RPKI Validation - Revisited

draft-huston-rpki-validation-00.txt

Geoff Huston
George Michaelson
APNIC

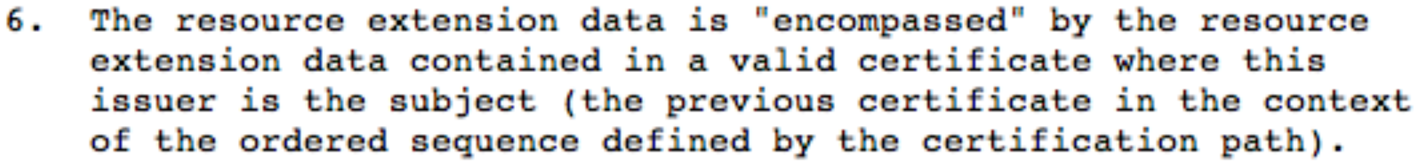
RPKI Validation

RFC6487 defined validation recursively

- For a certificate to be "valid":
  - the certificate must satisfy a number of criteria,
    - Syntax correctness, validity dates, etc
  - and there must exist an ordered sequence of certificates (1..n)
  - where:
    - Certificate 1 is issued by a trust anchor
    - Certificate x's Subject Name value matches Certificate x+1's Issuer Name value
    - Certificate 'n' is the certificate to be validated
    - Certificates 1 through n-1 are also "valid" according to this same criteria

