Secondary Certificates

Solving the Easier-to-Attack problem

Secondary Certs Are Easier To Attack

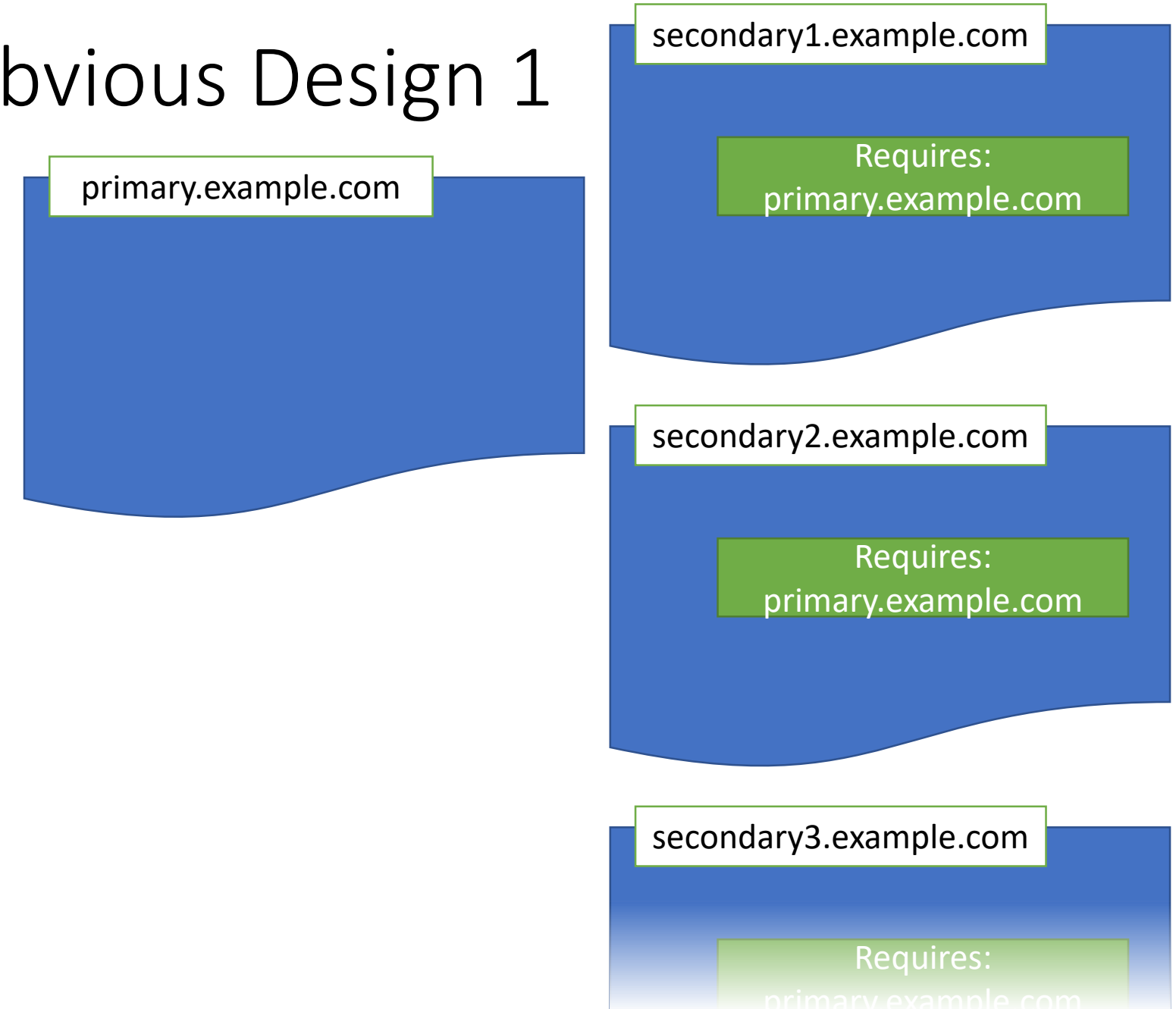
Misissued certs are less traceable

- **Without:** Attacker needs cert containing both attacker's domain and victim domain; this cert will appear in CT logs
- **With:** Attacker can use separate certs for the two domains, with no recorded link between them in CT logs

Compromised certs are easier to use

- **Without:** Attacker needs to hijack a TCP connection
  - Subvert IP routing or DNS resolution
- **With:** Attacker needs to induce navigation to an attacker-controlled origin

Obvious Design 1



- Explicit statement of primary certificate
- Have to either:
  - Successfully predict which domain will get the first connection
  - List the full set of domains in each certificate
- Doesn’t permit CDNs to coalesce across customers

