SIDROPS
Internet-Draft
Intended status: Standards Track
Expires: 18 September 2026

J. Snijders
BSD
T. Bruijnzeels
RIPE NCC
T. Harrison
APNIC
W. Ohgai
JPNIC

17 March 2026

# The Erik Synchronization Protocol for use with the Resource Public Key Infrastructure (RPKI)

draft-ietf-sidrops-rpki-erik-protocol-04

## Abstract

This document specifies the Erik Synchronization Protocol for use with the Resource Public Key Infrastructure (RPKI). Erik Synchronization can be characterized as a data replication system using Merkle trees, a content-addressable naming scheme, concurrency control using monotonically increasing sequence numbers, and HTTP transport. Relying Parties can combine information retrieved via Erik Synchronization with other RPKI transport protocols. The protocol’s design is intended to be efficient, fast, easy to implement, and robust in the face of partitions or faults in the network.

## Status of This Memo

## Copyright Notice



## Table of Contents

## 1. Introduction

This document specifies the Erik Synchronization Protocol for use with the Resource Public Key Infrastructure (RPKI) [RFC6480]. Erik Synchronization can be characterized as a data replication system using Merkle trees [M1987], a content-addressable naming scheme [RFC6920], concurrency control using monotonically increasing sequence numbers [RFC0677], and HTTP transport [RFC9110]. Relying Parties can combine information retrieved via Erik Synchronization with other RPKI transport protocols ([RFC5781] and [RFC8182]). The protocol's design is intended to be efficient, fast, easy to implement [RFC1925], and robust in the face of partitions or faults in the network.

### 1.1. Background

The notion of cache-to-cache data replication of unvalidated data was documented in Section 3 of [RFC7115].

> Validated caches may also be created and maintained from other validated caches. Network operators SHOULD take maximum advantage of this feature to minimize load on the global distributed RPKI database. Of course, the recipient relying parties should re-validate the data.
>
> -- RFC7115, section 3

Historic records show that experiments have been performed in this space using, for example, peer-to-peer file sharing technology (see [P2P]), but no standardised and widely-deployed mechanism for cache-to-cache replication emerged since then. The authors hope that the Erik Synchronization protocol might be suitable to fill this gap and improve propagation speed of validly signed repository data as well as help reduce load on the global RPKI.

### 1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

### 1.3. Related Work

The reader is assumed to be familiar with the terms and concepts described in "Maintenance of duplicate databases" [RFC0677], "An Infrastructure to Support Secure Internet Routing" [RFC6480], "The RPKI Repository Delta Protocol (RRDP)" [RFC8182], "Manifests for the Resource Public Key Infrastructure (RPKI)" [RFC9286], "A Digital Signature Based on a Conventional Encryption Function" [M1987].

### 1.4. Glossary

This section describes the terminology and abbreviations used in this document. Though the definitions might not be clear on a first read, later on the terms will be introduce with more detail.

Erik relay  An intermediate between CA publication repositories and Relying Parties.

FQDN  The fully qualified domain name of a RPKI repository instance referenced in an end-entity certificate’s Subject Information Access (SIA) extension’s id-ad-signedObject accessDescription.

Hash  A message digest calculated for an object using the SHA-256 algorithm.

ErikIndex  The relay’s Merkle root for a given FQDN.  An ErikIndex is an ordered listing of ErikPartition object hashes.

ErikPartition  An ordered listing of the manifest objects’ hashes, manifestNumber values, thisUpdate values, and their certificates’ SIA extension values.

## 2.  Informal Overview

Erik Synchronisation is an architecture to reliably distribute RPKI repository data from cache to cache using so-called Erik relays. Relays maintain a validated cache themselves and can be clients of other relays.  While this property suggests that a group of relays should converge to the exact same state, the distributed nature of the RPKI prevents relays from achieving strict synchronization.

In this synchronization protocol, Merkle trees are used to determine whether differences exist between client and relay.  Merkle trees are hierarchical data structures: the hash value of each node is computed recursively by hashing the concatenated hash values of the node’s children.  The hash of the ErikIndex represents the entire dataset related to a given FQDN.  If the ErikIndex hash is not the same between two replicas, the relay provides the client with hashes of smaller and smaller portions of the to-be-replicated dataset until the exact list of out-of-sync or missing objects is identified. Sequence numbers are then used to determine whether these differences are relevant enough for the client to fetch.  All data, except for ErikIndex objects, is fetched using static addresses derived from object hashes.  This approach reduces unnecessary data transfer between caches which contain mostly similar data.

The client starts by making an inventory of its local state and then querying an Erik relay for the relay’s current ErikIndex for a given FQDN.  If the ErikIndex is different compared to the previous run (or compared to the ErikIndex derived from locally cached objects). After having acquired the current ErikIndex, clients can determine which ErikPartitions need to be fetched.  The client then can compare the _manifestNumber_ sequence number and _thisUpdate_ for each manifest listed in the ErikPartition, and proceed to fetch (purportedly) newer versions of manifests of interest.  Whenever a relay has manifests with a lower sequence number on offer, the client can ignore those.  The client now has sufficient information to proceed to fetch any missing Certificates, Signed objects, and CRLs. With the information contained within manifests, clients can fetch addressed by content (by hash) and store by name (or some other scheme).

## 3. Erik Synchronization Data Structure Definitions

The messages used in this protocol to construct and determine synchronization states are encoded following Distinguished Encoding Rules (DER) [X.690]. ErikIndex and ErikPartition objects are formally defined as follows.

```
RpkiErikSynchronization-2025
  { iso(1) member-body(2) us(840) rsadsi(113549)
    pkcs(1) pkcs9(9) smime(16) mod(0)
    id-mod-rpkiErikSynchronization-2025(TBD) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN

-- EXPORTS ALL --

IMPORTS
  CONTENT-TYPE, Digest, DigestAlgorithmIdentifier
  FROM CryptographicMessageSyntax-2010 -- in [RFC6268]
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs-9(9) smime(16) modules(0) id-mod-cms-2009(58) }

  AccessDescription, KeyIdentifier
  FROM PKIX1Implicit-2009 -- in [RFC5912]
  { iso(1) identified-organization(3) dod(6) internet(1) security(5)
    mechanisms(5) pkix(7) id-mod(0) id-mod-pkix1-implicit-02(59) }
;

ContentInfo ::= SEQUENCE {
  contentType      CONTENT-TYPE.&id({ContentSet}),
  content      [0] EXPLICIT
                     CONTENT-TYPE.&Type({ContentSet}{@contentType}) }

ContentSet CONTENT-TYPE ::= {
  ct-rpkiErikIndex | ct-rpkiErikPartition, ... }

ct-rpkiErikIndex CONTENT-TYPE ::=
  { TYPE ErikIndex IDENTIFIED BY id-ct-rpkiErikIndex }

id-ct-rpkiErikIndex OBJECT IDENTIFIER ::=
  { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
    pkcs-9(9) id-smime(16) id-ct(1) erikindex(55) }

ct-rpkiErikPartition CONTENT-TYPE ::=
  { TYPE ErikPartition IDENTIFIED BY id-ct-rpkiErikPartition }

id-ct-rpkiErikPartition OBJECT IDENTIFIER ::=
```

```
     { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
       pkcs-9(9) id-smime(16) id-ct(1) erikpartition(56) }

   ErikIndex ::= SEQUENCE {
     version [0]      INTEGER DEFAULT 0,
     indexScope       IA5String,
     indexTime        GeneralizedTime,
     hashAlg          DigestAlgorithmIdentifier,
     partitionList    SEQUENCE (SIZE(1..ub-Partitions)) OF PartitionRef
   }

   ub-Partitions INTEGER ::= 256

   PartitionRef ::= SEQUENCE {
     hash             Digest,
     size             INTEGER (100..MAX) }

   ErikPartition ::= SEQUENCE {
     version [0]      INTEGER DEFAULT 0,
     partitionTime    GeneralizedTime,
     hashAlg          DigestAlgorithmIdentifier,
     manifestList     SEQUENCE (SIZE(1..MAX)) OF ManifestRef }

   ManifestRef ::= SEQUENCE {
     hash             Digest,
     size             INTEGER (1000..MAX),
     aki              KeyIdentifier,
     manifestNumber   INTEGER (0..MAX),
     thisUpdate       GeneralizedTime,
     locations        SEQUENCE (SIZE(1..MAX)) OF AccessDescription }
   END
```

### 3.1. General Syntax

At the top level the content of an Erik object is an instance of ContentInfo.

#### 3.1.1. contentType

The contentType is an OID specifying the type of payload in the object, in this profile either id-ct-rpkiErikIndex or id-ct-rpkiErikPartition.

#### 3.1.2. content

The content field contains an instance of ErikIndex or ErikPartition.

### 3.2. ErikIndex

An ErikIndex represents all current manifest objects available under a given FQDN and thus the complete state of the repository as it is known to the relay.

#### 3.2.1. The version field

The version number of the ErikIndex object MUST be 0.

#### 3.2.2. The indexScope field

The indexScope field contains the fully qualified domain name of the Signed Object location of the manifests referenced through this particular ErikIndex. The FQDN MUST be in the "preferred name syntax", as specified by Section 3.5 of [RFC1034] and modified by Section 2.1 of [RFC1123].

#### 3.2.3. The indexTime field

The indexTime is the most recent partitionTime value among the ErikPartitions referenced from the ErikIndex at hand. Use of the most recent partitionTime value (rather than the local system's notion of "now") provides idempotency for distributed system implementations. The field's value is a rough indication when the ErikIndex was generated and can be used for troubleshooting and measurement purposes.

For the purposes of this profile, GeneralizedTime values MUST be expressed UTC (Zulu) and MUST include seconds (i.e., times are YYYYMMDDHHMMSSZ), even where the number of seconds is zero. GeneralizedTime values MUST NOT include fractional seconds. See Section 4.1.2.5.2 of [RFC5280].

#### 3.2.4. The hashAlg field

This field contains the OID of the hash algorithm used to hash the ErikPartitions. The hash algorithm used MUST conform to the RPKI Algorithms and Key Size Profile specification [RFC7935].

#### 3.2.5. The partitionList field

This field is a sequence of PartitionRef instances. There is one PartitionRef for each current ErikPartition. Each PartitionRef is a tuple consisting of the hash of the partition object and the size of the partition object.

Information elements are unique with respect to one another and sorted in ascending order of the hash.

### 3.3. ErikPartition

An ErikPartition represents the set of manifest objects under a given FQDN which have the same first octet of the Authority Key Identifier (AKI) in common with each other. Each ErikPartition is an ordered listing of ManifestRef instances which contain the manifest objects' hashes, object sizes, AKI, manifestNumber values, thisUpdate values, and their respective end-entity certificates' SIA extension values. Using the first octet of the AKI as the partition key facilitates even and deterministic distribution of ManifestRefs across ErikPartitions.

#### 3.3.1. The version field

The version number of the ErikPartition object MUST be 0.

#### 3.3.2. The partitionTime field

The partitionTime is the most recent thisUpdate value among the manifests referenced from the ErikPartition at hand. Use of the most recent manifest thisUpdate value (rather than the local system's notion of "now") provides idempotency for distributed system implementations. The field's value is a rough indication when the ErikPartition was generated and can be used for troubleshooting and measurement purposes.

For the purposes of this profile, GeneralizedTime values MUST be expressed UTC (Zulu) and MUST include seconds (i.e., times are YYYYMMDDHHMMSSZ), even where the number of seconds is zero. GeneralizedTime values MUST NOT include fractional seconds. See Section 4.1.2.5.2 of [RFC5280].

#### 3.3.3. The hashAlg field

This field contains the OID of the hash algorithm used to hash the manifest objects referenced in this ErikPartition. The hash algorithm used MUST conform to the RPKI Algorithms and Key Size Profile specification [RFC7935].

#### 3.3.4. The manifestList field

This field is a sequence of ManifestRef instances. There is one ManifestRef for each current manifest. A manifest is nominally current until the time specified in nextUpdate or until a manifest is issued with a greater manifestNumber, whichever comes first (see Section 4.2.1 of [RFC9286]).

A ManifestRef is a structure consisting of the hash of the manifest object, the size of the manifest object, the manifest issuer's key identifier, the manifestNumber, and the thisUpdate contained within the object, and a sequence of AccessDescription instances from the manifest's End-Entity certificate's Subject Information Access extension.

Information elements are unique with respect to one another and sorted in ascending order of the hash.

## 4. Client-side Processing

Clients start by acquiring and processing an ErikIndex, which represents the relay's current Merkle tree head for a given FQDN. A client MUST verify the requested FQDN exactly matches the indexScope value in the ErikIndex, and if not proceed to use a different relay.

