

# An Experiment in Implementing a Stateless TCP DNS Server

Geoff Huston

APNIC

# IP Networking 101

There are two major transport protocols in IP:

- TCP when reliable data transfer is needed
- UDP for simple lightweight transactions

# IP Networking 102

Coping with large responses for transactions – What happens when the response size exceeds the path MTU?

- Use UDP with IP level fragmentation and reassembly to rebuild the protocol data unit
  - but firewalls often drop trailing IP fragments
  - IPv6 UDP path MTU handling is not well suited to transaction apps
- Use TCP segmentation and reassembly to rebuild the protocol data unit
  - switching to TCP implies additional load on the server, introduces limits on server transaction throughput, and adds additional delay in the elapsed time for the transaction

# IP Networking 666

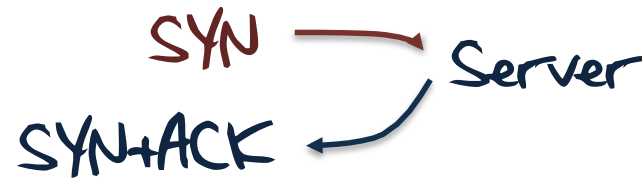
## Fire up the Bad Idea Factory

Why not combine UDP with TCP segmentation and reassembly?

- The client runs a conventional TCP application
- The server runs a stateless UDP-style application, but formats its output using TCP framing  
i.e. “Stateless TCP”

# The Server's Perspective

## 1. SYN Response



Flip the IP source and destination fields

Flip the TCP source and destination ports

Use any old sequence number

Offer a reasonable MSS (1220)

Offer no other TCP options

# The Server's Perspective

## 2. Request Response



Start with a sequence numbers given in the Request

Send an ACK

[Generate the response PDU]

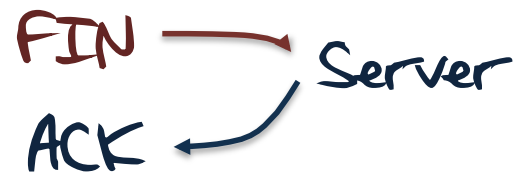
Chop the response into 512 octet segments – add TCP headers

Send the response packet train back to back

Send a FIN

# The Server's Perspective

## 3. FIN Response



Flip the IP addrs, TCP ports and ack/sequence fields  
increment ack field  
send ACK

