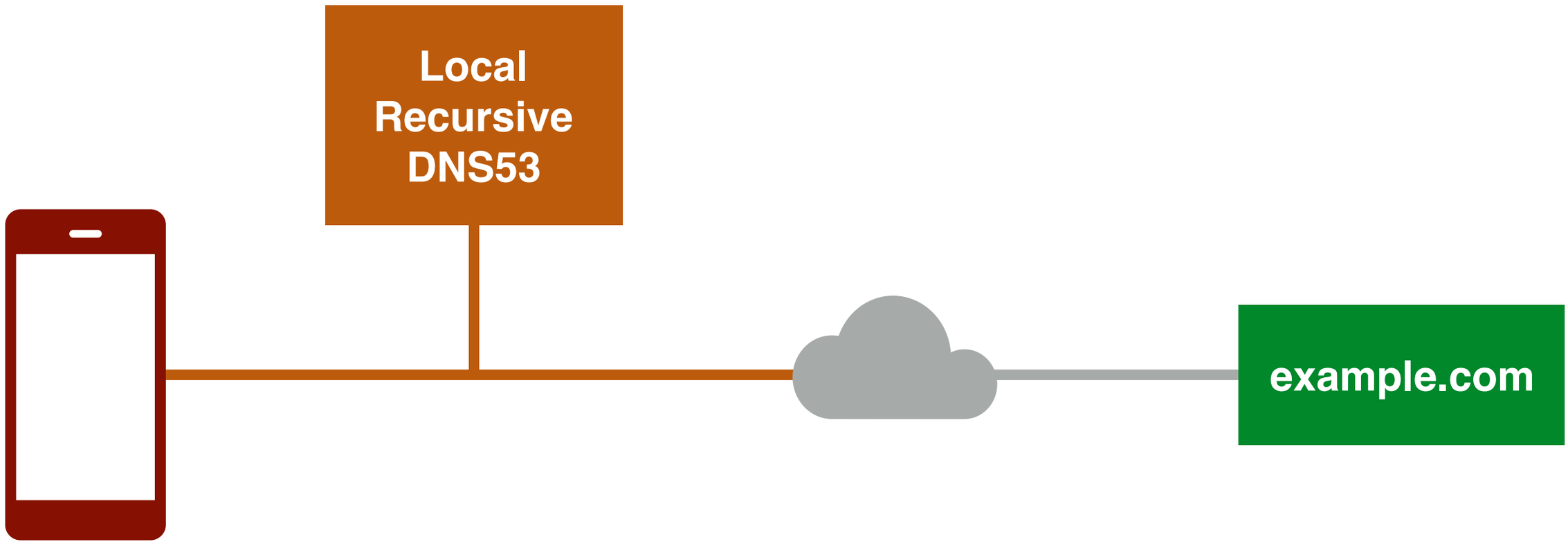
Designated Encrypted DNS Servers

draft-pauly-dprive-adaptive-dns-privacy-01

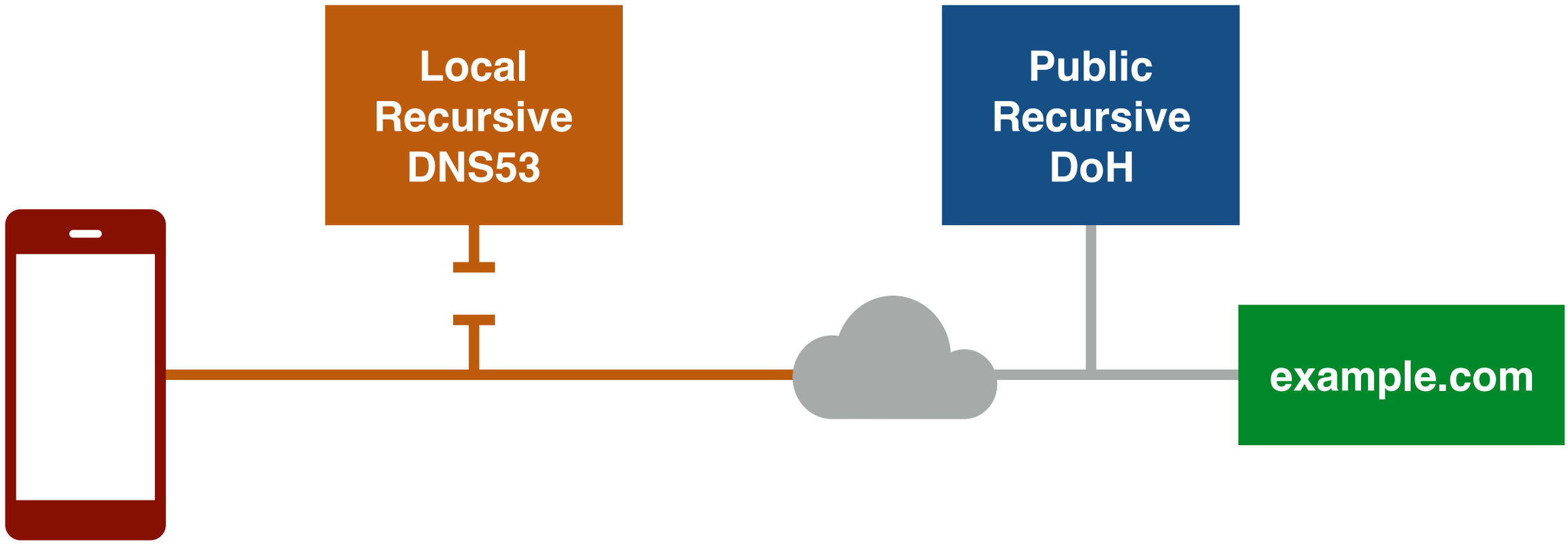
Tommy Pauly, Chris Wood,
Eric Kinnear, Patrick McManus

DPRIVE
IETF 106, November 2019, Singapore

Status Quo DNS



Public Recursive



Goals

Improve DNS privacy of client requests without requiring a fixed public resolver