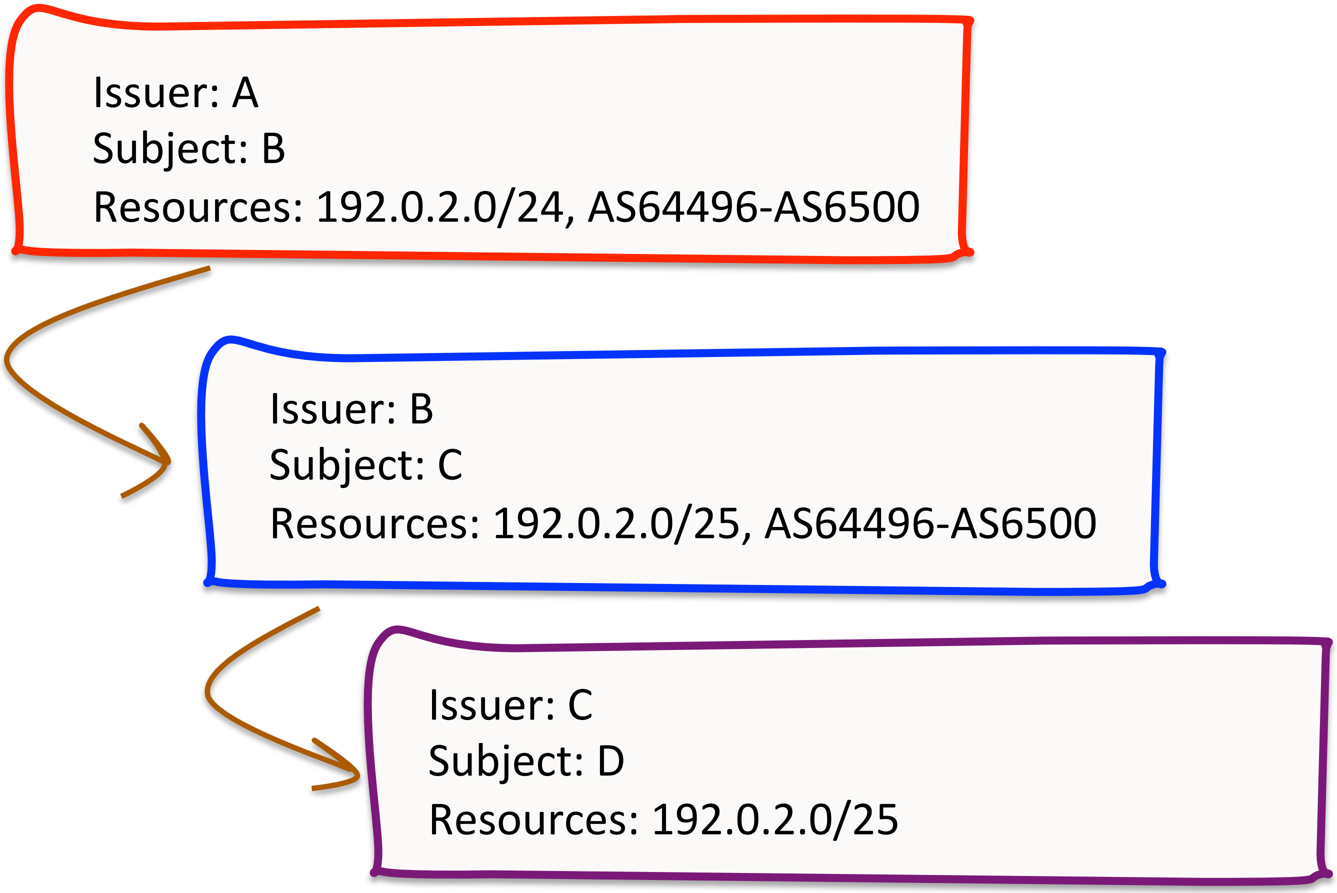
RPKI Validation

RFC6487 defined a validation criteria for the RFC3779 Number Resource extension:



```
6.  The resource extension data is "encompassed" by the resource
    extension data contained in a valid certificate where this
    issuer is the subject (the previous certificate in the context
    of the ordered sequence defined by the certification path).
```

This is Valid



Local Trust Anchor

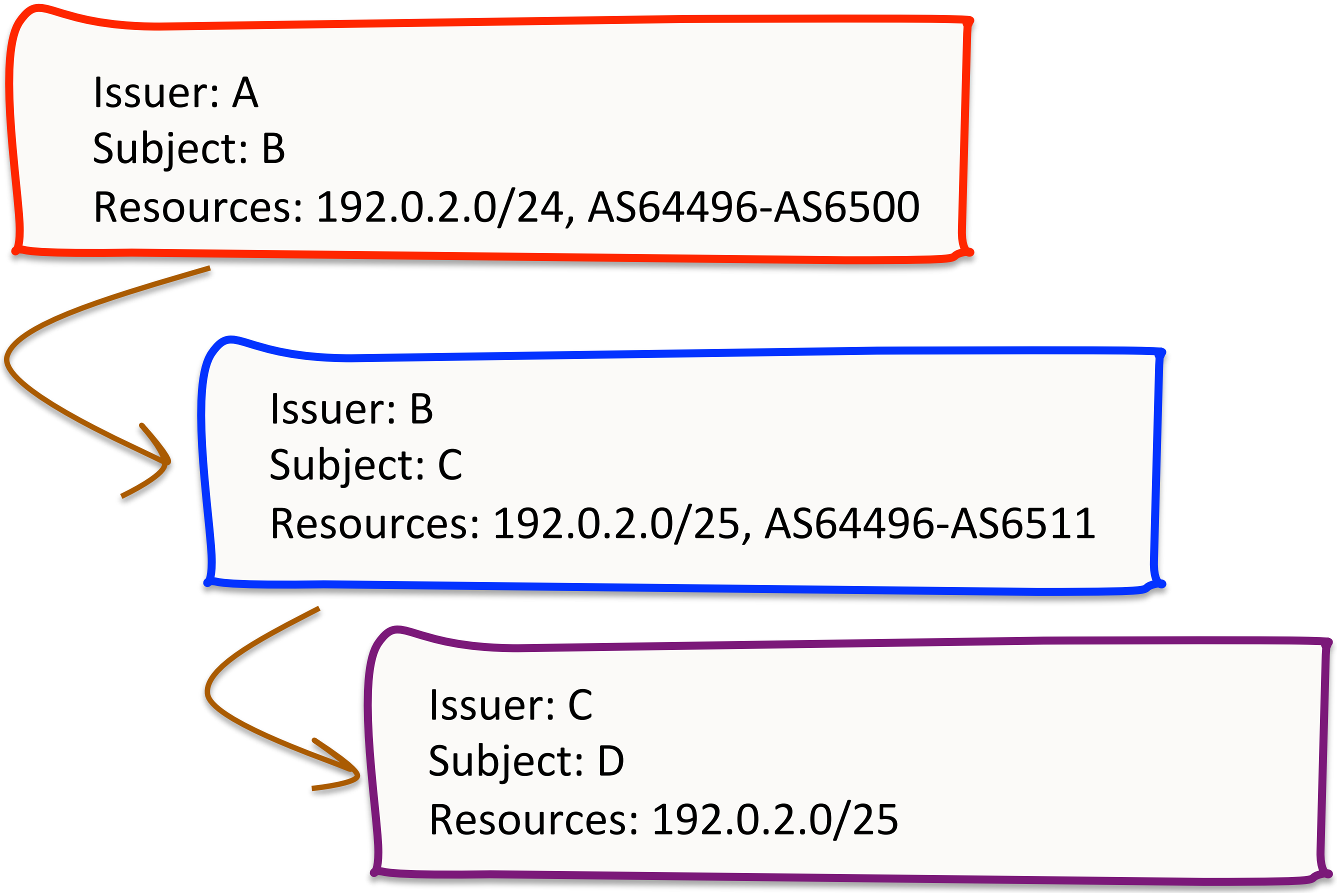
Issuer: A
Subject: B
Resources: 192.0.2.0/24, AS64496-AS6500

