

Comparação entre Ferramentas de Emulação

Gerson Porciúncula Siqueira¹, Eduardo Maroñas Monks²

¹Faculdade de Tecnologia Senac Pelotas (FATEC)
Caixa Postal 96015560 – Pelotas – RS – Brazil

{gersonporciuncula, emmonks}@gmail.com

Resumo. *Este artigo tem como objetivo analisar as ferramentas de emulação de redes, comparando as suas funcionalidades e medindo seu desempenho, quanto ao consumo de recursos da máquina hospedeira.*

Abstract. *This article is objetivo analyse the tools of networks emulation comparing their features and measuring its performance, as the consumption of resources on the host machine.*

1. Introdução

Atualmente as redes de computadores estão em todos os lugares, por esse motivo a área está em total evidência, cada vez mais fica-se dependente da Internet. Com o advento da Computação Móvel[UFOP 2014], faz pensar que a computação deve estar em todo lugar, para fazer com que essas informações sejam trocadas de uma maneira rápida e confiável o uso de redes de computadores se torna imprescindível.

Desta forma, novas tecnologias, dependem cada vez mais da Internet para troca de informações, por isso, passa-se a ter redes cada vez mais complexas, como essa tendência maior ao passar dos anos e ainda a migração do endereçamento das redes atuais para o IPv6 [NICBR 2014a], protocolo que existe a um certo tempo, mas poucos detêm conhecimento, há uma demanda enorme sobre profissionais da área, estes profissionais devem estar preparados para que sejam capazes de entender, planejar e administrar a complexidade destas novas tecnologias, tirando a máxima performance destes novos meios. Estes além de todo o conhecimento que já devam possuir, ainda necessitam estar sempre se atualizando, como cursos voltados para a área são raros e geralmente não se adequam ao horário, uma solução é o uso de laboratórios virtuais. Estes laboratórios são um meio de simular redes de pequeno a médio porte, através destas simulações, consegue-se fazer um estudo mais direcionado, focando mais no assunto desejado.

Utilizando-se de ferramentas de emulação de redes obtém-se algumas vantagens sobre a utilização de dispositivos reais, tais como a praticidade e o custo baixo para montar cenários complexos. Desta forma fez-se um estudo sobre quais simuladores de rede se adaptam melhor as necessidades dos profissionais e alunos. Sendo a proposta inicial, chegar a qual simulador existente atualmente no mercado atenda de uma forma eficaz, abrangendo as tecnologias existentes e com uma precisão mais próxima dos dispositivos físicos.

2. Ferramentas

O uso de simuladores e emuladores, tornou-se algo necessário, principalmente por existir uma dificuldade em poder trabalhar com dispositivos reais. Utilizando-se ambientes de

simulação, pode-se ter o estudo e a avaliação mais profunda do equipamento. Por exemplo, pode-se simular como um protocolo vai se comportar em redes distintas, sem ter que fazer a implantação deste em um ambiente de produção, algo impraticável por custo, tempo ou ainda fora das normas da empresa. Como o uso de simuladores é um ambiente controlado pode-se fazer inúmeros testes em que nada afetam o andamento do ambiente real. Entretanto há alguns problemas relacionados ao uso de simuladores, no ambiente simulado não a incidência de interferências externas, problemas com quedas de energia e usuários não ocorrem. Por vezes, acontece de que o que foi criado no ambiente simulado, não funcione com o mesmo desempenho no ambiente real.

No mercado de Redes de Computadores, destacam-se os simuladores GNS3 (*Graphical Network Simulator*) Cisco Packet Tracer, CORE (*Common Open Research Emulator*), eNSP (*Enterprise Network Simulation Platform*). Cada um com alguma suas características, mas de um modo geral todos atendem o que se comprometem, que é a simulação de redes.

2.1. Cisco Packet Tracer

Com o foco de auxiliar o estudante a compreender estas tecnologias de redes, especialmente tecnologias que envolvem equipamentos da Cisco System [Cisco 2014], empresa a qual domina boa parte do segmento de redes [Electrodata 2014], observou-se que necessitava-se de um modo de qualificar os profissionais para trabalharem com seus equipamentos. Desta forma surgiu a Cisco Networking Academy, com o principal propósito, de acompanhar o ritmo de evolução dos sistemas de rede, fornecendo currículos e ferramentas educacionais que ajudem o aluno a compreender a complexidade das tecnologias de informação e comunicação.

A partir disso, surgiu o Cisco Packet Tracer que é conhecido mais como um programa educacional ou aprendizagem (*e-learning*) do que propriamente um simulador. Serve como um complemento para os equipamentos físicos, permitindo criar redes com um número quase ilimitado de dispositivos, incentivando a livre prática, descoberta e resolução de problemas [Netacad 2014]. Apesar de ser considerado um programa gratuito, apenas usuários cadastrados em uma instituição de ensino, que esteja associada a Cisco poderá fazer *download* do *software*, que atualmente está na versão 6.0.1.0011. A instalação é simples, pode ser instalado no Microsoft Windows [Microsoft 2014], Ubuntu [Brasil 2014] e Fedora [Fedora 2014].

