# Курбатов Ярослав, РПО МКН

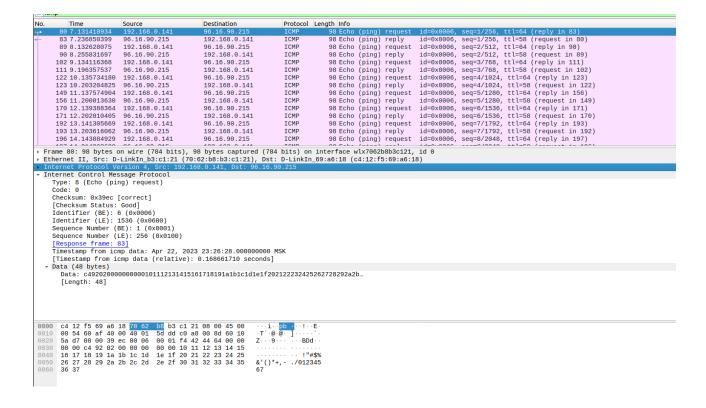
# 1. Wireshark: ICMP

в unix в ping опция для задания кол-ва запросов -n

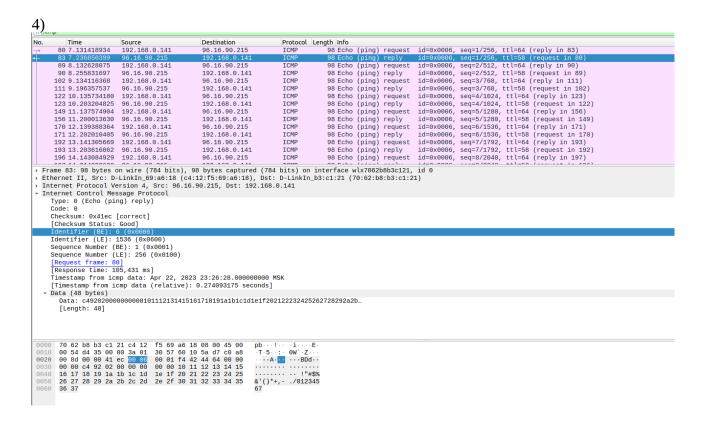
## 1. Ping

/rt Londs												
No		Time	Source	Destination	Protocol							
→		7.131418934	192.168.0.141	96.16.90.215	ICMP				request		seq=1/256,	
4	83	7.236850399	96.16.90.215	192.168.0.141	ICMP	98	Echo	(ping)	reply	id=0x0006,	seq=1/256,	ttl
	89	8.132628075	192.168.0.141	96.16.90.215	ICMP	98	Echo	(ping)	request	id=0x0006,	seq=2/512,	ttl
	90	8.255831697	96.16.90.215	192.168.0.141	ICMP	98	Echo	(ping)	reply	id=0x0006,	seq=2/512,	ttl
	102	9.134116368	192.168.0.141	96.16.90.215	ICMP	98	Echo	(ping)	request		seq=3/768,	
		9.196357537	96.16.90.215	192.168.0.141	ICMP			(1 5)	reply		seq=3/768,	
	122	10.135734180	192.168.0.141	96.16.90.215	ICMP	98	Echo	(ping)	request	id=0x0006,	seq=4/1024	, tt
	123	10.203204825	96.16.90.215	192.168.0.141	ICMP	98	Echo	(ping)	reply	id=0x0006,	seq=4/1024	, tt
	149	11.137574904	192.168.0.141	96.16.90.215	ICMP	98	Echo	(ping)	request	id=0x0006,	seq=5/1280	, tt
	156	11.200013630	96.16.90.215	192.168.0.141	ICMP	98	Echo	(ping)	reply		seq=5/1280	
	170	12.139388364	192.168.0.141	96.16.90.215	ICMP	98	Echo	(ping)	request	id=0x0006,	seq=6/1536	, tt
	171	12.202010405	96.16.90.215	192.168.0.141	ICMP	98	Echo	(ping)	reply	id=0x0006,	seq=6/1536	, tt
	192	13.141305669	192.168.0.141	96.16.90.215	ICMP	98	Echo	(ping)	request	id=0x0006,	seq=7/1792	, tt
	193	13.203616062	96.16.90.215	192.168.0.141	ICMP	98	Echo	(ping)	reply	id=0x0006,	seq=7/1792	, tt
			192.168.0.141	96.16.90.215	ICMP				request		seq=8/2048	
			00 40 00 04F	400 400 0 444	TOWN				1		0/0040	
			n wire (784 bits), 98									
Figure 1 Tipe 1 Tipe 2												
	<pre> 0101 = Header Length: 20 bytes (5) &gt; Differentiated Services Field: 0x00 (DSCP: CSO, ECN: Not-ECT)</pre>											
			vices Field. 0x00 (D3	CF. CSO, ECN. NOT-ECT	)							
	Total Length: 84 Identification: 0x60af (24751) Flags: 0x40, Don't fragment0 0000 0000 0000 = Fragment Offset: 0 Time to Live: 64											
	Protocol: ICMP (1) Header Checksum: AvEddd [velidation disabled]											
	Header Checksum: 0x5ddd [validation disabled]											
	[Header checksum status: Unverified] Source Address: 192.168.0.141											
		Source Address: 192.108.0.141 Destination Address: 96.16.90.215										
I	Desti	inacion Addres	68: 90.10.90.215									

- 1) IP-адрес моего хоста -- 192.168.0.141. IP-адрес назначения 96.16.90.215 (см. описание IP-пакета)
- 2) Протокол ICMP (хотя и работает поверх протокола IP) по сути находится на 3-ем уровне модели ISO/OSI, под транспортным уровнем, поэтому понятия порта тут нет.