Discover many different encrypted DNS servers, with clear indications of when to use them

Define how clients can correctly interact with enterprise resolvers, locally-hosted content, and local network policy

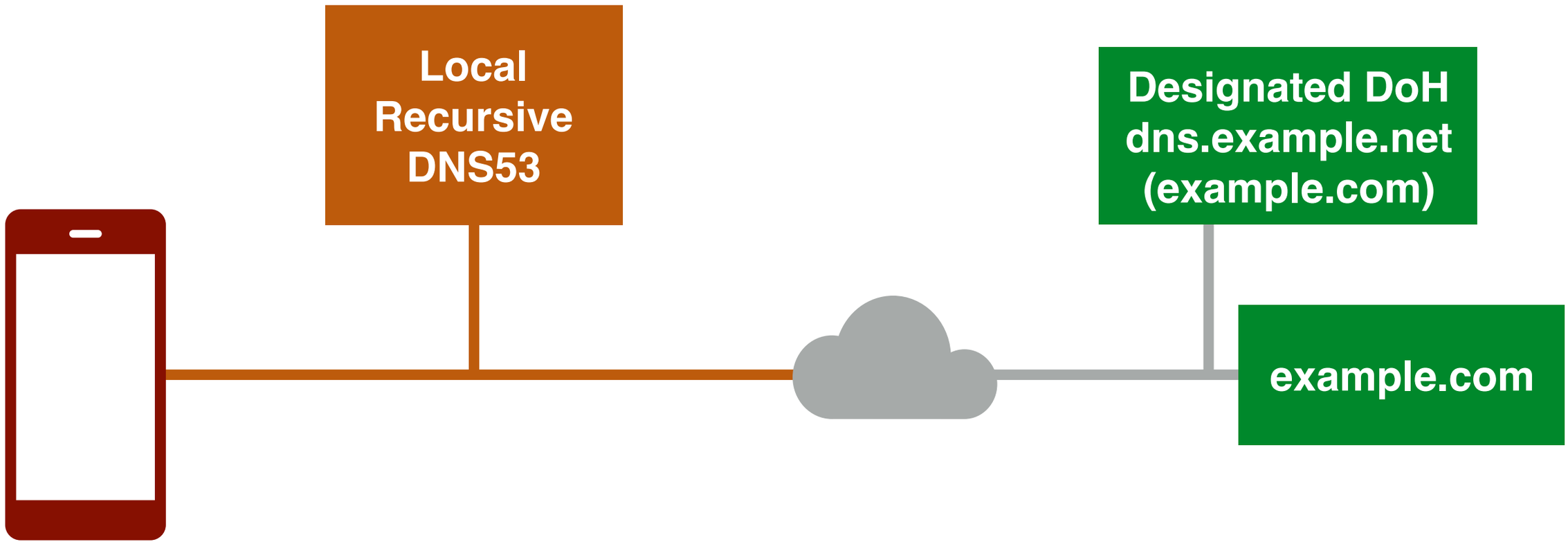
Provide a mechanism for making oblivious queries using a proxy in untrusted situations

