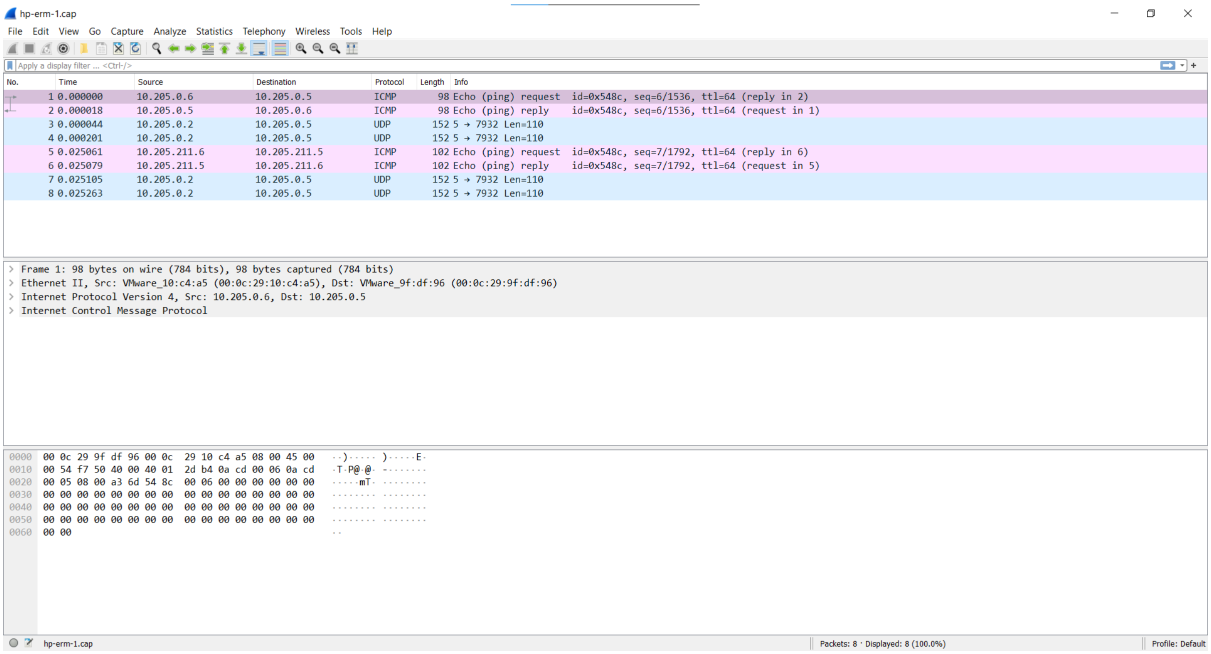
1. The downloaded pcap (Packet Capture) for wireshark:

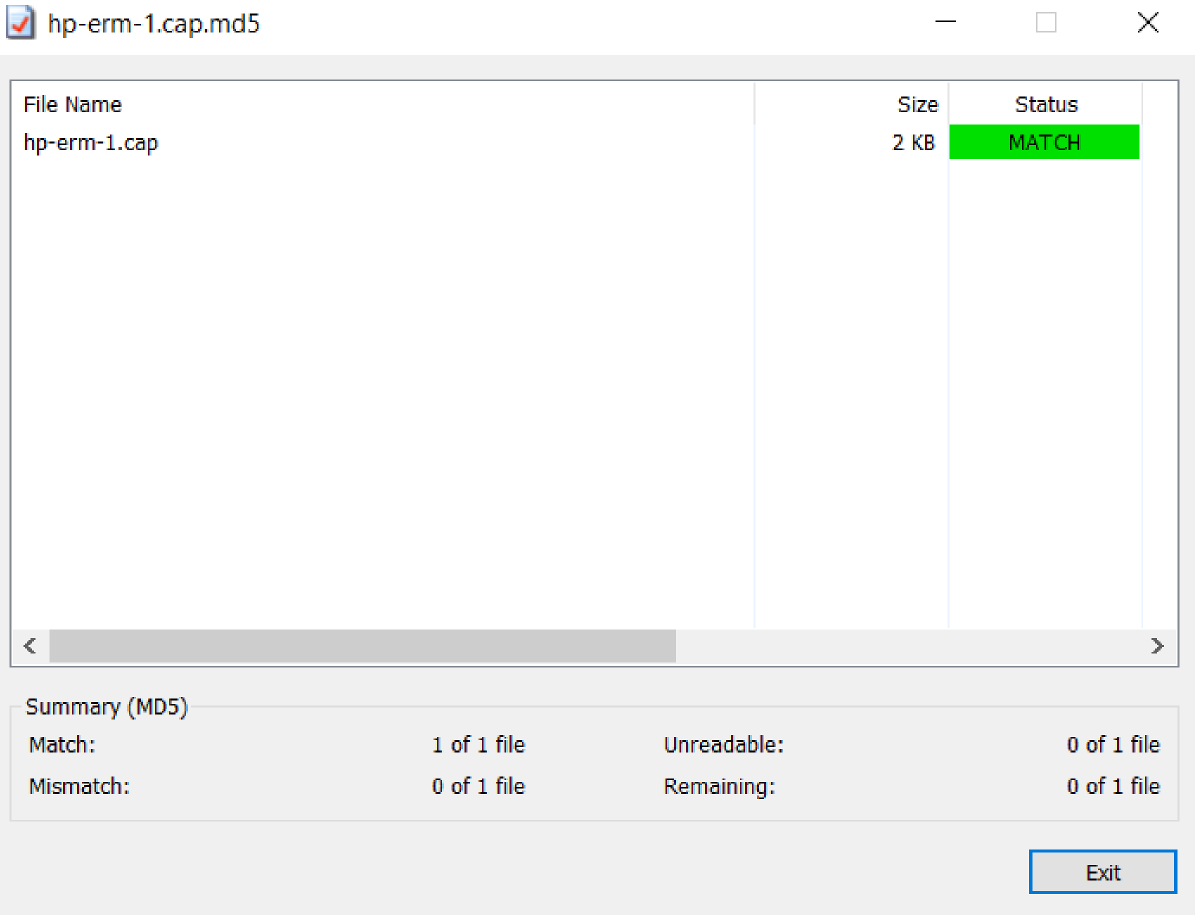

hp-erm-1.cap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.     Time      Source          Destination     Protocol  Length  Info
      1 0.000000  10.205.0.6      10.205.0.5      ICMP        98 Echo (ping) request  id=0x548c, seq=6/1536, ttl=64 (reply in 2)
      2 0.000018  10.205.0.5      10.205.0.6      ICMP        98 Echo (ping) reply    id=0x548c, seq=6/1536, ttl=64 (request in 1)
      3 0.000044  10.205.0.2      10.205.0.5      UDP        152 5 → 7932 Len=110
      4 0.000201  10.205.0.2      10.205.0.5      UDP        152 5 → 7932 Len=110
      5 0.025061  10.205.211.6    10.205.211.5    ICMP       102 Echo (ping) request  id=0x548c, seq=7/1792, ttl=64 (reply in 6)
      6 0.025079  10.205.211.5    10.205.211.6    ICMP       102 Echo (ping) reply    id=0x548c, seq=7/1792, ttl=64 (request in 5)
      7 0.025105  10.205.0.2      10.205.0.5      UDP        152 5 → 7932 Len=110
      8 0.025263  10.205.0.2      10.205.0.5      UDP        152 5 → 7932 Len=110
```

```
> Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: VMware_10:c4:a5 (00:0c:29:10:c4:a5), Dst: VMware_9f:df:96 (00:0c:29:9f:df:96)
> Internet Protocol Version 4, Src: 10.205.0.6, Dst: 10.205.0.5
> Internet Control Message Protocol
```

```
0000  00 0c 29 9f df 96 00 0c  29 10 c4 a5 08 00 45 00   ··)····· )·····E·
0010  00 54 f7 50 40 00 40 01  2d b4 0a cd 00 06 0a cd   ·T·P@·@· -·······
