CENTRO UNIVERSITÁRIO UNIVATES
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE ENGENHARIA DA COMPUTAÇÃO

SEGURANÇA DE TI NA PREFEITURA MUNICIPAL DE LAJEADO: A CAMADA DE ENLACE DE DADOS DA REDE LOCAL

Cristiano André Lenz

Lajeado, junho de 2017

Cristiano André Lenz

SEGURANÇA DE TI NA PREFEITURA MUNICIPAL DE LAJEADO: A CAMADA DE ENLACE DE DADOS DA REDE LOCAL

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Tecnológicas do Centro Universitário UNIVATES, como parte dos requisitos para a obtenção do título de bacharel em Engenharia da Computação.

Orientador: Luis Antônio Schneiders

Lajeado, junho de 2017

RESUMO

O desenvolvimento e abrangência das redes de computadores atualmente refletem na imprescindibilidade da segurança destas para evitar as mais variadas formas de ameaças e ataques que podem causar grandes prejuízos nas organizações. A presente proposta consiste na avaliação das ameaças associadas às redes de acesso local, considerando a camada de enlace de dados do modelo de referência RM-OSI/ISO. Com base nesta avaliação inicial, são produzidos ataques na infraestrutura de rede da Prefeitura Municipal de Lajeado, visando a identificação das principais vulnerabilidades de cada ameaça e também as possíveis alternativas de mitigação. Para esse propósito são considerados ataques de falsificação de endereço MAC (*MAC spoofing*), *MAC address table overflow*, ataques ao serviço DHCP (dentre eles o *DHCP starvation* e o *rogue DHCP server*), ataque ao protocolo ARP (*ARP spoofing*), ataque ao *spanning tree*, tempestade de *broadcast e* ataque de *VLAN hopping* utilizando *switch spoofing* e *double tagging*. São propostas e aplicadas soluções eficazes de contramedida para cada ataque realizado com sucesso. Por fim, é realizada uma análise de eficiência das contramedidas, apresentando os resultados dessa implementação e comprovando que ocorreu um aumento na segurança da rede.

Palavras-chaves: Segurança em redes. Redes de computadores. Camada de enlace de dados. Ameaças. Ataques.

ABSTRACT

The development and coverage of the currently computer networks reflects the indispensability of these security to prevent the several forms of threats and attacks which can cause huge losses in organizations. The present proposal consists of the evaluation of the threats associated to local access networks, considering the data link layer of the RM-OSI/ISO reference model. Based on this initial evaluation, attacks are generated on the Lajeado City Hall network infrastructure, aiming to identify the main vulnerabilities of each threat and also possible mitigation alternatives. For this purpose are considered MAC spoofing attacks, MAC address table overflow, DHCP service attacks (including DHCP starvation and rogue DHCP server), attack on the ARP protocol (ARP spoofing), spanning tree attack, broadcast storm and VLAN hopping attack using switch spoofing and double tagging. Effective countermeasure solutions are proposed and applied for each successful attack. Finally, an analysis of the efficiency of the countermeasures is carried out, presenting the results of this implementation and proving that there has been an increase in the security of the network.

Keywords: Network security. Computer networks. Data link layer. Threats. Attacks.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACL	*Access Control List*
ARP	*Address Resolution Protocol*
BPDU	*Bridge Protocol Data Unit*
CAIS	Centro de Atendimento a Incidentes de Segurança
CAM	*Content Addressable Memory*
CERT.br	Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil
CETIC.br	Centro de Estudos sobre as Tecnologias da Informação e da Comunicação
DAI	*Dynamic ARP Inspection*
DHCP	*Dynamic Host Configuration Protocol*
DMZ	*DeMilitarized Zone*
DoS	*Denial of Service*
FDB	*Forwarding Database*
ICMP	*Internet Control Message Protocol*
ID	Número Identificador
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IP	*Internet Protocol*
IPv4	*Internet Protocol version 4*
IPv6	*Internet Protocol version 6*
ISO	*International Organization for Standardization*
LAN	*Local Area Network*
MAC	*Media Access Control*
MAN	*Metropolitan Area Network*
NIC.br	Núcleo de Informação e Coordenação do Ponto BR
OSI	*Open Systems Interconnection*

PC	*Personal Computer*
PPS	*Packets Per Second*
RM-OSI/ISO	*Reference Model - Open Systems Interconnection / International Organization for Standardization*
RSTP	*Rapid Spanning Tree Protocol*
STP	*Spanning Tree Protocol*
TCP	*Transmission Control Protocol*
TCP/IP	*Transmission Control Protocol / Internet Protocol*
UDP	*User Datagram Protocol*
VLAN	*Virtual Local Area Network*
WAN	*Wide Area Network*

SUMÁRIO

1 INTRODUÇÃO

Conforme é exposto por Comer (2007), as redes de computadores têm crescido explosivamente e a comunicação via computador transformou-se em uma parte essencial da infraestrutura de todos. Portanto, conforme reportado por Tanenbaum e Wetherall (2011), como milhões de cidadãos comuns usam as redes para os mais variados fins, surge um ponto fraco atrás de outro, tornando a segurança um problema de grandes proporções e, segundo Júnior, Suavé, Moura e Teixeira (1999), um dos aspectos mais importantes dos ambientes de serviços em redes.

É corroborado por Tanenbaum e Wetherall (2011) que, para que uma rede torne-se mais segura, com frequência é necessário lidar com adversários inteligentes, dedicados, e às vezes, muito bem subsidiados, além do fato de que, segundo Nakamura e Geus (2007), quanto maior a evolução nos avanços tecnológicos, maiores são as vulnerabilidades que aparecem e que devem ser tratadas com a sua devida atenção.

Isto leva Campbell (1997) a afirmar que a segurança em uma rede de computadores é uma necessidade que, conforme exposto por Nakamura e Geus (2007), vem transcendendo o limite da produtividade e da funcionalidade pois, enquanto a velocidade e a eficiência em todos os processos de negócios significam uma vantagem competitiva, a falta de segurança pode resultar em grandes prejuízos e na falta de novas oportunidades de negócios. Portanto, a segurança de uma estrutura de rede é essencial para o bom andamento das organizações, fazendo com que elas precisem ser protegidas.

No que tange essa segurança, Nakamura e Geus (2007) exemplificam que, de tempos em tempos, os noticiários são compostos por alguns crimes “da moda”, que vêm e vão. Como

resposta, o policiamento é incrementado, o que resulta na inibição daquele tipo de delito. Os criminosos passam então a praticar um novo tipo de crime, que acaba virando notícia, de tal forma que faz com que o ciclo continue. Não diferente, esse mesmo comportamento pode ser observado no mundo da informação, de modo que também se deve ter em mente que a segurança deve ser contínua e evolutiva, já que o "arsenal" de defesa usado pela organização pode funcionar contra determinados tipos de ataques, porém, ser falho contra novas técnicas desenvolvidas para driblar esse "arsenal" de defesa.

É observado por Nakamura e Geus (2007) que o crescimento da importância e até mesmo da dependência do papel da tecnologia nos negócios, somado ao aumento da facilidade de acesso e ao avanço das técnicas usadas para ataques e fraudes eletrônicos, resultam no aumento do número de incidentes de segurança. Porém, é interessantemente também demonstrado por Nakamura e Geus (2007) que os ataques que vêm causando os maiores problemas para as organizações são aqueles que acontecem justamente a partir de sua própria rede, ou seja, os ataques internos. Este fato faz com que a complexidade da segurança na rede aumente, pois a proteção deve ocorrer não somente contra os ataques vindos de rede externa, mas também contra aqueles que podem ser considerados internos. Em 2007 foi apontado por Nakamura e Geus (2007) que, apesar de as pesquisas mostrarem que o número de ataques partindo da *Internet* era maior do que os ataques internos, os maiores prejuízos são causados por esses.

Quanto aos incidentes internos, Ribeiro (2006) cita que existe pouca preocupação no que diz respeito aos recursos de segurança dos equipamentos de conectividade das redes locais. Esta situação é uma das principais causas desse problema, pois faz com que, não só a quantidade de ataques provenientes da rede local seja maior, como também seja maior o seu poder de destruição, já que o atacante muitas vezes vai dispor de informações privilegiadas que um invasor externo normalmente não possui.

Portanto, neste trabalho é proposto a realização de um projeto de segurança em rede de computadores que abranja a camada de enlace de dados, também conhecida como ligação de dados, *link* de dados ou nível 2, do modelo de referência *Reference Model - Open Systems Interconnection / International Organization for Standardization* (RM-OSI/ISO), também conhecido apenas como modelo *Open Systems Interconnection* (OSI). Durante o trabalho é abordada esta camada estudada, elencando as principais ameaças e ataques que utilizam as vulnerabilidades de segurança dos equipamentos no nível de enlace de dados para prejudicar

uma rede de computadores. Posteriormente estas ameaças são executadas com sucesso em um ambiente de testes vulnerável. São buscadas soluções de contramedida para evitar ou mitigar cada um destes ataques e ameaças. E finalmente, utilizando o ambiente real de rede de computadores da Prefeitura Municipal de Lajeado, são aplicadas as defesas encontradas e então é realizada uma análise de eficiência delas, buscando os resultados dessa implementação e verificando se ocorreu um aumento na segurança da rede.

1.1 Tema

O tema do trabalho consiste na análise das principais ameaças e ataques à segurança de rede de computadores existentes na camada de enlace de dados (nível 2) do modelo de referência RM-OSI/ISO, buscando defesas para cada uma destas.

1.2 Problema de pesquisa

Quais são as ameaças e os ataques existentes na camada de enlace (nível 2) do modelo de referência RM-OSI/ISO em uma rede local e como mitigar estes no ambiente operacional da sede administrativa da Prefeitura Municipal de Lajeado, para se obter um ambiente de rede de computadores mais seguro?

1.3 Hipótese

Acredita-se que as principais ameaças, vulnerabilidades e ataques existentes na camada de enlace de dados (nível 2) do modelo de referência RM-OSI/ISO podem ser definidas e compreendidas. Devido a rede local estudada possuir como nó central um *switch* gerenciável, também supõem-se que é possível mitigar, ao menos parcialmente, as principais ameaças encontradas. Desta forma, presume-se que é possível alcançar um ambiente de rede local mais seguro na sede administrativa da Prefeitura Municipal de Lajeado através deste trabalho.

1.4 Objetivo geral

Objetiva-se propor e implementar um ambiente de redes de computadores mais seguro em relação a ameaças da camada de enlace de dados (nível 2) do modelo de referência RM-OSI/ISO, mitigando ao máximo o risco de concretização de uma ameaça a partir da exploração das vulnerabilidades inerentes aos elementos, protocolos, serviços e *hardware* desta camada.

1.5 Objetivos específicos

Como objetivos específicos do trabalho podem ser citados:

a) Pesquisar bibliografias que abordem este assunto, a fim de ressaltar a importância deste trabalho na área de segurança em redes de computadores.

b) Identificar as principais ameaças e ataques existentes na camada de enlace de dados (nível 2) de uma rede de computadores.

c) Buscar soluções efetivas para eliminar ou mitigar os riscos provenientes destas ameaças e vulnerabilidades, reduzindo ou eliminando o sucesso em ataques à rede de acesso local.

d) Analisar o resultado das ações de segurança realizadas na rede de computadores da Prefeitura Municipal de Lajeado, apresentando os resultados obtidos no processo.

1.6 Justificativa e relevância do trabalho

Os dados estatísticos levantados pelo Centro de Estudos sobre as Tecnologias da Informação e da Comunicação (CETIC.br) (2010) entre 2006 e 2010 mostram que, quanto maior for a empresa (ou maior for o cenário de funcionários conectados à rede nesta empresa), maior é a chance de ocorrer um acesso interno não autorizado e menor é o porcentual de acesso externo não autorizado. Porém, estas categorias de problemas de segurança encontrados em empresas, não foram mais listadas nas estatísticas do CETIC.br a partir de 2010, o que expõe uma carência de dados quanto aos tipos de ataques mais empregados atualmente.

Conforme dados fornecidos pelo Centro de Atendimento a Incidentes de Segurança (CAIS) (2016), os incidentes de segurança reportados, de forma geral, vêm aumentando em grande quantidade nos últimos anos. Entretanto, apesar do Núcleo de Informação e Coordenação do Ponto BR (NIC.br) (2016) destacar, com base nos dados do Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil (CERT.br) (2016), que os incidentes de segurança reportados em 2015 diminuíram em aproximadamente 31% em relação a 2014, neste mesmo intervalo de período houve um crescimento em determinadas modalidades de ataques.

De maneira geral, o CERT.br (2016) cita que os números de cada categoria de ataque relacionados também atestam que há uma redução nos relatos de tentativas com sucesso de ataques tradicionais, que tem como propósito a infiltração em uma organização passando por "brechas no *firewall*". Em contrapartida houve um crescimento, de 2014 a 2015, de 47% nas notificações de ataques que se encaixam na categoria de "outros", o que demonstra que a forma de execução de ataques está migrando para formas alternativas, onde a segurança em redes é mais negligenciada.

Porém estas entidades referidas, que mensuram e tratam dos incidentes de segurança existentes no Brasil, não oferecem dados de incidentes e ataques relacionados especificamente ao nível 2 (camada de enlace de dados) de uma rede de computadores ou de ataques recentes que sejam oriundos da rede local interna. Este fator, somado aos dados já mencionados anteriormente, acabam por demonstrar a relevância da proposta deste trabalho.

A implementação deste trabalho se justifica, pois, considerando os vários projetos de segurança em redes de computadores, eles se concentram em tratar de outros aspectos da área de segurança em redes em sua maioria, como os projetos que abordam especificamente a segurança em redes *wireless* (sem fio), como por exemplo os de Neto (2004), Andrade, Soares, Coutinho e Abelém (2004), Duarte (2003), Peres e Weber (2003), Gimenes (2005), Amaral e Maestrelli (2004), Bof (2010), Schweitzer, Sakuragui, Carvalho e Venturini (2005), Verissimo (2002), ou Silva e Souza (2003).

Além destes, um grande número de trabalhos que abordam a segurança de redes trata especificamente de *firewall*, podendo ser citados os artigos de Hunt (1998), Ioannidis, Keromytis, Bellovin e Smith (2000), Al-Shaer, Hamed, Boutaba e Hasan (2005), Al-Shaer e Hamed (2004), Lyu e Lau (2000), Wool (2004), Fernandes e Zanona (2010), Alécio e Pereira (2014), Sampaio (2011), Baqui (2012) e Gheorghe (2006).

Quanto ao objetivo específico da investigação deste trabalho, cita-se apenas o trabalho de Ribeiro (2006), que descreve as vulnerabilidades dos protocolos e equipamentos da camada 2, citando os ataques clássicos nesta camada, porém sem apresentar defesas específicas para cada um dos ataques citados, tampouco trazendo a implementação destas em um ambiente operacional.

Considerando estes fatores, os elementos apresentados nesta seção justificam e apontam a necessidade de novos estudos quanto aos problemas de segurança da camada de enlace de dados (nível 2) do modelo de referência RM-OSI/ISO, bem como aos métodos de prevenção.

1.7 Delimitação do trabalho

São abordadas ameaças, ataques, vulnerabilidades e defesas de segurança somente relacionados ao nível 2 do modelo de referência RM-OSI/ISO (camada de enlace de dados) de uma rede local, consequentemente limitando-se aos equipamentos que sejam relevantes apenas a este nível. Também, para implementação deste trabalho é considerada apenas a infraestrutura de rede da sede administrativa da Prefeitura Municipal de Lajeado e o seu ponto central, composto pelo *switch core*.

1.8 Estrutura do trabalho

O presente trabalho é composto por sete capítulos, conforme descritos a seguir:

O primeiro capítulo aborda a apresentação do trabalho em linhas gerais, introduzindo o mesmo, relatando o seu tema, problema de pesquisa, hipótese, objetivos gerais e específicos. Também é apresentada a justificativa que destaca a relevância do estudo para o campo científico.

No segundo capítulo é realizada uma revisão bibliográfica sobre os fundamentos da segurança em redes de computadores, abordando as características que lhe dizem respeito, os termos que fazem parte dela e o cenário nacional de incidentes. Após é fundamentado o conceito teórico geral do que é uma rede de computadores e os principais aspectos que a compõe, além de demonstrar o modelo de referência em camadas. Esta revisão também

aborda as principais ameaças existentes relacionadas a camada de enlace de dados e os métodos de prevenção possíveis para mitigar cada uma destas ameaças.

O capítulo três explica a metodologia de pesquisa empregada na elaboração do trabalho, classificando-a de acordo com cada aspecto do projeto proposto. Também são apresentados os equipamentos e as ferramentas utilizados no trabalho.

O quarto capítulo é utilizado para descrever o atual cenário operacional da Prefeitura Municipal de Lajeado em que é implementado o projeto.

No capítulo cinco são relacionadas as implementações das técnicas de ataque e defesa para cada uma das ameaças, e depois é analisado o resultado dessas implementações.

No sexto capítulo são apresentadas as conclusões e considerações finais deste trabalho, relatando os resultados obtidos através da execução deste.

O capítulo sete se destina a relacionar as referências bibliográficas utilizadas na elaboração deste documento.

2 REVISÃO BIBLIOGRÁFICA

Nesta seção são abordados os fundamentos da segurança em Tecnologia da Informação, das redes de computadores e os ataques mais comuns, associados às redes locais, considerando a camada de enlace de dados do modelo OSI, além dos métodos de prevenção ou mitigação relacionados.

2.1 Segurança em redes de computadores

De acordo com Soares, Lemos e Colcher (1995), o termo segurança é utilizado com o significado de minimizar a vulnerabilidade de bens (qualquer coisa de valor) e recursos, e, sendo assim, a segurança acaba sendo relacionada à necessidade de proteção contra o acesso ou a manipulação, intencional ou não, de informações confidenciais por elementos não autorizados, e a utilização não autorizada do computador ou de seus dispositivos periféricos. Além disso, Nakamura e Geus (2007) informam que seguir a concepção de que a segurança e o ambiente cooperativo devem andar juntos trará, além de bons negócios, grandes benefícios à economia global e também a garantia de sobrevivência.

Segundo Tanenbaum e Wetherall (2011) a segurança é um assunto abrangente e contém inúmeros tipos de problemas. Já Comer (1998) destaca que a segurança em um ambiente de interligação em redes é, ao mesmo tempo, importante e difícil: importante devido a informação ter importância significativa e difícil porque requer um entendimento da ocasião e do modo como os participantes podem confiar um no outros, assim como uma compreensão dos detalhes técnicos do *hardware* da rede e dos protocolos.

Ainda se cita que a segurança é marcada pela evolução contínua, em concordância com Nakamura e Geus (2007), cujos novos ataques têm como resposta novas formas de proteção, que levam ao desenvolvimento de novas técnicas de ataques, de tal forma que se cria um ciclo. A segurança também deve ser contínua e evolutiva, já que o "arsenal" de defesa usado em uma organização pode funcionar contra determinados ataques, mas ser falho contra novas técnicas desenvolvidas para driblar esse "arsenal" de defesa.

Comer (2007) ressalta que as redes não podem ser simplesmente classificadas como seguras ou não seguras, pois o termo não é absoluto, cabendo a cada organização definir o nível de acesso que é permitido ou negado. Isto se enfatiza pela afirmação de Ramos (2006) de que não existe segurança absoluta, já que, por mais medidas e precauções que sejam tomadas, jamais será possível endereçar todas as possíveis situações de prejuízo. Assim, Nakamura e Geus (2007) estipulam que afirmar que uma organização está 100% segura é, na realidade, um grande erro. Apesar disto, Comer (1998) destaca que, embora uma organização não possa impedir totalmente um crime, medidas básicas de segurança podem desencorajar ele, tornando-o significativamente mais difícil de ser impetrado.

Conforme mencionado por Tanenbaum e Wetherall (2011), em sua forma mais simples, a segurança preocupa-se em impedir que pessoas mal intencionadas leiam ou, pior ainda, modifiquem secretamente mensagens enviadas a outros destinatários. Também ela possui a preocupação que as pessoas tentem ter acesso a serviços remotos que não estão autorizadas a usar. Além disso, a segurança lida com meios para saber se uma mensagem supostamente verdadeira é falsa. Finalmente, a segurança ainda trata de situações em que mensagens legítimas são capturadas e reproduzidas, além de lidar com pessoas que tentam negar o fato de ter enviado certas mensagens. Dessa maneira, Comer (1998) clarifica que segurança implica inclusive garantia da integridade de dados e proteção contra acesso não autorizado dos recursos do computador, contra pessoas investigando ou enviando mensagens falsas na rede e contra a interrupção de serviços.

Assim, de acordo com Tanenbaum e Wetherall (2011), tornar uma rede segura envolve muito mais que apenas mantê-la livre de erros de programação, sendo necessário lidar com adversários inteligentes, dedicados e, às vezes, muito bem subsidiados, considerando o fato que a maior parte dos problemas de segurança é causada de forma proposital por pessoas mal intencionadas, que procuram obter algum benefício, chamar atenção ou prejudicar alguém. Isto converge para fomentar a condição de que, em conformidade com Tanenbaum e

Wetherall (2011), para manter uma rede segura, é necessário ter em mente que as medidas utilizadas para interromper a atividade de adversários eventuais terão pouco impacto sobre os adversários “mais espertos”. Concomitantemente, Comer (1998) traz à tona que a ideia de que um sistema de segurança só é tão fraco quanto seu ponto mais fraco é bastante conhecida e, conforme explicado por Ramos (2006) e por Nakamura e Geus (2007), não é por acaso que é no elo mais fraco da corrente que os ataques acontecem, ou ainda, citando diretamente, Ramos (2006, p. 24) declara que “a força de uma corrente é igual à força de seu elo mais fraco”.

2.1.1 Segurança física, lógica e operacional

A fim de classificar a segurança, Comer (1998) introduz que fornecer segurança de informações requer proteção não apenas dos recursos físicos, como também dos recursos abstratos, o que entra em sincronia com o fato que a segurança pode ser dividida entre física e lógica.

No que tange a segurança física, Soares, Lemos e Colcher (1995) explicam que medidas que garantem a integridade física dos recursos de um sistema são indispensáveis para assegurar a segurança de um sistema como um todo. Já Gil (2000) expõe que a segurança física corresponde à constatação de bom estado operacional dos recursos materiais (instalações, *hardware*, suprimentos) que compõem e dão sustentação aos sistemas de informações computadorizadas, enquanto Comer (1998) nota que, desta forma, a segurança física estende-se a cabos, pontes e roteadores que constituem a infraestrutura de uma rede. Portanto, conforme proposto por Comer (1998), embora a segurança física seja raramente mencionada, ela desempenha, na maioria das vezes, um papel preponderante no plano de segurança geral.

No que se refere a segurança lógica, Gil (2000) orienta que ela diz respeito a alterações, modificações ou erros dos recursos tecnológicos (processos e resultados) componentes de certo sistema de informação computadorizado, sendo assim, de acordo com Campbell (1997), uma defesa em nível de *software*. Fazendo uma analogia entre ambas, Comer (1998) explica que, enquanto a segurança física geralmente classifica pessoas e recursos em categorias amplas, a segurança lógica geralmente precisa ser mais restritiva.

Alguns autores ainda trazem um terceiro tipo de segurança, a segurança operacional. Já que, conforme citado por Behringer (2016), os problemas operacionais são uma diferente categoria de segurança. Segundo Rigo e Oliveira (2010) a segurança operacional, ou humana, visa estipular normas de utilização a serem obedecidas pelas pessoas envolvidas em todo o processo tecnológico corporativo, sejam elas usuários leigos ou profissionais de TI. Complementando, para Behringer (2016) existem dois tipos de problemas de segurança operacional: o primeiro é o de erros de configuração acidentais, onde as falhas cometidas são acidentais por natureza, sendo, de longe, o tipo mais frequente de problemas operacionais. O segundo tipo de problemas de segurança operacional são os erros de configuração deliberados, sendo estes, portanto, deliberados por natureza, mas variando em seu grau de malignidade, sendo estes mais propensos a resultar em uma violação, uma vez que existe a intenção maliciosa com o objetivo de quebrar a política de segurança.

2.1.2 Política de segurança

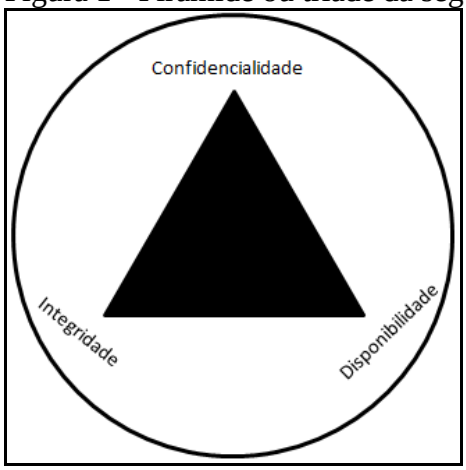
Quanto a necessidade de proteção, Soares, Lemos e Colcher (1995) atentam que ela deve ser estabelecida em termos das possíveis ameaças e riscos e dos objetivos de uma organização, formalizados nos termos de uma política de segurança, tal qual também é explicitado por Comer (1998), que cita que antes que uma organização possa reforçar a segurança da rede, ela deve assessorar os riscos e desenvolver uma política clara em relação ao acesso à proteção das informações.

Conforme Nakamura e Geus (2007), a política de segurança trata do documento que norteará todas as ações relacionadas à segurança, sendo que esta consiste, de acordo com Júnior, Suavé, Moura e Teixeira (1999), em uma série de decisões que irão em conjunto determinar a postura de uma organização com relação à segurança. Soares, Lemos e Colcher (1995) ainda definem que uma política de segurança é um conjunto de leis, regras e práticas que regulam como uma organização gerencia, protege e distribui suas informações e recursos. Cabe ressaltar que Comer (1998) cita que essa política deverá explicitar quem terá acesso a cada setor da informação, as regras que alguém deve observar ao difundir a informação entre terceiros e uma declaração de como a organização vai reagir às violações, ou seja, em concordância com o que é dito por Júnior, Suavé, Moura e Teixeira (1999), essa política de segurança deve determinar os limites de tolerância e os níveis de respostas às violações que possam ocorrer.

2.1.3 Aspectos e serviços básicos de segurança

Ramos (2006) informa que os principais aspectos da segurança da informação são basicamente três: confidencialidade, integridade e disponibilidade. Estes três aspectos formam o que é conhecido como pirâmide ou tríade da segurança da informação, que pode ser observada na Figura 1. Da mesma forma, Nakamura e Geus (2007) dividem as propriedades mais importantes para a segurança entre sigilo, integridade e disponibilidade.

Figura 1 - Pirâmide ou tríade da segurança da informação



Fonte: Comer (1998, p. 21).

Ramos (2006) cita que, quando falamos de confidencialidade estamos, basicamente, falando de sigilo. Portanto, a confidencialidade, segundo Tanenbaum e Wetherall (2011), está relacionada ao fato de manter as informações longe de usuários não autorizados. Assim, conforme Comer (2007), a confidencialidade de dados se refere à proteção contra acesso não autorizado a dados, questionando se os dados estão protegidos contra acesso sem autorização, o que ainda é evidenciado por Ramos (2006), que reitera que preservar a confidencialidade de uma informação significa garantir que apenas as pessoas que devem ter conhecimento a seu respeito poderão acessá-la, sendo que essa afirmação se reforça ainda mais quando Soares, Lemos e Colcher (1995) citam que o serviço de confidencialidade fornece proteção aos dados intercambiados contra revelação não autorizada da informação neles transportada e Gil (2000) explica que, consequentemente, a ausência de confidencialidade compreende na quebra de sigilo do sistema computadorizado, seu processo e informações.

A segunda propriedade, a da integridade de dados, é descrita por Comer (1998) como sendo a proteção contra mudanças não autorizadas, já de acordo com Júnior, Suavé, Moura e Teixeira (1999), a integridade de dados é o serviço que permite determinar se uma parte da informação não foi alterada enquanto atravessava o sistema de informação, enquanto que para

Comer (2007) a integridade de dados se refere à proteção contra mudança, indagando se os dados que chegam ao receptor são exatamente os mesmos que foram enviados, ao passo que, para Soares, Lemos e Colcher (1995), o serviço de integridade de dados atua no sentido de proteger os dados intercambiados contra ataques ativos que implicam na modificação, remoção ou injeção não autorizada de unidade de dados.

O terceiro aspecto, que se trata da disponibilidade de dados, segundo Comer (1998), é a garantia de que pessoas de fora não tenham a capacidade de impedir o acesso legítimo aos dados através da saturação do tráfego de uma rede, sendo assim, como é elucidado por Comer (2007), a disponibilidade de dados se refere à proteção contra a interrupção de serviço, ou seja, se os dados permanecem acessíveis para uso legítimo. Neste contexto, Ramos (2006) destaca que uma informação disponível é aquela que pode ser acessada por aqueles que dela necessitam, no momento em que precisam.

Comer (2007) cita estes mesmos três aspectos como sendo os de proteção de uma rede, adicionando um quarto item, que seria a privacidade. Ela se refere à habilidade de um remetente se manter anônimo, interrogando se a identidade do remetente é revelada.

Seguindo o argumento de Júnior, Suavé, Moura e Teixeira (1999), os serviços de segurança fornecem cinco tipos de serviços básicos para a construção de aplicações distribuídas altamente seguras: identificação, autenticação, autorização, criptografia e auditoria.

Ainda sobre os autores, a identificação se relaciona ao conceito de identificar o usuário na rede, sendo esta uma entidade que existe em uma única ou em várias localizações, posto isto que as identidades e suas características são usadas para determinar a que recursos na rede esse usuário tem acesso liberado.

A autenticação, em uma forma geral descrita por Tanenbaum e Wetherall (2011) cuida do processo de definir com quem você está se comunicando antes de revelar informações sigilosas ou ingressar em uma transação comercial, ou seja, conforme explanado por Júnior, Suavé, Moura e Teixeira (1999), a autenticação procura provar que um cliente é, de fato, quem ele está afirmando ser, através da determinação de quem está executando um pedido, certificando-se que o pedido originou-se de uma pessoa em particular e que este é um pedido autêntico, consequentemente, de acordo com Kaufman, Perlman e Speciner (2002), provando quem você é.

O terceiro tipo de serviço básico é a autorização, que Júnior, Suavé, Moura e Teixeira (1999) clarificam que também é chamada de controle de acesso. Comer (2007) desvenda que a autorização se refere à responsabilidade sobre cada item de informações e como tal responsabilidade é delegada a outros, enquanto que Soares, Lemos e Colcher (1995) informam que, este também intitulado controle de acesso possui mecanismos que são usados para garantir que o acesso a um recurso é limitado aos usuários devidamente autorizados, consequentemente, de acordo com Kaufman, Perlman e Speciner (2002), a autorização define o que você está autorizado a fazer.

O próximo tipo de serviço básico abordado é a criptografia, que Soares, Lemos e Colcher (1995) divulgam ter surgido da necessidade de se enviar informações sensíveis através de meios de comunicação não confiáveis, ou seja, em meios onde não é possível garantir que um intruso não irá interceptar o fluxo de dados para leitura ou para modificá-lo. Dessa forma, de acordo com Júnior, Suavé, Moura e Teixeira (1999), os serviços de criptografia permitem que informações sejam enviadas por meio de um *hardware* não confiável e sejam entregues a um destinatário de modo que possa ser detectado, dependendo do mecanismo de criptografia aplicado, se a informação foi ou não distorcida ou espionada em sua jornada. Comer (2007) ainda reforça que a criptografia assegura que o conteúdo de uma mensagem permaneça confidencial apesar do seu grampeamento, realizando tal feito através do embaralhamento de *bits* das mensagens de tal modo que somente o receptor pretendido possa recompor a mensagem. Desta forma, alguém que intercepte uma cópia da mensagem cifrada não poderá extrair informações.

O último serviço básico de segurança é o de auditoria. Júnior, Suavé, Moura e Teixeira (1999) descrevem a auditoria como sendo o processo que mantém registros detalhados sobre quem fez o que com qual objeto, para que se possa determinar como um ataque foi efetuado e que informação ficou comprometida, implicando, segundo Gil (2000), na validação e avaliação do controle interno de sistemas de informações em processamento eletrônico de dados.

2.1.4 Ameaças

Em se tratando da definição do que é uma ameaça, Stallings (2008) informa que ela é nada mais que um possível risco que pode explorar uma vulnerabilidade, ou seja, conforme descrito por Soares, Lemos e Colcher (1995), uma ameaça consiste em uma possível violação

da segurança de um sistema, ou ainda, como foi definido por Júnior, Suavé, Moura e Teixeira (1999) ela se constitui em uma pessoa ou uma organização que pode querer atentar contra a segurança desse sistema, e, por fim, Ramos (2006) especifica que ela é um evento que tem potencial em si próprio para comprometer os objetivos da organização, seja trazendo danos diretos ou prejuízos decorrentes de situações inesperadas. Desenvolvendo um pouco mais a explicação, Stallings (2008) inteira que uma ameaça à segurança é um potencial para violação da segurança, que existe quando há uma circunstância, capacidade, ação ou evento que pode quebrar a segurança e causar danos aos ativos. Aqui se faz necessário explicar primeiramente que, conforme Ramos (2006), o termo ativo se refere a tudo aquilo que possua valor e que necessita de algum tipo de proteção ou cuidado por conta disso.

A fim de classificar as formas de ameaças, Soares, Lemos e Colcher (1995) citam que elas podem ser consideradas passivas ou ativas. Ameaças passivas são as que, quando realizadas, não resultam em qualquer modificação nas informações contidas em um sistema, em sua operação ou em seu estado. Enquanto que as ameaças ativas são as que envolvem a alteração da informação contida no sistema, ou modificações em seu estado ou operação. Não obstante, Soares, Lemos e Colcher (1995) também classificam as ameaças entre acidentais, que são as que não estão associadas à intenção premeditada, ou intencionais, que são as que variam desde a observação dos dados com ferramentas simples de monitoramento das redes, a ataques sofisticados baseados no conhecimento do funcionamento de um sistema.

2.1.5 Riscos

Quando se trata de risco em segurança, Ramos (2006) afirma que o risco nada mais é que a medida que indica a probabilidade de uma deliberada ameaça se concretizar, combinada com os impactos (tamanho do prejuízo que a concretização de uma determinada ameaça causará) que ela trará. Conforme Nakamura e Geus (2007), serão contra estes riscos que as organizações têm de lutar, por meio das técnicas, tecnologias e conceitos de segurança. Segundo, Ramos (2006) estes riscos não podem ser completamente eliminados, mas sim trazidos para dentro de patamares aceitáveis.

Ramos (2006) informa que o risco é a principal métrica gerencial de segurança da informação, adicionando que, quanto maior a probabilidade de uma determinada ameaça acontecer e o impacto que ela trará, maior será o risco agregado a este incidente. É também citado por Ramos (2006) que escala de um risco é dada por meio da combinação de dois

fatores, onde o primeiro fator é a probabilidade de ocorrência da ameaça medida através da combinação da sua frequência com a avaliação de vulnerabilidades e o segundo fator são as consequências trazidas pela ocorrência do incidente.

2.1.6 Vulnerabilidades

Em termos de segurança, uma vulnerabilidade, para Ramos (2006), é a ausência de um mecanismo de proteção ou uma falha existente em um mecanismo de proteção, ou ainda, conforme descrito por Júnior, Suavé, Moura e Teixeira (1999), é um ponto fraco de projeto de sistema que pode ser explorado com más intenções, enquanto que, para Soares, Lemos e Colcher (1995), ela é qualquer fraqueza que pode ser explorada para se violar um sistema ou as informações contidas nele, segundo Ramos (2006), criando situações que podem ser exploradas por uma ameaça, acarretando em prejuízos.

Stallings (2008) ainda atenta para o fato que, uma vez que for descoberta, esta vulnerabilidade será rapidamente difundida entre uma comunidade de atacantes e poderá ser explorada regularmente até ser corrigida. Portanto, segundo Ramos (2006), serão as vulnerabilidades que permitirão que as ameaças se concretizem, o que faz com que ele conclua que uma vulnerabilidade por si só não causa prejuízos, sendo a sua exploração por uma determinada ameaça o que realmente causa prejuízos.

2.1.7 Ataques

Quanto ao o que configura um ataque, Soares, Lemos e Colcher (1995) afirmam que ele configura na realização de uma ameaça intencional. Júnior, Suavé, Moura e Teixeira (1999) definem um ataque como sendo o que ocorre quando uma ameaça tenta levar vantagem sobre uma vulnerabilidade, enquanto que para Stallings (2008), um ataque à segurança é um ato inteligente que é uma tentativa deliberada para escapar dos serviços de segurança e violar a política de segurança de um sistema.

