VIRTUALIZAÇÃO
REDES VIRTUAIS

Autor: Me. Ricardo César Ribeiro dos Santos

Revisor: Luciana de Castro Lugli

INICIAR

Introdução

Algumas das aplicações em que a virtualização se destaca é a criação de serviços de rede e a realização de conexões de sistemas situados tanto no mesmo sistema hospedeiro quanto externamente. O recurso de criar servidores virtuais já é uma realidade no que se refere à prestação de serviços para empresas e clientes por meio de serviços em parques proprietários com a suíte da Apache ou até mesmo por serviços como o Amazon AWS e a suíte de serviços de nuvem do Google.

Esses serviços dependem de uma infraestrutura que possui componentes virtualizados e componentes físicos, trabalhando em harmonia para criar uma plataforma de alta disponibilidade e com desempenho otimizado.

Nesta unidade, serão detalhados os mecanismos e os componentes que dão suporte às operações de grandes *data centers* e criam redes de alto desempenho, com altas taxas de transferência e de alta confiabilidade.

Redes Virtuais

Uma das aplicações de virtualização é a criação de servidores virtuais para atender usuários de um serviço. O tráfego dos dados é gerido por componentes virtuais que replicam as funcionalidades de componentes reais, como já acontece em outros ambientes virtualizados.

A grande diferença em ambientes de redes virtuais é que, ao contrário de outros contextos de virtualização, os componentes não estão todos internos à máquina virtual, contando com componentes virtuais contidos somente dentro do *hypervisor* . Um dos exemplos mais clássicos é o *switch* virtual, que permite a comunicação de máquinas virtuais dentro do mesmo sistema computacional.

Existem duas formas principais de se conectar uma máquina virtual a outros *hosts* de rede, mediante um *switch* virtual externo ou um *switch* virtual interno. Os dois casos permitem – como os próprios nomes já indicam – conexões a *hosts* externos ao *hypervisor* , que estejam executando sobre o mesmo *hypervisor* , respectivamente.

O *Switch* Virtual Interno

No caso do *switch* virtual interno, o sistema virtualizado não é visto pelos outros *hosts* da rede, somente outras máquinas virtuais que estejam executando sobre o mesmo *hypervisor* . Dessa forma, o *hypervisor* utiliza uma área da memória para emular um *switch* , criando uma área de comunicação em memória para a rede.

O *switch* interno se comunica com a interface de rede de máquina virtual da mesma forma que uma interface de rede física se comunica com um *switch* real. A Figura 4.1 ilustra como o processo é realizado.

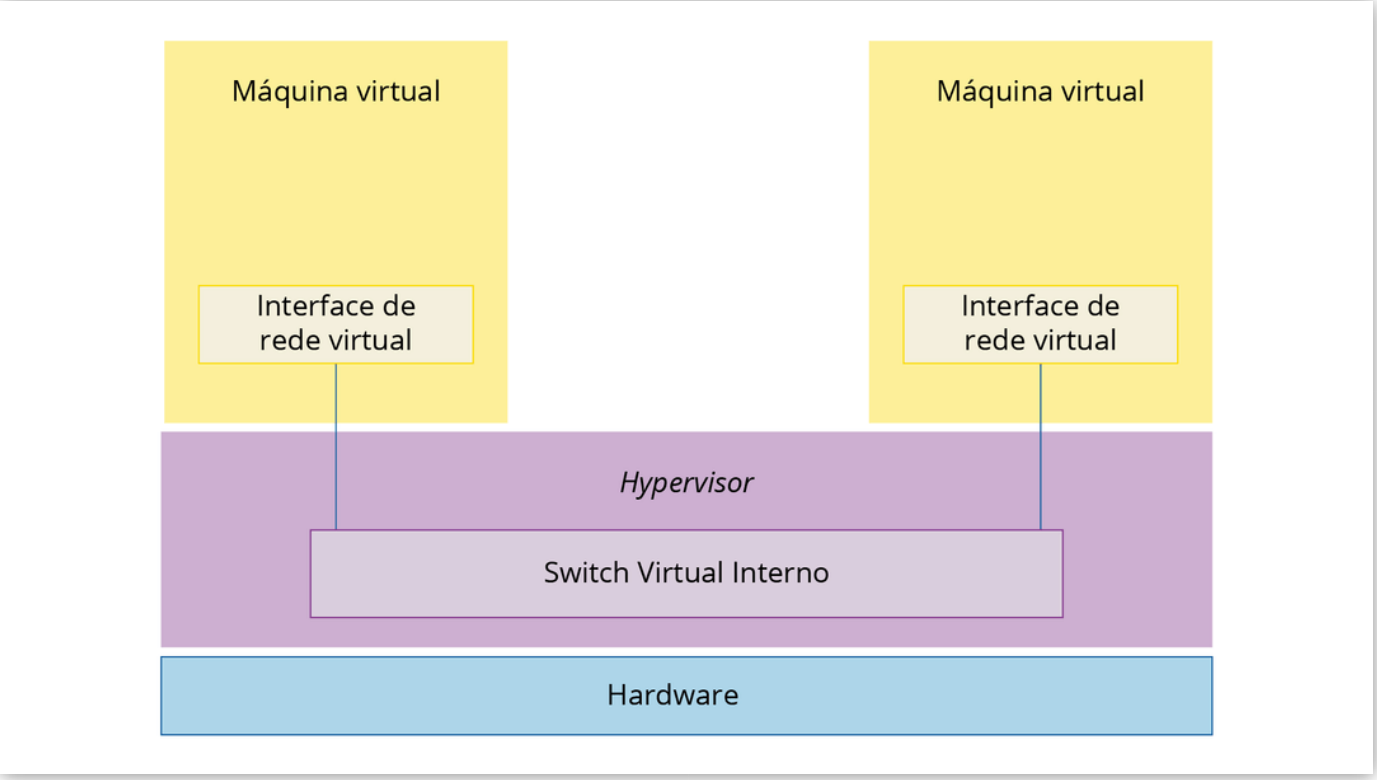


Figura 4.1 - Ligação de duas máquinas virtuais em um switch virtual interno
Fonte: Elaborada pelo autor.

O *switch* virtual interno é uma abstração para uma área de memória que permite a comunicação entre máquinas virtuais. Porém, o *hypervisor* ainda mantém válidos os protocolos de rede para as máquinas virtuais; o protocolo IP ainda se aplica, bem como é necessário instalar e configurar servidores etc.

