



# Simulação de ataque DDoS através da ferramenta CORE network emulator

Clarel Spies  
Luiza Rabuski

*Universidade de Santa Cruz do Sul - UNISC*

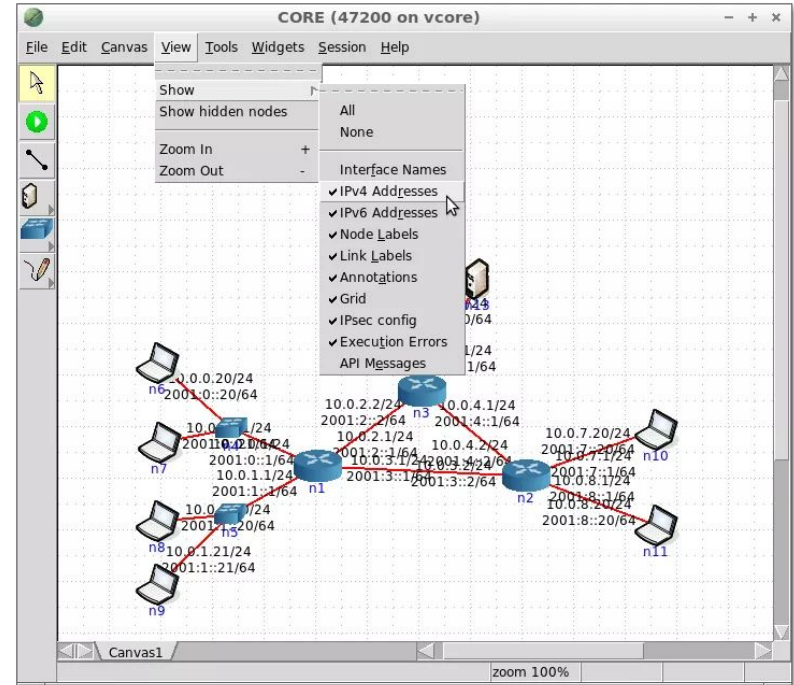
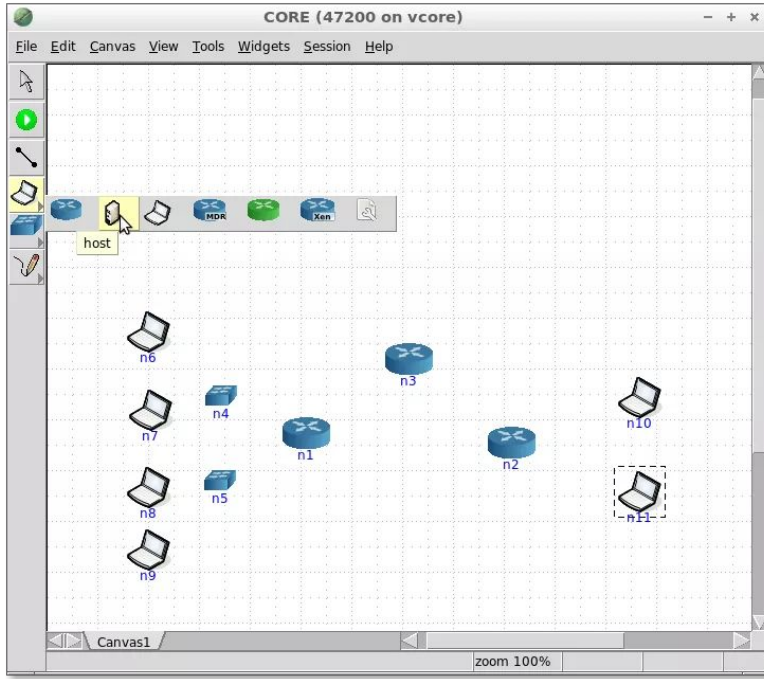
# Roteiro

- Introdução ao CORE
- Arquitetura CORE
- DDoS
- Tipos de Ataque
- Simulação de ataque DDoS Ping Flood
- Referências

