Research Center for
Information Security

Proposal for Application-friendly HTTP Authentication Extension

Yutaka OIWA
RCIS, AIST

September 9, 2010
IIW East Coast I

National Institute of
Advanced Industrial Science
and Technology
AIST

独立行政法人
産業技術総合研究所

HTTP Auth and Web apps.

- Many people consider that HTTP Auth is not very useful with real web applications
  - It lacks a cute UI
  - Modal dialog interrupts user experience
  - No (clean) way to accept guest users
  - No logout, no timeout
  - Etc…

  - Form-based auth is used widely

Form auth, from security-side

- HTML Form auth is very insecure
  - Plaintext sent to the remote server
  - All behavior of authentication controlled by server

  - Very easy to exploit phishing attack
  - No way to introduce stronger authentications
    - Even if someone introduced an extension to HTML, it could be easily bypassed by phishing sites

Harden HTTP auth!

- Reintroducing HTTP auth is the way to solve
  - E.g. HTTP Mutual authentication [draft-oiwa-http-mutualauth-07]
  - Digest is obsolete/historic and almost deprecated, but still better than Basic/Form
- Protocols can ensure use of cryptographic primitives for stronger protection
- But to use it, it should be application-friendly

My current proposal

- A small extension to current HTTP auth.
  - Optional HTTP authentication
    - [draft-oiwa-http-mutualauth-07, Section 4.7]
  - Authentication-Control: header
    - [draft-oiwa-http-mutualauth-07, Section 10]

  - I want to extend/finalize the design and propose to IETF for standardization
  - So, I need feedbacks/opinions

Optional HTTP authentication

- New header with “successful” responses
  - “It's OK, but you MAY authenticate if you want”
  - Compared with 401 status:
    - “Oh, No, you MUST authenticate to access it”
  - Useful with guest-enabled accesses

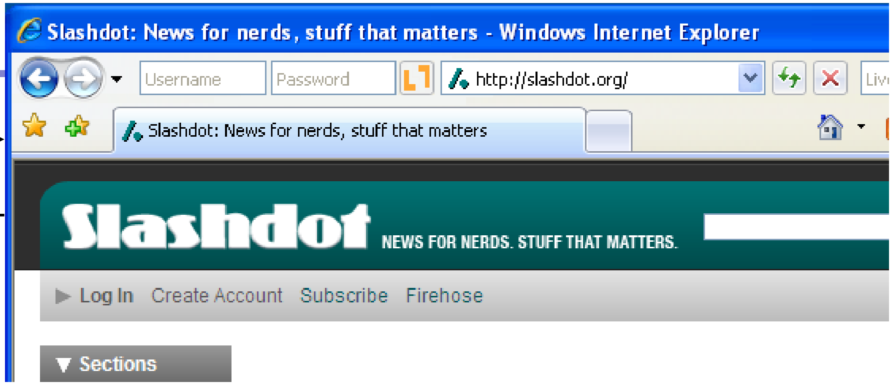
Optional HTTP authentication

```
GET /index.htm HTTP/1.1

HTTP/1.1 200 OK
Optional-WWW-Authenticate: Basic
           realm="user-customized contents"




GET /index.htm HTTP/1.1
Authorization: Basic AfewDG3fher=

HTTP/1.1 200 OK
```

Imaginary figure

GET / HTTP/1.1

- HTTP/1.1 200 OK
- Optional-WWW-Auth.: Mutual …

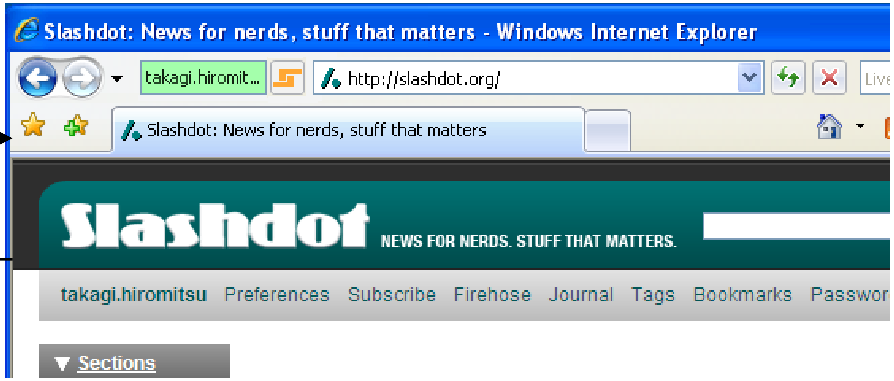

Slashdot: News for nerds, stuff that matters - Windows Internet Explorer
Username
Password
http://slashdot.org/
Live
Slashdot: News for nerds, stuff that matters
Slashdot
NEWS FOR NERDS. STUFF THAT MATTERS.
Log In
Create Account
Subscribe
Firehose
Sections

