

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

J. Snijders
BSD
T. Bruijnzeels
RIPE NCC
T. Harrison
APNIC
W. Ohgai
JPNIC
15 October 2025

The Erik Synchronization Protocol for use with the Resource Public Key
Infrastructure (RPKI)
draft-spaghetti-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

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 18 April 2026.

Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
1.1. Background	3
1.2. Requirements Language	4
1.3. Related Work	4
1.4. Glossary	4
2. Informal Overview	5
3. Erik Synchronization Data Structure Definitions	5
3.1. General Syntax	7
3.1.1. eContentType	7
3.1.2. eContent	7
3.2. ErikIndex	7
3.2.1. The version field	8
3.2.2. The indexScope field	8
3.2.3. The indexTime field	8
3.2.4. The hashAlg field	8
3.2.5. The partitionList field	8
3.3. ErikPartition	9
3.3.1. The version field	9
3.3.2. The partitionTime field	9
3.3.3. The hashAlg field	9
3.3.4. The manifestList field	9
4. Client-side Processing	10
5. Querying an Erik Relay	10
5.1. Fetching objects by hash	11
5.2. Fetching ErikIndex objects	11
6. Transport Error Detection and Handling	11
7. Setting Up an Erik Relay	11
8. Comparison with other RPKI transport protocols	12
8.1. Comparison with Rsync	12
8.2. Comparison with RRDP	12
8.2.1. Garbage Collection	13
9. Open Questions	13

10. Operational Considerations	13
10.1. Scaling considerations	13
10.2. HTTP Compression	14
11. Security Considerations	14
12. IANA Considerations	14
12.1. S/MIME Module Identifier	14
12.2. SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)	15
12.3. Well-Known URI	15
13. References	15
13.1. Normative References	15
13.2. Informative References	16
Appendix A. Implementation status	18
Appendix B. Example objects	19
B.1. Example ErikIndex	19
B.2. Example ErikPartition	23
Acknowledgements	28
Authors' Addresses	28

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 introduced 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 Partition identifiers and associated ErikPartition objects' 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 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 Index calculated from the locally cached objects). With the ErikIndex in hand, the client can determine which ErikPartition are missing and fetch accordingly. 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

In this synchronization protocol the `_signal layer_` makes use of DER-encoded messages [X.690].

`_Design note: DER encoding was selected for its canonical properties and because RPKI cache implementations already support ASN.1 encoding._`

```

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 PKIX1Implicit88 -- in [RFC5280]
  { iso(1) identified-organization(3) dod(6) internet(1) security(5)
    mechanisms(5) pkix(7) id-mod(0) id-pkix1-implicit(19) }
;

EncapsulatedContentInfo ::= SEQUENCE {
  eContentType      CONTENT-TYPE.&id({ContentSet}),
  eContent          [0] EXPLICIT OCTET STRING
    (CONTAINING CONTENT-TYPE.&Type({ContentSet}{@eContentType})) OPTIONAL }

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) identified-organization(3) dod(6) internet(1) private(4)
    enterprise(1) snijders(41948) erikindex(826) }

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

ub-Partitions INTEGER ::= 1024

PartitionRef ::= SEQUENCE {
  identifier        INTEGER (1..ub-Partitions),

```

```

    hash          Digest,
    size          INTEGER (100..MAX) }

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

id-ct-rpkiErikPartition OBJECT IDENTIFIER ::=
    { iso(1) identified-organization(3) dod(6) internet(1) private(4)
      enterprise(1) snijders(41948) erikpartition(827) }

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,
    location      SEQUENCE SIZE (1..MAX) OF AccessDescription }
END

```

3.1. General Syntax

The content of an Erik object is an instance of EncapsulatedContentInfo.

3.1.1. eContentType

The eContentType is an OID specifying the type of payload in this object.

3.1.2. eContent

The eContent is the payload of the Erik object encapsulated in an OCTET STRING.

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 this ErikIndex. The field's value roughly indicates 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].

Design note: using the most recent partitionTime, rather than the local system's notion of "now", helps reduce churn in distributed systems.

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 3-tuple consisting of the partition identifier, 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 partition identifier.

3.3. ErikPartition

An ErikPartition represents a subset of manifest objects available under a given FQDN. Each ErikPartition is an ordered listing of the manifest objects' hashes, manifestNumber values, thisUpdate values, and their end-entity certificates' SIA extension values.

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 contained within this ErikPartition. The field's value roughly indicates 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].

Design note: using the most recent manifest thisUpdate value, rather than the local system's notion of "now", helps reduce churn in distributed systems.

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 fetching 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.

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:
c2d0427bc5a32c42eealab5663d592b1fc29c7d4ef16ab0b5eld631d039dcc21.

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

5.2. Fetching ErikIndex objects

The URIs to fetch ErikIndex objects can be constructed using the following Well-Known URI template with the erik keyword as suffix and the FQDN as parameter: https://{relay_host}/.well-known/erik/index/{FQDN}.

For 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.

6. 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.

7. 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]).

Design notes: a decision must be made on a deterministic "manifest-to-partition" assignment scheme. Job's proof-of-concept relay (see Appendix A) uses the first few octets of the the Manifest's AKI as a stable partition assignment scheme. Other strategies could be to assign manifests to ErikPartitions based on the "hour-of-day" of the CMS signing timestamp, or the first few octets of the SHA-256 of the manifest object._

8. 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.

8.1. 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.

8.2. 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.

8.2.1. 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.

9. Open Questions

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

- * Which of the possible deterministic manifest-to-partition assignment strategies yield the best results? AKI?
- * Are deterministic and cheap Snapshots possible? If so, what is the best archive format for Snapshots? The ustar/gzip combo might not easily yield deterministic results across different implementations.
- * Is the concept of Differentials/Deltas needed in Erik Synchronization?
- * What will be the upper bound for the number of partitions? (ub-Partitions)

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, therefore it is RECOMMENDED for clients and relays to offer support for compressed content coding, as described in Section 8.4.1 of [RFC9110].

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
TBD	id-mod-rpkiErikSynchronization-2025	[this-draft]

12.2. SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)

The IANA is requested to allocate for this document in the "SMI Security for S/MIME CMS Content Type (1.2.840.113549.1.9.16.1)" registry:

Decimal	Description	References
TBD	id-ct-rpkiErikIndex	[this-draft]
TBD	id-ct-rpkiErikPartition	[this-draft]

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

For the current proof-of-concept phase, OIDs for id-ct-rpkiErikIndex and id-ct-rpkiErikPartition were assigned from a PEN arc. Implementers are not expected to make any special accommodations to retain backwards compatibility with these PEN OIDs, after IANA assigns OIDs.

12.3. Well-Known URI

An URI Suffix in the Well-Known URIs registry specific to Erik synchronization will be requested. See <https://github.com/protocol-registries/well-known-uris/issues/67> for the request.

The proposed suffix is erik.

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>>.

- [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>>.
- [rpkitouch] Snijders, J., "rpkitouch", September 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 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, Montreal, Osaka, São Paulo, 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://fnllwqoupfrhso6643whm6lpkgsftjtc6crpehmyz2o7pffirngy7rad.ion/> Erik relay service via Tor