Júnior, Suavé, Moura e Teixeira (1999) ainda informam que os ataques podem ser enquadrados em duas categorias: ataques ativos e ataques passivos. Segundo os próprios Júnior, Suavé, Moura e Teixeira (1999), nos ataques passivos, as ameaças apenas observam a informação que trafega no sistema, ou ainda um ataque passivo pode ser determinado,

conforme Kaufman, Perlman e Speciner (2002), como aquele em que o intruso examina, mas não modifica, o fluxo de mensagens. Stallings (2008) enfatiza que o objetivo de um ataque passivo é obter informações que estão sendo transmitidas, desta forma, descobrindo ou utilizando estas informações do sistema, mas não afetando os seus recursos. Como consequência, e ainda em concordância com Stallings (2008), ataques passivos incluem acesso não autorizado ao tráfego de rede entre navegador e servidor e obtenção de acesso a informações que deveriam ser restritas, através de análise de tráfego e leitura não autorizada de uma mensagem de arquivo. Realizando uma ligação com o próximo tipo de ataque, Júnior, Suavé, Moura e Teixeira (1999) evocam que um intruso passivo pode operar por meios ativos para facilitar um ataque passivo.

Adentrando na elucidação dos aspectos de um ataque ativo, Júnior, Suavé, Moura e Teixeira (1999) esclarecem que ele altera o sistema na tentativa de levar vantagem sobre as vulnerabilidades. Quanto a sua extensão, Kaufman, Perlman e Speciner (2002) informam que um ataque ativo é aquele em que o intruso pode transmitir mensagens, reproduzir mensagens antigas, modificar mensagens em trânsito ou excluir mensagens selecionadas, ou seja, tal qual dito por Stallings (2008), ataques ativos incluem simulação de outro usuário, alteração de mensagens em trânsito entre cliente e servidor e alteração de informações, envolvendo, ainda segundo Stallings (2008), alguma modificação do fluxo de dados ou a criação de um fluxo falso.

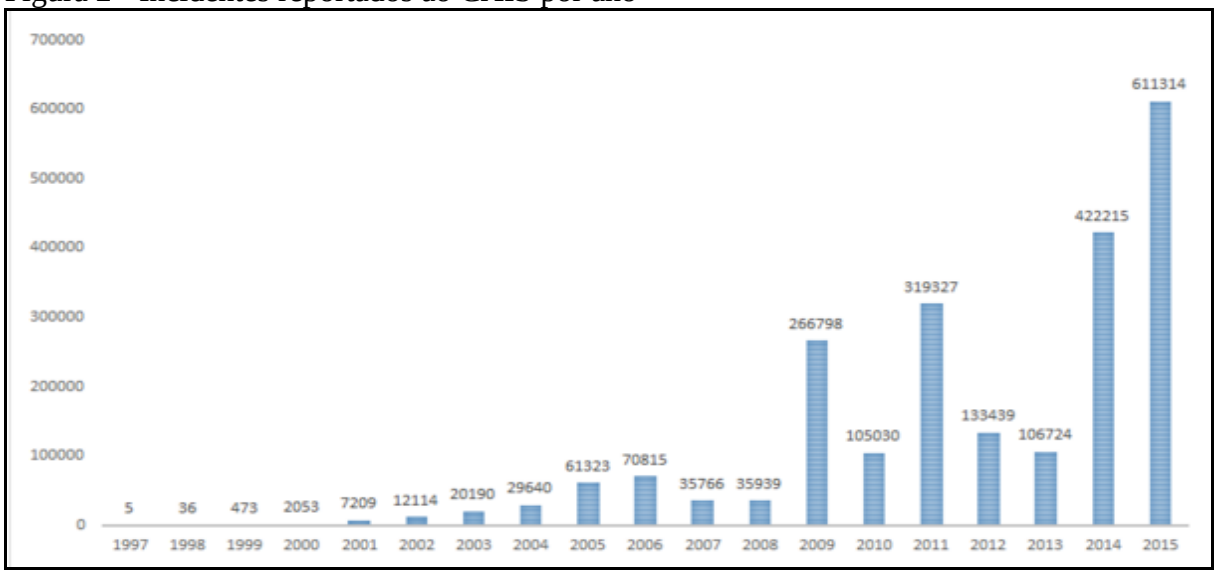
2.1.8 Cenário nacional de incidentes de segurança em redes

Dados estatísticos levantados pelo CETIC.br (2010) no período de 2006 até 2010, mostram uma relação entre os problemas de segurança de acessos externo e interno não autorizados encontrados em empresas, onde, de forma resumida, se pode extrair que em empresas de pequeno porte (de 10 a 49 funcionários) o número de acessos não autorizados externos é maior que o de internos, porém a situação se inverte conforme o número de funcionários vai crescendo, tornando o percentual de acessos internos não autorizados maior do que os acessos externos não autorizados. Essa diferença de percentual ainda vai aumentando para os casos de acessos não autorizados internos conforme aumenta o número de funcionários, demonstrando que, quanto maior for a empresa (ou maior for o cenário de funcionários conectados à rede nesta empresa), maior é a chance de ocorrer um acesso interno não autorizado e menor é porcentual de acesso externo não autorizado. Porém, estas

categorias de problemas de segurança encontrados em empresas não foram mais relacionadas nas estatísticas do CETIC.br dos anos a partir de 2010.

Conforme dados fornecidos pelo CAIS (2016), que podem ser conferidos na Figura 2, os incidentes de segurança reportados, de forma geral, vêm crescendo vertiginosamente nos últimos anos.

Figura 2 - Incidentes reportados ao CAIS por ano

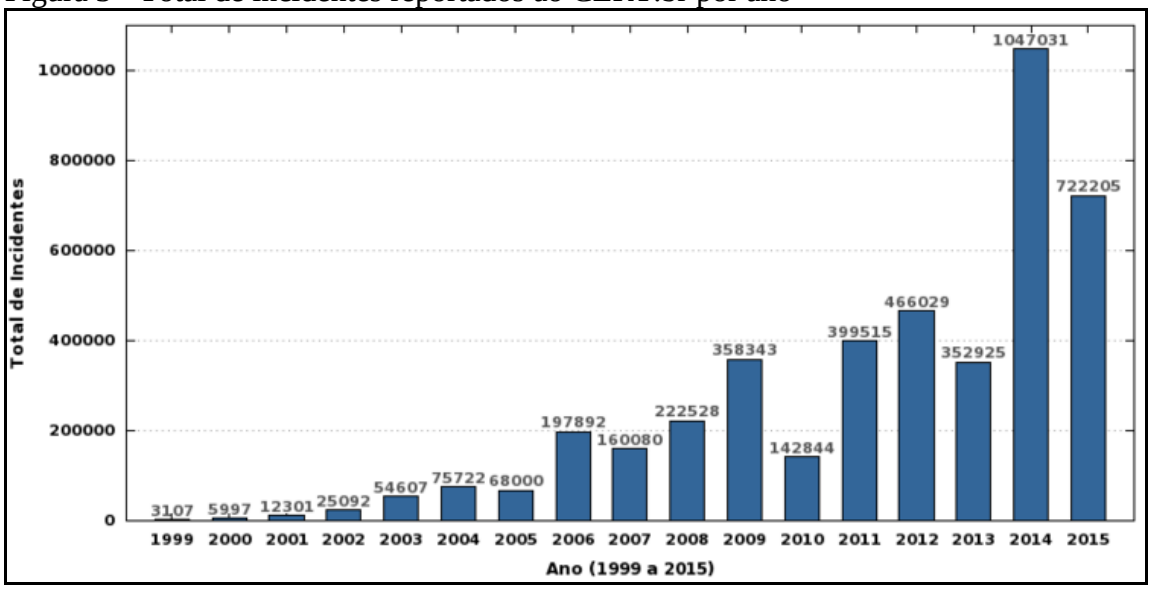


Fonte: CAIS (2016).

Por outro lado, apesar do NIC.br (2016) ressaltar, com base nos dados do CERT.br (2016), que os incidentes de segurança reportados em 2015 diminuíram em cerca de 31% em relação a 2014, conforme demonstrado pela Figura 3, no mesmo intervalo de período houve um crescimento em determinadas modalidades de ataques, como o de varreduras, técnica que tem o objetivo de identificar computadores ativos e coletar informações sobre eles, que teve um aumento de 48% nos incidentes relatados.

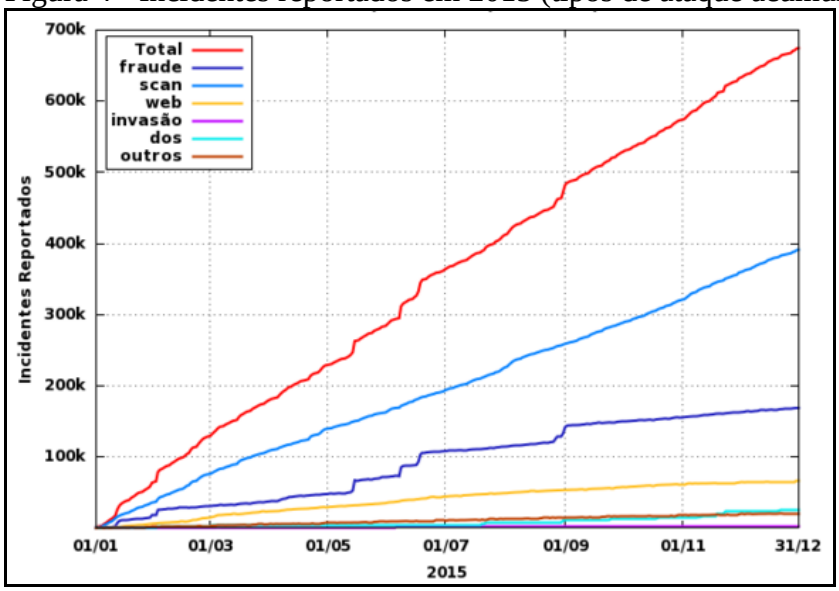
Já os dados de incidentes de segurança obtidos pelo CERT.br (2016), quando classificados por tipos de ataques durante o ano resultam nos gráficos da Figura 4 e da Figura 5.

Figura 3 - Total de incidentes reportados ao CERT.br por ano



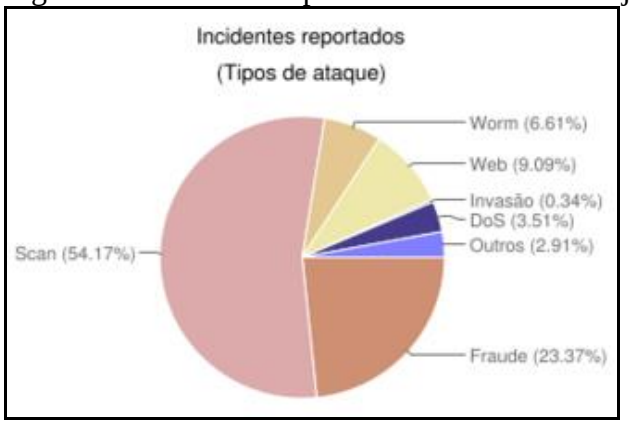
Fonte: CERT.br (2016).

Figura 4 - Incidentes reportados em 2015 (tipos de ataque acumulados)



Fonte: CERT.br (2016).

Figura 5 - Incidentes reportados ao CERT.br de janeiro a dezembro de 2015



Fonte: CERT.br (2016).

Generalizando, o CERT.br (2016) cita que os números de cada categoria de ataque relacionados também demonstram que existe uma diminuição nos relatos de tentativas com sucesso de ataques tradicionais, que tem como objetivo entrar em uma organização passando por "brechas no *firewall*", e um aumento em 2015 de 47% com relação a 2014 de notificações de ataques que se enquadram na categoria de "outros", o que demonstra que a forma de realização de ataques está migrando para meios alternativos onde a segurança em redes é mais negligenciada.

2.2 Redes de computadores

Tanenbaum e Wetherall (2011) recordam que o velho modelo de um único computador atendendo a todas as necessidades computacionais da organização foi substituído por outro em que os trabalhos são realizados por um grande número de computadores separados, porém interconectados. Esses sistemas são chamados redes de computadores, termo que serve para indicar, de acordo com Tanenbaum e Wetherall (2011), um conjunto de computadores autônomos interconectados por uma única tecnologia, sendo que eles informam que dois computadores estão interconectados quando podem trocar informações.

Já nas palavras de Soares, Lemos e Colcher (1995) uma rede de computadores é formada por um conjunto de módulos processadores (sendo estes qualquer dispositivo capaz de se comunicar através do sistema de comunicação por troca de mensagens) capazes de trocar informações e compartilhar recursos, interligados por um sistema de comunicação, ao passo que este sistema de comunicação vai se constituir de um arranjo topológico interligando vários módulos processadores através de enlaces físicos (meios de transmissão) e de um conjunto de regras com a finalidade de organizar a comunicação (protocolos). Forouzan (2010), por sua vez, explica que uma rede é um conjunto de dispositivos (normalmente conhecido como nós) conectados por *links* de comunicação, sendo que um nó pode ser um computador, uma impressora ou outro dispositivo de envio e/ou recepção de dados, que estejam conectados a outros nós da rede.

Comer (2007, p. 598) explica que protocolo é "um projeto que especifica os detalhes de como os computadores interagem, incluindo o formato das mensagens que trocam e como erros são tratados", já para Toledo (2005), protocolos são basicamente a parte do sistema operacional da rede que é encarregada de ditar as normas para a comunicação entre os dispositivos, enquanto que, para Kurose e Ross (2010, p. 6) "um protocolo define o formato e

a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou recebimento de uma mensagem ou outro evento".

No nível mais baixo, Comer (2007) explica que toda a comunicação entre computadores envolve codificar dados em uma forma de energia e enviar essa energia ao longo de um meio de transmissão. Ele também ressalta que, uma vez que os dispositivos de *hardware* conectados a um computador executam a codificação e a decodificação dos dados, os programadores e os usuários não precisam entender os pormenores da transmissão.

Campbell (1997), Comer (2007), Kurose e Ross (2010), Soares, Lemos e Colcher (1995), Tanenbaum e Wetherall (2011) e Toledo (2005) informam que esta conexão pode ser feita por fios de cobre (que são compostos por cabos como o coaxial e o cabo par trançado), fibras de vidro (conhecidas como fibra óptica), ou ainda através de redes sem fio, como micro-ondas, ondas de infravermelho, ondas de rádio eletromagnéticas, transmitidas por enlaces de satélites de comunicações ou luz de laser, o que leva todos essas formas de transmissão a demonstrar a citação de Soares, Lemos e Colcher (1995) de que qualquer meio físico capaz de transportar informações eletromagnéticas é passível de ser usado em redes de computadores.

2.2.1 Endereçamento de *hardware* na rede

Como já é possível ser constatado, e é relatado por Soares, Lemos e Colcher (1995), todas as estações devem ser capazes de reconhecer se uma mensagem ou pacote entregue deve ser passado a uma outra estação, ou se tem como destino a própria estação. Para que realize tal feito, Soares, Lemos e Colcher (1995) explicam que qualquer rede eficiente tem a necessidade de definir mecanismos de endereçamento que permitam as suas estações decidir que atitude devem tomar ao receber uma mensagem ou pacote.

Tendo em vista esta situação, é citado por Comer (2007) que, fisicamente, qualquer sinal enviado através de uma rede compartilhada alcança todas as estações acopladas, deste modo, quando um conjunto de *bits* for transferido através de uma rede local, os sinais elétricos transportando estes *bits* alcançam todas as estações. Para resolver que dois computadores possam se comunicar diretamente através de um meio compartilhado neste cenário, Comer (2007) expõe que a maioria das tecnologias de rede usa um esquema de endereçamento para fornecer comunicação direta.

Soares, Lemos e Colcher (1995) descrevem que este endereço irá consistir em uma maneira de reconhecer univocamente cada uma das estações conectadas à rede. Para este fim, Comer (1998) explica que a todos os computadores conectados a uma rede é atribuído apenas um único endereço numérico. Comer (2007) informa que este valor numérico único designado a cada estação de rede é chamado de endereço físico, endereço de *hardware*, endereço de acesso de meios ou endereço *Media Access Control* (MAC).

Para compreender as particularidades do endereço MAC, primeiramente cabe trazer que Chowdhury (2002) informa que o sistema binário é representado com zero (0) ou um (1), e cada um desses valores é conhecido como *bit*. É explicado por Campbell (1997) e Comer (1998) que os endereços MAC possuem 48 *bits*, onde, segundo Corrêa (2009), os primeiros 24 *bits* são aplicados para determinar o fabricante da placa de rede, ao passo que os últimos 24 *bits* são atribuídos pelo fabricante da placa, além do fato corroborado por Campbell (1997) e Comer (1998) de que os endereços MAC devem ser únicos dentro de uma rede, para que não haja conflito.

A partir da definição desses endereços, é necessário explicar que, seguindo Comer (1998), um pacote enviado através de uma rede tem um campo de endereço de destino que possui o endereço do destinatário, sendo que Comer (2007) amplia essa explicação informando que, além dessa especificação do receptor pretendido, cada pacote transmitido através de uma rede também possui um endereço que especifica o remetente. É importante citar que, segundo Comer (1998, p. 22) “o endereço de destino aparece na mesma localização em todos os pacotes, possibilitado ao *hardware* da rede identificar o endereço de destino facilmente”. Portanto, de acordo com Comer (2007), quando um remetente transmite um pacote pela rede, ele inclui este endereço de *hardware* do receptor desejado.

Com isso em mente, antes de encaminhar um pacote, Comer (2007) e Comer (1998) relatam que um transmissor, ou remetente, deve saber o endereço do destinatário e o deve inserir no campo de endereço de destino deste pacote. É acrescentado por Comer (2007) que, além do remetente ter que colocar o endereço do receptor no campo de endereço de destino, ele deve também colocar o seu próprio endereço no campo de endereço de origem do pacote. Sendo que essa inclusão do endereço do remetente torna mais fácil a geração de uma resposta pelo receptor.

Seguindo a explanação de Comer (1998) e Comer (2007), é importante citar que o *hardware* de interface de rede é projetado para examinar os campos de endereço em pacotes

que passam através da rede e, assim, para aceitar somente aqueles em que o endereço de destino combina com o endereço da estação, ou seja, somente as designadas a um dos endereços especificados, sendo que Comer (1998) acrescenta que, sempre que o sistema operacional é inicializado, ele inicializa também essa interface que oferecerá um conjunto de endereços a serem reconhecidos.

2.2.2 LAN

Campbell (1997) descreve uma *Local Area Network* (LAN) ou rede local, como sendo um grupo de computadores que são conectados entre si dentro de uma certa área, enquanto que, para Soares, Lemos e Colcher (1995), pode-se caracterizar uma rede local como sendo uma rede que permite a interconexão de equipamentos de comunicação de dados numa pequena região. Já para Comer (1998), Comer (2007), Forouzan (2010), Kurose e Ross (2010), Soares, Lemos e Colcher (1995) e Tanenbaum e Wetherall (2011), a LAN é uma rede particular que se concentrada na abrangência de uma pequena área geográfica tal como um único edifício, uma residência, campus universitário ou escritório. Soares, Lemos e Colcher (1995) flexibilizam um pouco a descrição de faixa de abrangência ao informar que as redes locais de computadores são sistemas cujas distâncias entre os módulos processadores (*hosts*) se enquadram na faixa de alguns poucos metros a alguns poucos quilômetros.

Entretanto, para Campbell (1997), no mundo real, as LANs geralmente são definidas mais por sua função do que por suas características físicas, portanto, para ele, uma rede local é um sistema em que computadores e dispositivos periféricos conectados podem compartilhar informações, programas, impressoras, serviços, dentre outros elementos. Não obstante, segundo Campbell (1997), as LANs variam grandemente em tamanho, sendo possível formar uma a partir de apenas dois computadores colocados um ao lado do outro na mesma sala, ou com milhares em um mesmo edifício.

2.2.2.1 *Ethernet*

De acordo com Tanenbaum e Wetherall (2011), muitos projetos para redes pessoais, locais e metropolitanas foram padronizados com o nome IEEE 802. Entre os sobreviventes destes padrões, um dos mais importantes é o padrão IEEE 802.3, ou *Ethernet*.

Não apenas a *Ethernet* é o padrão empregado no ambiente em que é aplicado este trabalho, como também ela, segundo Comer (2007) é uma tecnologia de rede bem conhecida e extensamente utilizada. Embora Comer (2007) informe que outras tecnologias tenham sido inventadas, a *Ethernet* domina todas as LANs, se tornando, conforme Campbell (1997) o tipo mais popular de rede local, ou ainda, em conformidade com Tanenbaum e Wetherall (2011) sendo provavelmente o tipo de rede de computação mais utilizado no mundo.

No que refere a sua história de criação, Comer (1998), Comer (2007), Tanenbaum e Wetherall (2011) explicam que *Ethernet* é o nome dado a uma tecnologia de rede local popular, de comutação de pacotes, criada pelo Centro de Pesquisa da Corporação Xerox em Palo Alto (*Xerox Palo Alto Research Center*) no início da década de 1970 por Bob Metcalfe e David Boggs, quando estes implementaram a primeira rede local utilizando um único cabo coaxial.

Mais tarde, em 1978, segundo Comer (1998), Comer (2007) e Tanenbaum e Wetherall (2011), devido ao sucesso da rede *Ethernet* da Xerox as empresas *Digital Equipament Corporation*, *Intel Corporation* e Xerox cooperaram, para desenvolver um padrão de produção da *Ethernet*. O Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE), que conforme Júnior, Suavé, Moura e Teixeira (1999) é um dos órgão públicos responsáveis pela padronização da tecnologia da informação, seguindo as colocações de Comer (1998), Comer (2007) e Tanenbaum e Wetherall (2011) criou uma versão compatível do padrão, com uma pequena mudança, fazendo com que este padrão de produção da *Ethernet* se torna-se o padrão IEEE 802.3 em 1983, e, conforme Comer (2007), levando o IEEE a controlar os padrões *Ethernet* a partir de então.

Complementando, Soares, Lemos e Colcher (1995) informam que a IEEE 802, a qual pertence o padrão IEEE 802.3, é uma iniciativa do IEEE que define os padrões para os níveis físico e enlace de redes locais de computadores. Por fim, Tanenbaum e Wetherall (2011) explicam que, como a *Ethernet* e o padrão IEEE 802.3 são praticamente idênticos, muitas pessoas usam os termos "*Ethernet*" e "IEEE 802.3" para indicar a mesma coisa.

2.2.2.1.1 *Ethernet* comutada e o *switch*

É importante ressaltar que em uma rede de comutação de pacotes, conforme Comer (1998), as mensagens são divididas em pequenas unidades, nomeadas de pacotes, que são

multiplexados por meio de conexões entre máquinas de alta capacidade. Soares, Lemos e Colcher (1995) reforçam que os sistemas por comutação de pacotes se caracterizam justamente pelo fato da mensagem ser quebrada em quadros ou pacotes antes da transmissão ser efetuada.

Seguindo o raciocínio de Comer (1998), um pacote, que geralmente contém apenas pequenas unidades de informações, transporta uma identificação que capacita o *hardware* da rede a enviar as informações a determinado destino. Assim, o *hardware* da rede envia os pacotes aos seus respectivos destinos, onde o *software* junta-os novamente em um arquivo único. Ainda cabe ressaltar que, de acordo com Comer (2007), para ajudar a distinguir entre a ideia geral de transmissão de pacotes e a definição específica de pacotes para uma dada tecnologia de *hardware*, alguns utilizam o termo quadro para denotar a definição de um pacote utilizado com um tipo específico de rede.

Seguindo o assunto da *Ethernet*, Comer (1998), Comer (2007) e Tanenbaum e Wetherall (2011) informam que a versão original da *Ethernet*, também chamada de *Ethernet* clássica, logo começou a evoluir para longe da arquitetura primária de um cabo longo único, devido aos problemas associados a encontrar interrupções ou conexões partidas. Isto levou a um tipo diferente de padrão de fiação em que cada estação tem um cabo dedicado esticado até um *hub* central (equipamento que simplesmente conecta todos os fios eletricamente, como se eles fossem únicos).

É necessário explicar que, de acordo com Comer (1998), Forouzan (2010) e Soares, Lemos e Colcher (1995), o *hub*, também chamado de concentrador, é dispositivo eletrônico central que fornece interconexão entre os nós de uma rede local, estimulando os sinais num cabo *Ethernet*. Porém os *hubs* possuem vários inconvenientes, como o citado por Tanenbaum e Wetherall (2011) de que eles não aumentam a capacidade, assim, conforme mais e mais estações forem sendo acrescentadas, cada estação recebe uma fatia cada vez menor da capacidade fixa, o que leva, por fim, a LAN a ficar saturada.

As demandas sufocantes citadas por Chowdhury (2002), por largura de banda, segurança, administração, complexidade e microssegmentação, somadas as demandas citadas por Soares, Lemos e Colcher (1995) de maiores taxas de transmissão e melhor usabilidade dos meios físicos, aliados à evolução contínua da microeletrônica, começaram a alterar a construção desses equipamentos concentradores (*hubs*). Conforme Tanenbaum e Wetherall (2011), isto levou a uma solução para lidar com o aumento da carga denominada de *Ethernet*

comutada ou *switched Ethernet*. Forouzan (2010), Kurose e Ross (2010) e Tanenbaum e Wetherall (2011) citam que a *Ethernet* comutada, se caracteriza por ser uma rede em que os *hubs* centrais foram substituídos por dispositivos chamados *switches*, ou comutadores, que por sua vez são usados para conectar diferentes computadores e encaminhar as mensagens ao seu destino.

Forouzan (2010), Kurose e Ross (2010) e Soares, Lemos e Colcher (1995) definem que esse tipo de elemento central denominado *switch*, ou comutador, é um dispositivo que interliga várias linhas de comunicação, tendo a função de receber quadros da camada de enlace e encaminhá-los para enlaces de saída. Tanenbaum e Wetherall (2011) ressalta que por fora, um *switch* se parece com um *hub*, inclusive mantendo as mesmas vantagens de um *hub*, que seriam a facilidade em acrescentar ou remover uma nova estação conectando ou desconectando um fio, e a facilidade em encontrar a maioria das falhas, pois um cabo ou porta com defeito normalmente afetará apenas uma estação.

Porém, Tanenbaum e Wetherall (2011) enaltecem que internamente o *switch* é muito diferente. Os *switches* só enviam quadros às portas para as quais esses quadros são destinados e quando uma porta do *switch* recebe um quadro *Ethernet* de uma estação, o *switch* verifica os endereços *Ethernet* para saber para qual porta o quadro se destina. Depois, o *switch* encaminha o quadro por sua placa interna de alta velocidade até a porta de destino, que transmite o quadro no fio para que ele alcance a estação intencionada, o que faz com que Tanenbaum e Wetherall (2011) esclareçam que nenhuma das outras portas sequer saberá que o quadro existe.

Com isso, Kurose e Ross (2010) afirmam que o comutador em si é transparente até aos *hosts*, ou seja, um *host* endereça um quadro a outro *host* (em vez de endereçar o quadro ao *switch*) que prontamente envia o quadro à LAN, inconsciente de que um *switch* irá receber o quadro e encaminhá-lo a outros *hosts*. Do ponto de vista de Soares, Lemos e Colcher (1995), o modo de operação usual dos *switches* dita que primeiramente eles recebem e armazenam os quadros, depois eles processam o endereço de destino e estabelecem um circuito entre as portas de origem e de destino, enquanto durar a transmissão do quadro.

Apesar de as camadas de uma rede ainda não terem sido abordadas, cabe citar que Kurose e Ross (2010) informam que os *switches* encaminham pacotes baseados em endereços MAC, ao invés de endereços de *Internet Protocol* (IP), atuando assim na camada de enlace de dados do modelo RM-OSI/ISO.

Em concordância com Tanenbaum e Wetherall (2011), em um *hub*, todas as estações estão no mesmo domínio de colisão. Já em um *switch*, cada porta é seu próprio domínio de colisão independente, o que leva Forouzan (2010) a reforçar que, em um *switch*, o domínio de colisão é dividido em N domínios, sendo N o número de portas deste, fazendo com que Soares, Lemos e Colcher (1995) especifiquem que a ideia utilizada pelos *switches* é justamente segmentar a rede, para aprimorar o seu desempenho, fornecendo a cada uma de suas portas, que podem estar ligadas a uma ou mais estações, uma taxa de transmissão na rede igual à do seu enlace de entrada/saída.

Cabe destacar que Kurose e Ross (2010) informam que o *switch* tem a maravilhosa propriedade de montar sua tabela de endereços MAC correspondentes a cada porta de forma automática, dinâmica e autônoma, ou seja, sem nenhuma intervenção de um administrador de rede ou de um protocolo de configuração, sendo que essas tabelas são denominadas de tabelas *Content Addressable Memory* (CAM). Em alguns modelos de *switch*, como explicado por Extreme Networks (2011), temos uma *Forwarding Database* (FDB), que é uma base de dados de encaminhamento mantida pelo *switch* com todos os endereços MAC recebidos em todas as suas portas. Assim o *switch* usará a informação desta base de dados FDB para decidir se um quadro deve ser encaminhado ou filtrado. Segundo Kurose e Ross (2010, p. 352), “em outras palavras, comutadores são autodidatas”. Corrêa (2009) conclui que a tabela CAM, ou FDB, de um *switch* é a responsável por armazenar e relacionar endereços MAC, portas e parâmetros de *Virtual Local Area Network* (VLAN), tornando possível que ele encaminhe de forma correta os quadros que passam por ele. A Figura 6 ilustra essa tabela CAM descrita.

Figura 6 - Exemplo de tabela CAM

```
BrainSW05#show mac address-table dynamic
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

  10    000d.9dd2.b5a5    DYNAMIC     Fa0/1
  10    000e.3834.b638    DYNAMIC     Fa0/21
  10    000e.3834.b954    DYNAMIC     Fa0/19
  10    000e.3888.d251    DYNAMIC     Fa0/1
  10    000e.3892.ddd1    DYNAMIC     Fa0/4
  10    0012.8053.f49c    DYNAMIC     Fa0/15
  10    0012.8053.f6e4    DYNAMIC     Fa0/12
  10    0012.8053.f76f    DYNAMIC     Fa0/9
  10    0012.8053.f77b    DYNAMIC     Fa0/10
  10    0012.8055.becf    DYNAMIC     Fa0/22
  10    0012.8055.c076    DYNAMIC     Fa0/7
  10    0012.8055.cc42    DYNAMIC     Fa0/20
  10    0012.8081.e07c    DYNAMIC     Po1
  10    0012.80b5.a124    DYNAMIC     Fa0/23
  10    0012.80bb.e3d4    DYNAMIC     Fa0/3
  10    0012.80bb.e3d5    DYNAMIC     Fa0/8
  10    0012.80bb.e457    DYNAMIC     Fa0/13
  10    0012.da8a.c496    DYNAMIC     Fa0/6
```

Fonte: Ortega (2012).

No que trata da segurança na rede, Tanenbaum e Wetherall (2011) informam que a mudança nas portas em que os quadros são enviados também tem benefícios nessa área. Como a maioria das interfaces de LAN possui um modo promíscuo, em que todos os quadros são dados a cada computador, não apenas os endereçados a ele, com um *hub*, cada computador conectado pode ver o tráfego enviado entre todos os outros computadores, gerando uma grande falha de segurança, mas com um *switch*, o tráfego é encaminhado apenas para as portas às quais ele é destinado. Essa restrição acaba oferecendo melhor isolamento, de modo que o tráfego não escapará com facilidade nem cairá em mãos erradas.

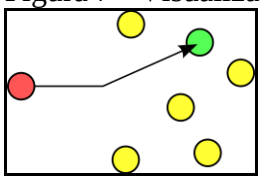
Considerando todos esses fatores Kurose e Ross (2010, p. 352) concluem que "o comutador é "mais esperto" do que um *hub*". Assim Kurose e Ross (2010) podem dizer que os *switches* têm como vantagens, em relação aos *hubs*, a eliminação de colisões, os enlaces heterogêneos e a facilidade na gestão da rede, além da afirmação de Soares, Lemos e Colcher (1995), de que eles possuem taxas efetivas de transmissão bem maiores, o que faz com que Tanenbaum e Wetherall (2011) afirmem que os *hubs* são espécies em extinção.

2.2.2.1.2 Tipos de transmissão de dados

No que se refere a tecnologia de transmissão de dados em uma rede de computadores, autores como Chowdhury (2002), Comer (1998), Comer (2007), Forouzan (2010) e Kurose e Ross (2010) informam que ela pode ser dividida entre *unicast*, *broadcast* e *multicast*.

A comunicação *unicast*, segundo Kurose e Ross (2010), ocorre onde um único nó de fonte envia um pacote a um único nó de destino, ou seja, seguindo Forouzan (2010), existem apenas uma origem e um destino, logo, a relação entre a origem e o destino é um-para-um, mesmo que, de acordo com Tanenbaum e Wetherall (2011), os pacotes tenham que percorrer máquinas intermediárias no percurso deste caminho. A Figura 7 ajuda a exemplificar estas definições de *unicast*, ilustrando um nó (círculo vermelho) enviando um pacote para apenas um outro nó (círculo verde) em uma rede com vários nós (círculos amarelos).

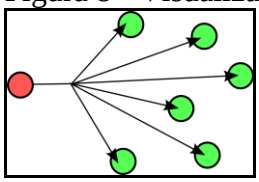
Figura 7 - Visualização do esquema *unicast*



Fonte: Cobbaut (2015).

Conforme Comer (2007), muitos aplicativos que usam uma rede se baseiam em uma outra técnica, conhecida como *broadcasting* (ou difusão). Para Tanenbaum e Wetherall (2011), os pacotes de *broadcast* enviados por qualquer máquina são recebidos por todas as outras. Portanto, Comer (2007) explica que o *broadcast* se trata de uma forma de transmissão em que uma cópia de um pacote é entregue a cada computador em uma rede, já na visão de Kurose e Ross (2010), no *broadcast* a rede provê um serviço de entrega de um pacote enviado de um nó de fonte a todos os outros nós da rede, enquanto que Forouzan (2010) define que na comunicação *broadcast*, a relação entre a origem e o destino é um-para-todos, assim existindo apenas uma origem, mas possuindo todos os demais *hosts* (Comer (2007) e Kurose e Ross (2010) explicam que os *hosts* são dispositivos finais conectados a uma rede) como destinos. A Figura 8 ajuda a elucidar o funcionamento do *broadcast*, ilustrando um nó (círculo vermelho) enviando um pacote para todos os demais nós na rede (círculos verdes).

Figura 8 - Visualização do esquema *broadcast*



Fonte: Cobbaut (2015).

Para tornar o uso de *broadcast* eficiente, Comer (2007) informa que a maioria das tecnologias de redes estende o esquema de endereçamento, de modo que, além de designar a cada computador um endereço, os projetistas de rede definem um endereço especial reservado especificamente para *broadcast*. Comer (1998) delibera que, por convenção, esse endereço de difusão (*broadcast*) para envio simultâneo a todas as estações da rede tem todos os *bits* ajustados ao 1. Assim, seguindo Chowdhury (2002) e Corrêa (2009), um *broadcast* enviado para um endereço físico (MAC) tem o endereço de destino como sendo o 0xFFFFFF, ou FF-FF-FF-FF-FF-FF. Enquanto que, no nível lógico, Kurose e Ross (2010) trazem à tona que o endereço de *broadcast* de um protocolo IP é usualmente o 255.255.255.255, e, mesmo em uma sub-rede onde esse IP de *broadcast* varia, ele será sempre o último endereço existente em qualquer rede.

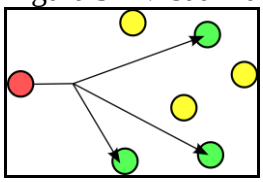
Apesar da viabilidade aparente demonstrada em usar *broadcast*, Comer (2007) adverte que, em alguns casos, enviar por *broadcast* é ineficiente, já que, embora cada estação possa ser configurada para descartar os pacotes desnecessários, o processamento e o descarte de um pacote consome recursos computacionais. Paralelamente, Forouzan (2010) explica que

o *broadcast* pode provocar uma quantidade enorme de tráfego e necessitar de uma grande quantidade de largura de banda na rede.