Then, clients can decide whether or not to fetch ErikPartition objects listed on the ErikIndex, for instance, by checking whether the object associated with the hash was already fetched at some point in the client's past.

Before using a ErikPartition, the client MUST verify that all URIs in the accessLocations in the id-ad-signedObject accessMethod instances in the ErikPartition are encompassed in the requested indexScope. A client can then decide whether or not to fetch a given manifest object, by comparing the manifestNumber and thisUpdate with what's locally cached and what's offered by the remote relay.

A client can compute which products listed in the manifest's fileList need to be fetched from one relay or another in order to achieve a successful fetch. A client MUST verify that the URI in the accessLocation in one of the id-ad-signedObject accessMethod instances in the manifest's Subject Information Access (SIA) is encompassed in the requested indexScope.

As there is no concept of 'sessions' (like in RRDP), clients can interchangeably use different Erik relays. When one Erik relay generates a HTTP error, the client can try fetching the requested object from another Erik relay. To improve reliability, clients should alternate among different relays in successive query and fetch attempts.

Clients SHOULD use HTTPS, unless the local system is unable to establish TLS connections to any of its configured Erik relays.

## 5. Querying an Erik Relay

### 5.1. Fetching objects by hash

This specification uses "Named Information" identifiers mapped to .well-known HTTP/HTTPS URLs for object retrieval, as described in [RFC6920].

For example, issuance #54 of ripe-ncc-ta.mft has the following SHA256 digest:
c2d0427bc5a32c42eea1ab5663d592b1fc29c7d4ef16ab0b5e1d631d039dcc21.

To fetch the aforementioned object from an relay hosted at relay.example.net, a client would access the following HTTP URL:
https://relay.example.net/.well-known/ni/sha-256/
wtBCe8WjLELuoatWY9WSsfwpx9TvFqsLXh1jHQOdzCE

Responses to requests for paths starting with /.well-known/ni/ are immutable.

### 5.2. Fetching ErikIndex objects

The URIs to fetch ErikIndex objects can be constructed using the following Well-Known URI template with the relay's hostname as authority, the erik keyword as suffix, the string /index/, followed by a FQDN as parameter, i.e.: https://{relay_hostname}/.well-known/erik/index/{FQDN}.

As an example, the URI to fetch an ErikIndex for the rpki.ripe.net FQDN from a relay at relay.example.net would be:
https://relay.example.net/.well-known/erik/index/rpki.ripe.net.

A client MAY use the If-Modified-Since HTTP header when fetching ErikIndex objects.

Responses to requests for paths starting with /.well-known/erik/index/ are mutable.

## 6. Prefetching Objects in Bulk

Object prefetching is a bulk distribution mechanism for data that clients might need in the near future. Prefetching is useful for efficiently bootstrapping empty caches and for clients to catch up on recently discovered objects with a single bulk request.

This section outlines the general structure of prefetch request responses and defines various .well-known/erik/ endpoints.

### 6.1. General Structure of Prefetch Responses

In this protocol, a Prefetch Response (PR) is simply a concatenated sequence of zero or more DER-encoded objects in gzip [RFC1952] compressed form. Because DER-encoded objects are self-delimiting, no marker is used between the objects and there is no end-of-sequence indicator.

A Prefetch Response is non-deterministic: objects contained within the PR may appear in arbitrary order or in duplicate. A PR may contain RPKI signed objects, ErikIndex objects, and ErikPartition objects. The set of objects in a PR need not be self-consistent or complete from the perspective of top-down validation. PRs merely serve to reduce network traffic by prefill unvalidated cache areas.

Different prefetching strategies have different trade-offs in terms of coverage and accuracy. On the one hand aggressive prefetching might waste bandwidth (i.e. clients download data they might not end up using, or data they had already cached), on the other hand, sending many individual requests for each and every object increases network overhead and latency. As a rule of thumb: FQDN Snapshots (Section 6.2) are useful for clients with no prior state and Time-trimmed Tail Queues (Section 6.3) are useful in the steady state (when synchronizing every few minutes).

For efficiency, relay implementations are RECOMMENDED to combine as many DER-encoded objects into as few gzip members as possible (see Section 2.3 of [RFC1952] for information on gzip members).

### 6.2. Prefetching FQDN Snapshots

A snapshot contains all RPKI & Erik objects a relay associated with a given FQDN as of the time of the snapshot’s production. Snapshots are not necessarily produced in lockstep with updates to the relay’s merkle trees, therefore clients MUST perform a tree comparison after fetching a snapshot.

As an example, the URI to fetch a snapshot for the rpki.ripe.net FQDN from a relay at relay.example.net would be: https://relay.example.net/.well-known/erik/snapshot/rpki.ripe.net. Responses to requests for paths starting with /.well-known/erik/snapshot/ are mutable.

Relays MUST NOT include ErikIndex and ErikPartition objects in snapshots.

See Appendix C for an example how to construct a FQDN snapshot.

### 6.3. Prefetching Time-trimmed Tail Queues

A relay’s time-trimmed tail queues are buffers that contains all RPKI & ErikPartition objects that relay discovered in the last 5 and 10 minutes, respectively. Such buffers are not necessarily produced in lockstep with updates to the relay’s merkle trees, therefore clients MUST perform a tree comparison after fetching a tail queue buffer.

As an example, the URI to fetch the 5 minute tail from a relay at relay.example.net would be: https://relay.example.net/.well-known/erik/tail/5min. Responses to requests for paths starting with /.well-known/erik/tail/ are mutable.

Relays MUST NOT include ErikIndex objects in time-trimmed tail queue buffers.

See Appendix D for an example how to pre-calculate Prefetch Responses using a time-trimmed tail queue concept.

## 7. Transport Error Detection and Handling

The client MUST calculate the hashes of fetched objects and verify they are the same as the expected hashes (which are embedded in the URIs through which the objects were retrieved). If there is a hash mismatch, the client may try fetching the object from a different Erik relay or treat this as a _failed fetch_ (see Section 6.6 of [RFC9286]) and try again at a later point in time in a next validation run.

## 8. Setting Up an Erik Relay

Erik relays can be operated by any party, without permission from or coordination with publication point operators or CAs. Relays are made accessible via either HTTP or HTTPS or both.

Relays generate and make accessible ErikIndexes and ErikPartitions derived from their current validation state, the client then cherry-picks which objects (if any) it wishes to fetch.  In turn, relays fetch fresh data from other relays, or from CA-designated publication points accessible via Rsync ([RFC5781]) and RRDP ([RFC8182]).

## 9. Comparison with other RPKI transport protocols

Ignoring obvious mechanical "on the wire" differences between Erik, Rsync, and RRDP; there are a number of concept differences between the protocols.  Rsync and RRDP can be described as "general purpose" synchronisation protocols: they could be used to transfer any arbitrary set of files, on the other hand the Erik protocol is RPKI-specific: part of its signaling layer are RPKI manifest objects, which RPs require as recourse for validation anyway.  This property by itself causes a small deduplication in the data to be transferred.

### 9.1. Comparison with Rsync

In Rsync, the server and the client construct and transfer a full listing of all available objects, and then transfer objects as necessary.  In effect, this allows clients to 'jump' to the latest repository state, regardless of the state of the local cache.

A major downside of Rsync is that the list of files itself can become a burden to transfer.  As of June 2025, in order to merely establish whether a client is synchronized or not with the RIPE NCC repository at rpki.ripe.net, as much as 5.8 megabytes of data are exchanged without exchanging any RPKI data.

Experimentation suggests that when synchronizing once an hour, Erik consumes less network traffic than Rsync generally would consume which, in turn, is less network traffic than RRDP would.

### 9.2. Comparison with RRDP

The key concept in RRDP is that the client downloads a "journal", containing all add/update/delete operations and replays this journal to arrive at the current repository state.

A major downside of RRDP is that (depending on the RRDP polling interval) clients end up downloading data which has become outdated. Imagine a hypothetical CA which issues and revokes a ROA every 10 minutes and a client that synchronizes every 60 minutes; in effect the client must fetch 5 outdated states, wasting bandwidth.

Experimentation suggests that when synchronizing every 15 minutes, Erik consumes less network traffic than RRDP generally would consume which, in turn, is less network traffic than Rsync would consume.

#### 9.2.1. Garbage Collection

In contrast to RRDP, the Erik protocol has no concept of server-specific "stateful" sessions that persist across polling attempts. This obviates the need for withdraw instructions as part of the protocol exchange: clients can simply delete objects that are no longer referenced from their current validation state and refetch them later on if needed.

## 10. Operational Considerations

### 10.1. Scaling considerations

As of July 2025, the global Internet's RPKI churn rate appears to be 2 new objects per second. The ecosystem is estimated to be composed of ~ 5000 RPKI cache instances and ~ 50 repository servers. Assuming 10 minute fetching intervals and 150 metadata requests per synchronization run (for exchange of Merkle tree data), an Erik relay serving all the Internet's RPKI cache instances would probably need to be able to sustain serving an average of at least 11,000 HTTP requests per second. This order of magnitude in terms of scaling requirements can easily be handled by a single commodity server.

### 10.2. HTTP Compression

Using gzip compression on average tends to yield a 20% reduction in RPKI object size when fetching individual objects, therefore it is RECOMMENDED for clients and relays to offer support for compressed content coding, as described in Section 8.4.1 of [RFC9110]. Prefetch responses (Section 6) are always in gzip compressed form.

Using a previous version of a RPKI object as a compression dictionary for a newer version enables delivery of a delta-compressed version of the changes, usually resulting in significantly smaller responses than what can be achieved by compression alone. Clients can facilitate delta compression by sending an Available-Dictionary request header, using a previously fetched version of the RPKI object as the dictionary. It is RECOMMENDED for clients and relays to make use of Compression Dictionary Transport ([RFC9842]).

## 11. Security Considerations

This document makes no changes to RPKI certificate validation procedures.

Paraphrasing Section 11 of [RFC6810]: The RPKI relies on object, not server or transport, trust. That is, the Regional Internet Registry root trust anchors are distributed through some out-of-band means, and can then be used by each relying party to validate certificate chains and Signed Objects. The inter-cache relationships are based on this object security model; hence, any cache-to-cache transport is assumed to be unreliable at times. See Section 5 of [RFC8182] for more security considerations.

To avoid certain forms of replay attack, clients MUST verify purported indexScope, ManifestRef location values, and manifest Subject Information Access (SIA) extensions match the expected FQDN.

Byzantine events or faults in relay-to-client communication can be overcome by the client rotating requests for objects among different Erik relays.

## 12. IANA Considerations

### 12.1. S/MIME Module Identifier

The IANA is requested to add an item to the "SMI Security for S/MIME Module Identifier" registry as follows:

| Decimal | Description | References |
|---|---|---|
| TDB | id-mod-rpkiErikSynchronization-2025 | [this-draft] |

### 12.2. SMI Security for S/MIME CMS Content Type

The IANA has allocated for this specification in the "SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)" registry as follows:

| Decimal | Description | References |
|---|---|---|
| 55 | id-ct-rpkiErikIndex | [this-draft] |
| 56 | id-ct-rpkiErikPartition | [this-draft] |

Upon publication of this document, IANA is requested to reference the RFC publication instead of this draft.

### 12.3. Well-Known URI

IANA is requested to assign an URI Suffix for Erik Synchronization in the Well-Known URIs registry as follows.

URI Suffix  erik
Reference:  [this draft]
Status:  permanent
Change Controller:  IETF
Related Information  -

#### 12.3.1. Protocol Registries Request at Github

This section is to be removed before publishing as an RFC.

See https://github.com/protocol-registries/well-known-uris/issues/67 for a copy of this request.

### 12.4. Media Types

IANA is requested to register the media types "application/rpki-erikindex" and "application/rpki-erikpartition" in the "Media Types" registry as follows.

#### 12.4.1. ErikIndex Media Type

Type name:  application
Subtype name:  rpki-erikindex
Required parameters:  N/A
Optional parameters:  N/A
Encoding considerations:  binary
Security considerations:  This media type contains no active content.
Interoperability considerations:  N/A
Published specification:  draft-ietf-sidrops-erik-protocol
Applications that use this media type:  RPKI operators
Fragment identifier considerations:  N/A
Additional information:
Content:  This media type is a RPKI ErikIndex object, as defined in draft-ietf-sidrops-erik-protocol.
Magic number(s):  N/A
File extension(s):  N/A
Macintosh file type code(s):  N/A
Person & email address to contact for further information:  Job Snijders (job@bsd.nl)
Intended usage:  COMMON
Restrictions on usage:  N/A
Author:  Job Snijders (job@bsd.nl)
Change controller:  IETF

#### 12.4.2. ErikPartition Media Type

Type name:  application

Subtype name:  rpki-erikpartition
Required parameters:  N/A
Optional parameters:  N/A
Encoding considerations:  binary
Security considerations:  This media type contains no active content.
Interoperability considerations:  N/A
Published specification:  draft-ietf-sidrops-erik-protocol
Applications that use this media type:  RPKI operators
Fragment identifier considerations:  N/A
Additional information:

Content:  This media type is a RPKI ErikPartition object, as defined in draft-ietf-sidrops-erik-protocol.