Issuer: B
Subject: C
Resources: 192.0.2.0/25, AS64496-AS6500

Issuer: C
Subject: D
Resources: 192.0.2.0/25

Certificate being
Tested for validity

This is not Valid



Local Trust Anchor

Issuer: A
Subject: B
Resources: 192.0.2.0/24, AS64496-AS6500

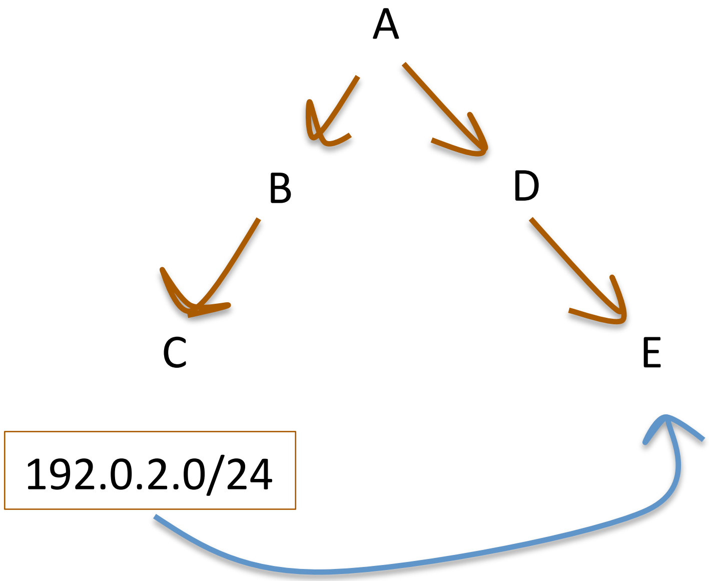
Issuer: B
Subject: C
Resources: 192.0.2.0/25, AS64496-AS6511

