Simple Web Discovery (SWD)
draft-jones-simple-web-discovery-04

Abstract

Simple Web Discovery (SWD) defines an HTTPS GET based mechanism to discover the location of a given type of service for a given principal starting only with a domain name.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Simple Web Discovery (SWD) defines an HTTPS GET based mechanism to discover the location of a given type of service for a given principal starting only with a domain name. SWD requests use query parameters to specify a URI for the principal and another URI for the type of service being sought. If the request is successful then the response, by default, is a JavaScript Object Notation (JSON) **[RFC4627]** object containing an array of URIs that point to where the principal has instances of services of the requested type.

For example, let us say that a requester wants to discover where Joe keeps his calendar. The requester could take Joe's e-mail address, joe@example.com, and use its domain to create an HTTPS GET request of the following form (with long lines broken for display purposes only):

```
GET /.well-known/simple-web-discovery
  ?principal=joe@example.com
  &service=urn:example:service:calendar HTTP/1.1
Host: example.com

HTTP/1.1 200 OK
Content-Type: application/json

{
 "locations": ["https://calendars.example.net/calendars/joseph"]
}
```

Note: The request-URI is left unencoded in the above example for the sake of readability. The query parameters above would actually be encoded as ?
principal=joe%40example.com&service=urn%3Aexample%3Aservice%3Acalendar.

1.1. Notational Conventions

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 Key words for use in RFCs to Indicate Requirement Levels **[RFC2119]**.

2. Simple Web Discovery Request

Domains that support SWD requests SHOULD make an SWD server available for their domain at the path /.well-known/simple-web-discovery. The syntax and semantics of /.well-known are defined in **RFC 5785** [RFC5785]. /.well-known/simple-web-discovery MUST point to an SWD server compliant with this specification.

SWD servers MUST support receiving SWD requests via TLS 1.2 **[RFC5246]** and MAY support other transport layer security mechanisms of equivalent security. SWD servers MUST reject SWD requests sent over plain HTTP or any other transport that does not provide both privacy and validation of the server's identity.

An SWD server is queried using an HTTPS GET request with the previously specified path along with a query segment containing a form encoded using the application/x-www-form-urlencoded encoding algorithm as defined in **HTML 4.01** [W3C.REC-html401-19991224]. The form MUST contain two name/value pairs that MUST appear exactly once, principal and service. Both name/value pairs MUST have values that are set to URIs **[RFC3986]**. If any of the previous requirements are not met in an SWD request, then the request MUST be rejected with a 400 Bad Request.

The SWD request form MAY contain additional name/value pairs but if those name/value pairs

are not recognized by the SWD server then the SWD server MUST ignore them for processing purposes.

The `principal` query component is a URI that identifies an entity. The `service` query component is a URI that identifies a service type. The semantics of the SWD query is "Please return the location(s) of instances of the specified service type associated with the specified principal". The definition of URIs used to identify principals and services are outside the scope of this specification.

SWD servers MAY also be located on ports other than 443 (the default HTTPS port), provided they use TLS on those ports. The means by which an SWD client would know to use any alternative ports are out of scope for this specification.

2.1. "simple-web-discovery" Subdomain

It may be difficult or impossible for some domains wanting to support SWD requests to make an SWD server available for their domain at the path `/.well-known/simple-web-discovery`. For instance, in the case of hosted domains, no web server may be running on the domain host at all.

For that reason, SWD servers for a domain MAY be located on a specific subdomain of that domain: `simple-web-discovery`. For example, the SWD server for the domain `example.com` MAY be located at the URI `https://simple-web-discovery.example.com/.well-known/simple-web-discovery`.

SWD clients MUST first attempt to make an SWD request to the domain's `/.well-known/simple-web-discovery` endpoint, and then if that fails, they MUST then attempt to make the request to the SWD endpoint at the `simple-web-discovery` subdomain for the domain.

3. Simple Web Discovery Responses

3.1. Response Containing One or More Locations

A 200 OK response to an SWD request that contains the information requested MUST return content of type application/json **[RFC4627]**. The JSON response MUST contain a JSON object that contains a member pair whose name is the string `locations` and whose value is an array of strings that are each a URI pointing to a location where the desired service type belonging to the specified principal can be found. There are no semantics associated with the order in which the URIs are listed in the array.