Discovering Encrypted Resolvers

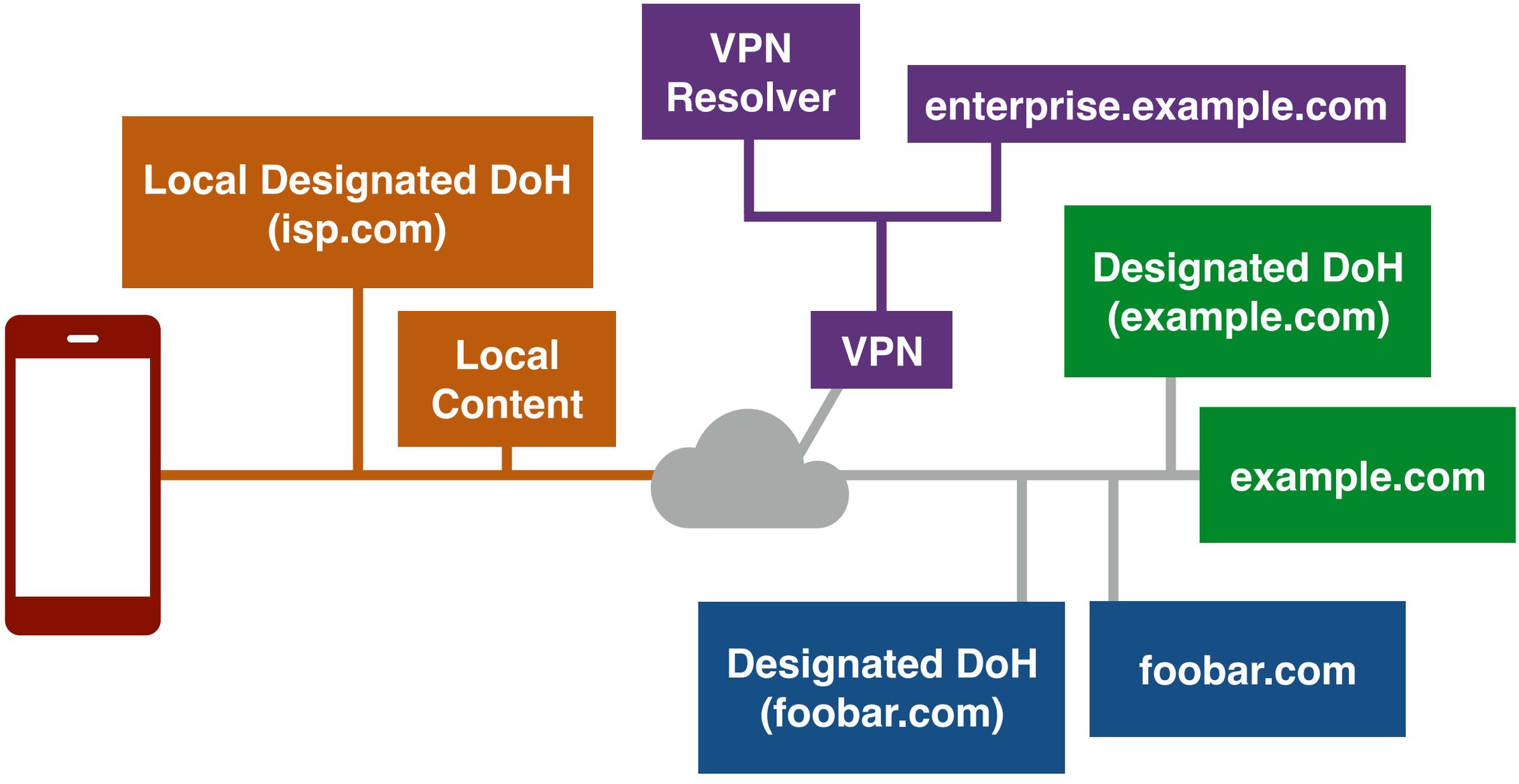
DNS records can designate a particular resolver for encrypted DNS