# The Server's Perspective

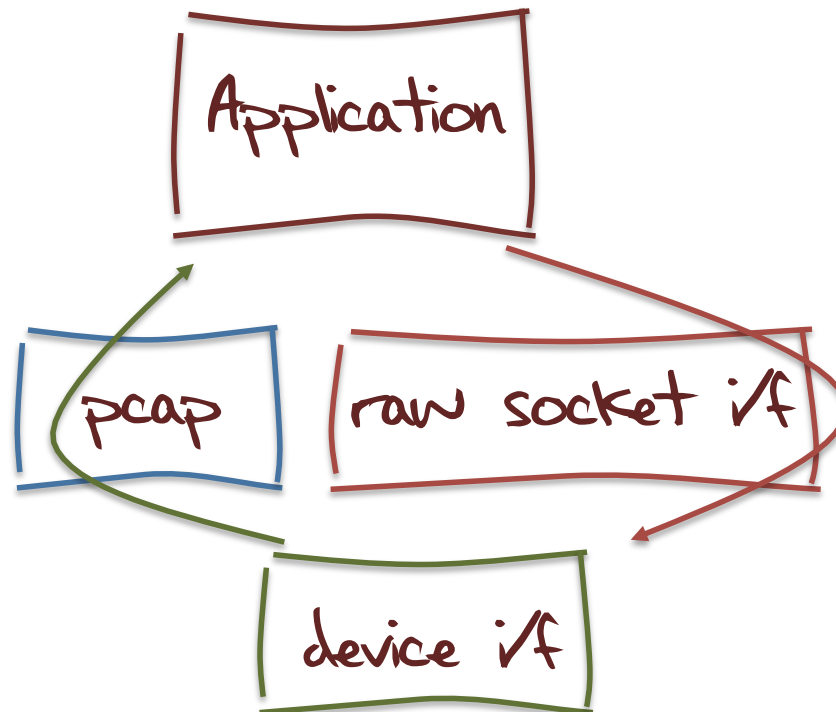
4. all else

No server response



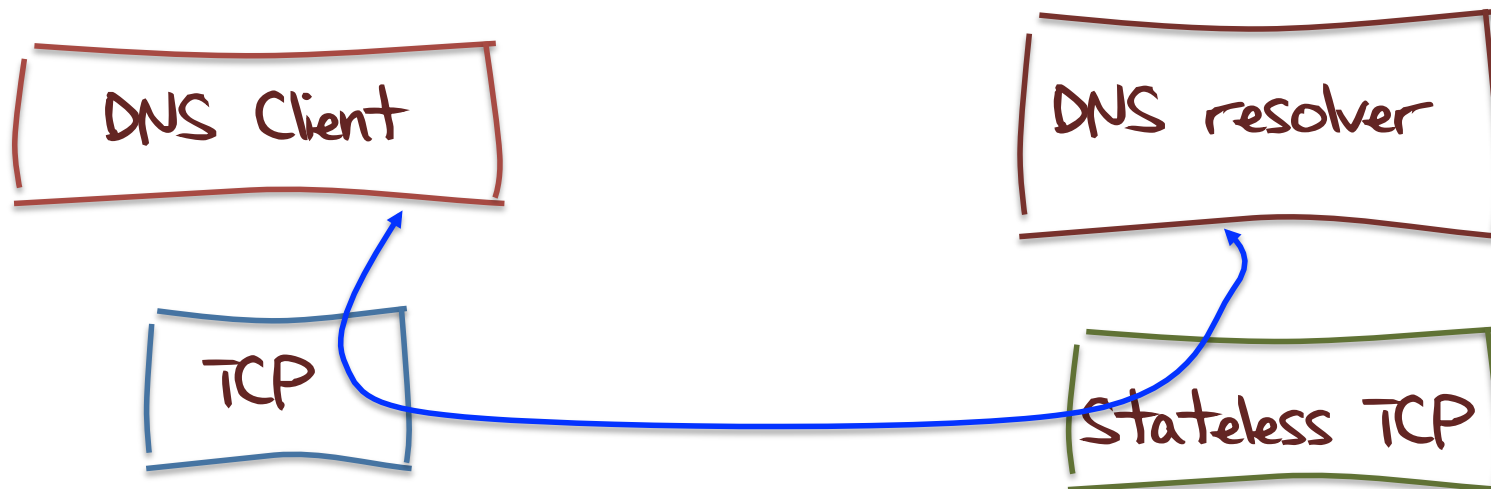
# Can this be coded?

A user space implementation of a stateless DNS TCP server that avoids kernel TCP processing



So far so good..

Can we use this approach to create a hybrid model of a TCP DNS client speaking to a stateless TCP DNS resolver?



# DNS and Stateless TCP

To test if this approach could work I used a prototype config of a stateless TCP facing the client, and a UDP referral to a DNS resolver as the back end



# It Worked!

\$ dig +tcp @server rand.apnic.net in any

```
client.55998 > server.domain: S, cksum 0x9159 (correct), 2201103970:2201103970(0) win 65535 <mss 1460>
server.domain > client.55998: S, cksum 0x82b9 (correct), 1256795928:1256795928(0) ack 2201103971 win 65535 <mss 1220>
client.55998 > server.domain: ., cksum 0x9986 (correct), 1:1(0) ack 1 win 65535
client.55998 > server.domain: P, cksum 0x41b2 (correct), 1:35(34) ack 1 win 6553530304+ ANY? rand.apnic.net. (32)
server.domain > client.55998: ., cksum 0x9964 (correct), 1:1(0) ack 35 win 65535
server.54054 > backend.domain: 30304+ ANY? rand.apnic.net. (32)
backend.domain > server.54054: 30304* q: ANY? rand.apnic.net. 6/0/2 rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: ., cksum 0x421a (correct), 1:232(231) ack 35 win 6553530304* q: ANY? rand.apnic.net. 6/0/2
    rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: F, cksum 0x987c (correct), 232:232(0) ack 35 win 65535
client.55998 > server.domain: ., cksum 0x987d (correct), 35:35(0) ack 232 win 65535
client.55998 > server.domain: ., cksum 0x987c (correct), 35:35(0) ack 233 win 65535
client.55998 > server.domain: F, cksum 0x987b (correct), 35:35(0) ack 233 win 65535
server.domain > client.55998: ., cksum 0x987c (correct), 232:232(0) ack 36 win 65535
```

