10º **ENTEC – Encontro de Tecnologia: 28 de novembro a 3 de dezembro de 2016**

PROTÓTIPO DE UMA REDE DEFINIDA POR SOFTWARE UTILIZANDO A FERRAMENTA MININET

Pedro Henrique da Costa Avelar; Luiz Cláudio Theodoro[1]
[1] *Universidade Federal de Uberlândia*
pedro.avelar@hotmail.com; lclaudio@feelt.ufu.br

Resumo

A rede IP atual é extremamente rígida e de difícil manutenção. Switches e Roteadores devem ser configurados manualmente por técnicos para que uma mudança seja aplicada, causando um enorme custo para as empresas de telecomunicações. Além disto, tais impedimentos atrasam a inovação nas redes. Alternativas para isto se baseiam em Redes Definidas por Software (SDNs), onde os Switches se tornam elementos mais simples que são controlados por um Controlador externo. Isto cria uma facilidade na programabilidade das redes, visto que o controlador pode facilmente alterar as regras que os Switches rodam.

Além disto, algumas SDNs permitem que outros protocolos diferentes do IP possam funcionar em uma rede onde também há Hosts IP. Isto cria possibilidades enormes para inovação em redes.

O propósito deste trabalho é a demonstração de uma ferramenta de prototipagem e simulação de Redes Definidas por Software, chamado mininet, a fim de demonstrar as capacidades desta.

Palavras-chave: Internet do Futuro. Openflow. Controlador Usado. Palavra4. Palavra5.

1 Introdução

A Internet conecta cada vez mais dispositivos no mundo, crescendo sem parar. Novos dispositivos capazes de se conectar à internet são produzidos todos os dias e esta demanda fez com que os endereços IPv4 fossem esgotados há algum tempo atrás. Várias soluções de curto e longo prazo foram feitas a fim de tornar as redes mais flexíveis e de se prolongar a vida do protocolo IP, duas do último tipo são o IPv6, que aumenta a quantidade de endereços disponíveis, mas altera o formato do pacote e requer uma adoção lenta e gradual, e o NAT (Network Address Translation) que permite que uma sub-rede utilize somente um endereço IP da internet, mas que dificulta o acesso de dispositivos fora da sub-rede para dispositivos dentro dela.

Outras discussões sobre como melhorar a Internet, mais radicais, pretendem à suplantação total do protocolo IP. Estas ideias tem sido bastante discutidas nos ultimos anos (REXFORD; DOVROLIS, 2010), mas sempre encontraram uma incrível inércia. Esta inércia se deve em parte à dificuldade de se testar tais arquiteturas assim como o custo financeiro que seria causado devido à troca de equipamentos para atender ao novo protocolo. Desta forma, as Redes Definidas por Software (SDN - Software Defined Networks) são uma alternativa que busca gerar uma transição mais fácil.

O conceito base de SDN é definido de várias formas diferentes, mas segundo (KREUTZ et al., 2015) uma rede SDN se baseia em quatro pilares:

- A camada de Dados e de Controle são separadas.

- As decisões de roteamento são baseadas em fluxos ao invés de destino.
- A lógica de controle é movida para fora dos dispositivos de roteamento, ficando em controladores centralizados.
- A rede é programável, através de software que interage com o controlador.

Uma das soluções SDN é o padrão OpenFlow, que define um padrão de comunicação entre os Controladores OpenFlow e os Switches com capacidade OpenFlow, desta forma tornando os switches meramente em elementos de roteamento das camadas 1 e 2, que executam as regras programadas nas suas tabelas de fluxo, impostas pelo controlador, com o tráfego recebido.

