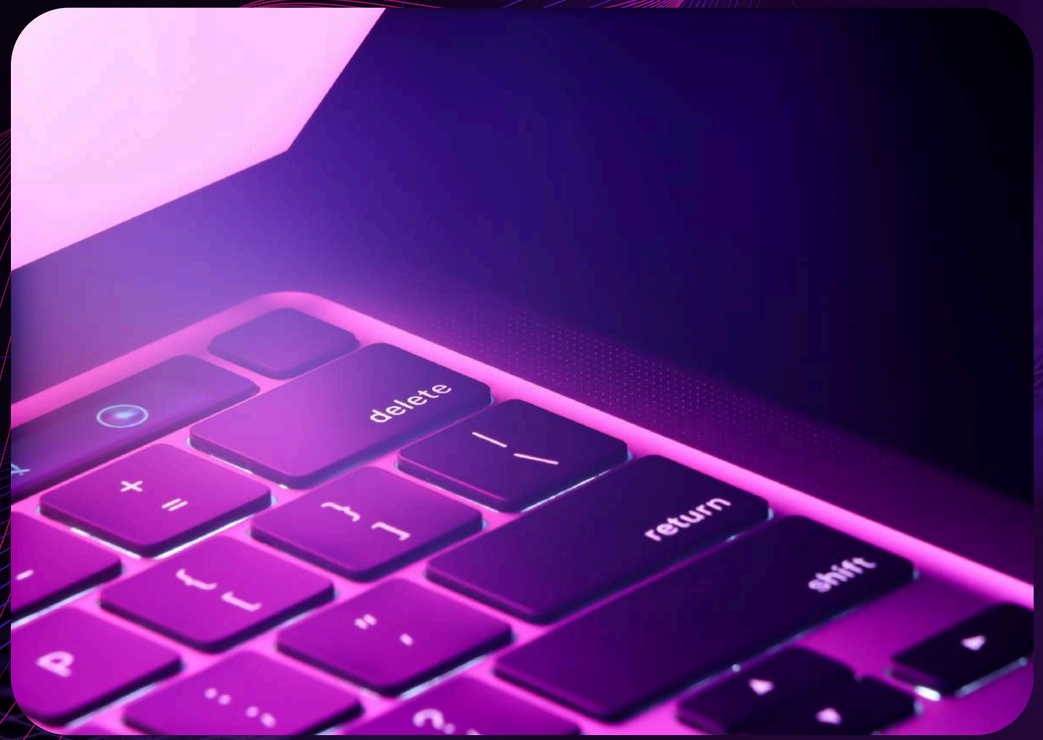
File Upload Vulnerabilities

Mikołaj Mazur, Tyberiusz Boberek, Mateusz Kurdziel

Czym Jest File Upload Vulnerability?



podatność przesyłania plików:
Luka bezpieczeństwa, występująca gdy aplikacja pozwala użytkownikom na przesłanie plików bez ich odpowiedniej weryfikacji oraz obsłużenia.

Gdzie Wystepuje?

- Formularze z możliwością przesłania plików
- Systemy CMS i panele administracyjne
- Systemy obiegu dokumentów i Helpdeski
- Funkcje importu danych

Dlaczego Wystepuje?

brak lub słaba weryfikacja plików

ufanie walidacji po stronie klienta

błędna konfiguracja serwera

Brak lub Słaba Weryfikacja Plików

- Całkowity Brak Ograniczeń
- Niepoprawna Walidacja dla Obfuskacji
- Brak Weryfikacji Rozmiaru Pliku (w tym ZIP)
- Brak Weryfikacji Rozszerzenia

Ufanie Walidacji po Stronie Klienta

- **Weryfikacja Tylko w JavaScript**
- **Atrybut “accept” w HTML**
- **Zaufanie do Nagłówka Content-Type**
- **Brak Powtórnej Weryfikacji na Backendzie**

