Caracterização Remota de Comportamento de Roteadores IPv6

**Rafael Almeida, Elverton Fazzion, Osvaldo Fonseca,
Dorgival Guedes, Wagner Meira, Ítalo Cunha**

[1] Departamento de Ciência da Computação
Universidade Federal de Minas Gerais

{rlca,elverton,osvaldo.morais,dorgival,meira,cunha}@dcc.ufmg.br

Abstract. *Even though the IETF standardizes protocols and provides implementation guidelines, many implementation decisions are left to vendors and developers. The behavior of network devices are also dependent on their configuration. Over the years, researchers have developed many measurement techniques to characterize network devices. These characterization techniques are useful for network management, operation, troubleshooting, and security. With the depletion of IPv4 address space and increasing IPv6 adoption, characterization techniques for IPv6 devices are increasingly important. Unfortunately, characterization techniques for IPv6 devices remain incipient. In this work we propose and evaluate a new characterization technique for IPv6 devices along a network path that takes the behavior of intermediate devices into account.*

Resumo. *Apesar da IETF padronizar protocolos e prover diretivas de implementação, muitas decisões de implementação são deixadas a cargo de fabricantes e desenvolvedores. O comportamento de dispositivos também depende de suas configurações. Ao longo dos anos, pesquisadores desenvolveram várias técnicas de medição para caracterizar dispositivos de rede. Estas técnicas de caracterização de dispositivos são úteis para gerenciamento, operação, resolução de falhas e segurança de redes. Com o esgotamento dos endereços IPv4 e a crescente adoção do protocolo IPv6, técnicas de caracterização de dispositivos na Internet IPv6 são cada vez mais importantes. Porém, técnicas de caracterização para dispositivos IPv6 ainda são incipientes. Neste trabalho propomos e avaliamos uma nova técnica de caracterização de dispositivos IPv6 ao longo de um caminho na Internet que considera características de dispositivos intermediários.*

1. Introdução

A Internet é formada por milhares de redes autônomas interligadas. Essas redes, denominadas sistemas autônomos, podem ter diferentes políticas internas, que impactam decisões como a escolha de enlaces físicos, de dispositivos de interconexão e da configuração destes dispositivos. As políticas internas de um sistema autônomo, em geral, são privadas e consideradas segredos de negócios. No entanto, técnicas de caracterização e *scanning* (e.g. Nmap) podem expor tais políticas. Neste cenário, podemos destacar algumas questões relativas à segurança das redes: (1) A topologia interna de um sistema autônomo pode ser reconstruída através de técnicas de traceroute e IP *aliasing*. Essa

informação pode ser utilizada por um atacante para identificar pontos de falhas ou gargalos na rede. (2) Um atacante pode expor características dos dispositivos utilizados dentro de um sistema autônomo através de técnicas de coleta de assinaturas (*fingerprinting*) e utilizar tais informações para explorar vulnerabilidades conhecidas nos dispositivos identificados. Técnicas de coleta de assinaturas de dispositivos de rede devem ser estudadas, pois tem grande potencial de expor informações consideradas sigilosas pelos sistemas autônomos. O Nmap [Lyon 2009], por exemplo, é uma ferramenta que constrói a assinatura de um dispositivo de rede e através dessa informação infere o sistema operacional em execução no dispositivo.

Grande parte das técnicas de coleta de assinaturas levam em conta os diferentes comportamentos dos dispositivos de rede em diferentes situações, por exemplo, podem considerar as diferentes características dos cabeçalhos da camada de rede e da camada de transporte dos pacotes enviados pelo dispositivo alvo. O IETF (Internet Engineering Task Force) é uma entidade que propõe padrões de tecnologias e protocolos para a Internet, através dos RFCs (Request for Comments). No entanto, os padrões propostos pelo IETF podem deixar algumas decisões a cargo dos fabricantes de dispositivos, desenvolvedores e administradores de redes. Existe, portanto, uma heterogeneidade de comportamento entre os diferentes dispositivos de rede. Essa heterogeneidade é explorada por técnicas de coleta de assinaturas.

A coleta de assinaturas para dispositivos na Internet IPv4 é bastante estudada, no entanto existem poucos trabalhos (e ferramentas) que focam no protocolo IPv6. Segundo a Google, o tráfego de pacotes IPv6 na Internet passou dos 12% em Junho de 2016 [Google 2016]. Dessa forma, estudos que revelam aspectos da segurança da Internet, em particular da Internet IPv6, são de grande importância, dado que nosso entendimento sobre as peculiaridades de segurança da Internet IPv6 ainda é limitado.