Magic number(s):  N/A

File extension(s):  N/A

Macintosh file type code(s):  N/A

Person & email address to contact for further information:  Job Snijders (job@bsd.nl)
Intended usage:  COMMON
Restrictions on usage:  N/A
Author:  Job Snijders (job@bsd.nl)
Change controller:  IETF

## 13. References

### 13.1. Normative References

[RFC1034]  Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[RFC1123]  Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, DOI 10.17487/RFC1123, October 1989, <https://www.rfc-editor.org/info/rfc1123>.

[RFC1952]  Deutsch, P., "GZIP file format specification version 4.3", RFC 1952, DOI 10.17487/RFC1952, May 1996, <https://www.rfc-editor.org/info/rfc1952>.

[RFC2119]  Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC6810] Bush, R. and R. Austein, "The Resource Public Key Infrastructure (RPKI) to Router Protocol", RFC 6810, DOI 10.17487/RFC6810, January 2013, <https://www.rfc-editor.org/info/rfc6810>.

[RFC6920] Farrell, S., Kutscher, D., Dannewitz, C., Ohlman, B., Keranen, A., and P. Hallam-Baker, "Naming Things with Hashes", RFC 6920, DOI 10.17487/RFC6920, April 2013, <https://www.rfc-editor.org/info/rfc6920>.

[RFC7935] Huston, G. and G. Michaelson, Ed., "The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure", RFC 7935, DOI 10.17487/RFC7935, August 2016, <https://www.rfc-editor.org/info/rfc7935>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9110] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/info/rfc9110>.

[RFC9286] Austein, R., Huston, G., Kent, S., and M. Lepinski, "Manifests for the Resource Public Key Infrastructure (RPKI)", RFC 9286, DOI 10.17487/RFC9286, June 2022, <https://www.rfc-editor.org/info/rfc9286>.

[X.690] ITU-T, "Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, ISO/IEC 8825-1:2021, February 2021, <https://www.itu.int/rec/T-REC-X.690-202102-I/en>.

### 13.2. Informative References

[M1987] Merkle, R., "A Digital Signature Based on a Conventional Encryption Function", Advances in Cryptology -- CRYPTO ’87 Proceedings, Lecture Notes in Computer Science, Vol. 293, DOI 10.1007/3-540-48184-2_32, 1988, <https://doi.org/10.1007/3-540-48184-2_32>.

[P2P] Austein, R., Bush, R., Elkins, M., and L. Johansson, "RPKI Over BitTorrent", March 2012, <https://www.ietf.org/proceedings/83/slides/slides-83-sidr-9.pdf>.

[RFC0677] Johnson, P. and R. Thomas, "Maintenance of duplicate databases", RFC 677, DOI 10.17487/RFC0677, January 1975, <https://www.rfc-editor.org/info/rfc677>.

[RFC1925] Callon, R., "The Twelve Networking Truths", RFC 1925, DOI 10.17487/RFC1925, April 1996, <https://www.rfc-editor.org/info/rfc1925>.

[RFC5781] Weiler, S., Ward, D., and R. Housley, "The rsync URI Scheme", RFC 5781, DOI 10.17487/RFC5781, February 2010, <https://www.rfc-editor.org/info/rfc5781>.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <https://www.rfc-editor.org/info/rfc6480>.

[RFC7115] Bush, R., "Origin Validation Operation Based on the Resource Public Key Infrastructure (RPKI)", BCP 185, RFC 7115, DOI 10.17487/RFC7115, January 2014, <https://www.rfc-editor.org/info/rfc7115>.

[RFC8182] Bruijnzeels, T., Muravskiy, O., Weber, B., and R. Austein, "The RPKI Repository Delta Protocol (RRDP)", RFC 8182, DOI 10.17487/RFC8182, July 2017, <https://www.rfc-editor.org/info/rfc8182>.

[RFC9842] Meenan, P., Ed. and Y. Weiss, Ed., "Compression Dictionary Transport", RFC 9842, DOI 10.17487/RFC9842, September 2025, <https://www.rfc-editor.org/info/rfc9842>.

[rpki-erik-demo]
Harrison, T., "rpki-erik-demo", March 2026, <https://github.com/APNIC-net/rpki-erik-demo>.

[rpkitouch]
Snijders, J., "rpkitouch", December 2025, <https://www.github.com/job/rpkitouch>.

# Appendix A.  Implementation status

This section is to be removed before publishing as an RFC.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs.  Please note that the listing of any individual implementation here does not imply endorsement by the IETF.  Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors.  This is not intended as, and must not be construed to be, a catalog of available implementations or their features.  Readers are advised to note that other implementations may exist.

According to RFC 7942, "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

## A.1.  Experimental Erik Relay Services

A few experimental Erik relays are available, each running on slightly different schedules.  Client implementers are encouraged to round-robin between these instances to observe results.

http://relay.rpki-servers.org/  Dublin, Osaka, Sydney - anycasted distributed computing cluster

http://dub.rpki-servers.org/  Dublin, Ireland, - distributed computing cluster (6 machines, NFS backend)

http://atl.rpki-servers.org/  Atlanta, USA, - distributed computing cluster (2 machines, NFS backend)

http://miso.sobornost.net/  Amsterdam, NL, single node

http://nyc.rpki-servers.org/  New York, USA, - single node

http://fnllwqoupfrhso6643whm6lpkgsftjtc6crpehmyz2o7pffirnqy7rad.onion/ Erik relay service via Tor

### A.2. Software

An experimental Erik static content generator was developed by Job Snijders in the form of [rpkitouch] using C.

An experimental Erik client and server were developed by Tom Harrison from APNIC in the form of [rpki-erik-demo] using Perl.

## Appendix B. Example objects

Included in this section are an ErikIndex for rpki.ripe.net and an ErikPartition referenced from the aforementioned ErikIndex, both Base64 encoded.

