
 INSTITUTO DE INFORMÁTICA UFG	Universidade Federal de Goiás Instituto de Informática	 UFG
Professor: Kleber Vieira Cardoso		
Laboratório: algumas ferramentas de rede, medição de atrasos, perdas, captura de pacotes.		

A menos que haja alguma orientação contrária, os exercícios se referem à máquina virtual (Kali Linux) disponibilizada para uso no VirtualBox (usuário: **root**, senha: **toor**).

1. Há várias ferramentas (ou programas) que acompanham as distribuições do sistema operacional Linux e servem para obter informações sobre a rede e as aplicações distribuídas. Geralmente, esses programas possuem um manual *online* no sistema que é acessível através do comando **man <programa>** ou **info <programa>**. A seguir, é apresentada uma lista de programas úteis. Utilize a documentação *online* ou procure na Web informações sobre esses programas e se familiarize com os mesmos.

- **ping**
- **traceroute**
- **mtr**
- **netstat**
- **netcat (ou nc)**
- **wireshark**

2. Utilizando os comandos listados no item anterior, obtenha as informações pedidas a seguir:

ping

- a) Execute o comando **ping** para o domínio **google.com**. Quais informações podem ser obtidas com a execução desse comando?
- b) Descubra o endereço IP que está associado a **reitoria.ufg.br**.
- c) Realize outro **ping** para o **google.com** mas enviando apenas 10 pacotes. Quantos pacotes foram transmitidos, recebidos e perdidos? Qual o RTT (*round-trip time*) máximo?
- d) Adicione um intervalo de espera de 2 segundos entre o envio de cada pacote.
- e) Realize um **ping** informando como parâmetro apenas o endereço IP. Há diferenças em relação ao comando informando o *host*?

traceroute

- a) O **traceroute** não vem instalado por padrão em algumas distribuições linux, portanto, é necessário realizar sua instalação antes de continuar as atividades abaixo.
 1. Acesse o terminal e instale o **traceroute** com o comando: **sudo apt-get install traceroute**
 2. Liste os parâmetros para utilizar o **traceroute** com o comando: **traceroute --help**
- b) Descubra quantos roteadores o separam do *site* da UFG (**ufg.br**).
- c) Agora repita o item anterior para o *site* do Google (**google.com**). Qual o número máximo de saltos e o tamanho dos pacotes enviados?
- d) Para o *site* do google, limite o número de saltos para 5.
- e) Mostre o endereço IP do *site* analisado.
- f) Configure o número de pacotes de teste por salto para 6. O que mudou na resposta, e o que representa essa informação nova?
- g) Verifique o tempo leva para enviar e receber um pacote de 1400 *bytes* do seu computador até o *site* IETF (**www.ietf.org**). Utilize 100 sondas para obter essa estimativa. Há variações significativas em cada amostra? Comente a respeito.

mtr

- a) O **mtr** não vem instalado por padrão em algumas distribuições linux, portanto, é necessário realizar sua instalação antes de continuar as atividades abaixo.
 1. Acesse o terminal e instale o **mtr** com o comando: **sudo apt-get install mtr**
 2. Liste os parâmetros para utilizar o mtr com o comando: **mtr --help**
- b) Utilize a ferramenta **mtr** indicando como destino o seguinte site: **kyoto-u.ac.jp**. Espere uns 60 segundos e informe qual *host* apresenta perda de pacotes?
- c) Após executar o **mtr** é possível identificar alguma diferença em relação ao **traceroute**?
- d) A ferramenta **mtr** permite modificar o modo de visualização, reiniciar as estatísticas e modificar a ordem dos campos visualizados. Explore essas funcionalidades para se familiarizar com a ferramenta.