Neste trabalho propomos um método para construção de assinaturas de dispositivos de rede para a Internet IPv6 que leva em conta não só informações do dispositivo alvo como também de seus vizinhos. Nossa assinatura é construída através de medições ativas e leva em conta aspectos do protocolo IPv6 que não foram explorados pelos poucos trabalhos existentes sobre o assunto. Consideramos os diferentes comportamentos dos dispositivos de rede em relação aos campos Traffic Class e Flow Label do cabeçalho IPv6. Em nossos experimentos, medimos e classificamos dispositivos de rede em cinco clusters diferentes. Aproveitamos para avaliar a viabilidade da técnica de coleta de assinaturas proposta por Vanaulbel et al. [Vanaubel et al. 2013] que é baseada nos TTLs iniciais de pacotes enviados pelos dispositivos.

Nosso método pode ser utilizado para identificar problemas de vazamento de informações confidenciais dos sistemas autônomos e auxiliar os administradores de redes na mitigação destes problemas. Por exemplo, se um atacante consegue associar um tipo de assinatura a um determinado modelo, tipo ou configuração de dispositivo de rede ele pode utilizar essa informação para explorar alguma vulnerabilidade previamente conhecida do dispositivo. Nosso método pode ser útil para obtenção da topologia de uma rede no nível de roteadores, complementando técnicas custosas de associação de endereços (IP *aliasing*), uma vez que interfaces que possuem o mesmo comportamento podem ser associadas a um mesmo roteador.

2. Fundamentação Teórica

O cabeçalho IPv6 trouxe muitas novidades em relação ao cabeçalho IPv4. Aqui introduzimos duas delas, o campo Flow Label e o campo Traffic Class. Em nosso trabalho estamos interessados no comportamento dos dispositivos de rede em relação a estes campos.

Alguns dispositivos de rede podem levar em conta um identificador de fluxo durante o processamento dos pacotes. Um balanceador de carga, por exemplo, pode utilizar o identificador de fluxo para encaminhar todos os pacotes de um mesmo fluxo para a mesma interface de saída, evitando que estes pacotes cheguem fora de ordem no destino final. O identificador de fluxo é tradicionalmente formado pela quíntupla: endereço de origem, endereço de destino, protocolo, porta de origem e porta de destino. Em um pacote IPv4, as informações da camada de transporte (como porta de origem e de destino) podem ser facilmente alcançadas através do campo Internet Header Length (IHL). No entanto, o campo IHL não existe no cabeçalho IPv6. Se um pacote IPv6 possui cabeçalhos de extensão, é necessário seguir a cadeia de Next Headers para alcançar o cabeçalho da camada de transporte. Isso impede que o tradicional identificador de fluxo seja calculado de forma eficiente. Para contornar este problema, o campo Flow Label foi introduzido no cabeçalho IPv6 [Amante et al. 2011]. O campo Flow Label permite a identificação do fluxo sem que as informações do cabeçalho da camada de transporte sejam necessárias. O campo Flow Label é especificado pela RFC 6437 [Amante et al. 2011]. Segundo a RFC, quando a este campo é atribuído um valor diferente de zero, é esperado que o pacote chegue ao destino final sem que o valor do Flow Label seja alterado. A RFC indica ainda que um dispositivo que encaminha um pacote cujo Flow Label é zero, pode inicializar o campo para um valor diferente de zero.

Além do Flow Label, o protocolo IPv6 também introduziu o campo Traffic Class. O campo Traffic Class é utilizado por dispositivos de rede para identificar diferentes classes de pacotes IPv6. Segundo a RFC 2460 [Deering and Hinden 1998], que especifica o protocolo IPv6, é esperado que o Traffic Class dê suporte aos serviços diferenciados (DiffServ), assim como o campo TOS do cabeçalho IPv4. Dispositivos de rede podem utilizar este campo de diferentes maneiras, inclusive de forma experimental e não padronizada. A RFC em questão determina que dispositivos de rede que suportam o campo Traffic Class podem modificar os valores do campo ao criar, encaminhar ou receber um pacote IPv6, sem restrições. Um dispositivo não deve assumir que o valor do campo Traffic Class é o mesmo de quando o pacote foi criado, ou seja, ele pode ter sido modificado por um dispositivo intermediário.

