# Identifying Open Resolvers in IPv6

Dario Gomez dario at lacnic dot net
Alejandro Acosta alejandro at lacnic dot net

# What is all this about?

Simple: As you know identifying Open Recursive DNS Servers in the IPv4 world is very easy, we just tried a different approach in IPv6

# Why we did it?

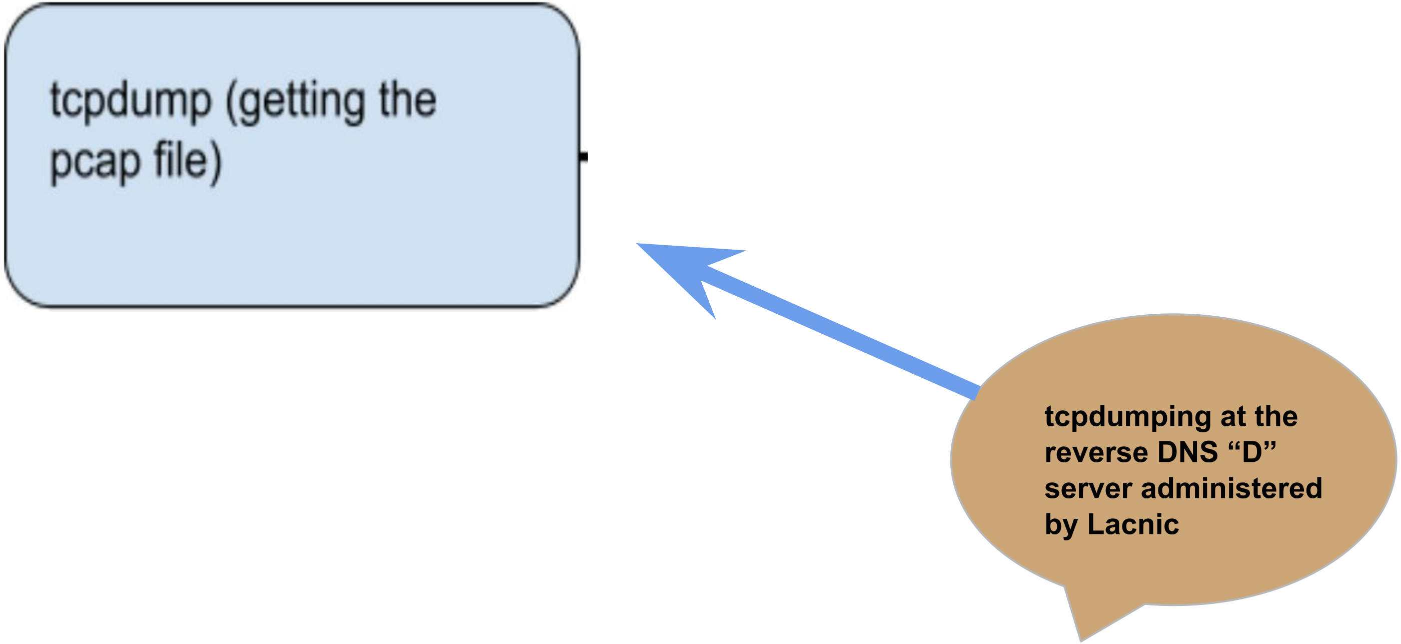
Open DNS resolvers are a bad idea for a few reasons:

- They allow outsiders to consume resources that do not belong to them.
- Attackers may be able to poison the cache of an open resolver.
- Open resolvers are being used in widespread DDoS attacks with spoofed source addresses and large DNS reply messages.

