RESOURCE TAGGED ATTESTATIONS

draft-michaelson-rpki-rta-01.txt

ggm@apnic.net

March 19 2018

OVERVIEW

- Overall Goals
- Its just CMS
- Use cases
- We have code
- Where to next?

OVERALL GOAL

- A Mechanism to permit any transaction to be associated with some Internet number resources

DESIGN GOALS

- Be Self contained
- Support multi-signing
- Do not constrain what is signed

BE SELF CONTAINED

- does not require to be published in a repository
- does not require any products to be found from a repository
- so can be **private** between two or more parties
- obviously the RP needs a trust anchor

SUPPORTS MULTI SIGNING

- so can aggregate separate resources into a super-set
- can also aggregate separate signers into a declared relationship
- but applicable to a single signer, single resource

SIGN ANYTHING

- RTA object itself has an OID assigned in the RPKI OID space
  - not intended to be seen in repository, but legal
- Detached Signature model permits anything to be signed
  - Actual object is not contained in the RTA nor in the repository

WHY?

- This is about provisioning and B2B activity outside of BGP
    - Things that happen ‘about’ resources but before or irrelevant to routing
- This is about putting **trust** into private business with resources
    - Whatever needs to be said about resources, should be verifiable
- You can model lots of things in this, because its **general**
    - Doesn’t specify what is signed or the semantic intent
    - Doesn’t say how to apply the resources to what is signed

ITS JUST CMS

ASN.1 IS WHAT IT WAS

- Initial design used home-brew ASN.1
  - we wrote zipfiles of collected bits
  - it was horrible
- Russ Housely advised use of CMS

CMS IS WHAT IT IS

- We defined a CMS conforming ASN.1 object
  - Extended to include a list of 3779 encoded resources
  - Extended to include a list of signers
- Object includes the “bag” of X.509 over the EE cert, to the TA
  - This is a stand-alone signed product: can validate as-is
- Does not **have** to appear in a repository to be validated
  - **Can** appear in a repository, EE certificates may well do

HOW IT WORKS: MAKING A SIGNED PRODUCT

- obtain an EE certificate (or set of certificates)
  - containing the resources you want to sign with
  - can be superset, can be explicitly only those relevant
- sign over a checksum of the relevant object and associated metadata
  - document, JSON, binary blob: not our business what it is
  - metadata includes lists of signers, Internet resources
    - list of signers means a group-sign can be identified as complete
    - list of resources is indicative of the applicability to the signed object

HOW IT WORKS: VALIDATION

- validation uses bag of X.509 to build path to TA
  - can then crypto validate to EE cert
    - can then prove signature over object
    - verify list of required signers all present
    - verify relevant internet number resources are covered by certificates
    - *object has to be fetched or provided OOB to validate signature*
  - cannot attest to applicability of the resources to that object
    - **out of scope**: validation limited to detached sig and crypto checks

USE CASES

‘BRING YOUR OWN IP’ PROVISIONING

- when you need to show ownership, send an RTA inline
- provisioning systems typically now send JSON over HTTPS/REST
  - specify the structural forms of ‘what is signed’
  - e.g. UUID inside customer provisioning system
- minimal friction provisioning
  - at this point in the cycle, the actual AS may not be known
  - response may include AS to be put into a ROA, but thats unconstrained

‘LICENSE TO OPERATE’

- can show anyone you have authority over a specific set of resources
- ‘signed letter of authority’
  - that LOA as PDF you pass around, but now with digital signatures
- minimally invasive on current process
  - but the process is now capable of being improved
- remote hands in DC?
  - identify the IPs you want to permit them to work on

WHAT DO PEOPLE THINK?

