

**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
SUL DE MINAS GERAIS
Câmpus Inconfidentes

MARCIO FELICIANO DO PRADO

**VMWARE X XEN:
UM ESTUDO COMPARATIVO DA EFICIÊNCIA ENERGÉTICA**

INCONFIDENTES-MG

2013

MARCIO FELICIANO DO PRADO

**VMWARE X XEN:
UM ESTUDO COMPARATIVO DA EFICIÊNCIA ENERGÉTICA**

Trabalho de Conclusão de Curso apresentado como pré-requisito de conclusão do curso de Graduação Tecnológica em Redes de Computadores no Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – Câmpus Inconfidentes, para obtenção do título de Tecnólogo em Redes de Computadores.

Orientador: Prof. André Luigi Amaral Di Salvo

INCONFIDENTES-MG

2013

MARCIO FELICIANO DO PRADO

**VMWARE X XEN:
UM ESTUDO COMPARATIVO DA EFICIÊNCIA ENERGÉTICA**

Data de aprovação: ____ de _____ 2013

Orientador: Prof. André Luigi Amaral Di Salvo (IFSULDEMINAS – Câmpus Inconfidentes)

Prof. M. Sc. Marcelo Augusto dos Reis (IFSULDEMINAS – Câmpus Inconfidentes)

Prof. Bruno Amarante Couto Rezende (IFSULDEMINAS – Câmpus Inconfidentes)

Dedicatória

Dedico este trabalho a minha mãe, Maria das Graças Feliciano do Prado, ao meu pai, João Domingos do Prado, a minha irmã Rosangela Feliciano do Prado e a minha noiva Flávia Tatiane Fernandes, que me incentivaram durante todo o curso.

Agradecimentos

Primeiramente, agradeço a Jesus Cristo por abençoar-me sempre e por ter deixado Nossa Senhora interceder por mim, fazendo com que eu conseguisse passar no vestibular.

Agradeço meus professores e, especialmente, ao professor André Luigi Amaral Di Salvo que aceitou me orientar.

Agradeço ao IFSULDEMINAS – Reitoria, que me concedeu um incentivo à qualificação (PIQ) que foi muito útil.

Por fim, agradeço mais uma vez Jesus Cristo, pelo dom do estudo.

Epígrafe

"Porque Deus amou o mundo de tal maneira que deu o seu Filho unigênito, para que todo aquele que nele crê não pereça, mas tenha a vida eterna." João (3-16).

Resumo

A virtualização de servidores é uma técnica antiga que permite a execução de diversos sistemas operacionais em um único servidor. Pode ser realizada de diferentes maneiras, cada uma com seus prós e contras. Na arquitetura x86, as opções de virtualização alteram o nível de privilégios (anéis) padrão. As soluções baseadas em hipervisores incluem a virtualização completa e a paravirtualização. Além destes dois tipos ainda existe a virtualização assistida por hardware, que foi utilizada neste trabalho, implementada com os dois hipervisores mais conhecidos VMware e Xen.

Este trabalho teve como objetivo, avaliar quantitativamente, qual destes hipervisores é mais eficiente quanto a economia de energia elétrica. Inicialmente foram monitorados 3 serviços instalados em um servidor sem a utilização das técnicas de virtualização. Os serviços foram: (i) Apache; (ii) Internet Information Services (IIS); (iii) Active Directory (AD). Em seguida foram monitorados os hipervisores VMWare ESXi 5.1 e XCP 1.6, ambos com 6 servidores virtuais cada um (sendo 03 serviços Apache, 02 serviços IIS e 01 serviço AD). Comprovou-se consideravelmente economia de energia elétrica quando utilizada técnicas de virtualização, independente do hipervisor utilizado. Houve uma diferença favorável ao VMware ESXi 5.1 de 0,59 kWh. Essa diferença é aparentemente pequena, mas, se o VMware ESXi 5.1 mantivesse o mesmo padrão de consumo das 72 horas, ele poderia ficar ligado 12 horas a mais que o XCP 1.6.

Palavras-chave – Energia, otimização, virtualização, VMware, Xen, Servidores.

Abstract

Server virtualization is an ancient technique that allows you to run multiple operational systems on a single server. It can be performed in different ways, each of them with its pros and cons. In the x86 architecture, the virtualization options alter the privilege standard level (rings). Solutions based on hypervisors include full virtualization and paravirtualization. Besides these two types, there is also a hardware-assisted virtualization, which was used in this experiment and implemented with the two most popular hypervisors: VMware and Xen.

This study aimed to assess quantitatively which of these hypervisors is more efficient when it comes to saving electricity. Initially services were monitored 3 installed on a server without the use of virtualization techniques. The services were: (i) Apache, (ii) Internet Information Services (IIS) and (iii) Active Directory (AD). Then, VMWare ESXi 5.1 hypervisors and XCP 1.6 were monitored, both with six virtual servers each (being 03 Apache services, 02 services and IIS service 01 AD). From this experience, it was proved that there is a considerable saving of electricity when the virtualization techniques were used, regardless of the hypervisor used in the experiment. There was also a difference in favor of VMware ESXi 5.1 to 0.59 kWh. Although this difference is apparently small, if the VMware ESXi 5.1 maintain the same consumption pattern of the 72 hours of the experiment, it could stay on 12 hours more than the XCP 1.6.

Keywords - Energy, optimization, virtualization, VMware, Xen, Servers.

Sumário

1. Introdução.....	11
2. Revisão literária.....	12
2.1 Virtualização de servidores.....	12
2.1.1 Breve Histórico.....	12
2.1.2 Termos da Virtualização de Servidores.....	12
2.1.3 Tipos de Hipervisores	13
2.2 Vantagens	13
2.2.1 Otimização do Hardware	13
2.2.2 Economia de Energia	14
2.3 Desvantagens	15
2.4 Nível de privilégio da arquitetura X86.....	16
2.5 Tipos de virtualização de servidores	17
2.5.1 Virtualização Completa	17
2.5.2 Paravirtualização	18
2.5.3 Virtualização Assistida por Hardware	18
2.6 Tendências da virtualização.....	19
2.6.1 Data Center Dinâmico	19
2.7 Hipervisores.....	20
2.7.1 VMware	20
2.7.2 Xen.....	21
3. Materiais.....	24
3.1 Softwares utilizados	24
3.1.1 Hipervisores	24
3.1.2 Gerenciadores de Hipervisores.....	24
3.1.3 Sistemas Operacionais	24
3.1.4 Software para servidores.....	25
3.1.5 Geradores de Carga de Trabalho	25
3.2 Hardwares utilizados	26
3.2.1 Notebook.....	26
3.2.2 Servidor.....	27
3.2.3 Medidor de energia elétrica.....	27
4. Metodologia.....	28
4.1 Avaliação da precisão do Kill A Watt P4400.....	28
4.1.1 Primeira Etapa.....	28
4.1.2 Segunda Etapa.....	30

4.1.3 Terceira Etapa.....	31
4.2 Carga de trabalho	33
4.2.1 Servidores Web Convencionais.....	33
4.2.2 Servidores Web Virtualizados.....	34
4.2.3 Servidor AD	35
4.3 Monitoramento dos servidores convencionais.....	36
4.3.1 Servidor Apache	37
4.3.2 Servidor IIS	38
4.3.3 Servidor AD	40
4.4 Monitoramento dos hipervisores.....	41
4.4.1 Ativação da Virtualização Assistida por Hardware.....	42
4.4.2 Hipervisor VMware ESXi 5.1	42
4.4.3 Hipervisor XCP 1.6	45
5. Resultados e discussão	47
5.1 Resultados.....	47
5.1.1 Diferença de consumo entre VMware ESXi 5.1 x XCP 1.6.....	47
5.1.2 Simulação de consumo	48
5.1.3 Economia monetária	49
5.2 Discussão	49
5.2.1 Diferença de desempenho dos hipervisores	49
6. Conclusão	50
7. Referências bibliográficas	51
8. Glossário	53

1. Introdução

A virtualização de servidores é uma técnica antiga, que surgiu entre o fim da década de 60 e o início de 70. Esta técnica permite a execução de diversos sistemas operacionais em um único servidor (FERNANDES, 2010).

Nos dias atuais, é a técnica central dos *data centers*, pois, aumenta o grau de utilização dos servidores, reduz a complexidade do gerenciamento, espaço físico, requisitos de energia e refrigeração (MARIN, 2011).

A virtualização pode ser realizada de diferentes maneiras, cada uma com seus prós e contras. Na arquitetura x86, as opções de virtualização alteram o nível de privilégios (anéis) padrão. As soluções baseadas em hipervisores incluem a virtualização completa e a paravirtualização (VERAS; KASSICK, 2011). Além destes dois tipos ainda existe a virtualização assistida por hardware, que foi utilizada neste experimento, implementada com os dois hipervisores mais conhecidos VMware e Xen.

O VMware ESXi é a arquitetura de hypervisor mais recente da VMware. Ele possui uma arquitetura ultrafina que não depende de um sistema operacional de uso geral. O VMware ESXi cria um novo patamar para a segurança e a confiabilidade, pois sua base de código menor representa uma “superfície de ataque” reduzida, com menos código para corrigir (VMware, 2013).

O Xen é um hipervisor livre, licenciado nos termos da GNU General Public Licence (GPL). É composto por três componentes fundamentais: (i) hipervisor Xen; (ii) domínio convidado privilegiado; (iii) domínio convidado não privilegiado (CARISSIMI, 2009).

Como essas duas ferramentas podem apresentar diferentes resultados em diferentes variáveis, neste trabalho elas foram avaliadas com base na eficiência energética promovida pelo uso da virtualização. Foi escolhida essa abordagem, uma vez que, com a crescente busca pela economia dos recursos naturais, vem crescendo o número de estudos sobre a virtualização de servidores.

2. Revisão literária

2.1 *Virtualização de servidores*

A virtualização de servidores é uma técnica que permite a execução de diversos sistemas operacionais em um único servidor. Sendo assim, com esta técnica, utilizando uma única máquina pode-se manter diversos sistemas operacionais, com diferentes aplicações sendo executadas simultaneamente (RODRIGUES, 2008).

2.1.1 *Breve Histórico*

Segundo Fernandes (2010), a virtualização é uma técnica antiga que surgiu entre o fim da década de 60 e o início de 70. Foi naquela época que surgiram os primeiros *mainframes*, que por serem computadores grandes e caros, geralmente tinham que ser compartilhados por um grande número de usuários. Nesse sentido houve um forte incentivo ao uso da virtualização, já que quase sempre usuários precisavam utilizar sistemas operacionais diferentes e ficava inviável ter um *mainframe* para cada pessoa.

Conforme Machado e Maia (2007), foi nesta época que o CTSS (Compatible Time-Sharing System) foi desenvolvido. Esse suportava no máximo 32 usuários interativos, por meio de comandos em um terminal permitia compilar e executar seus programas. O CTSS foi a base para outros sistemas operacionais de tempo compartilhado, como o MULTICS.

Na década de 80, o hardware passou a ser mais barato e os *mainframes* foram substituídos por *desktops*, diminuindo o interesse pelo uso da virtualização, pois, cada usuário podia ter o seu computador físico exclusivo.

Recentemente, o uso virtualização voltou a se tornar popular. Os grandes e caros *mainframes* do passado, são hoje os também grandes e caros servidores (ALMEIDA, 2012).

2.1.2 *Termos da Virtualização de Servidores*

Segundo Salgado (2011), o sistema operacional que executa sobre uma plataforma de virtualização é denominado hóspede. A plataforma de virtualização onde o hóspede executa é denominada hospedeiro ou sistema nativo. A camada de virtualização que implementa as máquinas virtuais é chamada de monitor de máquina virtual (MMV) ou hipervisor (hypervisor).

2.1.3 Tipos de Hipervisores

Segundo Veras e Kassick (2011) os hipervisores são classificados em dois tipos:

a) *Bare Metal*

Executa diretamente no *hardware* do servidor. Controla o *hardware* e o acesso do sistema operacional convidado. O papel do hipervisor nativo é compartilhar os recursos de hardware entre as máquinas virtuais, de forma que cada uma delas imagina ter recursos exclusivos.

b) *Hosted*

Aplicação que fornece um ambiente de execução para outras aplicações. Executa sob um sistema operacional nativo como se fosse um processo deste. A camada de virtualização é composta por um sistema operacional hóspede e um *hardware* virtual, que são criados sobre os recursos de *hardware* oferecidos através do sistema operacional nativo.

2.2 Vantagens

2.2.1 Otimização do Hardware

O avanço tecnológico é tão grande que o poder de processamento dos servidores aumentou muito e estes equipamentos sem a virtualização acabam ficando subutilizados. Segundo Troy e Helmke (2012), a maioria destes equipamentos operam há uma porcentagem muito a baixo da sua capacidade total de processamento, entre 10% a 15% da capacidade. De acordo com Veras (2011), com a virtualização, estes números podem subir para pelo menos 60%, o que otimiza bastante o uso destes equipamentos.

Segundo Ferreira (2012), as empresas adotaram a virtualização com o objetivo de reduzir os custos de *hardware*, energia ou mesmo evitar a construção de um novo *data center*, devido à proliferação de servidores físicos. A figura 1 ilustra a arquitetura tradicional e a virtualização.