The JSON object MAY contain other members but a receiver of the object MAY ignore any member pairs whose name it does not recognize.

3.2. 401 Unauthorized Response

An SWD server MAY respond to a request with a 401 Unauthorized Response, as described in **RFC 2616** [RFC2616], Section 10. Per the RFC, the request MAY be repeated with a suitable Authorization header field. Authorization information may be communicated in this manner, including a JSON Web Token **[JWT]**.

3.3. Other HTTP 1.1 Responses

An SWD server MAY return other HTTP 1.1 responses, including 404 Not Found, 400 Bad

Request, and 403 Forbidden. SWD implementations MUST correctly handle these responses.

4. IANA Considerations

This specification registers a well-known URI suffix value relative to "/.well-known/" in the IANA Well-Known URI registry defined in **RFC 5785** [RFC5785]:

URI suffix:
> simple-web-discovery

Change controller:
> IETF

Specification document:
> [[this document]]

5. Security Considerations

SWD responses can contain confidential information. Therefore a, general approach is used to require TLS in all cases. But TLS can only provide for privacy and server validation, it cannot validate that the requester is authorized to see the results of a query. The exact mechanism used to determine if the requester is authorized to see the result of the query is outside the scope of this specification.

Because SWD responses can contain confidential information, the requestor may need authorization to receive them. Standard HTTP authorization mechanisms MAY be employed to request authorized access, including the use of an HTTP Authorization header field in requests, which in turn, may contain a JSON Web Token **[JWT]**, among other authorization data formats.

When the SWD server for a domain is located at the `simple-web-discovery` subdomain, a TLS certificate will need to be present for that subdomain.

6. References

6.1. Normative References

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

[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 (**TXT**, **PS**, **PDF**, **HTML**, **XML**).

[RFC3986]   Berners-Lee, T., Fielding, R., and L. Masinter, "**Uniform Resource Identifier (URI): Generic Syntax**," STD 66, RFC 3986, January 2005 (**TXT**, **HTML**, **XML**).

[RFC4627]   Crockford, D., "**The application/json Media Type for JavaScript Object Notation (JSON)**," RFC 4627, July 2006 (**TXT**).

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

[RFC5785]   Nottingham, M. and E. Hammer-Lahav, "**Defining Well-Known Uniform Resource Identifiers (URIs)**," RFC 5785, April 2010 (**TXT**).

[W3C.REC-html401-19991224]   Hors, A., Raggett, D., and I. Jacobs, "**HTML 4.01 Specification**," World Wide Web Consortium Recommendation REC-html401-19991224, December 1999 (**HTML**).

6.2. Informative References

[JWT]   Jones, M., Bradley, J., and N. Sakimura, "**JSON Web Token (JWT)**," October 2012.

Appendix A. Document History

[[to be removed by the RFC editor before publication as an RFC]]

-04

- Specified that the SWD server for a domain may be located at the `simple-web-discovery` subdomain of the domain and that SWD clients must first try the endpoint at the domain and then the endpoint at the subdomain.
- Removed the `SWD_service_redirect` response, since redirection can be accomplished by pointing the `simple-web-discovery` subdomain to a different location than the domain's host.
- Removed `mailto:` from examples in favor of bare e-mail address syntax.
- Specified that SWD servers may also be run on ports other than 443, provided they use TLS on those ports.

-03

- Changed "requests use the x-www-form-urlencoded format" to "requests use query parameters" and changed "an x-www-form-urlencoded form" to "a form encoded using the application/x-www-form-urlencoded encoding algorithm", both at the suggestion of Julian Reschke.
- Updated examples to use "urn:example:" URNs rather than "urn:example.org:" URNs, also at Julian's suggestion.
- Applied applicable editorial improvements from JOSE specs to SWD.
- Updated references to related specifications.

-02

- Update examples to use example.{com,net,org} domain names.
- Provide encoded representation of the request-URI query parameters for the first example request.
- Changed "200 O.K." to "200 OK".
- Respect line length restrictions in examples.
- No normative changes.

-01

- Refresh draft before expiration of -00. No normative changes.

-00

- Initial version.

Authors' Addresses

Michael B. Jones
Microsoft
Email: mbj@microsoft.com
URI: http://self-issued.info/

Yaron Y. Goland
Microsoft
Email: yarong@microsoft.com