- Proposal uses Service Binding (SVCB/HTTPSSVC) records to indicate DoH URIs
- DNSSEC signing proves that the owner of a name designated a specific DoH service

Designated DNS Server



Designated DNS Server(s)



SVCB/HTTPSSVC Records

draft-ietf-dnsop-svcb-httpssvc-01

RRType that can be queried alongside A/AAAA

Encodes service information, such as:

- Alt-Svc (i.e., a related QUIC endpoint)
- ESNI keys

SVCB/HTTPSSVC Records

draft-ietf-dnsop-svcb-httpssvc-01

RRType that can be queried alongside A/AAAA

Encodes service information, such as:

- Alt-Svc (i.e., a related QUIC endpoint)
- ESNI keys
- **DoH URI**
- **Public Encryption Key for Oblivious DoH**

DoH URI in HTTPSSVC

Directly on queried name:

```
example.com.        7200  IN HTTPSSVC 1 . (
                                dohuri=https://doh.example.net/dns-query
                                odohkey="..." )
```

Using aliasing:

```
example.com.        7200  IN HTTPSSVC 0 svc.example.net.
svc.example.net.    7200  IN HTTPSSVC 2 svc1.example.net. (
                                dohuri=https://doh.example.net/dns-query
                                odohkey="..." )
```

Common Questions

Why use DoH for encrypting DNS?

Why use DNSSEC for validating records?

How does the system get bootstrapped?

Choice of Protocol

Focusing on DoH for now

- Allows possibility of connection reuse with HTTP
- Easy migration to QUIC via HTTP/3
- Allows for easy proxying

Can designate DoT servers as well

Signing Server Designation

If DoH server designations are not signed, an attacker can steer traffic to themselves

DNSSEC provides a mechanism to tie the designation to the zone owner

Provides a public record of designations

May be a barrier to entry for some; are there other good options that aren't just inventing something equivalent?