O Packet Tracer possui uma GUI (*Graphical User Interface*) utilizando o recurso de arrastar e soltar (*drag-and-drop*) como mostra a Figura 1, com suporte a conteúdos multimídia. Como faz o uso de vários protocolos, pode-se construir simulações complexas, consegue-se configurar cenários com uma grande diversidade. Na versão mais atual do *software* Packet Tracer, ele suporta os protocolos 802.11, 802.1q, Ethernet, VLAN, VTP, DTP, STP, RSTP, HTTP, DHCP, Telnet, SSH, TFTP, DNS, TCP, UDP, IPv4, IPv6, ICMP, ICMPv6, ARP, HDLC, PPP, Frame Relay, CDP, NAT, ACL. Sendo possível a utilização de protocolos de roteamento, bem como RIPv1, RIPv2, EIGRP, BGP, OSPF, entre outros [Netacad 2014].

Com o Packet Tracer pode ser demonstrado conceitos técnicos complexos e projetos de sistemas de redes, estes sistemas podem simular uma LAN e WAN [Tanebaum 2011] facilmente, utilizando os roteadores e protocolos disponíveis, pode-se

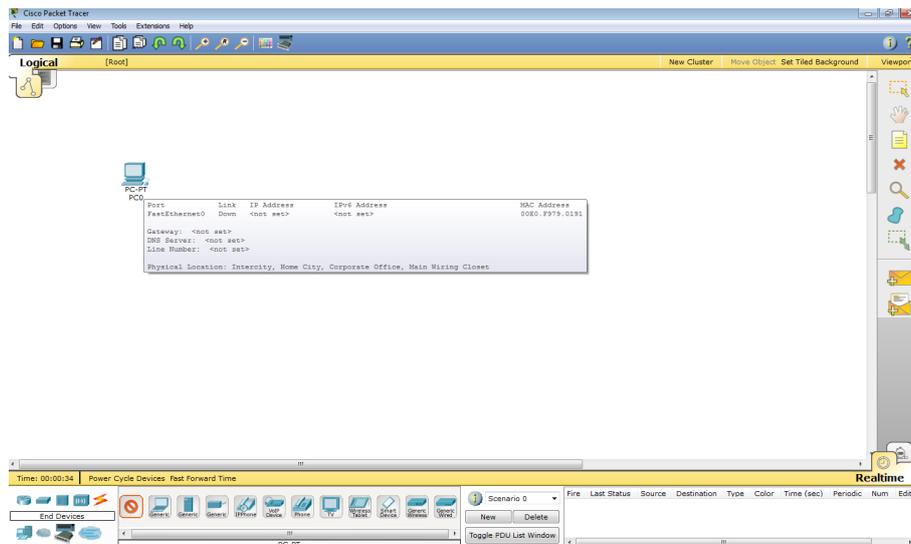


Figura 1. A interface *drag-and-drop*.

conectar várias LANs, montando um ambiente bem próximo de empresas e instituições de ensino.

As atividades podem ser desenvolvidas individualmente ou em grupos, o enfoque do programa é que os usuários pratiquem seu uso por linha de comando, para configurar os dispositivos. Ele torna-se uma importante ferramenta para os instrutores, pois permite a criação de tarefas personalizadas e orientadas, que apresenta um retorno quase que instantâneo aos usuários por meio do *Activity Wizard* (Figura 2).



Figura 2. O *Activity Wizard* permite a criação de atividades de aprendizagem customizadas.

O Packet Tracer trabalha com dois modos de operação para visualizar o comportamento de uma rede, o modo de **tempo real** e o **modo de simulação**. No de tempo real, a rede se comporta como fosse formada por dispositivos reais, com respostas imediatas em tempo real a todas as atividades da rede. O modo de tempo real fornece aos

alunos uma alternativa viável para equipamentos reais, permitindo que eles ganhem experiência em configuração antes de trabalharem com equipamentos reais. No modo de simulação, o usuário pode ver e controlar os intervalos de tempo, o funcionamento interno da transferência de dados e a propagação de dados em uma rede. Isso ajuda os usuários a compreender os conceitos fundamentais por trás das operações de rede.

2.2. CORE

O CORE é um emulador de redes, que serve principalmente para a construção de redes virtuais. Foi inicialmente desenvolvido pela Boeing [NICBR 2014b], com o nome inicial de Emunes, em seguida o nome do projeto foi alterado para CORE. Como a empresa não tinha mais como dar continuidade as pesquisas sobre a ferramenta, passou os estudos para a Marinha Americana. Além do CORE a Marinha possui outras ferramentas de simulação e emulação, sendo que algumas podem ser adicionadas, para trabalhar em conjunto com o CORE. A instalação pode ser feita via repositório, utilizando os pacotes ".deb" ou ".rpm" [Ferreira 2008], através da compilação dos códigos fonte do *software* que podem ser baixado ou ainda no *site* do desenvolvedor, há um máquina virtual que pode ser baixada e executada no VMWare [VMWare 2014] ou Virtual Box [Oracle 2014].

O Core pode ser conectado a dispositivos físicos, ele fornece um GUI de fácil uso como na Figura 3 e pode ser instalado nos sistemas operacionais Linux [Linux 2014] e FreeBSD [FreeBSD 2014]. Separa as processos das máquinas, faz uso de paravirtualização [UFRJ 2014]. Como ele faz uso de sistemas operacionais para simular seus *hosts*, pode ser instalado aplicações como Apache [Apache 2014], MySQL [MySQL 2014], PHP [PHP 2014].

