Home > Network and technology service docs > Janet CSIRT > Technical advice > Investigating a Denial of Service attack > What we saw

What we saw

We left the monitor in place for two days, until our log file began to grow rapidly indicating a new attack in progress. The following entries are typical of what was observed:

```
[**] IDS253 - DDoS shaft synflood outgoing [**]

06/12-14:30:46.599036 8:0:20:1B:22:A9 -> 0:D0:D3:56:D1:30 type:0x800

len:0x3C

98.76.54.111:1008 -> 12.34.56.78:6666 TCP TTL:30 TOS:0x0 ID:59926 DF

**S***** Seq: 0x28374839 Ack: 0x3E0D641F Win: 0xFFFF

[**] IDS253 - DDoS shaft synflood outgoing [**]

06/12-14:30:46.602703 8:0:20:1B:22:A9 -> 0:D0:D3:56:D1:30 type:0x800

len:0x3C

98.76.54.93:1020 -> 22.44.66.88:6667 TCP TTL:30 TOS:0x0 ID:59936 DF

**S***** Seq: 0x28374839 Ack: 0x3E0D641F Win: 0xFFFF

 [**] IDS253 - DDoS shaft synflood outgoing [**]

06/12-14:30:46.769474 8:0:20:1B:22:A9 -> 0:D0:D3:56:D1:30 type:0x800

len:0x3C

98.76.54.224:1009 -> 12.34.56.78:6667 TCP TTL:30 TOS:0x0 ID:59940 DF

**S***** Seq: 0x28374839 Ack: 0x3E0D641F Win: 0xFFFF
```

Each entry corresponds to a packet matched against an attack signature. We have three packets, sourced apparently from three different hosts on our network (98.76.54.111, 98.76.54.93, 98.76.54.224) and targeted at two machines (12.34.56.78 and 22.44.66.88). Several characteristics recorded by Snort are interesting, but most especially the layer 2 (MAC) addresses. As we would expect, the destination MAC address (0:D0:D3:56:D1:30) is in each case that of our router - the gateway to the rest of our network and, ultimately, the outside world. But we also see that the source MAC addresses are all the same (8:0:20:1B:22:A9). Although the IP addresses indicate the traffic is coming from three different machines, the layer 2 information shows they in fact originate from the same ethernet card.

This uniquely identifies the real source of the attack. If you keep records of system hardware addresses, associating the card with a named system is then trivial. If records aren’t available, broadcast pings on the network followed by inspection of the arp cache can help. For example, on an NT workstation connected to the departmental LAN:

```
C:\> ping 98.76.54.255

C:\> arp -a

Interface: 98.76.54.101 on Interface 2

Internet Address  Physical Address       Type

98.76.54.2          08-00-xx-xx-xx-xx     dynamic

98.76.54.11         00-30-xx-xx-xx-xx     dynamic

98.76.54.22         00-60-xx-xx-xx-xx     dynamic

98.76.54.24         08-00-xx-xx-xx-xx     dynamic

98.76.54.25         08-00-20-1b-22-a9     dynamic

98.76.54.30         00-60-xx-xx-xx-xx     dynamic

98.76.54.31         08-00-xx-xx-xx-xx     dynamic

98.76.54.32         00-30-xx-xx-xx-xx     dynamic
```

Otherwise it is a matter of checking machines by hand.

In our case, a detailed examination of host 98.76.54.25 (a Sun Solaris system) revealed the presence of several trojan versions of system binaries (including /bin/login and netstat). Sun’s ?ngerprint database [5] [1] was particularly helpful here. Linux users might use rpm’s verify capability to a similar end, assuming that the rpm database has not itself been tampered with.

Source URL: https://community-stg.jisc.ac.uk/library/janet-services-documentation/what-we-saw

Links
[1] https://community.ja.net/library/janet-services-documentation/references-3