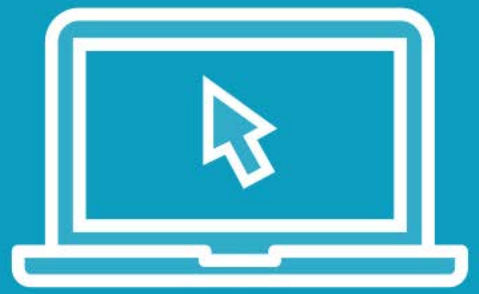
Cross-site Request Forgery (CSRF)

Christian Wenz

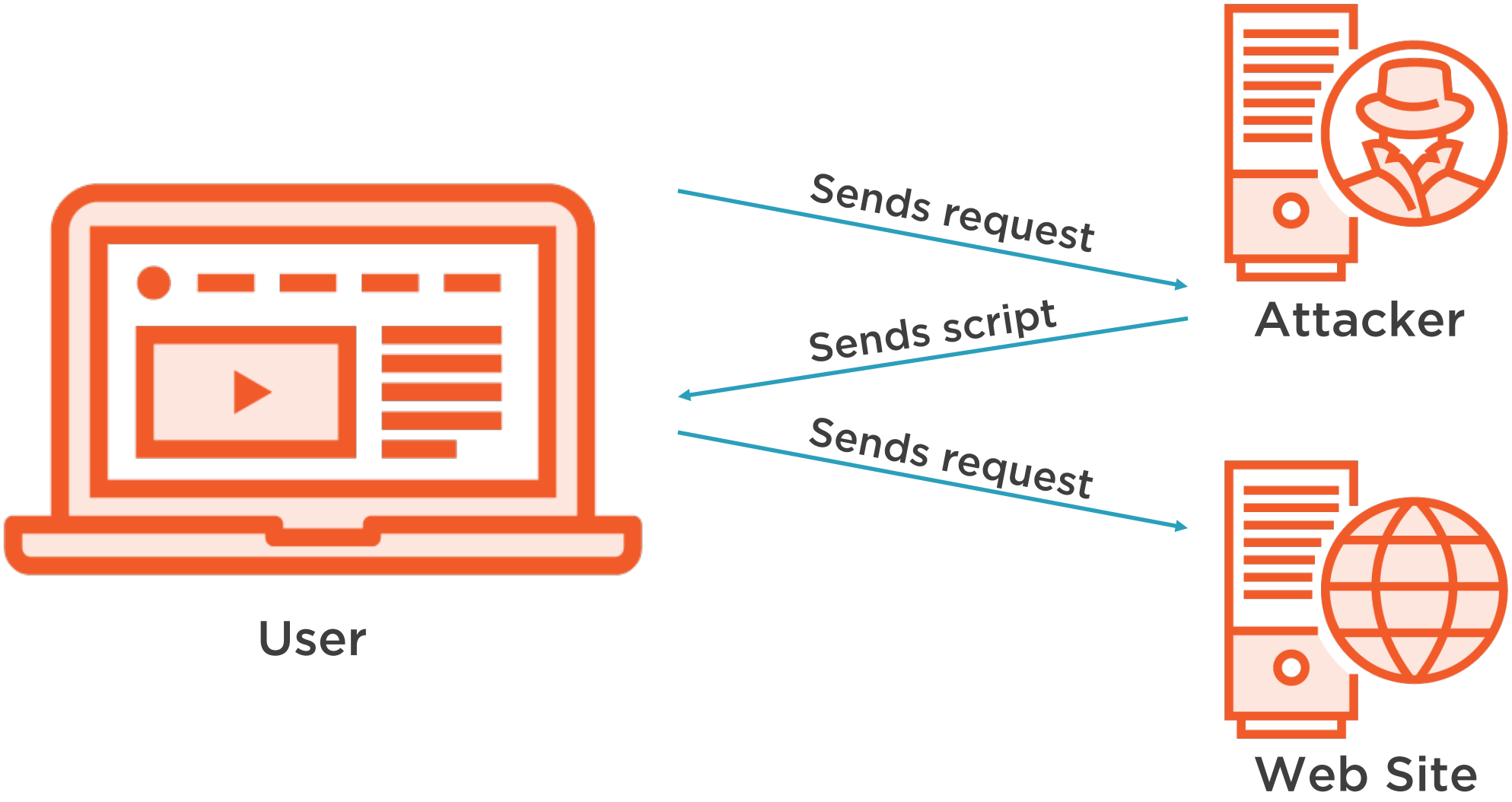
@chwenz

Demo



Prompting Users to Buy

Cross-site Request Forgery



Countermeasures

Add

random token

(nonce)

Implementation depends on the site and its purpose!

```
$name = 'token-' . mt_rand();

$token = random_bytes(32);

$_SESSION[$name] = $token;
```

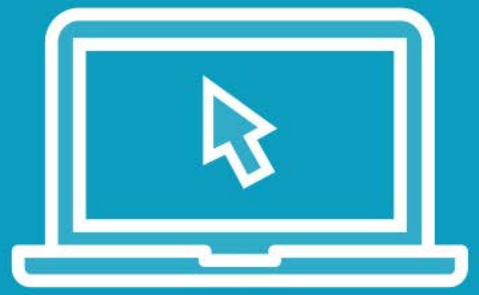
◀ Creating and storing the token

```
<input type="hidden"
       name="_csrfname"
       value="token-123456">

<input type="hidden"
       name="_csrfvalue"
       name="a1b2c3d4e5f6">
```

◀ HTML form markup

Demo



Prompting Users to Buy (Revisited)

Preventing Clickjacking

```
header('X-FRAME-OPTIONS', 'DENY'); //or: SAMEORIGIN
```

```
header(
  'Content-Security-Policy', "frame-ancestors: 'none'");
  // or: 'self', or domain/URI list
```

Summary

Cross-Site Request Forgery is possible if an attacker can predict the HTTP request

Add a random token to protect against CSRF

Prevent framing to protect against Clickjacking