Università di Genova

PHP (4)

Marina Ribaudo, marina.ribaudo@unige.it

Access control

"You may need to introduce access control to your system for a few reasons. The first and or most obvious reason is to **allow some people to see (or do) what you want them to see/do** while keeping the others out. However, you must also **know who did what and when**, so that they can be held accountable for their actions."

https://www.feistyduck.com/library/apache-security/online/apachesc-CHP-7.html

Access control

- **Identification**
  - L’utente presenta la sua identità
- **Authentication**
  - Verifica se l’utente può accedere al sistema
- **Authorization**
  - Verifica se l’utente può accedere ad una risorsa particolare
- **Accountability**
  - Capacità di dire chi ha avuto accesso ad un risorsa e quando e se la risorsa è stata modificata

Access control

- HTTP è stateless e nasce per lo scambio di risorse

- Problemi di **autenticazione e autorizzazione**
  - Basic authentication
  - Digest authentication
  - Form-based authentication

Access control

Altre tecniche, più o meno semplici da implementare, sono

- URL nascoste
  // si usava all'inizio del web, oggi non va più bene!
- Controllo basato sull'indirizzo IP o sul nome di dominio
  // usato nelle intranet aziendali

Basic Authentication

I **controlli** basati sull'identità dell'utente possono essere **demandati al server web** sfruttando la **Basic Authentication di HTTP**

Quando si cerca di accedere a informazioni protette con Basic Authentication

- il browser visualizza una **finestra di dialogo** che richiede le credenziali all'utente
- le credenziali vengono scambiate tra browser e server per tutta la durata dell'interazione