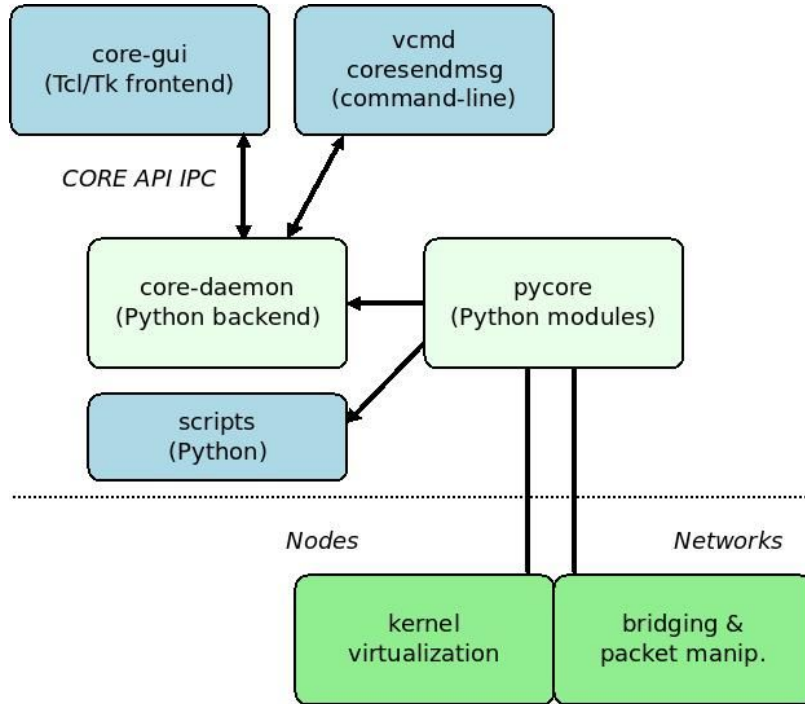
# Common Open Research Emulator (CORE)

- Permite emular redes de computadores em uma ou várias máquinas.
- Representação real de rede de computadores.
- Interface gráfica GUI.
- Sistema é modular.
- Dois modos de operação da ferramenta: Editar e Executar.

# GUI



# Arquitetura



Core-daemon (backend) administra sessões de emulação.

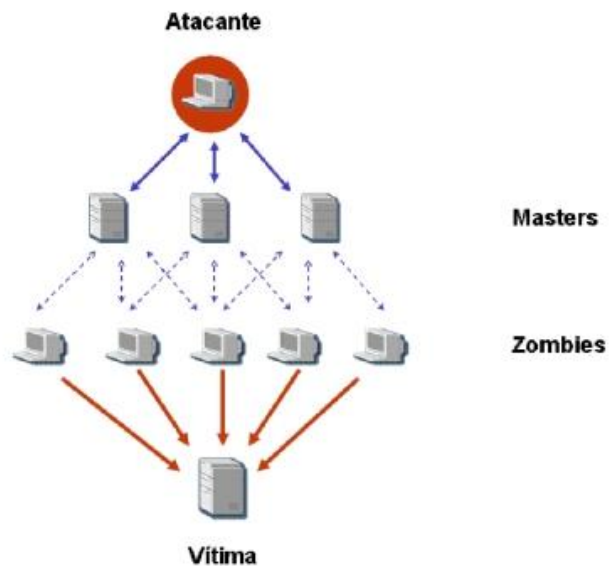
Constrói redes emuladas usando o componente kernel virtualization para nodos virtuais e o componente bridging and packet manipulation para redes virtuais.

O componente core-daemon é controlado através da interface gráfica do usuário, o CORE-GUI (frontend).

Utiliza módulos Python.

# Distributed Denial of Service (DDoS)

É uma tentativa de fazer com que um serviço online fique indisponível.



# Tipos de Ataque

- SYN Flood

O objetivo é consumir os recursos de rede e processamento da vítima enviando mais mensagens do que o servidor pode suportar.

Estas mensagens possuem pacotes SYN que requisitam a abertura de uma conexão com o alvo.

O servidor aceita a conexão e fica ocioso esperando por uma resposta do cliente confirmando que a conexão foi estabelecida.