netstat

- a) Liste todas as portas com **netstat**. Quais informações são retornadas? Dica: após o comando **netstat** utilize *pipe* (i.e., |) e o comando **more** (ou **less**) para auxiliar na visualização da lista retornada pelo **netstat**.
- b) Liste separadamente todas as conexões ativas que utilizam o protocolo **UDP** e depois o **TCP**.
- c) Mostre as estatísticas de todos os protocolos. Além disso, filtre apenas as conexões que utilizam os protocolos **TCP** e **UDP**.
- d) Execute um navegador com algumas páginas abertas e outros programas que utilizam TCP. Execute o **netstat** para exibir os programas e o PID dos processos que estão executando em cada porta TCP.
- e) Execute o **netstat** para monitorar continuamente as portas com o comando do exercício anterior.

netcat | nc

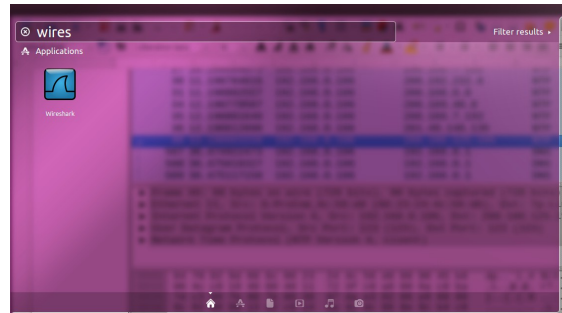
- a) Faça a varredura da faixa de portas 70-100 do domínio **ufg.br** e informe qual porta é possível conectar e qual protocolo está executando em cada uma.
- b) Utilize os mesmos parâmetros do exercício anterior, mas informando apenas o endereço IP do domínio **ufg.br**. Quais diferenças podem ser observadas em relação à execução do comando do exercício anterior? (dica: verifique a documentação do **netcat** (digite: **man netcat**) para descobrir qual parâmetro utilizar).
- c) Utilizando o programa **netcat** (ou **nc**), estabeleça um canal de comunicação local, criando um pequeno chat (troca de mensagens de texto). Será necessário abrir dois terminais onde pelo menos um deverá informar o endereço de domínio **localhost** para a rede local. Realize alguns testes com UDP e TCP. Há diferenças entre os dois protocolos? Comente a respeito.
- d) Responda as mesmas perguntas do exercício anterior, mas agora estabeleça um canal de comunicação com outro(a) colega do laboratório. Pelo menos um(a) terá que saber o endereço IP do equipamento do(a) outro(a) (dica: use o comando **ifconfig eth0** para saber IP utilizado pela máquina).

wireshark

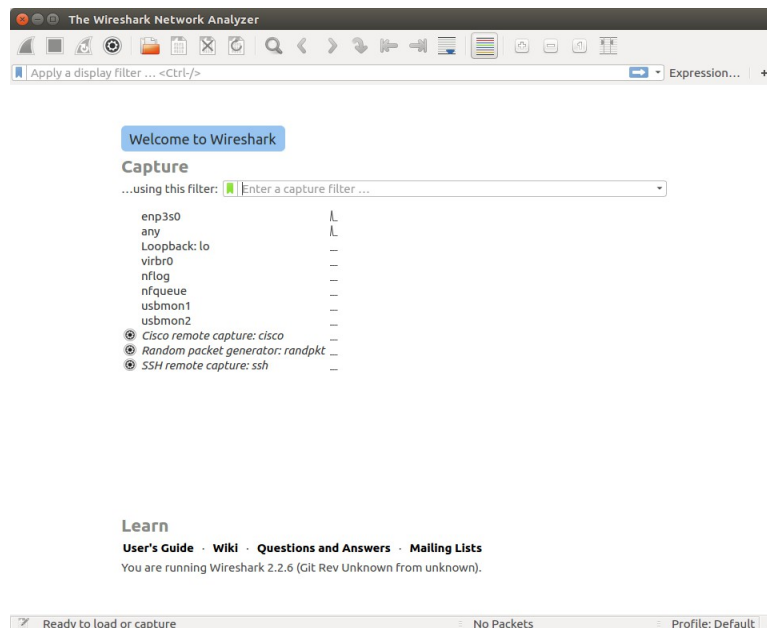
3. A captura de pacotes de uma rede de computadores é uma prática comum com diferentes propósitos. Um desses propósitos é entender o funcionamento dos protocolos de comunicação e da própria rede. Há várias ferramentas disponíveis para realizar essa tarefa, uma delas é o **Wireshark**. Para instalar e configurar o **wireshark** siga as instruções abaixo:

- Instale o Wireshark com o comando: **sudo apt-get install wireshark**
- O Wireshark possui extensa documentação que pode ser acessada a partir da própria ferramenta (*menu => help*) ou em seu site Web: wireshark.org/docs/.
 - Uma parte da documentação que será especialmente útil nesta atividade de laboratório é a referente a filtros: wiki.wireshark.org/DisplayFilters.

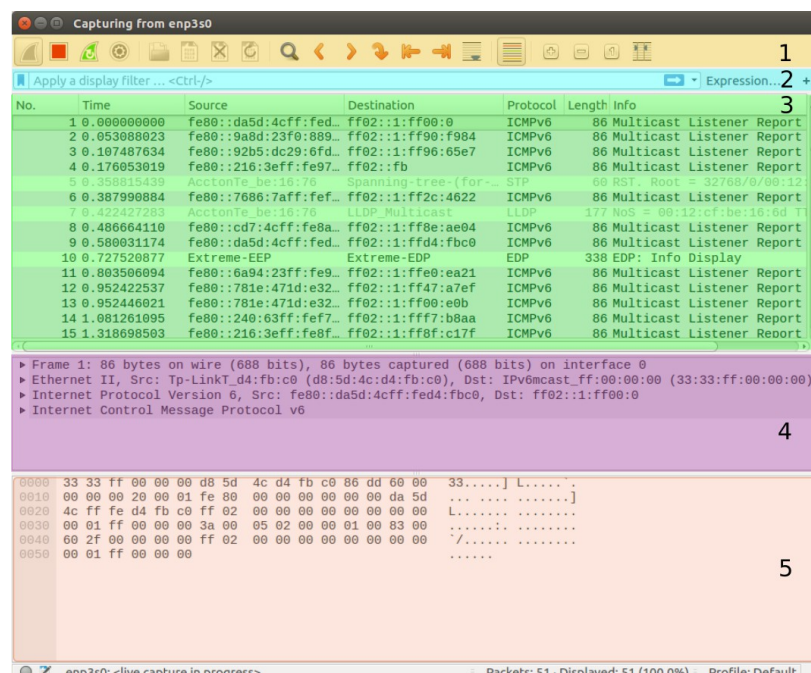
a) Para iniciar o Wireshark, acesse os programas (na interface gráfica), procure por Wireshark e clique no ícone da ferramenta.



b) Com Wireshark em execução, selecione uma das interfaces que deseja monitorar e realizar a captura dos dados. Nos equipamentos do laboratório, selecione a interface **eth0**.



c) A tela de captura é dividida em 5 partes, conforme ilustrado na figura a seguir: 1) barra de tarefas que possibilita parar e reiniciar a captura além de outras ações, 2) campo de inserção de filtros, 3) a tabela com os pacotes capturados, 4) descrição do pacote selecionado (em camadas) e 5) o conteúdo do pacote capturado (em hexadecimal e caracteres “visualizáveis”, se possível).



d) Agora, para utilizar o filtro, inicie a ferramenta **ping** indicando como destino `google.com`. A seguir, inicie o Wireshark, selecione a interface `eth0` e adicione no campo de filtro `icmp`. Desse modo, será possível ver todos os pacotes ICMP (*Internet Control Message Protocol*) enviados pela ferramenta **ping** (i.e., pacotes do tipo ICMP *Echo request*) e as respostas correspondentes retornadas por `google.com`. (i.e., pacotes do tipo ICMP *Echo reply*). Observe com atenção os endereços IP de **origem** (*source*) e de **destino** (*destination*) para identificar o sentido em que trafegam os pacotes.

