

1. При помощи анализатора протоколов Wireshark исследовать и описать работу команды ping.

Например: ping 8.8.8.8;

Команда ping в Windows (и в командной строке, CMD) используется для проверки сетевого соединения между вашим компьютером и другим устройством (сервером или другим компьютером). Она отправляет небольшие пакеты данных (эхо-запросы) и измеряет время, которое они затратили на дорогу туда и обратно. Это позволяет определить, есть ли связь между компьютерами, насколько она надежна, и какие могут быть проблемы с сетью.

Подробнее о работе команды ping:

1. 1. Отправка эхо-запросов:

Команда ping отправляет ICMP эхо-запросы на указанный IP-адрес или доменное имя.

2. 2. Измерение времени:

Отслеживается время, которое требуется для отправки запроса, его получения и возвращения ответного пакета.

3. 3. Оценка соединения:

По результатам измерения времени и количества утерянных пакетов можно судить о качестве сетевого соединения.

4. 4. Информация о соединении:

В результате работы ping вы получаете информацию о времени отклика (RTT), количестве утерянных пакетов и другие данные, полезные для диагностики сети.

Для чего используется команда ping:

- **Проверка связи:**

Можно быстро проверить, существует ли сетевое соединение с конкретным устройством.

- **Оценка производительности:**

Время отклика (ping) может быть использовано для оценки качества сетевого соединения.

- **Выявление проблем:**

Если пакеты не возвращаются или задерживаются, это может указывать на проблемы с сетью.

- **Устранение неполадок:**

Ping помогает выявить и устранить проблемы, связанные с подключением к сети.

Пример использования:

Код

```
ping google.com
```

Эта команда отправляет эхо-запросы на сервер Google и отображает информацию о времени отклика и количестве утерянных пакетов.

IP адрес 8.8.8.8 = <https://dns.google/>

Ping 8.8.8.8

304	72.458246	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=27/6912, ttl=128 (reply in 305)	
305	72.480112	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=27/6912, ttl=108 (request in 304)	
306	72.742803	192.168.0.131	47.103.37.176	TCP	66 57321 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM		
307	72.973472	47.103.37.176	192.168.0.131	TCP	66 80 → 57321 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1440 SACK_PERM WS=512		
308	72.973781	192.168.0.131	47.103.37.176	TCP	54 57321 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0		
309	72.975228	192.168.0.131	47.103.37.176	TCP	54 57321 → 80 [FIN, ACK] Seq=1 Ack=1 Win=65280 Len=0		
310	72.979509	192.168.0.131	10.1.1.81	TCP	66 57322 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM		
311	73.206441	47.103.37.176	192.168.0.131	TCP	60 80 → 57321 [FIN, ACK] Seq=1 Ack=2 Win=29696 Len=0		
312	73.206825	192.168.0.131	47.103.37.176	TCP	54 57321 → 80 [ACK] Seq=2 Ack=2 Win=65280 Len=0		
313	73.353526	192.168.0.131	149.154.167.99	TLSv1.2	237 Application Data		
314	73.395491	149.154.167.99	192.168.0.131	TLSv1.2	167 Application Data		
315	73.446909	192.168.0.131	149.154.167.99	TCP	54 49899 → 443 [ACK] Seq=5372 Ack=4950 Win=254 Len=0		
316	73.479268	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=28/7168, ttl=128 (reply in 317)	
317	73.501223	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=28/7168, ttl=108 (request in 316)	
318	73.908811	192.168.0.131	10.1.1.81	TCP	66 [TCP Retransmission] 57322 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM		
319	74.494705	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=29/7424, ttl=128 (reply in 320)	
320	74.516719	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=29/7424, ttl=108 (request in 319)	
321	75.510903	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=30/7680, ttl=128 (reply in 322)	
322	75.532777	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=30/7680, ttl=108 (request in 321)	
323	75.937956	192.168.0.1	192.168.0.255	UDP	368 43283 → 20002 Len=326		
Frame 304: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{BDE25619-C0F9-4160-8231-761A128802E3}, id 0							
Ethernet II, Src: MicroStarINT_9b:7d:13 (d8:43:ae:9b:7d:13), Dst: TpLinkTechno_24:e0:d7 (84:d8:1b:24:e0:d7)							
Internet Protocol Version 4, Src: 192.168.0.131, Dst: 8.8.8.8							
Internet Control Message Protocol							
Type: 8 (Echo (ping) request)							
Code: 0							
Checksum: 0x4d40 [correct]							
[Checksum Status: Good]							
Identifier (BE): 1 (0x0001)							
Identifier (LE): 256 (0x0100)							
Sequence Number (BE): 27 (0x001b)							
Sequence Number (LE): 6912 (0x1b00)							
[Response frame: 305]							
Data (32 bytes)							
Data: 6162636465666768696a6b6c6d6e6f7071727374757677616263646566676869							
[Length: 32]							

