

RESCAP
Internet-Draft
Expires: December 21, 2002

J. Beck
Sun Microsystems
June 22, 2002

ResCap Requirements
draft-ietf-rescap-req-01.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on December 21, 2002.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

Abstract

A variety of resource identifiers have been widely deployed on the Internet as a means of identifying various resources, services, and destinations. However, a means of attaching a set of attributes or characteristics to a given resource identifier and subsequently assessing those attributes or characteristics has not been specified and deployed.

Two tasks are envisioned. The first task will be to define a general resolution protocol that will translate resource identifiers to a list of attributes. The second task will be to define an administrative model and update protocol that can be used to set up and maintain the information the resolution protocol accesses.

Internet-Draft

RESCAP Requirements

June 2002

This document defines the requirements for these two protocols.

Table of Contents

1.	Discussion	3
2.	Introduction	3
3.	General requirements	3
3.1	Security	3
4.	Resolution protocol requirements	4
4.1	Scalability	4
4.2	Reply data	4
4.3	Granularity	4
4.4	Expiration	5
4.5	Referral	5
4.6	Security	5
4.7	Server location	6
4.8	Preferences	6
4.9	Simplicity	6
4.10	Replication and Synchronization	6
5.	Administrative update protocol requirements	7
5.1	Access control	7
5.2	Inheritance	7
5.3	Scalability	7
5.4	Security	7
5.5	Replication and Synchronization	8
6.	Security Considerations	8
7.	Acknowledgements	8
	References	8
	Author's Address	9
A.	Issues	9
B.	Revision history	9
B.1	Initial revision (beck-00): April 15, 1999	9
B.2	beck-00 -> beck-01: May 14, 1999	9
B.3	beck-01 -> beck-02: June 15, 1999	10
B.4	beck-02 -> beck-02a: November 23, 1999	10
B.5	beck-02a -> beck-03: December 1, 1999	10
B.6	beck-03 -> rescap-00: December 9, 1999	10
	Full Copyright Statement	12

Internet-Draft

RESCAP Requirements

June 2002

1. Discussion

This draft is being discussed on the ResCap mailing list at <rescap@cs.utk.edu>. Subscription requests can be sent to <rescap-request@cs.utk.edu> (send an email message with the word "subscribe" in the body). More information on the mailing list along with an archive of back messages is available at <<ftp://cs.utk.edu/pub/rescap>>.

2. Introduction

An attribute is a named characteristic of a resource identifier with a value that is meaningful in the context in which it is used. An attribute may be requested from a ResCap server. An example of an attribute might be a content type that an email address is capable of receiving.

A capability is a named set of one or more attributes that is meaningful in the context in which it is evaluated. A capability may be requested from a ResCap server. An example of a capability would be the complete set of content types that an email address could receive.

A particularly important resolution service of the general type described in the Abstract is one which, when given a mail address identifying a particular mail recipient, will return a series of attributes describing the capabilities of that recipient. This differs from a directory service in that no searching or other advanced query operations are involved. Likewise; this is not a discovery protocol.

3. General requirements

3.1 Security

The protocols must include support for authenticated messages between clients and servers. Specifically, it must be possible for a rescap client requesting information to reliably identify the server who is the source of the response. The client should be able to require a server to authenticate itself. The question as to whether the server is authorized to respond with the information requested is outside the scope of the protocols. See the Security Considerations section for more information.

Similarly, it must be possible for a rescap server to reliably identify the client who is making the request for information. The server should be able to require the client to authenticate itself. The question as to whether the client is authorized to request the

information is outside the scope of the protocols. See the Security Considerations section for more information.

Because the principal purpose of the protocols is the dissemination of public information, support for confidentiality services is explicitly not required from the protocols. Nevertheless, specific attributes or capabilities may themselves be encrypted, unbeknownst to the protocol. Also, local security policies may require the use of a secure transport layer that includes confidentiality services. See the Security Considerations section for more information.

[4. Resolution protocol requirements](#)

Throughout the rest of this section, "the protocol" refers to the resolution protocol.

[4.1 Scalability](#)

The protocol must be highly scalable both for number of entries in the database and number of entries per second resolved.

Example: Mail services with tens of millions of users could easily expect tens of millions of requests per day for client attribute information.

[4.2 Reply data](#)

The protocol must be able to support attributes and capabilities that

are arbitrarily long text or binary values. The protocol should be optimized for the exchange of relatively short length resource identifiers and attribute values. By way of example, the distinction between short and long could be determined by whether or not the request and response fit in an ordinary UDP packet.