Essa arquitetura tem algumas características interessantes. Primeiro, como toda a ligação se processa na memória, a comunicação ocorre de forma mais

veloz do que se fosse em uma rede física. Isso garante que as comunicações, caso passem por outra máquina virtual, sejam praticamente transparentes para algum *host* que esteja se conectando pela rede externa. Inclusive, é aconselhável, caso existam duas máquinas virtuais que necessitem trocar dados de forma muito intensa, que sejam instaladas no mesmo sistema computacional para otimizar a comunicação.

Outro efeito colateral é que, como o tráfego ocorre na memória, internamente ao sistema hospedeiro, a comunicação é indetectável por outros *hosts* de rede. Essa característica é positiva do ponto de vista de segurança, uma vez que um atacante não conseguiria ver o que trafega pela rede, sendo, portanto, mais difícil de ser detectado por um usuário malicioso. Porém, do ponto de vista de gerência de rede, como as ferramentas tradicionais não detectam o tráfego, erros na rede podem ser difíceis de detectar.

Para a depuração de rede em ambientes virtualizados, existem ferramentas especializadas que podem detectar e analisar esse tipo de tráfego.

O *Switch* Virtual Externo

Caso o administrador determine que uma máquina virtual tenha a possibilidade de se comunicar com uma rede externa ao servidor que contenha máquina virtual, cabe ao *hypervisor* mediar a ligação externa da interface de rede virtual.

Tal processo é feito por meio da conexão da interface de rede virtual ao *driver* de rede do servidor físico. Porém, é possível que mais de uma máquina virtual deseje se comunicar com o mundo exterior ao sistema hospedeiro. Desse modo, o *hypervisor* cria um *switch* virtual com conexão externa para abstrair a conexão de rede.

É importante notar que o conceito do *switch* virtual externo existe mesmo quando há somente uma máquina virtual que irá se comunicar com o sistema externo. No entanto, é comum simplificar o sistema e falar sobre a ligação direta da interface de rede virtual ao *driver* da placa de rede.

A conexão de máquinas virtuais à rede externa é ilustrada na Figura 4.2 a

seguir.

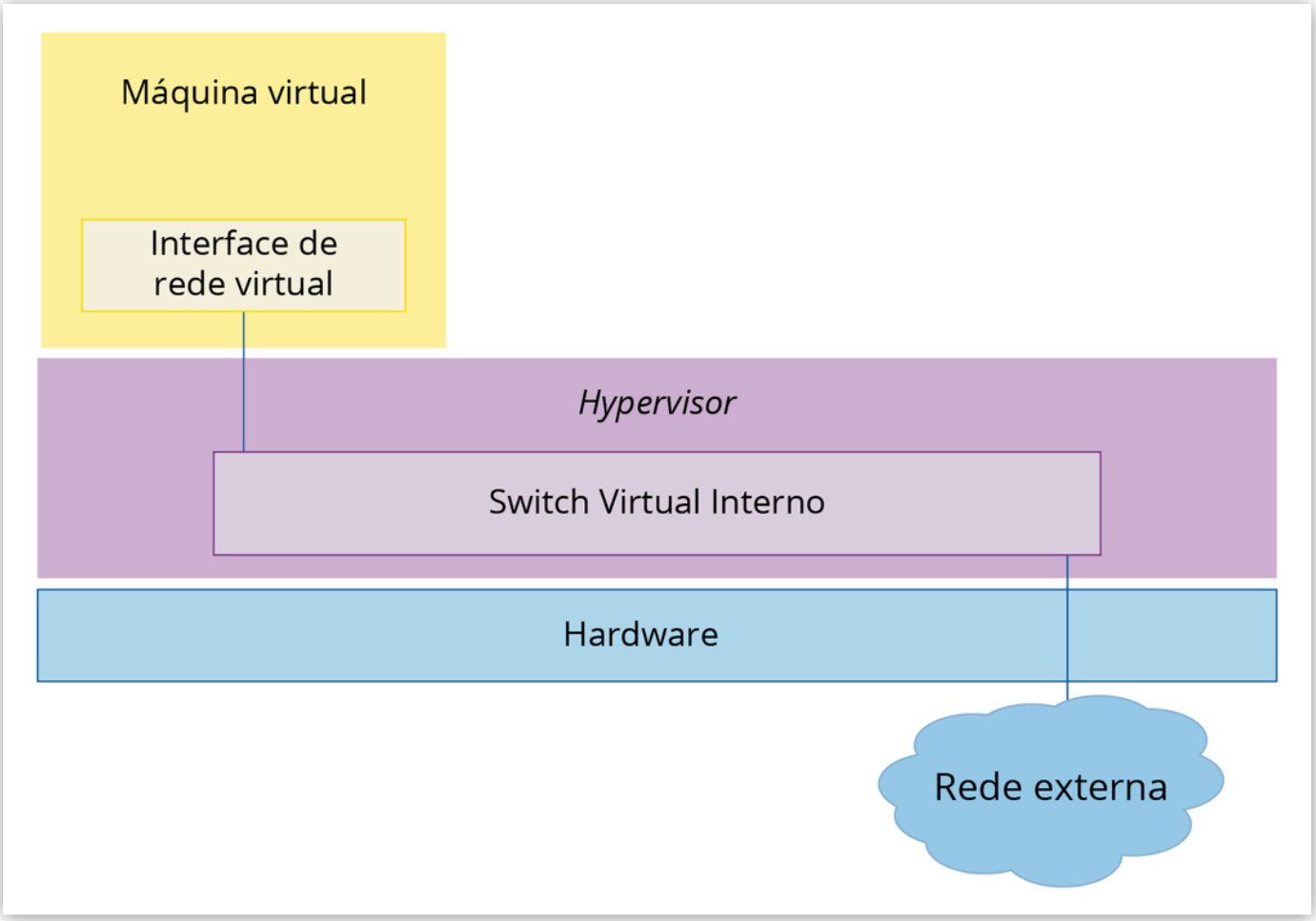


Figura 4.2 - Ligação de uma máquina virtual a uma rede externa por meio do switch virtual externo
Fonte: Elaborada pelo autor.

Os outros *hosts* de rede irão ver a máquina virtual como um *host* de rede, como se fosse uma máquina executando um servidor. Outras máquinas poderão se comunicar com o servidor virtualizado e poderão enviar requisições que serão tratadas e respondidas conforme a necessidade.