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

Appendix B. Example objects

Included in this section are an ErikIndex for rpk.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/rpk.ripe.net>.

```

MIIRxwYKKwYBBAGCxlYGOqCCK7cEgiuzMIIRrxYNcnBraS5yaXB1Lm5ldBgPMjAyNTA5MjMw
NzQzMjRaBg1ghkgBZQMEAgEwgiuAMCkCAQAEIMQZXNo1VHt7ExBd0EM0uv+/oqftDzHqo3m
KNRkIPj8AgI7/zApAgEBBCAoNYzf0h4GMfXQgF6r+ZBztQCpIMjA+Hjo/MximoIk5AICRZAw
KQIBAgQgOf7md80+smDgtzzFzZWJr2nsPRTEvG3zO6or479L/gACaj2VMCkCAQMEIMNDymEH
tI02zXKe3S3iRylo9wNJ7d67xd6/X6qIpYTuAgJGWzApAgEEBCB9S1ZqZS7oRGL4flxkG1zW
85irPg8RC6ACjrmU5LdPqgICPZYwKQIBBQqGwChTaYoMduMdh0l/2M15X42Fha3lscqvjSoR
Sg1BA5MCAky6MCKCAQYEIDLuxOgSRpTAGyuRKRz59hRRvSmN64rqWFwyi3q7M04HAgI/+zAp
AgEHBBCDF+bRH/b3ZLAeGkHFZ+fXqjYdw3XhoEssgA2GQF6ucJAICR/QwKQIBCAQgpbF75mXQ
vks5nE1JkMcAiQ6JrV886N5Hjm/fOvPr2WkCAki9MCKCAQkEIBQSG0aKaex+jJuJ/V50d2Jt
2Ml+yHiNZgh4QZhrIOFEAgJJiTApAgEKBCA58F26WIwAkHsp70tCBzFLLIgFAVFPgrQGp92z
uzwT5AICOAMwKQIBCWqgPu7gxZWUyYwG9Vb6VDJZEQR8EkVi8vF+ZMmyZXjHg6wCAkZbMCKC
AQWEIjJrocdQZxfxV5uLc2F4IwHgRhfeleB/rFwwNSE/yL42AgI5mzApAgENBCA+UHXU3nBN
xFH0IcSfy0yso0niOQy9+CplPZWcf9ESQICSLYwKQIBDgQgw8Z/uL9p7y7Rj/u2qeK8BJgF
4Ehn3XF6CI2XWsl9LgUCAkMsMCKCAQ8EIC/TzbiXWkH8Yh2mP8iodVK7X1GO9QeSsTO/5Qek
CeaUAgI8zDapAgEQBCDVpNTqAs38MWvEqLDzJ1TFckJ9CVgYmt6FSFSQ42xygAICQMwYwKQIB
EQQgOZeS+kJLCc4PvMVB41k36n7cxbtefYYg/94q/UmpXacCAkWPmCKCARIEIMQ/ozLa/Pjn
RU7q6BdUaQTdiPcapM7R3nZ8BGBtB6kYAgI/MDApAgETBCAHDUEkVfsUAOXfLipmGKRvE/VI
NzGXRmv2cn2eah/qtQICQyswKQIBFAQgft7T086QzTfzkdIA4mbDWNTHvfr9cv7Qkf6qSWp
iEsCAjDWMCKCARUEIFJgtglgsk63fKXz00AWpXOF4dTJa16+Kj8EGaVglxcEAgJD+DapAgEW
BCDXPJFNnFRND7Tt38BckhSjmyhesDE6JQvozfxV2qWj+gICSLYwKQIBFwQgIPAqhHGmrcdE
SBvsFxoAUMWU0c7FnxaVomXjUGDvHWGCAk/nMCKCARGeIEwrNbh0U2GXmOBcdXXG/Zq+tRYq
OOhOVkgmJWktHYxUAGI0BzApAgEZBCAGJ6OlmTISMftkbnawVG6o6AD648aLZeXGgb/nSn1
LQICSLgWQKIBGgQgbPQOuOGKA92uXYX+jGoay7rU3CWee7y4+7JgcMyE2AECak2IMCKCARsE
IDx6NSUB07GVhDfDaAKg85VA1syyVuuPe+eH1+FysJnXAgJD+DapAgEcBCAXh9E7r1iTkXXU
X8jdIvbh0A4nrYlNtQ86zT0DbmYS0AICS+4wKQIBHQqg48JuqRvkZAgHgfzYOZg5meIZ/AxK
MExy7maWh499kFwCAkMsMCKCAR4EIFreiG1PQYEn2NqKd/V9BHKJKyHjCRLnviEl2sKqs0Wy
AgJHJzApAgEfBCDwm7LyCy/LF2CBQLBm/HI+leFRcwPi9+QSN/Uo4uNAYQICRYcwKQIBIAQg
EGsvGiA5MU8RmJ3UPnpTrzg5edaDZjaOIjjJxJ1FiRECAkZaMCKCASEEIAEli5NaZGpClrxR
IhTRBmNSNyLnnlNeFq/hZbjeGIRrAgI4ATApAgEiBCCvjQN8pjyQdwv4npj5yKleFGb2280j
MNCgchn/TgCj6QIC0ZswKQIBIwQgy8lv2EvCBuW5PEe5WF9Mgv/tY3zMUaGM2YIJjAW/Y4kC
Aky4MCKCASQEIK8S9WY8uEH+ol3ywoyrIlaAZEvhMLmY7qQZPoePLagPAgI8zDapAgElBCBI
b3IEFsxamFjwc0Mb6SErStl9vycM+ks917jbexU+ewICPzAwKQIBJgQgRUF0S817BPFYDGHr
f8Ypk/PhqmnSLzrl03atfj/6thgCAkJgMCKCASCEIP/gBrFfzcCA5FTNm+kEX50gGvpXcNV
4c+UsIYkuJzEAgJOVDapAgEoBCB2HaaEs07TKd8Sa+dPd74VLxQIyKoLLYjjda21x89PPwIC
PMswKQIBKQqg0rKzmpODTukXeU4rb9WPX2RUPPgNcLoMTfynBiCf0s0CALWAMCKCASoEIB2U
82jsoBByaKImCjYPgMosPq7kebkZYTnm7VGVx40PAgI7DDApAgErBCCn4mFQ8D9ZyZhEkOT3
g2UqG2/iN/aIGUas65A9NqhohgICoZiWQKIBLAQg3SuXjf7EBPmHbz5dAzBe28SCOPnonS8+

```

jS3B3kSHDPGcaJmcMCKCAS0EIK4vPdPDJ3rJl5keyW1aQncvIogHGtW3AhjGmiUwHODFAgI2
azApAgEuBCABO4uMPvD4uVPNdmiWYq9jfTr/ZfeWOG7jqDgSsHIARgICRY0wKQIBLwQg2yRc
rxYv0gYnSd8oH0rH9yzo5F6CQnEZ4AK659+CZwECAk2GMCKCATAEIHylJjOmb1X7tpQ+hhvG
rSL7Gub/Ati/ZiKl2pM6CxFpAgJGXDAPAgExBCDaeWd7ItQNpvh4Wi/6NJmesMcLwmJyYOF
XgDOEQ/adwICozMwKQIBMGqgf4PqCNiyzaVwZALeZSZSo6jKqTY9j9PjTerq7ChU/fECAkja
MCKCATMEIH5zpoCHWQfrqiJk87T0LYoNyWFDPIcvGSMJTs4VDDyKAgI7MzApAgE0BCAirpuD
iutSDpC2o36up5TcENBmn+QBCq7iqUv/LZUfMAICSlgwKQIBNQqg//kB5S436tznJCDNt1/v
jJjliHfdN5bkWEXajoYPlsACAjzMMCKCATYEIPsf1Qxbdluw4f8w9Vf1Ptbc3dURvkApzeOn
aJ7lY6suAgJKVzApAgE3BCAoiXuOwRBpkz3DEGrMwRsXNfiPS2mRjyk7M2TSSnmGwWICPmMw
KQIBOAQgiMm+9YwiHo3zFWW9MyxUNG7zFN2djCMWZ7Te9MqauAsCAkcMCKCATkEIOjC0lLT
lFyc6jMtezlsVnF5l8j/DLbdk2jD8B27GYdpAgJCYDAPAgE6BCBXPauC3JNUR0lT+7XDu89F
v5KUDUEGNFZkUZVMt+b1QICRMiWkQIBOWqgDU5QaUAeD/kOyh94j30C1xbM/GfBNnlxKraJ
QwQ22r0CAj/8MCKCATWEIOBYxCoKbHifuOXXYKV+x5vtRwdSWlgEVdOso7nHIVjGAgI5nDAP
AgE9BCCyhZlF4NlPdqBYiZ35JGqt7kFgPfADwZs/pL231D1KDAICozMwKQIBPgQgvQyEeAbx
WWhkAdp59KTMk6DHq0lscxqRvxbf/urHClScaKpWMCKCAT8EIO903XHgxVATHAwuhNWcpWwY
J4sRtchwHB/SVkBtqn9eAgI3ODAPAgFABCD5c8CSzCIVL/SOHxLBTJpxcfqDXKhJmF58s00x
xU30kQICSYwwKQIBQQqgKbyx6u9nWR8mpvFBDHGutbNqCa22wkNPa3WYmBXP7M8CAkjaMCKC
AUIEIKUMi0hOCEio3gemMf7BVbdtDv1MjKazMXa6rVUaarVqAgJEXDAPAgFDBCCfY0q2Ggwe
cwkJltxl1+PlHRWJw5x6wCVtdssAuI//wgICQMYwKQIBRAQgAZfbJoikGBYDm0+vBIW0M1JZ
PPzGHJ4cWB+Ql2EP1gACAKsJmCKCAUUEII2qWecjRcWhW7pxxf29VrG8UXKBO2woJX7z+aT9
BINmAgIlnzApAgFGBCBYlSBEz6wAnl0l8ozEUIELAlMmqLoJUJ3fHZrethqS9wICQ/cwKQIB
RwQgKEQlaT9x7BsXFEMaetDgenADknBxsXhdXsqYArIt/FgCAkGSMCKCAUGEIF2kp9NZAHl+
HFPLk4+sAYFzv3J45frBnjecNI+ua5AmAgI/MDAPAgFJBCCwjoBtFb0R4DzoUrRYEwG7n0j3
yTS41wMi3GDAq3bKigICRMMwKQIBSgQgXW7ZhJY3Q4B0N/gb+PiOuYf1I/SQcxrm8MmArfaU
iesCAj8wMCKCAUSeIKDpYvcxg5dN5v4SC6cOVkcmYunl6w9sTWI54y0M6T++AgJfKdAPAgFM
BCCJEC2pnbSHsAVLkMDTcqbfdHElUwRa7ZPcLsaJhZTHAICTlQwKQIBTQQgKOV9QZ0lGnsK
PXgm0mWH84+NyPSAYHAAY08XcXNh2TgCAkTCMCKCAU4EIMyKM1NKY0xrfLBECPMga73tbtib
+MsQUHPdp5dJsBj8AgJivjApAgFPBCAX7oVWy+iuIs/3E4o79ykQlkgLlNtcwhOTVY3ri3N8
awICPMswKQIBUAQg5q66DsfGviCnY2QurViJW9Pj0QS1IULj9nKB9qLfNvsCAkJgMCKCAVEE
INBr2RynL0fYlv3kChzeVctoWid0UX95/FN292Clh8CagI7MjApAgFSBCAUfQEnV+fPPoyd
ZkbjmoCYBw4gWebvRhvsFxAIEvEo4wICRycwKQIBUwQgEPGOT59Yjv0mpWVJZdqp4ohc93rE
dFVlTzaOfV/IAVQCALJQMCKCAVQEIGzpxOH8NE+bV3pQEcSj473dW5uvW4v0LAWkWqNEBbVi
AgJH8jApAgFVBCCIY99guAcOGUeIHMCoRA4lGZMhKnt4g/x29AZbP+3JjgICQMYwKQIBVgQg
Z9DT88fkjGmJJ05dyjvyX8L8VLPAjLfbeg3VKANYSX4CAjDYMCKCAVcEIBsyD47yBjRiv5IC
4ApGyuc0iOQ83+ds+lPwPX3fY5jhAgJNhZApAgFYBCAnrR6ExogF072thbuvi3AT0R4M0m0Y
mbulkdnUCkMvbAICR/QwKQIBWQqgTk/L34ToQpRgWfPQwTSNK77xkjhOMANXWpR2se22ynsC
Aj/8MCKCAVoEICMyz9/tu+V3JwZ8MViHoCzPUMIcZg8+q1IbjIoJtozVAgJDKzApAgFbBCBN
SKD4zohF+nW4sYZblDDAPSlnmL3wYuA8fzJvANrxIQICQMcwKQIBXAQgircchc+d/kc6J9bz4
AwYM+FDD5FYsTE8ns0ScxLojED4CAjmbMCKCAV0EIO2ZoR83S8ofRgPI2YrYoD9yVp57mfW6
QcZ74anFGS56AgI//DAPAgFeBCBPvKhdgeRa2L+L6WjpO27zoJGX9nvfFCiocIEL6iHrDQIC
SlgwKQIBXwQg8jwdAgVrYCXxipFZrmcR00kOeWtrqrrxHXUTzG359IECAj5jMCKCAWAEIM2j
ntkBhb9bliQe78oxeWT/IEDvQcKBPWxVyDvPiQEUAgJMuzApAgFhBCAIh/zMmD+SVhj3JVl1
2QZmMpC+26y12F8Ak7AiIA66tQICQZiWkQIBYgQgqlp0vHJFSWdlLjZqldp82S4jIfnpI1Bg
OyZFmLCPGIMCAKGSCKCAWMEIMOT9zPT03YIZOFF/SFESqccXjojVGW+DsHyJk7HK3xdAgI+
ZTApAgFkBCA6Rh20Q7/0hjAlJHGe+v/dGERHj4zdZ29axLt/YWmu4AICQMcwKQIBZQQgYxkg
x2WcquxlJNi9OiB8bPakkFkMhmzIyROeDydzNQMCaj/8MCKCAWYEIBH/lykwYg1J2Q9/DvHS
SUXdDO32g+3RwLyeLBAoxMiIAGJHKDAPAgFnBCC6uyOvK4LsQwWuBJCBF4r5DU4j+KJXvUFZ
vfsGEHfbpwICRY8wKQIBaAQg8EY69QtWbahOfG92DEVHy2eHCUCUDsjlYSmuHmITxsACAKmM