0020  00 05 08 00 a3 6d 54 8c  00 06 00 00 00 00 00 00   ·····mT· ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0060  00 00                                              ··
```

2. Running  the file (hp-erm-1.cap) with Hash Tool (HashCheck) to verify integrity:

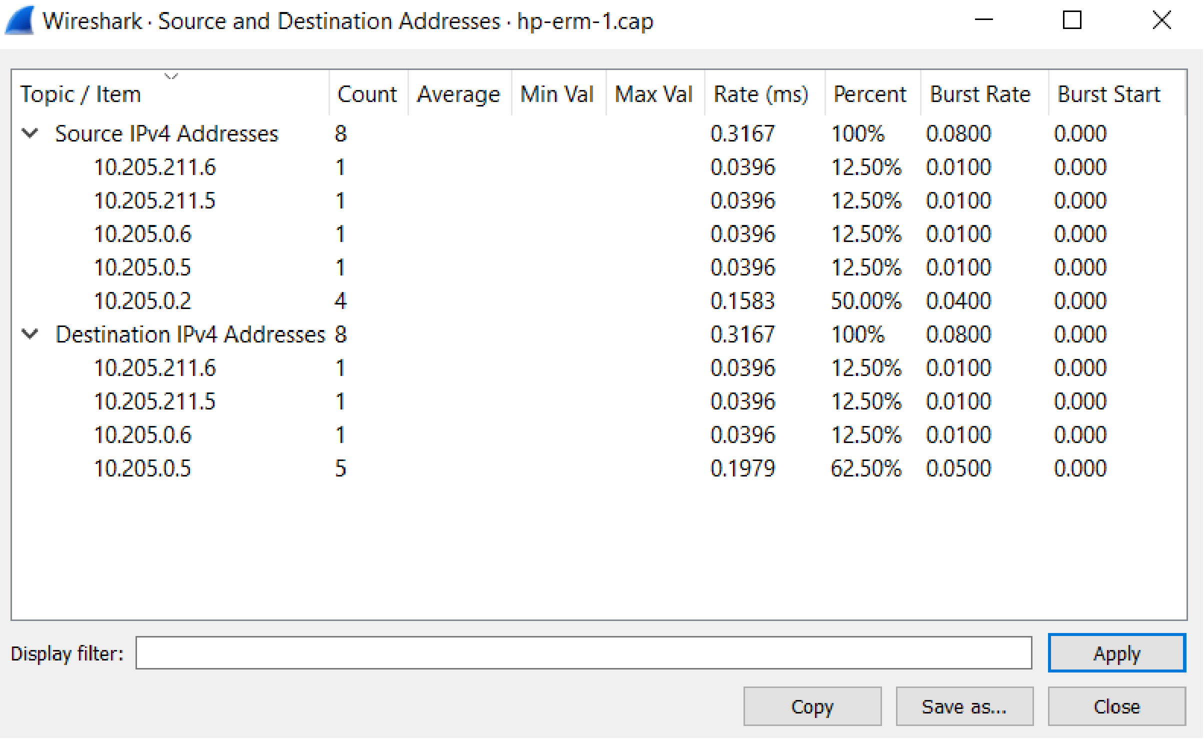


| File Name | Size | Status |
|---|---|---|
| hp-erm-1.cap | 2 KB | MATCH |

Summary (MD5)

| | | | |
|---|---|---|---|
| Match: | 1 of 1 file | Unreadable: | 0 of 1 file |
| Mismatch: | 0 of 1 file | Remaining: | 0 of 1 file |

3. Source and Destination IP addresses:

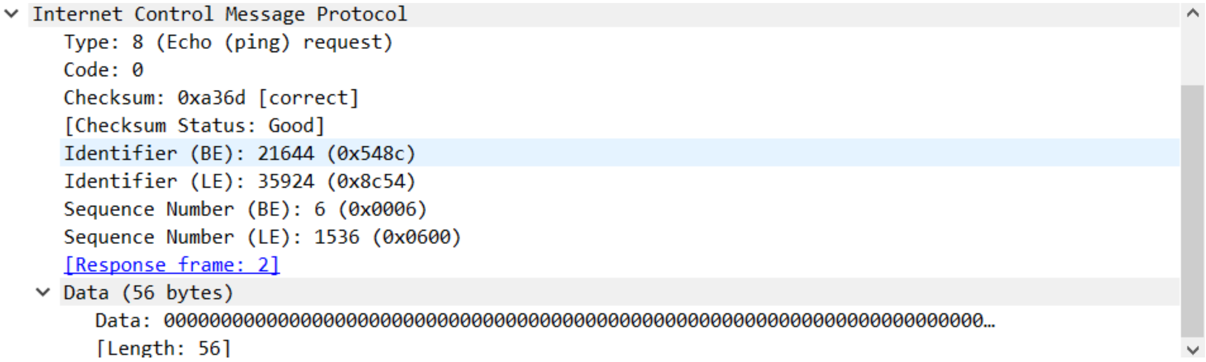


Wireshark · Source and Destination Addresses · hp-erm-1.cap

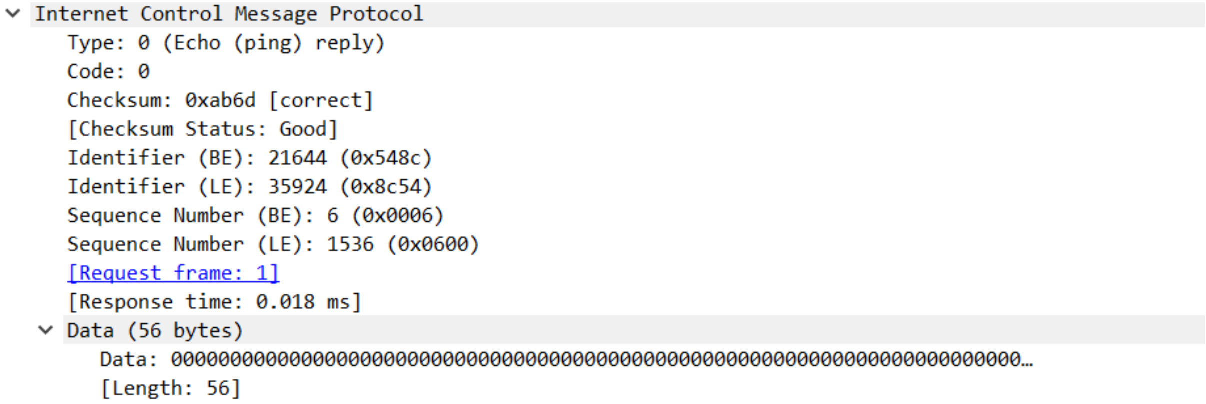
| Topic / Item | Count | Average | Min Val | Max Val | Rate (ms) | Percent | Burst Rate | Burst Start |
|---|---|---|---|---|---|---|---|---|
| Source IPv4 Addresses | 8 | | | | 0.3167 | 100% | 0.0800 | 0.000 |
| 10.205.211.6 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.211.5 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.0.6 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.0.5 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.0.2 | 4 | | | | 0.1583 | 50.00% | 0.0400 | 0.000 |
| Destination IPv4 Addresses | 8 | | | | 0.3167 | 100% | 0.0800 | 0.000 |
| 10.205.211.6 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.211.5 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.0.6 | 1 | | | | 0.0396 | 12.50% | 0.0100 | 0.000 |
