

Galetes (Cookies) en PHP

Gestió de dades persistents al client

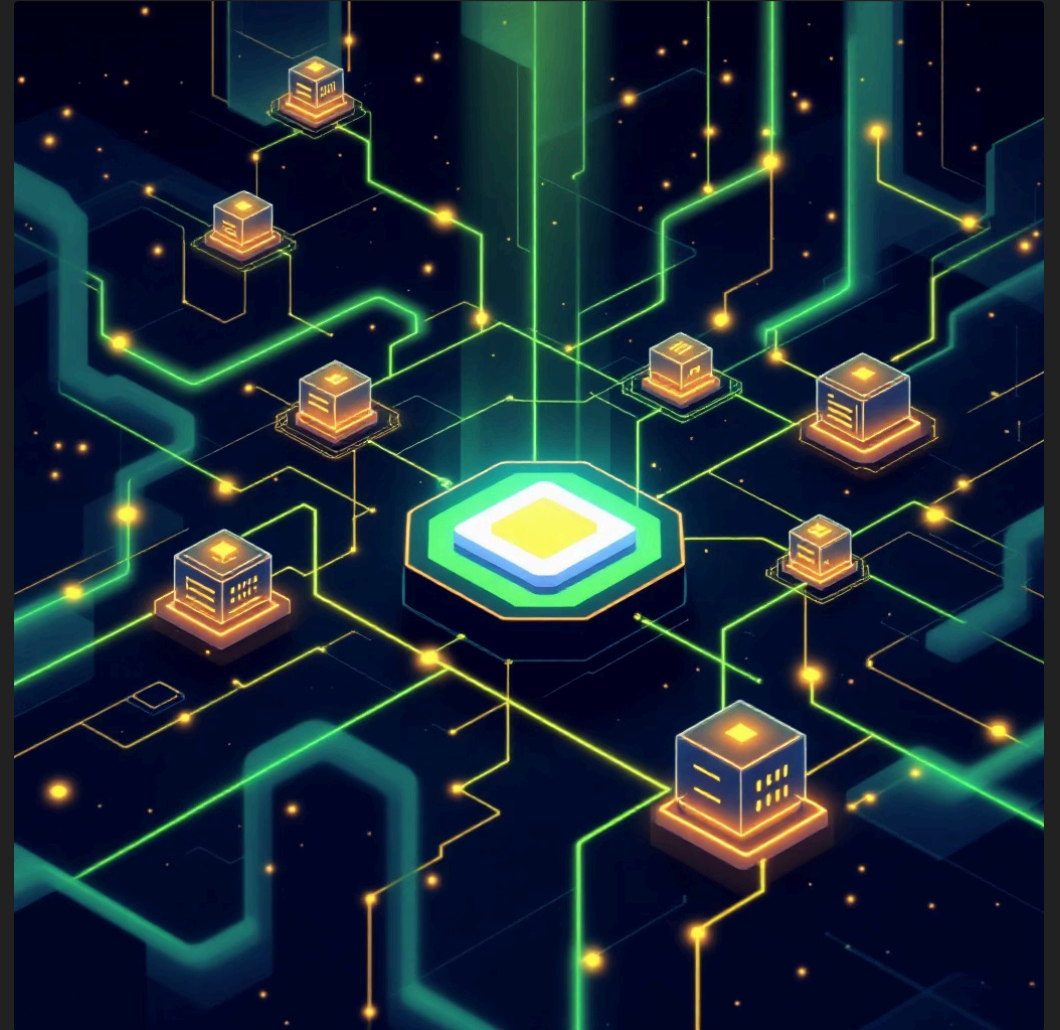
Curs: ASIX2 - SM9 Implantació d'aplicacions web



Què són les galetes?

Les galetes són petits fitxers de text emmagatzemats al navegador del client que resolen un problema fonamental: HTTP és un protocol sense estat (stateless). Això significa que cada petició és independent i el servidor no "recorda" les anteriors.

Les galetes permeten mantenir informació entre diferents sol·licituds, creant una experiència d'usuari coherent i personalitzada. S'emmagatzemen al navegador i es transmeten automàticament amb cada petició HTTP al mateix domini.



Característiques i usos principals



Emmagatzematge

Mida limitada de ~4KB per galeta. Cada domini pot tenir 20-50 galetes màxim.



Caducitat

Temps de vida configurable. Poden caducar en tancar el navegador o persistir durant dies, mesos o anys.



Seguretat

Opcions de seguretat: només HTTPS (secure) i no accessible via JavaScript (HttpOnly).



Transmissió

S'envien automàticament amb cada petició HTTP al servidor corresponent.

Les galetes són essencials per recordar preferències d'usuari, mantenir sessions actives, fer seguiment analític, personalitzar continguts i gestionar carrets de compra en comerços electrònics.

