Cross-Site Request Forgery

1 min

Cross-Site Request Forgery (CSRF) is a web-specific

Preview: Docs Loading link description

vulnerability

in which an attacker tricks a victim into making malicious requests to a website where that victim is authenticated.

CSRF usually involves some element of

Preview: Docs Loading link description

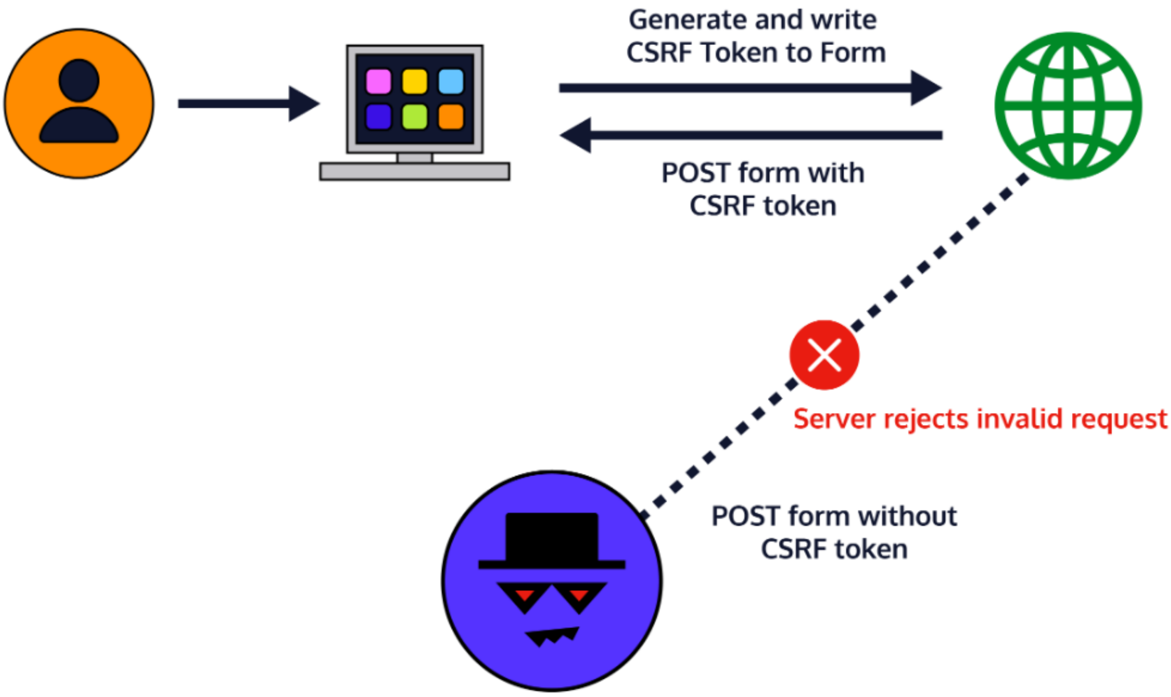
social engineering

, but the victim doesn’t necessarily have to click a link. All that matters is that the request comes from the victim’s web browser, where they are logged in.

How CSRF Works:

1. Victim’s Session: The victim logs into a trusted website (e.g., www.bank.com) and has an active session with authenticated credentials (usually maintained by a session cookie).
2. Malicious Request: The victim unknowingly visits a malicious website or clicks on a malicious link while still logged into the trusted website. This malicious site sends a crafted request to the trusted website (e.g., transferring money from the victim’s account).
3. Automatic Authentication: Because the victim is already authenticated with the trusted website, their browser automatically attaches the session cookie to the forged request, making it appear as a legitimate request from the victim.
4. Execution of Malicious Action: The trusted website receives the request with the valid session cookie and processes it, performing the action without the victim’s consent or knowledge.

XSS vulnerabilities exacerbate CSRF vulnerabilities, and properly defending against CSRF requires first defending against XSS.


Generate and write
CSRF Token to Form
POST form with
CSRF token
Server rejects invalid request
POST form without
CSRF token