| 10.205.0.5 | 5 | | | | 0.1979 | 62.50% | 0.0500 | 0.000 |

The Internet Control Message Protocol (ICMP) isn’t designed for carrying data like TCP(Transport Control Protocol) and UDP (User Datagram Protocol).
The purpose of ICMP packets is to carry error messages and implement simple management functions. Unlike HTTP and DNS, it doesn’t wrap and carry protocols.
As an error messaging protocol, the structure of an ICMP packet is designed to provide the necessary information to the recipient.
Error data in ICMP is carried in 2 values: the type and the code.
In the first figure of captured packets, no error(s) has been shown. In addition, the purpose of ping is to determine if the system at a certain IP address (10.205.0.5 in the example above) exists and is currently functional, and that a route to that system can be found.

The images below will demonstrate an ICMP ping request and response in Wireshark



```
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xa36d [correct]
    [Checksum Status: Good]
    Identifier (BE): 21644 (0x548c)
    Identifier (LE): 35924 (0x8c54)
    Sequence Number (BE): 6 (0x0006)
    Sequence Number (LE): 1536 (0x0600)
    [Response frame: 2]
  Data (56 bytes)
      Data: 0000000000000000000000000000000000000000000000000000000000000000…
      [Length: 56]
```



```
v Internet Control Message Protocol
     Type: 0 (Echo (ping) reply)
     Code: 0
     Checksum: 0xab6d [correct]
     [Checksum Status: Good]
     Identifier (BE): 21644 (0x548c)
     Identifier (LE): 35924 (0x8c54)
     Sequence Number (BE): 6 (0x0006)
     Sequence Number (LE): 1536 (0x0600)
     [Request frame: 1]
     [Response time: 0.018 ms]
   v Data (56 bytes)
        Data: 000000000000000000000000000000000000000000000000000000000000000000000000…
        [Length: 56]
```

As shown above, the first 2 values in the packet are the type and the code, indicating the purpose of the packet (type: 8, code: 0 for the request packet) and (type: 0, code: 0 for the response packet).

Next, the packet contains a checksum which is important since a single bit flip in the type or code can convey a completely different error message. In both packets it is shown as correct.

After that, ping packets contain identifiers and sequence numbers. Since ICMP is a **stateless protocol** these values help to match a response received by the sender to the corresponding request.

The ICMP protocol is designed to provide error information and perform simple diagnostic actions (like ping). Ping involves sending an ICMP ping request and looking for an ICMP ping response.
The main purpose of this protocol is to determine whether a system at a particular IP address exists and is operational or not.
It can be used for mapping a network during the reconnaissance phase of an attack. ICMP packets should be blocked at the network boundary, and unusual ICMP traffic from a host may indicate scanning by an attacker in preparation for indirect movement through the network.

The captured data packet does not show outdated or malformed packets.