MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE MESTRADO EM SISTEMAS E COMPUTAÇÃO

FÁBIO SANTOS DE ARAÚJO

UMA MÉTRICA PARA CLASSIFICAÇÃO DE VULNERABILIDADES BASEADA NA TOPOLOGIA DA REDE

Rio de Janeiro
2015

INSTITUTO MILITAR DE ENGENHARIA

FÁBIO SANTOS DE ARAÚJO

UMA MÉTRICA PARA CLASSIFICAÇÃO DE VULNERABILIDADES BASEADA NA TOPOLOGIA DA REDE

Dissertação de Mestrado apresentada ao Curso de Mestrado em Sistemas e Computação do Instituto Militar de Engenharia, como requisito parcial para obtenção do título de Mestre em Ciências em Sistemas e Computação.

Orientador:
Prof. Anderson Fernandes Pereira dos Santos- D.Sc.

Rio de Janeiro
2015

INSTITUTO MILITAR DE ENGENHARIA

Praça General Tibúrcio, 80 – Praia Vermelha

Rio de Janeiro – RJ CEP: 22290-270

004.65 Araújo, Fábio Santos de
A663m

Uma métrica para classificação de vulnerabilidade baseada na topologia da rede / Fábio Santos de Araújo, orientado por Anderson Fernandes P dos Santos – Rio de Janeiro: Instituto Militar de Engenharia, 2015.

58p. : il.

Dissertação de Mestrado – Instituto Militar de Engenharia: Rio de Janeiro, 2015.

1. Curso de Sistemas e Computação – teses, dissertações. 2. Vulnerabilidades. I. Santos, Anderson Fernandes P dos. II. Uma métrica para classificação de vulnerabilidade baseada na topologia da rede. III. Instituto Militar de Engenharia.

INSTITUTO MILITAR DE ENGENHARIA

FÁBIO SANTOS DE ARAÚJO

UMA MÉTRICA PARA CLASSIFICAÇÃO DE VULNERABILIDADES BASEADA NA TOPOLOGIA DA REDE

Dissertação de Mestrado apresentada ao Curso de Mestrado em Sistemas e Computação do Instituto Militar de Engenharia, como requisito parcial para obtenção do título de Mestre em Ciências em Sistemas e Computação.

Orientador: Prof. Anderson Fernandes Pereira dos Santos–D.Sc.

Aprovada em 26 de agosto de 2015 pela seguinte Banca Examinadora:

_____

Prof. Anderson Fernandes Pereira dos Santos–D.Sc. do IME – Presidente

_____

Prof. Maria Claudia Reis Cavalcanti–D.Sc. do IME

_____

Prof. Sidney Cunha de Lucena– D.Sc. da UNIRIO

Rio de Janeiro
2015

Dedico este trabalho a minha esposa e filhos.

AGRADECIMENTOS

Agradeço a todos que de alguma maneira fizeram parte desta pesquisa, seja pelo apoio ou pela compreensão às minhas ausências por vezes necessárias.

Aos professores Maria Claudia Reis Cavalcanti e Sidney Cunha de Lemos por terem aceitado participar de minha banca examinadora.

Em especial, ao meu orientador e professor major Anderson, por compartilhar seus conhecimentos e pela atenção ao longo deste trabalho.

Aos meus colegas de trabalho, em especial: Rafael Novaes do Lago e Sergio Medeiros da Nóbrega, que me apoiaram incondicionalmente para a concretização deste trabalho.

A Marinha do Brasil, por todas as oportunidades fornecidas para o meu aprimoramento pessoal e profissional durante o transcurso de minha carreira militar.

Por fim, agradeço especialmente aminha esposa Elaine e meus filhos Kaio Fabio, Kalebe e Kaique por entenderem as diversas horas de ausência durante o período da pesquisa.

Fábio Santos de Araújo

"Se você encontrar um caminho sem obstáculos, ele provavelmente não leva a lugar nenhum."

Frank Clark

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS

ABNT	-	Associação Brasileira de Normas Técnicas
CVE	-	*Commom Vulnerability and Exposure*
CVSS	-	*Common Vulnerability Scoring System*
EUA	-	Estados Unidos da América
FIRST	-	*Forum of Incident Response and Security Teams*
IME	-	Instituto Militar de Engenharia
ISO	-	*International Organization for Standardization*
MAV	-	*Median Active Vulnerability per Day*
NIST	-	*National Institute of Standards and Technology*
NVD	-	*National Vulnerability Database*
SCAP	-	*Security Content Automation Protocol Validated Products*
SP	-	*Special Publication*
TI	-	Tecnologia da Informação
TTNV	-	*Time to Next Vulnerability*
USGCB	-	*United States Government Configuration Baseline*
VFD	-	*Vulnerability Free Day*
WEKA	-	*Waika toEnvironment for Knowledge Analysis*
XML	-	*eXtensible Markup Language*

RESUMO

A principal dificuldade encontrada pelos usuários de *software* ao consultar um relatório com as vulnerabilidades encontradas é saber o risco caso estas sejam exploradas. Os repositórios trazem informações genéricas, não levando em consideração o escopo ou particularidades. Muitos usuários utilizam as informações obtidas priorizando sem um padrão definido, de acordo com os seus próprios conhecimentos.

Este trabalho apresenta uma métrica baseada na topologia da rede, onde as vulnerabilidades são classificadas de acordo com o cenário apresentado da rede em produção e um ativo considerado como principal pelo usuário.

Tem como base a associação da rede a uma estrutura de árvore n- área através de um algoritmo sugerido na pesquisa, o repositório de vulnerabilidade NVD de onde serão retirados os dados importantes e a métrica a fim de ponderar a classificação feita pelo relatório retirado do portal do NVD.

ABSTRACT

The main difficulty encountered by software users to consult a report with the vulnerabilities found is to know the risk IF they are exploited. Repositories bring generic information, not taking into account the scope or features. Many users use the information obtained without prioritizing a set pattern, according to their own knowledge.

This paper presents a metric based on network topology, where the vulnerabilities are classified according to the displayed network scenario in production and anactive regarded as the main user.

Is based on the association of the network to an n-area tree structure through na algorithm suggested in the survey, the NVD vulnerability repository where important data Will be removed and the metric in order to consider the classification made by the withdrawn report portal the NVD.

1 INTRODUÇÃO

O uso da tecnologia se tornou cada vez mais comum e complexo no meio da sociedade, encontrada em vários tipos de aparelhos e utilizadas por pessoas de diversas faixas etárias. Em paralelo surgem novas ferramentas e aplicativos e concomitantemente um grande número de novas vulnerabilidades de *software*. Com este cenário aumenta a preocupação com a segurança das informações e nascem vários repositórios que caracterizam e classificam a vulnerabilidade a fim de orientar os esforços na defesa de ativos.

Um importante repositório é o *National Vulnerability Database* (NVD), que segundo Zhang (ZHANG, 2011b), é uma fonte de dados pública que mantém informações padronizadas sobre relatórios de vulnerabilidades de *software*, repositório este que em seus primeiros dez anos publicou informações sobre mais de 43.000 vulnerabilidades em mais de 17.000 aplicativos.

Porém estes repositórios utilizam métricas padronizadas que caracterizam as vulnerabilidades de forma ampla, genérica, sem considerar as particularidades de nenhuma topologia de rede.

1.1 MOTIVAÇÃO

Atualmente várias empresas têm um contingente de funcionários reduzido, o que leva ao acumulo de várias funções. Na maioria das organizações a área de tecnologia e Informação (TI) é a mais afetada onde raramente se observa uma divisão responsável, somente, pela segurança das informações.

Segundo (BALDISSERA, 2006), para as organizações a informação passou a ser um dos principais ativos, necessitando da proteção da sua confidencialidade, manutenção da integridade e disponibilidade.

No final do ano de 2010 o *Wikileaks* divulgou em sua página centenas de documentos elaborados por autoridades norte-americanas sobre países como Rússia, Paquistão entre outros (MAYNARD, 2011). Esse foi nos tempos atuais, o maior exemplo da importância da informação em uma organização ou governo.

1.2 OBJETIVOS

Propor uma métrica para a classificação de vulnerabilidade baseando-se na topologia de uma rede em produção e no seu principal ativo.

O trabalho terá como foco as informações, logo as partes da estrutura mais importantes nas redes analisadas serão os servidores de banco de dados. Porém nada impede que possa ser considerado um *software* ou serviço como principal ativo, por exemplo.

Além do objetivo geral apresentado, este trabalho possui o objetivo específico de apresentar um vetor que classificará as vulnerabilidades de acordo com o risco sobre um ativo principal em rede em produção.

1.3 CONTRIBUIÇÕES ESPERADAS

As contribuições esperadas pelo trabalho são:

1) Métrica para a classificação de vulnerabilidades.

2)Realizar estudo de casos a fim de validar a métrica.

1.4 ORGANIZAÇÃO DO TRABALHO

A dissertação está estruturada da seguinte forma: no capítulo 2 são apresentados os conceitos básicos, com o referencial teórico necessário para o entendimento deste trabalho, o capítulo 3 o conceito da métrica sugerida; o capítulo 4 descrito a arquitetura do protótipo e como a métrica foi implementada, o capítulo 5 apresenta o emprego da métrica através de estudos de caso; no capítulo 6 trabalhos relacionados; e no capítulo 7 a conclusão do trabalho, suas contribuições e sugestões de trabalhos futuros.

2 CONCEITOS BÁSICOS

Na sociedade atual, onde a informação tem sido considerada o principal patrimônio, a segurança cresce ao passo que os sistemas de comunicações evoluem. Alguns conceitos utilizados no assunto evoluíram outros surgiram em um cenário de sofisticação e complexidades dos novos ataques.

Atualmente somente o uso de um bom antivírus não é suficiente para a manutenção dos princípios básicos da segurança da informação, e sim habilidades para se manter sempre a frente das ameaças.

Para entender um pouco mais do assunto proposto por esta pesquisa será explorado alguns conceitos importantes.

2.1 AMEAÇA

Redes de computadores são expostas a diversos tipos de ameaças à segurança da informação, como sabotagem, roubo de dados, alterações de dados.

Ameaça pode ser definida como a possibilidade de um agente externo, ou fonte de ameaça, explorar acidentalmente ou propositalmente uma vulnerabilidade (NIST SP 800-30).

O termo ameaça se refere ao risco ou possível perigo que uma situação ou circunstância leva perigo a algo ou alguém, e em TI esta expressão pode ser associada ao risco de um agente externo explorar as vulnerabilidades afim de atingir as informações.

Por isso a importância de sempre ao falar em ameaça ter bem claro os conceitos de risco e vulnerabilidade.

2.2 VULNERABILIDADE

Falha ou fraqueza de procedimento, implementação ou controle interno de um sistema de informação que possa acidentalmente ou propositalmente ser explorada, resultando em uma brecha de segurança (NIST SP 800-30).

Em outras palavras vulnerabilidades são fraquezas que podem permitir a quebra de um ou mais princípios de segurança da informação: confidencialidade, integridade e disponibilidade.

Pionti (PIONTI, 2001), se refere a vulnerabilidade como sendo o ponto onde qualquer sistema é fragilizado a um ataque, condição causada muitas vezes pela ausência das medidas de proteção.

2.3 RISCO

Probabilidade de uma fonte de ameaça explorar uma vulnerabilidade, resultando em um impacto, dano, para uma organização (NIST SP 800-30).

Para este trabalho a análise dos riscos terá como foco o dano que pode ser causado ao ativo principal, as informações, caso uma ameaça obtenha êxito na exploração das vulnerabilidades existente em uma rede em produção. Danos estes relacionados, principalmente, a quebra dos três princípios básicos da segurança da informação:

- **CONFIDENCIALIDADE –** proteger informações contra sua revelação para alguém não autorizado interna ou externamente. Consiste em proteger a informação contra leitura e ou cópia por alguém que não tenha sido explicitamente autorizado pelo proprietário daquela informação (DE OLIVEIRA, 2011).
- **INTEGRIDADE -** proteger a informação contra modificação sem a permissão explícita do proprietário daquela informação. A modificação inclui ações como escrita, alteração de conteúdo, alteração de status, remoção e criação de informações (DE OLIVEIRA, 2011).
- **DISPONIBILIDADE -** consiste na proteção dos serviços prestados pelo sistema de forma que eles não se tornem indisponíveis sem autorização, assegurando ao usuário o acesso aos dados sempre que deles precisar. Isto pode ser chamado também de continuidade dos serviços (DE OLIVEIRA, 2011).

2.4 REDE DE COMPUTADORES

Outro conceito importante a ser explorado para entendermos a pesquisa é a ideia de redes de computadores.

As redes de computadores abrangem qualquer sistema interconectado através de núcleos de processamento. O usuário pode acessar qualquer computador de outro terminal situado no sistema (COSTA, 2012).

Segundo Tanebaum (TANEBAUM, 1994), um dos objetivos das redes de computadores é o compartilhamento de recursos, onde os programas, dados e equipamentos estão disponíveis a todo o usuário independente da sua localização física.

Quando se fala em rede de computadores se refere as estruturas físicas (equipamentos) e lógicas (programas, protocolos) e neste trabalho será dado ênfase a estrutura lógica, analisando os *software* que contém vulnerabilidades listadas no portal do *National Vulnerability Database* (NVD).

2.5 NATIONAL VULNERABILITY DATABASE (NVD)

O *National Vulnerability Database* (NVD) (NVD,HOME), é um repositório do governo dos Estados Unidos da América (EUA), patrocinado pelo Departamento de Divisão de Segurança Cibernética Nacional de Segurança Interna, baseado em gerenciamento de segurança.

O repositório NVD contém vulnerabilidades em produtos validados por testes e verificações em sistemas como: *United States Government Configuration Baseline* (USGCB) (USGCB,NIST) e *Security Content Automation Protocol Validated Products*(SCAP) (SCAP,NIST).

O USGCB fornece orientações para as agências sobre o que deve ser feito para melhorar e manter uma definição de configuração eficaz centrada na segurança.

O SCAP consiste em um portal que fornece especificações existentes e emergenciais relevante para a automação de segurança do *National Institute of Standards and Technology* (NIST) (US DEPARTMENT OF COMMERCE), estas

constantes nas subséries especiais, *NIST Special Publication* (NIST SP), sobre recomendações de segurança.

O *Commom Vulnerability and Exposure* (CVE) (CVE,NIST) é o nome padronizado para vulnerabilidade de *software* e referência a alertas em banco de dados de vulnerabilidades conhecidas (KARLSSON, 2012).