Visto estas desvantagens relacionadas ao *broadcast*, Comer (2007) conta que, para se aproveitar da capacidade de *broadcast* sem desperdiçar recursos de processamento em outros computadores, a resposta foi encontrar uma forma restrita de *broadcasting* conhecida como *multicasting*. O que é reforçado por Tanenbaum e Wetherall (2011) quando estes citam que alguns sistemas de *broadcasting* também admitem a transmissão para um subconjunto de máquinas, o que se conhece como *multicasting*.

Conforme descrito por Comer (2007), no nível mais baixo o *multicast* opera de forma muito semelhante ao *broadcast*, porém, segundo Kurose e Ross (2010), o *multicast* habilita um único nó de fonte a enviar cópia de um pacote a um subconjunto de nós das outras redes. A fim de descrever ainda mais o que é um *multicast*, Forouzan (2010) informa que na comunicação *multicast*, existe apenas uma origem e um grupo de destinos, portanto, a relação é um-para-vários. Ele ainda faz um paralelo de que, nesse tipo de comunicação, o endereço de origem é um endereço *unicast*, mas o endereço de destino é um endereço de grupo, que define um ou mais destinos, sendo que o endereço de grupo identifica seus membros. A Figura 9 ajuda a visualizar o funcionamento do *multicast*, ilustrando um nó (círculo vermelho) enviando um pacote para um grupo de outros nós (círculos verdes) em uma rede com vários nós (círculos amarelos).

Figura 9 - Visualização do esquema *multicast*



Fonte: Cobbaut (2015).

2.2.3 VLANs como proposta de organização dos fluxos internos

Recordando através da citação de Forouzan (2010), uma estação é considerada parte de uma LAN se pertencer fisicamente a ela, sendo que o critério desta participação é geográfico. Porém Forouzan (2010) questiona, o que acontece se precisarmos de uma conexão virtual entre duas estações pertencentes a duas LANs físicas distintas. Tanenbaum e Wetherall (2011) replicam que, em resposta à esta solicitação e também a de usuários que, de forma geral, desejam maior flexibilidade, os fornecedores de redes começaram a buscar um meio de

recompor a fiação dos prédios inteiramente via *software*. Sendo que Kurose e Ross (2010) explicam que as dificuldades geradas pela falta de isolamento do tráfego, uso ineficiente de comutadores e o gerenciamento de usuários, foram o que motivaram esta busca, e podem ser resolvidas com um *switch* que suporte uma rede local virtual.

Seguindo a colocação de Forouzan (2010) e Tanenbaum e Wetherall (2011), esse conceito resultante chamado de rede local virtual, LAN virtual, ou *Virtual Local Area Network* (VLAN), foi padronizado pelo comitê IEEE 802 sendo denominado de padrão 802.1Q, que define o formato para identificação de quadros deste.

Para determinar do que se trata uma VLAN, Tanenbaum e Wetherall (2011) explicam que as VLANs permitem que a topologia física seja segmentada em diferentes topologias lógicas, sendo que Forouzan (2010) define uma VLAN como uma rede local configurada por *software* em vez de fiação física. Forouzan (2010) ainda informa que as VLANs agrupam estações pertencentes a uma ou mais LANs físicas em domínios de *broadcast*, portanto, as estações em uma VLAN se comunicam entre si como se pertencessem a um mesmo segmento físico. O que é consolidado por Forouzan (2010, p. 460) ao citar que "as VLANs criam domínios de *broadcast*".

Para explicar como funcionam os quadros dentro desse padrão, Tanenbaum e Wetherall (2011) informam que este novo formato contém uma *tag* de número identificador (ID) da VLAN, alterando o cabeçalho do padrão *Ethernet*. Segundo Tanenbaum e Wetherall (2011), mudar o cabeçalho do padrão *Ethernet* só é possível pois os campos VLANs só são realmente usados por equipamentos como os *switches*, e não pelas máquinas dos usuários. De acordo com Tanenbaum e Wetherall (2011), quando um quadro marcado chega a um *switch* que reconhece VLANs, o *switch* utiliza a ID da VLAN como um índice em uma tabela, para descobrir por meio de que portas deve enviar o quadro. Porém, Tanenbaum e Wetherall (2011) avisa que, como pode haver *switches* que não reconhecem a VLAN, o primeiro *switch* no caminho que a reconhece e toca em um quadro acrescenta os campos de VLAN, enquanto que o último neste caminho os remove.

Como vantagem relacionada a segurança em redes em se utilizar uma VLAN, Forouzan (2010) cita que as VLANs fornecem uma medida extra de segurança, já que pessoas pertencentes ao mesmo grupo podem enviar mensagens de *broadcast* com absoluta garantia de que os usuários nos demais grupos não receberão essas mensagens. Apesar dessa afirmação, Nakamura e Geus (2007) reiteram que as VLANs não podem ser consideradas

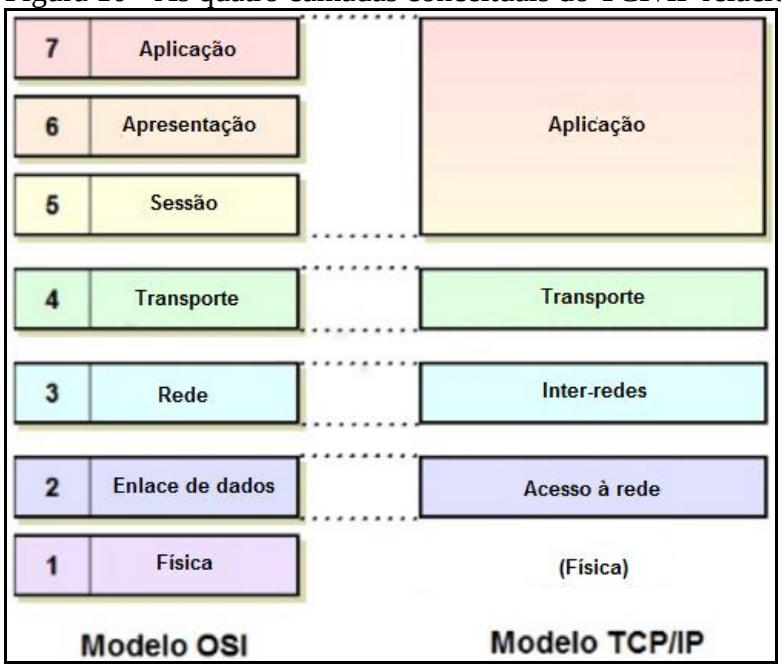
como mecanismos de segurança, devido a possibilidade de exploração das suas vulnerabilidades, as quais serão abordadas posteriormente no ataque de *VLAN hopping*, na seção 2.3.7. Portanto, segundo eles, elas podem ser consideradas apenas como uma segmentação de redes para otimizar a utilização de *broadcasts* e *multicasts*, além de reduzir o número de colisões.

2.2.4 Modelo de referência em camadas TCP/IP

Segundo Comer (2007), foram desenvolvidas várias ferramentas para ajudar os projetistas de protocolos a compreender as subpartes do problema de comunicação e planejar todo um conjunto de protocolos. Sendo que, de acordo com Soares, Lemos e Colcher (1995), dentre os princípios obtidos pela experiência adquirida no projeto de redes se destaca a ideia de estruturar a rede como um conjunto de camadas hierárquicas, cada uma sendo construída utilizando as funções e serviços oferecidos pelas camadas inferiores, o que se tornaria, em conformidade com Comer (2007), uma das ferramentas mais importantes, sendo chamada de modelo em camadas ou *layering model*. No que tange a segurança em redes, Tanenbaum e Wetherall (2011) relatam que a segurança não se ajusta nitidamente a nenhuma camada, mas sim todas as camadas da rede contribuem de alguma forma para a segurança.

Antes de abordar o modelo de referência *Transmission Control Protocol / Internet Protocol* (TCP/IP) em si, cabe uma breve explicação de que, conforme informado por Comer (1998), Comer (2007), Júnior, Suavé, Moura e Teixeira (1999) e Kaufman, Perlman e Speciner (2002), a organização conhecida como Organização Internacional para a Normalização (*International Organization for Standardization*, ISO) criou o modelo de referência *Open Systems Interconnection* (OSI), ou, de forma mais completa, modelo de referência *Reference Model - Open Systems Interconnection / International Organization for Standardization* (RM-OSI/ISO). Segundo Tanenbaum e Wetherall (2011), esse modelo se baseia em uma proposta desenvolvida pela ISO como um começo em direção à padronização internacional dos protocolos usados nas várias camadas. De acordo com Campbell (1997), Comer (1998), Comer (2007), Kaufman, Perlman e Speciner (2002) e Tanenbaum e Wetherall (2011), a ISO escolheu dividir o seu modelo de referência em sete camadas conceituais distintas. Sendo que, na Figura 10, é possível visualizar cada uma delas, comparando com as camadas do modelo TCP/IP, que será abordado posteriormente nesta seção.

Figura 10 - As quatro camadas conceituais do TCP/IP relacionadas às do RM-OSI/ISO



Fonte: Adaptado pelo autor com base em Kozierok (2005).

Porém, cabe informar que Comer (2007) destaca que as ideias sobre o projeto de protocolos tem mudado desde que o RM-OSI/ISO foi desenvolvido, fazendo com que muitos protocolos modernos não se encaixem no modelo. Isso é evidenciado por Kaufman, Perlman e Speciner (2002) e Tanenbaum e Wetherall (2011), ao observarem que as redes reais raramente se encaixam perfeitamente no modelo de sete camadas e que protocolos associados a esse modelo raramente são usados atualmente.

Fazendo um comparativo com o próximo modelo a ser abordado, Comer (1998) conta que o modelo RM-OSI/ISO, criado para descrever protocolos para uma única rede, não contém um nível específico de roteamento de interligação em redes igual ao dos protocolos do modelo TCP/IP. Comer (2007) justifica que o modelo de referência RM-OSI/ISO de sete camadas foi criado antes de a ligação inter-redes ser inventada, o que leva, por consequência, a este modelo RM-OSI/ISO não conter uma camada para protocolos para este fim. Além disso, o modelo de referência de sete camadas dedica uma camada inteira a protocolos de sessão, que se tornaram muito menos significativos à medida que os sistemas de computadores mudaram de grandes sistemas de tempo compartilhado para estações de trabalho pessoais.

Com estes problemas em mente, Campbell (1997) e Comer (2007) relatam que os pesquisadores do *Department of Defense* (Ministério da Defesa dos Estados Unidos) desenvolveram o TCP/IP, um novo modelo em camadas para tratar precisamente do problema

de conexão de redes com *hardwares* diferentes. Júnior, Suavé, Moura e Teixeira (1999) ainda informam que esse modelo evoluiu a partir de trabalhos iniciados no *Massachusetts Institute of Technology*, contando com a cooperação de várias empresas, e, de acordo com Soares, Lemos e Colcher (1995) sendo patrocinado pela *Defense Advanced Research Projects Agency*. Como Comer (1998) defende que esse modelo não surgiu através de uma comissão normatizada, e sim de pesquisas, Soares, Lemos e Colcher (1995) trazem à tona que os padrões da arquitetura TCP/IP não são elaborados por órgãos internacionais de padronização, como a ISO ou o IEEE.

Seguindo o que é dito por Júnior, Suavé, Moura e Teixeira (1999) e Tanenbaum e Wetherall (2011), este modelo ficou assim conhecido graças a seus dois principais protocolos, o *Transmission Control Protocol* (TCP) na camada de transporte e o *Internet Protocol* (IP) na camada de redes. Porém Júnior, Suavé, Moura e Teixeira (1999) informam que ele também é conhecido por pilha de protocolos *internet* e Comer (2007) diz que pode ser chamado de modelo inter-redes.

De acordo com Colcher (1995), Júnior, Suavé, Moura e Teixeira (1999), Soares, Lemos e Tanenbaum e Wetherall (2011) o modelo TCP/IP é composto por quatro camadas, apesar de Tanenbaum e Wetherall (2011) adotarem, para explicação dos protocolos, um modelo híbrido de 5 camadas. O que também é realizado por Comer (2007), ao citar que o modelo contém cinco camadas. No caso dos que abordam ele como tendo cinco camadas, é adicionada uma camada física anterior a camada de acesso à rede. Comer (1998) clarifica essa divergência ao explicar que, em tese, o *software* TCP/IP é organizado em quatro camadas conceituais construídas em uma quinta camada de *hardware*. A Figura 10 demonstra a estrutura dessas quatro camadas do modelo TCP/IP, relacionando-as com as sete camadas do modelo RM-OSI/ISO e fazendo um paralelo com uma possível quinta camada.

Comparando com o modelo RM-OSI/ISO, Tanenbaum e Wetherall (2011) explanam que ambos os modelos se baseiam no conceito de uma pilha de protocolos independentes e, além disso, as camadas deles tem praticamente as mesmas funcionalidades. Paralelamente, Soares, Lemos e Colcher (1995) trazem uma abordagem interessante ao citarem que os protocolos da arquitetura TCP/IP fornecem uma solução simples, porém bastante funcional, para o problema da interconexão de sistemas abertos, sendo que o fato de implementações de seus protocolos terem sido a primeira opção de solução não proprietária para a interconexão de sistemas fez com que essa arquitetura se tornasse um padrão *de facto*. Por outro lado, a estrutura organizacional da ISO, com membros representando vários países, se por um lado

aumenta o tempo de desenvolvimento dos padrões, por outro confere aos mesmos uma representatividade bem maior, tornando os padrões da ISO, por serem elaborados por uma instituição legalmente constituída para tal, padrões *de jure*.

Tendo como base todos estes aspectos acima relatados, a fim de especificar as funcionalidades de cada camada, será utilizado o modelo de referência TCP/IP.

2.2.4.1 Camada de acesso à rede

Serão tratadas aqui cada uma das camadas do protocolo TCP/IP individualmente, sendo que, conforme já abordado, para alguns autores existem cinco, para outros quatro. Os que abordam como sendo quatro tratam a camada física e de acesso à rede como fazendo parte da mesma camada, ou então não colocando a camada física como sendo parte do modelo.

Campbell (1997) e Comer (2007) e Tanenbaum e Wetherall (2011) abordam a camada física ao citarem cinco camadas, ao passo que Comer (2007) refere-se a ela como sendo a que corresponde ao *hardware* de rede básico. Estes fazem um paralelo com a camada física do modelo RM-OSI/ISO, chamada por Kaufman, Perlman e Speciner (2002) de *physical layer*, que para eles é a camada que entrega um fluxo de *bits* não estruturado através de uma ligação de algum tipo.

Assim, para Comer (2007), a camada física corresponde ao *hardware* de rede básico, ou, de acordo com Júnior, Suavé, Moura e Teixeira (1999), essa camada fornece as conexões elétricas necessárias, determinando as características para os sinais elétricos a serem usados nos meios físicos, ou, nas palavras de Campbell (1997), é nessa camada que se define qual será a interface entre o terminal e o equipamento de rede, tratando, segundo Tanenbaum e Wetherall (2011) da transmissão de *bits* normais por um canal de comunicação.

Continuando, dos que abordam a camada de acesso à rede no modelo TCP/IP, Comer (1998), Júnior, Suavé, Moura e Teixeira (1999) e Soares, Lemos e Colcher (1995) citam que ela também pode ser chamada de camada de interface de rede. Ao passo que Comer (1998) ainda informa que ela às vezes é denominada de camada de enlace de dados, assim como no modelo RM-OSI/ISO.

Comer (2007) informa que os protocolos da camada de acesso à rede especificam como organizar dados em quadros e como um computador transmite quadros através de uma rede. Já Tanenbaum e Wetherall (2011) explica que essa camada é a mais baixa do modelo e descreve o que os enlaces precisam fazer para cumprir os requisitos dessa camada de interconexão com serviço não orientado a conexões. Para Júnior, Suavé, Moura e Teixeira (1999) a camada de acesso à rede é a responsável pela transmissão de dados por meio de uma facilidade física comumente chamada de meio de comunicação, já Soares, Lemos e Colcher (1995) clarificam que é esse o nível que recebe os datagramas IP da camada de inter-redes e os transmite através de uma rede específica.

2.2.4.2 Camada de inter-redes

A camada de inter-redes pode também ser denominada, segundo Júnior, Suavé, Moura e Teixeira (1999), de camada de rede ou ainda, conforme Comer (2001), de camada de *internet*.

Comer (2007) explica que os protocolos da camada de inter-redes especificam o formato dos pacotes enviados através de uma rede, como também os mecanismos usados para encaminhar pacotes a partir de um computador através de um ou mais roteadores até o destino final, deste modo, Tanenbaum e Wetherall (2011) informam que a camada inter-redes integra toda a arquitetura da rede, mantendo-a unida.

Para Tanenbaum e Wetherall (2011) a tarefa desta camada é permitir que os *hosts* injetem pacotes em qualquer rede e garantir que eles trafegarão independentemente até o destino. Segundo Júnior, Suavé, Moura e Teixeira (1999), a camada de inter-redes é a responsável por descrever as tecnologias para interligação de redes, administrando o fluxo de pacotes por meio das mesmas, enquanto que Comer (1998), por sua vez descreve de forma mais detalhada que essa camada aceita um pedido para enviar um pacote originário da camada de transporte juntamente com uma identificação da máquina para qual o pacote deve ser enviado e encapsula o pacote em um datagrama IP, preenchendo o cabeçalho do datagrama e usando o algoritmo de roteamento para decidir se entrega o datagrama diretamente ou o envia para um roteador e passa o datagrama para a interface de rede apropriada para a transmissão. Ainda segundo Comer (1998) a camada de inter-redes também lida com datagramas de entrada, verificando sua validade, e usa o algoritmo de roteamento para decidir se o datagrama deve ser processado no local ou deve ser enviado.

Essa camada de inter-redes é relacionada, no modelo de referência RM-OSI/ISO, segundo Campbell (1997) e Comer (2007) Tanenbaum e Wetherall (2011), a camada de rede, sendo também chamada, seguindo Kaufman, Perlman e Speciner (2002), de *network layer*. Conforme Comer (2007), no modelo RM-OSI/ISO, a camada de redes é a que possui os protocolos que especificam como são atribuídos endereços e como são encaminhados pacotes de uma ponta a outra da rede.

2.2.4.2.1 IP

Para entendimento futuro dos ataques realizados, é interessante que seja explicado de forma introdutória o protocolo IP. Comer (2007) introduz que na pilha de protocolos TCP/IP, o endereço é especificado pelo Protocolo de *Internet* (IP). Segundo Júnior, Suavé, Moura e Teixeira (1999) o protocolo IP é o protocolo mais importante da pilha TCP/IP na camada de inter-redes, o que leva Tanenbaum e Wetherall (2011) a citar que a tarefa desta camada é justamente a de entregar pacotes IP onde eles são necessários.

Para compreender o endereçamento IP, de acordo com Chowdhury (2002), é importante estar familiarizado com o sistema binário e com a conversão de binário para decimal. Desse modo, ele informa que o sistema binário é representado com zero (0) ou um (1), e cada um desses valores é conhecido como *bit*. Adentrando na explicação, Chowdhury (2002) informa que oito dessas combinações de *bits* gera um "*byte*", que também é conhecido como octeto, clarificando que o valor de um *byte* pode variar de 0 a 255 em decimal, ou de 00000000 a 11111111 em binário.

Com isto em mente, podemos trazer as informações de Chowdhury (2002) e Comer (2007) de que o protocolo IP especifica que a cada *host* é atribuído um número de 32 *bits* único, conhecido como endereço de Protocolo de *Internet*, que é frequentemente abreviado de endereço IP e normalmente é representado como quatro valores decimais. Já que cada notação decimal corresponde a um *byte*, que representa o intervalo de 0 a 255, Comer (2007) certifica que os endereços IP em decimal pontilhada variam de 0.0.0.0 a 255.255.255.255. Porém é importante ressaltar que Chowdhury (2002) e Tanenbaum e Wetherall (2011) informam que esse endereço de 32 *bits* é conhecido como *Internet Protocol version 4* (IPv4), sendo que existe também endereços IP de 128 *bits* nos pacotes *Internet Protocol version 6* (IPv6).

A finalidade do IP, segundo Chowdhury (2002), é transportar datagramas por meio de um conjunto de interligação de redes, sendo que Kaufman, Perlman e Speciner (2002) citam que o seu trabalho é o de entregar dados através de uma rede. Comer (2007) explica que cada pacote enviado através de uma rede contém o endereço de IP do remetente (origem), bem como o receptor pretendido (destino).

Também é importante explicar do que se trata uma máscara de sub-rede. De acordo com Comer (2007) a máscara de endereço (*address mask*), assim como o endereço IP, também é armazenada como um valor binário de 32 *bits*, sendo que ela tem a função de separar o prefixo de rede com o 1 e marcar a porção do *host* com o 0. Tanenbaum e Wetherall (2011) esclarecem que esta separação é realizada submetendo a máscara a um AND com o endereço IP a fim de extrair apenas a parte da rede do endereço IP, conforme Comer (2007), a partir de um rápido cálculo de um roteador, definindo a rede ou sub-rede de um determinado endereço IP.

2.2.4.3 Camada de transporte

Comer (2007) define que os protocolos da camada de transporte especificam como assegurar a transmissão confiável, sendo que, para Comer (1998), a função da camada de transporte é prover a comunicação de um programa aplicativo para outro enquanto que para Soares, Lemos e Colcher (1995), a função básica da camada de transporte é permitir a comunicação fim a fim entre aplicações, ao passo que para Tanenbaum e Wetherall (2011) a finalidade dessa camada é permitir que as entidades pares dos *hosts* de origem e de destino mantenham uma conversação, o que é fornecido, segundo Comer (1998) através de transporte confiável, assegurando que os dados cheguem sem erros e em sequência.

Conforme Júnior, Suavé, Moura e Teixeira (1999), a camada de transporte retrata as tecnologias para o estabelecimento de conexões fim a fim e suporta o fluxo de dados entre dois *hosts*, garantindo a qualidade do serviço para a camada de aplicação, sendo que, de acordo com Júnior, Suavé, Moura e Teixeira (1999) e Tanenbaum e Wetherall (2011), os dois protocolos de ponta a ponta mais importantes definidos aqui são o TCP e o *User Datagram Protocol* (UDP).

Essa camada de transporte do modelo TCP/IP tem praticamente as mesmas funções da camada de mesmo nome do modelo RM-OSI/ISO, sendo que Kaufman, Perlman e Speciner

(2002) explicam que essa camada às vezes é chamada de *transport layer* e ela existe para estabelecer um fluxo de comunicação confiável entre um par de sistemas através de uma rede, colocando números de sequência em pacotes, segurando pacotes no destino até que eles possam ser entregues em ordem, e retransmitindo pacotes perdidos.

2.2.4.4 Camada de aplicação

Na camada de aplicação, de acordo com Comer (1998) e Soares, Lemos e Colcher (1995), os usuários usam programas aplicativos que acessam serviços disponíveis através de uma interligação em redes TCP/IP. Já nas palavras de Júnior, Suavé, Moura e Teixeira (1999), a camada de aplicação descreve as tecnologias usadas para fornecer serviços especializados para os usuários e administra os detalhes de uma aplicação em particular, assim para Comer (2007), cada protocolo dessa camada especifica como um aplicativo usa uma rede.

Comer (2007) esclarece que a camada de aplicação do modelo de referência TCP/IP corresponde às camadas de sessão, apresentação e aplicação do modelo de referência RM-OSI/ISO. Segundo Tanenbaum e Wetherall (2011) o modelo TCP/IP não tem as camadas de sessão ou de apresentação, pois não foi percebida qualquer necessidade para elas, já que, ao invés disso, as aplicações simplesmente incluem quaisquer funções de sessão e apresentação que forem necessárias.

2.3 Ameaças existentes na camada de enlace de dados

Nesta seção serão abordados vários ataques existentes na camada de enlace de dados encontrados, especificando como estes ataques são realizados e quais são os métodos de prevenção para mitigar estes ataques. Os ataques levantados consistem em ataques de falsificação de endereço MAC (*MAC spoofing*), *MAC address table overflow*, ataques ao serviço DHCP, dentre eles *DHCP starvation* e *rogue DHCP server*, ataque ao *Address Resolution Protocol* (*ARP spoofing*), tempestade de *broadcast* (*broadcast storm*), ataque ao *spanning tree* e ataque de *VLAN hopping* utilizando *switch spoofing* e *double tagging*.

2.3.1 Falsificação de endereço MAC (*MAC spoofing*)

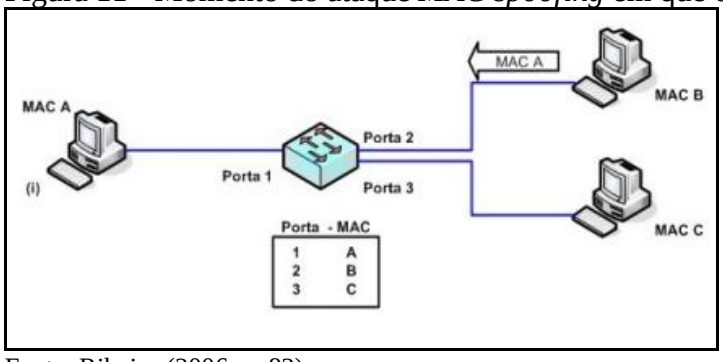
Para entendimento do primeiro ataque relacionado, a seção de revisão bibliográfica abordou anteriormente os fundamentos necessários, principalmente na seção 2.2.2.1.1 que trata sobre as particularidades do *switch* e da tabela CAM, porém cabe frisar novamente que, conforme Omar, Pinto e Saide (2013), um *switch* constrói e mantém dinamicamente uma tabela CAM ou FDB, conservando todas as informações MAC necessárias para cada porta, a fim de escolher para qual porta um determinado quadro deve ser encaminhado.

Com isto em mente, será tratado do ataque de falsificação de endereço MAC (*MAC spoofing*). Kurose e Ross (2010) levantam vagamente que a habilidade de introduzir pacotes na rede com uma fonte falsa de endereço MAC é conhecida como *MAC spoofing*, e é uma das muitas maneiras pelas quais o usuário pode se passar por outro, enquanto que nas palavras de Stallings (2008), a falsificação do endereço MAC usa endereços de MAC forjados para enganar um *host* para aceitar dados falsos. Corrêa (2009) adentra essas definições informando que o *MAC spoofing* é um tipo de ataque utilizado para substituir uma entrada na tabela CAM que contém um endereço MAC conhecido que aponta para uma determinada porta, fazendo com que ele aponte para outra porta, sendo que esta será comumente a que o atacante está conectado.

Pode ser constatado por Ribeiro (2006) que o *switch* vai apagar a entrada antiga e adotar uma nova entrada como verdadeira ao receber o mesmo endereço MAC em uma porta diferente. Sendo que Ribeiro (2006) descreve todo o processo desse ataque a partir dos seguintes passos retratados nas figuras posteriores.

Conforme Ribeiro (2006), na Figura 11 o atacante, neste caso a estação MAC B, dispara na rede um quadro com a falsa informação que o seu endereço MAC é o da estação MAC A.

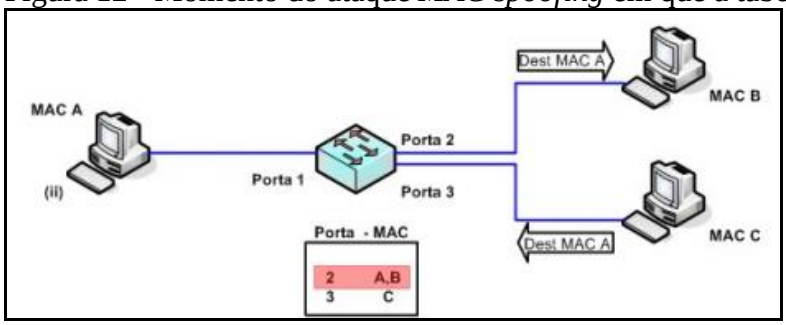
Figura 11 - Momento do ataque *MAC spoofing* em que o atacante lança na rede um pacote



Fonte: Ribeiro (2006, p. 82).

Continuando o ataque, Ribeiro (2006) informa que na Figura 12 o *switch* faz a atualização da sua tabela CAM com a informação falsa. Com isso, agora todo quadro enviado para o MAC da estação MAC A será encaminhado para a porta 2 do *switch*.

Figura 12 - Momento do ataque *MAC spoofing* em que a tabela CAM é atualizada



Fonte: Ribeiro (2006, p. 82).

É informado por Corrêa (2009) que através desse ataque é possível causar uma interrupção de serviços, também conhecida como *Denial of Service* (DoS). O DoS, segundo Comer (2007), é uma forma de ataque de rede na qual um serviço é sobrecarregado com tantos pacotes (usualmente de requisições incorretas ou fictícias) que ele não consegue responder para legitimar uma requisição. Já para Nakamura e Geus (2007), os ataques de negação de serviços (DoS) fazem com que recursos sejam explorados de maneira agressiva, de tal modo que usuários legítimos ficam impossibilitados de utilizá-los, ao passo que, para Forouzan (2010), o ataque de negação de serviços é aquele no qual um invasor monopoliza um sistema com tantas solicitações de serviço que ele acaba entrando em colapso e nega atendimento a todas as outras solicitações, enquanto que Stallings (2008), por sua vez, explica que um ataque de negação de serviço é uma tentativa de impedir que usuários legítimos de um serviço utilizem esse serviço.

Além do DoS, este ataque também pode ser utilizado como um ataque *man-in-the-middle*, que segundo Nakamura e Geus (2007) se trata de um ataque onde o atacante se coloca entre o usuário e o servidor, de modo que pode capturar os pacotes, modificá-los e reenviá-los para ambos os lados da conexão. Já para Corrêa (2009), este tipo de ataque, também conhecido como ataque do homem do meio, é uma situação na qual o atacante intercepta a comunicação entre dois *hosts* e falsifica as mensagens a fim de fazer-se passar por uma das partes.

2.3.1.1 Métodos de prevenção para a falsificação de endereço MAC (*MAC spoofing*)

Corrêa (2009) explica que esse tipo de ataque só pode ser bloqueado em *switches* gerenciáveis que possuem ferramentas de segurança para tal. Como por exemplo, o *port security* de alguns fabricantes de *switches* específicos, que segundo Watkins e Wallace (2008) é um modo onde um *switch* da Cisco, por exemplo, aprende de forma dinâmica os endereços MAC conectados a várias portas, sendo que estes endereços MAC aprendidos dinamicamente são adicionados a configuração do *switch*, evitando assim que um atacante realize *spoofing* de um endereço previamente aprendido.

2.3.2 *MAC address table overflow*

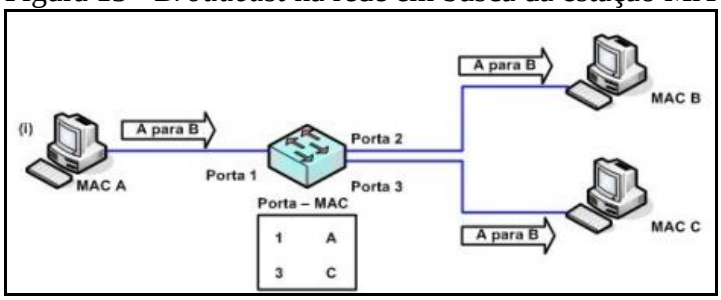
Na seção 2.2.2.1.1 é explicado o funcionamento da tabela CAM, porém Corrêa (2009) traz que um dos principais problemas de segurança, compreendendo o endereçamento da camada de enlace, é o *overflow* justamente dessa tabela CAM dos *switches*. Neste ataque denominado de *MAC address table overflow*, segundo Ribeiro (2006), um atacante pode provocar uma sobrecarga na tabela CAM enviando a uma porta *switch* um grande número de quadros com endereços MAC criados randomicamente, dessa forma causando uma inundação da tabela CAM.

Corrêa (2009) explica que fisicamente a tabela CAM é armazenada em uma memória normal e como tal, dispõe de tamanho limitado. Percebendo isso, em 1999, Ian Vitek criou uma ferramenta chamada "*macof*", que alterna inundações de endereços MAC de origem inválidos (cerca de 155000 por minuto). Essa ferramenta é capaz de encher rapidamente a tabela CAM do *switch* que está diretamente conectado ao *host* encarregado pela execução da ferramenta, além de também afetar os *switches* que estiverem conectados ao equipamento atacado.

De acordo com Nakamura e Geus (2007), esse envio de quadros à rede (*flooding*) usando endereços MAC ainda não utilizados que torna a tabela CAM cheia, faz com que o *switch* passe a atuar do modo *switch* para o modo *hub*. O que é reforçado por Corrêa (2009), que cita que o resultado desse ataque é um comportamento adotado pelo *switch* quando ele não consegue localizar um endereço em sua tabela, enviando os quadros recebidos para todas as suas portas e passando a se comportar como um *hub*. Para mais informações, essa diferença no funcionamento do *hub* e de um *switch* é fundamentada na seção 2.2.2.1.1 deste documento.

Seguindo a o que é explanado por Ribeiro (2006), primeiramente será ilustrada uma situação normal de uma comunicação entre dois pontos passando por um *switch* através do que é representado nas Figuras 13, 14 e 15. Na Figura 13 é possível visualizar o começo do processo, no caso, se o *switch* não conseguir localizar o endereço físico de destino (estação MAC A) em sua tabela CAM ele emite um *broadcast* para todas as portas do *switch*.

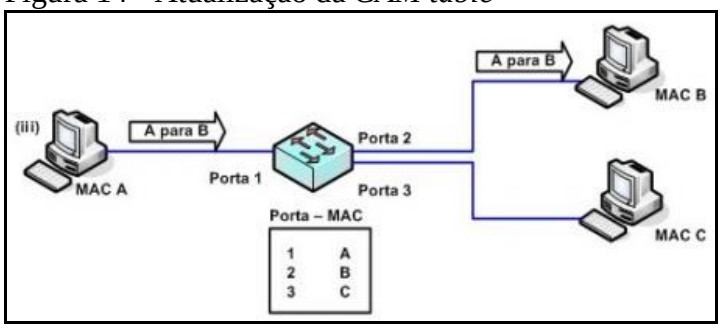
Figura 13 - *Broadcast* na rede em busca da estação MAC B



Fonte: Ribeiro (2006, p. 79).

Seguindo a explicação de Ribeiro (2006), na Figura 14 a estação MAC B recebe o *broadcast* e responde para a estação MAC A. O *switch* identifica que a estação MAC B está conectada na porta 2 e então atualiza sua tabela CAM.

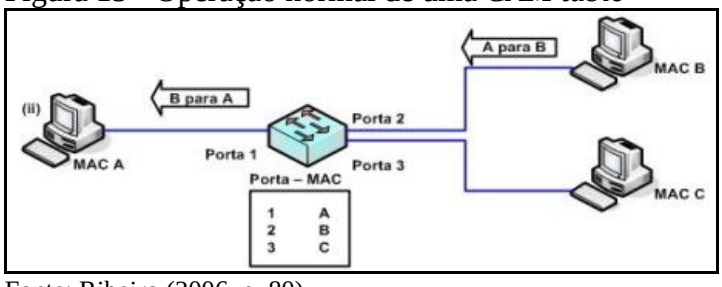
Figura 14 - Atualização da *CAM table*



Fonte: Ribeiro (2006, p. 79).

Finalizando a situação normal, Ribeiro (2006) traz, na Figura 15, que qualquer quadro *Ethernet* que a estação MAC A ou a estação MAC C endereçarem para a estação MAC B, será encaminhado diretamente para a porta 2, sem a necessidade de um *broadcast*.

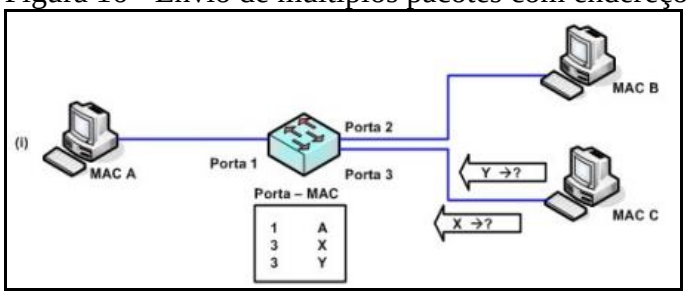
Figura 15 - Operação normal de uma *CAM table*



Fonte: Ribeiro (2006, p. 80).

Agora que a situação normal foi explicada seguindo o que é ilustrado por Ribeiro (2006), as Figuras 16 e 17 exemplificando o processo de um ataque *MAC address table overflow*. Começando, na Figura 16 o atacante envia milhares de quadros com endereços MAC falsos de origem aleatórias para portas com endereços MAC válidos da rede.

