VMWARE ESXi E A EVOLUÇÃO HISTÓRICA DAS FALHAS DE SEGURANÇA[1]

Wladimir Souza Campos Vergal[2]
Alexandre Garcia Aguado[3]

RESUMO

O presente estudo se propõe a verificar o quanto os fabricantes de Tecnologia da Informação (TI) efetivamente progridem no campo da Segurança da Informação (SI). Será apresentado o contexto atual desde as demandas que o mercado tem, as tecnologias e até os conceitos atuais da SI. Os recursos, as possibilidades e motivações levaram a escolha de uma linha de produtos, bastante popular, da área de infraestrutura de TI: o *VMWARE ESXi*. É um programa para virtualizar máquinas para uso como servidores. Para alcançar esse objetivo foi feito uma revisão de literatura e um estudo prático dividido em duas partes: num estudo quantitativo à partir de dados sobre as vulnerabilidades da linha de produtos e no estudo baseado em testes feitos em um laboratório montado para esse estudo com seis servidores *VMWARE*, abordando três vulnerabilidades documentadas para esses produtos. O resultado que abrangeu todas as fases do estudo foi de que o produto amadureceu no campo da Segurança de Informação, especificamente à partir de 2010, quando a versão ESXi 4.1 foi lançada.

Palavras-chave: Segurança da Informação; Vulnerabilidades; *VMWARE* ESXi;

ABSTRACT

This study proposes to determine how much the IT manufacturers effectively progress in the Information Security field .The current context will be presented from the demands that the market has, technologies and even the current concepts of information security. Resources, possibilities and motivations led to choose only a product line, quite popular in the IT infrastructure area: VMWARE ESXi. Is a program for virtualized machines for use as servers. To achieve this goal was made a literature review and a case study in two parts: a quantitative study starting from data on the line vulnerabilities products and the study Based on significant tests in a laboratory set up for this study with six servers VMWARE, addressing three vulnerabilities documented for these products. The result covering all phases of the study was that the product has matured in the field of Information Security, specifically the period starting from 2010, when the ESXi 4.1 version was released.

Keywords: *Information Security; Vulnerabilities; VMWARE ESXi;*

1 INTRODUÇÃO

Contexto e problemática

Desde a década de 60 havia computadores capazes de virtualizar[4] memória e na década de 80 um computador completo. Na década de 80, surgiram os pequenos computadores ou microcomputadores, inicialmente domésticos e hoje em dia temos servidores de grande porte baseados na mesma estrutura. E foi então que em 1998 a *VMWARE* colocou seu marco no mercado de microcomputadores registrando a patente do "*VMWARE Workstation* 1.0" e, no ano seguinte, lançando como seu primeiro produto. Rapidamente conseguiram expandir a empresa e fazer grandes parcerias com fabricantes como Dell, HP e IBM em 2001 quando lançaram a primeira versão de servidor dedicado à virtualização: o *VMWARE ESX Server* 1.0 (*VMWARE* a, 2014). A empresa foi criada e até hoje se especializa nesse segmento de virtualização de microcomputadores, trazendo tecnologias que permitem gerenciar e otimizar recursos físicos de Tecnologia de Informação (TI), pelo agrupamento e flexibilização no uso dos mesmos.

Atualmente a *VMWARE* já está na terceira geração de suas tecnologias de virtualização de servidores, denominada ESXi. Ela é baseada num sistema operacional[5] totalmente desenvolvido para isso com seu próprio núcleo o "*VMkernel*", depois de ter usado um sistema baseado em Linux nos servidores ESX. A virtualização de servidores atualmente encontra-se numa fase madura e bem disseminada nas grandes empresas e tende a ficar cada vez mais popular. (SYMANTEC, 2011, p.10) E dentro dessa

[1] Artigo baseado em Trabalho e Conclusão de Curso (TCC) desenvolvido em cumprimento à exigência curricular do Curso Superior de Tecnologia em Segurança da Informação, depositado em 19 de dezembro de 2015

[2] Tecnólogo em Tecnologia de Segurança da Informação – Fatec Americana – Centro Estadual de Educação Tecnológica Paula Souza ; Contato: wladsp@gmail.com

[3] Prof. Fatec Americana – Mestre em Tecnologia e Inovação; Contato: ale.garcia.aguado@gmail.com

[4] Virtualizar: nesse contexto se refere a disponibilização de um recurso de forma desconectada ao correspondente físico ou "real" (VMware b, p. 3).

[5] Sistema Operacional ou SO: Veras (2011, p 98) diz que "o SO é um alocador de recursos. ... atua como gerente destes recursos ... conforme necessário para execução de tarefas." . Os recursos a que se refere são, tradicionalmente, os recursos do microcomputador ou da máquina. Desde um celular com Android até um servidor com Windows Server 2012 necessitam de um e sobre eles é que rodamos os programas de usuário.

perspectiva que se inseriu o escopo deste trabalho, identificar como tem sido a evolução das diferentes versões dos produtos lançados pela *VMWARE* desde 2008, no que tange a segurança da informação.

A **proposta** deste estudo é através da empresa *VMWARE* e de seu produto o *VMWARE Server* estudar o quanto efetivamente os fabricantes de TI progridem no campo da Segurança da Informação (SI). Pela impossibilidade de se obter informações sobre os investimentos em SI durante o desenvolvimento de novos recursos ou produtos nem outras informações do gênero, a abordagem adotada foi avaliar o produto pelos resultados efetivos em SI. Resultado esse sob a perspectiva do mercado de TI, ou seja, dos seus clientes, que atualmente se preocupam e investem em SI para reduzir os prejuízos causados pelos incidentes.

O estudo ainda se concentrou apenas na família ESXi dentro da linha *VMWARE* Server, em parte porque a outra família, a ESX, foi descontinuada em 2008 e substituída pela família ESXi. Outro motivo é que, no momento em que foi desenvolvido esse estudo, o *VMWARE* Server ESXi já tinha 4 versões lançadas num período de tempo de sete anos e meio permitindo analisar quais versões estão maduras e conhecidas o suficiente pelo mercado de TI para serem comparadas. Dessa forma foi evitado que se expandisse o escopo para além do tempo disponível para desenvolver esse estudo.

Figura 1 - Tabela com Cronologia das versões do ESXi

***VMWARE* Versão**	**Data de lançamento**
v. 3.5 Build[6] 64607	20/02/2008
v. 4.0 Build 164009	21/03/2009
v. 4.1 Build 260247	13/07/2010
v. 5.0 Build 469512	24/08/2011
v. 5.1 Build 799733	10/09/2012
v. 5.5 Build 1331820	22/09/2013
v. 6.0 Build 2494585	12/03/2015

Fonte: Autoria Própria

Diante desta proposta este trabalho tem por objetivo contribuir para a seguinte problemática:

Como o *VMWARE ESXi* tem amadurecido no aspecto da segurança da informação ao longo dos lançamentos de novas versões com novos recursos?

Algumas **hipóteses** consideradas foram:

a) De fato houve uma evolução na segurança juntamente com o acréscimo de novos recursos;

b) As novas versões apresentam novas falhas ou voltam a apresentar falhas corrigidas anteriormente, seja devido aos novos recursos incorporados ou até mesmo nos recursos já existentes; e,

c) O estudo pode-se demonstrar inconclusivo não permitindo estabelecer se houve ou não uma evolução, talvez porque ela já tenha ocorrido em grande parte no lançamento do produto *VMWARE* ESXi que é sucessor do *VMWARE* ESX. Mostra-se aí a necessidade de um novo estudo, mais abrangente.

A **justificativa** para o estudo realizado foi a atenção crescente que a segurança da informação tem recebido pelos usuários em todos os níveis. Uma pesquisa recente mostra inclusive que nunca se investiu tanto em segurança da informação quanto agora, pelo menos na Inglaterra. De acordo com essa pesquisa feita pelo governo Inglês, pequenas empresas chegaram a investir mais do que 25% do orçamento de TI em segurança em 2014 sendo em média 15%. Já as grandes organizações investiram em média 11% no mesmo período. (B.I.S., 2014 p.7). Esta preocupação é fato percebido tanto para especialistas quanto para usuários de TI, mas há pouco material oficial sobre o assunto e não foi encontrada pesquisa como essa publicada no Brasil. Já o volume de incidentes no Brasil é contabilizado, e tem aumentado bastante recentemente. Especificamente na última avaliação divulgada por Marcos Ribeiro, responsável pelo monitoramento de incidentes no Brasil no Cert.br, o aumento de incidentes comparando 2013 com 2014 foi de 197% em geral e de 54% de aumento de incidentes envolvendo servidores (RIBEIRO, 2015).

Em contrapartida as empresas provedoras de serviços e produtos de TI afirmam que seus produtos são mais seguros. (*VMWARE* Education Services, 2012 p.56)

Nesse panorama foi interessante verificar se os fabricantes realmente estão dando foco para melhorar a segurança em TI através de melhorias nos produtos, ou se o foco é de apenas trazer novos

[6] Build – em TI, na área de desenvolvimento de sistemas usa-se esse termo no que equivaleria a um lote de produtos em produtos físicos

recursos e corrigir as falhas à medida que vierem à público, trazendo dessa forma, uma contribuição à área acadêmica de TI.

Metodologia

Quanto aos procedimentos, definiu-se como método para conduzir essa análise duas abordagens: a primeira de pesquisar vulnerabilidades já comprovadas e conhecidas pelo mercado de SI e buscar reproduzi-las num ambiente controlado e equivalente ao descrito na documentação para verificar em seguida se versões mais novas, em situação equivalente, apresentam alguma evolução em termos de segurança em relação a aquela vulnerabilidade. Para isso foi montado um laboratório com quatro servidores ESXi, dois servidores Linux e estações de trabalho para executar ferramentas de SI.

Outra abordagem foi a de escolher uma fonte confiável de informações sobre as vulnerabilidades que inclusive conta com uma avaliação qualitativa aceita pelo mercado do grau de risco que representa cada vulnerabilidade, de forma que foi possível obter a quantidade de vulnerabilidades totais por versão, e de separar a quantidade de vulnerabilidades graves para semelhante comparação.

Se por um lado temos então uma abordagem mais abrangente que incluiu todas as versões definidas para o estudo e todas as vulnerabilidades comprovadas, por outro temos uma análise numa amostra de três de um universo 35 vulnerabilidades graves. O estudo prático se aprofundou numa pesquisa detalhada dessas vulnerabilidades com a respectiva confrontação do seu comportamento em laboratório, trazendo mais conhecimento e experiência a esse estudo.

Pretende-se assim ter todos os dados e informações necessários para se chegar à uma conclusão satisfatória, no sentido de contribuir de fato como conhecimento ao autor e a comunidade.

Este trabalho está organizado da seguinte forma:

- Introdução: contém um breve histórico sobre virtualização de computadores, do fabricante e da linha de produtos foco desse estudo. Também objetiva situar em qual contexto se localiza o escopo desse trabalho;
- Conceitos básicos de SI: é o resultado de um estudo e pesquisa para trazer os conceitos de SI mínimos de forma a garantir o entendimento desse estudo;
- Virtualização: considerando a utilidade principal do produto em análise, sentiu-se a importância de dedicar um espaço maior aos conceitos relacionados à virtualização utilizando inclusive fontes atualizadas como a obra do Prof. Dr. ManoelVeras.de 2011;
- *VMWARE* ESXi e suas vulnerabilidades: após uma introdução para apresentar o banco de dados “NVD” e o critério de risco adotado, o “CVSS”, esta seção apresenta os detalhes do levantamento e organização dos dados bem como seu resultado, se houve ou não uma diferença expressiva da quantidade de vulnerabilidades na linha de produtos;
- Estudo prático: apresenta em detalhe todo o desenvolvimento das três vulnerabilidades testadas bem como o desenrolar dos procedimentos ocorridos em laboratório; e,
- Considerações finais: trará os resultados de cada parte do estudo, e qual a contribuição traz à problemática, para finalmente apresentar as considerações finais com base no estudo. Segue ainda alguns comentários sobre como os conhecimentos e a experiência contribuíram ao autor.

2 CONCEITOS BÁSICOS DE SEGURANÇA DA INFORMAÇÃO

Para entender o funcionamento dos dados utilizados nesse trabalho é importante que alguns conceitos, padrões e ferramentas sejam entendidos claramente. São conceitos básicos utilizados na área de SI e alguns padrões específicos, utilizados internacionalmente para o estudo e gerenciamento de vulnerabilidades e incidentes na área de TI.

2.1 Segurança da informação

A segurança da Informação ou simplesmente SI, é um segmento da segurança que envolve tópicos como os da segurança física, pessoal, de operações, de comunicações e de rede. A NBR ISO 17799 (ABNT, 2005, p.9), define segurança da informação como “a proteção da informação de vários tipos de ameaças para garantir a continuidade do negócio, minimizar o risco ao negócio, maximizar o retorno sobre os investimentos e as oportunidades de negócio”. Já Sêmola dá uma visão menos corporativa e que vemos com frequência na literatura especializada: “uma área do conhecimento dedicada a proteção de ativos da informação contra acessos não autorizados, alterações indevidas e sua indisponibilidade” (SÊMOLA, 2003, p. 43). Essa definição já deixa implícito as três bases clássicas da SI confidencialidade, integridade e disponibilidade, cujo o entendimento é importante para qualquer estudo nessa área.

Sêmola (2003, p.43) define **confidencialidade** como: “Toda informação deve ser protegida de acordo com o grau de sigilo de seu conteúdo, visando a limitação de seu acesso e uso apenas às pessoas para quem elas são destinadas. ” O interessante dessa visão é que ela ressalta que além de desejarmos

sigilo para as informações, a proteção está ligada ao grau da importância da informação em si. Um exemplo claro disso é o dado pelo CERT.br (2006, p.22) quando diz que não é desejado que alguém tenha acesso a todas informações de uma declaração de Imposto de Renda.

Goodrich e Tamassia (2013) acrescentam que a necessidade de manter segredo em determinadas informações é a essência da Segurança da Informação e já era aplicada desde a antiguidade. A cifra de César[7] é o exemplo utilizado com frequência inclusive citado por Singh (1999) como a técnica mais antiga documentada e aplicada militarmente.

Hoje em dia as técnicas têm que ser revistas e aperfeiçoadas com regularidade e os processos usados são mais complexos e desafiadores. Goodrich e Tamassia (2013) enumera as principais técnicas voltadas para confidencialidade como sendo: encriptação, controle de acesso, autenticação e segurança física.

A **encriptação** está envolvida quando a informação é transformada usando algoritmos que envolvem várias operações matemáticas baseadas numa cifra. Para decifrar a informação é necessário o conhecimento prévio da chave correta. Como exemplo de evolução, lembramos que, com o avanço da capacidade computacional dos equipamentos, a tecnologia WEP[8] usada para cifrar os dados em redes sem fio teve que ser substituída pois foi quebrada em 2001. Atualmente a técnica WPA2 é considerada segura e amplamente implementada e utilizada.

