OWASP

XSS, CSRF, SQL injection

UKEN Kraków 2024

Opis zagrożeń

Cross-Site Scripting (XSS)
Ataki polegające na wstrzykiwaniu złośliwego kodu JavaScript do strony internetowej w celu kradzieży danych użytkowników.

SQL Injection
Ataki polegające na wstrzykiwaniu kodu SQL do formularzy aplikacji w celu uzyskania nieuprawnionego dostępu do danych.

Cross-Site Request Forgery (CSRF)
Ataki polegające na wykorzystaniu uprawnionych sesji użytkowników do wykonania niechcianych akcji w ich imieniu.

CROSS-SITE SCRIPTING (XSS)

Ataki typu Cross-Site Scripting polegają na wstrzykiwaniu złośliwego kodu JavaScript do strony internetowej, który jest następnie wykonywany przez przeglądarkę użytkownika. Atakujący wykorzystuje luki w mechanizmach walidacji danych wejściowych, takich jak formularze lub pola komentarzy, aby wstrzyknąć kod, który może przechwytywać dane użytkowników, przekierowywać ich na fałszywe strony lub wykonywać inne złośliwe działania. Aby zapobiec atakom typu XSS, aplikacje powinny odpowiednio filtrować i kodować dane wprowadzane przez użytkowników oraz korzystać z mechanizmów zabezpieczających, takich jak nagłówki HTTP CSP (Content Security Policy).

SQL INJECTION

Ataki typu SQL Injection polegają na wstrzykiwaniu złośliwego kodu SQL do formularzy aplikacji internetowej w celu uzyskania nieuprawnionego dostępu do danych przechowywanych w bazie danych. Atakujący wykorzystuje luki w mechanizmach obsługi danych wejściowych, takich jak formularze, aby wprowadzić kod, który może modyfikować, odczytywać lub usuwać dane z bazy. Aby zapobiec atakom typu SQL Injection, aplikacje powinny stosować parametryzację zapytań SQL, filtrować i walidować dane wejściowe oraz unikać bezpośredniego łączenia zapytań SQL z danymi wprowadzanymi przez użytkowników.

CROSS-SITE REQUEST FORGERY (CSRF)

Ataki typu Cross-Site Request Forgery polegają na wykorzystaniu uprawnionych sesji użytkownika do wykonania niechcianych akcji w ich imieniu. Atakujący tworzy fałszywe żądania HTTP, które są automatycznie wysyłane do aplikacji, gdy użytkownik odwiedza złośliwą stronę lub klika w złośliwy link. Jeśli użytkownik jest zalogowany do aplikacji, atakujący może wykorzystać tę sesję do wykonania działań, takich jak zmiana hasła, wykonanie transakcji finansowych itp. Aby zapobiec atakom typu CSRF, aplikacje powinny stosować mechanizmy tokenu CSRF, które sprawdzają, czy żądanie pochodzi od prawidłowego źródła.

PRAKTYKI W ZAKRESIE BEZPIECZEŃSTWA APLIKACJI

Weryfikacja danych wejściowych:
należy zawsze sprawdzać i filtrować dane wprowadzane przez użytkowników, aby uniknąć ataków typu SQL Injection i XSS

Używanie mechanizmów uwierzytelniania i autoryzacji
każda aplikacja internetowa powinna wymagać uwierzytelnienia użytkowników oraz zapewnić odpowiednie poziomy autoryzacji, aby chronić dostęp do danych

Regularne aktualizacje
ważne jest regularne aktualizowanie aplikacji i frameworków, aby wyeliminować znane podatności i korzystać z najnowszych zabezpieczeń

Używanie szyfrowania danych
dane przesyłane pomiędzy użytkownikiem a aplikacją powinny być zaszyfrowane za pomocą protokołów takich jak HTTPS, aby chronić poufność i integralność informacji

Bezpieczne zarządzanie sesjami
aplikacje powinny zapewniać bezpieczne zarządzanie sesjami, aby uniknąć ataków typu CSRF i nieautoryzowanego dostępu