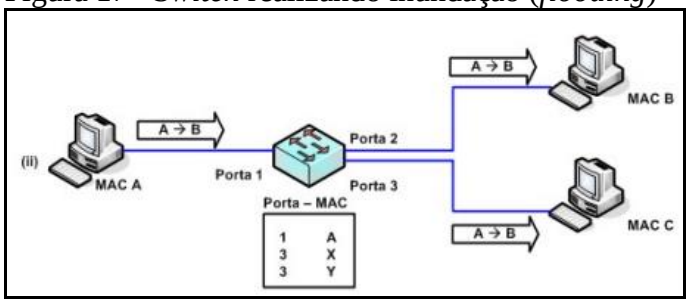
Figura 16 - Envio de múltiplos pacotes com endereços MAC aleatórios



Fonte: Ribeiro (2006, p. 80).

Concluindo o ataque, na Figura 17 Ribeiro (2006) demonstra que, por consequência a tabela CAM é inundada com a grande quantidade de endereços MAC e estoura sua capacidade de armazenamento passando a enviar todos os quadros trafegados na rede para todas as portas do *switch*, fazendo com que ele funcione como um *hub*.

Figura 17 - *Switch* realizando inundação (*flooding*)



Fonte: Ribeiro (2006, p. 81).

Em conformidade com Tanenbaum e Wetherall (2011), espiões e bisbilhoteiros adoram esse comportamento, já que, segundo Corrêa (2009), ele permite realizar a espionagem dos pacotes e, até mesmo, possibilitar um ataque de *man-in-the-middle* monitorando o tráfego da rede.

Omar, Pinto e Saide (2013) explicam que, como resultado deste ataque, alguns *switches* podem responder de duas maneiras. Na primeira, eles podem entrar em um modo "*fail-open*", operando em modo promíscuo e enviando o tráfego para todas as portas, como se fosse um *hub*, dando condições ao invasor de fazer um *sniffer* (interceptar e analisar o tráfego de dados) na rede capturando os pacotes que são enviados de um *host* a outro, dados da topologia de rede, configurações de equipamentos, serviços, dentre outras informações.

Na segunda maneira, Mangueira (2015) argumenta que alguns *switches* podem ser configurados como "*fail-closed*", sendo que neste modo os *switches* funcionam exatamente do modo inverso ao de um *switch "fail-open"*. Assim, em vez de efetuar o *broadcast* de todo o tráfego para todas as portas ele simplesmente para de encaminhar todo o tráfego, o que faz com que deva-se avaliar os riscos ao se escolher este tipo de configuração, pois um ataque a tabela CAM irá gerar uma negação de serviço, também conhecido como DoS, que segundo Omar, Pinto e Saide (2013), faz com o *switch* simplesmente pare de repassar os quadros.

2.3.2.1 Métodos de prevenção de *MAC address table overflow*

Segundo Nakamura e Geus (2007), esses ataques podem ser restringidos com o uso de listas de controle de acesso (*Access Control List*, ACL) baseados em endereços MAC também é recomendável, bem como o uso de tabelas *Address Resolution Protocol* (ARP) estática. Porém, Nakamura e Geus (2007) ressaltam que essa medida depende de uma avaliação quanto à escalabilidade e à carga administrativa gerada.

Corrêa (2009) resume que para resolver esse problema é preciso utilizar *switches* gerenciáveis que disponham de ferramentas de segurança para controlar os dados que trafegam por cada porta, além disso, deve ser realizado um monitoramento refinado dos dados que trafegam pela rede a fim de detectar o uso não autorizado de ferramentas de monitoramento. Sendo que esse ataque também pode ser mitigado com o uso do *port security* existente em alguns *switches*, assim como o da seção 2.3.1.1, ao passo que é explanado por Watkins e Wallace (2008) que o *port security* permite que o administrador do *switch* especifique o número máximo de endereços MAC que podem ser aprendidos em uma porta, impedindo assim um ataque de *overflow* da tabela CAM.

2.3.3 Ataques ao serviço DHCP

No que tange o protocolo afetado por este ataque, Comer (2007) introduz que, embora a especificação manual de IPs funcione quando um conjunto de computadores permanece fixo, ela não é suficiente se o conjunto de computadores muda rapidamente, o que leva Tanenbaum e Wetherall (2011) a observar que configurar manualmente cada computador é tedioso e passível de erros. Conforme é informado por Comer (2007), para gerenciar tais casos, foi desenvolvido o protocolo de configuração de *host* dinâmico (*Dynamic Host*

Configuration Protocol, DHCP) que propaga e automatiza a configuração de endereços IP, fornecendo um mecanismo que possibilita que um computador se junte a uma nova rede e obtenha um endereço IP automaticamente.

Para que o DHCP entre em funcionamento, Tanenbaum e Wetherall (2011) recomendam que cada rede precisa ter um servidor DHCP responsável pela configuração. Assim, ainda segundo Tanenbaum e Wetherall (2011), quando um computador é iniciado, ele tem um endereço MAC, mas não um endereço IP, o que faz com que, conforme Comer (2007) e Forouzan (2010), este computador difunda por *broadcast* uma *DHCP request*, também conhecida como *DHCP discover*, para qual um servidor envia uma *DHCP reply*, também conhecida como *DHCP offer*, ou seja, envie uma requisição DHCP, então o servidor DHCP consulta a sua base de dados para encontrar informações de configuração. A partir daí, se a base de dados possui uma entrada específica para o computador, o servidor retorna as informações da entrada, porém se nenhuma entrada existe para o computador, o servidor DHCP escolhe o próximo endereço IP disponível do conjunto e designa esse endereço ao computador.

Kurose e Ross (2010) ainda informam que um administrador de rede pode configurar o DHCP de modo que um determinado *host* receba o mesmo endereço IP toda vez que se conectar à rede ou que um *host* receba endereço IP temporário diferente sempre que se conectar à rede. Porém, Comer (2007) e Forouzan (2010) destacam que, na verdade, endereços fornecidos pelo DHCP não são permanentes e, ao invés disso, ele aluga (*lease*) o endereço por um endereço finito de tempo, o que de acordo com Tanenbaum e Wetherall (2011), serve para impedir que endereços de rede se percam. Neste processo de *leasing*, Comer (2007), Forouzan (2010) e Tanenbaum e Wetherall (2011) adicionam que quando a locação expira, o cliente tem de parar de usar o endereço IP ou então solicitar a renovação da locação, sendo que o servidor, por sua vez, tem a opção de concordar ou não com essa renovação. Se o servidor não concordar ou o cliente deixar de fazer uma solicitação, o cliente para de usar o endereço e retorna-o para o conjunto de endereços disponíveis, o que permite que este endereço seja atribuído a outro computador.

Segundo Forouzan (2010), o aspecto dinâmico do DHCP é necessário quando um *host* muda de uma rede para outra ou é conectado e desconectado de uma rede. Já o aluguel, para Comer (2007), é essencial para o funcionamento contínuo de um servidor pois permite que este servidor controle recursos e recupere endereços.

Tendo entendido o funcionamento do DHCP, serão abordados os problemas de segurança que envolvem esse serviço, sendo eles o *DHCP starvation* e o *rogue DHCP server*.

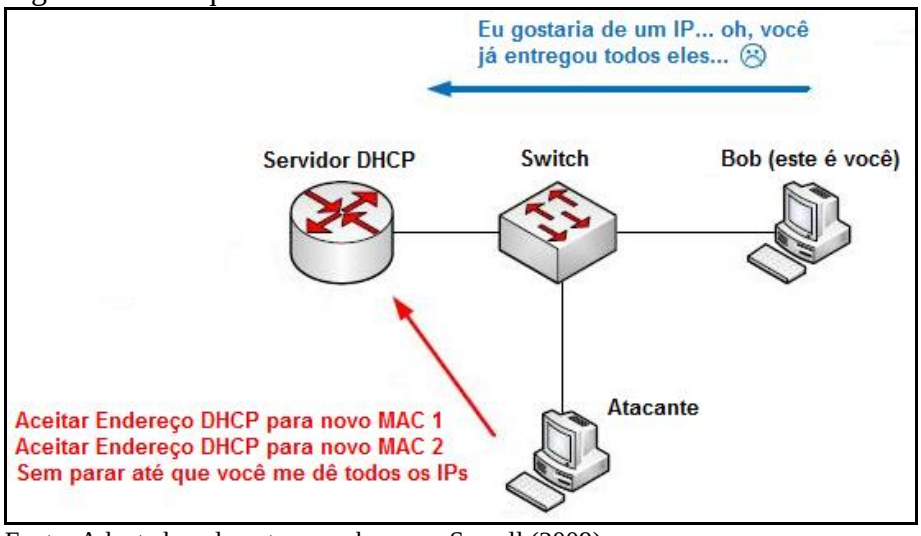
2.3.3.1 *DHCP starvation*

O primeiro dos ataques ao serviço de DHCP é o DHCP *starvation*, que conforme Omar, Pinto e Saide (2013), consiste em alocar todos os endereços IP disponíveis no servidor DHCP, onde o atacante de maneira simples, através de um *broadcast* usando requisições DHCP com endereços MAC falsos, obtém sucesso. Isso leva à negação de serviço na rede para os clientes habituais, abrindo uma brecha para os atacantes configurarem um servidor DHCP falso, e através dele enviar informações falsas para os clientes.

Ribeiro (2006) explica que este ataque pode ser implementado em servidores DHCP configurados para atribuir endereços dentro de uma faixa específica, onde um cliente não autorizado (atacante) gera um grande número de pedidos consecutivos usando endereços MAC diferentes, esgotando assim os endereços IP disponíveis nessa faixa. Esgotada a faixa de endereços, o servidor DHCP não será mais capaz de atribuir endereços solicitados por novos clientes enquanto não terminar o tempo de validade das configurações já atribuídas.

A Figura 18 ilustra um ataque de DHCP *starvation*. Nele o atacante realiza um número grande de requisições de DHCP com endereços MAC falsos até o servidor DHCP (neste caso um roteador) não possuir mais IPs disponíveis, fazendo com que os demais computadores, representados por “Bob”, fiquem sem endereços IPs para acessarem a rede.

Figura 18 - Ataque de *DHCP Starvation* sendo executado



Fonte: Adaptado pelo autor com base em Sowell (2009).

2.3.3.2 *Rogue DHCP server*

Com o ataque de *DHCP starvation* tendo sido realizado, Omar, Pinto e Saide (2013) explicam que abre a possibilidade de inserir na rede um servidor DHCP falso que será capaz de distribuir configurações adulteradas. Este ataque de servidor DHCP falso, que é o segundo a ser abordado no serviço DHCP, pode ser também descrito, conforme Banks (2012) e Watkins e Wallace (2008), como *Rogue DHCP Server*.

Ribeiro (2006) expõe que as atribuições de configurações inválidas podem provocar na rede a perda de conectividade dos clientes, atribuição de um IP de *gateway* (nó que se presume saber como encaminhar pacotes para outras redes) de uma estação pertencente ao atacante, fazendo com que os clientes encaminhem suas comunicações com destino a outra rede para a estação do atacante, permitindo a captura de dados. O que ainda é reforçado por Corrêa (2009), ao informar que a partir do momento que os clientes aceitarem as novas configurações DHCP, todo o tráfego da rede passará pela máquina do atacante, tornando assim muito fácil o monitoramento das informações e configurando em um ataque *man-in-the-middle*.

Segundo Omar, Pinto e Saide (2013), este ataque pode ser realizado em servidores que estejam configurados tanto para configuração por faixa de endereços quanto por reserva de configuração para endereços MAC específicos. Conforme Ribeiro (2006), no servidor que fornece uma faixa de endereços, tirando proveito da limitação de endereços IP disponível para atribuição a clientes DHCP, é possível trocar o servidor de DHCP legítimo por um servidor de DHCP controlado por um atacante. Ribeiro (2006) detalha que, na execução deste ataque o atacante pode primeiro realizar um ataque de DoS no servidor DHCP da rede, evitando assim a concorrência quando novos clientes na rede requisitarem configuração, já que como os pedidos são realizados em modo de difusão na rede, existe a possibilidade do cliente optar pela configuração oferecida pelo servidor legítimo da rede. Após realizado o ataque de DoS, é colocado em funcionamento o servidor DHCP do atacante que responderá aos pedidos de configuração dos clientes, porém atribuindo configurações diferentes as quais atenderão os interesses do atacante.

Já no servidor DHCP que reserva configurações para endereços MAC específicos, Omar, Pinto e Saide (2013) explicam que o ataque faz com que o servidor legítimo continue operando e o servidor falso seja conectado à rede. Lembrando que, no momento em que alguns clientes fazem pedidos de DHCP, estes são executados em *broadcast* pela rede e são

recebidos por todos os elementos dela. Portanto, os dois servidores DHCP, o legítimo e o falso, respondem aos pedidos quase que simultaneamente, e, neste caso, cabe ao cliente escolher uma das ofertas, sendo que geralmente a escolha é feita por aquela que for recebida primeiro. Embora este ataque não seja eficaz em todos os casos, vários clientes irão optar pela configuração fornecida pelo servidor falso.

Além desse problema ocorrer de forma proposital e maliciosa por um atacante, Brito (2013) evidencia que outro detalhe importante é que atualmente esse tipo de ataque ficou muito comum de ser executado de maneira involuntária, sem a intenção de atacar a rede, por um usuário menos experiente que negligentemente conecta um equipamento servidor de DHCP, como um desses roteadores *wireless* residenciais por exemplo, no ambiente corporativo para compartilhar a rede com seus dispositivos móveis. No caso citado de roteadores residenciais, cabe mencionar que eles normalmente estão com o serviço DHCP habilitado para tornar o equipamento *plug-and-play* (ligar e usar) para os usuários domésticos, sendo ainda que esse roteador residencial pode ser facilmente carregado e entrar despercebidamente na empresa, bastando o usuário conectá-lo na rede para executar involuntariamente um ataque de *rogue DHCP server* na rede.

2.3.3.3 Métodos de prevenção para ataques ao serviço DHCP

Sendo proposital ou não, estes ataques ao DHCP devem ser fortemente combatidos já que, como visto, geram graves problemas para os usuários finais, sendo que Omar, Pinto e Saide (2013) apontam que se proteger de ataques de *DHCP starvation* só se torna possível através de *switches* inteligentes que disponham de ferramentas de segurança que não permitam mais de um endereço MAC em uma porta.

Já se tratando de ataques de *rogue DHCP server*, é citado por Banks (2012), Brito (2013) e Watkins e Wallace (2008) que estes podem ser facilmente mitigados em alguns modelos de *switches* gerenciáveis através de um recurso denominado *DHCP snooping*. É através desse recurso de *DHCP snooping* que, conforme Banks (2012), Brito (2013) e Watkins e Wallace (2008), o administrador da rede configura a porta em que o servidor legítimo está conectado como sendo confiável (*trusted*) e todas as demais passam a não ser confiáveis (*untrusted*).

2.3.4 Ataque ao protocolo ARP (*ARP spoofing*)

O *Address Resolution Protocol* (ARP), segundo Chowdhury (2002), Corrêa (2009), Comer (1998), Comer (2007) e Tanenbaum e Wetherall (2011) é um dos protocolos mais utilizados para mapear endereços IP para endereço MAC, além de ser essencial para a comunicação no TCP/IP. Dentro de um mesmo segmento de rede, computadores trocam mensagens ARP uns com os outros para descobrir o endereço MAC baseados no endereço IP que possuem, assim, o ARP permite que um *host* encontre o endereço físico de um *host* de destino na mesma rede, se o endereço IP do destino for conhecido. Chowdhury (2002) explica que neste processo a origem enviará uma requisição ARP, geralmente um *broadcast*, para cada nó na rede.

Porém, Omar, Pinto e Saide (2013) expõe que, por ser um protocolo muito simples ele não tem muitos recursos de segurança, o que, segundo Corrêa (2009), acaba abrindo uma brecha para possíveis ataques, sendo que o mais conhecido destes ataques é o *ARP spoofing*, e consiste no envio de endereços IP ou MAC falsos causando ataques como a negação de serviços e o *man-in-the-middle*. Portanto, conforme alertado por Omar, Pinto e Saide (2013), os ataques de ARP podem deixar uma rede inteira sem comunicação, forçando o envio de mensagens ARP aleatórias e provocando erros de endereçamento incorreto nas estações.

Adentrando no princípio envolvido na técnica *ARP spoofing*, Ribeiro (2006) explica que ela consiste em enviar para a rede informações falsificadas de pacotes ARP sobre os endereços MAC e IP mantidos por cada sistema, enganando as estações de uma rede com relação ao endereço MAC de um determinado destino. Esta técnica tem como base a notificação de endereços falsos para dispositivos de rede, via *broadcast*, podendo ser utilizada, por exemplo, para fazer com que um atacante se coloque na posição de um *gateway* de uma rede, mesmo em rede segmentadas com uso de *switches*. Nakamura e Geus (2007) reforça que o envio de quadros com os endereços ARP falsos (*ARP spoofing*) faz com que o tráfego de outros equipamentos seja enviado para o equipamento do atacante, que captura os quadros e os redireciona para o equipamento verdadeiro, que nem percebe a diferença.

Omar, Pinto e Saide (2013) e Ribeiro (2006) ainda informam que a técnica pode ser aplicada devido ao ARP não possuir qualquer método de autenticação e, principalmente, de aceitação de mensagens não requisitadas para a atualização de ARP dos *hosts* da rede. Desta maneira as mensagens ARP falsas enganam os dispositivos de rede entregando os dados em

portas incorretas do *switch*, permitindo que o atacante tenha as informações destinadas a um sistema na rede, a vítima, direcionadas à sua porta de na rede.

2.3.4.1 Métodos de prevenção para o *ARP spoofing*

Assim como no ataque de *MAC spoofing*, na seção 2.3.1.1, Nakamura e Geus (2007) citam esses ataques podem ser restringidos com o uso de listas de controle de acesso (ACL) baseados em endereços MAC, bem como com o uso de tabelas ARP estáticas, apesar destas medidas dependerem de uma avaliação quanto à escalabilidade e à carga administrativa gerada.

Paralelamente Corrêa (2009) informa que uma das formas de se evitar estes ataques é adicionando os pares IP/MAC manualmente na tabela ARP dos computadores na rede, o que ajudaria a eliminar alguns cenários de ataque ARP *spoofing*, mas está longe de ser uma solução viável, uma vez que, em redes de grande porte, seria praticamente impossível manter um controle adequado.

Por fim, Bhaiji (2008) e Watkins e Wallace (2008) informam que alguns *switches* possuem recursos como o *Dynamic ARP Inspection* (DAI) e o *IP Source Guard* que podem ser utilizados para mitigar tanto o *ARP spoofing*, como o *MAC e IP spoofing*.

Watkins e Wallace (2008) explicam que o DAI trabalha de forma semelhante ao *DHCP snooping* usando portas confiáveis (*trusted*) e não confiáveis (*untrusted*). Respostas ARP são permitidas no *switch* em portas confiáveis (*trusted*), entretanto, se uma resposta ARP entra no *switch* em uma porta não confiável (*untrusted*), o conteúdo da resposta ARP será comparado à tabela de ligação do DHCP para verificar sua precisão. Se a resposta ARP for inconsistente com esta tabela de ligação DHCP ela será descartada e a porta será desativada.

Já o *IP Source Guard*, segundo Cisco (2013), fornece filtragem de endereços IP de origem em portas de camada de enlace de dados para prevenir que um *host* malicioso se passe por um *host* legítimo assumindo o endereço de IP desse *host* legítimo. O recurso usa *DHCP snooping* dinâmico e vínculo de fonte de IP estático para conciliar o endereço IP para *hosts* em portas de acesso não confiáveis (*untrusted*).

2.3.5 Tempestade de *broadcast* (*broadcast storm*)

Conforme já visto na seção 2.2.2.1.2, os pacotes de *broadcast* são comuns em todas as redes e utilizados por diversos protocolos, porém, segundo Cisco (2013), um nível desmoderado destes pacotes pode causar tráfego excessivo e degradação na rede, podendo até parar completamente a transmissão, desta forma impossibilitando o uso dos recursos de rede.

Conforme Tanenbaum e Wetherall (2011), um dos problemas relacionados ao *broadcast* é que, de vez em quando, uma interface de rede irá enfrentar uma pane e começará a gerar um fluxo infinito de quadros de *broadcast*. E se a rede realmente estiver sem sorte, alguns desses quadros gerarão respostas que trarão ainda mais tráfego. O resultado dessa tempestade de *broadcast* é que a capacidade da LAN inteira será ocupada por esses quadros e todas as máquinas em todas as LANs interconectadas serão danificadas, pois estarão processando e descartando todos os quadros que estiverem sendo transmitidos. Nas palavras de Kurose e Ross (2010), essa tempestade de *broadcast*, também conhecida como *broadcast storm*, gerada pela infindável multiplicação de pacotes *broadcast*, resulta na criação de uma quantidade tão grande de pacotes *broadcast* que a rede fica completamente inutilizável.

Watkins e Wallace (2008) informam que esse excesso de *broadcast* pode ser gerado por problemas de configuração, placas de rede defeituosas ou pela execução de um ataque de negação de serviço (DoS). Com isto em vista, Cisco (2013) cita que no caso de um ataque, pacotes de *broadcast* são enviados de maneira excessiva em uma mesma VLAN, o que levará o *switch* a utilizar ao máximo a sua capacidade de processamento, podendo prejudicar a performance substancialmente e até mesmo causar a interrupção dos serviços de rede.

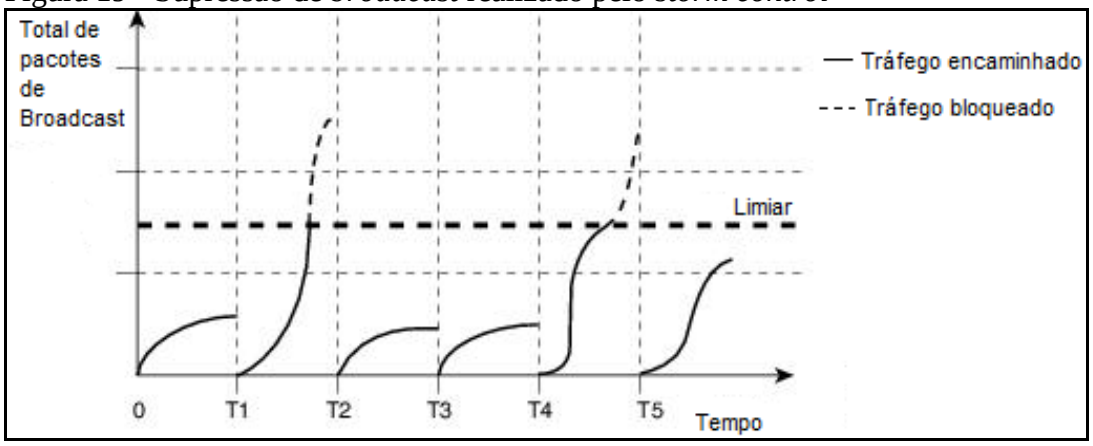
2.3.5.1 Métodos de prevenção para a tempestade de *broadcast* (*broadcast storm)*

Segundo Gomede (2012), os *switches* que permitem a segmentação da rede em várias VLANs acabam permitindo que o problema fique contido, já que o *broadcast* não atravessa as VLANs. Além disso, ele cita que o uso correto do *Spanning Tree Protocol* (STP), que é tratado na seção 2.3.6, pode evitar os problemas de *loops* que causam a tempestade de *broadcast*.

Adentrando em ferramentas de solução deste problema, a Cisco (2013) informa que nos *switches* de sua marca tem o recurso de "*traffic storm control*" (controle de tempestade de *broadcast*), que monitora, durante um determinado período de tempo, o nível de tráfego de

quadros de *broadcast* e, quando a quantidade de tráfego de *broadcast* exceder o limiar configurado entre esse período, o *switch* bloqueará o tráfego para evitar a sua sobrecarga. Esse método pode ser observado na Figura 19. Gomede (2012) complementa que a maioria dos *switches* atuais possuem este recurso de controle para o *broadcast storm* que evita que a rede fique congestionada, limitando o número de *broadcasts* simultâneos, atentando que, como esta solução bloqueia também *broadcast* legítimos, alguns problemas intermitentes podem ocorrer.

Figura 19 - Supressão de *broadcast* realizado pelo *storm control*



Fonte: Adaptado pelo autor com base em Cisco (2013).

2.3.6 Ataque ao *spanning tree*

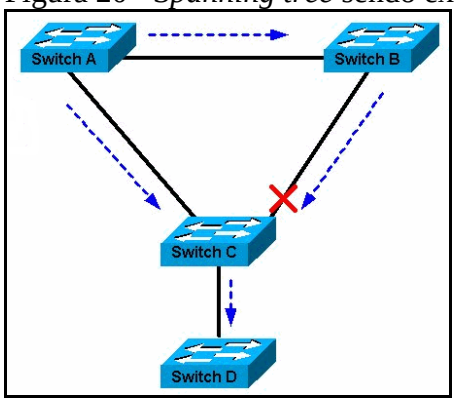
Corrêa (2009) introduz que um tipo de ataque bastante conhecido é a criação de *loops* na rede, que acontece quando duas portas relacionadas à mesma VLAN possuem ligações entre si, ou quando há dois ou mais caminhos entre dois *switches*. Nele os usuários com más intenções e que possuem acesso a infraestrutura de TI podem criar fisicamente *loops* na rede, ligando cabos cruzados em portas que pertencem ao mesmo *switch* e a mesma VLAN. Este tipo de ataque faz com que os pacotes de *broadcast* trafeguem por um longo período de tempo em sua rede, inundando (*flooding*) cada porta que pertence à VLAN atacada em cada *switch*, deteriorando bastante o desempenho, ou até mesmo derrubando a rede.

Para resolver esse problema de *loops* na rede, Corrêa (2009) cita que foi desenvolvido o *Spanning Tree Protocol* (STP) (IEEE 802.1D), que trabalha para desativar *links* que possam formar *loops*, verificando a possibilidade de implantação de ligações redundantes na rede. Segundo Ribeiro (2006) o STP é um protocolo orientado à camada 2 (enlace de dados) do modelo de referência RM-OSI/ISO, sendo que ele foi desenvolvido originalmente pela *Digital Equipament Corporation* e em seguida incorporado pelo padrão IEEE 802.1D.

O STP surgiu, segundo Comer (2007) e Forouzan (2010), para prevenir o problema de laços (*loops*) infinitos, originalmente em *bridges*. Após, ele foi sendo implementado também nos *switches*, para criar uma topologia sem a existência destes *loops* ou, conforme Tanenbaum e Wetherall (2011), interromper estes quando os *switches* são conectados de forma incorreta. Assim, segundo Kurose e Ross (2010), evitando a ciclagem da transmissão de quadros.

Comer (2007) explica que, no protocolo *spanning tree*, os *switches* que concordaram em encaminhar quadros formam um grafo que não contém qualquer ciclo (ou seja, uma árvore) e, segundo Forouzan (2010), apesar de não ser possível alterar a topologia física do sistema em virtude das conexões materiais entre os cabos, é possível criar uma topologia lógica que se sobrepõe àquela física. Portanto, é explicado por Tanenbaum e Wetherall (2011) que a solução para essa dificuldade é estabelecer a comunicação entre os *switches* e sobrepor a topologia real com uma *spanning tree* que alcance cada *switch*, fazendo com que algumas conexões potenciais entre os *switches* sejam ignoradas para que se construa uma topologia virtual livre de *loops* e sem ciclos, que é um subconjunto da topologia real. A Figura 20 ajuda exemplifica de forma simples o processo de *spanning tree* eliminando um *loop* em uma rede composta por vários *switches*.

Figura 20 - *Spanning tree* sendo executado



Fonte: Adaptado pelo autor com base em Cisco (2005).

É necessário informar ainda que, segundo Extreme Networks (2011), a partir do STP foi desenvolvido o *Rapid Spanning Tree Protocol* (RSTP), que fornece um algoritmo de *spanning tree* melhorado que aumenta a velocidade de convergência de redes e, em caso de ocorrer uma alteração ou falha de topologia de rede, recupera mais rapidamente a conectividade da rede confirmando a alteração localmente antes de propagar essa alteração para outros dispositivos através da rede.

Porém, Corrêa (2009) informa que com essa solução veio também outro tipo de problema, onde o atacante envia *broadcasts* de configuração STP ou mudanças de topologia

através do *Bridge Protocol Data Unit* (BPDU), forçando recálculos STP e aguardando que o atacante se torne o *root bridge*. O STP demora cerca de 30 a 45 segundos para reeleger um novo *root bridge* caso o antigo falhe, ocasionando assim em ataques de DoS.

Do ponto de vista de Omar, Pinto e Saide (2013), um ataque de negação no STP consiste em fazer um *spoofing* na ponte raiz. Essa ponte raiz (*root bridge*) é um *switch* eleito pelo algoritmo do STP para ser o caminho prioritário de tráfego da rede, ou seja, ela define o caminho padrão para tráfego no enlace redundante e que, em decorrência desta escolha, bloqueia todos os outros caminhos redundantes. O ataque de manipulação do STP explora o fato do protocolo STP não requerer nenhuma autenticação dos pacotes BPDU que são trocados entre os *switches*, assim, o atacante envia de sua estação um *broadcast* de BPDU especial na rede na tentativa de obrigar o STP a recalcular a ponte raiz determinando sua estação como a nova ponte raiz, provocando uma condição de negação de serviço na rede de cerca de 30 a 45 segundos, a cada tempo de recálculo para mudança da ponte raiz prioritária.

2.3.6.1 Métodos de prevenção para ataque ao *spanning tree*

Segundo Corrêa (2009), para evitar os ataques basta monitorar o tráfego da rede e verificar ocasionalmente as configurações dos *switches*.

Conforme Watkins e Wallace (2008), nos *switches* Cisco, por exemplo, outra maneira de mitigar a manipulação do STP inclui a configuração de *BPDU guard*, *PortFast* e *Root Guard*.

Watkins e Wallace (2008) explicam cada um destes métodos, citando que o recurso *BPDU Guard* é habilitado nas portas configuradas com o recurso de *PortFast*. Sendo que o recurso *PortFast* deve ser habilitado nas portas que se conectam a dispositivos finais, como PCs, já que ele reduz a quantidade de tempo necessário para a porta entrar em estado de encaminhamento depois de ser conectada. A lógica por trás do *PortFast* é que uma porta que se conecta a um dispositivo final não tem o potencial para criar um *loop* na rede, portanto a porta pode ser ativada mais cedo, pulando a escuta do STP e os estados de aprendizagem deste. Devido a estas portas *PortFast* serem conectadas a dispositivos de usuários finais, elas nunca devem receber um BPDU. Logo, se uma porta habilitada para *BPDU Guard* receber um BPDU, ela é desativada.

Já o *Root guard* é um recurso que pode ser habilitado em todas portas do *switch* na rede fora das quais o *root bridge* não deve aparecer (ou seja, todas as portas que não são uma porta raiz (*root*), a porta em cada *switch* que é considerada a mais próxima do *root bridge*). Se uma porta com *Root Guard* receber um BPDU superior, em vez de acreditar no BPDU, a porta entra em estado *root-inconsistent* (raiz-inconsistente), e, enquanto uma porta estiver neste estado, nenhum dado de usuário é enviado através dela. No entanto, após o BPDU superior parar, a porta retorna ao estado de encaminhamento.

2.3.7 Ataque de *VLAN hopping*

O que tange os aspectos fundamentais de uma VLAN (IEEE 802.1Q) foram tratados na seção 2.2.3, porém Omar, Pinto e Saide (2013) acrescentam que também é possível transportar dados e informações de VLANs entre *switches* através de *trunks*. Segundo Kurose e Ross (2010), *trunks*, também chamados de entroncamento, são interligações entre *switches* que permitem distinguir a que VLAN pertence determinado dado através de *tags*, levando em consideração que uma porta *trunk* pertence a todas as VLANs e quadros enviados a qualquer VLAN são encaminhados através do enlace *trunk* ao outro *switch*.

Assim, segundo Corrêa (2009), através da criação de VLANs é possível se atingir um excelente nível de segurança na camada de enlace, pois elas possibilitam a separação dos diferentes tipos de tráfego existente na rede. Porém Omar, Pinto e Saide (2013) advertem que ainda assim é preciso tomar cuidado, pois a má configuração desses *switches* pode permitir a ocorrência de ataques, sendo que Nakamura e Geus (2007) expõem que é possível injetar quadros em uma VLAN e estes serem direcionados a outras VLANs. No caso, será abordado o ataque de *VLAN hopping* utilizando a técnica de *switch spoofing* e de *double tagging*.

2.3.7.1 *VLAN hopping* com *switch spoofing*

Kurose e Ross (2010) relatam que saltar de canal em canal é denominado de *hopping*, logo, seguindo a explanação de Corrêa (2009), o ataque de *VLAN hopping* permite ao invasor tentar enviar dados para *hosts* que estão em outras VLANs. A implementação do padrão IEEE 802.1Q possui um modo padrão de funcionamento do *trunk* que permite ao atacante criar um *link* em modo *trunk* entre ele e o *switch*, dessa maneira conseguindo acesso a todos os *hosts* em todas as VLANs configuradas, “saltando” por entre as VLANs. Watkins e Wallace (2008)

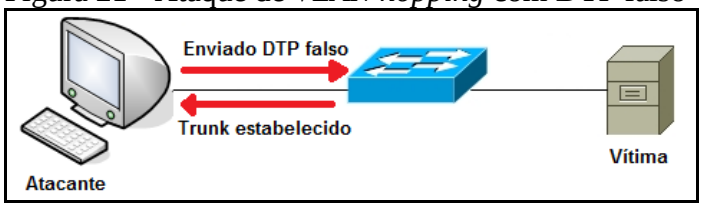
resumem que o *VLAN hopping* é um ataque que permite que o tráfego de uma VLAN passe para outra VLAN sem ser roteado primeiro.

Como exemplo, Rabelo (2014) traz que o DTP (*Dynamic trunk protocol*) é um protocolo proprietário Cisco utilizado entre *switches* diretamente conectados e que negocia de maneira automática a criação de enlaces *trunks* entre eles assim como o tipo de encapsulamento. Ainda segundo Rabelo (2014), esse protocolo para estabelecimento dinâmico de *trunks* denominado DTP, que existe nos *switches* da Cisco, possibilita o ataque de *VLAN hopping*, pois com a utilização deste protocolo DTP, um atacante pode colocar um *switch* na rede e se autoconfigurar com um *trunk* para capturar informações de VLAN que estejam passando por aquele segmento de rede, sendo esta técnica chamada de *switch spoofing*.

Watkins e Wallace (2008) citam dois métodos de execução do *switch spoofing* que podem ser utilizados para realizar o *VLAN hopping*. Um deste métodos consiste em introduzir um *switch* pirata na rede (*rogue switch*) e persuadir a porta do *switch* da rede a entrar em modo *trunk*, permitindo ao atacante ter acesso ao tráfego de todas as VLANs. Enquanto que para o outro método, Watkins e Wallace (2008) informam que algumas portas de um *switch* podem ter como padrão a autoconfiguração para entrar em modo *trunk*, o que significa que estas portas se tornarão automaticamente portas *trunk* ao receberem um quadro DTP. Assim, neste segundo método, o atacante pode tentar fazer a porta de um *switch* entrar no modo de *trunking* enviando mensagens falsas de DTP (*DTP spoofing*).

Para clarificar a visualização deste método, a Figura 21 demonstra a execução de um ataque de *VLAN hopping* através de *switch spoofing*, com o envio de um DTP falso para conseguir acesso *trunk* a porta de um *switch* da rede.

Figura 21 - Ataque de *VLAN hopping* com DTP falso



Fonte: Adaptado pelo autor com base em Certprepare (2014).

