway e posteriormente ser acrescido de novos serviços.

Assim, faremos com que o FreeBSD atue como o gateway da rede, compartilhando a conexão de internet com outros equipamentos com um ótimo desempenho, confiabilidade, disponibilidade e segurança.

O cenário

Utilizaremos como base uma rede com 200 microcomputadores que precisam de acesso à internet e apenas uma conexão.

Para realizar essa atividade utilizaremos um microcomputador com o sistema operacional FreeBSD, compilado com parâmetros do kernel para atuar como gateway, sendo ele a saída entre os microcomputadores e a internet (veja a **Figura 1**).

A conexão com a internet é via cable modem, onde o provedor, via DHCP (*Dynamic Host Configuration Protocol*), fornece um endereço IP para efetuarmos a conexão.

A instalação

Para realizar a instalação do FreeBSD podemos efetuar o download do mesmo através do endereço *www.freebsd.org*, onde se encontra o projeto oficial e toda a documentação detalhada.

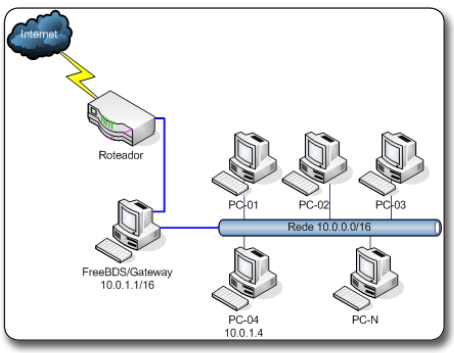


Figura 1. O cenário da rede

A versão utilizada foi a 7.2-RELEASE i386, que pode ser obtida no endereço apresentado na seção **Links** desse artigo. A princípio, baixe apenas o disco 1, pois só precisaremos dele. Em seguida salve a imagem em uma mídia de CD ou DVD.

Com a mídia gravada, precisamos então dizer ao microcomputador para inicializar a partir desta mídia (Boot). Para isso, na BIOS do microcomputador selecione a opção que melhor se adapte ao seu hardware. A BIOS do microcomputador nos fornece uma opção onde é possível optar pelo dispositivo de boot.



Nota do DevMan

Para atuar como gateway e possivelmente como um firewall, podemos utilizar um microcomputador mais antigo e que já não sirva mais para o uso diário.



Nota do DevMan

Caso tenha alguma dúvida sobre os requisitos mínimos para o FreeBSD no momento de instalação, aperte a tecla F1 e utilize a ajuda.

Inicialização (boot) do FreeBSD

O CD Loader do FreeBSD deve ser inicializado e se estiver tudo correto, será exibido algo semelhante ao que segue na **Figura 2**.

Agora começaremos o processo de instalação propriamente dito. Após o boot, veremos o instalador que se chama *sysinstall*. Ele tem a aparência semigráfica (não

tem mouse) e possui uma característica não linear. Dessa forma, existem opções que depois de selecionadas não direcionam o fluxo da instalação para a próxima etapa, sendo necessário acessar outras opções ou subopções para prosseguir a instalação, o que torna a instalação um pouco confusa.

O primeiro passo do *sysinstall* é selecionar o país: 31 para Brasil (**Figura 3**).

Feito isto, o próximo passo é selecionar o mapa do teclado: Brazil cp850 é uma boa opção quando não temos em mãos um teclado que possua o "ç" (um teclado ABNT2 contém cedilha). Mas essa opção pode ser facilmente alterada após a instalação (**Figura 4**).

A próxima janela é o menu principal, onde se encontram as opções mais avançadas. Uma boa prática é instalar somente os itens necessários para que o hardware e os componentes de softwares funcionem corretamente e com um objetivo bem definido. No nosso caso, como um gateway.

Manter o sistema o mais justo e refinado possível contribui com o tempo de inicialização, garante mais segurança, estabilidade e aumenta o tempo em que o sistema está disponível para uso (reduzindo dessa forma o número possível de falhas).

Dispomos de três opções de instalação, sendo que duas delas são indicadas apenas para quem já possui algum conhecimento:

• **Instalação simplificada:** onde apenas o básico para o funcionamento do FreeBSD é instalado. O *sysinstall* é mais simples, porém menos intuitivo, isso quer dizer que ele não exibe muita ajuda explicativa e visual (Express), o que para leigos dificulta um pouco;

• **Customizada:** exige um melhor conhecimento sobre todo o processo de instalação do FreeBSD, incluindo o conhecimento do sistema básico a ser instalado para funcionamento, a manipulação e particionamento dos discos (Custom).

A terceira opção, e a que utilizaremos neste artigo, é a padrão (*Standard*). Esta é a mais amigável. O *sysinstall* fornecerá os utilitários necessários para manipular os discos, selecionar os aplicativos a serem instalados, permitirá a configuração de alguns serviços e as interfaces de rede (observe a **Figura 5**).

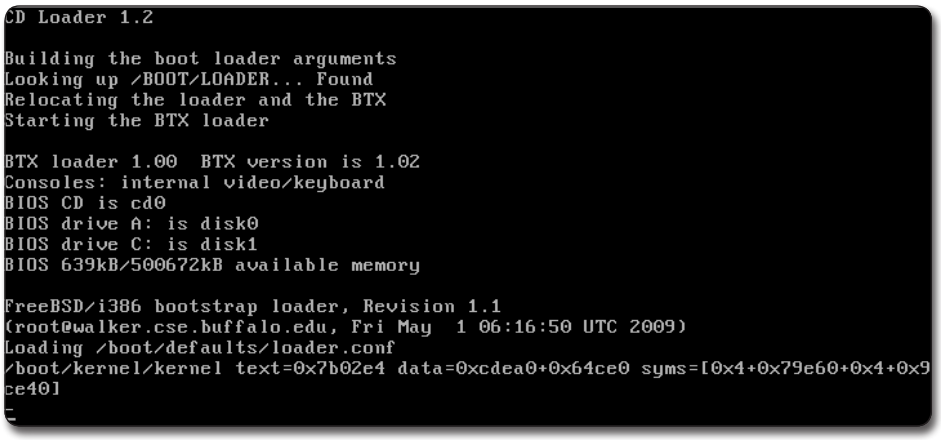


Figura 2. CD Loader FreeBSD

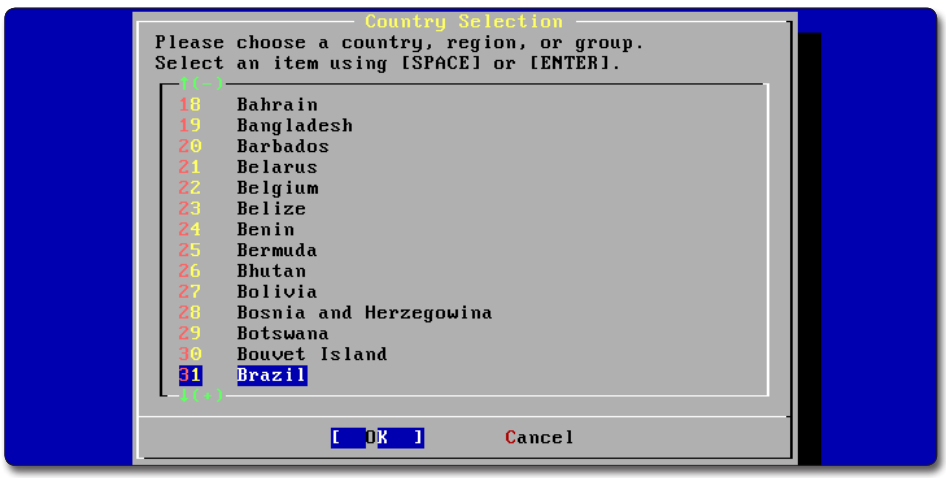


Figura 3. Escolha do país

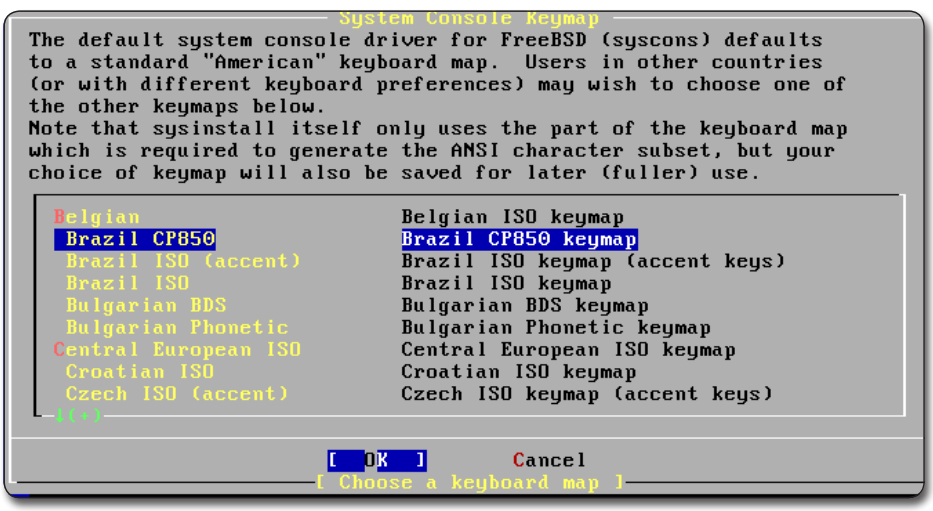


Figura 4. Mapa de teclado

O próximo passo é manipular os discos. Nos sistemas UNIX existe uma característica quanto à alocação e definição do espaço em disco chamado de *Slice*. Comparado a outros sistemas, seria as chamadas partições. No FreeBSD temos os chamados labels, que são subdivisões dentro da *Slice*.

No FreeBSD teremos então que indicar o quanto de espaço físico disponível será fatiado para que possamos dentro dessas fatias criar nossos rótulos (*Label*). Neste ponto, é interessante perceber que podemos optar entre ter uma fatia (*slice*) com vários rótulos (*labels*), ou termos várias fatias.

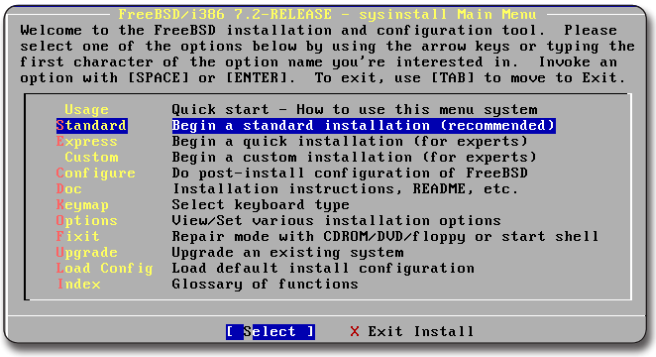


Figura 5. Menu principal

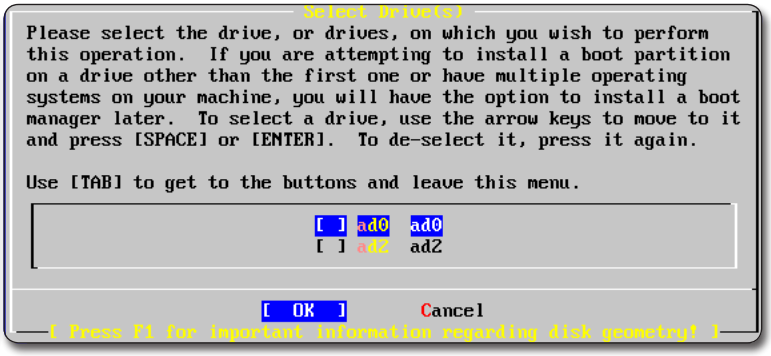


Figura 6. Seleção de discos

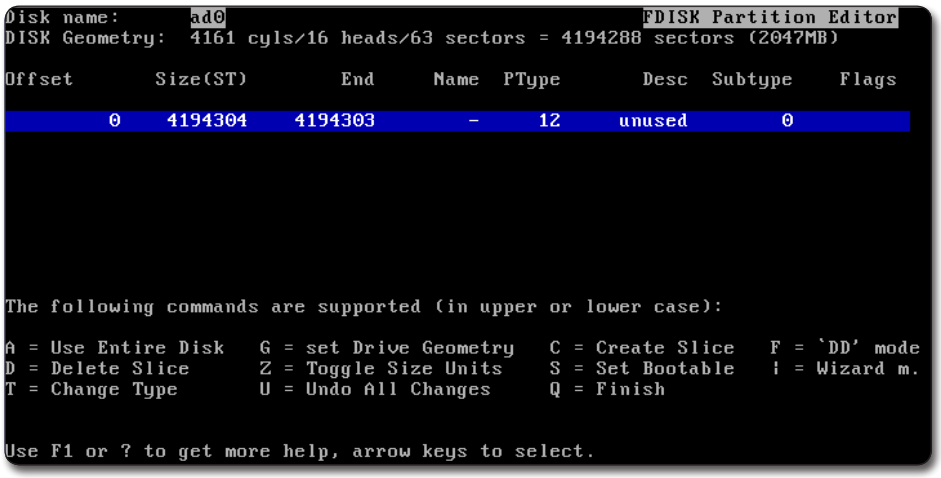


Figura 7. Disco sem partições e com espaço disponível

Label	Ponto de montagem	Descrição
A	/	Partição raiz. Contém a estrutura básica de todo o sistema, é obrigatório e requerida para o sistema de arquivos e funcionamento do FreeBSD.
B	Não tem	Espaço alocado como auxílio à falta de memória. Quando o sistema tem pouca memória RAM disponível e existe a necessidade de utilizar mais que o real existente, este espaço é usado como auxiliar. Aconselha-se que o tamanho seja duas a três vezes maior que a memória RAM física existente.
E	/var	Diretório onde são gerados os logs. Dependendo da capacidade do disco (ex.: 80Gb), aconselha-se alocar 5 Gb, mas nada impede que seja maior.
F	/usr	Partição onde ficam os binários compilados dos aplicativos utilizados pelo sistema e usuários.
G	/home	Partição que irá conter subdiretórios para cada usuário. Esses subdiretórios serão criados no sistema cada vez que um novo usuário for adicionado. Logo, ele contém os perfis dos usuários e seus respectivos diretórios e conteúdos.

Tabela 1. Configurações sugeridas

O FreeBSD também identifica os discos da maneira que são instalados fisicamente, logo, as opções de configuração na BIOS são ignoradas. Esse fato pode causar uma confusão quando configurarmos na BIOS qual disco contém o boot, pois o FreeBSD sempre assumirá a ordem em que forem encontrados os discos.

Os discos serão identificados como *ad0* para disco primário na IDE0, *ad1* para disco secundário na IDE0, *adb0* para disco primário na IDE1, *adb1* para disco secundário na IDE1 e assim sucessivamente (veja a **Figura 6**).

Quando é detectado apenas um disco, então o *sysinstall* não exibe o menu de seleção de discos (**Figura 6**) e já notifica a necessidade de particionar o disco encontrado.

Após a mensagem, o *sysinstall* invoca o utilitário para manipulação dos discos, o *fdisk*.

A nomenclatura para um rótulo/label/partição na primeira *slice* do disco primário da primeira IDE (IDE0) será identificado como *ad0s1a*, onde *ad0* representa o primeiro disco da primeira IDE, *s1* o Slice 1 e *a* o Label.

Sugere-se utilizar as especificações do projeto oficial para a estrutura nos discos. A **Tabela 1** foi retirada do site oficial. Nela podemos verificar quais são as estruturas básicas e o tamanho mínimo para efetuar a instalação do sistema.

O label "a /" (Partição raiz) é nomeado segundo o critério de identificação dos discos e configurado nos arquivos de inicialização (*/etc/fstab*). Logo, se trocarmos fisicamente os discos de posição, o disco primário passa a ser outro, afetando todo o sistema de arquivo, prejudicando o sistema como um todo.

Criaremos então uma *Slice* que irá conter nossos labels. Para isso, selecione o disco para podermos manipulá-lo.

É necessário que haja espaço suficiente disponível em disco para realizar a instalação. Caso o disco seja usado onde existam partições, e deseja-se removê-las deixando os discos totalmente livres para a criação das novas *Slices* para o FreeBSD, pode-se removê-las selecionando a partição e utilizando a opção D (D – Delete). Atenção, esse procedimento removerá a partição e seus conteúdos definitivamente, então se certifique que não existam dados importantes antes de realizar a remoção.

Havendo espaço disponível, o *fdisk* colocará em descrição (Desc) a palavra **unused**, conforme apresentado na **Figura 7**.

Para criar uma *slice*, utilize a opção C (Create). Em seguida será solicitado o tamanho dessa *slice*. É nesse momento que podemos dividir o disco caso exista a necessidade de instalar outros sistemas. O tamanho deve ser informado em Mb (Megabyte). Como criaremos apenas uma *slice*, usaremos o tamanho total disponível que já vem preenchido por padrão (ver **Figura 8**).

Confirmado o tamanho, será solicitado o tipo para esse espaço alocado (cada tipo de sistema de arquivos/partição requer sua numeração própria para identificação. Se colocássemos o número decimal 130 ela seria identificada e reconhecida pelo sistema como sendo *Swap*). O FreeBSD identifica a *Slice* através do número decimal 165, assim devemos informar ao *fdisk* este número, pois precisamos criar uma *slice* (**Figura 9**).

Confirmado o tipo da *Slice*, retornaremos ao *fdisk* e a *Slice* estará criada. Outros tipos de identificação podem ser visualizados e utilizados através da opção T = Change Type, porém em nossa instalação utilizaremos apenas o tipo para o FreeBSD (veja **Figura 10**).

O próximo passo é selecionar se queremos ou não instalar um gerenciador de *boot*. Quando existe mais de um sistema operacional instalado já com um gerenciador, a melhor opção é *None*, deixando a responsabilidade ao gerenciador de boot já existente. Caso queira instalar o gerenciador do FreeBSD com as suas características assumindo a tarefa de gerenciamento, selecione *BootMgr*.

A melhor opção neste cenário é selecionar a opção *Standard*, que irá instalar na MBR (*Master Boot Record*) o loader sem gerenciador, pois não temos outros sistemas instalados. Assim, podemos carregar diretamente nosso sistema reduzindo o tempo de inicialização (**Figura 11**).

Feito isso, aparecerá uma mensagem informando que é preciso criar os rótulos/partições (ver **Figura 12**).

Confirmado, o *sysinstall* apresentará o editor de partições onde criaremos então os rótulos/labels.

Criaremos dois labels, sendo uma a *swap*. Esta é a área de troca utilizada quando a memória RAM disponível não é suficiente e o sistema necessita de mais espaço (uma espécie de memória auxiliar).

No nosso exemplo, definiremos a área de *swap* com espaço de 512M. Então selecione o disco com a *slice* criada (*ad0s1*) e utilize a opção C (Create), como na **Figura 13**. Será questionado o tamanho do label, que

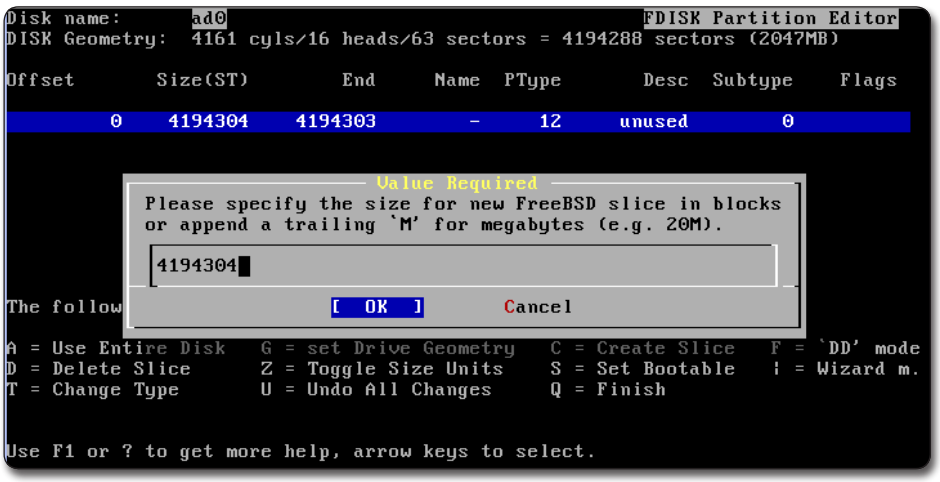


Figura 8. Criando a Slice

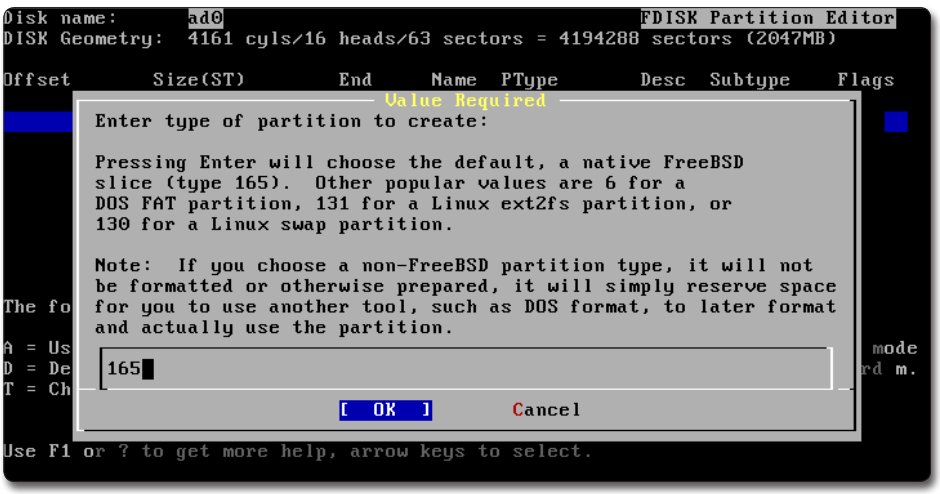


Figura 9. Tipo da Slice

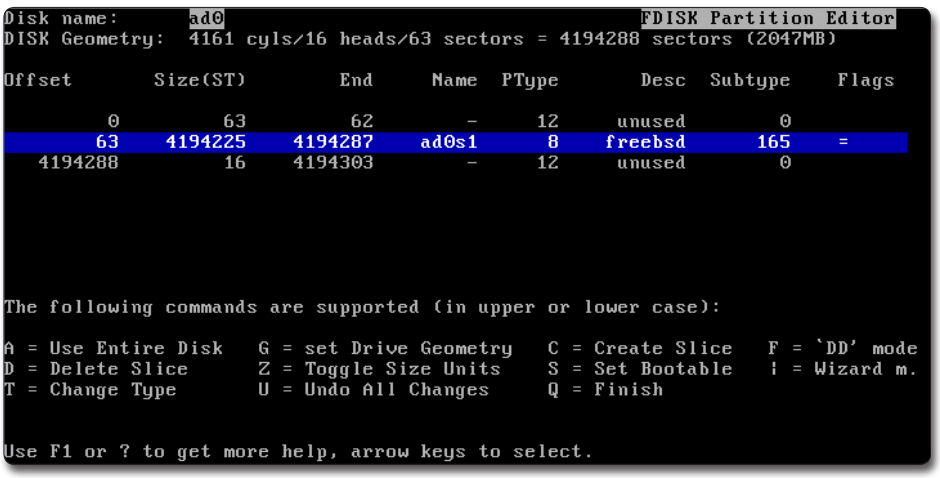


Figura 10. Slice criada

poderá ser informado usando as métricas: G para gigabytes, M para megabytes e C para cilindros.

Após confirmação, é preciso escolher o tipo da label/partição, que pode ser *swap* ou *fs* (*file system*). Nesse momento selecionaremos *swap*, com 512 Mb, conforme a **Figura 14**.

Feito isso, criaremos a partição raiz / com o restante do espaço disponível em disco. Lembrando que a criação dos labels e *slices* fica a critério do administrador, cabendo a ele decidir qual é a melhor opção de manipulação para o ambiente e para os serviços que irá servir, desde que a arquitetura básica de *slices* e labels exigida pelo FreeBSD seja mantida. Por exemplo, para um servidor Proxy, onde existe a necessidade de cache, é realizado um grande número de leitura e escrita. Nesta situação, separar o disco ou manter um rótulo específico para essa atividade ajuda no desempenho e na segurança dos sistemas. Para um servidor de arquivos ou um sistema de terminal com perfil móvel, é aconselhável separar os discos, onde um manteria o sistema operacional e seus aplicativos e em outro disco manteríamos o */home*, que contém todos os arquivos pertinentes a cada usuário. Essa divisão ajuda com o desempenho devido às operações de escrita e leitura dos discos, bem como facilitaria os backups.

Neste contexto, é importante termos em mente que o editor de discos dispõe de uma opção (A – Auto Defaults, apresentada na **Figura 14**) que manipula e cria os rótulos/labels automaticamente levando em consideração a arquitetura básica de discos exigida pelo FreeBSD.

Utilizaremos os mesmos passos das **Figuras 12** e **13** com a finalidade de aprendermos como criar nosso label raiz (/) com seu respectivo ponto de montagem. Atente apenas para o detalhe de selecionar o tipo de label, que deve ser do tipo *fs* (*file system*), e então será preciso especificar o ponto de montagem, nesse caso: raiz (/) – **Figura 15**.

Com as partições/labels criadas, pressione Q (*Finish*) para sair (**Figura 16**).

Neste momento, o próximo passo é selecionar os pacotes a serem instalados. Uma boa prática, como já citado, é instalar somente o necessário para o hardware e os componentes de software funcionarem, mas agora, para simplificar o artigo, vamos instalar todos os pacotes utilizando a opção *All* (**Figura 17**).

