Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: October 06, 2013

S. Hollenbeck
Verisign Labs
N. Kong
CNNIC

April 04, 2013

Security Services for the Registration Data Access Protocol
draft-ietf-weirds-rdap-sec-02

Abstract

The Registration Data Access Protocol (RDAP) provides "RESTful" web services to retrieve registration metadata from domain name and regional internet registries. This document describes information security services, specific requirements for RDAP, and approaches to provide RDAP security services.

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 October 06, 2013.

Table of Contents

1. Introduction

The Registration Data Access Protocol (RDAP) core is specified in two documents: "Registration Data Access Protocol Lookup Format" [I-D.ietf-weirds-rdap-query] and "JSON Responses for the Registration Data Access Protocol (RDAP)" [I-D.ietf-weirds-json-response]. One goal of RDAP is to provide security services that do not exist in the WHOIS [RFC3912] protocol, including authentication, authorization, availability, data confidentiality, and data integrity.

This document describes each of these security services from the perspective of RDAP requirements and applicability. Where applicable, informational references to requirements for a WHOIS replacement service [RFC3707] are noted.

2. Conventions Used in This Document

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].

2.1. Acronyms and Abbreviations

DNR: Domain Name Registry

RDAP: Registration Data Access Protocol

RIR: Regional Internet Registry

3. Information Security Services and RDAP

RDAP itself does not include native security services. Instead, RDAP relies on features that are available in other protocol layers to provide needed security services including authentication, authorization, availability, data confidentiality, and data integrity. A description of each of these security services can be found in RFC 4949 [RFC4949]. No requirements have been identified for other security services.

3.1. Authentication

WHOIS does not provide features to identify and authenticate clients. As noted in section 3.1.4.2 of RFC 3707 [RFC3707], there is utility in allowing server operators to offer "varying degrees of access depending on policy and need". Clients have to be identified and authenticated to provide that utility.

REQUIREMENT: RDAP MUST include an authentication framework that can accommodate anonymous access as well as verification of identities using a range of authentication methods and credential services.

REQUIREMENT: The RDAP authentication framework MUST use authentication methods that are fully specified and available to existing HTTP clients and servers.

REQUIREMENT: The RDAP authentication framework MUST be capable of supporting future authentication methods defined for use with HTTP.

APPROACH: RDAP clients and servers MUST implement the authentication framework specified in RFC 2617 [RFC2617]. The "basic" scheme can be used to send a client's user name and password to a server in plaintext, based64-encoded form. The "digest" scheme can be used to authenticate a client without exposing the client's plaintext password. If the "basic" scheme is used another protocol (such as HTTP Over TLS [RFC2818]) MUST be used to protect the client's credentials from disclosure while in transit (see Section 3.4).

The Transport Layer Security Protocol [RFC5246] includes an optional feature to identify and authenticate clients who possess and present a valid X.509 digital certificate [RFC5280]. Support for this feature is OPTIONAL.

3.1.1. Federated Authentication

The traditional client-server authentication model requires clients to maintain distinct credentials for every RDAP server. This situation can become unwieldy as the number of RDAP servers increases. Federated authentication mechanisms allow clients to use one credential to access multiple RDAP servers and reduce client credential management complexity. RDAP MAY include a federated authentication mechanism that permits a client to access multiple RDAP servers in the same federation with one credential.

Federated authentication mechanisms used by RDAP are OPTIONAL. If used, they MUST be fully supported by HTTP.

POSSIBLE APPROACH: The OAuth authorization framework [RFC6749] describes a method for users to access protected web resources without having to hand out their credentials. Instead, clients supply access tokens issued by an authorization server with the permission of the resource owner. Using OAuth, multiple RDAP servers can form a federation and the clients can access any server in the same federation by providing one credential registered in any server in that federation. The OAuth authorization framework is designed for use with HTTP and thus can be used with RDAP.