# Tipos de Ataque

- Ping Flood

O objetivo é inundar a rede do servidor com a intenção de sobrecarregá-la através da função Ping com mensagens do protocolo ICMP.

Através de várias máquinas executando o comando Ping para um mesmo destino, podemos sobrecarregá-lo com mensagens echo request e echo reply.

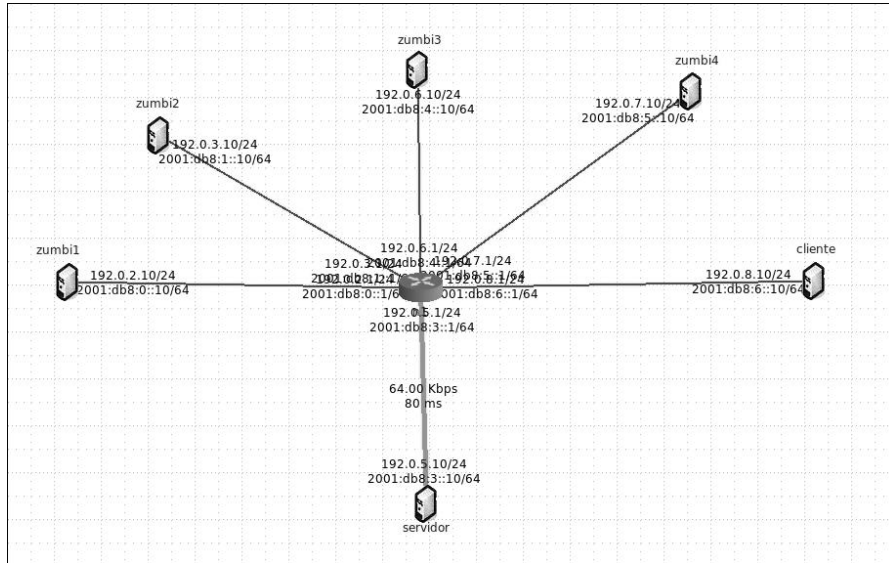
Podendo congestionar a rede e fazendo com que o servidor não consiga responder de forma eficiente ou de forma alguma.

- UDP Flood

Tem como objetivo congestionar a rede do servidor através do envio de diversos pacotes UDP para o servidor.



# Cenário para simulação



Formado por quatro máquinas chamadas de zumbis representando os atacantes.

Uma máquina cliente e uma máquina servidor.

Todos ligados a um Router e abstraindo sua rede interna para uma fácil compreensão.

A rede interna do servidor teve a capacidade limitada em 64.000 Kbps para que possamos encher o efeito do ataque DDoS em um ambiente reduzido.

# Cenário sem ataque DDoS

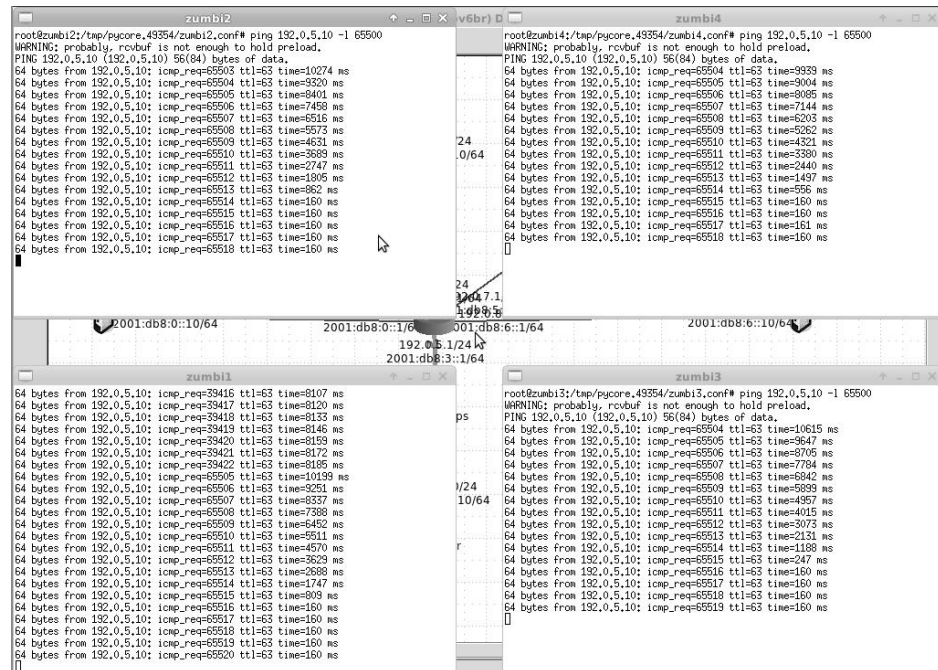
```
root@cliente:/tmp/pycore.51480/cliente.conf# ping 192.0.5.10
PING 192.0.5.10 (192.0.5.10) 56(84) bytes of data:
64 bytes from 192.0.5.10: icmp_req=1 ttl=63 time=320 ms
64 bytes from 192.0.5.10: icmp_req=2 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=3 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=4 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=5 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=6 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=7 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=8 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=9 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=10 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=11 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=12 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=13 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=14 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=15 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=16 ttl=63 time=160 ms
^C
--- 192.0.5.10 ping statistics ---
16 packets transmitted, 16 received, 0% packet loss, time 15019ms
rtt min/avg/max/mdev = 160,235/170,298/320,512/38,788 ms
root@cliente:/tmp/pycore.51480/cliente.conf# █
```