A forma de conexão dos *hosts* pode ser utilizada para a simulação de redes e testes de serviços em um ambiente controlado antes de ser colocado em operação atendendo usuários por meio da rede. Inclusive, fabricantes de ativos de rede já estão criando imagens virtuais de seus próprios equipamentos, permitindo-se que administradores de rede simulem o ambiente, realizem as configurações dos equipamentos e, posteriormente, repliquem as configurações para os ativos de rede reais.

praticar

Vamos Praticar

Máquinas virtuais que conectam somente ao *switch* virtual não têm como acessar a rede externa. Consequentemente, essa máquina virtual só pode ser acessada por outras máquinas virtuais que estejam conectadas ao *switch* virtual, não por fontes externas. Como nesse modelo a partição-pai não controla a interface de rede física, o *hypervisor* não administra a entrada e a saída de rede.

> *PORTNOY, M.* ***Virtualization Essentials*** *. 1. ed. Tradução: Ricardo César Ribeiro dos Santos. Indianapolis, Canada: Indiana Published, 2012.*

Com relação a redes virtuais, assinale a alternativa correta.

- **a)** O *switch* virtual é um componente físico que deve ser conectado ao hospedeiro da máquina virtual.
- **b)** Uma máquina virtual pode somente estar conectada a um *switch* virtual interno ou a um *switch* virtual externo, nunca a ambos.
- **c)** Uma interface de rede virtual pode somente estar conectada a um *switch* virtual interno ou a um *switch* virtual externo, nunca a ambos.
- **d)** O *switch* virtual externo, por padrão, já protege a máquina virtual contra ataques maliciosos.
- **e)** O *switch* virtual tem limite de portas. Quando cria um dispositivo desses, o administrador do sistema deve especificar se tem 8, 16, 24 ou 48 portas.

Sistemas Operacionais e a Disponibilidade

Nos serviços atuais disponíveis em rede, existe a preocupação de que, sempre que o usuário desejar, consiga se conectar com os servidores do serviço e realizar as transações de que necessita. Essa é a preocupação com a disponibilidade do serviço em rede, que, atualmente, é medido em percentual de tempo disponível, conforme ilustra a Tabela 4.1 a seguir.

Disponibilidade do Serviço	Tempo Anual Indisponível
99%	3,65 dias
99,9%	8,8 horas
99,99%	53 minutos
99,999% (“ *five nines”)*	5,3 minutos

Tabela 4.1 - Disponibilidade e tempo indisponível
Fonte: Adaptada de Portnoy (2012, p. 229).

Para que esses serviços possam atender aos parâmetros de disponibilidade, é necessário implementar um *data center* que crie um ambiente suficientemente estável para que tanto a infraestrutura da rede como os próprios servidores consigam lidar com o tráfego de rede e atender todos os clientes que enviam requisições.

reflita

Reflita

A quantidade de tempo que um serviço é acessível para os clientes é fundamental para empresas. Caso os usuários de um determinado serviço não consigam acessá-lo, ocorrem prejuízos financeiros e perda de usuários. Nota-se que uma disponibilidade aparentemente alta (de 99%) de um serviço ainda pode implicar uma indisponibilidade de 3,65 dias ao ano, o que corresponde a pouco mais da metade de uma semana ao ano. Atualmente, existem casos em que serviços on-line ficaram indisponíveis por menos tempo que isso e assumiram uma relevância tão grande, que foram reportados nos noticiários. Outras situações podem envolver forças alheias a razões técnicas, como as decisões judiciais que tiraram o WhatsApp do ar no Brasil e fizeram milhares de usuários migrarem para serviços similares. Disponibilidade é um tema delicado, que é tratado com muita seriedade por empresas de todos os ramos que disponibilizam serviços on-line.

Fonte: Portnoy (2012).

Na verdade, o problema de disponibilidade não tem uma única causa e uma única solução. É complexo e deve ser atendido de várias formas, tanto profilática (aplicando as medidas preventivas necessárias) quanto ativa, resolvendo os problemas quando aparecem.

O grande problema que a disponibilidade tenta resolver é a negação de serviços aos usuários de uma plataforma e, frequentemente, não é somente com *software* mais estável e *hardware* mais robusto que se resolve.

Uma das grandes armas em prol da disponibilidade é a redundância. Ela é utilizada a fim tanto de realizar proteção contra faltas quanto aumentar a capacidade de processamento de um sistema.

Em *hardware* , são utilizados sistemas redundantes, desde a instalação de múltiplos coolers, para garantir que o sistema fique refrigerado em caso de falha de algum deles, até a instalação de mais de uma fonte de alimentação. Porém, existem formas de utilizar o *software* para a criação de sistemas mais estáveis, com destaque para a criação de *clusters* , ou clusterização.

No caso da clusterização, duas ou mais máquinas são ligadas em rede, realizando compartilhamento de recursos e armazenamento de *software* . Caso um nó do sistema venha a sofrer uma falha, outro nó entra no lugar sem que o usuário perceba.

Essa configuração é realizada em nível de sistema operacional – ou de *hypervisores* . Cada um dos elementos executa sobre uma plataforma física, mas também se comunica por meio da rede para negociar a execução em conjunto. A Figura 4.3 contém um diagrama que ilustra esse sistema em funcionamento, do ponto de vista do sistema operacional.