305	72.480112	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=27/6912, ttl=108 (request in 304)	
306	72.742803	192.168.0.131	47.103.37.176	TCP	66 57321 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM		
307	72.973472	47.103.37.176	192.168.0.131	TCP	66 80 → 57321 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1440 SACK_PERM WS=512		
308	72.973781	192.168.0.131	47.103.37.176	TCP	54 57321 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0		
309	72.975228	192.168.0.131	47.103.37.176	TCP	54 57321 → 80 [FIN, ACK] Seq=1 Ack=1 Win=65280 Len=0		
310	72.979509	192.168.0.131	10.1.1.81	TCP	66 57322 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM		
311	73.206441	47.103.37.176	192.168.0.131	TCP	60 80 → 57321 [FIN, ACK] Seq=1 Ack=2 Win=29696 Len=0		
312	73.206825	192.168.0.131	47.103.37.176	TCP	54 57321 → 80 [ACK] Seq=2 Ack=2 Win=65280 Len=0		
313	73.353526	192.168.0.131	149.154.167.99	TLSv1.2	237 Application Data		
314	73.395491	149.154.167.99	192.168.0.131	TLSv1.2	167 Application Data		
315	73.446909	192.168.0.131	149.154.167.99	TCP	54 49899 → 443 [ACK] Seq=5372 Ack=4950 Win=254 Len=0		
316	73.479268	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=28/7168, ttl=128 (reply in 317)	
317	73.501223	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=28/7168, ttl=108 (request in 316)	
318	73.908811	192.168.0.131	10.1.1.81	TCP	66 [TCP Retransmission] 57322 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM		
319	74.494705	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=29/7424, ttl=128 (reply in 320)	
320	74.516719	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=29/7424, ttl=108 (request in 319)	
321	75.510903	192.168.0.131	8.8.8.8	ICMP	74 Echo (ping) request	id=0x0001, seq=30/7680, ttl=128 (reply in 322)	
322	75.532777	8.8.8.8	192.168.0.131	ICMP	74 Echo (ping) reply	id=0x0001, seq=30/7680, ttl=108 (request in 321)	
323	75.937956	192.168.0.1	192.168.0.255	UDP	368 43283 → 20002 Len=326		
Frame 305: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{BDE25619-C0F9-4160-8231-761A128802E3}, id 0							
Ethernet II, Src: TpLinkTechno_24:e0:d7 (84:d8:1b:24:e0:d7), Dst: MicroStarINT_9b:7d:13 (d8:43:ae:9b:7d:13)							
Internet Protocol Version 4, Src: 8.8.8.8, Dst: 192.168.0.131							
Internet Control Message Protocol							
Type: 0 (Echo (ping) reply)							
Code: 0							
Checksum: 0x5540 [correct]							
[Checksum Status: Good]							
Identifier (BE): 1 (0x0001)							
Identifier (LE): 256 (0x0100)							
Sequence Number (BE): 27 (0x001b)							
Sequence Number (LE): 6912 (0x1b00)							
[Request frame: 304]							
[Response time: 21,066 ms]							
Data (32 bytes)							
Data: 6162636465666768696a6b6c6d6e6f7071727374757677616263646566676869							
[Length: 32]							

C:\Users\Admin>ping 8.8.8.8

Обмен пакетами с 8.8.8.8 по с 32 байтами данных:
Ответ от 8.8.8.8: число байт=32 время=21мс TTL=108
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=108
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=108
Ответ от 8.8.8.8: число байт=32 время=22мс TTL=108

Статистика Ping для 8.8.8.8:

Пакетов: отправлено = 4, получено = 4, потеряно = 0
(0% потерь)

Приблизительное время приема-передачи в мс:

Минимальное = 21мсек, Максимальное = 22 мсек, Среднее = 21 мсек

C:\Users\Admin>

2. При помощи анализатора протоколов Wireshark исследовать и описать работу команды tracert

Например: `tracert -d -w 1 8.8.8.8;`

Команда tracert в командной строке Windows (cmd) используется для трассировки пути, по которому пакеты данных проходят от вашего компьютера до указанного адреса (IP-адрес или доменное имя). Она отображает все промежуточные маршрутизаторы, которые пакеты пересекают на пути к целевому хосту.

Как работает tracert:

1. 1. Отправка пакетов ICMP:

tracert отправляет серию пакетов ICMP (Internet Control Message Protocol) с постепенно увеличивающимся значением Time To Live (TTL).

2. 2. TTL и время жизни пакета:

TTL – это поле в заголовке пакета, которое определяет, сколько хопов (маршрутизаторов) пакет может пройти до того, как будет удален.

3. 3. ICMP "Time Exceeded" сообщения:

Каждый маршрутизатор на пути уменьшает значение TTL на 1. Когда TTL достигает нуля, маршрутизатор отправляет ICMP "Time Exceeded" сообщение обратно на источник.

4. 4. Определение маршрута:

tracert использует эти ICMP "Time Exceeded" сообщения, чтобы определить, какие маршрутизаторы находятся на пути к целевому хосту.

5. 5. Вывод информации:

tracert отображает список всех маршрутизаторов, через которые проходят пакеты, а также время, затраченное на прохождение пакета через каждый маршрутизатор.

Зачем нужна tracert:

- **Диагностика сетевых проблем:**

tracert помогает выявлять проблемы в сети, например, если какие-то маршрутизаторы не отвечают или пакеты теряются.

- **Понимание сетевого маршрута:**

tracert позволяет узнать, какой маршрут выбирается для передачи данных от вашего компьютера до удаленного хоста.

- **Определение задержек:**

tracert показывает, сколько времени занимает прохождение пакета через каждый маршрутизатор, что может помочь выявить источники задержек.

Пример использования:

Код

```
tracert www.google.com
```

Этот пример покажет трассировку пути к сайту google.com.

Существует несколько параметров командной строки, которые можно использовать с помощью команды TRACERT, несмотря на то, что параметры не являются обычно требуются для стандартных неполадок. В следующем примере синтаксис команды показывает все возможные варианты:

Tracert -d -h **максЧисло** -j **списокУзлов** -w **Таймаут target_host**

Что делают параметры:

-d Указывает, что адреса не следует преобразовывать в имена хостов

-h Указывает максимальное количество переходов для поиска цели.

