DDoSKiller: Identificando e Mitigando Ataques em Infraestruturas de Rede Programáveis

Ariel G. Castro, Francisco G. Vogt, Marcelo C. Luizelli
Universidade Federal do Pampa

***Abstract*—A telemetria *in-band* emergiu recentemente como uma alternativa para a coleta de informações da rede, permitindo aumentar a visibilidade sobre o estado da infraestrutura de rede. Neste trabalho, apresenta-se a arquitetura inicial do DDoSKiller – um protótipo capaz de fornecer ao operador da rede maior visibilidade sobre o estado da infraestrutura. O protótipo identifica e mitiga ataques DDoS (*Distributed Denial-of-Service*) utilizando hardware programável e um arcabouço de softwares. A arquitetura proposta é composta de cinco componentes principais de software responsáveis por (i) coletar dados de teletria, (ii) analisar e identificar ataques, (iii) mitigar ataques identificados, (iv) visualizar estatísticas coletadas e (v) configurar o ambiente.**

***Index Terms*—Telemetria *in-band*, Identificação de Anomalias, Ataque DDoS, Redes Programáveis.**

I. Introdução

A telemetria *in-band* é um paradigma emergente de monitoramento de infraestruturas de rede que permite maior flexibilidade na coleta dos dados [1]. Permite-se, por meio da programação de dispositivos de encaminhamento, a inclusão de informações de monitoramento (por exemplo, tamanho da fila de um dispositivo, tempo de processamento de um pacote) nos próprios pacotes dos tráfegos de rede ativos. Essas informações fornecem maior cobertura sobre o estado da rede, auxiliando no gerenciamento da infraestrutura de rede e suas aplicações [2], [3].

Com a utilização de telemetria *in-band*, é possível aumentar o nível de visibilidade (isto é, cobertura dos dispositivos de rede) e ajustar a granularidade desejada sobre a coleta dos dados. Um maior nível de visibilidade da infraestrutura de rede permite que uma gama maior de problemas e anomalias como, por exemplo, detecção de *heavy hitters* e de ataques DDoS [4], [5] sejam identificados com maior acurácia em comparação com abordagens tradicionais de coleta de dados (por exemplo, a partir do protocolo SNMP). A coleta dos dados de telemetria é feita de forma sistemática, auxiliando o operador, que passa a possuir maior controle sobre o estado da rede, de forma a atender às necessidades de diferentes aplicações com a granularidade desejada.

Abordagens recentes [6], [7], [8], [9] têm focado em utilizar os dados de telemetria para identificar e prever comportamentos anômalos em infraestruturas de rede. Entretanto, os trabalhos até então desenvolvidos são limitados quanto a operacionalização em infraestruturas de rede reais. Pouco ainda foi feito para materializar a telemetria *in-band* em produtos comerciais. Nesse artigo, propõe-se uma arquitetura inicial para identificar anomalias em infraestruturas de rede com suporte à programabilidade. A ideia consiste em programar interfaces de rede (com suporte a programabilidade) para coletarem em tempo real dados úteis para o gerenciamento das mesmas. Por exemplo, coletar níveis de ocupação de fila e os fluxos identificados como *heavy hitters* ajudam na rápida identificação de ataques DDoS à infraestrutura. Para apoiar a correta operacionalização dos algoritmos executados em *hardware*, projeta-se uma arquitetura de *software* para gerenciar a coleta de dados (por exemplo, a frequência), e mais importante, para orquestrar de maneira automática a identificação de eventos anômalos (por exemplo, por meio de algoritmos de agrupamento). Por fim, apresenta-se o protótipo inicial da solução, o qual é implementado com a utilização de *hardwares* programáveis emergentes (por exemplo, SmartNICs Netronome) e soluções em *software* para a visualização dos dados coletados.

O restante do artigo está organizado como segue. Na Seção II, descreve-se os trabalhos mais recentes relacionados à telemetria *in-band*. Na Seção III, descreve-se de maneira geral a arquitetura proposta para a identificação de anomalias em infraestruturas de rede. Na Seção IV, apresentam-se os detalhes do protótipo desenvolvido. Por fim, conclui-se o trabalho com as considerações finais e perspectivas de trabalhos futuros na Seção V.