O repositório NVD, além do CVE, contém informações das características da vulnerabilidade, os *softwares* que contém esta vulnerabilidade e uma especificação, um *score* resultante da métrica NVD, que categoriza a vulnerabilidade.

Ao longo dos anos este repositório têm se tornado o padrão utilizado pelos fornecedores para divulgar *patches* e pelos pesquisadores para realizar estudos sobre vulnerabilidades de *software*.

2.6 MÉTRICA NVD

A categorização das vulnerabilidades segue os valores fornecidos pelo *Common Vulnerability Scoring System* (CVSS) (CVSS, SIG).

O CVSS fornece uma estrutura aberta para divulgar as características das vulnerabilidades de TI. Consiste em três grupos: Base, Temporal e Ambiental, como ilustrado na figura 2.1. Cada grupo produz uma pontuação numérica (0 a 10). O grupo de base representa as características intrínsecas de uma vulnerabilidade que são constantes ao longo do tempo. O grupo temporal reflete as características de uma vulnerabilidade que varia com o tempo. O grupo Ambiental representa as características de uma vulnerabilidade que são exclusivos ao usuário (OF INCIDENT RESPONSE).

O repositório NVD leva em consideração apenas o valor obtido pelos componentes do grupo de base, assim esta pesquisa se limitará as definições para este grupo, a explicação mais detalhadas dos demais poderá ser visualizada no Guia do CVSS (OF INCIDENT RESPONSE).

FIG 2.1 Grupos CVSS, adaptado (OF INCIDENT RESPONSE)

No grupo de base os componentes *Access Vector*, *Access Complexity*, e *Authentication* capturam a forma como a vulnerabilidade é acessada e se condições adicionais são necessários para explorá-la. *Confidentiality impact*, *Integrity impact* e *Avaliability impact* medem como uma vulnerabilidade, se explorada, irá afetar diretamente um ativo de TI na perda de confidencialidade, integridade e disponibilidade.

As métricas do grupo de base são especificadas pelos analistas de vulnerabilidade ou fornecedores de aplicativos.

O resultado de cada métrica dos grupos é alcançado aplicando algumas propriedades e equações do Guia do CVSS, assim como na ferramenta disponibilizada e utilizada pelo site do NVD, *Common Vulnerability Scoring System Version2Calculator (*NVD, Calculator-v2).

2.7 RELATÓRIO NVD

No portal virtual do NVD é possível ter acesso aos relatórios contendo as informações das vulnerabilidades contidas no banco de dados do NVD, relatórios este que neste trabalho será chamado de *XML Vulnerability Feeds* (NVD, Data). Estes Relatórios são atualizados e disponibilizados em arquivo formato *eXtensible Markup Language* (XML) diariamente, assim que surgem novas vulnerabilidades ou *patches* para as vulnerabilidades já lançadas no repositório.

O arquivo XML fornecido contém vários elementos caracterizando cada vulnerabilidade das quais serão relatadas as mais importantes:

- <*entry*id="CVE-ano-dígitos">: identificador, CVE, da vulnerabilidade, onde o ano é o que a vulnerabilidade foi publicada e os dígitos o identificador único de cada vulnerabilidade;
- <*vuln:published-datetime*>: data da primeira publicação da vulnerabilidade;
- <*vuln:last-modified-datetime*>: data da última modificação ou publicação de *patch* para a vulnerabilidade;
- <*cpe-lang:fact-ref*>: softwares que contém a vulnerabilidade;
- <cvss:*score*>: pontuação base CVSS. Esta pontuação pode variar de 0 a 10, onde a gravidade da vulnerabilidade é classificada de três formas distintas, gravidade baixa, quando o *score* varia de 0 a 3,9; média quando de 4 a 6,9 e alta quando de 7 a 10; e
- <vuln:*summary*>: características da vulnerabilidade.

2.8 CICLO DE VIDA DA VULNERABILIDADE

Quando se fala em vulnerabilidade é importante entender o seu ciclo de vida, conceito utilizado por várias pesquisas e sistemas que tratam do assunto de segurança.

O ciclo se inicia pela descoberta da vulnerabilidade em determinado *software* pelo próprio fabricante ou por entidades externas destinadas a este fim. Após a descoberta é criado um *exploit* e comunicado o vendedor que por sua vez inicia o desenvolvimento de *patches*. O ciclo se encerra com a divulgação e aplicação das soluções.

3 MÉTRICA PROPOSTA

Durante a elaboração da métrica foi identificada a necessidade da pesquisa de trabalhos anteriores a fim de determinar os fatores que seriam explorados. Cabe ressaltar que não se assemelham a linha de pesquisa proposta, apenas orientam quais condições são mais citadas quando se sugere uma métrica na área de segurança das informações.

3.1 MATRIZ DE ANÁLISE DE RISCO

Baldissera (BALDISSERA, 2006), baseando-se nas normas ABNT ISO 27.002:2005, desenvolveu a matriz de análise de risco que é aplicada para quantificar a importância do perímetro da informação visando o aprimoramento do planejamento de segurança.

Em sua pesquisa, Baldissera considera o fluxo da informação distribuído e compartilhado e para planejar uma segurança efetiva se utiliza do conceito de perímetro. Sêmola (Sêmola, 2003), exemplifica perímetro utilizando uma casa que por padrão tem duas portas, e caso não se realize um estudo qual porta seja mais vulnerável, poderá haver um desperdício de investimento na segurança.

A figura 3.1 mostra o conceito utilizado no trabalho, onde um ataque em um perímetro mais próximo à informação requer uma melhor proteção.

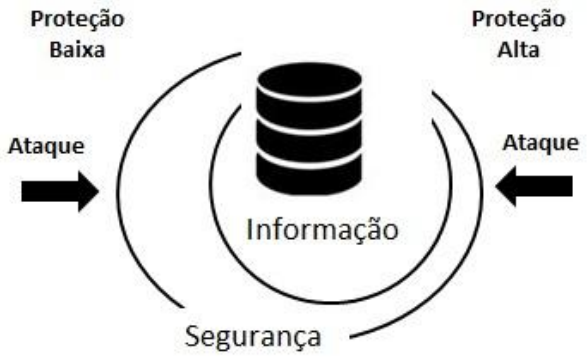

FIG 3.1 Exemplo do Conceito de Perímetro

3.2 MODELO PARA QUANTIFICAÇÃO DE SEGURANÇA

Dacier (DACIER, 1996), propõe uma abordagem para dar suporte aos administradores de sistemas computacionais no monitoramento da segurança de seus ativos. A abordagem é baseada na modelagem do sistema como um grafo de privilégios, expondo as vulnerabilidades de segurança.

Neste trabalho, o grafo de privilégios é comparado a um labirinto, onde cada aresta corresponde a uma porta que permite o acesso de um nó para o outro. As portas são equipadas com fechaduras de diferentes qualidades que pode ser caracterizada por um parâmetro que determina a taxa de sucesso do processo de ruptura.

Os fatores tempo e esforço utilizados pelo atacante para quebrar as fechaduras e obter êxito são utilizados para caracterizar e avaliar a segurança e duas propriedades são observadas:

- A segurança é diretamente proporcional ao tempo necessário para que um atacante obtenha sucesso; e
- A segurança é diretamente proporcional ao esforço necessário para a implementação de um ataque.

3.3 DESENVOLVIMENTO DA MÉTRICA

A métrica desenvolvida neste trabalho está fundamentada nos conceitos, já apresentados, de perímetro e labirinto.

Para iniciar o desenvolvimento da métrica foi realizada uma comparação de uma rede a uma estrutura de árvore n-área, onde o nó raiz seria o ativo principal e os nós filhos os outros elementos da rede, assim podendo fazer um paralelo com os níveis da árvore e aplicar os conceitos mencionados.

Para esclarecer o procedimento sugerido foi demonstrada a associação da figura 3.2 a uma árvore n-área seguindo os passos de um algoritmo na forma narrativa que será chamado de Métrica Sugerida.

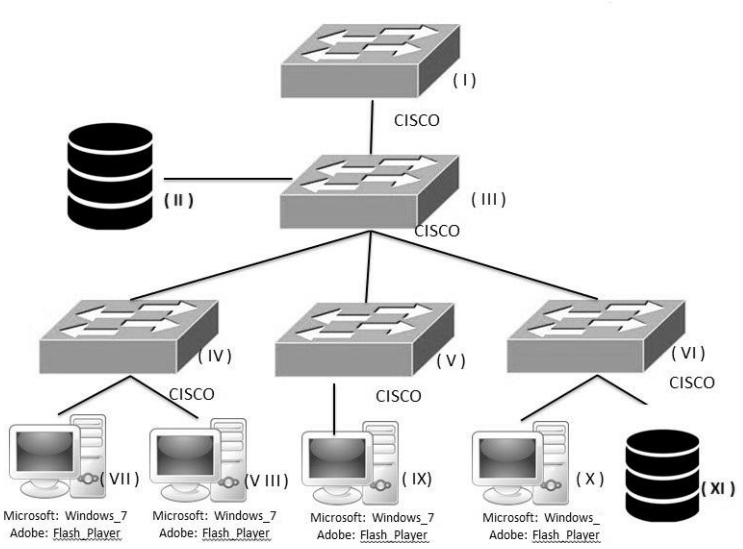

FIG 3.2 Exemplo de Rede Lógica em Produção

Algoritmo Métrica Sugerida

Início

1. Identificar o Ativo Principal.
2. Considerar este Ativo como raiz da árvore.
3. Verificar se existem elementos diretamente ligados a raiz.

Se Ativo Principal contém elementos diretamente ligados **Então**

4. Considerar elementos como nós filhos da raiz.
5. Verificar se existem nós filhos com a mesma configuração.

Se existem filhos iguais **Então**

6. Unificar os filhos iguais em somente um nó filho.

Fim Se

Senão

7. Considerar árvore montada, fim do algoritmo.

Fim Se

Repita

8.Verificar para cada nó filho se existem elementos diretamente ligados.

Para cada nó filho que preencha esta condição **faça**

Se filho contém elementos ligados **então**

9. Considerar elementos como nós filhos.

10. Verificar se existem nós filhos com a mesma configuração.

Se existem filhos iguais **Então**

11. Unificar os filhos iguais em somente um nó filho.

Fim Se

Fim Se

Fim Para

Até Não existir elementos ligados diretamente a nenhum nó filho.

Fim

Exemplificando, considere a estrutura da figura 3.3, admitindo como ativo principal o servidor de banco de dados II e que os *switchs* da Cisco tem as mesmas configurações assim como os *Desktop* dos usuários.

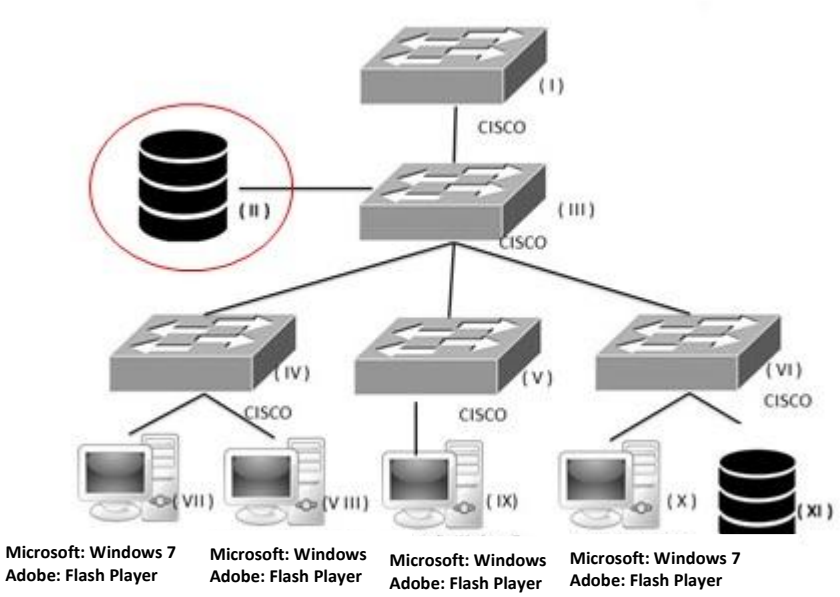

FIG 3.3 Passos 1 e 2 do algoritmo Métrica Sugerida

As figuras a seguir exemplificam cada passo do algoritmo, de acordo com a estrutura apresentada anteriormente.

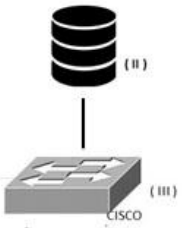

FIG 3.4 Passos 3 e 4 do algoritmo Métrica Sugerida

Não existem nós com a mesma configuração, assim os passos 5 e 6 são desconsiderados.