O **controle de acesso** é constituído de regras e políticas que definem e restringem quais informações devem ser acessíveis para cada grupo de usuários, e nenhuma informação a mais. Em outros tempos, por redução de custos e falta de tecnologia, o controle se restringia a ter acesso ou não ao banco de dados todo, por exemplo.

Finalmente entende-se por **autenticação** a determinação exata e única da identidade ou função de uma pessoa. São utilizadas várias formas para se obter esses resultados e as vezes uma combinação deles: " Algo que você é": como a imagem da sua digital ou uma captação das veias da palma da mão viva (BRADESCO), "Algo que você sabe": alguma informação que só a pessoa deve saber , como uma senha uma informação pessoal ou de cadastro; "Algo que você tem": pode ser um cartão de tarja magnética ou chip ou um "gerador" de chaves numéricas conhecido por "*token*".

Figura 2 – As três bases da Autenticação

Fonte: Adaptado de (GOODRICH e TAMASSIA, 2013, p.5)

E finalmente a **segurança física** que consiste na utilização de barreiras físicas para limitar o acesso aos recursos computacionais e variam desde portas com chaves mecânicas ou eletrônicas, passando por salas com paredes reforçadas chegando até paredes com malhas de cobres embutidas para criar o efeito físico chamado gaiola de Faraday, que impede a passagem das ondas eletromagnéticas que poderiam expor alguma informação fora do ambiente desejado (GOODRICH e TAMASSIA, 2013, p.4-5).

[7] Cifra de César se refere a uma prática adotada pelo Imperador Romano e seus comandantes ao trocarem informações sigilosas. Para a tecnologia da época, o simples fato de trocar mensagens escritas e não verbais e ainda fazer uma troca de letras foi o suficiente para nunca ocorrer um incidente, uma quebra de sigilo (SINGH, 1999, Introdução).

[8] WEP: é o primeiro protocolo de rede sem fio (1999) e tem como objetivo dar segurança à essas comunicações. Foi muito popular, até que várias falhas ou vulnerabilidades foram descobertas e trazidas à público em 2001, acelerando o uso de novas tecnologias como a WPA2, que é muito mais segura.

Retomando os três conceitos fundamentais, a **Integridade** é definida por Sêmola (2003, p.43) como o objetivo de se manter a informação da forma que seu autor a criou, protegendo de qualquer alteração por pessoas não autorizadas, seja a alteração intencional ou acidental. Outros autores, como Whitman (2012, p. 14) estende o conceito incluindo os sistemas utilizados para o processamento, armazenamento e transmissão desses dados. Ele lembra que existem técnicas utilizando operações matemáticas conhecidas por "função hash" para que se identifique facilmente quando qualquer parte da informação foi alterada do seu estado inicial. Da mesma forma, tanto no armazenamento quanto na transmissão de dados acrescentam-se "bits de paridade[9]", entre outras técnicas, para permitir que se detecte e reverta qualquer alteração da informação seja no destino da informação seja na leitura de um sistema de discos ou memória. Um exemplo semelhante ao dado pelo CERT.br (2006, p.22) é que usamos uma proteção eletrônica como o firewall[10] e um antivírus para proteger os dados de declaração de nosso imposto de renda contra um possível ataque. Ataque esse onde um invasor modifique os dados ou o programa antes ou durante o envio da declaração de Imposto de Renda à Receita Federal.

Concluindo esses conceitos temos o terceiro pilar da SI como a **disponibilidade** da informação que Sêmola (2003, p.43) explica da seguinte forma: "Toda informação deve estar disponível aos seus usuários no momento em que os mesmos dela necessitem para qualquer finalidade".

Vale lembrar que para alcançar esse objetivo todos os sistemas envolvidos sejam servidores, aplicações, estrutura de transmissão de dados entre outros devem ser projetados levando a disponibilidade em consideração e muitas vezes contar com uma redundância. Inclusive a virtualização de servidores e estrutura de redes colabora na estruturação de centros de processamento de dados ou *Data Center* permitindo altos níveis de disponibilidade. O armazenamento dos dados mais críticos conta com tecnologias que suportam a falha física ou lógica de um disco utilizando tecnologia "RAID", ou sistemas de redundância que replicam **tudo em outro local**: a máquina virtual, todos os dados necessários, as aplicações e o sistema operacional.

Em obras modernas outros objetivos são considerados também, mas variam dependendo do autor ou até do ambiente / empresa que for o objeto da SI. Sêmola (2003), por exemplo, comenta em sua obra os objetivos de legalidade para informações que é o de atender as leis e normas do país enquanto a autenticidade é a garantia de que o remetente de uma informação é autêntico, semelhante a uma assinatura em uma carta. Quando não se consegue atender a esses objetivos temos falhas e situações bem definidas e conceituadas no ramo de TI e SI que serão ao assunto nos próximos parágrafos.

2.2 Ameaças e vulnerabilidades

As **vulnerabilidades**, no ramo de SI, são definidas no livro de Whitman (2012, p.11) como sendo uma falha ou fraqueza no sistema ou mecanismo de proteção que o deixa sujeito a um ataque ou dano. Alguns exemplos de vulnerabilidades são falhas nos programas, portas do sistema desprotegidas ou falhas de autenticação. Existem vulnerabilidades bem conhecidas que foram estudadas, documentadas e divulgadas; outras continuam latentes e desconhecidas.

Goodrich e Tamassia (2013, p.2) complementam ao ressaltar que a análise das vulnerabilidades, permite determinar a gravidade daquela falha e qual a facilidade de se reproduzir o ataque. Dessa forma se direciona com maior eficiência as ações preventivas como identificar as máquinas vulneráveis, remover o programa mal-intencionado ou corrigir o sistema através da aplicação de *patches*: correções nos programas para corrigir as vulnerabilidades.

Este estudo é justamente sobre as vulnerabilidades conhecidas, analisando a quantidade que elas ocorrem pois há uma ligação direta com as vulnerabilidades e sua consequência que são as ameaças.

As **ameaças** são os agentes que causem impacto ao negócio ao violar a confidencialidade, integridade ou disponibilidade das informações e os ativos relacionados. Entende-se que o dano deve ser causado pelo uso de alguma vulnerabilidade do mesmo. (SÊMOLA, 2003 p.47)

As ameaças são classificadas frequentemente com base na intencionalidade do agente, que é a classificação adotada por Sêmola (2003), dividindo as ameaças em naturais, involuntárias e voluntárias.

As **ameaças naturais** são aquelas causadas por fenômenos da natureza, fora do controle do homem como: terremotos, maremotos, furacões, incêndios iniciados naturalmente, enchentes entre outras.

Para as **ameaças involuntárias** existe um agente, mas que não tem consciência ou intenção de causá-la. Os exemplos são os danos causados por erros, acidentes e muitas falhas em geral. Um exemplo relacionado a esse estudo seria um conjunto de servidores virtuais ficar indisponível devido a uma vulnerabilidade no código do *VMWARE ESXi* que cause seu travamento numa situação aleatória.

[9] bits de paridade: são dados gerados por operações matemáticas ou lógicas à partir do dado original de forma e criar mecanismos que permitam detectar falhas em armazenamento ou transmissão desses dados.
[10] firewall: programa ou dispositivo de monitora e gerencia as comunicações num computador ou rede para reduzir a ação de ataques ou bloquear fluxo de dados não autorizados à circular pela rede.

Já as **ameaças voluntárias,** reúne as ameaças causadas por agentes humanos que tem a intenção de causar o dano, seja para benefício próprio ou não. Como exemplo podemos citar crackers[11], ladrões cibernéticos, incendiários entre outros. Um exemplo real é a falha no código original do *VMWARE ESXi* 3.5, documentada sob a identificação CVE-2008-2097, que permite que um funcionário mal-intencionado com uma conta de acesso ao servidor consiga modificar seu nível de acesso e ganhar acesso pleno, conhecido por *root*. Esta falha pode ser corrigida pela instalação de pacotes de correções ESXe350-200805501-I-SG . lançado 28 dias após a divulgação da falha em 2008.

É interessante acrescentar ainda o conceito do que é um **ataque** na área de SI, que é entendido como a tentativa de violar um sistema de segurança e/ou a política de segurança de um sistema através de um método ou técnica. A Norma RFC 2828 enfatiza ainda que deve ser um ato inteligente, deliberado do agente. (RFC 2828 apud STALLINGS, 2002, p. 6)

3 VIRTUALIZAÇÃO

Manoel Veras (2011, p.22) define a virtualização, em título de mesmo nome, como sendo a tecnologia que atende à demanda dos negócios atuais no aspecto de que a TI deve se configurar na medida da demanda das aplicações e dos negócios, e com agilidade.

A virtualização aproveita o fato comprovado por muitos institutos de pesquisa como o IDC, que revela que ao longo do tempo apenas 15% da capacidade dos servidores é utilizada. Há então uma ociosidade de recursos de 85% (IDC apud VERAS, 2011, p.23). Mas existem outros custos e questões envolvidos, pois a empresa que tem a necessidade de ter servidores e uma infraestrutura de TI que lhe dê garantias mínimas quanto a segurança das informações ali processadas terá que prover um espaço reservado exclusivamente para eles, consumindo um espaço físico caro: pois exige acesso físico restrito e controlado, ar condicionado, instalação elétrica potente, proteção contra incêndio entre outros recursos na maioria dos casos. Um típico *Data Center*, ainda que pequeno.

O *Data Center* tradicional tem máquinas dedicadas para apenas uma função por exemplo uma para executar algumas aplicações, outra máquina para o banco de dados, uma terceira para controle da infraestrutura como os usuários e seus atributos e mais uma para proteção e controle do uso da Internet. Mesmo que fosse possível colocar tudo em apenas uma ou duas máquinas ainda assim não seria o recomendado devido a diversos motivos defendidos pelas boas práticas de TI, sendo que a principal é que afetaria a disponibilidade dos serviços. É fácil entender: se essa única máquina parar ou precisar de manutenção muito possivelmente a empresa terá que parar!

Mas se pudermos concentrar tudo em duas máquinas de maior capacidade e um conjunto de discos com redundância teríamos o problema resolvido. Melhor ainda se ambas tiverem utilidade, mas ocorrendo um incidente que apenas uma seja capaz de atender sozinha toda a demanda de computação automaticamente. A virtualização permite tudo isso atualmente: pois é possível criar máquinas logicas sem contato direto com a máquina física, e com sistemas operacionais dos mais diversos, sem afetar a compatibilidade com as aplicações.

Veras lembra inclusive que o cenário globalizado exige um ritmo de mudanças rápido nas empresas, elas afetam a organização e os processos, e uma TI dinâmica, pronta para mudança ajuda a empresa ser mais competitiva. Ao mesmo tempo a estabilidade continua sendo uma necessidade, na verdade cada vez maior. (VERAS, 2011, p.22). Qualquer indisponibilidade em qualquer parte da infraestrutura de TI é cada vez menos tolerada. Novamente a virtualização viabiliza isso também, facilitando a implementação de recursos de alta disponibilidade com vários níveis de qualidade e, portanto, com uma variedade de custos. No topo desses recursos, por exemplo, pode-se manter toda uma instalação redundante em outro local, pronta para o funcionamento a qualquer momento pois terá os dados continuamente atualizados.

Como a virtualização de servidores já é uma realidade no mundo de TI há vários anos já temos dados de pesquisa a respeito, como a feita pela empresa de pesquisas Americana, *Forrester Research* (2009), buscando entender a motivação das empresas para utilizar a tecnologia de virtualização. Ela entrevistou os responsáveis pela TI de vinte e nove empresas incluindo vice-presidentes e diretores de 3º nível e as principais descobertas foram:

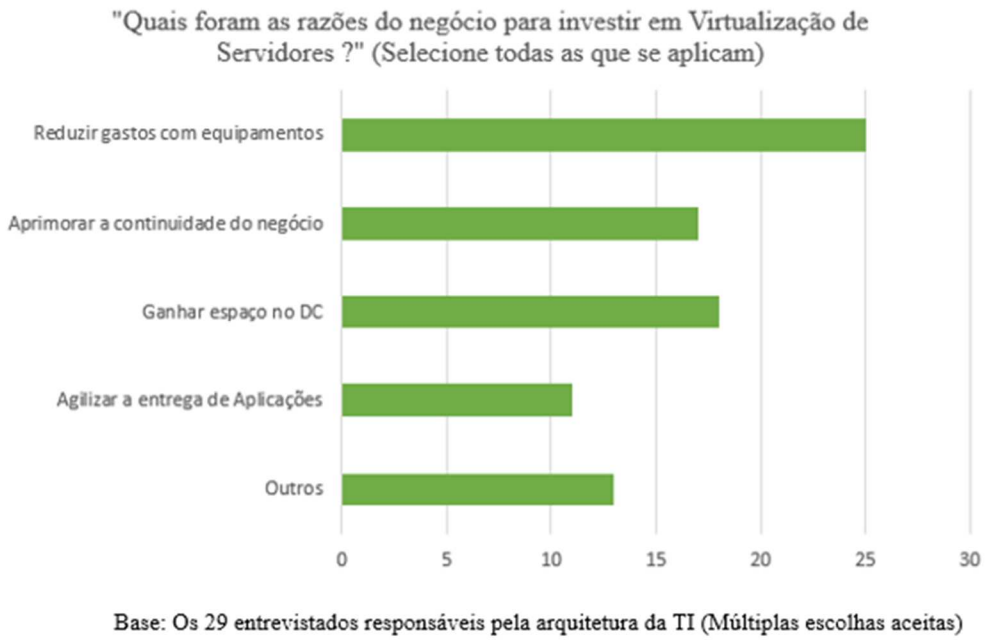
- A TI se tornou mais eficiente;
- A empresa passou a ter um *Time to Market* mais rápido; e,
- A TI passou a ser menos imprevisível, com um aumento da disponibilidade.

O estudo também trouxe a informação de que as empresas buscaram a tecnologia de virtualização por causa das demandas de reduzir os custos de equipamentos, melhorar a condição de recuperação à desastres e da continuidade do negócio, economizar ou liberar espaço físico no *Data Center* reduzindo junto

[11] Craker - *hackers* mal-intencionados.

o consumo de energia elétrica, acelerar a entrega de novas aplicações entre outros motivos (FORRESTER, 2009, p.3). O gráfico na figura 3 ilustra esse levantamento.