Trabalha a partir da camada três do modelo OSI [Buis 2014], simulando as camadas física e de enlace, essas camadas podem ser emuladas utilizando o *software* Emane que também é desenvolvido pela Marinha Americana [EMANE 2014]. O CORE é um *software Open Source* [OpenSource 2014], desenvolvido em Python [Python 2014] e reutiliza programas como Netgraph [Netgraph 2014] e Netem [Foundation 2014].

Possui como limitação a quantidade de nós, ou *hosts*, essa limitação se dá pela capacidade da máquina hospedeira, caso a máquina que esteja a rodar o *software* não suporte mais o número de nós pode se usar uma aplicação do mesmo, que habilita a continuação da rede em outras máquinas, simulando a rede em diferentes computadores, foi desenvolvido para rodar em computadores simples, por isso grande parte das simulações podem ser executadas em máquinas de baixo porte.

O CORE não usa um *hardware* específico, usa *switchs* e roteadores genéricos, utiliza o Quagga para fazer a função dos roteadores. O Quagga é um conjunto de *software* de roteamento, fornece a implementação de alguns protocolos de roteamento para a plataforma Unix [Quagga 2014].

Como o simulador utiliza o Quagga, ele aceita os protocolos que estão definidos no *software* para roteamento e os protocolos aceitos no Linux para as máquinas.

2.3. eNSP

A empresa Huawei [Huawei 2014], de origem chinesa vem crescendo bastante no segmento corporativo [Abril 2014]. Como seus equipamentos estão se espalhando por vários

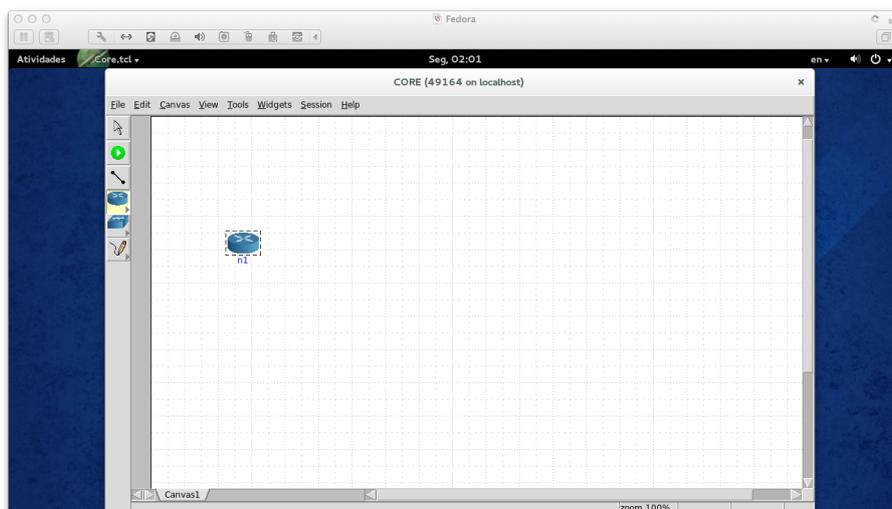


Figura 3. A interface do Core.

países, a empresa notou que deveria agir da mesma forma que a Cisco fez, ou seja, criar um método para qualificar os profissionais que trabalham com seus dispositivos.

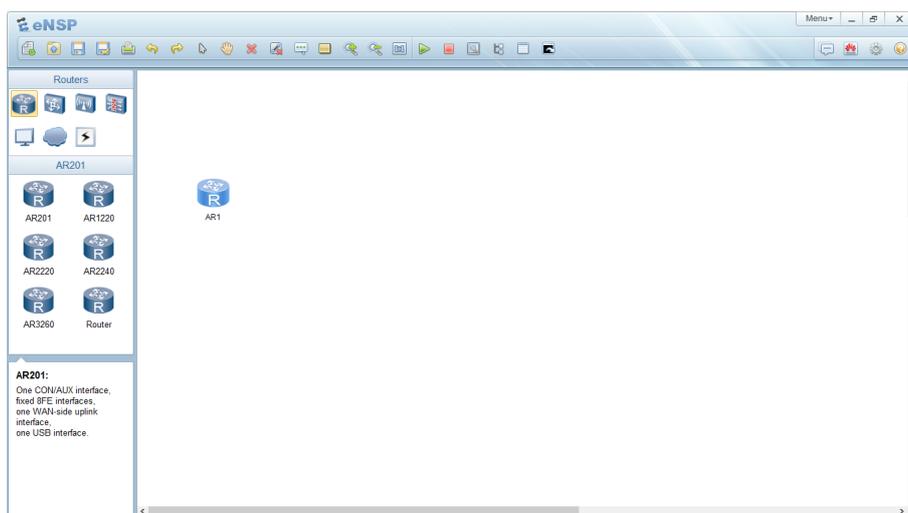


Figura 4. Interface Huawei.

Este método foi desenvolver o eNSP, um simulador livre [Licença 2014], que pode ser baixado pelo *site* da empresa, o principal função do simulador é auxiliar na obtenção do certificado HCDA (Huawei Certified Datacom Associate), se assemelha ao CCNA (Cisco Certified Network Associate). Atualmente esta na versão 1.2.00.350 , tem uma GUI com o uso bem simples, conforme Figura 4, usando o método de arrastar e soltar para montar os cenários de estudo.

