htpasswd – The file to store passwords

Htpasswd files are used when password protecting a website or a directory using HTTP Authentication and Apache's htaccess files.

The htpasswd file contains username in plain text (unencrypted) and a hashed (encrypted) password. Here's an example:

```
andreas:$apr1$dHjB0/..$mkTTbqwpK/0h/rz4ZeN8M0
john:$apr1$IHaD0/..$N9ne/Bqnh8.MyOtvKU56j1
```

Each line contains a username and a password separated by a colon ":". You can not see the actual passwords as they are hashed (encrypted) using a complex algorithm. The default algorithm is different from platform to platform. On Windows the passwords are hashed using MD5, and on Linux its based on a system function called "crypt()". The htpasswd generator on this site uses MD5 which means that the hashed passwords can be used on both Windows and Linux.

Filename

Normally the htpasswd file is named .htpasswd, but you are actually free to name your password file what every you like. It is perfectly valid to use a name like "passwords.txt" which may seem more appealing to Windows users. However there is one catch. Apache is usually configured to prevent access to .ht* files – starting with ".ht". If you name your password file "passwords.txt", a user can access it, and retrieve all valid usernames. Since the passwords are hashed he can't use them directly, but it will help him gain access using brute force. **It is therefore recommended to name a password file .htpasswd.**

Generating password

Hashed passwords can be generated with the command-line tool htpasswd (htpasswd.exe on Windows) which is part of a normal Apache installation. You can also create passwords using the htpasswd generator on this site, or create passwords yourself using PHP.

Various Options used:

Options

-b
Use batch mode; *i.e.*, get the password from the command line rather than prompting for it. This option should be used with extreme care, since **the password is clearly visible** on the command line. For script use see the -i option. Available in 2.4.4 and later.

-i
Read the password from stdin without verification (for script usage).

-c
Create the passwdfile. If passwdfile already exists, it is rewritten and truncated. This option cannot be combined with the -n option.

-n
Display the results on standard output rather than updating a file. This is useful for generating password records acceptable to Apache for inclusion in non-text data stores. This option changes the syntax of the command line, since the passwdfile argument (usually the first one) is omitted. It cannot be combined with the -c option.

-m
Use MD5 encryption for passwords. This is the default (since version 2.2.18).

-B
Use bcrypt encryption for passwords. This is currently considered to be very secure.

-C
This flag is only allowed in combination with -B (bcrypt encryption). It sets the computing time used for the bcrypt algorithm (higher is more secure but slower, default: 5, valid: 4 to 31).

-d
Use crypt() encryption for passwords. This is not supported by the httpd server on Windows and Netware. This algorithm limits the password length to 8 characters. This algorithm is **insecure** by today's standards. It used to be the default algorithm until version 2.2.17.

-s
Use SHA encryption for passwords. Facilitates migration from/to Netscape servers using the LDAP Directory Interchange Format (ldif). This algorithm is **insecure** by today's standards.

-p
Use plaintext passwords. Though htpasswd will support creation on all platforms, the httpd daemon will only accept plain text passwords on Windows and Netware.

-D
Delete user. If the username exists in the specified htpasswd file, it will be deleted.

-v
Verify password. Verify that the given password matches the password of the user stored in the specified htpasswd file. Available in 2.4.5 and later.

passwdfile
Name of the file to contain the user name and password. If -c is given, this file is created if it does not already exist, or rewritten and truncated if it does exist.

username
The username to create or update in passwdfile. If username does not exist in this file, an entry is added. If it does exist, the password is changed.

password
The plaintext password to be encrypted and stored in the file. Only used with the -b flag.

```
docker run -d \
  -p 5000:5000 \
  --restart=always \
  --name myregistry6 \
  -v `pwd`/auth:/auth \
  -e "REGISTRY_AUTH=htpasswd" \
  -e "REGISTRY_AUTH_HTPASSWD_REALM=Registry Realm" \
  -e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd \
  -v `pwd`/certs:/certs \
  -e REGISTRY_HTTP_TLS_CERTIFICATE=/certs/dockerrepo.crt \
  -e REGISTRY_HTTP_TLS_KEY=/certs/dockerrepo.key \
  Registry:2
```

```
curl --insecure -u "test:password"
http://myregistrydomain.com:5000/v2/_catalog
```

```
wget --no-check-certificate --http-user=test
--http-password=password
http://myregistrydomain.com:5000/v2/_catalog
```

```
docker container run --rm -it --name ucp -v
/var/run/docker.sock:/var/run/docker.sock docker/ucp:2.2.4  install
--host-address 172.31.24.145 --interactive
```