Figura 3 Gráfico da entrevista pela *Forrester Research*



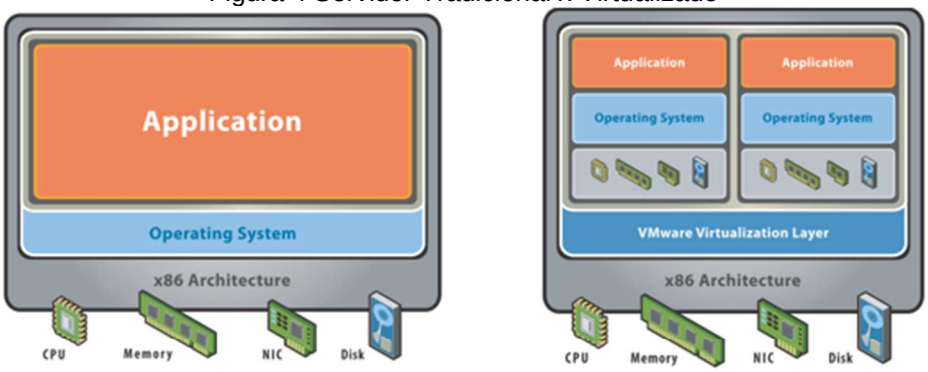
Fonte: Adaptado de FORRESTER, 2009

Assim acabamos percebendo que as empresas querem e precisam do melhor dos dois mundos: da agilidade, mas também da estabilidade obtida por uma disponibilidade maior dos serviços. A virtualização é capaz disso, mas quais são as opções, as tecnologias e como funcionam?

Para eliminar essas dúvidas é necessário entender os conceitos e as formas de virtualização: durante o desenvolvimento desse trabalho foram encontrados vários critérios para classificar as tecnologias de virtualização e o mais popular está relatado, por exemplo, num artigo da *VMWARE* que nos servirá também como introdução ao assunto.

A virtualização em geral permite um melhor uso dos recursos da máquina e uma independência ou desconexão da máquina e sistema operacional trazendo versatilidade e agilidade, pois além de podermos executar, por exemplo um Firewall Linux e uma aplicação Windows sobre a mesma máquina, poderei a qualquer momento trocar a aplicação por uma baseada em outra distribuição Linux. Na figura 4 há uma tabela que ilustra esses fatos.

Figura 4 Servidor Tradicional x Virtualizado

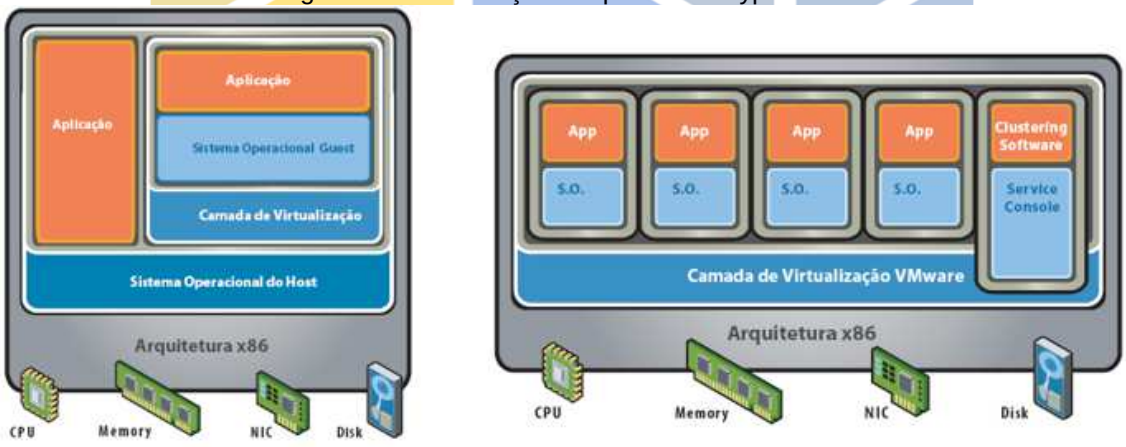


Sem Virtualização	Após a virtualização
Um único SO por máquina	Capaz de gerenciar SO e aplicações como unidades simples encapsulando-as em máquinas virtuais
Ligação forte entre aplicação e máquina	Máquina sem dependência do SO ou de aplicações
Subutilização de recursos	Máquinas virtuais disponíveis para qualquer sistema
Infraestrutura inflexível e cara	

Fonte: Adaptado de *VMWARE* (*VMWARE* b, p. 3)

Já dentre as várias tecnologias de virtualização uma das classificações possíveis são aquelas que utilizam o conceito de *Hypervisor*, que é um tipo específico e novo de aplicação semelhante ao sistema operacional, mas totalmente focado no gerenciamento e na interface dos recursos físicos. Podendo assim reduzir a interface com o usuário ao mínimo, removendo por exemplo a interface gráfica. Opção que aliás aumenta a segurança ao diminuir a superfície de ataque. A *VMWARE* trabalha dessa forma desde a geração anterior, com o *VMWARE* ESX Server de 2001. A Microsoft iniciou em 2008 com o lançamento do Windows Server 2008 que trouxe a tecnologia de virtualização Hyper V e a interface apenas texto denominada *core*

Figura 5 - Virtualização hospedada x Hypervisor



Arquitetura hospedada (hosted)	Arquitetura "hypervisor" ou Bare-Metal
Instalada e executada como aplicação	Núcleo Lean virtualization-centric kernel
Depende de um S.O. hospedeiro para fazer a interface e gerenciamento dos recursos físicos	Console de serviço para agentes Helper Applications
VMWARE Player, Workstation, Oracle VirtualBox, MS Virtual PC	*VMWARE* ESX Server, *VMWARE* ESXi Server, Microsoft Hyper V e Citrix Xen Server
TIPO 2	**TIPO 1**

Fonte: Adaptado de *VMWARE* (*VMWARE* b, p. 4)

Já na obra Virtualização de Manoel Veras encontramos uma classificação mais completa, que nos permite ir mais a fundo pois divide as tecnologias de virtualização em três categorias:

A categoria de **Virtualização Total** se distingue por permitir total desconexão da máquina física e virtual, pois o *Hypervisor* cria um sistema virtual completo. Tem como pontos fortes a facilidade ao migrar de servidores físicos, já existentes, para máquinas virtuais pois não é necessário fazer nenhuma modificação na aplicação tampouco no Sistema Operacional (SO). É considerado muito seguro também por isolar bem as diversas máquinas virtuais em execução. Por outro lado, pode ter problemas de performance, pois os *drivers*[12] das placas têm que ser genéricos já que *drivers* específicos, pela grande variedade de placas, somariam uma quantidade considerada inviável pelo autor. Outra limitação de desempenho é que as instruções do SO convidado, passam por duas translações binárias até ser executada na CPU física já que o este SO roda no anel 1 e o VMM no anel 0. A **Paravirtualização** busca melhorar o desempenho da categoria anterior ao custo de perder uma característica positiva já que o SO convidado é necessariamente modificado de forma que as instruções de CPU que representarem risco a estabilidade são modificadas para buscar o *Hypervisor* e não o CPU diretamente, o restante das instruções tanto do SO quanto da aplicação do usuário podem ser executadas diretamente no CPU físico. Dessa forma o SO convidado é executado no anel 0, embora exista uma camada de virtualização entre esse anel e o CPU.

Os fabricantes de CPU, à partir das dificuldades da paravirtualização tomaram a iniciativa de adicionar recursos nos novos CPUs de 32 bits e modificou bastante a estrutura em relação a clássica x86, ao desenvolver a atual geração de CPUs de 64 bits com recursos valiosos para virtualização: o Intel VT e AMD-V. Surgiu então a 3ª categoria denominada **Virtualização assistida pela máquina (*Hardware*).** Resumidamente esses CPUs permitem a execução do Hypervisor abaixo do anel 0, de forma que a máquina virtual tem a sua disposição os tradicionais anéis 0, 1, 2 e 3 para a aplicação de usuário.

Para essas máquinas temos praticamente só vantagens: o desempenho é muito bom, o SO convidado não precisa ser modificado, e mantemos a segurança pela isolação das máquinas virtuais. O ESXi utiliza essa tecnologia e é por isso que só podemos instala-lo em servidores com processadores 64 bits com as tecnologias de virtualização compatíveis, que na prática já inclui uma grande variedade de máquinas a alguns anos (VERAS, 2011, p.104-109).

Além disso, Veras utiliza também a classificação em *Hypervisor* tipo 1 e tipo 2, como já apresentado e ainda acrescenta que o *Hypervisor* tipo 1 tem dois tipos ou situações: *Hypervisor* monolítico e *Hypervisor microkernelizado*. No primeiro todos os drivers, programas de interface com a máquina, estão no próprio *Hypervisor* que emula as placas para o SO convidado, por outro lado o *Hypervisor microkernelizado* utiliza drivers na máquina virtual eliminando uma camada, como pode se ver na figura abaixo. Dessa forma reduzimos ainda mais o *Hypervisor* aumentando a segurança e o desempenho (VERAS, 2011, p.102).

2.1 *VMWARE* ESXi

O ESXi ou *VMWARE* vSphere Hypervisor, ao contrário do que muitos pensam é uma parte do conjunto de soluções para Data Center da *VMWARE* que é gratuita, e é uma evolução em relação a tecnologia ESX, que foi descontinuada em 2008. Não há uma versão paga para se instalar na máquina hospedeira *(host*), usa-se ou mantém-se a mesma instalação. O que muda são os outros itens do pacote que permitem gerenciar vários servidores ESXi e os recursos mais avançados para Data Center através do *vCenter Server* entre outros produtos. Listamos na figura 6 as principais capacidades da versão 5.1 , de 2012 que é última contemplada na análise de vulnerabilidades.

Figura 6 - Máximos do ESXi 5.1

Recurso	Capacidade Máx.
CPUs / Host	160
Máquinas Virtuais / Host	512
CPUs Virtuais/ Host	2048
Memória RAM / Host	2 TB
Virtual Disks / Host	2048

Fonte: *VMWARE* c (p. 2-4, tradução nossa)

O fabricante *VMWARE* enfatiza em sua divulgação do produto como também em material de treinamento que o ESXi 5.1 tem uma série diferenciais, muito focados na segurança da informação:

- *Uma ocupação mínima de espaço em disco o máximo de 70 Mb indicando um código enxuto, com menos possibilidades de ataques;*

[12] Drivers - programas que gerenciam componentes da máquina, muitas vezes desenvolvidos pelo fabricante para uso em uma linha de sistemas operacionais, por exemplo um driver de placa de vídeo nVidia para Windows 8.

- *Possibilidade de ser instalado e iniciado numa máquina sem disco, utilizando partida por rede (auto deploy), dispositivo USB ou cartão SD;*
- *Memory hardening* – um reforço na segurança da memória através da alocação aleatória, não previsível, de seus componentes como: Kernel, drivers, bibliotecas, aplicações para o administrador, entre outras. Que dificulta ataques que ocorrem em outros Kernel onde o endereço é fixo ou previsível;
- Integridade dos módulos – uma assinatura digital assegura a integridade e a autenticidade dos módulos já mencionados à medida que são carregados pelo *VMKernel;*
- *Trusted Platform Module (TPM)* – é um recurso opcional feito em parceria com fabricantes de máquinas, hoje bem difundido, para a utilização de chaves criptográficas com armazenamento de chave diretamente em chips na máquina que permitem restrições à alterações não autorizadas. Baseado nisso é possível ter ótimas garantias quanto a uma plataforma confiável para se verificar e afirmar que um processo de partida e carga dos *drivers* foi genuína; e,
- *Hypervisor* tipo 1*(bare-metal)*

(*VMWARE* EDUCATION SERVICES, 2012, p. 56-57, tradução nossa)

4 *VMWARE* ESXi E SUAS VULNERABILIDADES

Antes de explanar sobre as vulnerabilidades do *VMWARE* ESXi e estabelecer comparações entre as versões se faz necessário apresentar a fonte dessas informações que é o banco de dados de vulnerabilidades - NVD, e é organizado pelo NIST, sigla para *National Institute of Standards and Technology*.

O NIST é um instituto que faz parte do Departamento do Comércio Norte Americano, com objetivo semelhante ao nosso Instituto Nacional de Metrologia, Qualidade e Tecnologia, o Inmetro. Foi criado em 1901 com o objetivo de desenvolver a indústria e a tecnologia em geral. Sendo um órgão oficial desse porte num país com o porte tecnológico e financeiro como os Estados Unidos, é fácil perceber sua importância e confiabilidade no mercado internacional.

O banco de dados NVD de "*National Vulnerability Database*", é um repositório público de vulnerabilidades em TI, de abrangência mundial e que se auto define como "o repositório oficial de dados padronizados para gerenciamento das vulnerabilidades" (NIST, 2014 About).

Percebemos sua importância por exemplo quando vemos Mauricie Keulen e Virginia Franqueira da Universidade de Twente na Holanda redigir um artigo somente para a análise da importância do banco de dados NVD frente aos ataques à Segurança de Informação. Eles afirmam que o banco de dados NVD é utilizado por especialistas do mundo todo e é o mais abrangente repositório público de vulnerabilidades no meio acadêmico. Sendo utilizado na prática, tanto para consulta como para relatar vulnerabilidades (KEULEN e FRANQUEIRA, 2008, p.2). É interessante saber também que o NIST gerencia as descobertas de falhas feitas por consultorias de segurança no mundo todo, mantendo-as em sigilo até que o fabricante do produto teste e publique uma correção.

As vulnerabilidades listadas no Banco NVD utilizam um índice para classificá-las. Sua terceira versão já está disponível na *Internet* desde junho de 2015 e é denominado **CVSS,** uma sigla para "*Common Vulnerability Scoring System*". Esse índice se propõe a resolver os problemas mais comuns no gerenciamento de riscos de TI, através de **Índices padronizados** sendo que a padronização inclui todos os programas e equipamentos, um f**ramework aberto** para garantir a transparência e Prioridade do risco considerando o ambiente tornando a análise contextual quando necessário.

O índice é composto de vários fatores agrupados em três grupos de métricas: as métricas de base que são compostas de seis fatores, sendo três deles o impacto sobre a tríade da SI: confidencialidade, integridade e disponibilidade São considerados também: o vetor de acesso, a complexidade de acesso e nível de autenticação. O outro grupo é composto pelas métricas temporais: É uma classe opcional que inclui três fatores avaliando o nível de exposição, de solução e de confirmação. As métricas do contexto são também opcionais e dão parâmetros ao profissional de SI. Trazem os parâmetros de nível de efeito colateral sobre os danos econômicos ou humanos que a vulnerabilidade pode causar; nível de abrangência dos alvos e três índices personalizados pelo analista de SI relacionados com a tríade CIA .

Todas vulnerabilidades listadas no banco NVD **possuem uma identificação única e padronizada denominada CVE**. De acordo com a organização sem fins lucrativos Mitre (2015), para se obter uma identificação para uma vulnerabilidade nova, basta um pesquisador procurar um dos fabricantes reconhecidos como CNA ou um centro de respostas a incidentes como o Cert/CC e seguir os procedimentos exigidos. Com essa denominação padrão e única, pode-se divulgar e discutir soluções publicamente com eficiência.

Consultorias de SI testam e buscam vulnerabilidades de produtos lançados no mercado e para comprovar a vulnerabilidade muitas vezes utilizam o *proof of concept* ou PoC: Prova de conceito ou PoC em SI é um procedimento e/ou programa que demonstre claramente a existência de uma vulnerabilidade e

muitas vezes do risco. São feitos de forma que se possa reproduzir a mesma situação por qualquer profissional da área, permitindo assim aos fabricantes e profissionais de SI simularem o problema, estudar e criar uma solução por correção do programa ou solução de contorno, por exemplo através de uma configuração. Nesse estudo chamaremos simplesmente de “provas”.