Source: http://dns.measurement-factory.com/surveys/openresolvers.html

# Ok.., too much talk. What is the approach you previously mentioned ! (⅕)

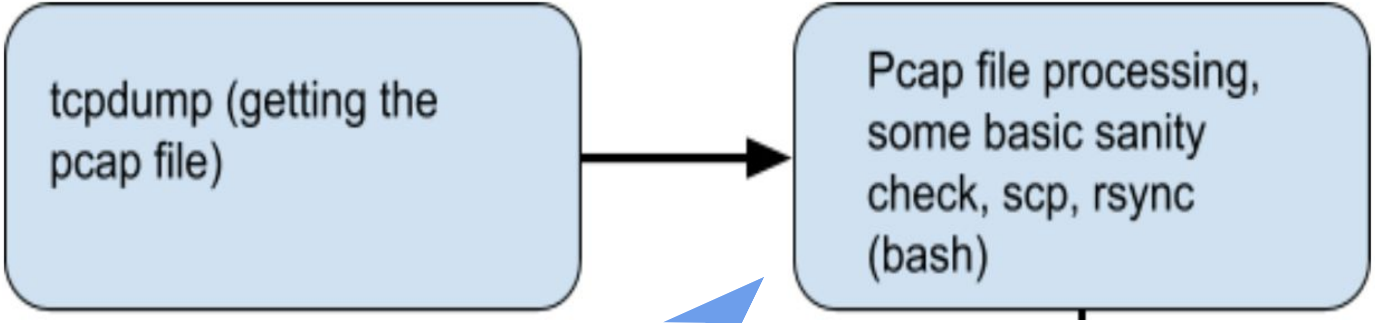
# Ok.., too much talk. What is the approach you previously mentioned ! (2/5)



tcpdump (getting the pcap file)

tcpdumping at the reverse DNS “D” server administered by Lacnic

# Ok.., too much talk. What is the approach you previously mentioned ! (3/5)

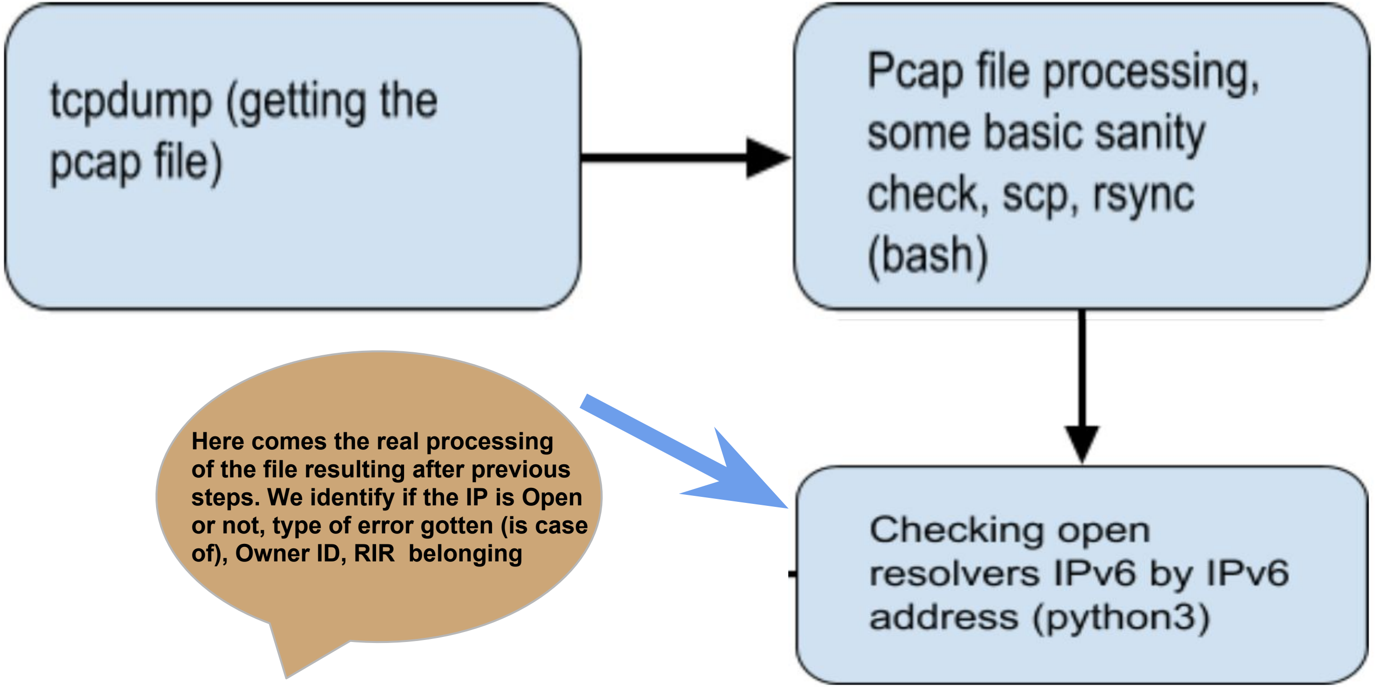


tcpdump (getting the pcap file)