<u>Same URL for guest/authenticated contents</u>

GET / HTTP/1.1

Authorization: Mutual wa=…

- HTTP/1.1 200 OK
- Authorization-info: …


Slashdot: News for nerds, stuff that matters - Windows Internet Explorer
takagi.hiromit...
http://slashdot.org/
Live
Slashdot: News for nerds, stuff that matters
Slashdot
NEWS FOR NERDS. STUFF THAT MATTERS.
takagi.hiromitsu
Preferences
Subscribe
Firehose
Journal
Tags
Bookmarks
Passwor
Sections

Authentication-Control: Header

- Server-supplied “hints” for precise behavior of clients regarding authentications
  - Not mandatory for clients to obey any hints
  - But, it may give better user experiences if followed

Authentication-Control: Header

```
GET /private HTTP/1.1


HTTP/1.1 401 Authentication Needed
WWW-Authenticate: Basic realm="protected"
Authentication-Control: Basic mode=non-modal,
  location-when-unauthenticated="/login.php"
```

- Accepts several key-value pairs for hints

Authentication mode

- Modal or non-modal

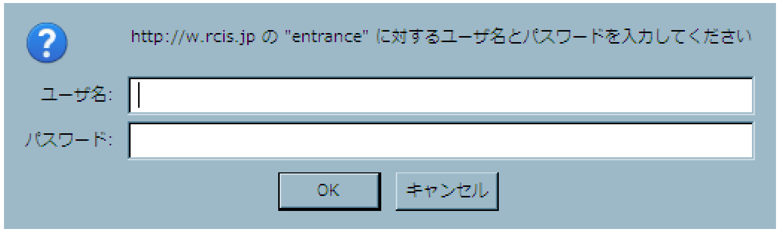
  (not in the current Mutual auth draft)

  - *modal*: authentication to be performed before processing 401 message body
    - Need to press “cancel” to see message body
    - Default for Basic/Digest
  - *Non-modal*: authentication will not block the message body
    - Default for proposed Mutual authentication
      - Use secure chrome-area UI for authentication
    - Maybe also useful with Basic/Digest
      - Use information bar
      - Mozilla's Account Manager proposal

Authentication mode

- Modal

GET / HTTP/1.1



http://w.rcis.jp の "entrance" に対するユーザ名とパスワードを入力してください

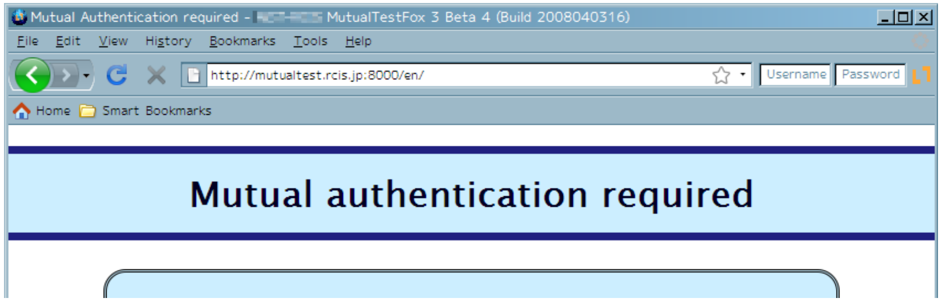
ユーザ名:

パスワード:

- Blocked by modal dialog, user may cancel auth.