Para fazer a instalação, primeiro deve-se fazer um cadastro no *site* da empresa e efetuar o *download*. A instalação é bem simples, com dois idiomas para instalação. Possui a opção de utilização de grande quantidade de protocolos, como o Ethernet, 802.11, 802.1q, 802.3 VTP, DTP, STP, RSTP, HTTP, DHCP, Telnet, SSH, TFTP, DNS, TCP, UDP, IPv4, IPv6, ICMP, ICMPv6, ARP, VLAN, NTP, ACL, HDLC, PPP, Frame Relay, CDP,

NAT, SNMP, GRE, VPN, MPLS, MSDP, IGMP, VRRP e o roteamento pode ser feito utilizando rota estática, RIP, OSPF, BGP, e IS-IS.

2.4. GNS3

O GNS3 é um simulador que suporta dois dos principais fabricantes de dispositivos de rede, Cisco e Juniper [Forbes 2014]. Possui o código-fonte aberto (*open source*), e utiliza a licença GPL [Software Foundation 2014], tendo como finalidade estar mais perto das redes reais. Utilizando uma GUI intuitiva semelhante a Figura 5, pode ser instalado nos sistemas operacionais Microsoft Windows, MacOS X [Apple 2014] e na maioria das distribuições Linux.

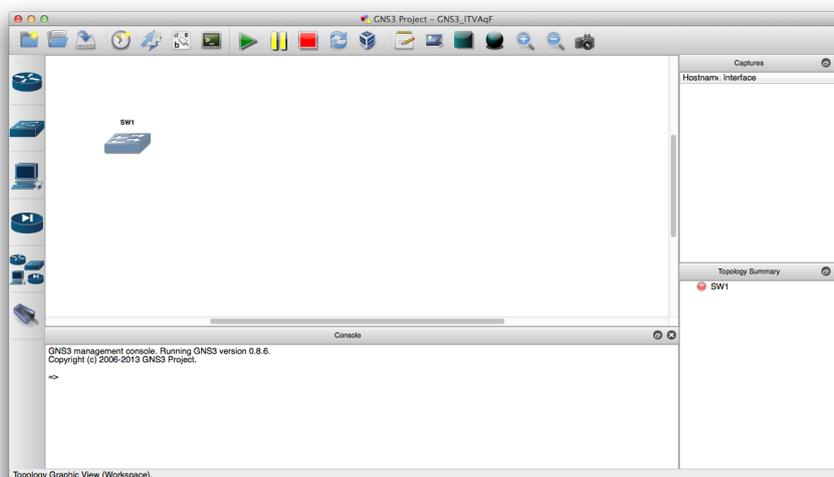


Figura 5. Interface GNS3.

A fim de se manter muito próximo, ele usa os emuladores Dynamips [GNS3 2014], que é um dos mais conhecidos emuladores do Cisco IOS [IOS 2014], o VirtualBox para fazer a instalação de sistemas operacionais e o Qemu que se trata de um emulador de máquinas *open source* [Qemu 2014]. Ainda pode fazer a emulação do Juniper Junos [Juniper 2014] e Cisco ASA, PIX e IPS. É indicado para profissionais que buscam as certificações Cisco CCNA, CCNP e CCIE, bem como Juniper JNCIA, JNCISE JNCIE, podendo ainda exercitar através da emulação de máquinas no VirtualBox, quem busca conhecimento e certificações Redhat (RHCE, RHCT) e Microsoft (MSCE, MSCA).

Por sua proximidade com a rede real, ele utiliza imagens dos próprios sistemas operacionais como Cisco IOS, PIX, ASA, IPS ou Juniper Junos.

3. Comparação

A primeira comparação que pode ser feita, é através da necessidade de cada *software* tem, para que possa ser instalado, esta necessidades podem ser observadas na Tabela 1,

Para fazer a comparação entre as ferramentas comentadas anteriormente, foi desenvolvido um cenário de teste, que foi executado em todas as ferramentas estudadas, com a finalidade de obter resultados sobre o desempenho das mesmas. Foi feita a comparação dividindo as ferramentas que possuem versões apenas para o Microsoft Windows e as que

possuem somente para sistemas operacionais Linux. Portanto ficando o GNS3 e o eNSP instalados no Microsoft Windows 7 e o Core e Packet Tracer no Ubuntu 14.4. O *software* Cacti [Cacti 2014] foi utilizado para gerar os gráficos do consumo dos simuladores, comparando o consumo de memória e de processador.

Tabela 1. Requisitos *software*.

<i>Hardware</i>	CORE	eNSP	GNS3	Packet Tracer
Processador	Dual Core 2Ghz	Dual Core 2Ghz	Dual Core 2Ghz	Pentium 300 Mhz
Memória RAM	2GB	2GB	3GB	100MB
Disco	3MB	1GB	1GB	250MB

3.1. Cenário

O cenário da Figura 6 foi desenvolvido para que as LAN distintas se comuniquem através do protocolo de roteamento dinâmico OSPF [Battisti]. Optou-se pelo OSPF por consumir um pouco mais de recursos do equipamento, trocam informações apenas sobre as rotas alteradas e tem a capacidade de dividir o cenários de teste em areas, permitindo o roteamento dentro da area e entre elas. Foi implantado em todos os *software* o mesmo endereçamento, para que os *softwares* possam ser comparados sobre as mesmas condições. Em anexo segue as configurações de alguns dos roteadores utilizados.