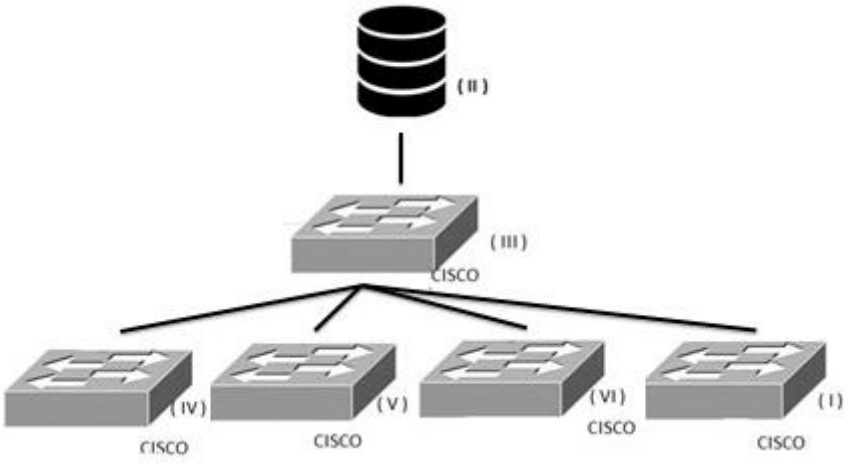

FIG 3.5 Passos 8 e 9 do algoritmo Métrica Sugerida

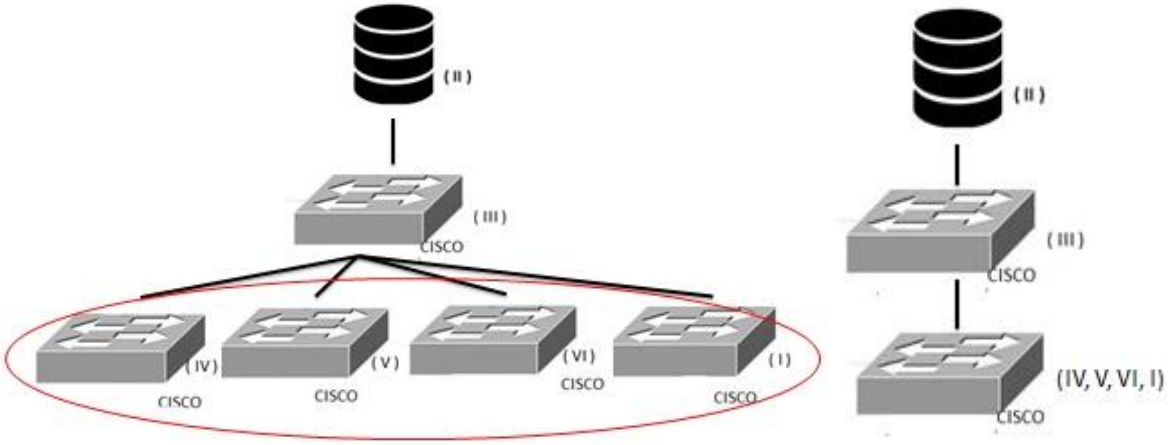

FIG 3.6 Passos 10 e 11 do algoritmo Métrica Sugerida

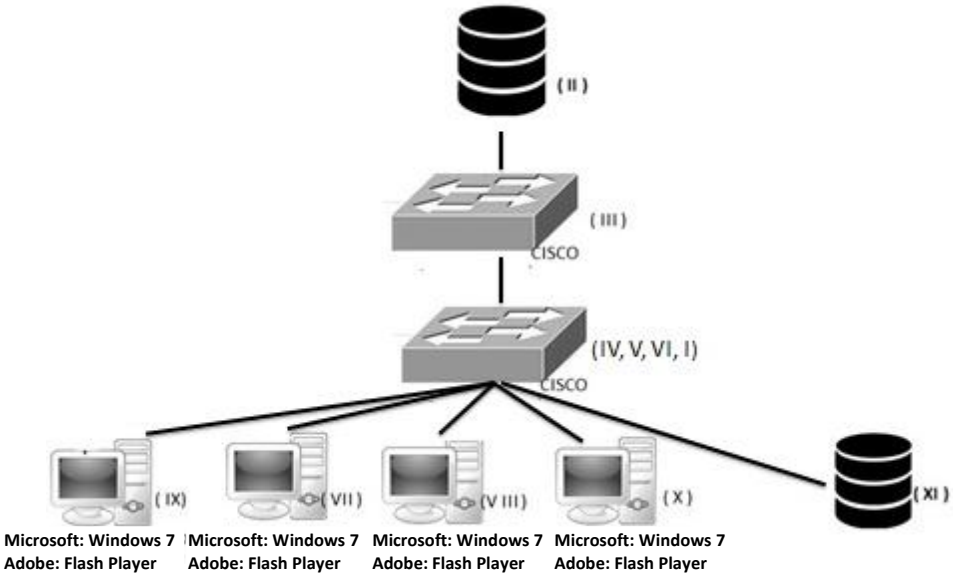

FIG 3.7 Passos 8 e 9 do laço repita do algoritmo Métrica Sugerida

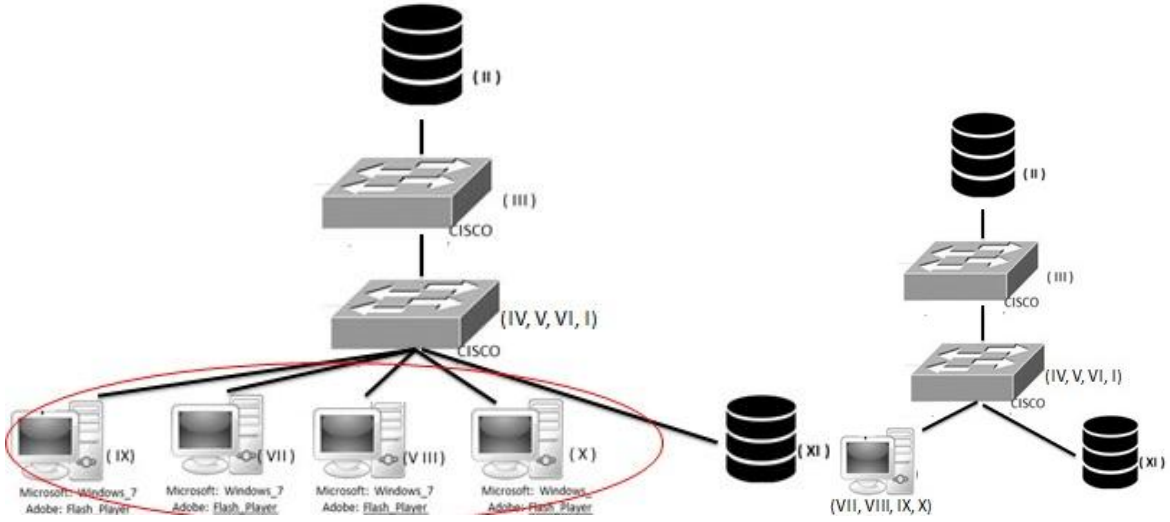

FIG 3.8 Passos 10 e 11 do laço repita do algoritmo Métrica Sugerida

Associar a rede a uma estrutura de árvore também possuí uma vantagem adicional, a flexibilidade da escolha do ativo principal. Neste exemplo, há dois ativos principais, uma vez que é classificado desta forma quando contem um SGBD, conforme figura 3.9.

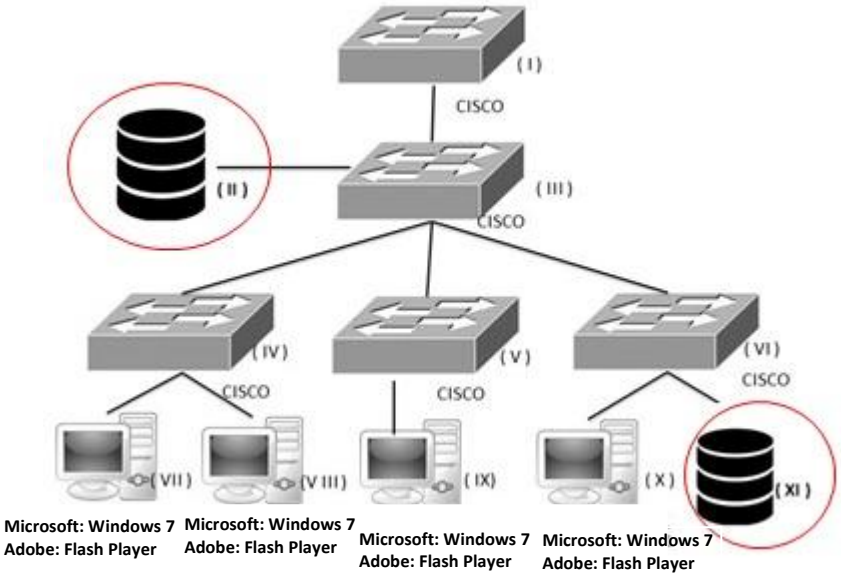

FIG 3.9 Ativos da Rede em Produção

Nesta mesma rede, caso seja considerado como ativo principal o servidor de Banco de Dados XI e aplicado o algoritmo Métrica Sugerida, terá como resultado a estrutura de árvore apresentada na figura 3.10.

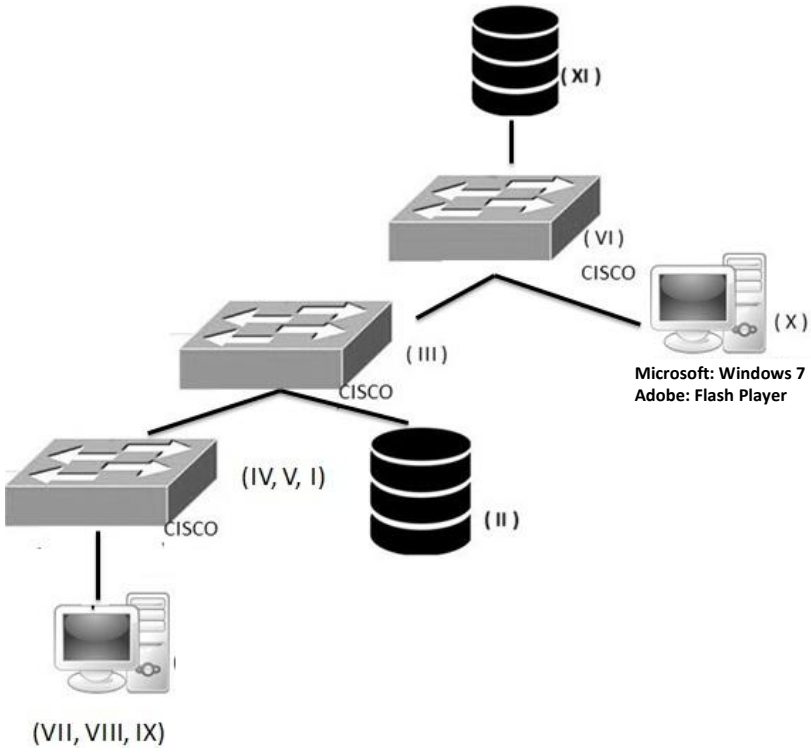

FIG 3.10 Árvore equivalente

A figura 3.11 apresenta a diferença das árvores resultantes do algoritmo considerando cada raiz como um SGDB diferente.

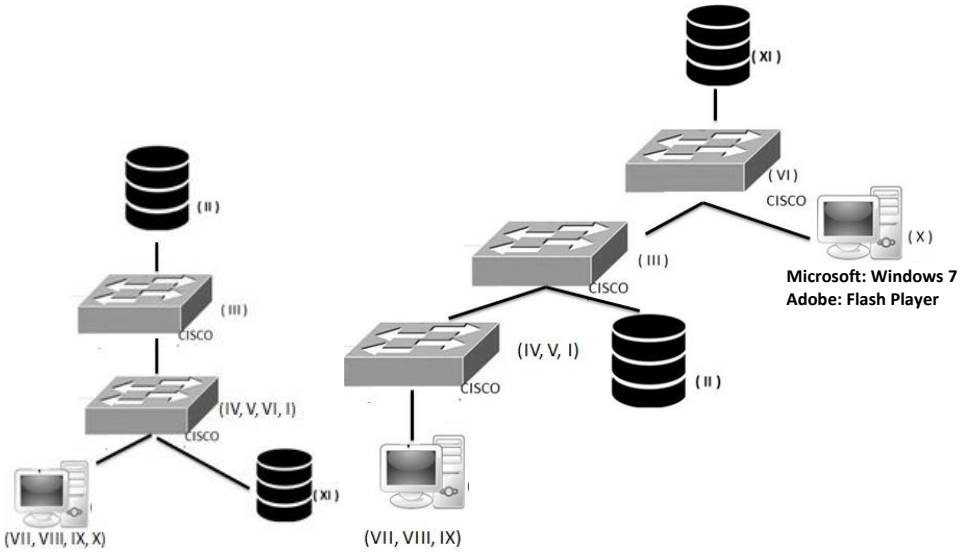

FIG 3.11 Árvores equivalentes

Cabe ressaltar que foi realizada uma pesquisa na tentativa de trabalhar na associação da rede com a estrutura de árvore binária, aproveitando suas propriedades para limitar a métrica, porém, ao verificar algumas redes em produção concluiu-se que grande parte delas não apresentaria as propriedades necessárias de árvore binária já nos níveis iniciais.

O nível da árvore foi uma propriedade importante neste estudo, pois a distância dos nós até a raiz, que é o ativo principal, foi utilizado como referência para se aumentar os fatores estabelecidos nesta pesquisa para a métrica, o esforço e tempo.

Foi considerado que quanto mais próximo a vulnerabilidade do ativo principal, maior a probabilidade do atacante obter sucesso e acessar de alguma forma as informações.

Pensando estabelecer alguns parâmetros de ponderação do *Score* NVD, foi decidido dividir a estrutura em três partes iguais, figura 3.12, de forma a realizar associações as três classificações distintas às vulnerabilidades feitas pelo repositório NVD.

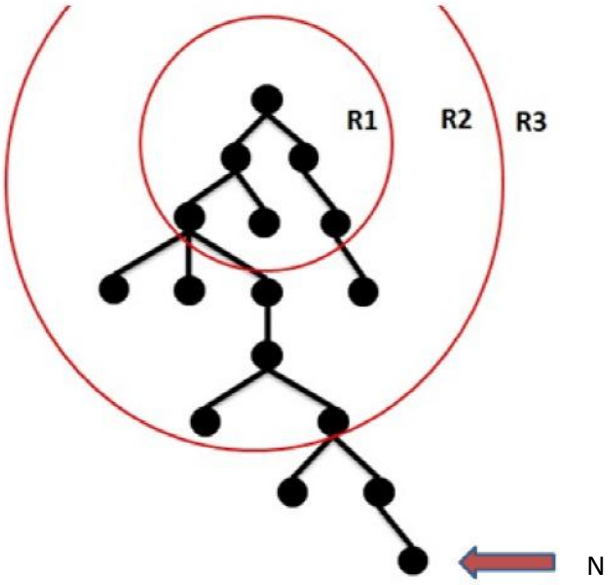

FIG 3.12 Divisão da árvore

A figura 3.12 elucida a divisão que seguirá os seguintes parâmetros:

- A estrutura será dividida em três partes iguais: R1 = (1\3) * N; R2 = (2\3) * N e R3 = (3\3) * N, onde N é o maior valor do maior nível da árvore.
- Os *softwares* dos nós com os níveis compreendidos entre 1 ≤ N ≤ R1, mais próximos ao ativo principal, e relatados no *XML Vulnerability Feeds com Score* <7, terão os seus valores acrescidos de 3. O objetivo é elevar a gravidade das vulnerabilidades de forma a aumentar a prioridade em suas correções.
- Os *softwares* dos nós com os níveis compreendidos entre R1 < N ≤ R2 e relatados no *XML Vulnerability Feeds* terão seu *score* mantidos.
- Os *softwares* dos nós com os níveis compreendidos entre R2 < N ≤ R3, mais afastados do ativo principal, e relatados no *XML Vulnerability Feeds com Score* >3.9, terão os seus valores decrescidos de 3. O objetivo é diminuir a gravidade da vulnerabilidade de forma a despriorizar suas correções.