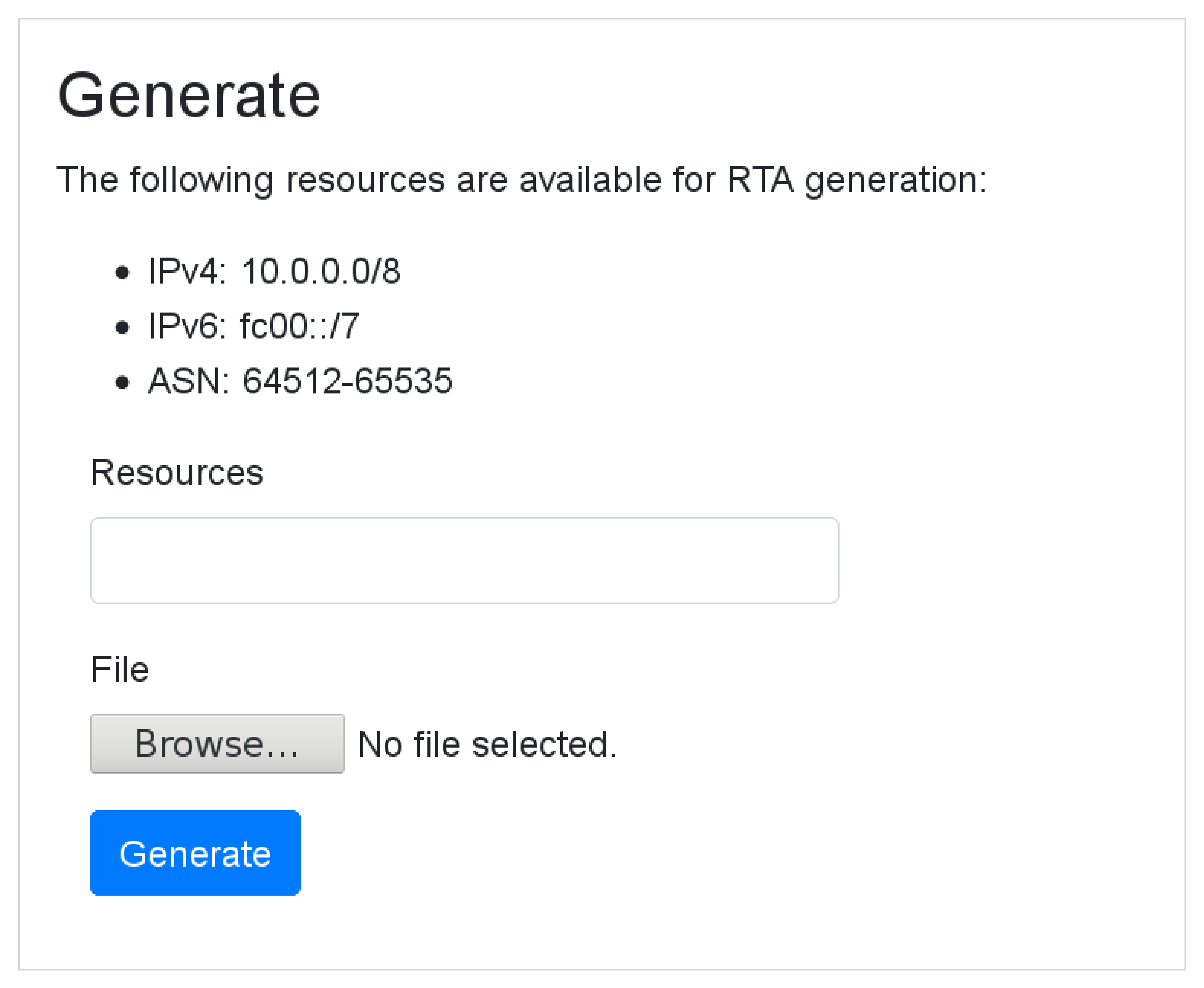
- we asked some operators of big infrastructure
  - they like it better than the alternatives right now
  - it fits into their business model
- this is workable for both their, and the resource holders interests
  - nobody else has to be involved in conducting business
  - it is better than asynchronous ‘proof by use’ publication methods
    - require use of a DNS registry, DNS server, Whois service, web update
- gets the registry out of the way: we just do the trust part.