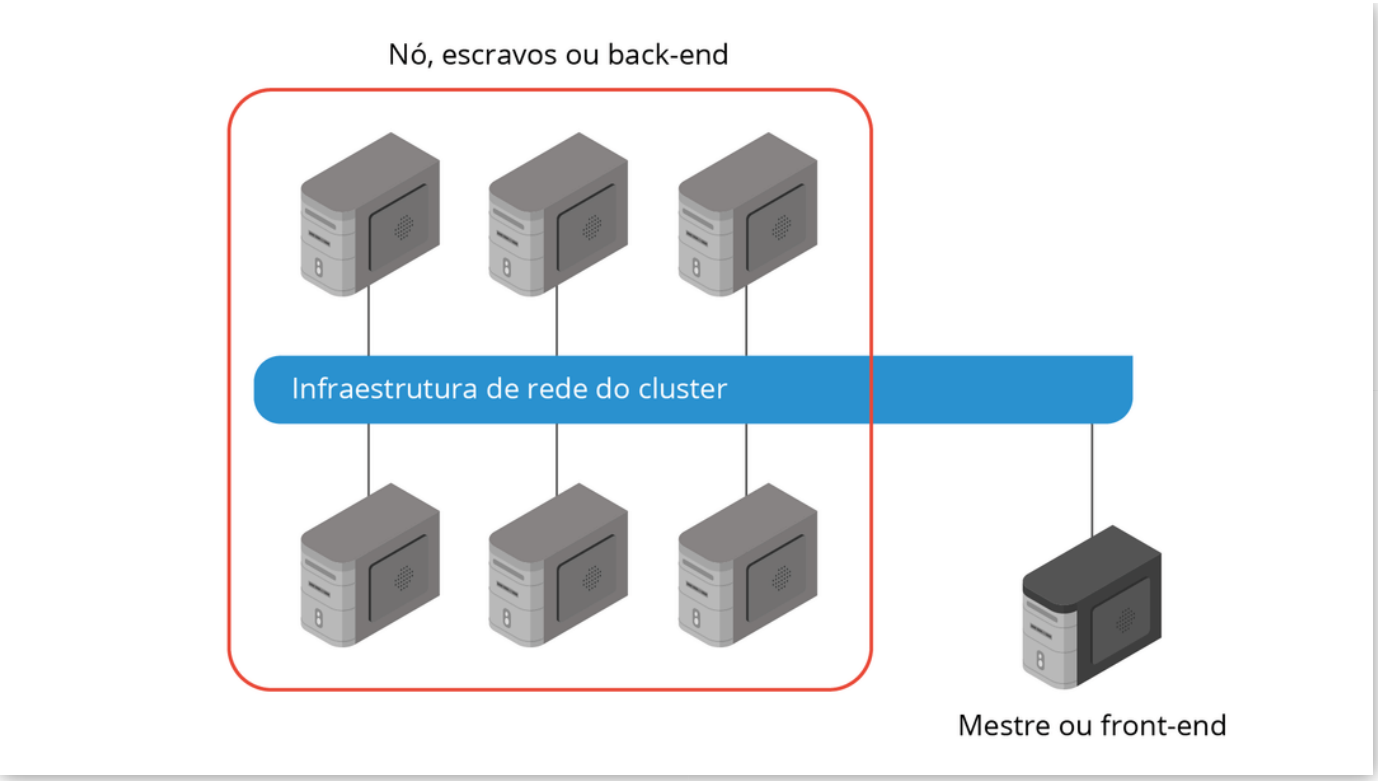


Figura 4.3 - Arquitetura de rede de cluster mostrando os nós escravos e o nó mestre

Fonte: Adaptada de Fekade e Maksymyuk (2016, p. 3.456).

Grosso modo, em um *cluster* , várias máquinas se comportam como se fossem uma só, respondendo a solicitações e criando arquivos de maneira unificada. A arquitetura conta com um grupo de computadores que age como o *back-end* , realizando as operações necessárias para a aplicação. Entretanto, existe um nó com uma função especial, o nó mestre, que é o responsável por lidar com as conexões que vêm dos usuários.

O *cluster* garante a tolerância a falhas de sistemas de *hardware* , caso um deles precise de manutenção, os outros servidores podem absorver a carga de processamento excedente. Esse mecanismo de balanceamento de carga garante que todos os nós consigam operar de maneira autônoma e suprir as deficiências dos demais em caso de necessidade.

O compartilhamento de sistemas também garante que dados armazenados possam ser replicados, garantindo-se a redundância à falha do armazenamento.

Outra estratégia de criação de serviços de redundância é a criação de

sistemas passivos. Existe um sistema ativo, que processa as requisições dos usuários, e um outro, passivo, que se mantém sincronizado com o ativo.

Em caso de falha do sistema ativo, o passivo toma seu lugar de maneira transparente para o usuário, a fim de manter a disponibilidade do serviço o mais alta possível.

É necessário que o sistema tenha todos os equipamentos replicados, o que pode acarretar custos duplicados para a criação do parque tecnológico. Entretanto, a virtualização pode auxiliar a manter os custos de implantação mais baixos e oferecer a mesma funcionalidade ao administrador do sistema.

praticar

Vamos Praticar

Faltas prolongadas de serviços em companhias que dependem de tais *softwares* acarreta perda de produtividade, que se soma aos riscos de negócio e provoca prejuízos. Quando companhias provedoras desses serviços consideram os benefícios de virtualização, o aumento de disponibilidade está entre os primeiros lugares da lista.

> *PORTNOY, M.* ***Virtualization Essentials*** *. 1. ed. Tradução: Ricardo César Ribeiro dos Santos. Indianapolis, Canada: Indiana Published, 2012.*

Com relação à disponibilidade de sistemas, assinale a alternativa correta.

- **a)** A disponibilidade de sistemas é independente da virtualização. É possível alcançar o mesmo resultado com recursos de *hardware* e *software* .
- **b)** A virtualização não contribui para o aumento de disponibilidade do serviço. Essa métrica tem relação somente com a infraestrutura de rede.

- **c)** Tipicamente, os usuários têm alta tolerância a falhas de serviço. Dessa forma, quando há uma falta, raramente empresas sofrem efeitos adversos.
- **d)** As faltas de serviço sempre acontecem por questões técnicas sob controle das empresas.
- **e)** Um *cluster* é o conjunto de computadores interligados, que, por si só, já garantem disponibilidade total.

Gerência de Tráfego e Isolamento de Tráfego

Para garantir a disponibilidade dos serviços e servidores, a infraestrutura de rede deve suportar alta disponibilidade também. Para que isso seja possível, a arquitetura da infraestrutura de rede deve ser executada de maneira eficiente.

Na busca de criar uma estrutura de rede que atenda às necessidades de disponibilidade para estruturas de *data center* , foram desenvolvidos dois paradigmas para projeto de redes, um derivado do outro. O modelo *core-access* , desenvolvido pela Cisco, lançou as bases sobre as quais o modelo *Spine and Leaf* posteriormente foi elaborado.

Neste tópico, serão abordadas as características desses dois modelos e como podem beneficiar a estrutura de *data centers* .

Ativos de Rede Necessários

Todo ambiente de rede deve contar com um roteador para realizar a conexão da rede interna com a rede externa. Em uma conceituação bastante simplificada, um roteador delimita uma rede. Por essa razão, quando se deseja conectar uma estrutura qualquer à internet, utiliza-se um roteador conectado a um cabo pelo qual o provedor disponibiliza o serviço para o cliente.