II. Trabalhos Relacionados

Aplicações de monitoramento de infraestruturas de rede são chave para a correta operação e identificação de comportamentos e anomalias. Muitas aplicações de monitoramento se beneficiam da identificação dos fluxos de rede (ou conjunto de) que mais consomem recursos (por exemplo, largura de banda) nas infraestruturas para identificar ataques, congestionamentos e para a engenharia de tráfego – apenas para mencionar alguns exemplos [10]. Os fluxos (ou o conjunto) de rede com o maior volume de dados são conhecidos como *heavy hitters*. A identificação dos *heavy hitters* consiste na contagem dos fluxos que mais contribuem para a utilização dos recursos nas infraestruturas [11].

Com a consolidação das redes definidas por software (SDN – *Software-defined Networks*), a identificação de *Heavy Hitters* tem o potencial de ser feita de maneira *online* pelo próprio plano de dados [10]. Ao se utilizar a identificação de *heavy hitters* no plano de dados, elimina-se a sobrecarga da transferência contínua de dados entre os planos de dados e de controle. Para além disso, a contagem dos fluxos tende a ser mais precisa que abordagens tradicionais de monitoramento baseadas em protocolos como, por exemplo, NetFlow ou SFlow. A

programabilidade do plano de dados permite redefinir o processamento dos pacotes nos dispositivos, incluindo o suporte a novos protocolos e funcionalidades. A programabilidade do plano de dados emergiu recentemente com a consolidação da linguagem P4 [12] e de dispositivos de encaminhamento programáveis (por exemplo, Barefoot Tofino e SmartNICS). A linguagem P4 é independente de protocolo (isto é, os protocolos são definidos pelo programador) e independente de alvo (isto é, o compilador utilizado gera um código objeto de acordo com a arquitetura utilizada).

Esforços recentes no contexto de telemetria *in-band* propõem instruir como esses dados são coletados pelos dispositivos de rede. [13] propõe que instruções simples sejam encapsuladas em pacotes e possam ser executadas em dispositivo programáveis para identificar problemas como *micro-burst* e congestionamento de rede. Para isso, considera-se que cada dispositivo seja responsável por implementar funções bem definidas: (a) *switches* programáveis encaminham e executam programas em pacotes *Ethernet* modificados – chamados de *Tiny Packet Programs* (TPPs), enquanto (b) *hosts* computam informações sobre o estado da rede, interagindo com o plano de controle quando necessário. Utilizando um dispositivo NetFPGA, observa-se um *overhead* baixo na inclusão das instruções nos TPPs, porém a variedade de aplicações possíveis nos TPPs ainda é restrita ao (i) conjunto de instruções existentes e (ii) e às próprias requisições – já que estas devem ser inicializadas pelos *hosts*.

EverFlow [14] e PathDump [4] coletam informações em infraestruturas de rede de Data Centers (DCN – *Datacenter Network*) para auxiliar no *debugging* e gerenciamento da infraestrutura, identificando a origem de falhas que podem causar problemas de roteamento, balanceamento de carga e potenciais *loops*. Ambos os trabalhos utilizam dispositivos de encaminhamento não programáveis para rastrear o caminho dos pacotes através da insfraestrutura de rede. EverFlow [14] prioriza a coleta de dados de telemetria a partir dos dispositivos de encaminhamento, enquanto que PathDump [4] visa distribuir essa função entre *hosts* e os próprios dispositivos de encaminhamento. Em suma, ambos são capazes de realizar diagnósticos sobre uma quantidade similar de problemas. Sonata [7] e NetVision [8] consideram, entretanto, que cada nó da infraestrutura de rede desempenha uma determinada função no processo de coleta dos metadados da rede. Sonata fornece uma *interface* declarativa para expressar requisições de telemetria em alto nível, as quais são compiladas e executadas nos dispositivos da infraestrutura de rede. Por sua vez, NetVision fornece uma interface que é responsável por abstrair as requisições de telemetria e gerar pacotes *probes* que são encaminhados através de caminhos pré-computados na infraestrutura para coletarem os dados de telemetria.

Apesar das abordagens existentes realizarem esforços para diagnosticar diferentes problemas na rede, pouco ainda foi feito materializar uma solução para identificar anomalias em infraestuturas de rede. Neste trabalho, com mencionado anteriormente, dá-se o primeiro passo na direção da construção de uma solução que integre as funcionalidades do plano de dados às camadas de gerenciamento de infraestrutura.

