



Avaliação de desempenho: Virtualizadores de rede.

RF – Relatório Final

Autor: Jean Menossi

Data: 03/04/2016

1. Introdução

Diversas soluções hoje são compatíveis com o desenvolvimento do FIBRE, da interface para o usuário final ao hardware disponível, diversas propostas tem surgido, algumas com um nível de entrega e finalização muito maiores que outros. Nesse contexto, foi apresentada a proposta de trabalho com OpenStack que se mostrou inviável pela realização e reprodução de trabalhos que viriam a complementar as necessidades que o FIBRE necessita para inovar desde a entrega ao experimentador e experiência de utilização ao controle de recursos internos da ilha. Muitos desses trabalhos com OpenStack, mesmo com resultados positivos se demonstraram inviáveis de reprodução e customização para o ambiente, seja por falta de documentação ou por abandono da continuidade por parte dos criadores. Ainda focando no contexto que a testbed de internet para o futuro é voltada para soluções ainda experimentais, a proposta de trabalho com OMF 6 ganhou força pelo know how, documentação e comunidade extremamente ativa em relação a outros frameworks. Mesmo com uma nova abordagem de tratamento, alguns recursos ainda se perpetuam como necessários para o uso da testbed, sendo um deles a necessidade de virtualização, paralelizando o trabalho de diversos experimentadores dentro dos recursos físicos do FIBRE. Tendo essa tecnologia hoje como ponto vital do projeto e a capacidade de se trabalhar de diversas formas via API, a proposta se concentra na viabilidade e estudo sobre os virtualizadores de rede disponíveis que possam trabalhar com a nova proposta de OMF 6 com mais qualidade, maior entrega, escalabilidade e menor custo possível em termos de resultados, proporcionando ao experimentador uma pesquisa cada vez mais próxima da realidade, como o objetivo da testbed propõe. O relatório portanto, concentra as informações que foram adquiridas em pesquisa e benchmark de utilização de dois virtualizadores de rede disponíveis hoje e com maior visibilidade: FlowVisor e OpenVirtex. Para que houvesse um embasamento maior na escolha do virtualizador, visto que se inicia um período de mudanças, faz se necessário avaliar os recursos principais e possíveis alternativas, que sejam viáveis para o objetivo que se propõe dentro do contexto em que o FIBRE entrega. Este relatório tenta concentrar essas informações da forma mais cognitiva possível com testes de desempenho e avaliação entre o FlowVisor e OpenVirtex, com o auxílio de ferramentas já conhecidas voltadas exclusivamente para testes com controladores.

2. Ambiente de testes

Para condizer com os recursos disponíveis, todos os testes foram realizados em máquinas virtuais com configurações e recursos similares aos disponíveis na tesbed, visto que o objetivo principal é verificar a aplicabilidade de alguns recursos. Sendo assim, para cada teste uma nova instância de VM com Ubuntu 14.04 64 bits foi utilizada, rodando apenas o experimento vigente para diminuir qualquer interferência ou impedir qualquer indisponibilidade por parte do hospedeiro. Os recursos reservados para cada instância de VM, tanto para FlowVisor quanto OpenVirtex são os que seguem:

- 4 Gigas de memória.
- 8 GB de disco rígido.
- 4 Threads de processador.

Demais especificações se fazem desnecessárias para contextualizar os resultados apresentados na próxima seção.

O virtualizador escolhido foi o VirtualBox em ambiente Windows pela facilidade de manuseio e extração de resultados em computador local. Não existe nenhum motivo prévio apontando o VirtualBox como sendo superior ao Xen e outros virtualizadores, a escolha foi totalmente voltada para a facilidade em manter um versionamento através de snapshot de VM's e manuseio.

3. Lições aprendidas

Essa seção será subdivida em quatro partes: Levantamento de informações sobre o FlowVisor, levantamento de informações sobre o OpenVirtex, ferramenta de benchmark para controladores OpenFlow e por fim os resultados alcançados. Essa estrutura tenta seguir de forma cognitiva a construção de informações suficientes para uma avaliação concisa dos resultados apresentados.