### B.1. Example ErikIndex

This object was retrieved from http://miso.sobornost.net/.well-known/erik/index/rpki.ripe.net.


```
MIIoRgYLKoZIhvcNAQkQATeggig1MIIoMRYNcnBraS5yaXBlLm5ldBgPMjAyNjAxMDgyM
zIwNTRaMAsGCWCGSAFlAwQCATCCKAAwJgQgteOE8pPUend8kUR6qmLyVUJW58GNqxuv9u
J7hNLi8kYCAkJ4MCYEIHaraMXmjcDabOQIkh+26R7qgymR+xJGjAAAnB7iDzekAgJIRjA
mBCD7A2eyMRTOhBOw+K1awbIDfSIWDqEfx2wwnH/ssOcaDgICQNEwJgQg6r3dpoqqtP18
48s1V2ll7e6xQ1bOLleJ5dW6QHh4r6cCAkrBMCYEICMCi/TwPWOH4JqX3dvNAmS8+m8a8
uCfcpHckqhKzq49AgI+VzAmBCBdf+oy9h6oYy7Ni1FLcjE/FT55ZAVKbphxoAd6uAu5CQ
ICThMwJgQgxYjeLOzCuVG37kP7lx6tE0tzp9M6U/NxS0p0a+5ZIb4CAkNOMCYEIBd1/77
YpybBiJDIW41tx6qFg3HD7zUg9oKrqWTI6F6JAgJLljAmBCAsmgS1P20sDJfK8jzBye8a
gUStHxjRT6Vjvawtm5HhHgICS5YwJgQgzlGoWkadYriQpcBRR9tn+opJ0x9OsSa9RSk+x
Jb2fWsCAk4SMCYEILYFTuRIifKf922SiNtmPnVVHLNnun2RSe2vFvhSeoDsAgI7BzAmBC
CMkyd1oQFMenpKU88Dg7TEP8w0sxH20rnPAroelJbTKQICSRowJgQgDByPsQbQE9DcyeK
Lu27VICKrtBMkJk/OygYTvvVZyKACAj5TMCYEIHMAxD5xBnppLv2EuLnuRbktX9f58Zyu
OWZw4Bj5En1TAgJNPzAmBCDkEwmCGqAjsZPkfUm5i0tmDua7rEHvJcYa95fS68ZQuwICR
PYwJgQgqiRd4u6+mbQLT7maVQnu6k/2k2X8a9sW57/6+OehXigCAkGjMCYEIE4SnNsvI4
rlXxOR5ZEe/Pwvt6bjnSaoCif7ab5Hbjv+AgJGnDAmBCBmKBJTmC3J/Sj7NnFWZk7Q234
riaaXykvRD20oty0ZqgICRp8wJgQg5Fgfag3shlS+jmdPySCaGbL3MRLHtjqK4esqnimO
2VUCAkT1MCYEIHrthfZ3L6haWgUGgunM4WPYaN3CB9Fi5kHjfq0MO9bRAgJGnjAmBCBGj
seJNUXkpN6OMKV2CEBvsqi7SWSsIe69xCg8pY6a0AICM5IwJgQgF0ORPc6Z4wjWPrk0I2
GNZUI5Ydq2i94ujoa6LpOKR5oCAkhHMCYEIDogoD9JqhB/qYYJ0lk0Ot2oVWe0KAku6dR
A0PDhp5twAgJLlTAmBCBRp1q9Jjh8HPtrCdJuoHlZSmJesaMH4gmCgbbSzRLwqgICUjUw
JgQg2x86JI1NWZ0Isw2G81u/TFIacQcMUP5KY+7oBwLnAusCAjlcMCYEIN6ald+lseUkl
fTphACppwLpSg/sO0o/vpFtbj3Rji5pAgJSMjAmBCBU7TRmN2Ol5MsO4N6KRl/2BAQhpI
EAmx5ointD8kXkHwICUjQwJgQgae2HBmn6vf2fJT/y7XbnuWo6NUXN3nYDT3g/zsGT9M0
CAkQjMCYEIL2OHInTlcfJctbeu0VfAzY/KFwFPhttEozeCqAVq8ldAgJOEjAmBCDAl869
bttF9BYGOwK/kkf7npDCinlMpu3jejhVg77aVwICSewwJgQgdV5dr99Mesm82uSSaTr5f
laqjDpG48S+asnR74SKitwCAkxpMCYEIK3vJ8icN4xK9gqlg9ad3kiz/hUfIP4ebZo8wV
TDBIubAgJJGzAmBCAYmoT9JryRbijv5nnKOc9nq9DxN9J1K98QKcgzaMvD5wICSe4wJgQ
gfcDoA5zq2u3IJjQW+dLoe7PsrhYewA4VkR3qjFt4e78CAjytMCYEIKWjW2RIbdezwJGI
YVw6LfRpcPjs1V+A8+joGG3aT/NoAgI9gjAmBCBpdyBWzebD4iGbFSVavuuoVvfp89943
```

```
RBgifoXP6gxhwICU90wJgQgq8PPWUFcKSmRSWxExdOiJKGC1ER3PAWLc2GQHL7R85ICAj
2DMCYEIO3d1sApCGkBXz/06+eCwygPBTa/IaCEYxSXgc/UfcQ1AgJFyDAmBCCSk5WYxB4
hU6doIMpLrfI9vKZlGyWKLUQGpqpyVZn81wICQacwJgQgf/kA9qoqMY3k2D8rzCN7YBvh
6N8aPXo3uYYB4kw+yXUCAlI1MCYEIOcHtxPyWHA/qgkca09niQc0bQJMBUM6Pt6GTCvzx
sFyAgI9gjAmBCA0Qg1uJakxAqpuGUq+Kw/7HIpqiWpzm+ej0rcSzu7ujwICWn4wJgQg61
JIv53XhGq0AA9B12oCQ+s1JvOoM6oU4DX9txhJj2QCAjyJMCYEIOIPL03mqDvKULLGm7G
h81QqgGKvR7bFrTiKDtKTMXsdAgI9gTAmBCDl0+OCKZRKXXw+zrqoBJRZcjT3RbhQ3jXa
k0ETShkX7QICO9owJgQgl8Hqw9QFeGH69AdgpwBw+0Ul08iSBGCGC36vfeG5kYsCAjsGM
CYEILWPQn2IZLnWpVryPboErGugOso6VpmyxkrvrCfeYiMIAgJGnTAmBCA6LZX3QfyOiZ
v5dXllXBMENaFNn5hwG6QYkk1il4dlKAICTucwJgQgvKcMhYPi4NdxvYa9HBSZ6SONWgY
AadPHtVxQYwq61VICAkdzMCYEIMeorZamZ7UWPcViw8Voj8eeHpID1LcYMdbOZozX7Bd2
AgI/KTAmBCDPQMe0Evnwn1NrrYNZPPZXloPO7Z1vF5m5J2NraaNUjAICS5YwJgQg5Zmpf
uETw3YjQIpIFSSrqtKcxlVR04jYLTqFSGTTvRcCAj5VMCYEIPRrKLcHW6zTeJCWXquLqp
SHz9OcNORMFkbw4HhAchc/AgJLljAmBCBp2AI4s92ufELx1TMCSlUjastbar5/bXLQ7l2
9xzi83QICP/4wJgQgvFOEtAchlJaDaym9hOU63BQ48+cpY5CtzBtL/90ZgQwCAkrCMCYE
IDs8UFQ9qbtwcHs2cYpjEyGWrhv3Ai5C1oKRUUx526BnAgI+VzAmBCC7SbJIm5/2L6znj
g6FvqgWRsFRi2TO8fw3r20XA5p6HgICSEcwJgQgh7yY3te3gI0LKK+l3vA3XbbvEN97X0
Mbae/+rwtVQSYCAkhFMCYEIKaVKX9r+hEBLo/v1n1wFj2q+UUWewcrjIt+yov3isdLAgJ
GnDAmBCBDkUOMPLGnT0K8zu1YXpnbUm2M98F4QvKh1rfw4dyuvQICQ00wJgQg9M7/tmwh
nE92zsVYz/S8FWbpCCfciGMlfNPdu1MFQ2wCAkDRMCYEIGjYKfRU3i3BQoKN0nFHq5zuB
Cg3/qh7POEtoI8e1WTZAgI//zAmBCC2JMivUXv3/zIsUiEkXyBohgyzf03HTaik2EyJrI
q9uAICTGowJgQg4h1xw2qVNaDfMH8SfbB4w5Fyps/cMYmBuYxWeT4zRHkCAjbiMCYEIM/
8ZhZ3CbuEGwBuTFAEbnP9nBPoGSNtAg92MH5F2ikdAgJO5TAmBCDdIeMvdsibEFcMMDnw
vGBadMXSLM3YQOfcriBNzgN2UgICTGYwJgQgllY7gmnDjFFvmpDMKrqYKDshKYKcpKYvI
CWvB8C/B+YCAkkZMCYEIAIQEQ7SkajiaEjZFxsmf0KworoXAjV4XXaZ+hqKjhDLAgJFyj
AmBCCTZe0oogg10rpxb9RThmOPxVeBzywrslOoVrWwjXvzVgICTGswJgQg7mGTp/zvlRJ
btFyXIekSQ6HQ58tfSpkot3HmzoFLD+cCAjleMCYEIFmaxCT52zYoJHpd1dt2zPPYZD0y
izbyV9MnitmLQ3MgAgJIQzAmBCACNx3IoeoqeqclIcXFSLmRHKQA2x9lsUcChkQDUO93Y
gICQ08wJgQgurGtxGxiz9efkspthtOoyyMojRsXXbMadlAEpuAOdzYCAj8rMCYEIGP/gT
JbZBCFCtbEzBCsmf9rSe+Ec9W14loYdo4LnzBfAgJJGjAmBCCQRnyclxcMJ42wDovHo2S
jazfcg7d5ir3kK0dfCZQZMgICP/4wJgQggeYSilmj9A5COVGCvujFMa3BNvJn1JNzbd2C
O7sXq/QCAkafMCYEIL7l0nzPqj6mFyzSMfcN9W33Pc7cYjRJVbwsLQ6jx5hyAgJO5jAmB
CBAUHuJhjtDBF7EBYqUf8yxdFHLE2HKuNanb1i/hkSwaAICRPYwJgQgzFnz5ckeEYWFQG
kPhX4b05boAuo0ZOgo2icxLW7fTyUCAk09MCYEIMMCMTiOMgi9TO8HQUW6012+X/lwnAn
uVsZdHDQGGEm/AgI//jAmBCBVzs23DRHsMUoMF48qIjnbuvggLBSZh9vNMA4DonV5EgIC
RcowJgQgB/budfyslLESuj91pUDFEdU/UA6oNjORbWYL2lqCj5sCAj2AMCYEIIJfegBmy
EQJdTVGQf8JvNoAXkC0OwoLimbZAf/c5onVAgJJGjAmBCCPiXNTbCqHbBMS840cT2BDdP
pPK5KRlPmLaPtsOliSbQICVy4wJgQgKke+YEYEGccTsCwQ4qosckkOEOtpLdYH/e/yOk+
wyVUCAkxpMCYEIDsO9ouxgT6ZDTZ/2bfabpoQjUTlFlrVK5EoTwjBhszBAgJBpjAmBCCm
RG8jmSGhz6qX04UyMO7sXE9dHwdZo/ouzvcCzi+WEgICMeswJgQglruzQ8bh7Kyt8uqdU
RlJ3nJ6DIYeKOZ/iv6bzMHvJvICAlI3MCYEIP5HU9JnOUWo0uP/il8IyWlsS2oh2YEOpW
seOq7ncDUjAgJMazAmBCCnowTNTOXdmORzq4MNkc81HHVo8QRswrDoJP+ZxmxUEQICQac
wJgQgp0y7WiuZPFoiK30xAPD7p6UHDwVYVX16ptfn3xB1a54CAkNPMCYEIIxtpX4DFbOC
M2/b9v9SijhehnvTUvqg4LTgRU3bx9iKAgJA0zAmBCDF7fDp6ceTkPS6QisrUBSapQX3b
EX7yDWpcZN86cRLrQICO9owJgQg1CQ4vdZezesq8ySPTCt5ZNv6UIBa+uubuUFhMUkhhd
kCAkQhMCYEIPc/PBxijzP1QKlIhoEbynPyplNcWNKOxD1sKxJ3s+kIAgJJGzAmBCB6Ci1
A3Dd2J/7RQKZSCKttO/6UZ8I80FRtt+E41bB+hAICQNIwJgQgnI7yJ7eaz4lKHbuQH0rv
TCLrjD6mMvxRRt3Gj79qZxsCAk4TMCYEIC185VSnjtR6lPTGbJTUjek9GxHkMgkUb8qaE
```