3. Determinando a Assinatura de um Dispositivo

Para construir a assinatura de um dispositivo alvo de rede IPv6 levamos em conta seu comportamento, dos dispositivos antecessores a ele (que estão entre o ponto de medição e o dispositivo alvo), e também do dispositivo sucessor a ele. O dispositivo sucessor tem papel importante pois utilizamos os pacotes com origem nele para descobrir se o dispositivo alvo faz ou não modificações nos pacotes enviados do ponto de medição. Já os dispositivos antecessores são levados em conta pois, se entre o ponto de medição e o dispositivo alvo há um dispositivo que modifica campos do cabeçalho IPv6, então precisamos saber isso para diferenciar modificações realizadas pelo dispositivo alvo de modificações realizadas por dispositivos intermediários.

Como exemplo vamos considerar a rota de Internet IPv6 $[R_1, R_2, R_3, R_4]$. Essa rota possui 4 dispositivos de rede. Vamos considerar R_3 como o dispositivo alvo. Neste caso, R_1 e R_2 são os dispositivos antecessores e R_4 o dispositivo sucessor ao dispositivo alvo. Para determinar sua assinatura, nossa metodologia leva em conta pacotes com origem em R_1, R_2, R_3 e em R_4. Estamos particularmente interessados em 2 tipos de informações do dispositivo alvo R_3: (i) as modificações que ele faz nos campos Traffic Class e Flow Label ao encaminhar um pacote para R_4 e (ii) os valores que ele utiliza para inicializar os campos Traffic Class e Flow Label quando cria pacotes ICMP Time Exceeded e ICMP Echo Reply.

3.1. Sondas de Medição

Inicialização dos campos Flow Label e Traffic Class. Enviamos sondas com TTL limitado utilizando a técnica do Paris traceroute, mantendo o identificador de fluxo constante. Para cada dispositivo alvo, em ordem crescente de distância a partir do ponto de medição, enviamos duas sondas ICMP Echo Request com TTL limitado. Na primeira sonda mantemos os campos Traffic Class e Flow Label zerados, e na segunda inicializamos estes campos com valores diferentes de zero. Essas duas sondas fazem com que os dispositivos alvo respondam com pacotes ICMP Time Exceeded. Com essas respostas inferimos como o dispositivo alvo inicializa os campos Flow Label e Traffic Class para pacotes Time Exceeded. Para cada dispositivo alvo, enviamos uma terceira sonda ICMP Echo Request destinada ao endereço IP identificado no dispositivo alvo pelas sondas acima. A sonda ICMP Echo Request faz com que o dispositivo alvo responda com um pacote ICMP Echo Reply. Com essa última sonda sabemos como o dispositivo alvo inicializa os campos Flow Label e Traffic Class para pacotes Echo Reply.

Modificação dos campos Flow Label e Traffic Class. Os pacotes ICMP Time Exceeded copiam também o cabeçalho do pacote recebido pelo dispositivo alvo (o pacote cujo TTL expirou e resultou na criação da resposta ICMP Time Exceeded). O cabeçalho copiado permite identificar qual Flow Label e Traffic Class foram observados pelo dispositivo alvo, em particular, se o Flow Label e Traffic Class foram modificados por algum dispositivo antecessor.

Impacto de rotas assimétricas. Rotas na Internet são assimétricas: um pacote de resposta criado pelo dispositivo pode trafegar de volta até o ponto de medição (*reverse path*) por um caminho diferente do trafegado pela sonda (*forward path*). Neste caso não podemos determinar qual dispositivo fez a modificação (ou inicialização) nos campos Flow Label e Traffic Class do pacote de resposta, apenas detectar a inicialização ou modificação dos campos. Note que o cabeçalho encapsulado por pacotes ICMP Time Exceeded não é modificado no caminho de volta até o ponto de medição, de forma que podemos identificar quais dispositivos modificam os campos Flow Label e Traffic Class no caminho do ponto de medição até um destino (*forward path*).

Impacto da perda de pacotes. Nossa metodologia depende fortemente das informações dos pacotes de respostas dos dispositivos considerados para a composição da assinatura. Nossa ferramenta reenvia as sondas de medições (faz até 3 reenvios) para tentar obter uma resposta do dispositivo. Dispositivos que não respondem impedem identificar precisamente qual dispositivo realizou as modificações nos campos do pacote. Neste caso, associamos qualquer modificações observadas ao dispositivo sucessor.

