

Introdução ao BB-Gen

Rony Recalde

Universidade Estadual de Campinas

Resumo

O BB-Gen é um gerador de pacote simples baseado em CLI escrito em Python sobre a biblioteca Scapy para gerar fluxos de pacotes formatados como arquivos PCAP, tomando parâmetros definidos pelo usuário como entradas baseadas nos cabeçalhos definidos em um programa P4. Ele pode criar pacotes nativamente para diferentes protocolos padrão e personalizados. O objetivo é criar arquivos PCAP para serem usados com um amplo conjunto de geradores de tráfego (por exemplo, pktgen-dpdk, NFPA, TCPDUMP, etc.) ajudando os desenvolvedores de rede a validar a rede e executar testes de desempenho sobre os destinos.

KETWORDS

P4, Redes Definidas por Software, Análise de desempenho

Objetivo

Apresentar a ferramenta BB-Gen, mesma que foi desenvolvida na UNICAMP e que tem como finalidade criar pacotes que geram fluxos de pacotes formatados como arquivos PCAP a partir de uma entrada de arquivo P4, neste trabalho encontra-se focado ao uso do BB-Gen, alíás de apresentar a linguagem de programação de redes P4.

1 SDN & p4

Nos últimos anos, as redes se expandiram exponencialmente com o aumento do número de usuários, da infraestrut-

tura física e o surgimento de novos e avançados serviços de rede. As redes passaram a evoluir do hardware para o software. Agora, as redes podem ser definidas por software, sendo o hardware usado apenas como infraestrutura física. O processamento flexível de pacotes de caminho rápido em infraestruturas SDN (Software Defined Networking), a maioria notavelmente alimentada pela independência de protocolo DSL (Domain Specific Language) abordagem de P4 [1], exacerbar os desafios de transportar avaliação de desempenho adequada de dispositivos de caminho de dados. Devido à evolução demandas de tráfego e padrões de comunicação, processamento de pacotes diferentes tecnologias (por exemplo, DPDK, Netmap), diversas topologias de comunicação e casos de uso vire responder a pergunta tradicionalmente simples “ Quão rápido é este dispositivo de datapath é extraordinariamente meticuloso. Genérico existente geração de tráfego e ferramentas de benchmarking baseiam-se definições tradicionais e fixas de traços de fluxo que fornecem meios sintéticos para gerar e analisar o tráfego de teste e configurar o caminho de dados. P4 é uma linguagem de domínio específico para expressar como os pacotes são processados pelo plano de dados de um dispositivo de rede programável.[7,10] Executando avaliação experimental rica de diversas aplicações P4 em alvos heterogêneos (por exemplo, Netmap / DPDK / ODP, ARM,x86, 10G / 100G NICs), experimentamos, em primeira mão, os requisitos esforços para re-

alizer tal trabalho devido aos fatores combinatórios envolvidos. Dada uma aplicação P4, a flexibilidade da bção de P4 se torna uma maldição ao gerar rastreamentos de tráfego correspondentes e entradas de tabela P4 para o dispositivo de destino sob teste (DUT). Nesta demo, apresentamos o BB-Gen [12–14] para superar os riscos acima mencionados da P4 ágil avaliação de desempenho, mostra sua simplicidade e eficácia para gerar tráfego de rede e entradas de tabela P4 para diferentes casos de uso P4 com o aumento da complexidade, enquanto aumenta o número de tabelas (ou seja, l2fwd l3fwd, VXLAN, NAT), número personalizado de entradas e tamanhos de pacotes, compiladores de destino e plataformas.

2 BB-GEN DEMO

Seguindo uma abordagem, priorizando a simplicidade, a reutilização de componentes e a reprodutibilidade, o BB-Gen é um criador de pacotes que gera fluxos de pacotes como arquivos PCAP a partir de uma entrada de arquivo P4. O BB-Gen complementa outras ferramentas de benchmarking em vez de tentar substituí-las, usando ele podemos escolher a função de rede. O programa P4 é alimentado no BB-Gen, que contém dois módulos principais: (i) Core: composto pelos componentes Parser and Header / Data Generator; Primeiro, o código P4 é analisado para obter informações de cabeçalho de pacote. De acordo com as distribuições de valor de campo (escolhidas pelo usuário), os descritores de pacotes são gerados; (ii) Scapymodule: para montar os pacotes incluídos no descritor de lista de pacotes resultante do arquivo PCAP final.