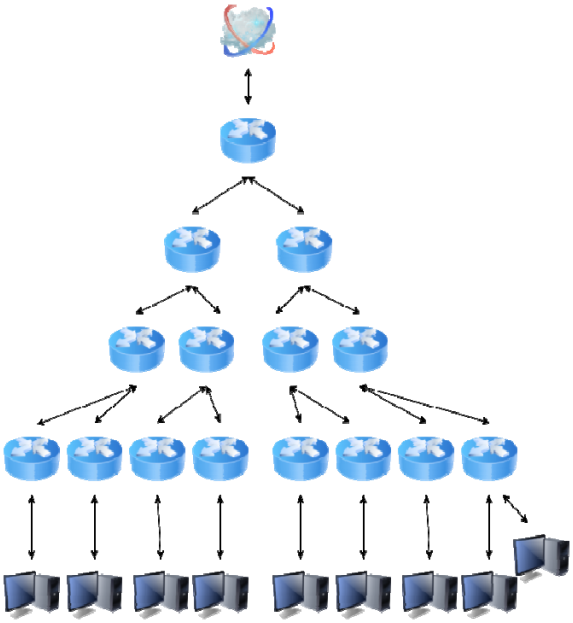
Além disto, várias ferramentas foram criadas a fim de se possibilitar a prototipação de redes e de aplicações de resdes. Uma destas é o software mininet, que utiliza uma virtualização leve ao nível do Sistema Operacional e permite que o usuário simule uma rede com conexões Ethernet, com Switches com capacidade Openflow e que permite a integração destes com controladores. Desta forma torna-se possível o teste de novas aplicações de redes com apenas um computador (LANTZ; HELLER; MCKEOWN, 2010).

O objetivo deste trabalho é portanto a demonstração da capacidade de simulação de Redes Definidas por Software pelo software mininet através da simulação de uma rede simples.

2 Materiais e Métodos

Para demonstrar a utilidade de redes SDN, testaremos a adaptabilidade de uma rede SDN. Para tal, utilizaremos a topologia em forma de árvore demonstrada na figura 1 e realizaremos algumas operações na rede a fim de testar como a rede será capaz de se reconfigurar para acomodar a falha no link. Adotando o apresentado em (HANDIGOL et al., 2013) e (HELLER et al., 2013) como norte para a implementação e teste de nossos experimentos, utilizamos a API Python fornecida pelo mininet para definir uma topologia e os testes a serem realizados. Assim, através do script produzido é possível a reprodução do experimento por outras pessoas.

Figura 1: Uma rede simples em forma de árvore



Utilizou-se o controlador padrão do mininet, que implementa o aprendizado da rede pelo switch. Este, entretanto, não suporta uma rede multicaminho e não permite que a mesma tenha laços dentro de si.

O procedimento seguido nos testes foi o seguinte:

1. Inicializa-se o Wireshark e a coleta de dados, filtrando para que se colete apenas dados referentes ao protocolo OpenFlow.

2. Cria-se a instância do mininet, rodando o script que define a rede e os testes desta forma os testes a seguir são executados sequencialmente.
3. Testa-se a conectividade entre os hosts realizando ping entre cada par de hosts.
4. Testa-se a velocidade das conexões entre os hosts com o host de saída.
5. Remove-se uma conexão aleatória, simulando uma falha na rede.
6. Testa-se novamente a conectividade com o ping para constatar se a rede funciona como esperado.

Desta forma, obteremos dados relacionados à conectividade da rede e da velocidade da conexão entre cada máquina e a máquina que representa a saída para a internet. Também, após ser simulada uma falha na rede testa-se o ping novamente a fim de verificar se rede funciona como esperado.

3 Resultados

Após o teste ser realizado várias vezes e os dados de cada vez serem coletados, realizou-se uma média dos valores apresentados na **Tabela 1:** Medidas Realizadas. Os casos onde os danos na rede causassem a impossibilidade de comunicação entre dois Hosts não foram incluidos na **Tabela 1:** Medidas Realizadas visto que nada poderia ter sido feito naquela situação.

Tabela 1 – Medidas Realizadas.