Uma alternativa para esta decisão é, depois de instalado, verificar quais hardwares o FreeBSD reconheceu, avaliar os softwares necessários e depois fazer uma instalação customizada, tanto com kernel, quanto com os aplicativos necessários.

No exemplo, selecione *All* (Todos), pressione *Enter*, confirme as mensagens e use a opção *X Exit* para retornar ao menu anterior e prosseguir a instalação (veja **Figura 17**).

Agora devemos selecionar a origem dos pacotes a serem instalados. No nosso caso

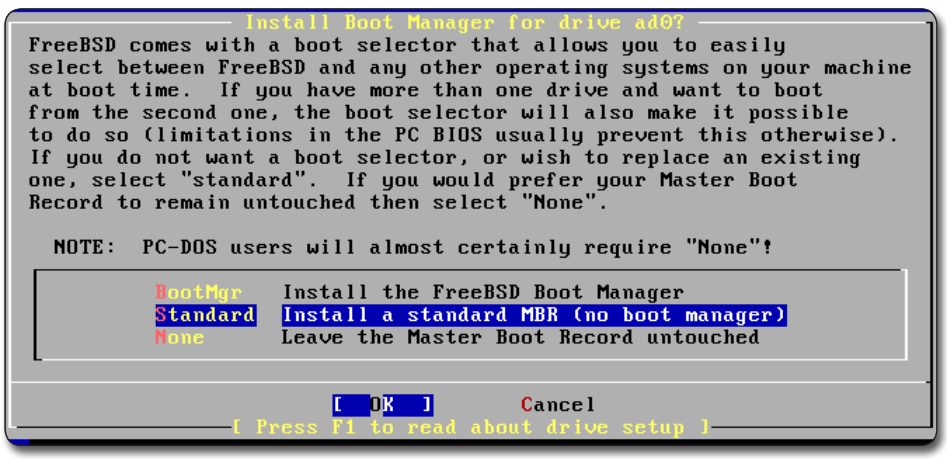


Figura 11. Instalação do gerenciador de boot

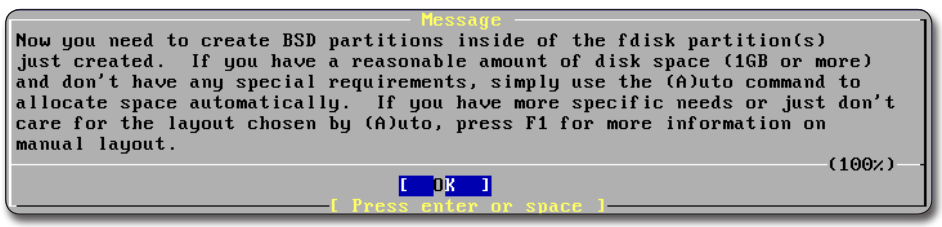


Figura 12. Mensagem para criação de partições

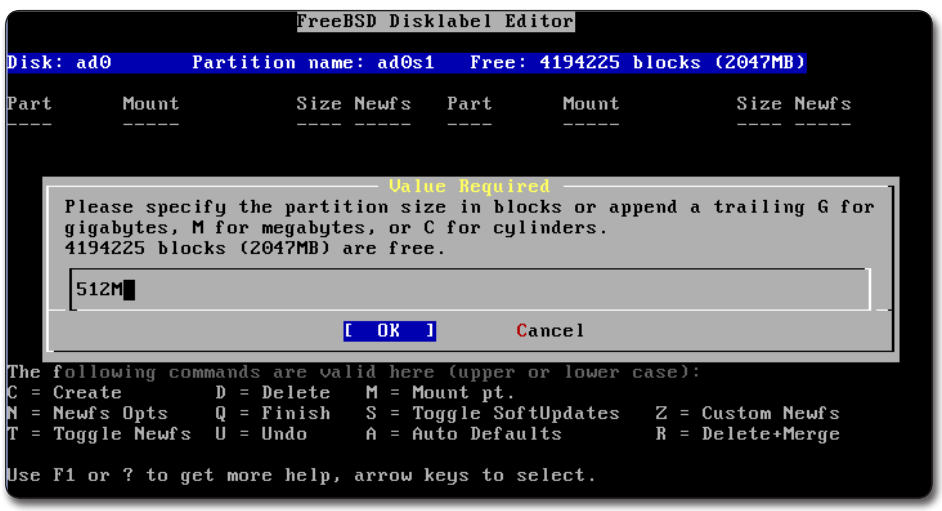


Figura 13. Criando a área de troca swap

Nota

O Secure Shell ou SSH é, simultaneamente, um programa e um protocolo de rede que permite a conexão com outro computador na rede de forma a executar comandos de uma unidade remota com a vantagem da conexão entre o cliente e o servidor ser criptografada.

é a imagem que gravamos no CD que foi baixada do site oficial. Entretanto, como podemos notar, as origens podem variar. Selecionaremos a opção CD/DVD de acordo com a **Figura 18**.

Depois de confirmado, *sysinstall* faz o questionamento se realmente queremos aplicar as alterações e manipulações dos discos realizadas nas etapas anteriores. Atenção neste momento, pois será afetada toda a estrutura dos discos envolvidos levando à perda de possíveis dados (**Figura 19**).

Feito isso, todas as configurações serão aplicadas e o sistema começará a ser copiado para os discos. Caso tudo ocorra bem, os próximos passos serão as configurações finais.

As questões a seguir são referentes aos serviços que desejamos ativar. Todas elas poderão ser ativadas após a instalação. Dessa forma, selecione sim nas opções *gateway* e *ssh*, as demais fica a seu critério (**Figura 20**).

Em seguida, as configurações serão aplicadas e algumas outras questões surgirão como o tipo do mouse, horário UTC (para o UTC Brazil as opções são: **2 -9 - 8**). Configure-os e depois selecione a opção *Exit* para retornar ao instalador (**Figura 21**).

Neste momento seremos direcionados à mensagem de confirmação para a instalação da coleção de pacotes/ports, onde podemos selecionar entre as diversas categorias os que serão copiados localmente no disco, ficando a disposição para futura instalação.

Note que neste ponto não estamos instalando os pacotes e sim os copiando para o disco para futuras instalações.

Ports é uma estrutura de diretórios localizada em */usr/ports* que possui um vasto número de aplicativos nativos aptos a serem instalados facilitando muito a administração e atualização dos aplicativos. É um repositório de aplicativos a disposição para futuras instalações, por exemplo, se quiséssemos instalar futuramente o servidor HTTP Apache, poderíamos ir até os ports em */usr/ports/www* e instalar o Apache13.

Com os ports é preciso pouca ou nenhuma intervenção no que diz respeito às de-

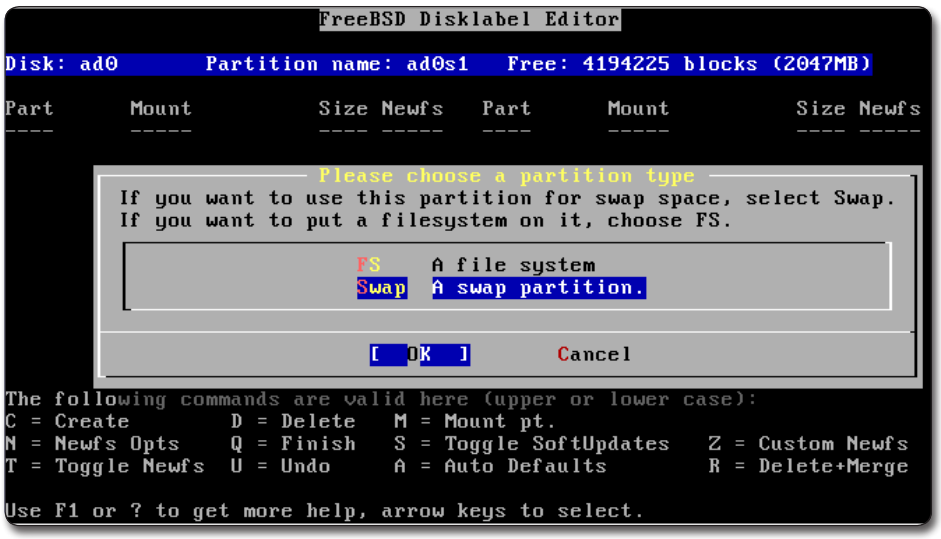


Figura 14. Selecionando o tipo de partição

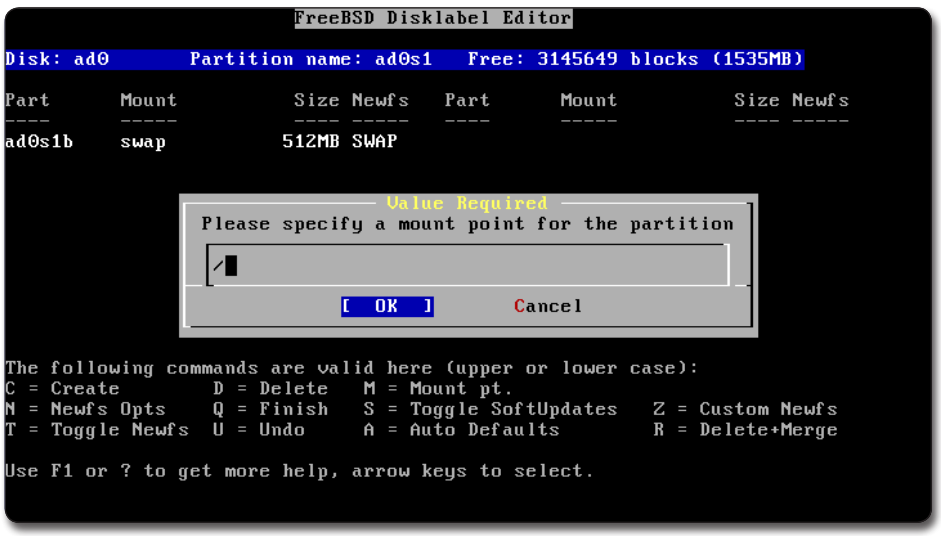


Figura 15. Criando a label e especificando o ponto de montagem

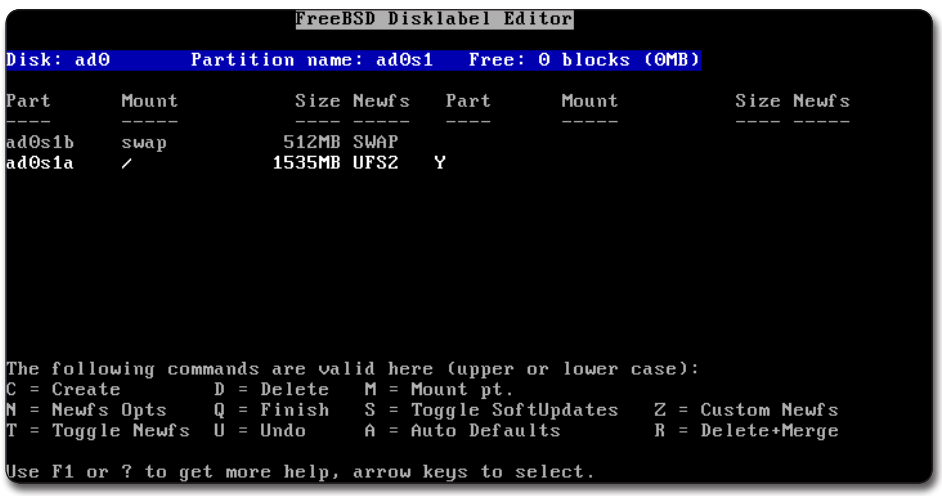


Figura 16. Partições criadas

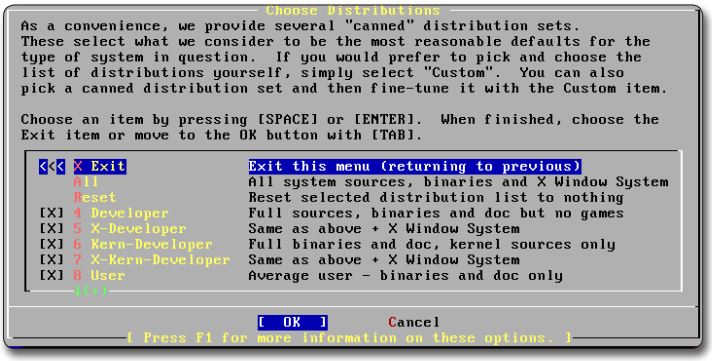


Figura 17. Confirmando pacotes a serem instalados

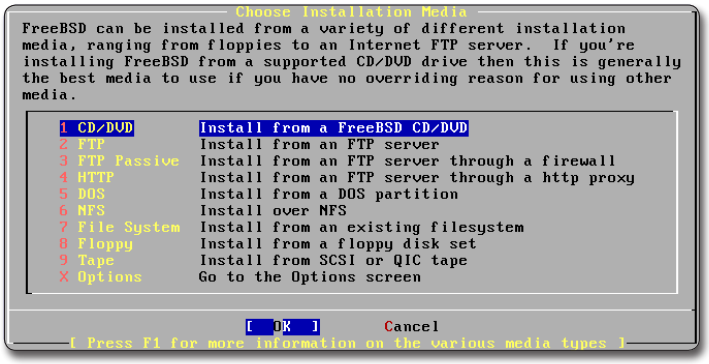


Figura 18. Selecionando o local onde obter os pacotes para instalação

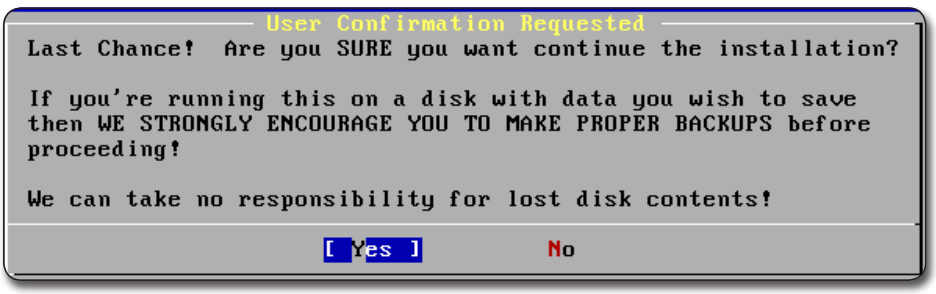


Figura 19. Confirmação da instalação

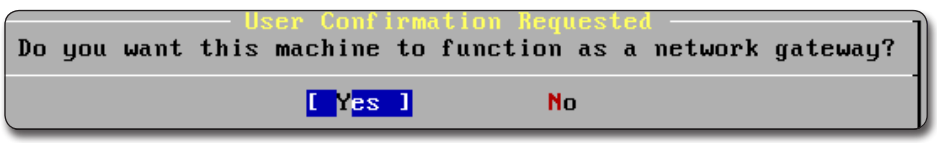


Figura 20. Ativando a opção de gateway

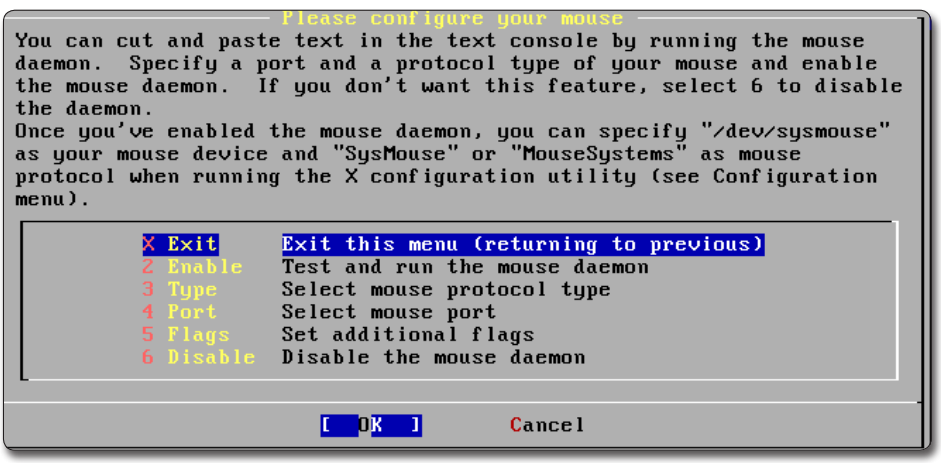


Figura 21. Configurando o mouse e usando opção Exit

pendências entre aplicativos e bibliotecas, pois os arquivos de construção e compilação dos códigos fontes já estão preparados para serem instalados no FreeBSD (veja a **Figura 22**).

Pressione *Enter* e depois selecione a opção *Install*. Confirmando o aviso sobre a realização da operação, uma nova mensagem será exibida referente à criação da conta de usuário para administração do servidor, confirme. O próximo passo é criar esta conta no sistema. Devemos evitar o uso do usuário *root* e com isso prevenimos possíveis problemas de segurança. Após a instalação podemos utilizar o *su*, utilitário que eleva o privilégio de um usuário comum aos privilégios de super usuário/*root*.

Mas para utilizar o *su* após o término da instalação, este usuário tem que obrigatoriamente pertencer ao grupo *wheel*. Podemos então já adicioná-lo ao grupo através do *sysinstall* neste ponto, ou adicioná-lo mais tarde após a instalação, com o comando *pw user mod nome-do-usuário -G wheel*.

Para criar um novo usuário selecione a opção *User* (**Figura 23**).

Preencha os dados nos campos como segue e adicione o usuário ao grupo *wheel*, como mencionamos (veja **Figura 24**).

Confirme em OK e selecione a opção *Exit*. A próxima mensagem é referente à configuração da senha para o usuário administrador do sistema, no caso *root*. O *root* é o usuário com maior nível de privilégio, então cuidado com a senha escolhida para ele.

Após informar e confirmar a senha de root, uma nova mensagem de confirmação questionando se desejamos revisar as configurações realizadas anteriormente é apresentada. Selecione *NO* e prossiga. No menu principal, selecione X *Exit Install* (**Figura 25**).

Então *sysinstall* exibe uma nova mensagem questionando a saída. Em caso de confirmação, avisa para remover os discos e mídias de boot, de acordo com o exposto na **Figura 26**.

Depois de remover o CD reinicie o computador. Feita a instalação, uma boa prática é trocar o boot do computador na BIOS para o disco primário e remover a verificação de presença do teclado. Isso evita que nosso computador deixe de ser iniciado e fique parado aguardando uma intervenção humana caso encontre uma mídia com boot, além de colaborar com o tempo do processo de inicialização.

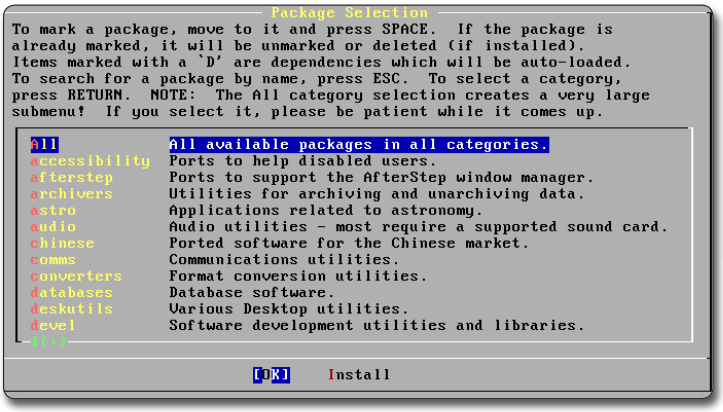


Figura 22. Selecionando pacotes

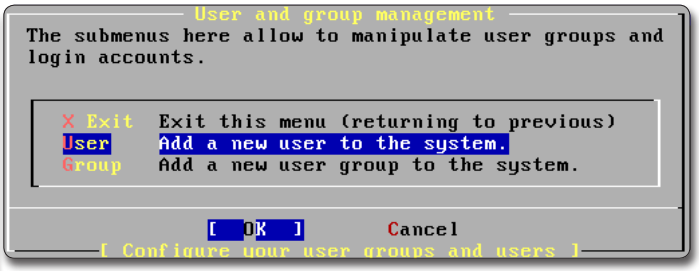


Figura 23. Adicionando uma conta de usuário

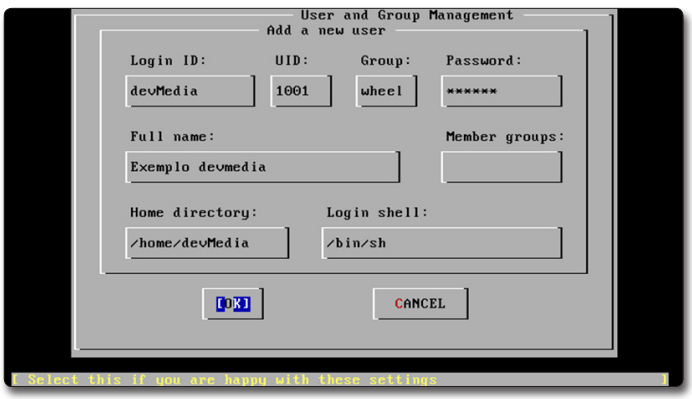


Figura 24. Criando uma conta de usuário

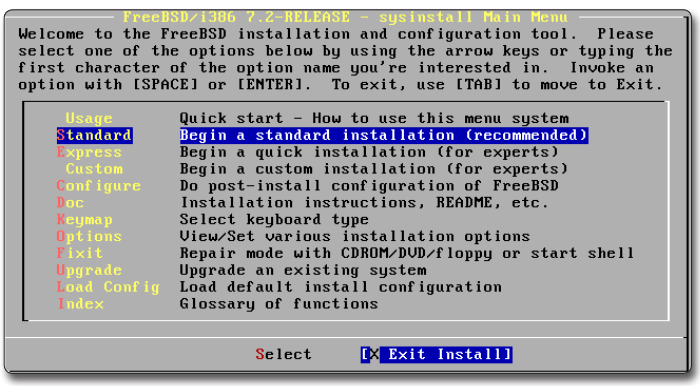


Figura 25. Executando a instalação, após as configurações

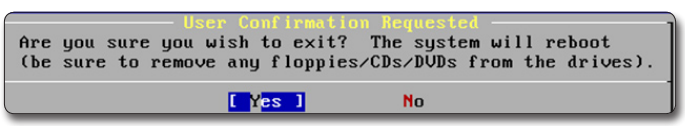


Figura 26. Finalização da instalação



Figura 27. Boot loader no disco

Neste ponto o FreeBSD deve estar instalado e o boot deve acontecer normalmente (veja **Figura 27**).

Configurando o FreeBSD

Com o FreeBSD instalado, iremos então configurá-lo para que possa conectar-se à internet com uma placa de rede via cable modem e compartilhar a internet com os demais computadores da rede.

Efetue o login no sistema com o usuário criado, # *devMedia*. Será exibida uma mensagem de boas vindas indicando alguns links úteis referentes ao FreeBSD. Use o comando *su* - e informe a senha cadastrada para o usuário root na instalação. Ele irá elevar o nível de privilégio do usuário *devMedia* e também ajustar as variáveis de ambiente com as de root (*$ su -*), lembrando que o usuário *devMedia* deve fazer parte do grupo *wheel*.

Iniciaremos as configurações do sistema verificando se as placas de rede foram detectadas e estão funcionando corretamente. Para isso, usaremos os comandos *pciconf* e *ifconfig*.

Verificando placas de rede com *pciconf*: #pciconf –lv

O retorno deverá exibir informações não apenas das interfaces de rede, mas também de todos os dispositivos detectados. As interfaces devem estar entre as opções com o nome e modelo do fabricante.

Depois de identificadas, verificamos com o comando *ifconfig* se as interfaces foram detectadas e instaladas corretamente. O retorno deverá ser algo semelhante à **Figura 28**, onde verificamos que foram detectadas as duas interfaces de rede: *re0* e *rl0*. #ifconfig

O arquivo de configuração do sistema (arquivo onde informamos quais serviços serão iniciados junto com sistema, também usado para modificar valores padrões de variáveis na inicialização do sistema) é o */etc/rc.conf*. Através dele podemos configurar, por exemplo, o comportamento das placas de rede, o *encoding*, o mapa de teclado e a zona de horário. Um exemplo completo do arquivo pode ser encontrado em */etc/defaults/rc.conf*.

Agora ajustaremos os parâmetros necessários para configurar nosso FreeBSD para atuar como Gateway da rede. Dessa forma, edite o arquivo com editor *vi*, que por padrão vem instalado, adicionando ao *rc.conf* as linhas apresentadas na **Figura 29**.

Aqui vale uma breve explicação sobre os parâmetros. Eles sempre obedecem à sintaxe **nome=valor**. Quando alteramos um valor, na verdade estamos sobrescrevendo os valores padrão. Os nomes e valores padrão estão em um arquivo de mesmo nome, porém em outro diretório (*/etc/defaults*).

