Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 4, 2015

A. Newton
ARIN
J. Aehlen
RIPE NCC
C. Martinez
LACNIC
J. Snijders
Independent

July 3, 2014

Route Policy Extensions for RDAP
draft-newton-weirds-route-policy-00

Abstract

This document describes extensions to the RDAP JSON data model to express route policy.

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 http://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 January 4, 2015.

1. Introduction

The base JSON data model for RDAP contains objects for domain name registries (DNRs) and Regional Internet Registries (RIRs). It does not contain objects for Internet Routing Registries (IRRs).

This document describes extensions to the RDAP data model to express route policy.

2. RDAP Compliance

Servers expressing route policy extensions SHOULD add the following string to rdapConformance array: "route_policy_level_0". The following is an example:

An example rdapConformance data structure:

```
"rdapConformance" :
[
  "rdap_level_0",
  "route_policy_level_0"
]
```

Figure 1

JSON names for extensions in RDAP should be prefixed with an extension identifier. The extension identifier used by this extension is "rp0".

This document defines one extenion to the RDAP autnum object class ("rp0_policies") and three new RDAP object classes which map to RPSL classes: route, route set, and autnum set. The RDAP equivalent of an RPSL maintainer object is the entity object class. Each of object classes defined by this document may have the links, entities, and other common data structures defined by RDAP.

3. Autnum Route Policies

The rp0_policies is an array containing objects. Each object in the array contains one member, the value of which is a string containing routing policy. Use of the rp0_policies array is as an optional member to the RDAP autnum object class.

The name of the object member MUST be one of the following names:

import

mp-import

import-via

export

mp-export

export-via

An example rp0_policies data structure:

```
"rp0_policies":
[
  { "import-via":
     "AS6777 from AS-ANY EXCEPT (AS1103 AND AS1103) accept ANY" },
  { "export-via":
     "AS6777 to AS-ANY EXCEPT (AS1 AND AS1103) announce AS-SNIJDERS" }
]
```

Figure 2

As rp0_policies is an array, the order of the objects in the array is to be observed during processing.

The following is an example of a JSON object representing an autnum with route policy extensions. For illustrative purposes, it does not include rdapConformance or notices data structures.

```
{
  "handle" : "XXXX-RIR",
  "startAutnum" : 10,
  "endAutnum" : 15,
  "name": "AS-RTR-1",
  "type" : "DIRECT ALLOCATION",
  "status" : [ "allocated" ],
  "country": "AU",
  "remarks" :
  [
    {
      "description" :
      [
        "She sells sea shells down by the sea shore.",
        "Originally written by Terry Sullivan."
      ]
```

```
    }
  ],
  "links" :
  [
    {
      "value" : "http://example.net/autnum/xxxx",
      "rel" : "self",
      "href" : "http://example.net/autnum/xxxx",
      "type" : "application/rdap+json"
    }
  ],
  "events" :
  [
    {
      "eventAction" : "registration",
      "eventDate" : "1990-12-31T23:59:60Z"
    },
    {
      "eventAction" : "last changed",
      "eventDate" : "1991-12-31T23:59:60Z"
    }
  ],
  "entities" :
  [
    {
      "handle" : "XXXX",
      "vcardArray":[
        "vcard",
        [
          ["version", {}, "text", "4.0"],
          ["fn", {}, "text", "Joe User"],
          ["kind", {}, "text", "individual"],
          ["lang", {
            "pref":"1"
          }, "language-tag", "fr"],
          ["lang", {
            "pref":"2"
          }, "language-tag", "en"],
          ["org", {
            "type":"work"
          }, "text", "Example"],
          ["title", {}, "text", "Research Scientist"],
          ["role", {}, "text", "Project Lead"],
          ["adr",
            { "type":"work" },
            "text",
            [
              "",
```

```
              "Suite 1234",
              "4321 Rue Somewhere",
              "Quebec",
              "QC",
              "G1V 2M2",
              "Canada"
            ]
          ],
          ["tel",
            { "type":["work", "voice"], "pref":"1" },
            "uri", "tel:+1-555-555-1234;ext=102"
          ],
          ["email",
            { "type":"work" },
            "text", "joe.user@example.com"
          ],
        ]
      ],
      "roles" : [ "registrant" ],
      "remarks" :
      [
        {
          "description" :
          [
            "She sells sea shells down by the sea shore.",
            "Originally written by Terry Sullivan."
          ]
        }
      ],
      "links" :
      [
        {
          "value" : "http://example.net/entity/XXXX",
          "rel" : "self",
          "href" : "http://example.net/entity/XXXX",
          "type" : "application/rdap+json"
        }
      ],
      "events" :
      [
        {
          "eventAction" : "registration",
          "eventDate" : "1990-12-31T23:59:60Z"
        },
        {
          "eventAction" : "last changed",
          "eventDate" : "1991-12-31T23:59:60Z"
        }
```

```
      ]
    }
  ],
  "rp0_policies":
  [
    { "import-via":
       "AS6777 from AS-ANY EXCEPT (AS1103 AND AS1103) accept ANY" },
    { "export-via":
       "AS6777 to AS-ANY EXCEPT (AS1 AND AS1103) announce AS-SNIJDERS" }
  ]
}
```

4. Route Object Class

The RDAP route object class is an RDAP representation of the RPSL route class.