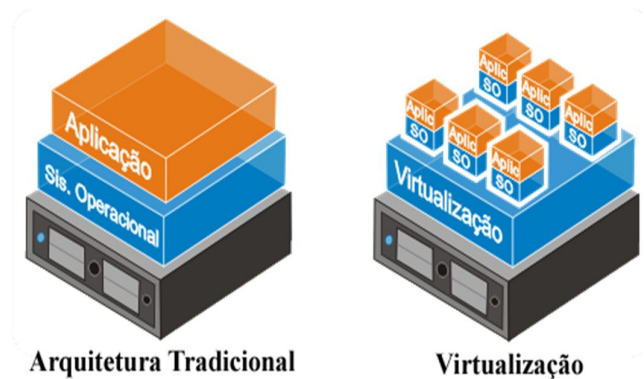


Figura 1 - Arquitetura tradicional e a virtualização. Fonte: ITS (2011).

2.2.2 Economia de Energia

Geralmente, os *data centers* são grandes consumidores de energia elétrica, devido a grande quantidade de equipamentos instalados nesses ambientes. Servidores convencionais e sistemas de climatização são alguns exemplos. Esse elevado consumo de energia resulta em um custo alto de manutenção e operação, fazendo-se assim, necessário um controle eficiente do uso de energia elétrica, bem como a otimização destes ambientes (MARIN, 2011).

De acordo com Salgado (2011), a exemplo de outras áreas de conhecimento, a computação está entrando na era do desenvolvimento sustentável. Profissionais da área vem propondo e implementando algumas formas de se utilizar mais racionalmente e eficientemente os recursos energéticos. As técnicas são várias, porém, há muito que evoluir.

Uma boa solução para este problema é a utilização de *data centers* baseados em servidores virtualizados. Com vários serviços executando em uma mesma instância física, pode-se atingir um alto nível de utilização do seu *hardware*. Servidores virtuais podem ser migrados de uma máquina física a outra sem grandes custos computacionais. Deste modo, o sistema pode ser gerenciado de forma a distribuir eficientemente os serviços entre os servidores. Pode-se concentrar todos os serviços em alguns servidores, permitindo que os outros sejam desligados (ALMEIDA, 2012).

Podemos perceber que a economia de energia utilizando a virtualização de servidores pode ser grande. Segundo Veras (2011), um servidor em plena carga ou a 15% consome praticamente a mesma quantidade de energia e nos *data centers* onde se utiliza a virtualização de servidores a taxa de consolidação geralmente é maior que 10:1, ou seja, 1 servidor virtualizado economiza energia de 9 outros servidores. Esta economia além de diminuir custos pode ajudar na disponibilidade dos serviços hospedados. Em caso de pane do

sistema de energia, o *data center* poderá ficar mais tempo ligado por meio dos *nobreaks* e geradores pois o consumo de energia é menor.

Outro fator que contribui para o consumo de energia é o espaço ocupado pelos servidores. Um *data center* geralmente é dimensionado de acordo com a quantidade de servidores, *storages* e equipamentos de telecomunicação. Quanto maior o número de equipamentos maior é o espaço físico e maiores são os gastos com refrigeração. Marin (2011), afirma que aproximadamente 100% da energia elétrica usada no *data center* é convertida em calor que precisa ser retirado do ambiente.

Para Salgado (2011), a utilização da virtualização de servidores garante uma eficiência energética por dois fatores: (i) menos *hardware* é necessário, pois ele é utilizado de forma eficiente; (ii) conforme a demanda, servidores podem ser desligados, poupando a energia de alimentação desses equipamentos e dos seus sistemas de refrigeração. Cada servidor virtualizado pode economizar 7.000 kWh de energia e deixar de emitir quatro toneladas de dióxido de carbono por ano.

2.3 Desvantagens

Existem alguns impedimentos no uso da virtualização, pois ela poderá ter alterações de desempenho dependendo da carga de trabalho. Conforme Veras e Kassick (2011), alguns aplicativos e cargas de trabalho poderão apresentar exigências específicas de *hardware* que talvez não sejam suportadas em um ambiente virtual.

A outra desvantagem diz respeito à necessidade de fazer atualizações de *hardware* ou *software* no servidor hospedeiro. Em ambientes computacionais, esta é uma prática bastante comum, e os servidores utilizados para virtualização não fogem a esta regra. Porém, quando atualizações desse tipo são necessárias, elas não tornam indisponíveis apenas o servidor hospedeiro, mas também, os hóspedes que ela possui, mesmo que momentaneamente. No entanto, já existe uma solução que suprime completamente esta desvantagem. Atualmente, é possível efetuar a movimentação de máquinas virtuais de um servidor hospedeiro para outro, sem que exista a necessidade de desligar os hóspedes. Para tal, é necessário que as máquinas utilizem uma área compartilhada de armazenamento externo e uma conexão de rede de alta velocidade entre os servidores físicos. Desta forma, como os servidores acessam a mesma área de dados, basta que o conteúdo de memória seja transferido, via rede, de um servidor para outro. Após a atualização, as máquinas virtuais podem, novamente ser movidas para a sua máquina de origem (RIBAS, 2008).

2.4 Nível de privilégio da arquitetura X86

De acordo com a empresa VMware (2013), a arquitetura x86 é o padrão do setor de servidores, e os fabricantes líderes do mercado, como Dell, IBM, HP e Fujitsu-Siemens adotam esta arquitetura. Por este motivo, é abordado o nível de privilégio desta arquitetura.

Segundo Neiva (2010), a arquitetura x86 provê quatro modos de operação para o processador, identificados de 0 a 3, denominados de anéis de proteção. O anel 0 foi projetado para abrigar o *kernel* do sistema operacional, o anel 1 e 2 os *drivers* de dispositivos e o anel 3 as aplicações de usuários.

Nos sistemas operacionais convencionais, como o *Windows* e o *Linux*, apenas os anéis 0 e 3 são utilizados. O anel nível 0 detém os maiores privilégios de execução e é usado pelo sistema operacional. O anel nível 3, de menor privilégio, é utilizado por processos do usuário. As instruções de máquina do processador são então divididas em instruções não privilegiadas que executam em modo usuário (anel 3), e instruções privilegiadas que executam em modo protegido (anel 0). Se um processo de usuário tenta executar uma instrução privilegiada, ocorre uma exceção (trap) que deverá ser tratada adequadamente pelo sistema operacional. Entretanto, tratando-se da arquitetura x86, existem instruções que afetam o funcionamento do processador, como instruções privilegiadas, mas que são executadas em modo usuário sem gerar exceções (traps). Essas instruções são denominadas de "sensíveis", e seu uso em sistemas operacionais nativos (não virtualizados) não causam problemas. No entanto, em ambientes virtualizados, as instruções sensíveis devem ser tratadas adequadamente, pois podem ser feitas por um processo em execução em um hóspede, e assim alterar o comportamento do sistema operacional nativo ou de outro hóspede. A figura 2 ilustra no nível de privilégio da arquitetura x86 (VERAS; KASSICK, 2011).

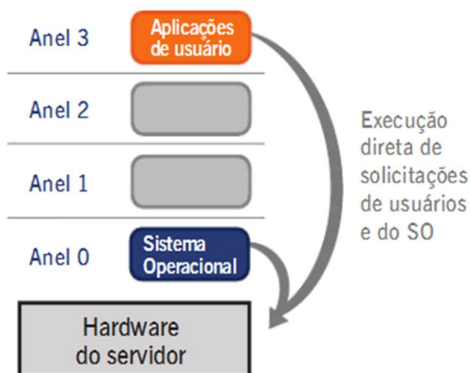


Figura 2 - Nível de privilégio da arquitetura x86. Fonte: VERAS; KASSICK (2011).

2.5 Tipos de virtualização de servidores

A virtualização pode ser realizada de diferentes maneiras, cada uma com seus prós e contras. Na arquitetura x86, as opções de virtualização alteram o nível de privilégios (anéis) padrão. As soluções baseadas em hipervisores incluem a virtualização completa e a paravirtualização (VERAS; KASSICK, 2011). Além destes dois tipos ainda existe a virtualização assistida por *hardware*.

2.5.1 Virtualização Completa

Conforme Veras e Kassick (2011), a virtualização completa (ou total) realiza a completa abstração do sistema físico, criando um sistema físico virtual completo, sobre o qual o sistema operacional hóspede é executado. Não é necessário fazer qualquer modificação no sistema operacional hóspede ou em suas aplicações. Este tipo de virtualização facilita a migração de máquinas virtuais entre servidores hospedeiros, pois existe total independência das aplicações e dos recursos físicos do servidor. Desse modo, a segurança é facilitada pelo isolamento entre as máquinas virtuais, já que cada instância da máquina virtual é um processo do sistema operacional hospedeiro.

Uma desvantagem da virtualização completa é o desempenho, pois o hipervisor verifica a execução de todas as instruções privilegiadas ou sensíveis feitas pelo sistema operacional hóspede, e as substitui por ações equivalentes controladas. Outro ponto desfavorável da virtualização completa é a dificuldade de emular o funcionamento dos dispositivos de E/S, por conta da diversidade dos dispositivos existentes. A solução empregada consiste em implementar *hardwares* virtuais que emulam dispositivos genéricos, o que pode causar a subutilização dos dispositivos de E/S (VERAS, 2011). A figura 3 ilustra a arquitetura de privilégio da virtualização completa.

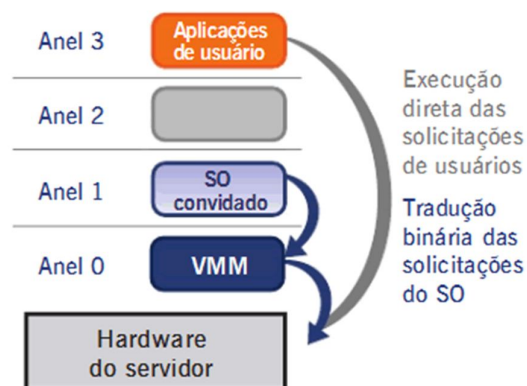


Figura 3 - Nível de privilégio da virtualização completa no x86. Fonte: VERAS; KASSICK (2011).

2.5.2 Paravirtualização

Segundo Almeida (2012), a paravirtualização é uma alternativa para contornar os problemas de desempenho e subutilização de recursos da virtualização completa. Na paravirtualização, o sistema operacional hóspede é alterado para chamar o hipervisor sempre que for executar uma instrução sensível. As instruções não privilegiadas, aquelas realizadas pelos processos de usuários, podem ser executadas diretamente sobre o processador nativo.

Para Veras e Kassick (2011), em relação aos dispositivos de *E/S*, os hipervisores que empregam paravirtualização permitem que as máquinas virtuais empreguem os *drivers* do dispositivo físico real sob o seu controle, o que é interessante, pois este aspecto otimiza o desempenho. A principal desvantagem da paravirtualização é a necessidade de modificação do sistema operacional hóspede, o que pressupõe acesso ao código-fonte.

O Xen é um exemplo de sistema baseado em paravirtualização, que virtualiza o processador e a memória, usando um *kernel Linux* modificado e virtualizando o sistema de *E/S* com *drivers* de dispositivos customizados. A figura 4 ilustra a arquitetura de privilégio da paravirtualização.

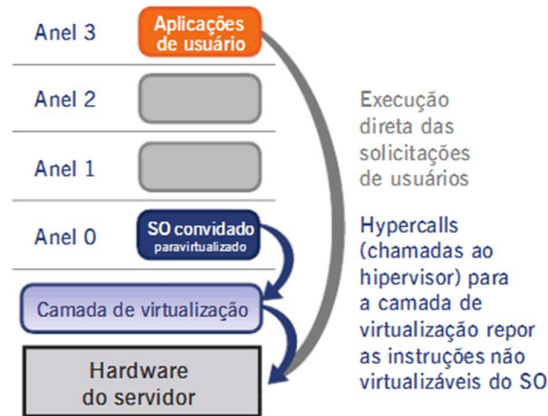


Figura 4 - Nível de privilégio da paravirtualização no x86. Fonte: VERAS; KASSISC (2011).

2.5.3 Virtualização Assistida por Hardware

Conforme Salgado (2011), há uma relação custo/benefício entre a virtualização total e a paravirtualização. Enquanto a primeira permite o uso de um sistema operacional hóspede sem modificações, a segunda precisa alterá-lo para substituir instruções privilegiadas e sensíveis por *hypercalls*, mas oferece um melhor desempenho. Sendo assim, os fabricantes Intel e AMD investiram em extensões na arquitetura x86, para suportar a virtualização e melhorar o desempenho da solução como um todo. Essas extensões são genericamente

denominadas de virtualização assistida por hardware (Hardware Assisted Virtualization - HAV). Este movimento da AMD e da Intel praticamente eliminou as vantagens de desempenho dos sistemas baseados em paravirtualização, que tinham o ônus de modificar o sistema operacional para funcionar.