MCKCAWKEIJpcOYmZ8MujSJJaFYvM5K5j9lBxhofkljxcaIqGxoPPYAgJQtzApAgFqBCAlGv05
Rh5OQ7mwJv87osh8bQPbAuDkUGJ3ANiMppwZRGICNNiWQIBawQgkQr4E3vuKamjxV2L3GUL
Ib56KS4xv9wQGg2HjboCz5oCAkpyMCKCAWWEIILZR9bVw2sUzhd4chOS4zB3l3agMaYZBmWb
SI9qNa6SagJBkzApAgFtBCDLGfJhrJjU7OwGDJ4ixNMvZmLpV+D2NMjp92cKzLFDuBQICPy8w
KQIBbgQg/6GZNBQON3bz9atx2tBF2S2RZqKGVcjpAsCBu6OhEvACAjc3MCKCAW8EIJDgoKBE
E8zDAXt36MhQ20IMJnG3Ofv/myfiKrfGtcPdAgJMuzApAgFwBCD/nzdfcH5Slgb0lblVbIHq
MlaA85t5iBnJSG6IzKttEwICSMawKQIBcQQg+0fv3Yt6vW5CIdlVipe5GVuaBojCaQYKSWdT
73gCs8cCAjTUMCKCAXIEINTtL6cL26i/1lPuKkpnwoD4Ph7L0Q719SRILWNHKnA0AgJivTap
AgFzBCBuWOIXRwE/Q+ziP5kEspdY+PssCCT0T3PS2mDIFhJ8eAICPMswKQIBdAgqkdbdzZz2w
SqQLl8RWlyRQEuN2F2f8uwk6Ug7ksSiJe0ECAky6MCKCAXUEIOy9+08NQoiPOemcy1lH2yNS
I4UXNDPq3pullxT5Pqs+AgJAxjApAgF2BCBT/WixKHycPN22wsCcJO7D2+joskxG9q/ObJqa
QtIh9QICP/owKQIBdwQgFlfQmWGbI8zHeGrsrGGjoyFs332p5f9PlnNYA7WFAv4CAkvvMCKC
AXgeIK417E+03KRfdd8tCuZ/EshTg09Tbt9eZ+KI94MRAH39AgI/+jApAgF5BCA8W0kiSgm9
ST/HtL915rilGBZxmz7PbPtXvmCD5MQh6gICSMawKQIBegQg7nJOr9Sn1Nuk485VKrSHzF5+
Wlywrl+eTRzfBrOT95wCAj/8MCKCAXseIMzFBzjNexi2DyyGC6ZLDdRfD/ngU0gXDH74jKyf
e8c2AgJCYDApAgF8BCAi+tuZqcUztTPAGg/rubUNuADN5I3nfcZpk1/vZksxwICozQwKQIB
fQQgeGel demh7yKilG6hliZ35idSvJFM4Wgz9WQZTi+vaAgCAkMqMCKCAX4EIOKJbuD690mx
Q9jCeBwmvMw2fg6XkYVmBmHhJphuNFIIAgJBkzApAgF/BCBzbNjPRXMUBbtdjsEMzfMa+MtH
TGVGaJaAP8W4LvSYZAICLNwwKgICAIAEIBmwuy0E0KEftqKhrKS1FoRmcJZD2Mx5MtwfOcYT
YnLAAgI00zAqAgIAgQQgs7q9Uie3n8/cL3NwGZfm+CHHTteiwUHYMxPcGVAHQSMCAjv/MCoC
AgCCBC5tAa5suux9IjOhjNXzo2JQrOmd3bZt4yolbA4RoBaUwICPZYwKgICAIMEIO02WG64
XV2nvwEm037cQ5zw5wmrElNIETszis9B7V+SAGJLIjAqAgIAhAQg8y9txNa25EqOsNBVWGGJ2
jl3PDrVqtVlEH9ca+4Bt4w4CAkMsMCoAgCFBCC56nWQBPPxY7mWrXV0XXcih+lJKZ0wZ4dS
bof4hX+LEwICQ/QwKgICAIEILenjosqW4L0iHrliQ+JUzskMRKgOa4BEGWj5vWRjclLagI8
yjaqAgIAhwQgglsxl5WVw9seADyiGaE72CYnvZ4nEqLcmQe8MqpEvoCAj2XMCoAgCIBCD3
vFCHTEh+KPeAhRn2SIOMe6fjkDgC/4ggHcaq9P8lJAICQywwKgICAIEIKK5PWakgV6iKlqG
ldKlGbeXGdwLrWWIyl08Gi69c73RagI/LzAqAgIAigQguXQmiZhvcosCTZdjTTvmOk56i6M3
HI4DIwed6siSerICakvuMCoAgCLBCDIndMvo3ogQRAKXF8uEic0LpuLLolrS0YPSlOEaLlP
mwICQZQwKgICAIEILO5JUDxl091w20Ude5lnpYiuUZwHKlywKU2sX+/TQn0AgI4BDAqAgIA
jQQgyDjhvs1ob7Dc6yYbN08t5R0TYitL3G9uyiT957fwlRcCAkcmMCoAgCOBCCkqHZUbgVm
3zReAZqBd2MdDV+iS2WcsNALBxwH8snp0wICRYwKgICAIEIDW9whMf/QNCi+PKtfHbydjf
3jt+m1lIUj6eFyJcWqQIAgJHJzAqAgIAkAQg7E4WFHqRlHrB+HnZrQ+/4AtECG8D97J3JqWT
Yg+X8bgCAj3MCoAgCRBCBhVOInUbYQkuVELKIRji/vu7cgkar/aFqq/XoKbClQaQICNAUw
KgICAJIEIDllMvLbs9VlYVEHOYUgJ7hb6XlsDonJt+mEL+lwyB4AgJNhzaqAgIAkwQgE5e8
KYVNqnWPJKAcMHGf3Lgvt2BJNiLnyi4Laz2odqgCAkcoMCoAgCUBCC4NrKPK3eyKwjbc8Y8
o2jl7lVlmq4ZfrnjbYs6kiu2wIComgwKgICAJUEIHSzD38FxLhd7dvfUIBlc0SXUJKcNv5M
6LNER0EirdCeAgI+YzaqAgIAlgQgr55IZx76I8XqDtSN0uRcYSj8PNgZNfezaUPZ31V8EvsC
AkZaMCoAgCXBCBjyIayUhs54RGrfnchDlz57+Y7KitWUEgYLS0B114abAICPmIwKgICAJgE
IHzwBCY7/XyKsIZCI5YtPL31L76j2X91xkLIQMTPr18SagI7MjaqAgIAMQg29bTNYldheAO
SxXftQWRJqXf6QYTWlos4kvHsmk0JysCAkcoMCoAgCaBCBEXQTxsZXRqIISbBqGakhu4B/T
OfIrod8D6rm+sQqXEGICSL8wKgICAJSIEIORHLMbPwcm9fivUBCRqivIiFW+4qFi+enRZZRAp
pAZgAgI/MDAqAgIANaQgokB+ngs/047/lCN68n4ibrUAOW2zHBYu7pSF+2oabLMCAj5kMCoC
AgCdBCDCwJ00dbMZtU35d3QC+LOJCGF6lUZviiGSR7t9E7kiqWICQZQwKgICAJ4EIAkL7mck
RoqLjWguzWm+Ex/sRiUD0JjvREkU9DAqODbAgJOVDAqAgIANwQgRuDth3cAlOsUwaepMGIt
OcvVqyVMebhfpHOZu6xzfVUCAkTEMCoAgCgBCC07q37BjH9GRci34q6NmV+oP0md8NhUBg1
ng0lSuz8UwICPMwwKgICAKEEIEvDhTg+59Xw04TMSndzamKo+g4pdrrc7YD7bq7y+WpBAGI/
LzAqAgIAogQglqCQC5fqXUhrSiITZBuJAw+jgGHrxpvn+Uiz/RfSjyYCAjgDMCoAgCjBCBa
NixpBhu/UtsarRfcp9ZQbZ0C9ayMkRoZ23r5pYfjYAICQZMwKgICAKQEIEJtIllaLExnXUZC