Por uma questão de ética a consultoria pode notificar primeiro o fabricante e o NIST ou eventualmente outro órgão neutro, já com um CVE definido, aguardando inclusive que o fabricante reconheça formalmente a falha. Depois de desenvolvida e testada a correção, o que pode levar meses, é revelado ao público todas as informações. A partir deste momento também que a vulnerabilidade é inclusa na lista pública do CVE e no banco de dados NVD.

Antes de concluirmos essa introdução é necessário saber que o preparo dessa análise levou a pesquisa nas aplicações Web que dão acesso ao banco de dados NVD e a aplicação do site “CVE Details” foi escolhida em virtude das opções de pesquisa oferecidas e a agilidade do site. Ainda assim nem todas informações necessárias eram apresentadas nos arquivos de texto exportados pela aplicação, por isso foi necessário fazer seis consultas, uma por versão, e registrar os dados manualmente relacionando cada vulnerabilidade com a versão ou versões afetadas, resultando numa planilha. É importante observar também que foram consideradas nessa planilha apenas vulnerabilidades de documentação madura, outras consultas que não o “CVE Details” dão acesso a vulnerabilidades no NVD em estágio de rascunho ou “candidatas” que não foram consideradas, mas que por algum motivo se tornam públicas. Uma parte bastante ilustrativa da planilha se encontra no anexo A.

A figura 7 mostra a totalização do volume de vulnerabilidades por versão e por ano de descoberta, base fundamental para o estudo e foi obtida por consultas no CVE Details.

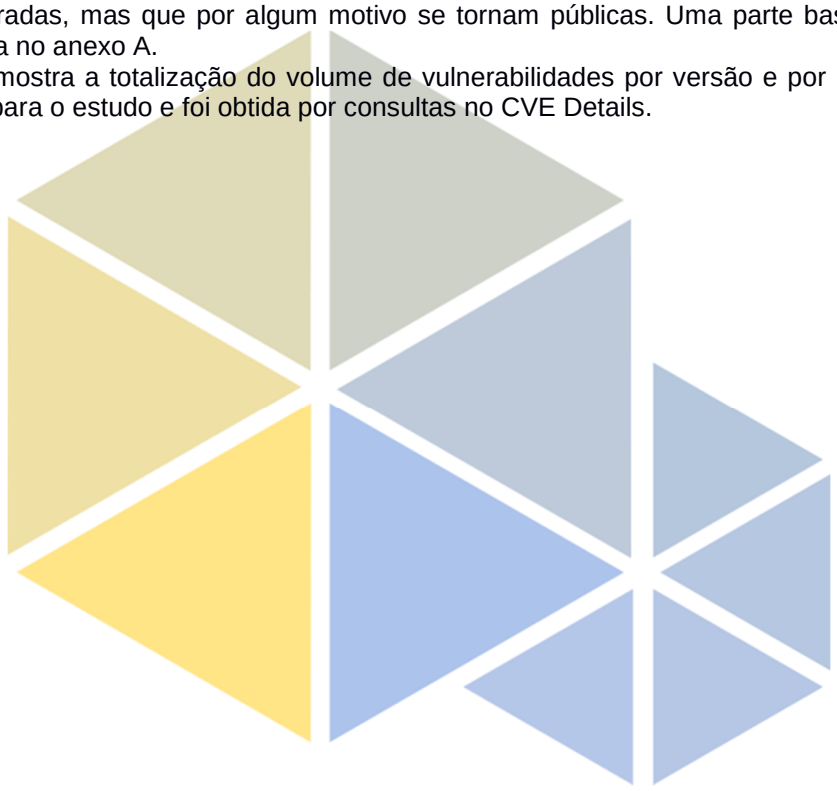


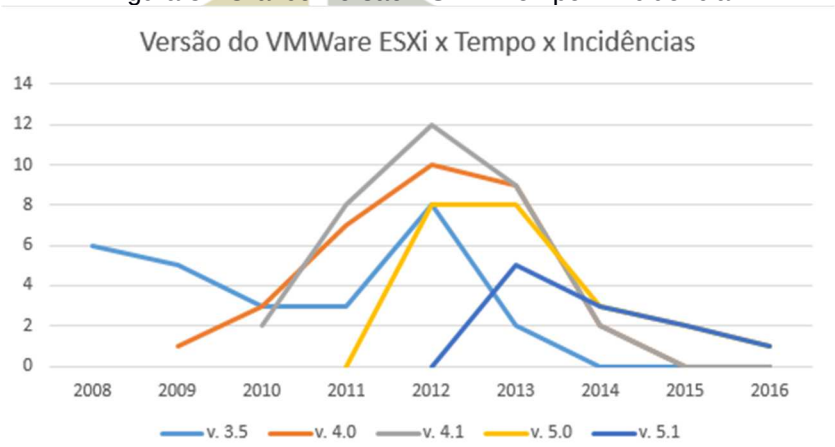
Figura 7 - Gráfico de Vulnerabilidades do ESXi x Ano

		Total de vulnerabilidades / ano									Total
Lançamento	Versão	2008	2009	2010	2011	2012	2013	2014	2015	2016	
Fev. 2008	v. 3.5	6	5	3	3	8	2	0	0	0	27
Mai. 2009	v. 4.0		1	3	7	10	9	2	0	0	32
Jul. 2010	v. 4.1			2	8	12	9	2	0	0	33
Ago. 2011	v. 5.0				0	8	8	3	2	1	22
Set. 2012	v. 5.1					0	5	3	2	1	11
Set. 2013	v. 5.5						1	1	2	1	5

Fonte: Autoria própria

Com base nessas informações foi feito um estudo buscando determinar quais versões já estavam estáveis no sentido de que pouca ou nenhuma vulnerabilidade nova será ainda descoberta em 2015[13]. A figura 8 mostra o gráfico para auxiliar na visualização desses dados, e confirmar em que ano as versões apresentam grande queda de novas vulnerabilidades.

Figura 8 - Gráfico Versão ESXi x Tempo x Incidência



Fonte: autoria própria

Esta análise levou a eliminar apenas a versão 5.5, penúltima versão lançada, pois mostrou que com exceção da primeira versão de 2008 que levou 5 anos para estabilizar, as demais atingiram estabilidade em apenas três anos, o que é coerente considerando o grande aumento no estudo das vulnerabilidades e segurança em TI na década iniciada em 2010.

4.1 A apresentação dos dados

A partir da planilha com todas as vulnerabilidades (anexo A), foram filtradas e contadas as vulnerabilidades totais para cada versão analisada e em seguida nova coleta foi feita filtrando apenas as vulnerabilidades com CVSS maior ou igual a 7, que pelo critério estabelecido são consideradas como graves. Dessa forma se chegou a tabela que está reproduzida na figura 9.

Figura 9 - Tabela Versão de ESXi x Volume de Vulnerabilidades

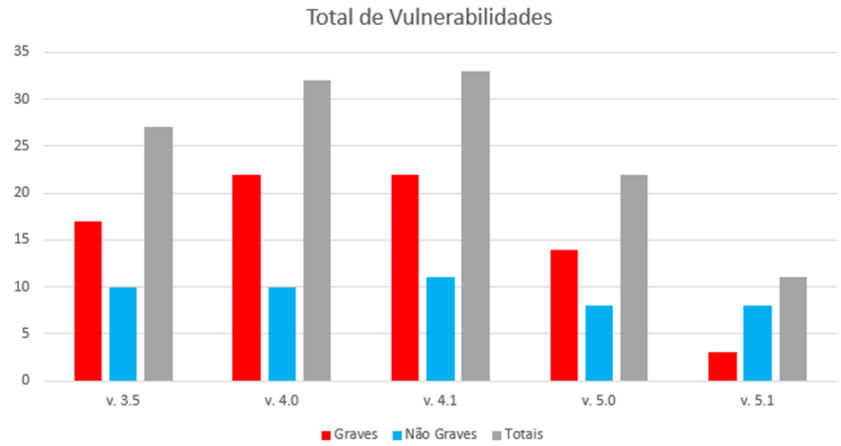
	Vulnerabilidades		
Versão	Graves	Não Graves	Totais
v. 3.5	17	10	27
v. 4.0	22	10	32
v. 4.1	22	11	33
v. 5.0	14	8	22
v. 5.1	3	8	11

[13] Uma revisão feita em março de 2016, utilizando as mesma fontes e critérios demonstrou que as hipóteses se confirmam: Desde a publicação do trabalho como TCC em 2015 surgiu apenas uma vulnerabilidade identificada como CVE-2015-6933, de gravidade média (6.5), afetando todas as versões desde 5.0 a 6.0 com data de publicação em Jan. de 2016. Para as versões anteriores nenhuma vulnerabilidade nova foi descoberta desde 2014.

Fonte: Autoria própria a partir de dados do NVD

Observando a tabela e o respectivo gráfico apresentado na figura 10 percebemos que a tendência de redução de vulnerabilidades em geral como também das mais graves é expressiva. Se compararmos a maior incidência, que ocorre sobre a versão 4.1 contra a versão mais recente estudada, a versão 5.1: temos uma redução de 66% na contagem total e de 86% das vulnerabilidades graves.

Figura 10 - Gráfico comparativo de vulnerabilidades ESXi



Fonte: Autoria própria a partir de dados do NVD

Uma vez que a queda da quantidade de vulnerabilidades se mostra expressiva o estudo prático pode trazer uma nova visão quanto ao problema proposto pelo detalhamento e validação que ele permite e foram essas motivações que levaram a sua execução.

5 ESTUDO PRÁTICO

Após o estudo da evolução das vulnerabilidades do ponto de vista quantitativo, conforme abordado na 4ª seção, houve o cuidado de aprofundar o estudo com um caráter mais prático através da criação de um ambiente real, onde fosse possível a simulação de algumas vulnerabilidades apontadas no capítulo anterior, afim de ratificar a existência destas e experimentar a possível mitigação no decorrer das diferentes versões do *VMWARE*.

A exploração de vulnerabilidades foi abordada através da pesquisa de publicações na Internet de provas ou PoC mencionados antes. Já a comprovação da existência de uma falha foi atingida pelo estudo da vulnerabilidade e o uso de procedimentos e ferramentas de segurança com funções específicas para testar cada vulnerabilidade do produto. Essas ferramentas fornecem relatórios de conformidade às melhores práticas e normas de SI e possuem módulos específicos para cada vulnerabilidade conhecida, apontando assim todas as vulnerabilidades conhecidas de um ambiente de TI. É o "*Nessus Vulnerability Scanner*" da *Tenable Network Security* em plataforma Windows, o popular NMAP gratuito e de código aberto e o Metasploit da empresa Americana Rapid7 ambos em plataforma Linux.

Para esse estudo foi criado um ambiente de testes com servidores ESXi em várias versões diferentes que permitisse submeter servidores diferentes à uma mesma prova ou ferramenta.

Perante a dificuldade de satisfazer os requisitos mínimos de memória desses vários servidores e pela agilidade e versatilidade que se teria em tê-los todos operando ao mesmo tempo, optou-se por usar máquinas virtuais. Para tanto foi utilizada uma prática comum entre profissionais em virtualização de computadores conhecida por *Nested,* de aboletado ou alojado, que será detalhada mais adiante.

5.1 Cenário

Na primeira análise das demandas para a criação do ambiente de testes, viu-se que seria produtivo utilizar a técnica conhecida por ***Nested Hypervisors***.

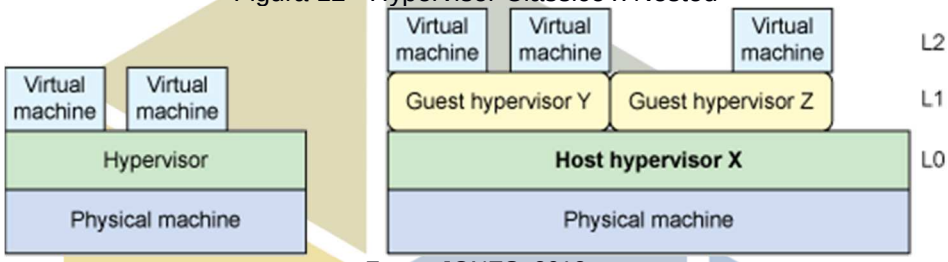
Nested traz a idéia de servidores arranjados como bonecas russas ou como uma cebola, onde um está contido noutro conforme figura 11.

Figura 11 - "Nested", bonecas Russas Matrioska

Fonte: Acervo particular do autor, 2015

Num conceito mais específico, Ben Yehuda (2010, p.1) explica que o *Hypervisor* na configuração clássica é capaz de executar vários sistemas operacionais, cada um na sua respectiva máquina virtual. Utilizando a técnica *Nested*, o *Hypervisor* executa vários *Hypervisor* diferentes, cada um na máquina virtual relacionada. E estes por sua vez executarão sistemas operacionais diversos.

Figura 12 - Hypervisor Clássico x Nested



Fonte: JONES, 2012

Foi então utilizando esta técnica[14] que foram criadas várias máquinas virtuais para compor o ambiente de teste. O "*Hypervisor* base" para os demais ou *Hypervisor host* como indicado na figura 12 foi na verdade o *VMWARE Workstation* instalado em uma máquina com Windows 7 Pro 64 bits, ilustrado pela figura 13.

Figura 13 - Diagrama das máquinas virtuais

Fonte: Adaptado de <http:// http://news.microsoft.com/imageGallery/> e http://blogs.*VMWARE*.com/workstation>

As máquinas virtuais criadas seguem detalhadas na figura 14 abaixo, junto com a descrição das máquinas físicas.

Figura 14 - Reprodução da tabela de descrição de máquinas

Sistema	Tipo	Configuração	Função
Windows 7 Pro 64 bits	Física	CPU AMD com 4 núcleos de 4.2 Ghz, 8Gb de RAM	Hospedeira das máquinas virtuais e das ferramentas Nessus e Wireshark
Windows 7	Física	CPU AMD com 2 núcleos de 2.6	Plataforma para ambiente de

[14] Foi verificado também se haveria alguma contraindicação do fabricante para esse arranjo em específico e a VMware d (2015) confirma num artigo recente que é possível rodar servidores ESXi ou ESX sobre VMware workstation num ambiente de testes ou de estudo.

