CSCI-1680
DNS

Nick DeMarinis

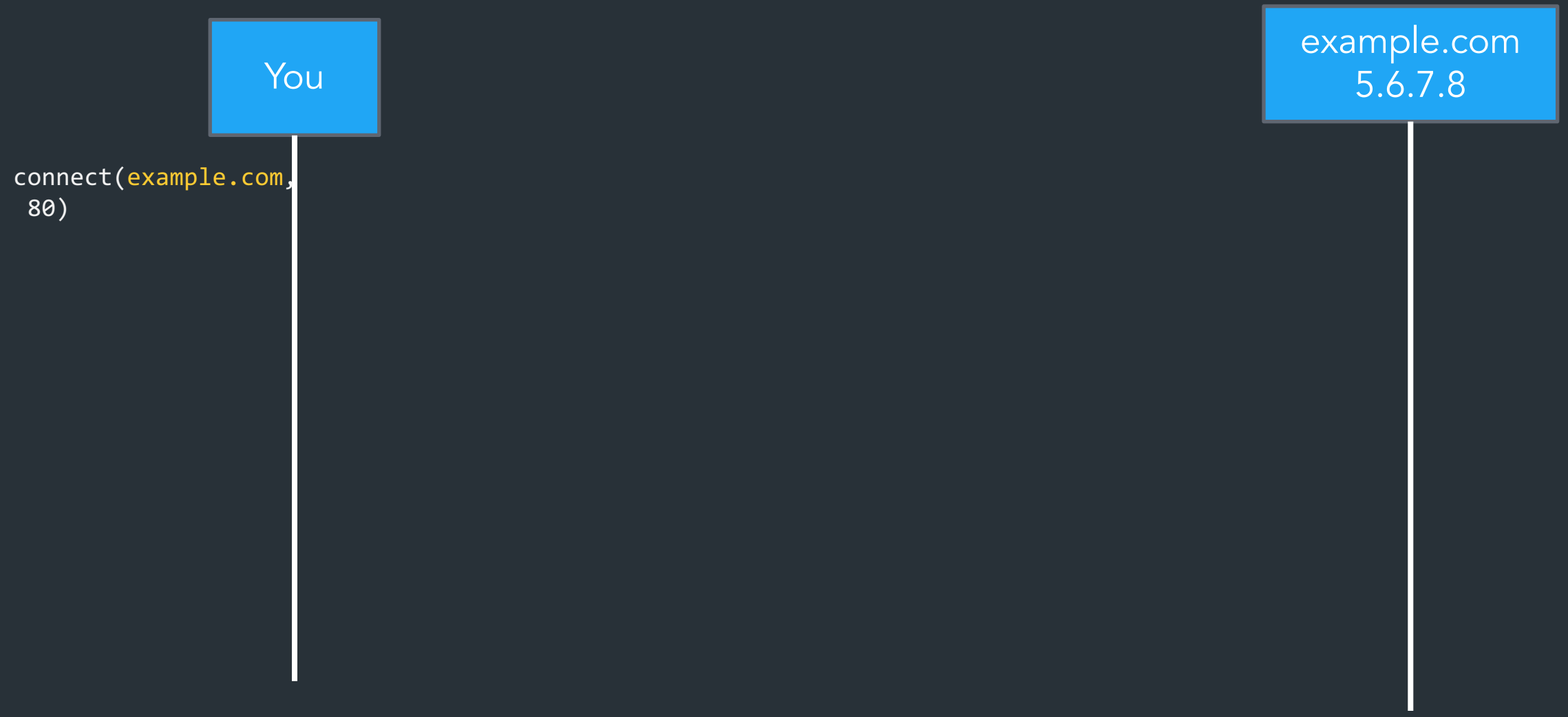
Based partly on lecture notes by Rodrigo Fonseca, Scott Shenker and John Jannotti

Administrivia

- TCP milestone II: sign up for a meeting soon (by Monday at latest—don't stress about having it all done)
- TCP gearup III: tonight (11/9), 5-7pm
- HW4: TBA, but due after TCP