4 ARQUITETURA E IMPLEMENTAÇÃO DO PROTÓTIPO SUGERIDO

Foi idealizado e implementado o protótipo que será utilizado para empregar a métrica sugerida e auxiliar no estudo de casos no final desta pesquisa.

O resultado final será um vetor que classificará as vulnerabilidades de uma rede em produção de acordo com o risco sobre um ativo principal, permitindo assim, quantificar as métricas que permitirão auxiliar o usuário quanto a prioridade nas correções das brechas encontradas. Este vetor prioriza as vulnerabilidades de acordo com os conceitos e as métrica apresentadas no Capítulo 3.

O conceito do protótipo se resume a três fases:

- Consultar os dados de um repositório de vulnerabilidade válido, para a nossa pesquisa o NVD;
- Confrontar os dados do repositório com os dados de uma rede em produção; e
- Aplicar a métrica visando ponderar os dados do repositório e gerar o vetor de classificação das vulnerabilidades.

4.1 ARQUITETURA E IMPLEMENTAÇÃODO PROTÓTIPO

Para facilitar o entendimento será explicada a arquitetura em paralelo com a implementação do sistema de acordo com as três fases descritas. De um modo geral, a arquitetura pode ser visualizada, de forma objetiva, nas figuras 4.1 e 4.2 e explicada de forma detalhada nos próximos tópicos.

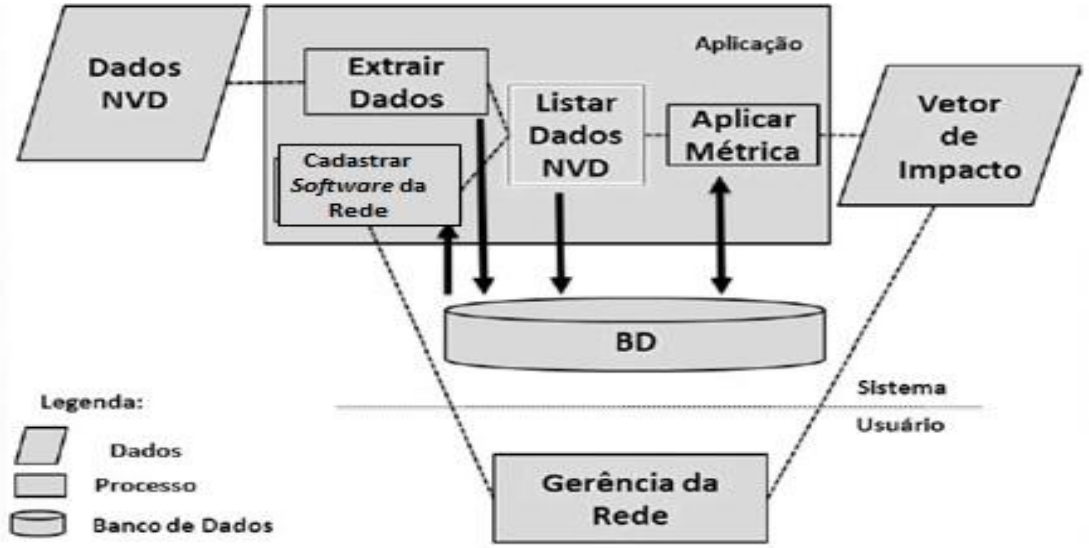

FIG 4.1 Arquitetura Proposta

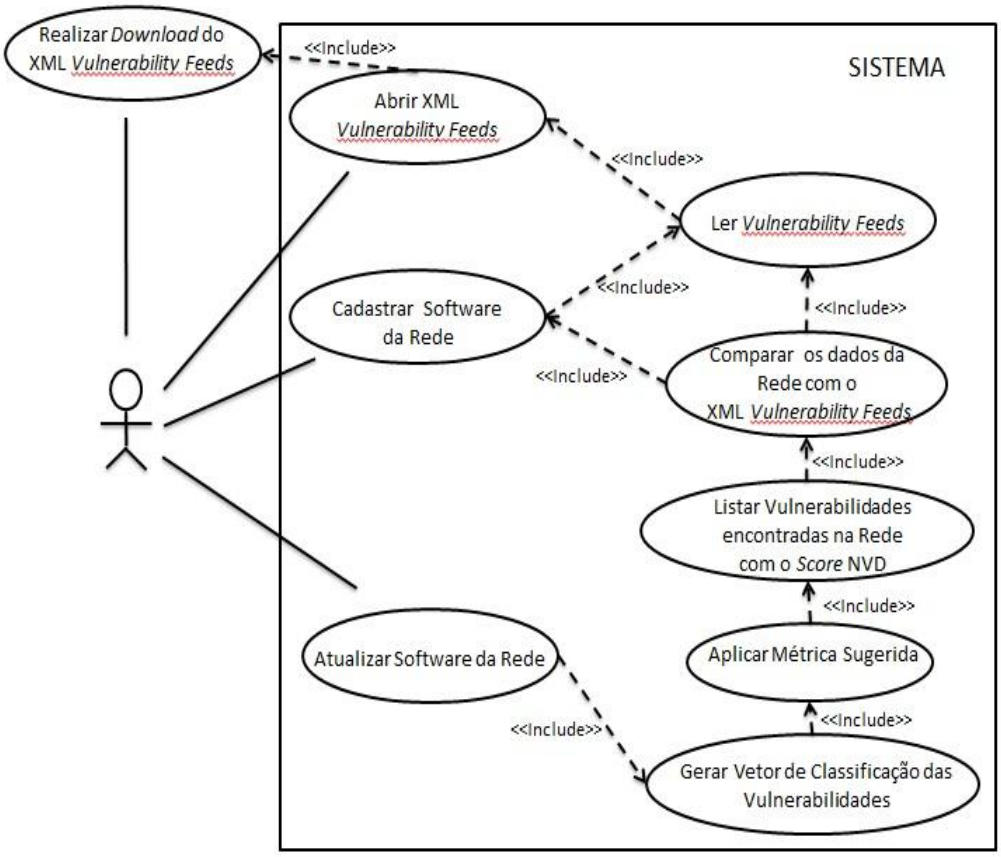

FIG 4.2 Diagrama de Caso de Uso

4.1.1 CONSULTAR OS DADOS DO REPOSITÓRIO

O início do processo é realizado a partir do *download* do XML *Vulnerability Feeds* no portal do NVD. Como já citado no Capítulo 2, o relatório fornecido é dinâmico, atualizado assim que alguma alteração ou solução apareça, porém para o protótipo a fidelidade das informações será diretamente proporcional a preocupação do cliente em realizar de forma periódica o *download* e leitura do arquivo.

Após o carregamento do XML *Vulnerability Feeds*, figura 4.3, as informações relevantes são armazenadas para o início do processo. Estas informações foram definidas no item 2.7 para a geração do vetor de classificação, conforme figura 4.4.

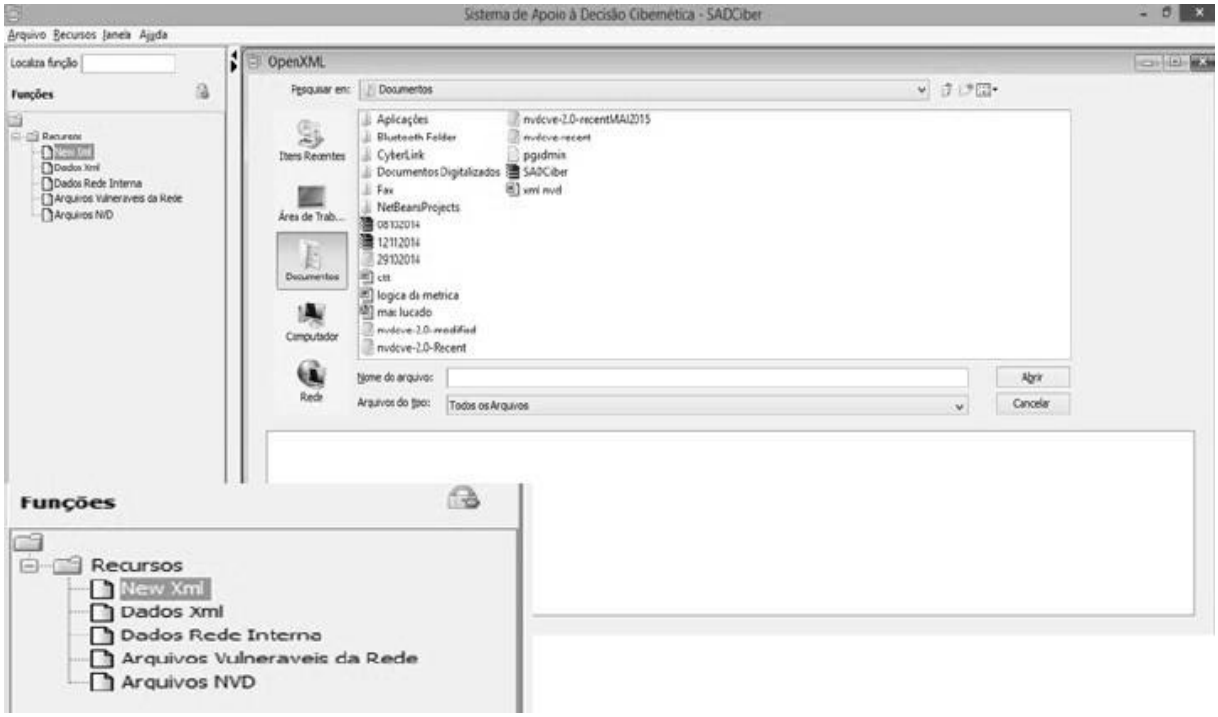

FIG 4.3 Abrir XML

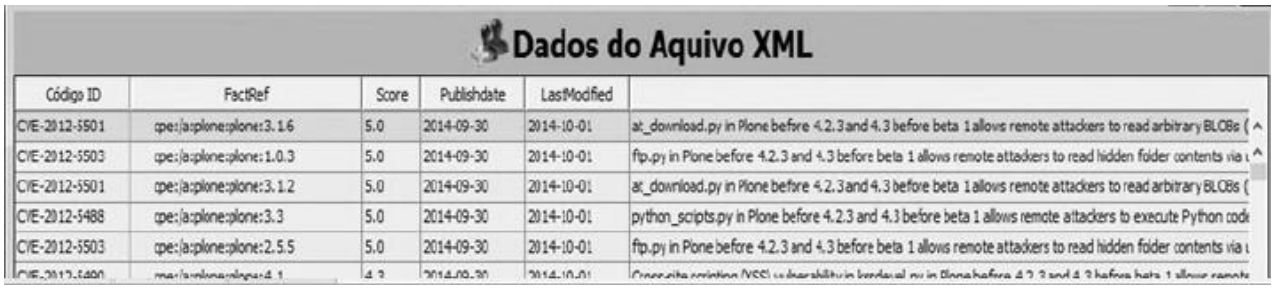

FIG 4.4 Dados extraídos do XML *Vulnerability Feeds*

Para a extração dos dados foram utilizados padrões utilizando bibliotecas JAVA, como JDOM.

4.1.2 ALINHAMENTO DOS DADOS DO REPOSITÓRIO E REDE

Os dados dos ativos apresentados no NVD possuem nomenclatura complexa, por exemplo, é apresentado novell:opensuse:12.3. Por isso o sistema já oferece ao usuário uma base inicial com todos os *softwares* conforme figura 4.5.

Com a topologia da rede já estruturada de acordo com o capítulo 3 e aliado à base de ativos, o sistema calculará o vetor com a métrica desenvolvida, conforme figuras 4.5 e 4.6.

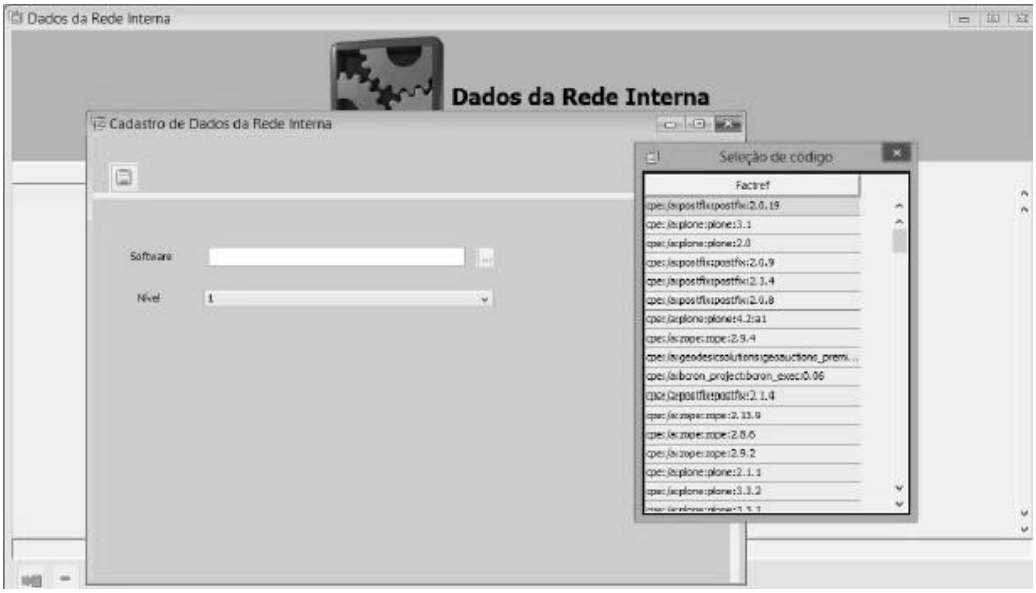

FIG 4.5 Dados extraídos do XML *Vulnerability Feeds*

4.1.3 APLICANDO A MÉTRICA SUGERIDA

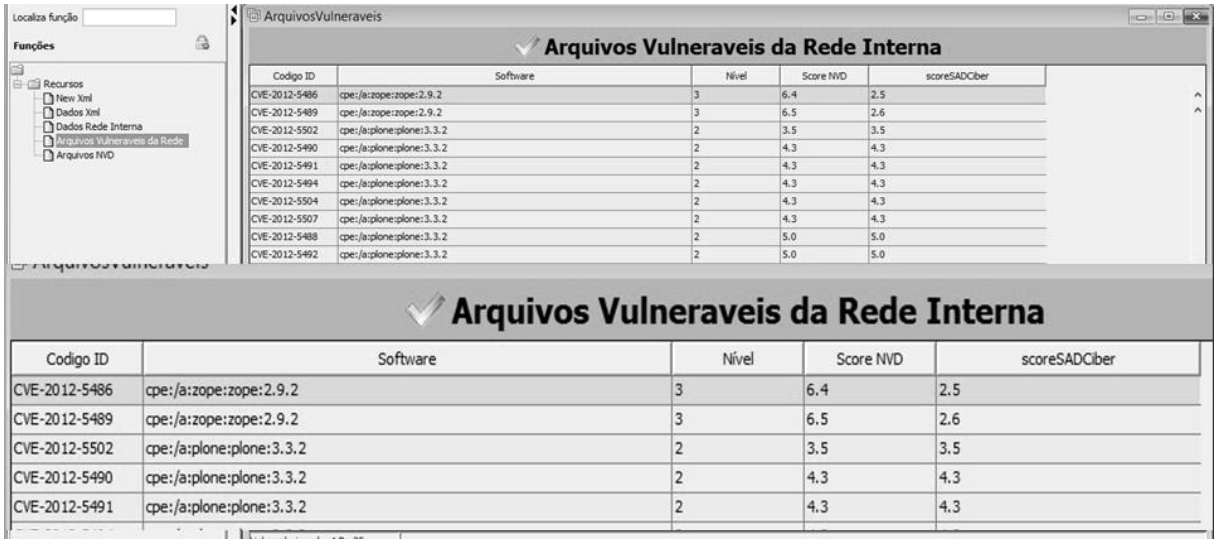

FIG 4.6 Vetor de classificação em função da métrica sugerida

Através da consulta do vetor final fornecido pelo sistema o usuário poderá envidar esforços para corrigir as falhas em sua rede, aplicando medidas de segurança necessárias como instalação de *firewall* e atualização ou substituição de *softwares* comprometidos. Com as alterações realizadas poderá reiniciar o processo com o cadastramento dos novos *softwares* ou a leitura de um novo XML *Vulnerability Feeds*.