WE HAVE CODE

WE HAVE CODE

- Simple proof of concept
  - `https://github.com/APNIC-net/rpki-rta-demo`
- Uses simple mods to dragon system
  - to extract the EE certificate/key pair for signing
- working testbed live in APNIC
  - `http://rpki-testbed.apnic.net/rta`

GENERATE



Generate

The following resources are available for RTA generation:

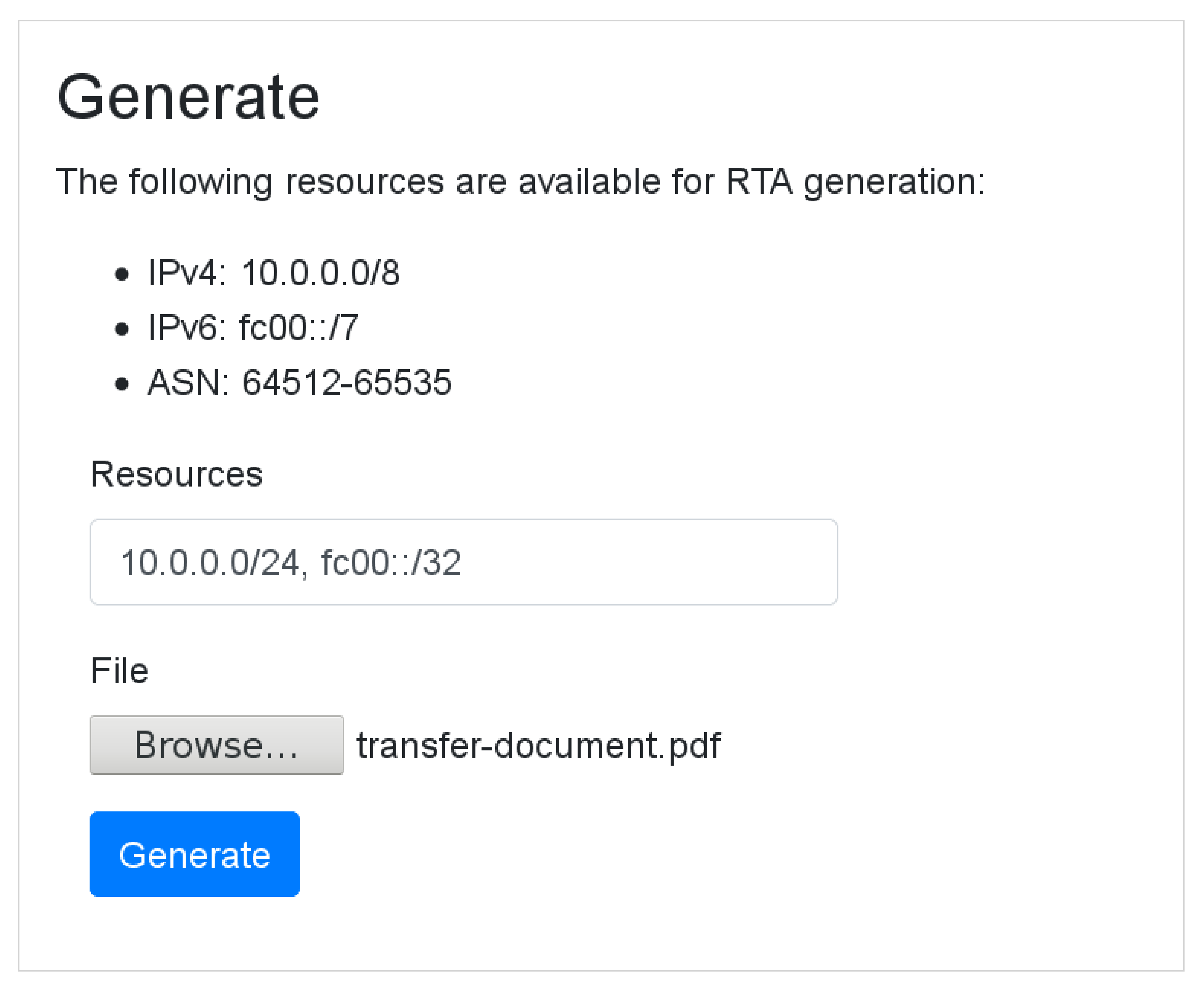
- IPv4: 10.0.0.0/8
- IPv6: fc00::/7
- ASN: 64512-65535