arC0Vucxn909KxMrRKemPxKvB2bQAgJOUTAqAgIApQQg6gJi3c/MAuJxFV0yUc5B2bmFTBKd
8FzLaoMBwJYQyJ8CAkTAMCoCagCmBCAlrfuygRz/Ir7ZkrPUe9p4Rpg9vKlSoEn0sJzluM8E
LgICPy4wKgICAKcEIodvizJLM604sLN7KoizKWmm4UioRulm9gr2SHSC2Nd3AgI/LjAqAgIA
qAQg5HV5XtWcW5u+MliilDZmQpdW0+FsA/RQAHEyw0JaQSoCAkGSMCoCagCpBCCxjlyZvKV+
4waavugsqEmC9jnJCQPF+HFqB9cfwzRwxwICPMswKgICAKoEIOyuxv9jKsR/NtqvYmdj5/lP
yUq6gDqXiTQs/pMe4N5oAgJBlDAqAgIAqWQg4+vGsAxV4Ht+l+xqELNM1T/x302Xp7hFffj1
97Gvd5wCAkGTMCoCagCsBCCAqbZwusWzJxat6N2i/Rb+1MWKRGWd4KnP6eHG/jMYNQICRyYw
KgICAK0EIDmdH/DeDDAcvGX4XPf82kzmHEFmzCJco37zyWJhJXjAgJBk jAqAgIAArgQghZvl
JOdIcIDjYiDLcWZ2LJphfkmDqkadrY4nTfE5zECAkMsMCoCagCvBCCyjm7WhojlyKdqQLPi
M4L1MKh2s6eDG396EkwbLHa3AQICU+gwKgICALAEIK7P40FO2XS7mLyFa4UpesUvsZxsmi5+
pcrpDsFu78sgAgI9lzaqAgIAsQQgg4JPqGViXRg3GFBilTtZKjVwhRDpxeB+Gnlgv4yn0sC
AkZaMCoCagCyBCAKUL6DC90lM1/jLa2Sdi10rSWaSYKJscSXAUVJpiQokgICQZMwKgICALME
IEBRG5NI0ixjttKDR5iHTcDKbpW4sLyuu/3aQ22hWPj3AgI+YzAqAgIAAtAQgVB/+e0OP0oBx
Hs3oQCqZ/qEdWm6+aelC1Ty3z3vgoyQCAk2IMCoCagC1BCCL/70uF9gdMCXi jxThwj7wL47Q
S0QBspMPgU5LjuPYSAICSyIwKgICALYEIILr6tKk2wQAx0MfxtHZDUwuHrfr7HVEuUtXN3jB
B697AgI8ADAqAgIAWqgPlxT8ipyGINO9i5Ly/yLmNXU1+OdVrzCfYWbDAGsc5ACAKJgMCoC
AgC4BCDW00hy68Q10ETFUrYWQTp2VsLcQDvZ49URxfbup66KVwICPZcwKgICALkEIDou16H0
Jw5bKTT4eeGwQ4Tfa9AltvpDnj9k+n3AYS3bAgJJizAqAgIAugQg7br4DtGjjP3coqWDRr4V
8CZuzaWGxpTSLHrfXtqTWNcCaj/5MCoCagC7BCB7q69j9Ux/mKwTcHwFHE84rRcxYdYgKuUH
A4B5nMZ+QwICP/owKgICALWEIM+YZHUZWQyE9IxYloBaC2RDSRHD5yMG7pwkGDqh708+AgJH
8DAqAgIAVQqgKOYwTxHkUkfJ3k4+pi6TpOafj7kOnzNNR+0pFUzzlLECAj5jMCoCagC+BCCN
7A2TSITXQ39jU7SCNY9DNKLMu6MaIz0X7TluPzh8jAICozQwKgICAL8EIIDNtWA+clMRGQRz
hwwKQJgxjiWXG32sm4/XxJ89M9WiAgJExDAqAgIAwAQg3Ga6ranqXbMq5frKEF3gs5crvyce
GL9vg51jRgNe3lkCAkMqMCoCagDBBCDDJPgkdEztcfImEqezOuJCrRSECoJ8n6OwlSpG35P
0AICULcwKgICAMIEIGo510jqMJ/6uV4aM66Ccs7RkrCWVwPD3xgsrVSsDRKKAjJDKzAqAgIA
wwQgBP7rR9MyK7N+csFpnDz5Y/k0uUva57AVbCqb5SBRk5cCAjzLMCoCagDEBCBPZtYttDeA
QJhK84UVm9qHTn248Z0mATix9mxQh++kegICP/wwKgICAMUEILF+KzZKi4HjUJ9395oYFMyf
juPYorlgBfIBUgex5m2FAgJH8jAqAgIAxgQguDxTODjzj6QosUbHKqGMVtYgZG4nV+9di/ad
DRJQ2PkCAjzKMCaCagDHBCDrh4tE3YJ7CJPM2i0Ju4HK6jaytmplxCGz3pTQywwvowICQ/cw
KgICAMgEIGemjtdrFSbCzW4o8Nj0CGajSBRks6v5+RwFGsNcTnAMAgI/+zAqAgIAyQQgbc1v
5v98UfjKYn0Fyn55t5bBWR4vmW77sCell1vt4ocCaj/7MCoCagDKBCDkOJ9vGgnTdy14jloP
y/CxswJPMcPOwJVWbflsnlis6AICSywwKgICAMSEIBTgYcoS7IkCsOTEUuUY9wJXcb0SuPir
gdVr1A2YEuUpAgJL7TAqAgIAzAQgsGheWVvilMicZS0xaI4HdiOu/OfXXnldXetUaKKpPqoC
AjQIMCoCagDNBCB8Hh8omWkEpLKouQjM5a1nj5PDULQ/LRkXBD9R06xkuwICSlgwKgICAM4E
ICcQyQeXm1fLSAMNYvyJ+0ohVIPXUX9RpecJr+QuYcHOAgJDLDAqAgIAzwQgc+YPSnZZzRdQ
D/o8m05H84xop4WkulnE14Yuk0pTaaCAkP3MCoCagDQBCCUopVQsXjJW5FTbKoQemKZEyv
3sHSUWJEjN4BGWr0ugICRMwKgICANEEINzwqsQ9DADESbaS0IxLCFK4XiVaSN58f2D5Qkmi
8mqfAgI/+zAqAgIA0gQgK5ZGjRsQ/avSf+ZWoBno8EBX5GgndYK0EVEGgIajpo4CAk8fMCoC
AgDTBCAzzjJFmnlOGtRNSj+XuRdNpeDNqHMiUKvCRAKIrtiTAICQywwKgICANQEIGj12Lz9
xh004PnzXm22rDlixCevYvyBtMLnPZpoEKMAgJFjzAqAgIA1QQgEM+PRSwcFJ9jNNwgW7gb
HfeSaPlIn9cW1VLLBhCSASACaDHMCaCagDWBCDICK0Xb4gK7A7hCR7QTdSxcavQkCi+rtok
thd4HemXswICOMyWgICANcEINSubGag/zG7itqj78MrzqAV0IjhUXgfkI+Wn97bbIYGAgi8
yTAqAgIA2AQgzrrf0mo3oZNH12kaP5YJg0/68fUA3EYrhD07CuvkpAECAj/4MCoCagDZBCDr
XG0aBsLQMUN1WW4Hfg3hIJ2W4732xOeQ/zh8UggZCwICQyswKgICANoEIJN8F23BOUxr9ytR
7OiBUYoxUtYoSMGZF5LMgUg7XnvJAgJJizAqAgIA2wQgScKLPMmcID4UIrEe0slhM6UYFG6E
Z4Mhp2m/iTifFmQCAj/7MCoCagDcBCCpnAxOmkkLvXWJSQP8mw8tMTCS5iqRozTi3ofJUxGo
mwICT+wwKgICAN0EIIge5weMmbhVulqwGbiT3SJfr89xtgVhwxA0U9Z4SK5xAgI5mzAqAgIA
3gQgRr110877dhCJ58SLBBj73ZL1z+tx/avHaewgJYXWbLYCAki/MCoCagDfBCDqHlZveqkE

