Implementação de um Mecanismo Para Detecção de Mensagens *Router Advertisement* Maliciosas e Servidores DHCPv6 Falsos em Redes de Equipamentos Legados

Rafael G. P. Azambuja[1], Tiago A. Rizzetti[2]

[1]Centro de Tecnologia – Universidade Federal de Santa Maria (UFSM)
Av. Roraima, 1000 – 97.105-900 – Santa Maria – RS – Brazil

[2]Colégio Técnico Industrial de Santa Maria – Universidade Federal de Santa Maria (UFSM)
Av. Roraima, 1000 – 97.105-900 – Santa Maria – RS – Brazil

rafaelazambuja@redes.ufsm.br, rizzetti@ctism.ufsm.br

Abstract. *This work presents the implementation of a mechanism for detecting and mitigating threats in legacy IPv6 networks, focusing on DHCPv6 servers and rogue routers. Using a test network infrastructure, the developed tool identified rogue devices and disabled untrusted ports on switches, providing detailed logs. Although it was effective in the test environment, future work aims to integrate this solution with monitoring systems, automate configurations and improve scalability in complex environments with different switch models.*

Resumo. *Neste trabalho é apresentado a implementação de um mecanismo para detecção e mitigação de ameaças em redes IPv6 legadas, focando em servidores DHCPv6 e roteadores invasores. Utilizando uma infraestrutura de rede de teste, a ferramenta desenvolvida identificou dispositivos invasores e desabilitou portas não confiáveis em switches, fornecendo registros detalhados. Embora tenha sido eficaz no ambiente de teste, futuros trabalhos visam integrar essa solução a sistemas de monitoramento, automatizar configurações e melhorar a escalabilidade em ambientes complexos com diversos modelos de switches.*

1. Introdução

Nos últimos anos, a migração para o *Internet Protocol version 6* (IPv6) tornou-se cada vez mais crucial à medida que o espaço de endereçamento *Internet Protocol version 4* (IPv4) tem-se esgotado [NRO 2011]. No entanto, essa transição apresenta desafios significativos, especialmente em redes que dependem de dispositivos legados desenvolvidos antes da concepção de técnicas e ferramentas referentes à segurança em redes IPv6. Esses dispositivos muitas vezes carecem dos recursos necessários para lidar com as ameaças de segurança associadas ao IPv6. Questões relacionadas à preocupações quanto a segurança de configuração e atribuições de endereços IPv6 são discutidas em diversas RFCs, principalmente as RFCs 6104, 6105, 7113 e 7610.

A RFC 6104 apresenta um problema observado em redes *Dual Stack* ou IPv6: a possibilidade da presença de mensagens *Router Advertisement* (RA), descrevendo três possíveis cenários. O primeiro cenário envolve a configuração equivocada de um equipamento, resultando no envio de mensagens RA com informações incorretas. O segundo cenário envolve um cliente acidentalmente enviando mensagens RA. Isto pode ocorrer, por exemplo, conectando uma interface errada de um roteador na rede. O

terceiro cenário descreve um atacante enviando intencionalmente mensagens RA, como tentativa de um ataque *man-in-the-middle* ou *Denial of Service* [Chown and Venaas 2011]. O documento propõe alguns métodos para mitigação de mensagens RA maliciosas, como o uso de *Access Control List* (ACL), e o bloqueio destas mensagens por dispositivos de camada 2, como proposto pela RFC 6105.

A solução proposta pela RFC 6105 é baseada em redes onde os nós estão interligados por dispositivos de camada 2. O mecanismo sugerido, *RA-Guard*, é implementado em dispositivos de camada 2 para a filtragem de mensagens RA, avaliando critérios como portas confiáveis e endereços de camada 2 e 3 [Levy-Abegnoli et al. 2011]. Notas sobre possíveis falhas em implementações do *RA-Guard* são descritas na RFC 7113, como, por exemplo, a falha do dispositivo em analisar mensagens fragmentadas e mensagens RA que contenham cabeçalhos de extensão [Gont 2014].

Enquanto os documentos descritos anteriormente abordam questões referentes a mensagens RA, clientes utilizando os métodos de configuração de endereço IPv6 propostos pelas RFCs 3315 e 3736 estão vulneráveis a servidores *Dynamic Host Configuration Protocol version 6* (DHCPv6) invasores. Uma solução proposta para este problema, análoga ao *DHCP Snooping*, é descrita na RFC 7610. O *DHCPv6-Shield* é implementado em um dispositivo de camada 2, de tal maneira que este realize a filtragem de mensagens DHCPv6 com base em portas confiáveis [Gont et al. 2015].