Os fabricantes de processadores (Intel e AMD) alteraram o funcionamento dos anéis de proteção dos processadores x86. O anel nível 0 é utilizado pelo sistema operacional e executado em modo protegido, e o anel nível 3 (de menor privilégio), é empregado pelos processos que executam em modo usuário. Os anéis 1 e 2 não são usados pelos sistemas operacionais convencionais. A Intel e AMD, de formas diferentes, mas com o mesmo pensamento, criaram dois novos modos de operação para os anéis de proteção do processador: modo "root" e modo "non-root". Os anéis de 0 a 3 executam no modo "non-root" e o hipervisor executa no anel adicional de maior prioridade (modo root). Portanto o hipervisor passou a ter total prioridade sobre o sistema operacional. Neste novo modo de operação, as instruções privilegiadas e sensíveis executadas pelo sistema operacional hospede causam um desvio (trap) para o hipervisor que tem a responsabilidade de tratar adequadamente a ocorrência dessas ações (VERAS; KASSISC, 2011). A figura 5 ilustra a arquitetura de privilégio da virtualização assistida por hardware.

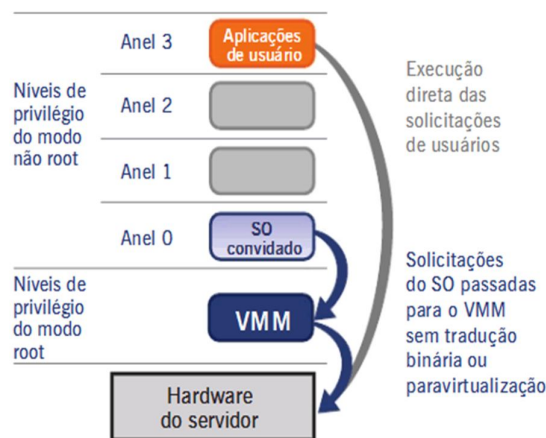


Figura 5 - Nível de privilégio virtualização assistida por hardware. Fonte: VERAS; KASSISC (2011).

2.6 Tendências da virtualização

2.6.1 Data Center Dinâmico

Conforme Sousa et al (2009), a virtualização muda a maneira como um *data center* é gerenciado. O conceito de virtualização desfaz um modelo tradicional utilizado, que é o de associar cada aplicação ou serviço a uma máquina específica. A virtualização permite

melhorar a utilização dos servidores que tradicionalmente rodam a 10 a 15% da sua capacidade. A execução de várias máquinas virtuais em um servidor faz com que sejam ocupados os ciclos ociosos do processador e seja eliminada a necessidade de outro servidor físico, reduzindo o consumo de energia e o espaço físico utilizado. O conceito de *data center* dinâmico permite alocar os recursos de forma imediata mediante a demanda.

2.7 Hipervisores

2.7.1 VMware

O VMware é um dos mais populares *softwares* de virtualização para a arquitetura x86. O tipo de virtualização oferecido por essa ferramenta é a virtualização completa. O VMware é uma infraestrutura completa de virtualização, fornecendo *softwares* para ambientes *desktop* a ambientes de *data centers*. Os produtos disponibilizados dividem-se em três categorias: gerenciamento e automação, infraestrutura virtual e plataformas de virtualização (GROSMANN et al., 2012).

Segundo a empresa VMware (2013), o VMware ESXi é a arquitetura de hypervisor mais recente da VMware. Ele possui uma arquitetura ultrafina que não depende de um sistema operacional de uso geral. O VMware ESXi cria um novo patamar para a segurança e a confiabilidade, pois sua base de código menor representa uma “superfície de ataque” reduzida, com menos código para corrigir.

Com o pequeno espaço ocupado e a mesma confiabilidade de um *hardware*, o VMware ESXi pode ser integrado diretamente aos servidores x86. O VMware ESXi foi desenvolvido objetivando a simplicidade. A inicialização por menu e as configurações automáticas tornam o ESXi a maneira mais fácil de conhecer a virtualização da VMware (VMWARE, 2013).

O VMware ESXi é instalados diretamente no *hardware* do servidor hospedeiro, inserindo uma camada de virtualização entre o hardware e o sistema operacional hóspede. O ESXi particiona um servidor físico em várias máquinas virtuais seguras e portáteis que podem ser executadas lado a lado no mesmo servidor físico. Cada máquina virtual representa um sistema completo com processadores, memória, rede, armazenamento e *BIOS* de modo que o sistema operacional e os aplicativos de software possam ser instalados e executados na máquina virtual sem qualquer modificação. As máquinas virtuais também são totalmente isoladas umas das outras pela camada de virtualização, evitando, desta maneira, que uma

falha ou um erro de configuração em um hóspede afete os demais (TROY; HELMKE, 2012). A figura 6 ilustra várias máquinas virtuais em um mesmo servidor físico.

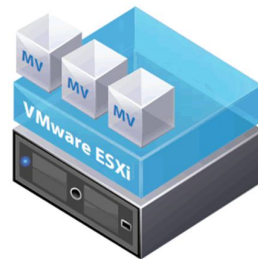


Figura 6 - Vários servidores virtuais em uma mesma máquina. Fonte: VMWARE (2013).

Segundo Veras (2011), o VMware ESXi utiliza da virtualização completa, porém, também suporta sistemas operacionais hóspedes *Linux* paravirtualizados para aprimorar o desempenho da máquina virtual.

2.7.2 Xen

Conforme Carissimi (2009), o Xen é um monitor de máquina virtual (hipervisor) em *software* livre, licenciado nos termos da *GNU General Public Licence (GPL)*, para arquiteturas x86, que permite vários sistemas operacionais hóspedes serem executados em um mesmo sistema hospedeiro.

O Xen é composto por três componentes fundamentais, hipervisor Xen, domínio convidado privilegiado e domínio convidado não privilegiado. O hipervisor Xen é a camada base, sobre essa camada estão um ou mais sistemas operacionais hóspedes, os quais o hipervisor escala sobre os processadores disponíveis. Há sempre no mínimo um sistema operacional convidado na arquitetura Xen. Essa é uma máquina virtual automaticamente inicializada no *boot* do Xen e, na terminologia empregada, corresponde ao domínio convidado 0 ou, simplesmente dom0. Os demais sistemas convidados são denominados de domU, onde o "U" vem do inglês *unprivileged* (VERAS; KASSICK, 2011). A figura 7 ilustra a arquitetura do Xen.

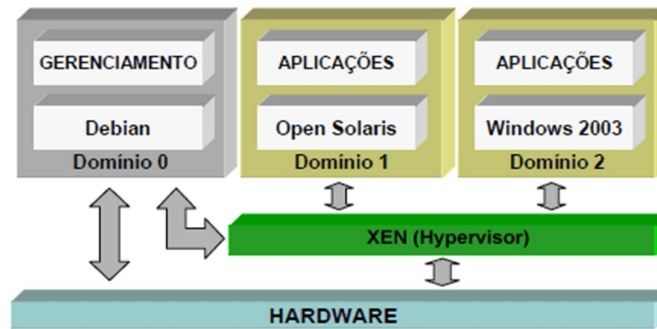


Figura 7 - Estrutura do Xen. Fonte: AZEVEDO (2010).

a) *Hipervisor Xen*

De acordo com Almeida (2011), o hipervisor Xen fica entre o *hardware* e os hóspedes. O hipervisor executa diretamente sobre o *hardware* e se torna um tipo de interface entre todas as requisições de hardware (processador, interrupções e disco) para os hóspedes. Ao separar os hóspedes do *hardware*, o Xen está apto a executar múltiplos sistemas operacionais de forma segura e independente.

Conforme Salgado (2011), o hipervisor Xen não possui nenhum tipo de *driver* de dispositivo, sendo assim, o Xen não acessa efetivamente o *hardware* de *E/S*, apenas o gerencia. Os acessos aos dispositivos são feitos pelos *drivers* existentes no domínio convidado 0 (dom0) de forma coordenada com o hipervisor. Devido a essa estrutura o dom0 é um domínio privilegiado.

b) *Domínio Convidado Privilegiado*

Durante a inicialização, o domínio hóspede privilegiado (dom0) é o primeiro sistema operacional hóspede automaticamente a ser carregado pelo hipervisor Xen. O dom0 é o único domínio convidado que tem privilégios especiais para acessar e administrar o hipervisor Xen. (VERAS; KASSICK, 2011).

c) *Domínio Convidado não Privilegiado*

Os domínios convidados não privilegiados, denominados domU, são carregados e inicializados a partir do dom0. Em um sistema computacional hospedeiro podem existir vários domU e cada um deles executa de forma totalmente isolada e independente dos demais domínios existentes na máquina hospedeira. Um domínio hóspede executa de forma não privilegiada. Todos os acessos a recursos compartilhados (processador, memória, *E/S*) devem ser realizados por meio do dom0 e do hipervisor Xen (SALGADO, 2011).

d) *Xen e a virtualização completa*

O Xen exigia, até a sua versão 2, que o sistema operacional de um domínio hóspede fosse modificado justamente para substituir as chamadas de sistemas nativas por chamadas ao hipervisor (paravirtualização). A partir da versão 3, isso não é mais necessário, ou seja, é possível o emprego de sistemas operacionais não modificados, o que permite a execução dos sistemas da família *Microsoft Windows* - virtualização completa. No entanto, essa opção só é possível quando o processador possui suporte por *hardware* para virtualização (*Intel-VT_x ou AMD-V*) (VERAS; KASSICK, 2011).

Segundo Carissimi (2009), para oferecer suporte tanto para a paravirtualização como para a virtualização total, o Xen distingue os domU entre paravirtualizados (domínios U-PV) e virtualizados (domínios U-HVM). Os domínios U-PV têm consciência de que não tem acesso direto ao hardware e reconhecem a existência de outras máquinas virtuais. Os domínios U-HVM não tem essa consciência, nem reconhecem a existência de outras máquinas virtuais. Isso está relacionado ao fato de que os domínios U-PV possuem *drivers* específicos para acesso à rede e ao disco para interagirem com as suas contra-partidas no domínio 0. Já as máquinas dos domínios U-HVM não possuem esses *drivers* (não foram modificados) e iniciam como um sistema convencional procurando executar a *BIOS*.

3. Materiais

Para real conclusão de qual hipervisor, VMware ou Xen, possui maior eficiência na economia de energia elétrica, foram realizados vários testes. Esses testes visaram basicamente medir o consumo de energia elétrica do servidor em diversas situações que serão descritas mais a frente.

Buscando um melhor entendimento, a seguir será descrito os softwares utilizados.

3.1 Softwares utilizados

3.1.1 Hipervisores

a) VMware ESXi 5.1

O VMware ESXi 5.1 é a versão free do hypervisor da VMware, é uma alternativa para um ambiente pequeno (CASTRO, 2012).

b) XCP 1.6

XCP (Xen Cloud Platform), é uma solução de virtualização do *Xen Open Source*, o XCP proporciona um ambiente de virtualização e cloud computing, ele inclui o Xen Hypervisor (GALOSSO, 2013).

3.1.2 Gerenciadores de Hipervisores

a) vSphere Client

Segundo a VMware (2013), o *vSphere Client* é a interface administrativa principal do VMware ESXi 5.1. Ele permite que os administradores de TI gerenciem as principais funções do VMware ESXi 5.1 de qualquer lugar do mundo.

b) Open Xen Manager

De acordo com Hamacker (2013), o *Open Xen Manager* é um *software* multiplataforma, desenvolvido em *python* que gerencia o hipervisor XCP.

3.1.3 Sistemas Operacionais

a) Debian Squeeze

De acordo com o site do Debian (2013), o Debian Squeeze é um sistema operacional livre que atualmente usa o *kernel Linux*.

b) Windows Server 2012

Windows Server 2012 é um Sistema Operacional Multitarefa. Sua principal característica é a forma como se apresenta e é fácil aprendê-lo porque sua interface gráfica é uniforme em todos os aplicativos (BRADESCO, 2013).

3.1.4 Software para servidores

a) Apache

De acordo com o site do IBICT - Instituto Brasileiro de Informação em Ciência e Tecnologia (2013), o servidor *Apache* é o mais bem sucedido servidor *web* livre. Foi criado em 1995 por Rob McCool, então funcionário do NCSA (National Center for Supercomputing Applications). Suas funcionalidades são mantidas através de uma estrutura de módulos, permitindo inclusive que o usuário escreva seus próprios módulos, utilizando a *API* do *software*. É o servidor *web* mais utilizado atualmente no mundo.

b) Internet Information Services (IIS)

O IIS (Internet Information Services) permite compartilhar informações com usuários na *Internet*, *intranet* ou *extranet*. O Windows Server 2012 oferece o IIS 8, que é uma plataforma Web unificada que integra o IIS, o *ASP.NET*, o *Windows Communication Foundation* e o *Windows SharePoint Services*.

c) Active Directory (AD)

O AD (Active Directory) é um serviço de diretório nas redes *Windows*. Serviço de diretório é um conjunto de atributos sobre recursos e serviços existentes na rede, isso significa que é uma maneira de organizar e simplificar o acesso aos recursos de sua rede centralizando-os; bem como, reforçar a segurança e dar proteção aos objetos da database contra intrusos, ou controlar acessos dos usuários internos da rede. O AD mantém dados como contas de usuários, impressoras, grupos, computadores, servidores, recursos de rede, etc.

3.1.5 Geradores de Carga de Trabalho

a) JMeter

JMeter é uma aplicação *desktop* de código aberto, 100% feita em *Java* desenvolvida para executar testes funcionais e medir o desempenho de aplicações *Web*.

b) *Cobian Backup*

Cobian Backup é um *software* com recursos para você configurar e agendar *backups* automaticamente em sistemas *Windows*.

c) *Crontab e Shell Script*

Crontab é um recurso dos sistemas *Linux* que permite o agendamento da execução de determinados comandos ou *scripts*.

