The secret-token URI Scheme

draft-nottingham-how-did-that-get-into-the-repo-00

Abstract

This document registers the "secret-token" URI scheme, to aid in the identification of authentication tokens.

Note to Readers

/rfc editor: please remove this section before publication_

The issues list for this draft can be found at https://github.com/mnot/I-D/labels/how-did-that-get-into-the-repo [1].

The most recent (often, unpublished) draft is at https://mnot.github.io/I-D/how-did-that-get-into-the-repo/ [2].

Recent changes are listed at https://github.com/mnot/I-D/commits/gh-pages/how-did-that-get-into-the-repo [3].

See also the draft’s current status in the IETF datatracker, at https://datatracker.ietf.org/doc/draft-nottingham-how-did-that-get-into-the-repo/ [4].

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 https://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 February 17, 2019.
1. Introduction

It has become increasingly common to use bearer tokens as an authentication mechanism.

Unfortunately, the number of security incidents involving accidental disclosure of these tokens has also increased. For example, we now regularly hear about a developer committing an access token to a public source code repository, either because they didn’t realise it was included in the committed code, or because they didn’t realise the implications of its disclosure.

This specification registers the "secret-token" URI scheme to aid prevention of such accidental disclosures. When tokens are easier to unambiguously identify, they can trigger warnings in Continuous Integration systems, or be used in source code repositories themselves. They can also be scanned for separately.

For example, if cloud.example.net issues access tokens to its clients for later use, and it does so by formatting them as secret-token URIs, tokens that "leak" into places that they don’t belong are
easier to identify. This could be through a variety of mechanisms; for example, if repo.example.com can be configured to refuse commits containing secret-token URIs, it helps its customers avoid accidental disclosures.

secret-token URIs are intended to aid in identification of generated secrets like API keys and similar tokens. They are not intended for use in controlled situations where ephemeral tokens are used, such as things like CSRF tokens.

1.1. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

This document uses ABNF [RFC5234], including by reference the following rules: ALPHA, DIGIT.

2. The secret-token URI scheme

The secret-token URI scheme identifies a token that is intended to be a secret.

secret-token-URI = secret-token-scheme "::" token
secret-token-scheme = "secret-token"
token = 1*tchar
tchar = ALPHA / DIGIT / "-" / "_" / "." / "~"

Note that characters outside the range allowed by tchar MUST be percent-encoded ([RFC3986], Section 2.1).

When a token is both generated and presented for authentication, the entire URI MUST be used, without changes.

For example, given the URI:

secret-token:E92FB7EB-D882-47A4-A265-A0B6135DC842%20foo

This string (character-for-character, case-sensitive) will both be issued by the token authority, and required for later access.
3. IANA Considerations

This document registers the following value in the URI Scheme registry:

- Scheme name: secret-token
- Status: permanent
- Applications / protocols that use this scheme: none yet
- Contact: iesg@iesg.org
- Change Controller: IESG
- References: [this document]

4. Security Considerations

The token ABNF rule allows tokens as small as one character. This is not recommended practice; applications should evaluate their requirements for entropy and issue tokens correspondingly.

This URI scheme is intended to prevent accidental disclosure; it cannot prevent intentional disclosure.

If it is difficult to correctly handle secret material, or unclear as to what the appropriate handling is, users might choose to obfuscate their secret tokens in order to evade detection (for example, removing the URI scheme for storage). Clear guidelines and helpful tools are good mitigations here.

5. References

5.1. Normative References


5.2. URIs


Author’s Address

Mark Nottingham

Email: mnot@mnot.net
URI: https://www.mnot.net/