Diversos dispositivos legados não dispõem dos mecanismos *RA-Guard* e *DHCPv6-Shield*. Para alguns destes dispositivos é possível utilizar ACLs para IPv6, como mencionado na RFC 6104 para filtragem de mensagens RA e DHCPv6, avaliando critérios como tipo de mensagem *Internet Control Message Protocol version 6* (ICMPv6), endereço de camada 2 e camada 3 de origem, e porta UDP/TCP de origem ou destino. Porém, uma grande parcela destes dispositivos não possibilitam a configuração destes critérios de avaliação para ACLs.

O presente artigo apresenta a implementação de um mecanismo de detecção e mitigação dessas ameaças. Esse mecanismo é projetado para operar com um mínimo de recursos e dependências externas, tornando-o adequado para redes de equipamentos legados.

1.1. Objetivos

Para a implementação do mecanismo proposto neste trabalho, estabeleceu-se os seguintes objetivos. Primeiramente, criar um cenário de estudo e teste, identificando dispositivos de interconexão e agrupando-os com base em recursos como suporte ao *Secure Shell* (SSH) e configuração de ACLs IPv6, bem como características comuns, como *Management Information Bases* (MIB) para tabelas MAC, parâmetros para requisições *Simple Network Management Protocol* (SNMP) e comandos de *Command Line Interface* (CLI). Em seguida, mapear a rede do cenário, identificando subredes, *Virtual Local Area Networks* (VLAN) e portas confiáveis, armazenando essas informações em uma base de dados.

O mecanismo proposto deve ser receptivo a mensagens RA e DHCPv6 de todas as VLANs e subredes do cenário proposto, analisando o endereço MAC de origem. Isso inclui o desenvolvimento de um método para consultar a tabela MAC dos dispositivos de interconexão, identificar portas com o MAC de dispositivos invasores e compará-las

com as portas confiáveis. Quando encontrada a porta de conexão final entre o dispositivo de interconexão e o invasor, o mecanismo deve desabilitar esta porta e registrar essa ação em um servidor de *logs* remoto para documentação e notificação aos administradores da rede.

2. Trabalhos Relacionados

Embora não existam muitos trabalhos que lidam especificamente com o problema de servidores DHCPv6 e roteadores invasores em redes de equipamentos legados, nesta seção serão apresentados trabalhos que abordam soluções com base no bloqueio dinâmico de tráfego malicioso.

Em [Naveed et al. 2010] é proposta a integração da ferramenta Snort com roteadores Cisco para a geração automatizada de ACLs com base em mensagens de alerta geradas pelo Snort. Em [Fu et al. 2017] é implementado um mecanismo para *firewalls* que visa bloquear ataques de negação de serviço e reduzir a carga do *firewall*. Utilizando *logs* syslog gerados pelo *firewall* como referência, são executados comandos em um roteador para geração de ACLs com intuito de bloquear endereços de rede contidos em ataques maliciosos.

No trabalho de [Bakker et al. 2016], os autores utilizam o conceito de *Software-defined network* (SDN) para tornar uma infraestrutura de rede em um *firewall* virtual. Com uso do protocolo *OpenFlow* em redes SDN, os autores estabelecem um *firewall* baseado em ACLs que utiliza tabelas de tráfego *blacklisted* e *whitelisted*.

Para o problema apresentado na Seção 1, estas soluções se tornam complexas demais. Como será discutido na Seção 3, o cenário de testes não permite que soluções baseadas em ACLs sejam possíveis. Soluções baseadas em SDN requerem uma reestruturação da rede. Isto diverge da proposta deste trabalho. Com base na Seção anterior, e como será discutido na Seção 5, a solução proposta neste trabalho visa o menor número de configurações e ferramentas, e espera que seja possível a integração com outras ferramentas.

3. Cenário de Testes

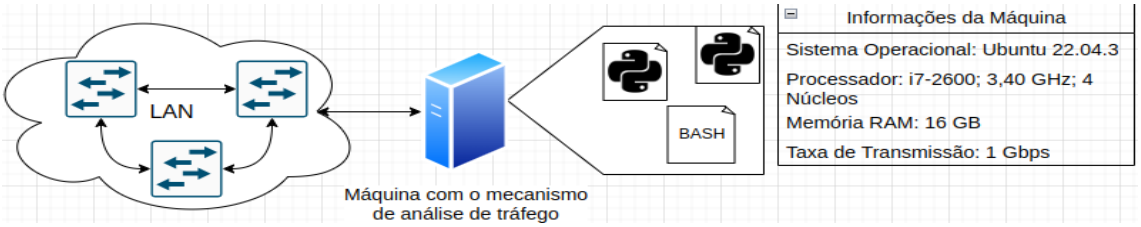

Figura 1. Cenário de Testes

Para execução dos objetivos estabelecidos na Seção 1.1., foi utilizada a infraestrutura de rede do Colégio Técnico Industrial de Santa Maria (CTISM). É importante ressaltar que o CTISM ainda não implementou, em toda sua rede, suporte ao IPv6. Portanto, assume-se que não existem servidores DHCPv6 e roteadores IPv6 legítimos na rede geral. Embora a rede do CTISM seja composta por diversas VLANs e subredes, para simplicidade da execução dos testes, é considerada apenas uma subrede e uma VLAN. O cenário de testes e as informações da máquina que contém o mecanismo proposto são

representadas na Figura 1.