ICMP-тип первого запроса — echo (значение 8 в поле Туре). Значение подтипа — 0 (единственный возможный при type=8). Также в пакете есть поле identifier, sequence number, checksum, timestamp. Размер identifier, checksum, SN-2 байта.

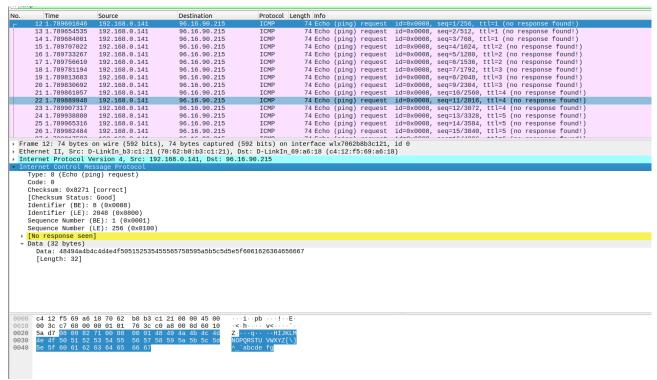


IC < P-тип первого ответа — echo reply (значение 0 в поле Туре). Значение подтипа — 0 (единственный возможный при type=0). Пакет имеет те же поля, что и запрос. SN, checksum, identifier занимают 2 байта.

#### 2. Traceroute

traceroute -I nsa.gov (-I чтобы использовать ICMP)

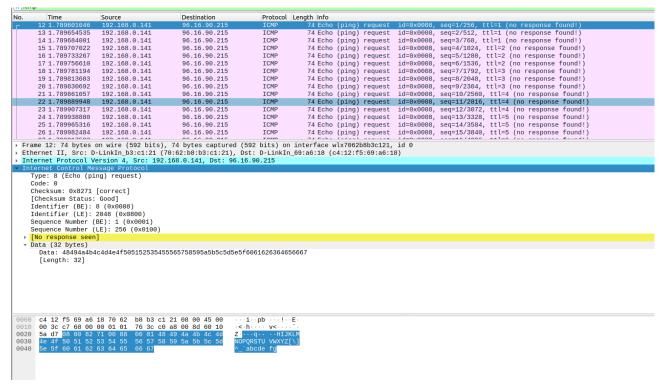
1)



можно видеть, что сам ICMP-пакет ничем не отличается от ICMP-пакета из первого задания. Есть различие на уровне IP-протокола (TTL=1).

2)

Ответ с сообщением об ошибке:



видно, что тип ICMP-пакета -1 – TTL Exceeded. Кроме того, пакет содержит данные ICMP-запроса. Также среди полей пакета – только подтип и checksum.

3)

последние 3 пакета, которые пришли пакета на исходный (мой) хост, это ICMP echo reply от искомого хоста (nsa.gov, 184.84.25.195). Собственно, это последние пакеты, потому что ICMP-пакеты дошли до искомого хоста и он ответил на 3 наших запроса.

4)

```
-s-compnet/homeworks/09/task_c$ traceroute -I nsa.gov
traceroute to nsa.gov (184.84.25.195), 30 hops max, 60 byte packets

1 192.168.1.1 (192.168.1.1) 2.046 ms 2.458 ms 2.647 ms

2 ip.178-70-210-1.avangarddsl.ru (178.70.210.1) 5.426 ms 5.967 m

3 so-2-1-0-s16.M120-1-PSKV.nwtelecom.ru (212.48.194.38) 7.359 ms
                                                                                                                                   5.967 ms 6.645 ms
.359 ms 7.691 ms 7.938 ms
                                                                                                                             7.359 ms
      109.172.24.67 (109.172.24.67) 19.658 ms 19.871 ms 20.327 ms ae53.edge4.Stockholm2.Level3.net (213.249.107.129) 20.104 ms * ae2.3204.edge7.Amsterdam1.level3.net (4.69.162.181) 46.017 ms
                                                                                                                                                     21.072 ms
      ae2.3204.edge7.Amsterdam1.level3.net (4.69.102.181) 40.01.
212.72.47.190 (212.72.47.190) 86.589 ms 86.851 ms 87.191
ae2.r02.ams01.icn.netarch.akamai.com (23.210.55.40) 45.504
ae8.r02.par01.icn.netarch.akamai.com (95.100.192.225) 65.4
ae5.r02.mad01.icn.netarch.akamai.com (95.100.192.119) 80.0
ae2.r02.mad01.ien.netarch.akamai.com (23.210.58.41) 103.30
                                                                                                                         87.191 ms
45.504 ms
                                                                                                                                                 48.873 ms
                                                                                                                                                                          50.964 ms
                                                                                                                             65.498 ms 69.021 ms
                                                                                                                                                                              76.434 ms
                                                                                                                             80.013 ms
                                                                                                                                                    80.948 ms
103.514 ms
                                                                                                                                                                               81.292 ms
                                                                                                                         103.306 ms 103.
441 ms 72.833 ms
                                                                                                                                                                              103.684 ms
       as20940-gw-max.cw.net (195.2.24.118)
       a184-84-25-195.deploy.static.akamaitechnologies.com (184.84.25.195) 74.713 ms 76.723 ms 76.079 ms
```