Issuer: C
Subject: D
Resources: 192.0.2.0/25

Certificate being
Tested for validity

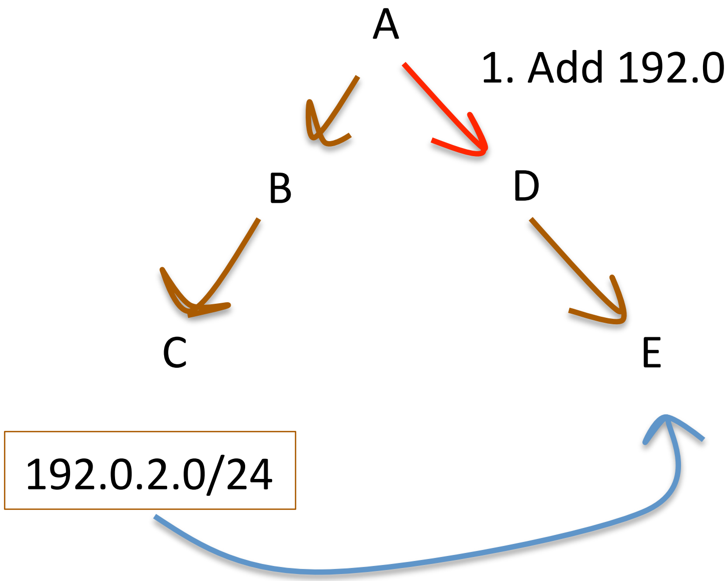
Implications

How to maintain validity of certified resources during a process of movement of resources between CA’s (registries)



Implications

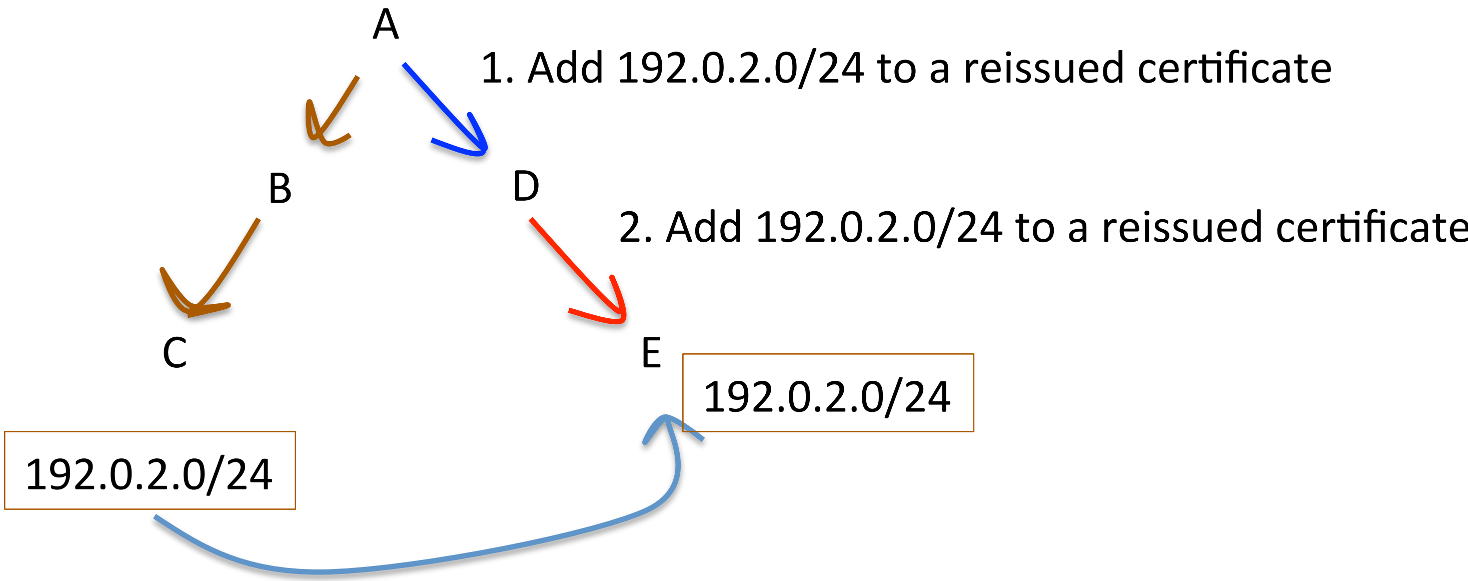
How to maintain validity of certified resources during a process of movement of resources between CA's (registries)



