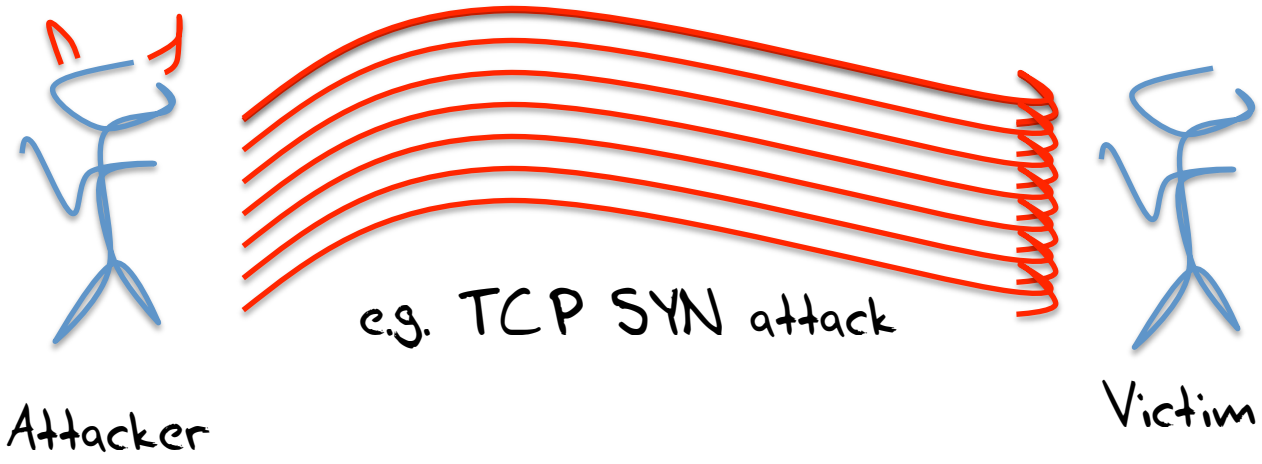
NTP and Evil

Geoff Huston, Randy Bush

The Evolution of Evil

- It used to be that you sent evil packets to your chosen victim

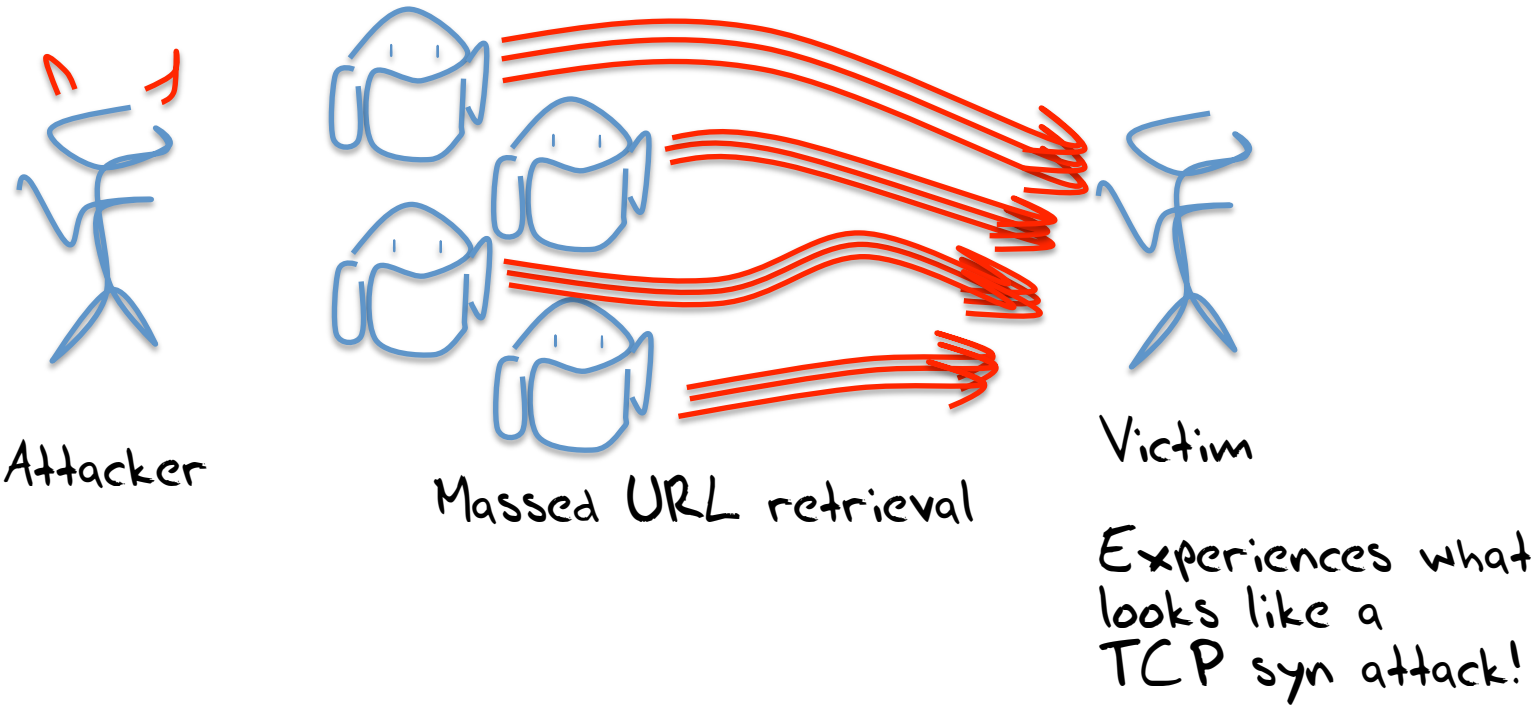
  but this exposed you, and limited the damage you could cause



The Evolution of Evil

- Then you enrolled a bot army to send evil

  which kept you hidden and increased the damage leverage



The Evolution of Evil

- But now you co-opt the innocent into the evil cause, and use uncorrupted servers to launch the attack

  which hides the attacker(s) and uses the normal operation of servers to cause damage