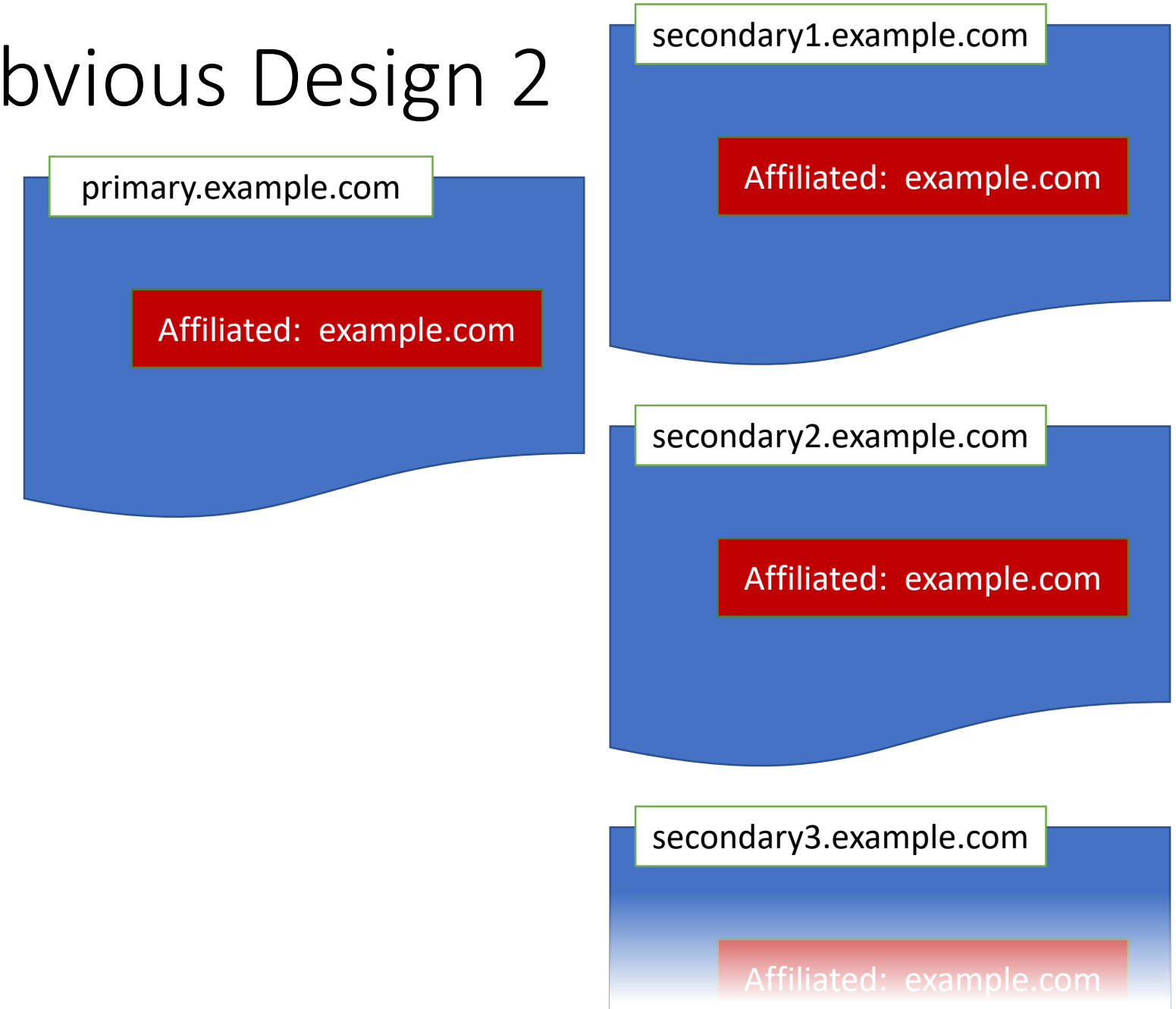
Beyond the Cert's Authority

- Want a pool of server certificates which can be used on a connection
  - Require a path from a new secondary certificate to the certificate in TLS
  - Don't require listing the full set of primary certificate domains
- Use a new extension containing some property a CA can validate
  - Probably still a domain name
- Want to support multiple use cases:
  - Origins with many domains
    - CDNs which manage certificates for customers look like this
  - CDNs where customers bring certificates

Beyond Secondary-to-Primary

- Customers won’t (and shouldn’t) add other customers to their certificates
- CDNs want to coalesce across unrelated domains they serve
- => Need to be able to satisfy Requires from another Secondary Cert

Obvious Design 2



- All secondaries share affiliation with primary
- Easy for multi-cert origins to do
- Deployment path for BYOC CDNs less clear

That Light At the End of the Tunnel…?