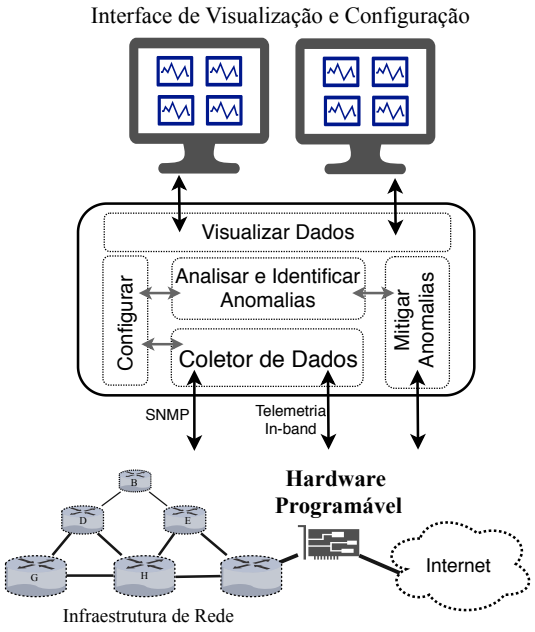

Fig. 1. Visão Geral da Solução Proposta

III. Arquitetura da Solução Proposta

Nesta seção, descreve-se em alto nível a arquitetura proposta de monitoramento de anomalias em infraestruturas de rede programáveis.

Como mencionado anteriormente, a solução proposta é baseada (i) na capacidade de coletar e computar informações de telemetria diretamente em interfaces de rede programáveis e (ii) em um arcabouço de software capaz de orquestrar a coleta e análise de dados em tempo real. A Figura 1 ilustra a visão geral da arquitetura da solução proposta. A arquitetura é composta por quatro componentes principais: (i) coletor de dados de telemetria, (ii) módulo de análise e identificação de anomalias, (iii) módulo de mitigação de anomalias, (iv) módulo de visualização e (v) o módulo de configuração da solução. Os módulos são descritos a seguir.

O coletor de dados é o módulo responsável por, periodicamente, solicitar dados de telemetria à interfaces de rede programáveis. A interface de rede é programada para "contar" os conjuntos de fluxos de rede que consomem maior volume de dados. O módulo de coleta pode, também, solicitar dados de monitoramento através de protocolos tradicionais (por exemplo, SNMP – *Simple Network Monitoring Protocol*).

Uma vez que o módulo de coleta de dados tenha solicitado os dados de telemetria mais recentes, os dados são gerenciados pelo módulo de análise e pode-se identificar possíveis anomalias. Este módulo é responsável por armazenar os dados e determinar o tempo de permanência na base de dados. Como o comportamento da infraestrutura pode variar ao longo do tempo, é importante manter apenas um subconjunto dos dados de telemetria coletados. Ademais, o componente constantemente analisa os dados através de algoritmos de agrupamento

Fig. 2. Exemplo da interface de monitoramento gerado com o *framework* Grafana.

(por exemplo, DenStream) capazes de identificar anomalias em tempo real. Caso seja identificado alguma anomalia (por exemplo, um ataque DDoS), o módulo de mitigação pode automaticamente interagir com a interface de rede programável e instruí-la em como proceder. Por exemplo, caso fosse detectado um ataque DDoS à infraestrutura, a interface de rede programável poderia bloquear o tráfego malicioso. Os dados coletados e analisados são visualizados a partir de uma interface web. A interface permite ao operador monitorar e identificar comportamentos anômalos. Ainda, os módulos mencionados são configuráveis a partir de um módulo próprio de configuração.

IV. Protótipo

Nesta seção, desecreve-se a implementação da arquitetura conceitual descrita na seção anterior. Para materializar a arquitetura proposta, desenvolveu-se (i) os algoritmos que executam diretamente na interface de rede (para a coleta de dados de telemetria) e (ii) os módulos capazes de coletar e visualizar os dados. É importante notar que o protótipo ainda está em fase de construção e, portanto, algumas funcionalidades ainda estão em desenvolvimento.

Para o protótipo, utiliza-se uma interface de rede programável Netronome SmartNIC Agilio CX 40GB[1]. A interface tem capacidade de processar 40 Gbits, o que permite ao protótipo escalar para infraestruturas de rede de médio e grande porte. A interface de rede é programada com a linguagem P4 para executar o algoritmo proposto por Basat et al. [11], o qual identifica o volume de dados que trafega entre os subconjuntos de fluxos de rede [11].

A interface de visualização foi desenvolvida através do *framework* Open Source Grafana[2]. O *framework* é capaz de interagir com qualquer base de dados e permite a personalização das séries temporais que serão visualizadas pelo operador da rede. A Figura 2 ilustra uma possível visualização dos dados coletados pelas interfaces de rede. É possível observar gráficos ilustrando séries temporais que representam a utilização dos recursos da infraestrutura de rede.