Resources

File

Generate

GENERATE



Generate

The following resources are available for RTA generation:

- IPv4: 10.0.0.0/8
- IPv6: fc00::/7
- ASN: 64512-65535

Resources

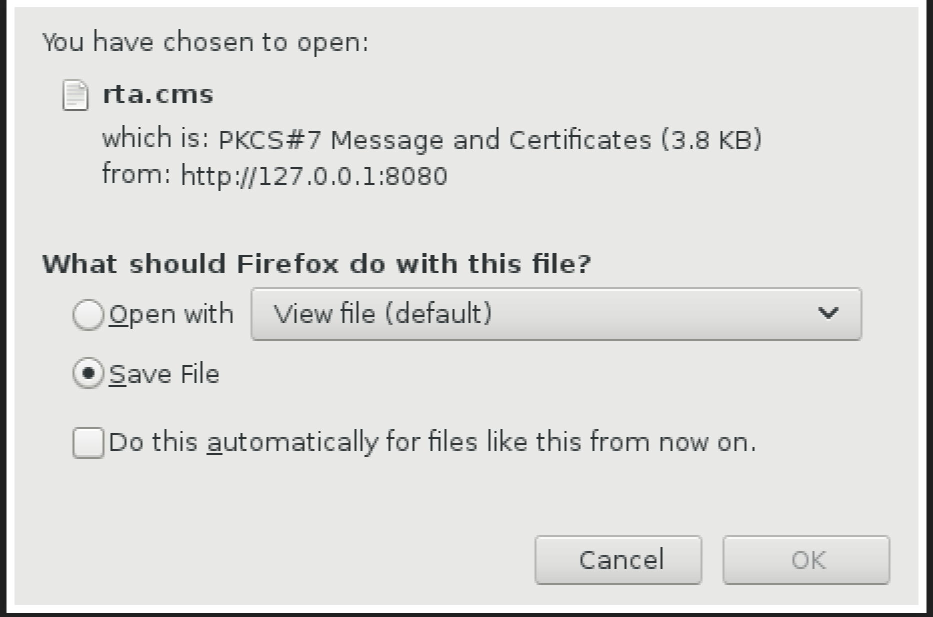
10.0.0.0/24, fc00::/32

File

transfer-document.pdf

Generate

DOWNLOAD



You have chosen to open:

rta.cms

which is: PKCS#7 Message and Certificates (3.8 KB)
from: http://127.0.0.1:8080

What should Firefox do with this file?