Já o *Shell Script* é definido por Virgilio (2013) como uma poderosa ferramenta de automação de instruções. Com um arquivo de texto executável o usuário ou sistema é capaz de executar uma seqüência de operações, instruções e testes.

3.2 *Hardwares utilizados*

A figura 8 mostra como foi montado o experimento.



Figura 8 - Hardwares utilizados.

3.2.1 *Notebook*

Para gerar carga no servidor, foi utilizado um *notebook* HP modelo ProBook 4320s, com 4 GB de RAM, processador Core i3 370M e placa de rede 10/100/1000, com sistema operacional Debian Squeeze.

3.2.2 Servidor

Foi utilizado um computador com a placa mãe Intel DH61HO e processador Core i3 3320, ambos com suporte a Virtualização Assistida por Hardware. O computador contava com 8 GB de RAM, placa de rede 10/100/1000 e 500 GB de HD. A fonte que alimentava tais componentes era a PX-300RMG da marca K-MEX, com potência real de 200 W.

3.2.3 Medidor de energia elétrica

Durante todas as situações, foi utilizado para medir o consumo de energia elétrica o aparelho *Kill A Watt P4400*.

Possui um *display LCD* no qual são mostradas medidas relativas ao consumo de energia do equipamento elétrico conectado a ele, entre elas estão: *Volts, Amperes, Watts, Hertz, Volt-Amperes* ou *kWh* e tempo que o aparelho está ligado. Com essas medidas é possível analisar o consumo individual de um eletrodoméstico.

De acordo com a P3 International, fabricante do equipamento, a precisão do *Kill A Watt P4400* é de 99,8%, porcentagem que está dentro das normas que regulamentam este tipo de equipamento. A figura 9 mostra o *Kill A Watt P4400*.



Figura 9 - Kill A Watt P4400

4. Metodologia

O tipo de virtualização escolhido foi a Virtualização Assistida por Hardware, por possuir melhor desempenho que a Virtualização Completa, e por ser possível virtualizar tanto sistemas *Linux* quanto Sistemas *Windows*.

4.1 Avaliação da precisão do Kill A Watt P4400

Buscando realizar o experimento com a maior precisão possível e livre de erros, foi realizada uma avaliação da precisão do *Kill A Watt P4400*.

Para realizar esta avaliação, foram utilizadas duas lâmpadas incandescentes, uma de 60 W e outra de 100 W. O intuito era basicamente monitorar o consumo das lâmpadas através do *Kill A Watt P4400*, e depois conferir os valores utilizando uma equação da física.

Segundo Halliday, Resnick e Walker (2009), para calcular o consumo de energia elétrica é necessário que se conheça a potência do aparelho e o tempo de utilização dele. O cálculo não leva em consideração as variações do consumo e é justamente por isso que foram utilizadas lâmpadas incandescentes, pois as mesmas possuem baixa variação de consumo.

Equação utilizada: Energia Elétrica = Potência x Δ tempo

Segundo Ferreira (2012), a medida de energia elétrica mais utilizada é o *kWh*, por este motivo ela foi utilizada.

O consumo de energia elétrica das lâmpadas foi monitorado em 3 etapas:

- 1ª etapa: lâmpada de 60 W,
- 2ª etapa: lâmpada de 100 W,
- 3ª etapa: lâmpadas de 60 W e 100 W ligadas em paralelo.

Cada etapa durou 1h40m, sendo feita a leitura do *Kill A Watt P4400* de 20 em 20 minutos.

4.1.1 Primeira Etapa

Foi monitorado nessa etapa o consumo da lâmpada de 60 W.

A figura 10 mostra a lâmpada de 60 W, desligada para garantir a qualidade da imagem, conectada no *Kill A Watt P4400*.



Figura 10 - Lâmpada de 60 W conectada no Kill A Watt.

Durante 1h40m, a lâmpada de 60 W consumiu um total de 0,10 kWh. Conforme a figura 11.

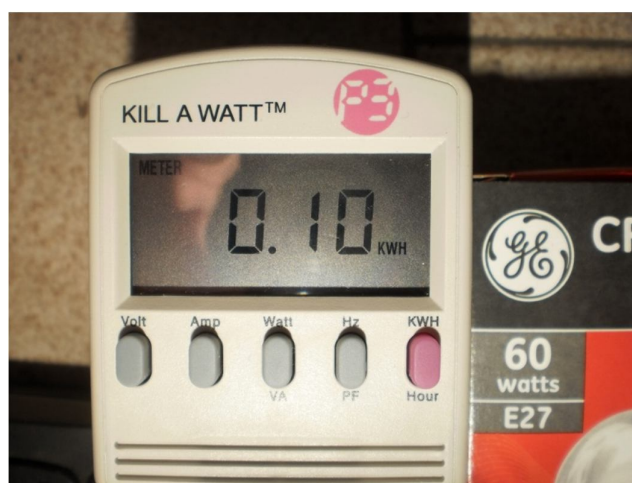


Figura 11 - Consumo acumulado da lâmpada de 60 W.

A tabela 1 detalha o consumo durante o monitoramento.

Tabela 1 - Consumo detalhado da lâmpada de 60 W

TEMPO	CONSUMO ACUMULADO
0 minuto	0 kWh
20 minutos	0,02 kWh
40 minutos	0,04 kWh
1 hora	0,06 kWh
1 hora 20 minutos	0,08 kWh
1 hora 40 minutos	0,10 kWh

Os cálculos, a seguir, foram feitos utilizando-se o tempo final (1h40m).

Energia Elétrica = Potência x Δ tempo

Energia Elétrica = ? kWh

Potência = 60 W = 0,06 kW

Δ tempo = 1h40m = 1,667 h

Energia Elétrica = 0,06 x 1,667

Energia Elétrica = 0,10 kWh

Segundo os cálculos acima, durante 1h40m uma lâmpada de 60 W consome 0,10 kWh, justamente o valor informado pelo Kill A Watt P4400.

4.1.2 Segunda Etapa

Nessa etapa foi monitorado o consumo da lâmpada de 100 W.

A figura 12 mostra a lâmpada de 100 W, desligada para garantir a qualidade da imagem, conectada no Kill A Watt P4400.



Figura 12 - Lâmpada de 100 W conectada no Kill A Watt.

Durante 1h40m, a lâmpada de 100 W consumiu um total de 0,16 kWh. Conforme a figura 13.

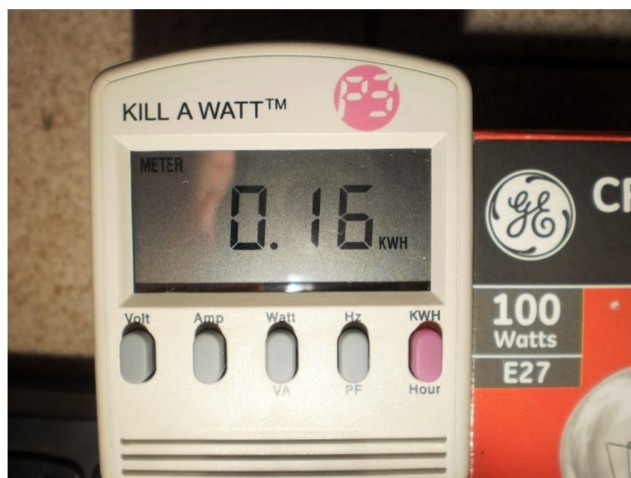


Figura 13 - Consumo acumulado da lâmpada de 100 W.

A tabela 2 detalha o consumo durante o monitoramento.

Tabela 2 - Consumo detalhado da lâmpada de 100 W

TEMPO	CONSUMO ACUMULADO
0 minuto	0 kWh
20 minutos	0,03 kWh
40 minutos	0,06 kWh
1 hora	0,10 kWh
1 hora 20 minutos	0,13 kWh
1 hora 40 minutos	0,16 kWh

Os cálculos, a seguir, foram feitos utilizando-se o tempo final (1h40m).

Energia Elétrica = Potência x Δ tempo

Energia Elétrica = ? kWh

Potência = 100W = 0,1 kW

Δ tempo = 1h40m = 1,667 h

Energia Elétrica = 0,1 x 1,667

Energia Elétrica = 0,1667 kWh

Segundo os cálculos acima, durante 1h40m uma lâmpada de 100 W consome 0,1667 kWh, justamente o valor, sem as duas últimas casas decimais, informado pelo Kill A Watt P4400.

4.1.3 Terceira Etapa

Nessa etapa foi monitorado o consumo das lâmpadas de 60 W e 100 W ligadas em paralelo.

A figura 14 mostra a lâmpada de 60 W em paralelo com a de 100 W, desligadas para garantir a qualidade da imagem, conectadas no *Kill A Watt P4400*.



Figura 14 - Lâmpadas em paralelo conectadas no Kill A Watt.

Durante 1h40m, a lâmpada de 60 W em paralelo com a de 100 W, consumiram um total de 0,26 kWh. Conforme a figura 15.



Figura 15 - Consumo acumulado das lâmpadas em paralelo.

A tabela 3 detalha o consumo durante todo monitoramento.

Tabela 3 - Consumo detalhado da lâmpada de 60 W em paralelo com a de 100 W.

TEMPO	CONSUMO ACUMULADO
0 minuto	0 kWh
20 minutos	0,05 kWh
40 minutos	0,10 kWh
1 hora	0,16 kWh
1 hora 20 minutos	0,21 kWh
1 hora 40 minutos	0,26 kWh

Os cálculos, a seguir, foram feitos utilizando-se o tempo final (1h40m).

Energia Elétrica = Potência x Δ tempo

Energia Elétrica = ? kWh

Potência = 160W = 0,16 kW

Δ tempo = 1h40m = 1,667 h

Energia Elétrica = 0,16 x 1,667

Energia Elétrica = 0,2667 kWh

Segundo os cálculos acima, durante 1h40m, uma lâmpada de 60 W e 100 W, ligadas em paralelo consomem 0,2667 kWh, justamente o valor, sem as duas últimas casas decimais, informado pelo *Kill A Watt P4400*.

Com a avaliação da precisão do *Kill A Watt P4400*, pode-se considerar que o aparelho tem um nível ótimo de precisão, apesar de exibir somente duas casas decimais após a vírgula.

4.2 Carga de trabalho

4.2.1 Servidores Web Convencionais

Para gerar carga de trabalho nos servidores *web*, foi utilizado o JMeter que estava instalado no *notebook* Probook 3420s. O JMeter foi configurado para gerar uma carga de 10 usuários simultaneamente e de modo contínuo. Essa carga de trabalho ocorreu durante todo o tempo de monitoramento (3 dias).

A figura 16 mostra as configurações referentes ao grupo de usuários. As duas configurações mais importantes para o momento são: número de usuários virtuais (10) e o contador de interação que está habilitado o modo infinito.

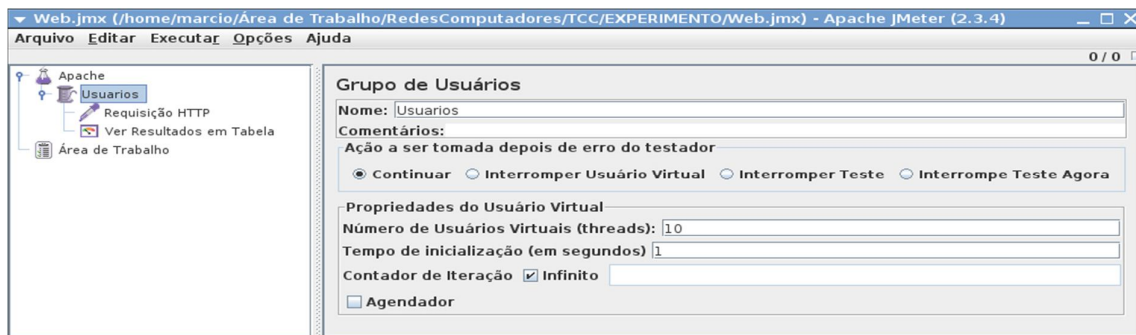


Figura 16 - Grupo de usuários virtuais.

A figura 17 mostra as configurações referentes as requisições *HTTP*. As configurações mais importantes para o momento são: *IP* do servidor (10.0.0.3), número da porta (80), além de estar também habilitada a opção “Recuperar todos os recursos embutidos a partir de arquivos *HTML*”. Essa configuração fez com que todos os arquivos, que estavam na página inicial e hospedados nos servidores *web*, fossem baixados a cada requisição. Esses arquivos somaram um total de 151 *KB*.