Resultado do teste Ping para o servidor em ambiente sem ataque.

Não houve nenhuma perda de pacotes durante as tentativas de sucesso.

Houve 16 envios e 16 respostas com um tempo de resposta contínuo em 160ms.

# Cenário com ataque DDoS



```
zumbi2
root@zumbi2:/tmp/pycore_49354/zumbi2.conf# ping 192.0.5.10 -l 65500
WARNING: probably, rcvbuf is not enough to hold preload.
PING 192.0.5.10 (192.0.5.10) 56(84) bytes of data,
64 bytes from 192.0.5.10: icmp_req=65503 ttl=63 time=10274 ms
64 bytes from 192.0.5.10: icmp_req=65504 ttl=63 time=9320 ms
64 bytes from 192.0.5.10: icmp_req=65505 ttl=63 time=9401 ms
64 bytes from 192.0.5.10: icmp_req=65506 ttl=63 time=7458 ms
64 bytes from 192.0.5.10: icmp_req=65507 ttl=63 time=6516 ms
64 bytes from 192.0.5.10: icmp_req=65508 ttl=63 time=5573 ms
64 bytes from 192.0.5.10: icmp_req=65509 ttl=63 time=4631 ms
64 bytes from 192.0.5.10: icmp_req=65510 ttl=63 time=3689 ms
64 bytes from 192.0.5.10: icmp_req=65511 ttl=63 time=2747 ms
64 bytes from 192.0.5.10: icmp_req=65512 ttl=63 time=1896 ms
64 bytes from 192.0.5.10: icmp_req=65513 ttl=63 time=962 ms
64 bytes from 192.0.5.10: icmp_req=65514 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65515 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65516 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65517 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65518 ttl=63 time=160 ms

zumbi4
root@zumbi4:/tmp/pycore_49354/zumbi4.conf# ping 192.0.5.10 -l 65500
WARNING: probably, rcvbuf is not enough to hold preload.
PING 192.0.5.10 (192.0.5.10) 56(84) bytes of data,
64 bytes from 192.0.5.10: icmp_req=65504 ttl=63 time=9939 ms
64 bytes from 192.0.5.10: icmp_req=65505 ttl=63 time=9004 ms
64 bytes from 192.0.5.10: icmp_req=65506 ttl=63 time=8005 ms
64 bytes from 192.0.5.10: icmp_req=65507 ttl=63 time=7144 ms
64 bytes from 192.0.5.10: icmp_req=65508 ttl=63 time=6203 ms
64 bytes from 192.0.5.10: icmp_req=65509 ttl=63 time=5252 ms
64 bytes from 192.0.5.10: icmp_req=65510 ttl=63 time=4321 ms
64 bytes from 192.0.5.10: icmp_req=65511 ttl=63 time=3380 ms
64 bytes from 192.0.5.10: icmp_req=65512 ttl=63 time=2440 ms
64 bytes from 192.0.5.10: icmp_req=65513 ttl=63 time=1497 ms
64 bytes from 192.0.5.10: icmp_req=65514 ttl=63 time=556 ms
64 bytes from 192.0.5.10: icmp_req=65515 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65516 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65517 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65518 ttl=63 time=160 ms

zumbi1
64 bytes from 192.0.5.10: icmp_req=39416 ttl=63 time=8107 ms