One option

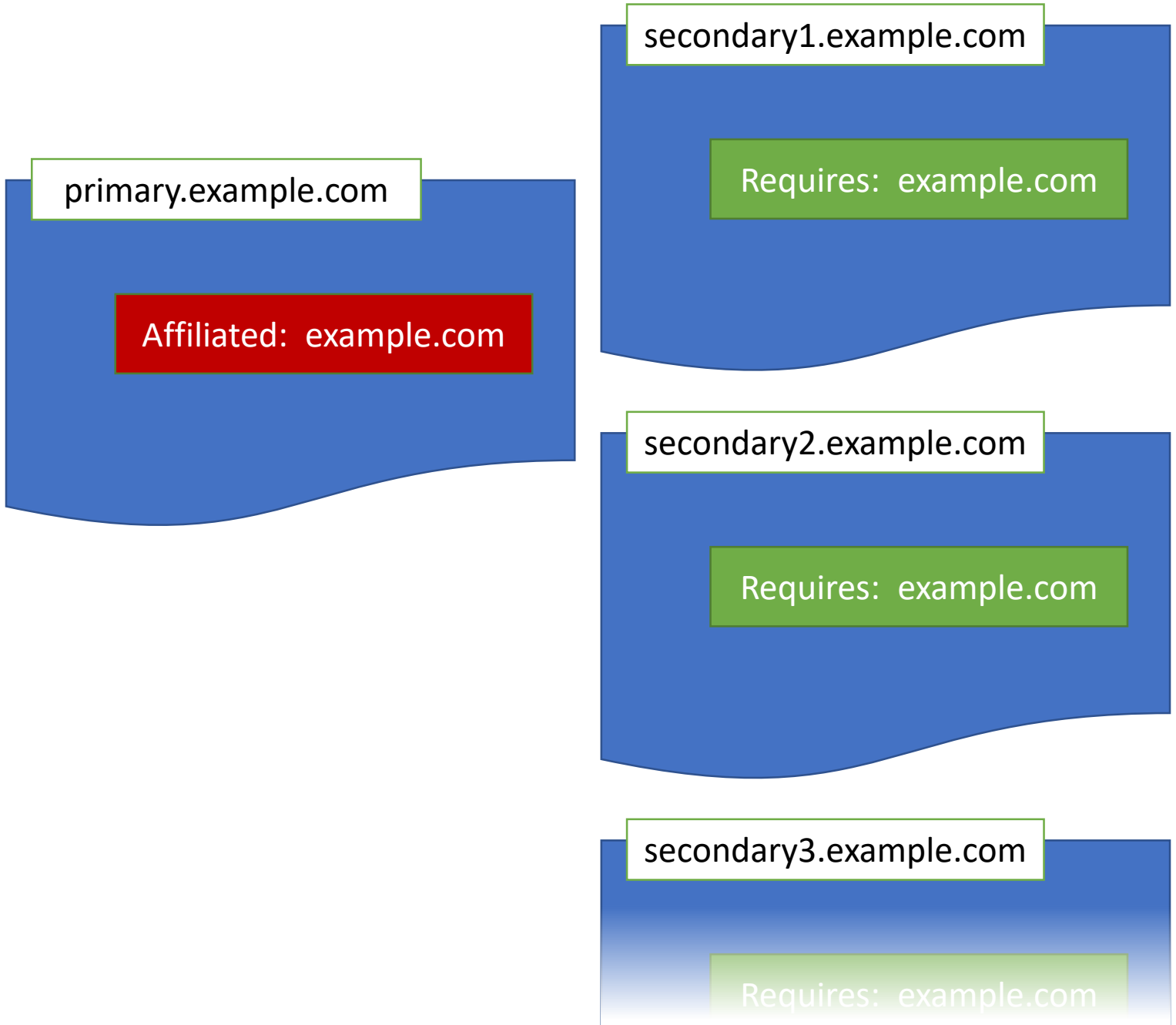
- CDN proves ownership of affiliated domain
- CDN signs token consenting to inclusion in new certificate
- Customer presents token when requesting certificate
- Customer gives certificate to CDN