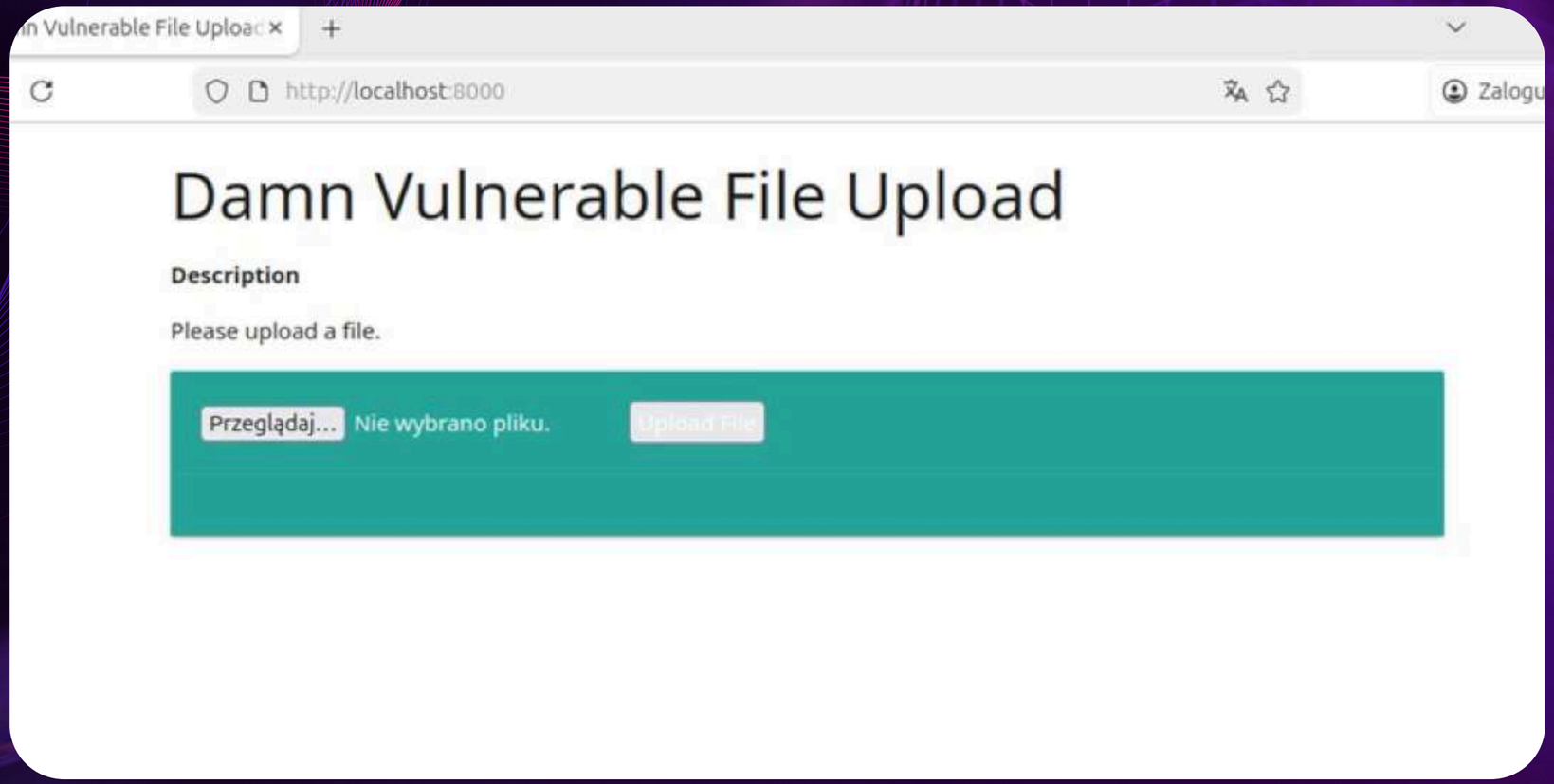
Błedna Konfiguracja Serwera

- Używanie "Czarnych List"
- Uprawnienia do Wykonywania
- Przechowywanie Wewnątrz "Web Root"
- Obsługa Niestandardowych Rozszerzeń
- Przewidywalne Nazwy Plików
- Brak Izolacji

Jakie Skutki Niesie?

- Remote Code Execution (RCE)
- Reverse Shell - przejęcie serwera
- DoS & Storage Exhaustion
- Content Spoofing - nadpisywanie plików