3.2. Assinatura dos dispositivos

Levamos em conta uma assinatura para os dispositivo de rede IPv6 que é formada pela dupla $\langle \text{MOD}, \text{INI} \rangle$. A entrada MOD guarda informações sobre as modificações que o dispositivo alvo faz nos campos Traffic Class e Flow Label ao encaminhar um pacote para o dispositivo sucessor. A entrada INI guarda informações sobre como o dispositivo alvo inicializa (e como dispositivos no caminho de volta até o ponto de medição modificam) os campos Traffic Class e Flow Label nos pacotes ICMP Time Exceeded e Echo Reply que ele envia. Cada entrada da assinatura é uma quádrupla. A tabela 3.2 sumariza a assinatura. A seguir, detalhamos os valores que cada um dos campos pode assumir e como determinamos cada campo.

Campo	Valor	Significado
MOD	$\langle \text{FL}_\text{Z}, \text{TC}_\text{Z}, \text{FL}_\text{N}, \text{TC}_\text{N} \rangle$	Campos que o dispositivo modifica
INI	$\langle \text{FL}_\text{TE}, \text{FL}_\text{ER}, \text{TC}_\text{TE}, \text{TC}_\text{ER} \rangle$	Campos que o dispositivo inicializa

Tabela 1. Campos da assinatura de dispositivos.

O campo MOD da assinatura guarda as modificações que o dispositivo alvo faz ao encaminhar um pacote. Ele é formado pela quádrupla $\langle \text{FL}_\text{Z}, \text{TC}_\text{Z}, \text{FL}_\text{N}, \text{TC}_\text{N} \rangle$. Cada entrada assume valor booleano e indica se o dispositivo modifica os campos Traffic Class (TC) e Flow Label (FL) quando estes campos têm valor zero (Z) e diferente de zero (N).

O campo INI da assinatura guarda como o dispositivo alvo inicializa os campos de seus pacotes. Ele é formado pela quádrupla $\langle \text{FL}_\text{TE}, \text{FL}_\text{ER}, \text{TC}_\text{TE}, \text{TC}_\text{ER} \rangle$. Cada entrada indica como o dispositivo inicializa os campos Flow Label (FL) e Traffic Class (TC) para pacotes ICMP Time Exceeded (TE) e pacotes ICMP Echo Reply (ER). Cada campo pode assumir o valor ZERO, quando não inicializa o campo; COPY, quando o valor do campo é o mesmo da sonda que provocou a resposta ICMP; ou DIFF quando o valor é diferente de zero e diferente do valor do campo na sonda que provocou a resposta.

A medição é feita salto a salto, como no traceroute. Inicialmente sabemos o comportamento do ponto de medição e conseguimos calcular a assinatura de seu sucessor (primeiro salto) enviando sondas para ele e para o segundo salto. Fazemos isso até alcançar um destino.

No nosso exemplo, para detectar se o dispositivo alvo R_3 faz modificações em um pacote IPv6 ao encaminhá-lo para R_4 verificados os valores do Traffic Class e do Flow Label no cabeçalho encapsulado por pacotes ICMP Time Exceeded criados por R_4. Para verificar como dispositivo R_3 inicializa os campos Traffic Class e Flow Label, verificamos o cabeçalho IPv6 dos pacotes que ele envia. Note que esta verificação é impossível caso um dos dispositivos entre R_3 e o ponto de medição também faça modificações. Para respostas DIFF, inferimos que algum dispositivo no caminho do dispositivo alvo até o ponto de medição (*reverse path*) modifica os campos Traffic Class e Flow Label, ou que o dispositivo alvo inicializa os campos. Para respostas COPY, inferimos que os campos foram inicializados pelo dispositivo alvo e não foram modificados no caminho de volta até o ponto de medição, pois apenas o dispositivo alvo tem acesso aos valores utilizados ao inicializar a sonda. Para respostas ZERO, inferimos que o dispositivo alvo não inicializa os campos Flow Label e Traffic Class e que nenhum dispositivo no caminho de volta até o ponto de medição modifica os campos.