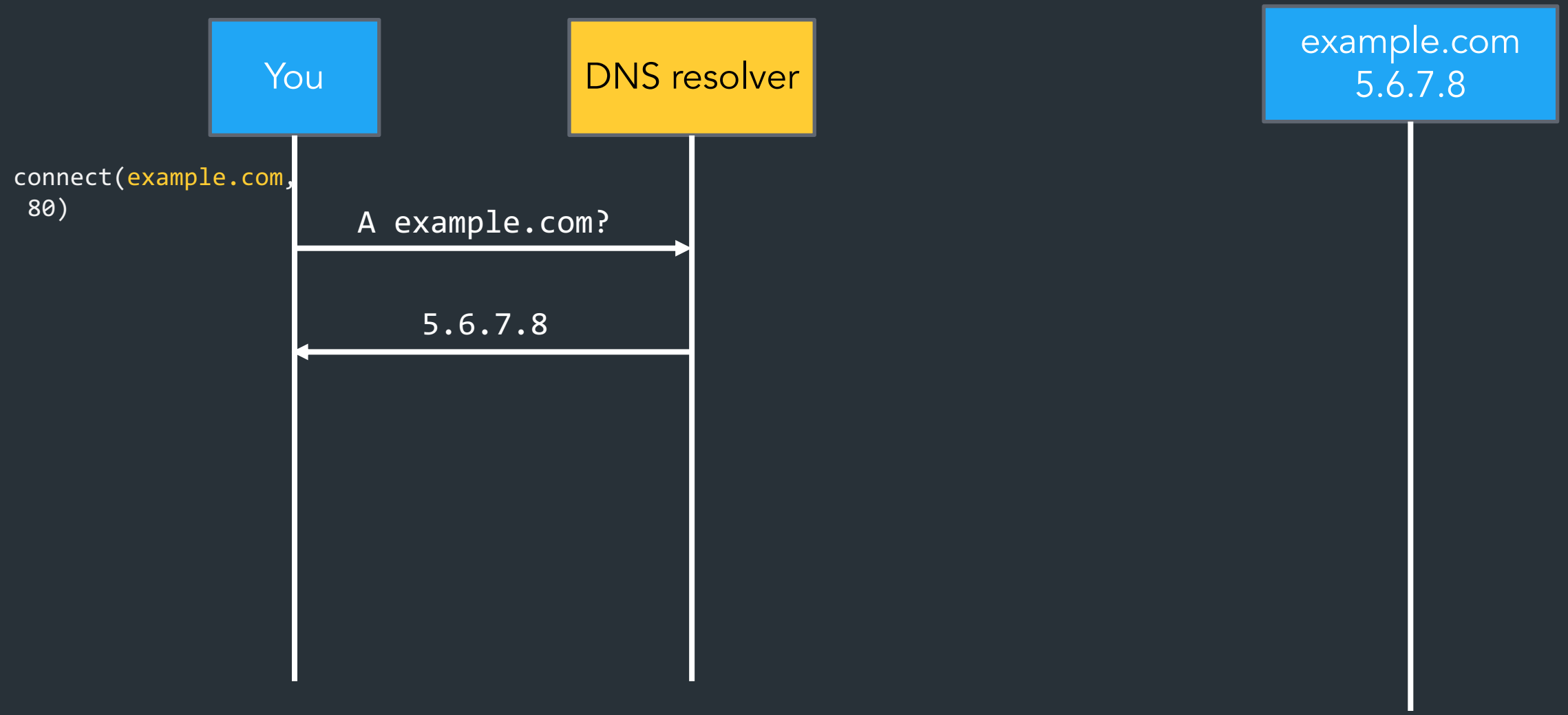
The story so far

POV: You want to connect to some website



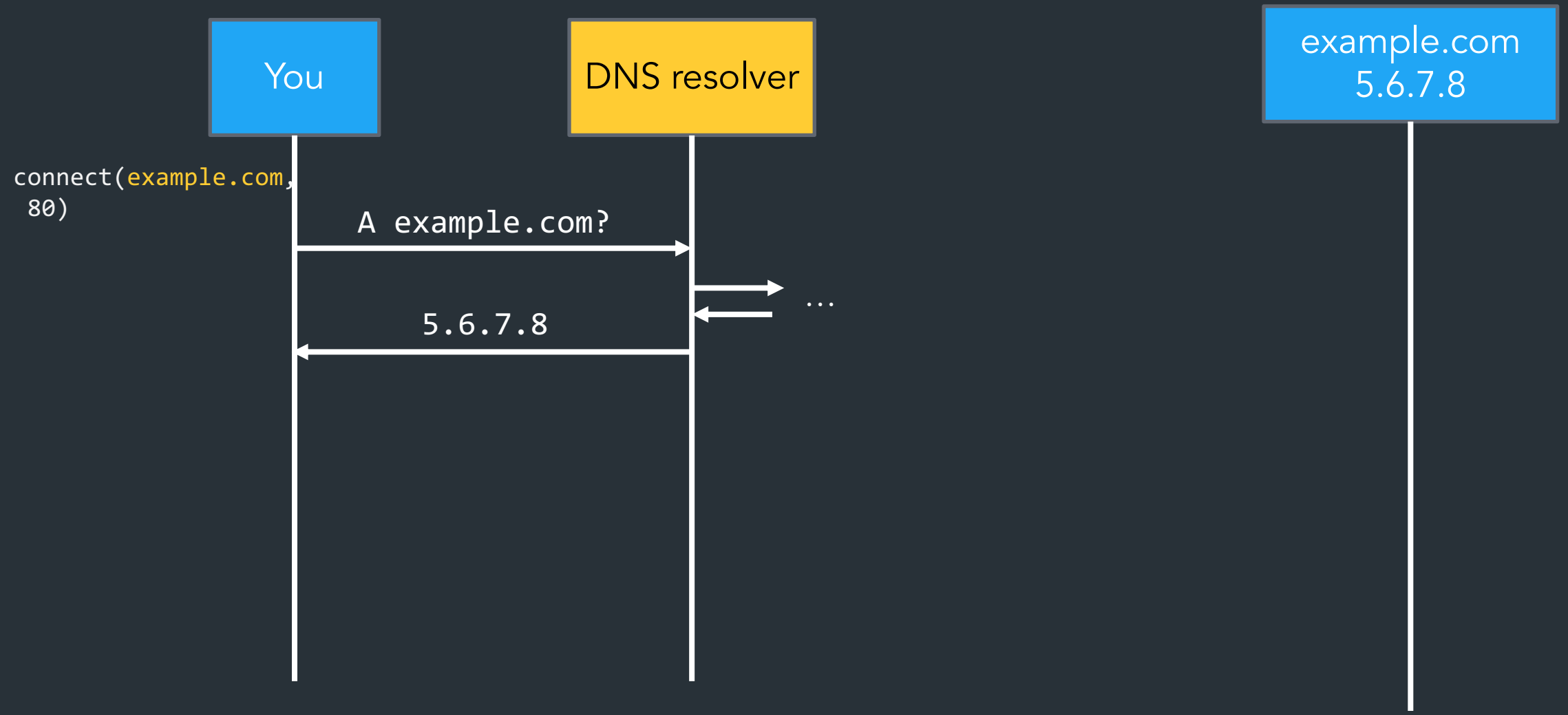
The story so far

POV: You want to connect to some website



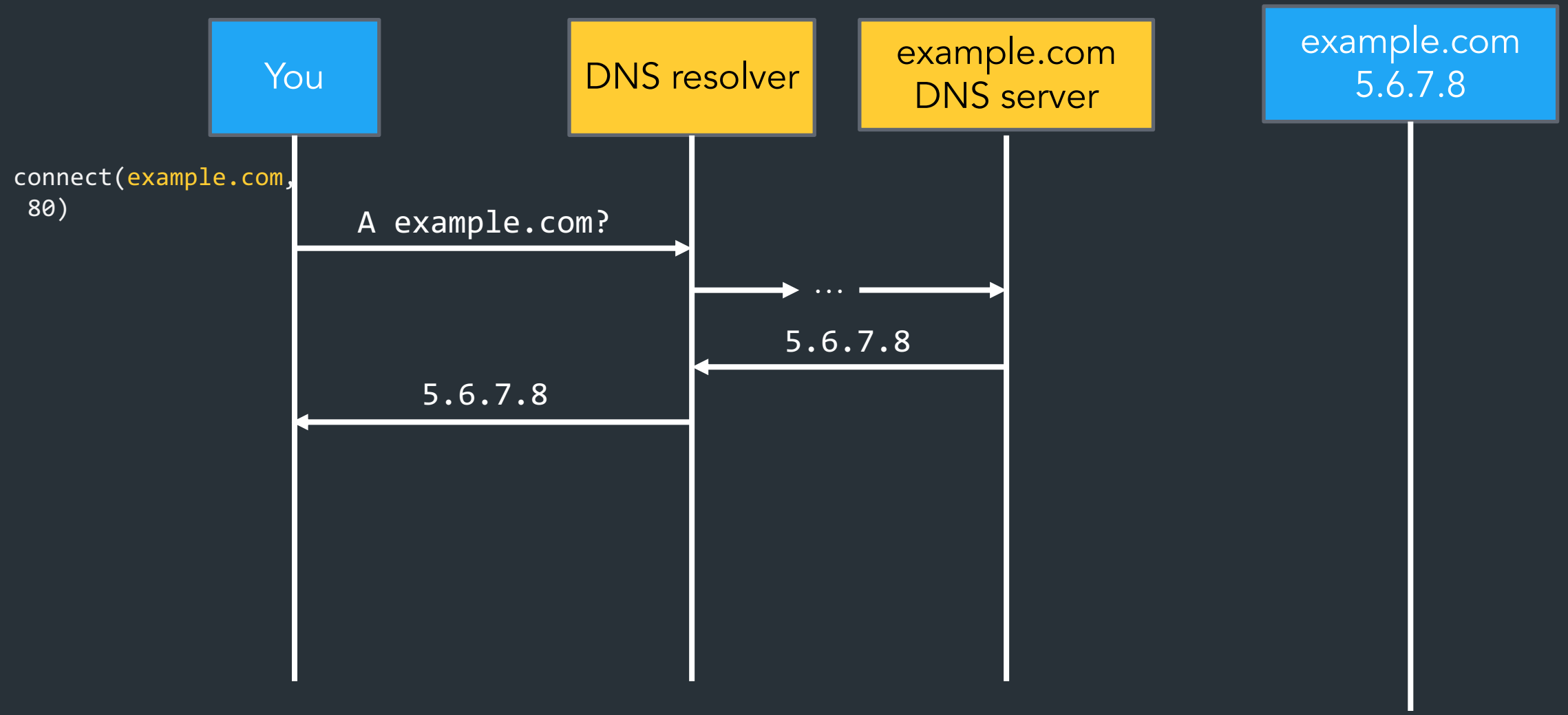
The story so far

POV: You want to connect to some website



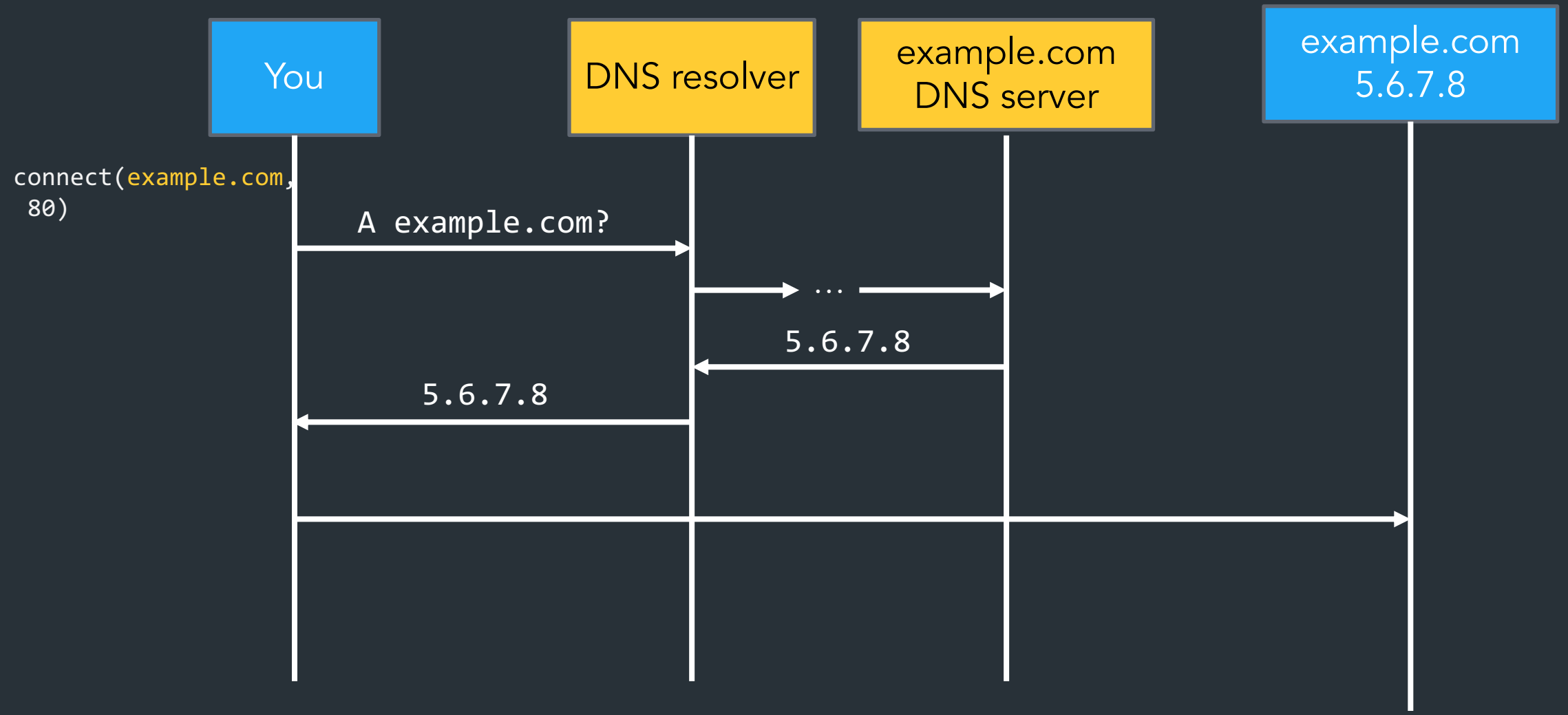
The story so far

POV: You want to connect to some website



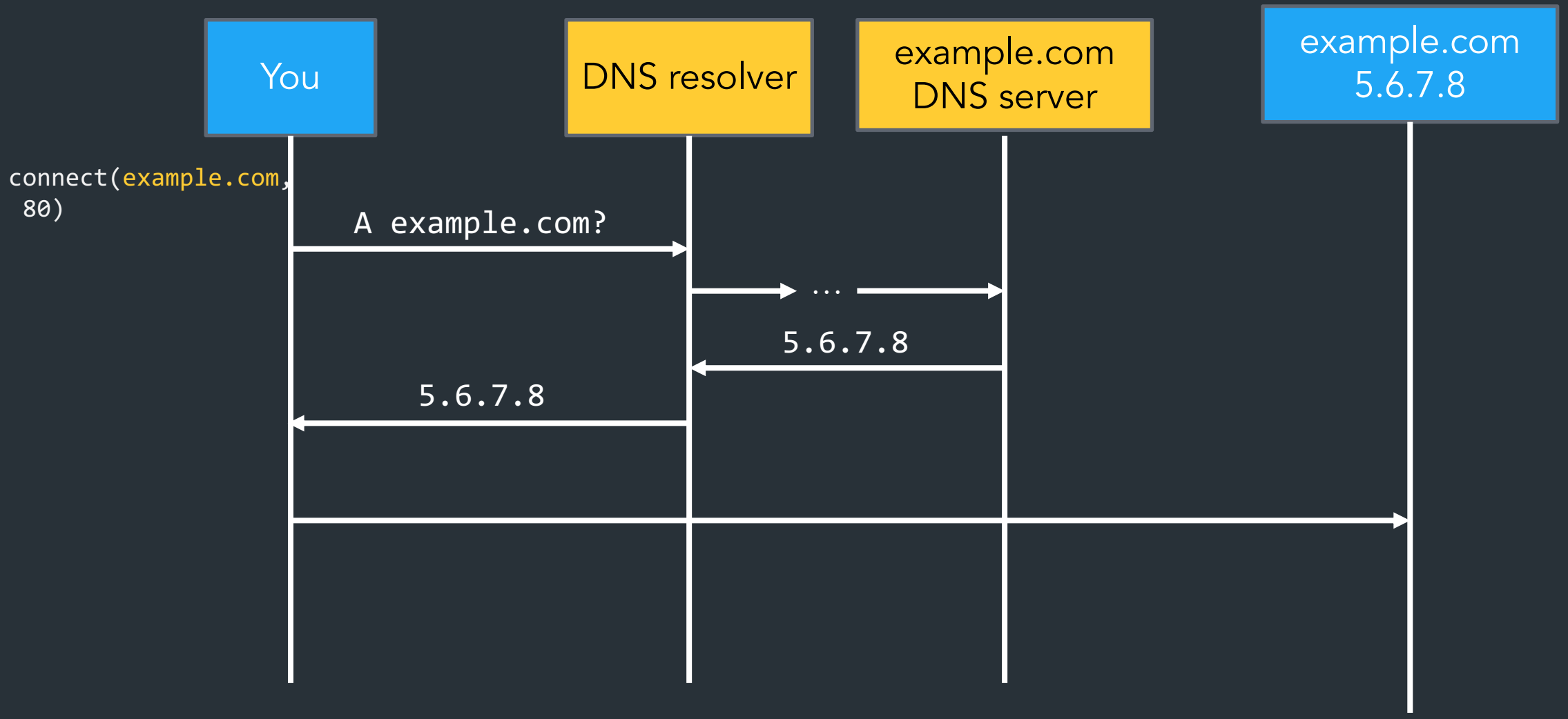
The story so far

POV: You want to connect to some website



Warmup

Q: If the randomsite.com's DNS server goes down, can another DNS server still resolve randomsite.com?



How it scales: caching

DNS Resolvers cache responses to avoid doing recursive/iterative queries

- Many messages => extra computation, extra latency

```
$ dig cs.brown.edu @10.1.1.10
;; ANSWER SECTION:
cs.brown.edu.           1800      IN      A       128.148.32.12
```

How it scales: caching

DNS Resolvers cache responses to avoid doing recursive/iterative queries

- Many messages => extra computation, extra latency