64 bytes from 192.0.5.10: icmp_req=39417 ttl=63 time=8120 ms
64 bytes from 192.0.5.10: icmp_req=39418 ttl=63 time=8133 ms
64 bytes from 192.0.5.10: icmp_req=39419 ttl=63 time=8146 ms
64 bytes from 192.0.5.10: icmp_req=39420 ttl=63 time=8159 ms
64 bytes from 192.0.5.10: icmp_req=39421 ttl=63 time=8172 ms
64 bytes from 192.0.5.10: icmp_req=39422 ttl=63 time=8185 ms
64 bytes from 192.0.5.10: icmp_req=65505 ttl=63 time=10199 ms
64 bytes from 192.0.5.10: icmp_req=65506 ttl=63 time=9251 ms
64 bytes from 192.0.5.10: icmp_req=65507 ttl=63 time=8337 ms
64 bytes from 192.0.5.10: icmp_req=65508 ttl=63 time=7388 ms
64 bytes from 192.0.5.10: icmp_req=65509 ttl=63 time=6452 ms
64 bytes from 192.0.5.10: icmp_req=65510 ttl=63 time=5511 ms
64 bytes from 192.0.5.10: icmp_req=65511 ttl=63 time=4570 ms
64 bytes from 192.0.5.10: icmp_req=65512 ttl=63 time=3629 ms
64 bytes from 192.0.5.10: icmp_req=65513 ttl=63 time=2688 ms
64 bytes from 192.0.5.10: icmp_req=65514 ttl=63 time=1747 ms
64 bytes from 192.0.5.10: icmp_req=65515 ttl=63 time=909 ms
64 bytes from 192.0.5.10: icmp_req=65516 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65517 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65518 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65519 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65520 ttl=63 time=160 ms

zumbi3
root@zumbi3:/tmp/pycore_49354/zumbi3.conf# ping 192.0.5.10 -l 65500
WARNING: probably, rcvbuf is not enough to hold preload.
PING 192.0.5.10 (192.0.5.10) 56(84) bytes of data,
64 bytes from 192.0.5.10: icmp_req=65504 ttl=63 time=10615 ms
64 bytes from 192.0.5.10: icmp_req=65505 ttl=63 time=9647 ms
64 bytes from 192.0.5.10: icmp_req=65506 ttl=63 time=8705 ms
64 bytes from 192.0.5.10: icmp_req=65507 ttl=63 time=7784 ms
64 bytes from 192.0.5.10: icmp_req=65508 ttl=63 time=6842 ms
64 bytes from 192.0.5.10: icmp_req=65509 ttl=63 time=5899 ms
64 bytes from 192.0.5.10: icmp_req=65510 ttl=63 time=4957 ms
64 bytes from 192.0.5.10: icmp_req=65511 ttl=63 time=4015 ms
64 bytes from 192.0.5.10: icmp_req=65512 ttl=63 time=3073 ms
64 bytes from 192.0.5.10: icmp_req=65513 ttl=63 time=2131 ms
64 bytes from 192.0.5.10: icmp_req=65514 ttl=63 time=1188 ms
64 bytes from 192.0.5.10: icmp_req=65515 ttl=63 time=247 ms
64 bytes from 192.0.5.10: icmp_req=65516 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65517 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65518 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=65519 ttl=63 time=160 ms
```

O cliente tentará acessar o servidor em meio a um ataque do tipo Ping Flood realizado por 4 maquinas zumbis.

