

P2PSIP  
Internet-Draft  
Intended status: Standards Track  
Expires: January 5, 2012

J. Rosenberg  
jdrosen.net  
C. Jennings  
Cisco  
M. Petit-Huguenin  
Stonyfish  
July 4, 2011

Proportional Quota in REsource LOcation And Discovery (RELOAD)  
draft-petithuguenin-p2psip-proportional-quota-01

Abstract

This document defines an extension to RELOAD [I-D.ietf-p2psip-base] that limits the number of a specific kind element that can be stored by one RELOAD peer.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, except to format it for publication as an RFC or to translate it into languages other than English.

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 5, 2012.

Copyright Notice

Copyright (c) 2011 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 (<http://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 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

- 1. Introduction . . . . . 3
- 2. Terminology . . . . . 3
- 3. Quota Algorithm . . . . . 3
- 4. Overlay Configuration Document Extension . . . . . 4
- 5. Security Considerations . . . . . 5
- 6. IANA Considerations . . . . . 5
- 7. Acknowledgements . . . . . 5
- 8. References . . . . . 5
  - 8.1. Normative References . . . . . 5
  - 8.2. Informative References . . . . . 5
- Appendix A. Release notes . . . . . 5
  - A.1. Modifications between  
draft-petithuguenin-p2psip-proportional-quota-01 and  
draft-petithuguenin-p2psip-proportional-quota-00 . . . . . 5
  - A.2. Modifications between  
draft-petithuguenin-p2psip-proportional-quota-00 and  
draft-petithuguenin-vipr-reload-usage-00 . . . . . 6
  - A.3. Running Code Considerations . . . . . 6
- Authors' Addresses . . . . . 6

## 1. Introduction

The base specification of RELOAD defines two variables to set the storage quota. The first one is the maximum size of an element of a specific kind, the second one is the maximum number of elements of a specific kind that can be stored on a specific peer. The combination of the two variables permits to limit the quantity of data that a peer have to store.

For kinds that are used with an access control policy that does not restrict the Resource-IDs, a different algorithm is needed to force storing nodes to behave. This document describes a quota algorithm that limits the number of elements of a specific kind that a node can store in the overlay, independently of the Resource-ID used. Another way to look at this quota algorithm is that an entity must provide a number of peers that is proportional to the number of elements of a specific kind that this entity wants to store.

It is important to understand that this quota works only for an overlay comprising only peers, so with a configuration file containing a <clients-permitted> element set to false.

## 2. Terminology

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 [RFC2119].

## 3. Quota Algorithm

A peer responsible for storing kinds using the quota algorithm described in this document MUST maintain a count of the number of unique entries being stored per signer for each kind. This operation does not require to add a field containing the Node-ID as the Node-ID of the signer is always available in the signature field of each element stored.

For example if a peer is storing 5 Resource-IDs and at each of those 5 there are two keys whose first 16 bytes correspond to a particular Node-ID, it means this node is currently storing 10 unique dictionary entries for that Node-ID.

When performing the quota checks for an entry, the peer starts by verifying that the size of the entry is consistent. It then takes the <max-count> configuration parameter for this overlay, which measures the amount of entries of a specific kind a particular node

can store when a `<max-count-per>SIGNER</max-count-per>` configuration parameter is also present. That value is multiplied by the number of replicas used by the topology plugin (i.e. 3 for Chord) and then divided by the fraction of the overlay owned by this peer. If the result is less than one, it is rounded up to two. This is the maximum number of unique entries that can be stored for this signer. If the peer is not yet storing this many entries for that Node-ID, the store is allowed.

Note that when evaluating a Store Request containing multiple entries per kind, the count of unique entries used for the evaluation is incremented after each successful check, but the count will be reset to its initial value if one of the check fails.

The algorithm takes in account only the duplications made by the topology plugin. If another level of duplication is done at the application level, the `<max-count>` value must be adjusted accordingly.

Note that because of imperfect distribution of Resource-IDs across the ring, new entries can be rejected even if the total count is under the limit. It is the responsibility of the storing entity to calculate the maximum acceptable probability of rejection and to increase the number of peers accordingly.

#### 4. Overlay Configuration Document Extension

This document extends the overlay configuration document by defining a new element in the "urn:ietf:params:xml:ns:p2p:quota" namespace.

The `<max-count-per>` element changes the meaning of the `<max-count>` variable. The value "PEER" forces the `<max-count>` to be interpreted as been per storing peer, which is the default quota algorithm when this extension is not used. The value "SIGNER" forces the `<max-count>` to be interpreted as been per signer, which is the algorithm defined by this document.

The Compact Relax NG Grammar for this element is:

```
namespace pqt = "urn:ietf:params:xml:ns:p2p:quota"

kind-parameter &= element pqt:max-count-per { max-count-per-type }
max-count-per-type | = "PEER"
max-count-per-type | = "SIGNER"
max-count-per-type | = xsd:string # extensions
```

## 5. Security Considerations

TBD

## 6. IANA Considerations

This document adds the following URN to the "XML Namespaces" class of the "IETF XML Registry":

```
urn:ietf:params:xml:ns:p2p:quota
```

## 7. Acknowledgements

This document was written with the xml2rfc tool described in [RFC2629].

## 8. References

### 8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[I-D.ietf-p2psip-base]  
Jennings, C., Lowekamp, B., Rescorla, E., Baset, S., and H. Schulzrinne, "REsource LOcation And Discovery (RELOAD) Base Protocol", draft-ietf-p2psip-base-15 (work in progress), May 2011.

### 8.2. Informative References

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

## Appendix A. Release notes

This section must be removed before publication as an RFC.

- A.1. Modifications between  
draft-petithuguenin-p2psip-proportional-quota-01 and  
draft-petithuguenin-p2psip-proportional-quota-00

- o Changed "storing peer" to "signer" as this works also for replica peers.
  - o The default quota algorithm is per storing peer, not per Resource-ID.
  - o Changed the constants in the XML extension accordingly.
  - o Added running code considerations for reference implementation.
- A.2. Modifications between draft-petithuguenin-p2psip-proportional-quota-00 and draft-petithuguenin-vipr-reload-usage-00
- o Instead of having a StorageQuota parameter that gives the maximum number of entries, reused the max-count parameter (that is mandatory anyway) and changes its meaning.
  - o Removed the 3x multiplier to account for the application layer COPY, as it is application specific.
  - o Removed the additional 3x multiplier to compensate for imperfect distribution, and moved the responsibility to the storing nodes.
- A.3. Running Code Considerations
- o Reference Implementation (<http://debian.implementers.org/testing/source/reload.tar.gz>). Marc Petit-Huguenin. Implements version -01.

#### Authors' Addresses

Jonathan Rosenberg  
jdrosen.net  
Monmouth, NJ  
US

Email: [jdrosen@jdrosen.net](mailto:jdrosen@jdrosen.net)  
URI: <http://www.jdrosen.net>

Cullen Jennings  
Cisco  
170 West Tasman Drive  
San Jose, CA 95134  
USA

Phone: +1 408 421-9990  
Email: [fluffy@cisco.com](mailto:fluffy@cisco.com)

Internet-Draft

Proportional Quota

July 2011

Marc Petit-Huguenin  
Stonyfish

Email: [marc@stonyfish.com](mailto:marc@stonyfish.com)