5 ESTUDO DE CASOS

Foram criados dois cenários para se realizar o estudo de casos e, assim, verificar a viabilidade da métrica sugerida neste trabalho. Os cenários são extratos de uma rede em produção de organizações militares que seguem um determinado padrão e onde os usuários acumulam funções variadas não tendo tempo suficiente para focar nas atividades rotineiras de segurança da informação.

5.1 REDE COM UM ATIVO PRINCIPAL

O cenário apresentado foi retirado de uma organização pequena onde se tem somente um ativo principal, servidor de dados de mensagens, conforme figura 5.1.

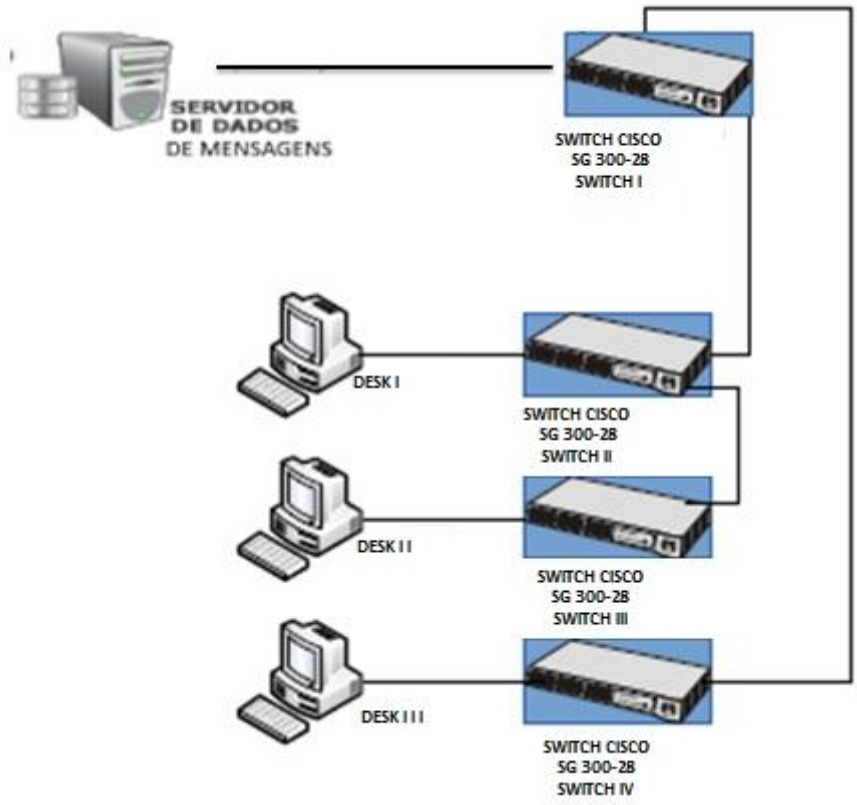

FIG 5.1 Rede em produção

A figura 5.2 Apresenta a estrutura da árvore equivalente após aplicar o algoritmo apresentado no Capítulo 3.

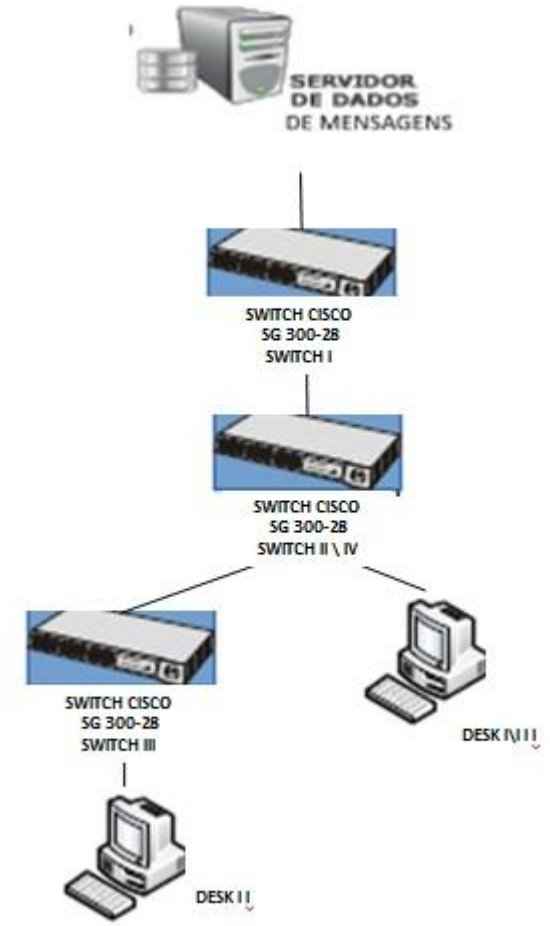

FIG 5.2 Rede estruturada em árvore n-área

Os *softwares* dos equipamentos que compõem a rede e seus respectivos Score, caso apareça no XML *Vulnerability Feeds*, são descrito a seguir:

- Servidor de Dados – novel *opensuse*:13.2 - *Score*: 6.8.
- Switch CISCO – Cisco *Secure Access Control System*:5.5 - *Score*: 4.3.
- Desk - McAfee ePolicyOrchestrator- *Score*: N\C;
  - Libreoffice:4.3.6 - *Score*: 6.8;
  - Mozilla:Firefox:38.1.0 - *Score*: 4.3;
  - Correio Eletrônico Lotus Notes- *Score*: N\C;
  - Adobe:Acrobat:10.1.14- *Score*: 4.3;
  - Adobe:*Flash_Player*:19.0.0.185 - *Score*: 10.0;

- WinRAR- *Score*: N\C;
- PDF *Creator*- *Score*: N\C;
- Java *Runtime*- Score: N\C; e
- Ubuntu_Linux:15.04- Score: 5.0.

Afigura 5.3 apresenta a forma gráfica da classificação dos *softwares* que contém alguma vulnerabilidade segundo o score do XML *Vulnerability Feeds*. Os nós chamados de *Desk* estão desdobrados em vários nós com os seus respectivos *softwares*.

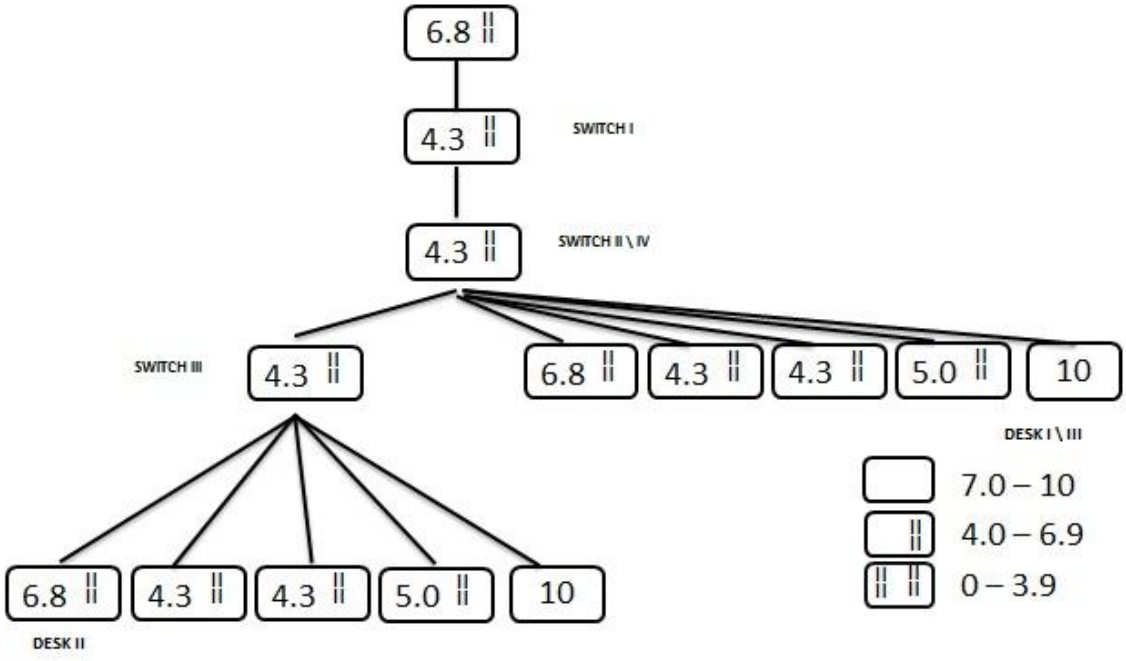

FIG 5.3 Representação gráfica da classificação das vulnerabilidades de acordo com o score XML *Vulnerability Feeds*

De acordo com a representação gráfica a rede apresenta aproximadamente 15% das vulnerabilidades classificadas como risco alto e as demais como de risco médio. Como já relatado nesta pesquisa uma classificação que é generalizada, que não considera o escopo da rede em produção.

Aplicando a métrica sugerida no capítulo 3 e o protótipo descrito no capítulo 4 resulta em uma nova classificação das vulnerabilidades, conforme figura 5.4,

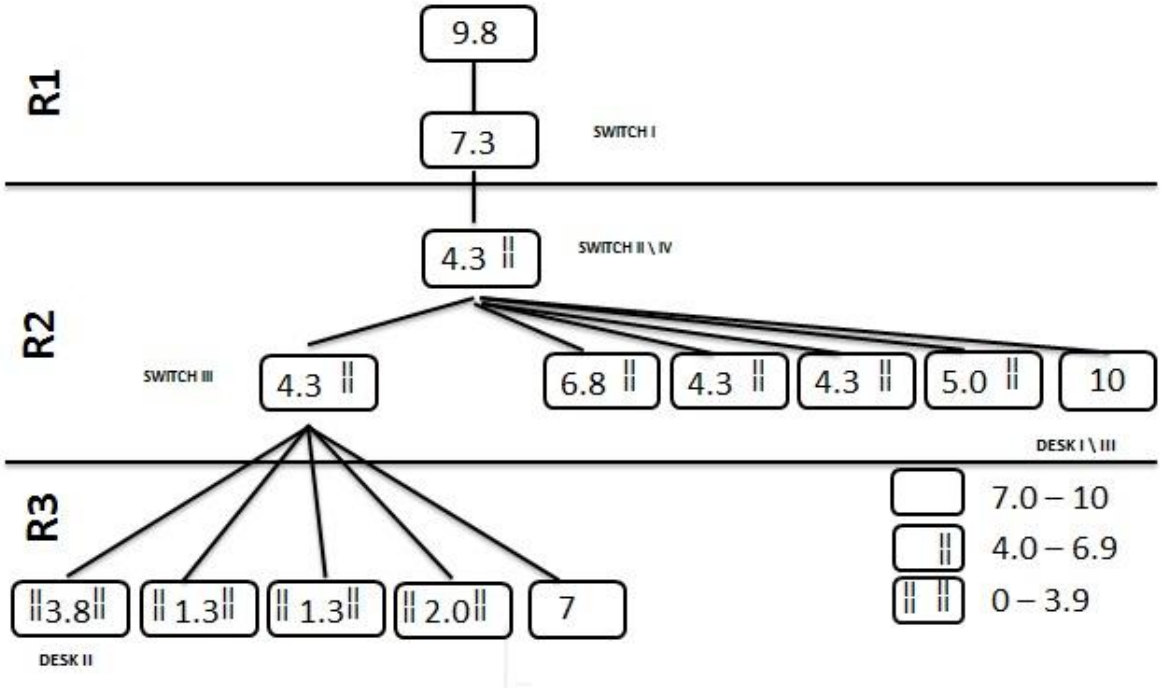

FIG 5.4 Representação gráfica da classificação das vulnerabilidades de acordo com a métrica sugerida

Na representação da classificação após a aplicação da métrica ocorreu a melhor distribuição das vulnerabilidades. Aproximadamente 28% são classificadas como de baixo risco, enquanto 42% risco médio e as restantes alto risco. Como resultado do conceito de perímetro aplicado na métrica pode-se observar ainda que as vulnerabilidades com riscos mais altos se encontram mais próximas da raiz principal, do ativo principal, assim orientando o usuário a reparar essas brechas aumentando o esforço e o tempo para o atacante obter sucesso.

5.2 REDE COM DIVERSOS ATIVOS PRINCIPAIS

O próximo cenário apresenta uma rede que com diversos ativos que podem ser considerados como principal, conforme a figura 5.5. Esta configuração é importante para demonstrar a flexibilidade apresentada pela métrica sugerida, uma mesma rede pode conter estruturas de árvores diferentes de acordo com o escopo observado.