La funció `setcookie()`

Per crear una galeta en PHP, utilitzem la funció `setcookie()`. És fonamental cridar-la abans de qualsevol sortida HTML.

```
setcookie(nom, valor, caducitat, ruta, domini, segur, httponly);
```

01	02	03
nom	valor	caducitat
Identificador únic de la galeta (obligatori)	Dada que es vol emmagatzemar	Timestamp UNIX que defineix quan expira
04	05	06
ruta	domini	segur
Directorí on la galeta està disponible	Domini on la galeta és vàlida	Només transmetre via HTTPS (true/false)
07		
httponly		
Inaccessible des de JavaScript (true/false)		

Exemples de creació i caducitat

Exemples de creació

Galeta simple:

```
setcookie("nom_usuari", "Joan");
```

Amb caducitat (1 hora):

```
setcookie("nom_usuari", "Joan",  
time() + 3600);
```

Amb seguretat completa:

```
setcookie("nom", "Joan",  
time() + 3600, "/", "",  
false, true);
```

Càlcul de temps

Format: Timestamp UNIX (segons des de 01/01/1970)

- `time() + 3600` → 1 hora
- `time() + 86400` → 1 dia
- `time() + (7*24*60*60)` → 1 setmana
- `time() + (30*24*60*60)` → 1 mes

❏ **Important:** Si la caducitat és 0 o s'omet, la galeta caduca en tancar el navegador.

Lectura i accés segur

Per llegir galetes, utilitzem l'array superglobal `$_COOKIE`. És crucial sempre verificar l'existència abans d'accedir-hi.

Comprovació bàsica

```
if (isset($_COOKIE['nom_usuari']))  
{  
    $nom =  
    $_COOKIE['nom_usuari'];  
    echo "Hola, " . $nom;  
} else {  
    echo "No hi ha galeta";  
}
```

Funció d'accés segur

```
function obtenir_galeta($nom,  
$per_defecte = "") {  
    if (isset($_COOKIE[$nom])) {  
        return $_COOKIE[$nom];  
    }  
    return $per_defecte;  
}  
  
$nom =  
obtenir_galeta('nom_usuari',  
'Anònim');
```

Prevenició d'atacs XSS

```
echo htmlspecialchars(  
    $_COOKIE['nom'],  
    ENT_QUOTES,  
    'UTF-8'  
);
```

Escapar sempre les sortides per prevenir Cross-Site Scripting.

Modificació i esborrament

Modificar galetes

Per modificar una galleta existent, cridem `setcookie()` amb el mateix nom:

```
// Galleta inicial
setcookie("comptador", "1",
time() + 3600);

// Modificació
if
(isset($_COOKIE['comptador']))
{
    $nou =
    $_COOKIE['comptador'] + 1;
    setcookie("comptador", $nou,
time() + 3600);
}
```

📌 El canvi no es veu fins a la següent càrrega de pàgina.

Esborrar galetes

Establir la caducitat al passat:

```
setcookie("nom_usuari", "",
time() - 3600);
```

Funció auxiliar:

```
function
esborrar_galeta($nom) {
    setcookie($nom, "", time() -
3600, "/");
}

esborrar_galeta('nom_usuari');
```

Esborrar totes:

```
foreach ($_COOKIE as $nom =>
$valor) {
    setcookie($nom, "", time() -
3600, "/");
}
```



Errors comuns i consideracions

⚠ Headers Already Sent

Incorrecte: HTML abans de
setcookie()

```
<html>
<?php setcookie("nom", "valor");
?>
```

Correcte: setcookie() primer

```
<?php setcookie("nom", "valor");
?>
<html>
```

Regla: setcookie() ha d'anar abans de qualsevol sortida HTML.

🕒 Disponibilitat immediata

```
setcookie("prova", "valor");

// Això serà FALSE
if (isset($_COOKIE['prova'])) {
    echo "Existeix"; // No
    s'executarà
}
```

Les galetes estan disponibles en la
següent sol·licitud. Solució: redirigir
després de crear-les.

🔒 Validació obligatòria

```
if (isset($_COOKIE['edat'])) {
    $edat =
    filter_var($_COOKIE['edat'],
        FILTER_VALIDATE_INT);

    if ($edat !== false && $edat > 0)
    {
        echo "Edat: " . $edat;
    }
}
```

Mai confiar directament en les
galetes. El client pot modificar-les!

Seguretat i limitacions

🚫 No emmagatzemar

- Contrasenyes
- Dades bancàries
- Informació confidencial
- Dades personals sensibles

Les galetes poden ser interceptades i modificades pel client.

📦 Limitacions tècniques

- Mida màxima: ~4KB per galeta
- Nombre màxim: 20-50 per domini
- Poden ser desactivades per l'usuari
- Poden ser modificades al client

✅ Bones pràctiques

- Usar `httponly` (prevenir XSS)
- Usar `secure` per HTTPS
- Establir caducitat adequada
- Validar i sanititzar sempre
- Especificar ruta clarament



Recursos i contacte

Resum clau

- Les galetes emmagatzemen dades al navegador del client
- Usar `setcookie()` abans de qualsevol sortida HTML
- Llegir amb l'array superglobal `$_COOKIE`
- Esborrar establint la caducitat al passat
- Sempre validar les dades rebudes
- Tenir en compte limitacions de mida i seguretat

Eines de depuració

Veure totes les galetes:

```
print_r($_COOKIE);
```

Navegadors: Chrome DevTools (Application → Cookies), Firefox (Inspector → Emmagatzematge)

Referències

- [Manual oficial PHP](#)
- [MDN Web Docs: HTTP Cookies](#)

? Preguntes?

Contacte:

jordi.binefa@fje.edu