Pcap file processing, some basic sanity check, scp, rsync (bash)

After the pcap file is gathered then we move it to another server, process it to be sure it is ok, and few more things

# Ok.., too much talk. What is the approach you previously mentioned ! (4/5)



tcpdump (getting the pcap file)

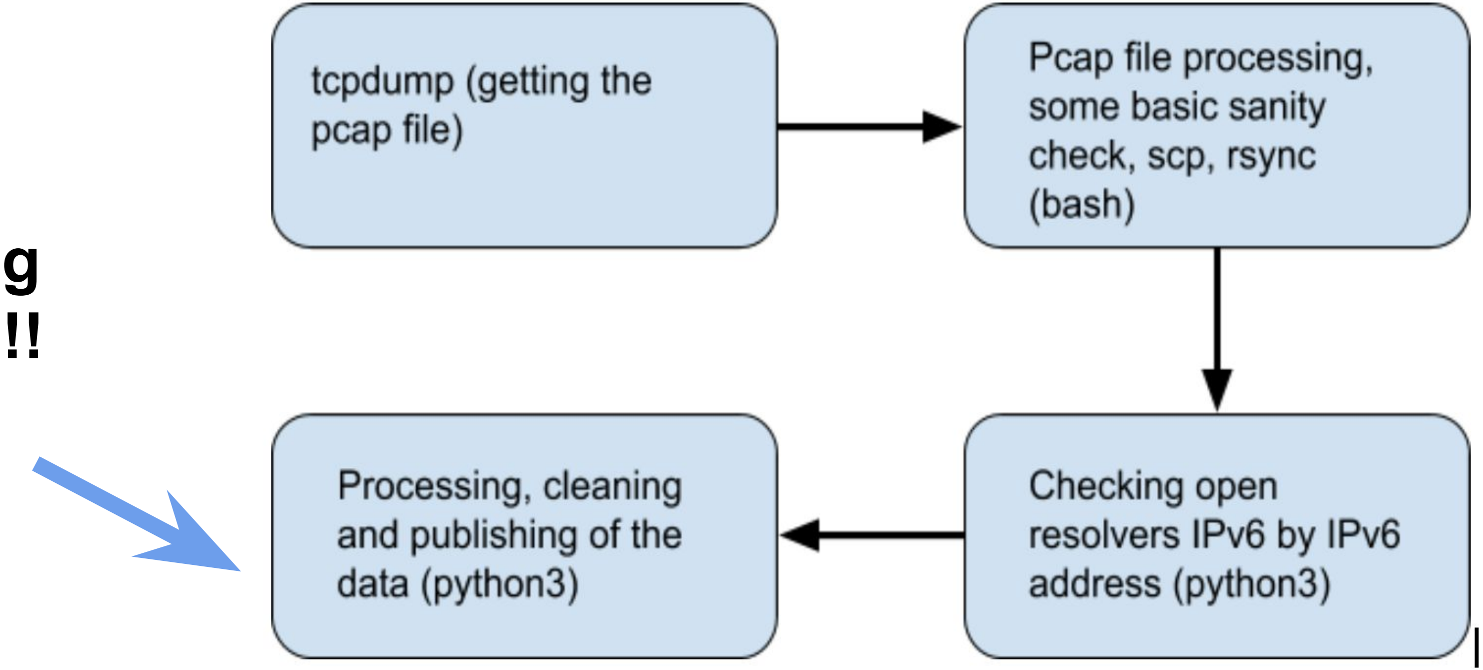
Pcap file processing, some basic sanity check, scp, rsync (bash)

Checking open resolvers IPv6 by IPv6 address (python3)

Here comes the real processing of the file resulting after previous steps. We identify if the IP is Open or not, type of error gotten (is case of), Owner ID, RIR  belonging