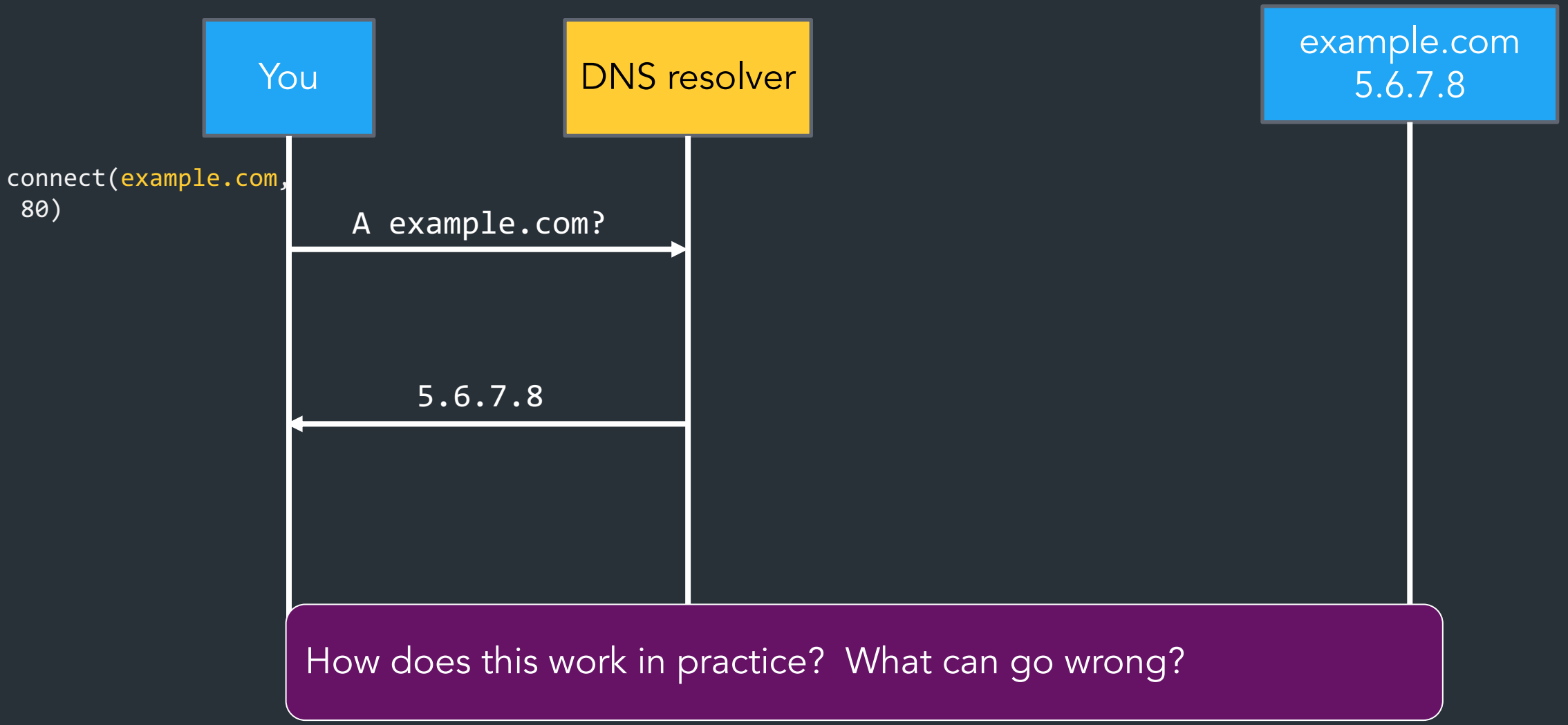
How long to cache?

=> Every record has a TTL (in seconds), delete when it expires

```
$ dig cs.brown.edu @10.1.1.10
;; ANSWER SECTION:
cs.brown.edu.           1800      IN      A       128.148.32.12
```

```
$ dig cs.brown.edu @10.1.1.10
; <<>> DiG 9.10.6 <<>> cs.brown.edu @10.1.1.10
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8536
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1


;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1220
;; QUESTION SECTION:
;cs.brown.edu. IN A


;; ANSWER SECTION:
cs.brown.edu.           1800      IN      A       128.148.32.12


;; Query time: 69 msec
;; SERVER: 10.1.1.10#53(10.1.1.10)
;; WHEN: Tue Apr 19 09:03:39 EDT 2022
;; MSG SIZE  rcvd: 57
```

Today


You
DNS resolver
example.com
5.6.7.8
connect(example.com, 80)
A example.com?
5.6.7.8
How does this work in practice? What can go wrong?

How it scales: caching

DNS Resolvers cache responses to avoid doing recursive/iterative queries

- Many messages => extra computation, extra latency

How long to cache?