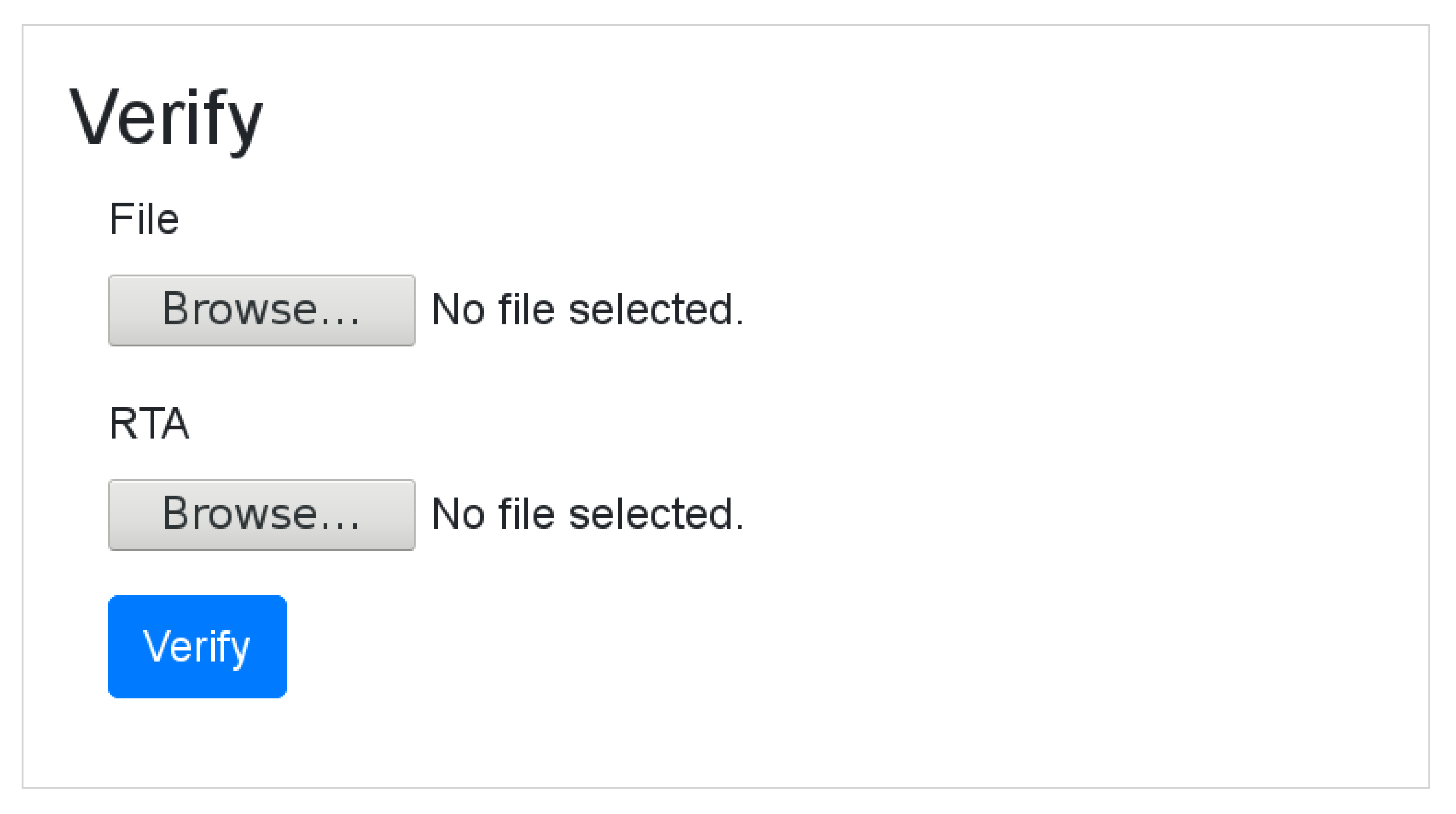
○ Open with View file (default)

● Save File

□ Do this automatically for files like this from now on.