1. Add 192.0.2.0/24 to a reissued certificate

Implications

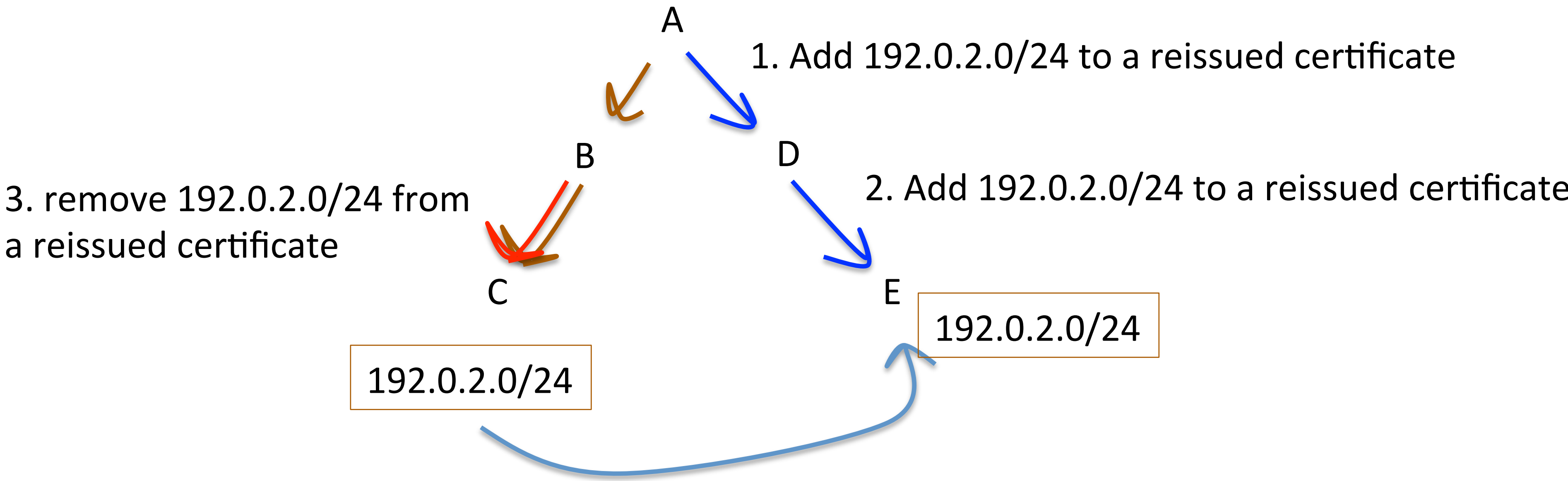
How to maintain validity of certified resources during a process of movement of resources between CA's (registries)



A
1. Add 192.0.2.0/24 to a reissued certificate
B
D
2. Add 192.0.2.0/24 to a reissued certificate
C
E
192.0.2.0/24
192.0.2.0/24

Implications

How to maintain validity of certified resources during a process of movement of resources between CA’s (registries)