=> Every record has a TTL (in seconds), delete when it expires

```
$ dig cs.brown.edu @10.1.1.10
;; ANSWER SECTION:
cs.brown.edu.           1800      IN      A       128.148.32.12
```

Related: redundant services via DNS

Can return multiple answers for one record
 => If a client can't connect to first result, can try next one

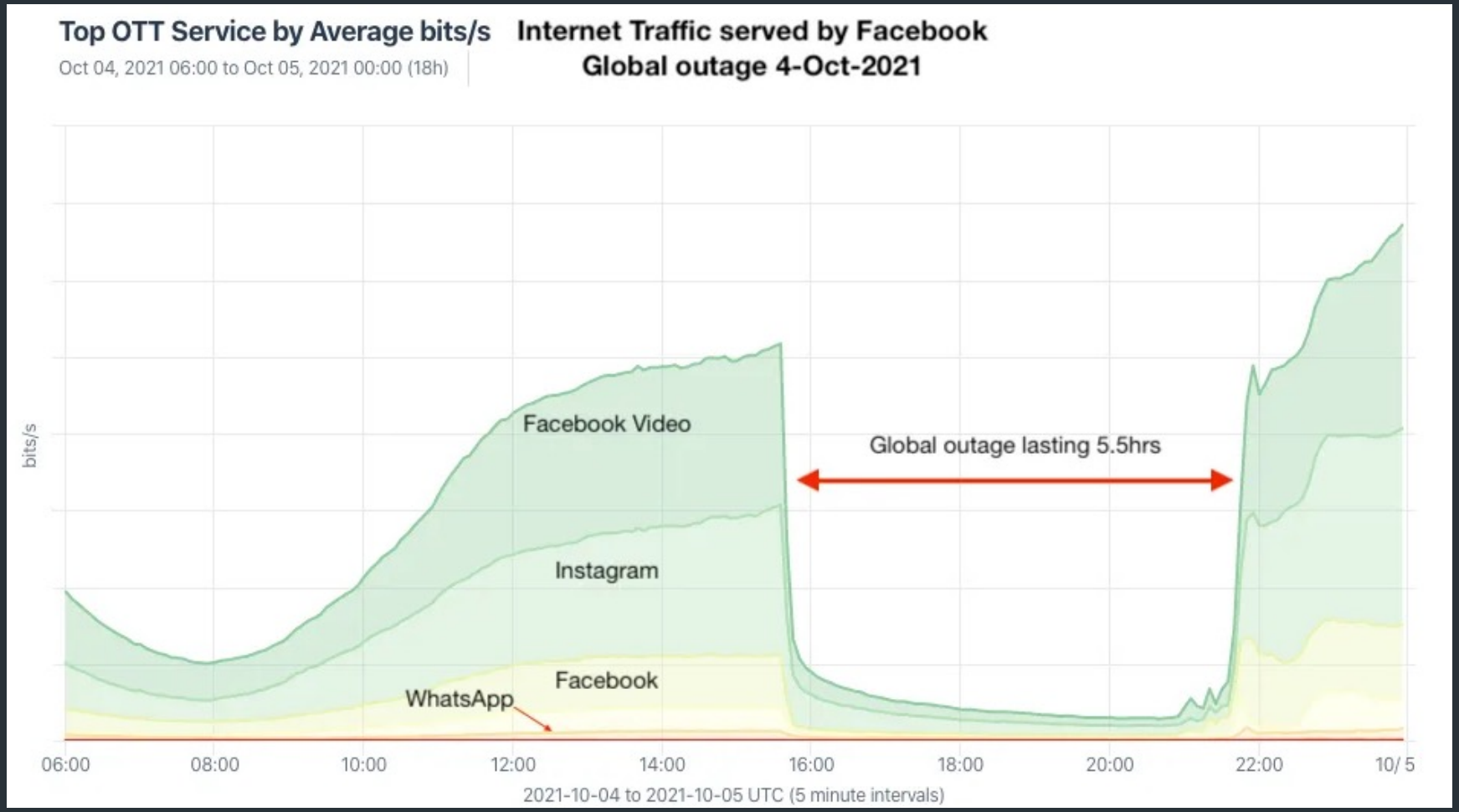
```
$ dig nytimes.com

;; ANSWER SECTION:
nytimes.com. 111 IN A 151.101.65.164
nytimes.com. 111 IN A 151.101.1.164
nytimes.com. 111 IN A 151.101.129.164
nytimes.com. 111 IN A 151.101.193.164


;; Query time: 40 msec
;; SERVER: 10.1.1.10#53(10.1.1.10)
;; WHEN: Thu Nov 09 08:42:41 EST 2023
;; MSG SIZE  rcvd: 104
```

DNS server usually shuffles answers on each response—why?

Facebook DNS outage (2021)

BGP configuration bug: Facebook withdraws all routes for its DNS servers to the Internet

=> Facebook DNS unreachable—not even Facebook could access their systems!



Traffic graph

Many writeups here

```
user@host$ dig @1.1.1.1 facebook.com # CloudFlare
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 5153
;facebook.com.                  IN      A
user@host$ dig @8.8.8.8 facebook.com # Google Public DNS
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 43224
;facebook.com.                  IN      A
user@host$ dig @208.67.222.222 facebook.com # OpenDNS
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 7643
;facebook.com.                  IN      A
user@host$ dig @176.103.130.130 facebook.com # AdGuard
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 5434
;facebook.com.                  IN      A
```

Source

DNS record types

RR Type	Purpose	Example
A	IPv4 Address	`128.148.56.2`
AAAA	IPv6 Address	`2001:470:8956:20::1`

More: https://en.wikipedia.org/wiki/List_of_DNS_record_types

DNS record types

RR Type	Purpose	Example
A	IPv4 Address	`128.148.56.2`
AAAA	IPv6 Address	`2001:470:8956:20::1`
CNAME	Specifies an alias ("Canonical name")	`systems.cs.brown.edu. 86400 IN CNAME systems-v3.cs.brown.edu.` `systems-v3.cs.brown.edu. 86400 IN A 128.148.36.51`
NS	DNS servers for a domain	`cs.brown.edu. 86400 IN NS br1.brown.edu`
MX	Mail servers	`MX <priority> <ip>` `eg. MX 10 1.2.3.4`
SOA	Start of authority	Information about who owns a zone
PTR	Reverse IP lookup	`7.34.148.128.in-addr.arpa. 86400 IN PTR quanto.cs.brown.edu.`
SRV	How to reach specific services (eg. host, port)	`_minecraft._tcp.example.net 3600` `SRV <priority> <weight> <port> <server IP>`

More: https://en.wikipedia.org/wiki/List_of_DNS_record_types

Reverse DNS

What if we want to map IP address => domain name?

Reverse DNS

What if we want to map IP address => domain name?

Leverages hierarchy in IP addresses, but in reverse

=> How? reverse the numbers: 12.32.148.128, then look that up

What happens when you register a new domain?

Registering a new domain

Your new startup helpme.com

- Get a block of addresses from ISP
  - Say 212.44.9.0/24
- Register helpme.com at namecheap.com (for ex.)
  - Provide name and address of your authoritative name server (primary and secondary)
  - Registrar inserts RR pair into the **.com** TLD server:
    - helpme.com NS dns1.helpme.com
    - dns1.helpme.com A 212.44.9.120
- Configure your authoritative server (dns1.helpme.com)
  - Type A record for www.helpme.com
  - Type MX record for helpme.com

Registering a new domain

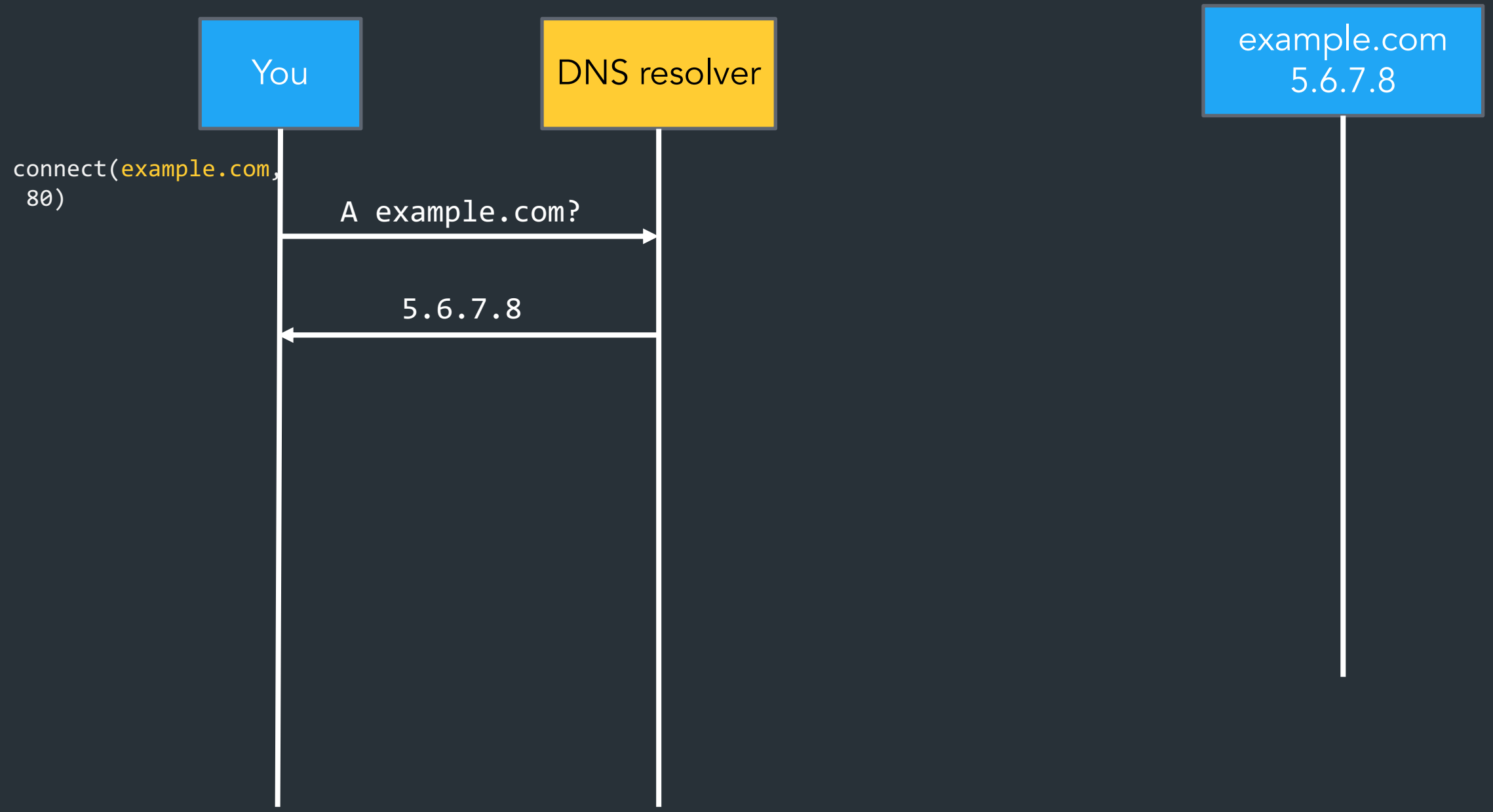
Your new startup helpme.com

- Get a block of addresses from ISP
  - Say 212.44.9.0/24
- Register helpme.com at namecheap.com (for ex.)
  - Provide name and address of your authoritative name server (primary and secondary)
  - Registrar inserts RR pair into the **.com** TLD server:
    - helpme.com NS dns1.helpme.com
    - dns1.helpme.com A 212.44.9.120
- Configure your authoritative server (dns1.helpme.com)
  - Type A record for www.helpme.com
  - Type MX record for helpme.com

