
# OS Fingerprinting with ICMP

Laura Chappell, Senior Protocol Analyst
Protocol Analysis Institute [lchappell@packet-level.com]
www.packet-level.com – www.podbooks.com
HTCIA Member, IEEE Associate

Operating System (OS) fingerprinting is the process of learning what operating system is running on a device. This can be used by the curious network administrator when they see a new device on the network. Most likely, however, OS fingerprinting is done by an unwarranted party on your network. Just as a bank robber may examine the outside of a bank and watch the comings and goings of employees before robbing the bank, a hacker typically may perform a reconnaissance process on your network prior to launching an attack.

Internet Control Message Protocol (ICMP) is a protocol used to send error messages across a TCPIP network. Many people recognize ICMP as the protocol used by the *ping* utility. ICMP is also used with the standard *trace route* utility.

Besides offering wonderful connectivity tests functionality, ICMP can also be used as part of a reconnaissance scan on a network. In particular, ICMP can be used to perform an active OS fingerprint scan. In this article we will examine the typical ICMP packets that cross the cable when an OS fingerprint operation is performed on your network.

*Note: Ofir Arkin, founder of the SYS-Security Group, began research on using ICMP for OS fingerprinting in the winter of 2000. His document "ICMP Usage in Scanning" (http://www.sys-security.com/html/projects/icmp.html) offers a detailed analysis of ICMP's capability as an OS fingerprinting tool and the various responses received from different operating systems.*

If you place an analyzer outside your network firewall chances are good that you will experience numerous ICMP packets hitting that firewall. People are probably trying to discover and learn about your network on a daily basis with ping and/or trace route. You will most likely also see ICMP messages that indicate OS fingerprinting scans are underway.

Figure 1 shows an ICMP fingerprint operation as seen on Sniffer Pro.

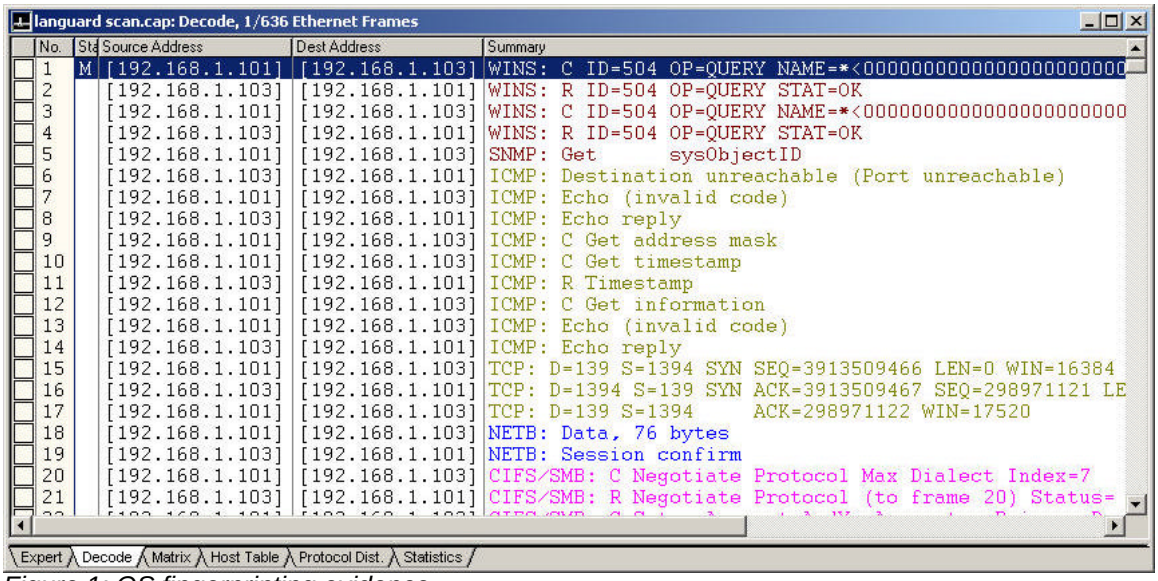


*Figure 1: OS fingerprinting evidence.*

The trace file shown in Figure 1 is available online at www.packet-level.com in the Library/Trace File area (**languard_scan** – available in .cap/.dmp/.pkt formats). Download the file and open it up in your analyzer to see inside each packet.