-j Указывает свободный исходный маршрут по списку хостов

-w Ожидает указанное время ожидания в миллисекундах для каждого ответа

target_host Указывает имя или IP-адрес целевого хоста

tracert -d -w 1 8.8.8.8

18393	1138.988862	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=31/7936, ttl=1 (no response found!)
18394	1138.989359	192.168.0.1	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18395	1138.989933	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=32/8192, ttl=1 (no response found!)
18396	1138.990219	192.168.0.1	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18397	1138.990780	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=33/8448, ttl=1 (no response found!)
18398	1138.991047	192.168.0.1	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18399	1139.478359	192.168.0.131	255.255.255.255	UDP	406 56348 → 7117 Len=364
18400	1139.478504	192.168.0.131	255.255.255.255	UDP	406 56349 → 7117 Len=364
18401	1139.478638	192.168.0.131	255.255.255.255	UDP	406 56350 → 7117 Len=364
18402	1139.478747	192.168.0.131	255.255.255.255	UDP	406 56351 → 7117 Len=364
18403	1139.478848	192.168.0.131	255.255.255.255	UDP	406 56352 → 7117 Len=364
18404	1140.009931	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=34/8704, ttl=2 (no response found!)
18405	1140.012597	100.119.0.1	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18406	1140.013962	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=35/8960, ttl=2 (no response found!)
18407	1140.016249	100.119.0.1	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18408	1140.017327	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=36/9216, ttl=2 (no response found!)
18409	1140.019704	100.119.0.1	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18410	1140.031415	192.168.0.131	149.154.167.99	TLSv1.2	237 Application Data
18411	1140.074657	149.154.167.99	192.168.0.131	TLSv1.2	167 Application Data
18412	1140.924780	192.168.0.131	149.154.167.99	TCP	54 49899 → 443 [ACK] Seq=80283 Ack=73659 Win=251 Len=0
18413	1141.023138	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=37/9472, ttl=3 (no response found!)
18414	1141.085364	192.168.0.131	31.13.72.52	TCP	123 49877 → 5222 [PSH, ACK] Seq=4054 Ack=3746 Win=508 Len=69 [TCP PDU reassembled in 19739]
18415	1141.106696	31.13.72.52	192.168.0.131	TCP	60 5222 → 49877 [ACK] Seq=3746 Ack=4123 Win=380 Len=0
18416	1141.117719	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=38/9728, ttl=3 (no response found!)
18417	1141.210601	31.13.72.52	192.168.0.131	TCP	125 5222 → 49877 [PSH, ACK] Seq=3746 Ack=4123 Win=380 Len=71 [TCP PDU reassembled in 19641]
18418	1141.258371	192.168.0.131	31.13.72.52	TCP	54 49877 → 5222 [ACK] Seq=4123 Ack=3817 Win=508 Len=0
18419	1141.609475	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=39/9984, ttl=3 (no response found!)
18420	1142.123807	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=40/10240, ttl=4 (no response found!)
18421	1142.131716	85.21.224.191	192.168.0.131	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
18422	1142.133999	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=41/10496, ttl=4 (no response found!)
18423	1142.139806	85.21.224.191	192.168.0.131	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
18424	1142.142055	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=42/10752, ttl=4 (no response found!)
18425	1142.145550	85.21.224.191	192.168.0.131	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
18426	1142.412082	HP 85145:67	Broadcast	LLC	60 S.P. func=RNR, N(R)=64; DSAP NULL LSAP Individual, SSAP NULL LSAP Command
18427	1143.151453	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=43/11008, ttl=5 (no response found!)
18428	1143.613000	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=44/11264, ttl=5 (no response found!)
18429	1143.671858	192.178.241.146	192.168.0.131	ICMP	134 Time-to-live exceeded (Time to live exceeded in transit)
18430	1143.706561	192.168.0.131	208.115.231.190	TCP	55 [TCP Keep-Alive] 49714 → 443 [ACK] Seq=1 Ack=1 Win=255 Len=1
18431	1143.708005	208.115.231.190	192.168.0.131	TCP	60 [TCP Keep-Alive] 443 → 49714 [ACK] Seq=0 Ack=2 Win=501 Len=0
18432	1143.708094	192.168.0.131	208.115.231.190	TCP	54 [TCP Keep-Alive ACK] 49714 → 443 [ACK] Seq=2 Ack=1 Win=255 Len=0
18433	1143.749822	208.115.231.190	192.168.0.131	TCP	66 [TCP Dup ACK 40#19] 443 → 49714 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
18434	1143.883921	192.168.0.131	149.154.167.99	TLSv1.2	237 Application Data
18435	1143.928545	149.154.167.99	192.168.0.131	TLSv1.2	167 Application Data
18436	1143.976222	192.168.0.131	149.154.167.99	TCP	54 49899 → 443 [ACK] Seq=80466 Ack=73772 Win=255 Len=0
18437	1144.092552	192.168.0.131	149.154.167.99	TLSv1.2	253 Application Data
18438	1144.105024	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=45/11520, ttl=5 (no response found!)
18439	1144.152341	149.154.167.99	192.168.0.131	TLSv1.2	585 Application Data
18440	1144.198192	192.168.0.131	149.154.167.99	TCP	54 49899 → 443 [ACK] Seq=80665 Ack=74303 Win=253 Len=0
18441	1144.618162	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=46/11776, ttl=6 (no response found!)
18442	1144.621801	192.178.241.146	192.168.0.131	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
18443	1144.623969	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=47/12032, ttl=6 (no response found!)
18444	1144.627371	192.178.241.146	192.168.0.131	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
18445	1144.628771	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=48/12288, ttl=6 (no response found!)
18446	1144.631867	192.178.241.146	192.168.0.131	ICMP	70 Time-to-live exceeded (Time to live exceeded in transit)
18447	1145.646993	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=49/12544, ttl=7 (no response found!)
18448	1146.110647	192.168.0.131	8.8.8.8	ICMP	106 Echo (ping) request id=0x0001, seq=50/12800, ttl=7 (no response found!)

```
C:\Users\Admin>tracert -d -w 1 8.8.8.8
```