Explicaremos os parâmetros que precisamos por hora, porém é indicado verificar o arquivo padrão para conhecer melhor:

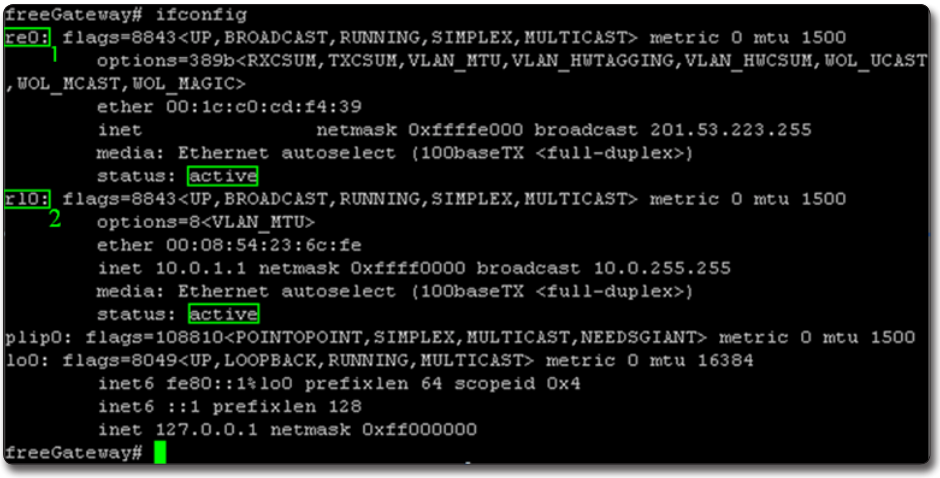


Figura 28. Verificando as placas de rede

```
freeGateway# cat /etc/rc.conf

# -- sysinstall generated deltas -- # Mon Feb 15 01:54:45 2010
# Created: Mon Feb 15 01:54:45 2010
# Enable network daemons for user convenience.
# Please make all changes to this file, not to /etc/defaults/rc.conf.
# This file now contains just the overrides from /etc/defaults/rc.conf.
gateway_enable="YES"
keymap="br275.cp850"
sshd_enable="YES"
hostname="freeGateway"
natd_enable="YES"
natd_interface="re0"
firewall_enable="YES"
firewall_script="/usr/local/etc/rc.d/rc.firewall"
ifconfig_re0="DHCP"
ifconfig_rl0="inet 10.0.1.1 netmask 255.255.0.0"
freeGateway#
```

Figura 29. Arquivo /etc/rc.conf

• **gateway_enable="YES"**: Opção que habilita a função de gateway. Seria o mesmo que utilizar o *syscontrol* com o comando *sysctl net.inet.ip.forwarding=1* que habilita funções em tempo real no kernel;

• **keymap="br275.cp850"**: Podemos especificar aqui o mapa de teclado a ser utilizado. Os mapas conhecidos podem ser encontrados em */usr/share/syscons/keymaps*. O padrão informado é indicado quando usamos teclado conhecido em outros sistemas, como us internacional;

• **sshd_enable="YES"**: Ativando o SSH (Secure Shell, usado para efetuar conexões e executar comandos remotamente);

• **hostname="freeGateway"**: Especificamos aqui o nome do microcomputador. Pode ser escolhido a seu critério;

• **natd_enable="YES"**: Ativando o serviço de NAT (*Network Address Translation*). Conhecido também como *masquerading*, o NAT irá mascarar os endereços dos microcomputadores da rede local de maneira que seja possível a comunicação através do link de internet;

• **natd_interface="re0"**: É preciso especificar em qual interface o NAT deve atuar. Use o *ifconfig* para identificar a interface, pois deve ser a que possui conexão com a rede externa;

• **firewall_enable="YES"**: Precisamos ativar o firewall com a finalidade de realizar o repasse de pacotes entre as interfaces de rede e também para que o NAT funcione corretamente;

• **firewall_script="/usr/local/etc/rc.d/seu arquivo de firewall"**: Apontamos onde está localizado o arquivo contendo as regras para o firewall;

• **ifconfig_re0="dhcp"**: No cenário atual, o IP da rede externa é atribuído via DHCP (*Dynamic Host Configuration Protocol*). Deve-se informar qual é a interface que receberá o IP através do DHCP (para que a placa de rede seja configurada de forma automática/dinâmica é necessário que o pacote *dhcpclient* esteja instalado – no cenário atual com a opção *Standard* selecionada no inicio da instalação ele estará presente);

• **ifconfig_rl0="inet 10.0.1.1 netmask 255.255.0.0"**: No mesmo cenário a outra interface precisa estar configurada com a rede local (interna) com um endereço IP estático, logo adicionamos o endereço IP e sua respectiva máscara manualmente.

As duas regras de firewall serão: uma para limpar as regras existentes, garantindo que as novas serão as regras válidas; e outra que irá realizar o NAT entre as redes.

Agora crie o arquivo dentro do diretório */usr/local/etc/rc.d/*. No nosso exemplo se chamará *rc.firewall*. Depois adicione as seguintes regras, uma por linha:

• **ipfw –f flush:** Limpa as regras existentes;

• **ipfw add 500 divert natd via re0:** Regra que utiliza o *socket divert* para filtrar o tráfego e direcionar os pacotes da interface interna para a interface externa.

A seguir, altere o dono e o grupo do arquivo para *root* com o comando *chown root.root rc.firewall*. Em seguida dê permissão de execução apenas para o usuário *root* (*chmod 700 firewall*). Com isso garantimos que usuários comuns não poderão alterar o conteúdo do arquivo de *firewall* e o utilize com outras finalidades.

Existem duas maneiras de aplicarmos as regras de *firewall*: informar no *rc.conf* (através do parâmetro *firewall_script*) ou do arquivo *rc.local* (este arquivo é lido por último no processo de *boot*, ou seja após iniciar todos o processos de inicialização este arquivo é lido com a finalidade de inicializar serviços extras, como regras de firewall por exemplo).

Existe um detalhe quando utilizamos o *rc.local* para inicialização de serviços, precisamos adicionar ao arquivo de *firewall* a extensão .sh, que o identificará como sendo um *script* (*/usr/local/etc/rc.d/rc.firewall.sh*) para que ao final do processo de *boot* ele possa ser reconhecido e executado. As permissões de execução do script de *firewall* (*firewall.sh*) devem estar

ajustadas com permissão de execução (*chmod 700*).

Lembre-se, escolha apenas um dos métodos, não há efeitos colaterais em usar um ou outro. Para o nosso exemplo utilizaremos a primeira opção (*rc.conf*) (veja a **Figura 30**).

Este script será executado toda vez que o sistema for inicializado e será interpretado pelo */sbin/sh*. Para verificar as regras de *firewall* ativas use *ipfw show* (**Figura 30**).

Mas para que o NAT e o *firewall* funcionem corretamente são necessários mais alguns ajustes. Existem duas maneiras de realizar esses ajustes: ativar os módulos e serviços estaticamente no kernel construindo e criando um novo kernel; ou através das variáveis do *sysctl* (mais informações sobre *sysctl* podem ser obtidas no endereço *http://doc.fug.com.br/handbook/configtuning-sysctl.html*).

Vamos utilizar a opção estática no kernel, pois o módulo necessário para realização do NAT deve ser adicionado ao kernel para que trabalhe em conjunto e sincronia. Duas das principais vantagens de configuração no kernel são a segurança e a estabilidade.

Devemos ajustar o arquivo de configuração do kernel (utilize o editor de texto disponível, o vi) adicionando as opções necessárias, e então construir o novo kernel e instalá-lo (com o comando *make*, que será explicado mais adiante).

Lembrando que devemos ter os pacotes com os fontes instalados. Caso tenha realizado a instalação mínima, os fontes podem ser encontrados no CD de instalação através do *sysinstall*, selecionando as opções *Distributions* > *Kernels* > *GENERIC*. O código fonte do kernel será copiado dentro do diretório */boot*. Como selecionamos no inicio da instalação a opção *Standard*, o código fonte do kernel deverá estar presente.

Com os fontes do kernel disponíveis, precisamos criar um arquivo de configuração para que ao compilar o novo kernel o compilador saiba quais módulos e configurações ele deve manter ou retirar.

Existe um arquivo genérico de configuração do kernel e ele se encontra no diretório */usr/src/sys/i386/conf*, denominado *GENERIC*. Vamos copiá-lo, mantendo sempre o original intacto e editar a cópia, lembrando que devemos estar com permissão de *root* para executar tais atividades.

Esse arquivo contém os parâmetros passados no momento de compilação do kernel. Podemos então ajustá-lo de modo que fique adequado ao nosso hardware e serviços, removendo todos os demais drivers e opções que não sejam necessárias. Um arquivo bem ajustado deixa o kernel mais leve, além de ajudar com o tempo de construção/compilação do kernel e de inicialização após instalado, uma vez que há menos código/módulos.

Então adicionaremos os parâmetros necessários para o funcionamento como gateway. No nosso exemplo copiaremos o arquivo GENERIC (este é apenas uma arquivo modelo) com outro nome (KERNCUSTOM). Teremos então dois arquivos, o modelo base (GENERIC) e o outro ao qual iremos modificar (KERNCUSTOM). Para realizar a cópia faça *#cp –p /usr/src/sysi386/conf/GENERIC /usr/src/sysi386/conf/KERNCUSTOM*.

Manter o mesmo nome do arquivo de configuração e do próprio kernel (ident) ajuda a manter a organização. Vale ressaltar que não há restrições quanto ao nome dado (veja **Figura 31**).

Então adicionamos os seguintes parâmetros (as linhas que iniciam com # são comentários):

- *ident KERNCUSTOM* – Nome de identificação do kernel. Não há relação com o nome do arquivo, é apenas uma boa prática para identificar e/ou versionar e organizar as possíveis variações de kernel;
- *options IPFIREWALL* – Habilitando o módulo do IPFW (*firewall*);
- *options IPFIREWALL_FORWARD* – Habilita o repasse de pacotes entre interfaces de rede interna e externa;
- *options IPFIREWALL_DEFAULT_TO_ACCEPT* – Comportamento padrão para o *firewall*, definido como aberto (caso essa linha seja omitida o padrão será fechado, portanto as regras no arquivo de *rc.firewall* deverão ser alteradas);
- *options IPFIREWALL_VERBOSE* – Habilitando o log do firewall no *syslogd*;
- *options IPFIREWALL_VERBOSE_LIMIT=1000* – Limite do log em Mb;
- *options IPDIVERT* – Habilitando o uso de *socket divert* por padrão na porta 8668.

```
freeGateway# pwd
/usr/local/etc/rc.d
freeGateway# cat /usr/local/etc/rc.d/rc.firewall
#!/bin/sh

/sbin/ipfw -f flush
/sbin/ipfw add 500 divert natd via re0

freeGateway# chmod 700 rc.firewall
freeGateway# ls -la
total 6
drwxr-xr-x   2 root  wheel  512 Feb 24 19:48 .
drwxr-xr-x  15 root  wheel  512 Feb 18 23:57 ..
-rwx------   1 root  wheel   71 Feb 24 19:42 rc.firewall
freeGateway# ipfw show
00500   574  154186 divert 8668 ip from any to any via re0
65535 21258 9781132 allow ip from any to any
freeGateway#
```

Figura 30. Arquivo /usr/local/rc.d/rc.firewall

```
freeGateway# cd /usr/src/sys/i386/conf/
freeGateway# ls
.cvsignore      DEFAULTS        GENERIC.hints   Makefile        PAE
CUSTOM          GENERIC         MAC             NOTES           XBOX
freeGateway# cp -p /usr/src/sys/i386/conf/GENERIC /usr/src/sys/i386/conf/KERNCUS
TOM
freeGateway# ls
.cvsignore      GENERIC         MAC             PAE
CUSTOM          GENERIC.hints   Makefile        XBOX
DEFAULTS        KERNCUSTOM      NOTES
freeGateway# vi /usr/src/sys/i386/conf/KERNCUSTOM
```

Figura 31. Copiando e editando o arquivo de configuração do kernel

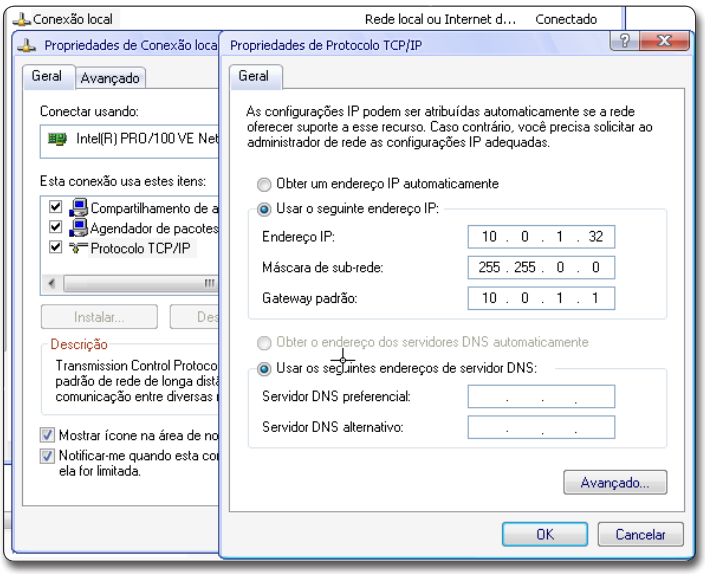


Figura 32. Configurando o Windows XP para usar o gateway



Figura 33. Teste de conexão, bem sucedido

Editado o arquivo KERNCUSTOM, devemos então retornar ao diretório */usr/src* para que consigamos executar os comandos para construção e instalação do novo kernel.

A construção do kernel usando o arquivo de configuração que foi editado, a depender do microcomputador, pode ser demorada. Para isso, execute:

```
# make buildkernel KERNCONF=KERNCUSTOM
```

Se nenhum erro foi encontrado, instale o novo kernel usando o make:

```
#make installkernel KERNCONF=KERNCUSTOM
```

Se tudo ocorreu bem, teremos então o FreeBSD pronto para atuar como gateway. Assim, basta reiniciar o computador e ligar suas conexões de rede (interna e externa) nas interfaces corretas. Lembre-se das configurações do *rc.conf*, devemos nos atentar em ligar as conexões de rede às placas corretas.

Configurando clientes Windows

Para configurar o gateway em uma máquina com Windows XP siga os passos: *Iniciar > Painel de controle > Conexões de rede > Conexão local > Propriedades > (Na lista) Protocolo TCP/IP > Propriedade*. Adicione o computador na mesma rede que o gateway e configure a opção gateway com o endereço do nosso servidor (**Figura 32**).

O DNS foi omitido por segurança, mas ele deve ser adicionado. Use o DNS (*Domain Name System*) do seu provedor ou algum de domínio público.

Para testar a conexão, tente navegar ou utilize o ping. Para isso, acesse: *Iniciar > Executar > cmd* e no prompt de comando digite: *ping www.devmedia.com.br* (como na **Figura 33**). Para que tudo esteja funcionando corretamente devemos ter 100% de pacotes enviados e 100% recebidos.

Temos então um sistema FreeBSD robusto, estável e seguro atuando como gateway da rede. Com a atual configuração, também poderíamos adicionar um Proxy, como Squid, controlar o tráfego, entre outros serviços, o que pode ser apresentado em um próximo artigo.

Conclusão

O FreeBSD é um sistema de baixíssimo custo. Ele oferece estabilidade e segurança, além de funcionar com as mais diversas arquiteturas de computadores, desde os de pequeno até os de grande porte.

Neste artigo, fizemos do FreeBSD o nosso gateway, permitindo que outras máquinas da rede local utilizem a internet a partir de um só link de internet. Concentrado todo o tráfego em um único ponto, nosso gateway torna mais fácil o controle e gerenciamento do tráfego através da adição de novos serviços, redirecionamentos de portas através de regras de *firewall*, tudo isso sem falar no aumento da segurança, pois alguém mal intencionado teria que passar pelo nosso gateway até chegar a outros pontos.

Podemos a partir de agora acrescentar ao FreeBSD instalado novos recursos, como: escrever regras de *firewall* mais adequadas dando permissão apenas às portas e serviços necessários; um proxy para restringir e controlar o acesso à internet; entre outras configurações que possibilitarão um melhor uso e administração da rede como um todo.



Rogério Tomassoni de A. Junior
tomassonijr@gmail.com
Bacharel em análise de sistemas (FAI - Faculdade de Administração e Informática), Certificado LPIC101 000160469, Certificado ITIL Foundation, atualmente exercendo a função de administrador de redes, amante do Unix e Linux e membro do site viva o Linux.

Versão do FreeBSD utilizada
ftp://ftp.freebsd.org/pub/FreeBSD/releases/i386/ISO-IMAGES/7.2/7.2-RELEASE-i386-disc1.iso

Sistema de créditos DevMedia

Conteúdo **completo** ao seu alcance

Com o sistema de créditos DevMedia você pode comprar individualmente mais de 4.000 vídeo aulas, 8000 artigos, além de palestras e vídeos com formações completas.

TODAS as vídeo aulas que você sempre quis ver a partir de

R$ 0,99



ECONOMIA: você não precisa assinar 8 produtos diferentes para ter acesso ao conteúdo completo do site!

Saiba mais sobre o Sistema de Créditos!

www.devmedia.com.br/creditos

DevMedia

Configurando o Servidor DHCP3

DHCP no Ubuntu como ferramenta de administração de redes

Racy Rassilan

Resumo DevMan

De que se trata o artigo:

O presente artigo demonstra como funciona o DHCP, e como implementá-lo para garantir uma melhor administração de redes Windows e Linux em uma organização.

Para que serve:

Serve principalmente para organizações em que a rede sempre está sofrendo alterações, como: aumento de computadores, troca de servidores e troca de endereços IPs em servidores.

Em que situação o tema é útil:

Existem várias situações, dentre elas: quando surge a necessidade de adicionar novas máquinas à rede, não sendo necessário definir as configurações manualmente ponto a ponto. Dessa forma, quando for necessário mudar o endereço IP do servidor, por exemplo, isso será repassado automaticamente às máquinas da rede independente do Sistema Operacional utilizado.

O presente artigo tem por objetivo ensinar a configurar um servidor DHCP3 (*Dynamic Host Configuration Protocol* ou Protocolo de Configuração Dinâmica de Endereço de Rede) no Sistema Operacional Linux Ubuntu, tornando a administração de redes Linux e Windows mais eficiente. Isso só será possível depois que o leitor compreender o que é DHCP e como esta maravilhosa ferramenta funciona, demonstrando que sua utilização torna a administração de redes um trabalho prazeroso e transparente ao usuário independente do SO utilizado pelo mesmo.

DHCP

Antes da existência do DHCP, ser um administrador de uma grande rede de computadores era um trabalho árduo e muitas vezes ineficiente devido a toda e qualquer alteração no servidor como IP, Maskara, Gateway e DNS. Isto acabava causando a necessidade dessas configurações serem repassadas às máquinas clientes manualmente ou mesmo quando surgia uma nova máquina na rede. O que tornava o processo demorado e susceptível a diversos erros, como IPs duplicados na rede e, consequentemente, perda de comunicação entre as máquinas e a internet.

Com o surgimento do DHCP em outubro de 1993, veio a grande solução a este árduo processo de configuração, e por ser tão eficiente sobrevive até hoje como a melhor solução de Configuração Dinâmica de Endereços de Rede.

Um servidor DHCP é um serviço de protocolo de redes existente no Gnu/Linux utilizado para gerenciar as faixas de IPs que podem existir na rede, ou seja, é um mecanismo para gerenciar o protocolo TCP/IP liberando IPs válidos para as estações de uma rede. Isto torna possível ao administrador da rede de uma empresa ou instituição, que geralmente tem um grande número de máquinas, não necessite configurá-las manualmente digitando o IP, a Maskara, o Gateway e o DNS tomando todo seu tempo e causando muita dor de cabeça sempre que surgirem novas máquinas.

Funcionamento do DHCP

O protocolo DHCP funciona de uma forma muito inteligente. Inicialmente ele estará rodando no servidor verificando constantemente se há alguma máquina na rede que solicitou seu serviço via *Broadcast* (mensagem enviada através de UDP a todas as máquinas na rede). Se houver, então o servidor DHCP captura esta mensagem e responsabiliza-se por liberar quatro informações: IP, Maskara, Gateway e DNS, que foram explicitamente informados no seu arquivo de configuração. Onde o IP liberado para a maquina solicitante será randômico, de acordo com a faixa definida



Nota do DevMan

Transparente: Geralmente em todo o texto na área da computação ele está presente e muitos o confundem. Este termo não significa transparência de ver através de algo como um vidro, mas sim o fato de esconder as definições e funcionamento técnico aos usuários comuns, como é o caso do DHCP. O usuário não precisa saber qual o IP de sua máquina e muito menos como esse endereço foi obtido.



Nota do DevMan

Maskara: Conhecido como sub-net, ou máscara de sub-rede, ao contrário do IP, que é formado por valores entre 0 a 255, a maskara é formada unicamente por valores 0 e o valor 255 servindo como indicativo de qual classe um determinado IP de rede pertence. O IP classe A terá a maskara 255.0.0.0; o IP classe B 255.255.0.0; o IP classe C, que é o mais utilizado, 255.255.255.0.

DNS: Domain Name System (Sistema de Resolução de Nomes) é através deste sistema que quando se digita um endereço no navegador ele consegue acessá-lo. Exemplo: quando se digita o endereço www.devmedia.com.br no navegador é necessário que um servidor DNS leia essa String e retorne ao navegador da máquina solicitante o IP da máquina na Internet que contém os arquivos que compõem a estrutura do site relacionado a este endereço, no caso o IP 74.53.3.20. Experimente digitar este IP no navegador, irá carregar o site da DevMedia.

no seu código de configuração, e se este ainda não estiver sendo usado por alguma máquina on-line na rede.

Depois de toda essa verificação por parte do servidor DHCP, será finalmente liberado um pacote do servidor para a máquina solicitante que contém uma configuração válida para que o computador da rede possa utilizar. Finalmente, a máquina solicitante conseguirá agrupar-se na rede e navegar na internet sem causar nenhum conflito com as que já estavam on-line.

O administrador pode definir que o DHCP funcionará de uma das quatro formas seguintes: *Automática*, *Dinâmica*, *Manual* ou *todas juntas*, através da edição do arquivo *dhcpd.conf*.

• **Automática:** Nesta configuração, uma determinada quantidade de IPs é definida para utilização na rede. Por exemplo, imagine que no código foi especificado que o servidor iria fornecer IPs no intervalo de 192.168.10.1 a 192.168.10.50. Dessa forma, toda hora que uma máquina cliente fizer uma solicitação, será direcionado a esta máquina um IP que ainda não esteja sendo utilizado dentro desta faixa;

• **Dinâmica:** Muito parecido com o Automático. Entretanto, nesta configuração especifica-se o tempo que a máquina poderá ficar conectada na rede. Isso é útil quando o administrador deseja definir um tempo limite de conexão. Mas essa configuração geralmente não é utilizada, pois os funcionários utilizam as maquinas de uma corporação constantemente e seria estressante toda hora chegar alguém falando que a maquina saiu da rede;

• **Manual:** Com essa configuração torna-se possível definir que uma máquina X sempre tenha o mesmo IP, independente do lugar e horário que ela se encontre conectada na rede. Isso se torna possível devido à utilização do endereço *MAC* da placa de rede no código de configuração do DHCP, que garantirá sempre um IP específico para este endereço *MAC*, conhecido amplamente como endereço físico da placa de rede.

A principal vantagem de ter um servidor DHCP gerenciando a rede é que bastando conectar um computador ou notebook em um terminal gerenciado pelo servidor já será o suficiente para termos acesso à internet. Isto acontece por que o DHCP reconhece o computador e libera as configurações de IP, Maskara, Gateway e DNS automaticamente via rede. Essa vantagem é muito útil em duas vertentes, uma quando se muda constantemente o computador de lugar no caso do notebook, e outra quando se faz necessário colocar uma nova máquina na rede.

TCP – Transmission Control Protocol	UDP – User Datagram Protocol
É orientado à conexão, tornando-se muito confiável uma vez que se utiliza de uma sessão para realizar a conexão entre os hosts e posteriormente iniciar a comunicação.	Não é orientado à conexão, tornando-se um protocolo não confiável, pois o UDP ao contrário do TCP já inicia a comunicação sem saber se a máquina destino está on-line.
Fragmenta a mensagem em partes numeradas (pacotes), garantindo a sequência dos pacotes.	Não garante a sequência dos pacotes e não confirma suas entregas.
Implementa controle de erros, como: o reenvio de pacotes caso algum não chegue ao destino ou chegue corrompido.	Não implementa controle de erros. As aplicações que utilizam UDP é que ficam responsáveis por tentar tratar a confiabilidade do transporte de dados.
É mais lento, utiliza alta sobrecarga podendo oferecer suporte à conexão apenas ponto a ponto, ou seja, máquina a máquina.	É mais rápido, utiliza baixa sobrecarga podendo oferecer suporte à conexão ponto a ponto e de ponto a vários pontos.

Tabela 1. Comparação entre o protocolo TCP e o UDP

Protocolos UDP & TCP/IP

Como podemos notar, o DHCP trabalha em função destes protocolos de rede: o UDP para mensagens via *Broadcast*; e o TCP/IP para comunicação e troca de arquivos. Então nada mais justo do que compreender um pouco sobre estes três protocolos que são vitais para o funcionamento das redes de computadores:

• **UDP:** ou *User Datagram Protocol*, é um padrão de protocolo de rede definido pela RFC 768. Este padrão é utilizado por alguns programas para transportar dados de forma mais rápida entre dois hosts (computadores) do que o TCP. O UDP consegue essa velocidade por não verificar se os dados foram entregues sem erros e se chegaram ao destinatário;

• **TCP:** ou *Transmission Control Protocol*, é o principal protocolo de rede devido a sua alta confiabilidade. Este protocolo executa importantes funções de verificação de integridade dos dados, o que garante sua confiabilidade, porém, não consegue ser tão rápido quanto o UDP. Por outro lado, também garante que os dados cheguem ao destinatário de forma incorruptível;

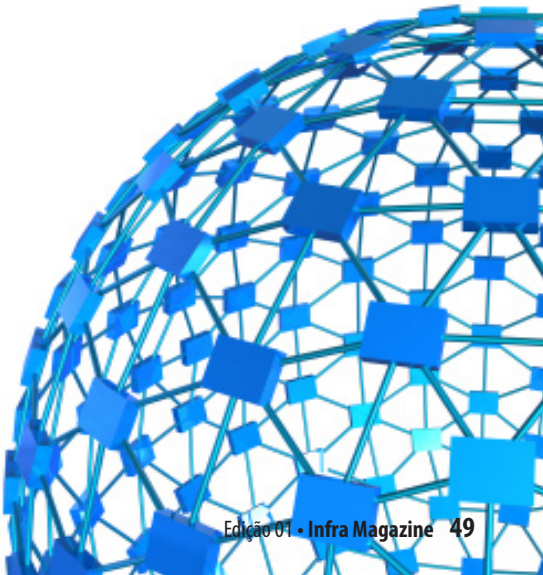
• **IP:** ou *Internet Protocol*, é o protocolo conhecido como protocolo de endereçamento. O protocolo IP tem a finalidade de endereçar as mensagens (dados) que trafegam na rede por este ser o protocolo responsável em definir o endereço IP único de cada maquina na rede.