UDP is a Fine Protocol

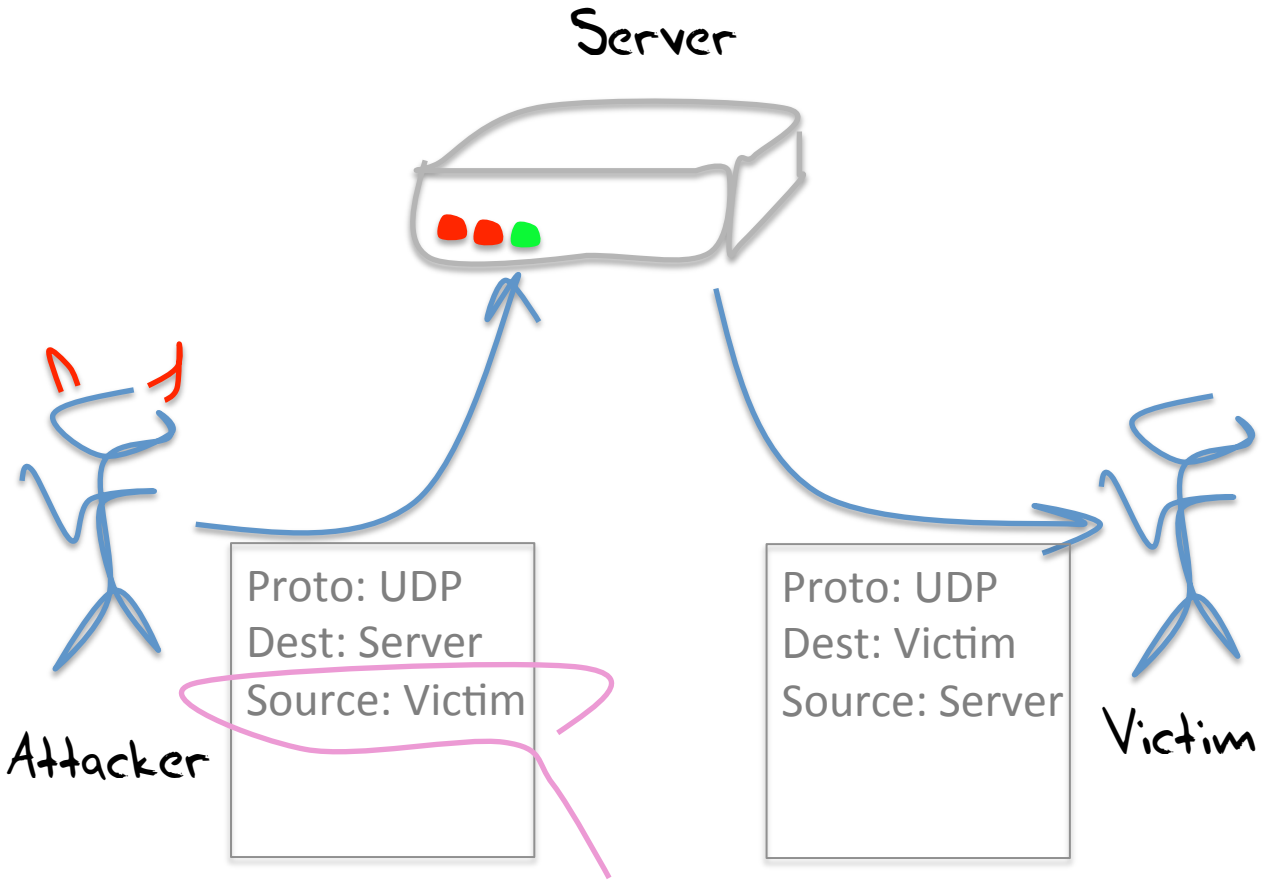
- UDP is used whenever you want a fast and highly efficient short transaction protocol
- Send a query to a server (one packet)
- And the server sends an answer (one packet)
- UDP works best when the question and the answer are small (<512 bytes), but can work on larger transactions*
- Although it's not as reliable as TCP

* The fine print (yes, you'll need to magnify this to read it!)
Some UDP applications use multiple UDP packets for large answers (e.g. NTP). Some rely of IP level fragmentation (e.g. DNS with EDNS0)
The problem with relying on fragmentation is firewall filtering and NATs (the trailing frags have no transport level header to assist in locating the NAT binding , as fragme
And the problem with multiple UDP packets is that the onus for reliable reassembly is pushed into the application, which may not necessarily do this well!, And the send
with no flow control, which can be bad as well

UDP Mutation

- Unlike TCP there is no handshake between the two parties
  - Send the server a UDP packet
  - The server flips the source and destination IP addresses and responds with a UDP packet
  - The server never checks the authenticity of the source address
- This allows a simple reflection attack

UDP Reflection Attack



note fake source!

UDP and DDOS Reflection Attacks

This works “best” for a UDP-based service when:

- The service is widely used
- Servers are commonplace
- Servers are poorly maintained (or unmaintained)
- Clients are not “qualified” by the server (i.e. anyone can pose a query to a server)
- The answer is far bigger than the question

Hmmmmm

What could that be?

The DNS!!!

- UDP-based query response service
  UDP is now almost ubiquitous for the DNS – EDNS0 wiped out the last vestiges of TCP fallback for most DNS resolvers
- The service is widely used
  Everybody is a client of the DNS
- Servers are commonplace
  Resolvers are scattered all over the Internet
- Servers are poorly maintained (or unmaintained)
  There are some 30 million open resolvers
- Clients are not “qualified” by the server (i.e. anyone can pose a query to a server)
  authoritative DNS name servers are promiscuous by design
  Many DNS resolvers are unintentionally promiscuous
- The answer is be far bigger than the question
  Just ask the right DNS question!