Packets number 7 through 17 are the OS fingerprinting operation in action. We will examine the entire set of ICMP packets (including packet 6) to see how the OS fingerprinting operation works.

### SNMP Query – ICMP Response

In Figure 1, packets 6 through 14 are ICMP packets. Packet 6 is simply an ICMP Destination Unreachable/Port Unreachable response to Packet 5. In Packet 5, the fingerprinter (192.168.1.101) has sent an SNMP Get sysObject ID request. Since the target device (192.168.1.103) does not support SNMP services, it sends back an ICMP Destination Unreachable/Port Unreachable message. This ICMP response by itself does not tell us anything about the target operating system. It does tell us, however, that the target does not support SNMP services.

### ICMP Echo

Packet 7 is the actual start of the OS fingerprinting operation. This packet is an ICMP Echo request packet, but it is not properly formed. ICMP Echo requests are used by the ping utility to establish whether a device is up and running on a network. In Packet 7, however, the fingerprinter has sent an ICMP Echo request packet with an invalid code. This is a malformed packet. If we look inside Packet 7, we would see the ICMP

header contains type 8 (echo) and code 19. In a typical ICMP Echo request, the type number should be 8, but the code should be 0.

By examining the response to an invalid ICMP echo request, the fingerprinter can determine if the target system examines the ICMP Echo request's code field at all. Some operating systems will look at the type field *and* the code field Others may look only at the type field and ignore an invalid code field.

In Packet 8, we see the target respond back with a standard ICMP Echo reply packet. This indicates that the target did not process the invalid code field. This gives the fingerprinter one clue about the OS running on the target system.

**ICMP Get Address Mask**

The next ICMP packet sent from the fingerprinter is an ICMP Get Address Mask request (ICMP Type 17). The ICMP Get Address Mask request was originally defined as a packet sent by diskless workstations to obtain a subnet mask at boot time. This address mask packet can also be used when one host wants to know the address mask of another host on the network. It is not a common packet to see on a network. This packet is being used for the purpose of OS fingerprinting since many operating systems do not support or respond to the Get Address Mask request. In our trace, we can see that the target machine does not respond to this request. Now we know a little more about the target OS.

**ICMP Get Timestamp**

In Packet 10, the OS fingerprinter sends a Get Timestamp request (ICMP Type 13) to the target. The ICMP Get Timestamp request allows one host to query another host for the current time. Initially, this was defined as a way for a sender to determine the latency time across a network. In this case, however, it is not being used to determine the latency time; it is being used to perform an OS fingerprint operation. In our trace we can see that the target does respond to a Get Timestamp request. Again, we have learned a bit about the type of operating system we might be targeting.

**ICMP Get Information**

In Packet 12, the OS fingerprinter has now sent an ICMP information request (Type 15) packet to the target. The ICMP information request process was defined to support diskless work stations during boot time. Using the information request packet, the diskless work station could discover there network address. On most networks today however, BOOTP and DHCP provide a better mechanism for IP address discovery. It is considered unusual to see a get information request cross a typical

network. In our trace we can see that the target does not respond to this packet. Again, this is a hint as to the type of operating system that is running on the target.

**Just for Good Measure…**
To finish the ICMP OS fingerprinting operation, the fingerprinter sends another ICMP Echo request using an invalid code (Code 19) to the target. The OS fingerprinter receives a reply back again from the target.

Packets 7-17 depict an unusual sequence of ICMP packets on a network. Typically, on your network you may see ICMP Echo requests (with valid codes) and possibly some ICMP Redirect packets. When you see this sequence (or a similar sequence) of ICMP packets on your network, you are being OS fingerprinted.

**Notes:**
In this example, LANguard Network Scanner (www.gfi.com) was used to perform an OS fingerprint operation as part of its standard vulnerability scan process.

RFC LIST (www.ietf.org):
RFC 792: Internet Control Message Protocol
RFC 1122: Requirements for Internet Hosts – Communication Layers
RFC 1256: Requirements for Internet Hosts – Application and Support
RFC 1349: Type of Service in the Internet Protocol Suite
RFC 1812: Requirements for IPv4 Routers

---

Laura Chappell is the Senior Protocol Analyst for the Protocol Analysis Institute. She is the author of numerous books and self-paced courseware available online at www.packet-level.com and www.podbooks.com. Laura also lectures on analysis, optimization and cybercrime. Her course schedule is online at www.packet-level.com.