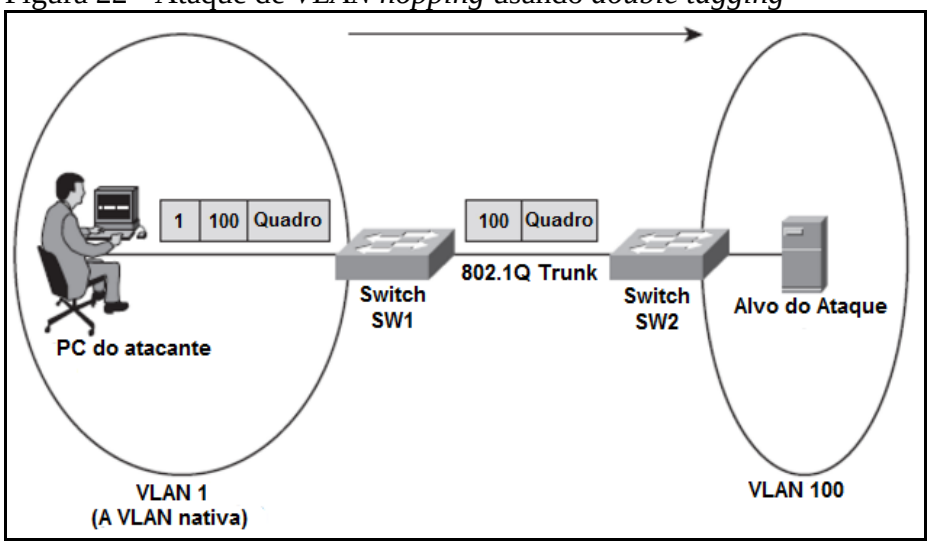
2.3.7.2 *VLAN hopping* com *double tagging*

Watkins e Wallace (2008) explicam que um *trunk* marca com uma etiqueta, ou *tag*, cada quadro indicando a que VLAN ele pertence, processo chamado de "*tagging*" ou etiquetamento. O *VLAN hopping* utilizando *double tagging* é um ataque onde o atacante anexa duas etiquetas, ou *tags*, de VLAN no pacote, daí o nome *double tagging* ou etiquetamento duplo.

Ainda conforme Watkins e Wallace (2008), neste ataque, o primeiro rótulo contém um *tag* válido para VLAN em que ele participa, o qual é apagado ao chegar no primeiro *switch*, sendo o quadro com a segunda *tag* é enviado para o destino. A segunda *tag*, que é falsa, agora está visível para o segundo *switch* que o pacote encontra. Assim, o cabeçalho do quadro indicará que ele é destinado para uma entidade em uma VLAN diferente da original (*tag* 1) e pertence a VLAN apontada no *tag* falso. Portanto, o quadro é enviado para a entidade da rede a qual originalmente o atacante não tinha acesso. Por fim, OmniSecu (2016), explica que o ataque de *double tagging* vai funcionar somente se o atacante estiver conectado a interface que pertence à VLAN nativa da porta *trunk*.

A Figura 22 ajuda a ilustrar um ataque de *VLAN hopping* com *double tagging*, onde o *Personal Computer* (PC) do atacante vincula ao quadro *tags* da VLAN 1 (VLAN nativa) e da VLAN 100 para, após o *switch* 1 (SW1), ficar com o quadro apenas com a *tag* da VLAN do alvo do ataque e efetuar o ataque.

Figura 22 - Ataque de *VLAN hopping* usando *double tagging*



Fonte: Adaptado pelo autor com base em Watkins e Wallace (2008, p. 215).

2.3.7.3 Métodos de prevenção para *VLAN hopping*

Segundo Aaron (2013) e Watkins e Wallace (2008), para prevenir este ataque é recomendado que, nos *switches*, sejam realizadas duas medidas. Primeiro que sejam colocadas todas as portas de borda para o modo "*access*", ou seja, deixar apenas os *links* que realmente são reservados para o entroncamento entre os *switches* como *trunk* e os demais como portas de acesso. Esse modo "*access*" vai forçar que a porta aja como uma porta de acesso, desabilitando qualquer chance dessa se tornar uma porta *trunk* e enviar tráfego para múltiplas VLANs.

A segunda medida é que não seja utilizado o protocolo DTP (*Dynamic Trunking Protocol*), desabilitando ele manualmente em todas as portas, a fim de prevenir que portas *access* se configurem dinamicamente para formar uma relação *trunk* com um potencial atacante, e, com isso, fechando ao máximo a possibilidade de autoconfiguração. Por fim, Watkins e Wallace (2008) também recomendam desabilitar as portas não utilizadas, colocando-as em *shutdown* ou em uma VLAN sem uso.

Aaron (2013), OmniSecu (2016) e Watkins e Wallace (2008) informam que também é recomendado alterar a VLAN nativa, que geralmente é a VLAN 1, para outro valor, fazendo com que essa VLAN não seja utilizada. Com essa recomendação e a dada anteriormente de desabilitar o DTP nas portas *trunk*, é possível garantir que um *switch* invasor (*rogue switch*) não consiga realizar o VLAN *hopping*. Portanto, essas acabam sendo as principais orientações para mitigar esse tipo de ataque.

3 MATERIAIS E MÉTODOS

Quanto a metodologia de pesquisa, do ponto de vista do modo de abordagem do problema, a pesquisa implementada neste trabalho é classificada como de caráter qualitativa no que concerne à análise das principais ameaças e ataques existentes na camada de enlace de dados (nível 2) do modelo de referência RM-OSI/ISO, as soluções existentes para estas ameaças e ataques, e nos resultados obtidos com a implementação destas soluções em um ambiente simulado e em um cenário real. Deslauriers (1991) expõe que essa metodologia qualitativa é capaz de produzir novas informações, segundo Yin (1989), a partir dos estudos intrínsecos ao tema estudado. Conforme definido por Goldenberg (1997), nesse tipo de pesquisa o pesquisador não pode fazer julgamentos, tampouco permitir que seus preconceitos e crenças contaminem a pesquisa. Leopardi (2002) ainda cita que, desta maneira, não serão utilizados instrumentos de medida precisos, mas serão aplicados dados específicos para cada caso, fazendo a avaliação de programas ou propondo programas.

Quanto aos seus objetivos, o cunho da pesquisa deste trabalho é classificado como sendo exploratório e descritivo. Exploratório, de acordo com Gil (2007), devido a este tipo de pesquisa buscar proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a construir hipóteses. Gil (2007) ainda explica que essa metodologia tem como objetivo o aprimoramento de ideias ou a descoberta de intuições, já que aprimorará as noções sobre as ameaças e ataques existentes na camada de enlace de dados (nível 2) e as soluções existentes para mitigar tais ameaças e ataques. Fazendo uma ligação com a pesquisa realizada no trabalho, Chemin (2015) expõe que, dessa forma, a análise qualitativa irá elaborar recomendações para o progresso de práticas já existentes, exibindo resultados não definitivos e passíveis de novos estudos posteriores.

Complementando o cunho exploratório, na perspectiva de seus objetivos, a pesquisa executada também é descritiva, já que, segundo Triviños (1987), exige dos investigadores uma série de informações sobre o que se deseja pesquisar, fazendo com que esse tipo de estudo pretenda descrever os fatos e fenômenos de determinada realidade.

Foram aplicados procedimentos técnicos de pesquisa bibliográfica e experimental para a concepção deste trabalho. No que tange a parte bibliográfica, ela se relaciona ao levantamento de informações sobre toda a revisão bibliográfica executada no trabalho, abrangendo o que é uma rede de computadores, os modelos e camadas existentes, ressaltando a camada que é abordada, os equipamentos vinculados a ela, as ameaças e as soluções para estas. Informações estas que, conforme reforçado por Fonseca (2002), são obtidas consultando materiais já elaborados, sendo estes basicamente livros e artigos científicos. Chemin (2015) comenta que as vantagens dessa forma de pesquisa referem-se à fonte rica e estável de dados, tendo um baixo custo.

Enquanto que a etapa experimental ocorre no momento em que as ameaças elencadas são concretizadas através de ataques, executados nos ambientes de testes de rede local, criados com as vulnerabilidades necessárias para sua execução, e no cenário real, após as defesas terem sido devidamente implementadas. Garces (2010) explica que estudos desse tipo ocorrem em situações controladas e pressupõe a interferência do pesquisador na realidade, por meio da manipulação de variáveis, ou seja, são pesquisas em que se manipulam os dados mutáveis, concedendo um tratamento específico para apurar seus efeitos. De forma mais específica, Gil (2007) explana que a pesquisa experimental consiste em determinar um objeto de estudo, selecionar as variáveis que seriam capazes de influenciá-lo, definir as formas de controle e de observação dos efeitos que a variável gera no objeto.

Para que tal procedimento de técnica de pesquisa experimental seja realizado, primeiramente são executados cada um destes ataques à camada de enlace de dados listados através de ferramentas específicas, equipamentos necessários e técnicas em um ambiente de testes controlado e modificado de forma específica para a realidade necessária para a execução de cada ataque. Isto é realizado no primeiro momento a fim de entender o funcionamento de cada um dos ataques e comprovar sua verídica eficiência antes de tentar reproduzi-lo no ambiente real, após a solução para o ataque ter sido aplicada.

Como ferramentas foram utilizadas o “*macchanger*”, que é um programa utilitário para manipulação de endereços físicos MAC das interfaces de rede presentes em um sistema

Linux. Também o "*macof*", que realiza uma inundação na rede local de pacotes de endereço MAC aleatórios, e o "*arpspoof*", que é utilizada para execução do ataque de *arp spoofing*, da biblioteca de ferramentas de auditoria de rede e testes de penetração "*dsniff*". Além destas, foi utilizada a ferramenta "*DHCP Server for Windows*", que é um *software* que implementa um servidor de DHCP de maneira simples em um computador com sistema operacional Windows. Também foi utilizada a ferramenta "*Yersinia*", que é um *framework* que tira vantagens das vulnerabilidades de diferentes protocolos de rede. Outra ferramenta utilizada foi o "*ping*", que é um programa básico que permite a um usuário verificar se um endereço IP em particular existe e pode aceitar requisições, através do envio de uma *Echo Request* (requisição de resposta) por *Internet Control Message Protocol* (ICMP) para uma interface especificada na rede e o aguardo de uma resposta, assim, testando a conectividade e determinando o tempo de resposta. E por fim, foi utilizada a própria interface de configuração de alguns modelos de placa de rede.

Para criar os ambientes de testes onde os ataques são simulados, foram utilizados diversos computadores e notebooks, com sistemas operacionais Windows e Linux, conectados entre si. Para cada ameaça foi criado um cenário diferente para se adaptar à realidade de como a ameaça poderia explorar uma vulnerabilidade na rede de maneira efetiva para realizar um ataque com sucesso. Dentre os equipamentos utilizados para conexão dos *hosts* se encontram um *switch* da 3Com de 24 Portas e modelo SuperStack3 3300 XM 3C16985B, um *switch* TP-Link modelo TL-SF1024, um *switch* TP-Link modelo TL-SF1008D e um roteador D-Link modelo DIR-100. O roteador foi utilizado em situações em que é necessário um servidor de DHCP para entregar endereços IPs aos *hosts* da rede criada.

Para ilustrar e facilitar o entendimento de cada ambiente de testes criado, o mesmo é representado utilizado o *software* de simulação de redes *Cisco Packet Tracer* na versão 7.0.

Com o objetivo de observar o fluxo de pacotes na rede e entender o funcionamento dos ataques e como funcionam suas prevenções, é utilizada a ferramenta *Wireshark* na versão 2.2.5 nos PCs com sistema operacional Windows e na versão 2.0.2 nos PCs com sistema operacional Linux, que é um *software* analisador de protocolos de rede com recursos de captura de dados e informações detalhadas da composição de cada pacote de dados que trafega pela rede.

Posteriormente, são implementados os métodos de prevenção elencados para cada um destes ataques no s*witch core* Extreme Networks de modelo Summit X460-24t, existente no cenário operacional da Prefeitura Municipal de Lajeado.

Após as técnicas de prevenção terem sido implementadas no *switch core*, o processo de tentativa de ataque é repetido, utilizando os mesmo métodos e ferramentas para verificar se o ataque foi efetivamente mitigado de maneira satisfatória.

4 CENÁRIO

O cenário real e operacional em que são implementadas as defesas para mitigar as ameaças de nível de enlace de dados elencadas é a infraestrutura de rede da Prefeitura Municipal de Lajeado. Ela pode ser classificada como uma MAN, por integrar diversas LANs dispersas pelo município de Lajeado em uma rede única. Essas LANs são configuradas em diversas secretarias, escolas, postos de saúde e demais prédios governamentais. Todas estas redes são interligadas por um nó central situado na sede administrativa do município.

Por questões de logística e acesso aos equipamentos, para efeito deste trabalho, é considerada apenas a infraestrutura de rede da sede administrativa. As demais redes do município, portanto, não fazem parte do escopo do presente trabalho.

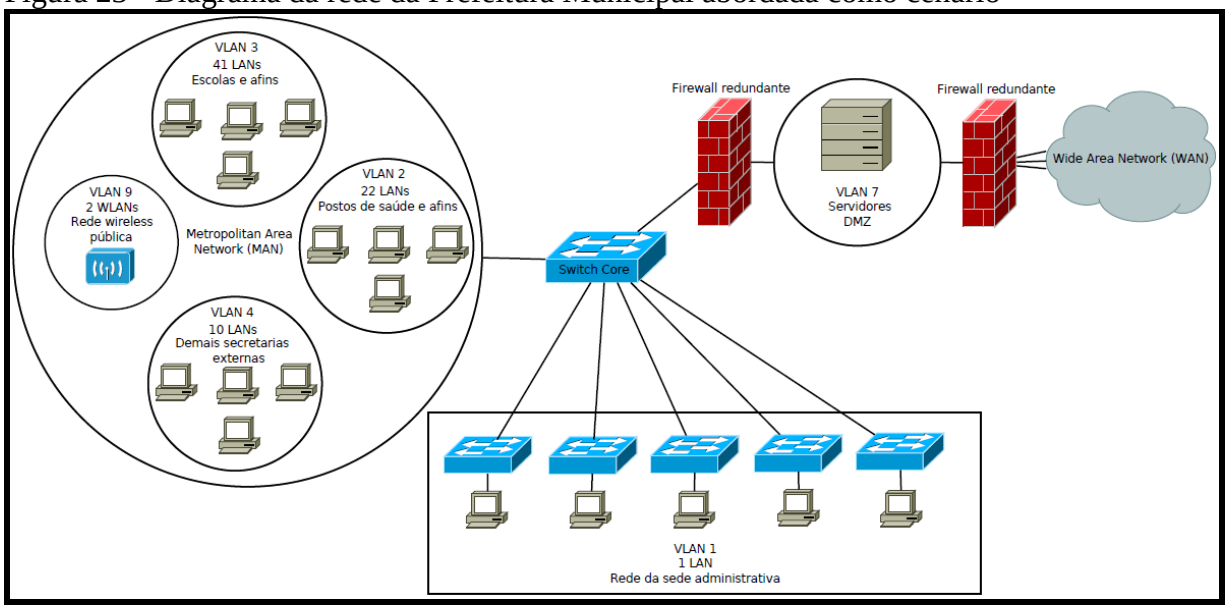
4.1 *switch core*

O *switch core*, que é nó central da rede, é o ponto que interliga todas as demais redes e é por onde passa todo o tráfego de dados executado na Prefeitura Municipal de Lajeado, tanto de saída, como de entrada ou interno. Sendo o objeto de estudo para implementação de defesas neste trabalho, ele é o s*witch core* da *Extreme Networks* de modelo *Summit X460-24t*, versão de *boot ExtremeXOS version* 12.5.0.14 instalado em um sistema operacional Linux 3.13.0-86-generic. Seguindo os conceitos de Extreme Networks (2011), este *switch core* possui o *ExtremeXOS*, que é o *software* que fará o gerenciamento deste *switch* e de suas funções. Nele as configurações e a visualização de qualquer informação são realizadas por inserção de comandos específicos.

Cabe ressaltar que este *switch core* está localizado dentro do *data center* próprio da Prefeitura. Este *data center* possui várias medidas de segurança implementadas no meio físico, como, por exemplo, porta corta fogo, *no-breaks* redundantes, gerador, ar-condicionados de precisão redundantes, acesso biométrico, dispositivos de controle de temperatura e umidade, sistema de combate a incêndio, monitoramento de alteração na alimentação de energia elétrica e alertas diversos através de chamadas telefônicas ou por e-mail.

O esquema da rede da Prefeitura Municipal de Lajeado, que é utilizado como cenário para implementação dos tratamentos para as ameaças à camada de enlace de dados, pode ser ilustrado da forma demonstrada na Figura 23.

Figura 23 - Diagrama da rede da Prefeitura Municipal abordada como cenário



Fonte: Autor.

A Figura 23 reforça o papel central na rede que o *switch core* possui e ilustra que a rede da Prefeitura Municipal de Lajeado pode ser classificada como uma *Metropolitan Area Network* (MAN), já que, conforme Comer (2007), Forouzan (2010) e Tanenbaum e Wetherall (2011), existe a interconexão de várias LANs dispersas pela cidade em uma rede única. Nesta rede são interligadas 75 LANs externas, incluindo escolas, postos de saúde e demais prédios governamentais do município.

Ainda seguindo a Figura 23, é apresentado que é neste *switch core* que são conectados equipamentos de servidores, os equipamentos de *firewall*, os *links* de acesso à *Internet* (que fornecem conexão com a *Wide Area Network* (WAN)) e os outros *switches*, de vários

modelos, que cascateiam a conexão para os computadores da sede administrativa da Prefeitura Municipal de Lajeado. É necessário informar que a WAN é caracterizada por Campbell (199**7)**, Comer (1998), Comer (2007), Forouzan (2010) e Tanenbaum e Wetherall (2011) como uma rede que possui tecnologias para possibilitar uma comunicação que abrange uma grande área geográfica, como múltiplas cidades, países, continentes ou até mesmo possuir uma cobertura mundial, abrangendo o mundo todo.

4.2 VLANs

Devido às vantagens trazidas por estas, existem VLANs configuradas na rede da Prefeitura. Estas são as que se relacionam na Tabela 1, as quais são separadas de acordo com suas funções e os fins necessários para cada situação.

Tabela 1 - Listas de VLANs existentes no cenário de implementação

VLAN	**Função**
1	Rede interna da sede administrativa
2	Pontos relacionados à Secretaria da Saúde
3	Pontos relacionados à Secretaria da Educação
4	Pontos relacionados à demais secretarias externas
5	*Link* de *Internet* 1
6	*Link* de *Internet* 2
7	*DeMilitarized Zone* (DMZ)
8	Destinada servidor *blade*
9	VLAN para *wireless* pública do parque
10	*Link* de *Internet* disponibilizada para acesso público na rede *wireless* do parque
11	*Link* de *Internet* 3
100	Gerenciamento dos ativos de rede
4095	VLAN *untagged* para configuração

Fonte: Autor.

5 IMPLEMENTAÇÃO DAS TÉCNICAS DE ATAQUE E DEFESA E ANÁLISE DOS RESULTADOS

Nesta seção são levantados os resultados da execução dos ataques e também as propostas para mitigar as ameaças detectadas, prevenindo futuros ataques à camada de enlace de dados da rede local da Prefeitura Municipal de Lajeado. Cabe informar que aqui não é abordado o ataque de *VLAN hopping*, tendo em vista que o mesmo não pôde ser reproduzido com sucesso durante a execução deste trabalho e, para que ele tenha um real impacto negativo na rede, é necessário que este seja executado em conjunto com um dos outros ataques abordados.

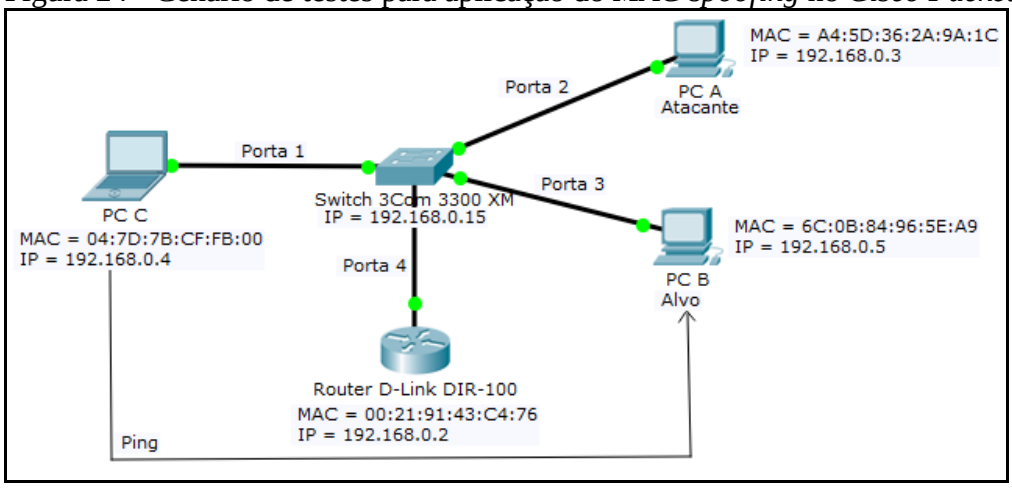
Se faz necessário explicar que após cada aplicação de alterações relacionadas a seguir, as quais são realizadas no *switch core* através da execução de comandos específicos com o propósito de mitigar as ameaças, é importante que elas sejam salvas, conforme Extreme Networks (2011), através da execução do comando "*save*" e da escolha da opção de sobrescrever o atual arquivo de configuração ou salvar estas configurações em um novo arquivo. Isto fará com que as alterações efetuadas sejam mantidas após uma eventual reinicialização do *switch core*.

5.1 Falsificação de endereço MAC (*MAC spoofing*)

Detalhada na seção 2.3.1, a ameaça de falsificação de endereço MAC, ou *MAC spoofing*, se concretiza a partir da obtenção de um endereço MAC de algum alvo e a inserção do mesmo endereço MAC na interface do atacante, mascarando o seu endereço MAC verdadeiro.

Para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido, foi utilizado um ambiente de testes composto por um *switch* 3Com SuperStack3 3300 XM 3C16985B, 3 PCs e um roteador modelo D-Link DIR-100. Neste *switch* foram conectados o PC "A" na porta 2, a partir do qual será executado o ataque, o PC "B" na porta 3, que será o alvo do ataque, isto é, o endereço MAC a ser clonado pelo atacante, o PC "C" na porta 1, a partir do qual serão monitorados os resultados e efeitos do ataque, e o roteador na porta 4 para distribuir endereços IP dinamicamente para cada um dos *hosts* da rede. A Figura 24 apresenta a estrutura do ambiente de testes.

Figura 24 - Cenário de testes para aplicação do *MAC spoofing* no *Cisco Packet Tracer*



Fonte: Autor.

Através do PC "A" foi acessada a interface *web* do *switch* 3Com SuperStack3 3300 XM 3C16985B (pelo endereço IP 192.168.0.15 fixado nele). Na Figura 25 é possível visualizar a tabela CAM deste *switch* antes do ataque ser realizado.

Figura 25 - Tabela CAM do *switch* 3Com antes da execução do *MAC spoofing*

Display Database Entries (100 at a time)

Unit	Port	VLAN	Mac Address	Status
			Ageing Time = 1800 secs	
1	4	1	00:21:91:43:c4:76	Learned
1	1	1	04:7d:7b:cf:fb:00	Learned
1	3	1	6c:0b:84:96:5e:a9	Learned
1	2	1	a4:5d:36:2a:9a:1c	Learned
			Total = 144 Perm = 0	

Fonte: Autor.

O PC “C” ficará executando o comando *ping* para o endereço IP do PC “B” a fim de simular um “cliente” na rede que utiliza o “servidor” PC “B”. Enquanto o *ping* está sendo executado do PC “C” para o PC “B”, o PC “A” atacante não consegue observar os pacotes de ICMP do *ping* trafegando pela rede antes do ataque ser realizado, pois eles são pacotes *unicast* endereçados apenas aos endereços MAC do PC “B” e do PC “C”.

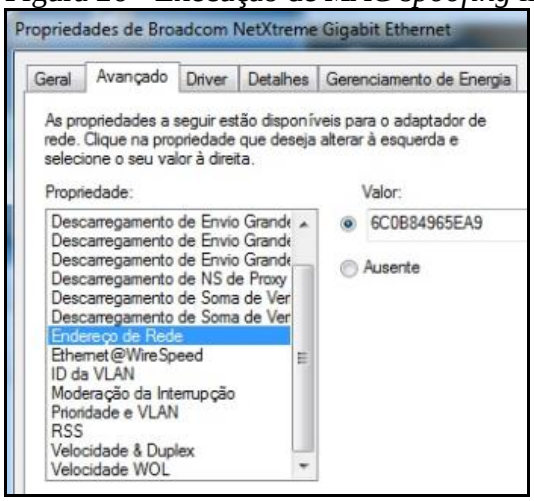
O primeiro passo da execução do ataque foi de descobrir o endereço MAC do alvo. Para tal, o atacante pode executar o comando *ping* para um endereço IP ou para um domínio (neste caso atentando para o endereço IP que responde) existente na rede local. Posteriormente é necessário executar o comando “*arp -a*”. Este comando lista a tabela ARP no computador do atacante, onde é apresentada toda a relação de endereços MAC vinculados ao endereço IP da tabela ARP, incluindo o do alvo desejado.

Com o endereço MAC do alvo obtido, o PC “A” pode executar a técnica de *MAC spoofing* de maneiras distintas dependendo do sistema operacional que ele utiliza. Em sistema operacional Linux é possível realizar este ataque instalando a ferramenta “*macchanger*” e executando o seguinte comando que fará a troca do endereço MAC da interface de rede do atacante:

macchanger -m <MAC_do_alvo> <interface_de-rede>

Já no sistema operacional Windows, que foi utilizado para o ataque, é possível realizar o ataque de *MAC spoofing* utilizando a própria interface de *driver* de algumas placas de rede, que permitem a troca de endereço MAC desta interface. Para execução desse ataque no PC “A”, que possui Windows 7, foi acessada a interface da placa de rede do computador pela tela de “Status de Conexão Local” e clicado no botão de “Propriedades”, após foi selecionada na aba “Avançado” a propriedade de “Endereço de Rede” e preenchido o campo “Valor” com o endereço MAC do PC “B”, conforme demonstrado na Figura 26. Por fim, as alterações foram concluídas clicando no botão “OK”.

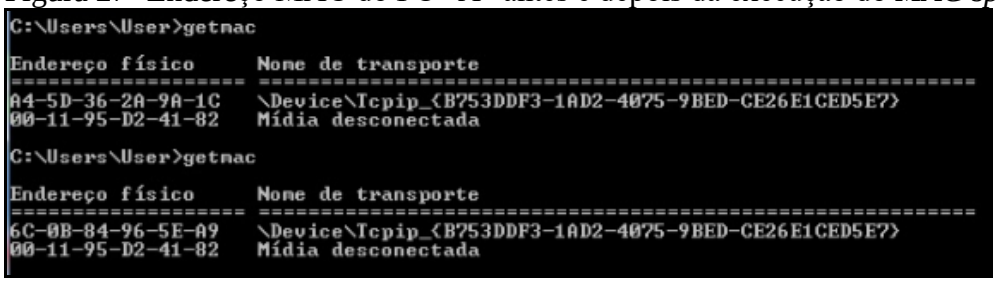
Figura 26 - Execução de *MAC spoofing* no PC “A”



Fonte: Autor.

O resultado das ações realizadas acima é demonstrado na Figura 27, que apresenta o endereço MAC da interface de rede, antes e depois da execução do ataque.

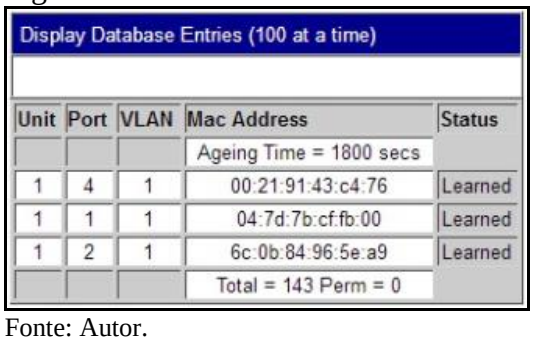
Figura 27- Endereço MAC do PC “A” antes e depois da execução do *MAC spoofing*



Fonte: Autor.

Como resultado prático do ataque, o PC “A” envia um pacote do protocolo ARP informando que ele é o computador com o endereço MAC do PC “B”. Consequentemente a tabela CAM do *switch* 3Com SuperStack3 3300 XM 3C16985B é atualizada, removendo o vínculo anterior do MAC do PC “B” com a porta 3 e agora vinculando o endereço MAC falsificado do PC “B” à porta 2, que é a do PC “A”. Os novos valores desta tabela são apresentados na Figura 28.

Figura 28 - Tabela CAM do *switch* 3Com após a execução do *MAC spoofing*

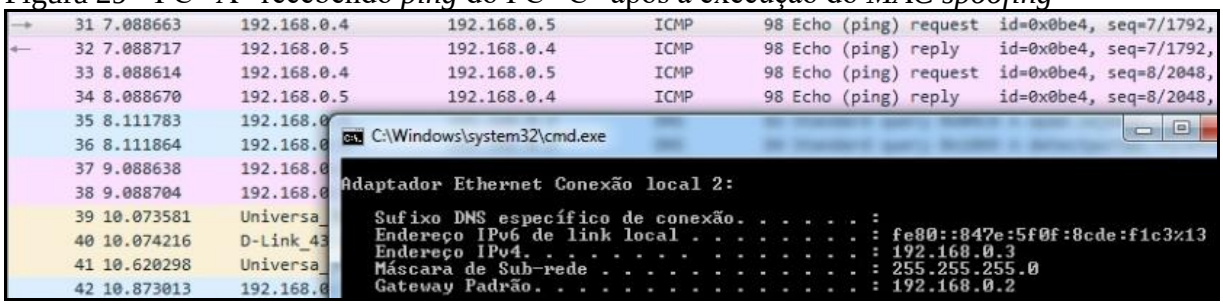


Unit	Port	VLAN	Mac Address	Status
			Ageing Time = 1800 secs	
1	4	1	00:21:91:43:c4:76	Learned
1	1	1	04:7d:7b:cf:fb:00	Learned
1	2	1	6c:0b:84:96:5e:a9	Learned
			Total = 143 Perm = 0	

Fonte: Autor.

Como resultado do ataque, o PC “C” continua executando o comando *ping* para o IP 192.168.0.5 sem perder pacotes durante o ataque, porém, agora o *ping* não é mais recebido e devolvido pelo PC “B”, mas sim pelo PC “A” atacante. Assim, este ataque faz com que o PC “B” fique totalmente fora desta conexão e com que o PC “C” seja enganado. A Figura 29 ilustra o PC “A”, de endereço IP 192.168.0.3, recebendo os pacotes de *ping* destinados ao endereço IP 192.168.0.5, que são exibidos pelo *Wireshark*.

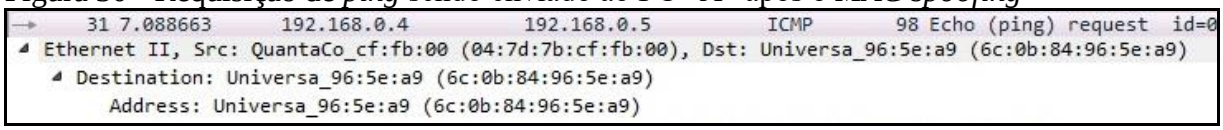
Figura 29 - PC “A” recebendo *ping* do PC “C” após a execução do *MAC spoofing*



Fonte: Autor.

A Figura 30 exibe um pacote ICMP de requisição (*request*) do comando *ping* do PC “C”, que deveria estar sendo enviado para o PC “B”, sendo agora recebido pelo PC “A” atacante. Através dela é possível ver que os pacotes de ICMP do comando *ping* chegam até a placa de rede do PC “A” com o campo de destino preenchido com o MAC falsificado do PC “B”.

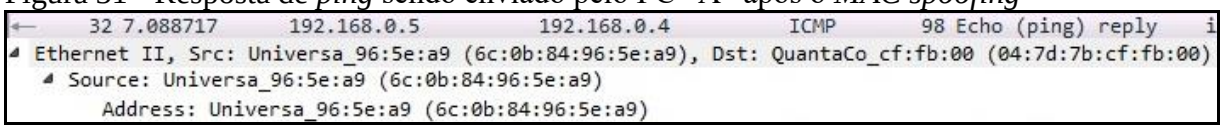
Figura 30 - Requisição de *ping* sendo enviado ao PC “A” após o *MAC spoofing*



Fonte: Autor.

Já a Figura 31 apresenta o pacote ICMP de resposta (*reply*) ao comando *ping* sendo enviado pelo PC “A” para o PC “C”. Também aqui é possível observar o endereço MAC falsificado do PC “B” no campo de fonte do pacote enviado pelo PC “A”.

Figura 31 - Resposta de *ping* sendo enviado pelo PC “A” após o *MAC spoofing*



Fonte: Autor.

Com a execução do ataque de *MAC spoofing*, é possível causar um DoS, já que qualquer serviço do PC "B" será negado ao PC "C", ou um *man-in-the-middle*, já que o PC "A" pode interceptar, registrar e alterar todos os pacotes enviados a ele sem que o PC "C" perceba. Com estes dados é possível observar que o ataque de falsificação de MAC, ou *MAC spoofing*, foi executado com sucesso neste ambiente de teste vulnerável apresentado.

5.1.1 Aplicando defesa ao *MAC spoofing* na infraestrutura operacional da Prefeitura

Diferentemente dos testes de ataque, as proteções contra *MAC spoofing* serão aplicadas no *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

De acordo com Extreme Networks (2011), o *software* ExtremeXOS possui o comando de *lock-learning* que limita e fixa o endereço MAC aprendido em cada porta. Para executar o *lock-learning* é necessário informar o seguinte comando:

configure ports <portas> vlan <nome_da_vlan> lock-learning

Este comando tem por finalidade fazer com que a entrada dinâmica de endereço MAC da tabela FDB da VLAN e da porta especificada seja convertida em uma entrada estática fixa. Este comando também atualizará o limite de aprendizado para esta porta e VLAN para o valor 0, fazendo com que nenhuma nova entrada seja aprendida. Todas as novas fontes de endereço MAC serão marcadas como *blackhole* (uma entrada *blackhole* na FDB irá descartar todos os pacotes endereçados para ou recebidos de um endereço MAC específico) com o rótulo "Bb" e tentativas de aprendizado do mesmo MAC em outras portas terão os pacotes descartados.

Para remover um endereço MAC fixado com esse comando é necessário informar o seguinte comando:

configure ports <portas> vlan <nome_da_vlan> unlock-learning

Quando o endereço MAC fixado for removido com a opção *unlock-learning*, o limite de aprendizado de endereços MAC será redefinido para ilimitado e todas as entradas associadas na FDB serão liberadas.

Foi executado este comando para fixar o endereço MAC das portas em que estão conectados os equipamentos servidores, o *firewall* e os *links* de acesso à *Internet*. Cada uma

destas portas teve as suas respectivas VLANs adicionadas ao comando. Como por exemplo, para habilitar o *lock-learning* na porta 2 foram executados os seguintes comandos:

configure ports 2 vlan "Default" lock-learning

configure ports 2 vlan "SAUDE" lock-learning

configure ports 2 vlan "ESCOLAS" lock-learning

configure ports 2 vlan "SECRETARIAS" lock-learning

configure ports 2 vlan "DMZ" lock-learning

Ainda seguindo o que é informado por Extreme Networks (2011), para mostrar as entradas fixadas no *switch* é necessário usar o comando "*show fdb ports* <*porta*>". As entradas com endereço MAC fixado estarão sinalizadas com "l". Como na Figura 32, que mostra a defesa aplicada através da execução do comando *lock-learning* para fixar o endereço MAC do servidor de *firewall* nas cinco diferentes VLANs da porta 2.

Figura 32 - *Lock-learning* aplicado na porta 2 do *switch core*

```
(Software Update Required) * X460-24t.8 # show fdb ports 2
Mac                    Vlan       Age  Flags         Port / Virtual Port List
-------------------------------------------------------------------------------
00:e0:7d:da:c3:0e    Default(0001) 0000 spm    l       2
00:e0:7d:da:c3:0e      SAUDE(0002) 0000 spm    l       2
00:e0:7d:da:c3:0e    ESCOLAS(0003) 0000 spm    l       2
00:e0:7d:da:c3:0e SECRETARIAS(0004) 0000 spm    l       2
00:e0:7d:da:c3:0e        DMZ(0007) 0000 spm    l       2
```

Fonte: Autor.

Para testar a efetividade dessa proteção aplicada, foi repetido o ataque de *MAC spoofing* na infraestrutura operacional da Prefeitura. Para tal execução ocorrer foi falsificado o endereço MAC da porta 2 (onde atualmente está conectado o servidor de *firewall*) na VLAN 2. Para realizar o ataque de *MAC spoofing* foi utilizado um notebook com Linux, conectado na porta 15 do *switch core*. Após conexão, foi criada uma interface de rede na VLAN 2 neste notebook e aplicado o comando da Figura 33 que executará a técnica de troca de endereço MAC original do notebook para o endereço falsificado do servidor que está na porta 2.