Co-Opting the DNS for Evil

- DNS DDOS attacks are now very commonplace
- They can (and do) operate at sustained gigabit speeds
- Efforts to mitigate tend to degrade the quality of the service as well as affecting the victim

What other UDP services are susceptible?

chargen?

snmp?

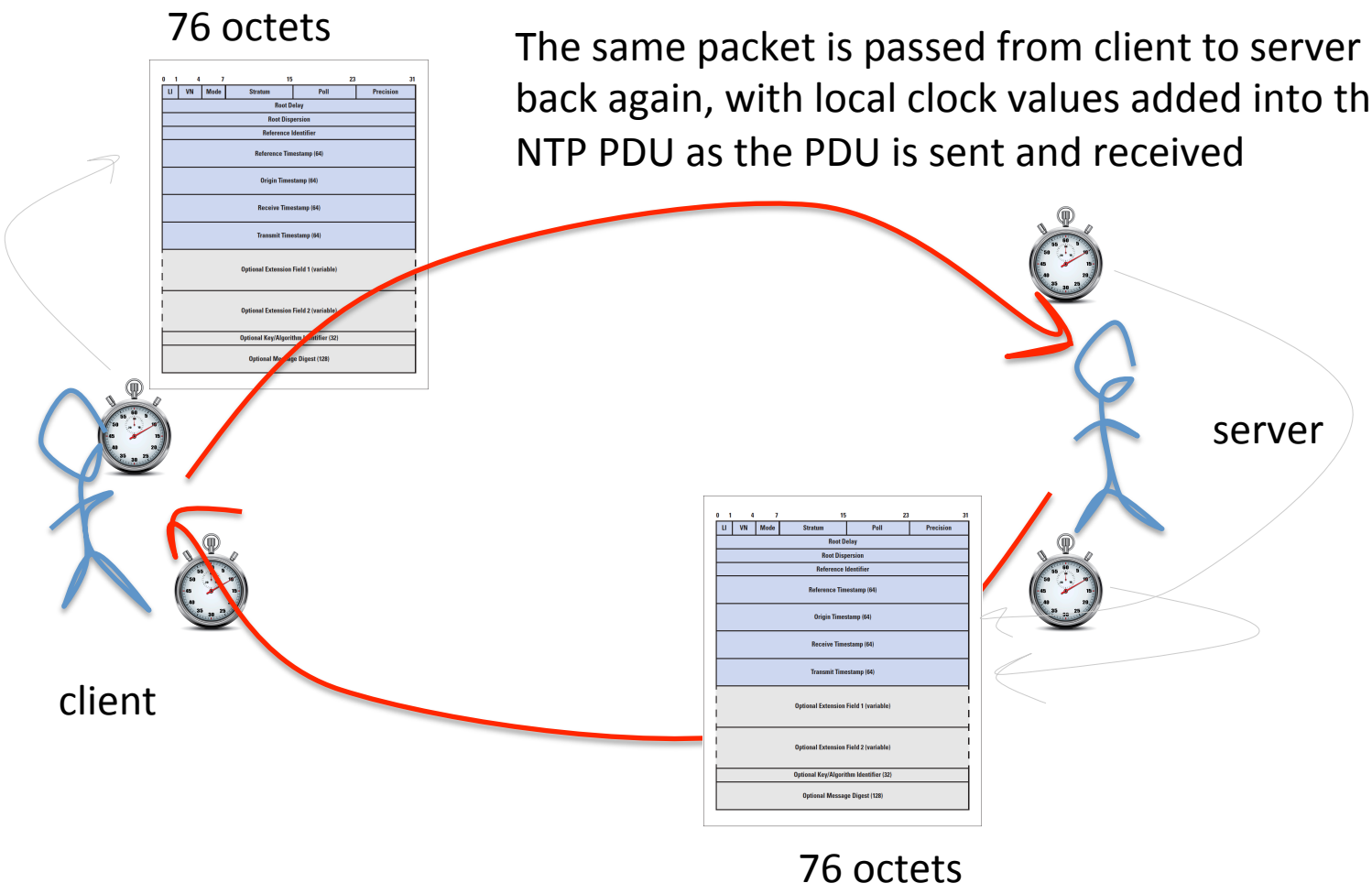
It's as easy as 1, 2, 3!