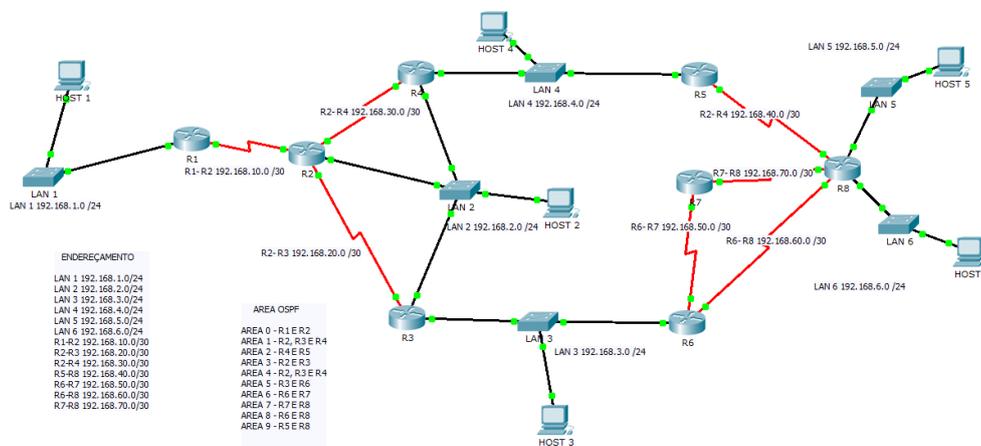


Figura 6. Cenário de teste.

3.2. Microsoft Windows

Para Windows foi utilizado o eNSP e o GNS3, usando como parâmetro de comparação o quanto de recurso da máquina eles consomem. Na Figura 7 imagem "A)", aparece o consumo de processador e memória do eNSP e na imagem "B)", o consumo do GNS3 é demonstrado. O GNS3 usa 100% de processamento em todos os núcleos da máquinas e em torno de 2 GB de memória RAM, quanto está executando a simulação do cenário proposto. Já o eNSP utilizou apenas algo em torno de 13.5 % em cada núcleo do processador mas utilizou o dobro de memória em comparação ao GNS3.

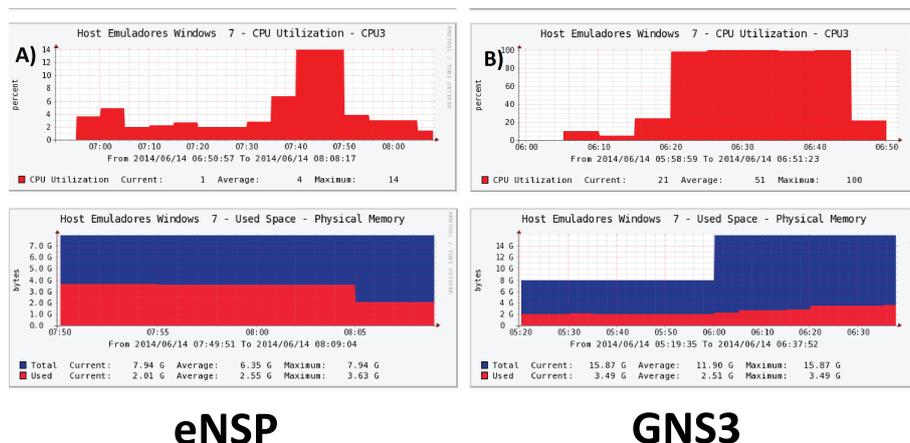


Figura 7. Comparação eNSP e GNS3.

3.3. Linux

O Cisco Packet Tracer e o CORE, foram instalados em uma máquina virtual, sendo o Ubuntu o sistema base. Na Figura 8, imagem "A", o consumo dos recursos do Core é apresentado e ao lado (imagem "B"), os gráficos de recursos utilizados pelo Packet Tracer, nota-se que o consumo de ambos é bem baixo, usando em torno de 50 MB de memória RAM e uma carga de processador um pouco mais baixa no Packet Tracer.

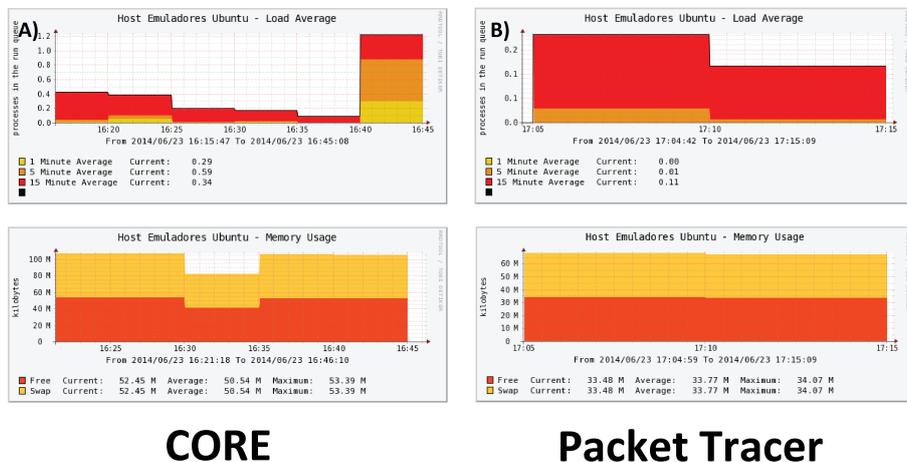


Figura 8. Comparação CORE e Packet Tracer.