Servers must either provide the information requested, provide a referral indicating from where the information may be requested, or indicate the information is not available. Servers may forward requests on behalf of clients, but the forward must be transparent to the client.

[4.3 Granularity](#)

A mechanism needs to exist whereby a subset of capabilities for a resource can be fetched. I.e., the protocol request syntax should be able ask for one or more features instead of all of them at once. However, the client also needs to be able to ask for all capabilities known to the server without naming all of them.

Beck

Expires December 21, 2002

[Page 4]

Internet-Draft

RESCAP Requirements

June 2002

Example: A client might only want to know the S/MIME capabilities of a recipient, but not care about its media features.

[4.4 Expiration](#)

Some means to indicate the expected lifetime of a capability is required, so that a client application can judge whether, or when, the information should be considered stale.

The expectation is that data will be infrequently updated, and that the granularity for time-stamps would be in seconds.

The protocol should also support a mechanism for indicating the "last changed date" of a given attribute.

Example: The ResCap server may believe that the recipient is only temporarily unable to receive large mail messages.

[4.5 Referral](#)

Some sort of request referral mechanism is needed. In other words,

the protocol must support a mechanism whereby a response can indicate "I don't know, but go ask the ResCap server at address X." or "use the following URL to retrieve the ResCap response you requested." That is, the response might be a simple DNS name, or it might be a full ResCap URL.

Example: A server might delegate all requests for S/MIME certificate information to a different server that keeps track of that type of information.

4.6 Security

The protocol must be able to handle authenticated queries. The protocol must also support transmission of signed and/or encrypted responses. The protocol should allow for a ResCap server to hold and return "pre-signed" responses. This would allow it to hold capability information signed by the controller of the resource to which it refers, and also to sign responses off-line to avoid the need to perform signature computations at the time of responding to a query. This may imply a need to apply a signature to just part of a response.

Servers may require clients to use authenticated requests. Clients are not required to be able to create authenticated queries. Such clients will not be able to make requests of servers requiring authenticated queries; such servers might regard this as a feature.

Clients may require servers to use authenticated responses. Servers must be able to create authenticated responses.

Controls on which attributes will be announced should exist.

Note that consideration of transport-level security is out of scope here; an appropriate mechanism such as TLS or IPSec should be used instead.

Example: A server might give less information to a client that is unauthenticated than to one that is authenticated. Some information from the server may be important enough for the server to want to prevent tampering, or even to prevent snoopers from reading.

[4.7](#) Server location

DNS resource records should be used to determine a protocol server.

Example: The ResCap server that provides responses for resources at "example.com" might itself be running on a host other than "example.com", such as if the administrator of "example.com" has outsourced ResCap services to another company.

[4.8](#) Preferences

Preferences for a set of capabilities, if needed, are to be a supplied in a schema-specific fashion.

Example: A recipient might strongly prefer image/tiff files over image/jpeg because s/he can display image/tiff on his/her system without launching an external application.

[4.9](#) Simplicity

The protocol should be sufficiently simple that it allows implementation of client and/or server functionality in very small, low cost devices (e.g. telephones, modems, printers, smart-cards, etc.).

[4.10](#) Replication and Synchronization

We need to define the model for consistency between replicas - e.g. if a client is using one replica for queries and has to fail over to another, what assumptions can the client make about the consistency between the two replicas?

We should consider whether to assume strict master-slave as this choice would severely limit scalability of the service. Multi-master

seems preferable, but this impacts the protocol as hooks are needed to allow clients to make sense of the data if they need to get it from multiple servers.

[5.](#) Administrative update protocol requirements

Throughout the rest of this section, "the protocol" refers to the

administrative update protocol.

[5.1](#) Access control

Authentication of anyone updating the database is required.

Example: Individual mail users should be able to update some or all of the information about them in the database, but such updates must be done with authentication to prevent others from maliciously entering false information.

[5.2](#) Inheritance

The protocol must support inheritance. Specifically, mechanisms must be provided by which administrators can set default values for members of their administrative domains.

If a change is made to a default resource capability value that has previously been inherited by some other resource, then the inheritor should receive the new value.

Example: The media features for all addresses at a particular mail server might be the same because the mail server processes all messages at all addresses.

[5.3](#) Scalability

The protocol must support atomic processing of request messages. This processing does not include updating any replicated sources for the information nor. The client should receive a completion response from the server, but the underlying protocol (if used over UDP) might require the client to retransmit its request at appropriate intervals until it receives such a response.

[5.4](#) Security

The protocol must include support for authenticated messages. Servers and clients must use authenticated requests and responses to effect changes to attributes or capabilities.