- NTP is a simple UDP query/ response protocol, where the NTP server listens on UDP port 123
- Time is important for network-distributed services
- So we've deployed a lot of NTP servers to distribute time across the network

NTP and UDP Reflection Attacks

- UDP-based query response service
  UDP is ubiquitous for NTP
- The service is widely used
  Time is widely distributed
- Servers are commonplace
  NTP servers are scattered all over the Internet
- Servers are poorly maintained (or unmaintained)
  NTP tends to be operated in a "configure and forget" mode
- Clients are not "qualified" by the server (i.e. anyone can pose a query to a server)
  NTP is not necessarily promiscuous
  But it is often configured in a promiscuous mode
- The answer is far bigger than the question
  Not normally…

NTP transactions are symmetric



NTP

The NTP server's time response is the same size as the NTP time query

> Which limits the types of attacks that are effective, as this becomes indirection rather than indirection + amplification

But the NTP folk added another hook into the model

- The NTP command and control channel is also implemented in UDP, using the same UDP port

NTP

The NTP server's time respons[illegible] the same size as the NTP tim[illegible]

> Which [illegible] effective, as this becom[illegible]ion + amplifi[illegible]

But the NT[illegible]k into the model

- The NTP [illegible] control channel is also implemented in UDP, using the same UDP port

Ooops!

NTP Command and Control

ntpdc – the "special" NTP query program

"monlist" returns the IP addresses of the last (up to) 600 systems that this NTP server has interacted with

```
ntpdc –c monlist <server>
```

(There are other commands, but "monlist" provides the highest amplification)

One UDP packet of 220 bytes input generates up to 100 x 468 byte UDP packets in response

That's an impressive amplification factor of 212!)

6perf — ssh — 104×72

```
$ ntpdc -c monlist 127.0.0.1
```

drafts — ssh — 147×6

```
listening on lo0, link-type NULL (BSD loopback), capture size 65535 bytes
22:07:42.821666 IP 127.0.0.1.44354 > 127.0.0.1.123: NTPv2, Reserved, length 192
```