tUIpfTlq0bE/ZPSaVyFMXs8VnSmxMcV3HgICNmwwKgICAOAEIJKicAXor8ec7/putzWFTulw
fASlj/rGEc00QirHGn8RAgI/+DAqAgIA4QQgbBKfuN+i8Pld6VoA/J3vDqEETrE4l8HclHEk
AjyyZ0YCAkcMCoCAGDiBCDjOq8wNSc72ET5lFT5GJsHZEuToPnADpjbAUglu29vdgICT+sw
KgICAOMEIMMunP77bxEi/TPfmbv23tejLS09sgsF17QLi1XIJQvQAgI8ADAqAgIA5AQg/HIJ
56MNRwTWffObMxudYyliOM/mlwmIZSWZqMnG9NYCAkGRMCoCAGDlBCBqwcPVkU9f9bFaLeN9
ncVWpiTwA6r5lOzi2X2NaTZDsgICPzAwKgICA0YEINoi8+Mdwb2e9sMkOJNFuFgIjbUshx7F
9nndrhI5xESoAgI8yzAqAgIA5wQgEawwnpqhtkuBYzdGSLagC/usXOPAhOlGb4lEJsK4W6cC
AkWPMCoCAGDoBCC04/5rdbpG3uJKFxGFQtXoANDwBx0P5TSvlQLSTxWEZAICThIwKgICA0kE
IFxWi8kv2Pa0lD22STJF/R8ZfJCfb65u+a4QLV2pdZA0AgJDLDAqAgIA6gQgtqdiQL6I340j
mwEZUblbBSXuBLhoxVnRlLPUJlKH/QECAjgCMCoCAGDrBCB8juePUk3yDnCBhfHskziOT5YS
Sr0c00AtmQnz7xwKwWIC0ZgWkgICA0WEIKBPTnXfmtX/4RhWcmSNdRWlMBhL+WlGqhl08CDl
jY6qAgI4ozAqAgIA7QQgxmbLfDqRcWU62ii6Z2WHqBonGhMHISHIhe2g+4pwd0cCAj2XMCcC
AgDuBCCSUCebgymnj8Kya4QX9Z4woH+IskKwWl0UsuC4C4PobgICPZcwKgICA08EIASdcveE
GDqHzjQ0ukJh3cVmlb2T9Lt0GmHoDOPVbkTfAgJDLDAqAgIA8AQgWVfUlP4yqiM6lAEfVcUk
94LD/12uGlZ/c9MLVDCPri8CAjv/MCoCAGDxBCC1FQ8kF18T89kQqh2sPXHQhGnLo0dhSCAN
PGvW5C2ZzAIC0mcwKgICAPIEIB66DhxH5jXOyLOhw4HNNH6aJI97nJ48jctiEw0j6+uAgI8
zDAqAgIA8wQgJa1EXD4yCDaHAKNxBRVGGPqxgoRBWzyMPhMOPs95edcCAjpnMCoCAGD0BCAo
3FTszzQJmMseV2Ksx2VvFY3GIYVCADmG9MZS+ZSjAAICNzgwKgICAPUEIEkeQ+/DpMinvxL3
WftXjKm6jWPN/1ICRxWPJLVOSgRVAgJAXTAqAgIA9gQg3G2pqs0/o48IXglytPlhDx5Az+JV
esKoS7azwQfbbWECAjv/MCoCAGD3BCDPz9CX9T+wd03RGoejDfd8g45l1NVkTWkQv1Op+Any
vAIC0ZkWKgICAPgEIE2kjZzd4f06F4OkC6l8FyDLzSj3fPkL0ziut8C1BWDxAgI+sJAqAgIA
+QQgn4XbqAvQ7X/bQhbDeU783K0sYqWCs/UJ14FDEK2Js18CAjUWMC0CAGD6BCBwTbJlB3u
B/5UGCue0sbfe+gwwI1r85ZWjRNhfqjudAICQESwKgICAPsEIKD3lTXhwZ+14VI+2zealLHg
GhiJ29RpVfDkdbFTCmlSagI/fDAqAgIA/AQgq99/XBu3PCo412fjNZ2Y1ur8JuyRCZUAeC7E
mMKkG9cCAkJgMCoCAGD9BCCGUNfF0q+JK9WdgWcJvqqghhnaJr7ByVCpqqccIcrGBQICNNMw
KgICAP4EIMV4EJ78f5L126U8DQpKc6WS+/0rxvm6LlVJvANUHW0AgI9ljAqAgIA/wQgH50b
PHPlVD2ttD8e6YpSB2T+yyl52SyUhztJcJy4aa8CAkP4

B.2. Example ErikPartition

This object was retrieved from http://miso.sobornost.net/.well-known/ni/sha-256/c2zSaUVzFAW7XY7BDGXzGvjLR0xlRmo2gD_FuC70mGQ.

MIIs2AYKKwYBBAGCxlYGO6CCLMgEgizEMIIswBgPMjAyNTA5MjMwNzAxMDNaBgIghkgBZQME
AgEwgiygMIHJBCAG6UIxgyjIq982vxxKMJv/bowmDbMWO3LvqqHevfX87QICB4QEFH/v69Ff
vGUUh6yvZ0JPR4PUPrQgAgIwLxgPMjAyNTA5MjMwNDAXMDRmHYwdAYIKwYBBQUHMAuGaHJw
a2kucmlwZS5uZXQvcmlwY3NpdG9yeS9ERUZBVUxULzlhLzI2ZWY2MyliNzJmLTJmMmMmOGRm
Yy0yYmM4NDUxOTIyN2EvMS9mLV9yMFYtOFpSU0hySzluUW55SGc5US10Q0EubWZ0MIHJBCAJ
C/5rzn8EctTLdokjtJcRbxjNkGTUDXvS/ul5awnE/6gICCGAEFh8rqG33h4iPb4WSqAam/+ow
RsB6AgIwnxgPMjAyNTA5MjIyMzAxMTlaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQv
cmVwb3NpdG9yeS9ERUZBVUxULzYwLzVlNjc4Ni02Mzc3LTQyMjQyYmEwNi1kYzQ3NjllZmYx
ZjUvMS9meXVvYmZlSGlJOXZOWktvQnFiXzZqQkd3SG8ubWZ0MIHJBCAN0Jm2BGv14ec2gL92
ALy6rhSrwWaoZ9RvLtuakvPH/AICB84EFH+oFqMzjTJ0HNT/+PSd2yzIOP3FAGIEDBgPMjAy
NTA5MjMwNjAwNTdaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwY3NpdG9yeS9E
RUZBVUxULzUzL2E0NTI4YS05MGU2LTrkOTETyTUyYyYlMmMmN2VhNDg1YzYvMS9mNmdXb3pP
TkluUWwMxUF80OUozYkxNzRfYlUubWZ0MIHJBCAQcLeR03m/qXv7ORS2kMM/0AGF7sbyiKRM
FWUjx4lyHAICB84EFH+S33YQ8Ej00/RMM2lBxsEVfvDiAgIwLhgPMjAyNTA5MjIyMzAwMzla
MHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwY3NpdG9yeS9ERUZBVUxULzE3LzU4