Comparação entre UDP e TCP

É sempre interessante verificar como os protocolos se comportam perante a necessidade de estabelecer uma conexão para entrega de dados, pois o DHCP só funciona pela existência deles.

A **Tabela 1** apresenta um comparativo do funcionamento dos dois protocolos.

A partir de agora fica mais fácil entender como funciona o processo de comunicação entre a máquina solicitante e o servidor DHCP. Quando uma máquina está na rede e precisa de configurações válidas, ela irá enviar uma solicitação de configurações válidas via *Broadcast* através do protocolo UDP por ser mais rápido. E o servidor DHCP irá liberar para a máquina um IP para esta conseguir utilizar o protocolo TCP para suas tarefas que devem ser seguras, como tráfego de dados incorruptíveis.



Instalando o Servidor DHCP

Agora que já foi transmitido o conhecimento de como o DHCP funciona, é hora de colocar o mesmo na prática instalando o DHCP.

O processo de instalação no Ubuntu é muito simples, basta entrar no *shell* como *root*, digitar o seguinte comando: *apt-get install dhcp3-server* e esperar pela realização do download e instalação do pacote.

Configurando o Servidor DHCP

Depois de instalado o pacote *dhcp3-server*, o serviço estará pronto para ser configurado. **Basicamente tudo que envolve configuração no Linux, seja qual distribuição for, as configurações são feitas por intermédio da edição de arquivos** e o principal arquivo que deverá ser editado para configurar o servidor DHCP é o *dhcpd.conf*, encontrado no diretório */etc/dhcp3/*.

Para tal, serão realizados três passos que devem ser seguidos à risca no servidor para que tudo funcione corretamente.

Primeiro Passo: Estando como *root* no *Shell*, abra o arquivo com o *vim* da seguinte forma:

```
vim dhcpd.conf
```

Ele deve mostrar um conteúdo similar à **Listagem 1**.

Listagem 1. Código fonte padrão do DHCP quando ele é instalado.

```
#
# Sample configuration file for ISC dhcpd for Debian
#
# Attention: If /etc/ltsp/dhcpd.conf exists, that will be used as
# configuration file instead of this file.
#
# $Id: dhcpd.conf,v 1.1.1.1 2002/05/21 00:07:44 peloy Exp $
#
# The ddns-updates-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
# have support for DDNS.)
ddns-update-style none;
# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;
default-lease-time 600;
max-lease-time 7200;
# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
#authoritative;
# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;
# No service will be given on this subnet, but declaring it helps the
# DHCP server to understand the network topology.
#subnet 10.152.187.0 netmask 255.255.255.0 {
#}
# This is a very basic subnet declaration.
#subnet 10.254.239.0 netmask 255.255.255.224 {
#  range 10.254.239.10 10.254.239.20;
#  option routers rtr-239-0-1.example.org, rtr-239-0-2.example.org;
#}
# This declaration allows BOOTP clients to get dynamic addresses,
# which we don't really recommend.
#subnet 10.254.239.32 netmask 255.255.255.224 {
#  range dynamic-bootp 10.254.239.40 10.254.239.60;
#  option broadcast-address 10.254.239.31;
#  option routers rtr-239-32-1.example.org;
#}
# A slightly different configuration for an internal subnet.
#subnet 10.5.5.0 netmask 255.255.255.224 {
#  range 10.5.5.26 10.5.5.30;
#  option domain-name-servers ns1.internal.example.org;
#  option domain-name "internal.example.org";
#  option routers 10.5.5.1;
#  option broadcast-address 10.5.5.31;
#  default-lease-time 600;
#  max-lease-time 7200;
#}
# Hosts which require special configuration options can be listed in
# host statements.   If no address is specified, the address will be
# allocated dynamically (if possible), but the host-specific information
# will still come from the host declaration.
#host passacaglia {
#  hardware ethernet 0:0:c0:5d:bd:95;
#  filename "vmunix.passacaglia";
#  server-name "toccata.fugue.com";
#}
# Fixed IP addresses can also be specified for hosts.   These addresses
# should not also be listed as being available for dynamic assignment.
# Hosts for which fixed IP addresses have been specified can boot using
# BOOTP or DHCP.   Hosts for which no fixed address is specified can only
# be booted with DHCP, unless there is an address range on the subnet
# to which a BOOTP client is connected which has the dynamic-bootp flag
# set.
#host fantasia {
#  hardware ethernet 08:00:07:26:c0:a5;
#  fixed-address fantasia.fugue.com;
#}
# You can declare a class of clients and then do address allocation
# based on that.   The example below shows a case where all clients
# in a certain class get addresses on the 10.17.224/24 subnet, and all
# other clients get addresses on the 10.0.29/24 subnet.
#class "foo" {
#  match if substring (option vendor-class-identifier, 0, 4) = "SUNW";
#}
#shared-network 224-29 {
#  subnet 10.17.224.0 netmask 255.255.255.0 {
#    option routers rtr-224.example.org;
#  }
#  subnet 10.0.29.0 netmask 255.255.255.0 {
#    option routers rtr-29.example.org;
#  }
#  pool {
#    allow members of "foo";
#    range 10.17.224.10 10.17.224.250;
#  }
#  pool {
#    deny members of "foo";
#    range 10.0.29.10 10.0.29.230;
#  }
#}
```

Note que a maior parte do arquivo fonte da **Listagem 1** é constituída por comentários ("#") que definem especificações de uso, que podem ser pesquisadas afim de um aprofundamento em um estudo futuro por sua parte.

Para simplificar as coisas, exclua todas as linhas. Dessa forma o código ficará mais limpo, contendo apenas suas especificações, tornando o processo de alteração mais rápido e simples.

Segundo Passo: Adicione as linhas como informado na **Listagem 2**.

Ao término da edição do arquivo salve-o.

Listagem 2. Código fonte para configuração do DHCP.

```
option domain-name "ServidorDHCP";
ddns-update-style none;
authoritative;
subnet 192.168.10.0 netmask 255.255.255.0 {

  range 192.168.10.1 192.168.10.200;
  option domain-name-servers 200.165.132.148, 200.165.132.155;
  option routers 192.168.10.170;
  option broadcast-address 192.168.10.255;
  default-lease-time 600;
  max-lease-time 7200;

}

host micro1{

  hardware ethernet 00:0d:87:ee:dd:a7;
  fixed-address 192.168.10.100;

}
```

Listagem 3. Código fonte para configuração do arquivo interfaces.

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
```

Explicação

As três primeiras linhas devem existir por padrão para o funcionamento do DHCP.

A linha *subnet 192.168.10.0 netmask 255.255.255.0* serve como a criação de uma classe, que neste caso foi denominada *subnet 192.168.10.0 netmask 255.255.255.0{...}*. Dentro dessa classe serão realizadas as principais especificações, que são:

- *range 192.168.10.1 192.168.10.200* – Define o intervalo de IPs liberados para as máquinas da rede pela classe. Podendo, por exemplo, liberar outra faixa como *10.10.10.1 10.10.10.158*, entretanto para isso será necessário criar uma classe *10.10.10.0*, caso contrário não irá funcionar;
- *option domain-name-servers* – Define quais são os servidores de resolução de nomes (DNS). Sempre é bom utilizar estes: *200.165.132.148* e *200.165.132.155*, uma vez que os mesmos são da própria Oi/Telemar;
- *option routers 192.168.10.170* – Define qual o IP do servidor que irá liberar para a rede os IPs. Geralmente este IP é o da segunda placa de rede *eth1*;
- *option broadcast-address 192.168.10.255* – Define qual IP será utilizado para enviar uma mensagem para todas as máquinas na rede.

A linhas *default-lease-time 600* e *max-lease-time 7200* definem respectivamente os tempos mínimo e máximo que o servidor DHCP irá liberar um IP válido para uma máquina. Após o esgotamento do tempo mínimo, ele oferece (de forma automática) um pouco mais de tempo caso o usuário solicite ficar mais tempo logado na máquina. Caso não queira passar pelo transtorno de mais de 10 pessoas te procurando dizendo que não está conseguindo acessar a internet remova essas duas linhas.

Note que foi realizada uma configuração mista, ou seja, dentro do *subnet 192.168.10.0 netmask 255.255.255.0 { ... }* foram realizadas especificações da distribuição geral dos IPs pela rede. E depois foi definido um IP fixo para uma determinada máquina na rede através de associação do MAC da placa de rede ao IP escolhido nesta sessão:

```
host micro1{

  hardware ethernet 00:0d:87:ee:dd:a7;
  fixed-address 192.168.10.100;

}
```

Geralmente o mais utilizado é simplesmente configurar a faixa de IP para a rede e pronto, como foi feito na classe *subnet 192.168.10.0 netmask 255.255.255.0{...}*. Mas também demonstramos que é possível especificar via MAC qual IP você deseja que uma determinada máquina tenha, conforme vimos na classe *host micro1{...}*. Isso possibilita fazer a configuração mista, ou seja, automática e via MAC ao mesmo tempo.

Terceiro Passo: Agora é importante reiniciar o serviço. Para isso, acesse como *root* o diretório */etc/init.d/* e digite os seguintes comandos:

```
./dhcp3-server stop
```

Depois:

```
./dhcp3-server start
```

Sempre que se realiza alguma configuração no SO Linux, é extremamente recomendável parar e iniciar o serviço correspondente, pois só assim terá certeza que as configurações entraram e estão em vigor.

Configurando os Clientes DHCP no Linux

Esta etapa é muito simples, sendo necessária a realização de dois passos. Basta ir até as máquinas da rede e editar o arquivo *interfaces*, encontrado em */etc/network/*.

Primeiro Passo: Editar o arquivo *interfaces* utilizando o *vim* ou *vi*.

- Conteúdo a ser colocado dentro do arquivo de configuração *interfaces*:
- *auto lo*: Indica que as placas de rede deverão ser ativadas automaticamente durante o boot;
- *iface lo inet loopback*: Utilizado para co-

munication entre aplicações do sistema e demais aplicativos;

• *auto eht0*: Este ponto define quais das placas deverá ser ativada e configurada de acordo com a linha abaixo;

• *iface eht0 inet dhcp*: Define que a configuração da rede será via DHCP, onde esta máquina solicitará ao servidor os quatro padrões de configuração, que são: endereço IP, Maskara, Gateway e DNS.



Figura 1. Ícone de configuração de rede denominado "conexão local"

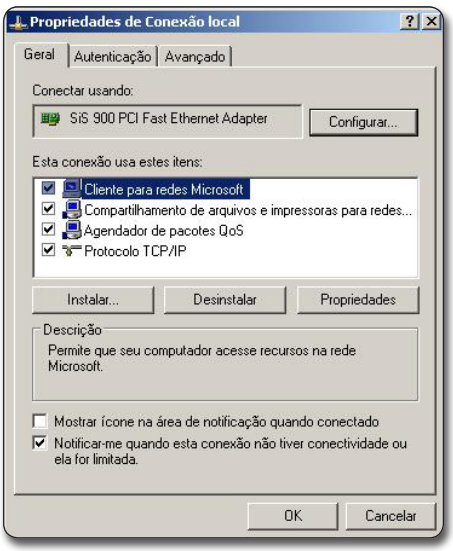


Figura 2. Janela de Propriedades de Conexão local

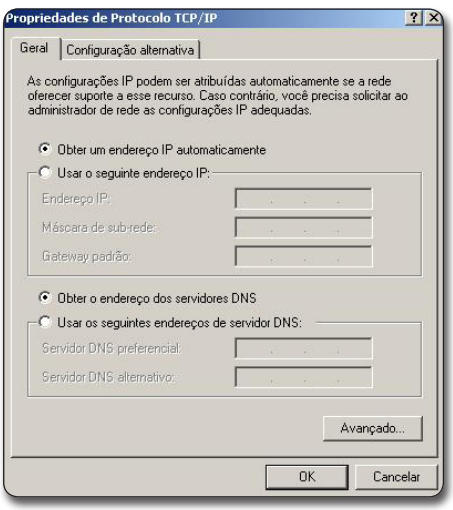


Figura 3. Janela de Propriedades de Protocolo TCP/IP

O conteúdo do arquivo ficará como o mostrado na **Listagem 3**.

Feito isso basta salvar o arquivo.

Segundo Passo: Agora que o arquivo está configurado, será necessário reiniciar o serviço de rede. Para isso acesse o diretório */etc/init.d* e digite os seguintes comandos:

```
./networking stop
```

Depois:

```
./networking start
```

Com isso esta máquina conseguirá adquirir todas as configurações do servidor que foi configurado anteriormente.

Configurando os Clientes DHCP no Windows XP

Configurar computadores que tenham o SO Windows a adquirirem suas configurações de rede via DHCP assim como no Linux também é um processo muito simples. Estando em uma máquina Windows siga os seguintes passos:

Primeiro Passo: Acesse Iniciar > Configurações > Conexões de Rede. Então será mostrada uma janela com um ícone chamado "Conexão local", como o mostrado na **Figura 1**.

Agora clique com o botão direito sobre este ícone. Surgirá um menu flutuante. Neste menu selecione "Propriedades", sendo exibida uma janela de configurações da rede, como a da **Figura 2**.

Segundo Passo: Localize a opção "Protocolo TCP/IP", dê um clique sobre ela e posteriormente clique no botão "Propriedades". Logo em seguida será aberta outra janela, semelhante à **Figura 3**.

Terceiro Passo: Estando com a janela da **Figura 3** aberta, basta marcar as opções como mostrado na figura, que são: "Obter um endereço IP automaticamente" e "Obter o endereço dos servidores DNS". Clique no botão "Ok"; será retornada a janela da **Figura 2**, nela basta clicar novamente em "Ok" e pronto, o computador está configurado para adquirir todas as configurações de rede via DHCP.

Para enriquecer o conteúdo desta matéria, o vídeo a seguir expõe os conceitos e demonstra na prática o que é necessário para configurar o servidor DHCP.

Conclusão

Através deste artigo ficou claro que configurar um servidor DHCP é muito simples, e possibilita ao administrador de redes administrá-la com mais facilidade, pois quando for necessário mudar o IP do servidor ou o gateway, não precisaremos informar essas mudanças máquina a máquina. Isso permitirá ao administrador de redes uma administração que não incomode e/ou permita que os usuários tenham acesso direto a estes processos, tornando sua administração transparente ao usuário.

A computação é uma ciência muito dinâmica e esse dinamismo torna os processos organizacionais mais eficientes. Portanto, é necessário que todos os profissionais de TI sempre estejam preparados para utilizar seus subsídios, garantindo não só o mero funcionamento de computadores em suas organizações, mas sim utilizar da melhor solução possível em termos de gestão de TI.

Compreender como funcionam as redes de computadores torna-se vital, uma vez que não se pode implementar uma solução para algo sem conhecer seu funcionamento, e muito menos se nem souber da existência de mecanismos computacionais que auxiliam sua administração, como é o caso do DHCP.

Agora você tem conhecimento suficiente para montar um servidor DHCP distribuindo as faixas de IPs ou especificando por MAC qual IP uma determinada máquina deve ter. Isto tornará a administração um processo mais fácil e eficaz, evitando transtornos como fazer tudo ponto a ponto na rede. Então administrador, mão na massa que essa tecnologia vai lhe garantir uma administração descomplicada e estará colocando-o no quadro de profissionais que usam o melhor em gestão de TI.



Racy Rassilan
racy@pos.facom.ufu.br
É graduado em Ciência da Computação pela Universidade Presidente Antonio Carlos - UNIPAC de Teófilo Otoni - MG e Mestrando em Banco de Dados pela Universidade Federal de Uberlândia - UFU - MG. Atualmente é desenvolvedor de software e administrador de servidores Linux. Atuando principalmente com desenvolvimento de aplicações Desktop e Web com as linguagens JAVA e PHP.

Artigo sobre configuração do serviço DHCP feito pela UNICAMP.
http://www.ccuec.unicamp.br/revista/infotec/linux/linux19-1.html

Artigo do site vivaolinux que aborda uma maneira fácil e rápida de configurar o servidor DHCP no Linux Conectiva.
http://www.vivaolinux.com.br/artigo/Servidor-DHCP-rapido-e-facil-para-Conectiva-Linux

Tutorial que explica como deve ser o conteúdo do arquivo dhcpd.conf no Linux Ubuntu.
http://thiagodfreitas.wordpress.com/2007/11/13/arquivo-de-configuracao-etcdhcp3dhcpdconf-no-ubuntu/

Artigo do site vivaolinux que mostra como implementar um servidor DHCP.
http://www.vivaolinux.com.br/artigo/Implementando-um-servidor-DHCP/

Tutorial feito pelo MIT explicando como configurar um servidor DHCP no Linux Red Hat Enterprise.
http://web.mit.edu/rhel-doc/3/rhel-sag-pt_br-3/s1-dhcp-configuring-server.html

Blog com tutorial que explica como configurar um servidor DHCP no Linux Debian e Ubuntu utilizando DHCP3.
http://www.xjulio.info/blog/configurando-um-servidor-de-dhcp-no-ubuntu-e-debian-com-o-dhcp3-server

Site que visa responder algumas perguntas básicas sobre DHCP, como: O que é DHCP? Para que serve? Como funciona? etc.
http://www.rnp.br/newsgen/9705/n1-2.html

Site de Julio Battisti com um excelente artigo explicando sobre os protocolos de redes.
http://www.juliobattisti.com.br/artigos/windows/tcpip_p11.asp

Site Guia do Hardware com um artigo que visa explicar as máscaras de sub-rede.
http://www.gdhpress.com.br/redeseservidores/leia/index.php?p=cap3-2

FÓRUM
DEVMEDIA
O lugar perfeito para você ficar por dentro de tudo o que acontece nas tecnologias do mercado atual



No novo Fórum da DevMedia você vai encontrar canais específicos de Delphi, ASP.NET, Java, Banco de Dados e Engenharia de Software; além de ter contato com profissionais qualificados da área para troca de informações, sugestões e muito mais.

ACESSE AGORA
www.devmedia.com.br/forum

DevMedia

Virtualização com o Hyper-V

Conheça a arquitetura e como configurar

Jorge Barata

O conceito de virtualização não é novo, ele foi introduzido pela IBM nos mainframes na década de 60, principalmente porque havia a necessidade de redução de custos sem perder serviços, o que só foi possível com a consolidação de várias máquinas em uma única. Atualmente, devido à grande procura de equipamentos, aplicativos e sistemas operacionais, que exigem cada vez mais máquinas poderosas, as empresas começaram a considerar a virtualização como a solução para seus problemas. Virtualização é um método que permite ao usuário de computador instalar e usar simultaneamente mais de um sistema operacional na mesma máquina. O objetivo central dessa tecnologia é justamente gerar mais produtividade ao ambiente de TI (Tecnologia da Informação) das empresas, com cada vez menos gastos e investimentos em manutenção e equipamentos.

Para isso já existem softwares que fazem esse gerenciamento e podem ser aplicados de acordo com as necessidades da empresa. Nesse contexto, o Hyper-V foi lançado com suporte a vários sistemas operacionais do mercado. Ele é um software que permite você executar e virtualizar vários sistemas operacionais na mesma máquina, cada uma se comportando como se fossem independentes. Os ganhos com virtualização vão desde a otimização dos recursos de hardware e o maior nível de disponibilidade para as aplicações até a redução de gastos obtida com a consolidação do parque de servidores.

Neste sentido, este artigo abordará o conceito de virtualização, os principais softwares e, por fim, apresentaremos uma introdução sobre virtualização com o Hyper-V.

Resumo DevMan

De que se trata o artigo:

Este artigo abordará o conceito de virtualização, os principais softwares e, por fim, apresentaremos uma introdução sobre virtualização com o Hyper-V.

Para que serve:

Virtualização é um método que permite ao usuário instalar e usar simultaneamente mais de um sistema operacional na mesma máquina.

Em que situação o tema é útil:

Empresas que tenham interesse em gerar mais produtividade ao ambiente de TI com cada vez menos gastos e investimentos em manutenção e equipamentos.

Conceitos de virtualização

Pode-se definir virtualização como: "Emular, simular, mascarar ambientes isolados, capazes de rodar diferentes sistemas operacionais dentro de uma mesma máquina, utilizando-se ao máximo a capacidade do hardware, que por muitas vezes fica ociosa em determinados períodos. Esse aproveitamento é maior devido à possibilidade de fornecer ambientes de execução independentes a diferentes usuários em um mesmo hardware.".

Seguem abaixo outras definições importantes sobre virtualização:

- **Hypervisor:** trata-se de uma plataforma de virtualização que permite executar vários sistemas operacionais em um único computador físico chamado de host. A principal função do Hypervisor está na execução de ambientes isolados de cada máquina virtual e no gerenciamento do acesso entre os sistemas operacionais convidados (guests) que estão executando em máquinas virtuais e os recursos de hardware do computador (host);
- **Virtualização Completa (Full Virtualization):** É quando uma máquina virtual simula todo o hardware, permitindo que um sistema operacional guest seja executado de modo isolado;
- **Para-virtualização (Para-virtualization / Enlightenment):** Um sistema operacional modificado sendo executado em um hypervisor. Na para-virtualização, o SO Guest e o Hypervisor colaboram para obter o máximo em desempenho;
- **Virtualização de Aplicação:** Um método para mascarar e executar aplicações em um SO ou vindo de um SO, tanto local quanto via rede;



• **Virtualização de Apresentação:** Neste modelo de virtualização apenas a camada de apresentação de um aplicativo é executada na máquina cliente. O processamento e uso de memória ficam no servidor que estiver provendo esta modalidade de virtualização;
• **Hosted Virtualization:** Alguns autores conceituam esta técnica como uma virtualização baseada no Tipo-2 de Hypervisores, em que um sistema operacional guest é executado sobre um sistema operacional host que intermedia toda a comunicação da máquina virtual com o hypervisor;
• **Hardware Virtualization Assistance:** Melhoramentos no processador que suportam guests não-modificados (full virtualization); Intel-VT / AMD-V.

A experiência nos mostrou que a virtualização em larga escala pode oferecer benefícios significativos. Mas também aprendemos que, ao mesmo tempo, a virtualização também traz uma série de desafios – como o gerenciamento de "ilhas" de ativos físicos e ativos virtualizados sem gerenciamento. Por isso, é muito comum o uso de ferramentas de teste para simplificar e acelerar o gerenciamento de toda a infraestrutura da virtualização e todos os processos associados a ela. É preciso se certificar de que a adoção da tecnologia de virtualização não trará resultados que comprometam os benefícios ofertados por ela.

A virtualização pode ajudar as organizações a avançar em direção a um estado mais ágil, proporcionando muito mais flexibilidade aos sistemas de TI e dinamizando as mudanças. Através da adoção dessas melhores práticas, os usuários podem aproveitar todos os benefícios dessa tecnologia:
1) Desenvolver uma estratégia coesa de virtualização por toda a infraestrutura de TI;
2) Integrar o gerenciamento das máquinas físicas e virtuais com ferramentas baseadas em diretivas para reduzir a complexidade do sistema;
3) Aumentar a segurança do servidor/desktop virtual, provendo dessa forma a disponibilidade (ou continuidade) dos negócios;
4) Conduzir as mudanças de cultura necessárias para gerenciar com eficiência os recursos em um ambiente virtual.

Alguns virtualizadores existentes

No mercado já existem diversos softwares que atuam na virtualização. Com esses aplicativos é possível investir em uma tecnologia segura e confiável obtendo a disponibilidade de melhores recursos e o aumento da produtividade. Vamos agora conhecer quatro opções:

1)VMware

A solução VMware Server proporciona integração entre diferentes sistemas operacionais, simulando redes com várias versões do Linux e do Windows. Ele é uma opção para se testar (sem a necessidade de um computador dedicado a esta tarefa) outros sistemas operacionais e aplicativos de forma mais simples, ou até mesmo para manter a compatibilidade de sistemas operacionais e aplicativos antigos (legados) e recentes. Algumas de suas características são:
• Suporta sistemas operando na tecnologia 64 bits, incluindo Windows e Linux;
• Monitora e fiscaliza a infraestrutura em um console central de gerência;
• Tem suporte a dois processadores SMP virtual (SMP virtual é um módulo adicional ao VMware que possibilita que uma única máquina virtual possua mais de 1 processador).

2)Virtual PC

O Virtual PC é a solução desenvolvida pela Microsoft para virtualização do computador. Assim como as outras ferramentas, ele permite rodar diversos sistemas operacionais em uma única máquina, atendendo usuários do Windows XP.

Uma das vantagens do software é a ágil reconfiguração dos equipamentos. Ele otimiza a produtividade dos profissionais de suporte técnico, pois permite facilmente a troca de sistemas operacionais sem a necessidade de realizar logins ou logouts sempre que atendem a um chamado. Algumas de suas características são:
• Suporte para até 4 adaptadores de rede por máquina;
• Configurações baseadas na linguagem XML;
• Suporte para até 4GB de memória;
• Possui Virtual Machine Additions, que oferece alto nível de integração entre os sistemas;



Nota do DevMan

Máquina Virtual: Uma máquina virtual é um ambiente, isolado ou não, de computação que é implementado em software e que compartilha dos recursos de hardware do computador físico para que vários sistemas possam funcionar simultaneamente em um único computador.

Os recursos que podem ser compartilhados são: processadores, memória, disco rígido, placas de rede, placa de vídeo e demais.

• Roda a maior parte dos sistemas x86 sem a necessidade de drivers personalizados.

3)Virtual Server R2 SP1

Outro produto da Microsoft são os recursos do Virtual Server. Sua instalação pode ser realizada apenas em sistemas operacionais servidor, como Windows Server 2000 e 2003.

O Virtual Server fornece aos administradores um maior controle, conectando máquinas virtuais e permitindo automatização durante o processo. Algumas de suas características são:
• Suporte para conectividade permitindo cluster de todas as máquinas virtuais executando sobre um host;
• Suporta a tecnologia 64 bit;
• Melhorias no Hyper-threading.

4)XEN

O Xen é uma opção opensource para virtualização. Com funções semelhantes ao VMware, ele se caracteriza por ter como base o Linux. Em resumo, o Xen utiliza um conceito chamado para-virtualização, onde o sistema operacional rodando dentro da máquina virtual tem a ilusão de estar sendo executado diretamente sobre o hardware. Ele se encarrega de organizar as requisições feitas pelas máquinas virtuais e repassá-las ao sistema principal.

Voltado para solução em servidores, ele possibilita, como as outras soluções, vários servidores virtuais em uma mesma máquina. Para Giani Maldaner, Diretor Técnico da SISNEMA, o XEN é uma opção acessível e robusta àqueles que querem entrar no mundo virtualizado. "Com a adoção cada vez maior do Linux entre as empresas e organizações, soluções que atendam essa demanda se tornam fundamentais, ainda mais quando elas representam vantagens

com um baixo custo de implementação", contextualizou o Diretor.

Algumas de suas características são:

- Baixo custo na implementação;
- Através do conceito de para-virtualização ele obtém alto rendimento, inclusive em arquiteturas (x86);

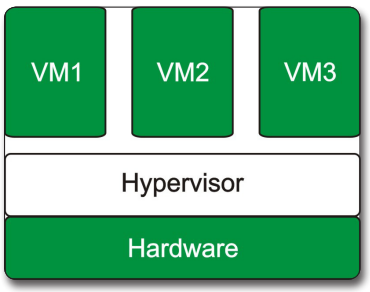


Figura 1. Tipo-1 de hypervisor

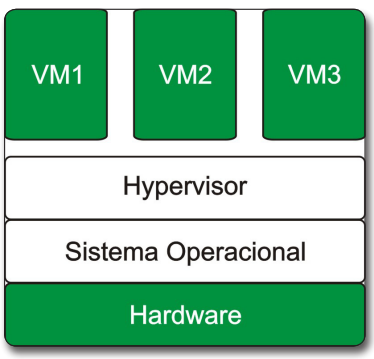


Figura 2. Hypervisores Tipo-2 executam as VMs pelo sistema operacional

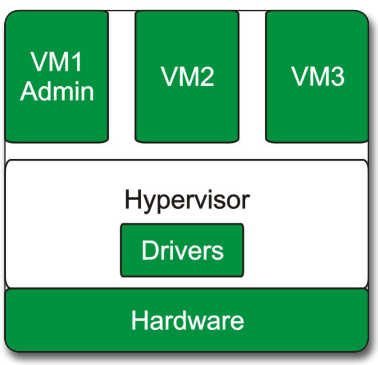


Figura 3. Modelo de Hypervisor Monolítico

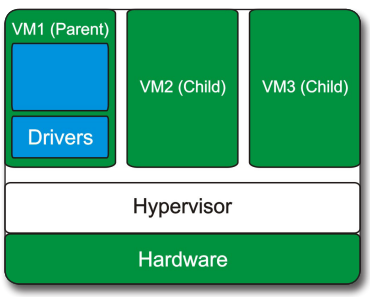


Figura 4. Modelo de Hypervisor Microkernelizado

- Sustentação para 32 usuários com memória de 4GB;
- Suporte para tecnologia 64 bits e x86;
- Além de ter o código fonte aberto, proporcionando maior integração com outras tecnologias.

Hypervisor

A criação do hypervisor é um marco na evolução da computação, pois apresenta uma forma de superar as limitações de arquitetura e o alto custo do uso de servidores mainframe.

Existem vários tipos de hypervisores pelos quais eles podem ser categorizados, por exemplo, se é executado diretamente no hardware físico ou pelo sistema operacional (hosted by). Hypervisores também podem ser categorizados pelo modelo de implantação, ou seja, se eles são monolíticos ou microkernelizados. Conheceremos um pouco mais sobre a arquitetura de hypervisores na próxima seção.

Tipos de arquitetura de Hypervisores

Tipo-1

Este tipo de hypervisor é executado entre a camada de hardware e as máquinas virtuais cliente (ver **Figura 1**). É executado diretamente no modo nativo (bare metal), em vez de dentro de um ambiente de sistema operacional. Geralmente fornece o melhor desempenho, disponibilidade e segurança dentre as demais arquiteturas de hypervisor.

Seguem alguns produtos virtualizadores que implementam este tipo de hypervisor:

- Microsoft Hyper-V;
- Citrix XenServer;
- VMware ESX Server.

Tipo-2

Este tipo de arquitetura necessita do sistema operacional para que sejam executadas as máquinas virtuais. Neste caso, o SO é o host (ver **Figura 2**).

Comparando-se a **Figura 2** com a **Figura 1**, na plataforma de hypervisor do tipo-2 existe uma camada a mais separando o hardware das máquinas virtuais guest, que é o sistema operacional host. Este nível extra resulta em um esforço maior exercido pelo hardware físico, causando uma sobrecarga que limita o desempenho e o número de máquinas virtuais que podem ser hospedadas.

Como exemplos deste Tipo-2 de hypervisor temos:

- Microsoft Virtual Server;
- VMware Server;
- Microsoft Virtual PC.

Modelos de Hypervisores

Do mesmo modo que existem tipos de arquitetura (Tipo-1 e Tipo-2), o hypervisor possui dois modelos: monolítico e microkernelizado, como veremos logo abaixo.

Hypervisor Monolítico

Neste modelo (ver **Figura 3**), os drivers são hospedados e gerenciados pelo hypervisor e traz alguns benefícios e inconvenientes. Como exemplo de benefício, o hypervisor não precisa de um controle, da partição pai ou sistema operacional, pois os sistemas operacionais guest interagem diretamente com o hardware utilizando os drivers do hypervisor. O inconveniente é que a variedade de modelos de placas-mãe, placas de rede, placas de vídeo e outros dispositivos, dificulta o desenvolvimento específico do hypervisor.

Hypervisor Microkernelizado

Neste modelo (ver **Figura 4**), o hypervisor não necessita de drivers desenvolvidos especificamente para ele, pois ele possui um sistema operacional atuando como Root ou partição pai. Esta partição fornece um ambiente de execução necessário para os drivers de dispositivo por acessar diretamente o hardware do computador host. Os drivers dos dispositivos físicos são instalados no sistema operacional em execução na partição pai, não sendo necessária a instalação desses drivers nos sistemas operacionais convidados. Dessa forma, quando o sistema operacional convidado precisa acessar o hardware físico, ele simplesmente se comunica com a partição pai.