[5.5](#) Replication and Synchronization

SRV and/or NAPTR resource records may be used to determine a protocol server. [SRV, NAPTR]

[6.](#) Security Considerations

Security issues are discussed in sections [2.1](#), [3.6](#), [4.1](#) and [4.4](#) of this memo.

It is also worth noting that the data which these protocols will be designed to deal with are meant to be public, not private.

However, support for authentication should be included in the resolution protocol to permit administrators to restrict the attributes that are returned in response to a request according to a locally specified security policy. The security policy may require the use of a transport security protocol, e.g., TLS [TLS], to provide additional security services not supported by these protocols.

[7.](#) Acknowledgements

The author would like to thank Paul Hoffman, Graham Klyne, Ned Freed, James Galvin and Keith Moore for their assistance.

References

- [RFC2052] Gulbrandsen, A. and P. Vixie, "A DNS RR for specifying the location of services (DNS SRV)", [RFC 2052](#), October 1996.
- [RFC2168] Daniel, R. and M. Mealling, "Resolution of Uniform Resource Identifiers using the Domain Name System", [RFC 2168](#), June 1997.
- [RFC2246] Dierks, T., Allen, C., Treese, W., Karlton, P., Freier, A. and P. Kocher, "The TLS Protocol Version 1.0", [RFC 2246](#), January 1999.
- [RFC2533] Klyne, G., "A Syntax for Describing Media Feature Sets", [RFC 2533](#), March 1999.

Internet-Draft

RESCAP Requirements

June 2002

Author's Address

John Beck
Sun Microsystems
901 San Antonio Road M/S U-MPK-17-202
Palo Alto, CA 94303-4900
US

Phone: +1-650-786-8078
EMail: john.beck+rescap@sun.com

[Appendix A](#). Issues

1. Does [section 3.6](#) need further clean-up, particularly the first paragraph?
2. Are sections [4.1](#) and [4.4](#) redundant?
3. [Section 3.10](#) currently lays out no requirements, but asks some questions and raises some issues which need to be addressed in order for the proper requirements to be determined.

[Appendix B](#). Revision history

[B.1](#) Initial revision (beck-00): April 15, 1999

[B.2](#) beck-00 -> beck-01: May 14, 1999

- o Added note to Abstract about this not being a discovery protocol.
- o Examples added throughout.
- o [Section 1.2](#) renamed from "Long replies" to "Reply data", and clarified.
- o Clarified [section 1.4](#), and added note about support for a "last changed date".
- o [Section 1.6](#) renamed from "Privacy" to Security, and clarified, largely by absorbing [section 2.2](#) (Privacy). As a result, [section 2.3](#) (Inheritance) renumbered to 2.2 .

- o New [section 4](#) (Acknowledgements) added, and old sections [4](#) (References) and 5 (Author's Address) renumbered to 5 and 6.

Beck

Expires December 21, 2002

[Page 9]

Internet-Draft

RESCAP Requirements

June 2002

[B.3](#) beck-01 -> beck-02: June 15, 1999

- o New [section 1.9](#) (Simplicity) added.

[B.4](#) beck-02 -> beck-02a: November 23, 1999

- o Added notes about infrequent updates and granularity of seconds to [section 1.4](#) (Expiration).
- o Added note about transport-level security being out of scope to [section 1.6](#) (Security), and did some word-smithing.
- o [Section 1.7](#) (Server location): specific references to SRV and NAPTR replaced by general reference to DNS. Deleted corresponding references from [section 5](#) (References). Also rewrote example in [section 1.7](#) .
- o [Section 1.8](#) (Preference) changed to be schema-specific.
- o Added note about the effect of changing a default resource to [section 2.2](#) (Inheritance).
- o Added note about public vs. private data to [section 3](#) (Security Considerations).
- o Added [appendix X](#) (Revision history).

[B.5](#) beck-02a -> beck-03: December 1, 1999

- o New [section 1](#) (Introduction) and 2 (General requirements); old sections [1](#) through [6](#) renumbered to 3 through 8.
- o New [section 4.3](#) (Scalability) added.

- o [Section 3.2](#) (Reply data) rewritten.

[B.6](#) beck-03 -> rescap-00: December 9, 1999

- o ResCap is now a WG, so document name changes.
- o Added note about authentication to Security Considerations.
- o Added [appendix I](#) for open issues.
- o Added sections [2.1](#) and [4.4](#), and added to 3.6 (all on Security).

Beck

Expires December 21, 2002

[Page 10]

Internet-Draft

RESCAP Requirements

June 2002

- o Added sections [3.10](#) and [4.5](#) (on Replication and Synchronization).

Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an

"AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.