6perf — ssh — 104×72

```
$ ntpdc -c monlist 127.0.0.1
remote address          port local address      count m ver code avgint  lstint
===============================================================================
localhost               44354 127.0.0.1             8 7 2      0    9333       0
bogong.rand.apnic.net     123 88.198.69.72       4040 1 4    1d0     128       7
drongo.rand.apnic.net     123 88.198.69.72       3273 1 4    1d0     624      28
nong.rand.apnic.net       123 88.198.69.72       3538 1 4    1d0     176      36
wattle.rand.apnic.net     123 88.198.69.72       3910 1 4    1d0     622      92
199.102.79.186            123 88.198.69.72       2514 3 4    1d0     231     107
202.158.221.222           123 88.198.69.72       3504 1 4    1d0     624     115
amchur.rand.apnic.net     123 88.198.69.72       1327 1 4    1d0     339     277
junk.rand.apnic.net       123 88.198.69.72       2730 1 4    1d0    1024     595
thickie.rand.apnic.net    123 88.198.69.72       3922 1 4    1d0     615     639
mirin.rand.apnic.net      123 88.198.69.72       3165 1 4    1d0     615     654
hoon.rand.apnic.net       123 88.198.69.72       3421 1 4    1d0     564     669
matong.rand.apnic.net     123 2a01:4f8:140:50c5::69:72     3190 1 4    1d0    1022     715
h.labs.apnic.net          123 88.198.69.72       2126 1 4      1    1024     791
ponzu.rand.apnic.net      123 88.198.69.72       2676 1 4    1d0    1024     820
twerp.rand.apnic.net      123 88.198.69.72       3784 1 4    1d0     565     938
gronggrong.rand.apnic.    123 2a01:4f8:140:50c5::69:72     2793 1 4    1d0    1005     979
dipstick.rand.apnic.ne    123 88.198.69.72       2241 1 4    1d0     756    1817
ip-50-63-188-5.ip.secu  53441 88.198.69.72          2 3 4    1d0       1  237998
myserver.serverforumho  52177 88.198.69.72          4 7 2    1d0  403258  242836
scanresearch1.syssec.r  10151 88.198.69.72          6 6 2    1d0  106974  525008
www.openresolverprojec  57915 88.198.69.72          6 7 2    1d0  116153  559858
58.215.177.51           49830 88.198.69.81          2 3 4    1d0   55199  614627
internetsurvey-6.errat  54660 88.198.69.72          2 7 2    1d0      11  624948
gorenje.rand.apnic.net    123 88.198.69.72        283 1 4    1d0     971  769504
kunden.telemaxx.net     36957 88.198.69.81          1 7 0    1d0       0  876626
ntpresearch.cymru.com   16090 88.198.69.72          1 6 2    1d0       0  917511
mail.mante.nl           15123 88.198.69.81          2 7 2    1d0     211  941274
static.98.220.46.78.cl  58481 88.198.69.72          2 7 2    1d0       5 1104975
host.premiumdedicateds  52304 88.198.69.72          1 7 0    1d0       0 1147424
62-210-137-230.rev.pon  40470 88.198.69.81          2 7 0    1d0     212 1172336
freebsdgirl.org         21608 88.198.69.72          2 7 2    1d0      13 1206822
buri.und.pw             52149 88.198.69.81          2 7 0    1d0  131998 1255102