Agora que já temos uma fundamentação teórica sobre o assunto, partiremos para o uso específico do Hyper-V.

A Arquitetura do Hyper-V

A **Figura 5** apresenta os principais pontos da arquitetura do Hyper-V.

O Partition (Partição) é a unidade básica de isolamento suportada pelo hypervisor. Uma partição é composta por um espaço de endereço físico e um ou mais processadores virtuais. Além disso, a uma partição podem ser atribuídos os recursos de hardware específico (memória, dispositivos e ciclos de CPU) e as permissões de determinados direitos de acesso.

Já a Partição Parent (Pai) trata-se de uma partição que cria partições filho (child) e contém uma pilha de virtualização (Virtualization Stack) para controlar as partições filhas (Child Partition).

A Partição Child (Filha) é o software rodando dentro de uma partição. O hóspede pode consistir de um sistema operacional cheio de recursos como o Windows XP ou um kernel pequeno, para fins especiais.

Além destes, outros conceitos presentes na arquitetura do Hyper-V são:

• **Hypercalls:** servem para a comunicação ente as camadas das partições com o hypervisor;

• **Virtualization Stack (Pilha de Virtualização):** os componentes de virtualização hospedados na partição pai são mencionados coletivamente como a pilha de virtualização. A pilha de virtualização é executada na partição pai e faz acesso direto ao hardware do computador host. Na implementação do Microsoft Hyper-V do modelo hypervisor Tipo-1, a pilha de virtualização é composta pelos seguintes componentes:

- Virtual Machine Management Service;
- Virtual Machine Worker Service;
- Virtual Devices (Dispositivos Virtuais);
- Virtualization Infrastructure Driver;
- Windows Hypervisor Interface Library;
- Virtualization Service Providers (VSP);
- Virtual Machine Bus (VMBus).

• **Virtualization Services Providers (VSP):** estão hospedados na partição pai e proporcionam uma forma de publicação de serviços para partições filho, fornecendo recursos de I/O relacionados com os Virtualization Service Clients (VSCs) rodando nas partições filho. VSPs são o ponto de extremidade do servidor e VSCs são o ponto de extremidade do cliente que viabilizam as comunicações cliente/servidor para troca de informações sobre as funcionalidades dos dispositivos. Todas as comunicações entre os VSPs e os VSCs são feitas sobre o VMBus;

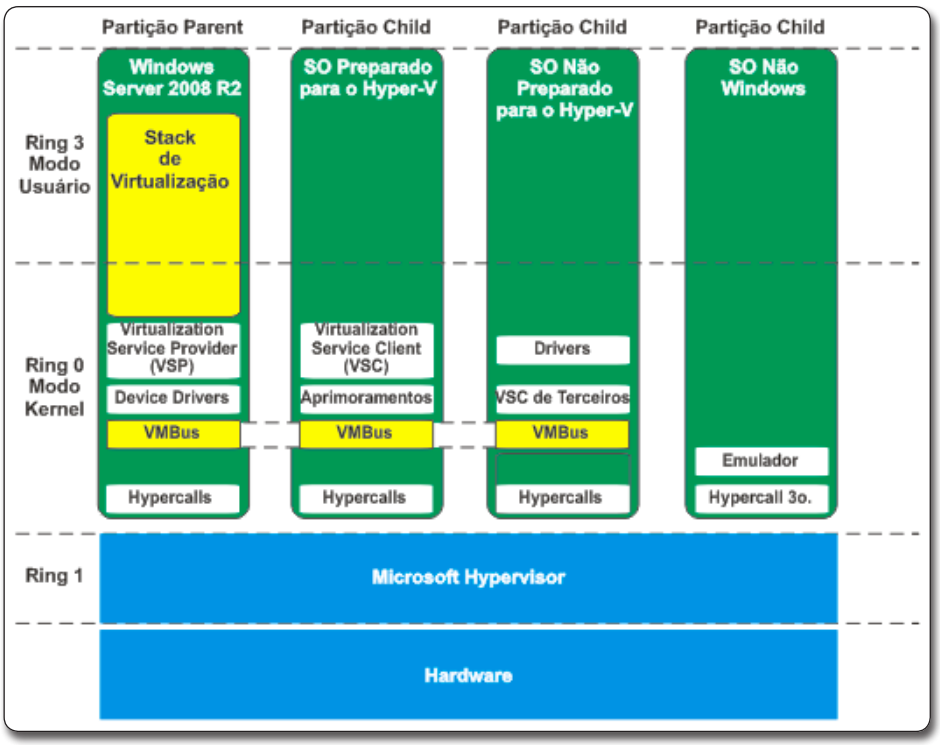


Figura 5. Arquitetura do Hyper-V

• **Virtualization Services Clients (VSC):** são dispositivos sintéticos que residem na partição filha que usam recursos de hardware fornecidos pelos VSPs na partição pai, comunicando sobre o VMBus. VSCs são automaticamente disponibilizados para a instalação quando os Integration Services (serviços de Integração) estão instalados na partição filha, permitindo que o sistema operacional virtualizado utilize dispositivos de hardware sintéticos (dispositivos virtuais mapeados diretamente para os dispositivos físicos). Sem o Integration Services instalado, uma partição filha só pode usar dispositivos emulados (drivers projetados para suportar sistemas operacionais antigos);

• **Virtual Machine Bus (VMBus):** canal de comunicação entre a partição pai e as partições filhas. O objetivo do VMBus é fornecer um mecanismo de comunicação altamente otimizado entre as partições virtuais, diferente de outras técnicas que são mais lentas devido à maior sobrecarga que impõe a emulação.

Recursos do Hyper-V

A partir da arquitetura apresentada na seção anterior, o Hyper-V oferece os seguintes recursos:

• **Amplo Suporte a Sistemas Operacionais:** executa, simultaneamente, diferentes tipos de sistemas operacionais incluindo os de 32 bits e 64 bits. Além disso, permite o uso de sistemas operacionais em diferentes plataformas de servidor, como Windows e algumas distribuições Linux, como o Suse e Rad Hat;

• **Extensibilidade:** O Hyper-V tem como base os padrões de interface Windows Management Instrumentation (WMI) e as interfaces de programação de aplicativo (APIs) para permitir que, independente de fornecedores de software (ISVs), os desenvolvedores criem rapidamente ferramentas personalizadas, utilitários e melhorias para a plataforma de virtualização;

• **Network Load Balancing:** O Hyper-V inclui capacidades de switch virtual que

fornecem a possibilidade de usar o Windows Network Load Balancing (NLB) para serviço de balanceamento de carga em máquinas virtuais rodando em servidores diferentes;

• **Arquitetura Microkernelizada:** O Hyper-V tem um hypervisor com microkernel de 64 bits que permite que a plataforma forneça uma ampla gama de dispositivos de suporte (drivers), um melhor desempenho e maior segurança;

• **Virtualização Assistida por Hardware:** O Hyper-V exige e usa as tecnologias de virtualização por hardware Intel VT ou AMD-V;

• **Arquitetura de Hardware Compartilhado:** O Hyper-V inclui as arquiteturas de Virtualization Service Provider (VSP) e Virtualization Service Client (VSC), o que proporciona maior acesso e utilização dos recursos de hardware, como discos, rede e vídeo;

• **Quick Migration:** O Hyper-V oferece a possibilidade de migrar uma máquina virtual em execução de um computador host físico para outro com um mínimo de paralisação, alavancando capacidades de alta disponibilidade do Windows Server 2008 com as ferramentas de gerenciamento do System Center;

• **Escalabilidade:** O Hyper-V inclui suporte para múltiplos processadores e núcleos no host e melhoria da memória dentro de máquinas virtuais. Esse suporte permite que ambientes de virtualização sejam dimensionados para suportar um grande número de máquinas virtuais em um determinado host, continuando a potencializar a rápida migração para a escalabilidade em vários hosts. O Hyper-V R2 provê mais dois recursos de escalabilidade para sua infraestrutura:

• *Suporte para mais de 64 Processadores Lógicos no Pool de Processadores do Host:* O número de processadores lógicos suportados pelo Hyper-V neste lançamento quadruplicou em relação à versão anterior. Isto permite tirar vantagem nos negócios em larga escala, a fim de maximizar os benefícios da consolidação da carga de trabalho dos servidores existentes em uma maior densidade. Além disso, o Hyper-V pode suportar até quatro processadores lógicos por máquina virtual, permitindo que os sistemas operacionais de servidor possam utilizar-se do recurso de multiprocessamento para aplicações que precisem do recurso;



Figura 6. Controle deslizante do Dynamic Memory. Mais performance ou maior escalabilidade?

• *Suporte ao Core Parking:* Permite que um core sem uso, do processador, entre em estado inativo economizando energia.

• **Suporte a Multiprocessadores Simétricos:** O Hyper-V oferece suporte para até quatro processadores em um ambiente de máquina virtual, a fim de tirar vantagem de aplicações multithreaded em execução em uma máquina virtual;

• **Snapshots de Máquinas Virtuais:** O Hyper-V oferece a capacidade de tirar snapshots de uma máquina virtual em execução para que você possa facilmente voltar ao estado anterior e assim melhorar soluções de backup e recuperação de desastres;

• **Live Migration:** Presente a partir do Hyper-V R2, o Live Migration permite mover um máquina virtual de um servidor Hyper-V para outro sem cair a conexão da rede e sem qualquer tempo de inatividade percebido pelo usuário ou a interrupção de outros serviços que o desempenho possa retardar por alguns segundos. O Live Migration pode fornecer alta disponibilidade a servidores e aplicações rodando em cluster Hyper-V num ambiente de datacenter;

• **Cluster Shared Volumes:** É um novo recurso de Failover Clustering disponível no Windows Server 2008 R2 que fornece um único e consistente namespace para que todos os nós do cluster vejam o mesmo armazenamento. Cluster Shared Volumes são altamente recomendados para cenários de Live Migration;

• **Suporte para Unidades de Armazenamento Hot-Swap:** Na versão R2 do Hyper-V, pode-se adicionar ou remover discos rígidos virtuais (VHDs) e discos pass through de uma máquina virtual em execução sem a necessidade de desligar e reiniciar a mesma;

• **Modo de Compatibilidade de Processadores:** Este modo, presente no Hyper-V R2, permite a migração de uma VM entre máquinas host com a mesma arquitetura de processador (AMD ou Intel). Isto torna mais fácil a atualização da infraestrutura de hosts Hyper-V, simplificando a migração das VMs de máquinas host rodando em hardware antigo a máquinas host rodando em um hardware mais novo. Ele também oferece mais flexibilidade para a migração de máquinas virtuais entre os nós de um cluster;

• **Desempenho:** O Hyper-V pode prover diversos recursos para aprimorar seu desempenho. Dentre eles, podemos destacar o suporte a 384 VMs com processador virtual simples, 256 VMs com processador virtual dual e 128 VMs com processadores virtuais quad.

Recentemente fomos informados que serão disponibilizados, em um service pack do Windows Server 2008 R2, dois novos recursos importantes:

• **Remote FX:** Onde os usuários serão capazes de trabalhar remotamente em um ambiente de desktop Windows Aero, assistir vídeo full-motion, desfrutar de animações Silverlight, e executar aplicativos 3D dentro de uma VM Hyper-V;

• **Dynamic Memory:** Com esta tecnologia, será permitida uma maior densidade (escalabilidade) de servidores e estações virtuais por servidor físico e, o mais importante, sem degradar a performance do servidor host. Se você usa o Memory Overcommit da VMware, agora terá a mesma funcionalidade, porém sem degradação de performance (ver **Figura 6**). Os clientes vão poder nos pedir para que comecemos a mover a barra deslizante para aumentar a densidade e ainda minimizar o impacto no desempenho. Em resumo, memória dinâmica é um aprimoramento no gerenciamento de memória do Hyper-V projetado para uso em produção, que permite aos clientes obterem uma maior consolidação de VMs.

Agora que já conhecemos bastante os recursos do Hyper-V, vamos para a prática!

No vídeo elaborado para esta matéria, aprenda como adotar virtualização no seu ambiente de trabalho.

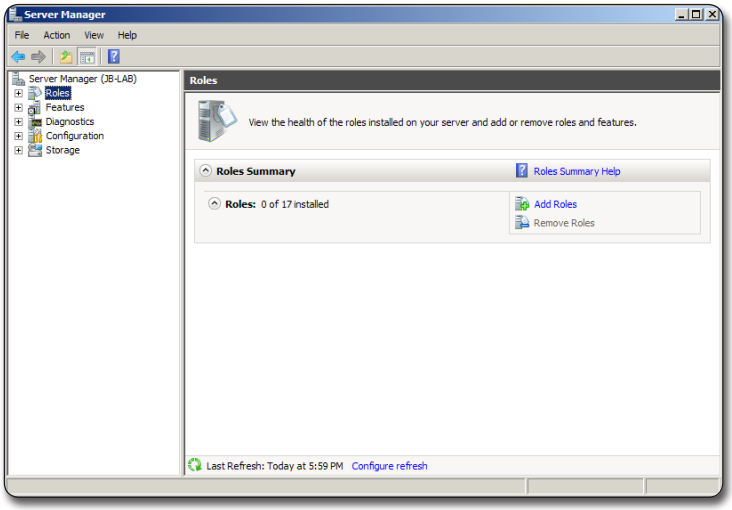


Figura 7. Server Manager

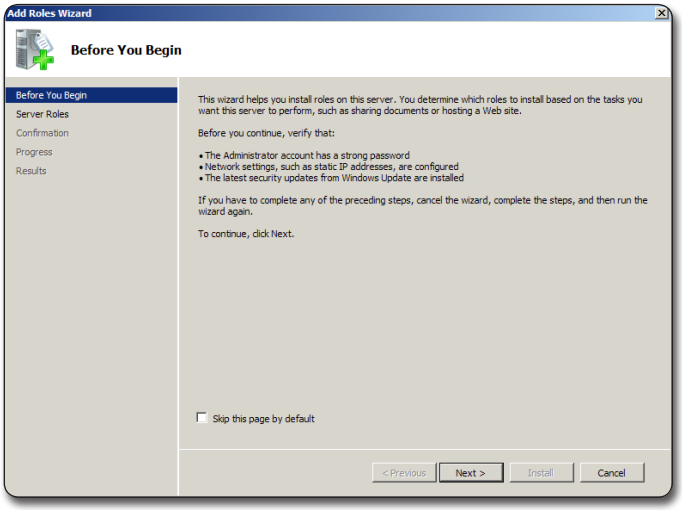


Figura 8. Informações antes de iniciar.

Instalando o Hyper-V e criando uma Máquina Virtual

Para a instalação do Hyper-V será necessário habilitarmos a role hyper-V no Windows Server 2008. É primordial que tenhamos uma configuração de hardware capaz de suportar a virtualização.

Os requisitos do sistema para viabilizar a habilitação do Hyper-V são:

• Windows Server 2008 Standard, Enterprise ou Datacenter Server x64 Edition;

• Suporte à Virtualização Assistida por Hardware (Hardware Assisted virtualization) – Intel VT ou AMD-V;

• O Suporte ao Data Execution Protection (DEP) deve estar habilitado no hardware – AMD NX bit (no execute bit) ou Intel XD bit (execute disable bit).

Neste laboratório, foi utilizado um servidor com a configuração de hardware abaixo:

- Centrino Duo P8600 2.4 Ghz, 64-bits com tecnologia VT;
- Memória Ram de 3.00 GB;
- HD com partição NTFS de 50Gb;
- Windows Server 2008 R2 Enterprise Edition.

Para instalarmos a role Hyper-V, devemos estar em *Server Manager*. Para isso, clicamos em *Start*, vamos para *Administrative Tools* e clicamos em *Server Manager* (ver **Figura 7**).

Em seguida, clicamos em *Add Roles*. O *wizard* iniciará, selecione *Next*. Agora é preciso escolher a opção *Hyper-V* em *Roles* e clicar em *Next* (ver **Figuras 8** e **9**).

Após selecionar o role do Hyper-V e clicar em *Next*, aparecerá uma tela com uma breve introdução ao Hyper-V informando que é importante identificar qual dispositivo de rede será configurado para as redes virtuais. Depois veremos a tela em que você escolherá o dispositivo de rede e, por fim, aparecerá uma tela de confirmação das opções escolhidas (ver **Figuras 10** e **11**).

Após alguns minutos, aparecerá uma janela informando que devemos reiniciar o servidor para completar a instalação (ver **Figura 12**).

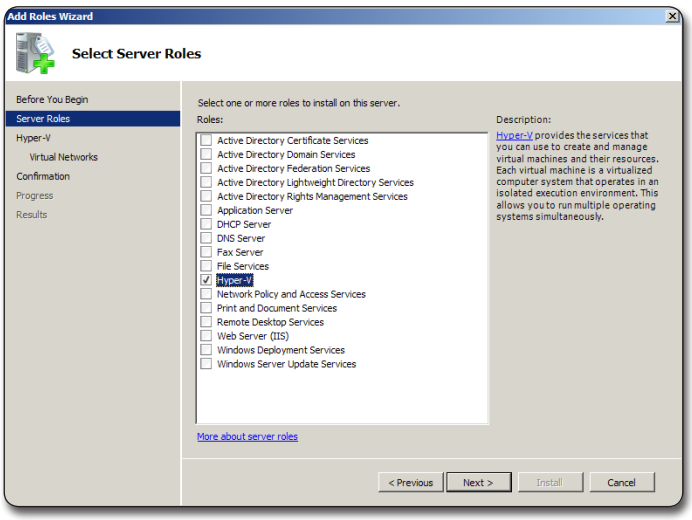


Figura 9. Selecionando a role Hyper-V

Após reiniciarmos o servidor, será exibida uma janela informando que a instalação foi realizada com sucesso. Com isso, a role do Hyper-V já está instalada e pronta para trabalharmos.

A próxima etapa será a criação da Máquina Virtual de Exemplo. O primeiro passo para isso é acessarmos o *Server Manager*, *Roles*, *Hyper-V* e em *Summary* escolhermos a opção *Hyper-V Manager*, conforme apresentado na **Figura 13**.

No menu *Action* selecionamos *New* > *Virtual Machine*, e assim o Wizard para criação da máquina virtual terá início (ver **Figura 14**).

Neste momento devemos especificar o nome da VM e o caminho padrão onde ela será armazenada. O caminho pode ser configurado para um storage externo ou outra partição, de acordo com a **Figura 15**.

Em seguida, atribuímos a quantidade de memória RAM que a máquina virtual terá. No nosso exemplo, deixamos o padrão para o Windows 7, 512Mb (ver **Figura 16**).

Na janela da **Figura 17** selecionamos o adaptador de rede virtual, que permitirá acesso da VM à rede externa (LAN do host) e à Internet (WAN).

Na opção de conectar um disco virtual, podemos escolher entre criar um disco virtual, usar um disco existente e anexar um disco depois. Em nosso exemplo, preferimos usar a opção de usar um disco existente, onde podemos digitar o caminho (path) do local onde o disco virtual está armazenado ou selecionar, clicando em *Browse...* (ver **Figura 18**). Ao final, **Figuras 19** e **20**, serão apresentados um sumário da instalação com as opções escolhidas (nome, memória, rede e disco rígido) e a tela do Hyper-V Manager com a VM criada em estado de desligada.

Neste exemplo, a VM foi instalada a partir de uma imagem .ISO do Windows 7 Enterprise x64. A seguir, veremos o processo de instalação do Sistema Operacional.

Para realizarmos esta configuração, clicamos com o botão direito do mouse na VM e clicamos em *Settings...*, conforme a **Figura 21**.

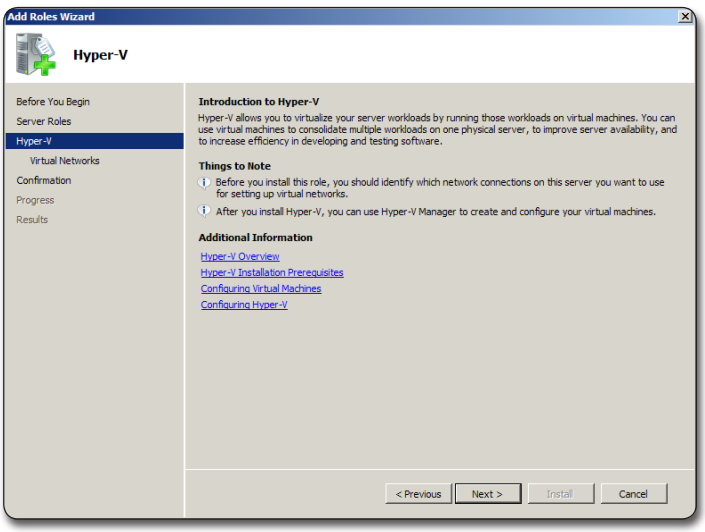


Figura 10. Breve introdução sobre o Hyper-V

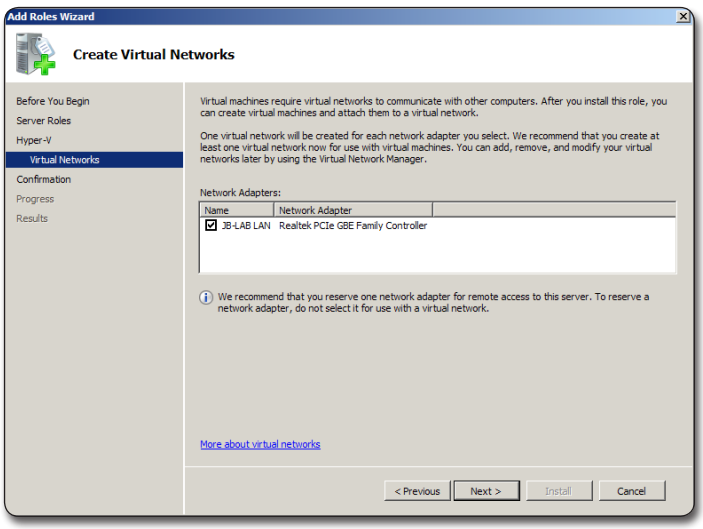


Figura 11. Selecione uma placa de rede para a rede virtual

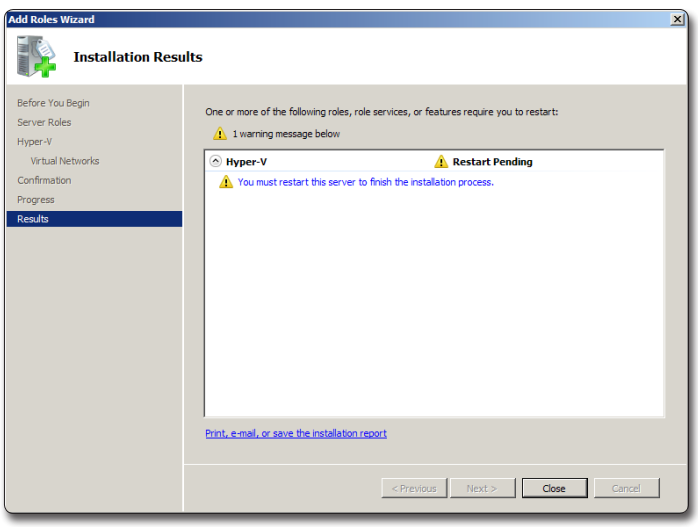


Figura 12. Resultado da instalação.

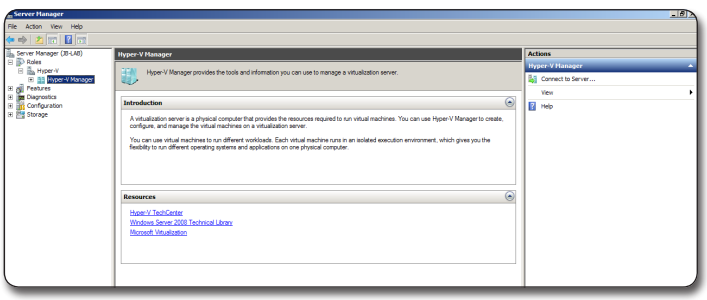


Figura 13. Hyper-V Manager

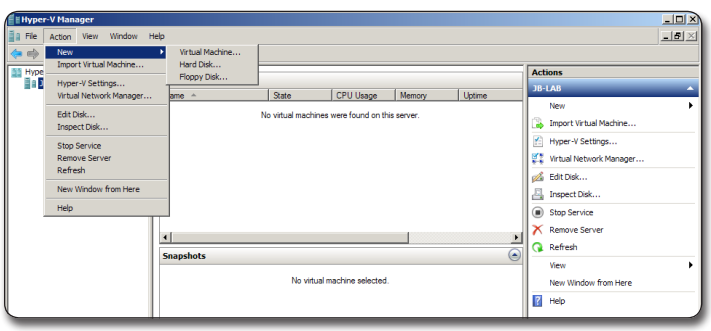


Figura 14. Criando uma VM

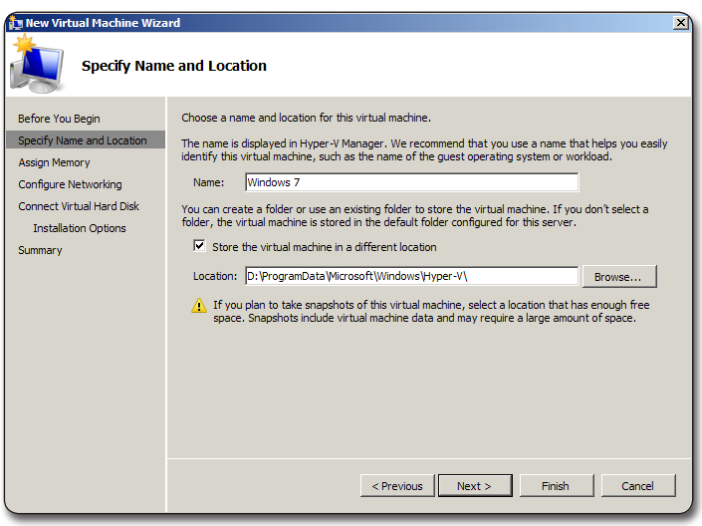


Figura 15. Especificando nome e local

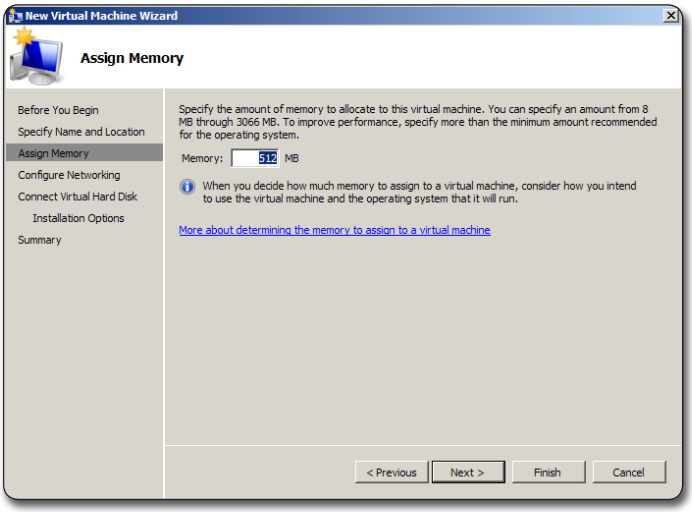


Figura 16. Memória RAM

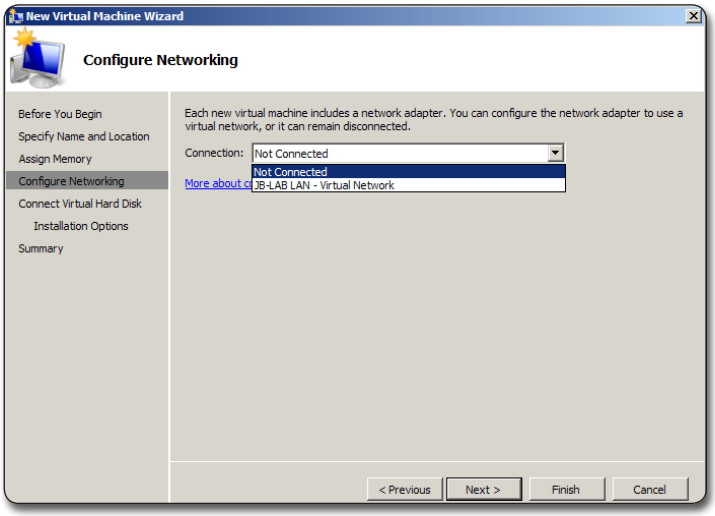


Figura 17. Configurando a rede.

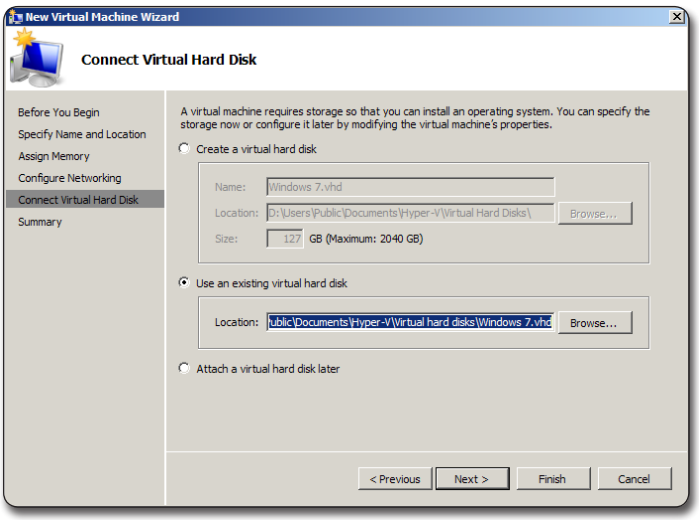


Figura 18. Conectando o disco virtual de 20Gb dinamicamente expansível

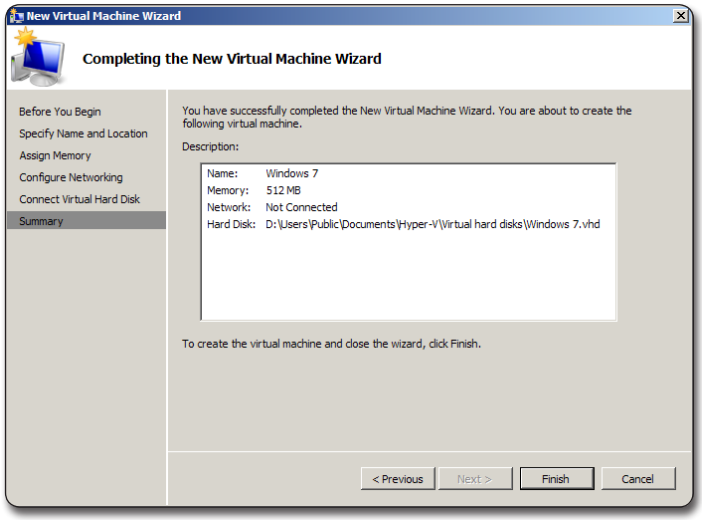


Figura 19. Sumário das configurações da VM.

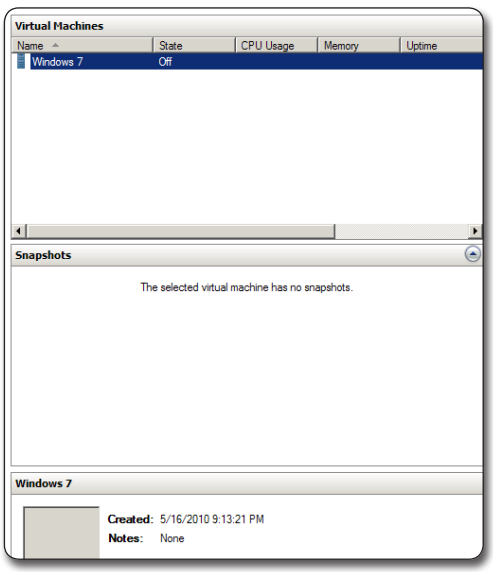


Figura 20. VM criada

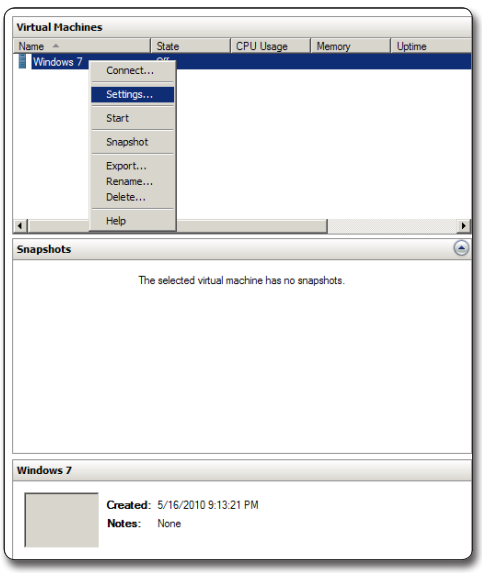


Figura 21. Acessando as configurações da VM.

Na opção *IDE Controller 1*, selecionamos a imagem do SO e aplicamos as definições para começarmos a instalar o mesmo, como pode ser visto na **Figura 22**.

Finalizada as configurações da VM, agora iremos conectar na VM, clicando com o botão direito na VM e selecionando *Connect*.... Após o procedimento anterior, aparecerá a tela da sessão informando que a VM encontra-se desligada. (ver **Figuras 23** e **24**).

Para iniciarmos a instalação do Sistema Operacional, precisamos ligar a VM, clicando no menu *Action*, opção *Start* (ver **Figura 25**).

Após ligarmos a VM, aparecerá a tela de carregamento dos arquivos da imagem de instalação do SO, e depois a tela principal de instalação em que aparecem as opções de idioma, formato de hora e layout do teclado (ver **Figuras 26** e **27**).

Após todo o processo de instalação, que demorará alguns minutos, o Windows 7 Enterprise x64 estará funcional e pronto para ser utilizado (ver **Figura 28**).

Na **Figura 29** podemos verificar as configurações da VM que acabamos de criar. Note que o Windows 7 está rodando com apenas 512Mb de memória RAM.

Conclusão

Neste artigo vimos os conceitos de virtualização, seus tipos, arquiteturas e exemplos de virtualizadores do mercado. Tivemos uma introdução sobre a nova ferramenta de virtualização da Microsoft, o Hyper-V, alguns de seus recursos e novidades que ainda serão implementadas, como o RemoteFX e o Dinamic Memory.

Por fim, aprendemos como instalar a role Hyper-V, através do Server Manager e no Hyper-V Manager, criamos uma máquina virtual e instalamos o Windows 7 Enterprise x64.

No próximo artigo conheceremos em detalhes as configurações do Hyper-V Manager e as opções de uma VM, como criar um switch virtual privado, criar um disco virtual, conhecer os tipos de discos virtuais que o Hyper-V pode oferecer e muito mais. Até a próxima!

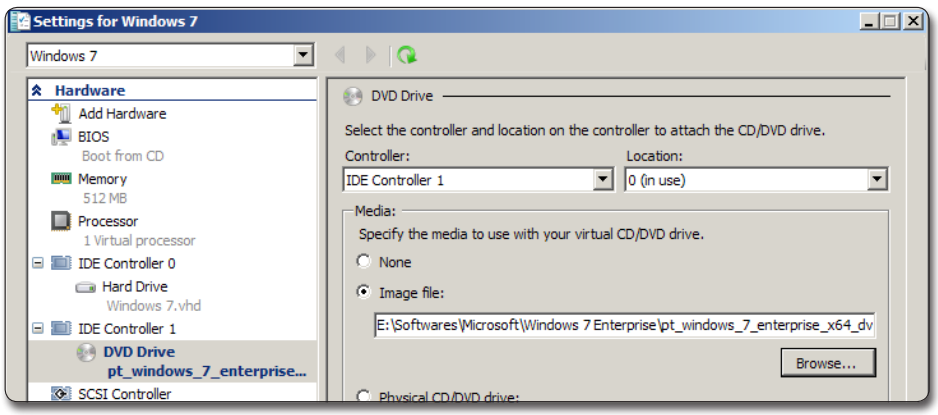


Figura 22. Selecionando a imagem.

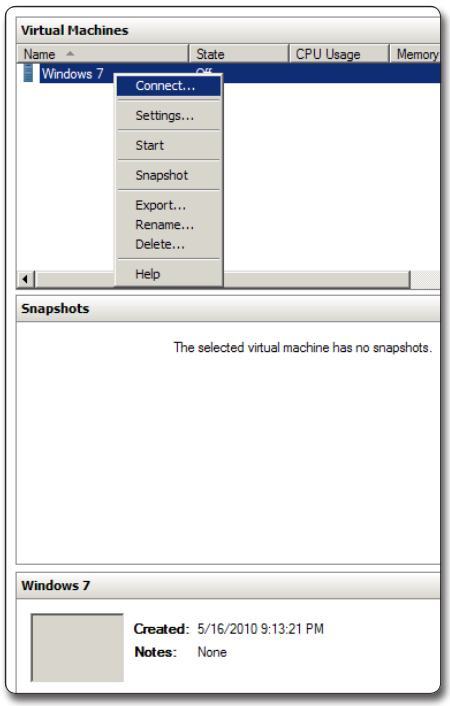


Figura 23. Conectando na VM

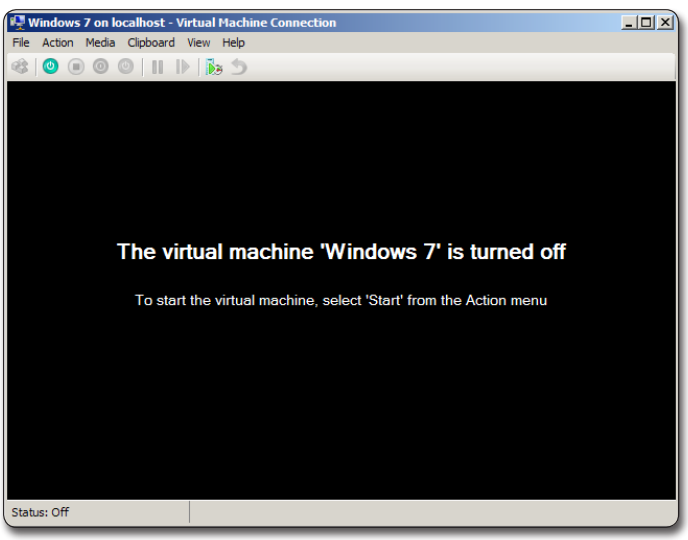


Figura 24. Sessão com a VM desligada.

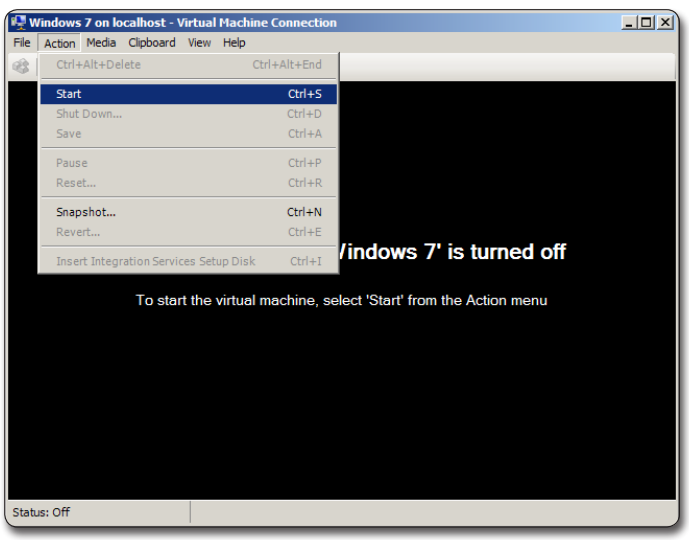


Figura 25. Ligando a VM.

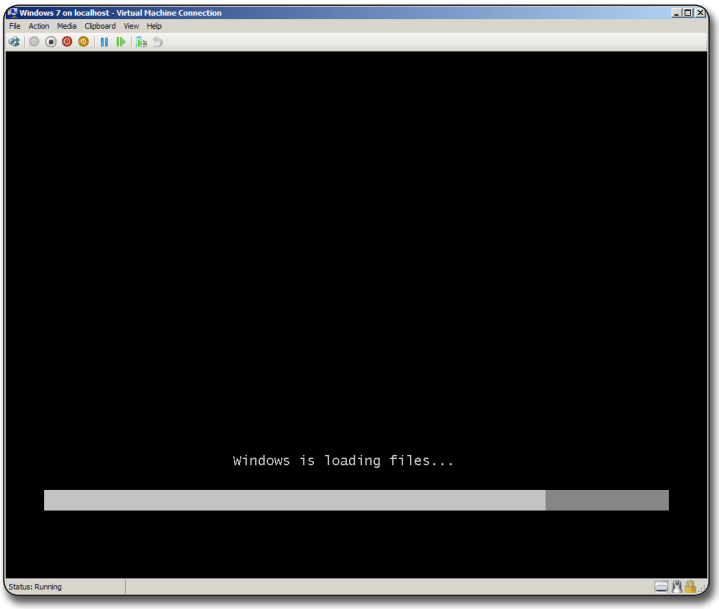


Figura 26. Iniciando a instalação do Windows 7.



Figura 27. Tela inicial da instalação do Windows 7

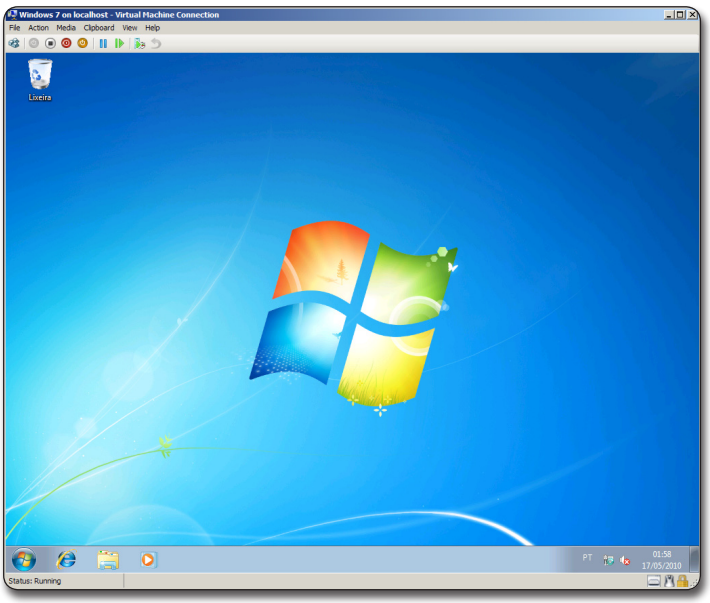


Figura 28. Windows 7 instalado

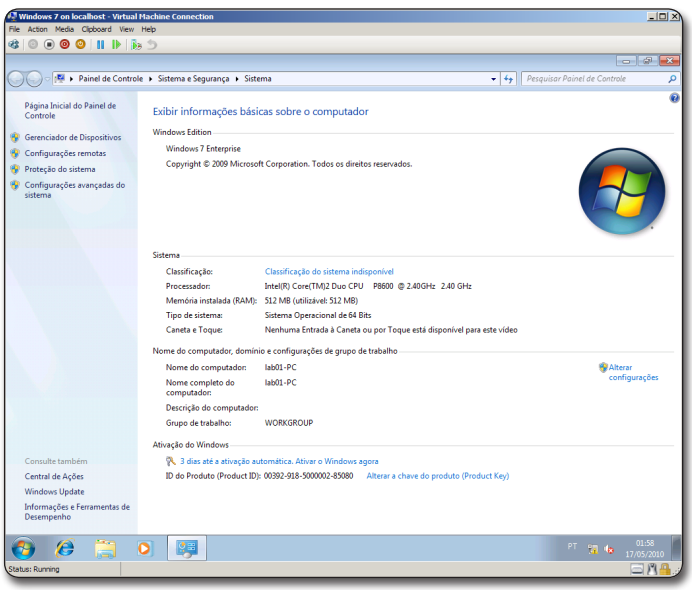


Figura 29. Informações do sistema



Jorge Barata
jorge@jorgebarata.eti.br
Atua no ramo de tecnologia desde 2003, é certificado MCSA em Windows Server 2003, MCT, MCTS em Windows Server 2008 (Active Directory, Network e Applications) e Windows Server Virtualization (Hyper-V), MCP em Windows Server 2003 e XP Professional, MCDST, MSP, ITIL Foundations V3 e CompTIA Security +. Formado em Sistemas de Inf. Especializado nas tecnologias da plataforma Microsoft, com foco em servidores (Windows Server System), Virtualização com Hyper-V e SCVMM e Windows Desktop.

Virtualização Microsoft
http://www.microsoft.com/brasil/servidores/virtualizacao/default.mspx

TechCenter de Virtualização
http://technet.microsoft.com/pt-br/virtualization/default.aspx

Blog do Fábio Hara
http://fabiohara.spaces.live.com/

LAUREANO, M. Máquinas Virtuais e Emuladores: Conceitos, Técnicas e aplicações. 1ª Ed. São Paulo: Novatec, 2006.

TULLOCH, MITCH, Understanding Microsoft Virtualizations Solutions. 2nd. Edition, Microsoft Press, 2010.

Jorge Barata - A Virtualização Em Ambiente de Testes Com o Hyper-V, ano 2008.

Monitoramento de Redes

Trabalhando com a ferramenta Nagios

Renata Aparecida Benini e Marcelo Santos Daibert

Resumo DevMan

De que se trata o artigo:

Este artigo apresenta uma definição sobre o uso de servidores e ativos de redes e a importância de monitorar uma rede de computadores. Apresenta também ferramentas de monitoramento e um estudo de caso com o Nagios, uma ferramenta para monitoramento de servidores e ativos de rede.