The following is an elided example of a route object showing the high level structure:

```
{
  "handle" : "XXX",
  "route" : "XXX",
  "origin" : 123,
  ...
  "entities" :
  [
    ...
  ],
  "links" :
  [
    ...
  ],
  ...
}
```

The "handle" member is the registry unique identifier of the route object, just as with other RDAP object classes. The "route" member is the IP address prefix as specified in RPSL. The "handle" and "route" values may be the same (and usually will be).

The "origin" member is an integer specifying an autonomous system number.

The other members of the object are:

memberOf - an array of strings, each containing the handle of a route set object.

inject - an array of strings, each containing a value as specified by RPSL.

components - a string containing a value as specified by RPSL.

aggregateBoundary - a string containing a value as specified by RPSL.

aggregateMtd - a string containing a value as specified by RPSL.

exportComps - a string containing a value as specified by RPSL.

holes - an array of strings, each containing a value as specified by RPSL.

entities - an array of entity objects, as specified by RDAP.

While the "memberOf" array contains route set object handles, the links array (as specified by RDAP) SHOULD contain links to each route set object using the "collection" link relationship.

Route objects are obtained from an RDAP server by appending the "/route" path to an RDAP base URL followed by either the handle of the route object or value of the "route" member.

5. Route Set Object Class

The route set object class is an RDAP representation of the route-set object in RPSL.

The following is an elided example of a route set object showing the high level structure:

```
{
  "handle" : "XXX",
  "members" :
  [
     ...
  ],
  ...
  "entities" :
  [
    ...
  ],
  "links" :
  [
    ...
  ],
  ...
}
```

The "handle" member is the registry unique identifier of the route set object, just as with other RDAP object classes. The "members" object member is an array of strings, each containing the handle of a route or route set object.

While the "members" array contains route or route set object handles, the links array (as specified by RDAP) SHOULD contain links to each route or route set object using the "item" link relationship.

Route set objects are obtained from an RDAP server by appending the "/routeSet" path to an RDAP base URL followed by the handle of the route set object.

6. Autnum Set Object Class

The autnum set object class is an RDAP representation of the as-set object in RPSL.

The following is an elided example of an autnum set object showing the high level structure:

```
{
  "handle" : "XXX",
  "members" :
  [
     ...
  ],
  ...
  "entities" :
  [
    ...
  ],
  "links" :
  [
    ...
  ],
  ...
}
```

The "handle" member is the registry unique identifier of the autnum set object, just as with other RDAP object classes. The "members" object member is an array of strings, each containing the handle of an autnum object (as defined by RDAP) or autnum set object.

While the "members" array contains autnum or autnum set object handles, the links array (as specified by RDAP) SHOULD contain links to each autnum or autnum set object using the "item" link relationship.

Autnum set objects are obtained from an RDAP server by appending the "/autnumSet" path to an RDAP base URL followed by the handle of the autnum set object.

7. Discussion

7.1. Other IRR Objects

RDAP contains many objects that are found in IRRs, but not all of them. The common object types are autnum objects and entity objects (for the mnter class of objects). RDAP does not have equivalences for route or set objects, and RPSL does not have equivalences for nameserver, domain, or network objects (though such objects do appear in registries that are both RIRs and IRRs).

There has been debate in the community over the usefulness of the entire RPSL data model. Therefore, this document starts with replicating only the parts of RPSL needed to express import/export policy for an autonomous system.

7.2. Decomposition of RPSL to JSON

Since RPSL is specified via language grammar, it requires an RPSL specific parser. This has been among one of its chief criticisms. RDAP uses JSON, so it requires no RDAP specific parser.

As the route policy community is not very large and is mostly composed of individuals who are not computer programmers as their primary tasking, the issue of a problem domain parser for route policy is important. But using a common data format such as JSON, network operators do not need to create a route policy specific parser to use the data. While seemingly trivial to some, it is an important consideration for such a small community.

This document does not attempt to map RPSL into JSON, instead leaving the policies as RPSL strings. This only solves the "lookup" problem, where RDAP can be consulted to get the policies. It does not solve the "parser" problem, where a special purpose RPSL parser is still needed.

Future versions of this document may decompose RPSL into JSON once a reasonable JSON serialization technique can be determined.

Authors' Addresses

Andrew Lee Newton
American Registry for Internet Numbers
3635 Concorde Parkway
Chantilly, VA 20151
US

Email: andy@arin.net
URI: http://www.arin.net

Johan Aehlen
RIPE Network Coordination Centre
Singel 258
Amsterdam 1016AB
NL

Email: jahlen@ripe.net
URI: http://www.ripe.net

Carlos M. Martinez
Latin American and Caribbean Internet Address Registry
Rambla Republica de Mexico 6125
Montevideo 11300
UY

Email: carlos@lacnic.net
URI: http://www.lacnic.net

Job Snijders
Independent
Theodorus Majofskistraat 100
Amsterdam 1065 SZ
NL

Email: job@instituut.net