# Ok.., too much talk. What is the approach you previously mentioned ! (5/5)

**Ongoing phase !!!**



tcpdump (getting the pcap file)

Pcap file processing, some basic sanity check, scp, rsync (bash)

Checking open resolvers IPv6 by IPv6 address (python3)

Processing, cleaning and publishing of the data (python3)

# What are going to be the final results

.- # of IPv6 Resolvers identified
.- % of IPv6 Resolvers identified per RIR
.- % of IPv6 OPEN IPv6 Resolvers identified
.- % of IPv6 OPEN IPv6 Resolvers identified by RIR
.- We also expect to identify some more information

# Partial results

5112 DNS Servers

164 Open Resolvers

4975 Non-Open Resolvers

239 Resolvers in Lacnic region

1489 Resolvers in Arin region

3177 Resolvers in Ripe region

244 Resolvers in APNIC region

17 Resolvers in AFRINIC region

224 Non-Open Resolvers in Lacnic region

15 Open Resolvers in Lacnic region

1423 Non-Open Resolvers in Arin region

78 Open Resolvers in Arin region

17 Non-Open Resolvers in Afrinic region

0 Open Resolvers in Afrinic region

3149 Non-Open Resolvers in Ripe region

36 Open Resolvers in Ripe region

210 Non-Open Resolvers in Apnic region

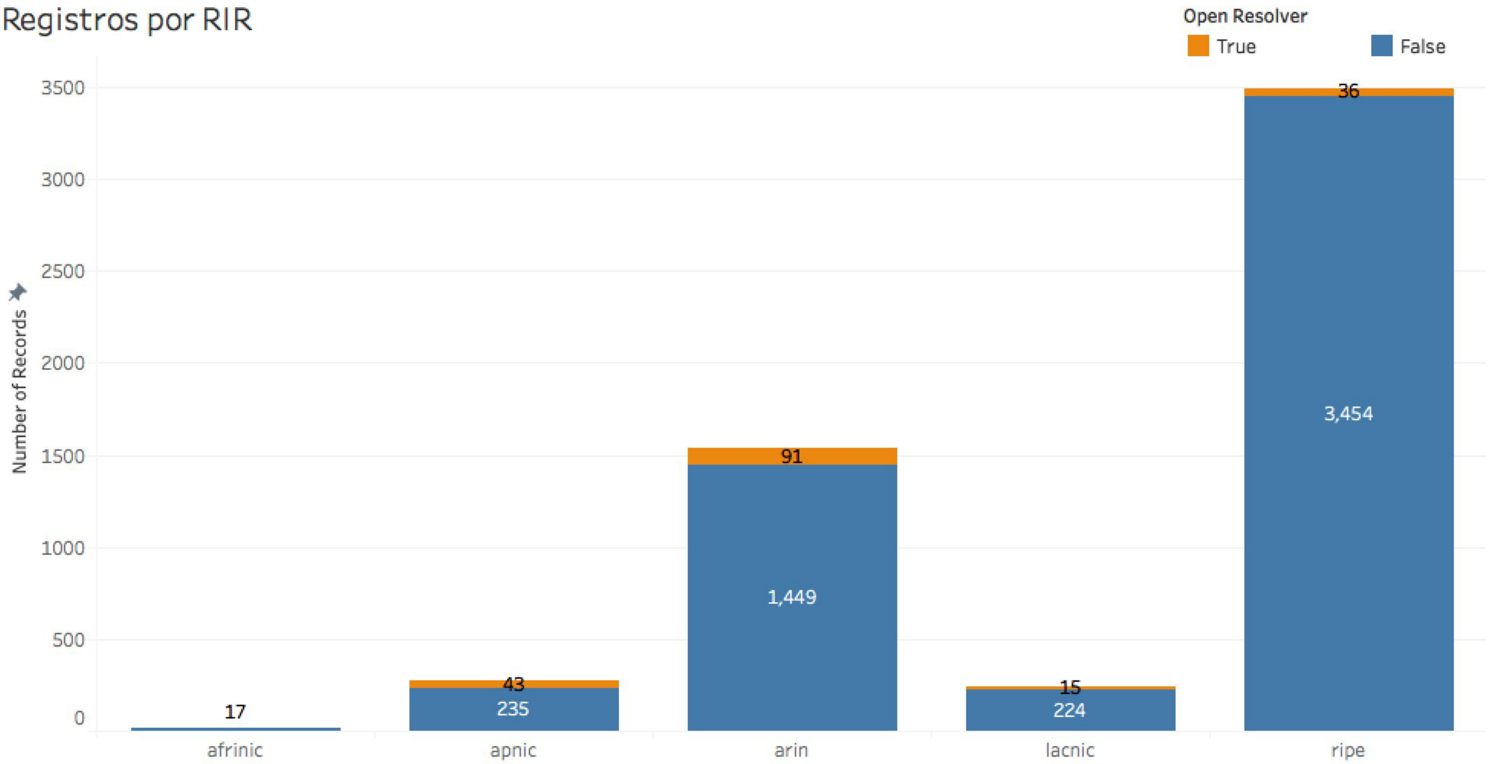
42 Open Resolvers in APNIC region

# Partial results

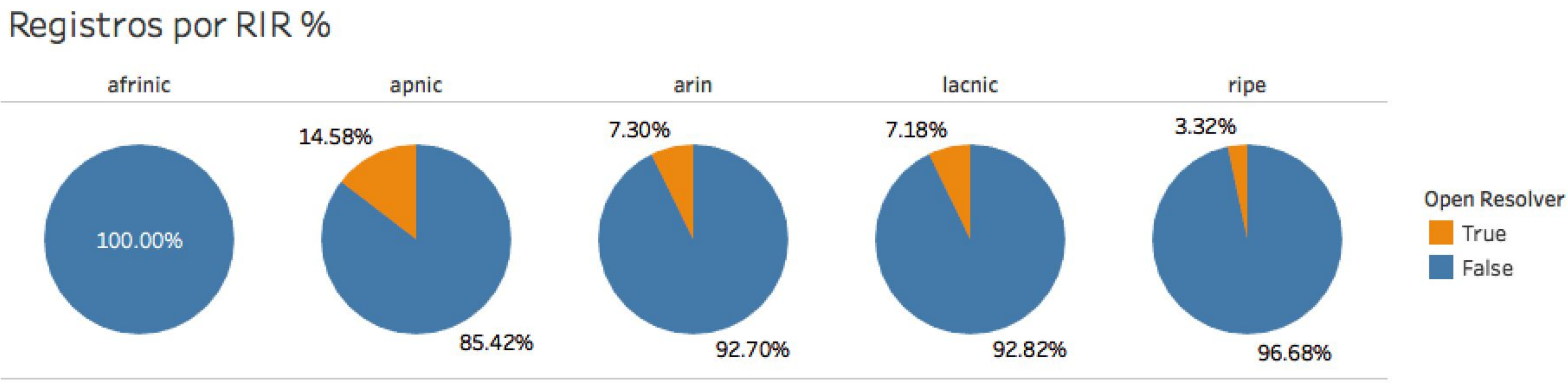
In Lacnic region we have already found the following Open Resolvers

```
2803:XXXX:0:XXXX::242 | noerror | VE-YYYY-LACNIC

2803:XXXX:4100:XXXX::5 | noerror | CO-YYYY-LACNIC

2803:XXXX:4200:XXXX::6 | noerror | CO-YYYY-LACNIC

2001:XXXX:3000:20:XXXX:7c1f:XXXX:f3c1 | noerror | MX-YYYY-LACNIC
```

# Partial results



# Partial results



# Miscellaneous finding

Number of resolvers per /64:

# Miscellaneous finding

Number of resolvers per /64:

2001:XXXX:XXXX:2/64 62

2a02:XXXX:0:XXXX/64 38

2a02:XXXX:0:XXXX/64 38

2001:XXXX:52:XXXX/64 34

2a02:XXXX:0:XXXX/64 32

# Utopy

.- Automation of the whole process (ongoing)
.- Alarm to notify owner ID about the Open Resolver (in few months)

# Questions / Comments