

Re: urllib interpretation of URL with ".."

Source: <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2007-06/msg03173.html>

- *From:* jjl@xxxxxxxxxx (John J. Lee)
 - *Date:* Mon, 25 Jun 2007 21:12:02 GMT
-

John Nagle <nagle@xxxxxxxxxxxx> writes:

Duncan Booth wrote:

"Martin v. Löwis" <martin@xxxxxxxxxxxx> wrote:

Is "urllib" wrong?

Section 5.2 is also relevant here. In particular:

g) If the resulting buffer string still begins with one or more complete path segments of "..", then the reference is considered to be in error. Implementations may handle this error by retaining these components in the resolved path (i.e., treating them as part of the final URI), by removing them from the resolved path (i.e., discarding relative levels above the root), or by avoiding traversal of the reference.

The common practice seems to be for client-side implementations to handle this using option 2 (removing them) and servers to use option 3 (avoiding traversal of the reference). urllib uses option 1 which is also correct but not as useful as it might be.

That's helpful. Thanks.

In Python, of course, "urlparse.urlparse", which is the main function used to disassemble a URL, has no idea whether it's being used by a client or a server, so it, reasonably enough, takes option 1.

Re: urllib interpretation of URL with ".."

(Yet another hassle in processing real-world HTML.)

Note that RFC 3986 obsoletes RFC 2396, and attempts to codify current good practice re generic URL syntax (URI and relative reference syntax, to use the precise terminology of the RFC). It discusses normalisation at length, quite sensibly and pragmatically. And very readable and useful it is too.

Somebody submitted a module implementing the URL splitting / joining algorithms specified in RFC 3986 for inclusion in Python 2.6 — I haven't looked at that recently...

See also RFC 3987.

John

.