timeserver3.intellicen    123 88.198.69.72       1111 4 4    1d0    1441 1270992
118.192.48.27           52147 88.198.69.72          2 7 2    1d0   65113 1319859
80.82.64.138            55340 88.198.69.81         16 7 2    1d0   39901 1632392
77.243.180.146            123 88.198.69.72       2325 7 2    1d0       0 2012254
hosted-by.hosthatch.co    123 88.198.69.81         30 7 2    1d0    5861 2012847
199.250.61.98             123 88.198.69.72      17813 7 2    1d0       0 2048375
cpc14-shep11-2-0-cust3    123 88.198.69.81       1536 7 2    1d0       1 2048462
reverse.hfbservers.com    123 88.198.69.81       5028 7 2    1d0       0 2048463
web7.sh3lls.net           123 88.198.69.81        468 7 2    1d0       0 2048466
65.52.24.110              123 88.198.69.81       4228 7 2    1d0       1 2048467
99-136-80-42.lightspee    123 88.198.69.81        824 7 2    1d0       0 2048471
cpe-71-64-205-157.cinc    123 88.198.69.81        428 7 2    1d0       0 2048472
c-68-42-230-87.hsd1.mi    123 88.198.69.81       1120 7 2    1d0       1 2048557
d192-24-125-122.try.wi    123 88.198.69.81        680 7 2    1d0       1 2048624
pool-108-39-84-238.nrf    123 88.198.69.81         40 7 2    1d0       1 2048699
71-17-76-12.regn.hsdb.    123 88.198.69.81         72 7 2    1d0       1 2048714
cpe-174-101-155-45.cin    123 88.198.69.81        120 7 2    1d0       1 2048750
                          123 88.198.69.81        508 7 2    1d0       0 2048781
vserver2.axc.nl           123 88.198.69.81        560 7 2    1d0       1 2048797
108-205-29-126.lightsp    123 88.198.69.81       1680 7 2    1d0       1 2048844
208.43.243.245-static.    123 88.198.69.81       9216 7 2    1d0       1 2048880
d149-67-104-1.clv.wide    123 88.198.69.81        120 7 2    1d0       1 2048885
cpe-184-57-165-8.cinci    123 88.198.69.81         52 7 2    1d0       0 2048935
c-98-209-219-102.hsd1.    123 88.198.69.81        120 7 2    1d0       1 2049069
168.61.144.13             123 88.198.69.81      24324 7 2    1d0       1 2049110
adsl-74-240-202-184.bn    123 88.198.69.81         36 7 2    1d0       1 2049118
24-241-114-160.dhcp.gn    123 88.198.69.81         24 7 2    1d0       1 2049178
host-200-6-122-182.iia    123 88.198.69.81         16 7 2    1d0       1 2049298
pool-71-179-43-110.blt    123 88.198.69.81         80 7 2    1d0       1 2049400
68-113-162-31.dhcp.plb    123 88.198.69.81        584 7 2    1d0       0 2049439
74-130-117-201.dhcp.in    123 88.198.69.81       2240 7 2    1d0       1 2049645
p5085394C.dip0.t-ipcon    123 88.198.69.81       1060 7 2    1d0       1 2049978
dal01.multiplay.co.uk     123 88.198.69.81         76 7 2    1d0       1 2050032
053f90e4.rdns.100tb.co    123 88.198.69.81        560 7 2    1d0       1 2050162
201-246-174-119.baf.mo    123 88.198.69.81         80 7 2    1d0       1 2050166
96.58.52.100              123 88.198.69.81         64 7 2    1d0       0 2050203
```