POSSIBLE APPROACH: OpenID [OpenID] is a decentralized single sign-on authentication system that allows users to log in at web sites with one ID instead of having to create multiple unique accounts. OpenID is decentralized. An end user can freely choose which OpenID provider to use, and can preserve their Identifier if they switch OpenID providers. [To be discussed: Is it possible to introduce OpenID into RDAP?]

POSSIBLE APPROACH: Section 7.4.6 of the Transport Layer Security Protocol [RFC5246] describes the specification of a client

certificate. Clients who possess and present a valid X.509 digital certificate, issued by an entity called "Certification Authority" (CA), could be identified and authenticated by a server who trusts the corresponding CA. A certificate authentication method can be used to achieve federated authentication in which multiple RDAP servers all trust the same CAs and then any client with a certificate issued by a trusted CA can access any RDAP server in the federation. This certificate-based mechanism is supported by HTTPS and can be introduced into RDAP.

3.2. Authorization

WHOIS does not provide services to grant different levels of access to clients based on a client's authenticated identity. As noted in section 3.1.4.2 of RFC 3707 [RFC3707], there is utility in allowing server operators to offer "varying degrees of access depending on policy and need". Access control decisions can be made once a client's identity has been established and authenticated (see Section 3.1).

REQUIREMENT: RDAP MUST include an authorization framework that is capable of providing granular (per registration data object) access controls according to the policies of the operator.

APPROACH: Server operators will offer varying degrees of access depending on policy and need in conjunction with the authentication methods described in Section 3.1. Some examples:

- Clients will be allowed access only to data for which they have a relationship.

- Unauthenticated or anonymous access status may not yield any contact information.

- Full access may be granted to a special group of authenticated clients.

The type of access allowed by a server will most likely vary from one operator to the next.

3.3. Availability

An RDAP service has to be available to be useful. There are no RDAP-unique requirements to provide availability, but as a general security consideration a service operator needs to be aware of the issues associated with denial of service. A thorough reading of RFC 4732 [RFC4732] is RECOMMENDED.

An RDAP service MAY use a throttling mechanism to limit the number of queries that a single client can send in a given period of time. If used, the server SHOULD return a 429 response code as described in RFC 6585 [RFC6585]. A client that receives a 429 response SHOULD decrease its query rate, and honor the Retry-After header if one is present.

3.4. Data Confidentiality

WHOIS does not provide the ability to encrypt data while in transit to protect it from inadvertent disclosure. Web services commonly use HTTP Over TLS [RFC2818] to provide that protection.

REQUIREMENT: RDAP or a protocol layer used by RDAP MUST include features to protect plaintext client credentials used for client authentication.

REQUIREMENT: The data confidentiality methods used by RDAP MUST be fully specified and available to existing HTTP clients and servers.

REQUIREMENT: RDAP MUST be capable of supporting future data confidentiality methods defined for use with HTTP.

OPTION: RDAP or a protocol layer used by RDAP MAY include features to encrypt client-server data exchanges.

APPROACH: As noted in Section 3.1, the HTTP "basic" authentication scheme can be used to authenticate a client. When this scheme is used HTTP Over TLS [RFC2818] MUST be used to protect the client's credentials from disclosure while in transit. HTTP Over TLS MAY also be used to protect client-server data exchanges if the policy of the server operator requires encryption. There are no current requirements for object-level encryption, but RDAP MUST NOT preclude support for this feature in the future.

3.5. Data Integrity

WHOIS does not provide the ability to protect data from modification while in transit. Web services commonly use HTTP Over TLS [RFC2818] to provide that protection. Digital signatures as described in RFC 4949 [RFC4949] are also used to provide data integrity. Note that this security service is often mistakenly associated with policy requirements focused on data accuracy; those requirements are out of scope for this protocol. The most specific need for this service is to provide assurance that HTTP redirection hints [RFC2616] are not modified.

REQUIREMENT: RDAP or a protocol layer used by RDAP MUST include features to protect HTTP 30x redirection hints from modification.

REQUIREMENT: The data integrity methods used by RDAP MUST be fully specified and available to existing HTTP clients and servers.