# It Worked!

## 1. TCP handshake

dig +tcp @server rand.apnic.net in any

```
client.55998 > server.domain: S, cksum 0x9159 (correct), 2201103970:2201103970(0) win 65535 <mss 1460>
server.domain > client.55998: S, cksum 0x82b9 (correct), 1256795928:1256795928(0) ack 2201103971 win 65535 <mss 1220>
client.55998 > server.domain: ., cksum 0x9986 (correct), 1:1(0) ack 1 win 65535
client.55998 > server.domain: P, cksum 0x41b2 (correct), 1:35(34) ack 1 win 6553530304+ ANY? rand.apnic.net. (32)
server.domain > client.55998: ., cksum 0x9964 (correct), 1:1(0) ack 35 win 65535
server.54054 > backend.domain: 30304+ ANY? rand.apnic.net. (32)
backend.domain > server.54054: 30304* q: ANY? rand.apnic.net. 6/0/2 rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: ., cksum 0x421a (correct), 1:232(231) ack 35 win 6553530304* q: ANY? rand.apnic.net. 6/0/2
    rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: F, cksum 0x987c (correct), 232:232(0) ack 35 win 65535
client.55998 > server.domain: ., cksum 0x987d (correct), 35:35(0) ack 232 win 65535
client.55998 > server.domain: ., cksum 0x987c (correct), 35:35(0) ack 233 win 65535
client.55998 > server.domain: F, cksum 0x987b (correct), 35:35(0) ack 233 win 65535
server.domain > client.55998: ., cksum 0x987c (correct), 232:232(0) ack 36 win 65535
```

# It Worked!

## 2. TCP request and referral to UDP DNS backend

```
dig +tcp @server rand.apnic.net in any
```

```
client.55998 > server.domain: S, cksum 0x9159 (correct), 2201103970:2201103970(0) win 65535 <mss 1460>
server.domain > client.55998: S, cksum 0x82b9 (correct), 1256795928:1256795928(0) ack 2201103971 win 65535 <mss 1220>
client.55998 > server.domain: ., cksum 0x9986 (correct), 1:1(0) ack 1 win 65535
client.55998 > server.domain: P, cksum 0x41b2 (correct), 1:35(34) ack 1 win 6553530304+ ANY? rand.apnic.net. (32)
server.domain > client.55998: ., cksum 0x9964 (correct), 1:1(0) ack 35 win 65535
server.54054 > backend.domain: 30304+ ANY? rand.apnic.net. (32)
backend.domain > server.54054: 30304* q: ANY? rand.apnic.net. 6/0/2 rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: ., cksum 0x421a (correct), 1:232(231) ack 35 win 6553530304* q: ANY? rand.apnic.net. 6/0/2
rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: F, cksum 0x987c (correct), 232:232(0) ack 35 win 65535
client.55998 > server.domain: ., cksum 0x987d (correct), 35:35(0) ack 232 win 65535
client.55998 > server.domain: ., cksum 0x987c (correct), 35:35(0) ack 233 win 65535
client.55998 > server.domain: F, cksum 0x987b (correct), 35:35(0) ack 233 win 65535
server.domain > client.55998: ., cksum 0x987c (correct), 232:232(0) ack 36 win 65535
```