É comum, por exemplo, no contexto residencial, os provedores de internet fornecerem ao cliente uma terminação de um cabo de serviço – seja fibra óptica, cabo coaxial ou até mesmo um cabo par trançado – e conectá-lo a um roteador. Somente a montante do roteador, que os equipamentos da rede residencial são instalados, conectados mediante cabos de rede ou por rede *wireless* .

saiba mais

Saiba mais

Ambas as arquiteturas que serão apresentadas nesta seção são modelos amplamente aceitos pela comunidade de desenvolvedores de soluções de redes de computadores. Entretanto, outros tipos de arquitetura estão disponíveis para implementação, caso sejam necessários. Um exemplo é a arquitetura de rede Cisco Digital Network Architecture, que foi criada pensando-se em ambientes de *deep learning* e inteligência artificial.

Para saber mais, acesse o link a seguir.

ACESSAR

Ao contrário do que ocorre no contexto residencial, os roteadores para *data centers* não têm interface *wireless* , somente sendo acessíveis pelo meio cabeado – fibras ópticas, cabos metálicos ou até mesmo cabos especializados para configuração dos ativos.

Modelo *Core-Access*

Nesse modelo, a rede é dividida em três camadas: *core* , agregação e acesso. A camada de *core* fica ligada diretamente à rede do provedor de internet e define o ponto de acesso da rede do cliente à rede externa. A camada de acesso conecta os usuários finais à rede corporativa, provendo segurança da rede. A camada de agregação realiza a conexão entre as duas camadas.

O modelo é ilustrado na Figura 4.4 a seguir.

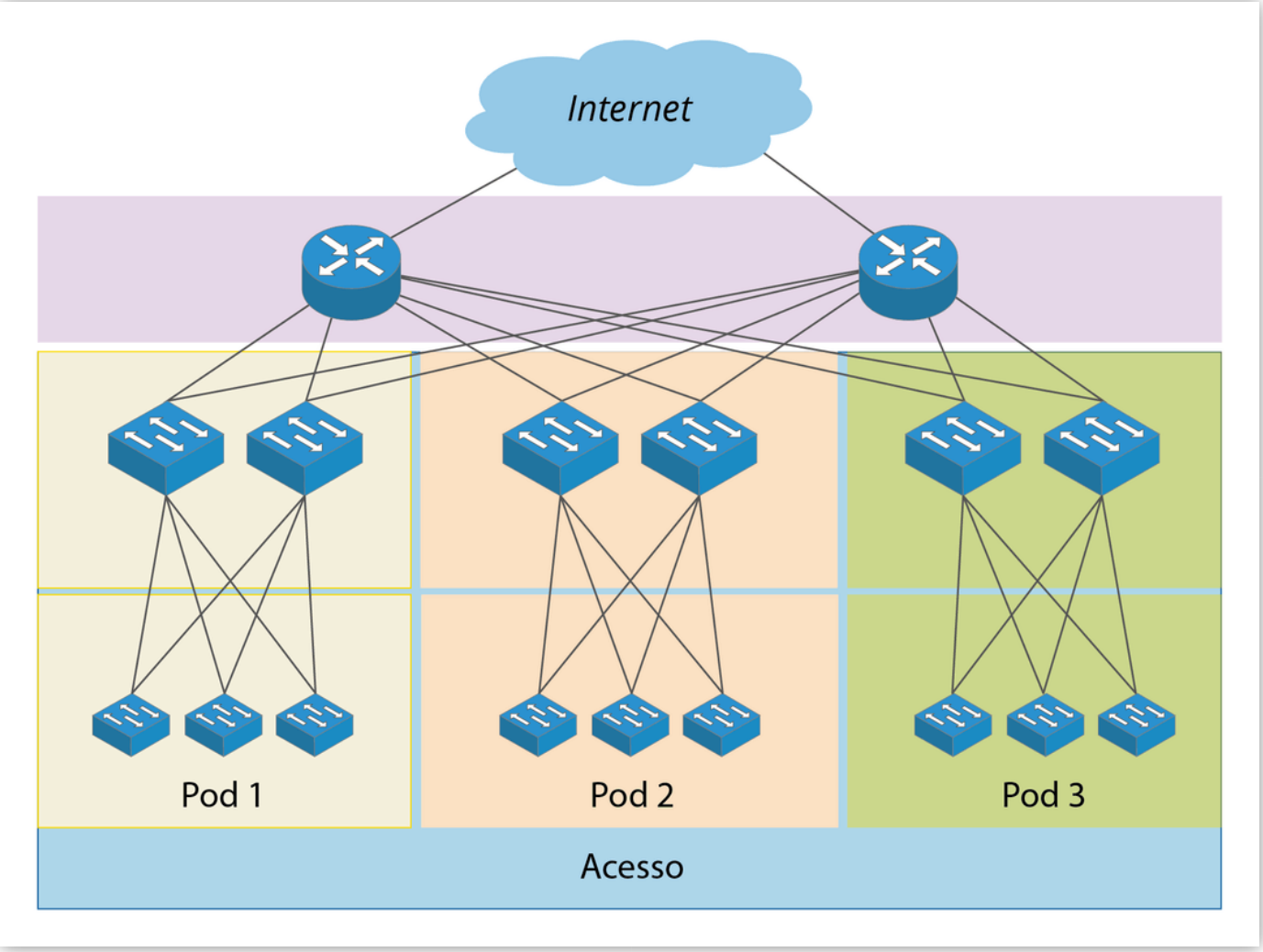


Figura 4.4 - Diagrama da arquitetura de rede core-access
Fonte: Adaptada de Cisco (2016, p. 3).

A camada de *core* é ligada à rede externa preferencialmente, de maneira redundante. A próxima camada, agregação, precisa realizar a comunicação entre os dispositivos que estão ligados a ela da maneira mais rápida o possível, a fim de ligar a camada de acesso à de *core* .

Entretanto, formam-se os *pods* , que são agrupamentos de *switches* de acesso ligados a um mesmo grupo de *switches* de agregação. Cada *pod* é isolado dos demais, sendo que a comunicação entre *pods* somente é possível por meio dos ativos de rede da camada de *core* . Essa característica isola o tráfego entre os *pods* e garante que os pacotes só alcancem os nós que devem alcançar.

Todavia, já foi visto que é possível criar *switches* virtuais no *hypervisor* , de forma a conectar mais de uma máquina virtual – com acesso externo ou não. Assim, toda a estrutura de rede das camadas de acesso pode ser virtualizada, simplificando a estrutura de rede e criando uma nova topologia.