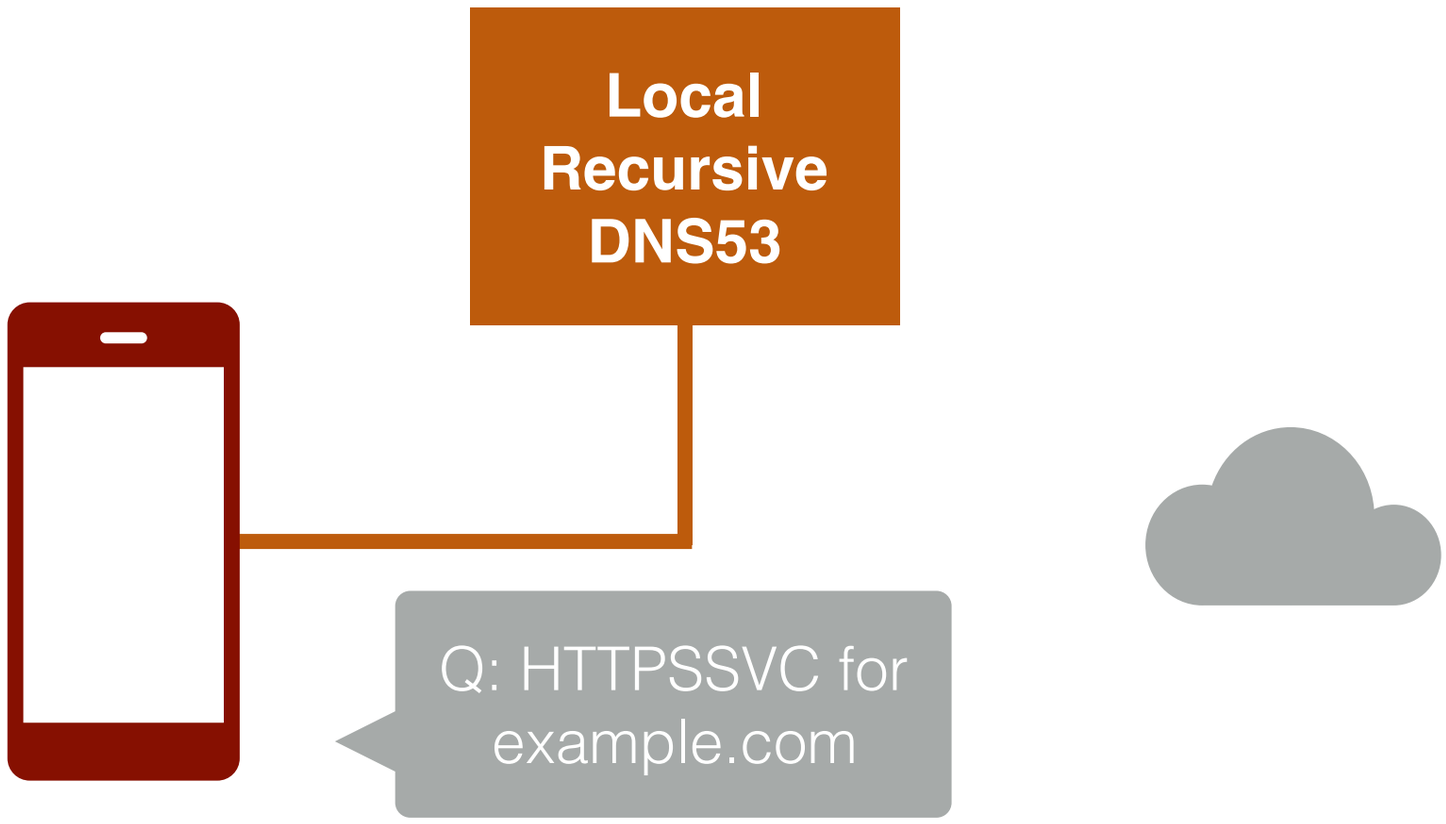
Bootstrapping

Client knows a small set of names for which it expects designated DoH servers

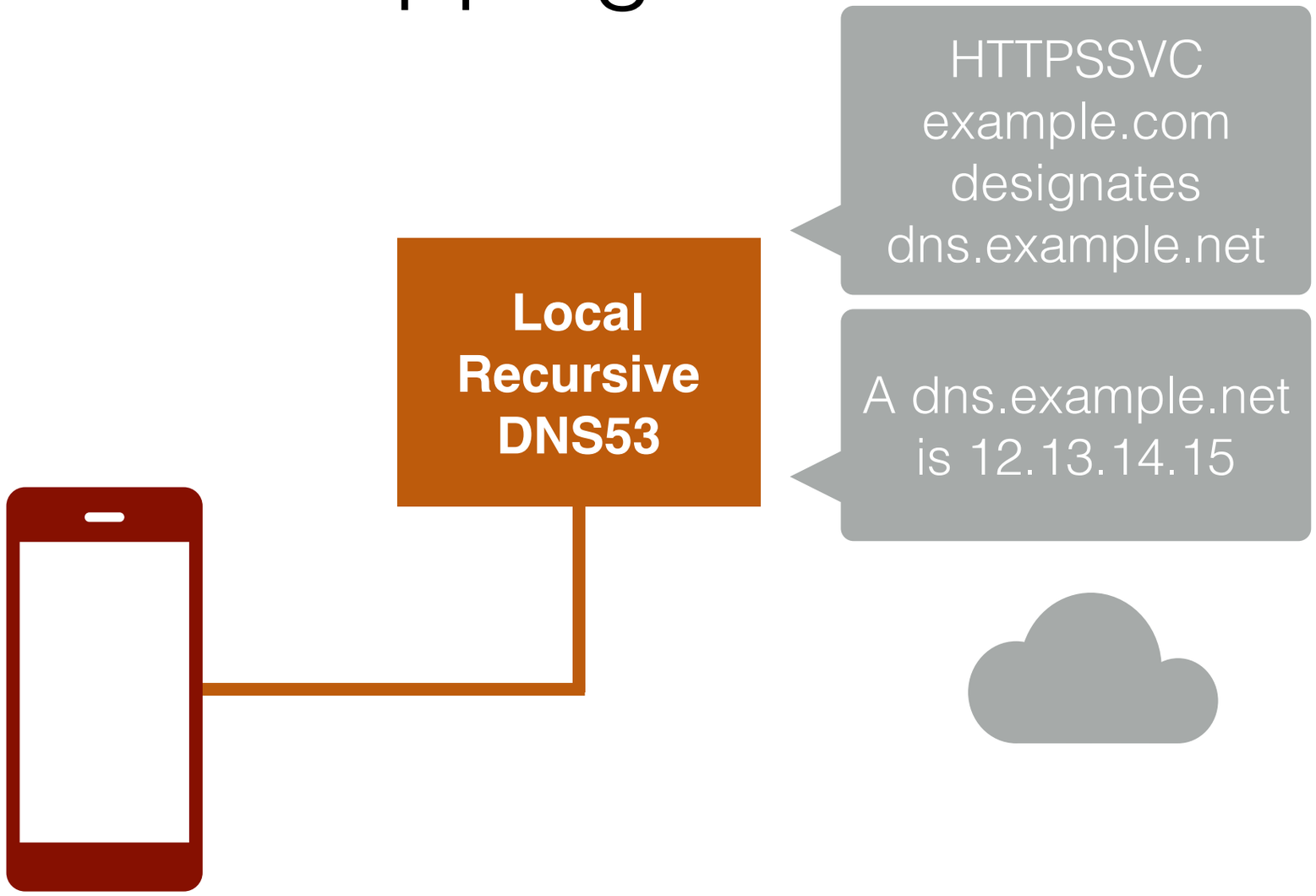
- Lookup those names over DNS53
- ... or know a few designations by default

Oblivious DoH allows doing lookups privately once the client has at >1 proxy and >1 DoH server