Pro 64 bits		Ghz, 4 Gb de RAM	linguagem C com Codeblocks
Linux Mint (Debian)	Virtual	CPU com 2 núcleos, 1.2 Gb de RAM	Plataforma gráfica para ambiente de linguagem C com Eclipse
Linux Debian 7 Wheezy	Virtual	CPU com 2 núcleos, 512 Mb de RAM	Plataforma texto para ferramenta NMAP e composição do ambiente de testes
Linux Kali 1.1	Virtual	CPU 3 núcleos, 512 Mb ou 2 Gb de RAM	Plataforma gráfica de ferramentas de segurança
ESX 3.5	Virtual	CPU com 1 núcleo, 1 Gb de RAM	Para comparação com ESXi
ESXi 4.0 u4	Virtual	CPU com 1 núcleo, 2 Gb de RAM	Para teste de vulnerabilidades
ESXi 4.1	Virtual	CPU com 1 núcleo, 2 Gb de RAM	Para teste de vulnerabilidades
ESXi 5.0 u2	Virtual	CPU com 1 núcleo, 4 Gb de RAM	Para teste de vulnerabilidades
ESXi 5.5	Virtual	CPU com 1 núcleo, 4 Gb de RAM	Para teste de vulnerabilidades

Fonte: Autoria própria

Do ponto de vista da rede de computadores, todas as máquinas tanto físicas como virtuais foram conectadas logicamente a um roteador doméstico que atribuiu automaticamente os endereços de rede às máquinas, assim como demais configurações para acesso à Internet. Dessa forma todas máquinas tinham acesso as demais sem restrições ou bloqueios e da mesma forma por estarem na mesma rede. Todas tinham acesso e configuração igual para usar o referido roteador que lhes deu acesso à Internet quando necessário. Na figura 15 apresentamos a relação dos endereços de rede e as respectivas máquinas.

Figura 15 - Lista completa de endereços de rede

Linux Wheezy	192.168.0.18	ESXi 4.0	192.168.0.25
Linux Wheezy2	192.168.0.19	ESXi 4.1	192.168.0.29
Linux Kali	192.168.0.32	ESXi 5.0	192.168.0.19
ESX 3.5	192.168.0.33	ESXi 5.5	192.168.0.12
Windows 7 Pro - hospedeira	192.168.0.10	Windows 7	192.168.0.20

Fonte: autoria própria

5.2 Desenvolvimento dos testes

Devido a natureza desse estudo, havia a necessidade de se obter os instaladores de várias versões do servidor ESXi para compor um ambiente de testes. No entanto o fabricante só disponibiliza ao público a última versão, que na ocasião do estudo é o ESXi 6.0. O que levou a uma pesquisa quase que constante por algum tempo, buscando pacotes de instalação de versões mais antigas de servidores *VMWARE* e que tivessem o mínimo possível de pacotes de correção já pré aplicados.

Dessa pesquisa alguns foram encontrados em sites de reputação como: ESXi 4.1 no site do fabricante Dell e o ESX 3.5 no site de uma Universidade renomada no Chile a Universidade de Talca, mas houveram outras fontes como o site longgeek..com . Por esse motivo foi necessário utilizar um recurso que permite garantir a integridade de arquivos eletrônicos: primeiramente obteve-se no fabricante a assinatura[15] das versões encontradas no formato MD5 e SHA1 para então confrontar com a assinatura gerada pelo programa da Microsoft "File *Checksum Integrity Verifier*" sobre os arquivos encontrados, conforme tabela abaixo. Todos os instaladores localizados se mostraram íntegros e inalterados em relação aos distribuídos pelo fabricante.

Figura 16 - Tabela de validação dos instaladores

Versão	Origem do instalador	Fonte da assinatura	Resultado
ESXi 4.1 build 260247	Longgeek.com	*VMWARE*	Confere
ESXi 4.1 build 260247	Dell.com	Dell	Confere

[15] Assinatura – nesse contexto se refere a uma operação que se aplica sobre um arquivo eletrônico resultando numa "soma" única que identifica aquele arquivo e que se for alterado irá alterar o resultado final, a soma ou "assinatura". Essas operações são padronizadas e MD5 e SHA1 são dois padrões utilizados pelo fabricante, sendo o último mais moderno e preciso.

ESX 3.5 u5	ftp.utalca.cl	*VMWARE*	Confere
ESXi 5.0 u3 build 914586	*VMWARE*	*VMWARE*	Confere
ESXi 5.5 u1	*VMWARE* [16]	*VMWARE*	Confere
ESXi 4.0 u4 build 504850	Redes de compartilhamento	*VMWARE*	Confere

Fonte: Autoria Própria

Com o cenário montado iniciou-se a busca por reproduzir algumas vulnerabilidades, o ponto inicial escolhido foi utilizar uma planilha com todas as vulnerabilidades publicadas pelo NIST para os servidores *VMWARE*, tendo sido tabulada pelo autor para permitir consultas das vulnerabilidades por versão de ESXi afetada. Como ela também continha informações classificando as vulnerabilidades, foi dado um foco nas que possuíam as seguintes características: Possuíam um "Score" considerado grave, identificado pela nota acima de 7, permitiam uma negação de serviço e também classificadas com uma complexidade média ou baixa de execução. Pois além de serem graves para o usuário possivelmente seriam mais rápidas de se reproduzir nos testes. Na figura 17 e no anexo A temos reprodução de um trecho do conteúdo[17] dessa planilha dividido em duas partes.

Figura 17 - Planilha de vulnerabilidades

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	ESXi Versions Affected							CVE ID	CWE ID	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication
		3.5	4.0	4.1	5.0	5.1	5.5										
22		X	X	X				CVE-2013-1405	287	DoS Exec Code Mem. C	02/2013	02/2013	10.0	None	Remote	Low	Not required
23					X	X	X	CVE-2014-3793		DoS +Priv	05/2014	06/2014	5.8	User	Local Ne	Low	Not required
24			X	X	X	X		CVE-2014-1208		DoS	01/2014	01/2014	3.3	None	Local Ne	Low	Not required
25			X	X	X	X		CVE-2014-1207		DoS	01/2014	01/2014	4.3	None	Remote	Medium	Not required
26					X	X	X	CVE-2015-1044		DoS	01/2015	02/2015	3.3	None	Local Ne	Low	Not required
27					X	X	X	CVE-2014-8370	264	DoS +Priv	01/2015	02/2015	6.4	None	Remote	Low	Not required

	H	R	S	T	U
	CVE ID	Confidentiality Impact	Integrity Impact	Availability Impact	
CVE-2013-1405		Complete	Complete	Complete	VMware vCenter Server 4.0 before Update 4b and 4.1 before Update 3a, VMware VirtualCenter 2.5, VMw
CVE-2014-3793		Partial	Partial	Partial	VMware Tools in VMware Workstation 10.x before 10.0.2, VMware Player 6.x before 6.0.2, VMware Fus
CVE-2014-1208		None	None	Partial	VMware Workstation 9.x before 9.0.1, VMware Player 5.x before 5.0.1, VMware Fusion 5.x before 5.0.1,
CVE-2014-1207		None	None	Partial	VMware ESXi 4.0 through 5.1 and ESX 4.0 and 4.1 allow remote attackers to cause a denial of service (N
CVE-2015-1044		None	None	Partial	vmware-authd (aka the Authorization process) in VMware Workstation 10.x before 10.0.5, VMware Pla
CVE-2014-8370		None	Partial	Partial	VMware Workstation 10.x before 10.0.5, VMware Player 6.x before 6.0.5, VMware Fusion 6.x before 6.0

Fonte: Visualização dos dados fornecidos no banco de dados NVD, no site CVE Details

Estudo da vulnerabilidade "ESXi vSphere API DoS"

A primeira vulnerabilidade escolhida foi a **CVE 2012-5703** pois satisfazia os critérios já mencionados incluindo a localização na *Internet* de uma prova para ela. A prova foi desenvolvida por Sebastian Tello e publicada pela consultoria "Core Security" (2012) em conjunto com a *VMWARE*, e é um código em linguagem Python 2 sem documentação. Segundo o boletim "*VMWARE* Advisory 2012-0016" (*VMWARE* e, 2013) a vulnerabilidade afeta apenas a versão 4.1 do ESXi. O referido boletim reconhece que existe uma falha na biblioteca de programas ou API de gerenciamento do ESXi, que permite um ataque cause uma negação desse serviço. A parada do serviço não afeta as máquinas virtuais diretamente, mas cessa a comunicação que permite qualquer tipo de gerenciamento, desde um simples console remoto até soluções centralizadas. Foi necessário dedicar algum tempo em testes, utilizando a máquina de testes Wheezy, buscando entender o funcionamento do código, já que ele permitia algumas configurações via parâmetros. Pela prática e pela pesquisa ficou comprovado que o servidor ESXi 4.1 em testes não era vulnerável mesmo sendo da versão falha e possuir o "build" registrado como o primeiro para a versão 4.1, o 260247.

Figura 18 - Reprodução da execução da prova em linguagem Python, vSphereDoS

[16] Os instaladores do ESXi 5.0 e ESXi 5.5 tiveram como origem a biblioteca do Autor, que os obteve diretamente do fabricante na época em que era a versão mais atual.

[17] A planilha mantém as colunas conforme o banco de dados NVD, e foi pela coluna "Vulnerability Types" (Tipos de vulnerabilidade) que se filtrou as que continham "DoS", de negação (ou parada) de serviço em inglês.

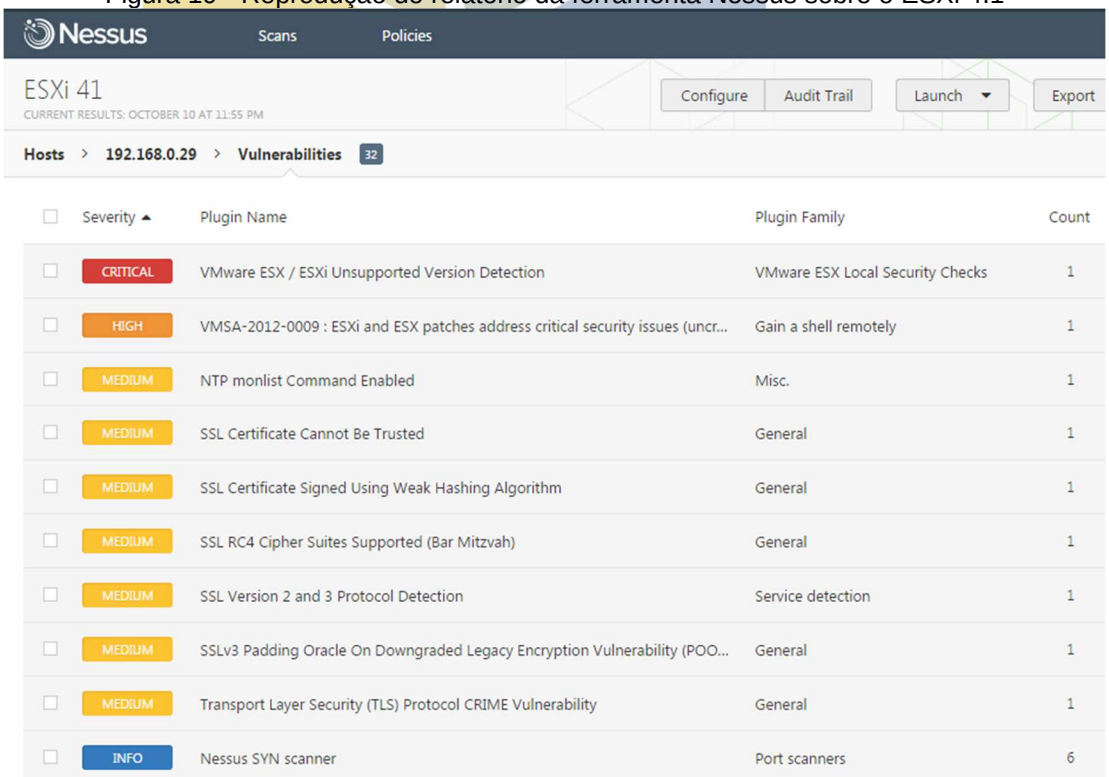
```
root@WheezyServer:~# python vSphereDoS.py
Usage: vSphereDoS.py [-p PORT] [-l] [-n] esx.example.com

vSphereDoS.py: error: No server host provided
root@WheezyServer:~# python vSphereDoS.py 192.168.0.29
 * Host seems to be up
 * Attacking 192.168.0.29:443 ...
 * Attack failed
```

Fonte: autoria própria

Na figura 18 pode-se ver a tela executando o programa prova sem parâmetros onde ele espera a indicação do servidor seja por nome ou endereço de rede. Em seguida o comando foi repetido apontando para o endereço do servidor ESXi 4.1, logo o programa confirma que recebeu uma resposta do servidor com a mensagem “Máquina parece estar disponível”, depois fica alguns segundos na mensagem indicando que está “Atacando” o servidor indicado na porta de rede padrão 443 para finalmente indicar que o ataque falhou. Um exame simples do programa indica que as outras mensagens possíveis seriam “Prova executada, mas não foi possível verificar se o servidor travou” ou ainda “Ataque bem sucedido: servidor travou”.

Durante essa pesquisa, os sites de segurança faziam referência a testes específicos para essa vulnerabilidade mencionando a ferramenta Nessus e um identificador “Nessus Id 62944”. Ao saber que existiam testes específicos para cada vulnerabilidade do *VMWARE*, ESXi, introduziu-se na rotina de testes verificações via rede utilizando essa e outras ferramentas. Para a análise dessa vulnerabilidade foram gerados relatórios[18], um verificando todos os servidores em teste. Outra busca foi executada apenas sobre o ESXi 4.1 e a parte do relatório pode ser visto na figura 20 onde se nota que não há nenhuma referência a essa vulnerabilidade na coluna “Plugin Name”. O primeiro relatório que segue anexo confirmou a não existência desta vulnerabilidade em nenhum dos servidores: ESX 3.5, ESXi 4.0, ESXi 4.1, ESXi 5.0 e ESXi 5.5, certificando que não seria possível reproduzir essa vulnerabilidade.