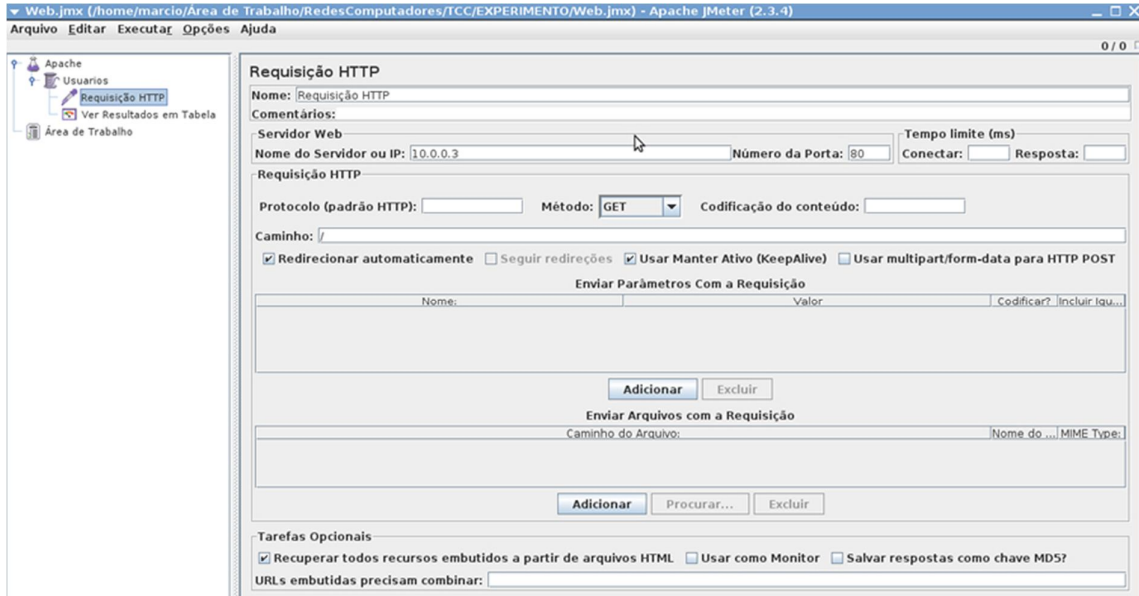


Figura 17 - Configurações das requisições *HTTP*.

4.2.2 Servidores *Web* Virtualizados

A carga de trabalho dos servidores *web* virtualizados foi idêntica a dos servidores convencionais, o que mudou foi somente a quantidade de grupos de usuários e, conseqüentemente o número de requisições *HTTP*. Tal situação é mostrada nas figuras 18 e 19.

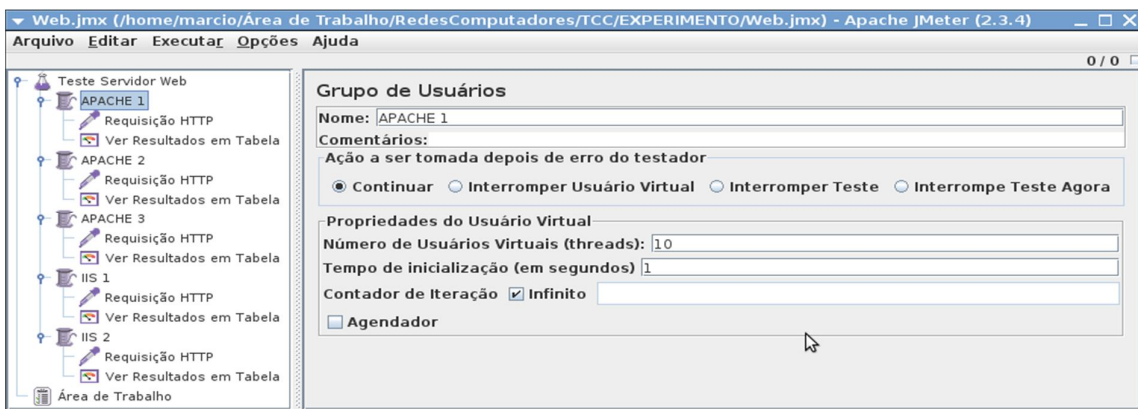


Figura 18 - Vários grupos de usuários virtuais.

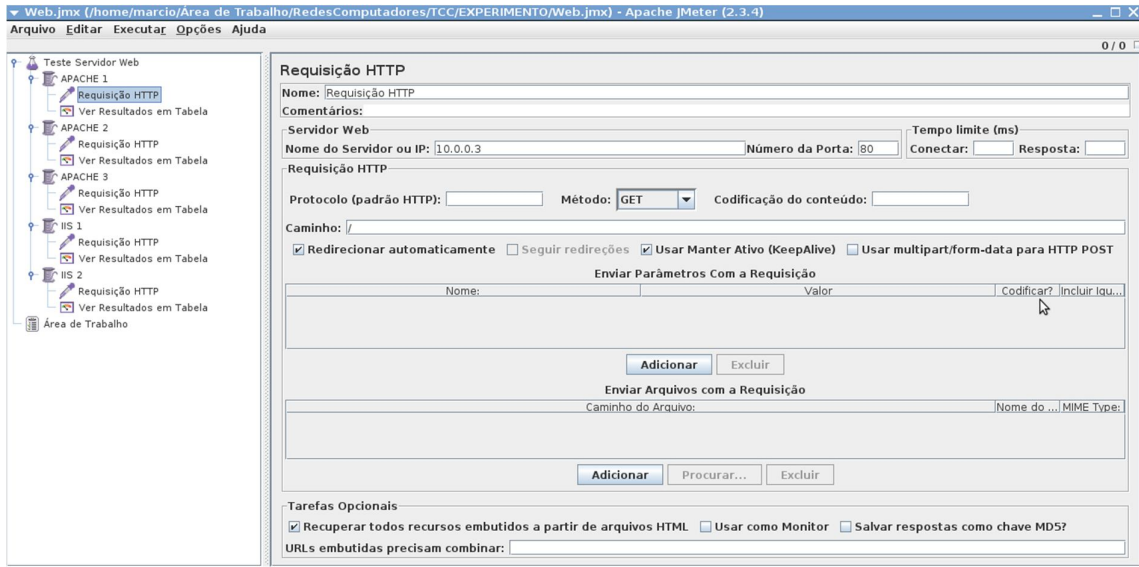


Figura 19 - Configurações das requisições HTTP.

4.2.3 Servidor AD

Para gerar carga de trabalho no Servidor AD, tanto no convencional quanto no virtualizado, foi utilizado o Cobian Backup e um Shell Script que foi executado através do Crontab.

O Cobian Backup foi instalado no próprio Servidor AD. Uma vez instalado, compactava a pasta c:\tcc que continha 37,8 GB e o arquivo compactado era armazenado na pasta c:\backup, a tarefa era executada de 8 em 8 horas. As figuras 20 e 21 mostram as configurações.

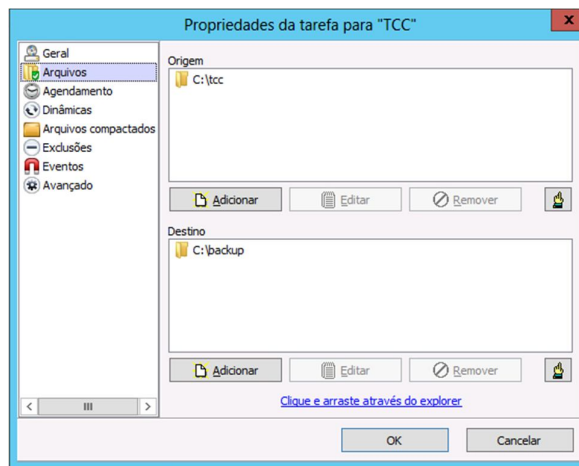


Figura 20 - Cobian Backup, pasta origem e destino.

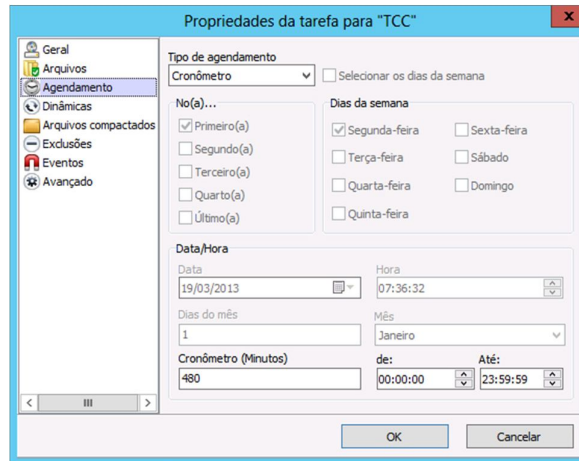


Figura 21 - Cobian Backup, agendamento de 8 em 8 horas.

Já o Shell Script ficou sendo executando através do Crontab no *notebook* Probook 3420s. De 2 em 2 minutos, o *notebook* enviava um arquivo de 195 MB para o Servidor AD.

Para ter acesso ao compartilhamento do AD no Debian Squeeze do *notebook*, foi preciso montar o diretório, e o seguinte comando foi utilizado.

```
#mount -t cifs //10.0.0.3/tcc /mnt/ad -o username=adminstrador,password=asd123*
```

O Shell Script a seguir com nome *carga-trabalho.sh* copiava o arquivo chamado *arquivo.zip* de 195 MB para a pasta */mnt/ad*, que foi montada anteriormente.

```
#!/bin/bash
cd /root/
cp arquivo.zip /mnt/ad/
```

No Crontab, que de 2 em 2 minutos executava o Shell Script anterior, foi adicionada a seguinte linha.

```
*/2 * * * * /root/carga-trabalho.sh
```

4.3 Monitoramento dos servidores convencionais

Todos os servidores (Apache, IIS e AD) foram instalados na mesma máquina, um por vez. Cada servidor foi monitorado durante 3 dias. Os servidores *web* receberam a carga de trabalho especificada no subcapítulo 4.2.1 e o servidor AD recebeu a carga de trabalho especificada no subcapítulo 4.2.2. A leitura do consumo de energia elétrica foi realizada de 12 em 12 horas.

A seguir, será explicitado o monitoramento detalhado de cada servidor, no qual haverá a exposição do cenário, taxa de utilização e consumo de energia.

4.3.1 Servidor Apache

a) Cenário

A figura 22 exemplifica o cenário.

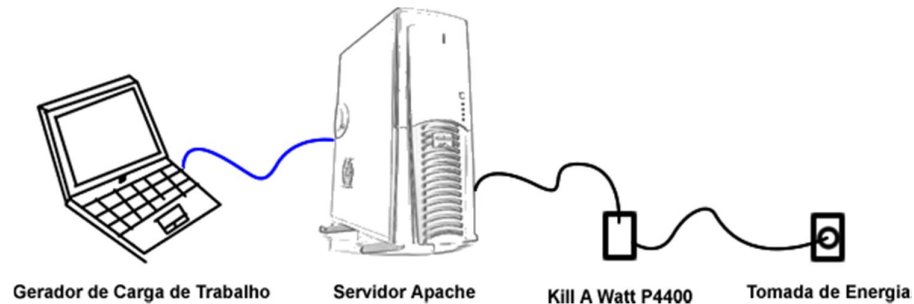


Figura 22 - Cenário do Servidor Apache.

b) Taxa de Utilização

Todas as vezes que foi utilizado o comando *top* do sistema Debian Squeeze, para avaliar a taxa de utilização do hardware, verificou-se que a máquina ficava subutilizada. Sendo que, com a carga de trabalho, mais os processos do sistema operacional, havia uma utilização de apenas 8% do processador e 1% da memória *RAM*.

c) Consumo de energia

Durante os 3 dias, conforme a figura 23, e com a carga de trabalho específica, o servidor Apache consumiu um total de 2,52 *kWh*, como pode-se verificar na figura 24.



Figura 23 – Tempo total do monitoramento do Servidor Apache



Figura 24 - Consumo acumulado do Servidor Apache.

A tabela 4 detalha o consumo durante todo monitoramento.

Tabela 4 - Consumo detalhado do Servidor Apache.

TEMPO	CONSUMO ACUMULADO
0 horas	0 kWh
12 horas	0,42 kWh
24 horas	0,84 kWh
36 horas	1,26 kWh
48 horas	1,68 kWh
60 horas	2,1 kWh
72 horas	2,52 kWh

4.3.2 Servidor IIS

a) Cenário

A figura 25 exemplifica o cenário.

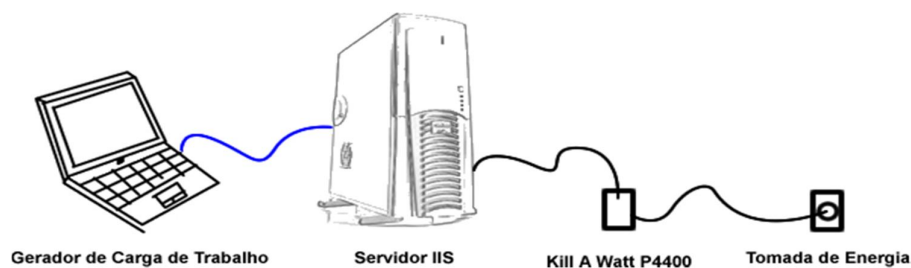


Figura 25 - Cenário do Servidor IIS.

a) Taxa de Utilização

Todas as vezes que foi utilizado o Gerenciador de Tarefas do sistema *Windows Server 2012*, verificou-se que a máquina também ficava subutilizada. Sendo que com a carga de trabalho, mais os processos do sistema operacional, havia uma utilização de apenas 6% do processador e 3% da memória *RAM*.

b) *Consumo de energia*

Durante os 3 dias, conforme a figura 26, e com a carga de trabalho específica, o Servidor IIS consumiu um total de 2,46 kWh, como pode-se verificar na figura 27.