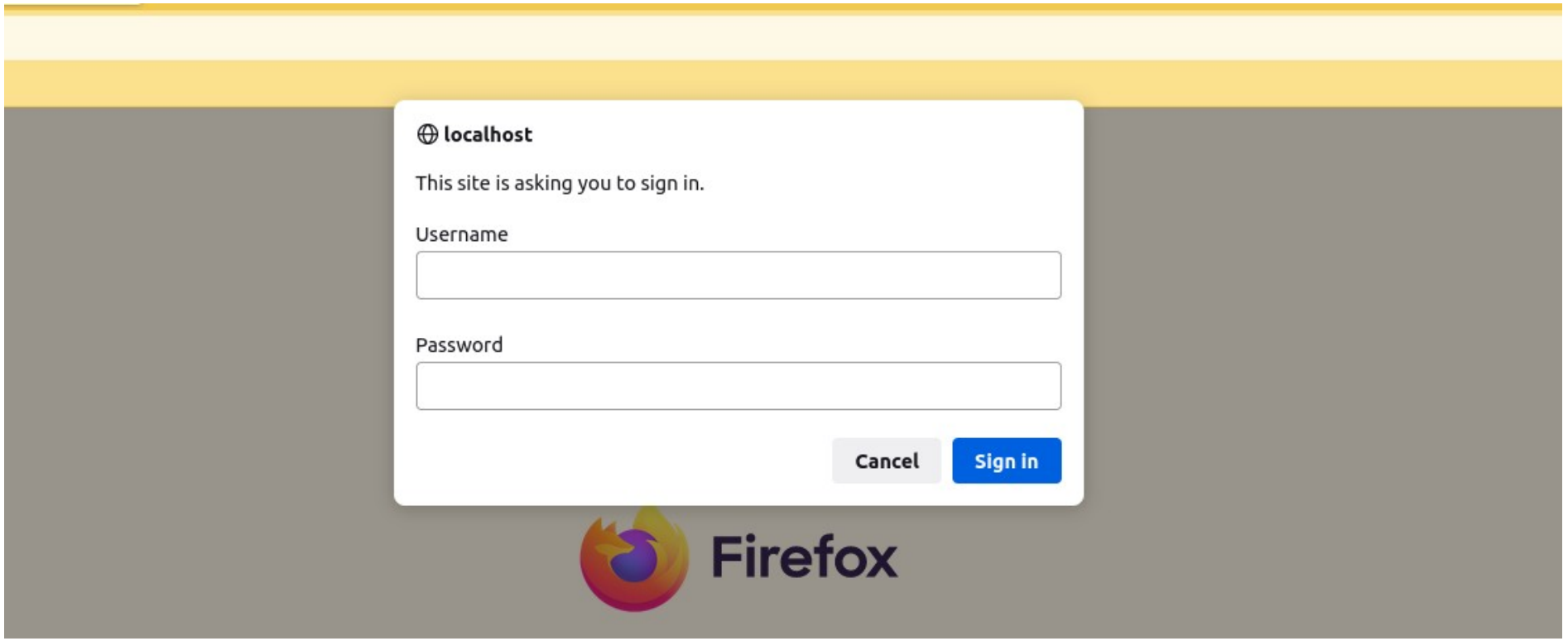
Basic Authentication: response

HTTP/1.1 401 Authorization Required
Date: Mon, 07 Nov 2022 15:01:03 GMT
Server: Apache/2.4.41 (Ubuntu)
WWW-Authenticate: Basic realm="Protect this directory"
Content-Length: 456
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=iso-8859-1

.....

Basic Authentication: response

- Visualizzazione nel browser

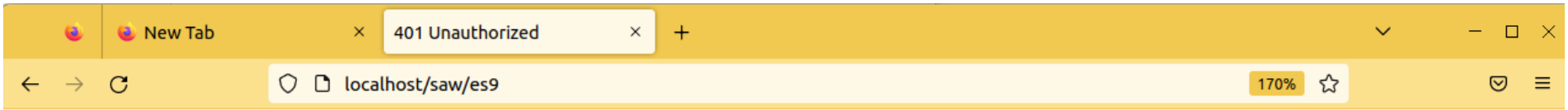


Basic Authentication: response

HTTP/1.1 401 Unauthorized
Date: Mon, 07 Nov 2022 16:07:33 GMT
Server: Apache/2.4.41 (Ubuntu)
WWW-Authenticate: Basic realm="Protect this directory"
Content-Length: 456
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=iso-8859-1

Basic Authentication: response

- Accesso negato



Unauthorized

This server could not verify that you are authorized to access the document requested. Either you supplied the wrong credentials (e.g., bad password), or your browser doesn't understand how to supply the credentials required.

Apache/2.4.41 (Ubuntu) Server at localhost Port 80

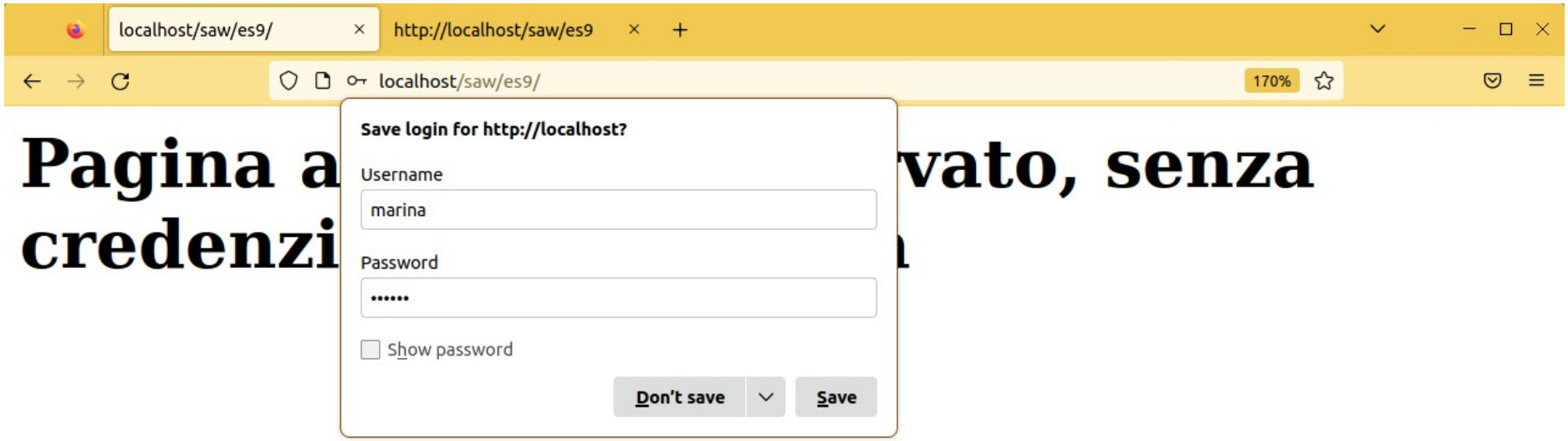
Basic Authentication: request

GET /saw/es9/ HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:106.0) Gecko/20100101 Firefox/106.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br