4. Conclusões

Pode-se concluir que os simuladores que se mostram mais completos são o GNS3 e o eNSP. O GNS3 é um *software* que necessita de um processamento elevado, isto deve-se ao fato de ele utilizar uma imagem, emulando a com o Dynamips, já o eNSP torna-se um destaque por ser o único que simula equipamentos da Huawei, mesmo sendo seu consumo de memória mais elevado, não chega a afetar o desenvolvimento de pequenos laboratórios. Ideal para usuários que buscam certificação nos equipamentos da Huawei. Tanto o GNS3 e o Huawei tem como principal recurso a comunicação com a interface de rede do *host* no qual ele está sendo executado.

O Cisco Packet Tracer é indispensável para uso em aulas, por ser leve e de fácil entendimento, mas não aconselhável para quem busca montar cenários para teste de desempenho dos dispositivos e protocolos, pois não é preciso. Já o Core se destaca por consumir poucos recursos, ser mais personalizável e utilizar o Quagga para simular o roteador, mas tem como ponto fraco não ter dispositivos de fabricantes conhecidos e não possuir opções de cabeamento diferentes, o cabo direto é a única opção.

Referências

- Abril, E. (2014). Huawei vê crescimento mais forte no segmento corporativo. Disponível em: <<http://exame.abril.com.br/negocios/noticias/huawei-ve-crescimento-mais-forte-de-vendas-do-segmento-corporativo-2>>. Acesso em: março 2014.
- Apache (2014). Apache software foundation - projects. Disponível em: <<http://projects.apache.org>>. Acesso em: maio 2014.
- Apple (2014). Os x. Disponível em: <<https://www.apple.com/br/osx/>>. Acesso em: junho 2014.
- Battisti.
- Brasil, C. U. (2014). O que é ubuntu. Disponível em: <<http://ubuntu-br.org/ubuntu>>. Acesso em: março 2014.
- Buis, P. (2014). The ISO Layering Model. Disponível em: <<http://www.cs.bsu.edu/homepages/peb/cs637/layering/ISO.htm>>. Acesso em: abril 2014.
- Cacti (2014). Cacti - the complete rrdtool-based graphing solution. Disponível em: <<http://www.cacti.net/>>. Acesso em: junho 2014.
- Cisco (2014). Sobre a cisco. Disponível em: <http://www.cisco.com/web/BR/sobre/sobre_cisco.html>. Acesso em: março 2014.
- Electrodata (2014). Produtos cisco. Disponível em: <<http://www.electrodata.com.br/producto-cisco.htm>>. Acesso em: junho 2014.
- EMANE (2014). What is emane? Disponível em: <<http://downloads.pf.itd.nrl.navy.mil/docs/core/core-html/emane.html>>. Acesso em: junho 2014.
- Fedora, C. (2014). Freedom. friends. features. first. Disponível em: <http://fedoraproject.org/pt_BR/>. Acesso em: março 2014.
- Ferreira, R. E. (2008). *Linux - Guia do Administrador do Sistema*. Novatec, 2th edition.
- Forbes (2014). Cisco vs juniper: Delight or die! Disponível em: <<http://www.forbes.com/sites/stevedenning/2011/05/20/cisco-vs-juniper-delight-or-die/>>. Acesso em: maio 2014.
- Foundation, L. (2014). Netem. Disponível em: <<http://www.linuxfoundation.org/collaborate/workgroups/networking/netem>>. Acesso em: julho 2014.

FreeBSD (2014). The freebsd project. Disponível em: <<http://www.freebsd.org/>>. Acesso em: junho 2014.

GNS3 (2014). Dynamips. Disponível em: <<http://www.gns3.net/dynamips/>>. Acesso em: maio 2014.

Huawei (2014). Huawei. Disponível em: <<http://www.huawei.com/br/>>. Acesso em: junho 2014.

IOS, C. (2014). Cisco ios technologies. Disponível em: <<http://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-technologies/index.html>>. Acesso em: abril 2014.

Juniper (2014). Juniper networks. Disponível em: <<http://www.juniper.net/br/pt/products-services/nos/junos/>>. Acesso em: junho 2014.

Licença (2014). Huawei Enterprise Support Community. Disponível em: <<http://support.huawei.com/ecommunity/bbs/10185315.html?p=1#p10321917>>. Acesso em: março 2014.

Linux (2014). Linux. Disponível em: <<http://www.vivaolinux.com.br/linux/>>. Acesso em: junho 2014.

Microsoft (2014). Windows. Disponível em: <<http://windows.microsoft.com/pt-br/windows/home>>. Acesso em: março 2014.

MySQL (2014). Mysql - the world's most popular open source database. Disponível em: <<http://www.mysql.com>>. Acesso em: maio 2014.