3.1 FlowVisor

Já bem conhecido nos ambientes de experimentação e o pioneiro na ideia de virtualização de redes SDN, o FlowVisor conta com mecanismos de isolamento de tráfego e abstração de topologia, de forma que seja possível criar vários layouts de entrega da rede utilizando de uma mesma estrutura física. Mesmo contando com diversos mecanismos de isolamento, o foco do relatório concentra-se no benchmark e não na explicação de cada um dos componentes da tecnologia oferecida.

Entretanto, um tópico importante a ser levantado é questão de overhead adicionado em cada um dos slices fornecidos. Parte do funcionamento do próprio FlowVisor utilizar as tags de VLAN e VLAN Priority para marcar e isolar cada tráfego requisitado em determinado layout, sendo que mensagens de PACKET_IN, em uma tentativa de diminuir o máximo possível de delay entre a requisição e a resposta com o ACTION. Nesse contexto, em seu primeiro artigo e primeira versão, a tecnologia apresenta o seguinte resultado em questão de overhead:

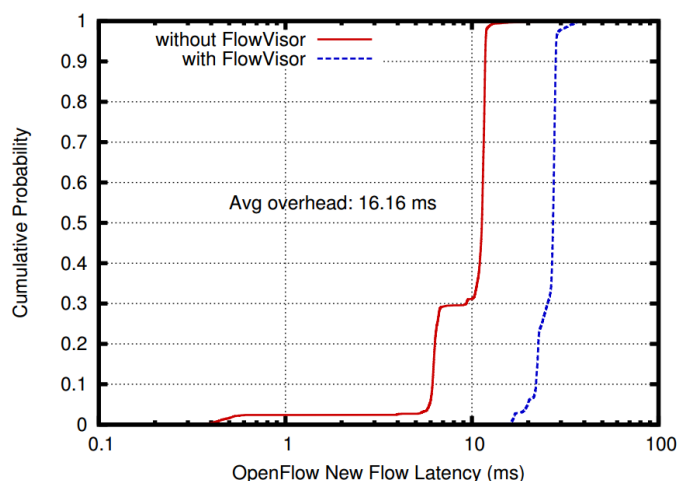


Figura 1 - Overhead FlowVisor [1]

Tendo em vista que o FlowVisor atua como um proxy entre o layout real e a abstração fornecida para o usuário, todo tráfego sofre uma inserção de overhead com um delay de pelo menos 16ms.

3.2 OpenVirtex

Sendo um dos principais componentes do Sistema Operacional de Rede – ONOS [5], o OpenVirtex tem uma fatia de uso garantido devido à entrega e sua arquitetura interna diferenciada. Fica como destaque o uso do MongoDB para armazenar links, dispositivos e conexões encontradas, dessa forma por mais que alguns links do layout físico deixem de funcionar, o virtualizador tem a capacidade de manter uma persistência entre links, tendo a base de dados como referência para reconfigurar o mesmo layout escolhido, característica importante a se levar em conta em ambientes de produção. Ainda, com essa mesma base de dados, alguns layouts podem ser salvos e reutilizados sempre que necessário, além da possibilidade de migração e snapshot da rede, utilizando o mesmo recurso de banco de dados. A Figura 2 demonstra o tempo de resposta de acordo com a quantidade de switches adicionados:

