Oblivious DoH

draft-pauly-dprive-oblivious-doh-01

Tommy Pauly, **Chris Wood**,
Eric Kinnear, Patrick McManus

DPRIVE
IETF 106, November 2019, Singapore

Oblivious DoH supports proxying *encrypted* queries between client and an untrusted resolver

Oblivious DoH

Requirements and Problem Statement

Client knowledge

- Name and public encryption key of *target* resolver
- Address of willing *proxy*

Assumptions

- ODoH targets and proxies do not collude

Goal: keep knowledge of query contents and origin IP address separate to all

HTTP Message Format

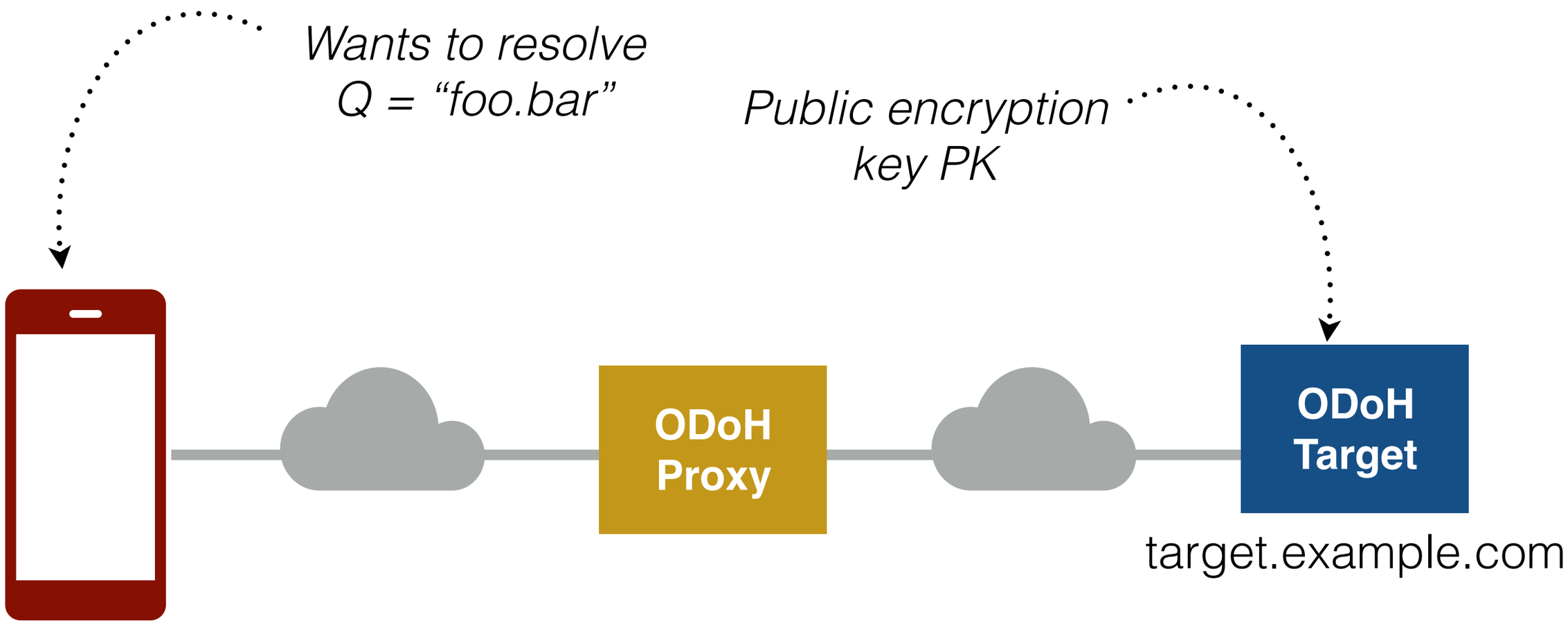
Client uses POST

Target location is specified in the path

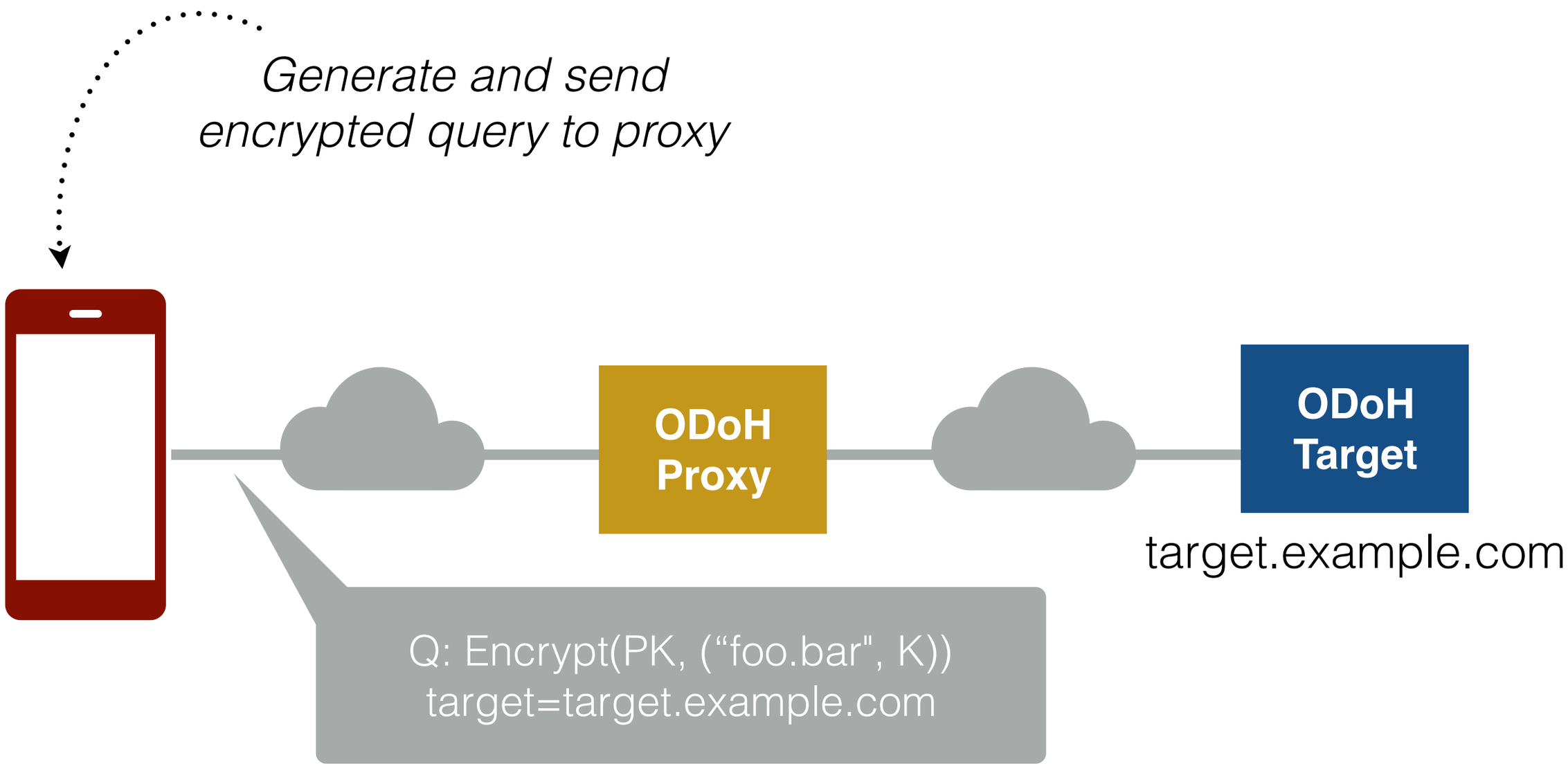
```
:method = POST
:scheme = https
:authority = dnsproxy.example.net
:path = /dns-query?targethost=target.example.net&targetpath=/dns-query
accept = application/oblivious-dns-message
cache-control = no-cache, no-store
content-type = application/oblivious-dns-message
content-length = 106

<Bytes containing the encrypted payload for an Oblivious DNS query>
```