Id.	País	Cidade	ASN
cph-dk	Dinamarca	Ballerup	59469
dac-bd	Bangladesh	Dhaka	24122
hnl-us	Estados Unidos	Honolulu, HI	6360
lax-us	Estados Unidos	Los Angeles, CA	2152
san-us	Estados Unidos	San Diego, CA	1909
sin-sg	Singapura	Singapore	37989
ylk-ca	Canadá	Barrie, ON	19764

Tabela 2. Pontos de medição utilizados para coleta

4. Coleta e caracterização dos dados

Coletamos assinaturas para um total de 19199 interfaces diferentes durante um período de 5 dias. Nossas medições foram realizadas em sete pontos de medição da infraestrutura do Ark, mantidos pela CAIDA (Center for Applied Internet Data Analysis) [Hyun 2016]. Os dispositivos estão localizados em redes distintas de cinco países, como é mostrado na tabela 2. Nossa lista de destinos possui 51928 endereços IP e foi formada a partir da *hitlist* IPv6 disponibilizada por Gasser et al. [Gasser et al. 2016]. A lista original possui cerca de 700 mil endereços IPv6 e construímos a nossa lista escolhendo aleatoriamente dois endereços IP para cada prefixo /48 da lista original.

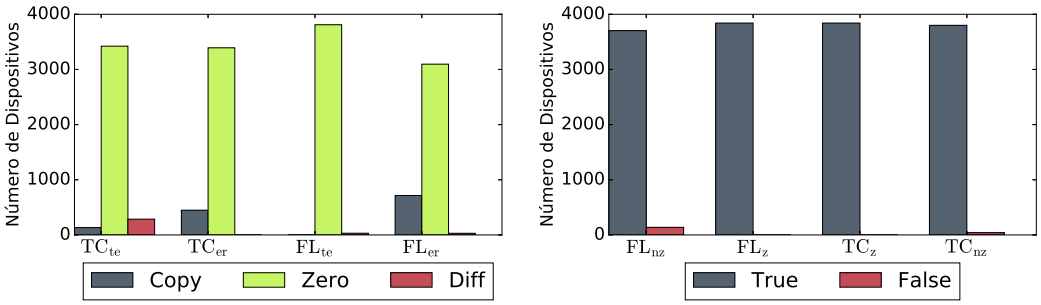
Para selecionar os dispositivos alvos, realizamos medições de traceroute partindo dos sete pontos de medição com destino a cada um dos 51928 endereços IP selecionados. Assim, cada um dos dispositivos que apareceram nesses caminhos são dispositivos alvos. Os dispositivos para os quais não conseguimos obter uma assinatura completa são desconsiderados da análise.

5. Resultados

Para identificar dispositivos com comportamento similar entre os dispositivos mensurados durante nossas medições, realizamos um agrupamento utilizando o algoritmo K-means, que recebe a assinatura de cada dispositivo e os agrupa em K classes definidas. É conhecido que um dos maiores desafios desse algoritmo é determinar o valor de K corretamente dado que é bastante subjetivo [Han et al. 2011]. Em nosso problema, não sabemos quais são os diferentes comportamentos presentes em nossas medições. Para contornar esse problema, utilizamos o método de Elbow para estimar o número de classes ideal que minimize o erro interno de cada grupo (i.e., consiga agrupar bem dispositivos similares) e, ao mesmo tempo, não especialize demasiadamente os mesmos (onde exista um grupo para cada dispositivo) [Kodinariya and Makwana 2013]. O método revelou que o valor de $K = 5$ é o melhor compromisso entre baixo erro interno e generalização dos grupos.

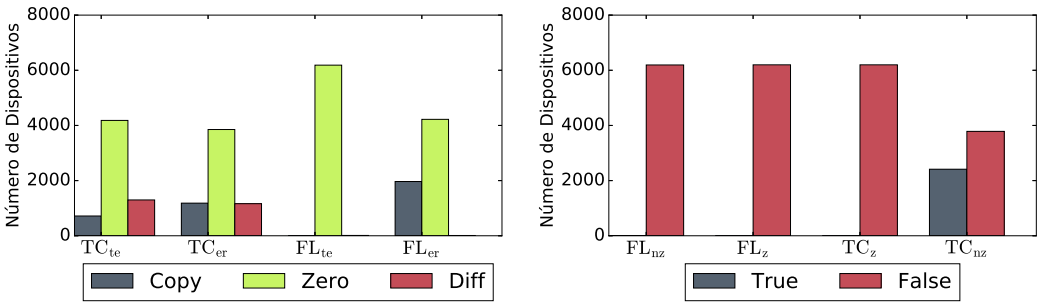
Dado a descoberta de cinco diferentes classes de dispositivos em nossa base, buscamos entender quais características as diferenciam. Através das figuras 1–5 é possível comparar os atributos de inicialização e modificação dos dispositivos em cada cluster, respectivamente. A seguir, descrevemos em detalhes cada um dos clusters identificados.