Przykład Ataku

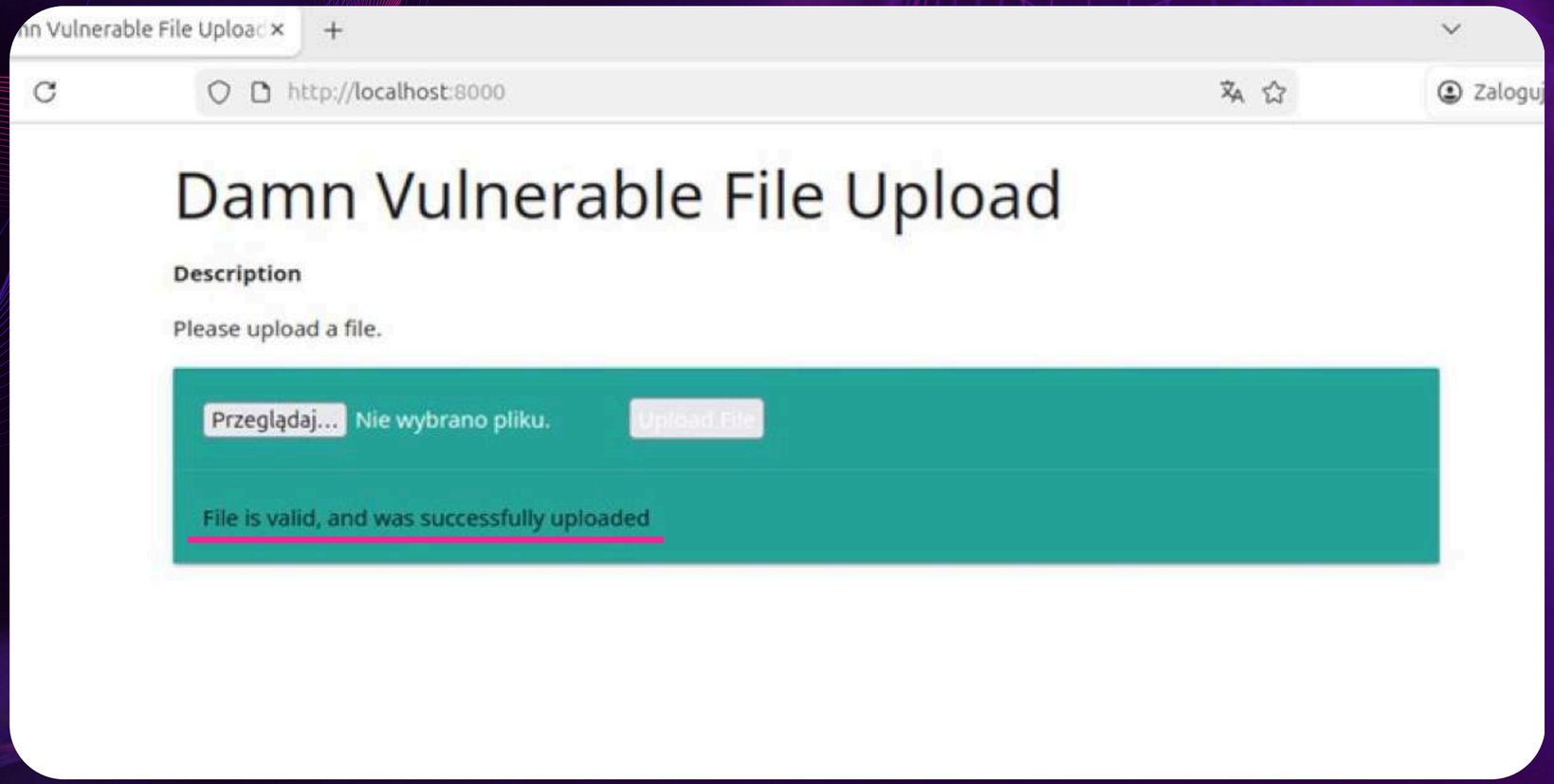


Przykład Ataku

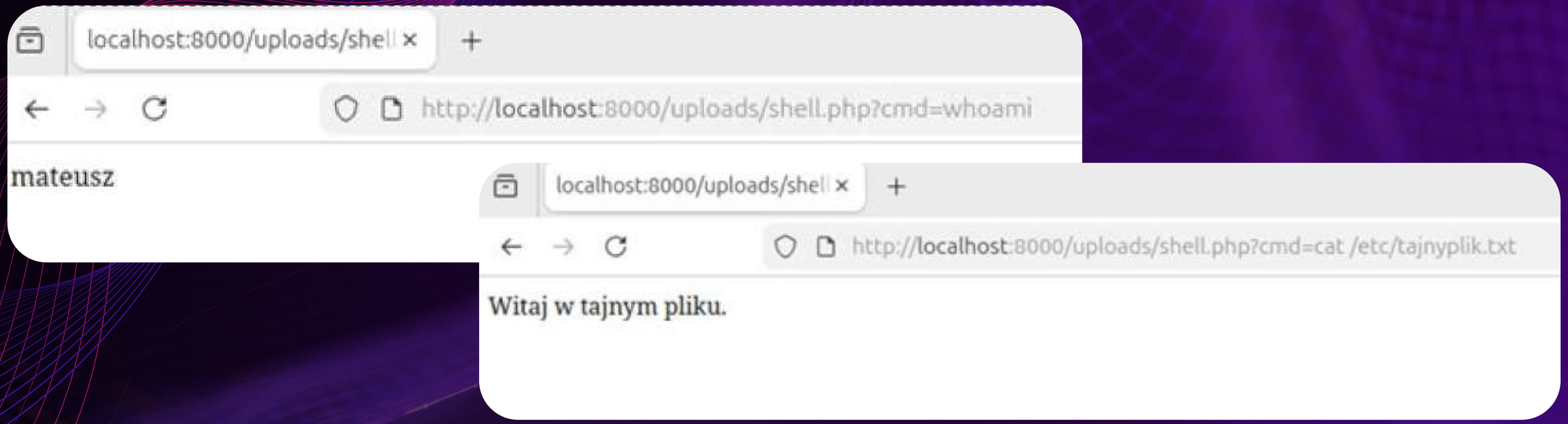
```php
<?php
system($_GET['cmd']);
?>
```