liMfXk/AgJGnDAmBCBF11WiXA+oi6JUeBpmKzJzplr0h6aTVk4Ap0rGUv6YjQICRCEwJg
QgUHkoaa053NIb6rBjXeCKaEWa3lRWwRqlYDZopKZBCUoCAj//MCYEIOfQKLO6T2FsUw0
+REFT5ST8h23c0Us+qcfCwP3ASMm2AgJBpjAmBCADKyjXhHu/uR06mEl/3u9w412q9ILN
2LMrz6fHsocZgAICQacwJgQgvHDCqBL+Cd9NkUo2cHtIJo3hJLbXyTTjNd4s6dyYVqECA
k4SMCYEICStmRJXJw+3RykPvfaJhfksPK10pkw74lr2qDpbwFeHAgJFyzAmBCA31pSCS3
abFXDVq3h5m15hI3mYABLtXwvURK5hhvswdwICTGowJgQgrWr/41cm9u63f8W/mQeLaL+
ExKTLX0vbBxZesryuNiICAlZZMCYEIEoCT8CNeyJuV+h0LnQYHoiKcHSRSQQisiSFjZ1f
bxykAgI4iTAmBCDiHzWMfdC2XqK1XPEO6adAHZ4IHRiJck8XYTVjd5kK5gICTT0wJgQgu
Sco5nBYFs3gheoNYZs4IM00QCbpWRlDpFCm06jTgO0CAkadMCYEIC/Z5qEkYIUZwOfWQJ
adgLZ9IaG/ZJIKvPsuUqA4DSH7AgJE8TAmBCDMRT7YgJF8aeHYiBvTwU+9pwWWg9R94ut
XV3PlzWnpUQICN7YwJgQgK5JsQ/pyQfwKf9JEcRjlGj3lgimLRzam43LwDdWYjj4CAk+6
MCYEIBYx/6j+5NoqnT2HXPLMVz0UBN08uKsJNJ0KmRnEYKu8AgJPujAmBCC9ScwKn4Kfi
imLQys1ccbxGF/ivq1EsxkoWfDmzTlwrQICOV4wJgQgyPitIFGHuqCiE/5BmqBrCxFdlg
RG/6SC5YuGrTo54yUCAk+6MCYEIKsjT+PpckG+SNAIP/OTFlmbXP5RlppvHWcTqMd9UsA
PAgI/KTAmBCCNWpEuR/O/dg1MC8XoBrYvi074JMCzDlw7/IsTirw0qQICTucwJgQgHC9l
aJvEL6SdsD/y50JuXTAVNnuDQ8/n6gOem+sZ6nUCAkJ5MCYEIBATmzo5nsK44W1XVu29N
zevnFGIhVWdUjbPLvw5z2l9AgJA0jAmBCA415eRHersYxqQa0SXtcpX+wBONb0duIX9GL
goRfD54AICUI0wJgQgJ9o4m5ihjDz2PeOAVmLW1MnSqUFcQMJ1pUfy0ha3b1oCAkNOMCY
EIICJz1R1pVw0ZN+D87tpVjg3rXMCliQfd3zBgrmi+o4IAgJOETAmBCCcH3nFpvyHDBBr
rem23RAV1/I8aaPEKze7GjjP7Jou6gICQnkwJgQg0CAOOY0UIm6LdiwIJdpJx6Gp5oZT/
CALU3uC9mdY9+ECAkQjMCYEICDQbujoaiLogH5KZZCB0O+mk9vfNy3WUhtGhEcmD+4UAg
I+VjAmBCA1Vq83/WkWjUk/0z4MSS7Ehp1hxGDw4zXe9R+saEWhtQICR3IwJgQgJHOn34h
41MGfHCji+eBnReWI/SwtOXVikuJ/LURjc9MCAkDTMCYEIAGZsMkSrwRb+Az5doOSAITP
AWw71Vs2b4AS4zkQqF6jAgIxFjAmBCBnBCS94dydJj0/+9CQ/90q6nwHDQcY0o2XCDO58
2b/oAICOV0wJgQg9HMP84oTzEJLHxN8NYe9aUcfYObxSS/R/I9ybijMGjwCAjvbMCYEIF
sR5FJEysleE+a8x1JHYraPdTNJZO9WlkcmFSDi2ybeAgI9gzAmBCCvO+Pfmnb6hxV9jQa
rbbZyXcwmStbzUNLt7Z2XgB8ThQICTT4wJgQgHqJSm08hTubeocg5lSGN7s8JyoX/MCwV
KwPaM1S7HLACAkT2MCYEIF8hcOz0I/OGcxn5BHAduTfsostzbGtFiv0GaefP8/dJAgJFy
zAmBCB/8EMy4qiSVib453clxhnB0eFZuXQjyCECtt9+kIhT9AICQaYwJgQgS3lRDpPpty
eaCOTeKbjpr0M0Wg2JYpdPkNZCkz50x8MCAkJ5MCYEIBuLhTXGwbyyAepysM3KsAociO4
qyJGFhVn69COeXCWfAgJHcjAmBCDU+5SF71OwTQXLM9XTwigbYfPgND+pRyqsW8TM5Ezt
EAICP/8wJgQgl7Jo7POs5oYrjesPf+WS0KF4yMIa8JDw3/1hLxK8UxACAk+4MCYEIHulo
rKSwLycSnr8Y8ytEBZ6rlJAiwnQF7zcH8EExg9XAgJA0zAmBCAu24La6Fb0ESnhiSkoBE
c33SSf5SJgUNcBVAYchjDHhwICO9swJgQg/QA7DeqcyHs3YQN3nYPhCWuSVQm0c7zYoqJ
e5VYMYwECAkrCMCYEIK8r5WrsWvSFugmdT6T8Xso9Si23P8sWh+6McBFaEv5MAgJLlzAm
BCDyd/TBCJVSTJIxeBZXnsjczukCkDN4oAevoURN+bVr9QICSe4wJgQgO671WYmnyGDGx
BEXbfi7aaXTXq6HIw/RGaMeBrLO+jwCAjiLMCYEIJiT0vxYQZtZVY0CdOfgGr+ZHwReJL
9FBX3c6UyDUks7AgI3tjAmBCAMfOg0g9vSHei8oi3YREkv9KWDqyJK9q2B5xIkHTS78AI
CTT8wJgQgz3+Xr1jzpHJ6Z8tCSAyifouWa+YUWXCJkZafqER/ISMCAkuXMCYEINvi+QH+
ihPR4qKykN7FjfhfhLEJzDj+CB9wVc4VBBTMAgI72zAmBCCGqyNIqULLk3DHO5YsIIk6Q
PtNmt3WyIF8Pb9PNjO88AICQaYwJgQgo08PeHFONB2JpC8+76tcnkgzUM8phHdJjyDWOS
uBA3gCAkhHMCYEIFC9Q2eKNfjpJxoyiEO914BqjQKee7sqAjkOEp+DZgfOAgJBpzAmBCA
Jcn+vmdgPTAeDvDdLT3s4BwC74pj01Hybmre+6OzW9AICOwYwJgQgX8tZ5evtpTUH4xpl
Hilwmvyu+lxSmuUFvdRM4/oglpACAk4PMCYEIKw/PXt4GnX7tsC0g9FSOfswPKcSLBtUx
Bog10sg5SaFAgJJ7zAmBCAM5qGBqSmY5V5qP89DQm/QN8F+HYHJxIJEWo8yDjlL1QICP/
8wJgQg4Im8hmLgxk5AOT2icXMnds2EKQ2xASb1qDqOspbzacoCAj8qMCYEINSMY+eQXlt
fmOBLg0PxRNAPl87T+sIge2NKTgze0EDuAgJEIjAmBCBlqehaAgJSlpSnZ/5qt730Y1/R
AalrxsgfQ6kTvIZdwQICT7swJgQgXzjEPbigCsa7K0D2Qpgg3ail1ATyskzRQpsLF6fqi

```
3sCAkntMCYEIKhSAlCG3xVLkNszRi6W2G06iHPEz7YJobZc/IePpV7PAgJBpjAmBCAJ3k
RgiX8r7KR4/NJ1FOmi/1TsnaOjPq59d06pBTfMRwICQnowJgQgscxbufrbZeOOPwxXGE4
XnyaggXcz3Iw5lR5g+/0KnxUCAjsFMCYEIG1QyGSS9YE+rgnFDXpG4gVqSphxIey9EsXo
mI6s/CJBAgJHcTAmBCBXoKnU7MAQCUtqxm73Uu660ApL5ZosjEQtF0yEORjJagICT7owJ
gQg8vAOROTRVH7vK72BdfYTHy2QuXEkOB7X1qmoN8/SKT0CAkadMCYEII8taUtzUfvfca
DrIjrrzNzGbo60TqoaN3Q2uLgKhU9uAgJA0jAmBCA8Jsi9rBUN1R3cJKEATddc1d2wHuv
aNrtR32zoye7ZMAICQaYwJgQgfEHGwDMQtsrp+sI9rnvOnAmY+xz1XzyRh8t3Q7otUIcC
AkNOMCYEIGdd1tErxsuk6+YQFyc08X80DVrlUsCgGrnf/N7GMnajAgI//TAmBCArr/RhU
xkeBxnfEo95YODmLW/B1LonMxwGCox2COYRzAICSEIwJgQgEweJtvNQL9bjf2iQ9AW+yp
JHzOHqhKf7acEDKCernEYCAkGnMCYEIFt7DobpP6uC036RRjyIAgQQ/42jtrOR++MD4QY
fbpd7AgJLljAmBCDr33bhYRLCqHfzhqVA9h0OEUHFpMgGdfuD6YxUdv6AxgICQ08wJgQg
6kyL6peOm+TBOBPVME8CAqB+ANHfBiQXmFG8Rv/TiYECAkhFMCYEIPH1EhAhYbq5kHXdp
rO2X9WrNqmml2Rs3wI5Q8tfecZ9AgJVhjAmBCC26nQBMIR6buqJ1zPG12wKCsV01mglKE
EXyMLiXiSvaAICPyswJgQg7xLqvkJLJgC4mqBnbzktogFz0Cgf6Mj065Z+melcod4CAku
VMCYEIAzWDR+QpUqlkBwLYXdI1j+1S+atzi3FUjI6UpY0zrDkAgJEIjAmBCBSDmLoPTTp
o2lThHrqC+EKh+d3oZYyINVkIF+UB2nS4wICPlcwJgQgFcOQIlwmcVYCKL8pT8vW2u7EC
ays8aVCMsNOQ4nMiqYCAlCMMCYEIBIYL0TPWI30bVJi6DloMw6BjsEzHdHgpY578GqFqJ
ENAgJPuTAmBCAtdStaax5m+bf0CkFCW4ahYk9h/bXYThkbAqjd/vrqlgICQnUwJgQgvQS
sn+BQYX8KlSqdWXRWw8JcJWMoWCUBb72YWX+8GlsCAkXJMCYEIE55GSoKFsJUZO5w6slR
nNn6Sg/7qGmF72JrQP54PR/FAgJA0jAmBCAvRh8/FWBFDyzhzx9DlT3EURcCuFG2bkmmS
uEKVca4ngICTT8wJgQg/3HwZWX3kCvSIgP353RrzmQHmNpLuDYnhJFgOe4VFnUCAkQhMC
YEICFmgQIz17HiFDtOzyzTGBbjW+J8zWIIEnhJDglbK/QBAgJA0zAmBCAG1OHcH22DLxF
OKmutn3iwIaB7y4p71vHkJTkfJvlCawICSe8wJgQgVXaF+fl946I23YBM2yugiNBAOOgU
KIMSNqjexcts4W0CAkNNMCYEIEn17fqWUHl/Ut0396kcdMPes4P0Ilumc9GACGdzPxSdA
gJBpjAmBCCKnq9O/mbcRISAeDyhs0Fj8uUuYtyZVFiWeuNEgQduKgICSe4wJgQgFSyVT0
sZ+KAZuPm5O/NLFAVmwJfj0Jh1fDK/sIjt9PkCAkXKMCYEIAjRw4V9a1Oy808Uj/6hFDv
HrZGtZzwcjlAgWmBa4lglAgJSNzAmBCDbhYX9msQB07vzhPv3+ZimS53DuyP1PkU9++WU
fREZZQICRPYwJgQg9eFpbDRspxIpYM5opguFlrRZMoLNlMaDgJZuPtmvxJ0CAjyvMCYEI
BDUK87HsfP++Nj7DP5JT1Ku+4pKLsObbZMnUtIEtz95AgJBpzAmBCBAkEDWSeCipOcgg9
gushL67+tk6gHb0fxuDbXrVq+r3wICSsIwJgQghxxpeO0zW++6xbLY+i2QNRl6Kho0o6g
3k1vRgyKMCgsCAj2CMCYEIF+D1twA/l9ygIBXNq5TlGzw6DjpFmVGmjw6CE0sG3GMAgJF
yjAmBCBRZOPIg7jm8bxv9tZLQJldOPiLfnvxLPf8YY9yXDzpmgICQnswJgQglQD4UBUMC
T5hNDMVNooytw52WqZGHvksHPJBU5QjGPUCAkNOMCYEIMspqi8R4vx0YcZrVCmoiKBaQ3
dgISYdD2o0yWcG3ULFAgJKwjAmBCBT6JKpPDGqs6qGA4Mkom6+zZp/dOCEMOPamYSLIsI
5vAICTGswJgQgvzO4UiG1l+t3BxPhCyDBdUj/dGegABaXn9U528y3FBoCAjiIMCYEIEdi
vwkozmgnX0g+45qzbq8I4GuTRGX3qaHk77f/D82gAgJKwTAmBCCWWHiS9m1FuFJ5saDtj
2/VUCmNm+QP9piNW6NTTzFC9AICR3IwJgQgzNVM/wCeeYiJoOKGJBMHIKftFtjL8mw85r
kXhTpARZQCAkT2MCYEIIf+dxLYN1WtEUUH6pknbxrdqSDYAjmV6rlaqGulZJhWAgJJGjA
mBCC6JIVZAn4XmVimESDDCezTQiCnTe/ToMjCqwsnyf2lhwICQnowJgQgj91577YttW8J
GjfyAWGQNmM61KlR57o7zKu7QQYlwX4CAlFgMCYEIAhVWwYn5y3bziyHlaPrX5QTq6AWp
/6y9zn8IdSNf+XUAgJEIzAmBCAhvg0yHLZdJ0Oi4gAoz6H1+IIGiXplUFbbZ5MeU233Sw
ICR3MwJgQgsaAWfywlBudRzWhmRwRl/+O7SPDX0FrldY6JPm71zRkCAkJ7MCYEIOmgadk
Hv93OoPy0zoX7wG+ZoX76OjLQtVUz2k7UIQHIAgI+VDAmBCD8L1cC9aPe9b5Fo1efCR6a
+/94EioZJG0NF/tk/3gHAgICQNIwJgQgzbSJuYykxcB4D/KCrmbGDVC7+Tmon++UW/FsY
SZECFwCAkNNMCYEIENgVDzKBFO96u84ktfYN7nVDGTdaRIzkxc/Z+syC2TPAgJEIjAmBC
CqqalHwNPiA2dukhMmKylGLq4Emj1CzBY+rPMYzAGNFQICSsMwJgQgPxuwJb1hJeTUKYF
Wlq9lmaKXkXArUIHXBSBCQgpfTHkCAkJ6MCYEIGZTSrHvZ2TJ03ZwVdu6meXVk6hBoHei
RLgYj4RfAo7iAgJVhDAmBCAT4X4YtJQLqZqvilWETxZ3WRkB/MXzPDzErEcQZVzpagICO
```

9owJgQgnlYBrq/ADKOxSLwWZ6jsZ+nbv2ytWqnEUYI/CArG79gCAk4SMCYEICxORErcH/
5qrlBu2QusFm74gtkBmiVee31VXbcFgir5AgI2DzAmBCAB2Ar/cFcjmNmZNcpszjRYO1z
xHjL3rKKD6QbHkXvuAQICRPUwJgQg+URpogPr6tZneDVkkkRsrLoJz+s5vlZUrjwn6ewS
Ci4CAknvMCYEIGD8A/+EQeRfLSj0SBVjapGsr4no0q6VP04qAo61FGeEAgJSNzAmBCBjn
fmnQ8Blq8pEldKJH/5N032PRKvu5+4plXaixvZp0QICQ00wJgQgIYutojOYLEnkTLsoI0
OsT1Ar4f6qdY5vp2p9MhDxmXACAkQhMCYEIJxd3o0dh+NGQkzBs3MaJdt41OKeTkgv+sB
KvfSmGwZTAgJA0jAmBCBhLW9KsT3LhN7Ludd/vzGkQaZiAkuVhWKCRUdAyYLh3QICP/0w
JgQgzXb8SwnjAJnNj5ZFe10JVZe1s2bye+zmxCfzVhOm5HYCAkXLMCYEIFImRd4H/v2g0
tNA5hdzQJQnKwN6U0/e4iuyJ4HjEA/0AgJRIDAmBCBn5tUk0D0D3lUgZMIc/qupQ/y9U6
/hDkr1Td/vSCuaKQICQacwJgQgGHOrmZ+52gwLIkj3DIHkz9zwCzBrHGVJ3de52oeeDiE
CAjozMCYEIP3rfr6Bxvdy/Sikb8tVO8ASb1wPr9q1a3JB2Irbm8IXAgI6MjAmBCBwoQKg
FCYLRx7NBruVMNqC4LrGm+r63aFYZVVVNiTkoAICOtowJgQgy/J2VVix3tYZ+nf/Q5eVc
BLod++R2HHKUqaccejU6RwCAkJ5MCYEIO2fBhl8+PH3wtE3wnXe5c78MeTEZyl/kyyiiW
J/EOQCAgJDTjAmBCBVBCrBYolj8pBdjPN/+2MCjWAvWXR0IZreSSz1Flv2hgICQ08wJgQ
gcPcgpSFhOXx2bp6EZIh1O/TQawqa92f6cfHB+frK4dACAj//MCYEIJOXUMlFlj1pyJna
g/1/RS0Tsc1a5zCB7G2aAKvcwGcDAgI7BzAmBCDRyTSvv0GkOx4DalLdBFw+29Y7HbUDT
3DhYUa8iudmHAICPyowJgQgv8z+XZ6YU/qi0+CrEn2zPPgH0vJhV5obNo05/L6MGsMCAj
vbMCYEIDizLsIr8vm7kQ4TpfMScQTyk3ueWKEevGNQqSZ2nJNbAgI7BjAmBCDzNuvUQSg
QI5283uMmyemP9B+BD+zXJmCdq4NC5/e0DgICQ04wJgQgoH+CnPLGmqvTv2FP+WtrZJtT
yHEmI5Qh8iETVIMwDVoCAkJ7MCYEII4Zkl3ibkXtoi9/tCcRwHIzMe4dk93NsX77x6+3K
lqLAgI9gTAmBCDm+6tqUlS/87xVMVBRxUxDwlrLmF3z/vpfGWa3/b3VmAICQskwJgQg9E
kg0mb6OcwZAuPIvglyhcee9OMkh7wPUl0c92doyJ8CAjSmMCYEIIs0cuOLf9fZ4BAqVjI
QWllmUUGh8t1nsIc9WZiNBbj+AgJHxzAmBCCCqfNU2Q3p2OzvbV49ppqLGySFilfujV0o
R0Ml0/tCtgICQsowJgQgj8fmavD/fbTWl/IPjS3DUeNuGq1fS+oBDaGljgXcdGACAkkaM
CYEIIixhEaYKjuSUm3DEmeLoJY5ew6+OXlzMgmajVOgK+LAAgI5XTAmBCBbu1540vVhbe
m1AFpnhqpZQCFeN4J37O3Ttj+t+C8YXQICPlYwJgQgYX4PVaUu5ZlKcoLWh/wKkXcdAeh
iAl/aE8CzteMupVkCAkT0

## B.2. Example ErikPartition

This object was retrieved from https://miso.sobornost.net/.well-known/ni/sha-256/AZmwyRKvBFv4DPl2g5IAhM8BbDvVWzZvgBLjORCoXqM.

