Curs 2023 - 2024

Autenticació en Nginx

Guillermo Vidal Frasquet

Desplegament d'Aplicacions Web

Pràctica

IES Jaume II El Just
Tavernes de la Valldigna

Continguts

1 Autenticación en Nginx

1.1 Requisits abans de començar la pràctica

Atenció, molt important abans de començar

- La pràctica 2.1 ha d'estar funcionant correctament.
- No començar la pràctica abans de tindre la 2.1 **funcionant i comprovada**.

1.2 Introducció

En el context d'una transacció HTTP, l'autenticació d'accés bàsica és un mètode dissenyat per a permetre a un navegador web, o un altre programa client, proveir credencials en la forma d'usuari i contrasenya quan se li sol·licita una pàgina al servidor.

L'autenticació bàsica, com el seu nom l'indica, és la forma més bàsica d'autenticació disponible per a les aplicacions Web. Va ser definida per primera vegada en l'especificació HTTP en si i no és de cap manera elegant, però compleix la seua funció.

Aquest tipus d'autenticació és el tipus més simple disponible però té importants problemes de seguretat que no la fan recomanable en moltes situacions. No requereix l'ús ni de cookies, ni d'identificadors de sessió, ni de pàgina d'ingrés.

1.3 Paquets necessaris

Per a aquesta pràctica podem utilitzar l'eina **openssl** per a crear les contrasenyes.

En primer lloc hem de comprovar si el paquet està instal·lat:

```
1 dpkg -l | grep openssl
```

I si no ho estiguera, instal·lar-ho.

1.4 Creació d'usuaris i contrasenyes per a l'accés web

Crearem un arxiu ocult anomenat **.htpasswd** en el directori de configuració `/etc/nginx` on guardar els nostres usuaris i contrasenyes (la -c és per a crear l'arxiu):

```
1 sudo sh -c "echo -n 'vostre_nom:' >> /etc/nginx/.htpasswd"
```

Ara crearem un **password** xifrat per a l'usuari:

```
sudo sh -c "openssl passwd -apr1 >> /etc/nginx/.htpasswd"
```

Aquest procés es podrà repetir per a tants usuaris com faça falta.

- Crea dos usuaris, un amb el teu nom i un altre amb el teu primer cognom.
- Comprova que l'usuari i la contrasenya apareixen xifratges en el fitxer:

```
cat /etc/nginx/.htpasswd
```

1.5 Configurant el servidor Nginx per a usar autenticació bàsica

Editarem la configuració del **server block** sobre el qual volem aplicar la restricció d'accés. Utilitzarem per a aquesta autenticació el lloc web de **Perfect Learn**:

```
sudo nano /etc/nginx/sites-available/nom_web
```

> Recorda que un **server block** és cadascun dels dominis (**server {...}** dins de l'arxiu de configuració) d'algun dels llocs web que hi ha en el servidor.

Hem de decidir quins recursos estaran protegits. Nginx permet afegir restriccions a nivell de servidor o en un **location** (directori o arxiu) específic. Per al nostre exemple, protegirem el **document root** (l'arrel, la pàgina principal) del nostre lloc.

Utilitzarem la directiva `auth_basic` dins del **location** i li posarem el nom al nostre domini que serà mostrat a l'usuari en sol·licitar les credencials. Finalment, configurem Nginx perquè utilitze el fitxer que prèviament hem creat amb la directiva `auth_basic_user_file`:

```
server {
    listen 80;
    listen [::]:80;

    root /var/www/guillermo/html/perfect-learn;
    index index.html index.htm index.nginx-debian.html;

    server_name tasca;
    location / {
        auth_basic  "Àrea restringida";
        auth_basic_user_file    /etc/nginx/.htpasswd;
        try_files $uri $uri/ =404;
    }
}
```

Una vegada acabada la configuració, reiniciem el servei perquè aplique la nostra política d'accés.

```
sudo systemctl restart nginx
```

1.5.1 Provant la nova configuració

Comprovació 1
Comprova des de la teua màquina física/amfitrió que pots accedir a `http://nom-lloc-web` i que sol·licita autenticació. **No t'autentiques**.

Comprovació 2
Comprova que si decideixes cancel·lar l'autenticació, se't negarà l'accés al lloc amb un error. Quin error és?

Lliura les captures de pantalla de les comprovacions, en les tasques habilitades en aules.

Una vegada us autentiqueu amb èxit, el navegador guardarà aquesta autenticació i no tornarà a demanar-vos usuari/contrasenya.
Arribats a aquest punt, si voleu tornar a provar d'autenticar-vos, haureu d'obrir-vos una **Nova finestra privada** del navegador o esborrar les **cookies**.