Figura 26 - Tempo total do monitoramento do Servidor IIS.



Figura 27 - Consumo acumulado do Servidor IIS.

A tabela 5 detalha o consumo durante todo monitoramento.

Tabela 5 - Consumo detalhado do Servidor IIS.

TEMPO	CONSUMO ACUMULADO
0 horas	0 kWh
12 horas	0,41 kWh
24 horas	0,82 kWh
36 horas	1,23 kWh
48 horas	1,64 kWh
60 horas	2,05 kWh
72 horas	2,46 kWh

4.3.3 Servidor AD

a) Cenário

A figura 28 exemplifica o cenário.

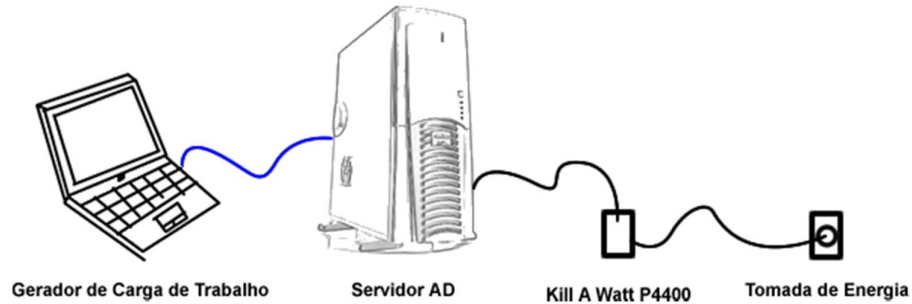


Figura 28 - Cenário do Servidor AD.

b) Taxa de Utilização

A maioria das vezes que foi utilizado o utilizando o Gerenciador de Tarefas do sistema Windows Server 2012, verificou-se que o hardware do servidor também ficava subutilizado. Sendo que com a carga de trabalho, mais os processos do sistema operacional, havia uma utilização de apenas 2% do processador e 2% da memória *RAM* e de 8 em 8 horas acontecia um pico de processamento, pois neste intervalo o Cobian Backup compactava a pasta *c:\tcc*.

c) Consumo de energia

Durante os 3 dias, conforme a figura 29, e com a carga de trabalho específica, o Servidor AD consumiu um total de 2,48 *kWh*, como pode-se verificar na figura 30.



Figura 29 – Tempo total do monitoramento do Servidor AD.



Figura 30 - Consumo acumulado do Servidor AD.

A tabela 6 detalha o consumo durante todo monitoramento.

Tabela 6 - Consumo detalhado do Servidor AD.

TEMPO	CONSUMO ACUMULADO
0 horas	0 kWh
12 horas	0,41 kWh
24 horas	0,82 kWh
36 horas	1,24 kWh
48 horas	1,65 kWh
60 horas	2,07 kWh
72 horas	2,48 kWh

4.4 Monitoramento dos hipervisores

Os dois hipervisores (VMware ESXi 5.1 e XCP 1.6) foram instalados na mesma máquina, um por vez.

Para conseguir uma utilização que ficasse entre de 30% a 65% do hardware da máquina, foram virtualizados, utilizando cada hipervisor, os seguintes servidores:

- 3 servidores Apache;
- 2 servidores IIS e
- 1 servidor AD.

Os servidores web virtualizados receberam a carga de trabalho especificada no subcapítulo 4.2.2 e o servidor AD virtualizado recebeu a carga de trabalho especificada no subcapítulo 4.2.3. A leitura do consumo de energia elétrica foi realizada de 12 em 12 horas.

Cada hipervisor foi monitorado durante 3 dias. A leitura do consumo de energia elétrica foi realizada de 12 em 12 horas.

4.4.1 Ativação da Virtualização Assistida por Hardware

Como o tipo de virtualização escolhido foi a Virtualização Assistida por Hardware, foi necessário habilitar este recurso na BIOS da placa mãe antes da instalação dos hipervisores, conforme é mostrado na figura 31.

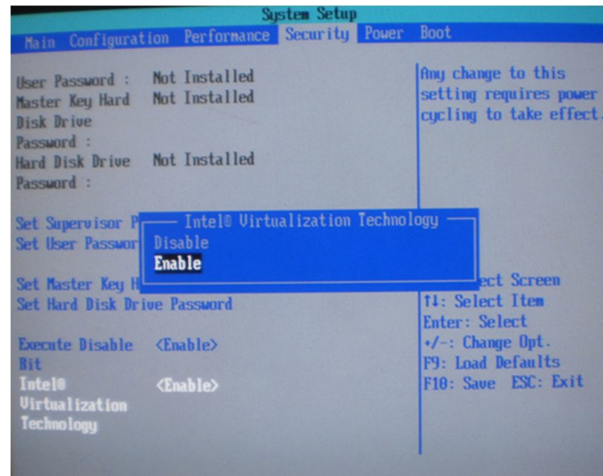


Figura 31 - Setup da placa mãe DH61HO.

A seguir, o monitoramento detalhado de cada hypervisor, expondo cenário, taxa de utilização e consumo de energia.

4.4.2 Hipervisor VMware ESXi 5.1

a) Cenário

A figura 32 exemplifica o cenário.

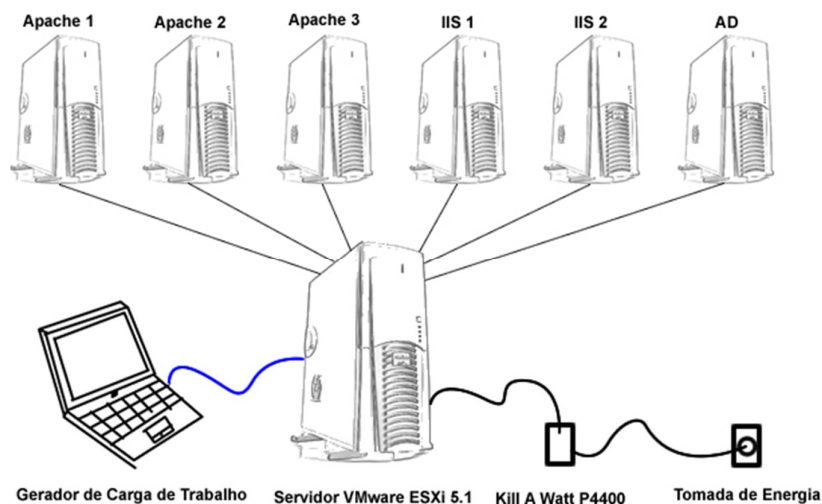


Figura 32 - Cenário do hipervisor VMware ESXi 5.1.

b) Taxa de utilização

Utilizando o VMware ESXi 5.1, com as 6 máquinas virtuais, a utilização do processador ficou entre 28% e 32%. O software utilizado para verificar esta utilização foi o vSphere Client, conforme mostra a figura 33.

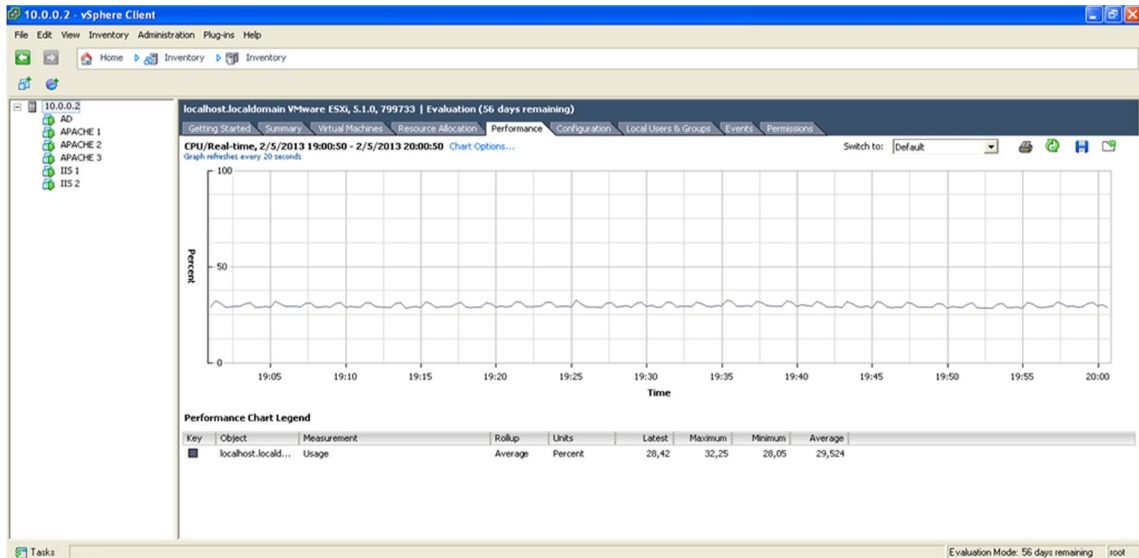


Figura 33 - Taxa de utilização do processador com o VMware ESXi 5.1.

Já utilização da memória RAM ficou entre 58% e 67%. O software utilizado para verificar esta utilização também foi o vSphere Client, como mostra a figura 34.

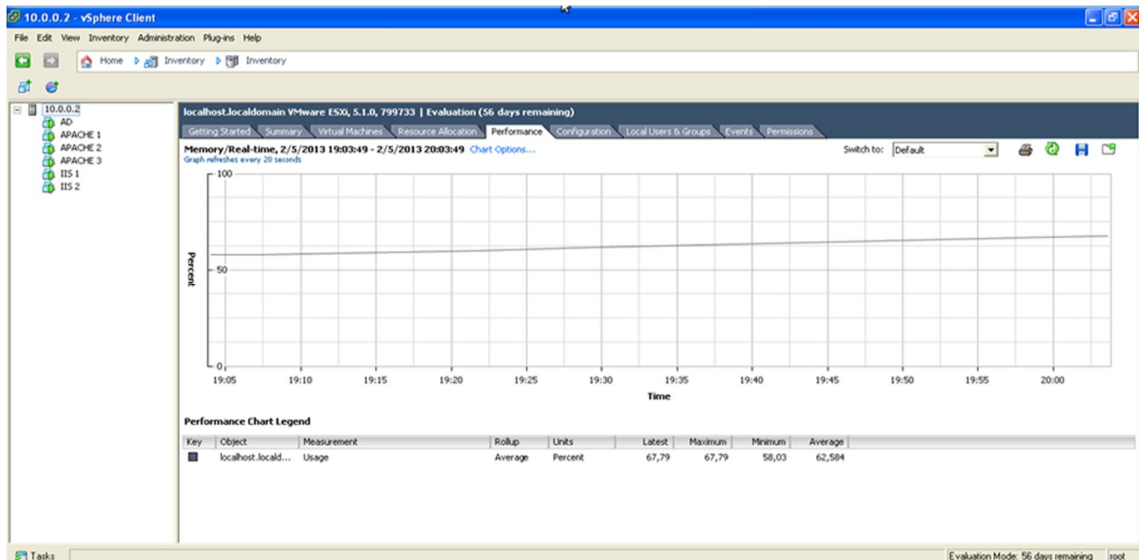


Figura 34 - Taxa de utilização da memória RAM com o VMware ESXi 5.1.

c) *Consumo de energia*

Durante os 3 dias, como retratado a figura 35, e com os 6 servidores virtualizados recebendo a carga de trabalho especifica, o hipervisor VMware ESXi 5.1 consumiu um total de 3,17 kWh, conforme a figura 36.



Figura 35 - Tempo total do monitoramento do hipervisor VMware ESXi 5.1.

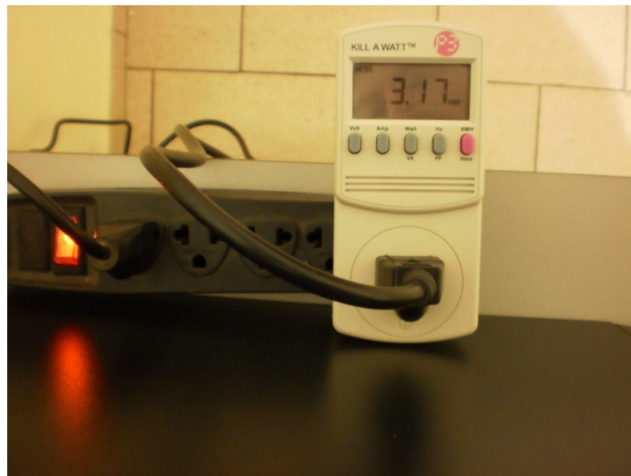


Figura 36 - Consumo acumulado do hipervisor VMware ESXi 5.1.

A tabela 7 detalha o consumo durante todo monitoramento.

Tabela 7 - Consumo detalhado do hipervisor VMware ESXi 5.1.

TEMPO	CONSUMO ACUMULADO
0 horas	0 kWh
12 horas	0,52 kWh
24 horas	1,04 kWh
36 horas	1,58 kWh
48 horas	2,11 kWh
60 horas	2,65 kWh
72 horas	3,17 kWh

4.4.3 Hipervisor XCP 1.6

a) Cenário

A figura 37 exemplifica o cenário.