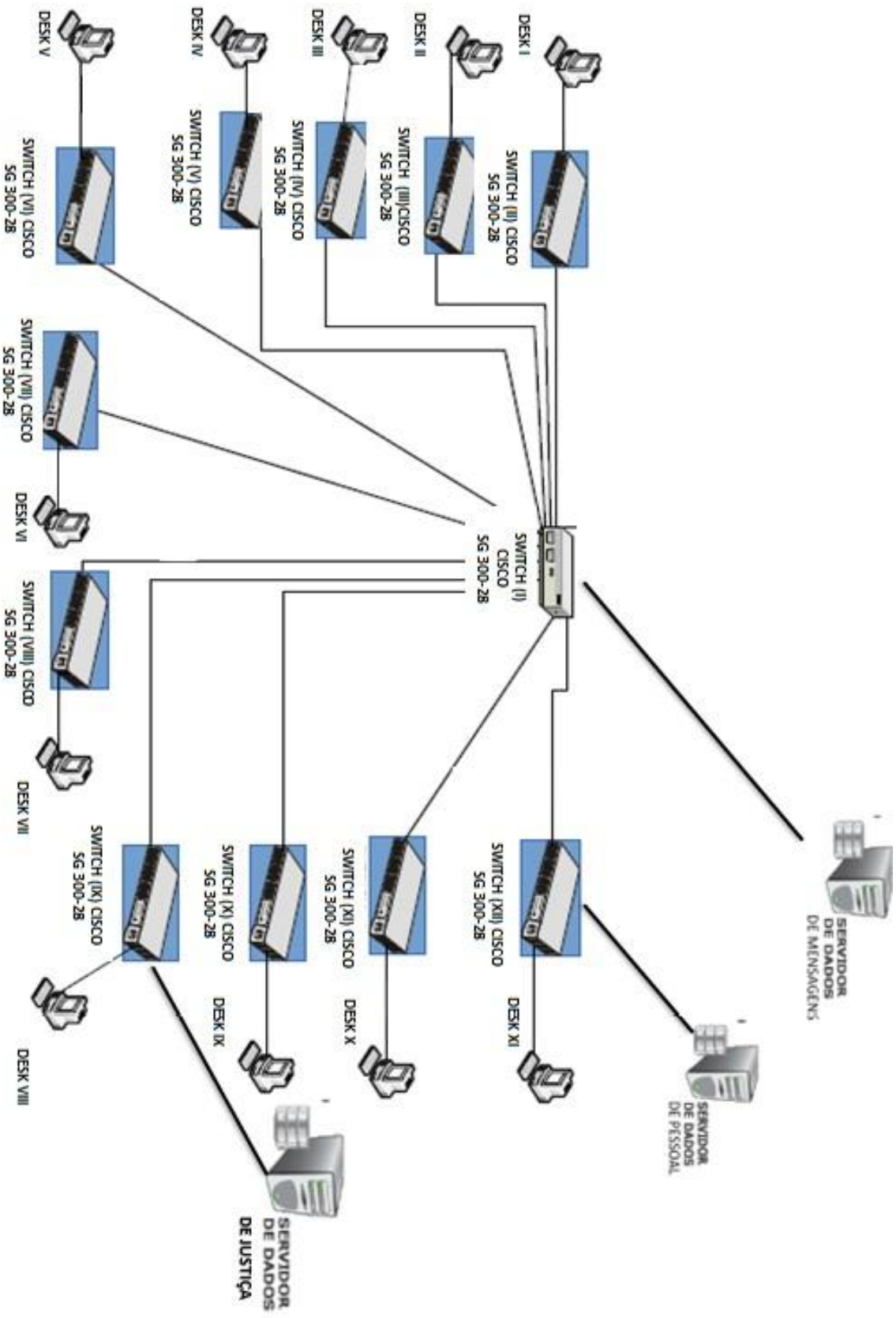

FIG 5.5 Rede em produção

As redes utilizadas nas organizações estudadas apresentam um mesmo padrão de configuração de *software*, logo os *desks*, *swiths* e servidor de mensagens seguem como descritos no item anterior. Os servidores de dados de pessoal e de justiça utilizam o oracle:mysql:5.5.43 com *score* 4.0 segundo o XML *Vulnerability Feeds*.

A figura 5.6 apresenta a estrutura da árvore equivalente considerando o servidor de dados de mensagem como ativo principal.

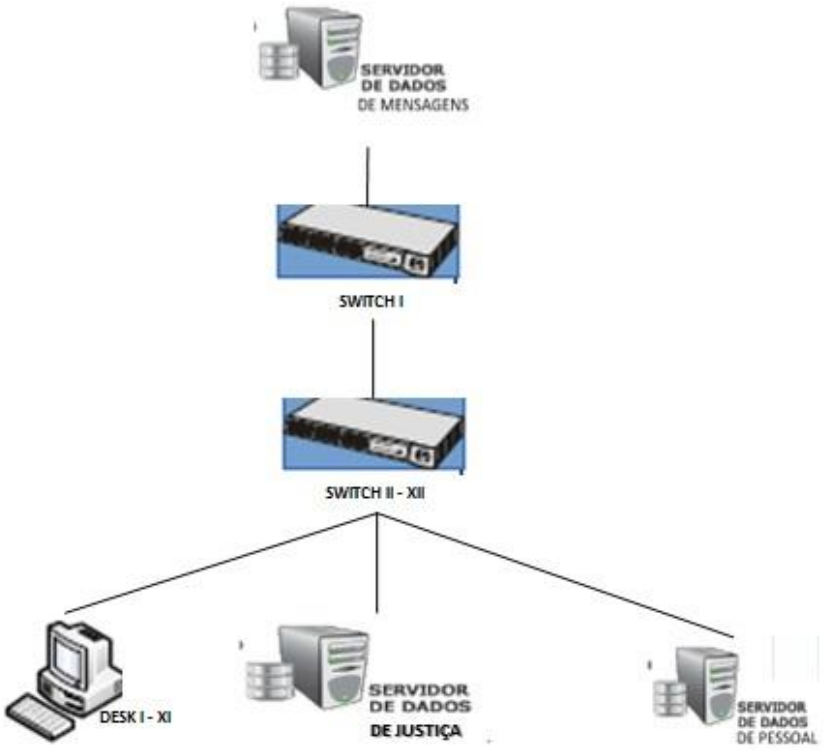

FIG 5.6 Rede estruturada com o servidor de dados de mensagens como ativo principal

Aplicando o algoritmo do capítulo 3 a complexidade da estrutura lógica da rede apresentada na figura 5.5 diminui consideravelmente, pois como a configuração dos *softwares* em cada equipamento se apresenta de forma padrão, ocorre um agrupamento das máquinas como descrito no passo 10 e 11 do algoritmo. O agrupamento em questão ocorre com os *switchs* II ao XII que aparecem no terceiro nível da árvore e em todos os *desk* do quarto nível.

Assim como apresentado anteriormente, a figura 5.7 apresenta a forma gráfica da classificação dos *softwares* que contém alguma vulnerabilidade segundo o score do XML *Vulnerability Feeds*.

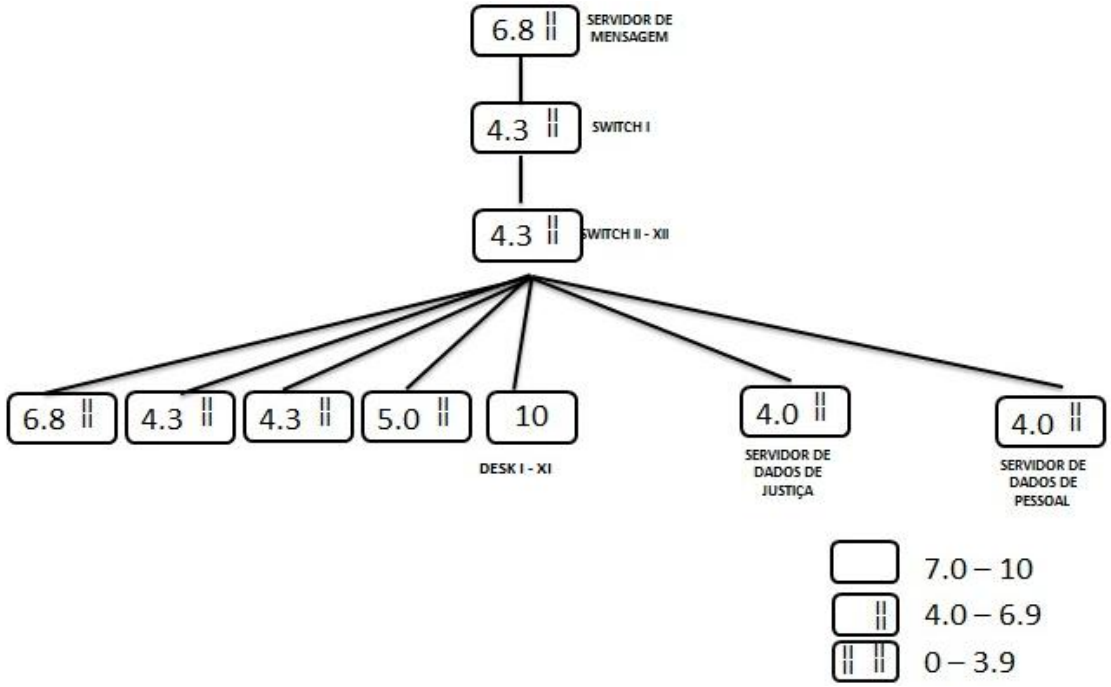

FIG 5.7 Representação gráfica da classificação das vulnerabilidades de acordo com o score XML *Vulnerability Feeds*

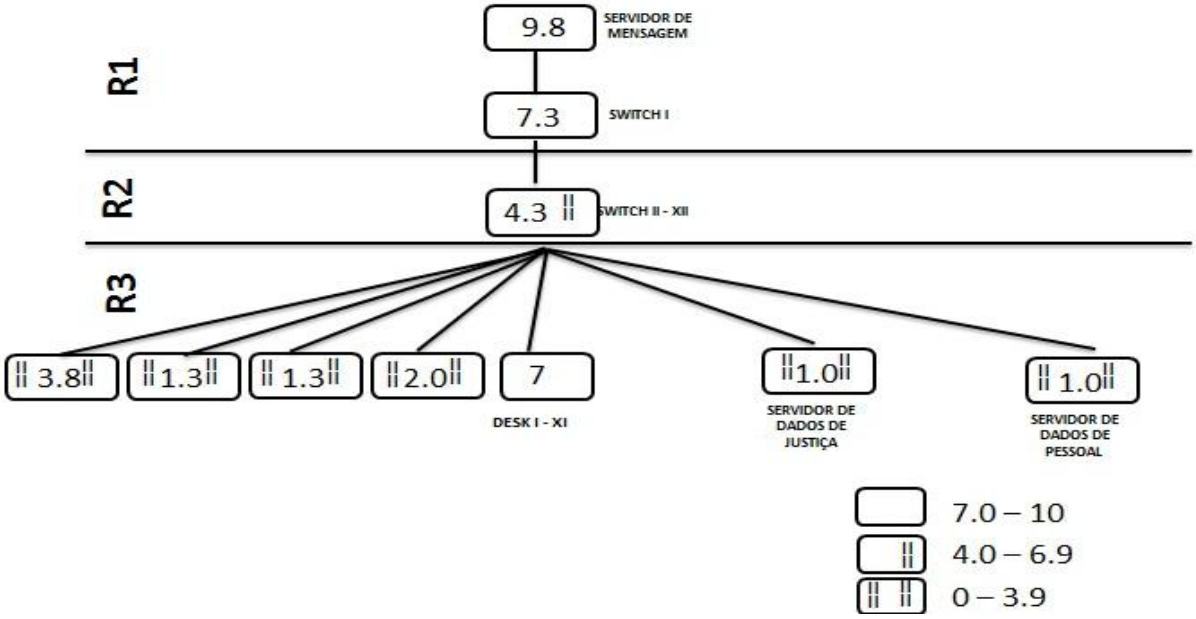

FIG 5.8 Representação gráfica da classificação das vulnerabilidades de acordo com a métrica sugerida

A figura 5.8 apresenta a nova classificação segundo a métrica sugerida na pesquisa. Analisando esta nova representação gráfica pode-se concluir que, diferentemente da classificação feita somente pelo XML *Vulnerability Feeds* que apresenta 90% risco médio, Existiu novamente uma melhor distribuição e assim uma melhor orientação ao usuário quanto a direcionar os esforços de segurança.

Nesta nova classificação se tem 30% de vulnerabilidade de risco alto, 10% de médio e 60% risco baixo. Assim o usuário aumentando a segurança nos *switchs* ele aumenta o esforço e o tempo que o atacante levaria para obter êxito.

Caso se faça o estudo considerando o servidor de dados de pessoal como ativo principal, se obtém a estrutura da árvore demonstrada na figura 5.9 assim como a classificação das vulnerabilidades encontradas segundo o XML *Vulnerability Feeds* e métrica sugerida nas figuras 5.10 e 5.11 respectivamente.

