## Ruby - Bug #20690

## URI.encode_www_form_component method escapes tilde when it's not supposed to

08/22/2024 09:33 AM - y4m4p (Masahiro Yamashita)

| | | | |
|---|---|---|---|
| **Status:** | Rejected | | |
| **Priority:** | Normal | | |
| **Assignee:** | | | |
| **Target version:** | | | |
| **ruby -v:** | ruby 3.2.2 (2023-03-30 revision e51014f9c0) [arm64-darwin21] | **Backport:** | 3.1: UNKNOWN, 3.2: UNKNOWN, 3.3: UNKNOWN |

**Description**

# Problem

As the title says,  Uri.encode_www_form_component is escaping tilde (~) where it should not according to the RFC3986.
Ref: https://datatracker.ietf.org/doc/html/rfc3986#section-2.3

There was a previous issue with the same problem, https://bugs.ruby-lang.org/issues/6696 which has been resolved and merged.

---

Since URI.escape is now obsolete and gone, the preferred way to escape URI characters are to choose from the following methods.

- URI.encode_www_form_component (or, URI.encode_www_form which uses this)
- CGI.escape
- ERB::Util.#url_encode
- WEBrick::HTTPUtils.#escape_form
- WEBrick::HTTPUtils.#escape

This issue does not occur when using any of the methods except URI.encode_www_form_component.
These preferred options are not compatible with each other, making it a poor experience for Ruby users.

### Minimum replication

```
# URI
irb(main):002:0> require 'uri'
=> true
irb(main):004:0> URI.encode_www_form_component("ruby~test")
=> "ruby%7Etest"

# CGI
irb(main):001:0> require 'cgi'
=> true
irb(main):003:0> CGI.escape("ruby~test")
=> "ruby~test"

# ERB
irb(main):009:0> require 'erb'
=> true
irb(main):012:0> ERB::Util.url_encode("ruby~test")
=> "ruby~test"

# Webrick
# needs webrick gem installed
[3] pry(main)> require 'webrick'
=> true
[4] pry(main)> WEBrick::HTTPUtils.escape_form("ruby~test")
=> "ruby~test"
[5] pry(main)> WEBrick::HTTPUtils.escape("ruby~test")
=> "ruby~test"
```

# Real world use-case problem

I have a use-case problem with using URI when implementing a OIDC with PKCE client / server setup.
When trying to generate a GET URI for Open ID Connect with PKCE, you need to provide a code_verifier value to the Authorization server, which actually requires you to use the tilde without escaping.

Ref: https://www.oauth.com/oauth2-servers/pkce/authorization-request/

> When the native app begins the authorization request, instead of immediately launching a browser, the client first creates what is known as a “code verifier“. This is a cryptographically random string using the characters A-Z, a-z, 0-9, and the punctuation characters -._~ (hyphen, period, underscore, and tilde), between 43 and 128 characters long.

The inconsistent encoding is problematic for the Server and Client both, since both ends needs to have an agreement on using tilde as an escaped character or an unescaped character.
Logically speaking, ~ should be sent AS-IS and not to be escaped, since the code_challenge value and code_verifier values are strictly checked using a Hash generation function Digest::SHA256.digest, which generates a different Hash value depending on ~(non escaped) or %7E(escaped).
These parameters from user inputs are usually not meant to be tampered with for security sake, so it's best to have tilde ~ to be unescaped from the beginning.

## Related issues

- https://bugs.ruby-lang.org/issues/6696
    - This seems to have been resolved and merged.

### History

**#1 - 08/26/2024 02:37 AM - y4m4p (Masahiro Yamashita)**

I labeled it as a bug, but maybe it's more of a feature request..? sorry if I'm getting it wrong.

I have also created a PR on the uri repo to address this particular case.
PR: https://github.com/ruby/uri/pull/117

This change might affect some Ruby users who rely on tilde being escaped for whatever reason, so I'm not sure if this should be included in a patch, so I'd want to hear the mainainer's thoughts on this.
Thank you.

**#2 - 09/05/2024 09:23 AM - naruse (Yui NARUSE)**

*- Status changed from Open to Rejected*

URI.encode_www_form_component is designed for HTML form submission.
It is not whatt RFC3986 defines as "Percent-Encoding", but defined as "application/x-www-form-urlencoded".
(The latest definition is https://url.spec.whatwg.org/#urlencoded-serializing)

As far as I understand, your use case is different from "application/x-www-form-urlencoded" because it doesn't unescape "~".
I think you should implement your own function for your use case.

**#3 - 12/06/2024 10:20 AM - mentalizer (Jakob Skjerning)**

While URI.encode_www_form_component might not deal with what RFC3986 defines as "Percent-Encoding", URI.encode_uri_component does (I think?) and that exhibits the same behavior:

```
3.3.4 :006 > URI.encode_uri_component("ruby~test")
 => "ruby%7Etest"
```

Perhaps this issue should be reopened and changed to target URI.encode_uri_component instead, or I can open another issue with a similar description, if need be?