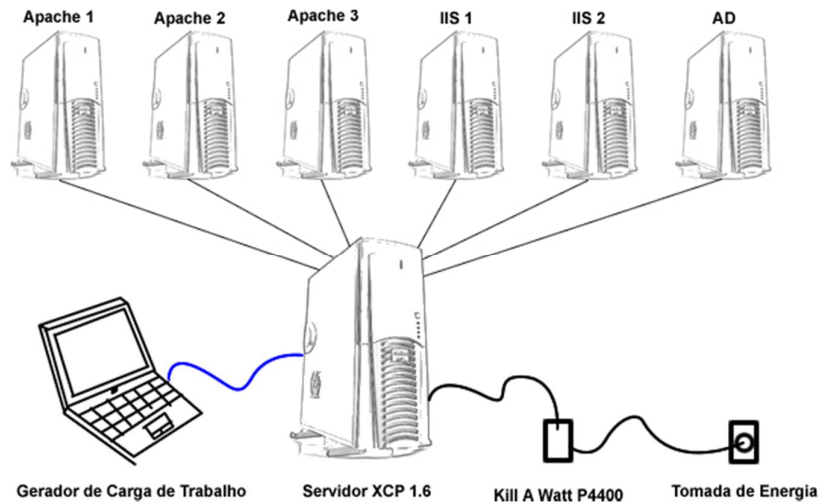


Figura 37 – Cenário do hipervisor XCP 1.6.

b) Taxa de Utilização

Utilizando o XCP 1.6, com as 6 máquinas virtuais, a utilização do processador ficou na faixa de 62% e a memória RAM na faixa de 78%. O software utilizado para verificar esta utilização foi o Open Xen Manager, conforme a figura 38.

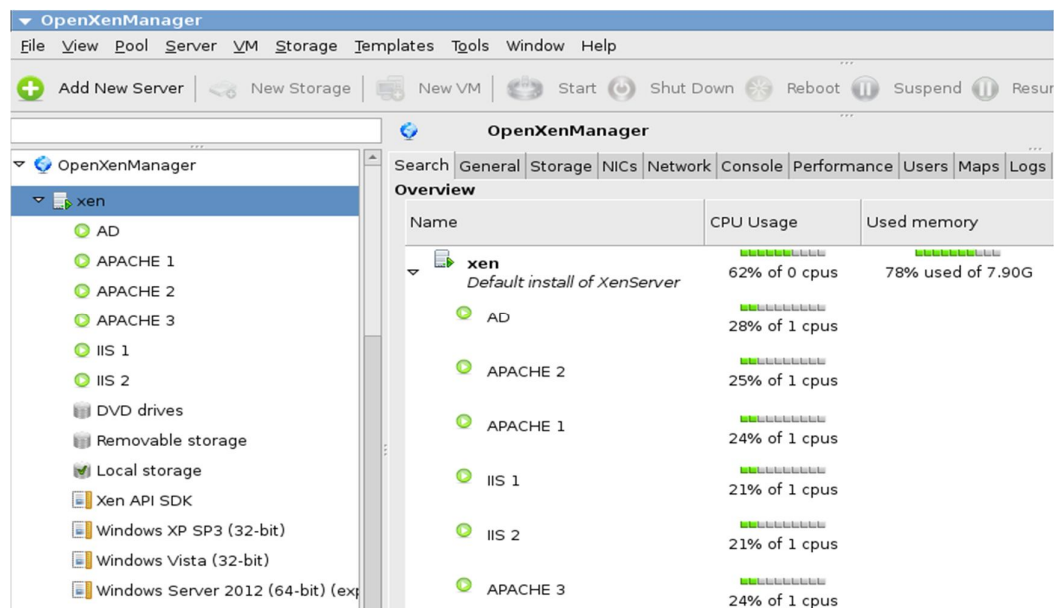


Figura 38 - Taxa de utilização do hardware com o XCP 1.6.

c) *Consumo de energia*

Durante os 3 dias , conforme a figura 39, e com os 6 servidores virtualizados recebendo a carga de trabalho especifica, o hipervisor XCP 1.6 consumiu um total de 3,76 *kWh*, conforme a figura 40.



Figura 39 - Tempo total do monitoramento do hipervisor XCP1.6.



Figura 40 - Consumo acumulado do hipervisor XCP 1.6.

A tabela 8 detalha o consumo durante todo monitoramento.

Tabela 8 - Consumo detalhado do hipervisor XCP 1.6.

TEMPO	CONSUMO ACUMULADO
0 horas	0 kWh
12 horas	0,62 kWh
24 horas	1,24 kWh
36 horas	1,88 kWh
48 horas	2,55 kWh
60 horas	3,14 kWh
72 horas	3,76 kWh

5. Resultados e discussão

5.1 Resultados

5.1.1 Diferença de consumo entre VMware ESXi 5.1 x XCP 1.6

Com base no experimento realizado, o hipervisor VMware ESXi 5.1 mostrou-se mais eficiente no quesito economia de energia elétrica do que o XCP 1.6.

Durante os 3 dias de monitoramento, o VMware ESXi 5.1, com os 6 servidores virtualizados recebendo as cargas de trabalho específicas, consumiu um total de 3,17 kWh. Já o XCP 1.6, nas mesmas condições, consumiu um total de 3,76 kWh.

Sendo assim, houve uma diferença favorável ao VMware ESXi 5.1 de 0,59 kWh. A tabela 9 detalha o consumo durante todo monitoramento do VMware ESXi 5.1 e do XCP 1.6.

Tabela 9 - Consumo detalhado dos hipervisores.

TEMPO	CONSUMO ACUMULADO	CONSUMO ACUMULADO
	VMWARE ESXi 5.1	XCP 1.6
0 horas	0 kWh	0 kWh
12 horas	0,52 kWh	0,62 kWh
24 horas	1,04 kWh	1,24 kWh
36 horas	1,58 kWh	1,88 kWh
48 horas	2,11 kWh	2,55 kWh
60 horas	2,65 kWh	3,14 kWh
72 horas	3,17 kWh	3,76 kWh

Para facilitar a visualização do consumo dos hipervisores, foi feito o gráfico 1.

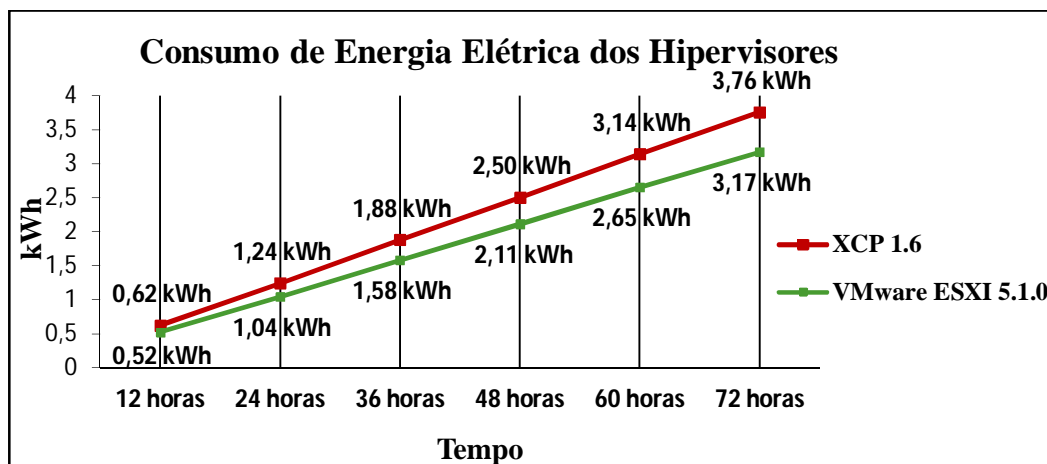


Gráfico 1 – Consumo XCP 1.6 e VMware ESXi 5.1

5.1.2 Simulação de consumo

Nos testes, foram virtualizados 6 servidores, sendo eles: 3 Apache, 2 IIS e 1 AD. Caso estes servidores fossem servidores convencionais, e recebessem a mesma carga de trabalho, provavelmente eles consumiriam um total de 14,96 kWh durante os 3 dias, como mostra a tabela 10.

Tabela 10 - Simulação do Consumo de Energia Elétrica dos Servidores Convencionais.

SERVIDORES	CONSUMO INDIVIDUAL	QUANTIDADE VIRTUALIZADO	TOTAL POR TIPO DE SERVIDOR
Apache	2,52 kWh	3	7,56 kWh
IIS	2,46 kWh	2	4,92 kWh
AD	2,48 kWh	1	2,48 kWh
Consumo Simulado Total			14,96 kWh

Com a virtualização desses 6 servidores, utilizando o VMware ESXi 5.1, a economia de energia foi de 11,79 kWh e utilizando o XCP 1.6 foi de 11,20 kWh, conforme a tabela 11.

Tabela 11 - Economia de energia utilizando a virtualização.

HIPERVISOR	CONSUMO	CONSUMO SIMULADO TOTAL	ECONOMIA
VMware ESXi	3,17 kWh	14,96 kWh	11,79 kWh
XCP 1.6	3,76 kWh	14,96 kWh	11,20 kWh

Para facilitar a visualização do consumo dos hipervisores e da simulação de consumo dos servidores convencionais, foi construído o gráfico 2.

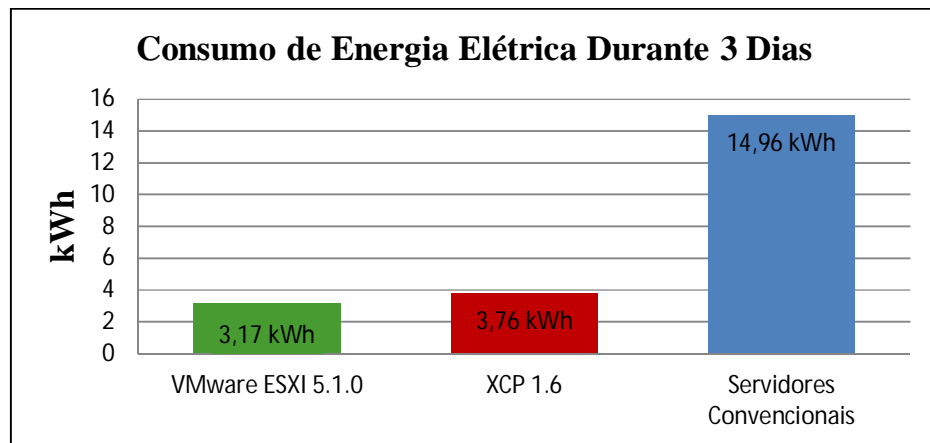


Gráfico 2 – Economia de energia elétrica

5.1.3 Economia monetária

Segundo o site da ANEEL (2013), o custo do *kWh* cobrado pela CEMIG (Companhia Energética de Minas Gerais) é de R\$ 0,347. Sendo assim, durante um ano, utilizando o VMware ESXi 5.1 a economia chegaria a R\$ 497,75 e utilizando o XCP 1.6 a economia seria de R\$ 472,75, conforme a tabela 12.

Tabela 12 - Economia monetária.

HIPERVISOR	CONSUMO DE ENERGIA DURANTE 3 DIAS	CONSUMO SIMULADO SERVIDORES CONVENCIONAIS	ECONOMIA DE ENERGIA DURANTE 3 DIAS	ECONOMIA DE ENERGIA DURANTE 1 ANO	ECONOMIA MONETÁRIA DURANTE 1 ANO
VMware ESXi 5.1	3,17 kWh	14,96 kWh	11,79 kWh	1.434,45 kWh	R\$ 497,75
XCP 1.6	3,76 kWh		11,2 kWh	1.362,67 kWh	R\$ 472,85

5.2 Discussão

5.2.1 Diferença de desempenho dos hipervisores

Durante essa avaliação, o VMware ESXi 5.1 obteve um melhor desempenho energético do que o XCP 1.6. As possíveis causas são:

(i): o VMware ESXi 5.1 gerencia melhor os hóspedes do que o XCP 1.6, gastando assim menos energia. Durante os testes dos hipervisores, o VMware ESXi 5.1 utilizou o hardware de forma mais eficiente.

(ii): o *Advanced Configuration and Power Interface* (ACPI) o sucessor do *DPMA*, usado em computadores 486 e *Pentium*. A diferença básica entre os dois é que no ACPI o gerenciamento de energia é feito pelo sistema operacional e não pelo *BIOS*. Isso permite um gerenciamento muito mais avançado, permitindo recursos como diminuir a frequência de operação do processador.

Supõe-se então que o ACPI do XCP 1.6 não seja tão eficiente quanto o do VMware ESXi 5.1 fazendo com que ele consuma mais energia.

Durante o trabalho, foi monitorando o servidor sem sistema operacional algum. O consumo de energia foi elevado, 4,02 *kWh*, durante os 3 dias. Isto comprova que se o ACPI tem influência direta no consumo de energia elétrica.

6. Conclusão

Ficou comprovado neste trabalho que existe real economia de energia elétrica com a virtualização, independente do hipervisor utilizado.

Houve uma diferença favorável ao VMware ESXi 5.1 de 0,59 kWh. Essa diferença é aparentemente pequena, mas, se o VMware ESXi 5.1 mantivesse o mesmo padrão de consumo das 72 horas, ele poderia ficar ligado 12 horas a mais que o XCP.