Cluster 1 (20%): A característica dos dispositivos do cluster 1 é que eles sempre modificam o valor do campo Flow Label e do campo Traffic Class ao encaminhar um pacote IPv6, independente do valor encontrado nesses campos. Além disso, a maior parte dos



(a) Atributos inicializados (b) Atributos modificados

Figura 1. Características do Cluster 1



(a) Atributos inicializados (b) Atributos modificados

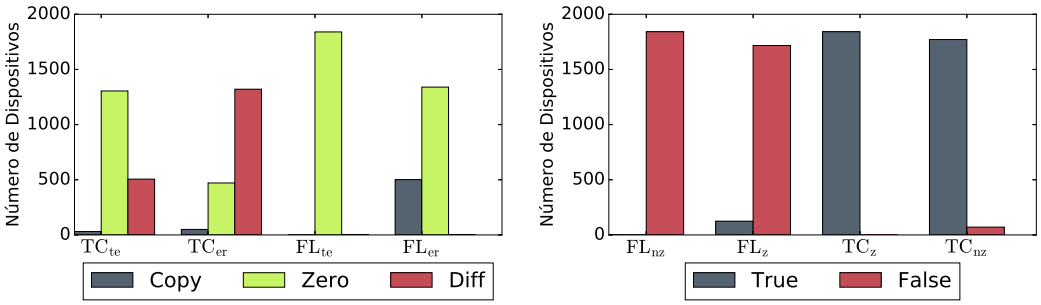
Figura 2. Características do Cluster 2

dispositivos nesse cluster inicializam os campos Traffic Class e Flow Label dos pacotes ICMP Time Exceeded e ICMP Echo Reply com ZERO.

Cluster 2 (32%): O cluster 2 é formado em sua maioria por roteadores que não modificam o campo Flow Label ao encaminhar um pacote. Os dispositivos deste cluster também não modificam o campo Traffic Class quando este está zerado, mas alguns dos dispositivos (menos da metade) modificam o Traffic Class quando este é diferente de zero. Além disso, grande parte dos dispositivos do cluster 2 inicializa os campos Traffic Class e Flow Label dos pacotes ICMP Time Exceeded e ICMP Echo Reply com ZERO.

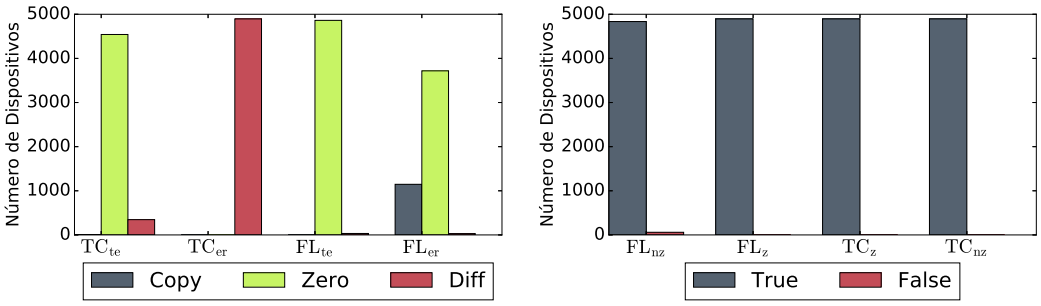
Cluster 3 (10%): O cluster 3 é formado, em geral, por dispositivos que modificam o valor do campo Traffic Class mas não modificam o valor do campo Flow Label ao encaminhar um pacote IPv6, independente do valor encontrado nesses campos. Quanto a inicialização dos campos dos pacotes ICMP, todos dispositivos deste cluster inicializam o Flow Label de pacotes Time Exceeded com ZERO, e grande parte inicializa o Flow Label dos pacotes ICMP Echo Reply também com ZERO, mas existe uma parte considerável de dispositivos que inicializa o Flow Label com o valor encontrado no pacote que causou a resposta ICMP (COPY). Em pacotes Time Exceeded, grande parte dos dispositivos inicializam o Traffic Class com ZERO, já em pacotes Echo Reply, a grande maioria inicializa com valores diferentes de 0 e diferentes dos pacotes que causaram a resposta.

