

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: June 16, 2021

A. Przygienda  
Juniper  
December 13, 2020

RIFT Keys Structure and Well-Known Registry in Key Value TIE  
draft-przygienda-rift-kv-registry-00

## Abstract

This document describes key structure of keys contained within RIFT key-value TIEs.

## Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

## 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 June 16, 2021.

## Copyright Notice

Copyright (c) 2020 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

Internet-Draft

[draft-ietf-przygienda-kv-registry](#)

December 2020

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Description . . . . .	<a href="#">2</a>
<a href="#">2.</a>	Key Type Registry . . . . .	<a href="#">2</a>
<a href="#">3.</a>	OUI Key Type . . . . .	<a href="#">2</a>
<a href="#">4.</a>	Well-Known Key Type . . . . .	<a href="#">3</a>
<a href="#">5.</a>	IANA Considerations . . . . .	<a href="#">3</a>
<a href="#">5.1.</a>	Key Type Registry . . . . .	<a href="#">3</a>
<a href="#">5.1.1.</a>	Requested Entries . . . . .	<a href="#">3</a>
<a href="#">5.2.</a>	Well-Known Key Type Registry . . . . .	<a href="#">4</a>
<a href="#">5.2.1.</a>	Requested Entries . . . . .	<a href="#">4</a>
<a href="#">6.</a>	Security Considerations . . . . .	<a href="#">4</a>
<a href="#">7.</a>	Acknowledgements . . . . .	<a href="#">4</a>
<a href="#">8.</a>	Normative References . . . . .	<a href="#">4</a>
	Author's Address . . . . .	<a href="#">4</a>

## [1.](#) Description

[I-D.ietf-rift-rift] specifies a topology information element (TIE) that can carry unstructured key value pairs of data. This document defines a registry for the keys to allow for vendor specific values being carried without risking a collision with future standardized values.

This document specifies also several well-known keys and their values including a registry to allow for easier interoperability between implementations.

## [2.](#) Key Type Registry

The first octet of every key is a value from a "RIFT Key Types Registry" which may specify the according key structure further.

## [3.](#) OUI Key Type

This section reserves a key type value to indicate a vendor specific key with further indication via Organizationally Unique Identifier (OUI) which organization the key belongs to. The value of that first octet is TBD1. The structure of the key is indicated in the figure

Figure 1.

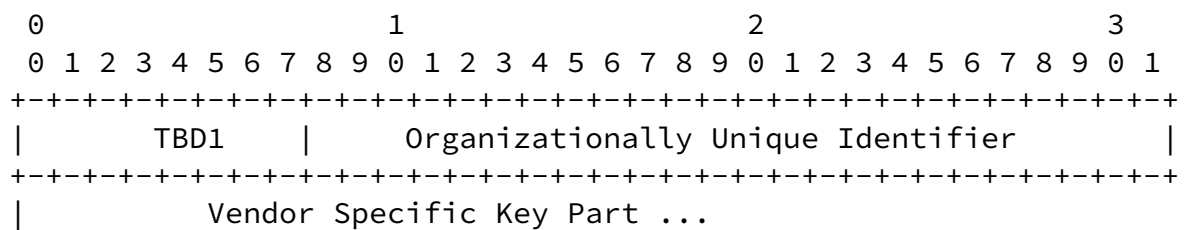


Figure 1: OUI Key

#### 4. Well-Known Key Type

This section reserves a key type value in Key Type Registry to indicate a well-known key that all implementations SHOULD support. The type is followed by a 3 octets value from a "RIFT Well-Known Key" Registry describing the structure of the value it carries. The resulting structure of the key is provided in Figure 2.

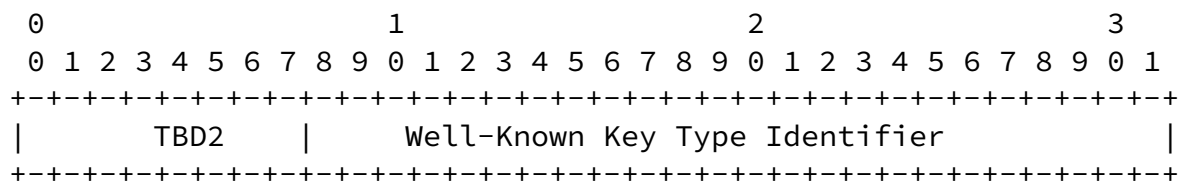


Figure 2: Well-Known Key

#### 5. IANA Considerations

This section requests registries that help govern RIFT key values via usual IANA registry procedures. All values not suggested as to be considered available for assignment. Allocation of new values is always performed via `Expert Review` action.

## [5.1.](#) Key Type Registry

This is registry for key types for keys contained in RIFT KV TIEs. The range of values is 0 .. [255](#)

### [5.1.1.](#) Requested Entries

Name	Value	Description
OUI	TBD1	Followed by 3 octets of OUI.
Well-Known Key	TBD2	Followed by 3 octets value from Well-Known Key Type Registry.

Przygienda

Expires June 16, 2021

[Page 3]

---

Internet-Draft

[draft-ietf-przygienda-kv-registry](#)

December 2020

## [5.2.](#) Well-Known Key Type Registry

This is registry for key types for well-known keys contained in RIFT KV TIEs. The range of values is 0 .. 2<sup>24</sup>-1

### [5.2.1.](#) Requested Entries

Name	Value	Description
Illegal	0	Not allowed.
MAC/IP Binding	TBD2	To be defined.
FAM Security Roll-Over Key	TBD2	To be defined.

## [6.](#) Security Considerations

This document introduces no new security concerns to RIFT or other specifications referenced in this document given that key value pairs are carried in TIEs that are already extensively secured by RIFT specification itself.

## [7.](#) Acknowledgements

To be provided.

## [8.](#) Normative References

[I-D.ietf-rift-rift]

Przygienda, T., Sharma, A., Thubert, P., Rijsman, B., and D. Afanasiev, "RIFT: Routing in Fat Trees", [draft-ietf-rift-rift-12](#) (work in progress), May 2020.

[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>>.

Author's Address

Tony Przygienda  
Juniper  
1137 Innovation Way

Sunnyvale, CA

USA

Email: [prz@juniper.net](mailto:prz@juniper.net)