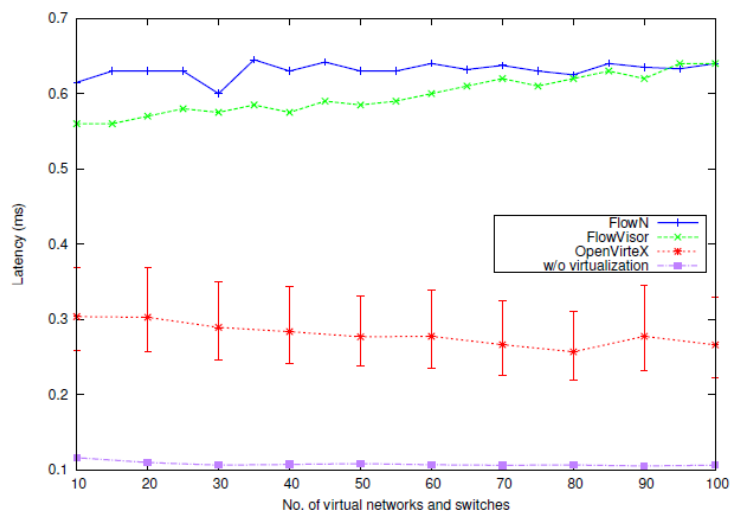


Figura 2 - Delay de resposta[2]

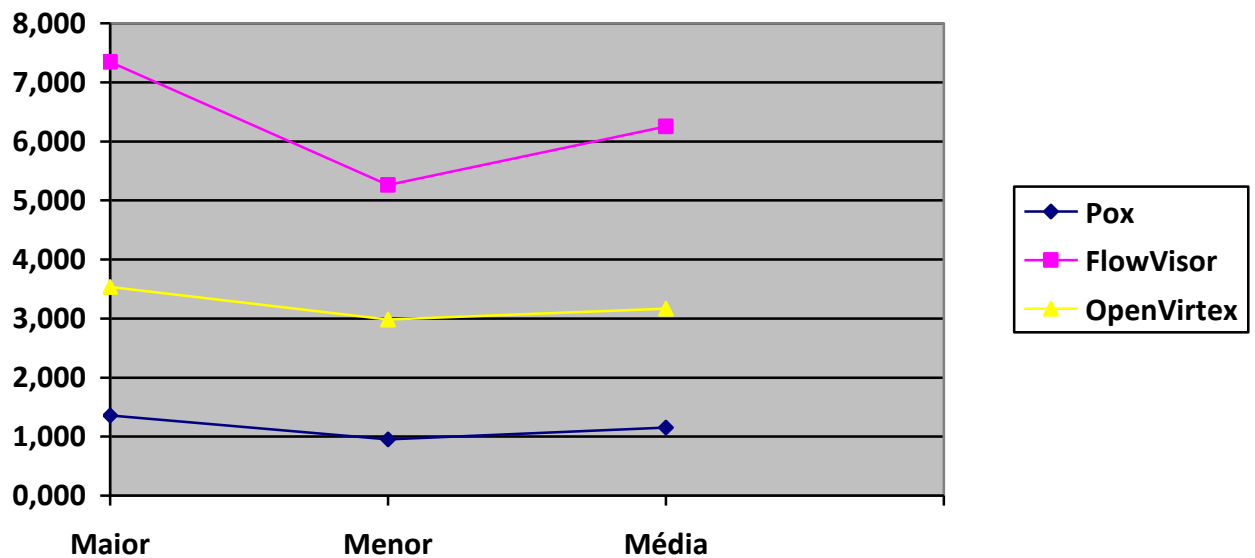
A questão de overhead fica bem mais simples de ser tratada, visto que faz parte de sua utilização apontar diretamente qual o escopo de endereçamento determinado ambiente virtual utilizará, ficando a cabo da southbound separar de acordo com a entrada e toda a arquitetura de northbound traduzir e reencaminhar os pacotes. Essa arquitetura bem definida e separada em cada um dos aspectos que abrangem a virtualização de uma rede definida por software demonstra grande preocupação dos desenvolvedores com desempenho e entrega, tornando o experimento pouco influenciado pela participação ativa do OpenVirtex como intermediário entre o controlador do experimentador e toda arquitetura física disponibilizada.