Inserting a Record in DNS, cont

- Need to provide reverse PTR bindings
  - E.g., 212.44.9.120 -> dns1.helpme.com
- Configure your dns server to serve the 9.44.212.in-addr.arpa zone
  - Need to add a record of this NS into the parent zone (44.212.in-addr.arpa)
- Insert the bindings into the 9.44.212.in-addr.arpa zone

What can go wrong?

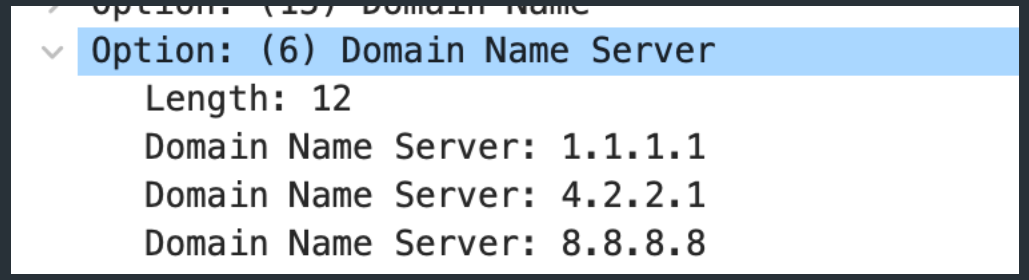

You
DNS resolver
example.com
5.6.7.8
connect(example.com,
80)
A example.com?
5.6.7.8

DNS Protocol

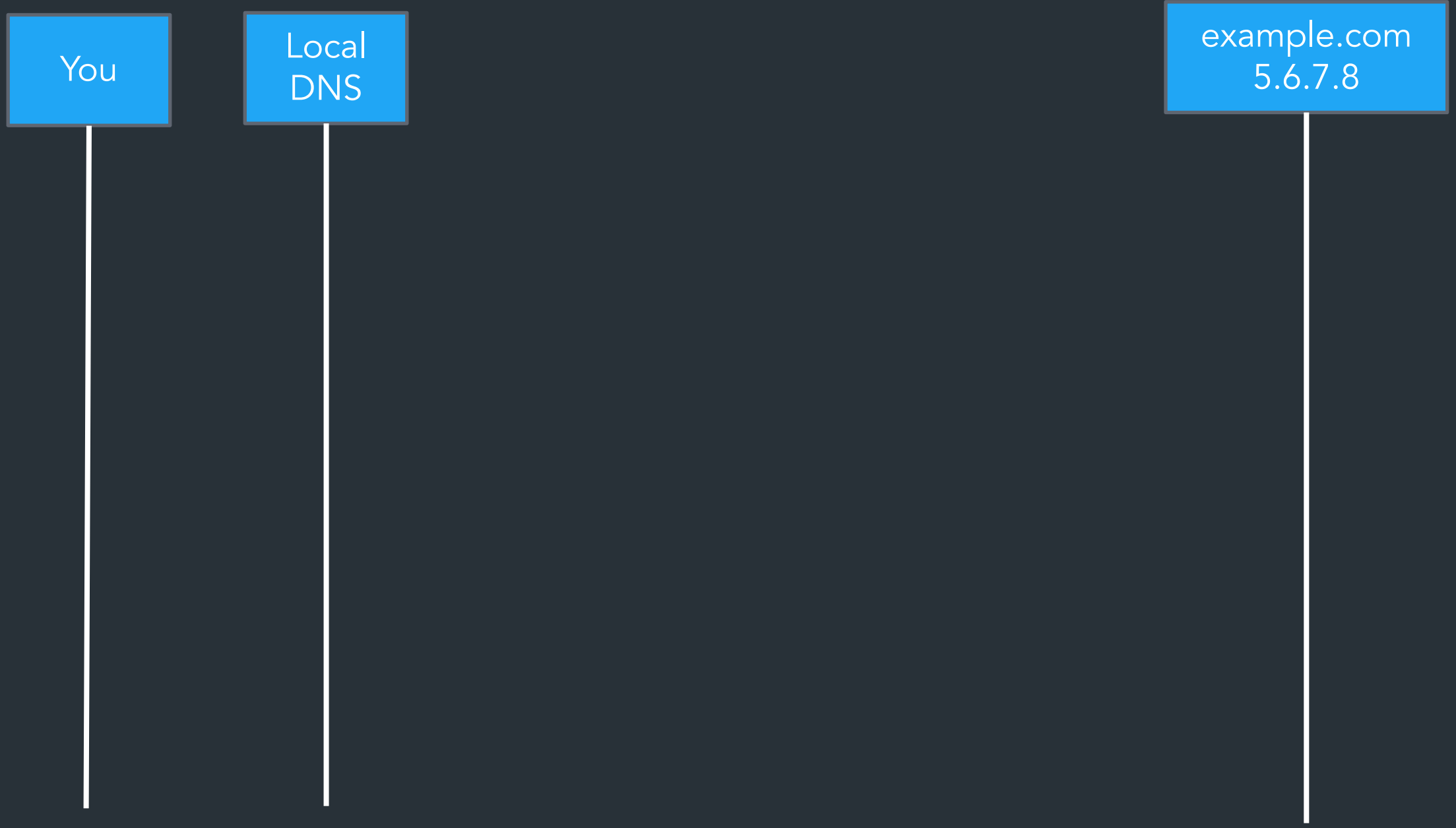
- TCP/UDP port 53
- Most traffic uses UDP
  - Lightweight protocol has 512 byte message limit
  - Can run over TCP (more on this later)
- A few options to request recursive queries, …

DNS Security

- You go to starbucks, how does your browser find www.google.com?
  - Ask local name server, obtained from DHCP

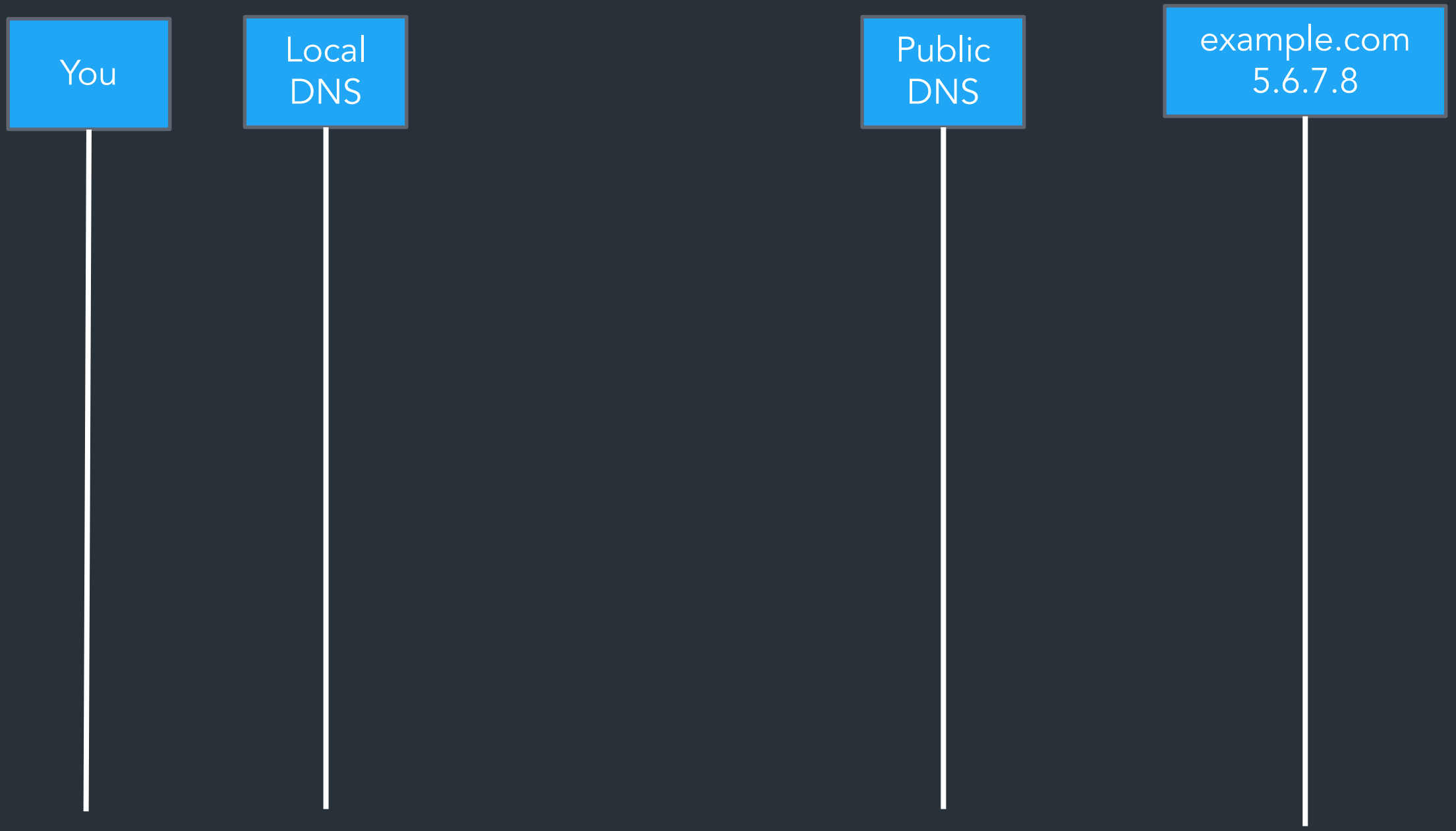


- Can you trust this DNS server?


