Yes, This Again
Hostnames are interesting
- Alcoholicsanonymous.org
- Cia.gov
- Glaad.org
- Encrypting hostnames during connection setup has been a “holy grail” of privacy
- SNI is an obvious place where the hostname leaks
Servers need SNI data to select certificates

Hard to establish a shared encryption context before TLS

Observer can see TLS certificate anyway
  ▪ Fixed in TLS 1.3

Active attacker can get the TLS certificate anyway
  ▪ Still true in TLS 1.3
LOTS OF HOLES TO PLUG

- DNS leaks the hostname before the connection is even open
  - Doh!
- SNI leaks the hostname in the Client Hello
  - Secondary Certs
- Client doesn’t know what innocuous hostnames are available
  - Alt-Svc SNI parameter
- Alt-Svc requires having spoken to the server before
  - ALTSVC DNS records
TARGET SCENARIO

- Host has many domains, only some of which are sensitive
  - (If the fact that clients connect to the host at all is sensitive, just use TOR.)

- Want client to present an innocuous domain in SNI

- Client still needs to validate the real domain
  - Certificate might be valid for the real domain as well (*.github.io)
    - 0-RTT is possible here
  - Otherwise, use Secondary Certs to request the certificate
    - 1-RTT best case
ALT-SVC EXTENSION FOR SNI

Hypothetical Alt-Svc records for [https://sensitive.example.com](https://sensitive.example.com):

- **Colocated Domain**
  - h2="innocence.org:443";ma=2635200;persist=true;sni=innocence.org

- **Wildcard Subdomain**
  - h2="www.example.com:443";ma=2635200;persist=true;sni=www.example.com

- **Omitting SNI**
  - h2="alternative.example.com:443";ma=2635200;persist=true;sni=""
ALTSVC RECORDS IN DNS

Avoids clients making initial requests with “exposed” SNI

_https._443.www.example.com. 60S IN ALTSVC
"h2="innocence.org:443"; ma=2635200; persist=true; sni=innocence.org"

Collateral benefits

- HTTP/QUIC connections without TCP exchange first
- Opportunistic Security without cleartext exchange first
QUESTIONS?