A

1. Add 192.0.2.0/24 to a reissued certificate

B

D

3. remove 192.0.2.0/24 from
a reissued certificate

2. Add 192.0.2.0/24 to a reissued certificate

C

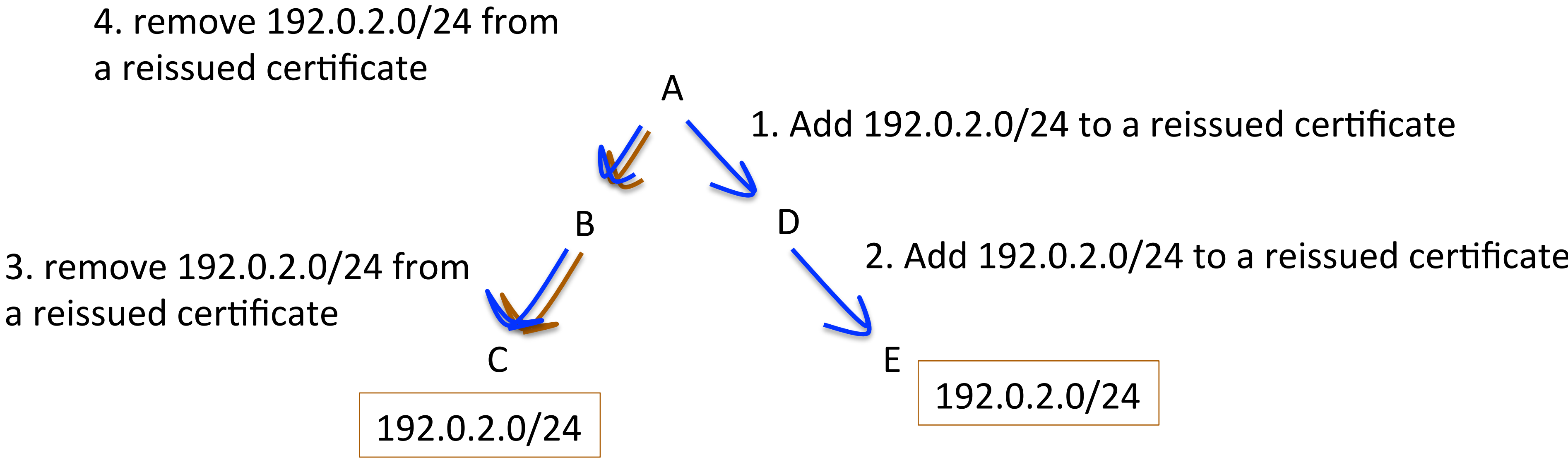
E

192.0.2.0/24

192.0.2.0/24

Implications

How to maintain validity of certified resources during a process of movement of resources between CA's (registries)



4. remove 192.0.2.0/24 from a reissued certificate

A

1. Add 192.0.2.0/24 to a reissued certificate

B

D

3. remove 192.0.2.0/24 from a reissued certificate

2. Add 192.0.2.0/24 to a reissued certificate

C

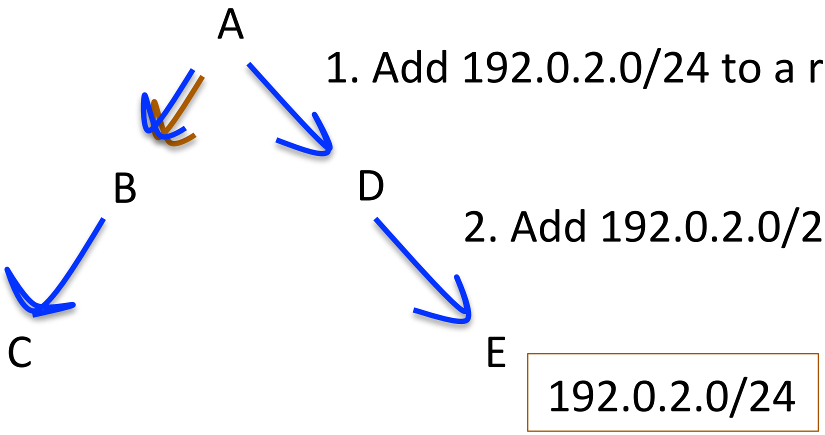
192.0.2.0/24

E

192.0.2.0/24

Implications

How to maintain validity of certified resources during a process of movement of resources between CA's (registries)