Simple attack

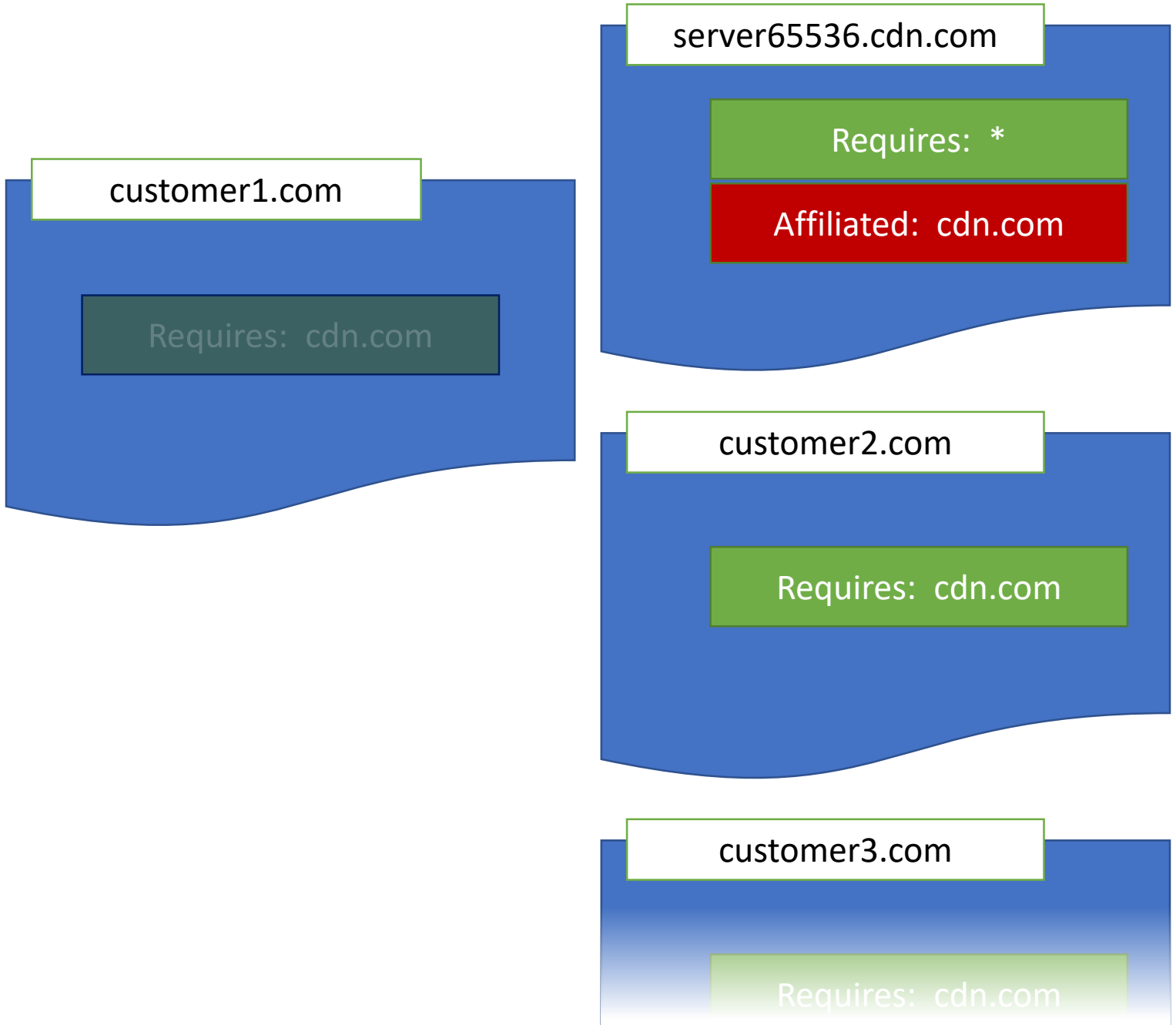
- *Premise: Customer certificate affiliated with CDN is already compromised*
- Attacker signs up as customer of CDN
- Attacker keeps the certificate affiliated with the CDN
- Induced navigation sufficient to use compromised certificate, *quod erat praeventio*

David Wheeler

All problems in computer science can be solved by another level of indirection.



- Certificates indicate an affiliation which must already be proven
- Multi-cert origins put both extensions in all certificates
- CDN customers put only one extension in their certificates



- CDN customers put only one extension in their certificates
- CDNs need to prove the CDN identity before using another customer's certs
  - One additional ExpAuth

Hard Hat Warning

- Exactly what's in these extensions
  - Single value, or list of values?
  - Domains only, or same types as supported by `subjectAltName`?
- Chaining rules
  - BYOC doesn't work if you can't meet the requirement with a previous Secondary Cert
  - Having to search all names for all previous certs for a match is a pain
    - Particularly if the list of things to look for is any of several values
- CA Amenability