YmUxOC0yZjM1LTRjNmEtYWE2YS03ZjcyZjQ0Yjg5ODIvMS9mNUxmZGhEd1NQUTC5RXd6YlVI
R3dSVi04T0kubWZ0MIHJBCAWJLJq5NgRwoTGQyJxWd8kyvlpHldouFIaEXTVe2eCigICCF8E
FH9W8olIyDLy1DS7RNXdQeiYBSFagIWNBgPMjAyNTA5MjMwNDAYMjlaMHYwdAYIKwYBBQUH
MAuGaHJwa2kucmlwZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxULzNhLzIxZmJjYS04MGUwLTRi
OGMtODYyMi00ZTg2YWQ2NGY3NzQvMS9mMWJ5aVvqSU12TFVOTHRFMWQ0T29TSmdHd1UubWZ0
MIHJBCAZtE8704WQcce5Xl54W0qWv/CMhndW7u/ZfNdcTsxlGICB84EFH/mixIjs9cDQwG8
lrE4quJ3hgo+AgIFBxgPMjAyNTA5MjMwNDAXMDdaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlw
ZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxULzhiLzRhMTExMS04ZjNjLTRlZGYtYjc3Yy0yZmEy
NjMxYmMzMWwvMS9mLWFMRLW1OTDF3TKRBYnlXclRpcTRuZUdDajQubWZ0MIHJBCAdR5bZgtQs
WmbdFbwYysmbREktzDzrYIAbMAd65FUGdQICB84EFH/3TavU6xqhFHFE4VsCZp6YL95NAGIG
RxxgPMjAyNTA5MjMwNTAwNTBaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcvVwb3Np
dG9yeS9ERUZBVUxULzdkLzBmYmJmNS1jODNkLTRiOTctOWNlYS0wNTVjNDlhYzY4MjgVMS9m
X2ROcTlUckdxRVVjVVRoV3dKbW5wZ3YzazAubWZ0MIHJBCAk8vdyICFceQZP/jwTVpdlm7oN
8aINdZ9yBdwal5r/wAICCBgEFH8DofjDNP2/S3je8MWS/wSQ3fSwAgIWUxgPMjAyNTA5MjMw
MDAwNDdaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxU
LzY2LzNmZTFhMC1jNmZkLTRiYzQtYWF1MS05ZWUwMDY5NDJiNGIvMS9md09oLU1NMF9iOUx1
Tjd3eFpMX0JKRGQ5TEEubWZ0MIHJBCAtzQRah0RvVm5t5cElINTCNV61ummb7bPmMHs2Q6B
qAICB84EFH9YvAhkEvTSxU+gcC2SziVJBOR5AgIUFBgPMjAyNTA5MjMwMDAwMzVaMHYwdAYI
KwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxUL2NjLzUzMDkzMy01
OTE1LTRmZjItYjFmOS01MDEwZDA1Zjk5YTgvMS9mMWk4Q0dRUz10TEZUNk1j3TFpMT0pVbHM1
SGsubWZ0MIHJBCAvHd/7VSM5LDf/9vv6NdM4wyOY6FW5vdaHrNFW4G+5agICB4QEFH8HV8Mx
mB4EK3RxnZUn0PZKEla0AgISPxxgPMjAyNTA5MjMwNDAXMzFAMHYwdAYIKwYBBQUHMAuGaHJw
a2kucmlwZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxULzc4LzRmOTZmOS01NjlmLTQzM2MtYj1h
OC02YTI2MTJkNDBmNTAvMS9md2RYd3pHWUhnUXJkSEUzTlNmUTlrb1RWclEubWZ0MIHJBCAv
uosewYqBhxMMwPdZG2Bb7HOXqIP4n3Xrkoa4ANMKtwICB84EFH9Wid5KfOdovzq12fhUboVs
yxk2AgIWmxgPMjAyNTA5MjMwMDAwNTZaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQv
cmVwb3NpdG9yeS9ERUZBVUxULzc3LzY0ZDUwNS0yMjA0LTRkY2EtYTk1ZS04YjUzNzI1ODRh
Y2QvMS9mMWFkM2twODUyaV9PclhaLUZSdWwXekxHVfKubWZ0MIHJBCAw/TNRg+KR3AigQmsm
fTMLWK6AgvtqWcYmyXrbYW2gJwICB84EFH/RimpJkQzDMdyREUrlm3GF1fMNAgIWNBgPMjAy
NTA5MjMwNDAYMjBaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcvVwb3NpdG9yeS9E
RUZBVUxUL2JhLzA5NDdmMi0yMmNhLTQ3Y2ItODlhZC0zZTUwYTVMDE5OTgvMS9mOUdLYWtt
UkRNTXgzSkVSU3VXYmNZWFY4dzAubWZ0MIHJBCA2FDd/pvoShAkRHeWEYTHP3p34AkM9F4UJ
yiPR4yK2vAICB4QEFH9za4igJATUMEvdrV/1UEpOm+e3AgIWmBgPMjAyNTA5MjMwNDAXMDFa
MHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxULzk5LzBm
MjVjZS1iOTc3LTQ4NTMtOWVjMy05ZTU2Y2I1NGZlZjcvMS9mM05yaUtBa0JOUXdtOTJ0WF9W
UVNtZ301N2MubWZ0MIHJBCA3ePsoey6JSimntoFj7+Uj2Jp7o3x9lxHSCHA+CvngQICCKUE
FH/K2J3xv5mjbykMw+8PHntNANuZAgIWDxgPMjAyNTA5MjMwNDAXMzJAMHYwdAYIKwYBBQUH
MAuGaHJwa2kucmlwZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxULzU1LzlmOGIxNi0yODRmLTQ1
MTItYjNkYy0wMTVhOWYxYjRiNTAvMS9mOHJZbmZHX21hTnZLUXpEN3c4ZWUwMENkVE0ubWZ0
MIHJBCA6w7i2aTE856qUwrqtHvcV2pGAYtPzW+ItBY72cJRyTQICB4QEFH+LIYNYmDp9eiU4
qdqbofNjTXGrAgIAwRgPMjAyNTA5MjMwMTAwMjRaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlw
ZS5uZXQvcvVwb3NpdG9yeS9ERUZBVUxUL2Q5LzcyY2QwMy1lYWEyLTQ5NjUtOTRlZi05OGRI
YjY0MDJjZGQvMS9mNHNoZzFpWU9uMTZKVGlwMnBladgwbE5jYXMubWZ0MIHJBCA78jSLajj1
eEmavbdMS0bHck7ucyNY/5Ghcb/BhvNRBQICB4QEFH8UzoEdt4XzUEvEsy8fTJtwzj9/AgIP
WBgPMjAyNTA5MjMwNDAXMDVaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcvVwb3Np
dG9yeS9ERUZBVUxUL2I2L2Y2OWRiZS1lMDclLTQ0MzgtYjUzYilMnJE2MGZhMWZiMDIvMS9m
eFRPZ1FPM2hmTlFTOFN6THg5TW0zRE9QMzgubWZ0MIHJBCA+Cnuiff/Cda/T10RjLbIbhGiA
6yw/wmdbNs/E5YV6gICB84EFH/uPzggc/LD5PzP/KOEbDNgmwmAgICbBgPMjAyNTA5MjMw

MjAxMDRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2E3LzhjY2ViZiMOWZhLTRjNDYtYWVlOC1jY2JmYTczNDI0YTcvMS9mLTrfT0NCeJhzUGtfTV84bzRURnNNMkNiQ1kubWZ0MIHJBCBC7monZzoZgZgRkbEUCIq3ToeoeEYEBRf/yR3KXgbPlwICB84EFH/JVqUrc1BKTx/zRUcZkpen9d6dAgIKTxgPMjAyNTA5MjMwNTAwNTRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2FhLzcxYWI2OS02OTY1LTRiNzAtOTY4ZC1iYjU3YTRlZjcxNTMvMS9mOGxXcFN0elVFcFBIX05GUnhtU2w2ZjEzcDAubWZ0MIHJBCBEoAA8szwjyi9uxLlSyErullXChsoH4xs3+leR8c4pgICB84EFH8Yitq1tVIIHsrIIcmwkdLlc7MVAgIRCxgPMjAyNTA5MjMwMTAwMzBaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzkwL2E2NTUyMilmN2M1LTQ4N2ItYjdiOC0yZTQ2MTQxNDZhYTQvMS9meG1LMnJXMTVnZ2V5c2doeWJDuu9VaHpzeFUubWZ0MIHJBCBH59+lgS6pkImGH3tVdnqkYpyPqpt5h0mEJaJjiaVlJAICB84EFH+F6ZA1Q5fjbAypA6DGIMdwnv3NAGICyhgPMjAyNTA5MjMwNjAwNDFaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzVlLzRhZTE3NS01NWQwLTQ4NGQtOGQxMS04YzlkNTgyM2JhZDkvMS9mNFhwa0RWRGwtTnNES2tEb01ZZ3gzQ2VfYzAubWZ0MIHJBCBIO5YvqeKZhIwAq5xNWIHFQYXDGHX++Snc9OR3VklQICB4QEFH+ZGMq28fNe8bXlg8soOem3SLFYAgITcBgPMjAyNTA5MjMwNDAXMDhaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzI2L2RmODJmYS0xMThkLTQyMTktYmExNi0xYTtk2ODNjOWQ2Y2IvMS9mNWtZeXJieDgxN3h0ZVDeElnNTZiZElzVmcubWZ0MIHJBCBUcrrYoOmnp+lr0dGSwp10yTIOji2xK0XGVtH/jO9z8QICB84EFH93NN/qEgZXQS6oZ928e4TRMr94AgIPSRgPMjAyNTA5MjMwNDAYMDRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzdlLzY4MDMyNC1lZTFmLTQwZjItODhkZi0xOTY5MzE5NjJkM2MvMS9mM2MwMy1vU0JsZEJMcWhuM2J4N2hORXl2M2cubWZ0MIHJBCBk8ncmsVu05KJS64WUkha4F/2Z6hhjsHEvOjGWkevNmwICB4QEFH9K5X7m4KnFEB/BSOelM0FQu6tGAgIF6hgPMjAyNTA5MjIyMzAwNTlaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzJiLzExZmE0NC04ODc5LTrkMTktYjQ2OC01YjVkoThmNTM2MWUvMS9mMHJsZnViZ3FjVVFIOEZLaDZVelFWQzdxMFkubWZ0MIHJBCBLoCeMud+A0BSpmrfdU2mQjsCbeBScISC/gPsgvPsQ0wICB84EFH9C0nwh2obH6fv0SuDlbJjz0vgLagIOORgPMjAyNTA5MjMwNDAYMjNaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2E5LzkyYjAyNC05ZDhmLTQ4NjktYTJhOS0wYWZiN2MzZmJmMzAvMS9mMEXtZkNIYWWhZnAtX1JLNE9Wc2lQUFMTQXMubWZ0MIHJBCBw8y/EDCcYMb7gP8H5htkrZe4PVZuH0MnAfrIjELHbvaICB84EFH81la5Bf0XuhQXXbOqhs0xFg5SgAgIPdhgPMjAyNTA5MjMwNDAYMDRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2U2L2M1NDY3Yi0zOTYyLTRiNzQtYWUwZC0zNDQ3M2JjOTFkODAvMS9melhWcmTGXlJlNkZCZGRzNnFHelRFV0RsS0EubWZ0MIHJBCBxZXk3VbhMC3pspd6F9s5+5rzXicGmDS87bxAXcC5v9AICB84EFH/8glhd9pt8lLFqYZMv/ItuM1FWAgIScxgPMjAyNTA5MjMwNDAXMTRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2U0LzUzMWQyYi1lMTcwLTQ5YTUtODQwNC05MmY3M2Y1NmZjNjIvMS9mX3lDV0Yzmm0zeVVzV3Boa3lfoGkyNHpVvlkubWZ0MIHJBCB0NGwcUNfXmz+slvqbdJuuAJNSASxeJtVG+WIScDUZEgICCBgEFH/g51kl1ToPMGTIDgRCd4i2g8acAAgIWoRgPMjAyNTA5MjMwMTAwNTRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzRjLzBjNzI0My0xZmZiLTQ5ZDItyjYjMS1iNDAYZThhMWQ5MzQvMS9mLURuVlRWT2c4dlpNZ09CRUozaUxhRHhwD0EubWZ0MIHJBCB0tjF6xFCQL8EVIImZnEluq5kDbLBDCWemir9s7lRBERwICCKUEFH9RXq0JXu2axMq8WrStC1hn2fAbAgIWsRgPMjAyNTA5MjMwNjAxMjBaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2E3LzVjMmE1OS02MDI1LTQwMGUtYWIyOC1mGE2MjRkNDAYMTivMS9mMUZlclFsZTdackV5cnhhdeSwTFdHZlo4QnMubWZ0MIHJBCB3xsZEFIEGsX7ixl02aTWQSKjQ7xeNF/o+rvPRBN0+MAICB84EFH/j1jtKW0BLX/g8vysVJaMEd/ZcAgID3RgPMjAyNTA5MjMwNjAwNTRAmHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2Y2LzgxYTY3MilmZTk4LTrkM2YtOWI4My0yY2I3YTNI NmU0MmYvMS9mLVBXTzBwYlFFdGYtRHlfs3hVbG93UjM5bHcubWZ0MIHJBCB4jbeXyKKmxw2qL7qN

knQE9sC8Mbp3PB65S4tWNTn7YgICCF8EFH8+Cye45NeY+Sud4Vfx2lpDzUnlAgIQ2BgPMjAy
NTA5MjMwNzAwNDFaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9E
RUZBVUxULzVmlL2EwYzlhYy0zYTQ3LTrkNmMtYWEhNS1hNDJlYzg3NzZmYmIvMS9mejRMSjdq
azElaJVLNTNoVl9iYVdrUE5TZVUubWZ0MIHJBCCB6I7Br1qhj6mFvJAZLnxbwXjsrWAGCTEbu
MY5nuQrj+gICB4QEFH8pZlevjQDCX9dfVuzIoos1FQVlAgIP0RgPMjAyNTA5MjMwNzAxMDNa
MHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxUL2QwL2U4
MGIzMy0zYmVlLTQ2ZWmtYjMlZC1iYWY5NWE1MDZkMTkvMS9meWxtVjYtTkFNSmYxMTlXN01p
aWl6VVZCWFUubWZ0MIHJBCCDkqyKnrs6AkI2WpmX5t8Dwi55XFHtP0mT5xiZiIGFcQICB84E
FH8Xj69KaELzcW4xdKvP33Md9Y8iAgIRDxgPMjAyNTA5MjMwNjAwNDlaMHYwdAYIKwYBBQUH
MAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxULzY2LzFiNzk2OC1jYTRkLTQ1
ZjQtYjY3MS0yZTdmNzg0ODl jZDMvMS9meGVQc jJRQjR2TnhiakYyUlduZmN4MzFqeUkubWZ0
MIHJBCCFDDrZBUzNkumvliEbQSPHMBB2WbYuFwD4ZYxe7f2+ogICCBgEFH8WgCjsDatmimfV
v29TWMqr4zeoAgILChgPMjAyNTA5MjMwMDAxMDVaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlw
ZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxUL2NjL2IxNTI4Ni1mZDRkLTQ5ZmUtYTY5ZS03ZmFk
ZjUwYTJlMzcwMS9meGFBS093TnEyYUtaOVdfYjYFOwXlxdmpONmcubWZ0MIHJBCCCKsbiYxCKs
Witft9wl//KbqqkzkoXS5g++SNBP01bHwICB84EFH8dWNYt3X5HryGW/XVLS/8meYkqAgIW
mRgPMjAyNTA5MjMwNDAXNTlaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3Np
dG9yeS9ERUZBVUxULzQ2LzI5MGE2MC03OTJmLTQ0NzUtYTlmlNC1lM2I5ZTBiYWU2YWlVMS9m
eDFZMWkzZGZrZXZJWmI5ZFVlel95WjVpU28ubWZ0MIHJBCCOXbx/KCDd6GY3H2VwggZj6ils
X1nuORBcFThWKwFRlWICB84EFH8RQWrjkp3ze/ZqRZzTrKY4kelGAgIWLxgPMjAyNTA5MjMw
MDAwMzBaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxU
LzJmLzZiMjJlOC0zZGNkLTQ0ZGQtOTUxNC02NzU2ZjdiMGwZGivMS9meEZCYXVPU25mTjc5
bXBGbk5Pc3BqaVI2VVkubWZ0MIHJBCCRYcPVPeYtsaCCsUTfyqBHbPAAwsIA/BFmYeDB5Isx
dgICB84EFH/WLkLAjhYBxFceDYi jSaBQnepeAgISixgPMjAyNTA5MjMwMDAwNTlaMHYwdAYI
KwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxULzC4L2I5MzRlZS1j
NGJhLTQzOTETODBiMCl1NzVhNWVhODlmZDEvMS9mOV1lUXNDT0ZnSEVWeDR0aUtOSm9GQ2Q2
bdQubWZ0MIHJBCCCTyRLdagxVGyugDu6T6ET54lM06pyPjve5S27+B9xA/QICB84EFH8z/EDS
4DM7vHveqyvYWZVDACDxAgIF5BgPMjAyNTA5MjIyMzAwMzVaMHYwdAYIKwYBBQUHMAuGaHJw
a2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxUL2I0LzY1YmYxZC1lYzFkLTQwZTYtODQ5
ZC03OTc5MGQ2NmQ3ZDMvMS9mela4UU5Mz016dThlOTZySzloWmxVTUJ3UEUubWZ0MIHJBCCU
2IRpBytL/B62SPheKly8NuGHwWPbU8iyvkhFLuXp3gICCb0EFH9r0aawRiXFcdgw+HixwCOC
R0CMAgIFPxpPMjAyNTA5MjMwNzAwNDRaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQv
cmVwb3NpdG9yeS9ERUZBVUxULzQ4L2J jNjZkNy01N2FiLTQ3NWQtOTZiYS04OWI2YzMyMzE1
YzIvMS9mMnZSchJCR0pjVngyREQ0ZUxIQUK0SkhRSXcubWZ0MIHJBCCYquVwfhlYmcmkDOxQ
2HCsPWT69GGOnzJ/g90EwQ8rIgICB84EFH8+CHSJ7iOpQk1T+sIW7kuOASgOAgIWLhgPMjAy
NTA5MjIyMzAwNDZaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9E
RUZBVUxUL2M2L2MyYzkh5ZC1 jZ jkxLTRkZmItOTRkZS1hN2M2M2Q1NjJlNTYvMS9mejRjZElu
dUk2bENUVlA2d2hidVM0NEJLQTQubWZ0MIHJBCCfyEBs3iVYfhQ3nM9w6BpR5fISnWVCBV62
cPxv9KUEQICB84EFH/7ARXErN3RtK3EBzeAcmrPwBr5AgILpRgPMjAyNTA5MjMwNjAxMDZa
MHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxULzYxLzUy
NjhmYy1 jZmVlLTRhYTgtOTdiZi1 jYmUwMjA2NjVlZmUvMS9mX3NCRmNTczNkRzByYlFITjRC
eWFzX0FHdmsubWZ0MIHJBCCn6U5KB9Ulxw7GIPFbaOmCIDKf8lpmKqI33H2DHZkZsgICB84E
FH9RIoN0dC3lRKqTBYxa09PRZCGZAgIOBgPMjAyNTA5MjMwNTAwNDlaMHYwdAYIKwYBBQUH
MAuGaHJwa2kucmlwZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxULzBhLzkhZkZODVhYS0xYjc5LTRh
MDItYTA5Mi0wMWViMDM2ODRmMDkvMS9mMUVpZzNSMExmVkcE1GakZvNzA5RmtJWmsubWZ0
MIHJBCCpi80UrfSzwVwcehboPdQHhrh6h9xGsVXEPoezAH0XwICB84EFH9qj1lVwkmKgmNv
mfj8njGeB3ceAgIPnBgPMjAyNTA5MjIyMzAwNDFaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlw
ZS5uZXQvcmlwV3NpdG9yeS9ERUZBVUxULzC0LzI3ZmNlMS0xMWNlLTlhMzItYTY0OS1lMDc2