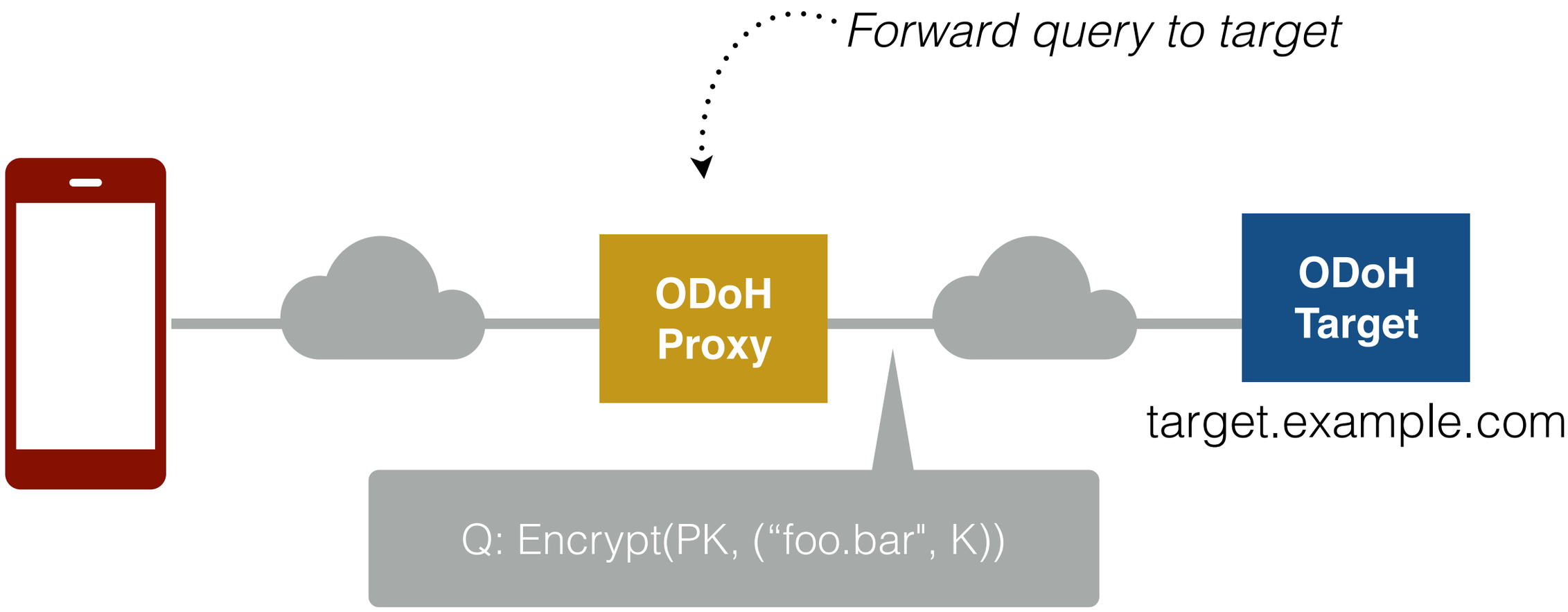
Oblivious Proxy



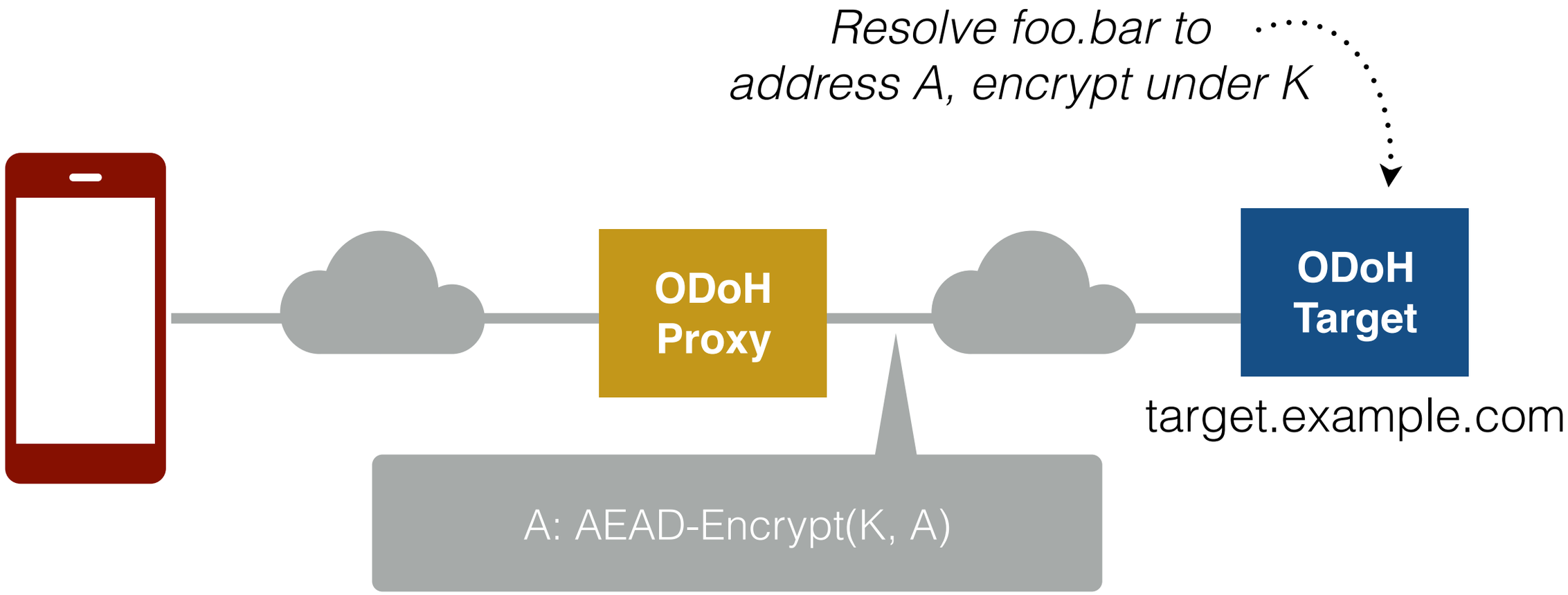
Oblivious Proxy



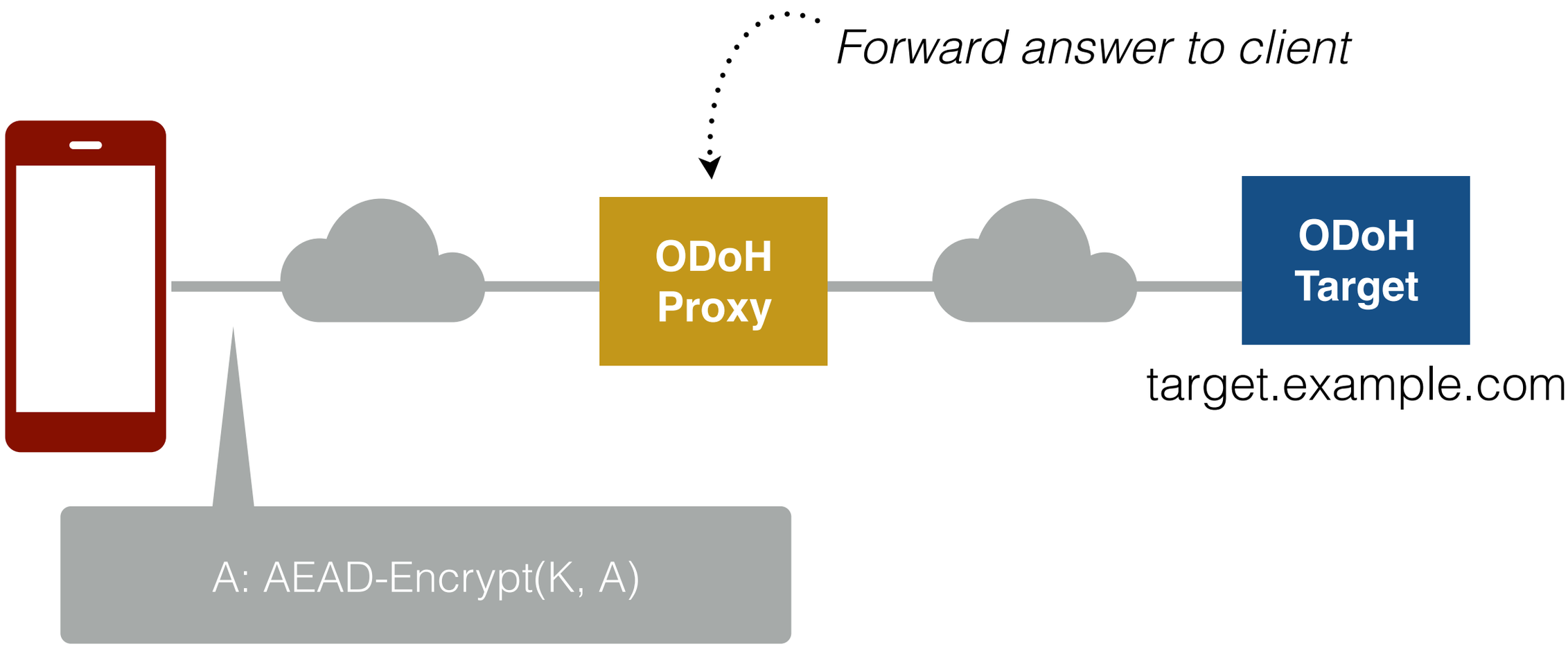
Oblivious Proxy



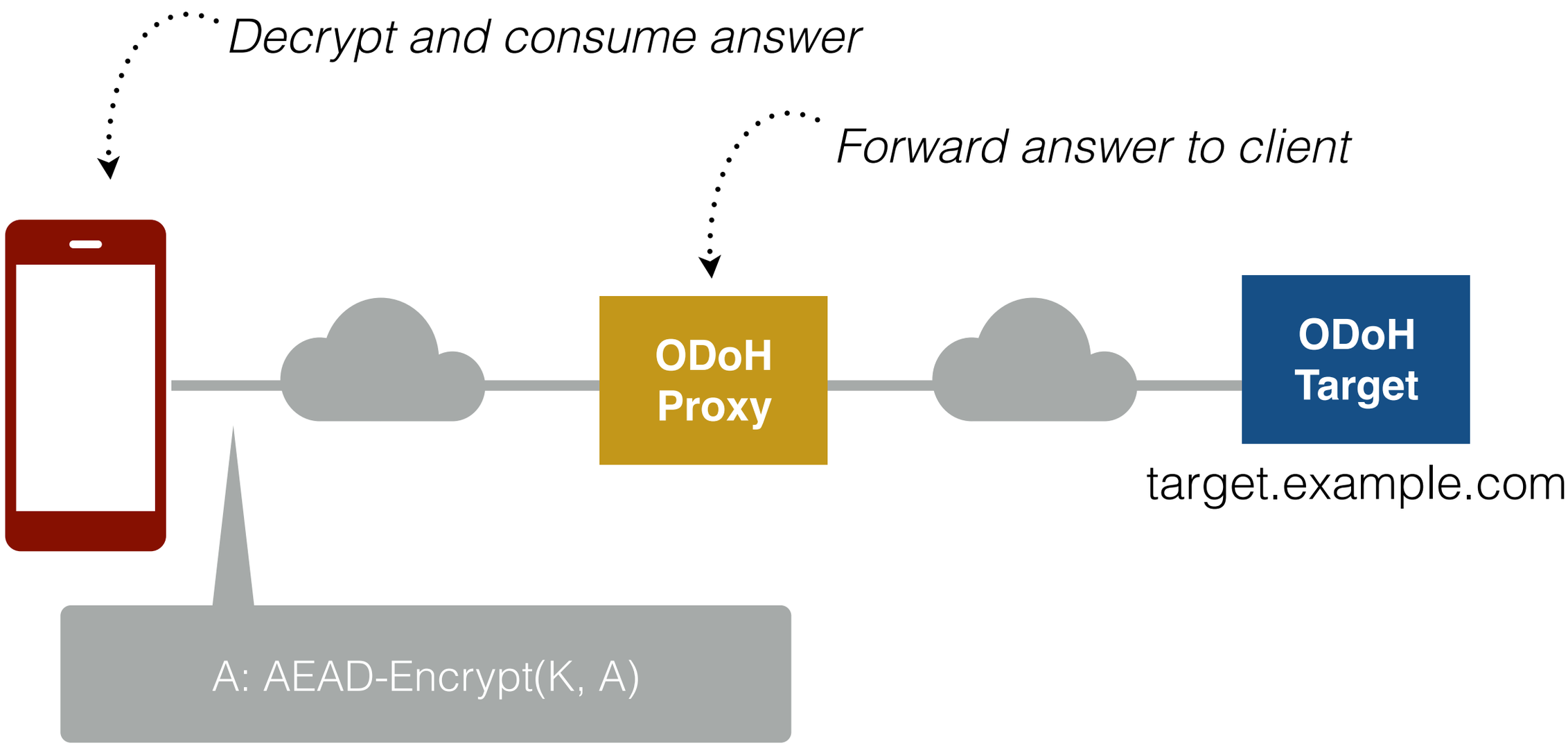
Oblivious Proxy



Oblivious Proxy

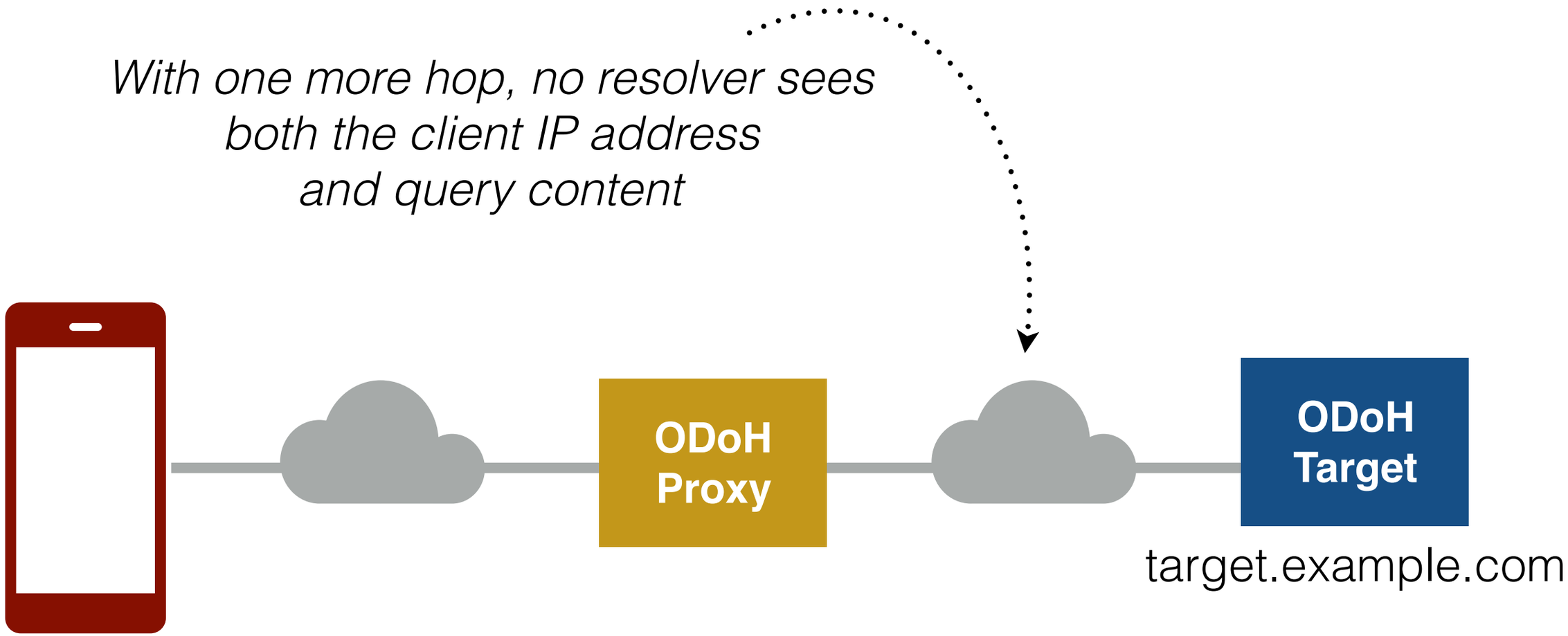


Oblivious Proxy

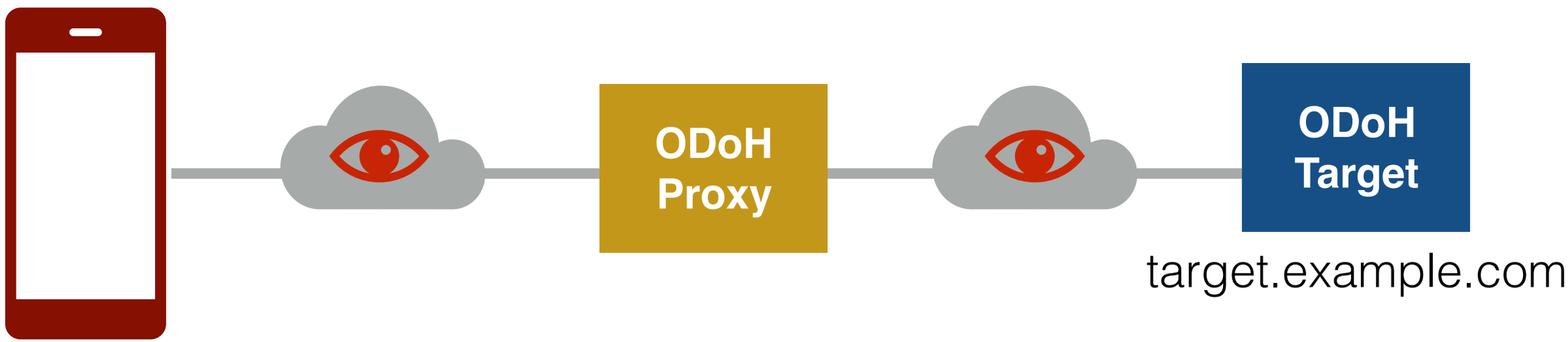


