Autorzy: Krzysztof Kościsz

Michał Kościsz

Bezpieczeństwo Systemów Informatycznych

# Raport Bezpieczeństwa

# Data wykonania raportu: 20.01.2022 Wersja 1.0

1. **Aplikacja**

Nazwa: Vulnerable Web Application

Technologia aplikacji: Apache 2.4.18, PHP 5.6.21

Back-end: MySql 5.0

Repozytorium: <https://github.com/OWASP/Vulnerable-Web-Application>

Opis:

Vulnerable-Web-Application to strona, która jest przygotowana dla osób, które interesują się testami penetracyjnymi i chcą mieć informacje na ten temat lub pracować. Strona jest dość prosta w instalacji i użytkowaniu.

Vulnerable-Web-Application obejmuje Command Execution, File Inclusion, File Upload, SQL oraz XSS. Dla kategorii wymagających bazy danych, tworzy bazę danych pod localhost za pomocą jednego przycisku podczas konfiguracji. W przypadku uszkodzonych lub zmienionych baz danych, można utworzyć bazę danych ponownie.

Licancja: Aplikacja jest licencjonowana na zasadach GNU General Public License v3.0.

1. **Klasyfikacja podatności**

Na liście poniżej przedstawiono skrótowy opis każdej z podatności. Każdy błąd został oznaczony kolorem, zgodnie z legendą:

****

* **Niskie zagrożenie –** wykorzystanie podatności ma niewielki bezpośredni wpływ na bezpieczeństwo aplikacji lub wymaga bardzo trudnych warunków do spełnienia (np. fizyczny dostęp do serwera).
* **Średnie zagrożenie –** wykorzystanie podatności może zależeć od zewnętrznych czynników (np. wymaga przekonania użytkownika do kliknięcia w łącze) lub może wymagać trudnych do spełnienia warunków. Ponadto wykorzystanie podatności zazwyczaj umożliwia dostęp tylko do ograniczonej ilości danych lub do danych o mniejszym poziomie istotności.
* **Wysokie zagrożenie –** wykorzystanie podatności umożliwia przejęcie pełnej kontroli nad serwerem lub urządzeniem sieciowym albo pozwala uzyskać dostęp (w trybie zapisu i/lub odczytu) do danych o dużym poziomie poufności i istotności. Wykorzystanie podatności pozwala także na uzyskanie dostępu do wrażliwych informacji, może wcześniej wymagać spełnienia pewnych warunków (np. posiadania konta użytkownika w wewnętrznym systemie).

1. **Lista odnalezionych podatności - szczegóły**

Niniejszy raport jest podsumowaniem testów bezpieczeństwa przeprowadzonych w celu wykrycia ewentualnych luk w oprogramowaniu. Przedmiotem sprawdzenia była aplikacja webowa Vulnerable Web Application.

W trakcie audytu szczególny nacisk położono na podatności mające lub mogące mieć negatywny wpływ na poufność, integralność oraz dostępność przetwarzanych danych. Testy bezpieczeństwa przeprowadzono zgodnie z powszechnie przyjętymi metodykami testowania aplikacji webowych, takimi jak: OWASP TOP10

* 1. **Podatności XSS (Cross-Site Scripting)**

Na czym polega istota podatności XSS? Po pierwsze jest to przede wszystkim atak na klienta korzystającego z podatnej webaplikacji (w przeciwieństwie np. do SQL injection, którego głównym celem jest część serwerowa). Po drugie, atak polega na wstrzyknięciu do przeglądarki ofiary fragmentu javascript który może być uruchomiony w przeglądarce.

W efekcie, atakujący ma możliwość wykonania dowolnego kodu skryptowego w przeglądarce

Poziom ryzyka:

Text

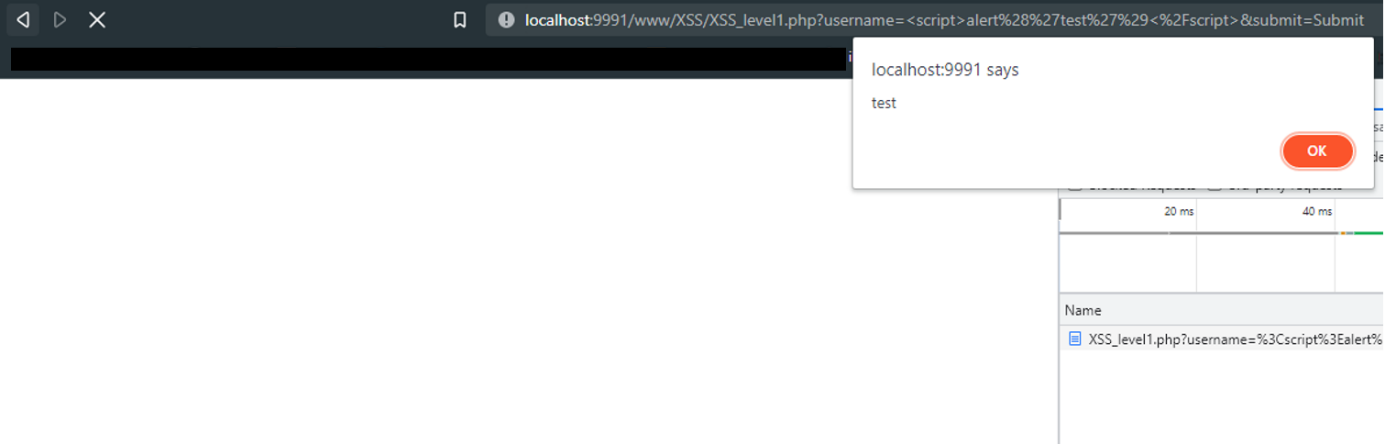
Description automatically generated with low confidence

Liczne wystąpienia, m.in. nazwa użytkownika w historii logowania, nazwy produktów itp.