YjUxNzIxYWQvMS9mMnFPWFZYQ1NZcUNZMilaLVB5ZU1aNEhkeDQubWZ0MIHJBCCsI4WNffQO
MibfaZfwiXVbi3HpxPKq0qlzWTjOSZ0K6AICB84EFH8qMrpCGWgNzdWPYQHlx67BVApzAgIW
nBgPMjAyNTA5MjMwNDAYMTZaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3Np
dG9yeS9ERUZBVUxULzU1LzE0M2U5ZS05NmU2LTQ3NGUtYWQyYy0yMmU2ZGY0NTg0YWYvMS9m
eW95dWtJWmFBM04xWTloQWVYSHJzRlVDbk0ubWZ0MIHJBCCv8OQ2OEccYyrWLhsQ9GBsMkL+
U1qihaQnp4htBZJ7xAICCF8EFH9QB30t2KZ6Gui2q9a7s0iQKKW7AgIWohgPMjAyNTA5MjMw
MjAxMDJAMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxU
LzVjL2MxYzFjZS1lYTU5LTRkY2YtYmNjYy0zZTdjYWRkODhjNzAvMS9mMUFIZlMzWXBub2E2
TGFyMXJlelNKQW9wYnMubWZ0MIHJBCCwbYH58Q1Hr3LbAk6xc1G6KS2L4M6cMSMGeGSmORmj
0wICB84EFH+0PeI3/QtqKHOJiWkh0losLtGoAgIWLrgPMjAyNTA5MjMwMjAwNDFAmHYwdAYI
KwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzA5L2Q0ZDEyYS1h
YjRlLTRkYmEtOTVkJZS1iYzYzNzEzMGRLNmUvMS9mN1E5NGpmOUMyb29jNGtqQ1NIU1dpd3Uw
YWcubWZ0MIHJBCC5qxrhGsdwpb32aH8xtHnZZzWsxXc/uTEJLN0VxpfIVKAICCBgEFH/xuBz+
arsRjpfGm1tmLq+YTy9qAgINBxgPMjAyNTA5MjMwMDAwNTRaMHYwdAYIKwYBBQUHMAuGaHJw
a2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzclLzE4NmExOC01ZDdmLTQzZWQtYjA2
YS1jZWE3ZWlZNTA1MzcwMS9mX0c0SFA1cXV4R09sLUF5VzJZdXI1aFBMMm8ubWZ0MIHJBCC6
bbXg9NGOUiRUF704Kkx7JJK2UGvi48vcupkrHBNoQICB84EFH8xNg/8GvlfHaZtgUBORmNR
LULnAgIUJBGPMjAyNTA5MjMwMjAwNDJAMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQv
cmVWb3NpdG9yeS9ERUZBVUxULzJkL2ZlZjVkJZC0zOGVlLTRiYzUtODJmZi01ODRkNzhhmjVm
OGMvMS9mekUyRF93YV9WOGRwbTJCUU1R1kxRXRTV2MubWZ0MIHJBCCDFP/VzX+fQNa0KUtt
8P0j6leHwFvSwrx0LHWDrdI3oAICB4QEFH9R3x306IZ+QeXn+S3n+dH6DBVNAGILgRgPMjAy
NTA5MjMwNjAxMTRaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9E
RUZBVUxUL2ZhlZVlNzMXZi01MjhiLTRlMDItYWQ2OC02ZDkwMzVhZjE1MzUvMS9mMUhmSGZU
b2huNUI1ZWY1TGVMNTBmb01GVTAubWZ0MIHJBCCDNGMH6y/iAIm2VqfNabCpnc1gk0F5FdiQx
b45atOgcBwICB84EFH+bTP3JsNnJwx4Ou4Hm8bHLvcnkAgIMexgPMjAyNTA5MjMwMjAwMzRa
MHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzViLzgy
NDZlYi04NDY5LTQyZGYtYTA4Zi05MGm0MjFmNTVhZGIVMS9mNXRXNX2NtdzJlUERIZzY3Z2Vi
eHNjdTl5ZVEubWZ0MIHJBCCDULZ9V9GVvzrX6km+uZVhinNo5qZQvNZyoPq6ckfF4AwICB84E
FH8dDjKYvtOn85+zskTtkYv2xNe/AgIUcRgPMjAyNTA5MjIyMzAwNDZaMHYwdAYIKwYBBQUH
MAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxUL2Y3L2U2NTA2ZS03NjglLTQ4
ZTctYTU4My0yMWFmM2RlZThlZTkVMS9meDBPTXBpLTA2ZnpuN095Uk8yUmlfYkUxNzgubWZ0
MIHJBCCDwykLsOqpzcVfKvfc4KEZhaQ5iW2klfJdHpTsrQ5klwICB4QEFH9+Xr1vpGnF43C/
wQbElKrR4duUAGIPBBgPMjAyNTA5MjMwNjAwMjNaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlw
ZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzIzL2MzZThlZC01OTdhLTQ4NWetOTRhMy05ODJj
ZmIzNTlhZWUvMS9mMzVldlcta2FjWGPjTf9CQnNUVXF0SGgyNVEubWZ0MIHJBCCDi1bm+jh+B
MioiQZihpFhx0f5+snIoLoL7iYheLRGulaICB4QEFH/xeuVHufJkHmBXf+VT3bb3SaCHAgID
2hgPMjAyNTA5MjMwNDAXMDlaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3Np
dG9yeS9ERUZBVUxUL2ElLzFiYzBmOS1iZDdjLTRmN2EtYTQwNC1hNDNmZjI1ZDQ0N2QvMS9m
X0Y2NVVlNThtUWVZRMrfNVZQZHR2ZEpvSWMubWZ0MIHJBCCDrroFYFly0MZks0keOew0RivBM
lIz4sQ5J3U+8VQ+ioAICB84EFH9NWRwrjSxpr31/eh1M6Ov06VlsAgIE9RgPMjAyNTA5MjMw
MjAwNDBaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxU
LzNlLzBkNjdhMiliYzZmL2RiOTMtYjYwNC1kNGY1YzkyZDkwOTUvMS9mMDFaSENPdExHbEhm
WDk2SFV6bzY4N3BXV3cubWZ0MIHJBCCDrySEF0Mb4O38e4Yom6HOu56DkNiYSZsnLEu0z9ENq
kwICB84EFH9DnZQkYjvxnOyecyYqzX9vX6pf1AgISMxgPMjAyNTA5MjIyMzAwNDBaMHYwdAYI
KwYBBQUHMAuGaHJwa2kucmlwZS5uZXQvcmlwVWb3NpdG9yeS9ERUZBVUxULzJhLzY5YTA5Zi01
ODA5LTQ2NTYtYjM1NS0xNDZmMjU0YWMxMzEvMS9mME9kbENRbV9HYzdKNXpKaXJOZjI5ZnFs
X1UubWZ0MIHJBCCDwigCSeW8JjL525MGaphNOYxotjA+2y7/nxhBX30rm0QICB4QEFH8Xtvky
OZ1YUJPRhzOU6/4bKfmMAGIOHBgPMjAyNTA5MjMwMTAwNDJAMHYwdAYIKwYBBQUHMAuGaHJw

a2kucmlwZS5uZXQvcmlVwb3NpdG9yeS9ERUZBVUxULzMxL2NjMDIwMC1iYjJjLTQ1ZjYtYWM3
NS04Mjc1NzdkZjhlZGMvMS9meGUyLVRJNW5WaffrOUdITTVUcl9oc3AtWXcubWZ0MIHJBCD2
OipMOZLdHyqDhwoMxCagn3UjHiGolihS07IdefH2zwICB84EFH8km5VEYgaD+Us4inVRpopk
k+0SAgICyXgPMjAyNTA5MjMwMTAwNDdaMHYwdAYIKwYBBQUHMAuGaHJwa2kucmlwZS5uZXQv
cmVwb3NpdG9yeS9ERUZBVUxULzhiLzdhdhYTA0ZS00ODA3LTQ5ODgtOTFwMy04NDIzOTdlMzA2
NDMvMS9meVNibFVSaUJvUDVTemlLZGFZbWltU1Q3UkkubWZ0

Acknowledgements

The authors wish to thank George Michaelson, Theo de Raadt, Bob Beck, Theo Buehler, and William McCall for the lovely conversations that led to this proposal.

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