Oblivious Proxy

With one more hop, no resolver sees both the client IP address and query content



Oblivious Proxy



Oblivious DoH

Deployment Concerns

Can targets deal with per-query public key encryption overhead?

What motivates an entity to proxy traffic?

How can the client know that the proxy and target are not the same entity?

Why use DoH as a proxy?

Oblivious DoH

Variants and Next Steps

Onion encrypt queries to avoid secure connections between client<->proxy and proxy<->target?

Generalize to “oblivious HTTP”?

"Oblivious HTTP"

Rather than including the encrypted payload for DNS, include an entire encrypted HTTP request

```
:method = POST
:scheme = https
:authority = proxy.example.net
:path = /dns-query?targethost=target.example.net&targetpath=/dns-query
accept = application/oblivious-message
cache-control = no-cache, no-store
content-type = application/oblivious-message
content-length = 106

<Bytes containing the encrypted HTTP request>
:method = GET
:scheme = https
:authority = target.example.net
:path = /dns-query?dns=AAABAAABAAAAAAAAA3d3dwdleGFtcGxlA2NvbQAAAQAB
accept = application/dns-message
```

Get involved!

Draft Issues and PRs

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

Oblivious DoH Library

https://github.com/chris-wood/odoh

Sample Proxy/Target

https://github.com/chris-wood/odoh-server