O comando executado é 'ping 192.0.5.10 -l 65500'. Onde 192.0.5.10 é o endereço do servidor e o -l 65500 representa o parâmetro preload.

Cada uma das maquinas zumbis estará enviando um comando Ping ao servidor com o parâmetro preload configurado 65500.

# Cenário com ataque DDoS

```
root@cliente:/tmp/pycore.51479/cliente.conf# ping 192.0.5.10
PING 192.0.5.10 (192.0.5.10) 56(84) bytes of data:
From 192.0.8.1 icmp_seq=132 Destination Host Unreachable
From 192.0.8.1 icmp_seq=135 Destination Host Unreachable
From 192.0.8.1 icmp_seq=138 Destination Host Unreachable
64 bytes from 192.0.5.10: icmp_req=1 ttl=63 time=12863 ms
64 bytes from 192.0.5.10: icmp_req=2 ttl=63 time=12876 ms
64 bytes from 192.0.5.10: icmp_req=3 ttl=63 time=12889 ms
64 bytes from 192.0.5.10: icmp_req=4 ttl=63 time=12902 ms
64 bytes from 192.0.5.10: icmp_req=5 ttl=63 time=12915 ms
64 bytes from 192.0.5.10: icmp_req=6 ttl=63 time=12929 ms
64 bytes from 192.0.5.10: icmp_req=7 ttl=63 time=12941 ms
64 bytes from 192.0.5.10: icmp_req=8 ttl=63 time=12954 ms
64 bytes from 192.0.5.10: icmp_req=9 ttl=63 time=12967 ms
64 bytes from 192.0.5.10: icmp_req=10 ttl=63 time=12980 ms
64 bytes from 192.0.5.10: icmp_req=11 ttl=63 time=12993 ms
64 bytes from 192.0.5.10: icmp_req=12 ttl=63 time=13006 ms
64 bytes from 192.0.5.10: icmp_req=13 ttl=63 time=13019 ms
64 bytes from 192.0.5.10: icmp_req=14 ttl=63 time=13032 ms
64 bytes from 192.0.5.10: icmp_req=15 ttl=63 time=13045 ms
64 bytes from 192.0.5.10: icmp_req=16 ttl=63 time=13058 ms
64 bytes from 192.0.5.10: icmp_req=17 ttl=63 time=13071 ms
64 bytes from 192.0.5.10: icmp_req=18 ttl=63 time=13084 ms
64 bytes from 192.0.5.10: icmp_req=129 ttl=63 time=12171 ms
64 bytes from 192.0.5.10: icmp_req=141 ttl=63 time=306 ms
64 bytes from 192.0.5.10: icmp_req=142 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=143 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=144 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=145 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=146 ttl=63 time=160 ms
64 bytes from 192.0.5.10: icmp_req=147 ttl=63 time=160 ms
^C
--- 192.0.5.10 ping statistics ---
147 packets transmitted, 26 received, +3 errors, 82% packet loss, time 19006ms
rtt min/avg/max/mdev = 160.243/9498.875/13084.330/5658.014 ms, pipe 140
root@cliente:/tmp/pycore.51479/cliente.conf#
```

1. Podemos ver que a conexão foi totalmente negada pelo servidor.
2. Foi realizada a comunicação de forma precária com um tempo grande para resposta do servidor.
3. O serviço já normalizado.

