

# Plan omówienia zagadnień

- 1 Co to jest same-origin policy?
- 2 XSS
- 3 CSRF
- 4 Co to jest CORS i ACAO response header?

# Co oznacza, że dwa zasoby pochodzą z tego samego źródła? (Origin)

Dwa zasoby są uważane za pochodzące z tego samego źródła, jeśli mają ten sam protokół, hosta i port.

The following table gives examples of origin comparisons with the URL `http://store.company.com/dir/page.html`:

URL	Outcome	Reason
<code>http://store.company.com/dir2/other.html</code>	Same origin	Only the path differs
<code>http://store.company.com/dir/inner/another.html</code>	Same origin	Only the path differs
<code>https://store.company.com/page.html</code>	Failure	Different protocol
<code>http://store.company.com:81/dir/page.html</code>	Failure	Different port ( <code>http://</code> is port 80 by default)
<code>http://news.company.com/dir/page.html</code>	Failure	Different host

# Co to jest same-origin policy?

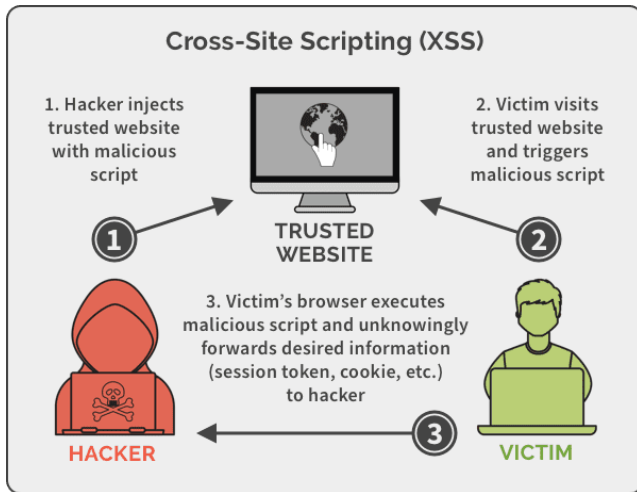
Same-origin policy (SOP) to mechanizm bezpieczeństwa, który ogranicza interakcję dokumentu lub skryptu załadowanego przez jedno źródło z zasobem z innego źródła.

Gdyby przeglądarka podchodziła do tej polityki rygorystycznie, to:

- Nie można by zamieścić na stronie z Originem A obrazków, skryptów, arkuszy CSS z Originu B
- Nie można by wywoływać zapytań HTTP z Originu A do Originu B
- Nie można by zapisywać i odczytywać ciasteczek Originu A, będąc na stronie innego Originu B

# XSS (Cross-Site Scripting)

XSS to luka w zabezpieczeniach aplikacji internetowej, która pozwala atakującemu na wstrzyknięcie złośliwego kodu (zwykle JavaScript) do treści wyświetlanej innym użytkownikom. Atakujący może wykorzystać XSS do kradzieży danych użytkowników, przejęcia sesji, manipulowania zawartością strony i innych działań.

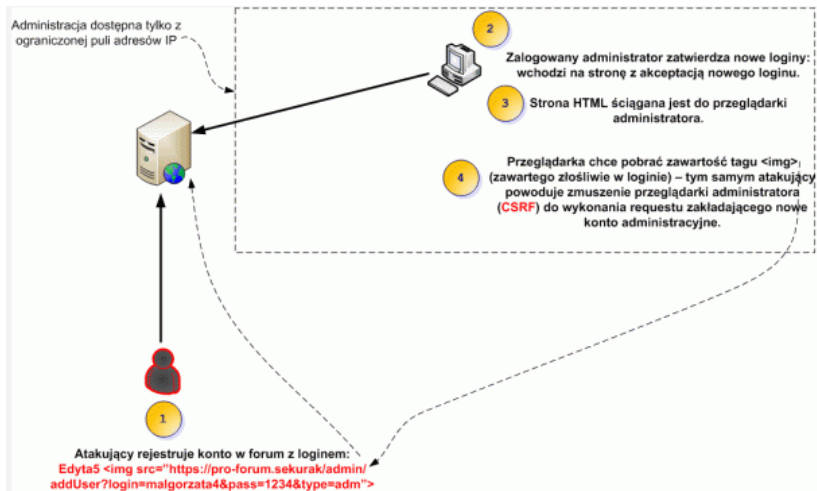


- 1 Reflected XSS: Złośliwy skrypt jest odzwierciedlany bezpośrednio w odpowiedzi serwera, zwykle poprzez parametry URL lub formularze
- 2 Stored XSS: Polegająca na umieszczeniu złośliwego skryptu po stronie serwerowej
- 3 DOM-based XSS: Złośliwy skrypt manipuluje DOM strony bezpośrednio w przeglądarce użytkownika

# CSRF (Cross-Site Request Forgery)

- Atak polegający na wymuszeniu żądania HTTP
- Wykonywany w imieniu zalogowanego użytkownika
- Wykorzystuje automatyczne wysyłanie cookies

# CSRF Przykład



- Losowy, unikalny token generowany przez serwer
- Dołączany do formularzy lub nagłówków
- Chroni przed atakami CSRF

# CORS (Cross-Origin Resource Sharing)

CORS (Cross-Origin Resource Sharing) to mechanizm bezpieczeństwa, który pozwala na kontrolowanie dostępu do zasobów na serwerze z innych domen niż ta, z której pochodzi żądanie. CORS umożliwia serwerom określenie, które domeny mają prawo do dostępu do ich zasobów, co pomaga zapobiegać nieautoryzowanym żądaniom cross-origin.

ACAO (Access-Control-Allow-Origin) to nagłówek odpowiedzi HTTP używany w mechanizmie CORS. Określa on, które domeny mają prawo do dostępu do zasobów serwera.

Może on przyjmować wartość konkretnej domeny (np. 'https://example.com') lub wartość '\*', co oznacza, że zasób jest dostępny dla wszystkich domen.

# CORS Przykład