The screenshot shows the Wireshark interface with the filter `icmp` applied. The packet list displays 29 packets of ICMP Echo (ping) requests and replies. The packet details pane shows the structure of a selected ICMP Echo (ping) request packet.

No.	Time	Source	Destination	Protocol	Length	Info
8679	1191.5188876...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8708	1192.3488419...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8709	1192.5102241...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8711	1193.3501851...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8712	1193.5104072...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8714	1194.3513964...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8715	1194.5193514...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8742	1195.3523303...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8743	1195.5172173...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8745	1196.3541992...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8746	1196.5196535...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8755	1197.3556208...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8756	1197.5192925...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...
8784	1198.3572767...	10.16.0.16	216.58.192.46	ICMP	98	Echo (ping) request id=...
8785	1198.5091828...	216.58.192.46	10.16.0.16	ICMP	98	Echo (ping) reply id=...

Frame 3264: 94 bytes on wire (752 bits), 94 bytes captured (752 bits) on interface 0
 Ethernet II, Src: Tp-LinkT_d4:fb:c0 (d8:5d:4c:d4:fb:c0), Dst: Dell_f5:f0:b4 (90:b1:1c:f5:f0:b4)
 Internet Protocol Version 4, Src: 10.16.0.1, Dst: 10.16.0.16
 Internet Control Message Protocol

```

0000  90 b1 1c f5 f0 b4 d8 5d 4c d4 fb c0 08 00 45 c0  .....] L....E.
0010  00 50 78 c6 00 00 40 01 ec f6 0a 10 00 01 0a 10  .Px...@. ....
0020  00 10 05 01 05 45 0a 10 58 02 45 00 00 34 68 ae  ....E.. X.E..4h.
0030  40 00 3f 06 66 e4 0a 10 00 10 0a 10 58 02 94 48  @.?.f... ..X..H
0040  00 16 b1 66 6e a6 21 17 7d 56 80 10 00 e5 d2 36  ...fn.!. }V....6
0050  00 00 01 01 08 0a 47 2b 80 9d 27 e2 f4 eb  .....G+ ..'...
  
```

Internet Control Message Protocol: Protocol Packets: 8786 · Displayed: 29 (0.3%) Profile: Default

e) Modifique o filtro anterior substitua o protocolo ICMP pelo IP de destino do `google.com`. Consulte a documentação do Wireshark disponível em wiki.wireshark.org/DisplayFilters, em caso de dúvida.

f) Para realizar esse exercício é necessário entender como a ferramenta Traceroute funciona. Para descobrir os equipamentos que estão no caminho até o equipamento de destino, Traceroute utiliza uma envia pacotes (UDP – *User Datagram Protocol*), manipulando o valor do TTL desses pacotes. Por padrão, inicialmente, são enviados 3 pacotes com o TTL igual a 1. O primeiro roteador decrementa o TTL, verifica que é zero e informa à origem que o pacote não pode ser reencaminhado porque o TTL expirou. Para tanto, utiliza uma mensagem ICMP *Time-to-Live Exceeded* (Type 11). A seguir, a origem gera 3 novos pacotes com TTL igual a 2, os quais atravessam o primeiro roteador, mas não o segundo, o qual informa à origem sobre a expiração do TTL. Esse processo continua até que o valor do TTL dos pacotes gerados pela origem é suficiente alto para alcançar o destino sem ser descartado ao longo do caminho.

Agora que já sabe como o Traceroute funciona, use o Wireshark com filtro adequado para capturar todos os pacotes utilizados pelo Traceroute. Com realizado em um exercício anterior, utilize o `google.com` como destino, i.e., **traceroute google.com**.

g) Execute os comandos realizados no exercício “c” ou “d” sobre a ferramenta **netcat**, mas em combinação o Wireshark. Configurando o filtro adequadamente, deve ser possível ler (no Wireshark) o texto enviado entre às duas aplicações.