3.3 OFLOPS

Uma das ferramentas mais conhecidas para testes de desempenho de switches, contanto com gerador tráfego baseado em NetFPGA, o OFLOPS[3] oferece uma gama de testes que podem ser executados em Switches ou controladores. A inserção dessa ferramenta neste relatório, compreende uma de suas capacidades, concentrado no benchmark de controladores testando diretamente o tempo de resposta entre uma requisição (PACKET_IN) e uma resposta (ACTION). A ferramenta específica para os testes é o CBENCH (Controller Benchmark), que simula 16 switches fazendo requisições simultaneamente. Para cada resposta, o CBENCH apresenta o resultado total de tempo da operação total (requisição/resposta).

3.4 Resultados

Antes de inserir os resultados, é importante deixar claro as particularidades criadas no ambiente para cada um dos testes. Cada um dos ambientes foi preparado para um slice em cada um dos virtualizadores para testes exclusivos de tempo de resposta entre um PACKET_IN e um ACTION. É vital além do overhead sobre cada fluxo por slice, mensurar o tempo de resposta entre um experimento com um controlador específico, forward de informações do FlowVisor (virtualizador) e o equipamento que participa de determinado FlowSpace (slice). Para cada um dos resultados, foram 3 rodadas de execução para comparação da média, que compõem o gráfico a seguir:



Para comparações mais assertivas, o mesmo teste foi feito com um controlador Pox, posicionando a diferença de tempo de resposta entre um controlador comum e os virtualizadores. O experimento contou com várias rodadas de CBENCH, cada qual separadamente sem nenhum outro experimento em execução para diminuir a influência dos

fatores externos. Mesmo que o FlowVisor seja o pioneiro na ideia, fica claro que o OpenVirtex tem uma maturidade maior de resultados, seja pela sua arquitetura, recursos disponibilizados ou mesmo a questão do overhead, fortemente presente nos resultados do FlowVisor. Como já explicado durante a explicação do OFLOPS, o CBENCH insere 16 switches fazendo requisições simultaneamente, entregando os resultados individuais e resumidos ao final do experimento. Ainda, vale a pena lembrar do delay natural de uma rede, uma vez que todos os controladores, virtualizadores e mesmo o CBENCH foram utilizados em máquina local. Esses experimentos podem demonstrar de forma significativa o quanto o FlowVisor pode influenciar nos resultados de um experimento, tornando o um problema latente que, nessa nova proposta de arquitetura deve ser corrigido para acompanhar as tendências e crescente atratividade dos recursos disponibilizados no ambiente de produção para acesso dos experimentadores.

4. Conclusão e avaliação

Considerando os resultados apresentados no tópico anterior, é visível a necessidade de troca do virtualizador atual oferecido (FlowVisor 0.8) pelo FIBRE. A questão aqui levantada aponta diretamente não somente a questão do overhead e os tempos de resposta, mas a estabilidade no desenvolvimento e atualização do software. Parte desse desenvolvimento contínuo deve-se pelo fato do OpenVirtex ser uma das peças chave para o funcionamento do ONOS[5], grande foco em pesquisas e utilização em redes SDN. A diferença começa nos requisitos mínimos para utilização: O FlowVisor não pede mais do que 1.5 giga de memória e um único core enquanto o OpenVirtex pede para pleno funcionamento 4 gigas, 4 cores, sendo 1 giga alocado apenas para heap do Java. Além disso, existem outros recursos que valem a pena ser pontuados como vantagem pelo OpenVirtex, sendo um deles o uso de MongoDB para a persistência de informações relacionadas aos nós e controladores de cada slice ativo. Ainda, é importante citar a ampla adoção do ONOS em diversos novos experimentos, o que cria automaticamente uma relação com o OpenVirtex. Os problemas enfrentados em utilizar a última versão apresentada do FlowVisor (1.4) criam uma dependência de utilização de tecnologia legada, impossibilitando a adoção de novas funções e tornando a testbed cada vez mais presa em uma estrutura arcaica para os novos padrões e tecnologias. Levando em conta a experiência de uso com ambos os virtualizadores, é imprescindível colocar dois assuntos em discussão: O desenvolvimento de todo Ofelia aos moldes do FlowVisor e quais as dificuldades de migração para o OpenVirtex.