You
Local DNS
example.com
5.6.7.8

Great Firewall of CIT

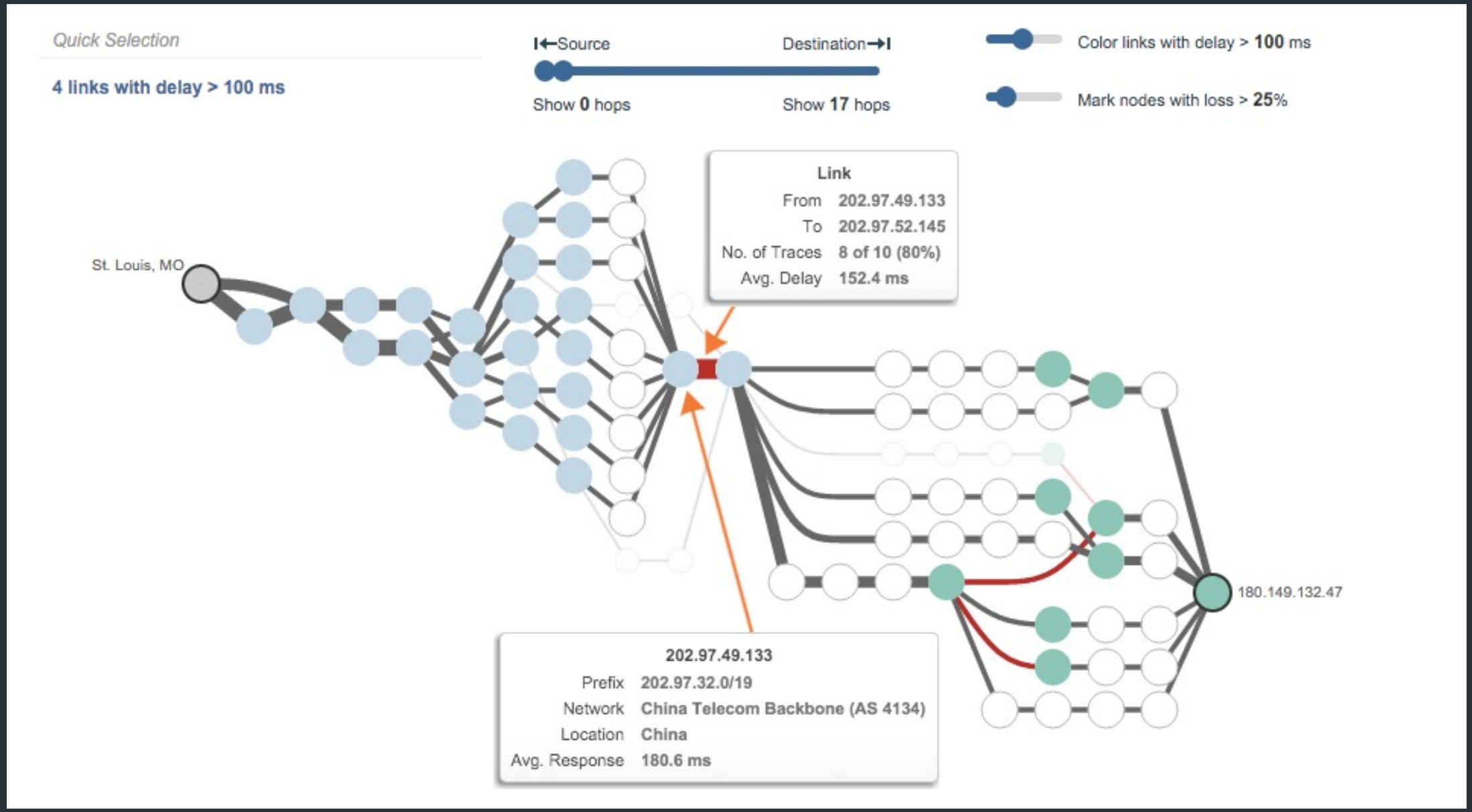
If attacker is on the path (say, it is the ISP, or a malicious version of TStaff), what could they do?


You
Local DNS
Public DNS
example.com
5.6.7.8

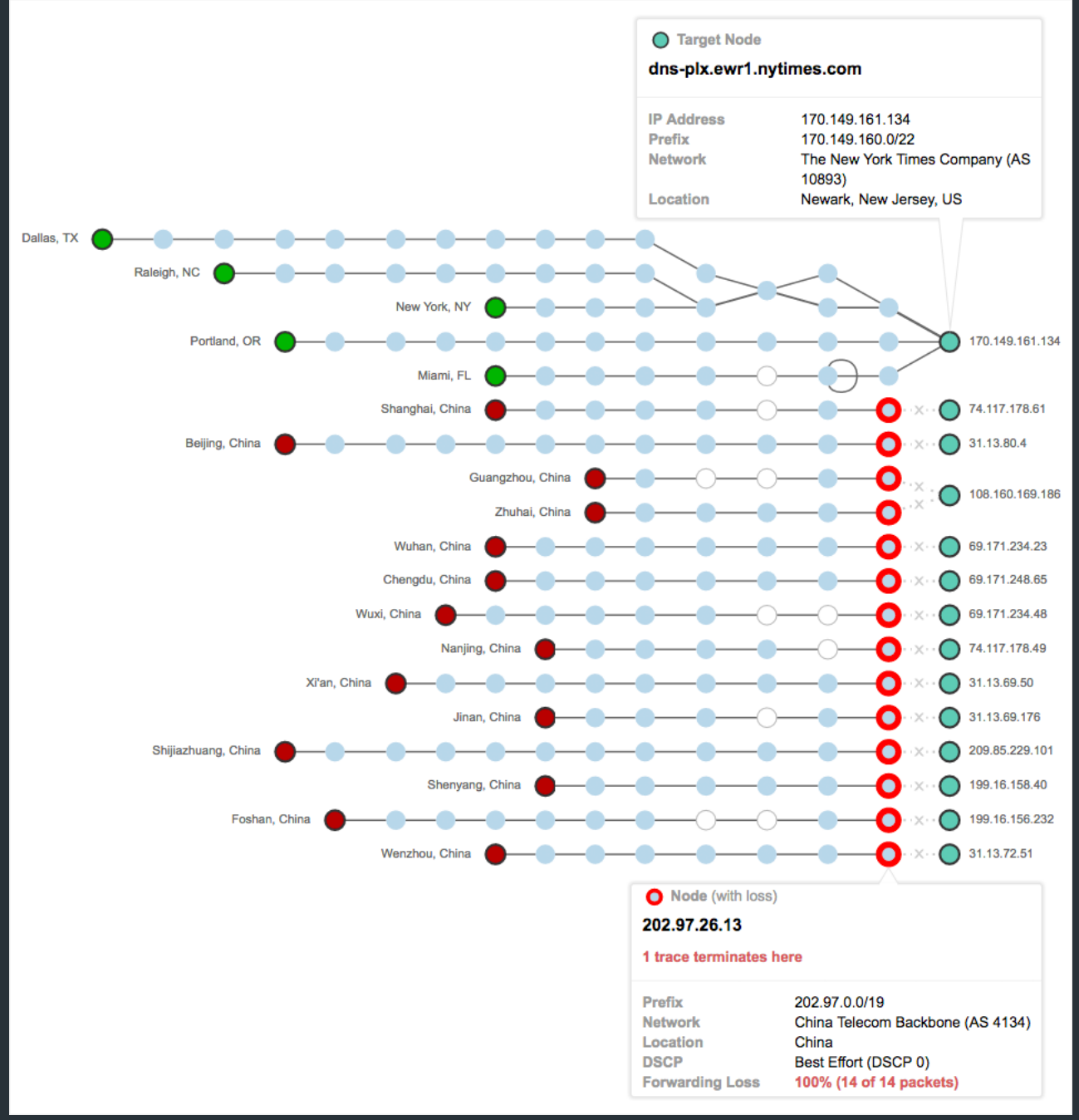
Great Firewall of CIT

If attacker is on the path (say, it is the ISP, or a malicious version of TStaff), what could they do?