+

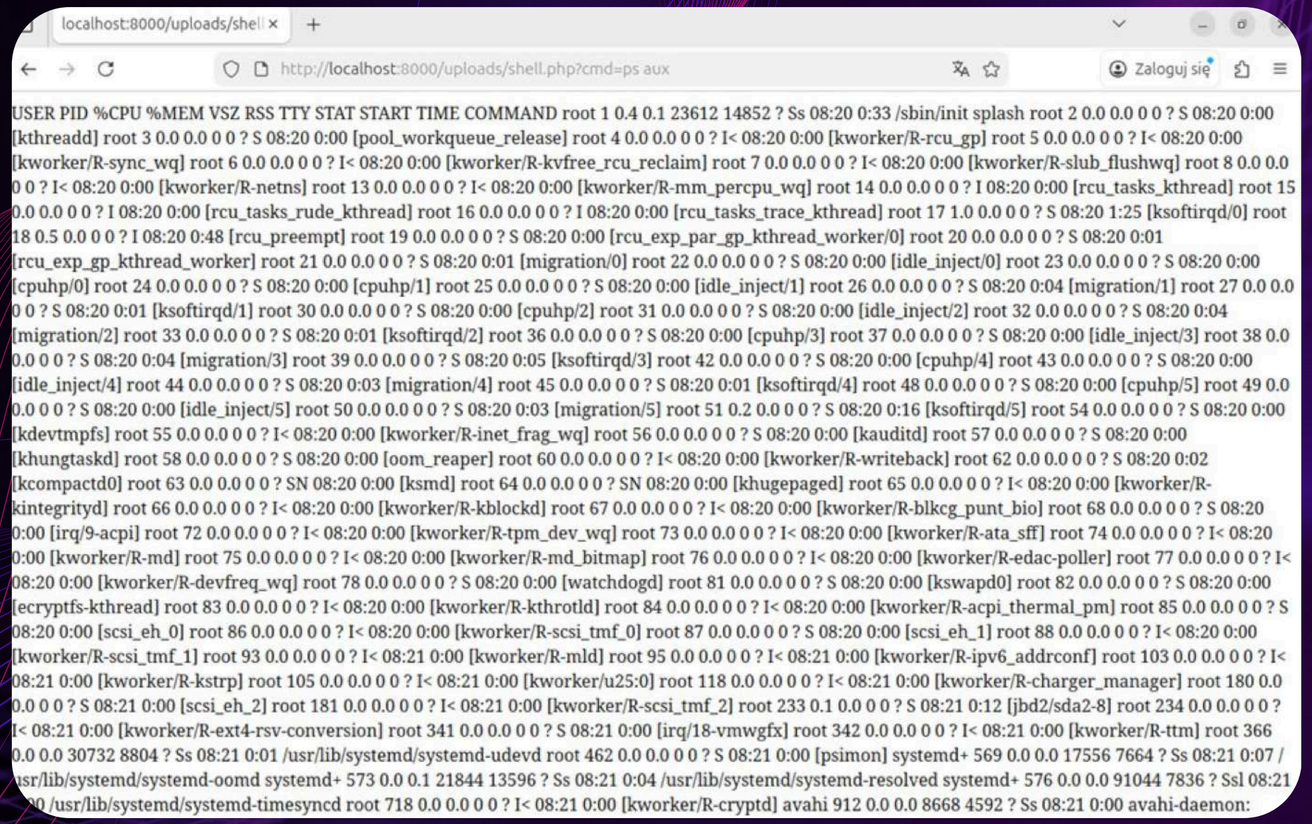
Przykład Ataku

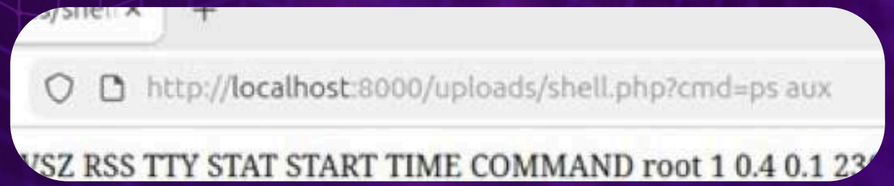


Przykład Ataku



Przykład Ataku





Obrona

filtracja i walidacja

walidacja po stronie serwera

przechowywanie danych

Filtracja i Walidacja

- Whitelisty (dozwolone typy)
- MIME-type
- Magic bytes

```php
// ------------------------------------------------------------
// Weryfikacja MIME-type
// ------------------------------------------------------------
$ALLOWED_MIME = ['image/jpeg', 'image/png', 'application/pdf'];
$finfo = new finfo(FILEINFO_MIME_TYPE);
$mime = $finfo->file($_FILES['file']['tmp_name']);
if (!in_array($mime, $ALLOWED_MIME)) {
    die("Niedozwolony MIME-type.");
}
```

```php
// ------------------------------------------------------------
// Sprawdzenie rozszerzenia z whitelist
// ------------------------------------------------------------
$ALLOWED_EXT = ['jpg', 'jpeg', 'png', 'pdf'];
$ext = strtolower(pathinfo($filename, PATHINFO_EXTENSION));
if (!in_array($ext, $ALLOWED_EXT)) {
    die("Niedozwolone rozszerzenie pliku.");
}
```

```php
// ------------------------------------------------------------
// Sprawdzenie magic bytes
// ------------------------------------------------------------
$fh = fopen($_FILES['file']['tmp_name'], 'rb');
$header = fread($fh, 8); // bierzemy pierwsze bajty
fclose($fh);
```

```php
// PNG
if ($mime === "image/png" && !str_starts_with($header, "\x89PNG")) {
    die("Not PNG!!!");
}

// PDF
if ($mime === "application/pdf" && !str_starts_with($header, "%PDF")) {
    die("Not PDF!!!");
}
```

Walidacja Po Stronie Serwera

- Bezpieczna nazwa pliku
- Analiza pod kątem złośliwej zawartości
- Rozmiar pliku

```php