4. remove 192.0.2.0/24 from a reissued certificate

A

1. Add 192.0.2.0/24 to a reissued certificate

B

D

3. remove 192.0.2.0/24 from a reissued certificate

5. revoke the previously issued certificate

2. Add 192.0.2.0/24 to a reissued certificate

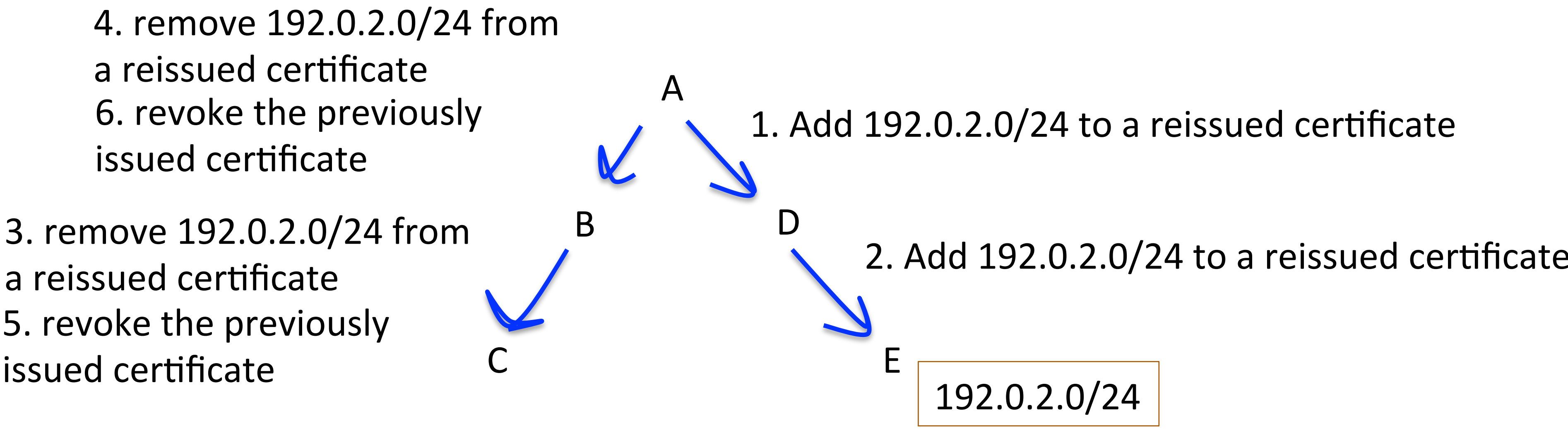
C

E

192.0.2.0/24

Implications

How to maintain validity of certified resources during a process of movement of resources between CA’s (registries)



4. remove 192.0.2.0/24 from a reissued certificate

6. revoke the previously issued certificate

1. Add 192.0.2.0/24 to a reissued certificate

3. remove 192.0.2.0/24 from a reissued certificate

5. revoke the previously issued certificate

2. Add 192.0.2.0/24 to a reissued certificate

192.0.2.0/24

Operational Synchronisation

- The "receiving side" uses a top down sequence of certificate re-issuance
- The "sending side" uses a bottom up sequence of certificate reissuance in order to avoid the side effect of unintended invalidity
  - But this requires careful synchronisation of issuance actions between CA's