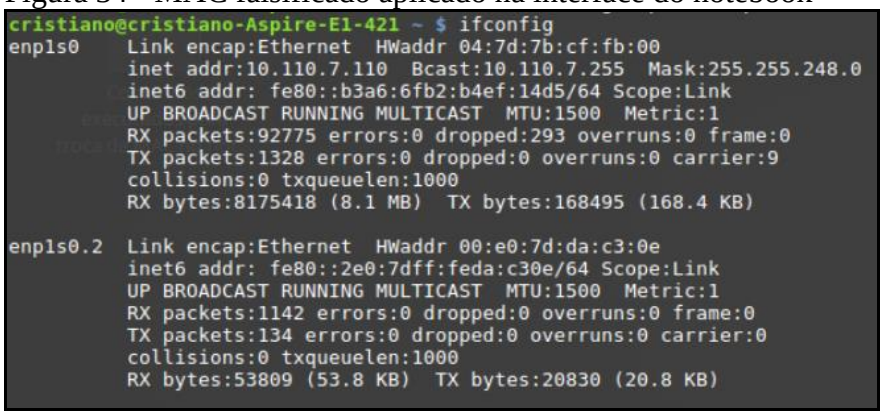
Figura 33 - Alterando o MAC para execução do *MAC spoofing* no *switch core*

```
cristiano@cristiano-Aspire-E1-421 ~ $ sudo macchanger -m 00:E0:7D:DA:C3:0E enp1s0
Current MAC:   04:7d:7b:cf:fb:00 (Quanta Computer Inc.)
Permanent MAC: 04:7d:7b:cf:fb:00 (Quanta Computer Inc.)
New MAC:       00:e0:7d:da:c3:0e (NETRONIX, INC.)
```

Fonte: Autor.

A Figura 34 exibe a interface da VLAN 2 do notebook com o MAC falsificado do servidor já aplicado.

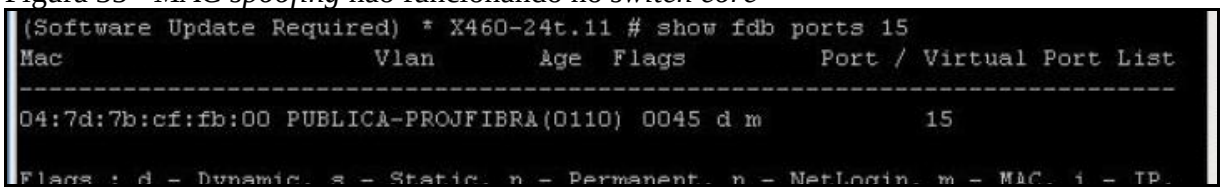
Figura 34 - MAC falsificado aplicado na interface do notebook



Fonte: Autor.

Agora o ataque já não funciona mais, como apresentado na Figura 35. Isso pode ser comprovado a partir da observação que o endereço MAC falsificado não é vinculado a porta 15 do atacante, assim a tabela FDB não sofra alterações referentes a trocar a porta do endereço MAC do alvo do ataque, mantendo ele na porta 2.

Figura 35 - *MAC spoofing* não funcionando no *switch core*



Fonte: Autor.

5.1.2 Análise dos resultados de prevenção ao *MAC spoofing*

Os testes de efetividade aplicados demonstram que a fixação de endereço MAC com a específica porta através do comando de *lock-learning* mitiga com sucesso a ameaça de *MAC spoofing* nestes endereços MAC. Com esta proteção sendo aplicada em portas que se conectam a equipamentos cujos endereços MAC mudam com pouca frequência (como servidores e links de acesso à *Internet*), é possível afirmar que a ameaça de *MAC spoofing* foi mitigada com sucesso nestes que são os pontos mais críticos da Prefeitura.

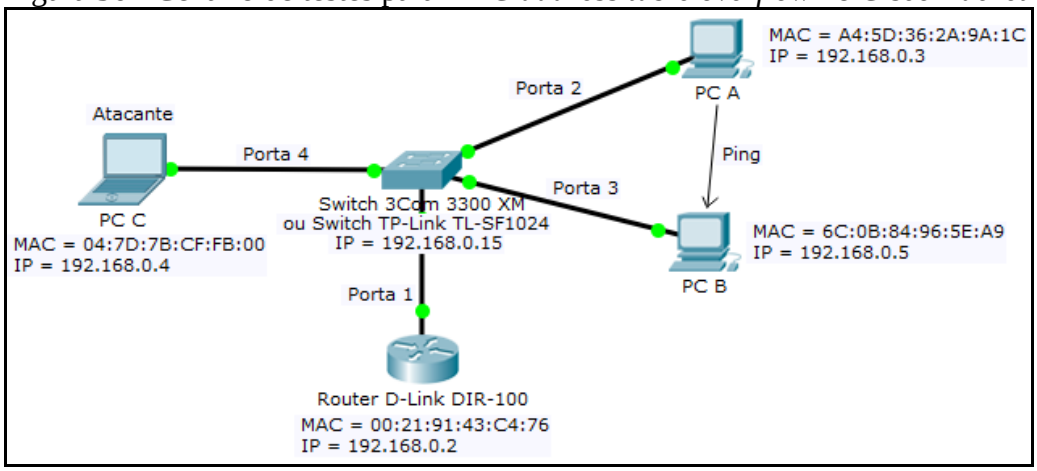
Com estas prevenções aplicadas, é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques do tipo *MAC spoofing* aos servidores conectados no *switch core*.

5.2 *MAC address table overflow*

Detalhada na seção 2.3.2, a ameaça de inundação da tabela de endereços MAC, ou *MAC address table overflow*, se concretiza a partir do envio de uma grande quantidade de pacotes do protocolo IPv4 contendo endereços MAC de origem e destino aleatórios, com o objetivo de sobrecarregar a capacidade da tabela CAM ou FDB do *switch*, causando a indisponibilidade deste equipamento (*fail-closed*) ou alterando o seu modo de operação para um de um *hub* (*fail-open*).

Foram realizados dois testes para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido. No teste 1 foi utilizado o *switch* 3Com SuperStack3 3300 XM 3C16985B em estado de "*fail-closed*", enquanto que no teste 2 foi utilizado o *switch* TP-Link TL-SF1024 em estado de "*fail-open*". No ambiente de testes, foram utilizados ainda 3 PCs e um roteador modelo D-Link DIR-100. No ambiente dos dois testes foram conectados ao *switch* utilizado no experimento o PC "A" na porta 2, que executará o comando de *ping* para o PC "B" na porta 3, o PC "C" na porta 4, a partir do qual será executado o ataque, e o roteador na porta 1, para distribuir endereços IP dinamicamente para cada um dos *hosts* da rede. A Figura 36 apresenta a estrutura do ambiente dos dois testes.

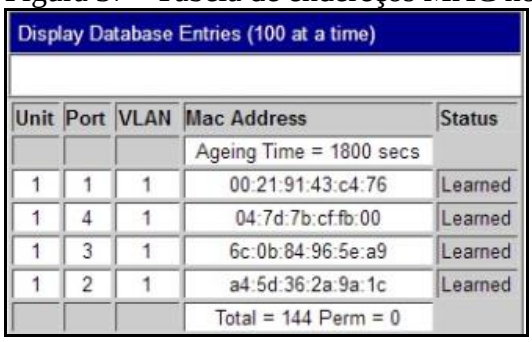
Figura 36 - Cenário de testes para *MAC address table overflow* no *Cisco Packet Tracer*



Fonte: Autor.

No teste 1, através do PC "C" foi acessada a interface *web* do *switch* pelo IP 192.168.0.15 fixado nele. Na Figura 37 é possível visualizar a tabela CAM deste *switch* funcionando normalmente.

Figura 37 - Tabela de endereços MAC no *switch* 3Com antes do *MAC table overflow*



Display Database Entries (100 at a time)

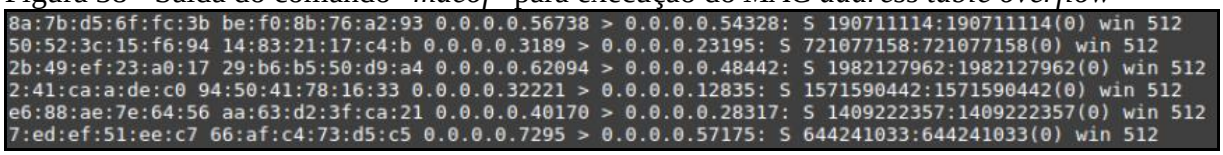
Unit	Port	VLAN	Mac Address	Status
			Ageing Time = 1800 secs	
1	1	1	00:21:91:43:c4:76	Learned
1	4	1	04:7d:7b:cf:fb:00	Learned
1	3	1	6c:0b:84:96:5e:a9	Learned
1	2	1	a4:5d:36:2a:9a:1c	Learned
			Total = 144 Perm = 0	

Fonte: Autor.

O PC “A” ficará executando o comando *ping* para o endereço IP do PC “B”, a fim de simular uma rede em funcionamento. Enquanto o *ping* está sendo executado, o PC “C” atacante não consegue observar os pacotes do *ping* trafegando pela rede antes do ataque ser executado, pois eles são pacotes *unicast*.

Para o PC “C” executar o ataque de *MAC address table overflow*, foi instalada a biblioteca de ferramentas “*dsniff*” em sistema operacional Linux. Desta biblioteca foi utilizada a ferramenta “*macof*” através do comando “*macof*”. A execução deste simples comando faz com que o PC “C” imediatamente injete uma grande quantidade de endereços MAC inválidos na tabela CAM ou FDB do *switch* pela porta 4. A Figura 38 demonstra parte da saída na tela feita pela execução do comando “*macof*” no PC “C”.

Figura 38 - Saída do comando “*macof*” para execução do *MAC address table overflow*

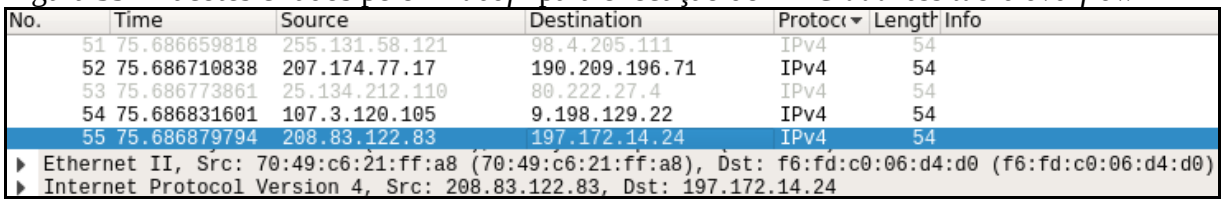


```
8a:7b:d5:6f:fc:3b be:f0:8b:76:a2:93 0.0.0.0.56738 > 0.0.0.0.54328: S 190711114:190711114(0) win 512
50:52:3c:15:f6:94 14:83:21:17:c4:b 0.0.0.0.3189 > 0.0.0.0.23195: S 721077158:721077158(0) win 512
2b:49:ef:23:a0:17 29:b6:b5:50:d9:a4 0.0.0.0.62094 > 0.0.0.0.48442: S 1982127962:1982127962(0) win 512
2:41:ca:a:de:c0 94:50:41:78:16:33 0.0.0.0.32221 > 0.0.0.0.12835: S 1571590442:1571590442(0) win 512
e6:88:ae:7e:64:56 aa:63:d2:3f:ca:21 0.0.0.0.40170 > 0.0.0.0.28317: S 1409222357:1409222357(0) win 512
7:ed:ef:51:ee:c7 66:af:c4:73:d5:c5 0.0.0.0.7295 > 0.0.0.0.57175: S 644241033:644241033(0) win 512
```

Fonte: Autor.

A Figura 39 ilustra que, para inundar a tabela CAM ou FDB com endereços MAC falsos, o PC “C” envia para a porta 4 uma grande quantidade de pacotes de protocolo IPv4 com endereços MAC de fonte falsos destinados a endereços MAC falsos, gerados randomicamente.

Figura 39 - Pacotes criados pelo “*macof*” para execução do *MAC address table overflow*



No.	Time	Source	Destination	Protoc	Length	Info
51	75.686659818	255.131.58.121	98.4.205.111	IPv4	54	
52	75.686710838	207.174.77.17	190.209.196.71	IPv4	54	
53	75.686773861	25.134.212.110	80.222.27.4	IPv4	54	
54	75.686831601	107.3.120.105	9.198.129.22	IPv4	54	
55	75.686879794	208.83.122.83	197.172.14.24	IPv4	54	

```
▸ Ethernet II, Src: 70:49:c6:21:ff:a8 (70:49:c6:21:ff:a8), Dst: f6:fd:c0:06:d4:d0 (f6:fd:c0:06:d4:d0)
▸ Internet Protocol Version 4, Src: 208.83.122.83, Dst: 197.172.14.24
```

Fonte: Autor.

O resultado das ações realizadas acima é demonstrado na Figura 40, que apresenta parte da tabela CAM do *switch* do teste 1, que foi inundada com endereços MAC falsos na porta 4.

Figura 40 - Tabela CAM do *switch* 3Com inundada

1	4	1	08:11:1a:47:69:9b	Learned
1	4	1	08:1f:bd:04:fc:a0	Learned
1	4	1	08:28:5d:73:f9:05	Learned
1	4	1	08:3d:70:2b:2c:be	Learned
1	4	1	08:45:4c:3a:b2:bb	Learned
1	4	1	08:45:d9:38:3b:8b	Learned
			Total = 6000 Perm = 0	

Fonte: Autor.

Como resultado do ataque de *MAC address table overflow*, a memória da tabela CAM ou FDB do *switch* "transborda" (*overflow*), fazendo com que o *switch* se comporte de duas maneiras distintas, dependendo do seu estado.

Com o sucesso da execução do ataque, no teste 1, que possui o *switch* que está no estado "*fail-closed*", é possível causar um DoS. Pois o *switch* para de encaminhar qualquer pacote, cancelando todo o tráfego na rede (incluindo o *ping* entre o PC "A" e o PC "C").

Já no teste 2, que possui o *switch* que está no estado "*fail-open*", é executado o mesmo ataque, porém agora possibilitando a execução de um *man-in-the-middle*. Pois o *switch* começa a funcionar como um *hub*, enviando todos os pacotes para todos os *hosts*, conforme demonstrado na Figura 41, onde o PC "C" está interceptando os pacotes enviados entre PC "A" e PC "B", podendo ainda registrar e alterar estes.

Figura 41 - *Switch* TP-Link TL-SF1024 funcionando como *hub* após ataque

No.	Time	Source	Destination	Protocol	Length	Info
19254	12.730596442	192.168.0.3	192.168.0.5	ICMP	74	Echo (ping) request
25580	13.744567494	192.168.0.3	192.168.0.5	ICMP	74	Echo (ping) request
33601	14.758513463	192.168.0.3	192.168.0.5	ICMP	74	Echo (ping) request

▾ Frame 19254: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
▾ Ethernet II, Src: HewlettP_2a:9a:1c (a4:5d:36:2a:9a:1c), Dst: Universa_96:5e:a9 (6c:0b:84:96:5e:a9)

Fonte: Autor.

5.2.1 Aplicando defesa ao *MAC address table overflow* na estrutura operacional da Prefeitura

Diferentemente dos testes de ataque, as proteções contra *MAC address table overflow* serão aplicadas no *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

Da mesma forma que no ataque de *MAC spoofing*, aqui a defesa pode ser feita pelo comando de *lock-learning*, que fixa o endereço MAC aprendido e bloqueia os novos aprendizados de endereço MAC na porta, limitando a apenas um endereço MAC por porta, conforme explicado na seção 5.1.1.

Para implementar defesa nas portas que possuem mais de um endereço MAC, de acordo com Extreme Networks (2011), o *software* ExtremeXOS possui o comando de *limit-learning* que limita a quantidade de endereços MAC em cada porta. Para executar o *limit-learning* é necessário informar o seguinte comando:

configure port <número_da_porta> vlan <nome_da_vlan> limit-learning <número_de_limite> action [blackhole|stop-learning]

Este comando tem por finalidade especificar o número de endereços MAC dinamicamente aprendidos pela FDB que são permitidos para a porta na VLAN especificada. Quando o limite de aprendizado for alcançado, em todas as novas fontes de endereço MAC serão aplicadas uma das duas ações a serem selecionadas. Na ação de *blackhole* as novas entradas de endereço MAC ficarão marcadas pelo rótulo “Bb” como *blackhole* (uma entrada *blackhole* na FDB irá descartar todos os pacotes endereçados para ou recebidos de um endereço MAC específico). Já na ação de *stop-learning* não serão criadas entradas de *blackhole* e os pacotes endereçados de um endereço MAC de uma nova fonte serão descartados, o que protege a FDB de ficar lotada com endereços MAC em *blackhole*.

Para remover o limite de endereços MAC especificado é necessário executar o seguinte comando, que define o aprendizado de novos endereços MAC na porta e VLAN especificada para um valor ilimitado:

configure port <número_da_porta> vlan <nome_da_vlan> unlimited-learning

Foi utilizado este comando de *limit-learning* para limitar os endereços de MAC aprendidos nas portas que não possuíam o *lock-learning* ainda implementado, por possuírem mais de um endereço MAC na porta e estes mudarem com frequência, como as ligadas aos demais *switches* da sede administrativa da Prefeitura. Cada uma destas portas teve as suas respectivas VLANs adicionadas ao comando e um número de limite definido de forma adequada a quantidade de endereços MAC que é possível aprender naquela porta. Como exemplo de demonstração, para visualizar as entradas bloqueadas, foi habilitado o *limit-learning* na porta 9 com ação de *blackhole* através do seguinte comando:

configure port 9 vlan "LINK_BWNET_2" limit-learning 1 blackhole

Também foram mantidas as configurações de *lock-learning* realizadas na seção 5.1.1.

Para testar a efetividade dessa proteção aplicada, foi repetido o ataque de *MAC address table overflow* na estrutura operacional da Prefeitura. Para tal foi conectado um notebook com Linux na porta 9 do *switch core* e, após, foi executado novamente o comando "*macof*".

Agora o ataque já não funciona mais, como apresentado na Figura 42, onde é permitido apenas um endereço MAC e os novos endereços MAC falsos são marcadas pelo rótulo "Bb" como *blackhole* para envio e recebimento de pacotes.

Figura 42 - FDB do *switch core* com os endereços MAC falsos em *blackhole*

```
(Software Update Required) * X460-24t.4 # show fdb port 9
Mac                   Vlan       Age  Flags         Port / Virtual Port List
-------------------------------------------------------------------------------
04:7d:7b:cf:fb:00 LINK_BWNET_2(0011) 0000 spm           9
00:0a:cf:27:53:01 LINK_BWNET_2(0011) 0023 d m  Bb       9
00:0c:19:76:e4:94 LINK_BWNET_2(0011) 0025 d m  Bb       9
00:0c:e7:7a:da:b7 LINK_BWNET_2(0011) 0027 d m  Bb       9
00:16:e9:36:44:7b LINK_BWNET_2(0011) 0022 d m  Bb       9
00:17:d1:30:00:88 LINK_BWNET_2(0011) 0022 d m  Bb       9
00:21:8b:0e:7b:5b LINK_BWNET_2(0011) 0022 d m  Bb       9
```

Fonte: Autor.

Para não exaurir os recursos da FDB com entradas *blackhole*, foi selecionada a ação de *stop-learning* nas portas onde o *limit-learning* foi aplicado no *switch core*.

5.2.2 Análise dos resultados de prevenção ao *MAC address table overflow*

Os testes de efetividade aplicados demonstram que a especificação de um limite de endereços MAC aprendidos em uma porta através do comando de *limit-learning*, aliada a fixação de endereço MAC com a porta específica através do comando de *lock-learning*, mitiga com sucesso a ameaça de *MAC address table overflow*. Com a proteção de *lock-learning* aplicada nas portas em que se conectam os equipamentos que mudam seus endereços MAC com pouca frequência (como os servidores e os *links* de acesso à *Internet*), e a proteção de *limit-learning* aplicada nas portas em que se conectam os demais *switches* da rede da sede administrativa, onde há uma grande quantidade de endereços MAC inconstantes e variados, é possível afirmar que a ameaça de *MAC address table overflow* foi mitigada com sucesso nas portas do *switch core*.

Também cabe informar que o uso de tabelas ARP estáticas não é viável para implementação na rede do ambiente operacional da Prefeitura. Visto o tamanho e complexidade dela, este procedimento acabaria gerando tanta carga administrativa que limitaria muito a dinamicidade do ambiente de trabalho, gerando um gigantesco entrave de acesso à rede e tornando tal procedimento impraticável. Além disso, não é necessário a implementação de ACLs aqui, visto que a ameaça pode ser mitigada com outras defesas.

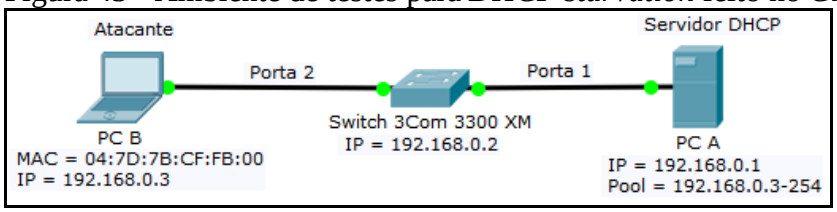
Com estas prevenções aplicadas é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques do tipo *MAC address table overflow* no *switch core*.

5.3 *DHCP starvation*

Detalhada na seção 2.3.3.1, a ameaça de *DHCP starvation* se concretiza a partir da criação de inúmeras requisições de DHCP com endereços MAC falsos até que todos os endereços IP disponíveis do servidor sejam alocados.

Para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido, foi utilizado um ambiente de testes composto por um *switch* 3Com SuperStack3 3300 XM 3C16985B e 2 PCs. Neste switch foram conectados o PC "A" na porta 1, que será o servidor de DHCP, e o PC "B" na porta 2, a partir do qual será executado o ataque. A Figura 43 apresenta a estrutura do ambiente de testes com os específicos endereços IP de cada *host* da rede.

Figura 43 - Ambiente de testes para *DHCP starvation* feito no *Cisco Packet Tracer*



Fonte: Autor.

No PC "A" foi instalado o *software* livre "*DHCP Server for Windows*", que é uma ferramenta para criação de um servidor DHCP de forma simples em um computador com sistema operacional Windows. Na Figura 44 é possível visualizar a tabela de clientes DHCP deste servidor do PC "A" antes do ataque ser realizado, com o endereço IP do PC "B" já definido.

Figura 44 - Tabela de clientes DHCP antes da execução do *DHCP starvation*

SERVER_0 DHCP Clients

Id	IP Address	Hostname	AutoConfig	Lease ends
04-7D-7B-CF-FB-00	192.168.0.3	cristiano-Aspire-E1-421	03/24/2017 17:13:52	Sat Mar 25 17:13:52 2017

Fonte: Autor.

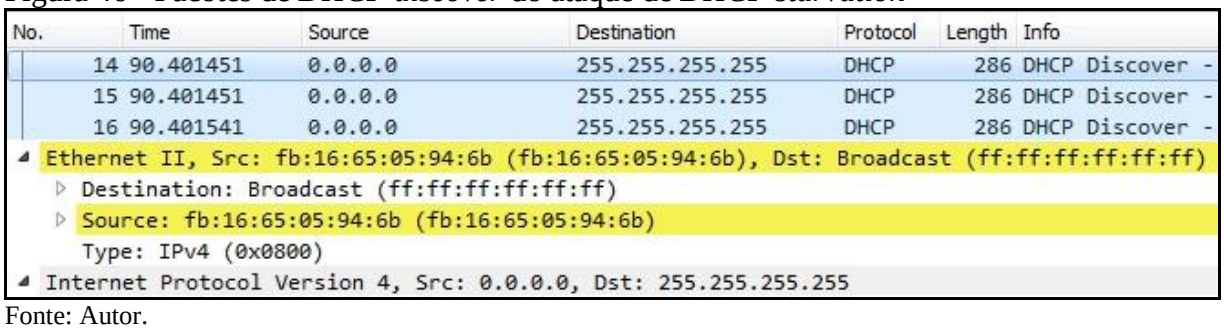
Para o PC "B" poder executar o ataque de *DHCP starvation*, foi instalada a ferramenta de rede "*Yersinia*". Após instalação, foi inicializada a ferramenta com o comando "*yersinia -I*", selecionada a interface a ser utilizada no ataque pressionando a tecla "i", pressionada a tecla "g" e escolhido o modo de protocolo DHCP, pressionada a tecla "x" para abrir o menu de ataques e, por fim, pressionada a tecla "1" para enviar uma inundação de pacotes *discover* de DHCP e iniciar o ataque. A Figura 45 ilustra a tela de saída apresentada na execução deste ataque.

Figura 45 - Execução do ataque de *DHCP starvation* no PC "B"

Fonte: Autor.

O resultado das ações realizadas acima é demonstrado na Figura 46, que apresenta alguns dos inúmeros pacotes de *DHCP discover* gerados com endereços MAC falsos criados aleatoriamente e enviados para o endereço de *broadcast* pelo PC "B".

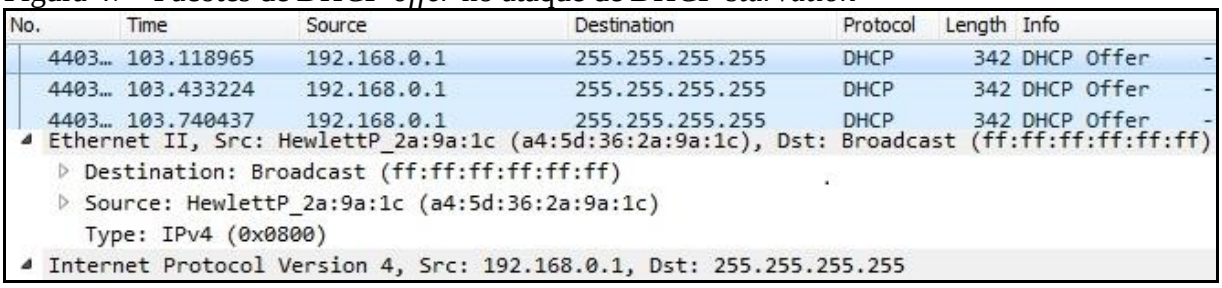
Figura 46 - Pacotes de *DHCP discover* do ataque de *DHCP starvation*



Fonte: Autor.

Cada um destes pacotes de *DHCP discover* do ataque receberá um pacote de *DHCP offer*, onde o servidor de DHCP atribui um endereço IP para este endereço MAC falso gerado. A Figura 47 ilustra estes pacotes de *DHCP offer* que são enviados para *broadcast* sendo recebidos pelo PC "B".

Figura 47 - Pacotes de *DHCP offer* no ataque de *DHCP starvation*



Fonte: Autor.

Como resultado do ataque, a tabela de clientes de DHCP do servidor do PC "A" ficará lotada, atribuindo endereços IP aos endereços MAC falsos gerados nas requisições, conforme demonstrado na Figura 48.

Figura 48 - Parte da tabela de clientes DHCP após ataque de *DHCP starvation*

74-C4-C5-3B-0C-6F	192.168.0.210		03/24/2017 17:31:38	Fri Mar 24 17:31:39 2017
DD-9D-2B-1F-BB-3F	192.168.0.211		03/24/2017 17:31:39	Fri Mar 24 17:31:40 2017
B9-69-78-45-E1-F7	192.168.0.212		03/24/2017 17:31:39	Fri Mar 24 17:31:40 2017
C5-C5-5A-20-96-E2	192.168.0.213		03/24/2017 17:31:39	Fri Mar 24 17:31:40 2017
1A-F0-D0-62-4D-87	192.168.0.214		03/24/2017 17:31:40	Fri Mar 24 17:31:41 2017
D8-91-4D-44-5A-70	192.168.0.215		03/24/2017 17:31:40	Fri Mar 24 17:31:41 2017

Fonte: Autor.

Com a execução do ataque de *DHCP starvation*, é possível causar um DoS, pois, ao conectar novos PCs na rede do ambiente de testes, nenhum deles obteve um endereço IP. Com estes dados é possível observar que o ataque de *DHCP starvation* foi executado com sucesso neste ambiente de teste vulnerável apresentado.

5.3.1 Aplicando defesa ao *DHCP starvation* na infraestrutura operacional da Prefeitura

Diferentemente dos testes de ataque, as proteções contra *DHCP starvation* serão aplicadas no *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

Da mesma forma que no ataque de *MAC address table overflow*, a defesa para os ataques de *DHCP starvation* pode ser feita pelos comandos de *lock-learning*, que fixa o endereço MAC aprendido e bloqueia os novos aprendizados de endereço MAC na porta, limitando a apenas um endereço MAC por porta, conforme explicado na seção 5.1.1, e de *limit-learning* que limita a quantidade de endereços MAC em cada porta, conforme explicado

na seção 5.2.1. Portanto foram mantidas as configurações de *lock-learning* e de *limit-learning* realizadas na seção 5.1.1 e 5.2.1.

Para testar a efetividade dessa proteção aplicada, foi repetido o ataque de *DHCP starvation* na infraestrutura operacional da Prefeitura. Para realizar tal ataque foi conectado um notebook com Linux na porta 9 do *switch core*. Após a conexão, foram repetidos os mesmos procedimentos de execução do ataque de *DHCP starvation* descritos na seção 5.3.

Agora o ataque já não funciona mais, como apresentado na Figura 49, onde os endereços MAC gerados aleatoriamente nos pacotes de *DHCP discover* que estão acima do limite de aprendizado permitido na porta, estão bloqueados ao serem marcados como *blackhole* pelo rótulo “Bb” (*blackhole* em envio e recebimento). Assim os pacotes de *DHCP starvation* com estes endereços MAC falsos não passarão pela porta do *switch core* e não chegarão ao servidor de DHCP.

Figura 49 - *Switch core* bloqueando os endereços MAC falsos do *DHCP starvation*

Fonte: Autor.

5.3.2 Análise dos resultados de prevenção ao *DHCP starvation*

Os testes de efetividade aplicados demonstram que a especificação de um limite de endereços MAC aprendidos em uma porta através do comando de *limit-learning*, aliada a fixação de endereço MAC com a porta específica através do comando de *lock-learning*, mitiga com sucesso a ameaça de *DHCP starvation*. Com esta proteção de *lock-learning* sendo aplicada em portas que se conectam a equipamentos cujos os endereços MAC mudam com pouca frequência (como servidores e *links* de acesso à *Internet*), e esta proteção de *limit-learning* sendo aplicada em portas que se conectam aos demais *switches* da rede em que há uma grande quantidade de endereços MAC inconstantes e variados, é possível afirmar que a ameaça de *DHCP starvation* foi mitigada com sucesso no servidor de DHCP que se conecta ao *switch core* da sede administrativa da Prefeitura.

Com estas prevenções aplicadas é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques do tipo *DHCP starvation* no servidor de DHCP conectado ao *switch core*.

5.4 *Rogue DHCP server*

Detalhada na seção 2.3.3.2, a ameaça de *rogue DHCP server* se concretiza a partir da inserção de um servidor DHCP falso na rede.

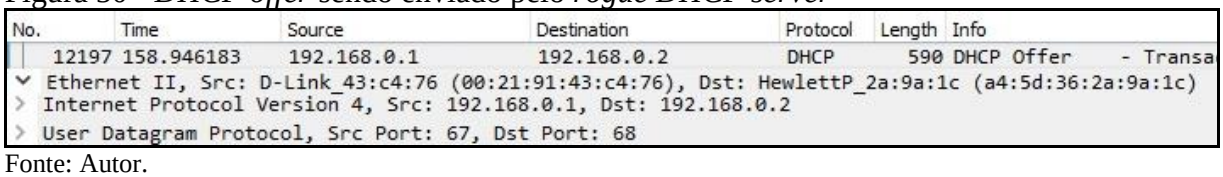
Diferentemente dos outros testes de ataque realizados, para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido, foi utilizado o próprio *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura e um roteador modelo D-Link DIR-100.

O roteador D-Link DIR-100, que possui endereço MAC 00:21:91:43:C4:76, foi configurado com o endereço IP 192.168.0.1 e serviço de DHCP foi habilitado neste com um alcance de endereços IP ofertados para os clientes indo de 192.168.0.2 até 192.168.0.250.

Com o roteador configurado como servidor de DHCP, o ataque de *rogue DHCP server* foi executado ao conectar este na porta 9 do *switch core* da sede administrativa da Prefeitura.

Como resultado desta ação realizada, foi ligado um novo computador na rede sem endereço IP e este, após enviar um pacote de *DHCP request* para o *broadcast* requisitando um endereço IP válido do servidor de DHCP local, recebeu como resposta um pacote de *DHCP offer* do servidor de DHCP falso. A Figura 50 ilustra esse pacote sendo recebido pela interface de rede do computador.

Figura 50 - *DHCP offer* sendo enviado pelo *rogue DHCP server*



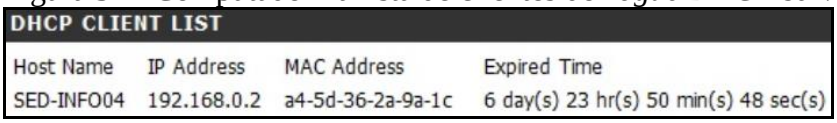
Fonte: Autor.

A partir da Figura 50 ainda se observa que o campo de fonte (*source*) do pacote vem com os endereços IP e MAC do roteador D-Link DIR-100, enquanto que o campo de destino (*destination*) vem com endereço MAC da interface de rede deste computador e o endereço IP que foi ofertado pelo *rogue DHCP server*. Ainda é importante observar que o protocolo

utilizado para transporte do pacote é o UDP, empregando a porta 67 para fonte (*source*) e a porta 68 para destino (*destination*). Este entendimento das portas utilizadas será essencial para explicar a defesa realizada na seção 4.4.1.

Como resultado prático do ataque, o computador terá atribuído a sua interface de rede um endereço IP falso, conforme demonstrado pela Figura 51 que mostra este computador na lista de clientes do servidor DHCP falso.

Figura 51 - Computador na lista de clientes do *rogue DHCP server*



DHCP CLIENT LIST			
Host Name	IP Address	MAC Address	Expired Time
SED-INFO04	192.168.0.2	a4-5d-36-2a-9a-1c	6 day(s) 23 hr(s) 50 min(s) 48 sec(s)

Fonte: Autor.

Com a execução do ataque de *rogue DHCP server*, é possível causar um DoS, já que todos os equipamentos que obterem um endereço IP falso do servidor DHCP falso terão negados qualquer serviço que exija a utilização de um endereço IP válido na rede, ou um *man-in-the-middle*, já que o servidor DHCP falso pode interceptar, registrar e alterar os pacotes dos seus clientes. Com estes dados é possível observar que o ataque de servidor DHCP falso, ou *rogue DHCP server*, foi executado com sucesso na infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

5.4.1 Aplicando defesa ao *rogue DHCP server* na infraestrutura operacional da Prefeitura

De acordo com Extreme Networks (2011), o *software* ExtremeXOS possui o recurso de *DHCP snooping* e o comando de *trusted-port*, que combinados permitem que apenas o servidor de DHCP nas portas selecionadas envie endereços IP aos *hosts* da rede. Para habilitar o *DHCP snooping* é necessário informar o seguinte comando:

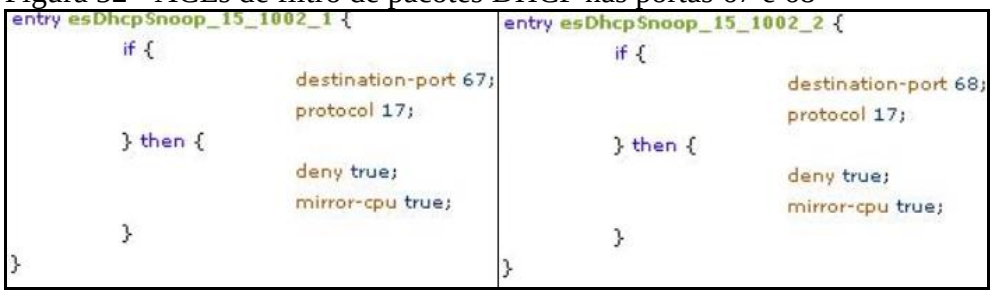
enable ip-security dhcp-snooping vlan <nome_da_vlan> ports <portas> violation-action drop-packet

Este comando tem por finalidade filtrar a mensagens de DHCP no *switch*. Todas as portas que não forem especificadas como confiáveis (*trusted*) através deste comando serão marcadas como não confiáveis (*untrusted*) ao habilitar o *DHCP snooping* nelas, e, portanto, os pacotes de DHCP nelas serão descartados. Com as portas configuradas com o *DHCP*

snooping, o *switch* vai examinar (*snoop*) os pacotes DHCP nas portas indicadas e construir uma base de dados de vínculos DHCP (*DHCP bindings database*) com o endereço IP, endereço MAC, ID de VLAN e número de porta vinculados aos pacotes recebidos. O *switch* encaminhará apenas pacotes DHCP de portas marcadas como confiáveis (*trusted*), enquanto que descartará os pacotes DHCP das demais portas com o *DHCP snooping* habilitado.