# Análise do ataque no Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=1/256, ttl=64 (no response found!)
2	0.000019	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=1/256, ttl=64 (no response found!)
3	0.000027	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=1/256, ttl=63 (no response found!)
4	0.000224	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=2/512, ttl=64 (no response found!)
5	0.000232	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=2/512, ttl=64 (no response found!)
6	0.000236	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=2/512, ttl=63 (no response found!)
7	0.000245	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=3/768, ttl=64 (no response found!)
8	0.000248	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=3/768, ttl=64 (no response found!)
9	0.000250	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=3/768, ttl=63 (no response found!)
10	0.000257	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=4/1024, ttl=64 (no response found!)
11	0.000260	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=4/1024, ttl=64 (no response found!)
12	0.000262	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=4/1024, ttl=63 (no response found!)
13	0.000268	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=5/1280, ttl=64 (no response found!)
14	0.000271	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=5/1280, ttl=64 (no response found!)
15	0.000273	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=5/1280, ttl=63 (no response found!)
16	0.000279	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=6/1536, ttl=64 (no response found!)
17	0.000281	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=6/1536, ttl=64 (no response found!)
18	0.000284	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=6/1536, ttl=63 (no response found!)
19	0.000290	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=7/1792, ttl=64 (no response found!)
20	0.000292	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=7/1792, ttl=64 (no response found!)
21	0.000294	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=7/1792, ttl=63 (no response found!)
22	0.000301	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=8/2048, ttl=64 (no response found!)
23	0.000303	192.0.3.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002a, seq=8/2048, ttl=64 (no response found!)

Dados da máquina zumbi2 capturados pelo Wireshark.

# Análise do ataque no Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
2477...	136.889125	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=49435/7105, ttl=63 (reply in 247704)
2477...	136.902115	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=50093/44483, ttl=63 (reply in 247709)
2477...	136.915136	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=50859/43974, ttl=63 (reply in 247714)
2477...	136.928125	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=51602/37577, ttl=63 (reply in 247719)
2477...	136.954088	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=52779/11214, ttl=63 (reply in 247729)
2477...	136.980114	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=53692/48337, ttl=63 (reply in 247739)
2477...	137.019107	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=55079/10199, ttl=63 (reply in 247754)
2477...	137.045129	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=56503/47068, ttl=63 (reply in 247764)
2477...	137.071139	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=58280/43235, ttl=63 (reply in 247774)
2477...	137.084107	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=59716/17641, ttl=63 (reply in 247779)
2477...	137.110131	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=60537/31212, ttl=63 (reply in 247789)
2477...	137.123124	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=61058/33518, ttl=63 (reply in 247794)
2478...	137.135942	192.0.8.1	192.0.8.10	ICMP	128	Destination unreachable (Host unreachable)
2478...	137.135968	192.0.8.1	192.0.8.10	ICMP	128	Destination unreachable (Host unreachable)
2478...	137.136395	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=61794/25329, ttl=63 (reply in 247807)
2478...	137.162132	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=63099/31734, ttl=63 (reply in 247817)
2478...	137.188110	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=64676/42236, ttl=63 (reply in 247827)
2480...	137.747117	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=65501/56831, ttl=63 (reply in 248050)
2480...	137.814223	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=4/1024, ttl=64 (no response found!)
2480...	137.814232	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=4/1024, ttl=64 (no response found!)
2484...	138.822331	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=5/1280, ttl=64 (no response found!)
2484...	138.822343	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=5/1280, ttl=64 (no response found!)
2488...	139.829995	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=6/1536, ttl=64 (no response found!)
2488...	139.830001	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=6/1536, ttl=64 (no response found!)
2489...	139.937098	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002c, seq=65504/57599, ttl=63 (reply in 248923)
2489...	139.952615	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=6/1536, ttl=63 (no response found!)
2489...	140.070107	192.0.8.10	192.0.5.10	ICMP	100	Echo (ping) request id=0x002d, seq=6/1536, ttl=63 (reply in 248982)

Dados da máquina cliente capturados pelo Wireshark.

# Referências

Core-dev (2012). Core 4.8 documentation. Disponível em: <http://downloads.pf.itd.nrl.navy.mil/docs/core/core-html/intro.html>. Acesso em 18 maio 2016.

Kumar, A., Sharma, A. K., and Singh, A. (2012). Performance evaluation of centralized multicasting network over icmp ping flood for ddos. International Journal of Computer Applications (0975–8887) Volume.

Oliveira, E., Aschoff, R., Lins, B., Feitosa, E., and Sadok, D. (2007). Avaliação de proteção contra ataques de negação de serviço distribuídos (ddos) utilizando lista de ips confiáveis. VII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais.

Orozco, A. M., Fernandes, A. P., and Costa, G. H. (2014). Simulação de syn flooding attackk no common open research emulator. Revista Competência , 7(1):161–173.