Cluster 4 (25%): O cluster 4, por sua vez, é formado por dispositivos que modificam o campo Traffic Class e Flow Label dos pacotes independente do valor destes campos. A diferença para o cluster 1 fica em como estes dispositivos inicializam os campos do



(a) Atributos inicializados (b) Atributos modificados

Figura 3. Características do Cluster 3



(a) Atributos inicializados (b) Atributos modificados

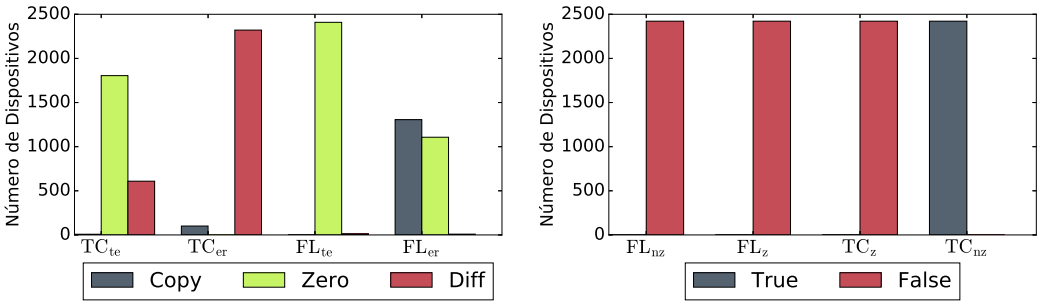
Figura 4. Características do Cluster 4

cabeçalho IPv6 nas mensagens ICMP de resposta. Todos os dispositivos do cluster 4 inicializam o Traffic Class para pacotes Echo Reply com valores diferentes de ZERO e diferentes das sondas que causaram a resposta (DIFF). No caso de pacotes Time Exceeded, os dispositivos, em geral, inicializam o Traffic Class com ZERO. Para o Flow Label, a característica dos dispositivos do cluster 4 é que eles inicializam este campo com ZERO tanto para pacotes com mensagens Time Exceeded quanto para mensagens Echo Reply.

Cluster 5 (13%): O cluster 5 é composto por dispositivos que modificam o Traffic Class quando este é diferente de zero mas não modificam quando é zero. Além disso, os dispositivos do cluster 5 não modificam o campo Flow Label ao encaminhar um pacote IPv6. Já as características de inicialização dos dispositivos do cluster 5 é que eles inicializam com ZERO o campo Flow Label dos pacotes com mensagem Time Exceeded, no entanto para pacotes com mensagem Echo Reply, estes dispositivos copiam o valor do pacote que causou a resposta.

5.1. Inicialização do Campo TTL

Durante nossos experimentos, verificamos a possibilidade de levar em conta também a inicialização do campo Hop Limit (TTL no IPv4) dos pacotes ICMP Time Exceeded e ICMP Echo Reply, como foi proposto por Vanaubel et al. [Vanaubel et al. 2013]. O trabalho de Vanaubel et al. mostrou que é possível classificar dispositivos de redes de forma bem definida levando em conta o TTL inicial de pacotes ICMP Time Exceeded e ICMP Echo Reply. No entanto, percebemos que tal classificação não é possível na Internet IPv6. Durante nossas medições, percebemos um comportamento que acreditamos ser



(a) Atributos inicializados (b) Atributos modificados

Figura 5. Características do Cluster 5

exclusivo do protocolo IPv6. Diferente do que Vanaubel et al. mostraram para a Internet IPv4, encontramos que 98,63% dos dispositivos na Internet IPv6 inicializam o Hop Limit com 64, tanto para pacotes ICMP Time Exceeded quanto para ICMP Echo Reply. De fato, a RFC 1700 recomendou em 1994 que o valor padrão para o TTL deveria ser 64, no entanto isso não era seguido. Acreditamos que este comportamento mostra maior aderência dos padrões e recomendações nas implementações IPv6 se comparado com o comportamento mostrado por Vanaubel et al.