Authorization: Basic bWFyaW5hOm1hcmluYQ==

Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
If-Modified-Since: Mon, 07 Nov 2022 14:03:41 GM
If-None-Match: "45-5ece1e4103d6a-gzip"

Basic Authentication: request



Basic Authentication: how to

Si può creare un file di testo **.htaccess** nella directory che si vuole proteggere e specificare delle direttive

Vedi: http://httpd.apache.org/docs/current/howto/auth.html

```
AuthType Basic
AuthName “Restricted Area for My Server”
AuthUserFile  /absolutepath/nomefilepwd
Require valid-user
```

** invece di scrivere valid-user, si possono elencare gli utenti che possono accedere, oppure un gruppo di utenti*

Basic Authentication: how to

Le **credenziali** degli utenti possono essere **salvate in un file di testo**, specificando la direttiva

AuthBasicProvider **file**

Con il file di testo ci sono problemi di efficienza (il file viene letto per ogni accesso alle risorse nell'area protetta) e si possono anche usare formati di storage come **dbm** e **dbd**, o **LDAP**

AuthBasicProvider **dbm**

AuthBasicProvider **ldap**

Basic Authentication: how to

Il file delle password si crea con il comando **htpasswd** (-c si deve usare solo la prima volta che si crea il file delle password)

htpasswd –c nomefilepwd nomeuser
New password: **********
Re-type new password: **********

Nota: Il file .htaccess e quello delle password devono essere **leggibili dal web server** (chmod 644)

Basic Authentication: how to

Perché tutto funzioni bisogna **abilitare un modulo di Apache** (mod_auth) e **modificare la direttiva AllowOverride** nel file di configurazione di Apache

Per chi fosse interessato/a
https://www.keycdn.com/support/htaccess-not-working

Il file .htaccess permette di specificare anche altre configurazioni per i server web che usano Apache

Per chi fosse interessato/a
https://www.keycdn.com/support/popular-htaccess-examples

Basic Authentication

- Basic authentication ha un certo numero di **svantaggi**
  - Le credenziali sono trasmesse in base64
  - Non esiste la possibilità di fare logout (si deve chiudere il browser)
  - Il form di login non può essere personalizzato
  - I proxy HTTP possono estrarre le credenziali dal traffico di rete

Basic Authentication

- Per ogni accesso a una pagina/risorsa protetta, il server deve leggere le credenziali dalla richiesta HTTP e poi accedere a
  - file .htaccess per le direttive
  - file/database/LDAP per la password
- Se il numero degli utenti cresce, questo meccanismo di autenticazione diventa inefficiente

Digest Authentication

- Permette l’autenticazione senza inviare le credenziali in base64
- Il server invia al client una challenge e il client risponde inviando un **hash** della soluzione della challenge e della password
- Il server verifica se il client possiede la password corretta

Digest Authentication

*"This module implements **HTTP Digest Authentication** (RFC2617), and provides an alternative to mod_auth_basic where the **password is not transmitted as cleartext**."*

It uses MD5… The MD5 calculations used in HTTP digest authentication is intended to be "one way", meaning that it should be difficult to determine the original input when only the output is known. If the password itself is too simple, however, then it may be possible to test all possible inputs and find a matching output (a brute-force attack) – perhaps aided by a dictionary or suitable look-up list, which for MD5 is readily available

*"Therefore, using **basic auth** and encrypting the whole connection using **mod_ssl** is a much better alternative."*

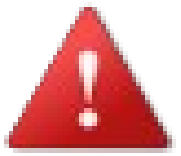
https://httpd.apache.org/docs/2.4/mod/mod_auth_digest.html

Form-based authentication

- Invece di lavorare a livello di protocollo HTTP si lavora a livello di applicazione web
- Come abbiamo visto, per le pagine ad accesso riservato, in risposta ad una richiesta da parte di un utente non ancora autenticato, l'applicazione restituisce il form per il login

Form-based authentication

“Applications are often given far less testing than the web server and potentially contain more security issues. Some files in the application, for example, may not be protected at all. Images are almost never protected. Often applications contain large amounts of code that are executed prior to authentication. The chances of an intruder finding a hole are much higher when application-level authentication is used.”



https://www.feistyduck.com/library/apache-security/online/apachesc-CHP-7.html