Трассировка маршрута к 8.8.8.8 с максимальным числом прыжков 30

1	<1 ms	<1 ms	<1 ms	192.168.0.1
2	2 ms	2 ms	2 ms	100.119.0.1
3	*	*	*	Превышен интервал ожидания для запроса
4	8 ms	6 ms	3 ms	85.21.224.191
5	*	*	*	Превышен интервал ожидания для запроса
6	3 ms	3 ms	3 ms	192.178.241.146
7	*	22 ms	22 ms	209.85.249.158
8	22 ms	22 ms	22 ms	216.239.43.20
9	22 ms	22 ms	21 ms	142.250.56.13
10	*	*	*	Превышен интервал ожидания для запроса
11	*	*	*	Превышен интервал ожидания для запроса
12	*	*	*	Превышен интервал ожидания для запроса
13	*	*	*	Превышен интервал ожидания для запроса
14	*	*	*	Превышен интервал ожидания для запроса
15	*	*	*	Превышен интервал ожидания для запроса
16	*	*	*	Превышен интервал ожидания для запроса
17	*	*	*	Превышен интервал ожидания для запроса
18	*	*	*	Превышен интервал ожидания для запроса
19	22 ms	22 ms	22 ms	8.8.8.8

Трассировка завершена.

```
C:\Users\Admin>
```

3. Исследовать ключ -r команды ping. Например: ping -r 9 yandex.ru;

/r <count> Указывает параметр "Маршрут записи" в заголовке IP-адреса для записи пути, принятого сообщением запроса эхо и соответствующим сообщением ответа на эхо (доступно только

в IPv4). Каждый прыжок в пути использует запись в параметре "**Маршрут** записи". Если это возможно, укажите *число* , равное или больше количества прыжков между источником и назначением. *Число* должно быть не менее 1 и не более 9.

```
C:\Users\Admin>ping -r 9 yandex.ru

Обмен пакетами с yandex.ru [5.255.255.77] с 32 байтами данных:
Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.

Статистика Ping для 5.255.255.77:
    Пакетов: отправлено = 4, получено = 0, потеряно = 4
    (100% потеря)

C:\Users\Admin>
```

4. ***Построить полный маршрут до какого-либо узла в Интернете;

```
C:\Users\Admin>tracert -h 45 8.8.8.8
```

Трассировка маршрута к dns.google [8.8.8.8]
с максимальным числом прыжков 45:

1	<1 ms	<1 ms	<1 ms	192.168.0.1
2	3 ms	2 ms	2 ms	100.119.0.1
3	*	*	*	Превышен интервал ожидания для запроса
.				
4	2 ms	2 ms	2 ms	85.21.224.191
5	*	*	*	Превышен интервал ожидания для запроса
.				
6	3 ms	3 ms	3 ms	192.178.241.146
7	23 ms	22 ms	22 ms	209.85.249.158
8	19 ms	18 ms	37 ms	216.239.43.20
9	22 ms	21 ms	21 ms	142.250.56.13
10	*	*	*	Превышен интервал ожидания для запроса
.				
11	*	*	*	Превышен интервал ожидания для запроса
.				
12	*	*	*	Превышен интервал ожидания для запроса
.				
13	*	*	*	Превышен интервал ожидания для запроса
.				
14	*	*	*	Превышен интервал ожидания для запроса
.				
15	*	*	*	Превышен интервал ожидания для запроса
.				
16	*	*	*	Превышен интервал ожидания для запроса
.				
17	*	*	*	Превышен интервал ожидания для запроса
.				
18	*	*	*	Превышен интервал ожидания для запроса
.				
19	21 ms	21 ms	21 ms	dns.google [8.8.8.8]

Трассировка завершена.

IP адреса которые определились, разве они не являются маршрутом?