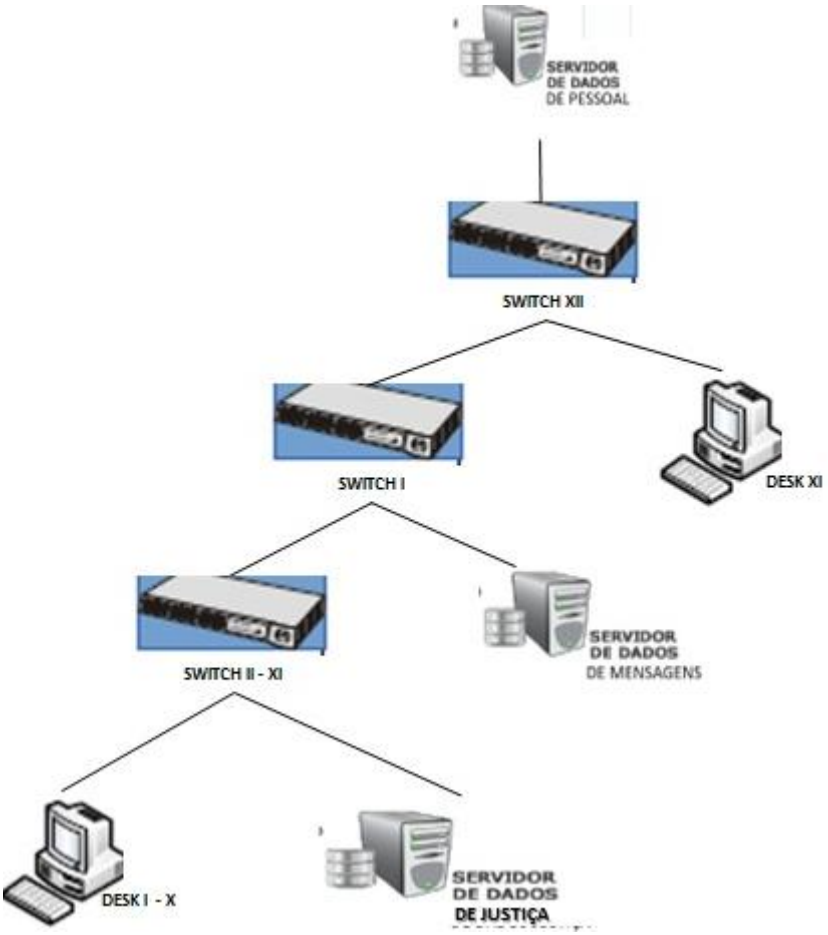

FIG 5.9 Rede estruturada com o servidor de dados de pessoal como ativo principal

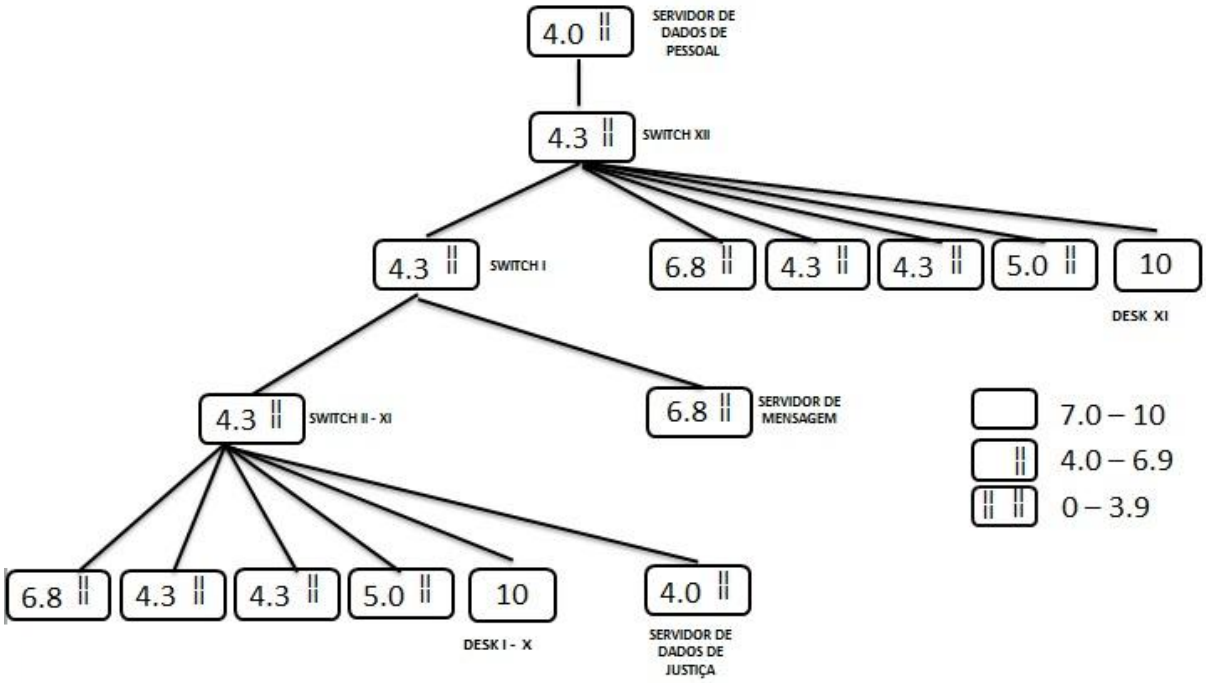

FIG 5.10 Representação gráfica da classificação das vulnerabilidades de acordo com o score XML *Vulnerability Feeds*

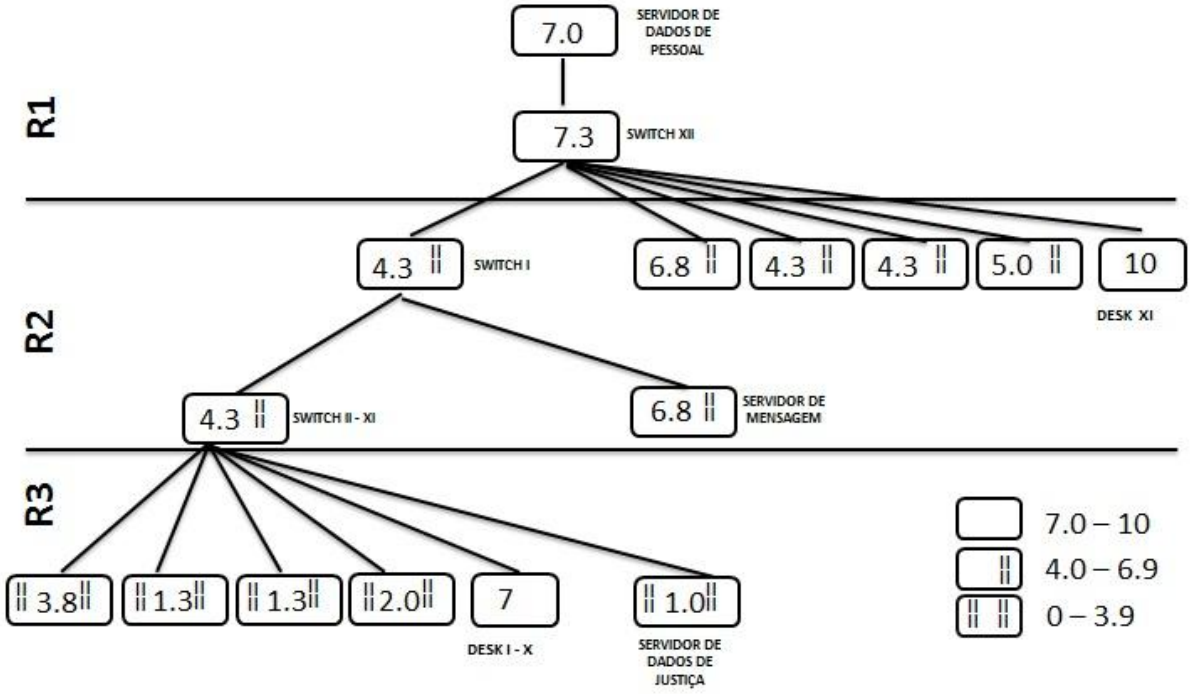

FIG 5.11Representação gráfica da classificação das vulnerabilidades de acordo com a métrica sugerida

Analisando o cenário apresentado, além das mesmas considerações já realizadas anteriormente sobre e a melhor distribuição da classificação da vulnerabilidade, foi observado que se têm dois grupos de *desk* distintos. Grupos que apresentam as mesmas vulnerabilidades, porém, com classificações diferentes. Assim, caso o usuário opte em realizar correções nestes grupos se teria uma prioridade a seguir, diminuindo o seu esforço e aumentando do atacante.

Talvez analisando de forma isolada, observando apenas a árvore, não tenha importância essas observações, porém, ao se verificar como foi estruturada a árvore, principalmente os nós *desk* I – X e *desk* XI, se conclui que a quantidade de máquinas que deveriam ser aplicadas as soluções se reduz bastante com a classificação diferenciada desses dois grupos.

O mesmo estudo pode ser realizado caso se tenha o servidor de dados de justiça como ativo principal, como na figura 5.12, 5.13 e 5.14.

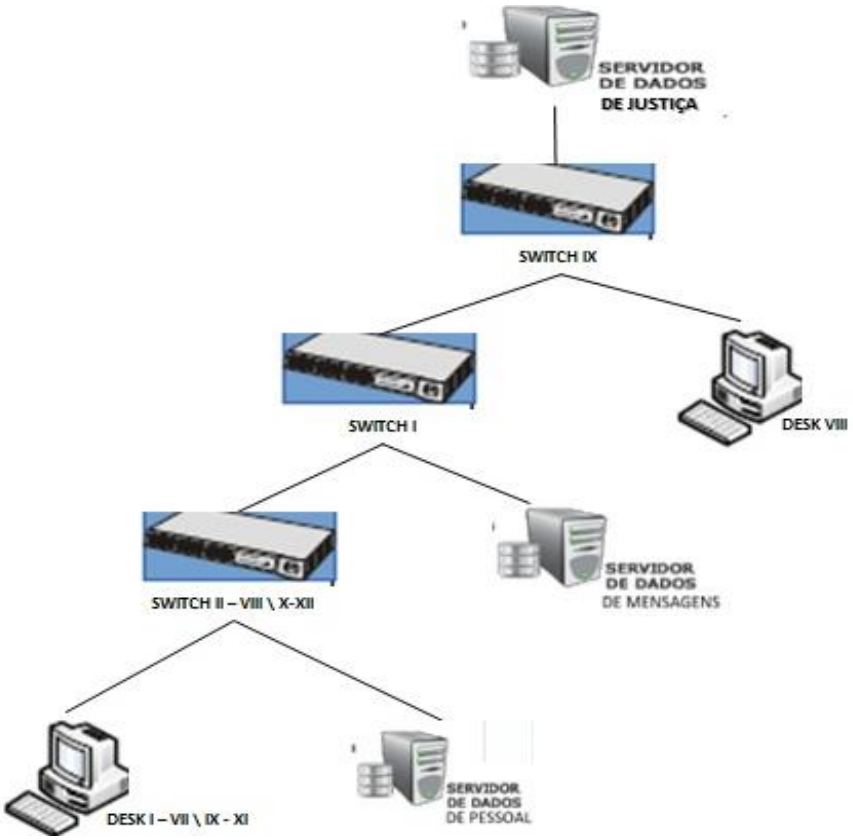

FIG 5.12 Rede estruturada com o servidor de dados de justiça como ativo principal

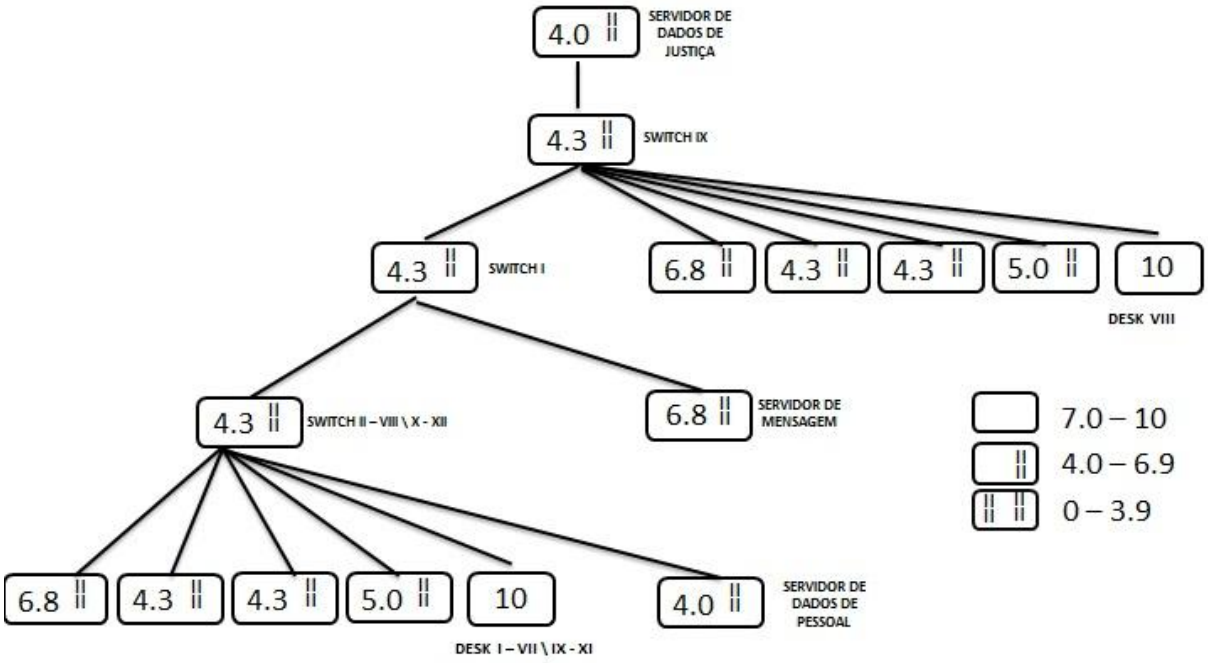

FIG 5.13 Representação gráfica da classificação das vulnerabilidades de acordo com o score XML *Vulnerability Feeds*

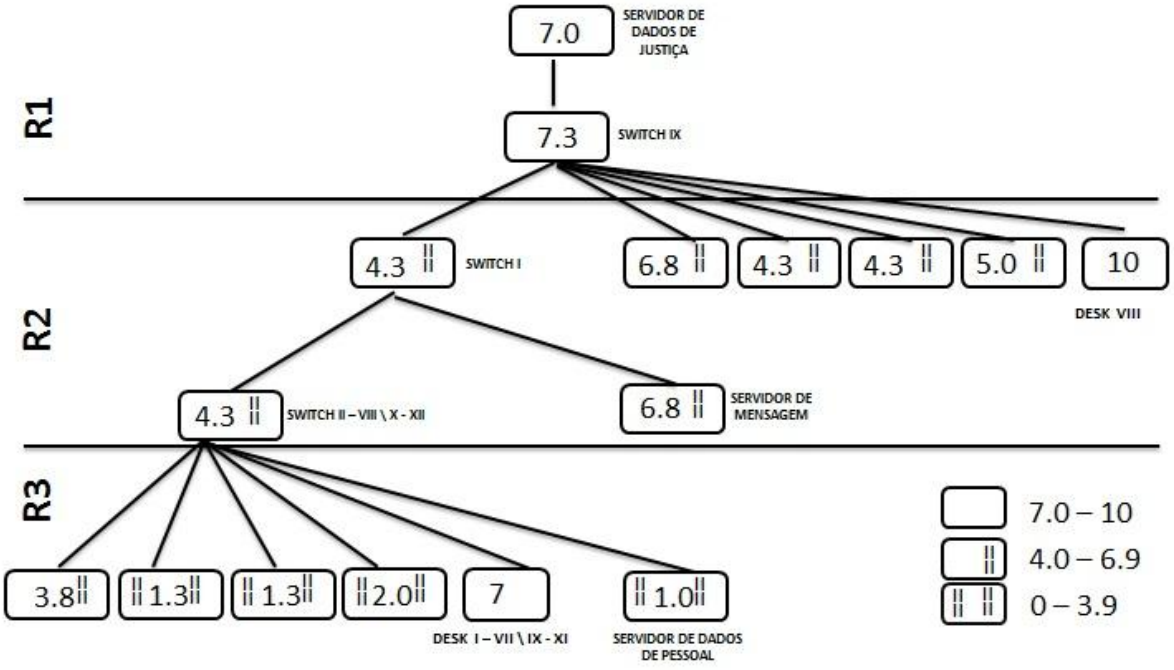

FIG 5.14Representação gráfica da classificação das vulnerabilidades de acordo com a métrica sugerida

Aparentemente as árvores apresentadas são idênticas ao caso anterior, porém analisando os detalhes se verificou que a classificação das vulnerabilidades se dá de forma diferente devido a disposição dos nós. Neste apresentado a prioridade deverá ser um switch e um grupo *desk* diferente do anterior. Com esta observação se chegou a uma conclusão que além dos pontos já levantados a métrica sugere um caminho diferente para cada ativo principal, assim considerando o que é realmente importante para o escopo apresentado.