Modelo *Spine-And-Leaf*

Com a virtualização dos ativos de rede correspondentes às camadas de acesso e agregação, a topologia de rede pode ser revista e simplificada. Ademais, com a utilização de máquinas virtuais, a comunicação entre os *pods* do modelo *core-access* se intensifica muito – seja por causa da clusterização de servidores físicos ou de servidores de tarefas que precisam trocar mensagens periodicamente.

Para sanar todos esses problemas, foi criado o modelo *spine-and-leaf* , em que existe uma espinha dorsal que conecta todos os ativos, chamados de folhas, nesta topologia, ilustrada na Figura 4.5 a seguir.

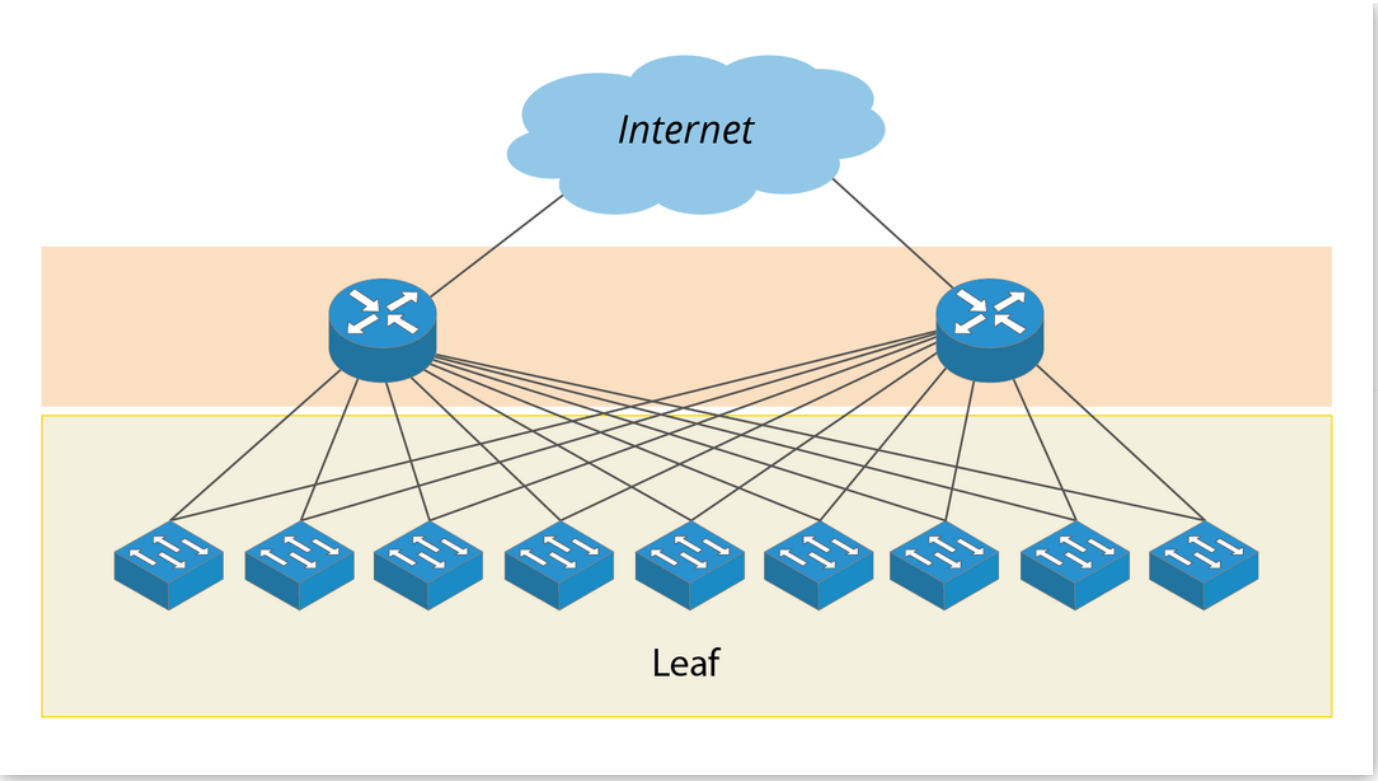


Figura 4.5 - Diagrama da arquitetura de rede spine-and-leaf
Fonte: Adaptada de Cisco (2016, p. 4).

Todos os ativos de rede da camada *leaf* são conectados aos dispositivos da camada *spine* . Esse tipo de topologia é denominado *full mesh* , que corresponde à ligação de todos os ativos de uma camada com todos os ativos de outra.

Com essa nova topologia, a comunicação entre os ativos da camada *leaf* é potencializada, uma vez que somente dois saltos são necessários, no máximo, para ir de um ponto da rede até outro ponto qualquer.

praticar

Vamos Praticar

O *data center* dá as fundações à tecnologia moderna, cumprindo um papel fundamental em expandir as capacidades de empresas. A arquitetura tradicional do *data center* utiliza uma arquitetura de três camadas, [a arquitetura *core-access*]. [...] Com a utilização de servidores virtualizados, as aplicações são instaladas de maneira distribuída [...], e o tráfego deve ter latências baixas e previsíveis.

> *CISCO.* ***Cisco Data Center Spine-and-Leaf Architecture*** *: Design Overview. Tradução: Ricardo César Ribeiro dos Santos. San Jose: Cisco Systems, 2016.*

Com relação à gerência e isolamento de tráfego, assinale a alternativa correta.

- **a)** A arquitetura *spine-and-leaf* isola o tráfego de rede em *pods* , que são agrupamentos de nós que estão ligados ao mesmo *switch* de acesso.
- **b)** A arquitetura *core-access* é dividida em três camadas, em que a mais superior é ligada à rede externa.
- **c)** A utilização de arquiteturas de rede predefinidas garante que as latências e o número de saltos na rede sejam bem definidos.
- **d)** Ao escolher uma arquitetura de rede, não se deve levar em consideração as aplicações que serão executadas pelo parque tecnológico.
- **e)** Ao utilizar arquiteturas de rede e isolar o tráfego, não é mais necessário considerar as taxas de transmissão dos meios físicos.

Disponibilidade para a Virtualização

Em tópicos anteriores, foram vistas tecnologias e técnicas para a criação de um ambiente que consiga manter a disponibilidade de um *data center* tão alta quanto possível. Agindo sobre a infraestrutura de rede e aplicando os conceitos de replicação de *hardware* e de *software* , é possível criar um sistema de alta disponibilidade, estável e eficiente.

