

# Certificates for STIR

STIR WG

IETF 89 (London)

Sean & Jon

# draft-peterson-stir-certificates-00

- Attempt to provide a certificate-based STIR credential system
  - This is one option, not excluding others
  - Still a lot to fill in, but this is the high-level idea
- Defines a certificate extension for telephones number ranges
- Defines ways of acquiring the certs
- Sketches techniques for cert validation

# Enrollment

- Document assumes a threefold method
  - Direct assignment
    - From numbering authorities, regulators, etc.
  - Delegation from above
    - From other number holders
  - Proof of possession
    - Last time here, we had “no opposition” to going forward with that
- Agnostic on “Golden root” versus alternatives
  - Assumes at least one CA, but there can be more

# Certificates for Ranges

- Certificates have long supported telephone numbers
  - X.509 DN, various OIDs
  - Some entities will only have one number
- But some entities will have authority over multiple numbers
  - Administrative domains could control millions of numbers
    - In non-continuous ranges
  - Includes service providers, enterprises, resellers, etc.
- Ideally, a service provider should not have to have one credential per number
  - The draft contains new syntax for number ranges

# Telephone # Extension

TNAuthorizationList ::= SEQUENCE SIZE (1..MAX) OF TNAuthorization

TNAuthorization ::= SEQUENCE SIZE (1..MAX) OF TNEnter

TNEnter ::= CHOICE {

    spid ServiceProviderIdentifierList,

    range TelephoneNumberRange,

    one E164Number }

ServiceProviderIdentifierList ::= SEQUENCE SIZE (1..3) OF

    OCTET STRING

-- When all three are present: SPID, Alt SPID, and Last Alt SPID

TelephoneNumberRange ::= SEQUENCE {

    start E164Number,

    count INTEGER }

E164Number ::= IA5String (SIZE (1..15)) (FROM ("0123456789"))

# Verifier Credential Acquisition

- Different methods of acquiring certs
  - Push (e.g., cert arrives with a SIP request)
    - MIME multipart body
  - Pull (e.g., verifier acquires cert on receipt of request)
    - Either dereferencing Identity-Info URI
      - (or creating a fetch based on the originating number)
      - Current recommendation is to use EST (RFC7030)
  - Prefetch (verifier gets top 500 keys) with pull
    - SIP SUBSCRIBE/NOTIFY mentioned in the text
  - Others? Probably – no need to choose one (but MTI?)
    - DANE? If you there's a DNS tree...

# Expiry, Revocation and Rollover

- All credentials will have a lifetime
  - Ordinary rollover
    - Sometimes keys will be compromised before their expiry
  - But telephone numbers change owners, get ported, transfer normally
- Some sort of real-time checking required
  - Pull method could encompass this check
    - As could the prefetch
  - OCSP checks, but adds some overhead
    - More investigation to be done here

# Open Issue: Private Key Provisioning

- Not specific to certificates
- How do signers acquire and manage private keys?
  - Self-generated and provisioned at the authority?
  - Generated by the authority and downloaded to devices?
- Intermediaries and enterprises
  - Provision keys for number blocks, sign on behalf of calls/texts passing by
  - May possess many keys
- What's the right tool to accomplish this?



# Open Issue: Public or Confidential Credentials?

- How much information are we willing to make public?
  - Should certs contain a subject (e.g., “AT&T”)
    - Okay when a call is received to know the originating carrier?
      - Receiving user vs. receiving carrier may be different
    - More seriously, can an attacker mine a public database to reveal who owns *all* numbers?
  - Will we introduce VIPR-like privacy leaks?
- Can we restrict access to the certs?
  - Identity-Info, say, could have short lived, unguessable URLs
  - How important is endpoint verification?
    - Does trust become transitive if endpoints rely on intermediary verifiers?

# Open Issue: Partial Delegation

- Authority over numbers conflates many powers
- Should it be possible to delegate authority over services?
  - e.g., my SMS provider can sign my texts (MESSAGE), but my voice provider signs my INVITEs
    - Yes, example is kind of contrived
    - Can I give my SMS provider a text-specific cert that would not enable them to sign voice calls?
- Too complex? Do we need this?

# Open Issue: Ranges by Reference

- The certificate extension for ranges could lead to big certs
  - If a provider has a million numbers, how do you handle it?
  - Ranges can also be discontinuous, due to porting
- Rather than including ranges by-value in a cert, explore alternatives?
  - A) Synthesizing certs for each number
    - Identity-Info URL would specify the desired number
    - Has interesting private properties
  - B) Putting a URL in a cert that lets you download its current number range
  - C) Extending OCSP or a similar protocol to ask if the scope of authority contains a particular number

**BACKUP**

# Which credentials do verifiers need?

- Can we uniquely identify the needed credential based on TN alone?
  - Depends on how many authorities there are
- How many authorities and delegates per number?
  - Some kind of hint needed to disambiguate
    - Identity-Info
    - CIDER “public key index value”

# DANE (RFC6698)

- DANE requires (many MUSTs) DNSSEC validation
- The four DANE usages defined for TLS
  - 0: Specifies the root cert of the CA this site got its cert from
  - 1: Specifies which cert (possibly of many CA certs) a site uses
  - 2: Specifies a cert you should use to validate the site's cert
  - 3: **Specifies a cert** (without implying anything about CAs)
- Selector:
  - 0: Full (gives a whole X.509 object)
  - 1: **Public key only** (equal to X.509 SubjectPublicKeyInfo)
- Matching types
  - 0: **Full** (gives the object specified in the selector as a literal)
  - 1: Small hash (SHA256)
  - 2: Big has (SHA512)