Métrica	Valor (Média entre os testes)
Tempo do primeiro ping	9.1666 ms
Tempo do segundo ping	6.0820 ms

Outro dado relevante é que do primeiro para o segundo ping, houve uma queda média de 13.0619 ms, o que demonstra as tabelas de fluxo sendo formuladas dentro dos switches pelo controlador.

4 Discussão

Como visto nos dados coletados, podemos ver que a rede se comporta de acordo com o esperado com a falha simulada. Assim, foi possível observar a utilidade do mininet para a prototipagem de redes definidas por software visto a facilidade de implementação da rede e dos testes realizados.

Foi possível também perceber o controlador realizando sua tarefa de programar os switches após as requisições de ping iniciais, quando os switches ainda não tinham nenhuma tabela preenchida e nenhum caminho aprendido.

5 Conclusão

Através da realização deste trabalho foi possível concluir que a ferramenta mininet é de fácil utilização para a prototipagem de redes ao permitir que as redes sejam testadas em software antes de implementadas fisicamente, com um código enxuto e fácil de ser lido.

Como consequência disto, podemos sugerir como trabalhos futuros a realização de experimentos com controladores personalizados, sobre internet do futuro como (ARAÚJO; CAMARGOS, 2011) ou até o teste de redes que não utilizem o protocolo IP como os apresentados em (VRIJDERS et al., 2014), (DAY; GOLDSTEIN; NOLAN, 2011), (AHLGREN et al., 2006), (ZHANG et al., 2010) e (SILVA et al., 2014) (SILVA et al.,)

Referências

AHLGREN, B. et al. A node identity internetworking architecture. In: IEEE. Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications. [S.l.], 2006. p. 1–6.

ARAÚJO, T. M.; CAMARGOS, L. J. Verificação da viabilidade do uso de idtrees como endereços na internet. In: VII ENTEC – Encontro de Tecnologia da UNIUBE, 2011

DAY, J.; GOLDSTEIN, F.; NOLAN, J. The last waltz and moving beyond tcp/ip. JOURNAL OF THE INSTITUTE OF TELE-COMMUNICATIONS PROFESSIONALS, ITP-INST TELECOMMUNICATIONS PROFESSIONALS SUNBURY TE, GREEN STREET, SUNBURY ON THAMES, MIDDLESEX, TW16 6QJ, ENGLAND, v. 5, p. 42–50, 2011.

HANDIGOL, N. et al. Reproducible network experiments using container-based emulation. In: ACM. Proceedings of the 8th international conference on Emerging networking experiments and technologies. [S.l.], 2012. p. 253–264.

HELLER, B. Reproducible network research with high-fidelity emulation. Tese (Doutorado) — Stanford University, 2013. Disponível em: <https://stacks.stanford.edu/file/druid: zk853sv3422/heller_thesis-augmented.pdf>. Acesso em: 29 Ago. 2016.

KREUTZ, D. et al. Software-defined networking: A comprehensive survey. Proceedings of the IEEE, IEEE, v. 103, n. 1, p. 14–76, 2015.

LANTZ, B.; HELLER, B.; MCKEOWN, N. A network in a laptop: rapid prototyping for software-defined networks. In: ACM. Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. [S.l.], 2010. p. 19.

REXFORD, J.; DOVROLIS, C. Future internet architecture: clean-slate versus evolutionary research. Communications of the ACM, ACM, v. 53, n. 9, p. 36–40, 2010.

SILVA, D. A. D. et al. Uml-based modeling entity title architecture (etarch) protocols. Citeseer. SILVA, F. et al. Entity title architecture extensions towards advanced quality-oriented mobility control capabilities. In: IEEE. 2014 IEEE Symposium on Computers and Communications (ISCC). [S.l.], 2014. p. 1–6.

VRIJDERS, S. et al. Prototyping the recursive internet architecture: the irati project approach. IEEE Network, IEEE, v. 28, n. 2, p. 20–25, 2014.

ZHANG, L. et al. Named data networking (ndn) project. Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC, Citeseer, 2010