OPTION: RDAP or a protocol layer used by RDAP MAY include features to provide message integrity checks.

REQUIREMENT: RDAP MUST be capable of supporting future JSON data integrity methods defined for use with HTTP.

OPTION: RDAP or a protocol layer used by RDAP MAY include features to provide data integrity by signing JSON-encoded objects.

APPROACH: HTTP Over TLS MAY be used to protect client-server data exchanges if the policy of the server operator requires message integrity. There are no current requirements for object-level data signing, but RDAP MUST NOT preclude support for this feature in the future.

4. IANA Considerations

This document does not specify any IANA actions. This section can be removed if this document is published as an RFC.

5. Security Considerations

One of the goals of RDAP is to provide security services that do not exist in the WHOIS protocol. This document describes the security services provided by RDAP and associated protocol layers, including authentication, authorization, availability, data confidentiality, and data integrity. Non-repudiation services were also considered and ultimately rejected due to a lack of requirements. There are, however, currently-deployed WHOIS servers that can return signed responses that provide non-repudiation with proof of origin. RDAP MUST NOT preclude support for this feature in the future.

As an HTML-based protocol RDAP is susceptible to code injection attacks. Code injection refers to adding code into a computer system or program to alter the course of execution. There are many types of code injection, including SQL injection, dynamic variable or function injection, include file injection, shell injection, and html-script injection among others. Data confidentiality and integrity services provide a measure of defense against man-in-the-middle injection attacks, but vulnerabilities in both client- and server-side software make it possible for injection attacks to succeed.

6. Acknowledgements

The authors would like to acknowledge the following individuals for their contributions to this document: Marc Blanchet, Jean-Philippe Dionne, Andrew Newton, and Linlin Zhou.

7. References

7.1. Normative References

[I-D.ietf-weirds-json-response]
Newton, A. and S. Hollenbeck, "JSON Responses for the Registration Data Access Protocol (RDAP)", draft-ietf-weirds-json-response-02 (work in progress), January 2013.

[I-D.ietf-weirds-rdap-query]
Newton, A. and S. Hollenbeck, "Registration Data Access Protocol Lookup Format", draft-ietf-weirds-rdap-query-03 (work in progress), March 2013.

[OpenID] OpenID Foundation, "OpenID Authentication 2.0 - Final ", December 2007, <http://specs.openid.net/auth/2.0>.

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

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC2617] Franks, J., Hallam-Baker, P.M., Hostetler, J.L., Lawrence, S.D., Leach, P.J., Luotonen, A., and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

[RFC4732] Handley, M., Rescorla, E., IAB, "Internet Denial-of-Service Considerations", RFC 4732, December 2006.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[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, May 2008.

[RFC6585] Nottingham, M. and R. Fielding, "Additional HTTP Status Codes", RFC 6585, April 2012.

[RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", RFC 6749, October 2012.

7.2. Informative References

[RFC3707] Newton, A., "Cross Registry Internet Service Protocol (CRISP) Requirements", RFC 3707, February 2004.

[RFC3912] Daigle, L., "WHOIS Protocol Specification", RFC 3912, September 2004.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.

Appendix A. Change Log

Initial -00: Adopted as working group document.

-01: Extensive text additions and revisions based on in-room discussion at IETF-85. Sections for data integrity and non-repudiation have been removed due to a lack of requirements, but both topics are now addressed in the Security Considerations section.

-02: Fixed document names in the Introduction. Modified text in Section 3.1.1 to clarify requirement. Added text to Section 3.3 to describe rate limiting. Added new data integrity section. Updated security considerations to describe injection attacks.

Authors' Addresses

Scott Hollenbeck
Verisign Labs
12061 Bluemont Way
Reston, VA 20190
US

Email: shollenbeck@verisign.com
URI: http://www.verisignlabs.com/

Ning Kong
China Internet Network Information Center
4 South 4th Street, Zhongguancun, Haidian District
Beijing 100190
China

Phone: +86 10 5881 3147
Email: nkong@cnnic.cn