Brak lub dowolne konto w systemie (w zależności od konkretnego wystąpienia).

Graphical user interface, text, application, email

Description automatically generated



Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Graphical user interface, text, application, email, website

Description automatically generated

* 1. **Podatności Sql Injection**

Poziom ryzyka:

Text

Description automatically generated

W aplikacji zidentyfikowano błędy SQL Injection, pozwalające napastnikom na pełny dostęp do bazy danych danej instancji W konsekwencji wykorzystania tej podatności możliwy jest:

Widoczna informacja na temat bazy danych oraz znajdujących się w niej tabel.

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

W aplikacji zidentyfikowano błędy SQL Injection, pozwalające napastnikom na pełny dostęp do bazy danych danej instancji YetiForce. W konsekwencji wykorzystania tej podatności możliwy jest: • Dostęp do hashy haseł użytkowników systemu, • Dostęp do danych wszystkich kontrahentów, • Dostęp do treści maili.

* 1. **Command Execution**

| curl <http://127.0.0.1/.testing/rce.txt>

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

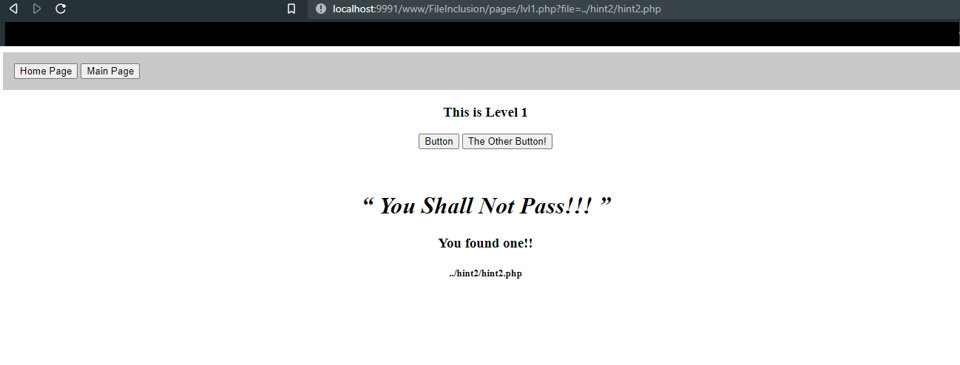
* 1. **File Inclusion**

Poziom ryzyka:

Text

Description automatically generated

****

****

Graphical user interface, text, application, email

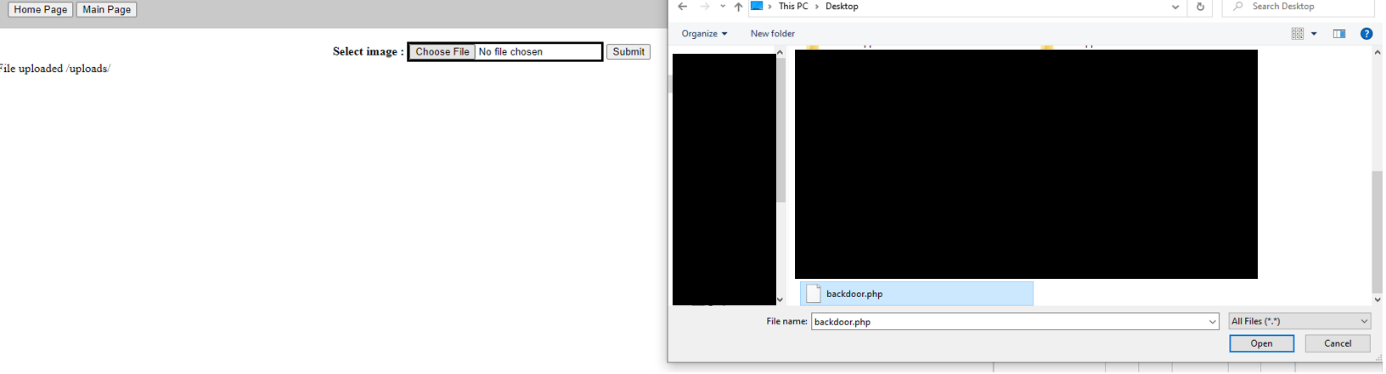
Description automatically generated

* 1. **File Uploads**

1. Poziom ryzyka:

Text

Description automatically generated

****

Graphical user interface, text, application

Description automatically generated

1. **Wnioski z testów**

W aplikacji zidentyfikowano kilka sposobów na wykonywanie dowolnego kodu po stronie serwera. Praktycznie tego typu podatność może zostać wykorzystana m.in. do:

* Przejęcia wszystkich danych przechodzących przez system. Mogą to być zarówno dane użytkowników systemu (loginy i hasła), jak i dane przetwarzane przez system, takie jak: listy kontrahentów, dane osobowe itp.
* Dalszych ataków na inne hosty znajdujące się w sieci wewnętrznej,

Zwraca również uwagę duża liczba podatności typu Cross-Site Scripting (XSS), które mogą zostać wykorzystane do wstrzyknięcia niepożądanego kodu.

Mechanizm importowania plików dostępny dla wszystkich użytkowników systemu YetiForce w niewystarczający sposób waliduje poprawność wysyłanych plików, pozwalając tym samym na wgranie pliku PHP do dowolnego katalogu serwera.

Ponadto, w systemie zostało zidentyfikowanych kilka podatności o mniejszym poziomie istotności, które jednak też mogą w pewien sposób ułatwiać napastnikom przejęcie kontroli nad systemem.

Wszystkie problemy zostały szczegółowo opisane w dalszej części raportu