Para que serve:

Para auxiliar os administradores de rede a configurarem seus ambientes de monitoramento, mostrando a importância de se ter uma rede monitorada. Apresenta também os procedimentos para configuração e uso da ferramenta Nagios.

Em que situação o tema é útil:

A utilização de serviços baseados em redes de computadores, além de mais cômodos, já que é possível acessá-los de forma remota, também contribuem para a rapidez e eficácia das comunicações como um todo. Entretanto, para que estes serviços sejam utilizáveis, há uma dependência de que servidores e ativos de redes estejam funcionando corretamente. Neste sentido, o monitoramento de servidores e ativos de rede se apresenta como uma solução rápida e eficaz para identificação de problemas, impactando diretamente no índice de disponibilidade destes equipamentos.

Com a evolução tecnológica, um importante diferencial que impactou a sociedade e impacta sobre medida os negócios são as redes de computadores. O que hoje é conhecido como Globalização (um fenômeno de integração política, cultural, social e econômica), foi impulsionado graças ao achatamento conceitual do globo terrestre, em especial ao avanço dos transportes (não há distâncias intransponíveis) e ao avanço das telecomunicações. Assim, as redes de telecomunicações passaram a ser uma necessidade para o estágio atual de evolução de nossa civilização. Deixou de ser um luxo, para ser um importante marco estratégico de empresas e necessário para as vidas de cidadãos comuns.

A tecnologia da informação teve seu avanço na década de 50 quando surgiu o primeiro sistema de computador, baseado em sistemas de informação e equipamentos complexos para armazenamento de informações, operados por pessoas altamente especializadas. Como tudo que envolve a informática teve um avanço muito acelerado, com as redes de computadores não foi diferente. Depois da fusão dos computadores com as comunicações, houve uma profunda mudança na forma como os sistemas computacionais eram organizados. O velho modelo de um computador atendendo a todas as necessidades computacionais de uma empresa foi substituído por um conjunto de computadores autônomos interconectados que podem trocar informações, conhecido como redes de computadores.

Na década de 80 as redes de computadores ganharam importância, especialmente com a diminuição dos custos dos computadores, tornando cada vez mais atraente a distribuição do poder computacional em módulos com processadores localizados em diversos pontos de uma organização. Assim, o uso de rede para compartilhamento de recursos passou a ser indispensável no cotidiano de uma empresa, e passaram a ser usadas também em ambientes domésticos. A rede de computadores não se limita a Internet, e está presente em várias atividades do cotidiano. São utilizadas em serviços bancários, no uso do cartão de crédito, em chamadas telefônicas, entre outros. Percebe-se que a cada dia há uma dependência maior na utilização destes serviços e, por consequência, na utilização das redes.

Nesse contexto, o uso de um computador sem que esteja conectado a uma rede está ficando sem utilidade. A facilidade e comodidade que é adquirida utilizando os serviços de rede fazem com que haja dependência desses serviços. Se o serviço de rede de uma empresa estiver inativo, todos os departamentos, diretamente ou indiretamente, são afetados. É comum, inclusive, que os serviços de uma empresa fiquem ligados a um computador servidor, onde é provido acesso a um sistema ou responsável pela comunicação interna (ler **Nota DevMan 1**).

O rápido crescimento e a proliferação de novas tecnologias têm mudado as características das redes de computadores nos últimos anos. O monitoramento, em tempo real, da infraestrutura de redes e seus



Nota do DevMan 1

Servidores e Clientes

Os servidores são sistemas que fornecem os serviços de redes, como por exemplo: e-mail, arquivos, web, impressão, compartilhamento de dados. Os servidores são responsáveis por gerenciar recursos de hardware (como disco e memória), de software (como planilhas e sistemas), e compartilhar informações (como o banco de dados de uma empresa). Já os clientes são aqueles que acessam os servidores e fazem de alguma forma uso dos serviços providos.

ativos vem se tornando indispensável na gestão da tecnologia da informação. Este monitoramento permite obter as informações necessárias sobre estes equipamentos de modo rápido, sintético, preciso e confiável, facilitando as tomadas de decisão do gestor no momento do planejamento, adequação e expansibilidade do parque tecnológico. A verificação do desempenho de serviços e a resolução de problemas diversos, como conectividade e integração de plataformas, também ocorrem mais facilmente.

As ferramentas disponíveis no mercado para monitoramento de sistemas e servidores permitem realizar uma análise nos processos e seus serviços de forma a identificar o mais cedo possível qualquer falha, buscando assim uma solução do problema antes mesmo que qualquer usuário possa ter notado. O monitoramento de serviços e ativos de rede é uma técnica que busca fazer um monitoramento ostensivo para que, quando houver um problema, os administradores de rede sejam os primeiros a serem notificados.

Neste contexto, o objetivo deste artigo é estabelecer uma análise sobre o monitoramento de redes de servidores e ativos de rede, comparando as ferramentas existentes no mercado como Nagios, The Dude e WhatsUP e estabelecer passos para configuração da ferramenta Nagios. Além disso, é feito um estudo de caso na instituição FAGOC – Faculdade Governador Ozanam Coelho, onde é utilizada a ferramenta de monitoramento Nagios para verificar os serviços de FTP (File Transfer Protocol, Protocolo de Transferência de Arquivos), SSH (Secure Shell), conexão dos servidores de e-mail e web.

Para alcançar os objetivos desse artigo, foi feito um estudo sobre as ferramentas de monitoramento de redes e feita uma comparação sobre os serviços e recursos que as ferramentas disponibilizam. Fazendo uma analogia das ferramentas, foram analisados os pontos fortes e pontos fracos de cada uma delas. Ao final, foi instalada a ferramenta Nagios 3.2.0 no sistema operacional Linux CentOs 5.0, e configurada para monitorar os servidores de e-mail, web, DNS (Domain Name System - Servidor de Nomes de Domínios) e LTSP (Linux Terminal Server Project).

Redes de computadores

O século XX foi um marco com o avanço no campo da informação. As pessoas buscam cada vez mais utilizar redes de computadores para se comunicar, armazenar e processar informações. Segundo Tanembaum, "*uma rede de computadores pode oferecer um meio de comunicação altamente eficaz para funcionários que trabalham em locais muito distantes um do outro*". As redes, hoje, são fortemente usadas em todas as áreas e por diversas classes de pessoas, seja para uso pessoal, para ler um e-mail ou acessar algum serviço de alguma rede social, até uso empresarial, para servir dados e aplicações para funcionários.

A tecnologia utilizada para a construção de uma rede pode ser:

- ponto a ponto, que consiste em conexão de pares individuais de máquinas que se comunicam direto sem passar por um intermediário onde tem um emissor e um receptor de dados;
- multiponto, que permite a conexão de várias máquinas compartilhando entre todos os dispositivos de rede um meio de transmissão único, através do qual toda a informação é transmitida.

Quanto à abrangência, uma rede pode ser definida em redes locais, as chamadas LANs (Local Area Network), que compartilham recursos e são redes privadas. As LANs abrangem em média 10 km e são utilizadas para conectar servidores, estações, periféricos ou qualquer outro recurso de rede que possuem dispositivos de processamento em um escritório, casa, ou edifícios. Quando uma rede é mais ampla são denominadas MANs (Metropolitan Area Network ou redes metropolitanas), que abrangem uma cidade. Já as redes WANs (Wide Area Network), conhecidas como redes geograficamente distribuídas, podem abranger um país ou continente, e temos como principal exemplo a Internet.

Uma outra estrutura de rede que vem ganhando uma importância relativa é a rede sem fio (*wireless*), sendo cada vez mais utilizada em dispositivos móveis, smartphones e celulares. É uma conexão de rede utilizada mais para acesso a Internet em escritórios, bares, aeroportos, parques e até mesmo em casa.

Assim, temos que uma rede permite que diversos equipamentos e recursos possam ser interligados e compartilhados, dando acesso a protocolos e requisitos de segurança de forma a permitir que o usuário possa se beneficiar com a utilização de serviços de rede.

Na próxima seção são apresentados conceitos de ativos e serviços de rede. O entendimento destes conceitos é importante para o entendimento da tarefa de monitoramento discutida neste artigo.

Ativos e serviços de redes

Uma rede de computadores é formada por diversos tipos de equipamentos, como computadores, roteadores, switch e serviços de rede. Esses equipamentos e serviços podem ser monitorados dentro de uma rede, evitando assim falhas ou até mesmo que uma rede interrompa o seu funcionamento.

Um servidor na rede pode conter funcionalidades de diferentes naturezas. Alguns tipos de servidores/serviços de rede são:

- O servidor FTP: que é o servidor onde os usuários têm acesso a arquivos em rede;
- O servidor web: é responsável por aceitar pedidos HTTP (*Hypertext Transfer Protocol*) de clientes, servindo com páginas web e arquivos de site;
- O servidor DNS: que é responsável pela distribuição de nome de redes, convertendo nomes em IPs;
- O servidor de arquivos: que é responsável por armazenar arquivos de clientes;
- O servidor Webmail: que é responsável pelo envio/recebimento de contas eletrônicas e armazenamento de e-mails;
- O servidor de Proxy: que é o responsável pelo armazenamento dos endereços de sites acessados, funcionando como um cache.

Para interligar os computadores na rede é necessária a utilização de alguns dispositivos como o roteador e switch. Esse último responsável por criar um barramento de comunicação entre os diversos dispositivos de rede que podem estar presentes na estrutura.

O roteador é um dispositivo que tem como característica selecionar a rota mais apropriada para transferir e receber protocolos na rede. É utilizado para fazer

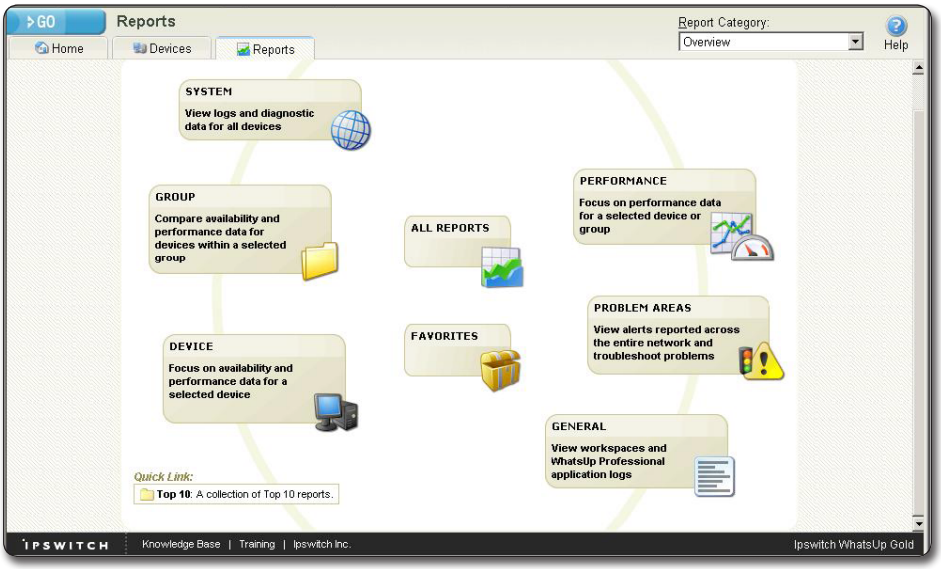


Figura 1. Interface de relatório do WhatsUp

a comunicação entre diferentes redes de computadores provendo a comunicação entre computadores distantes entre si.

Já o switch é um dispositivo utilizado em redes de computadores para reencaminhar quadros entre os diversos nós. Possuem diversas portas, assim como os hubs, e operam na camada acima dos hubs. A diferença é que ele segmenta a rede internamente, sendo que cada porta corresponde a um segmento diferente, o que significa que não haverá colisões entre os computadores de segmentos diferentes.

Alguns switches podem ser gerenciados de forma a permitir administrar as portas separadamente. O switch gerenciável normalmente tem outras funcionalidades, como a capacidade de criar VLANs (Virtual LAN), QoS (Quality of Service), firewall, entre outros. Assim é possível habilitar e desabilitar as portas, definir a velocidade da conexão, definir VLANs[1], configurar o spanning tree[2], e utilizar software de monitoramento que acessa o switch através de protocolos como o SNMP (Simple Network Management Procotol).

1 VLANS – Redes Locais Virtuais;

2 Spanning Tree Protocol (STP) – Protocolo para equipamento de rede que permite resolver problemas de loop em redes, determinando qual é o caminho mais eficiente entre cada segmento separado por switches.

Conhecendo as principais características de redes de computadores e de ativos e serviços de rede, definiremos na próxima seção o conceito de monitoramento de redes para que seja apresentada a ferramenta NAGIOS.

Monitoramento de redes

Ter um ambiente mapeado e monitorado é fundamental para o processo de crescimento de uma empresa. Já está mais do que comprovado que com um ambiente de TI bem planejado seu negócio tem mais chances de dar certo, mesmo para as empresas em que o principal foco não seja TI, pois todos dependem hoje da Internet e dos serviços que ela disponibiliza.

Assim, em um ambiente de rede é necessário considerar a eficácia no funcionamento dos equipamentos e serviços existentes. Com o aumento no uso das redes de computadores, aumentam também os problemas na rede, onde os usuários reclamam de acesso lento em horários de pico, indisponibilidade no sistema, problemas em downloads e em acessos em geral. É importante detectar qualquer problema na rede antes que o usuário possa notar a falha. Como é impossível evitar os problemas na rede, a melhor forma de resolver o impacto causado é através do monitoramento.

Monitorar rede é verificar o funcionamento de cada serviço e equipamento disponível. Para isso é necessário utilizar ferramentas que verificam o funcionamento adequado dos equipamentos e serviços, enviando relatórios e alertas aos administradores, prevenindo falhas, e fazendo com que sejam corrigidas antes que sejam notadas pelo usuário.

Ferramentas de monitoramento

As ferramentas de monitoramento de redes são utilizadas para que se possa ter um controle efetivo sobre todos os ativos nela disponíveis, verificando serviços e processos. Existem diversas ferramentas para fazer esse serviço e, para fazer sua implantação, é necessário verificar qual melhor atende às necessidades de cada estrutura de rede. Algumas opções conhecidas são The Dude, WhatsUP e Nagios, entre outras.

The Dude é uma ferramenta que faz a verificação automática de todos os dispositivos dentro de redes especificadas, desenha e faz o layout de mapas de rede, monitora serviços dos seus dispositivos e alerta em caso de problemas. É uma ferramenta gratuita e cumpre os objetivos simples de sistemas de gerenciamento de redes.

Já a ferramenta WhatsUP é um gerenciador e monitor de rede fácil de utilizar, permitindo a gerentes de TI transformar dados coletados da rede em informações estratégicas para seu negócio, como análise de tendência de planejamento de recursos estratégicos.

As ferramentas de gerenciamento The Dude e WhatsUP são similares nas suas funcionalidades. Destaca-se a ferramenta WhatsUP por possuir mais recursos e opções do que os disponíveis no The Dude, principalmente na forma de apresentar as informações que são disponibilizadas em vários tipos de relatórios e gráficos. A **Figura 1** apresenta a interface de menu para acesso aos relatórios disponibilizados pelo WhatsUp, que são os relatórios do sistema, dos dispositivos, das áreas com problemas, dos grupos e de performance. Na **Figura 2** são apresentados gráficos disponibilizados pela ferramenta The Dude mostrando informações da utilização de dispositivos.

Em uma análise comparativa, ambas as ferramentas possuem uma interface

simples que permitem acesso web, e são fáceis e intuitivas de operar para usuários acostumados com o ambiente de janelas, ícones e mouse. Além disso, possuem características diferentes quanto à organização dos menus, botões, além dos dispositivos gerenciados. O WhatsUP possui uma interface mais tradicional (**Figura 3**). Já o The Dude possui características diferentes quanto à organização das janelas e menus, o que não prejudica tanto a usabilidade (**Figura 4**). A interface padrão web do WhatsUP apresenta-se mais rica em opções do que o The Dude. Outra diferença entre essas ferramentas é que a WhatsUP é paga, enquanto a ferramenta The Dude é gratuita.

Outra ferramenta de monitoramento de redes muito utilizada é o Nagios, que monitora serviços de rede, recursos de computadores e equipamentos de rede. A próxima seção detalha o Nagios, relatando seus serviços e configuração.

Nagios

O Nagios é uma aplicação de monitoramento de redes de código aberto e licenciado pelo sistema GPL (General Public License) bastante popular. Ela permite monitorar tanto hosts quanto serviços, alertando-o quando ocorrerem problemas na rede. Ela é utilizada por administradores de redes para que possam ter um controle sobre os serviços e equipamentos de sua rede. Foi desenvolvida inicialmente para ser utilizada em sistemas operacionais Linux e, a partir da versão 3.0.4, já está sendo compatível com outros sistemas Unix.

O Nagios foi criado primeiramente com o nome de Netsaint, foi escrito e é atualmente mantido pela Ethan Galstad, e sua equipe de desenvolvedores é responsável por manter os plugins oficiais e não-oficiais. Ethan Galstad é o criador do projeto e um dos maiores mantenedores.

O Nagios busca uma forma prática de auxiliar os administradores de redes no processo de monitoração. As características principais do Nagios são: o monitoramento de serviços de rede como tráfego de dados de host e serviços que podem ser definidos pelo administrador da rede, além de monitorar serviços como SMTP (*Simple Mail Transfer Protocol*), POP3 (*Post Office Protocol*), HTTP (*HyperText Transfer Protocol*), NNTP (*Network News Transfer Protocol*), ICMP (*Internet Control Message Protocol*) e SNMP (*Simple Network Management Protocol*). O Nagios monitora também os recursos de servidores como logs do sistema, carga do processador, uso de memória e uso de disco.

Há uma infinidade de serviços que são monitorados por essa ferramenta, especialmente pela sua arquitetura baseada em plugins, onde qualquer administrador de rede pode desenvolver seu próprio plugin para monitorar qualquer particularidade de sua rede. Os plugins podem ser desenvolvidos em qualquer linguagem. Mas a

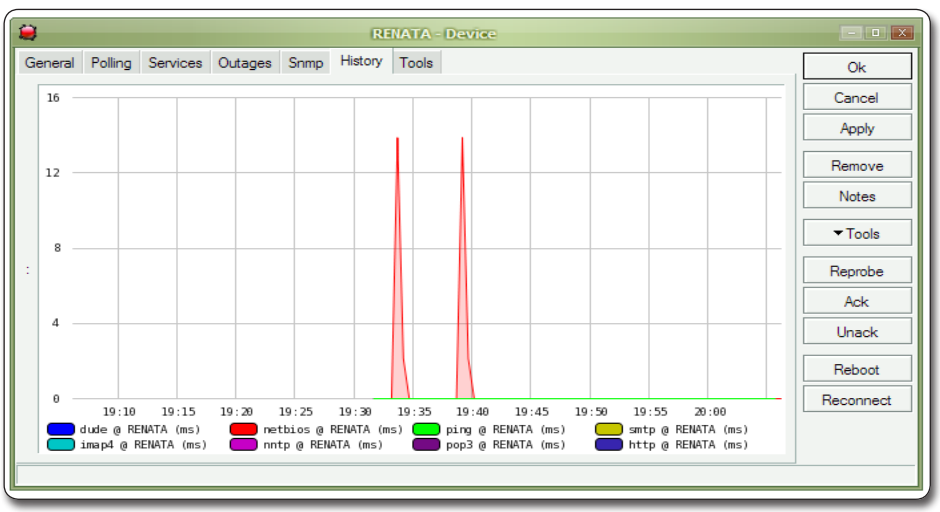


Figura 2. Gráfico com informações da utilização de dispositivos com The Dude

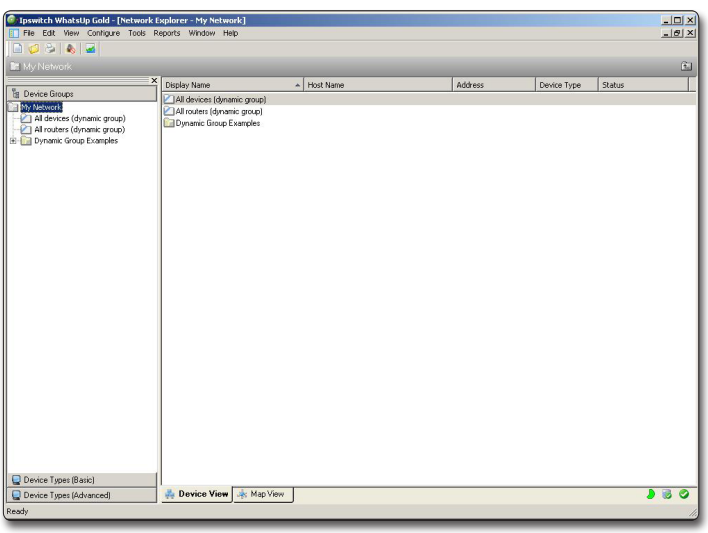


Figura 3. Interface principal do WhatsUp

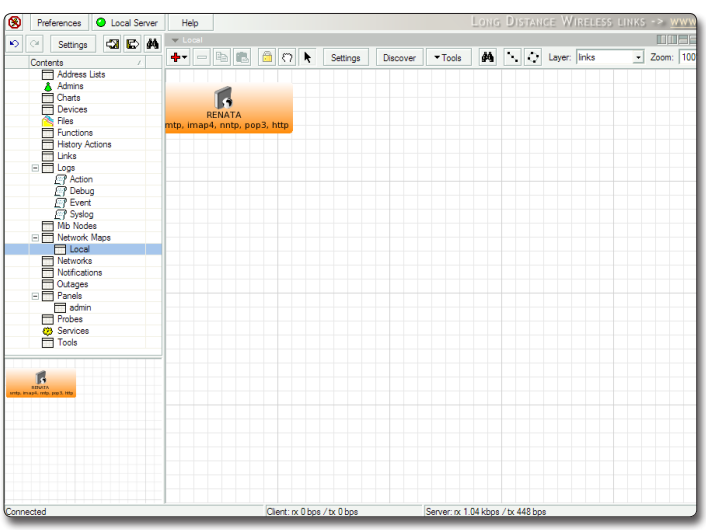


Figura 4. Interface principal do The Dude

grande maioria que existe é desenvolvido em perl e python.

A notificação dos possíveis problemas de rede é feita através de alertas enviados por e-mail, pager, SMS (*Short Message Service*), e outros meios definidos pelo administrador do sistema.

O serviço do Nagios é dado por algoritmos de verificação, podendo ser verificações simples como, por exemplo, o monitoramento via ping, ou monitoramento avançado podendo ser, neste caso, o monitoramento via o protocolo SNMP (*Simple Network Management Protocol*). Este protocolo permite que a máquina que está monitorando o host envie um OID (*Object Identifier*) para a máquina monitorada solicitando informações sobre o seu funcionamento. O protocolo então retorna a informação solicitada. O SNMP é o protocolo simples de gerência de rede que disponibiliza ao administrador da rede o acesso a diversas informações da máquina monitorada, entre elas: utilização de CPU, utilização de memória, configurações de rede, atividade de rede – download e upload –, usuários conectados, serviços ativos, entre outros. É possível ainda, na categoria de verificações avançadas, a utilização de plug-ins específicos que são disponibilizados em diversos sites especializados no assunto. Um exemplo é o site Nagios Exchange (*http://exchange.nagios.org/*), mantido pela empresa desenvolvedora do Nagios, que agrupa diversos plug-ins de seus usuários com diversas responsabilidades de monitoramento.