Download

VERIFY



Verify

File

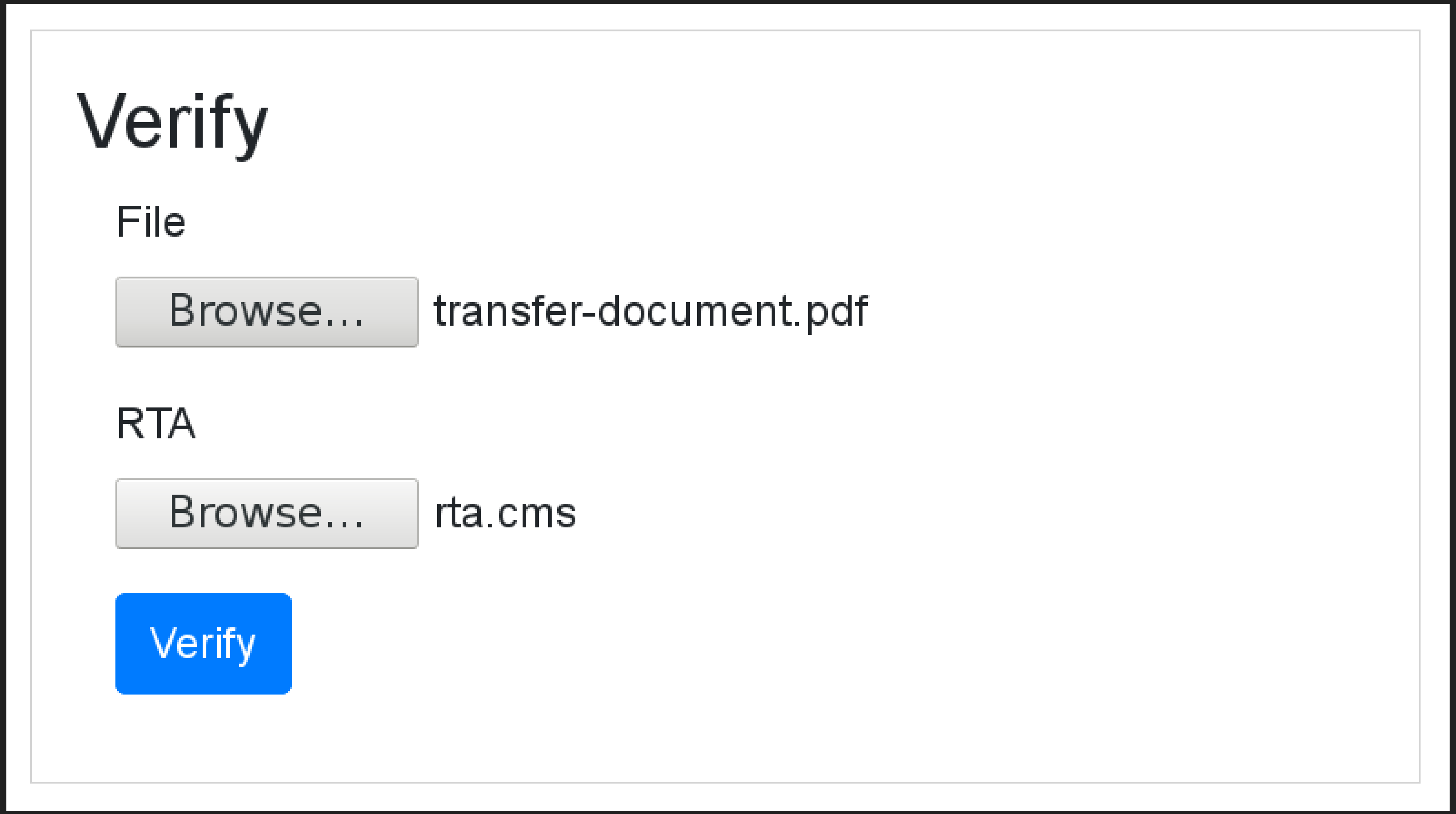
No file selected.

RTA

No file selected.

