Well-Known URIs

The `.well-known` standard, defined in RFC 8615, serves as a standardized directory within a website's root domain. This designated location, typically accessible via the `/.well-known/` path on a web server, centralizes a website's critical metadata, including configuration files and information related to its services, protocols, and security mechanisms.

By establishing a consistent location for such data, `.well-known` simplifies the discovery and access process for various stakeholders, including web browsers, applications, and security tools. This streamlined approach enables clients to automatically locate and retrieve specific configuration files by constructing the appropriate URL. For instance, to access a website's security policy, a client would request `https://example.com/.well-known/security.txt`.

The `Internet Assigned Numbers Authority` (`IANA`) maintains a registry of `.well-known` URIs, each serving a specific purpose defined by various specifications and standards. Below is a table highlighting a few notable examples:

URI Suffix	Description	Status	Reference
`security.txt`	Contains contact information for security researchers to report vulnerabilities.	Permanent	RFC 9116
`/.well-known/change-password`	Provides a standard URL for directing users to a password change page.	Provisional	https://w3c.github.io/webappsec-change-password-url/#the-change-password-well-known-uri
`openid-configuration`	Defines configuration details for OpenID Connect, an identity layer on top of the OAuth 2.0 protocol.	Permanent	http://openid.net/specs/openid-connect-discovery-1_0.html
`assetlinks.json`	Used for verifying ownership of digital assets (e.g., apps) associated with a domain.	Permanent	https://github.com/google/digitalassetlinks/blob/master/well-known/specification.md
`mta-sts.txt`	Specifies the policy for SMTP MTA Strict Transport Security (MTA-STS) to enhance email security.	Permanent	RFC 8461

This is just a small sample of the many `.well-known` URIs registered with IANA. Each entry in the registry offers specific guidelines and requirements for implementation, ensuring a standardized approach to leveraging the `.well-known` mechanism for various applications.

Web Recon and .well-known

In web recon, the `.well-known` URIs can be invaluable for discovering endpoints and configuration details that can be further tested during a penetration test. One particularly useful URI is `openid-configuration`.

The `openid-configuration` URI is part of the OpenID Connect Discovery protocol, an identity layer built on top of the OAuth 2.0 protocol. When a client application wants to use OpenID Connect for authentication, it can retrieve the OpenID Connect Provider's configuration by accessing the `https://example.com/.well-known/openid-configuration` endpoint. This endpoint returns a JSON document containing metadata about the provider's endpoints, supported authentication methods, token issuance, and more:

Code: json

```
{
  "issuer": "https://example.com",
  "authorization_endpoint": "https://example.com/oauth2/authorize",
  "token_endpoint": "https://example.com/oauth2/token",
  "userinfo_endpoint": "https://example.com/oauth2/userinfo",
  "jwks_uri": "https://example.com/oauth2/jwks",
  "response_types_supported": ["code", "token", "id_token"],
  "subject_types_supported": ["public"],
  "id_token_signing_alg_values_supported": ["RS256"],
  "scopes_supported": ["openid", "profile", "email"]
}
```

The information obtained from the `openid-configuration` endpoint provides multiple exploration opportunities:

1. `Endpoint Discovery`:
   - `Authorization Endpoint`: Identifying the URL for user authorization requests.
   - `Token Endpoint`: Finding the URL where tokens are issued.
   - `Userinfo Endpoint`: Locating the endpoint that provides user information.
2. `JWKS URI`: The `jwks_uri` reveals the `JSON Web Key Set` (`JWKS`), detailing the cryptographic keys used by the server.
3. `Supported Scopes and Response Types`: Understanding which scopes and response types are supported helps in mapping out the functionality and limitations of the OpenID Connect implementation.
4. `Algorithm Details`: Information about supported signing algorithms can be crucial for understanding the security measures in place.

Exploring the IANA Registry and experimenting with the various `.well-known` URIs is an invaluable approach to uncovering additional web reconnaissance opportunities. As demonstrated with the `openid-configuration` endpoint above, these standardized URIs provide structured access to critical metadata and configuration details, enabling security professionals to comprehensively map out a website's security landscape.