Os serviços de monitoração são feitos através de arquivos onde é definido o que será monitorado. A ferramenta já possui seus arquivos definidos, porém permite que o usuário crie modos próprios de monitoração. O principal arquivo da ferramenta é o *nagios.cfg*, que é onde está a configuração principal responsável pela operação do Nagios. Esse arquivo, por padrão, é criado automaticamente quando o Nagios é configurado. Outros arquivos também são necessários para o funcionamento da ferramenta, como o arquivo *resource.cfg*, que é utilizado para armazenar as variáveis que podem ser personalizadas com o objetivo de facilitar o acesso às chamadas aos plug-ins pela ferramenta. Já o arquivo *services.cfg* é utilizado para identificar os serviços que serão verificados e as métricas associadas a cada cliente monitorado.

A definição dos clientes que serão monitorados é feita no arquivo de cliente *host.cfg*, e esses clientes podem ser agrupados com o propósito de simplificar as notificações. O arquivo *contacts.cfg* define quem receberá a notificação caso a rede apresente algum problema ou restabeleça um serviço. Um ou mais contatos podem ser agrupados para receber as notificações dos clientes através do arquivo *contactgroups.cfg*.

Outros arquivos de configurações são o *timeperiods.cfg*, responsável pelo período de tempo em que será válido fazer as notificações e checagem dos serviços; e o *commands.cfg*, responsável pelas definições dos comandos a serem executados para monitoramento. Cada comando usado no parâmetro *check_command* dos hosts precisará estar configurado nesse arquivo.

Outro arquivo de configuração importante é o *CGI.cfg*, que é o arquivo de configuração das permissões de acesso dos usuários quando esta opção estiver ativa nas configurações principais.

O Nagios não possui interface para configuração na versão gratuita e nas versões CORE. A configuração é feita manualmente ao editar seus arquivos de configuração. Já na versão enterprise, a Nagios XI, toda e qualquer tipo de configuração pode ser feita pela sua interface. Existem outras ferramentas que fazem isso nas versões gratuitas, mas não são suportadas oficialmente pela desenvolvedora do Nagios. Um exemplo é o NagiosSQL (*http://www.nagiosql.org/*), uma ferramenta de administração web que possui uma interface que é definida após a instalação do Nagios para fazer a configuração do mesmo.

Antes de instalar e fazer a configuração do Nagios, é necessário preparar o ambiente em que será instalado. Como a aplicação irá gerar os arquivos CGI (*Common Gateway Interface*) para que a aplicação funcione perfeitamente, é necessário que esteja instalado um servidor Web, como o Apache, e algumas dependências como o compilador gcc (*GNU Compiler Collection*), e as bibliotecas GD (*http://www.libgd.org/*).

Em todas as versões do Nagios há uma interface Web informativa onde é possível acompanhar todo o monitoramento (**Figura 5**). Somente na versão enterprise estas interfaces se estendem para a mani-

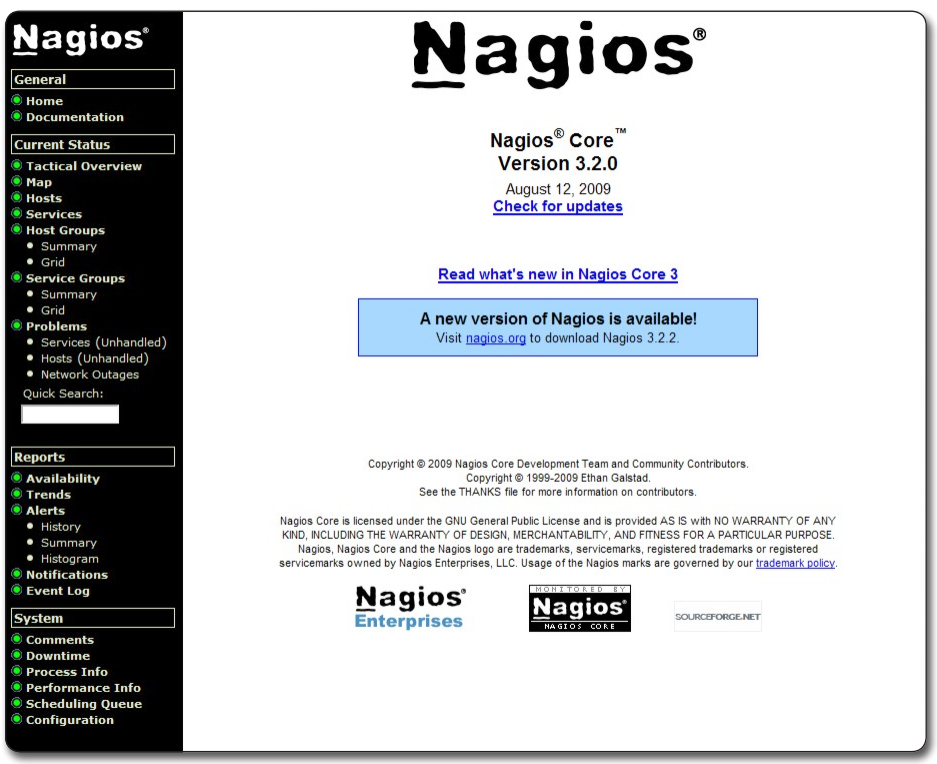


Figura 5. Interface informativa do Nagios

pulação dos arquivos de configuração. A interface das versões gratuitas é dividida em quatro partes: geral, monitoramento, relatórios e configuração.

Na parte geral (*General*) da interface existem os campos **home**, que direciona para a página principal do Nagios, e o campo **documentação** (*Documentation*), que se refere à documentação da versão do Nagios que está sendo utilizada. Na parte monitoramento (*Current Status*) tem-se o resumo de serviços, de hosts, de configuração, de monitoramento e o status da rede. É possível também visualizar os serviços e os hosts monitorados, inclusive separados por grupos que foram definidos, além do estado de cada um. Ainda na parte de monitoração, é possível visualizar o mapa da rede, os serviços com problema, a evolução e andamento do Nagios, o relatório de cada cliente, e visualizar os problemas na rede em ordem de relevância e ver onde está a causa dos problemas.

Em relatórios (*Reports*) é possível visualizar o intervalo de duração de cada problema ocorrido, a porcentagem de tempo em que um host ou serviço esteve em funcionamento, o histograma de um host ou serviço, a lista de alertas e as notificações do dia, e o resumo de alertas, filtrando os alertas por horas, serviços e/ou grupos.

Por último, na parte de configuração (*System*), é possível visualizar as configurações, os clientes e grupos de clientes, as dependências, os serviços e grupos de serviços, os comandos, os períodos de tempo e o escalonamento de notificações.

Conhecida esta visão geral do Nagios, veremos agora como trabalhar com ele na prática.

Estudo de Caso

O estudo de caso aqui apresentado é composto da implantação da ferramenta Nagios para monitoramento dos serviços do servidor PANDORA na rede da instituição FAGOC – Faculdade Governador Ozanam Coelho. Neste estudo de caso iremos considerar o monitoramento dos serviços HTTP, MySQL e disponibilidade pelo retorno do ping.

Atualmente a FAGOC possui duas redes separadas fisicamente, rede administrativa e a rede acadêmica, conforme representado na **Figura 6**. A rede administrativa interliga todos os computadores dos setores administrativos (secretaria, diretores, departamento de pessoal, tesouraria, cobrança, sala dos professores, entre outros). A rede acadêmica interliga todos os ambientes acadêmicos como laboratórios de informática e biblioteca.

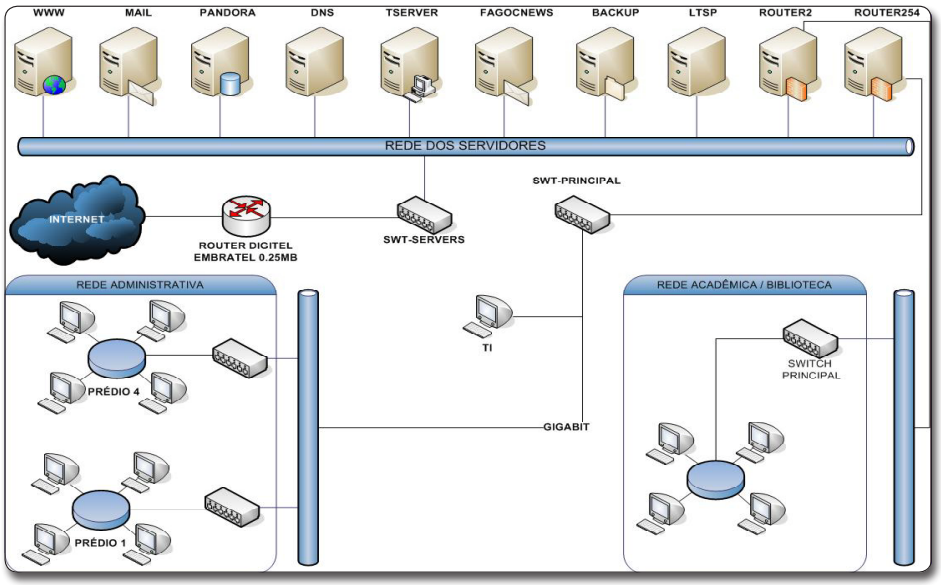


Figura 6. Rede administrativa e acadêmica da FAGOC

A FAGOC possui atualmente 9 (nove) servidores principais que são:

- WWW – que hospeda o site *www.fagoc.br*, possui instalado o sistema operacional Linux, e é o servidor de hospedagem dos serviços Web. Nele funcionam os serviços de SSH e FTP;
- Mail – É o servidor de e-mail, configurado com **cpnel** e **mailing**, e trabalha com os serviços de SMTP, POP, IMAP;
- DNS – É o servidor que tem como função fazer a resolução de nomes na rede;
- Terminal Server – É o servidor de terminal para conexão das estações de cliente remoto;
- FagocNews – É o servidor de mailing utilizado para envio de notícias sobre a instituição e cursos;
- LTSP – É o servidor que possibilita o uso de um computador por vários terminais de acesso, onde todos os comandos rodam no servidor, mas a saída será exibida nas estações que são os thin clients;
- Router rede acadêmica – É responsável pelos serviços de redes como firewall, proxy, DHCP e LTSP. A configuração de ambas as redes são controladas por um servidor DHCP existente em cada uma das duas redes. Para adicionar um novo computador na rede, este deve ser cadastrado no servidor DHCP. Isso é necessário para diminuir o número de conflitos de IP e melhorar a administração e suporte da rede;
- Router rede administrativa – É responsável pelos serviços de redes como firewall, proxy e DHCP;
- PANDORA – É o servidor que possui o sistema acadêmico e administrativo da instituição, e é o servidor de banco de dados MySQL, onde estão configurados os serviços Web, SSH, HTTP e FTP.

O processo de configuração do Nagios pode ser demorado, mas é possível apenas com os arquivos de configuração de exemplo (instalados juntamente com a ferramenta) fazer com que ela fique ativa e funcionando. O Nagios possui uma documentação que é instalada junto com a ferramenta e pode ser acessada pela interface web para obter mais informações.

O objetivo principal para o monitoramento do Nagios é definido no arquivo de *host.cfg*. A seguir será mostrada a configuração deste arquivo e o acesso ao Nagios através de sua interface web.

O arquivo *host.cfg* apresenta a definição dos dados do servidor que será monitorado, conforme apresentado na **Listagem 1**.

Os itens que devem ser configurados neste arquivo são:

• *Use* – define o template padrão do host, no exemplo é utilizado o generic-host;

• *Host_name* – define o nome da máquina monitorada;

• *Alias* – define um apelido para a máquina monitorada;

• *Address* – define o endereço de acesso;

• *Check_command* – é o nome abreviado do comando que deverá ser usado para verificar se o cliente está funcionando ou não;

• *Max_check_attempts* – é a diretiva usada para definir o número máximo de vezes que o programa irá checar pelo serviço até reportar um erro;

• *Notification_interval* – é definido o tempo de espera para renotificar o contato sobre a situação do estado do cliente;

• *Notification_period* – define o período de tempo no qual as notificações de eventos de um cliente podem ser envidas aos contatos;

• *Notification_options* – utilizada para determinar quando a ferramenta poderá enviar notificações para o usuário administrador, onde *d* = enviar notificação em estados DOWN (inativo), *u* = enviar notificações em estados UNREACHABLE (inacessível), e *r* = enviar notificações nas recuperações, ou seja, quando o host volta a ficar ativo.

Contact_groups – define o grupo de contatos que receberão as notificações dos clientes.

Listagem 1. Host definido em host.cfg.

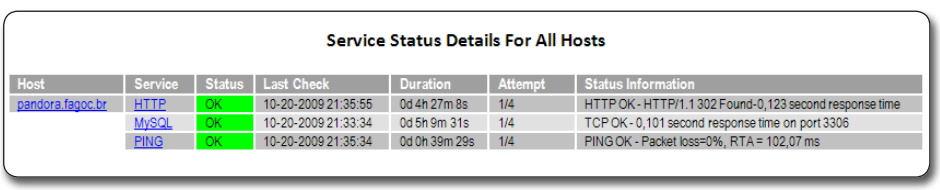
```
define host{
    use                     generic-host    ; Name of host template to use
    host_name               pandora.fagoc.br
    alias                   pandora.fagoc.br
    address                 201.39.137.212
    check_command           check-host-alive
    max_check_attempts      10
    notification_interval   120
    notification_period     24x7
    notification_options    d,u,r
    contact_groups          admins
    }
```

Após a configuração, o monitoramento pode ser acompanhado pelo acesso Web. Ao acessar o Nagios na interface web é possível obter as informações dos servidores configurados pelo menu da interface principal (**Figura 5**), que possui as opções geral (*General*), o menu para acesso às informações do monitoramento (*Current Status*), o menu de acesso aos relatórios (*Reports*) e o menu para visualizar as configurações (*System*).

Na interface principal, no menu *Current Status*, ao acessar a opção *Services* é exibido o nome do servidor (*Host*), os serviços que estão sendo monitorados (*Service*), a última verificação informando a data e a hora (*Last Check*), a duração da verificação (*Duration*), as tentativas de verificação (*Attempt*), além de informar o estado do serviço durante a verificação (*Status Information*), como representado na **Figura 7**.

No Nagios, através do menu *Reports*, na opção *Availability* são apresentados os relatórios de disponibilidade dos hosts e ativos de redes (impressoras, switch). A **Figura 8** mostra a visualização do relatório geral da disponibilidade do cliente, apresentando o tempo em que ficou ativo (UP), parado (DOWN) e inacessível (UNREACHABLE). No *Host State Breakdowns*, ainda na **Figura 8**, é possível ter um resumo do tempo de monitoramento dos serviços e o percentual de tempo do funcionamento dos serviços disponibilizados pelos servidores que estão sendo monitorados. Visualizando o relatório no período de 31 dias, foi possível verificar que nesse intervalo de funcionamento dos serviços do PANDORA, obteve-se 98.587% de disponibilidade, sendo que em 1,413% deste período os serviços ficaram parados e 0,000% inacessível.

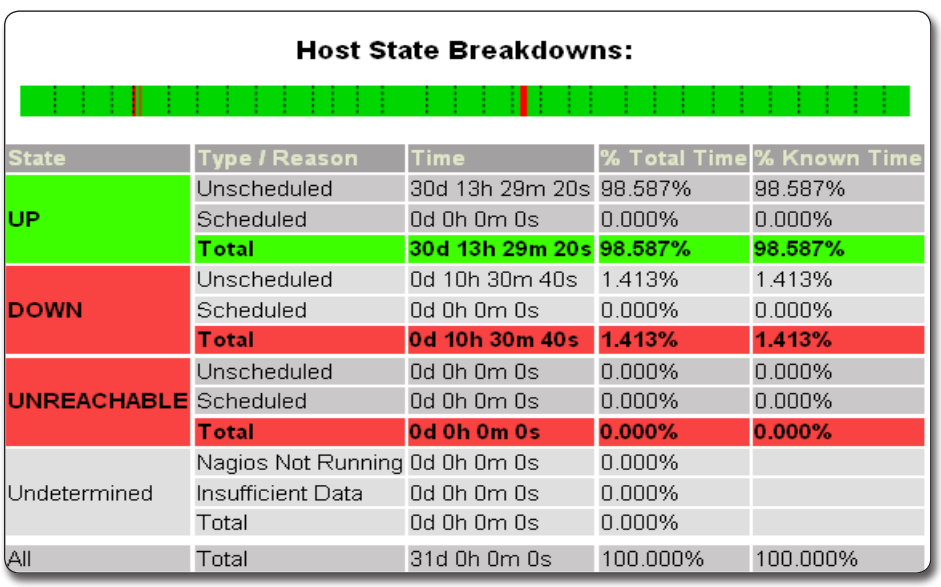
Na opção de relatórios é possível ainda detalhar o estado de funcionamento do cliente/servidor. O relatório detalhado apresenta um gráfico com o status do servidor, por período, visualizando por data, hora e dia da semana, o tempo exato em que os servidores ficaram ativos, parados ou inacessíveis. A **Figura 9** apresenta esse gráfico com o histórico do funcionamento do servidor PANDORA entre os dias 05 de outubro a 05 de novembro de 2009, onde é possível identificar nos dias 9 e 23 de outubro o período que o servidor obteve falha, ficando inativo. Quando há uma falha no servidor, é enviada uma noti-



Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
pandora.fagoc.br	HTTP	OK	10-20-2009 21:35:55	0d 4h 27m 8s	1/4	HTTP OK - HTTP/1.1 302 Found-0,123 second response time
	MySQL	OK	10-20-2009 21:33:34	0d 5h 9m 31s	1/4	TCP OK - 0,101 second response time on port 3306
	PING	OK	10-20-2009 21:35:34	0d 0h 39m 29s	1/4	PING OK - Packet loss=0%, RTA = 102,07 ms

Figura 7. Serviços monitorados pelo Nagios



Host State Breakdowns:

State	Type / Reason	Time	% Total Time	% Known Time
UP	Unscheduled	30d 13h 29m 20s	98.587%	98.587%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	**30d 13h 29m 20s**	**98.587%**	**98.587%**
DOWN	Unscheduled	0d 10h 30m 40s	1.413%	1.413%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	**0d 10h 30m 40s**	**1.413%**	**1.413%**
UNREACHABLE	Unscheduled	0d 0h 0m 0s	0.000%	0.000%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	**0d 0h 0m 0s**	**0.000%**	**0.000%**
Undetermined	Nagios Not Running	0d 0h 0m 0s	0.000%	
	Insufficient Data	0d 0h 0m 0s	0.000%	
	Total	0d 0h 0m 0s	0.000%	
All	Total	31d 0h 0m 0s	100.000%	100.000%

Figura 8. Relatório geral de disponibilidade

ficação, de acordo com a configuração feita, por e-mail e SMS para o grupo de contatos especificados no contacts_groups, e o aviso informará o estado do cliente, podendo ser WARNING (aviso) ou CRITICAL (crítico), o tipo da notificação (Notification Type), o serviço que foi verificado (Service), o host e address do cliente, e seu estado (state), além da data e hora da verificação (Date/Time).

Cada serviço de um host ou ativo é monitorado separadamente. A **Figura 10** apresenta o relatório de monitoração dos serviços, onde é mostrada a porcentagem referente ao tempo verificado, o tempo em que os serviços estiveram em estado ok, em estado de alerta, inacessível e em estado crítico.

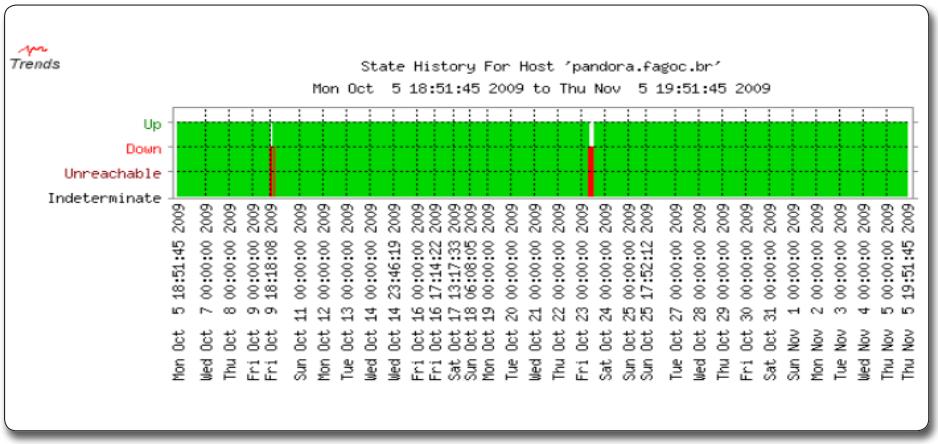


Figura 9. Gráfico do histórico de funcionamento do PANDORA

State Breakdowns For Host Services:

Service	% Time OK	% Time Warning	% Time Unknown	% Time Critical	% Time Undetermined
HTTP	98.270% (98.270%)	0.000% (0.000%)	0.000% (0.000%)	1.730% (1.730%)	0.000%
MySQL	95.389% (95.389%)	0.000% (0.000%)	0.000% (0.000%)	4.611% (4.611%)	0.000%
PING	95.609% (95.609%)	0.503% (0.503%)	0.000% (0.000%)	3.889% (3.889%)	0.000%
Average	96.423% (96.423%)	0.168% (0.168%)	0.000% (0.000%)	3.410% (3.410%)	0.000%

Figura 10. Relatório de monitoração dos serviços.

Conclusão

Este artigo buscou apresentar a importância de monitorar redes e ativos de computadores. Grandes são os desafios para o gerenciamento de uma rede de dados e seus ativos, tendo como premissa a manutenção de sua disponibilidade. Assim, com a utilização de ferramentas de monitoramento, é possível acompanhar o comportamento dos serviços disponibilizados e manter o desempenho, disponibilidade e estabilidade num ambiente computacional, podendo evitar falhas e problemas na rede.

Vimos também que com o aumento na utilização das redes, um sistema de gerenciamento de redes torna-se cada vez mais indispensável. Entretanto, muitas vezes não se sabe qual seria a melhor solução que atenderia às suas necessidades devido às diferentes características de cada ferramenta. Todas elas tratam um único assunto que é ajudar a manter o bom funcionamento da rede, ou seja, melhorando a confiabilidade e segurança dos dados e recursos disponíveis aos usuários da mesma.

Por fim, vimos que o Nagios é uma ferramenta open source de monitoramento de redes que verifica constantemente a disponibilidade do serviço, seja local ou remoto, e avisa por meio de e-mail ou celular sobre o problema ocorrido. Além disso, é possível obter relatórios de disponibilidade e configurar ações corretivas para os problemas ocorridos na rede. Com a utilização do Nagios foi possível detectar problemas com serviços de rede mais rapidamente, identificando o que poderia afetar a disponibilidade do sistema utilizado pela instituição FAGOC de forma confiável, e obtendo relatórios com informações precisas, possibilitando acompanhar o funcionamento dos serviços e da rede como um todo.



Renata Aparecida Benini
renatabenini@fagoc.br
É especialista em Manutenção e Gerência de Tecnologia da Informação pela FAGOC - Faculdade Governador Ozanam Coelho e Bacharel em Ciência da Computação também pela FAGOC.



Marcelo Santos Daibert
marcelo@daibert.pro
@msdaibert
É Mestre e Especialista em Ciência da Computação pela Universidade Federal de Viçosa, professor e coordenador do Curso de Bacharelado em Ciência da Computação da FAGOC - Faculdade Governador Ozanam Coelho, e Bacharel em Sistemas de Informação pela Faculdade Metodista Granbery. Gerente técnico da Optical Soluções em Informática Ltda.

DIAS, H. L. A importância do monitoramento de ativos de redes: Um estudo de caso com o sistema CACIC. Recife, PE: UPE, 2008. 64 f. Monografia (Bacharel em Engenharia da Computação) Universidade de Pernambuco, Recife, PE, 2008.
http://dsc.upe.br/~tcc/20082/TCC_Henrique_Dias_2008-2.pdf.

NETO, A. F., UCHÔA, J. Q. Ferramentas Livres para Monitoração de Servidores. FISL 7.0, Porto Alegre, RS, p. 149-154, abr. 2006.
http://www.ginux.ufla.br/files/artigo-ArlindoNeto,JoaquimUchoa.pdf.

PONTUAL, R. P. Projeto baseado no Nagios para monitoração de autorizadores de TEF. Recife, PE: UPE, 2009. 64 f. Monografia (Bacharel em Engenharia da Computação) Universidade de Pernambuco, Recife, PE, 2009.
http://dsc.upe.br/~tcc/20091/TCC_Rogerio_Pontual_Projeto-baseado-no-Nagios-------.pdf.

SOARES, L. F. G., LEMOS, G., COLCHER, S. Redes de Computadores: Das Lan's Man's e Wan's às Redes ATM. 2 ed. Rio de Janeiro, Ed. Campus, 1995.

TANENBAUM, A. S. Redes de Computadores. 3 ed. Rio de Janeiro, Ed. Campus, 1997.

Hospede seus sites de forma simples, rápida e ilimitada.



Plano **Básico**

- 1 domínio principal
- 50 domínios hospedados
- Transferência **ilimitada**
- Espaço **ilimitado**
- 10 contas de e-mail
- PHP, ASP e ASP.NET
- Painel de controle Plesk
- Suporte 24hs

Tudo isso por apenas

R$ **15,90** por mês
pagamento trimestral

Assine agora mesmo e descubra por que Hospedix é a hospedagem de sites preferida entre os profissionais.

- Conheça também nossos planos profissionais com registro de domínio grátis e isenção de taxas anuais enquanto for cliente.

HOSPEDIX
HOSPEDAGEM PROFISSIONAL
www.hospedix.com.br

30 DIAS GRÁTIS

Ao assinar, digite o código de desconto exclusivo para leitores desta revista e ganhe 30 dias inteiramente grátis.

RVSTMEDIA

As promoções anunciadas não são cumulativas com outras promoções e são válidas apenas para clientes Hospedix ativos.