# It Worked!

## 3. TCP response to client

```
dig +tcp @server rand.apnic.net in any
```

```
client.55998 > server.domain: S, cksum 0x9159 (correct), 2201103970:2201103970(0) win 65535 <mss 1460>
server.domain > client.55998: S, cksum 0x82b9 (correct), 1256795928:1256795928(0) ack 2201103971 win 65535 <mss 1220>
client.55998 > server.domain: ., cksum 0x9986 (correct), 1:1(0) ack 1 win 65535
client.55998 > server.domain: P, cksum 0x41b2 (correct), 1:35(34) ack 1 win 6553530304+ ANY? rand.apnic.net. (32)
server.domain > client.55998: ., cksum 0x9964 (correct), 1:1(0) ack 35 win 65535
server.54054 > backend.domain: 30304+ ANY? rand.apnic.net. (32)
backend.domain > server.54054: 30304* q: ANY? rand.apnic.net. 6/0/2 rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: ., cksum 0x421a (correct), 1:232(231) ack 35 win 6553530304* q: ANY? rand.apnic.net. 6/0/2
    rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: F, cksum 0x987c (correct), 232:232(0) ack 35 win 65535
client.55998 > server.domain: ., cksum 0x987d (correct), 35:35(0) ack 232 win 65535
client.55998 > server.domain: ., cksum 0x987c (correct), 35:35(0) ack 233 win 65535
client.55998 > server.domain: F, cksum 0x987b (correct), 35:35(0) ack 233 win 65535
server.domain > client.55998: ., cksum 0x987c (correct), 232:232(0) ack 36 win 65535
```

# It Worked!

## 4. FIN close

```
dig +tcp @server rand.apnic.net in any
```

```
client.55998 > server.domain: S, cksum 0x9159 (correct), 2201103970:2201103970(0) win 65535 <mss 1460>
server.domain > client.55998: S, cksum 0x82b9 (correct), 1256795928:1256795928(0) ack 2201103971 win 65535 <mss 1220>
client.55998 > server.domain: ., cksum 0x9986 (correct), 1:1(0) ack 1 win 65535
client.55998 > server.domain: P, cksum 0x41b2 (correct), 1:35(34) ack 1 win 6553530304+ ANY? rand.apnic.net. (32)
server.domain > client.55998: ., cksum 0x9964 (correct), 1:1(0) ack 35 win 65535
server.54054 > backend.domain: 30304+ ANY? rand.apnic.net. (32)
backend.domain > server.54054: 30304* q: ANY? rand.apnic.net. 6/0/2 rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: ., cksum 0x421a (correct), 1:232(231) ack 35 win 6553530304* q: ANY? rand.apnic.net. 6/0/2
    rand.apnic.net. SOA mirin.apnic.net. research.apnic.net.
    2009051502 3600 900 3600000 3600, rand.apnic.net. NS mirin.apnic.net., rand.apnic.net. NS sec3.apnic.net.,
    rand.apnic.net. MX kombu.apnic.net. 100, rand.apnic.net. MX karashi.apnic.net. 200, rand.apnic.net. MX
    fennel.apnic.net. 300 ar: sec3.apnic.net. A sec3.apnic.net, sec3.apnic.net. AAAA sec3.apnic.net (229)
server.domain > client.55998: F, cksum 0x987c (correct), 232:232(0) ack 35 win 65535
client.55998 > server.domain: ., cksum 0x987d (correct), 35:35(0) ack 232 win 65535
client.55998 > server.domain: ., cksum 0x987c (correct), 35:35(0) ack 233 win 65535
client.55998 > server.domain: F, cksum 0x987b (correct), 35:35(0) ack 233 win 65535
server.domain > client.55998: ., cksum 0x987c (correct), 232:232(0) ack 36 win 65535
```



But ...

Its just like UDP in almost every respect:  
no reliability, no flow control, and  
absolutely no manners whatsoever!

And its really a Bad Idea!

# Code and ACK

The FreeBSD code used here for the Stateless DNS proxy can be found at: <http://www.potaroo.net/tools/useless>

This Bad Idea was cooked up in collaboration with George Michaelson