O mapeamento da rede foi realizado através de requisições SNMP para os switches, agrupando-os em relação ao fabricante, o modelo, e às MIBs implementadas para consultas de tabela MAC. Para a obtenção das porta confiáveis, foram realizadas consultas às tabelas *Link Layer Discovery Protocol* (LLDP) e MAC. Estas portas foram armazenadas em uma base de dados que as relaciona com o endereço de rede do dispositivo.

A Tabela 1 apresenta uma relação dos switches da rede do CTISM quanto aos recursos de interesse para este trabalho, como a possibilidade de interagir com uma CLI para gerenciamento, parâmetros relacionados a mensagens SNMP, recursos RA-Guard e DHCPv6-Shield, e a possibilidade de configurar ACLs IPv6 avançadas, ou seja, ACLs que tenham critérios de avaliação como tipo ICMPv6.

Tabela 1. Relação de características dos switches

Fabricante	Modelo	Quantidade	CLI/SSH	SNMP	RA-Guard	DHCPv6-Shield	ACL IPv6 Avançada
Fabricante 1	Modelo A	14	Sim	v3	Não	Não	Não
	Modelo B	1	Não	v2c	Não	Não	Não
	Modelo C	1	Não	v2c	Não	Não	Não
	Modelo D	1	Sim	v3	Não	Não	Não
Fabricante 2	Modelo E	9	Sim	v3	Não	Não	Não
	Modelo F	3	Sim	v3	Não	Não	Não
	Modelo G	6	Sim	v3	Não	Não	Sim
	Modelo H	9	Sim	v3	Não	Não	Sim
Fabricante 3	Modelo I	5	Não	v3	Não	Não	Não
Fabricante 4	Modelo J	4	Sim	v3	Não	Não	Sim
Fabricante 5	Modelo K	1	Não	v3	Não	Não	Não
Fabricante 6	Modelo L	2	Sim	v3	Não	Não	Sim
Fabricante 7	Modelo M	5	Sim	v3	Não	Não	Não
	Modelo N	1	Sim	v3	Não	Não	Não
Total de switches: 62							

A principal coluna de interesse na Tabela 1 é a coluna “ACL IPv6 Avançada”. Se todos os switches permitissem a criação destas ACLs, o mecanismo proposto neste trabalho seria desnecessário neste cenário. Porém, dos 62 switches presentes na infraestrutura de rede, 41 não possuem este recurso, sendo estes pontos de vulnerabilidade quanto a mensagens RA e DHCPv6 maliciosas. A próxima coluna de interesse é a coluna “SNMP”. Como será mencionado no decorrer desta Seção, o protocolo SNMP será utilizado para operações de leitura e escrita. Na rede do CTISM operações de escrita e consulta às MIBs proprietárias são permitidas apenas por SNMPv3. Como os switches dos modelos B e C não possuem suporte ao protocolo SNMPv3, estes foram descartados do cenário de testes. No grupo dos switches vulneráveis ainda foi necessário agrupá-los quantos às MIBs implementadas para consultas nas tabelas MAC. A Tabela 2 apresenta esta relação.

Tabela 2. Relação de MIBs

Modelo	MIB
Modelo A	MIB do Fabricante 1
Modelo D	BRIDGE-MIB
Modelo E	MIB do Fabricante 2
Modelo F	MIB do Fabricante 2
Modelo I	MIB do Fabricante 3
Modelo K	BRIDGE-MIB
Modelo M	BRIDGE-MIB
Modelo N	BRIDGE-MIB

3.1. Modelo Proposto

O mecanismo proposto consiste em um nó na rede que possui interfaces de rede disponíveis nas VLANS e subredes de interesse. No cenário de teste, esse nó pertence a apenas uma VLAN e uma subrede. A análise de tráfego é feita por um *script* escrito em Python que utiliza o módulo *scapy*. O *script* Python procura nos pacotes recebidos por mensagens DHCPv6 e RA, analisando o endereço MAC de origem. Quando identificadas, é executado um *script* Bash que procura o MAC do dispositivo invasor nas tabelas MAC dos dispositivos de camada 2, através do protocolo SNMP. A escolha deste protocolo se dá pelo fato da diferença de comandos utilizados para gerenciamento por CLI implementados por cada fabricante. Quando encontrada uma porta que contenha o MAC, é procurada na base de dados de portas confiáveis se esta é uma porta segura. Caso não seja, então é enviado uma mensagem *snmpset* para desabilitar esta porta, e então é enviado, através de um segundo *script* Python, uma mensagem *syslog* para o servidor de *logs* remoto, informando a porta desabilitada e o dispositivo que contém a porta.