Netacad (2014). Cisco Network Academy. Disponível em: <<http://www.netacadbrasil.com.br/index.php/ferramentas/7-packet-tracer>>. Acesso em: maio 2014.

Netgraph (2014). Freebsd Man Pages. Disponível em: <<http://www.freebsd.org/cgi/man.cgi?netgraph%284%29>>. Acesso em: julho 2014.

NICBR (2014a). O que é o ipv6, em português claro. Disponível em: <http://www.youtube.com/watch?v=_JbLr_C-HLk&list=PLQq8-9yVHyOZaK8rXJ0S-dui4r_YiXu9s%3Frel%3D0>. Acesso em: março 2014.

NICBR (2014b). Palestra fisl emulador core. Disponível em: <<https://www.youtube.com/watch?v=GIOzCZO5Pzw>>. Acesso em: maio 2014.

OpenSource (2014). Comunidade Open Source. Disponível em: <<http://softwarelivre.org/open-source-codigo-aberto>>. Acesso em: maio 2014.

Oracle (2014). Welcome to virtualbox.org! Disponível em: <<https://www.virtualbox.org/>>. Acesso em: maio 2014.

PHP (2014). Php - hypertext preprocessor. Disponível em: <<http://www.php.net>>. Acesso em: maio 2014.

Python (2014). The official home of the python programming language. Disponível em: <<https://www.python.org/about/>>. Acesso em: maio 2014.

- Quagga (2014). Quagga routing software suite. Disponível em:
<<http://www.nongnu.org/quagga/>>. Acesso em: maio 2014.
- Qemu (2014). Qemu Open Source Processor Emulator. Disponível em:
<http://wiki.qemu.org/Main_Page>. Acesso em: abril 2014.
- Software Foudation, F. (2014). Gnu general public license. Disponível em:
<<http://www.gnu.org/copyleft/gpl.html>>. Acesso em: abril 2014.
- Tanebaum (2011). *Redes de Computadores*. Pearson, 5th edition.
- UFOP (2014). Computação móvel introdução. Disponível em:
<http://www.decom.ufop.br/vicente/disciplinas/2013_2/comp_movel/material/introducao.pdf>.
Acesso em: abril 2014.
- UFRJ (2014). Paravirtualizacao. Disponível em:
<http://www.gta.ufrj.br/grad/08_1/virtual/Virtualizaototalepara-virtualizao.html>.
Acesso em: marco 2014.
- VMWare (2014). Virtualização. Disponível em:
<<http://www.vmware.com/br/virtualization/>>. Acesso em: maio 2014.

5. Anexo

Em anexo é demonstrado os passos para configurar um roteador em cada um dos *softwares*, a configuração pode ser replicada para os outros, apenas mudando o endereçamento.

5.1. Anexo A

A seguir os passos para configuração do roteador R1 do cenário de testes no emulador CORE.

```
root@n1:/tmp/pycore.44332/n1.conf#vtysh
n1#configure terminal
n1(config)#hostname R1
R1(config)#interface eth0
R1(config-if)#ip address 192.168.1.1/24
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#interface eth1
R1(config-if)#ip address 192.168.10.1/30
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#router ospf
R1(config-router)#router id 1
R1(config-router)#network 192.168.1.0/24 area 0
R1(config-router)#network 192.168.10.0/30 area 1
R1(config-router)#exit
R1(config)#exit
R1#write
```

E a configuração aplicada.

```
R1#write terminal
hostname R1
service integrated-vtysh-config
debug ospf6 lsa unknown
interface eth0
 ip address 192.168.1.1/24
 ipv6 nd suppress-ra
interface eth1
 ip address 192.168.10.1/30
 ipv6 nd suppress-ra
interface lo
router-id 10.0.0.1
router ospf
 ospf router-id 0.0.0.1
 network 192.168.1.0/24 area 0.0.0.0
 network 192.168.10.0/30 area 0.0.0.1
ip forwarding
ipv6 forwarding
line vty
end
```

5.2. Anexo B

A seguir a configuração do roteador R4 do cenário de testes no eNSP.

```
<Huawei>system-view
[Huawei]sysname R4
[R4]interface GigabitEthernet 0/0/0
[R4-GigabitEthernet0/0/0]ip address 192.168.2.1 255.255.255.0
[R4-GigabitEthernet0/0/0]undo shutdown
[R4-GigabitEthernet0/0/0]quit
[R4]interface loopback 0
[R4-LoopBack0]ip address 10.0.0.4 255.255.255.255
[R4-LoopBack0]quit
[R4]interface GigabitEthernet 0/0/1
[R4-GigabitEthernet0/0/1]ip address 192.168.4.2 255.255.255.0
[R4-GigabitEthernet0/0/1]undo shutdown
[R4-GigabitEthernet0/0/1]quit
[R4]interface Serial 0/0/0
[R4-Serial0/0/0]ip address 192.168.30.2 255.255.255.252
[R4-Serial0/0/0]undo shutdown
[R4-Serial0/0/0]quit
[R4]ospf
[R4-ospf-1]area 1
[R4-ospf-1-area-0.0.0.1]network 192.168.30.0 0.0.0.3
[R4-ospf-1-area-0.0.0.1]quit
[R4-ospf-1]area 2
[R4-ospf-1-area-0.0.0.2]network 192.168.4.0 0.0.0.255
[R4-ospf-1-area-0.0.0.2]quit
[R4-ospf-1]area 4
[R4-ospf-1-area-0.0.0.4]network 192.168.2.0 0.0.0.255
[R4-ospf-1-area-0.0.0.4]quit
```