относительно большная задержка между ae53.edge4.Stockholm2.Level3.net (213.249.107.129) и ae2.3204.edge7.Amsterdam1.level3.net (213.249.107.129) (видимо, Стокгольм и Амстердам)

#### Программирование

Ruby MRI ~> 3.0.0

Задание А

ruby task\_a/task\_a.rb <network-interface-identifier>

## Задание В

ruby task\_b/task\_b.rb -ip-address=<...> --ports-range=<...> --mode=<...>

- --ip-address: IP-адрес для скана
- --ports-range: Отрезок портов для скана в формате start:end
- --mode: open или busy, busy занятые порты, open свободные (по поводу работы с UDP см.

Комментарии в коде)

```
(ktvy_venv) vtralpraxtsgprtnary:-/Bocuments/spbu/2023-sprtng/networks/spbu-masters-compnet/homeworks/09/task_b$ ruby task_b.rb --tp-address 127.0.0.1 --ports-range 1024:65535 --mode=busy TCP scan started 5432/TCP 6379/TCP 6379/TCP 7070/TCP 8000/TCP 9000/TCP 90000/TCP 9000/TCP 90000/TCP 9000/TCP 9000/TCP 9000/TCP 9000/TCP 9000/TCP 9000/TCP 9000
```

# Задание С

python 3.10.0

зависимости: task\_c/requirements.txt (сгенерировано pip freeze)

python task\_c/broad.py -- -port <port> --broadcast-period <float-seconds> --inactive-peer-period <float-seconds>

#### Параметры:

port: число, порт на который биндится приложение

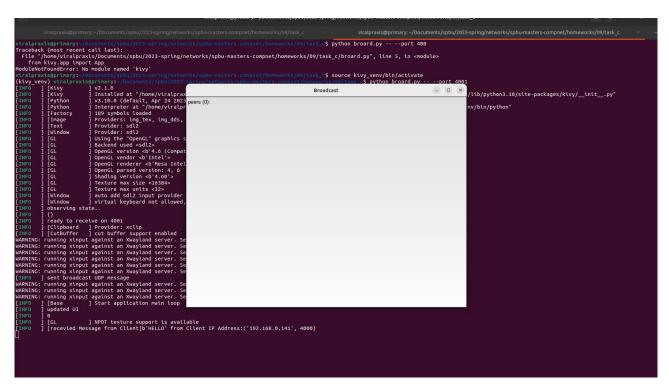
broadcast-period: float, кол-во секунд, раз в которое делается бродкаст

inactive-peer-period: float, кол-во секунд такое, что если в течение большего периода не

получены UDP-пакеты от хоста, он считается неактивным

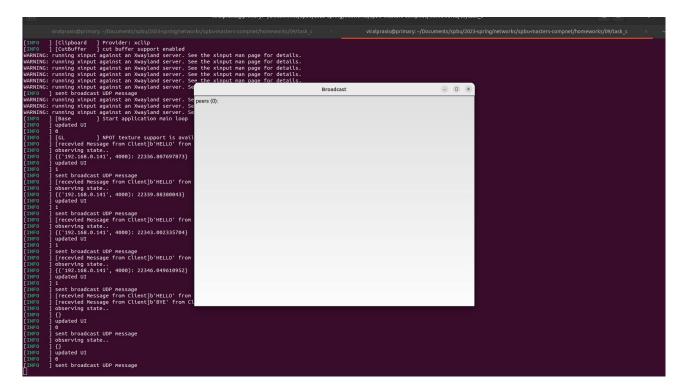
Запустим инстанс приложения на порту 4000 (python3 broard.py -- --port 4000):

Запустим еще один инстанс на порту 4001 (python3 broard.py -- --port 4001):



Через какое-то время второе приложение получит broadcast-пакет от первого и добавит его в список:

Теперь остановим первое приложение. Перед завершением работы оно пошлет пакет, сигнализирующий о завершении работы. Второе приложение пометит первое как неактивное и уберет его из UI:



То же самое произодет, если первое приложение не будет отсылать пакет перед остановкой, через ~12 секунд (дефолтное значение) второе приложение пометит первое как неактивное.