O primeiro ponto a ser considerado sobre o desenvolvimento são as características utilizadas no Ofelia ligadas diretamente ao FlowVisor. Tomando esse ponto como cerne da discussão, existem características próprias disponibilizadas pela API do virtualizador que entregam funcionalidades completas ao Expedient, como o mapeamento de links e nós. Outro ponto a ser levado em conta é a facilidade em gerar certificados de acesso que ligam diretamente o Expedient ao Opt-In, tornando a atualização e acesso aos recursos mais confiável. Porém, ainda nesse contexto, é importante ressaltar que foram encontrados

problemas em quase todas as novas versões apresentadas, inclusive a 1.4. Essa característica nos alerta sobre a dificuldade clara em moldar o Ofelia para as características puramente interessantes apenas ao FIBRE, seja por falta de documentação ou por complexidade de códigos não comentados, ficamos cada vez mais restritos ao que já foi de pleno funcionamento, porém com a dinâmica de desenvolvimento de novas tecnologias, a tendência a se tornar totalmente obsoleto e desinteressante à pesquisa é crescente. O outro ponto a ser levantado, remete novamente ao quanto o código do Ofelia está desenvolvido e pouco documentado estritamente ao FlowVisor: existe uma dificuldade muito grande em alterar a plataforma para novas tecnologias devido a quantidade de código e fluxos de desenvolvimento não documentados.

A experiência de utilização mostra duas vias de distintas de utilização: enquanto no FlowVisor criamos uma estrutura fortemente associada ao layout físico e em sequência o FlowSpace relacionado ao usuário, no OpenVirtex grande parte das informações (DPID e links) tem de ser adicionados manualmente em cada um dos ambientes virtualizados. Em um dos projetos apresentados por essa modalidade de extensão, foi possível visualizar que, para entregar o mesmo nível de entrega ao usuário final pelo OpenVirtex (como o layout disponível) várias alterações teriam de ser realizadas direto no Expedient, bem como no Opt-In para reserva de recursos. Diferente do FlowVisor o OpenVirtex não possui mecanismos de priorização de tráfego (já discutidos), porém em contrapartida o fato de OpenVirtex conseguir manter uma redundância de entrega através das informações diretamente relacionadas ao banco de dados.

Portanto, com todas as informações abordadas, acredito que seja indispensável a utilização de novas tecnologias na testbed, o OpenVirtex demonstra estabilidade e funcionamento suficientes para ser elegível à utilização, levando-se em conta as diferenças entre o FlowVisor e OpenVirtex já citadas levanta a questão de um certo esforço a ser despendido na questão de adaptação entre Opt-in, Expedient. Apresntando um desempenho pelo menos 50% maior que o FlowVisor, como fortemente demonstrado nas Figuras 1,2,3 respectivamente, o OpenVirtex demonstra uma robustez bem maior além da promessa de desenvolvimento continuado, justamente por ser parte de um grande projeto[5].

5. Referências Bibliográficas

- [1] SHERWOOD, Rob et al. Flowvisor: A network virtualization layer. **OpenFlow Switch Consortium, Tech. Rep**, p. 1-13, 2009.
- [2] AL-SHABIBI, Ali et al. Openvirtex: A network hypervisor. In: **Open Networking Summit 2014 (ONS 2014)**. 2014.
- [3] AL-SHABIBI, Ali et al. OpenVirteX: Make your virtual SDNs programmable. In: **Proceedings of the third workshop on Hot topics in software defined networking**. ACM, 2014. p. 25-30.

- [4] ROTSOS, Charalampos et al. OFLOPS: An open framework for OpenFlow switch evaluation. In: **Passive and Active Measurement**. Springer Berlin Heidelberg, 2012. p. 85-95.
- [5] BERDE, Pankaj et al. ONOS: towards an open, distributed SDN OS. In: **Proceedings of the third workshop on Hot topics in software defined networking**. ACM, 2014. p. 1-6.