Considerando o fato da infraestrutura do CTISM, na rede geral, não conter roteadores IPv6 e servidores DHCPv6 legítimos, todos os pacotes RA e DHCPv6 são tratados como maliciosos. Os testes realizados envolvem dispositivos conectados a diferentes switches, enviando mensagens RA e DHCPv6 ilegítimas na rede.

4. Resultados

Através da ferramenta implementada, foi possível atingir os objetivos estabelecidos. Para todos os modelos de interesse (Seção 3, Tabela 1), foram encontradas as portas nas quais o dispositivo invasor foi conectado. Para os testes realizados, foram necessários menos de cinco minutos para encontrar a porta que continha o MAC do dispositivo atacante. O tempo necessário depende apenas da quantidade de dispositivos para consulta, e do tempo de resposta destes.

A Figura 2 apresenta o um *log* enviado pela ferramenta ao servidor de *logs*. No *log* é informado o tipo de ataque detectado, o switch conectado diretamente ao dispositivo invasor, a porta que foi desativada, e o endereço MAC do atacante.

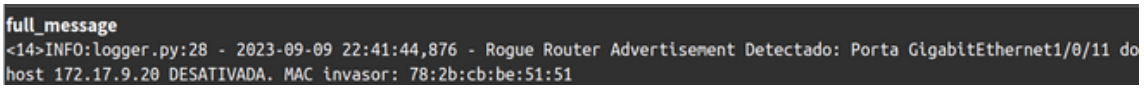

Figura 2. Log armazenado no servidor de logs

5. Conclusão e Trabalhos Futuros

No contexto da infraestrutura do CTISM, a implementação desta ferramenta foi realizada de forma simples, necessitando apenas de um interpretador Python, uma base de dados contendo as portas confiáveis, e três *scripts*. Implementações em redes com um maior número de dispositivos e um nível menor de padronização pode resultar em uma solução não escalável. A lista de portas confiáveis é composta de pontos de conexão entre switches de camada 3 e com servidores de produção. É importante notar que se entre o atacante e o dispositivo de camada 3 existirem um ou mais dispositivos não gerenciáveis, a desativação da porta implica no bloqueio de todo um segmento de rede.

Como trabalho futuro, pretende-se integrar este mecanismo a uma solução de monitoramento como o Zabbix. No decorrer deste trabalho, foi possível notar a complexidade de gerenciamento de uma rede com diversos modelos de switches de diferentes fabricantes. Um módulo para o Zabbix para gerenciamento de switches, onde fosse possível obter-se uma lista de portas confiáveis e a lista de dispositivos acabaria com a necessidade da base de dados utilizada neste trabalho. Ainda, a ferramenta não possui nenhum mecanismo para habilitar uma porta que por ela foi desabilitada. Através deste módulo, poderiam ser configuradas rotinas para verificação das portas desabilitadas e reabilitá-las. Por fim, diversas configurações, como parâmetros para mensagens SNMP, endereço do servidor de *log*, e endereços de dispositivos confiáveis são feitas de forma manual. O uso de um módulo Zabbix permitiria que isto ocorresse de forma automatizada.

Referências

Chown, T., Venaas, S. (2011) “Rogue IPv6 Router Advertisement Problem Statement”, https://datatracker.ietf.org/doc/html/rfc6104, Setembro.

Fu, S., Hsu, H., Kao, Y., Tsai, S. Tseng, C. (2017) “An autoblocking mechanism for firewall service”. In: IEEE. 2017 IEEE Conference on Dependable and Secure Computing. [S.l.], 2017. p. 531–532.

Gont, F. (2014) “Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)”, https://www.rfc-editor.org/rfc/rfc7113, Setembro

Gont, F., Liu, W. and Van de Velde, G. (2015) “DHCPv6-Shield: Protecting against Rogue DHCPv6 Servers”, https://www.rfc-editor.org/rfc/rfc7610, Setembro.

Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C. and Mohacsi, J. (2011) “IPv6 Router Advertisement Guard”, https://datatracker.ietf.org/doc/html/rfc6105, Setembro.

NRO. (2011) “Free Pool of IPv4 Address Space Depleted”, https://www.nro.net/ipv4-free-pool-depleted, Setembro.

Naveed, M., Nihar, S., Babar, M. (2010) “Network Intrusion Prevention by Configuring ACLs on the Routers, based on Snort IDS alerts”, https://www.researchgate.net/publication/224196023_Network_intrusion_prevention_by_configuring_ACLs_on_the_routers_based_on_Snort_IDS_alerts, Setembro.