6. Trabalhos Relacionados

Técnicas de coleta de assinaturas de dispositivos de rede são estudadas há algum tempo. No entanto, grande parte possui foco no protocolo IPv4 [Shu and Lee 2006, Vanaubel et al. 2013, Paxson 1997]. Técnicas específicas para o protocolo IPv6 começaram a ser estudadas recentemente, com a crescente popularização deste protocolo. O Nmap [Lyon 2009], por exemplo, é uma ferramenta de detecção de assinatura de dispositivos de rede bastante popular. A ferramenta trabalha enviando sondas destinadas ao dispositivo alvo, construindo uma assinatura do dispositivo e comparando essa assinatura com outras já conhecidas previamente armazenadas em um banco de dados. O Nmap possui um módulo de identificação para o protocolo IPv6. Nosso trabalho vai além no sentido de que considera informações não só do dispositivo alvo como também de um dispositivo sucessor para formar a assinatura, desta forma conseguimos identificar os campos do protocolo IPv6 que o dispositivo alvo modifica.

O trabalho de Vanaubel et al. [Vanaubel et al. 2013] utiliza uma assinatura baseada no TTL inicial de pacotes ICMP Time Exceeded e Echo Reply e mostra que com essa assinatura é possível inferir características de dispositivos de redes, inclusive a marca do dispositivo. Inspirados pelos resultados de Vanaubel et al., verificamos que dispositivos IPv6 inicializam o campo Hop Limit com valor 64.

7. Conclusões e Trabalhos Futuros

Neste trabalho desenvolvemos um método de construção de assinatura para dispositivos de rede IPv6. Nosso método utiliza uma assinatura que é baseada nas características dos dispositivos ao processar diferentes pacotes IPv6. Nossa assinatura leva em conta o comportamento não só do dispositivo alvo como também dos dispositivos antecessores e de seu dispositivo sucessor. Durante cinco dias coletamos assinaturas de 19199 interfaces

diferentes que foram classificadas em cinco clusters bem definidos através do algoritmo K-means. Mostramos também que 98.63% dos dispositivos IPv6 inicializam o campo TTL com o valor 64, limitando a utilidade deste campo na identificação de dispositivos IPv6.

Em trabalhos futuros pretendemos melhorar nossa técnica de identificação de roteadores adicionando mais campos à assinatura. Por exemplo, esperamos que os cabeçalhos de extensão de um pacote IPv6 possam fornecer boas informações para compor a assinatura do dispositivo. Outro aspecto que pretendemos estudar é a quantidade de dados (o tamanho do *payload*) de pacotes ICMP Time Exceeded. Estas características podem servir, portanto, como mais campos de assinatura de um dispositivo de rede.

Referências

Amante, S., Carpenter, B., Jiang, S., and Rajahalme, J. (2011). Ipv6 flow label specification. RFC 6437, RFC Editor.

Deering, S. E. and Hinden, R. M. (1998). Internet protocol, version 6 (ipv6) specification. RFC 2460, RFC Editor.

Gasser, O., Scheitle, Q., Gebhard, S., and Carle, G. (2016). Scanning the ipv6 internet: Towards a comprehensive hitlist. *Traffic Monitoring and Analysis Workshop (TMA)*.

Google (2016). Ipv6 adoption statistics in the internet. (2016, June 29) Retrieved from https://www.google.com/intl/en/ipv6/statistics.html.

Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

Hyun, Y. (2016). Archipelago (ark) measurement infrastructure. (2016, July 15) Retrieved from http://www.caida.org/projects/ark/.

Kodinariya, T. M. and Makwana, P. R. (2013). Review on determining number of cluster in k-means clustering. *International Journal*, 1(6):90–95.

Lyon, G. F. (2009). *Nmap network scanning: The official Nmap project guide to network discovery and security scanning*. Insecure.

Paxson, V. (1997). Automated packet trace analysis of tcp implementations. In *Proceedings of the ACM SIGCOMM '97 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '97, pages 167–179, New York, NY, USA. ACM.

Shu, G. and Lee, D. (2006). Network protocol system fingerprinting-a formal approach. In *Proc. IEEE International Conference on Computer Communications (INFOCOM)*.

Vanaubel, Y., Pansiot, J.-J., Mérindol, P., and Donnet, B. (2013). Network fingerprinting: Ttl-based router signatures. In *Proc. ACM Internet Measurement Conference (IMC)*, pages 369–376. ACM.