- Can sniff all DNS queries
- Send fake responses back first
- Could do this selectively, to direct facebook.com to cs.brown.edu, for example…



https://www.thousandeyes.com/blog/internet-censorship-around-the-world



https://blog.thousandeyes.com/monitoring-dns-in-china/

Public DNS

Public DNS resolvers provided by cloud companies and ISPs

- 8.8.8.8 (Google)
- 1.1.1.1 (Cloudflare)
- … and others

Why do this?

Changing DNS servers in response to blocking of Twitter in Turkey (2014)

Writeup, with more links: https://www.thousandeyes.com/blog/internet-censorship-around-the-world

“Helpful” ISPs

- Many ISPs hijack NXDOMAIN responses to “help” by offering search and advertisement related to the domain
- E.g., www.bicycleisntadomain.com doesn’t (currently) exist
  - Could return a page with search and ads on bicycles (or domain registrations?)

What can be done?

Some defenses against DNS spoofing/hijacking

What can be done?

Some defenses against DNS spoofing/hijacking

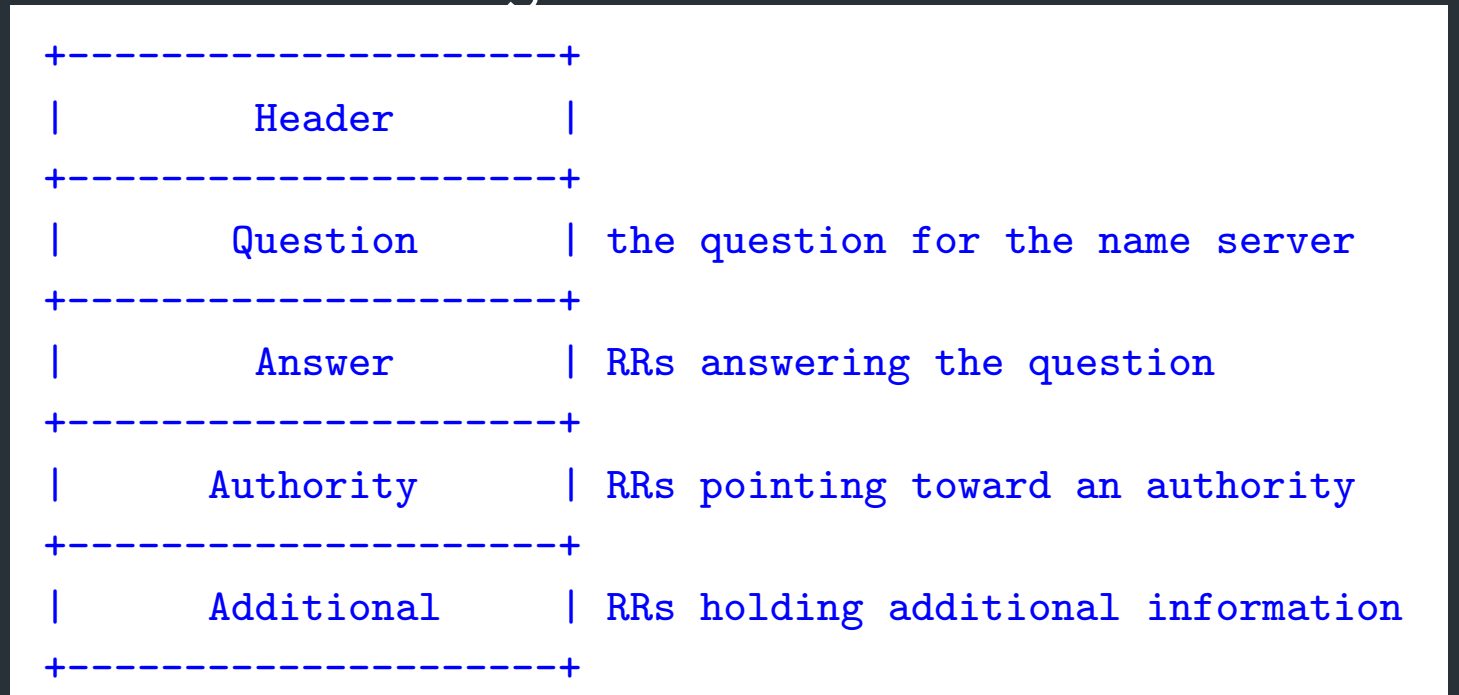
- DNSSEC: protocol to sign/verify hierarchy of DNS lookups
  - Expensive to deploy, hierarchy must support at all levels
  - APNIC DNSSEC monitor: https://stats.labs.apnic.net/dnssec
  - https://www.internetsociety.org/resources/deploy360/2012/nist-ipv6-and-dnssec-statistics-6/

- Tunneling DNS: client uses DNS via more secure protocol
  - DNS over HTTPS
  - DNS over TLS

More on DNS

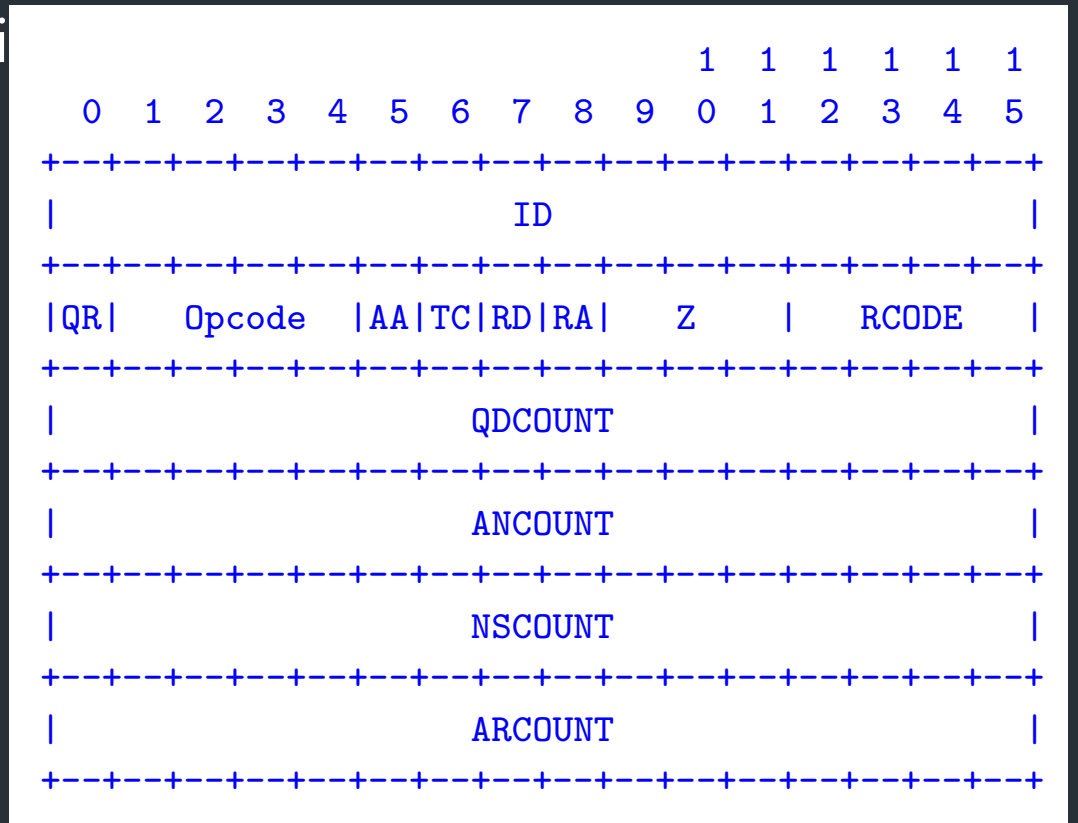
Structure of a DNS Message

- Same format for queries and replies
  - Query has 0 RRs in Answer/Authority/Additional
  - Reply includes question, plus has RRs
- Authority allows for delegation
- Additional for glue, other RRs client might need



Header format

- Id: match response to query; QR: 0 query/1 response
- RCODE: error code.
- AA: authoritative answer, TC: truncated,
- RD: recursion desired, RA: recursion avai



Other RR Types

- CNAME (canonical name): specifies an alias

```
www.google.com.              446199  IN      CNAME   www.l.google.com.
www.l.google.com.    300     IN      A       72.14.204.147
```

- MX record: specifies servers to handle mail for a domain (the part after the @ in email addr)
  - Different for historical reasons
- SOA (start of authority)
  - Information about a DNS zone and the server responsible for the zone
- PTR (reverse lookup)

```
7.34.148.128.in-addr.arpa. 86400 IN   PTR     quanto.cs.brown.edu.
```

Example

```
dig . ns

dig +norec www.cs.brown.edu @a.root-servers.net

dig +norec www.cs.brown.edu @a.edu-servers.net

dig +norec www.cs.brown.edu @bru-ns1.brown.edu

    www.cs.brown.edu.   86400 IN    A     128.148.32.110
```

Resource Records

All DNS info represented as resource records (RR)

name [ttl] [class] type rdata

- name: domain name
- TTL: time to live in seconds
- class: for extensibility, normally IN (1) “Internet”
- type: type of the record
- rdata: resource data dependent on the type

- Example RRs

```
www.cs.brown.edu.       86400  IN    A      128.148.32.110
cs.brown.edu.           86400  IN    NS     dns.cs.brown.edu.
cs.brown.edu.           86400  IN    NS     ns1.ucsb.edu.
```

