CSRF (Cross-Site Request Forgery)

1. Definición

O **CSRF** é un ataque no que un usuario autenticado nun sitio web (por exemplo, o seu banco en liña) **é inducido a realizar accións non desexadas** sen o seu consentimento, normalmente mediante a visita a outra páxina maliciosa.

O atacante **non rouba credenciais directamente**, pero aproveita a confianza que o sitio ten no navegador do usuario (cookies de sesión, tokens de sesión, etc.).

Cómo funciona

O ataque basease en **aproveitar a sesión válida** do usuario:

1. O usuario inicia sesión nun sitio web seguro (por exemplo `banco.com`) e ten unha **cookie de sesión activa**.
2. Sen pechar sesión, o usuario visita unha páxina maliciosa controlada polo atacante (`evil.com`).
3. Esa páxina maliciosa envía unha **solicitude HTTP ao sitio seguro**.
    - Exemplo: `POST https://banco.com/transferir` con parámetros de transferencias.
4. O navegador **engade automáticamente as cookies de sesión** do usuario na solicitude.
5. O servidor recibe a solicitude **como se fose válida**, e executa a acción sen que o usuario a autorizase conscientemente.

Exemplo práctico

Situación real: transferencia de cartos

```
<!-- Páxina maliciosa -->
<img src="https://banco.com/transferir?destino=12345&cantidade=1000">
```

- O navegador do usuario interpreta o `<img>` e envía unha petición GET a `banco.com` con a súa sesión activa.
- O banco procesa a transferencia sen saber que o usuario non quere facela.

3. Que pode facer un atacante con CSRF

Dependendo da aplicación web, un CSRF pode permitir:

- Realizar transaccións financeiras (transferencias de cartos).
- Cambiar contrasinais ou correos electrónicos de contas.
- Modificar configuracións de usuario.
- Engadir ou eliminar datos (posts, comentarios, arquivos, etc.).

- Calquera acción que o usuario poida facer e que se autentique mediante cookies ou tokens non protexidos.

Nota: Non permite ler datos directamente; é máis un ataque de “acción non autorizada” que de roubo de información.

4. Como se impide

Para evitar CSRF, as boas prácticas inclúen:

1. **Tokens CSRF**
    - Cada formulario xera un token único e secreto, asociado á sesión do usuario.
    - O servidor verifica que o token recibido coincide co esperado.
    - Así, unha petición externa sen token válido é rechazada.
2. **Cabeceras `SameSite` en cookies**
    - Configurar as cookies de sesión como `SameSite=Lax` ou `Strict` fai que o navegador **non envie cookies** en solicitudes de terceiros.
3. **Dobre verificación para accións críticas**
    - Pedir reconfirmación de contrasinais ou códigos temporais (2FA) antes de cambios sensibles.
4. **Validación do método HTTP**
    - Usar `POST` para accións que cambian estado, nunca `GET`.
    - Verificar que as solicitudes se realizan desde a propia aplicación.

Resumo

1. Usuario autenticado → banco.com (cookie activa)
2. Usuario visita evil.com → evil.com envía solicitude ao banco
3. Navegador engade cookies → servidor executa acción
4. Prevención: token CSRF, SameSite, POST, validación

CSRF é un ataque que **usa a confianza dun sitio web no navegador dun usuario autenticado** para facer accións non desexadas. A prevención céntrase en **tokens únicos, cookies restrinxidas e validación rigorosa das solicitudes**.