Figura 19 - Reprodução do relatório da ferramenta Nessus sobre o ESXi 4.1



Fonte: gerado pela ferramenta Nessus – Tenable Network Security

Estudo da vulnerabilidade Shellshock nos *VMWARE* ESXi

Em seguida foi estudada a falha no Linux conhecida como “Shellshock”, “Bashdoor” ou “Bash bug”, documentada sob Id **CVE 2014-6271** que embora não estivesse relacionada diretamente a qualquer versão

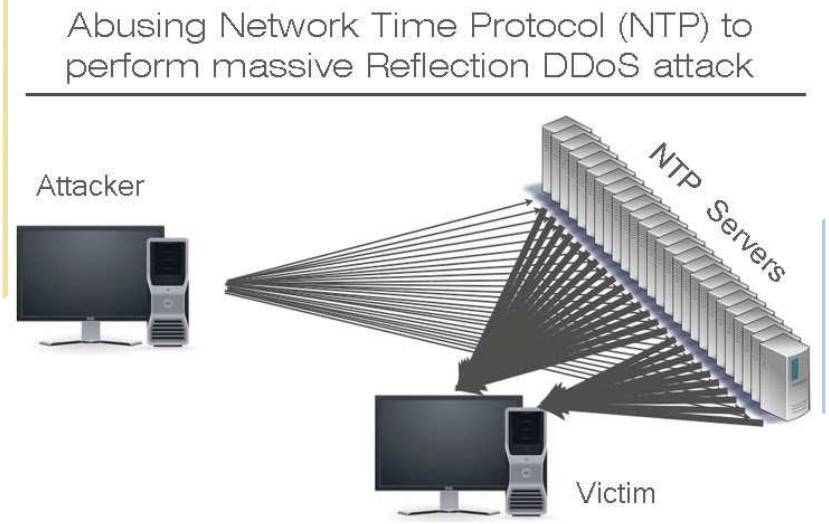
[18] Para obter uma cópia eletrônica dos relatórios da ferramenta Nessus favor acessar os endereços indicados no ANEXO B.

de *VMWARE* ESXi na lista do CVE, mas teve grande repercussão nas comunidades de SI por afetar um recurso ou módulo frequentemente utilizado em soluções que utilizam Linux, o Bash[19]. Uma pesquisa indicou que a vulnerabilidade afetaria apenas o ESX 4.0 e ESX 4.1 e mais uma lista com mais de 70 itens com várias versões de vários produtos *VMWARE*, mas não afetando nenhum dos servidores ESXi que são o escopo desse estudo. E realmente a ferramenta Nessus não encontrou a vulnerabilidade nos servidores ESXi nem mesmo no ESX 3.5 do ambiente de testes. Quando os servidores foram examinados em detalhe foi confirmado que os mesmos utilizam o Bash "Busybox" que não é afetado. Coincide também com a informação da *VMWARE* no boletim *VMWARE* Advisory VMSA 2014-0010.13 que este é o Bash utilizado atualmente em todas as versões de *VMWARE* Server ESXi do 3.5 ao 6.0.

Estudo da vulnerabilidade NTP Amplification DDoS

Prosseguindo o estudo, foi escolhida uma nova abordagem: à partir do relatório da Nessus ordenado por vulnerabilidade pesquisou-se a existência de uma respectiva prova publicada na Internet. Dessa forma chegou-se a uma prova na forma de um código fonte em linguagem C, na verdade uma prova parcial pois uma explicação no próprio código já avisa que é necessário modificá-lo de forma a repetir a consulta muitas vezes, à partir de várias máquinas até se atingir a parada do servidor. A vulnerabilidade tem como id **CVE 2013-5211** e o boletim *VMWARE* VMSA-2014-0002.4, reconhece a vulnerabilidade, mas não menciona a prova encontrada. A vulnerabilidade foi causada por um sistema muito utilizado em diversos Linux para implementar funções de sincronismo de relógio através de troca de informações via rede, conhecido por NTP Server. Devido a grande utilidade desses serviços raramente se bloqueia essas comunicações até porque normalmente são pacotes pequenos. Mas a falha permitia que servidores NTP conectados à Internet fossem induzidos a responder à uma consulta conhecida por "Monlist" para um endereço forjado, sem nem identificar o requisitante real. Esse mesmo requisitante oculto poderia induzir uma lista de servidores vulneráveis apontando para um único endereço, alvo do ataque, até que ele travasse caracterizando um ataque distribuído por negação de serviço ou DDoS.

Figura 20 - Ilustração de ataque DDoS usando NTP Amplificado



Fonte: Revista Exame <http://exame.abril.com.br/tecnologia/noticias/maior-ataque-ddos-da-historia-atinge-servidores-da-cloudflare>

A vulnerabilidade teve grande impacto na época de sua descoberta e exploração, pois permitiu um ataque a um grande prestador de serviços de Internet, o Cloudfare. O mesmo afirma manter hoje 2 milhões de páginas de *Interne*t entre elas: o instituto MIT, a banda Metallica, o fabricante Cisco, entre outros. Em 10 de fevereiro de 2014 o CEO da empresa reconheceu a existência de um ataque, que depois foi reconhecido com o maior ataque em volume até então, pois consumiu uma de banda de 400 Gb/s conforme detalhes da Revista Exame (2014).

Ao longo do estudo dessa falha a maioria das máquinas do ambiente de testes foi envolvida e devido à natureza técnica das ferramentas e do ambiente de rede montado, serão indicadas nas ilustrações pelo seu endereço na rede. Por esse motivo se reproduz novamente na figura 21 uma tabela relacionando as máquinas e seus endereços.

[19] Bash ou shell: o interpretador de comandos de sistema operacionais derivados do Unix. É através dele que o operador pode interagir mais diretamente com um sistema operacional. Nas distribuições Linux são conhecidos por bash na verdade vários módulos de função semelhante mas que foram desenvolvidos por equipes diferentes, como o Busybox (na verdade Ash), toybox, ...

Figura 21 - Tabela Máquinas x Endereço de rede

Linux Wheezy	192.168.0.18	ESXi 4.0	192.168.0.25
Linux Wheezy2	192.168.0.19	ESXi 4.1	192.168.0.29
Linux Kali	192.168.0.32	ESXi 5.0	192.168.0.19
ESX 3.5	192.168.0.33	ESXi 5.5	192.168.0.12

Fonte: Autoria própria

O estudo foi iniciado buscando ter mais evidências de que haveria um servidor ESXi suscetível. Por esse motivo foi verificado diretamente no console dos servidores se utilizavam a versão do NTP Server vulnerável através de um comando “ntpd –version” oque se confirmou pois retornou “ntpd - NTP daemon program - Ver. 4.2.4p6” tanto no ESXi 4.1, ESXi 5.0 quanto no ESXi 5.5. Para dar continuidade foi instalado e configurado nas duas máquinas Linux Debian do ambiente, o NTP Server. É um pacote que traz consigo uma ferramenta que permite estabelecer algumas conexões simulando tráfego e fazer consultas entre servidores NTP, inclusive a requisição “*monlist*” que se não for desativada ou restringida, dá origem a falha. O comando permite mostrar as máquinas e respectivos detalhes das últimas conexões feitas pelo servidor NTP, no limite de seiscentas máquinas.

Essa consulta *monlist* provavelmente foi criada para uso apenas em redes internas, mas em servidores NTP na Internet facilmente a lista pode atingir centenas oque geraria uma resposta à consulta bem maior do que os pacotes de dados mais comuns para NTP. Na figura abaixo temos um exemplo dessas consultas aplicada sobre os servidores ESXi 4.1, 4.0, 5.5 e ESX 3.5. É importante observar que apenas o ESXi 4.1 responde com uma lista.

Figura 22 - Testes com NTP Server e monlist

```
root@WheezyServer:~# ntpdc -n -c monlist 192.168.0.29
remote address          port local address      count m ver rstr avgint  lstint
===============================================================================
192.168.0.18           47535 192.168.0.29          28 7 2    580    106       0
192.168.0.10           65120 192.168.0.29           6 7 2    580      1    2333
root@WheezyServer:~# ntpdc -n -c monlist 192.168.0.25
ntpdc: read: Connection refused
root@WheezyServer:~# ntpdc -n -c monlist 192.168.0.12
192.168.0.12: timed out, nothing received
***Request timed out
root@WheezyServer:~# ntpdc -n -c monlist 192.168.0.33
192.168.0.33: timed out, nothing received
***Request timed out
```

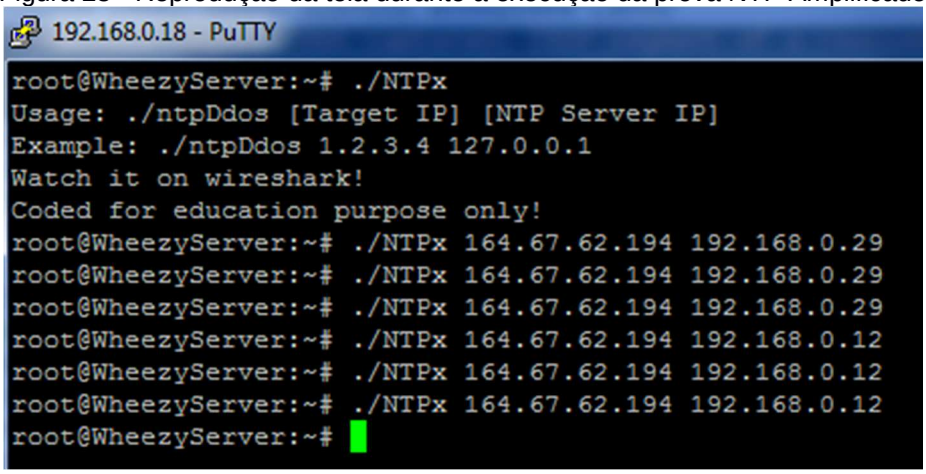
Fonte: autoria própria

Na sequência se iniciou os preparativos para montar o ambiente que pudesse executar a prova em si e logo de início houve uma razoável dificuldade de compilar a fonte da prova, utilizando um ambiente de programação com o *Codeblocks* sobre Windows. Mais adiante foi identificado que o código foi escrito usando bibliotecas de rede do ambiente Linux. Para entender minimamente o funcionamento do código e validar o cumprimento das necessidades do ambiente de programação foi montado um novo ambiente de programação utilizando um Linux gráfico com a interface Eclipse e seguido um tutorial[20] do Prof. Jamin (2015). Os programas de teste do professor foram compilados sem erros e estabeleceram comunicação do tipo servidor e cliente via rede como previsto. Validando então o ambiente e as bibliotecas de rede.

Em seguida, o código fonte da prova foi compilado dessa vez com sucesso e o executável copiado e executado na máquina Wheezy induzindo os servidores *VMWARE* à responder para um servidor NTP fora do ambiente, na Internet. Em simultâneo a ferramenta Wireshark monitorou e registrou as comunicações que ocorriam.

[20] O tutorial faz parte do material da matéria de Redes de Computadores na Universidade de Michigan, ministrada pelo Professor

Figura 23 - Reprodução da tela durante a execução da prova NTP Amplificado



Fonte: Autoria própria

A figura 23 mostra o console do Linux durante a execução apontando uma possível vítima no endereço 164.67.62.194 ou tick.ucla.edu por pacotes de dados enviados pelo ESXi 4.1 e 5.5 por três vezes cada

.

Figura 24 - Monitoração dos dados do ESX 4.1

No.	Time	Source	Destination	Protocol	Length	Info
15	1.85242400	192.168.0.18	192.168.0.10	SSH	118	Server: Encrypted packet (len=64)
16	1.85468500	192.168.0.29	164.67.62.194	NTP	194	NTP Version 2, private
17	1.85526600	192.168.0.18	192.168.0.10	SSH	134	Server: Encrypted packet (len=80)
18	1.85530600	192.168.0.10	192.168.0.18	TCP	54	53276→22 [ACK] Seq=65 Ack=145 Win=16229 Len=0
19	1.91598000	192.168.0.10	192.168.0.255	NBNS	92	Name query NB WPAD<00>
20	2.25193200	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
21	2.25270000	192.168.0.18	192.168.0.10	SSH	150	Server: Encrypted packet (len=96)
22	2.29909800	52.1.1.135	192.168.0.10	DIS	62	PDUType: Unknown
23	2.36498700	fe80::3464:4cdb:b19	ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
24	2.45284200	192.168.0.10	192.168.0.18	TCP	54	53276→22 [ACK] Seq=129 Ack=241 Win=16205 Len=0
25	2.61373500	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
26	2.61513500	192.168.0.18	192.168.0.10	SSH	118	Server: Encrypted packet (len=64)
27	2.61758500	192.168.0.29	164.67.62.194	NTP	194	NTP Version 2, private
28	2.61814900	192.168.0.18	192.168.0.10	SSH	134	Server: Encrypted packet (len=80)
29	2.61818900	192.168.0.10	192.168.0.18	TCP	54	53276→22 [ACK] Seq=193 Ack=385 Win=16169 Len=0
30	2.66596200	192.168.0.10	192.168.0.255	NBNS	92	Name query NB WPAD<00>
31	3.01008400	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
32	3.01191400	192.168.0.18	192.168.0.10	SSH	150	Server: Encrypted packet (len=96)
33	3.01295100	192.168.0.20	192.168.0.255	NBNS	92	Name query NB WLAD-PC<1c>
34	3.20982300	192.168.0.10	192.168.0.18	TCP	54	53276→22 [ACK] Seq=257 Ack=481 Win=16145 Len=0
35	3.32563800	192.168.0.10	192.168.0.20	DB-LSP	144	Dropbox LAN sync Protocol
36	3.32974700	192.168.0.20	192.168.0.10	DB-LSP	128	Dropbox LAN sync Protocol
37	3.32992800	192.168.0.20	192.168.0.10	DB-LSP	128	Dropbox LAN sync Protocol
38	3.32996100	192.168.0.10	192.168.0.20	TCP	54	53182→17500 [ACK] Seq=91 Ack=149 Win=16314 Len=0
39	3.35674300	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
40	3.35844900	192.168.0.18	192.168.0.10	SSH	118	Server: Encrypted packet (len=64)
41	3.36118100	192.168.0.29	164.67.62.194	NTP	194	NTP Version 2, private

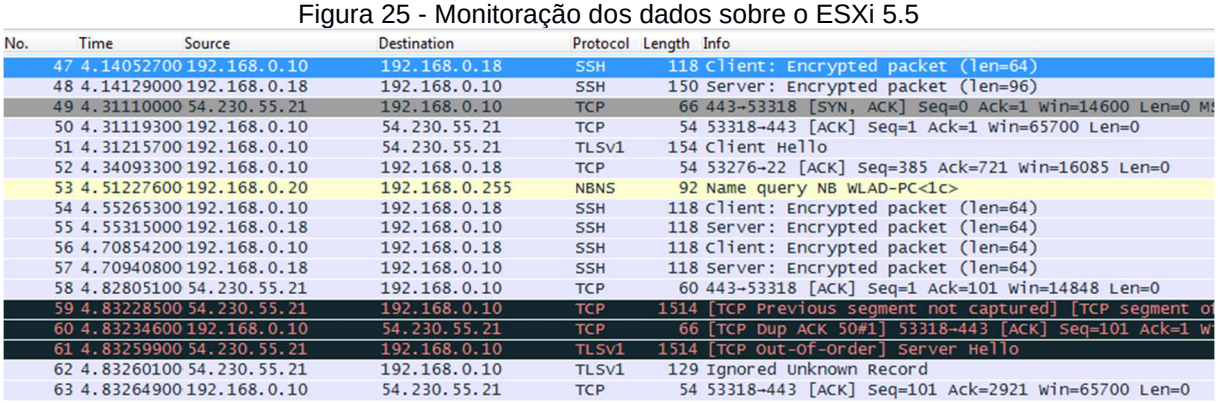
Fonte: Autoria própria

Na figura 24 vemos a primeira tela do *Wireshark* onde as colunas indicam *Source* para Origem, *Destination* para destino e *Protocol* para indicar o protocolo. Apenas observando o protocolo nos permitirá identificar os pacotes trocados para passar os comandos, que são os SSH. São vistos pacotes NTP se o servidor for vulnerável, pois atenderá ao programa prova enviando um pacote de protocolo NTP com destino ao endereço determinado para cada comando.