[1]https://www.netronome.com/products/agilio-cx/

[2]https://grafana.com/

V. Considerações Finais

Neste artigo, introduziu-se uma arquitetura inicial para a identificação de anomalias em infraestruturas de rede com suporte à programabilidade. A solução proposta consiste em programar interfaces de rede para coletarem em tempo real dados úteis para o gerenciamento das mesmas. O protótipo ainda está em fase de desenvolvimento e, portanto, pretende-se desenvolver e avaliar algoritmos para a identificação automática de anomalias de rede, além da correta integração entre os módulos da arquitetura.

References

[1] A. F. Q. Wu, J. Strassner and L. Zhang. (2016, Mar.) Network telemetry and big data analysis. [Online]. Available: https://tools.ietf.org/html/draft-wu-t2trg-network-telemetry-00

[2] Y. Z. C. Z. Jianzhe Liang, Jun Bi, "In-band network function telemetry," in *Proceedings of the Poster 2018 ACM Conference on SIGCOMM*, ser. SIGCOMM '18. New York, NY, USA: ACM, 2018, pp. 1–4.

[3] A. Gulenko, M. Wallschläger, and O. Kao, "A practical implementation of in-band network telemetry in open vswitch," in *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*. IEEE, 2018, pp. 1–4.

[4] P. Tammana, R. Agarwal, and M. Lee, "Simplifying datacenter network debugging with pathdump," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, Savannah, GA, 2016, pp. 233–248.

[5] Z. Liu, A. Manousis, G. Vorsanger, V. Sekar, and V. Braverman, "One sketch to rule them all: Rethinking network flow monitoring with univmon," in *Proceedings of the 2016 ACM Conference on SIGCOMM*, ser. SIGCOMM '16. New York, NY, USA: ACM, 2016, pp. 101–114.

[6] A. Putina, D. Rossi, A. Bifet, S. Barth, D. Pletcher, C. Precup, and P. Nivaggioli, "Telemetry-based stream-learning of bgp anomalies," in *Proceedings of the 2018 Workshop on Big Data Analytics and Machine Learning for Data Communication Networks*, ser. Big-DAMA '18. New York, NY, USA: ACM, 2018, pp. 15–20.

[7] A. Gupta, R. Harrison, M. Canini, N. Feamster, J. Rexford, and W. Willinger, "Sonata: Query-driven streaming network telemetry," in *Proceedings of the 2018 ACM Conference on SIGCOMM*, ser. SIGCOMM '18. New York, NY, USA: ACM, 2018.

[8] Z. Liu, J. Bi, Y. Zhou, Y. Wang, and Y. Lin, "Netvision: Towards network telemetry as a service," in *2018 IEEE 26th International Conference on Network Protocols (ICNP)*. IEEE, 2018, pp. 247–248.

[9] J. A. Marques, M. C. Luizelli, R. I. Tavares da Costa Filho, and L. P. Gaspary, "An optimization-based approach for efficient network monitoring using in-band network telemetry," *Journal of Internet Services and Applications*, vol. 10, no. 1, p. 12, Jun 2019.

[10] V. Sivaraman, S. Narayana, O. Rottenstreich, S. Muthukrishnan, and J. Rexford, "Heavy-hitter detection entirely in the data plane," in *Proceedings of the Symposium on SDN Research*, ser. SOSR '17. New York, NY, USA: ACM, 2017, pp. 164–176.

[11] R. Ben Basat, G. Einziger, R. Friedman, M. C. Luizelli, and E. Waisbard, "Constant time updates in hierarchical heavy hitters," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '17. New York, NY, USA: ACM, 2017, pp. 127–140.

[12] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 87–95, Jul. 2014.

[13] V. Jeyakumar, M. Alizadeh, Y. Geng, C. Kim, and D. Mazières, "Millions of little minions: Using packets for low latency network programming and visibility," in *Proceedings of the 2014 ACM Conference on SIGCOMM*, ser. SIGCOMM '14. New York, NY, USA: ACM, 2014, pp. 3–14.

[14] Y. Zhu, N. Kang, J. Cao, A. Greenberg, G. Lu, R. Mahajan, D. Maltz, L. Yuan, M. Zhang, B. Y. Zhao, and H. Zheng, "Packet-level telemetry in large datacenter networks," in *Proceedings of the 2015 ACM Conference on SIGCOMM*, ser. SIGCOMM '15. New York, NY, USA: ACM, 2015, pp. 479–491.