6 TRABALHOS RELACIONADOS

Durante a pesquisa não se encontrou trabalho que descreva objetivo semelhante ao desejado, assim alguns foram observados com a finalidade de orientar qual repositório de vulnerabilidades mais utilizado na determinação de soluções para o problema de segurança.

6.1 PREVISÃO DE VULNERABILIDADE DE SOFTWARE

Zhang (ZHANG, 2011b) realizou um estudo empírico sobre técnicas de aplicação de mineração de dados se baseando no NVD, com o objetivo de prever o tempo para a próxima vulnerabilidade.

A base de dados do NVD foi utilizada a fim de tentar construir um modelo para a utilização de uma métrica sugerida na pesquisa, chamada de *time to next Vulnerability* (TTNV), onde o objetivo principal seria determinar o tempo em que a próxima vulnerabilidade será encontrada em uma determinada aplicação.

A solução considera as informações abaixo:

- Tempo: dia e mês da data da publicação da vulnerabilidade, a tag <*vuln:published-datetime*> do XML *Vulnerability Feeds*. Através deste campo o autor verificou a possibilidade da existência de uma repetição padrão, um mesmo intervalo de tempo para o surgimento de vulnerabilidades em uma determinada aplicação.
- Versão: Foi pesquisado se existe um intervalo padrão de versões para ocorrer vulnerabilidades em um específico aplicativo, utilizando dados extraídos da tag *cpe-lang:fact-ref* do XML *Vulnerability Feeds*.

6.2 PREVISÃO DE ERRO DE SOFTWARE

Em seu trabalho, Jason (WRIGHT, 2014), se preocupa com os erros na especificação, desenvolvimento ou configuração de *software* de tal forma que a sua execução produza um resultado incorreto ou um comportamento inesperado.

Para tentar evitar tais erros foram sugeridas métricas para medir a exposição relativa dos utilizadores finais e formar uma base para a escolha de qual produto usar, levando em consideração a quantificação de falhas encontradas.

- *Vulnerability Free Day* (VFD), métrica utilizada para medir a probabilidade da não existência de vulnerabilidade ativa em um determinado dia.
- *Median Active Vulnerability per Day* (MAV), o número médio de vulnerabilidades que são conhecidos para um fornecedor específico.

As métricas são baseadas em medições do número de relatórios de vulnerabilidades e a taxa de desenvolvimento de *patches* para produtos de *softwares* individuais. Medições relacionadas com o conceito de ciclo de vida de vulnerabilidade.

Nesta pesquisa foram utilizados os campos *product* e *published-datatime* do XML *Vulnerability Feeds*.

O objetivo principal do trabalho é medir a exposição relativa dos usuários finais a fim de formar uma base para a escolha de um produto de *software* em relação a outros.

6.3 MODELO PARA AGREGAR MÉTRICA DE VULNERABILIDADE USANDO O NVD

Em (ZHANG, 2011a), é desenvolvido um modelo quantitativo para agregar as métricas de vulnerabilidades em uma rede corporativa, medindo a probabilidade de ocorrência das violações e a possibilidade de ocorrer uma invasão para elevar os privilégios.

O trabalho é baseado em grafo de ataque e utilizou dados extraídos do XML Vulnerability Feeds assim como as métricas do CVSS, como o *Access Complexity*. Ainda se utilizou como base o algoritmo desenvolvido em (HOMER, 2009) para a construção das métricas de segurança.

Em resumo a construção do grafo de ataque se dá com analise das vulnerabilidades de todas as máquinas da rede e assim avaliadas todas as

possibilidades, ou todos os caminhos que um invasor pode utilizar para se aproveitar das brechas e comprometer o sistema.

Uma conclusão importante para a pesquisa e utilizada no momento de formular a métrica sugerida é a ideia da quantidade de *softwares* com vulnerabilidades. Se um *host* tem uma aplicação com dez vulnerabilidades, terá menor risco para a segurança do que dez aplicações diferentes, cada um com uma vulnerabilidade. Assim um atacante pode estar familiarizado com uma destas aplicações e utilizar esta brecha.

6.4 COMPARATIVOS DOS TRABALHOS

TAB. 6.1 Tabela Comparativa entre os Trabalhos Relacionados

TRABALHOS RELACIONADOS	Usa o repositório NVD	Objetivo
ZHANG, 2011b	Sim	Determinar o tempo em que a próxima vulnerabilidade será encontrada em uma determinada aplicação.
WRIGHT, 2014	Sim	Fornecer uma base para a escolha de um produto de software em relação a outros
ZHANG, 2011a	Sim	Calculo do risco acumulativo

A TAB. 6.1 apresenta os trabalhos pesquisados para o desenvolvimento desta dissertação. Pode-se observar que não apresentam objetivos semelhantes a esta pesquisa, porém foram utilizados para verificar o repositório mais confiável para servir como base ao trabalho apresentado.

O trabalho apresentado nessa dissertação propõe uma métrica que utiliza os dados do XML Vulnerability Feeds a fim de classificar as vulnerabilidades tendo

como base um ativo principal. Classificação esta que apoiaria o usuário quanto a decisão de qual brecha corrigir.

7 CONCLUSÃO

A principal dificuldade encontrada pelos usuários ao consultar um relatório com as vulnerabilidades é saber qual o risco caso estas sejam exploradas para a sua rede.

Este trabalho propôs uma métrica para classificar as vulnerabilidades considerando a topologia da rede, tendo como base um ativo principal para um escopo considerado. A solução tem como base os dados extraídos das fontes cedidas pelo portal NVD, onde classifica as vulnerabilidades de uma forma genérica.

O protótipo proposto realiza a consulta no XML encontrado no portal e aplica as métricas sugeridas levando em consideração o ativo principal escolhido pelo usuário e a aplicação do algoritmo para a transformação da rede em uma árvore n-área.

Analisando os cenários criados a métrica sugerida em conjunto com o algoritmo para se associar a rede a uma estrutura de árvore n-área se apresentou de forma satisfatória apresentando algumas conclusões:

- Redução da complexidade da topologia da rede para analise – Com a aplicação do algoritmo proposto no capítulo 3, pode-se obter uma estrutura menor devido ao agrupamento dos equipamentos que possuem configurações idênticas. Assim redes extremamente podem ser avaliadas com um menor esforço, caso observado no capítulo 5.
- Priorização das vulnerabilidades mais próximas ao ativo principal – Aplicando a métrica obtemos uma classificação que sugere ao usuário correções daquelas brechas mais próximas ao ativo considerado como principal. Esta priorização foi importante para priorizar grupos que por vezes tinham configurações semelhantes, mas distâncias diferentes em relação a raiz da árvore.
- Fidelidade ao escopo apresentado – Mesmo como estruturas semelhantes, após aplicar o algoritmo e métrica, pode-se observar que

o modelo sugere prioridades diferentes, caminhos diferentes a ser seguido pelo usuário para a correção e aumento da segurança da rede.

7.1 TRABALHOS FUTUROS

Os trabalhos futuros serão baseados no projeto de uma arquitetura completa para a classificação, automatizando a entrada dos dados da rede, de forma que o sistema faça a leitura de toda a rede em produção e estruture a árvore de acordo com o ativo principal indicado e o algoritmo sugerido neste trabalho.

Estas melhorias retiram parte do trabalho executado pelo usuário, deixando o sistema responsável por praticamente todas as fases descriminadas no capítulo 4 desta pesquisa.

Além dessas melhorias, o sistema poderá comparar uma rede com vários ativos principais de forma a identificar uma classificação única para que o usuário corrija e aumente a defesa de todos estes ativos simultaneamente.

8 REFERÊNCIAS BIBLIOGRÁFICAS

BALDISSERA, T.A. e NUNES, R.C.**Impacto na implementação da norma nbr iso/iec 17799 para a gestão de segurança da informação em colégios: um estudo de caso.** Encontro Nacional de Engenharia da Produção. USA, 2011.

CASANAS, Alex Delgado Gonçalves; MACHADO, César de Souza.**O impacto da implementação da norma nbr iso/iec 17799–código de prática para a gestão da segurança da informação-nas empresas.**UFSC. Santa Catarina, 2001.

DACIER, M.,DESWARTE Y. e KAÂNICHE, M.**Models and tools for quantitative assesment of operational secury.** 1996.

DE OLIVEIRAMarcia Cristina; VIEIRA, Alexandre Timm.**Quantificação devulnerabilidades em segurança da informação avaliando maturidade de pessoas.**Ulbra.RS, 2011.

HOMER, J., OU, X. e SCHMIDT, D.A sound and **practical approach to quantifyng security risk in enterprise networks.** Kansas States University Technical Report, pags. 1-15, 2009,

MAYNARD, D.C.S. **O caso WIKILEAKS: desafio ao historiador do tempo presente.** XXVI Simpósio Nacional de História - ANPUH. São Paulo, 2011.

OF INCIDENT RESPONSE, F. e TEAMS, S.**A complete guideto the common vul - nerability scoring system version 2.0.**https://www.first.org/cvss/v2/guide. Acessado em 02 de Agosto de 2014.

OF STANDARDS, N. I. e TECHNOLOGY.**Common vulnerabilitirs and exposures the standard for information security.**https://cve.mitre.org/about/index.html. Acessado em 26 de Julho de 2014

OF STANDARDS, N. I. e TECHNOLOGY.**Common vulnerabilitirs scoring system version 2 calculator.**https://nvd.nist.gov/cvss.cfm?calculator&version=2,b. Acessado em 15 de Agosto de 2014

OF STANDARDS, N. I. e TECHNOLOGY.**National vulnerability database.**https://nvd.nist.gov/download.cfm,c. Acessado em 23 de Agosto de 2014

OF STANDARDS, N. I. e TECHNOLOGY.**Nvd common vulnerability scoring system support v2.**https://nvd.nist.gov/CVSS.aspx,d. Acessado em 26 de Julho de 2014

OF STANDARDS, N. I. e TECHNOLOGY.**Nvd/cve xml achema file.**https://nvd.nist.gov/schema/nvdcve.xsd,e. Acessado em 31 de Agosto de 2014

OF STANDARDS, N. I. e TECHNOLOGY.**The security content automation protocol.**https://scap.nist.gov/,f. Acessado em 24 de Julho de 2014

OF STANDARDS, N. I. e TECHNOLOGY.**The unides states government configuration baseline.**https://usgcb.nist.gov,g. Acessado em 24 de Julho de 2014

REMCO R. BOUCKAERT, EIBE FRANK, M. H. R. K. P. R. A. S. D. S.**WEKA Manual for Version 3.7.** The University of Waikator,2010.

TRAINA, A.J.M.**SLIM-TREE:Método de Acesso Métrico Baseado em Medidas de Dispersão.** Tese de Doutorado, Universidade Federal de Uberlândia, 2005.

VULNERABILITIES, C. e EXPOSURES. **Cve-ids have a new format.**https://cve.mitre.org/cve/identifiers/syntaxchance.html. Acessado em 14 de Dezembro de 2014

WRIGHT, J.L.**Software Vulnerabilities: Lifespan, Metrics and Case Study.**Tese de Doutorado, University of Idaho, 2014.

XINMING OU, WAYNE F. BOYER, M.A.M. **A scalable approach to attack graph generation.** 3th ACM Conference on Computer and Communications Security, págs. 336-345, 2006.

XINMING OU, S.G. e APPEL,A. **Mulval: A logic-based network security analyzer.** 14th USENIX Security Symposium, 2005.

ZHANG, ANNOP SINGHAL, J.H.X.O.**An empirical study of a vulnerability metric aggregation method.**Kansas States University Technical Report, págs. 1-8, 2011a.

ZHANG, S., CARAGEA, D. e OU, X. **An empirical study on using the national vulnerability database to predict software vulnerabilities.**Database and Expert Systems Applications, págs.217-231, 2011b.

9 APÊNDICE

Este apêndice visa descrever as interações entre os usuários e o sistema, fornecendo uma narrativa sobre como o sistema é utilizado. A descrição será feita através de um cenário, conjunto de sequência de interação. O caso de uso ilustrado no capítulo 4 desta pesquisa apresenta um ator principal, que pede ao sistema para que execute a interação com o portal NVD e aplique a métrica sugerida no capítulo 3, a fim de resultar no vetor de classificação das vulnerabilidades de acordo com o escopo apresentado.

Sistemas

Cenário Principal:

1. O usuário abre o XML no diretório armazenado
2. O sistema busca as informações importantes pré-determinadas e salvas no SGDB
3. O usuário navega pela lista de softwares armazenados no SGBD e seleciona os contidos em sua rede
4. O sistema compara os softwares adicionados pelo usuário com os armazenados no SGBD
5. O sistema lista as vulnerabilidades encontradas na rede com o score NVD
6. O sistema aplica a métrica sugerida
7. O sistema gera o vetor de classificação de acordo com a métrica aplicada
8. O usuário navega pela lista de softwares armazenados no SGDB e realiza a atualização das alterações realizadas em sua rede
9. Fim do sistema

Extensões:

3a: Não existem softwares armazenados no SGDB que pertença a rede

1: Volta ao passo 9 do cenário principal

9a: O usuário gera outro vetor de classificação

1: Volta ao passo 4 do cenário principal