MIIxEgYLKoZIhvcNAQkQATiggjEBMIIw/RgPMjAyNjAxMDgyMzAyMDhaMAsGCWCGSAFlA
wQCATCCMNswgdEEIAFg/0CdwFaUyfP3EyK5RmO+SHjEkYpJ03VcFje02/uaAgIIpQQUfz
4LJ7jk15j5K53hV/HaWkPNSeUCAhH4GA8yMDI2MDEwODE5MDA1NVowfjB8BggrBgEFBQc
wC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC81Zi9hMGM5
YWMtM2E0Ny00ZDZjLWFhMTUtYTQyZWM4Nzc2ZmJiLzEvZno0TEo3amsxNWo1SzUzaFZfS
GFXa1BOU2VVLm1mdDCB0QQgBNUfbUn0GcyCLtIB6SMWn3PzeAvWEckmuoPOieKTqQgCAg
ilBBR/ytid8b+Zo28pDMPvDx57TQJ1MwICF5cYDzIwMjYwMTA4MjAwMTExWjB+MHwGCCs
GAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzU1
LzlmOGIxNi0yODRmLTQ1MTItYjNkYy0wMTVkOWYxYjRiNTAvMS9mOHJZbmZHX21hTnZLU
XpEN3c4ZWUwMENkVE0ubWZ0MIHRBCAHXXSuVDo4xIJcp05FZytpg4kjD/qhya82XREhZN
cQDgICB84EFH9RIoN0dC31RKqTBYxaO9PRZCGZAgIPKxgPMjAyNjAxMDgxODAxMzBaMH4
wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFV
TFQvMGEvOTM4NWFhLTFiNzktNGEwMi1hMDkyLTAxZWIwMzY4NGYwOS8xL2YxRWlnM1IwT
GZWRXFwTUZqRm83MDlGa0laay5tZnQwgdEEIAfribv7yIvTN0Wj8JLMcqQVV5J/DDGHUE
cOrsmff15rAgIHzgQUf5LfdhDwSPQ79EwzbUHGwRV+8OICAhdPGA8yMDI2MDEwODIzMDE

```
wMFowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkv
REVGQVVMVC8xNy81OGJlMTgtMmYzNS00YzZhLWFhNmEtN2Y3MmY0NGI4OTgyLzEvZjVMZ
mRoRHdTUFE3OUV3emJVSEd3UlYtOE9JLm1mdDCB0QQgDIjUjD6duUu0FKQv9y05IBRfVc
l4+Kd1b9mDDIns65ICAghgBBR/UAd9LdimehrotqvWu7NIkCiluwICF8IYDzIwMjYwMTA
4MjIwMTMwWjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3Np
dG9yeS9ERUZBVUxULzVjL2MxYzFjZS1lYTU5LTRkY2YtYmNjYy0zZTdjYWRkODhjNzAvM
S9mMUFIZlMzWXBub2E2TGFyMXJ1elNKQW9wYnMubWZ0MIHRBCAaaM1Znthus6ISP96g3J
D81Tog+3zOPZGZ8AuZHf8H9QICB84EFH/j1jtKW0BLX/g8vysVJaMEd/ZcAgIE/BgPMjA
yNjAxMDgyMDAxMDlaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9y
ZXBvc2l0b3J5L0RFRkFVTFQvZjYvODFhNjcyLWZlOTgtNGQzZi05YjgzLTJjYjdhM2I2Z
TQyZi8xL2YtUFdPMHBiUUV0Zi1EeV9LeFVsb3dSMzlsdy5tZnQwgdEEIB+2rs4fnNPglk
dC0Oxj3YrmH7gcOCVQjNglyUOdsjI5AgIHzgQUfzP8QNLgMzu8e96rK9hZlUMBwPECAgc
FGA8yMDI2MDEwODIzMDExMVowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUu
bmV0L3JlcG9zaXRvcnkvREVGQVVMVC9iNC82NWJmMWQtZWMxZC00MGU2LTg0OWQtNzk3O
TBkNjZkN2QzLzEvZnpQOFFOTGdNenU4ZTk2cks5aFpsVU1Cd1BFLm1mdDCB0QQgI6Umay
dl8dJpPFLj+98hCRWwWetbdCBsDVGbXzToe9sCAgfOBBR/KjK6QhloDc3Vj2EB5ceuwVQ
KcwICF7sYDzIwMjYwMTA4MTYwMjA4WjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2ku
cmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzU1LzE0M2U5ZS05NmU2LTQ3NGUtYWQyY
y0yMmU2ZGY0NTg0YWYvMS9meW95dWtJWmFBM04xWTloQWVYSHJzRlVDbk0ubWZ0MIHQBC
AwH+fHKOlH39wY9R5xHxHmhMnb6GSHhpnVmpCHVeA8FwICB4MEFH8QnrYwNX69ximBGmL
mKxhXpxMqAgFQGA8yMDI2MDEwODIzMDE0NFowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9y
cGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC8xOC85NTAzMGUtYzVjYi00NWJmL
WEzZDktZjhiNjNkZWRlNTZjLzEvZnhDZXRqQTFmcjNHS1lFYVl1WXJHRmVuRXlvLm1mdD
CB0QQgMrEX8b0WrGlFZa/aGJQ1O81LoLp9UO84CBaveQjgEJ0CAgfOBBR/VoneSnznaL8
6tdn4VG6FbMsZNgICF7sYDzIwMjYwMTA4MTYwMTMwWjB+MHwGCCsGAQUFBzALhnByc3lu
YzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzc3LzY0ZDUwNS0yMjA0L
TRkY2EtYTk1ZS04YjUzNzI1ODRhY2QvMS9mMWFKM2twODUyaV9PclhaLUZSdWhXekxHVF
kubWZ0MIHRBCAzPSRrimqT6CeNH3o60zUEaMS3WUawEw4OZIl6p9RoIQICB84EFH8Yitq
1tVIIHsrIIcmwkDlIc7MVAgISKxgPMjAyNjAxMDgyMTAxMTdaMH4wfAYIKwYBBQUHMAuG
cHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvOTAvYTY1NTIyL
WY3YzUtNDg3Yi1iN2I4LTJlNDYxNDE0MWFhNC8xL2Z4aUsyclcxVWdnZXlzZ2h5YkNRT1
VoenN4VS5tZnQwgdEEIDZuHlY6ZmzvFkBtLu2X4/vNQhnzQRDO3eUq5luTSA9lAgIHzgQ
Uf/F65Ue58mQeYFd/5VPdtvdJoIcCAgT7GA8yMDI2MDEwODE2MDE0MVowfjB8BggrBgEF
BQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC9hNS8xY
mMwZjktYmQ3Yy00ZjdhLWE0MDQtYTQzZmYyNWQ0NDdkLzEvZl9GNjVVZTU4bVFlWUZkXz
VWUGR0dmRKb0ljLm1mdDCB0QQgO/APpgOd1ACqy0+8SO9WRmhqT9DddrWoyeDqQzL7bd8
CAghfBBR/VvKJSMgy8tQ0u0TV3g6hImAbBQICF7sYDzIwMjYwMTA4MTUwMTEwWjB+MHwG
CCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxUL
zNhLzIxZmJjYS04MGUwLTRiOGMtODYyMi00ZTg2YWQ2NGY3NzQvMS9mMWJ5aVVqSU12TF
VOTHRFMWQ0T29TSmdHd1UubWZ0MIHRBCBBea4tF3JE2mv6g3bWSBcOn1ws0Gw1yJcVg8W
bXTnL3wICB4QEFH83coPXwjWpPce9K4MXsnSPKF/3AgIArBgPMjAyNjAxMDgyMDAxMTJa
MH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFR
kFVTFQvOTIvMGZhZGZjLWY0OWItNDNlOC1iNjIxLTgzMWUyOTQ0ZjhmYS8xL2Z6ZHlnOW
ZDTmFrOXg3MHJneGV5ZEk4b1hfYy5tZnQwgdEEIELKBg9HoXTnEUmtGFfoSdAVlwNlZSY
G4PMsqnmb19iSAgIHzgQUf/yCWF32m3yUsWphky/8i24zUVYCAhOTGA8yMDI2MDEwODIz
MDEwN1owfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvc
nkvREVGQVVMVC9lNC81MzFkMmItZTE3MC00OWE1LTg0MDQtOTJmNzNmNTZmYzYyLzEvZl
95Q1dGMzJtM3lVc1dwaGt5XzhpMjR6VVZZLm1mdDCB0QQgQxrKR0GQ5PFg8dSnJkqQgNX
```