drafts — ssh — 147×6

```
listening on lo0, link-type NULL (BSD loopback), capture size 65535 bytes
22:07:42.821666 IP 127.0.0.1.44354 > 127.0.0.1.123: NTPv2, Reserved, length 192
22:07:42.821712 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821716 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821718 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821721 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821724 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821726 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821729 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821731 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821734 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821736 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821739 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821741 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821743 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821746 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821748 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821750 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821752 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821754 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821757 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821759 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821761 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821763 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821765 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821768 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821770 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821772 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821774 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821777 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821779 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821781 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821784 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821786 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821788 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821790 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821793 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821795 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821797 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821800 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821802 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821804 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821806 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821808 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821811 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821813 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821815 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821817 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821819 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821821 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821823 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821825 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821827 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821829 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821831 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821834 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821836 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821838 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821840 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821842 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821844 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821846 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821848 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821850 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821852 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821854 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
22:07:42.821857 IP 127.0.0.1.123 > 127.0.0.1.44354: NTPv2, Reserved, length 440
```

World's largest DDoS strikes US, Europe

Powered by SC Magazine

By Darren Pauli on Feb 11, 2014 1:55 PM (12 hours ago)
Filed under Security

8 Comments

New attack vector a sign of "ugly things to come".

A content delivery network provider has today been hit by what appears to be the world's largest denial of service attack, in an assault that exploits an emerging and frightening threat vector.

The Network Time Protocol (NTP) Reflection attack exploits a timing mechanism that underpins a way the internet works to greatly amplify the power of what would otherwise be a small and ineffective assault.

US-based DDoS protection outfit CloudFlare was hit with the attacks after an unnamed customer was targeted.

It is unclear how many websites and users were affected, although at least one French networking host reported a 350Gbps DDoS attack during the assault.

CloudFlare chief executive Matthew Prince said the attack tipped 400Gbps, 100Gbps larger than the previous record DDoS attack which used DNS reflective amplification.

