

# Teredo Stray Packet Analysis

Johannes Ullrich

This investigation started with Rick observing some odd UDP traffic hitting his firewall. In this case, the traffic came from 66.55.158.116 port 3544. The destination port was a "random" high port. If you would like to provide your packet captures, see the end of this article for the right filter.

Port 3544 is assigned to Teredo. However, Teredo itself uses this port to establish connections, not necessarily for the actual Teredo tunnel traffic itself. As a host establishes a Teredo connection, it will connect to a Teredo server on port 3544 and negotiated the details of the connection. During this negotiation, a IPv6 address is established for the host.

The IPv6 address used by Teredo will always start with "2001:0000:". This /32 prefix is reserved for Teredo. It is followed by the IPv4 address of the Teredo server, 16 bits worth of "Flags" defining the type of NAT used by the client's network, the UDP port used to connect back to the client and finally the clients public IPv4 address. To illustrate this, here a more "graphical" representation of a Teredo address:

**2001:0000:SSSS:SSSS:FFFF:UUUU:CCCC:CCCC**

**S = Server IP address, F = Flags, U = obfuscated UDP port, C = public client IP address**

in short: Everything needed to connect back to the client is encoded in the IPv6 address.

The Teredo packet itself is rather simple. The IPv6 packet is embedded in a IPv4 UDP packet. We got an IPv4 header, a UDP header followed by the IPv6 header. Wireshark for example does a nice job in analyzing Teredo traffic.

Now lets go back to Rick's packet trace. If all this is true, then the last 4 bytes of the embedded IPv6 destination address should match the public IPv4 address the packet was sent to. In Rick's case, this wasn't the case. The two addresses didn't match at all.

The only packets Rick saw were "Bubble Packets". Bubble packets are used by Teredo to keep the firewall open. Typically every 30 seconds, the Teredo client will send an empty IPv6 packet to the server. This will extend the connection timeout in most firewalls. For UDP, there is no state like for TCP. In order to preserve the resemblance of state, firewalls assume that if a host sent a packet out, there may be a reply coming back. Teredo takes advantage of this "statefull UDP" feature in modern firewalls.

This is why we would like to connect more of these packets to see if Rick's experience was just a "one off" oddity or if others are seeing the same traffic. Rick did not use Teredo or IPv6 for that matter. We are interested in your packet in particular if you are NOT using IPv6.

You can use tcpdump or windump to collect the traffic. You probably have to collect the traffic outside of your firewall. The tcpdump line is not perfect, and may capture some other UDP traffic (e.g. DNS). Before you send it to us, take a quick look at it to make sure you are not sending us any confidential data. As a reminder, the tcpdump command line to collect the traffic:

```
tcpdump -c100 -s0 -i any -w /tmp/teredo udp port 3544
```