Ainda sobre a mesma figura se ressalta que por três vezes se observa a sequência esperada: dois pacotes SSH indo e voltando do Wheezy (linhas 25 e 26) e finalmente na linha 27 um pacote NTP com origem no servidor ESXi e destino a máquina fora do ambiente de testes. As outras duas ocorrências são vistas na linha 16 e na 41. É uma confirmação de que este servidor ESX 4.1 é suscetível a ser usado num ataque DDoS, ou de ser uma vítima noutra configuração mais exposta à *Internet*.

Na mesma sessão foram executados por mais três vezes o programa de prova, dessa vez para provocar pacotes a partir do ESXi 5.5. Pode-se notar na figura 25 que os comandos foram passados nos momentos indicados pelas linhas 47, 54 e 56, mas nenhuma movimentação de pacotes com protocolo NTP foi notada, indicando que o servidor ESXi 5.5 não atendeu à solicitação e que portanto, na configuração padrão não é vulnerável.

Figura 25 - Monitoração dos dados sobre o ESXi 5.5



No.	Time	Source	Destination	Protocol	Length	Info
47	4.14052700	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
48	4.14129000	192.168.0.18	192.168.0.10	SSH	150	Server: Encrypted packet (len=96)
49	4.31110000	54.230.55.21	192.168.0.10	TCP	66	443→53318 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 M
50	4.31119300	192.168.0.10	54.230.55.21	TCP	54	53318→443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
51	4.31215700	192.168.0.10	54.230.55.21	TLSv1	154	Client Hello
52	4.34093300	192.168.0.10	192.168.0.18	TCP	54	53276→22 [ACK] Seq=385 Ack=721 Win=16085 Len=0
53	4.51227600	192.168.0.20	192.168.0.255	NBNS	92	Name query NB WLAD-PC<1c>
54	4.55265300	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
55	4.55315000	192.168.0.18	192.168.0.10	SSH	118	Server: Encrypted packet (len=64)
56	4.70854200	192.168.0.10	192.168.0.18	SSH	118	Client: Encrypted packet (len=64)
57	4.70940800	192.168.0.18	192.168.0.10	SSH	118	Server: Encrypted packet (len=64)
58	4.82805100	54.230.55.21	192.168.0.10	TCP	60	443→53318 [ACK] Seq=1 Ack=101 Win=14848 Len=0
59	4.83228500	54.230.55.21	192.168.0.10	TCP	1514	[TCP Previous segment not captured] [TCP segment o
60	4.83234600	192.168.0.10	54.230.55.21	TCP	66	[TCP Dup ACK 50#1] 53318→443 [ACK] Seq=101 Ack=1 W
61	4.83259900	54.230.55.21	192.168.0.10	TLSv1	1514	[TCP Out-Of-Order] Server Hello
62	4.83260100	54.230.55.21	192.168.0.10	TLSv1	129	Ignored Unknown Record
63	4.83264900	192.168.0.10	54.230.55.21	TCP	54	53318→443 [ACK] Seq=101 Ack=2921 Win=65700 Len=0

Fonte: Autoria Própria

Para uma última verificação, foi utilizada uma máquina com Linux Kali que já possui pré-instalado os pacotes de segurança NMap e Metasploit. Um teste utilizando o NMAP sobre os servidores nas versões 5.5 (identificado pelo endereço 192.168.0.12) e 4.1 (endereço 192.168.0.29) comprovou os resultados práticos conforme a figura 26 indica.

Figura 26- Teste NMap para NTP Amplificado ou NTP Monlist

```
Starting Nmap 6.00 ( http://nmap.org ) at 2015-10-18 22:37 BRST
Nmap scan report for 192.168.0.12
Host is up (0.00029s latency).
PORT    STATE         SERVICE
123/udp open|filtered ntp
MAC Address: 00:0C:29:09:64:0A (VMware)

Nmap done: 1 IP address (1 host up) scanned in 5.31 seconds
root@WheezyServer:~# nmap -sU -pU:123 -Pn -n --script=ntp-monlist 192.168.0.29

Starting Nmap 6.00 ( http://nmap.org ) at 2015-10-18 22:37 BRST
Nmap scan report for 192.168.0.29
Host is up (0.00031s latency).
PORT    STATE SERVICE
123/udp open  ntp
| ntp-monlist:
|   Other Associations (3)
|       192.168.0.18 (You?) seen 4 times. last tx was unicast v2 mode 7
|       192.168.0.32 seen 6 times. last tx was unicast v2 mode 7
|_      192.168.0.12 seen 1 time. last tx was unicast v2 mode 7
MAC Address: 00:0C:29:86:F1:54 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
root@WheezyServer:~#
```

Fonte: autoria própria

No primeiro comando vemos que o servidor responde, dizemos que a porta de rede 123 "está aberta", mas na sequência não atende ao comando *monlist*: não retorna com um pacote de dados. Situação diferente do segundo comando sobre o ESXi 4.1 que retorna uma lista com três endereços identificando servidores com quem trocou pacotes NTP. indicando assim como vulnerável. A ferramenta Metasploit através do módulo "auxiliary/scanner/ntp/ntp_monlist", apresentou resultados semelhantes, assim como o relatório da ferramenta Nessus .

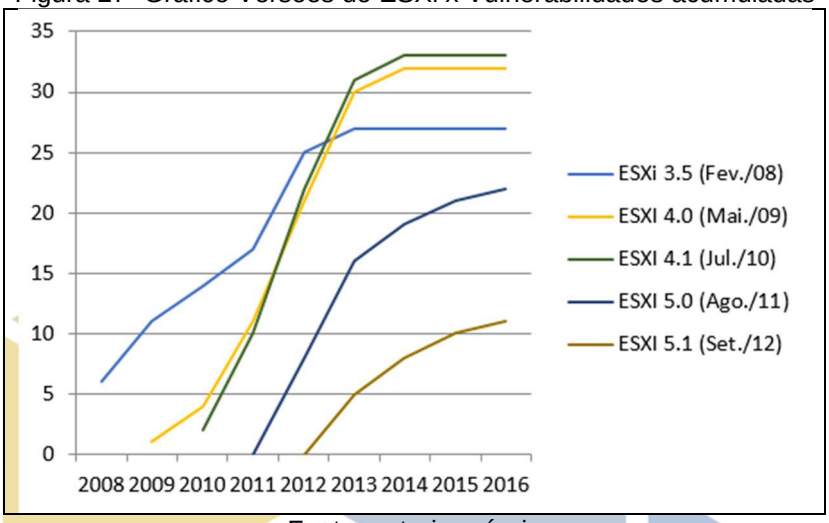
É necessário destacar que além do boletim de segurança da *VMWARE* acima mencionado existe um artigo da *VMWARE* identificado por KB2070193 com o objetivo de orientar como minimizar o risco até que se possa agendar uma parada para executar as atualizações recomendadas. Esse artigo afirma que todos o ESXi da versão 4.0 até 5.5 tem o serviço NTP afetado, mas tem que ser manualmente habilitado. Isso explica porque nos testes não foi detectado a vulnerabilidade, uma vez que todos servidores estavam na sua configuração padrão, recém instalado. O inesperado foi que a versão ESXi 4.1 testada tenha tido, novamente, um comportamento diferente dos demais. Esse tipo de imprecisão foi observado no estudo das outras vulnerabilidades, valorizando o estudo prático em laboratório por trazer essa experiência. Na conclusão à seguir esta experiência será descrita mais longamente.

6. CONSIDERAÇÕES FINAIS

Entendeu-se que para fundamentar bem as considerações é necessário, antes de tudo, apresentarmos em detalhes o resultado ou fatos que cada parte do trabalho prático nos trouxe.

Na seção 5 onde estudou-se as vulnerabilidades de uma forma mais abrangente e quantitativa, entendemos que os critérios adotados foram conservadores ao analisarmos apenas as versões com pelo menos três anos de uso[21]. Dessa forma a última versão analisada é o ESXi 5.1 lançado em Set. de 2012. Se observarmos o gráfico das vulnerabilidades acumuladas na figura 27, visualmente já se nota que, diferente do que esperava, o interesse por SI não levou as vulnerabilidades serem descobertas mais rápido. As linhas dos ESXi 4.0 e 4.1 crescem na mesma velocidade, com a mesma inclinação antes de estabilizar e as versões mais novas crescem um pouco mais devagar

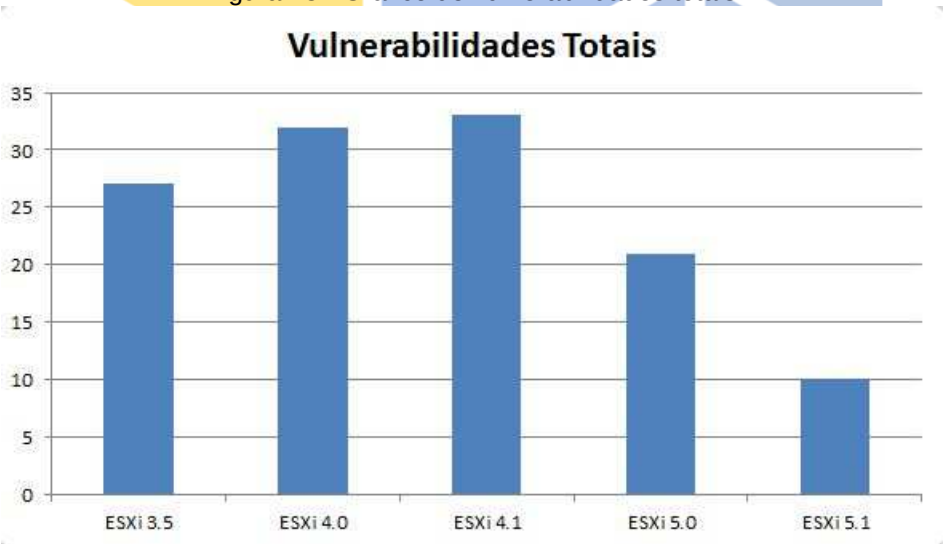
Figura 27- Gráfico Versões do ESXi x Vulnerabilidades acumuladas



Fonte: autoria própria

A diferença na verdade é que as versões mais novas apresentam um volume menor de vulnerabilidades, levando a confecção do gráfico mostrado na figura 28 para ilustrar isso. Essa análise então aponta para a hipótese de que estas são mais seguras que suas antecessoras.

Figura 28 - Gráfico de vulnerabilidades totais



Fonte: Autoria própria

[21] Como resultado da revisão feita em Mar. de 2016 os gráficos ilustrados nas Figuras 27 e 28 foram atualizados trazendo bem pouca alteração já que 2015 terminou sem novas vulnerabilidades descobertas e em 2016 apenas uma nova vulnerabilidade foi descoberta.

A primeira vulnerabilidade testada na seção 5.2 Desenvolvimento dos testes, foi a “ESXi vSphere API DoS” aonde o boletim da *VMWARE* admite uma falha na biblioteca (API) de gerenciamento do ESXi 4.1. Foi considerado que:

- A prova foi executada conforme publicada sem mensagem erros de execução e indicou que a tentativa se resultava em falha, resultado esse reafirmando pela ferramenta de verificação de vulnerabilidades Nessus; e,
- O instalador do servidor ESXi 4.1 estava integro e inalterado em relação ao que o fabricante fornecia e ainda que era exatamente o instalador referido no boletim, através da identificação de “build”.

Chegou-se, nesse caso, a um resultado inconclusivo uma vez que não se reproduziu a falha tampouco se comprovou a existência da mesma.

Novos testes foram feitos a partir da pesquisa da vulnerabilidade Shellshock, que afetou uma grande quantidade de produtos da *VMWARE* e outros fabricantes, mas de acordo com a documentação não afetaria nenhum ESXi, de qualquer versão. E de fato verificou-se que não havia a vulnerabilidade, pois o módulo defeituoso só foi utilizado na linha anterior ESX, que está fora do escopo desse estudo. Temos então um resultado inconclusivo, visto que todas versões do ESXi já são seguras em relação à essa falha

A última vulnerabilidade estudada foi a falha no servidor NTP, aonde já havia vários indícios da existência de falhas mas ainda alguma dúvida pois foi encontrado um documento afirmando que o serviço NTP estaria desativado na configuração padrão de todas as versões de ESXi estudadas. Mas oque se verificou foi o contrário:

- Três versões de ESXi utilizavam o módulo NTP falho;
- O servidor ESXi 4.1 respondeu à consultas NTP, indicando atividade;
- Ferramentas de segurança apontaram apenas o referido servidor como não conforme ou falho à respeito dessa vulnerabilidade em específico; e,
- A prova encontrada na Internet afetou o servidor ESXi 4.1 e não o servidor mais novo ESXi 5.5.

Desta forma, considerando essa vulnerabilidade se validou que pelo menos uma versão mais nova tem sua configuração padrão mais segura que a anterior, indicando um amadurecimento do produto nesse aspecto.

Ampliando as análises e considerando então todos os testes feitos, ficou comprovado que o produto *VMWARE* Server ESXi tem se demonstrado mais seguro, especificamente da versão 4.1 em diante. Diz -se isso porque o ESXi 4.1 tem atualmente a maior quantidade de vulnerabilidades graves e totais da sua linha e foi o único servidor que apresentou falha comprovada nos testes de laboratório. Durante esses testes foi observado, mesmo sem uma análise detalhada, que o fabricante passou a ter mais prudência nas configurações padrão, deixando menos cuidados à cargo do usuário, como por exemplo obrigando a se escolher uma senha segura para o usuário administrador em contraste a senha em branco permitida nas versões anteriores a 5.0.

Aproveitamos para sugerir uma continuidade desse estudo, ao adotar um escopo mais abrangente que inclua produtos de outros fabricantes e/ ou mais versões do servidor *VMWARE* que não puderam ser analisadas e comparadas nesse estudo por falta de um laboratório que comportasse mais servidores e pelas limitações de tempo do autor. Seria interessante incluir a versão lançada esse ano o ESXi 6.0, mas que exigiria mais memória virtual dos que as 12 Gb dedicados às quatro versões estudadas.

O estudo como um todo trouxe a oportunidade ao autor de vivenciar parte da rotina de gerenciamento de vulnerabilidades pelo uso de ferramentas de segurança e análise e estudo de boletins de consultorias e de fabricantes, bem como perceber a importância de validar a documentação fornecida, pois várias vezes detalhes apresentados nessas documentações não se comprovou na prática, tanto a favor como contra a segurança do produto.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). **ABNT ISO 17799:** Norma ABNT NBR ISO/IEC 17799:2005 – Código de prática para a Gestão da Segurança da Informação, Rio de Janeiro: ABNT, 2005.