// ------------------------------------------------------
// Czysta nazwa pliku (bez ../ itp.)
// ------------------------------------------------------
$original = $_FILES['file']['name'];
$filename = basename($original);
$filename = preg_replace("/[^A-Za-z0-9\._-]/", "_", $filename);
```

```php
// ------------------------------------------------------
// Limit rozmiaru
// ------------------------------------------------------
$MAX_SIZE = 5 * 1024 * 1024; // 5 MB
if ($_FILES['file']['size'] > $MAX_SIZE) {
    die("Plik za duży.");
}
```

Przechowywanie Danych

- Własne nazwy plików
- Narzucona ścieżka do pliku z dala od wrażliwych części aplikacji
- Brak uprawnień wykonywania

```php
// Folder NA ZEWNĄTRZ webroot, np. /var/uploads/
$UPLOAD_DIR = "/var/uploads/";

// ------------------------------------------------------
// Generowanie nowej, unikalnej nazwy
// ------------------------------------------------------
$newName = uniqid("file_", true) . "." . $ext;
```

Wspomagajki i Przykłady Podatnosci

Wspomagajki:

- OWASP Application Security Verification Standard (ASVS)
- OWASP File Upload Cheat Sheet
- OWASP Web Security Testing Guide

Przykłady:

- WordPress Plugin (Forminator)
  - CVE-2023-4596
  - CVE-2024-28890
- SAP NetWeaver (Visual Composer development server)
  - CVE-2025-31324

Dziekujemy Za Uwagę

część praktyczna

github.com/marmag0/BAIiM-file-upload-vulnerabilities

Bibliografia

- https://www.cve.org/CVERecord/
- https://owasp.org/www-project-web-security-testing-guide/stable/
- https://owasp.org/www-project-application-security-verification-standard/
- https://cheatsheetseries.owasp.org/cheatsheets/File_Upload_Cheat_Sheet.html
- https://github.com/LunaM00n/File-Upload-Lab