Verify

VERIFY



Verify

File

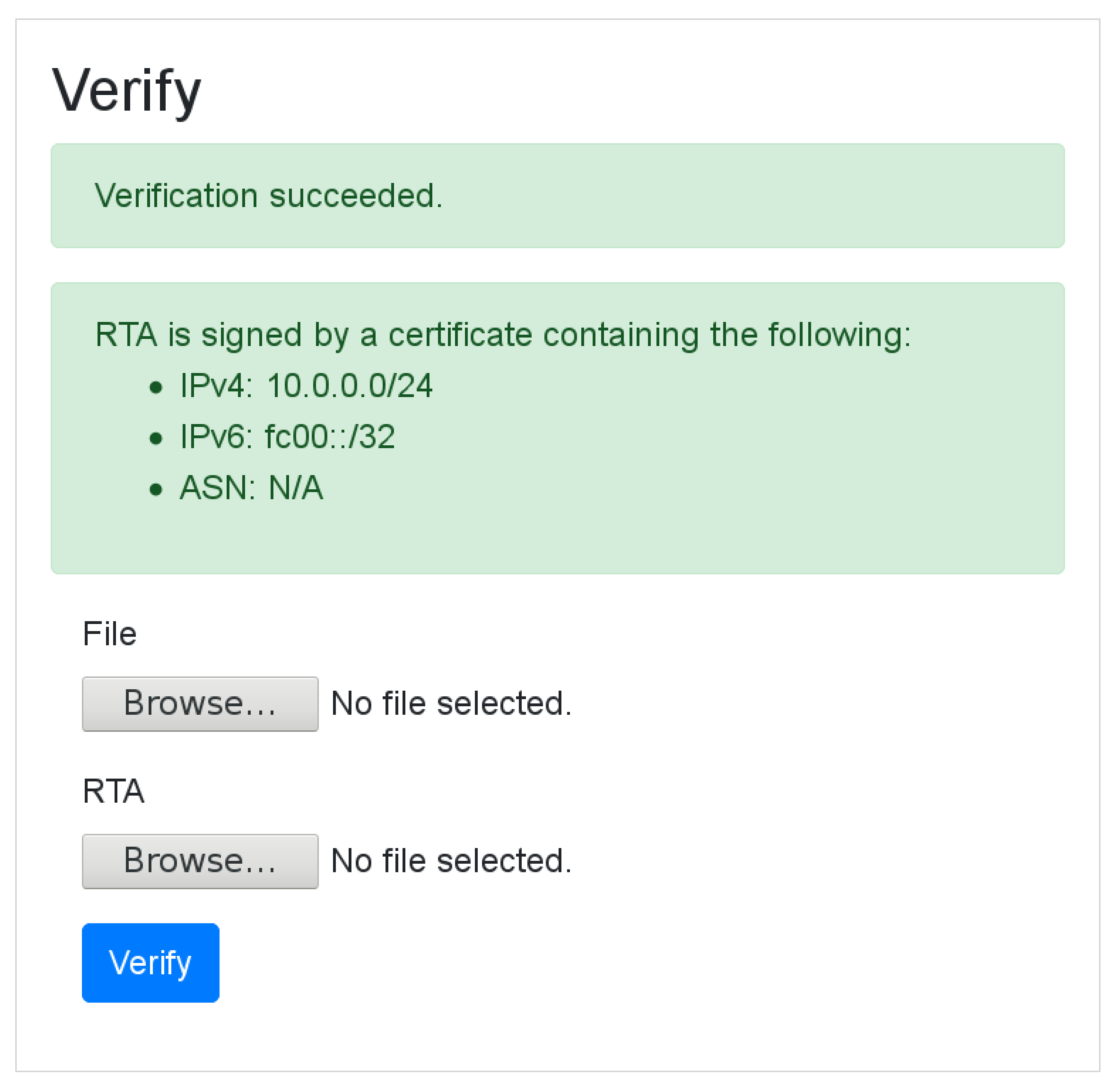
transfer-document.pdf

RTA

rta.cms

Verify

ASSOCIATED RESOURCES



Verify

Verification succeeded.

RTA is signed by a certificate containing the following:

- IPv4: 10.0.0.0/24
- IPv6: fc00::/32
- ASN: N/A

File

RTA

Associated Resources

WE HAVE CODE MODS FOR DRAGON RESEARCH

- Just needed a simple EE certificate export function
- We did git diffs but realistically a cleaner solution is needed
  - `https://github.com/APNIC-net/rpki.net/commit/890d47f6d56fadefcb5a5e14ac1115e11a168bcf`
- Integration into other RPKI production systems would be equally simple
  - Portal services should not have a problem
  - Signing large objects is best done locally hence key export
- CSR would be clean model
  - Specify resources to add to EE certificate
  - Private key stays private

WHERE TO NEXT?

WHERE TO NEXT?

- Please adopt this draft. We think its useful.
  - Proving authority over resources outside of BGP is useful
- We want to encourage use for BYO and related B2B uses
  - Which needs systems willing to issue an EE cert to a csr
  - and systems which export an EE cert and key from inside
  - and systems which implement RTA embedded inside themselves
- We think this is business-ready
  - BYO and other B2B is looking for this kind of service