Com o advento da virtualização, foi possível criar *backups* de servidores em serviço e criar *backups* secundários de servidores em operação sem necessitar de replicação de *hardware* .

Uma das primeiras linhas de frente para garantir que os serviços fossem entregues de forma estável para os clientes foi a criação de *backups* das máquinas virtuais. Como uma máquina virtual nada mais é do que um conjunto de arquivos, é possível criar uma cópia da máquina virtual, que possa ser colocada em serviço caso aconteça algo com a principal.

Ressalta-se que, para que essa técnica seja eficaz, é necessário que exista uma rotina de *backup* regular, de forma a evitar que configurações, dados ou *softwares* fiquem muito desatualizados em relação ao ambiente em produção.

Caso haja esse descompasso, pode ser que, em uma eventualidade, configurações dos clientes se percam e seja preciso realizar retrabalho para restaurar o estado do *backup* para o do ambiente em produção.

Essa é a técnica mais básica de todas, sendo que depende de cópias manuais (ou por meio de uma rotina automatizada do sistema operacional) para ser realizada. Também é importante notar que, enquanto o *backup* da máquina virtual é restaurado, é possível que os usuários do serviço não consigam se conectar ainda. Isso pode acontecer principalmente se uma máquina for corrompida e outra estiver se inicializando.

No ambiente virtualizado, também é possível utilizar um *backup* passivo dos servidores que estão executando. Nesse caso, uma máquina virtualizada executa em segundo plano e assume o controle da aplicação caso o servidor principal não esteja disponível.

Essas duas técnicas são uma transposição direta das técnicas utilizadas em *data centers* para o contexto de virtualização. O conceito de *cluster* , entretanto, deve ser adaptado para poder ser aplicado no contexto de virtualização.

No caso de um *cluster* de virtualização, a ligação entre as máquinas é feita mediante os *hypervisores* , e não do sistema operacional. Ou seja, as máquinas virtuais rodam sobre um grupo de sistemas computacionais abstraídos por meio de um grupo de *hypervisores* . Essa arquitetura é representada na Figura 4.6 a seguir.

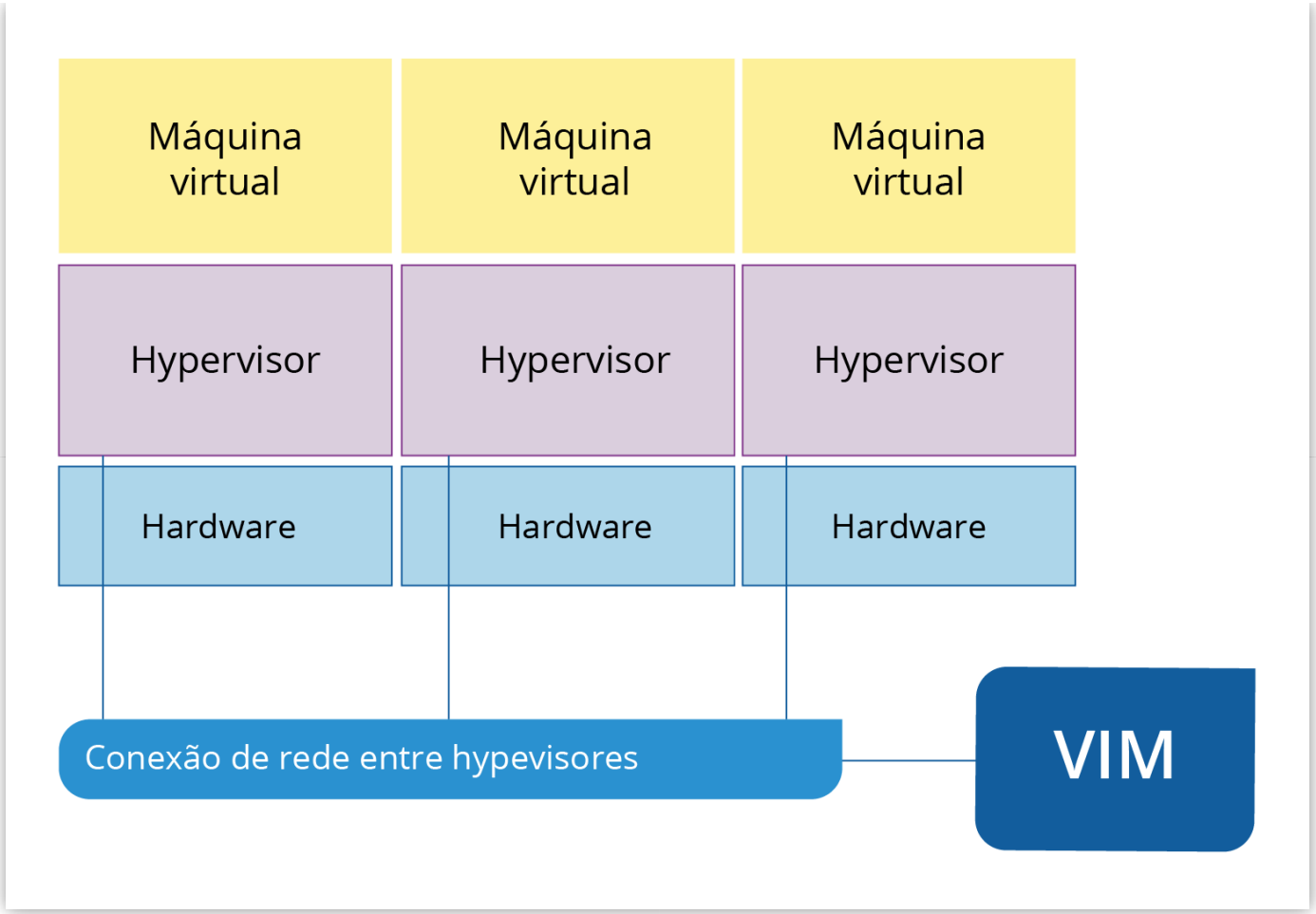


Figura 4.6 - Criação de um cluster por meio dos hypervisores
Fonte: Adaptada de Fekade e Maksymyuk (2016, p. 3.456).