E a configuração aplicada.

```
[R4]display current-configuration
sysname R4
router id 10.0.0.4
undo http server enable
nd user-bind detect retransmit 0 interval 0
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
interface Ethernet0/0/0
 shutdown
interface Ethernet0/0/1
 shutdown
```

```

interface Serial0/0/0
  link-protocol ppp
  undo shutdown
  ip address 192.168.30.2 255.255.255.252
interface Serial0/0/1
  link-protocol ppp
  shutdown
interface Serial0/0/2
  link-protocol ppp
  shutdown
interface Serial0/0/3
  link-protocol ppp
  shutdown
interface GigabitEthernet0/0/0
  undo shutdown
  ip address 192.168.2.1 255.255.255.0
interface GigabitEthernet0/0/1
  undo shutdown
  ip address 192.168.4.1 255.255.255.0
interface GigabitEthernet0/0/2
  shutdown
interface GigabitEthernet0/0/3
  shutdown
interface NULL0
interface LoopBack0
  ip address 10.0.0.4 255.255.255.255
ospf 1
  area 0.0.0.1
    network 192.168.30.0 0.0.0.3
  area 0.0.0.2
    network 192.168.4.0 0.0.0.255
  area 0.0.0.4
    network 192.168.2.0 0.0.0.255
user-interface con 0
user-interface vty 0 4
user-interface vty 16 20
return

```

5.3. Anexo C

A seguir a configuração do roteador R7 do cenário de testes no GNS3.

```

Router>enable
Router#configure terminal
Router(config)#hostname R7
R7(config)#interface Serial0/0/0
R7(config-if)#ip address 192.168.50.1 255.255.255.252
R7(config-if)#no shutdown

```

```
R7(config-if)#exit
R7(config)#interface Serial0/0/1
R7(config-if)#ip address 192.168.70.1 255.255.255.252
R7(config-if)#no shutdown
R7(config-if)#exit
R7(config)#interface loopback 0
R7(config-if)#ip address 10.0.0.7 255.255.255.252
R7(config-if)#exit
R7(config)#router ospf 1
R7(config-router)#network 192.168.50.0 0.0.0.3 area 6
R7(config-router)#network 192.168.70.0 0.0.0.3 area 7
R7(config-router)#end
R7#configure terminal
R7(config)#router ospf 1
R7(config-router)#router-id 10.0.0.7
R7(config-router)#end
R7#clear ip ospf process
```

Exibição das configurações aplicadas

```
R3#show running-config
Building configuration...
Current configuration : 866 bytes
version 12.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
hostname R7
spanning-tree mode pvst
interface Loopback0
 ip address 10.0.0.7 255.255.255.255
interface Serial0/0/0
 ip address 192.168.50.1 255.255.255.252
interface Serial0/0/1
 ip address 192.168.70.1 255.255.255.252
interface Vlan1
 no ip address
 shutdown
router ospf 10
 log-adjacency-changes
router ospf 1
 log-adjacency-changes
 network 192.168.50.0 0.0.0.3 area 6
 network 192.168.70.0 0.0.0.3 area 7
ip classless
line con 0
line aux 0
line vty 0 4
```

```
login
end
```

5.4. Anexo D

A seguir a configuração do roteador R3 do cenário de testes no Packet Tracer.

```
Router>enable
Router#configure terminal
Router(config)#hostname R3
R3(config)#interface FastEthernet 0/0
R3(config-if)#ip address 192.168.2.3 255.255.255.0
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#interface FastEthernet 0/1
R3(config-if)#ip address 192.168.3.1 255.255.255.0
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#interface loopback 0
R3(config-if)#ip address 10.0.0.3 255.255.255.255
R3(config-if)#exit
R3(config)#router ospf 1
R3(config-router)#network 192.168.20.0 0.0.0.3 area 2
R3(config-router)#network 192.168.2.0 0.0.0.255 area 4
R3(config-router)#network 192.168.3.0 0.0.0.255 area 5
R3(config-router)#end
R3#configure terminal
R3(config)#router ospf 1
R3(config-router)#router-id 10.0.0.3
R3(config-router)#end
R3#clear ip ospf process
```

Exibição das configurações aplicadas.

```
R3#show running-config
Building configuration...
Current configuration : 866 bytes
version 12.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
hostname R3
spanning-tree mode pvst
interface Loopback0
 ip address 10.0.0.3 255.255.255.255
interface FastEthernet0/0
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
```

```
interface FastEthernet0/1
  ip address 192.168.3.1 255.255.255.0
  duplex auto
  speed auto
interface Serial0/0/0
  ip address 192.168.20.2 255.255.255.252
interface Serial0/0/1
  no ip address
interface Vlan1
  no ip address
  shutdown
router ospf 10
  log-adjacency-changes
router ospf 1
  log-adjacency-changes
  network 192.168.20.0 0.0.0.3 area 3
  network 192.168.3.0 0.0.0.255 area 5
  network 192.168.2.0 0.0.0.255 area 4
ip classless
line con 0
line aux 0
line vty 0 4
  login
end
```