Na demonstração do funcionamento da ferramenta, os participantes receberão uma escolha de programas P4 (por exemplo, L2fwd, L3fwd IPv4 / v6, GRE, VXLAN) [9], além de outras opções de configuração definidas pelo usuário. Agora é possível testar a ferramenta com qualquer um desses exemplos disponibilizados. Para apresentar os resultados, nesse caso foi usada a ferramenta wireshark para fazer o análise do fluxo de pacotes, a forma de testar a funcionalidade do BB-Gen é analisando os pacotes gerados, percebendo se os protocolos definidos no programa p4 são os obtidos no wireshark. Segue o link do repositório com as informações: <https://github.com/RonyRecalde/IA369Z>

3 Conclusões

O BB-Gen permitirá confiar nos recursos de extensibilidade do Scapy para criar pacotes com uma sequência arbitrária de cabeçalhos que podem não estar de acordo com os padrões, mas um P4 pouco ortodoxo programa pode exigir isso.

REFERENCES

- [1] Pat Bosshart, Dan Daly, Glen Gibb, Martin Izzard, Nick McKeown, Jennifer Rexford, Cole Schlesinger, Dan Talayco, Amin Vahdat, George Varghese, and David Walker. 2014. P4: Programming Protocolindependent Packet Processors. *SIGCOMM Comput. Commun. Rev.* 44, 3 (July 2014), 87–95. <https://doi.org/10.1145/2656877.2656890>
- [2] Scott Bradner and Jim McQuaid. 1999. Benchmarking Methodology for Network Interconnect Devices. RFC 2544. RFC Editor. <http://www.rfc-editor.org/rfc/rfc2544.txt>
- [3] P4 Language Consortium. 2016. p4app. <https://github.com/p4lang/p4app>.
- [4] P4Language Consortium. 2017. P4 Runtime. <https://p4.org/p4-runtime/>.
- [5] L. Csikor, M. Szalay, B. Sonkoly, and L. Toka. 2015. NFPA: Network function performance analyzer. In 2015 IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN). 15–17. <https://doi.org/10.1109/NFV-SDN.2015.7387395>
- [6] Huynh TuDang, Han Wang, Theo Jepsen, Gordon Brebner, Changhoon Kim, Jennifer Rexford, Robert Soulé, and Hakim Weatherspoon. 2017. Whippersnapper: A p4 language benchmark suite. In Proceedings of the Symposium on SDN Research. ACM, 95–101.
- [7] Sándor Laki, Dániel Horpácsi, Péter Vörös, Róbert Kitlei, Dániel Leskó, and Máté Tejfel. 2016. High Speed Packet Forwarding Compiled from Protocol Independent Data Plane Specifications. In Proceedings of the 2016 ACM SIGCOMM Conference (SIGCOMM ’16). ACM, New York, NY, USA, 629–630. <https://doi.org/10.1145/2934872.2959080>
- [8] Andres Nötzli, Jehandad Khan, Andy Fingerhut, Clark Barrett, and Peter Athanas. 2018. P4pktgen: Automated test case generation for p4 programs. In Proceedings of the Symposium on SDN Research. ACM, 5.
- [9] P Gyanesh Patra, Fabricio Rodriguez, Daniel Feferman, Juan Sebastian Mejia, Christian Esteve Rothenberg, and Gergely Pongracz. 2018. MACSAD. <https://github.com/intrig-unicamp/macsad>.
- [10] P Gyanesh Patra, Christian Esteve Rothenberg, and Gergely Pongrácz. 2016. MACSAD: Multi-Architecture Compiler System for Abstract Dataplanes (Aka Partnering P4 with ODP). In Proceedings of the 2016 ACM SIGCOMM Conference (SIGCOMM ’16). ACM, New York, NY, USA, 623–624. <https://doi.org/10.1145/2934872.2959077>
- [11] P. G. Patra, C. E. Rothenberg, and G. Pongracz. 2017. MACSAD: High performance dataplane applications on the move. In 2017 IEEE 18th International Confer-

ence on High Performance Switch- Rothenberg. 2018. BB-
ing and Routing (HPSR). 1–6. Gen. <https://github.com/intrig->
<https://doi.org/10.1109/HPSR.2017.7963684>amp/bb-gen.

[12] Fabricio Rodriguez and P Gyanesh
Patra. 2018. BB-Gen: A Packet
Crafter for P4 Target Evaluation
- Sigcomm Demo. <https://www.youtube.com/watch?v=430KjOsgJf0>

[13] Fabricio Rodriguez, P Gyanesh
Patra, and Christian Esteve

[14] Fabricio Rodriguez Cesen,
Pattam Gyanesh Kumar Patra,
and Christian Esteve Rothenberg.
2018. BB-Gen: A Packet Crafter
for Data Plane Evaluation. In
SBRC 2018 - Salão de Ferramentas.
[http://www.sbrc2018.ufscar.br/wp-](http://www.sbrc2018.ufscar.br/wp-content/uploads/2018/04/180625_1.pdf)
[content/uploads/2018/04/180625_1.pdf](http://www.sbrc2018.ufscar.br/wp-content/uploads/2018/04/180625_1.pdf)