A fim de melhor entender como o *switch* descobre quais são os pacotes de DHCP, cabe informar que a implementação do *DHCP snooping* criará automaticamente listas de controle de acesso (ACLs) para cada uma das portas em que ele for configurado no *switch*. Essas ACLs definirão as regras de permissão para tráfego dos pacotes de DHCP. Conforme foi demonstrado na Figura 50, da seção 5.4, as portas utilizadas para trafegar pacotes DHCP são a 67 e a 68. Logo, o *switch* verificará se estas são as portas presentes no cabeçalho do pacote e tomará a devida ação para estes pacotes, conforme é demonstrado na Figura 52, obtida através da interface gráfica do *switch core*, que ilustra essas regras aplicadas em uma porta do *switch core* após aplicação do *DHCP snooping*.

Figura 52 - ACLs de filtro de pacotes DHCP nas portas 67 e 68



Fonte: Autor.

Para remover o *DHCP snooping* de uma porta é necessário informar o seguinte comando:

disable ip-security dhcp-snooping vlan <nome_da_vlan> ports <portas>

Para marcar uma porta como sendo confiável (*trusted*) para tráfego de pacotes DHCP é necessário executar o seguinte comando:

configure trusted-ports <número_da_porta> trust-for dhcp-server

Este comando fará com que todos os pacotes DHCP recebidos ou enviados por esta porta sejam encaminhados sem nenhum bloqueio.

Para demarcar uma porta como sendo confiável (*trusted*) no *switch* é necessário executar o seguinte comando:

unconfigure trusted-ports <número_da_porta> trust-for dhcp-server

No *switch core* da infraestrutura operacional da Prefeitura foi habilitado o *DHCP snooping* em todas as portas e VLANs com o comando já explicado anteriormente. Como por exemplo, para habilitar o *DHCP snooping* em todas as portas em duas VLANs específicas (de nomes “DMZ” e” SAUDE”) foram executados os seguintes comandos:

enable ip-security dhcp-snooping vlan DMZ ports all violation-action drop-packet

enable ip-security dhcp-snooping vlan SAUDE ports all violation-action drop-packet

Em seguida, foram configuradas como confiáveis (*trusted*) as portas 02 e 04, em que se conectam o servidor de DHCP (junto com o *firewall*) e o servidor de DHCP reserva da rede, com a execução do seguinte comando:

configure trusted-port 02 trust-for dhcp-server

configure trusted-port 04 trust-for dhcp-server

Ainda seguindo o que é informado por Extreme Networks (2011), para mostrar as configurações de *DHCP snooping* e as portas confiáveis (*trusted*) é necessário usar o comando “*show ip-security dhcp-snooping <nome_da_vlan>*”. A Figura 53 exibe estas configurações após a execução deste comando na VLAN de nome “SAUDE”.

Figura 53 - *DHCP snooping* aplicado na VLAN “SAUDE” do *switch core*

```
(Software Update Required) * X460-24t.24 # show ip-security dhcp-snooping SAUDE
DHCP Snooping enabled on ports: 2, 4, 7, 8, 11, 13,
                                14, 15, 16, 18, 19, 23, 24
Trusted Ports: 2, 4
```

Fonte: Autor.

Para testar a efetividade dessa proteção aplicada, foi repetido o ataque de *rogue DHCP server* na infraestrutura operacional da Prefeitura. Para tal execução ocorrer foi conectado o mesmo roteador D-Link DIR-100 com servidor de DHCP habilitado na porta 15 do *switch core*.

Agora o ataque já não funciona mais, conforme demonstrado pela Figura 54, onde o endereço MAC do servidor DHCP falso levantado na porta 15 é marcado como uma violação e os seus pacotes DHCP são descartados. Para verificar as violações de servidor DHCP é

necessário executar o comando "*show ip-security dhcp-snooping violations vlan <nome_da_vlan>*".

Figura 54 - Violação de servidor DHCP na porta 15 do *switch core*

```
(Software Update Required) * X460-24t.93 # show ip-security dhcp-snooping violat
ions vlan "PUBLICA-PROJFIBRA"
-----------------------------------
Port              Violating MAC
-----------------------------------
15              00:21:91:43:c4:76
```

Fonte: Autor.

5.4.2 Análise dos resultados de prevenção ao *rogue DHCP server*

Os testes de efetividade aplicados demonstram que a aplicação do recurso de *DHCP snooping* aliado a definição de portas confiáveis (*trusted*) mitiga com sucesso a ameaça de *rogue DHCP server*. Com essa proteção sendo aplicada em todas as portas e configurada corretamente nas portas que se conectam ao servidor de DHCP, é possível afirmar que a ameaça de *rogue DHCP server* foi mitigada com sucesso nas portas do *switch core*.

Com estas prevenções aplicadas é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques do tipo *rogue DHCP server*.

5.5 Ataque ao protocolo ARP (*ARP spoofing*)

Detalhado na seção 2.3.4, a ameaça de *ARP spoofing* se concretiza a partir do envio de endereços IP ou MAC falsos através de pacotes forjados do tipo ARP, fraudando a tabela ARP dos *hosts* da rede através da substituição do endereço MAC verdadeiro relacionado ao endereço IP pelo endereço MAC do atacante.

Para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido, foi utilizado um ambiente de testes composto por um *switch* 3Com SuperStack3 3300 XM 3C16985B, 3 PCs e um roteador modelo D-Link DIR-100. Neste *switch* foram conectados o PC "A" na porta 2, que executará o comando de *ping* para o PC "B" na porta 3, o PC "C" na porta 4, a partir do qual será executado o ataque, e o roteador na porta 1 para distribuir endereços IP dinamicamente para cada um dos *hosts* da rede. A Figura 36, da seção 5.2, apresenta a estrutura do ambiente de testes com os específicos endereços MAC e endereços IP de cada *host* na rede.

O PC “A” ficará executando o comando *ping* para o endereço IP do PC “B” a fim de simular uma rede em funcionamento, observando o comportamento desta após o ataque ser executado em ambos os *switches*. Enquanto o *ping* está sendo executado, o PC “C” atacante não consegue observar os pacotes do protocolo ICMP da execução do *ping* trafegando pela rede antes do ataque ser executado, pois eles são pacotes *unicast* endereçados apenas ao MAC do PC “B” e de resposta do PC “B” ao PC “A”.

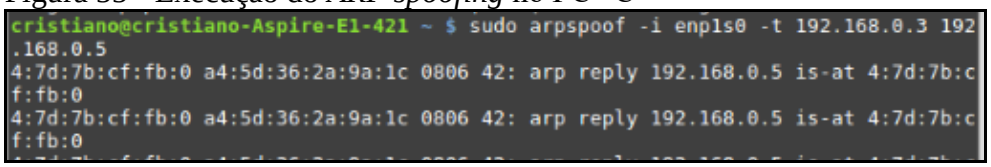
O primeiro passo da execução do ataque foi de descobrir o endereço MAC do alvo. Para tal, o atacante pode executar o comando *ping* para um endereço IP ou para um domínio (neste caso atentando para o endereço IP que responde) existente na rede local. Após é necessário executar o comando “*arp -a*”. Este comando lista a tabela ARP no computador do atacante, onde é apresentada toda a relação de endereços MAC vinculados ao endereço IP da tabela ARP, incluindo o do alvo desejado.

Com o endereço MAC do alvo obtido, para PC “C” poder executar a técnica de *ARP spoofing* é primeiro necessário habilitar o encaminhamento de pacotes na interface de rede que será utilizada para o ataque, através da execução do comando “*sudo bash -c ‘echo “1” > /proc/sys/net/ipv4/ip_foward*”. Depois, foi instalada a biblioteca de ferramentas “*dsniff*” em sistema operacional Linux. Desta biblioteca foi utilizada a ferramenta “*arpspoof*” através do seguinte comando:

arpspoof -i <interface> -t <endereço IP do alvo> <endereço IP que o alvo vai mapear para o endereço MAC do atacante>

A execução deste comando fará com que o atacante envie um pacote do protocolo ARP para o endereço MAC do alvo informando que o endereço MAC do endereço IP que será falsificado agora é o endereço MAC do atacante. Para execução do ataque foi utilizado o seguinte comando da Figura 55 no PC “C”, que demonstra parte da saída na tela feita pela execução do comando “*arpspoof*”, utilizando o PC “A” como alvo (o que terá a tabela ARP alterada) e o PC “B” como o que terá o vínculo de endereço IP alterado para o endereço MAC do PC “C” na tabela ARP do PC “A”.

Figura 55 - Execução do *ARP spoofing* no PC “C”



Fonte: Autor.

O resultado da ação realizada é o envio de pacotes do protocolo ARP demonstrado na Figura 56, onde o PC “C” está informando ao PC “A” que o endereço IP do PC”B” agora corresponde ao endereço MAC do PC “C”.

Figura 56 - Pacote ARP de falsificação enviado pelo PC “C”

No.	Time	Source	Destination	Protocol	Length	Info
112	71.246392472	QuantaCo_cf:fb:00	HewlettP_2a:9a:1c	ARP	42	192.168.0.5 is at 04:7d:7b:cf:fb:00
113	73.246868878	QuantaCo_cf:fb:00	HewlettP_2a:9a:1c	ARP	42	192.168.0.5 is at 04:7d:7b:cf:fb:00

Fonte: Autor.

Como resultado prático do ataque, o PC “A” atualiza a sua tabela ARP, alterando o vínculo do endereço IP do PC “B” para o endereço MAC do PC “C”, de acordo com o que é demonstrado na Figura 57.

Figura 57 - Tabela ARP do PC “A” antes e depois do ataque de *ARP spoofing*

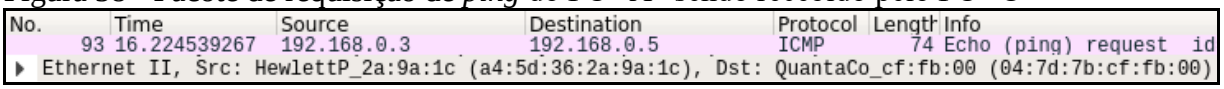
```
C:\Users\User>arp -a

Interface: 192.168.0.3 --- 0xd
  Endereço IP           Endereço físico       Tipo
  192.168.0.2           00-21-91-43-c4-76     dinâmico
  192.168.0.5           6c-0b-84-96-5e-a9     dinâmico

C:\Users\User>arp -a

Interface: 192.168.0.3 --- 0xd
  Endereço IP           Endereço físico       Tipo
  192.168.0.2           00-21-91-43-c4-76     dinâmico
  192.168.0.4           04-7d-7b-cf-fb-00     dinâmico
  192.168.0.5           04-7d-7b-cf-fb-00     dinâmico
```

Fonte: Autor.

Como resultado do ataque, o PC “A” continua executando o comando *ping* para o IP 192.168.0.5 sem perder pacotes durante o ataque, porém, agora o *ping* não é mais recebido e devolvido pelo endereço MAC do PC “B”, mas sim pelo endereço MAC do PC “C” atacante. Abrindo outro terminal, foi possível repetir o mesmo ataque com os endereços IP trocados no comando, assim alterando a tabela ARP do PC “B” e fazendo com que o PC “C” receba os pacotes de ambos os PCs na rede e encaminhe eles à ambos, realizando um *man-in-the-middle*. O encaminhamento de pacotes ocorre pois ele foi habilitado anteriormente com o comando “*echo “1” > /proc/sys/net/ipv4/ip_forward*”, descrito neste documento antes do ataque de *ARP spoofing* em si ser executado, e devido a tabela ARP do PC “C” estar com os endereços MAC dos PCs ainda corretos.

Para ilustrar este processo, a Figura 58 demostra um pacote de requisição (*request*) de *ping* sendo enviado pelo PC “A” para o endereço IP do PC “B”, mas sendo recebido pelo PC “C”.

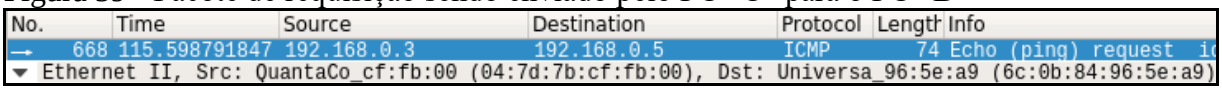
Figura 58 - Pacote de requisição de *ping* do PC “A” sendo recebido pelo PC “C”



Fonte: Autor.

A Figura 59 demonstra o pacote de requisição (*request*) de *ping*, originário do PC “A”, sendo enviado pelo PC “C” para o PC “B”.

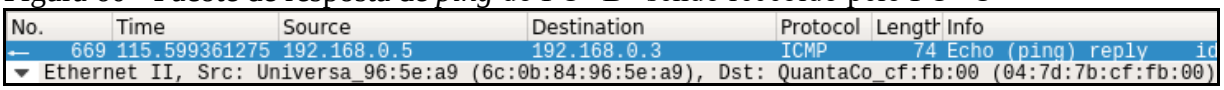
Figura 59 - Pacote de requisição sendo enviado pelo PC “C” para o PC “B”



Fonte: Autor.

Já a Figura 60 demonstra o pacote de resposta (*reply*) do comando *ping* sendo enviado do PC “B” para o endereço IP do PC “A”, porém sendo recebido pelo PC “C”. Posteriormente este pacote é encaminhado pelo PC “C” ao PC “A”. Desta maneira, o processo será repetido para todos os pacotes enviados entre ambos os PCs.

Figura 60 - Pacote de resposta de *ping* do PC “B” sendo recebido pelo PC “C”



Fonte: Autor.

Com a execução do ataque de *ARP spoofing* e sem habilitar o encaminhamento de pacotes é possível causar um DoS, já que qualquer serviço do PC “B” será negado ao PC “A”, ou, habilitando o encaminhamento de pacotes, é possível executar um *man-in-the-middle*, já que o PC “C” pode interceptar, registrar e alterar todos os pacotes transmitidos entre o PC “A” e o PC “B”. Com estes dados é possível observar que o ataque de *ARP spoofing* foi executado com sucesso neste ambiente de teste vulnerável apresentado.

5.5.1 Aplicando defesa ao *ARP spoofing* na infraestrutura operacional da Prefeitura

Diferentemente dos testes de ataque, as proteções contra *ARP spoofing* serão aplicadas no *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

De acordo com Extreme Networks (2011), o *software* ExtremeXOS possui o comando de *ARP validation* que utiliza a base de dados de vínculos de DHCP (*DHCP binding database*) para verificar a validade dos pacotes ARP que trafegam no *switch*. A base de dados

de vínculos de DHCP (*DHCP binding database*) já foi criada anteriormente a partir da implementação do *DHCP snooping* para mitigar o *rogue DHCP server* descrita na seção 5.4.1, portanto esta já existe no *switch core*. Já para executar o *ARP validation* é necessário informar o seguinte comando:

enable ip-security arp validation <nome_da_vlan> ports <portas> violation-action drop-packet

Este comando está ligado diretamente com o recurso de *DHCP snooping* e tem por finalidade usar a base de dados de vínculos DHCP (*DHCP bindings database*) para validar as entradas de pacotes do protocolo ARP nas portas especificadas.

Ainda seguindo o que é informado por Extreme Networks (2011), a fim de executar esta validação, o *ARP validation* segue várias regras executando pesquisas na base de dados de vínculos DHCP, como validando se nos pacotes ARP de requisição (*request*) o endereço MAC de origem *Ethernet* não corresponde ao endereço de *hardware* de origem, e, se o endereço IP de origem ou de destino é igual a um *multicast*, ou se o endereço IP de origem não está presente na base de dados DHCP ou se o endereço IP de origem existe na base de dados DHCP mas o endereço de *hardware* de origem não corresponde ao endereço MAC da entrada na base dados DHCP. Já para os pacotes ARP de resposta (*response*) será validado se o endereço IP de origem não está presente na base de dados DHCP ou se é presente, mas o endereço de *hardware* de origem não corresponde ao endereço MAC da entrada na base de dados DHCP, ou se o endereço IP de origem ou de destino não é igual a um *multicast*, ou se o endereço MAC de origem ou destino *Ethernet* não corresponde ao endereço de *hardware* de origem ou de destino.

É aplicando estas regras na base de dados de vínculos DHCP que o *switch* saberá quais pacotes ARP são válidos ou não.

Para remover o *ARP validation* é necessário executar o seguinte comando:

disable ip-security arp validation <nome_da_vlan> <portas>

Foi utilizado este comando de *ARP validation* para habilitar a validação de pacotes ARP em todas as portas e VLANs do *switch core*. Como por exemplo, para habilitar o *ARP validation* em todas as portas em duas VLANs específicas foram executados os seguintes comandos:

enable ip-security arp validation ESCOLAS portas all violation-action drop-packet

enable ip-security arp validation SAUDE portas all violation-action drop-packet

Para visualizar as configurações de *ARP validation* é necessário utilizar o comando "*show ip-security arp validation* <*nome_da_vlan*>". A Figura 61 exibe parte destas configurações, após a execução deste comando na VLAN de nome "SAUDE".

Figura 61 - Parte da tela que mostra a aplicação do *ARP validation*

```
-------------------------------------------------------------------
Port            Validation                      Violation-action
-------------------------------------------------------------------
2       DHCP                                    drop-packet
4       DHCP                                    drop-packet
7       DHCP                                    drop-packet
10      DHCP                                    drop-packet
```

Fonte: Autor.

Para testar a efetividade dessa proteção aplicada, foi repetido o ataque de *ARP spoofing* na infraestrutura operacional da Prefeitura. Para tal execução ocorrer foi conectado um notebook na porta 16 e utilizados os endereços IP 10.1.0.1 e 10.1.0.5 de dois servidores conectado ao *switch core* para que o notebook aplique o ataque.

Agora o ataque já não funciona mais, conforme demonstrado pela Figura 62, a qual foi extraída dos *logs* de eventos da interface *web* do *switch core* e mostra que uma violação de ARP foi detectada na porta 16, trazendo ainda os endereços IP e MAC que estão nesse pacote de violação. Apesar de não aparecer na Figura 62, a mensagem apresentada ainda traz a informação de que o tipo de violação é de vínculo inválido de endereços IP e MAC.

Figura 62 - *Logs* de violação de ARP ao aplicar o *ARP spoofing*

```
<Warn:ipSecur.arpViol> An ARP violation was detected on vlan Default port 16 violating IP 10.1.0.5 violating MAC 04:7D:7B:CF:FB:00
<Warn:ipSecur.arpViol> An ARP violation was detected on vlan Default port 16 violating IP 10.1.0.5 violating MAC 04:7D:7B:CF:FB:00
```

Fonte: Autor.

5.5.2 Análise dos resultados de prevenção ao *ARP spoofing*

Os testes de efetividade aplicados demonstram que a aplicação do recurso de *ARP validation*, aliado ao uso do *DHCP snooping*, mitiga com sucesso a ameaça de *ARP spoofing*. Com essa proteção aplicada em todas as portas é possível afirmar que a ameaça de *ARP spoofing* foi mitigada com sucesso nas portas do *switch core*.

Assim como na seção 5.2.2, aqui também cabe informar que o uso de tabelas ARP estáticas não é viável para implementação na rede do ambiente operacional da Prefeitura. Visto o tamanho e complexidade dela, este procedimento acabaria gerando tanta carga administrativa que limitaria muito a dinamicidade do ambiente de trabalho, gerando um gigantesco entrave de acesso à rede e tornando tal procedimento impraticável.

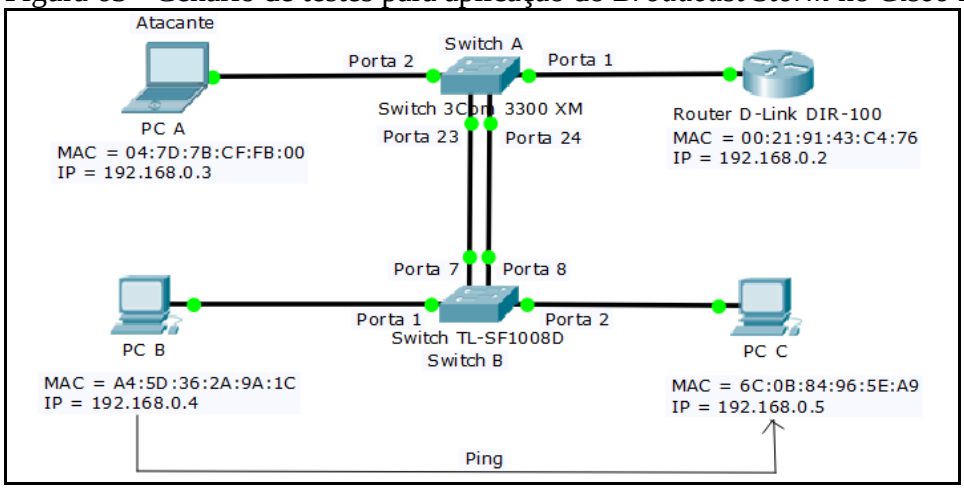
Com estas prevenções aplicadas é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques do tipo *ARP spoofing*.

5.6 Tempestade de *broadcast* (*broadcast storm*)

Detalhada na seção 2.3.5, a ameaça de tempestade de *broadcast*, ou *broadcast storm*, se concretiza a partir do envio de um número excessivo de pacotes do tipo ICMP para o endereço de *broadcast*, com a função de inundar a rede, deixando-a inutilizável.

Para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido, foi utilizado um ambiente de testes composto por um *switch* 3Com SuperStack3 3300 XM 3C16985B (que será o *switch* "A"), um *switch* TP-Link TL-SF1008D (que será o *switch* "B"), 3 PCs e um roteador modelo D-Link DIR-100. No *switch* "A" foram conectados o roteador na porta 1, para distribuir endereços IP dinamicamente para cada um dos *hosts* da rede, e o PC "A" na porta 2, a partir do qual será executado o ataque. Já no *switch* "B" foram conectados o PC "B" na porta 1 e o PC "C" na porta 2. No PC "B" foi executado o comando de *ping* para o PC "C", a fim de simular o funcionamento de uma rede local em operação. Os *switches* foram interconectados entre si, conectando a porta 23 do *switch* "A" à porta 7 do *switch* "B" e a porta 24 do *switch* "A" à porta 8 do *switch* "B". Em ambos os *switches* foi desabilitado o STP. A Figura 63 apresenta a estrutura do ambiente de testes.

Figura 63 - Cenário de testes para aplicação do *Broadcast Storm* no *Cisco Packet Tracer*



Fonte: Autor.

O ataque de *broadcast storm* foi executado pelo PC "A", em sistema operacional Linux, através do comando "*ping -b -f <endereço_de_broadcast>*", onde o endereço IP de broadcast é 255.255.255.255. O parâmetro "*-b*" permite que seja executado o comando *ping* para um endereço de *broadcast*, já o parâmetro "*-f*" faz com o comando *ping* executado crie uma inundação (*flood*), enviando um grande número de pacotes em um curto tempo, ou de forma mais precisa, os pacotes de saída do *ping* serão enviados tão rápido quanto eles voltam ou uma centena de vezes por segundo, o que for mais. A execução deste comando, aliado a má configuração de conexão dos *switches* e a ausência de STP, faz com que o PC "A" imediatamente injete uma grande quantidade de pacotes de *broadcast*, os quais serão respondidos e se multiplicarão de forma a inundar a rede. A Figura 64 ilustra uma pequena parte dos pacotes de *broadcast* do protocolo ICMP inundando a rede local, destinados ao endereço MAC de *broadcast* "FF:FF:FF:FF:FF:FF".

Figura 64 - Pacotes de *broadcast* de *ping* inundando a rede

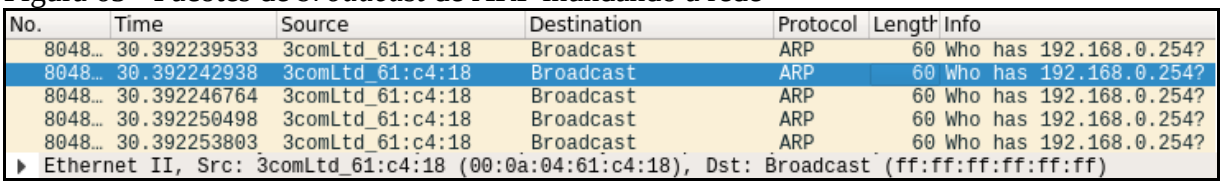
No.	Time	Source	Destination	Protocol	Length	Info
1086…	170.418501027	192.168.0.3	255.255.255.255	ICMP	98	Echo (ping) req
1086…	170.430531168	192.168.0.3	255.255.255.255	ICMP	98	Echo (ping) req
1086…	170.442479236	192.168.0.3	255.255.255.255	ICMP	98	Echo (ping) req
1086…	170.451006640	192.168.0.3	255.255.255.255	ICMP	98	Echo (ping) req
1086…	170.451039956	192.168.0.3	255.255.255.255	ICMP	98	Echo (ping) req
1086…	170.454472633	192.168.0.3	255.255.255.255	ICMP	98	Echo (ping) req

▸ Ethernet II, Src: QuantaCo_cf:fb:00 (04:7d:7b:cf:fb:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Internet Protocol Version 4, Src: 192.168.0.3, Dst: 255.255.255.255

Fonte: Autor.

Já a Figura 65 demonstra como o *switch* "A" tenta lidar com essa quantidade imensa de pacotes *broadcast* ao enviar ainda mais mensagens destinadas ao *broadcast* pelo protocolo ARP, com a intenção de descobrir para onde endereçar estes pacotes de ICMP.

Figura 65 - Pacotes de *broadcast* de ARP inundando a rede



Fonte: Autor.

Como resultado do ataque, após determinado tempo, a memória de ambos os *switches* transbordou e eles pararam de encaminhar qualquer pacote, cancelando todo o tráfego na rede local (incluindo o *ping* entre o PC “B” e o PC “C”).

Com a execução do *broadcast storm* é possível causar um DoS, já que qualquer serviço que tente ser acessado na rede será negado devido ao fato dos *switches* pararem de encaminhar qualquer pacote. Com esses dados é possível observar que o ataque de tempestade de *broadcast*, ou *broadcast storm*, foi executado com sucesso neste ambiente de teste vulnerável apresentado.

5.6.1 Aplicando defesa ao *broadcast storm* na estrutura operacional da Prefeitura

Diferentemente dos testes de ataque, as proteções contra *MAC address table overflow* serão aplicadas no *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

Uma das técnicas de defesa está na existência de 13 VLANs, já configuradas em um momento anterior a este projeto no *switch core* e em funcionamento atualmente, que dividem as LANs pelas funções exercidas em cada uma, conforme pode ser observado na Tabela 1. Esta segmentação da rede em várias VLANs permite que o *broadcast storm* fique contido a cada VLAN, isolando assim o domínio de execução do ataque.

Além desta configuração de VLAN, outra técnica de defesa executada foi a aplicação correta do STP. Primeiramente, para criar o domínio do protocolo de *spanning tree* é necessário informar o seguinte comando:

configure stpd <domínio_de_STP> mode dot1w

Além de criar um domínio de STP, Extreme Networks (2011) explica que este comando define o modo de operação do domínio de *spanning tree* para o 802.1w, que é utilizado para compatibilidade com o RSTP.

Posteriormente, para determinar o STP do domínio criado em cada VLAN e porta, é necessário executar o seguinte comando:

configure stpd <domínio_de_STP> add vlan <nome_da_VLAN> ports <portas>

Para remover o STP de uma VLAN em uma porta é necessário executar o seguinte comando:

c*onfigure stpd <domínio_de_STP> delete vlan <nome_da_VLAN> ports <portas>*

No *switch core* da infraestrutura operacional da Prefeitura foi habilitado o STP de nome de domínio "s0" nas VLANs configuradas das portas 18, 19, 20, 21 e 22, que estão conectados aos demais *switches* da sede administrativa. Como por exemplo, para habilitar o STP em todas as portas citadas na VLAN "ESCOLAS" foram executados os seguintes comandos:

configure stpd "s0" add vlan "ESCOLAS" ports 18

configure stpd "s0" add vlan "ESCOLAS" ports 19

configure stpd "s0" add vlan "ESCOLAS" ports 20

configure stpd "s0" add vlan "ESCOLAS" ports 21

configure stpd "s0" add vlan "ESCOLAS" ports 22

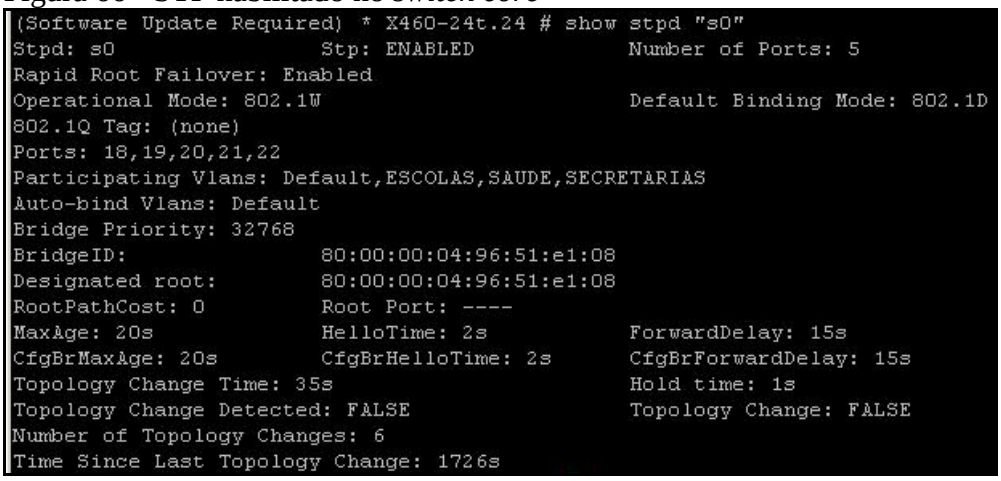
Por fim, para habilitar o STP e colocá-lo em operação, é necessário executar o seguinte comando:

enable stpd <domínio_de_STP> rapid-root-failover

Segundo Extreme Networks (2011), este comando também habilita o *rapid root failover*, para tempos de recuperação de falhas de STP mais rápidos. Assim, se o *link* da porta raiz (*root*) ativa cair, o *switch* recalcula o STP e elege uma nova porta raiz (*root*) que inicia imediatamente o encaminhamento, ignorando as fases de audição e aprendizado.

A Figura 66 exibe as configurações de STP aplicadas, apresentadas através da execução do comando "*show stpd <domínio_de_STP>*".

Figura 66 - STP habilitado no *switch core*



Fonte: Autor.

Por fim, a última técnica de defesa é a que o *software* ExtremeXOS possui com o comando de *rate-limit flood*, que, de acordo com Extreme Networks (2011), define um limite de pacotes do tipo escolhido que podem trafegar em uma porta por segundo, conhecido também como técnica de *storm control*. Para executar o *rate-limit flood* para limitação de pacotes *broadcast* é necessário informar o seguinte comando:

configure ports <portas> rate-limit flood broadcast <pps>

Este comando tem por finalidade fazer com que o número de pacotes de *broadcast* que ingressam no *switch* sejam limitados ao valor de pacotes por segundo (*Packets Per Second*, PPS) informado na execução do comando. Quando essa taxa de limite informada for excedida, a porta irá bloquear o tráfego deste tipo de pacotes, descartando eles até que o tráfego deste tipo de pacote fique novamente abaixo da taxa de limite configurada. Lembrando que o *switch* identificará os pacotes de *broadcast* através do endereço MAC de destino existente no cabeçalho dos pacotes.

Este comando também pode ser utilizado para limitar o tráfego de pacotes *multicast*, apenas substituindo no comando a palavra *broadcast* por *multicast*.

Para remover o limite de pacotes *broadcast* por segundo é necessário informar o seguinte comando:

configure ports <portas> rate-limit flood broadcast no-limit

Foi executado este comando para fixar uma taxa de limite de tráfego de pacotes *broadcast* e *multicast* em cada uma das portas do *switch core*. Cada porta teve o valor de limite configurado de acordo com sua funcionalidade e condizente com o tráfego de

broadcast que já atuava nessas antes da configuração ser realizada. Como por exemplo, para habilitar o *rate-limit flood* na porta 13 foram executados os seguintes comandos:

configure port 13 rate-limit flood broadcast 10000

configure port 13 rate-limit flood multicast 5000

Para testar a efetividade destas proteções aplicadas, foi repetido o ataque de *broadcast storm* na infraestrutura operacional da Prefeitura. Para realizar tal ataque foi conectado um notebook com Linux na porta 15 do *switch core*. Após conexão, foi executado novamente o comando "*ping -b -f 255.255.255.255*".

Agora o ataque já não funciona mais, como apresentado na Figura 67, onde, através do comando "*show ports rate-limit flood no-refresh*" é possível visualizar o grande número de pacotes *broadcast* descartados alguns segundos depois da execução do ataque na porta 15, os quais são apresentados na coluna de *Flood Rate Exceeded* (taxa de inundação excedida) pelo número de 27866 pacotes, sendo que este número tem a tendência de crescer constantemente conforme o ataque continuar sendo executado.

Figura 67 - Quantidade de pacotes de *Flood Rate Exceeded* na porta 15

Port Rate-Limit Discard Monitor

Port	Link State	Rx Pkt Count	Rx Byte Count	Rx Pkt Bcast	Rx Pkt Mcast	Flood Rate Exceeded
15	A	18883796	5461666768	18870609	3431	27866

Fonte: Autor.

Paralelamente a isto, cabe informar que o *switch core* também cria um *log* que registra que o *rate-limit flood* foi ativado na porta 15, conforme demonstrado pela Figura 68, extraída dos *logs* de eventos da interface *web* do *switch core*.

Figura 68 - *Log* de ativação de *rate-limit flood* na porta 15

```
03/31/2017 07:27:18.90 <Info:HAL.Port.RateLimit> Flood Rate Limiting activated on Port 15
```

Fonte: Autor.

5.6.2 Análise dos resultados de prevenção ao *broadcast storm*

Os testes de efetividade aplicados demonstram que a implementação de VLANs na rede, aliados a correta aplicação do STP e a especificação de uma taxa limite de pacotes *broadcast* permitidos em cada porta do *switch*, mitiga com sucesso a ameaça de *broadcast*

storm. Assim, com estas proteções citadas tendo sido aplicadas, é possível afirmar que a ameaça de *broadcast storm* foi mitigada com sucesso nas portas do *switch core*.

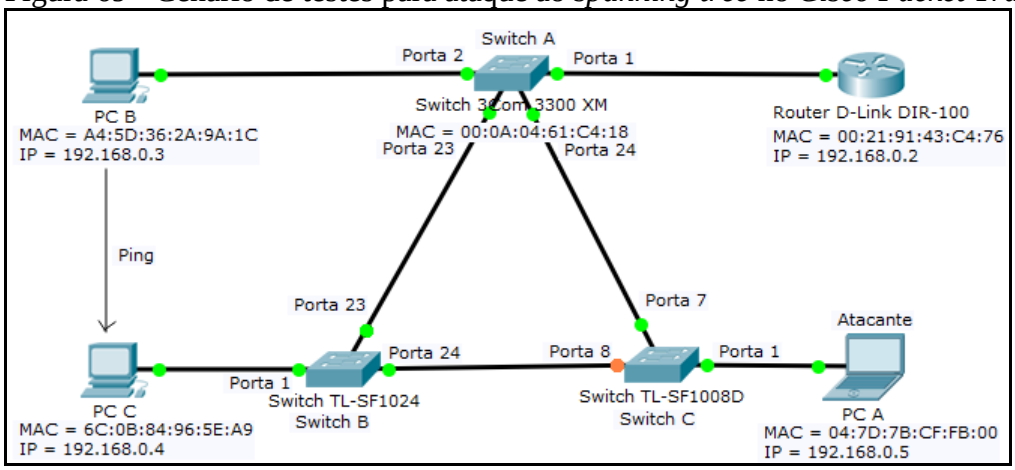
Com estas prevenções aplicadas é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques do tipo *broadcast storm* no *switch core*.