Tags

cloudflare, ddos, ntp, dos, security, infosec

Related Articles

- UK spooks launched DDoS attacks against Anonymous
- DoS attack hits US federal court

Most read | Most discussed

- World's largest DDoS strikes US, Europe
- Russia bans Bitcoin
- The Aussie cities with the best IT job prospects
- Weather bureau forced to dump water data project
- NSW Government hands down mobility rules

Latest Videos See all videos »

What you need to be naughty

To Do List

- ❑ Generate a list of open NTP hosts (zmap, for example is a good starting point)
- ❑ Write a simple script that sends monlist commands to the open server, with UDP source address spoofing
- ❑ Enlist some coercible hosts to generate some 2,500 monlist queries per second
- ❑ And the servers will respond with a 1G bps DDOS stream!
- ❑ Rinse, repeat and multiply

What you need to be nice

- Seal up your NTP
  - The following Team Cymru's secure template for NTP should help:

    http://www.team-cymru.org/ReadingRoom/Templates/secure-ntp-template.html

- Disable monlist
  - Upgrade NTP to at least version 4.2.7p26

Being Nice on a (cisco ios) Router

ios (recent 12.* releases)

```
access-list 46 remark utility ACL to block everything
access-list 46 deny any
!
access-list 47 remark NTP peers/servers we sync to/with
access-list 47 permit 10.0.0.1
access-list 47 permit 10.0.0.2
access-list 47 deny any
!
! NTP access control
ntp access-group query-only 46   ! deny all NTP control queries
ntp access-group serve 46        ! deny all NTP time and control by default
ntp access-group peer 47         ! permit sync to configured peer(s)/server(s)
ntp access-group serve-only 46   ! deny NTP time sync requests
```

Being Nice on a (cisco xr) Router

ios/xr

```
Ntp
server 10.0.0.1
Server 10.0.0.2
source Loopback0
update-calendar
!
! local packet transport service config
lpts pifib hardware police location 0/2/CPU0
flow ntp default rate 0
flow ntp known rate 64
!
! The input/loopback filter for xr
control-plane
management-plane
 inband
  interface all
  !!! oh, no config here for ntp, I guess LPTS handles it all?
```

Being Nice on a (juniper) Router

juniper

This is a firewall filter fragment for a loopback filter which assumes a default permit

```
term ntp {
    from {
      source-address {
         0.0.0.0/0;
         /* NTP servers to get time from */
         10.0.0.1 except;
         10.0.0.2 except;
         }
      protocol udp;
      port ntp;
      }
    then {
       discard;
       }
     }
```

The alternative is to use a loopback default deny filter, in which case you would need the inverse form of the filter to accept NTP packets from the configured servers:

```
term ntp {
     from {
       source-address {
          10.0.0.1/23;
          10.0.0.2/32;
          }
       protocol udp;
       port ntp;
       }
    then {
       count ntp-requests;
       accept;
       }
     }
```

Being nice on a host

/etc/ntp.conf

```
# By default, exchange time with everybody, but don't allow
# configuration.
#
restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery
#
# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1
restrict ::1
```

But…

- Being nice is not always possible
  - There is a significant volume of embedded functionality in appliances and consumerware
  - And enough of it includes NTP to be a problem that is not going to be "fixed" anytime soon
- Which leads to the underlying observation:

  that despite more than 15 years of lip service, without much actual support in our networks, Source Address Filtering really IS important!

How to be nice to each other

- Perform Source Address Validation filtering on all outgoing ports
  - i.e. deploy BCP38 in your network!

Some Useful Resources

NTP Monlist command:

http://www.eecis.udel.edu/~mills/ntp/html/ntpdc.html

Description of NTP attack

http://blog.cloudflare.com/understanding-and-mitigating-ntp-based-ddos-attacks

Sealing up NTP – a template for ntp.conf

http://www.team-cymru.org/ReadingRoom/Templates/secure-ntp-template.html

Open NTP servers

http://openntpproject.org

BCP 38

http://bcp38.info

BCP 38 tracking

http://spoofer.cmand.org//

Thanks!