Os *hypervisores* se comunicam para trocar informações e manter a estrutura de servidores virtuais sincronizada entre si. Ao invés de um nó mestre, entra uma VIM (*Virtual Infrastructure Manager*), que é responsável por agrupar mensagens sobre os atributos de todos os dispositivos da rede.

Dessa forma, mais de uma máquina virtual pode ser consolidada em um único ambiente físico, com a possibilidade de realizar *backups* e utilizar servidores virtualizados. Porém, é necessário planejar como executar o *cluster* : caso um sistema computacional tenha três máquinas virtuais executando sobre si e essas três máquinas forem ligadas em *cluster* , o máximo de desempenho que é possível se alcançar é o desempenho da máquina hospedeira.

praticar

Vamos Praticar

Se um servidor físico ficar *offline* , [...] um único serviço é impactado, em um sistema hospedeiro, entretanto, executando múltiplas máquinas virtuais, vários serviços podem ser impactados. Infraestruturas virtuais têm que ter a habilidade de se recuperar automaticamente e recuperar a carga de serviços com mais velocidade que infraestruturas físicas.

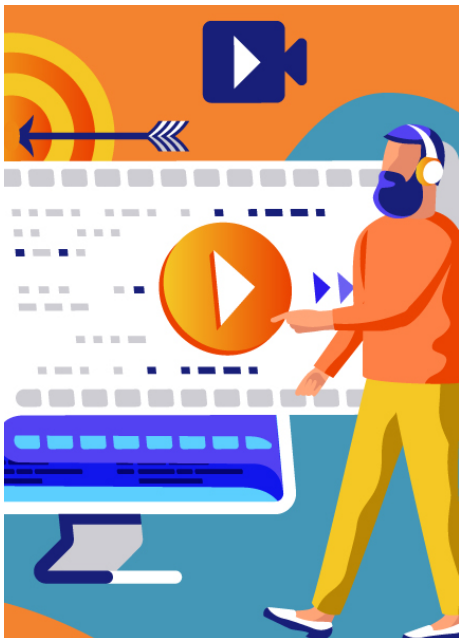
> *PORTNOY, M.* ***Virtualization Essentials*** *. 1. ed. Tradução: Ricardo César Ribeiro dos Santos. Indianapolis, Canada: Indiana Published, 2012.*

Quanto à disponibilidade em virtualização, assinale a alternativa correta.

- **a)** A criação de serviços *backup* , que entram em operação quando os principais ficam indisponíveis, em virtualização é muito custosa; logo, essa técnica não é utilizada.
- **b)** Ao criar servidores virtuais *backup* , não é uma boa prática clonar máquinas virtuais existentes. O ideal é sempre criá-las e reconfigurá-las quantas vezes for necessário.
- **c)** Não é aconselhável a criação de mais de um servidor virtual hospedado em uma única máquina física, já que máquinas virtuais não devem concorrer por recursos.
- **d)** Uma vez que uma máquina virtual seja clonada, não é necessário criar mais *backups* das máquinas virtuais.
- **e)** Para a criação de servidores virtuais de maneira eficiente, é aconselhável criar uma rotina de *backups* .

indicações

Material Complementar



FILME

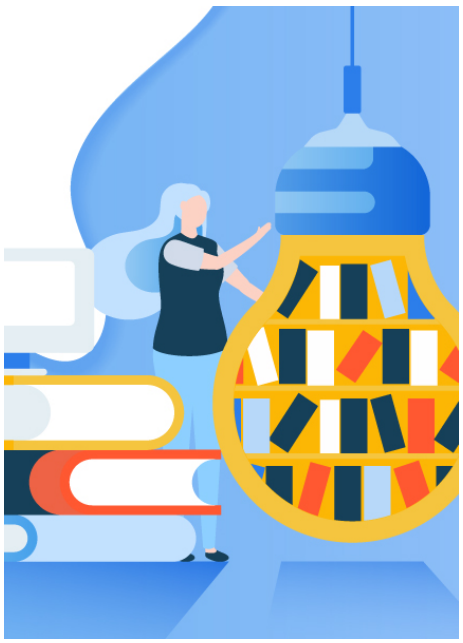
A Rede Social

Ano: 2010

Comentário: Este filme retrata o início do Facebook, com foco em seu criador, Mark Zuckerberg. O interessante de se observar no filme são as etapas no desenvolvimento de uma aplicação em nuvem e como uma companhia se estrutura ao redor de uma aplicação deste tipo.

Para conhecer mais sobre o filme, acesse o *trailer* a seguir.

TRAILER



LIVRO

Virtualização: Tecnologia Central do Datacenter

Editora: Brasport

Autor: Manoel Veras

ISBN: 8574527610

Comentário: Um livro modular que detalha como a virtualização se insere em *data centers* . Abordando tanto o aspecto teórico quanto a prática de virtualização, o livro oferece para o leitor um panorama sobre a tecnologia e as técnicas utilizadas para a implantação de servidores virtuais.

conclusão

Conclusão

Nesta unidade, foram vistas as formas de criação de redes virtuais e como é possível interligar *hosts* reais e virtuais por meio dessas redes, que utilizam interfaces de redes físicas e áreas de memória para realizar a consolidação de nós ligados em rede e máquinas virtuais distribuídos em localizações diversas.

Essas técnicas são vitais para a criação de serviços capazes de atender usuários mediante redes locais ou serviços em nuvem. Criando uma plataforma desse tipo, é possível manter *hardware* e *software* otimizados e executando serviços de maneira confiável.

Ao disponibilizar essa infraestrutura para os usuários, entretanto, é preciso sempre pensar nos requisitos de segurança, seja contra a ação de usuários maliciosos ou contra falhas de *hardware* . Nesse quesito, é sempre necessário atentar aos requisitos de segurança de rede e redundância de sistemas e de dispositivos para que o serviço possa se manter de maneira satisfatória.

referências

Referências Bibliográficas

CISCO. **Data Center Spine-and-Leaf Architecture** : Design Overview. Cisco White Papers. Disponível em: https://www.cisco.com/c/en/us/products/collateral/switches/nexus-7000-series-switches/white-paper-c11-737022.pdf . Acesso em: 03 jan. 2020.

FEKADE, B.; MAKSYMYUK, T.; JO, M. Clustering hypervisors to minimize failures in mobile cloud computing. **Wireless Communications and Mobile Computing** , Danag, vol. 16, fasc. 18, 3455-3465, fevereiro, 2017.

PORTNOY, M. **Virtualization Essentials** . Indianapolis, Canada: Indiana Published, 2012.