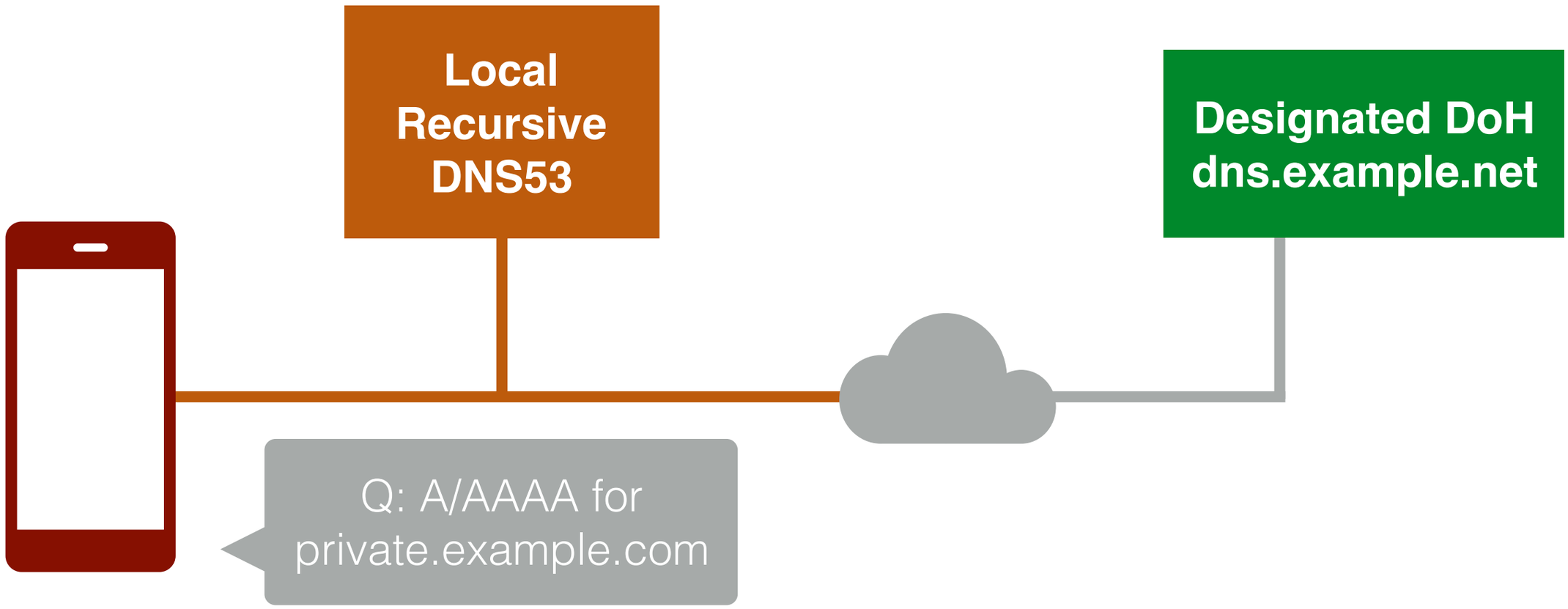
Bootstrapping



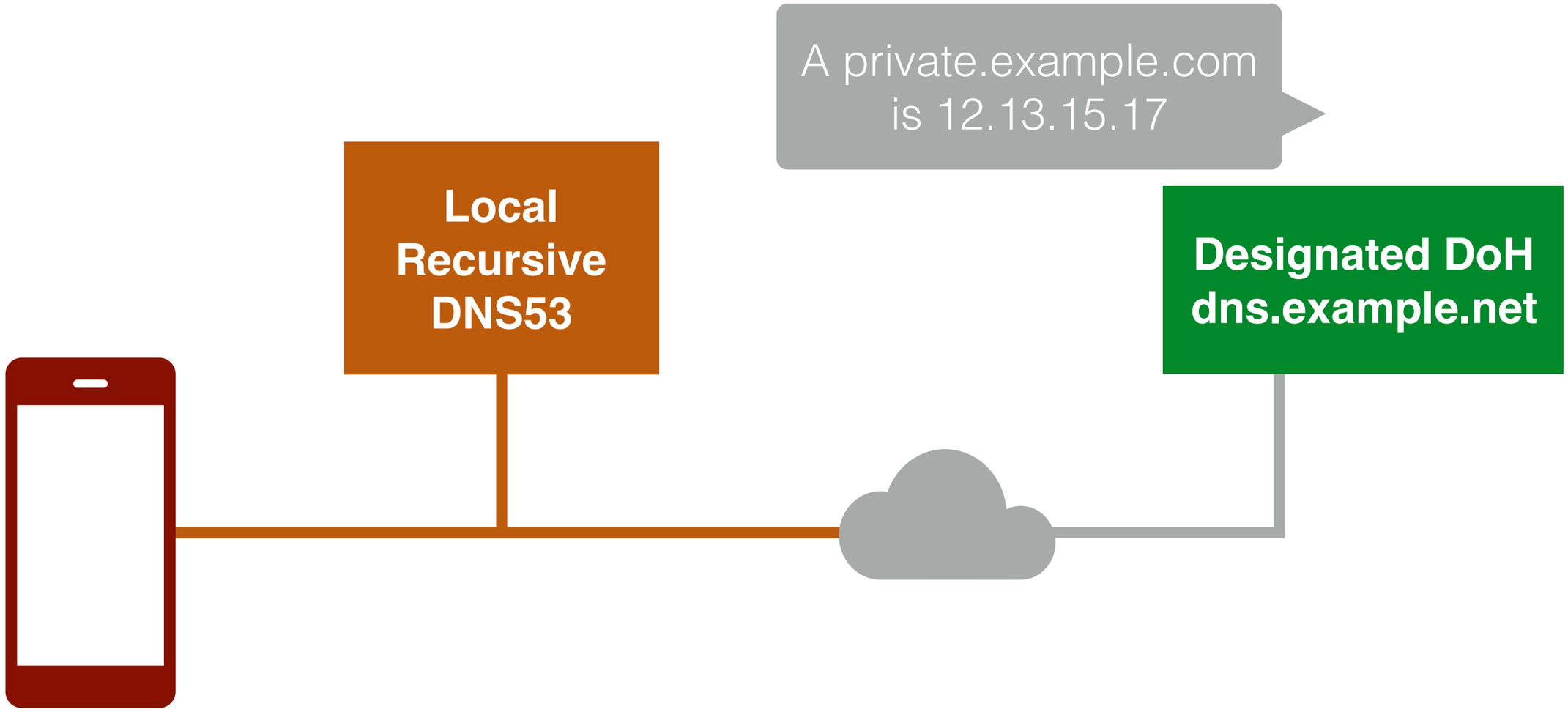
Bootstrapping



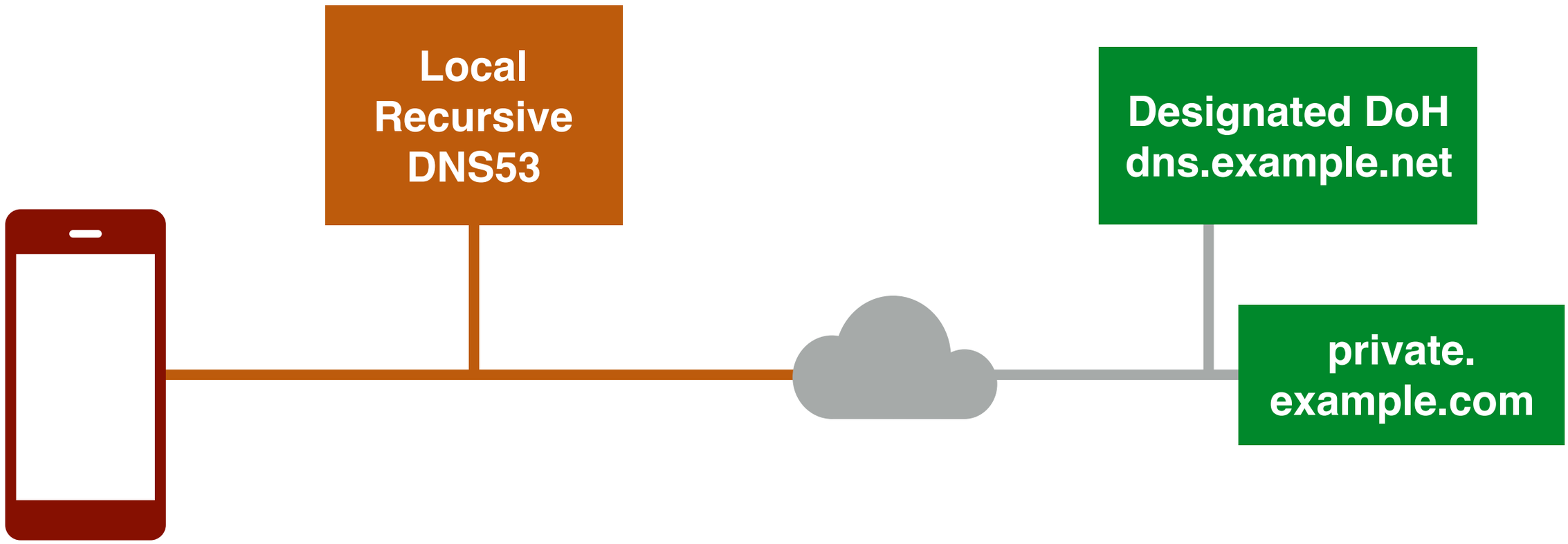
Bootstrapping



Bootstrapping



Bootstrapping



DoH Server Extended Info

"Web PvD"

Client fetches a JSON blob (provisioning domain details) from a DoH server over HTTPS as application/pvd+json

```
{
   "identifier": "dnsserver.example.net",
   "dnsZones": [ "example.com", "foobar.net" ],
   "dohTemplate": "https://dnsserver.example.net/dns-query
}
```

The list of zones are "default" domains to advertise

HTTPSSVC records can be pushed over HTTP/2 to pre-populate client cache

Open Issues

Multi-CDN deployment recommendations

Options for zones not ready to fully DNSSEC-sign

Guidance on when to re-use HTTP connections

Explain failure options and fallback considerations

https://github.com/tfpauly/draft-pauly-adaptive-dns-privacy/issues