gqJJzOfyMohP7WhCW3Q8CAgqPBBR/a9GmsEYlxXHYMPh4scAjgkdAjAICBl8YDzIwMjYw
MTA4MTcwMTAwWjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb
3NpdG9yeS9ERUZBVUxULzQ4L2JjNjZkNy01N2FiLTQ3NWQtOTZiYS04OWI2YzMyMzE1Yz
IvMS9mMnZScHJCR0pjVngyREQ0ZUxIQUk0SkhRSXcubWZ0MIHRBCBGDFHS/6xyUhCj8Zl
1U1kS9wP0gzCFREdGetcG3cLKMgICB84EFH8xKwnR9pDyVwC9Xc8HyRgMXpZjAgIA5BgP
MjAyNjAxMDgyMTAxMzJaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ld
C9yZXBvc2l0b3J5L0RFRkFVTFQvYmMvMjlkYWM5LTlhMTUtNDY2MS1hZjc4LTE1MjJlMj
k2NGZjZS8xL2Z6RXJDZEgya1BKWEFMMWR6d2ZKR0F4ZWxtTS5tZnQwgdEEIEoZODFpg9Q
Alydw6JYDtmG9KPMwOH50lZUGXPzouWz/AgIHzgQUf1i8CGQS9NLFT6BwLZLOJUls5HkC
AhU0GA8yMDI2MDEwODIzMDEwNlowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpc
GUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC9jYy81MzA5MzMtNTkxNS00ZmYyLWIxZjktNT
AxMGQwNWY5OWE4LzEvZjFpOENHUVM5TkxGVDZCd0xaTE9KVWxzNUhrLm1mdDCB0QQgSpT
Mk6O4NitlrWxkqmsUQoq7nb6KWB07+LKbVl4Ywz8CAgfOBBR/HVjWLd1+R68hlv11S7P/
JnmJKgICF7gYDzIwMjYwMTA4MTcwMTA1WjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa
2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzQ2LzI5MGE2MC03OTJmLTQ0NzUtYT
lmNC1lM2I5ZTBiYWU2YWIvMS9meDFZMWkzZGZrZXZJWmI5ZFV1el95WjVpU28ubWZ0MIH
RBCBUQl7ggC2lGraIQxbw+rir1iII4x8LWg5ys/sm6mvySQICB84EFH/WLkLAjhYBxFce
DYijSaBQnepeAgITQxgPMjAyNjAxMDgxNzAxMTNaMH4wfAYIKwYBBQUHMAuGcHJzeW5jO
i8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvNzgvYjkzNGVlLWM0YmEtND
M5MS04MGIwLWU3NWE1ZWE4OWZkMS8xL2Y5WXVRc0NPRmdIRVZ4NE5pS05Kb0ZDZDZsNC5
tZnQwgdEEIFoFa8a2/eSlNTOkJE9qPttQcC6+bFpdBs9LUTO56tvYAgIHzgQUf+aLEiNL
1wNDAbyWsTiq4neGCj4CAgYnGA8yMDI2MDEwODIwMDEwN1owfjB8BggrBgEFBQcwC4Zwc
nN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC84Yi80YTExMTEtOG
YzYy00ZWRmLWI3N2MtMmZhMjYzMWJjMzFjLzEvZi1hTEVpTkwxd05EQWJ5V3NUaXE0bmV
HQ2o0Lm1mdDCB0QQgXnHaL4UZ+iFqZDwh8JKjKcmmeBOmCNz543yoQiwlMn8CAgfOBBR/
qBajM40ydBzU//j0ndssyDj9xQICBSsYDzIwMjYwMTA4MTkwMjAxWjB+MHwGCCsGAQUFB
zALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzUzL2E0NT
I4YS05MGU2LTRkOTEtYTUyYy1mMDcxN2VhNDg1YzYvMS9mNmdXb3pPTk1uUWMxUF80OUo
zYkxNZzRfY1UubWZ0MIHRBCBlBtj8cbMdXpx/CKEWYwvo+YZAZmIOEaiK+HcvU9/M3AIC
CBgEFH8WgCjsDatmimfVv29TWMqr4zeoAgIMKhgPMjAyNjAxMDgyMTAxMjhaMH4wfAYIK
wYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvY2
MvYjE1Mjg2LWZkNGQtNDlmZS1hNjllLTdmYWRmNTBhMmUzNy8xL2Z4YUFLT3dOcTJhS1o
5V19iMU5ZeXF2ak42Zy5tZnQwgdEEIGd7MqCU/xUEGZqKrH1XkqfN3RqCtG/L1V9L8DQY
A1S1AgIHzgQUf7UOfTt/0olM+3DklGCLMgzCFcECAgDQGA8yMDI2MDEwODE3MDEzNFowf
jB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQV
VMVC85Ni9hOTE4Y2ItMjA0NC00ZjFjLTk3MDAtOTM2MDFhMzRmNjJmLzEvZjdVT2ZUdF8
wb2xNLTNEa2xHQ0xNZ3pDRmNFLm1mdDCB0QQgbJNFg5mgBHtCC8IDP0ZxfJZsrSOBp8KO
JRHb520LtpECAgfOBBR/dzTf6hIGV0EuqGfdvHuE0TK/eAICEGkYDzIwMjYwMTA4MTcwM
TIxWjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS
9ERUZBVUxULzdlLzY4MDMyNC1lZTFmLTQwZjItODhkZi0xOTY5MzE5NjJkM2MvMS9mM2M
wMy1vU0JsZEJMcWhuM2J4N2hORXl2M2cubWZ0MIHRBCBt7xqIdeziNrWHHxep3FfwH0OP
q5nSVE8fGBb8WLTWAwICB4QEFH8UzoEDt4XzUEvEsy8fTJtwzj9/AgIQdxgPMjAyNjAxM
DgxOTAxMDFaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2
l0b3J5L0RFRkFVTFQvYjYvZjY5ZGJlLWUwNzUtNDQzOC1iNTNiLWY2MTYwZmExZmIwMi8
xL2Z4VE9nUU8zaGZOUVM4U3pMeDlNbTNET1AzOC5tZnQwgdEEIG6nUCjrFXJF2aB9FoMk
5WimDuz4GdU4rL+jK6x7dQe8AgIHzgQUf/sBFcSs3dG0rcQHN4Byas/AGvkCAgzEGA8yM
DI2MDEwODE5MDExMVowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3
JlcG9zaXRvcnkvREVGQVVMVC82MS81MjY4ZmMtY2ZlZS00YWE4LTk3YmYtY2JlMDIwNjY

1ZWZlLzEvZl9zQkZjU3MzZEcwcmNRSE40Qnlhc19BR3ZrLm1mdDCB0QQgb+o8m06Q9iU6
EnPhdWx7hQf34aX7u8+BtkN6WDmO8JcCAgfOBBR/tD3iN/0LaihziSMJIdJaLC7RqAICF
00YDzIwMjYwMTA4MjAwMTI1WjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS
5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzA5L2Q0ZDEyYS1hYjRlLTRkYmEtOTVkZS1iYzY
zNzEzMGRlNmUvMS9mN1E5NGpmOUMyb29jNGtqQ1NIU1dpd3UwYWcubWZ0MIHRBCBzrSMA
19fNkYVSPFrfNwuIQW6pxyA/3xBmt+lL5h5xdAICB84EFH+F6ZA1Q5fjbAypA6DGIMdwn
v3NAgID6hgPMjAyNjAxMDgyMzAyMDhaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS
5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvNWUvNGFlMTc1LTU1ZDAtNDg0ZC04ZDE
xLThjOWQ1ODIzYmFkOS8xL2Y0WHBrRFZEbC1Oc0RLa0RvTVlneDNDZV9jMC5tZnQwgdEE
IHWhirnnIDJ5n+WI2Xvk54oI9JbIb2bvBy3fY7npA/trAgIHzgQUfzE2D/wa/V8dpm2BQ
E5GY1EtSWcCAhVEGA8yMDI2MDEwODE4MDE1MVowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly
9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC8yZC9mZWY1ZGQtMzhlZS00YmM
1LTgyZmYtNTg0ZDc4YTI1ZjhjLzEvZnpFMkRfd2FfVjhkcG0yQlFFNUdZMUV0U1djLm1m
dDCB0QQge+535dV/T4fBendUeAcDJiibStHT0H8cdANzm3c00sQCAgeEBBR/iyGDWJg6f
XolOKnam6HzSU1xqwICAeAYDzIwMjYwMTA4MTYwMTA3WjB+MHwGCCsGAQUFBzALhnByc3
luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxUL2Q5Lzc2Y2QwMy1lYWE
yLTQ5NjUtOTRlZi05OGRiYjY0MDJjZGQvMS9mNHNoZzFpWU9uMTZKVGlwMnB1aDgwbE5j
YXMubWZ0MIHRBCCASGa4GEb+lTv/4fJr94UgLuaT+ISjkIKKdEKkHovxHwICB84EFH/uP
zggc/LD5PzP/KOExbDNgmwmAgIDjBgPMjAyNjAxMDgxNzAxMTdaMH4wfAYIKwYBBQUHMA
uGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvYTcvOGNjZWJ
mLWY5ZmEtNGM0Ni1hZWU4LWNjYmZhNzM0MjRhNy8xL2YtNF9PQ0J6OHNQa19NXzhvNFRG
c00yQ2JDWS5tZnQwgdEEIIEq59M4+de5vhL+AIB9ZhUCn4d3FMAGscEdEk2Nqb90AgIHz
gQUfxFBauOSnfN79mpFnNOspjiR6UYCAhe3GA8yMDI2MDEwODIyMDEyOFowfjB8BggrBg
EFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC8yZi8
2YjIyZTgtM2RjZC00NGRkLTk1MTQtNjc1NmY3YjBjMGRiLzEvZnhGQmF1T1NuZk43OW1w
Rm5OT3NwamlSNlVZLm1mdDCB0QQgg/TeGB1EpvVQgaiS8VooOlY3CODqmdkfZl+6QeaG9
IcCAgeEBBR/F7b5MjmdWFCT0YczlOv+Gyn5jAICDzsYDzIwMjYwMTA4MTYwMTUzWjB+MH
wGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUx
ULzMxL2NjMDIwMC1iYjJjLTQ1ZjYtYWM3NS04Mjc1NzdkZjhlZGMvMS9meGUyLVRJNW5W
aFFrOUdITTVUcl9oc3AtWXcubWZ0MIHRBCCGMyVTGVLFMFffgmebFSD+ivU3zQMIPxNpC
ZMyZaCMUAICB84EFH8+CHSJ7iOpQk1T+sIW7kuOASgOAgIXThgPMjAyNjAxMDgyMDAxMz
RaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0R
FRkFVTFQvYzYvYzJjOTlkLWNmOTEtNGRmYi05NGRlLWE3YzYzZDU2MmU1Ni8xL2Z6NElk
SW51STZsQ1RWUDZ3aGJ1UzQ0QktBNC5tZnQwgdEEII5DEJkg54e32y2/B1WOyDyahzCEf
aERNpKcSA9PZdtHAgIHzgQUf+/r0V+8ZRSHrK9nQk9Hg9Q+tCACAhe3GA8yMDI2MDEwOD
IxMDIwNFowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXR
vcnkvREVGQVVMVC85YS8yNmVmNjMtYjcyZi00YjNjLThkZmMtMmJjODQ1MTkyMjdhLzEv
Zi1fcjBWLThaUlNIcks5blFrOUhnOVEtdENBLm1mdDCB0QQgj2n88kmDKSPaKWTt2cg1P
qjq2c7bnts/AV3HJWXqqjkCAgeEBBR/mRjKtvHzXvG15YPLKDnpt0ixWAICFI8YDzIwMj
YwMTA4MTkwMTU0WjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmV
wb3NpdG9yeS9ERUZBVUxULzI2L2RmODJmYS0xMThkLTQyMTktYmExNi0xYTk2ODNjOWQ2
Y2IvMS9mNWtZeXJieDgxN3h0ZVdEeXlnNTZiZElzVmcubWZ0MIHRBCCSkQuN9PCn151nB
GBrBmHej5rw8dfvnsEEW+nAvyfMjgICB4QEFH9R3x306IZ+QeXn+S3n+dH6DBVNAgIMoB
gPMjAyNjAxMDgyMTAxMTlaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5
ldC9yZXBvc2l0b3J5L0RFRkFVTFQvZmEvNWU3MzFmLTUyOGItNGUwMi1hZDY4LTZkOTAz
NWFmMTUzNS8xL2YxSGZIZlRvaG41QjVlZjVMZWY1MGZvTUZVMC5tZnQwgdEEIJdpko2c2
K9OwfCs7d25CWOpQ5QfTp8v4Ini0HEfstxvAgIHhAQUfwdXwzGYHgQrdHE3NSfQ9koTVr
QCAhNeGA8yMDI2MDEwODE5MDEzM1owfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJ

pcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC83OC80Zjk2ZjktNTY5Zi00MzNjLWI5YTgt
NmEyNjEyZDQwZjUwLzEvZndkWHd6R1lIZ1FyZEhFM05TZlE5a29UVnJRLm1mdDCB0QQgn
bFHY5o1eFI/JXrcZzLyJ1TdH3Z3csXzRZinE0Y5cCsCAgfOBBR/TVkcI60saUd9f3odTO
jrzulZbAICBhQYDzIwMjYwMTA4MTUwMTM1WjB+MHwGCCsGAQUFBzALhnByc3luYzovL3J
wa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxULzNlLzBkNjdhMi1iYzM2LTRiOTMt
YjYwNC1kNGY1YzkyZDkwOTUvMS9mMDFaSENPdExHbEhmWDk2SFV6bzY4N3BXV3cubWZ0M
IHRBCCnjXO+nhFRAnet66plShQY9ijeKIn2Jjn8FzzN7RbFMwICCBgEFH8DofjDNP2/S3
je8MWS/wSQ3fSwAgIXcxgPMjAyNjAxMDgyMTAyMDBaMH4wfAYIKwYBBQUHMAuGcHJzeW5
jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvNjYvM2ZlMWEwLWM2ZmQt
NGJjNC1hYWUxLTllZTAwNjk0MmI0Yi8xL2Z3T2gtTU0wX2I5TGVON3d4WkxfQkpEZDlMQ
S5tZnQwgdEEILygQ8fZPYY0mQD8zM4wMnIkkXbFYZks+kKXtCavjvupAgIHzgQUf0OdlC
Qm/Gc7J5zJirNf29fql/UCAhNTGA8yMDI2MDEwODE5MDE1N1owfjB8BggrBgEFBQcwC4Z
wcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC8yYS82OWEwOWYt
NTgwOS00NjU2LWIzNTUtMTQ2ZjI1NGFjMTMxLzEvZjBPZGxDUW1fR2M3SjV6SmlyTmYyO
WZxbF9VLm1mdDCB0QQgvgX58HkrAY/LuflVMbJZ7LzrWFSoZlhjbgUohc4JjkACAgfOBB
R/QtJ8IdqGx+n79Erg5WyY89L4CwICD1gYDzIwMjYwMTA4MTYwMjAzWjB+MHwGCCsGAQU
FBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxUL2E5Lzk4
YjAyNC05ZDhmLTQ4NjktYTJhOS0wYWZiN2MzZmJmMzAvMS9mMExTZkNIYWhzZnAtX1JLN
E9Wc21QUFMtQXMubWZ0MIHRBCC+S/81Xud1PzUBFp/H1MiaahJK+jQve1DYEWEVia39uA
ICB84EFH/JVqUrc1BKTx/zRUcZkpen9d6dAgILbhgPMjAyNjAxMDgxNzAxMDlaMH4wfAY
IKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQv
YWEvNzFhYjY5LTY5NjUtNGI3MC05NjhkLWJiNTdhNGVmNzE1My8xL2Y4bFdwU3R6VUVwU
EhfTkZSeG1TbDZmMTNwMC5tZnQwgdEEIL9qHDGaKh1R04cdIJLYSjtKeclEypWTBTqgxV
51LzV0AgIIpwQUf1FerQle7ZrEyrxatK0LWGfZ8BsCAhfRGA8yMDI2MDEwODIzMDE0OVo
wfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVG
QVVMVC9hNy81YzJhNTktNjAyNS00MDBlLWFiMjgtZjBhNjI0ZDQwOTEyLzEvZjFGZXJRb
GU3WnJFeXJ4YXRLMExXR2ZaOEJzLm1mdDCB0QQgv39HKTjopic782oac6tVyAIMW0PaCG
eYBa2nvVCuDKYCAggYBBR/4OdZNU6DzBkyA4EQneItoPGnAAICF8EYDzIwMjYwMTA4MTg
wMTUyWjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9y
eS9ERUZBVUxULzRjLzBjNzI0My0xZmZiLTQ5ZDItYjVjMS1iNDAyZThhMWQ5MzQvMS9mL
URuV1RWT2c4d1pNZ09CRUozaUxhRHhwd0EubWZ0MIHRBCC/mDUwr8XbN8QR3vi4bpdyNf
prvcn9Z/t23XZndp12IQICB4QEFH9za4igJATUMEvdrV/1UEpoM+e3AgIXtxgPMjAyNjA
xMDgxOTAxMjVaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBv
c2l0b3J5L0RFRkFVTFQvOTkvMGYyNWNlLWI5NzctNDg1My05ZWMzLTllNTZjYjU0ZmVmN
y8xL2YzTnJpS0FrQk5Rd1M5MnRYX1ZRU21nejU3Yy5tZnQwgdEEIMzFRfxAeVWe8poS1U
kK08BGZ3h9a4zX7Upp1hcBuxHYAgIIXwQUfyuobfeHiI9vhZKoBqb/6jBGwHoCAhe/GA8
yMDI2MDEwODIwMDExOFowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0
L3JlcG9zaXRvcnkvREVGQVVMVC82MC81ZTY3ODYtNjM3Ny00MjI0LWJhMDYtZGM0NzY5Z
WZmMWY1LzEvZnl1b2JmZUhpSTl2aFpLb0JxYl82akJHd0hvLm1mdDCB0QQgzZTr9egKcv
XVZQaiAvwJv7MZS6kvoBECP7s1InpsFaICAgfOBBR/0YpqSZEMwzHckRFK5ZtxhdXzDQI
CF7wYDzIwMjYwMTA4MTkwMjAzWjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlw
ZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxUL2JhLzA5NDdmMi0yMmNhLTQ3Y2ItODlhZC0zZ
TUwYTVmMDE5OTgvMS9mOUdLYWttUkRNTXgzSkVSU3VXYmNZWFY4dzAubWZ0MIHRBCDOHe
e/kR7uuSXgV8NAdqOlT3rD5l6+8gKpPOsQAIockAICB84EFH9K5X7m4KnFEB/BSoelM0F
Qu6tGAgIHCxgPMjAyNjAxMDgyMTAxNThaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBr
aS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvMmIvMTFmYTQ0LTg4NzktNGQxOS1iN
DY4LTViNWQ5OGY1MzYxZS8xL2YwcmxmdWJncWNVUUg4RktoNlV6UVZDN3EwWS5tZnQwgd
EEINQB8UTqGOQVuKymbJF9l032P1avGA5HYUZOzZBlJUOpAgIHzgQUf2qOXVXCSYqCY2+

Z+PyeMZ4Hdx4CAhC8GA8yMDI2MDEwODIxMDE1NlowfjB8BggrBgEFBQcwC4ZwcnN5bmM6
Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC83NC8yN2ZjZTEtMTFjZS00Y
TMyLWE2NDktZTA3NmI1MTcyMWFkLzEvZjJxT1hWWENTWXFDWTItWi1QeWVNWjRIZHg0Lm
1mdDCB0QQg1SElzrrX16Q+poOs2aDApHsXvKrqfpGpM/emj5l2WlUCAgfOBBR/HQ4ymL7
Tp/Ofs7JE7ZGL9sTXvwICFZEYDzIwMjYwMTA4MjAwMTMyWjB+MHwGCCsGAQUFBzALhnBy
c3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBVUxUL2Y3L2U2NTA2ZS03N
jg1LTQ4ZTctYTU4My0yMWFmM2RlZThlZTkvMS9meDBPTXBpLTA2ZnpuN095Uk8yUmlfYk
UxNzgubWZ0MIHRBCDZ3cCnM1f1TWqRljN/6JJMPxp4i98OoQ/iByRh11U2bAICB84EFH8
11a5Bf0XuhQXXbOqhs0xFg5SgAgIQlRgPMjAyNjAxMDgxNzAxMDVaMH4wfAYIKwYBBQUH
MAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvZTYvYzU0N
jdiLTM5NjItNGI3NC1hZTBkLTM0NDczYmM5MWQ4MC8xL2Z6WFZya0ZfUmU2RkJkZHM2cU
d6VEVXRGxLQS5tZnQwgdEEINySvZqi93Y/d9zNNx6gq0dxUhmQuVj1t4LKdJ+XtT5eAgI
HhAQUf35evW+kacXjcL/BBsTUqtHh25QCAhAjGA8yMDI2MDEwODIxMDExNFowfjB8Bggr
BgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9zaXRvcnkvREVGQVVMVC8yM
y9jM2U4ZWQtNTk3YS00ODVhLTk0YTMtOTgyY2ZiMzU5YWVlLzEvZjM1ZXZXLWthY1hqY0
xfQkJzVFVxdEhoMjVRLm1mdDCB0QQg4BXwVc2t3CYGZHmeWUo+F2MGbMmIsMpazuR20W1
ttMgCAgfOBBR/m0z9ybDZ48MeDruB5vGxy73J5AICDZsYDzIwMjYwMTA4MjMwMTAwWjB+
MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvcmVwb3NpdG9yeS9ERUZBV
UxULzViLzgyNDZlYi04NDY5LTQyZGYtYTA4Zi05MGM0MjFmNTVhZGIvMS9mNXRNX2Ntdz
JlUERIZzY3Z2VieHNjdTl5ZVEubWZ0MIHRBCDju+/+aQNhL3xABcZCmQwPlJsSt9PVczu
iI1LEhh0V9gICB4QEFH8pZlevjQDCX9dfVuzIoos1FQV1AgIQ8BgPMjAyNjAxMDgyMjAx
MzJaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlLm5ldC9yZXBvc2l0b3J5L
0RFRkFVTFQvZDAvZTgwYjMzLTNiZWUtNDZlYy1iMzVkLWJhZjk1YTUwNmQxOS8xL2Z5bG
1WNi1OQU1KZjExOVc3TWlpaXpVVkJYVS5tZnQwgdEEIOhXNpfSOMASGctVcKbvN/OB6ql
5D1pQacHPcCY/ZHiyAgIIGAQUf/G4HP5quxGOl+AyW2Yur5hPL2oCAg4nGA8yMDI2MDEw
ODE4MDExNFowfjB8BggrBgEFBQcwC4ZwcnN5bmM6Ly9ycGtpLnJpcGUubmV0L3JlcG9za
XRvcnkvREVGQVVMVC83NS8xODhhMTgtNWQ3Zi00M2VkLWIwNmEtY2VhN2ViMzUwNTM3Lz
EvZl9HNEhQNXF1eEdPbC1BeVcyWXVyNWhQTDJvLm1mdDCB0QQg7ofzO61BsU/Hy5uia4R
YPFpIBkpqq5VsbWs8rtg3Ff8CAgfOBBR/F4+vZAHi83FuMXZFad9zHfWPIgICEi4YDzIw
MjYwMTA4MTUwMTA4WjB+MHwGCCsGAQUFBzALhnByc3luYzovL3Jwa2kucmlwZS5uZXQvc
mVwb3NpdG9yeS9ERUZBVUxULzY2LzFiNzk2OC1jYTRkLTQ1ZjQtYjY3MS0yZTdmNzg0OD
ljZDMvMS9meGVQcjJRQjR2TnhiakYyUlduZmN4MzFqeUkubWZ0MIHRBCDu2djmK3gbyPB
qskEsLEV+na+Ot0G2TJurk/7Nc14YQQICB84EFH8km5VEYgaD+Us4inVRpopkk+0SAgID
6xgPMjAyNjAxMDgxODAxNDBaMH4wfAYIKwYBBQUHMAuGcHJzeW5jOi8vcnBraS5yaXBlL
m5ldC9yZXBvc2l0b3J5L0RFRkFVTFQvOGIvN2FhMDRlLTQ4MDctNDk4OC05MTAzLTg0Mj
M5N2UzMDY0My8xL2Z5U2JsVVJpQm9QNVN6aUtkVkdtaW1TVDdSSS5tZnQ=

## Appendix C.  Producing a FQDN Snapshot

The following terminal transcript illustrates how one could prepare a Prefetch Response for rpki.ripe.net using common command line utilities.

```
$ cd /var/cache/rpki-client/rpki.ripe.net/
$ export TMP="$(mktemp)"
$ find . -type f | xargs -r cat -- | gzip > "${TMP}"
$ mv "${TMP}" /var/www/htdocs/erik/snapshot/rpki.ripe.net
$ unset TMP

$ cd /var/www/htdocs/erik/snapshot/

$ ls -nl rpki.ripe.net
-rw-r--r--  1 1000  0  98392415 Dec 12 15:59 rpki.ripe.net

$ file rpki.ripe.net
rpki.ripe.net: gzip compressed data, from Unix

$ gzcat rpki.ripe.net | openssl asn1parse -inform DER | grep -c :d=0
109880
```

## Appendix D.  Producing a Time-Trimmed Tail Queue

The following terminal transcript is an overly simplistic illustration how one could prepare the 5min and 10min time-trimmed tail queue Prefetch Responses using common command line utilities.

```
$ cd /var/cache/rpki-client/
$ export TMP="$(mktemp)"
$ find * -mmin -5 -type f | xargs -r cat -- | gzip > "${TMP}"
$ mv "${TMP}" /var/www/htdocs/erik/tail/5min
$ find * -mmin -10 -type f | xargs -r cat -- | gzip > "${TMP}"
$ mv "${TMP}" /var/www/htdocs/erik/tail/10min
$ unset TMP

$ cd /var/www/htdocs/erik/tail/

$ ls -nl 10min 5min
-rw-r--r--  1 1000  0  946037 Dec 12 20:02 10min
-rw-r--r--  1 1000  0  642883 Dec 12 20:02 5min

$ gzcat 5min | openssl asn1parse -inform DER | grep -c :d=0
206
$ gzcat 10min | openssl asn1parse -inform DER | grep -c :d=0
398
```

## Acknowledgements

The authors wish to thank George Michaelson, Theo de Raadt, Bob Beck, Theo Buehler, William McCall, Jasdip Singh, and Michael Hollyman, for the lovely conversations that led to this proposal.  The authors wish to thank Sean Turner and Russ Housley for their review of the ASN.1 notation.

This protocol is named after Erik Bais, who passed away in 2024, as a small token of appreciation for his friendship.

## Authors' Addresses

Job Snijders
BSD Software Development
Amsterdam
The Netherlands
Email: job@bsd.nl
URI:   https://www.bsd.nl/

Tim Bruijnzeels
RIPE NCC
The Netherlands
Email: tim@ripe.net

Tom Harrison
APNIC
Australia
Email: tomh@apnic.net

Wataru Ohgai
JPNIC
Japan
Email: alt@nic.ad.jp