DNS Example

```
$ dig cs.brown.edu @10.1.1.10
; <<>> DiG 9.10.6 <<>> cs.brown.edu @10.1.1.10
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8536
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1


;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1220
;; QUESTION SECTION:
;cs.brown.edu. IN A


;; ANSWER SECTION:
cs.brown.edu.           1800    IN      A       128.148.32.12


;; Query time: 69 msec
;; SERVER: 10.1.1.10#53(10.1.1.10)
;; WHEN: Tue Apr 19 09:03:39 EDT 2022
;; MSG SIZE  rcvd: 57
```

When server doesn't know all info…

```
% dig +norec cs.brown.edu @j.root-servers.net

; <<>> DiG 9.10.6 <<>> +norec cs.brown.edu @j.root-servers.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61618
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 27


;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;cs.brown.edu. IN A


;; AUTHORITY SECTION:
edu. 172800 IN NS a.edu-servers.net.
edu. 172800 IN NS b.edu-servers.net.
edu. 172800 IN NS l.edu-servers.net.
edu. 172800 IN NS m.edu-servers.net.


;; ADDITIONAL SECTION:
a.edu-servers.net. 172800 IN A 192.5.6.30
b.edu-servers.net. 172800 IN A 192.33.14.30
c.edu-servers.net. 172800 IN A 192.26.92.30
d.edu-servers.net. 172800 IN A 192.31.80.30
e.edu-servers.net. 172800 IN A 192.12.94.30
```

What we have

IP addresses

- Used by routers to forward packets
- Fixed length, binary numbers
- Assigned based on where host is on the network
- Usually refers to one host

Examples

- 5.6.7.8
- 212.58.224.138
- 2620:6e:6000:900:c1d:c9f7:8a1c:2f48

Efficient forwarding: ✅
Human readable: ❌
Scalable for distributed services: ❌

What we have

IP addresses

- Used by routers to forward packets
- Fixed length, binary numbers
- Assigned based on where host is on the network
- Usually refers to one host

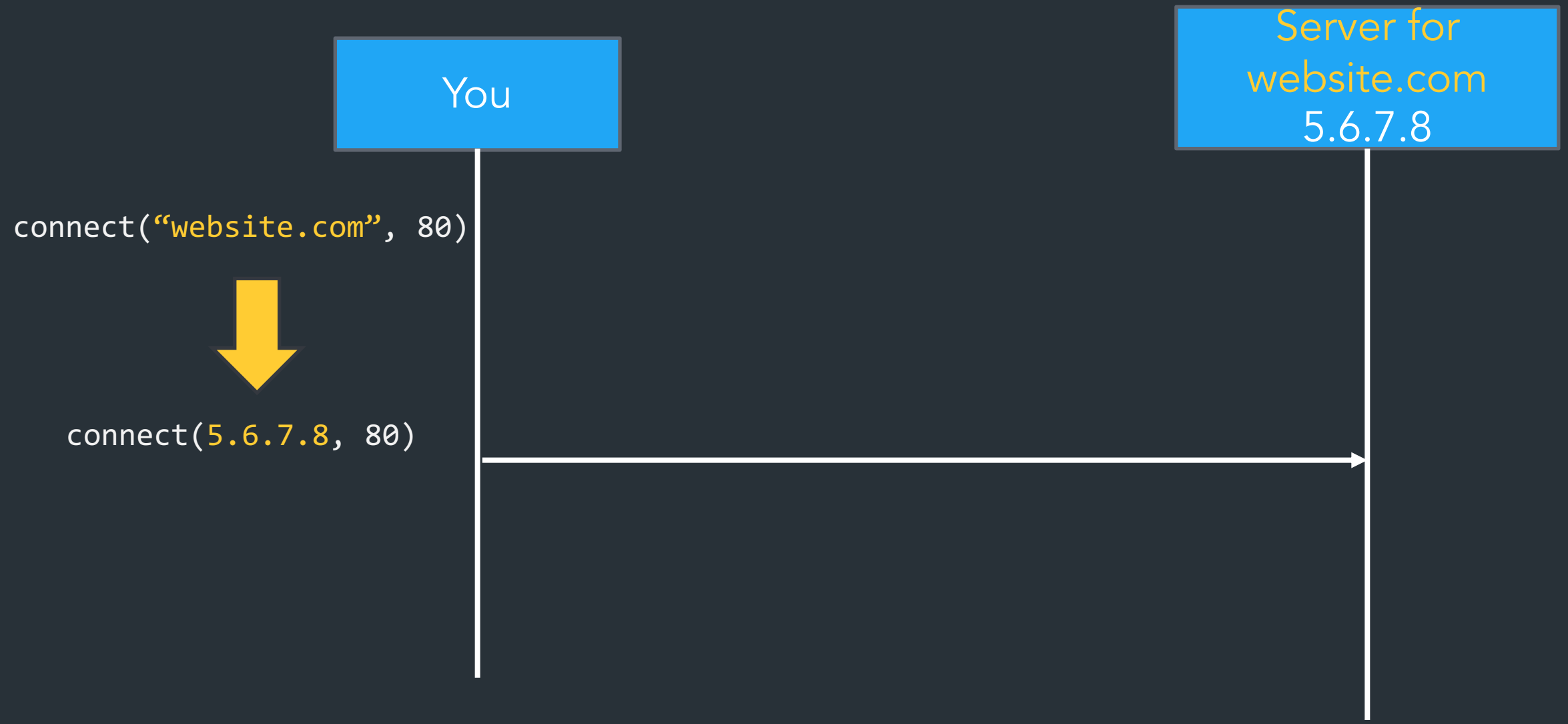
Examples

- 5.6.7.8
- 212.58.224.138
- 2620:6e:6000:900:c1d:c9f7:8a1c:2f48

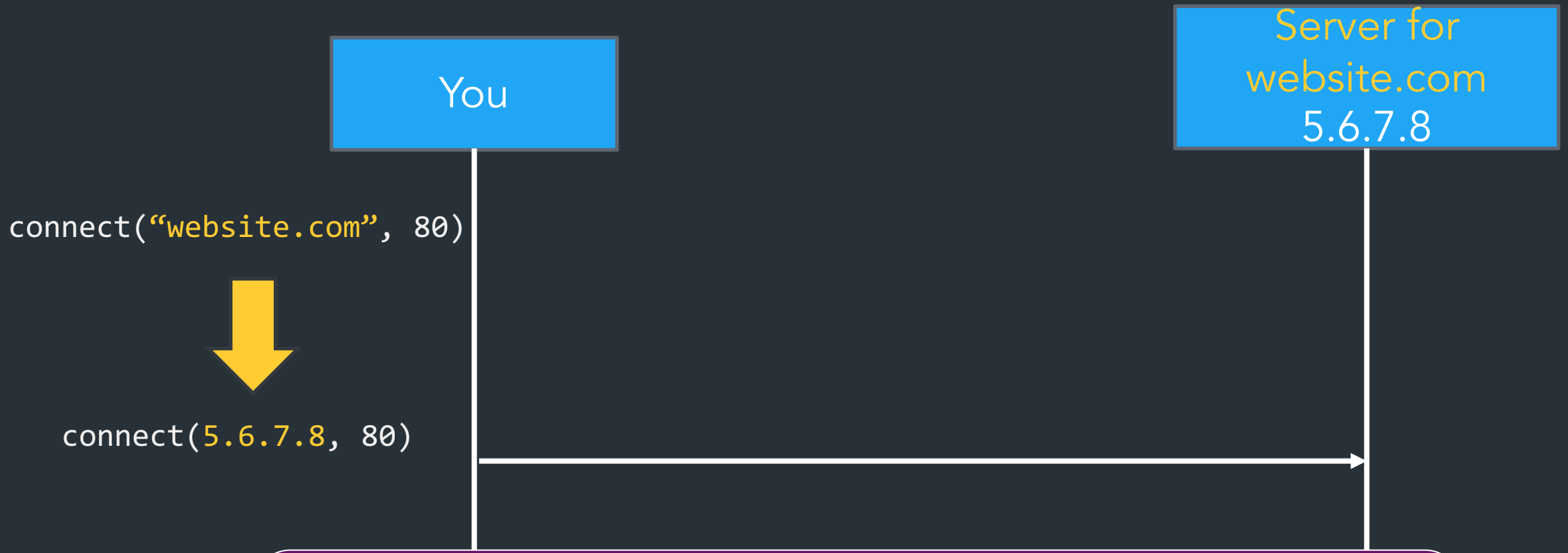
Efficient forwarding: ✅
Human readable: ❌
Scalable for distributed services: ❌

=> Need a new abstraction for “stuff” we are trying to access

What we want: a new abstraction for names



What we want: a new abstraction for names



Want: names
- Human-readable
- Variable length
- Don't need to care about where destination is/what server it is
  => Can refer to a service, not just a host

Some important details

- How do local servers find root servers?
  - DNS lookup on a.root-servers.net ?
  - Servers configured with *root cache* file
  - Contains root name servers and their addresses

```
.                       3600000  IN  NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     3600000      A     198.41.0.4
...
```

- How do you get addresses of other name servers?
  - To obtain the address of www.cs.brown.edu, ask a.edu-servers.net, says a.root-servers.net
  - How do you find a.edu-servers.net?
  - Glue records: A records in parent zone

Other uses of DNS

- Local multicast DNS
  - Used for service discovery
  - Made popular by Apple
  - This is how you learn of different Apple TVs in the building
- Load balancing
- CDNs (more on this later)

Reliability

- Answers may contain several alternate servers
- Try alternate servers on timeout
  - Exponential backoff when retrying same server
- Use same identifier for all queries
  - Don’t care which server responds, take first answer