- Non-modal

GET / HTTP/1.1



Mutual Authentication required - [redacted] MutualTestFox 3 Beta 4 (Build 2008040316)

http://mutualtest.rcis.jp:8000/en/

Mutual authentication required

- Content displayed first, user auths if want to continue

Location-when-unauthenticated

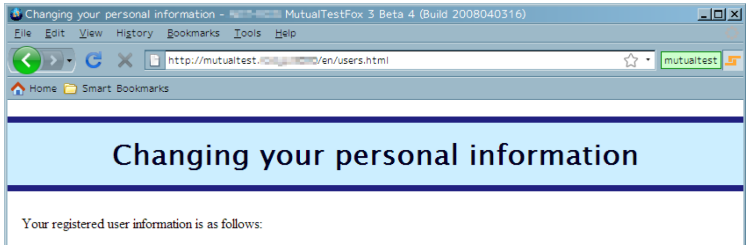
- Redirects browser for a specific URL if there is no authenticated user
  - Useful is there is a specific authentication page
    - Common design in recent web apps.
  - Browser will
    - Follow redirection, if it does not know the user
    - Try authentication, it already knows credentials

Location-when-logout

- Redirects if the user want to log-out
  - For the browsers with log-out facilities
  - Avoid users from being confused with obsolete authenticated contents or other messages
    - Especially useful, if current page is better not to be accessed as unauthenticated users
      - For example, a page of POST result

Location-when-logout

Authenticated page



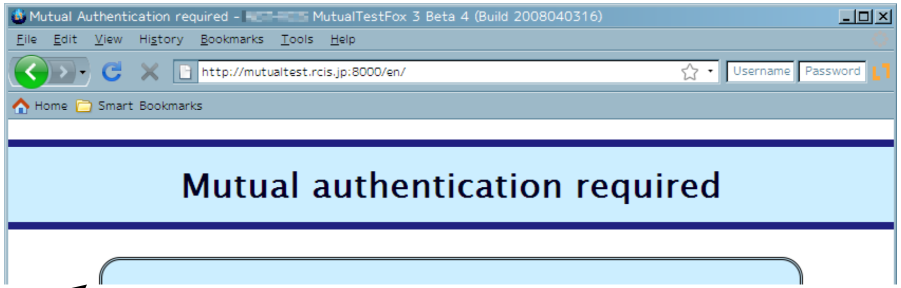
Changing your personal information - [redacted] MutualTestFox 3 Beta 4 (Build 2008040316)

http://mutualtest.[redacted]/en/users.html

Changing your personal information

Your registered user information is as follows:

Logout

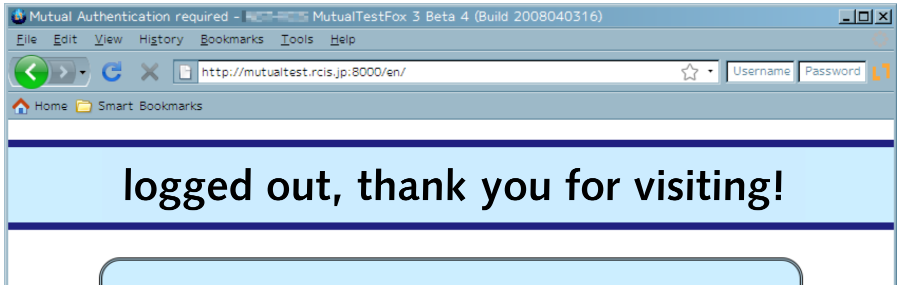


Mutual Authentication required - [redacted] MutualTestFox 3 Beta 4 (Build 2008040316)

http://mutualtest.rcis.jp:8000/en/

Mutual authentication required

Location-when-logout not specified:
a normal 401 error page



Mutual Authentication required - [redacted] MutualTestFox 3 Beta 4 (Build 2008040316)

http://mutualtest.rcis.jp:8000/en/

logged out, thank you for visiting!

Location-when-logout specified:
a special logout-page

Logout-timeout

- Request client-side timer for automatic logout

```
GET /private HTTP/1.1
Authorization: Mutual oa=……

HTTP/1.1 200 OK
Authentication-Control: Mutual logout-timeout=300
```

  - To be logged out in 300 sec from now
  - Timer reset if new logout-timeout specified
    - Can be used both as an idle timer and as a time limit

Future plan

- Currently, a part of our new auth proposal
- If peoples consider it useful,
  I'll write a separate draft to IETF

What's more?

- Please imagine combination of *your* webapp. with HTTP auth and my proposal
  - Our goal is to support almost all web. apps with small alteration of UI/page-flow design
  - If you think some feature is not implementable, please let me know
  - Proposals are welcome

Contact

- Yutaka OIWA
  - RCIS, AIST
    - Research Center for Information Security
    - National Institute of Advanced Industrial Science and Technology
    - https://staff.aist.go.jp/y.oiwa/ or http://bit.ly/yoiwa-aist-en
  - E-mail: y.oiwa@aist.go.jp
  - Twitter: @yoiwa (Japanese: @yoiwa_j)