Levando em consideração a incansável busca pela economia de recursos naturais nos dias de hoje, os resultados deste trabalho são de grande valor. Cabe aos profissionais da computação dar a importância devida para a virtualização, levando em consideração a real economia de energia elétrica que ela proporciona, independente do hipervisor utilizado.

7. Referências bibliográficas

ALMEIDA, João Paulo Vieira de. **Impacto de plataformas de virtualização no consumo energético: um estudo comparativo entre Xen e KVM.** 2012. 67 f. TCC (Graduação em Engenharia de Computação) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

ANEEL. **Conheça as Tarifas da Classe de Consumo Residencial de uma Concessionária.** Disponível em: <<http://www.aneel.gov.br/area.cfm?idarea=493>>. Acesso em: 08 maio 2013.

AZEVEDO, Marcus Vinícius do Patrocínio. **Otimização de recursos e economia de energia em clusters usando virtualização.** 2010. 76 f. Dissertação (Mestrado) - Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2010.

BRADESCO, Fundação. **O QUE É Windows?** Disponível em: <http://www.fundacaobradesco.org.br/vv-apostilas/win_p1.htm>. Acesso em: 08 abr. 2013.

CARISSIMI, Alexandre. **Virtualização: Princípios Básicos e Aplicações.** In: ERAD, 2009, Caxias do Sul: UFRGS, 2009. p. 40 - 68.

CASTRO, João. **Entendendo as limitações do Esxi 5.1 FREE.** Disponível em: <<http://communities.vmware.com/blogs/joaocastro/2012/09/18/entendendo-as-limita%C3%A7%C3%B5es-do-esxi-51-free>>. Acesso em: 08 abr. 2013.

DEBIAN. **Afinal de contas, o que é o Debian?** Disponível em: <<http://www.debian.org/intro/about#what>>. Acesso em: 08 abr. 2013.

FERNANDES, Almir Dominicini. **Avaliação experimental de técnicas de virtualização através de balanceamento de carga em clusters de computadores.** 2010. 86 f. Dissertação (Mestrado em Engenharia de Sistemas e Computação) - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2010.

FERREIRA, Josiane Bezerra. **Análise de Formas de Medição de Consumo de Energia Elétrica no Setor Residencial.** 2012. 74 f. Monografia (Graduação em Engenharia da Computação) - Departamento de Centro de Informática, Universidade Federal de Pernambuco, Recife, 2012.

FERREIRA, Osvaldo Manuel Dias. **O nível de implementação do cloud computing nas empresas portuguesas.** 2012. 103 f. Dissertação (Mestrado em Sistemas de Informação) - Universidade Portucalense Infante D. Henrique, Porto, 2012.

GALOSSO, Ricardo. **Instalando o XCP 1.6.** Disponível em: <<http://guiadoti.blogspot.com.br/2013/02/instalando-o-xcp-16.html>>. Acesso em: 17 abr. 2013.

GROSMANN, Diego et al. **Estudo comparativo sobre o uso do VMware e Xen Server na virtualização de Servidores.** Mossoró: Ifpi, 2012. 8 p.

HALLIDAY, David; RESNICK, Robert; WALKER, Jearl. **Fundamentos de Física.** 8. ed. Rio de Janeiro: Ltc, 2009. 394 p.

HAMACKER. **Administrando servidores Xen com o Linux.** Disponível em: <<http://hamacker.wordpress.com/2011/06/22/administrando-servidores-xen-com-o-linux/>>. Acesso em: 21 abr. 2013.

- IBICT. **O QUE É APACHE?** Disponível em:
<http://seer.ibict.br/index.php?option=com_content&task=view&id=235&Itemid=74>.
Acesso em: 08 abr. 2013.
- ITS. **Virtualização.** Disponível em: <<http://www2.itssolucoes.com.br/virtualizacao>>. Acesso em: 04 jan. 2013.
- MACHADO, Francis Berenger; MAIA, Luiz Paulo. **Arquitetura de sistemas operacionais.** 4. ed. Rio de Janeiro: Ltc, 2007. 324 p.
- MARIN, Paulo Sérgio. **Data centers - Desvendando cada passo: conceitos, projeto, infraestrutura física e eficiência energética.** Rio de Janeiro: Érica, 2011. 318 p.
- NEIVA, Adriana Silva. **Estudo comparativo de desempenho em ambiente tradicional e virtualizado aplicado a banco de dados em plataforma x86.** 2010. 113 f. Mestrado (Mestrado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, 2010.
- P3 INTERNATIONAL. **Kill A Watt: Empowers you to save \$100.** Disponível em:
<<http://www.p3international.com/products/special/P4400/P4400-CE.html>>. Acesso em: 12 jan. 2013.
- RIBAS, Marcelo. **Consolidação de servidores: estudo de caso.** 2008. 41 f. TCC (Graduação em Ciência da Computação) - Centro Universitário Feevale, Novo Hamburgo, 2008.
- RODRIGUES, Elder de Macedo. **Realocação de recursos em ambientes virtualizados.** 2009. 80 f. Dissertação (Mestrado em Ciência da Computação) - Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2009.
- SALGADO, Gustavo Trein. **Estudo sobre o impacto energético de máquinas virtuais em um sistema computacional físico.** 2011. 80 f. TCC (Graduação em Engenharia de Computação) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2011.
- SOUSA, Flávio R. C.; MOREIRA, Leonardo O.; MACHADO, Javam C.. Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios. **ERCEMAPI 2009**, Ceará, n. , p.1-26, 28 out. 2009. Anual.
- TROY, Ryan; HELMKE, Matthew. **VMware cookbook.** 2. ed. Sebastopol: O'reilly, 2012. 360 p.
- VERAS, Manoel. **Virtualização: componente central do datacenter.** São Paulo: Brasport, 2011. 364 p.
- VERAS, Manoel; KASSICK, Rodrigo. **Virtualização de Servidores.** Rio de Janeiro: RNP/ESR, 2011. 422 p.
- VIRGILIO. **O que é Shell Script?** Disponível em: <<http://www.vivaolinux.com.br/artigo/O-que-e-Shell-Script>>. Acesso em: 12 abr. 2013.
- VMWARE. **VMware ESX e VMware ESXi: Os hypervisores líderes do mercado com produção comprovada.** Disponível em:
<http://www.vmware.com/files/br/pdf/products/VMW_09Q1_BRO_ESX_ESXi_BR_A4_P6_R2.pdf>. Acesso em: 03 jan. 2013.

8. Glossário

AMD-V: nome dado a tecnologia de virtualização da AMD.

API (Application Programming Interface): um conjunto de funções e sub-rotinas usadas para ativar um determinado dispositivo no programa.

Aplicativos: programa executável destinado a auxiliar o usuário na realização de determinadas tarefas num computador.

Arquitetura x86: arquitetura de processadores criada e lançada pela Intel em 1978, essa arquitetura suportava aplicativos de 8 bits e 16 bits. Com a evolução dos processadores Intel, a arquitetura x86 passou a suportar aplicativos e sistemas de 32 bits. Atualmente está presente na maioria dos processadores Intel e AMD, equipando quase todos os desktops, notebooks e netbooks.

ASP.NET (Active Server Pages): linguagem de programação para a Internet.

Backup: cópia de segurança, que permitem o resgate de informações importantes ou programas em caso de falha do disco rígido.

BIOS (Basic Input Output Services): primeira camada de software do sistema, responsável por "dar a partida" no micro.

Boot: procedimento de carregar um sistema operacional na memória RAM principal, executado por um pequeno programa, contido no BIOS da memória ROM, que instrui o microprocessador sobre como proceder para localizar o sistema operacional no disco e carregá-lo na memória.

Cargas de trabalho: conjunto de todas as informações de entrada que um sistema recebe durante qualquer período de tempo determinado.

Data Center: é o local onde são concentrados os equipamentos de processamento e armazenamento de dados de uma empresa ou organização.

Desktop: termo designado ao computador de mesa não portátil.

Display LCD: modalidade de tela para monitores, composta por duas lâminas de vidro, entre os quais há uma camada de cristal líquido.

DPMA (Dynamic Power Management Architecture): sistema de gerenciamento de consumo de energia.

Drivers: itens de software que permitem que o computador se comunique com um acessório específico, como uma determinada placa. Cada acessório exige um driver específico.

E/S (entrada e saída): descreve qualquer troca de dados entre dois dispositivos (a memória RAM e o processador por exemplo) ou mesmo entre o computador e o usuário (um texto digitado no teclado (entrada), uma página impressa na impressora (saída)).

Extranet: recursos Web com acesso controlado disponibilizados para uma comunidade limitada de clientes ou parceiros de negócio.

GB (Gigabyte): unidade de quantidade de informações usado na especificação da capacidade de memória de computadores, tamanho de arquivos. Múltiplo do byte.

GNU: sistema operacional multitarefa e multiusuário, compatível com o Unix, distribuído pela Free Software Foundation.

GPL (General Public License): Licença Pública Geral, designação da licença para software livre idealizada por Richard Matthew Stallman em 1989.

h: abreviação de hora.

Hardware: corresponde a parte física do sistema computacional. São as placas, peças, partes, circuitos e componentes do sistema.

HD (forma abreviada para Hard disk): disco rígido, é o local onde são guardadas as informações. Elas permanecem gravadas mesmo quando o computador é desligado.

HTML (Sigla para Hypertext Markup Language): linguagem padrão para criação de páginas em um site.

HTTP (Sigla para Hyper Text Transfer Protocol): Protocolo de Transferência de Hipertexto, protocolo básico para o trânsito de dados e informações na Internet.

Intel-VTx: nome dado a tecnologia de virtualização da Intel.

Internet: maior conglomerado de redes de comunicações em escala mundial, ou seja, vários computadores e dispositivos conectados em uma rede mundial e dispõe milhões de dispositivos interligados pelo protocolo de comunicação TCP/IP que permite o acesso a informações e todo tipo de transferência de dados.

Intranet: conjunto de recursos Web usado internamente, em geral mantido dentro de um imite seguro conhecido como firewall.

IP (Sigla para Internet Protocol): é o protocolo responsável pelo roteamento de pacotes entre dois sistemas que utilizam a família de protocolos TCP/IP, desenvolvida e usada na Internet.

Java: linguagem de programação multiplataforma.

KB (Kilobyte): unidade de quantidade de informações usado na especificação da capacidade de memória de computadores, tamanho de arquivos. Múltiplo do byte.

Kernel: núcleo de um sistema operacional. Local onde estão as rotinas de instruções básicas para operação de um computador.

kWh (Sigla de quilowatt-hora): medida de energia usualmente utilizada em eletrotécnica.

Linux: Sistema operacional, similar ao Unix, de código aberto (Open Source) cujo núcleo foi criado pelo finlandês Linux Tovalds e mantido por uma grande comunidade desde 1991.

m: abreviação de minuto(s).

Mainframe: designação dada aos antigos computadores de grandes porte e desempenho.

MB (Megabyte): unidade de quantidade de informações usado na especificação da capacidade de memória de computadores, tamanho de arquivos. Múltiplo do byte.

Memória: Circuitos, componentes ou partes mecânicas de um computador que armazenam informações.

Microsoft: empresa líder mundial em softwares para computadores pessoais e empresariais, fundada, em 1975, por Willian (Bill) H. Gates e Paul Allen.

Nobreaks: equipamento que permite que o microcomputador e seus periféricos (quando ligados a ele) continuem a funcionar por um determinado período de tempo, quando há interrupção de fornecimento de energia elétrica.

Notebook: um computador portátil, leve, designado para poder ser transportado e utilizado em diferentes lugares com facilidade.

Placa mãe: é uma placa de circuito impresso, que serve como base para a instalação dos demais componentes de um computador, como o processador, memória RAM, os circuitos de apoio, as placas controladoras, os slots do barramento e o chipset.

Processador: É o componente fundamental do sistema computacional, responsável por todos os cálculos, decisões e controle do sistema.

Python: é uma linguagem de programação de alto nível , interpretada, imperativa, orientada a objetos, de tipagem dinâmica e forte.

RAM (Sigla para Random Acces Memory): memória de acesso randômico, área da memória de um computador, cujo conteúdo pode ser lido e gravado. Armazena temporariamente dados e instruções de que o processador necessita para execução de tarefas.

Rede: conjunto de computadores interligados, de modo a permitir aos usuários o compartilhamento de programas e arquivos.

Root: usuário especial usado para a administração do sistema.

Script: conjunto de comandos e parâmetros escritos numa determinada linguagem de programação para a execução automática de tarefas.

Servidor: computador que oferece recursos especializados, para os demais computadores da rede.

Software: o mesmo que Programa de Computador.

TI (Sigla para Tecnologia da Informação): é a área de conhecimento responsável por criar, administrar e manter a gestão da informação através de dispositivos e equipamentos para acesso, operação e armazenamento dos dados

top: comando de um sistema operacional Linux, usado para visualização de processos.

W (Sigla de watt): unidade de potência do Sistema Internacional de Unidades.

Web: sinônimo mais conhecido do serviço World-Wide-Web (WWW) da internet. É a interface gráfica da internet que torna os serviços disponíveis totalmente transparentes para o usuário e ainda possibilita a manipulação multimídia da informação.

Windows: sistema operacional, multitarefa desenvolvido pela Microsoft Corporation, a partir de 1983. Apresenta um ambiente de trabalho em que os aplicativos são dispostos na forma de janelas; daí o nome.