1.5.2 Tasques

Tasca 1

- Intenta entrar primer amb un usuari erroni i després amb un altre correcte. Pots veure tots els successos i registres en els logs `access.log` i `error.log`.
- Adjunta una captura de pantalla dels logs on es veja que intentes entrar primer amb un usuari invàlid i amb un altre vàlid. Indica on podem veure els errors d'usuari invàlid o no trobat, així com on podem veure el número d'error que us apareixia abans.

Lliura la tasca en aules.

Quan hem configurat el següent bloc:

```
location / {
    auth_basic  "Àrea restringida";
    auth_basic_user_file    /etc/nginx/.htpasswd;
    try_files $uri $uri/ =404;
}
```

L'autenticació s'aplica al directori/arxiu que li indiquem en la declaració del **location** i que en aquest cas l'arrel `/`.

Així doncs, aquesta restricció s'aplica al directori arrel o base on resideixen els arxius del lloc web i que és:

```
/var/www/guillermo/html/perfect-learn
```

I a tots els arxius que hi ha dins, ja que no hem especificat cap en concret.

Ara bé, provarem d'aplicar autenticació només a una part de la web. Intentarem que només es necessite autenticació per a entrar a la part de **portfoli**:

Aquesta secció es correspon amb l'arxiu `contact.html` dins del directori arrel.

Tasca 2
Esborra les dues línies que fan referència a l'autenticació bàsica en el **location** del directori arrel. Després d'això, afig un nou **location** davall amb l'autenticació bàsica per a l'arxiu/secció `contact.html` únicament.

Fixeu-vos que heu d'anar amb compte perquè l'última línia de l'arxiu ha de ser **}** que tanca la primera línia **server {** de l'arxiu.

1.6 Combinació de l'autenticació bàsica amb la restricció d'accés per IP

L'autenticació bàsica HTTP pot ser combinada de manera efectiva amb la restricció d'accés per adreça IP. Es poden implementar dos escenari:

- Un usuari ha de complir totes dues coses, estar autenticat i tindre una IP vàlida.
- Un usuari deu o bé estar autenticat, o bé tindre una IP vàlida.

Vegem com ho faríem:

1. Com permetre o denegar accés sobre una IP concreta (directives **allow** i **deny**, respectivament). Dins del **block server** o arxiu de configuració del domini web, que recordeu està en el directori `sites-available`:

```
location /api {
    #...
    deny  192.168.1.2;
    allow 192.168.1.1/24;
    allow 127.0.0.1;
    deny  all;
}
```

   L'accés es garantirà a la IP `192.168.1.1/24`, excloent a l'adreça `192.168.1.2`.

   Cal tindre en compte que les directives **allow** i **deny** s'aniran aplicant en l'ordre en el qual apareixen en l'arxiu.

   Ací apliquen sobre la `location /api` (això és només un exemple d'un hipotètic directori o arxiu), però podrien aplicar sobre qualsevol, inclosa tot el lloc web, la `location /`.

L'última directiva **deny all** vol dir que per defecte denegarem l'accés a tothom. Per això cal posar els **allow** i **deny** més específics just abans d'aquesta, perquè en avaluar-se en ordre d'aparició, si els posàrem davall es denegaria l'accés a tothom, ja que **deny all** seria el primer que s'avaluaria.

2. Combinar la restricció IP i l'autenticació HTTP amb la directiva **satisfy**.

   Si establim el valor de la directiva a **all**, l'accés es permet si el client satisfà totes dues condicions (IP i usuari vàlid). Si ho establim a **any**, l'accés es permet si se satisfà almenys una de les dues condicions.

```
location /api {
    #...
    satisfy all;

    deny  192.168.1.2;
    allow 192.168.1.1/24;
    allow 127.0.0.1;
    deny  all;

    auth_basic           "Administrator's Area";
    auth_basic_user_file conf/htpasswd;
}
```

1.6.1 Tasques

Tasca 3

Configura Nginx perquè no deixe accedir amb la IP de la màquina amfitriona al directori arrel d'un dels teus dos webs. Modifica el seu **server block** o arxiu de configuració. Comprova com es denega l'accés:

- Mostra la pàgina d'error en el navegador.
- Mostra el missatge d'error d'`error.log`.

Tasca 4

Configura Nginx perquè des de la teua màquina amfitriona s'haja de tindre tant una IP vàlida com un usuari vàlid, totes dues coses alhora, i comprova que sí que pot accedir sense problemes.