BEN YEHUDA, Muli et al. **The turtles project:** design and implementation of nested virtualization. Disponível em: <https://www.usenix.org/legacy/event/osdi10/tech/full_papers/Ben-Yehuda.pdf>. Haifa Israel: IBM Research 2010. Acesso em 31 out. 2015

BIS - Department for Business, Inovation & Skills. **2014 Information Security Breaches Survey**, Reino Unido 2014. Disponível em: <http://www.pwc.co.uk/audit-assurance/publications/2014-information-security-breaches-survey.jhtml>. Acesso em 7 nov. de 2014

CERT.BR (Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil). **Cartilha de segurança para Internet.** 2006. Disponível em <http://cartilha.cert.br/sobre/old/cartilha_seguranca_3.1.pdf>. Acesso em 10 maio 2015.

CORE SECURITY. ***VMWARE* vSphere Hypervisor Vulnerability**. Disponível em: <http://www.coresecurity.com/content/*VMWARE*-esx-input-validation-error>. Acesso em 31 out. 2015

FORRESTER RESEARCH. **The business value of virtualization**. Estados Unidos: Julho de 2009. Disponível em <http://www.*VMWARE*.com/files/pdf/solutions/Business-Value-Virtualization.pdf>. Acesso em 23 maio 2015.

FRANQUEIRA, V. Nunes Leal; VAN KEULEN, Maurice. **Analysis of the NIST database towards the composition of vulnerabilities in attack scenarios.** Holanda, 2008 Universidade de Twente. Disponível em < http://doc.utwente.nl/64664/1/TR_CTIT_08_08.pdf>. Acesso em: 4 maio 2015

GOODRICH, Michael T. **Introdução a segurança de computadores.** Porto Alegre: Bookman, 2013. 550p.
MITRE CORP. **CVE - Frequently Asked Questions**. <https://cve.mitre.org/about/faqs.html#b2>. Estados Unidos. Acesso em 31 out. 2015

NIST: National Institute of Standards and Technology. **National Vulnerability Database**, Estados Unidos, nov. 2014. Disponível em: <http:// nvd.nist.gov>. Acesso em 30 out. 2014.

REVISTA EXAME. **Maior ataque DDoS da história atinge servidores da CloudFlare**. Disponível em <http:// exame.abril.com.br/ tecnologia/noticias/ maior-ataque-ddos-da-historia-atinge-servidores-da-cloudflare>. Acesso em 2 nov.2015.

RIBEIRO, Marco. **Incidentes de segurança na Internet crescem 197% no Brasil.** Disponível em <http://nic.br/noticia/na-midia/incidentes-de-seguranca-na-internet-crescem-197-no-brasil/ >. Acesso em 15 dez.2015

SÊMOLA, Marcos. **Gestão da segurança da Informação**. 11.ed. Rio de Janeiro: Campus, 2003

SINGH, Simon. **The code book:** the evolution of secrecy from Mary Queen of Scots to quantum cryptography. 1st. ed. New York: Anchorbooks 1999.

STALLINGS, William. **Arquitetura e organização de computadores**: projeto para o desempenho. São Paulo: Pearson Prentice Hall, 2002.

SYMANTEC. **Pesquisa virtualização e evolução para a nuvem**: Resultados Brasil. Estados Unidos, 2011. Disponível em <https://www.symantec.com/content/pt/br/enterprise/images/theme/evolution/symantec-evolution-to-the-cloud-portugues.pdf>. Acesso em 13/12/2015.

JAMIN, Sugih. **Building socket programs.** Disponível em: <http://web.eecs.umich.edu/~sugih/courses/eecs489/common/notes/ide/>. Acesso em: 17 out. 2015

JONES, M. Tim. **Nested virtualization for the next-generation cloud**: *an introduction to nesting with KVM. S.l.p.: IBM. 2012. Disponível em:* http://www.ibm.com/developerworks/cloud/library/cl-nestedvirtualization/*. Aceso em Dezembro de 21015*

VERAS, Manoel. **Virtualização:** componente central do Datacenter. Rio de Janeiro: Brasport. 2011

VMWARE a. ***VMWARE* Milestones**, Estados Unidos, 2014. Disponível em: <http://www.*VMWARE*.com/company/news/media-resources/milestones.html>. Acesso em 25 out. 2014

VMWARE b. **Virtualization Overview.** Disponível em <https://www.*VMWARE*.com/pdf/virtualization.pdf**>.** Acesso em 1 maio 2015.
VMWARE c. **Configuration Maximums:** *VMWARE* vSphere 5.1. Disponível em <http://www.*VMWARE*.com/pdf/vsphere5/r51/vsphere-51-configuration-maximums.pdf>. Acesso em: 12 set 2015.

VMWARE d. **Support for running ESXi/ESX as a nested virtualization solution**, <http://kb.*VMWARE*.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2009916>. Acesso em 31 out. 2015

VMWARE e. ***VMWARE* Advisory 2012-0016.** Disponível em: <https://www.*VMWARE*.com/security/advisories/VMSA-2012-0016>. Acesso em 10 out. 2015

VMWARE Education Services. ***VMWARE* vSphere: Install, Configure, Manage.** 2012
WHITMAN, Michael. **Principles of Information Security.** 4th ed. Cengage Learning, 2001. 656 p.

ANEXO A - REPRODUÇÃO PARCIAL DA PLANILHA DE VULNERABILIDADES PARA A LINHA *VMWARE* ESXI

ESXi Versions Affected																				
3.5	4.0	4.1	5.0	5.1	5.5	CVE ID	CWE ID	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Confidentiality Impact	Integrity Impact	Availability
		X				CVE-2012-5703	20	DoS	11/12	03/13	5,0	None	Remote	Low	Not requi	None	None	Partial
X	X	X	X			CVE-2012-3289	94	DoS	06/12	06/12	7,8	None	Remote	Low	Not requi	None	None	Complet
X	X	X	X			CVE-2012-3288	20	DoS Exec Code I	06/12	11/13	9,3	None	Remote	Medium	Not requi	Complet	Complet	Complet
X	X	X	X			CVE-2012-2450		DoS Exec Code	05/12	11/13	9,0	None	Remote	Low	Single sys	Complet	Complet	Complet
X	X	X	X			CVE-2012-2449	119	DoS Exec Code (	05/12	11/13	9,0	None	Remote	Low	Single sys	Complet	Complet	Complet
X	X	X	X			CVE-2012-2448	119	DoS Exec Code (	05/12	05/12	7,5	None	Remote	Low	Not requi	Partial	Partial	Partial
X	X	X	X			CVE-2012-1518	264	+Priv	04/12	11/14	8,3	None	Local Networ	Low	Not requi	Complet	Complet	Complet
		X				CVE-2012-1517	119	DoS Exec Code (	05/12	11/13	9,0	None	Remote	Low	Single sys	Complet	Complet	Complet
X	X	X				CVE-2012-1516	119	DoS Exec Code (	05/12	11/13	9,0	None	Remote	Low	Single sys	Complet	Complet	Complet
X	X	X				CVE-2012-1515	264	+Priv	04/12	11/13	8,3	None	Local Networ	Low	Not requi	Complet	Complet	Complet
	X	X	X			CVE-2012-1510	119	Overflow +Priv	03/12	11/13	7,2	None	Local	Low	Not requi	Complet	Complet	Complet
	X	X	X			CVE-2012-1508	264	DoS +Priv	03/12	11/13	7,2	None	Local	Low	Not requi	Complet	Complet	Complet
	X	X	X	X	X	CVE-2013-5973	264		12/13	01/14	4,4	None	Local	Medium	Not requi	Partial	Partial	Partial
	X	X	X			CVE-2013-5970	20	DoS	10/13	10/13	7,1	None	Remote	Medium	Not requi	None	None	Complet
	X	X	X			CVE-2013-3658	22	Dir. Trav.	09/13	09/13	9,4	None	Remote	Low	Not requi	None	Complet	Complet
	X	X	X			CVE-2013-3657	119	DoS Exec Code (	09/13	09/13	7,5	None	Remote	Low	Not requi	Partial	Partial	Partial
	X	X	X	X		CVE-2013-3519	264	+Priv	12/13	03/14	7,9	None	Local Networ	Medium	Not requi	Complet	Complet	Complet
	X	X	X	X		CVE-2013-1661	20	DoS	09/13	09/13	4,3	None	Remote	Medium	Not requi	None	None	Partial
X	X	X	X	X		CVE-2013-1659		DoS Exec Code I	02/13	02/13	7,6	None	Remote	High	Not requi	Complet	Complet	Complet
	X	X	X	X		CVE-2013-1406	20	+Priv	02/13	11/13	7,2	None	Local	Low	Not requi	Complet	Complet	Complet
X	X	X				CVE-2013-1405	287	DoS Exec Code I	02/13	02/13	10,0	None	Remote	Low	Not requi	Complet	Complet	Complet
			X	X	X	CVE-2014-3793		DoS +Priv	05/14	06/14	5,8	User	Local Networ	Low	Not requi	Partial	Partial	Partial
	X	X	X	X		CVE-2014-1208		DoS	01/14	01/14	3,3	None	Local Networ	Low	Not requi	None	None	Partial
	X	X	X	X		CVE-2014-1207		DoS	01/14	01/14	4,3	None	Remote	Medium	Not requi	None	None	Partial
			X	X	X	CVE-2015-1044		DoS	01/15	02/15	3,3	None	Local Networ	Low	Not requi	None	None	Partial
			X	X	X	CVE-2014-8370	264	DoS +Priv	01/15	02/15	6,4	None	Remote	Low	Not requi	None	Partial	Partial
X						CVE-2008-4917	399	Mem. Corr.	12/08	05/13	7,2	None	Local	Low	Not requi	Complet	Complet	Complet
X						CVE-2008-4915	264	+Priv	11/08	05/13	6,9	Admin	Local	Medium	Not requi	Complet	Complet	Complet
X						CVE-2008-4281	22	+Priv Dir. Trav.	11/08	08/10	9,3	None	Remote	Medium	Not requi	Complet	Complet	Complet

I.

ESXi Versions Affected																		
3.5	4.0	4.1	5.0	5.1	5.5	CVE ID	CWE ID	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Confidentiality Impact	Integrity Impact	Availability
X						CVE-2008-2100	119	Exec Code Over	06/08	05/13	7,2	None	Local	Low	Not requi	Complet	Complet	Complet
X						CVE-2008-2097	119	Overflow +Priv	06/08	06/11	9,0	Admin	Remote	Low	Single sys	Complet	Complet	Complet
X						CVE-2008-0967		+Priv	06/08	05/13	6,9	Admin	Local	Medium	Not requi	Complet	Complet	Complet
X						CVE-2009-3733	22	Dir. Trav.	11/09	05/13	5,0	None	Remote	Low	Not requi	Partial	None	None
X	X					CVE-2009-2267		+Priv	11/09	05/13	6,9	None	Local	Medium	Not requi	Complet	Complet	Complet
X						CVE-2009-1805		DoS	06/09	08/10	4,0	None	Local	High	Not requi	None	None	Complet
X						CVE-2009-1244		Exec Code	04/09	05/13	6,8	Admin	Local	Low	Single sys	Complet	Complet	Complet
X						CVE-2008-4914		DoS	02/09	08/10	4,7	None	Local	Medium	Not requi	None	None	Complet
		X				CVE-2010-4573	287		12/10	01/11	9,3	None	Remote	Medium	Not requi	Complet	Complet	Complet
X	X	X				CVE-2010-4297		20 +Priv	12/10	12/10	7,2	None	Local	Low	Not requi	Complet	Complet	Complet
X	X					CVE-2010-1142	264	+Priv	04/10	05/13	8,5	None	Remote	Medium	Single sys	Complet	Complet	Complet
X	X					CVE-2010-1141	264	Exec Code	04/10	05/13	8,5	None	Remote	Medium	Single sys	Complet	Complet	Complet
X	X	X				CVE-2011-2146	200	+Info	06/11	11/14	2,1	None	Local	Low	Not requi	Partial	None	None
X	X	X				CVE-2011-2145	264		06/11	11/14	6,3	None	Local	Medium	Not requi	None	Complet	Complet
	X	X				CVE-2011-1789	310		05/11	05/11	5,0	None	Remote	Low	Not requi	None	Partial	None
X	X	X				CVE-2011-1787	362	+Priv	06/11	11/14	6,9	None	Local	Medium	Not requi	Complet	Complet	Complet
		X				CVE-2011-1786	399	DoS	05/11	10/11	5,0	None	Remote	Low	Not requi	None	None	Partial
	X	X				**CVE-2011-1785**	399	DoS	05/11	01/12	7,8	None	Remote	Low	Not requi	None	None	Complet
	X	X				**CVE-2011-0355**	399	DoS	02/11	09/11	7,8	None	Remote	Low	Not requi	None	None	Complet
	X	X				CVE-2010-3609		DoS	03/11	01/14	5,0	None	Remote	Low	Not requi	None	None	Partial

ANEXO B - LISTA DE ENDEREÇOS OU LINKS DA INTERNET PARA SE OBTER CÓPIA DOS RELATÓRIOS DAS FERRAMENTAS DE SEGURANÇA E DE MATERIAL COMPLEMENTAR

* - Para acesso a página ou “pasta” contendo todos os arquivos mencionados nesse anexo, favor acessar o endereço em versão encurtada, “short url”: <https://goo.gl/Frma5H>

* - Relatório da ferramenta Nessus para todos servidores *VMWARE*: ESX 3.5, ESXi 4.0, ESXi 4.1, ESXi 5.0 e ESXi 5.5.. Mencionado no estudo prático da vulnerabilidade vSphere API DoS, identificação CVE 2012-5703.
Ordenado por servidor, em formato Adobe Acrobat ou PDF . endereço: <https://goo.gl/M6W4Bo>
Ordenado por vulnerabilidades, em formato Hiper texto ou HTML, endereço: <https://goo.gl/TK1In9>

* - Relatório da ferramenta Nessus para o servidor ESX 4.1, mencionado no estudo prático da vulnerabilidade vSphere API DoS, identificação CVE 2012-5703. Em formato PDF
Endereço: <https://goo.gl/Boz5hv>

* - Cópia eletrônica do presente estudo, em formato Adobe Acrobat ou PDF. Endereço: https://goo.gl/Vcggsw

Alexandre Garcia Aguado

Mestre em Tecnologia e Inovação pela Faculdade de Tecnologia da Unicamp (2012) e Graduado em Tecnologia em Software Livre pelo Centro Universitário Salesiano de São Paulo (2007). Atualmente é professor no Instituto Federal de São Paulo - Campus Capivari, onde coordena o Projeto Jovem Hacker - Capivari. Antes de iniciar a carreira acadêmica foi Analista de Sistemas na Celestica Corporation, suportando os sites do Canadá, EUA, México e Brasil, desligando-se em Setembro/2009. Durante todo o ano de 2011 esteve em Angola como Voluntário através dos Salesianos de Dom Bosco onde coordenou as atividades de estruturação da área de Tecnologia da Informação, tendo como foco a criação de infraestruturas de T.I das obras Salesianas e estruturação dos programas de Formação Profissional em Informática de Jovens Angolanos.
Contato: ale.garcia.aguado@gmail.com
Fonte: CNPQ – Curriculo Lates