What is an open redirect?

An open redirect is a vulnerability where an application redirects users to a URL taken from user input without validating it, so an attacker crafts a link on the trusted domain that bounces the victim to a malicious site. On its own it mainly enables convincing phishing (the link starts with your real domain), but it also helps bypass allow-lists and amplifies attacks like SSRF and OAuth token theft. The fix is to never redirect to a raw user-supplied URL, using an allow-list or relative paths instead.

HOW IT WORKS

01 How it works and example

The attacker crafts a link using the site's redirect parameter:

- `https://trusted.example/login?next=https://evil.example` sends the user to the attacker after login.
- Bypasses of naive checks: `//evil.example` (protocol-relative), `https://trusted.example.evil.example`, whitespace/encoding tricks, or `@` confusion (`https://trusted.example@evil.example`).
- Chained into OAuth/OIDC flows, an open redirect can steal authorization codes or tokens.

Examples shown for defensive context.

> BORROWED TRUST
>
> *The danger is not the redirect itself, it is that the link starts on your trusted domain, so users and filters trust it. That is why open redirects supercharge phishing and help defeat URL allow-lists.*

HOW TO DEFEND

- Avoid user-supplied redirect targets. Prefer relative paths or a server-side mapping (a short token to a known destination).
- Allow-list destinations (exact hosts/paths) and reject anything else, after decoding.
- Reject protocol-relative and absolute external URLs where only internal redirects are intended.
- Show an interstitial for any unavoidable external redirect.
- Test every redirect parameter, including OAuth `redirect_uri` handling.

SOURCES

[1] OWASP: Unvalidated Redirects and Forwards

[2] MITRE CWE-601: URL Redirection to Untrusted Site

[3] OWASP Top 10