5.7 Ataque ao *spanning tree*

Detalhado na seção 2.3.6, a ameaça de ataque ao *spanning tree* se concretiza ao enviar para a rede local pacotes falsos do tipo BPDU, informando que a ponte raiz (*root bridge*) é o endereço MAC do atacante. Isto forçará os *switches* vulneráveis a recalcular o STP para incluir a nova ponte raiz (*root bridge*) e fará com que toda a rede fique sem conexão.

Para verificar a efetividade da ameaça e a presença da vulnerabilidade que permite um ataque bem sucedido, foi utilizado o ambiente de testes da Figura 69, composto por um *switch* 3Com SuperStack3 3300 XM 3C16985B (o *switch* "A"), um *switch* TP-Link TL-SF1024 (o *switch* "B"), um *switch* TP-Link TL-SF1008D (o *switch* "C"), 3 PCs e um roteador modelo D-Link DIR-100. No *switch* "A" foram conectados o roteador na porta 1, para distribuir endereços IP para cada um dos *hosts* da rede, e o PC "B" na porta 2, que executa o comando *ping* para o PC "C", que está conectado na porta 1 do *switch* "B". No *switch* "C" foi conectado o PC "A" na porta 1, a partir do qual será executado o ataque. Os 3 *switches* foram interconectados entre si, conectando a porta 23 do *switch* "A" à porta 23 do *switch* "B", a porta 24 do *switch* "B" à porta 8 do *switch* "C" e a porta 7 do *switch* "C" à porta 24 do *switch* "A".

Figura 69 - Cenário de testes para ataque ao *spanning tree* no *Cisco Packet Tracer*



Fonte: Autor.

O STP foi habilitado em todos os *switches*, elegendo o switch “A” como raiz (*root bridge*), conforme demonstrado nas configurações de STP do *switch* “A” exibidas na Figura 70.

Figura 70 - STP funcionando com o *switch* “A” de *root bridge*

```
stpState:       enabled          agingTime:      1800

Time since topology change:              0 hrs 20 mins 10 seconds
Topology Changes:                        1
Bridge Identifier:                       8000 000a0461c418
Designated Root:                         8000 000a0461c418
```

Fonte: Autor.

A Figura 71 demonstra um dos pacotes de STP do tipo BPDU, que envia a informação do endereço MAC do *switch* “A” para o endereço MAC do grupo do STP (01:80:C2:00:00:00), informando que o *root bridge* é o endereço MAC do *switch* “A”.

Figura 71 - Pacote do tipo BPDU com o STP funcionando com o *root bridge* correto

```
No.  Time          Source              Destination         Protocol  Length Info
     1 0.000000000 3comLtd_61:c4:18    Spanning-tree-(for-… STP         60 Conf. Root = 32768/0/00:0a:04:61:c4:18
▾ IEEE 802.3 Ethernet
  ▾ Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
      Address: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
  ▾ Source: 3comLtd_61:c4:18 (00:0a:04:61:c4:18)
      Address: 3comLtd_61:c4:18 (00:0a:04:61:c4:18)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
▾ Spanning Tree Protocol
    Protocol Identifier: Spanning Tree Protocol (0x0000)
    Protocol Version Identifier: Spanning Tree (0)
    BPDU Type: Configuration (0x00)
  ▾ BPDU flags: 0x00
      0... .... = Topology Change Acknowledgment: No
      .... ...0 = Topology Change: No
  ▾ Root Identifier: 32768 / 0 / 00:0a:04:61:c4:18
      Root Bridge Priority: 32768
      Root Bridge System ID Extension: 0
      Root Bridge System ID: 3comLtd_61:c4:18 (00:0a:04:61:c4:18)
    Root Path Cost: 0
  ▾ Bridge Identifier: 32768 / 0 / 00:0a:04:61:c4:18
      Bridge Priority: 32768
      Bridge System ID Extension: 0
      Bridge System ID: 3comLtd_61:c4:18 (00:0a:04:61:c4:18)
```

Fonte: Autor.

Para execução do ataque ao *spanning tree*, no PC “A” foi instalada a ferramenta de rede “*Yersinia*”. Após, ela foi inicializada com o comando “*yersinia -I*”, foi selecionada a interface a ser utilizada no ataque pressionando a tecla “i”, pressionada a tecla “g” e escolhido o modo de protocolo STP, pressionada a tecla “x” para abrir o menu de ataques e, por fim, pressionada a tecla “4” para executar a opção de “*Claiming Root Role*” e iniciar o ataque. A Figura 72 ilustra a tela de saída apresentada na execução deste ataque, onde é possível identificar que o PC “A” capturou a informação do endereço MAC do *root bridge* (00:0A:04:61:C4:18) de um pacote de tipo BPDU do STP e após alterou ele levemente,

lançando na rede pacotes de BPDU falsos informando que o novo *root bridge* é o endereço MAC 00:0A:04:60:C4:18.

Figura 72 - Execução do ataque ao *spanning tree* no PC “A”

```
RootId              BridgeId            Port                Iface Last seen
8000.000A0461C418 8000.000A0461C418 800C                enp1s005 Apr 18:27:24
8000.000A0460C418 8000.000A0460C418 800C                enp1s005 Apr 18:29:49
8000.000A0460C418 8000.000A0461C418 8018                enp1s005 Apr 18:27:25
```

Fonte: Autor.

O resultado das ações realizadas acima é demonstrado na Figura 73, que apresenta um dos pacotes de BPDU falsos, que foi recebido pelo *switch* “A”, sendo repassado por ele para toda a rede local a fim de informar que o novo *root bridge* é o de endereço MAC falso.

Figura 73 - Pacotes de BPDU com o *root bridge* falso

```
No.     Time        Source                Destination           Protocol  Length Info
     53 19.152573   3com_61:c4:18         Spanning-tree-(for-... STP         60 Conf. Root = 32768/0/00:0a:04:60:c4:18
▲ Spanning Tree Protocol
    Protocol Identifier: Spanning Tree Protocol (0x0000)
    Protocol Version Identifier: Spanning Tree (0)
    BPDU Type: Configuration (0x00)
  ▲ BPDU flags: 0x00
      0... .... = Topology Change Acknowledgment: No
      .... ...0 = Topology Change: No
  ▲ Root Identifier: 32768 / 0 / 00:0a:04:60:c4:18
      Root Bridge Priority: 32768
      Root Bridge System ID Extension: 0
      Root Bridge System ID: 3com_60:c4:18 (00:0a:04:60:c4:18)
    Root Path Cost: 18
  ▲ Bridge Identifier: 32768 / 0 / 00:0a:04:61:c4:18
      Bridge Priority: 32768
      Bridge System ID Extension: 0
      Bridge System ID: 3com_61:c4:18 (00:0a:04:61:c4:18)
```

Fonte: Autor.

Como resultado do ataque, todos os *switches* da rede atualizaram as configurações de STP para eleger o endereço MAC falso como o *root bridg*e, como no *switch* “A” da Figura 74.

Figura 74 - R*oot bridge* falso aplicado ao STP do *switch* “A”

```
stpState:       enabled         agingTime:      1800

Time since topology change:             0 hrs 28 mins 50 seconds
Topology Changes:                       3
Bridge Identifier:                      8000 000a0461c418
Designated Root:                        8000 000a0460c418
```

Fonte: Autor.

Com a execução do ataque ao *spanning tree*, é possível causar um DoS, pois com um *root bridge* falso os *switches* não conseguem encaminhar nenhum pacote (incluindo o *ping* entre o PC “B” e o PC “C”) por cada tempo de recálculo do novo *root bridge*. Como os

pacotes falsos de BPDU são enviados com certa frequência, estes recálculos que causam DoS irão persistir indefinidamente. Com estes dados é possível observar que o ataque ao *spanning tree* foi executado com sucesso no ambiente de testes vulnerável apresentado.

5.7.1 Aplicando defesa ao ataque ao *spanning tree* na infraestrutura operacional da Prefeitura

Diferentemente dos testes de ataque, as proteções contra *DHCP starvation* serão aplicadas no *switch core* Extreme Networks Summit X460-24t da infraestrutura operacional da sede administrativa da Prefeitura Municipal de Lajeado.

Para implementar a defesa, de acordo com Extreme Networks (2011), o *software* ExtremeXOS possui a opção de criação de ACLs. Para realizar tal criação de ACP é executado o comando "*edit policy <nome_da_ACL>.pol*".

Então, foi criada a ACL da Figura 75, que será usada para bloquear todos os pacotes BPDU em uma porta ou VLAN, já que todos eles usam o mesmo endereço MAC 01:80:C2:00:00:00 de destino do grupo de STP. Está informação pode ser observada através da Figura 71 da seção 5.7, onde temos um pacote STP funcionando corretamente na rede. O "*count bpdu*" da ACL serve para gerar *logs* sempre que a ACL executar uma negação de pacote BPDU.

Figura 75 - ACL criada para bloqueio de BPDU

```
entry bpdu {
if {
        ethernet-destination-address 01:80:c2:00:00:00;
    }
    then
    {
        deny;
        count bpdu;
    }
}
```

Fonte: Autor.

Após criada a ACL, está foi aplicada nas portas com o seguinte comando:

configure access-list <nome_da_ACL> ports <portas> ingress

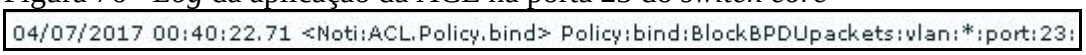
Também é possível aplicar por VLANs, substituindo o "*ports <portas>*" por "*vlan <nome_da_vlan>*" no comando acima.

Já para remover uma aplicação de ACL deve ser utilizado o seguinte comando:

configure access-list delete <nome_da_ACL> ports <portas>

Sempre que uma ACL for aplicada em uma porta, será gerado um *log* do evento, que pode ser conferido na interface *web* do *switch core*, conforme demonstrado na Figura 76.

Figura 76 - *Log* da aplicação da ACL na porta 23 do *switch core*



Fonte: Autor.

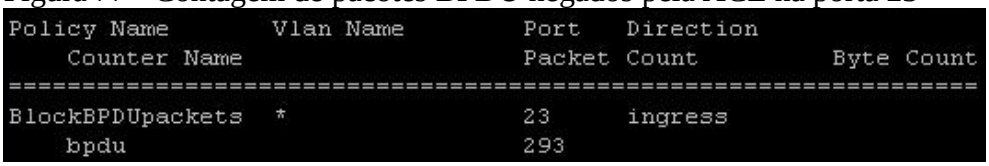
No *switch core*, foi aplicada a ACL em todas as portas, com exceção das portas em que estão conectados os demais *switches* da sede administrativa (as 18, 19, 20, 21 e 22), pois estas portas precisam trafegar os pacotes BPDU para o correto funcionamento do STP. Como exemplo, para aplicar a ACL criada de nome "*BlockBPDUpackets.pol*" na porta 1 foi utilizado o seguinte comando:

configure access-list BlockBPDUpackets.pol ports 1 ingress

Para testar a efetividade dessa proteção aplicada, foi repetido o ataque ao *spanning tree* na infraestrutura operacional da Prefeitura. Lembrando que o STP já está em execução nela e sua configuração de correto funcionamento pode ser observado na Figura 66, da seção 5.6.1. Para repetir a execução do ataque, foi conectado um notebook com Linux na porta 23.

Porém, agora o ataque já não funciona mais e o *root bridge* continuará sendo o mesmo que é apresentado na Figura 66, da seção 5.6.1. Os pacotes de BPDU que foram negados pela ACL na porta 23 podem ser conferidos na Figura 77, cujos os dados foram obtidos através da execução do comando "*show access-list counter ingress*".

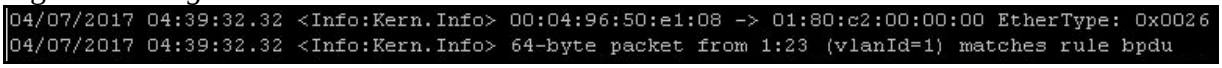
Figura 77 - Contagem de pacotes BPDU negados pela ACL na porta 23



Fonte: Autor.

Já o *log* de evento de acionamento da ACL aplicada pode ser conferido através do comando "*show log*", conforme demonstrado na Figura 78.

Figura 78 - *Log* de evento de acionamento da ACL



Fonte: Autor.

5.7.2 Análise dos resultados de prevenção ao ataque ao *spanning tree*

Os testes de efetividade aplicados demonstram que a criação e aplicação de uma ACL para negação de pacotes BPDU mitiga com sucesso a ameaça de ataque ao *spanning tree*. Com está proteção sendo aplicada nas portas em que não se conectam os demais *switches*, é possível afirmar que esta ameaça foi mitigada com sucesso se executada a partir destas portas.

Com estas prevenções aplicadas, é possível afirmar que a rede da Prefeitura Municipal de Lajeado está mais segura a ataques ao *spanning tree* no *switch core*.

6 CONCLUSÃO

O presente trabalho apresenta um levantamento das principais ameaças existentes a segurança em redes de computadores na camada de enlace de dados de uma rede, buscando soluções para mitigar estas ameaças e propondo uma posterior implementação destas soluções em um ambiente operacional real.

A pesquisa realizada na revisão bibliográfica mostrou aspectos importantes de como as redes de computadores são compostas e do que abrange a segurança nestas redes, a fim de melhor compreender os aspectos das ameaças à camada de enlace de dados que foram elencadas e detalhadas quanto ao seu funcionamento, o que levou a uma consequente investigação dos meios necessários para mitigar cada uma dessas ameaças.

O que foi exposto durante a revisão bibliográfica demonstrou que as ameaças existentes na camada de enlace de dados são pouco abordadas em trabalhos científicos, evidenciando uma relativa negligência com estas quando comparadas às ameaças que passam por outras camadas, vindas de meios externos as redes locais, e justificando a relativa escassez de material sobre estas ameaças. Porém, conforme abordado, os ataques realizados seguem tendências que demonstram que eles sempre acabam se concentrando nos pontos da rede que são mais suscetíveis, ditos como o “elo mais fraco”, que são justamente os que se localizam mais próximos aos usuários finais, evitando os meios mais conhecidos de ataques, já que serão neles que os maiores cuidados com a defesa serão realizados nas empresas. Estas informações comprovaram a importância do estudo e demonstraram uma possível futura tendência em ataques desse tipo.

Foi possível identificar e compreender as principais ameaças à camada de enlace de dados de uma rede local e executar, com sucesso, os procedimentos necessários para tornar cada uma destas ameaças em ataques nos ambientes de testes criados. A partir disto, foi observado que estes ataques identificados podem ser executados de maneira relativamente simples em uma rede local despreparada, o que aumenta o risco destas ameaças serem concretizadas em ataques, enfatizando a importância da execução deste trabalho e da aplicação de técnicas de defesa para cada uma destas ameaças.

Também foi possível encontrar soluções efetivas para implementar defesas para cada um dos ataques elencados e mitigar com sucesso o risco de concretização de cada um destes ataques à camada de enlace de dados, no *switch core* da rede local da Prefeitura Municipal de Lajeado.

Observa-se que o *switch* utilizado para execução do trabalho possuía ferramentas específicas necessárias para a mitigação de cada uma destas ameaças relacionadas (com exceção do ataque ao STP), o que demonstra que as empresas fabricantes destes equipamentos estão cientes do atual cenário de ameaças e consequentemente fornecem recursos necessários para implementar defesas satisfatórias.

Com a sucedida mitigação destes ataques elencados, é possível afirmar que a rede local do cenário onde o trabalho foi aplicado está mais segura às diversas ameaças internas provenientes da camada de enlace de dados (nível 2) do modelo de referência RM-OSI/ISO.

Com estes aspectos relatados em vista, concluísse que todos os objetivos propostos para este trabalho foram atingidos com sucesso durante a execução deste projeto.

Se propõe, para trabalhos futuros, replicar estas técnicas de defesa relatadas nos demais *switches* da sede administrativa e nas demais LANs da Prefeitura Municipal de Lajeado. Devido ao fato de ainda existir a possibilidade de execução destes ataques elencados nestes itens que não são objetos de estudo deste trabalho, porém se conectam ao *switch core*.

Ainda como trabalhos futuros, se indica a execução deste mesmo projeto em outros cenários operacionais diferentes. Por fim, se propõe a execução e mitigação de ataques relacionados a realizar saltos nas VLANs, como o *VLAN hopping*, que é executado através de técnicas de *switch spoofing* e *double tagging*, onde, a partir de uma VLAN inicial, é possível obter acesso ao tráfego em outras VLANs que normalmente não seriam acessíveis.

REFERÊNCIAS

AARON. **Understanding VLAN hopping attacks**. Nlogic, 2013. Disponível em: <https://www.nlogic.co/understanding-vlan-hopping-attacks/>. Acesso em: 20 abr. 2016.

ALÉCIO, Willian dos Santos; PEREIRA, Júlio César. **Implantação de firewall:** segurança em redes de computadores. Paranavaí: Universidade Paranaense (Unipar), 2014. Disponível em: <http://docplayer.com.br/storage/20/526563/1457828157/BerqPkvwQUGA_BJaOEGQDQ/526563.pdf>. Acesso em: 13 mar. 2016.

AL-SHAER, E. S.; HAMED, H. H. **Modeling and management of firewall policies**. IEEE Transactions on Network and Service Management, v. 1, n. 1, p. 2-10, 2004. Disponível em: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4623689&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D4623689>. Acesso em: 13 mar. 2016.

AL-SHAER, E.; HAMED, H.; BOUTABA, R.; HASAN, M. **Conflict classification and analysis of distributed firewall policies**. IEEE Journal on Selected Areas in Communications, Chicago, Estados Unidos, v. 23, n. 10, p. 2069-2084, 2005. Disponível em: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1514536&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D1514536>. Acesso em: 13 mar. 2016.

AMARAL, Bruno Marques; MAESTRELLI, Marita. **Segurança em redes wireless 802.11.** Rio de Janeiro: Centro Brasileiro de Pesquisas Físicas (CBPF), 2004. 38 p. Disponível em: <http://cbpfindex.cbpf.br/publication_pdfs/nt00204.2006_01_30_22_51_07.pdf>. Acesso em: 13 mar. 2016.

ANDRADE, Lidiane Parente; SOARES, Daniel Nelo; COUTINHO, Mauro Margalho; ABELÉM, Antônio Jorge Gomes. **Análise das vulnerabilidades de segurança existentes nas redes locais sem fio:** um estudo de caso do projeto WLACA. Pará/Amazônia: SERPRO/UFPA/UNAMA, 2004. 8 p. Disponível em: <https://www.researchgate.net/publication/228757441_ANALISE_DAS_VULNERABILIDADES_DE_SEGURANCA_EXISTENTES_NAS_REDES_LOCAIS_SEM_FIO_UM_ESTUDO_DE_CASO_DO_PROJETO_WLACA>. Acesso em: 13 mar. 2016.

BANKS, Ethan. **Five things to know about DHCP snooping**. Packet Pushers, 2012. Disponível em: <http://packetpushers.net/five-things-to-know-about-dhcp-snooping/>. Acesso em: 20 abr. 2016.

BAQUI, Ruben Bambi Tsimba. **Segurança em redes linux com firewall**. Curitiba: Universidade Tecnológica Federal do Paraná, 2012. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/1832/1/CT_GESER_II_2012_10.PDF>. Acesso em: 13 mar. 2016.

BEHRINGER, Michael. **Understanding operational security**. San Jose, Estados Unidos: Cisco Security, 2016. Disponível em: <http://www.cisco.com/c/en/us/about/security-center/understanding-operational-security.html>. Acesso em: 17 abr. 2016.

BHAIJI, Yusuf. **Security features on switches**. Cisco Press, 2008. Disponível em: <http://www.ciscopress.com/articles/article.asp?p=1181682>. Acesso em: 20 abr. 2016.

BOF, Edson. **Segurança em redes wireless**. Serra: Faculdade do Centro Leste (UCL), 2010. 58 p. Disponível em: <http://br.monografias.com/trabalhos-pdf/seguranca-redes-wireless/seguranca-redes-wireless.pdf>. Acesso em: 13 mar. 2016.

BRITO, Samuel Henrique Bucke. **DHCP snooping na mitigação de servidores falsos**. 2013. Disponível em: <http://labcisco.blogspot.com.br/2013/01/dhcp-snooping-na-mitigacao-de.html>. Acesso em: 20 abr. 2016.

CAIS (Centro de Atendimento a Incidentes de Segurança). **Estatísticas**. Brasília/Campinas/Rio de Janeiro: RNP (Rede Nacional de Ensino e Pesquisa), 2016. Disponível em: <https://www.rnp.br/servicos/seguranca/tratamento-incidentes/estatisticas>. Acesso em: 20 mar. 2016.

CAMPBELL, Patrick T. **Instalando redes em pequenas e médias empresas.** São Paulo: Makron Books, 1997. 343 p.

CERT.br (Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil). **Estatísticas dos incidentes reportados ao CERT.br**. 2016. Disponível em: <http://www.cert.br/stats/incidentes/>. Acesso em: 20 mar. 2016.

CERT.br (Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil). **CERT.br:** incidentes reportados (tipos de ataque acumulado). 2016. Disponível em: <http://www.cert.br/stats/incidentes/2015-jan-dec/tipos-ataque-acumulado.html>. Acesso em: 20 mar. 2016.

CERT.br (Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil). **Incidentes reportados ao CERT.br:** janeiro a dezembro de 2015. 2016. Disponível em: <http://www.cert.br/stats/incidentes/2015-jan-dec/tipos-ataque.html>. Acesso em: 20 mar. 2016.

CERTPREPARE. **VLAN hopping questions**. 2014. Disponível em: <http://www.certprepare.com/vlan-hopping-questions>. Acesso em: 28 abr. 2016.

CETIC.br (Centro de Estudos sobre as Tecnologias da Informação e da Comunicação). **TIC empresas 2006:** D5 - problemas de segurança encontrados. 2006. Disponível em: <http://www.cetic.br/tics/empresas/2006/geral/D5/>. Acesso em: 20 mar. 2016.

CETIC.br (Centro de Estudos sobre as Tecnologias da Informação e da Comunicação). **TIC empresas 2007:** D8 - problemas de segurança identificados. 2007. Disponível em: <http://www.cetic.br/tics/empresas/2007/geral/D8/>. Acesso em: 20 mar. 2016.

CETIC.br (Centro de Estudos sobre as Tecnologias da Informação e da Comunicação). **TIC empresas 2008:** D8 - problemas de segurança identificados. 2008. Disponível em: <http://www.cetic.br/tics/empresas/2008/geral/D8/>. Acesso em: 20 mar. 2016.

CETIC.br (Centro de Estudos sobre as Tecnologias da Informação e da Comunicação). **TIC empresas 2009:** D5 - problemas de segurança identificados. 2009. Disponível em: <http://www.cetic.br/tics/empresas/2009/geral/D5/>. Acesso em: 20 mar. 2016.

CETIC.br (Centro de Estudos sobre as Tecnologias da Informação e da Comunicação). **TIC empresas 2010:** D5 - problemas de segurança identificados. 2010. Disponível em: <http://www.cetic.br/tics/empresas/2010/geral/D5/>. Acesso em: 20 mar. 2016.

CHEMIN, Beatris F. **Manual da UNIVATES para trabalhos acadêmicos:** planejamento, elaboração e apresentação. 3. ed. Lajeado: Editora UNIVATES, 2015.

CHOWDHURY, Dhiman D. **Projetos avançados de redes IP:** roteamento, qualidade de serviço e voz sobre IP. Rio de Janeiro: Campus, 2002. 380 p.

CISCO. **Catalyst 6500 release 12.2SX software configuration guide**. 2013. Disponível em: <http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/12-2SX/configuration/guide/book/ipsrcgrd.html>. Acesso em: 20 abr. 2016.

CISCO. **Spanning tree protocol root guard enhancement**. 2005. Disponível em: <http://www.cisco.com/c/en/us/support/docs/lan-switching/spanning-tree-protocol/10588-74.html>. Acesso em: 20 abr. 2016.

COBBAUT, Paul. **Chapter 1:** general networking. 2015. Disponível em: <http://linux-training.be/networking/ch01.html>. Acesso em: 15 abr. 2016.

COMER, Douglas E. **Interligação em rede com TCP/IP**. Rio de Janeiro: Campus, 1998. 672 p.

COMER, Douglas E. **Redes de computadores e internet**. 4. ed. Porto Alegre: Bookman, 2007. 632 p.

CORRÊA, Gabriel de Figueiredo. **Tipos de ataques por camada**: camada de enlace. Vitória, 2009. Disponível em: <http://gabritech.blogspot.com.br/2009/10/tipos-de-ataques-por-camada-camada-de.html>. Acesso em: 20 abr. 2016.

DESLAURIERS, Jean-Pierre. **Pesquisa qualitativa:** enfoques epistemológicos e metodológicos. São Paulo: Vozes, 1991. Disponível em:

<http://brainwork.com.br/2012/01/26/switching-camada-2tabelas-cam-e-tcam/>. Acesso em: 20 abr. 2016.

DUARTE, Luiz Otávio. **Análise de vulnerabilidades e ataques inerentes a redes sem fio 802.11x**. São José do Rio Preto: UNESP, 2003. 55 p. Disponível em: <https://projetos.inf.ufsc.br/arquivos_projetos/projeto_193/Resumo%20-%20TCC%20An%E1lise%20de%20Vulnerabilidades%20e%20Ataques%20Inerentes%20a%20Redes%20Sem%20Fio%20802.doc>. Acesso em: 13 mar. 2016.

EXTREME NETWORKS. **ExtremeXOS Concepts Guide:** Software Version 12.5.2. California, Estados Unidos: Extreme Networks, 2011. Disponível em: <http://www.extremenetworks.com/wp-content/uploads/2014/02/EXOSConcepts12_5_2.pdf>. Acesso em: 15 abr. 2017.

FERNANDES, Anita Maria da Rocha; ZANONA, Arthur Felipe. **Inteligência Computacional Aplicada às Redes:** Ferramenta para Determinação de Regras de Firewall a Partir de Políticas de Segurança Lógica Utilizando RBC e Ontologias. São José: Universidade do Vale do Itajaí (UNIVALI), 2010. Disponível em: <http://periodicos.unesc.net/sulcomp/article/download/366/373>. Acesso em: 13 mar. 2016.

FONSECA, J. J. S. **Metodologia da pesquisa científica**. Fortaleza: UEC, 2002.

FOROUZAN, Behrouz A. **Comunicação de dados e redes de computadores**. 4. ed. Porto Alegre: AMGH, 2010. 1134 p.

GARCES, Solange B. B. **Classificação e Tipos de Pesquisas**. Universidade de Cruz Alta – Unicruz, 2010.

GHEORGHE, Lucian. **Designing and implementing linux firewalls and qos using netfilter, iproute2, NAT, and L7-filter**. 1. ed. Birmingham, Inglaterra: Packt Publishing, 2006. Disponível em: <https://www.packtpub.com/sites/default/files/SampleChapter-Designing-and-Implementing-Linux-Firewall-and-QOS.pdf>. Acesso em: 20 abr. 2016.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2007.

GIL, Antonio de Loureiro. **Auditoria de computadores**. 5. ed. São Paulo: Atlas, 2000. 236 p.

GIMENES, Eder Coral. **Segurança de redes wireless**. Mauá: Faculdade de Tecnologia de Mauá, 2005. 58 p. Disponível em: <http://www.tvprudente.com.br/apostilas/Rede/Redes.pdf>. Acesso em: 13 mar. 2016.

GOLDENBERG, M. **A arte de pesquisar**. Rio de Janeiro: Record, 1997.

GOMEDE, Everton. **O temido "broadcast storm"**. 2012. Disponível em: <http://evertongomede.blogspot.com.br/2012/12/o-temido-broadcast-storm.html>. Acesso em: 28 abr. 2016.

HUNT, Ray. **Internet/Intranet firewall security:** policy, architecture and transaction services. Computer Communications, Christchurch, Nova Zelândia, v. 21, n. 13, p. 1107-1123, 1998. Disponível em:

<http://www.sciencedirect.com/science/article/pii/S014036649800173X>. Acesso em: 13 mar. 2016.

IOANNIDIS, Sotiris; KEROMYTIS, Angelos D. BELLOVIN, Steve M.; SMITH, Jonathan M. **Implementing a distributed firewall**. Proceedings of the 7th ACM conference on Computer and communications security, New York, Estados Unidos, p. 190-199, 2000. Disponível em: <http://dl.acm.org/citation.cfm?id=353052>. Acesso em: 13 mar. 2016.

JÚNIOR, José Helvécio Teixeira; SUAVÉ, Jacques Philippe; MOURA, José Antão Beltrão; TEIXEIRA, Suzana de Queiroz Ramos. **Redes de computadores:** serviços, administração e segurança. São Paulo: Makron Books, 1999. 493 p.

KAUFMAN, Charlie; PERLMAN, Radia; SPECINER, Mike. **Network security:** private communication in a public world. 2. ed. New Jersey, Estados Unidos: Prentice Hall PTR, 2002. 713 p.

KOZIEROK, Charles M. **TCP/IP architecture and the TCP/IP model**. The TCP/IP guide, 2005. Disponível em: <http://www.tcpipguide.com/free/t_TCPIPArchitectureandtheTCPIPModel-2.htm>. Acesso em: 20 abr. 2016.

KUROSE, James F.; ROSS, Keith W. **Redes de computadores e Internet:** uma abordagem top-down. 5. ed. São Paulo: Addison Wesley, 2010. 614 p.

LEOPARDI, Maria Tereza. **Metodologia da pesquisa na saúde**. 2. ed. Florianópolis: Pallotti, 2002.

LYU, M. R.; LAU, L. K. Y. **Firewall security:** policies, testing and performance evaluation. Computer Software and Applications Conference, Taipei, China, p. 116-121, 2000. Disponível em: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=884700&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D884700>. Acesso em: 13 mar. 2016.

MANGUEIRA, Thais Pinto. **Transbordamento da tabela CAM ou em inglês CAM table overflow por meio da técnica de ARP poisoning, ARP spoofing, MAC flooding**. 2015. Disponível em: <http://docplayer.com.br/6180536-Tema-transbordamento-da-tabela-cam-ou-em-ingles-cam-table-overflow-por-meio-da-tecnica-de-arp-poisoning-arp-spoofing-mac-flooding.html>. Acesso em: 20 abr. 2016.

NAKAMURA, Emilio Tissato; GEUS, Paulo Lício de. **Segurança de redes em ambientes cooperativos**. São Paulo: Novatec Editora, 2007. 483 p.

NETO, Roberto Miyano. **A evolução dos mecanismos de segurança para redes sem fio 802.11**. Rio de Janeiro: PUC-Rio, 2004. 27 p. Disponível em: <http://www-di.inf.puc-rio.br/~endler/courses/Mobile/Monografias/04/Miyano-Mono.pdf>. Acesso em: 13 mar. 2016.

NIC.br (Núcleo de Informação e Coordenação do Ponto BR). **CERT.br aponta aumento de notificações de ataques a servidores web**. 2016. Disponível em:

<http://www.nic.br/noticia/releases/cert-br-aponta-aumento-de-notificacoes-de-ataques-a-servidores-web/>. Acesso em: 20 mar. 2016.

OMAR, Leila Aquima Agy; PINTO, Celso Mahomed; SAIDE, Nacir Amir. **Criptografia e segurança de dados**. Maputo, Moçambique: Universidade Eduardo Mondlane, 2013. Disponível em: <https://googlegroups.com/group/enginformaticadiurno2010/attach/49527ab9f5faaf14/Criptografia_ModeloOSI%20(DOPS).pdf?part=0.1>. Acesso em: 20 abr. 2016.

OMNISECU. **What is double tagging attack and how to prevent double tagging attack**. 2016. Disponível em: <http://www.omnisecu.com/ccna-security/what-is-double-tagging-attack-how-to-prevent-double-tagging-attack.php>. Acesso em: 28 abr. 2016.

ORTEGA, André. **Switching camada 2:** tabelas CAM e TCAM. Brainwork, 2012.

PERES, André; WEBER, Raul Fernando. **Considerações sobre segurança em redes sem fio**. Porto Alegre: ULBRA/UFRGS, 2003. 8 p. Disponível em: <http://ceseg.inf.ufpr.br/anais/2003/07.pdf>. Acesso em: 13 mar. 2016.

RABELO, Hugo. **CCNP switch:** guia de estudo para profissionais. 1. ed. Belo Horizonte: Fontan, 2014.

RAMOS, Anderson. **Security officer - 1:** guia oficial para formação de gestores em segurança da informação. 1. ed. Porto Alegre: Zouk, 2006. 460 p.

RIBEIRO, Luiz Cláudio. **Uma abordagem de segurança na camada 2**. Uberlândia: Faculdade de ciências aplicadas de minas (UNIMINAS), 2006. 117 p. Disponível em: <http://www.si.lopesgazzani.com.br/TFC/monografias/Uma%20abordagem%20de%20seguranca%20na%20camada%202.pdf>. Acesso em: 13 mar. 2016.

RIGO, Julian Mayer; OLIVEIRA, Welington Paulino. **Redes de computadores I:** introdução. São José dos Campos: TELECO, 2010. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialitil/pagina_1.asp>. Acesso em: 17 abr. 2016.

SAMPAIO, Alexandre Batista. **Implantação de políticas de segurança em redes de computadores com PIX firewall**. Curitiba: Universidade Tecnológica Federal do Paraná, 2011. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/394/3/CT_GESER_1_2011_02.pdf>. Acesso em: 13 mar. 2016.

SCHWEITZER, Christiane M.; SAKURAGUI, Rony R.; CARVALHO, Tereza Cristina; VENTURINI, Yeda Regina. **Tecnologias de redes sem fio:** WPANs, WLANs e WMANs desafios de segurança, vulnerabilidades e soluções. São Paulo/Karlstad: Universidade de São Paulo/Karlstad University Computer Science Department, 2005. 36 p. Disponível em: <ftp://www.linorg.cirp.usp.br/pub1/SSI/SSI2005/Microcursos/MC04.pdf>. Acesso em: 13 mar. 2016.

SILVA, Gilson Marques; SOUZA, João Nunes. **Uma análise dos mecanismos de segurança de redes locais sem fio e uma proposta de melhoria**. Uberlândia: Universidade Federal de Uberlândia (UFU), 2003. 8 p. Disponível em:

<http://www.gilsonmarques.com.br/artigos/seguranca_em_wlan_sbrc_wseg.PDF>. Acesso em: 13 mar. 2016.

SOARES, Luiz Fernandes Gomes; LEMOS, Guido; COLCHER, Sérgio. **Redes de computadores:** das LANS, MANs e WANs às redes ATM. Rio de Janeiro: Campus, 1995. 705 p.

SOWELL, Greg. **Layer 2 security:** protect you and your users. Bryan, Estados Unidos, 2009. Disponível em: <http://gregsowell.com/?p=1133>. Acesso em: 20 abr. 2016.

STALLINGS, William. **Criptografia e segurança em redes:** princípios e práticas. 4. ed. São Paulo: Pearson Prentice Hall, 2008. 492 p.

TANENBAUM, Andrew S.; WETHERALL, David. **Redes de computadores**. 5. ed. São Paulo: Pearson Prentice Hall, 2011. 582 p.

TOLEDO, Ismerino Roriz S. C. **Hardware completo:** dicas inéditas. 1. ed. Goiânia: Terra, 2005. 440 p.

TRIVIÑOS, A. N. S. **Introdução à pesquisa em ciências sociais:** a pesquisa qualitativa em educação. São Paulo: Atlas, 1987. 175 p.

VERISSIMO, Fernando. **Segurança em redes sem fio**. Rio de Janeiro: Universidade Federal do Rio de Janeiro (UFRJ), 2002. 90 p. Disponível em: <http://www.land.ufrj.br/~verissimo/cos871/bibref/wnsmono.pdf>. Acesso em: 13 mar. 2016.

WATKINS, Michael e WALLACE, Kevin. **CCNA security:** official exam certification guide. Indianapolis, Estados Unidos: Cisco Press, 2008. 637 p. Disponível em: <http://www.sc.mahidol.ac.th/scsosd/Doc/km/Network/CCNA/CCNASecurityOfficialExamCertificationGuide.pdf>. Acesso em: 28 abr. 2016.

WOOL, A. **A quantitative study of firewall configuration errors**. Computer, Tel Aviv, Israel, v. 37, n. 6, p. 62-67, 2004.

YIN, Ralph. **Case study research in applied social research methods series**. Sage Publitions Newbury Park. London, Inglaterra: New Delli, 1989. v. 5.