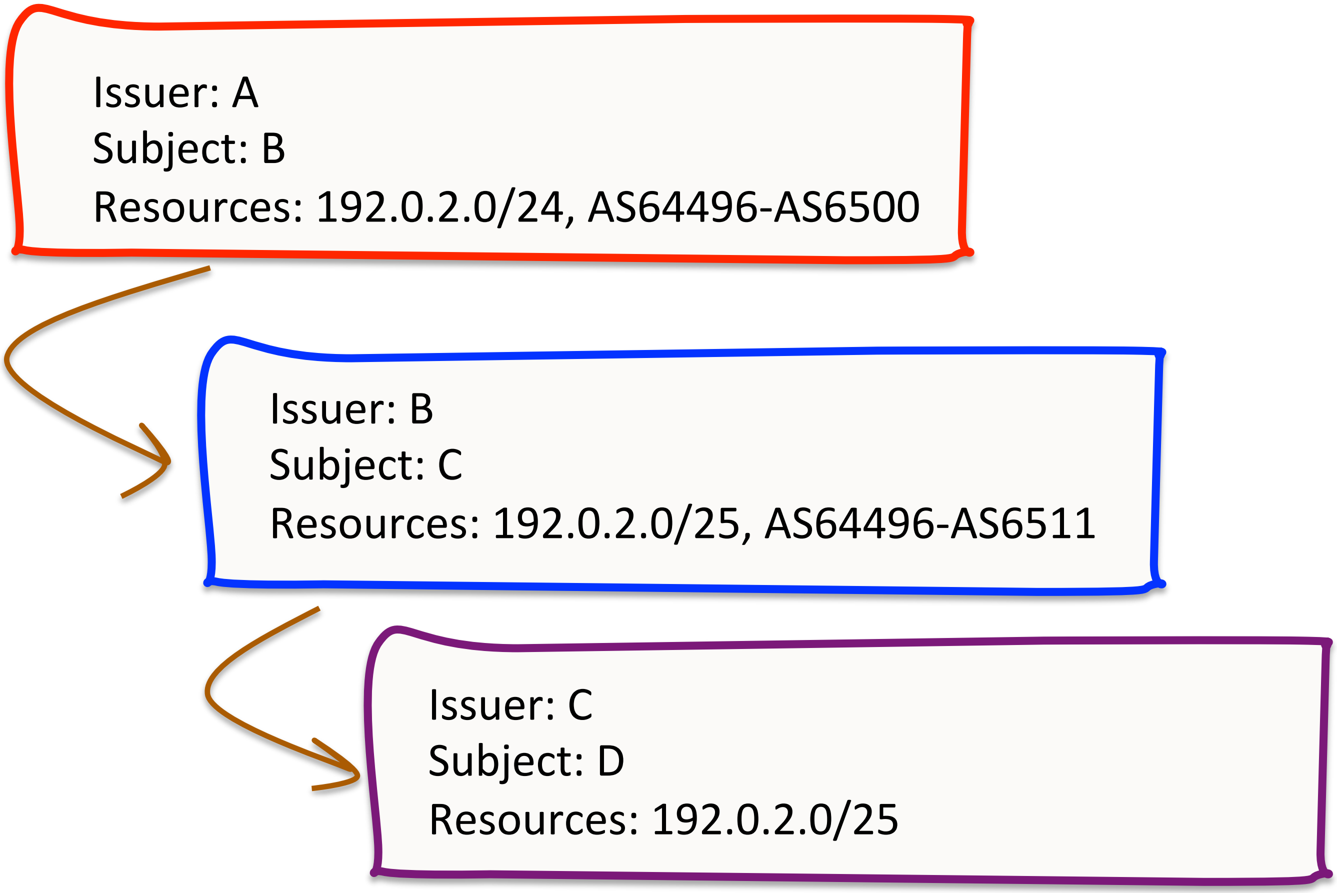
An Alternative Validity Test

Replace “*Is this certificate valid?*” with “*Is this certificate valid* ***for these resources****?*”

- i.e. add a specific set of resources to the validity question

```
6.  The resources specified in the validity test are "encompassed" by the resource
    extension data contained in all certificates that form the validation path.
```

This is Valid *for 192.0.2.0/25*



Local Trust Anchor

Issuer: A
Subject: B
Resources: 192.0.2.0/24, AS64496-AS6500

Issuer: B
Subject: C
Resources: 192.0.2.0/25, AS64496-AS6511

Issuer: C
Subject: D
Resources: 192.0.2.0/25

Certificate being
Tested for validity
with the resource
192.0.2.0/25

What about Resource Movement?

- No synchronisation of CA actions is necessary
  - Each CA can re-issue augmented or reduced subordinate certificates without needing to synchronise their actions with other Cas

What else changes?

Not much!

- A ROA is valid if the certificate used to sign the ROA is valid *against the resources listed in the ROA*
- Similar refinements can be used in other cases of RPKI certificate use

What about local cache operation?

If you need to specify a resource to undertake a validity test for a certificate then what about local cache operation? How can you pre-determine “valid” certificates in the local cache?

A revised Local Cache Management Approach

- Perform top-down local cache construction
- Add a data object to the local cache of each certificate
  - This object holds the intersection of the resources listed in the associated certificate and the resources in the data object associated with the “parent” certificate
- Use the resources in the associated data object instead of the resources listed in the certificate in all cases where “resources certified by this certificate” are used

Where is this draft going?

It may be useful
Or it may not

As of right now ,its just an idea

But we are interested to see if there are any flaws in our logic here.

Have we missed something critical in contemplating this refinement to the RPKI validity process?

This subtly different RPKI validation process could make the operational issues of CA coordination a lot easier, and thereby reduce some of the fragility in the operation of the RPKI and its potential application in secure routing.