

# ファイル処理の問題

## 1. 攻撃

ファイル処理が関係する攻撃として次のようなものがあげられる。

**ディレクトリトラバーサル** . . . サーバ内の**ファイル**へ不正アクセス

**OSコマンドインジェクション** . . . **OSのコマンド**の呼び出し

## ディレクトリトラバーサル

サーバ内のファイルに不正アクセスすると次のような影響が起こりうる。

- Webサーバ内の**ファイルの閲覧** → 情報漏えい
- Webサーバ内の**ファイルの改ざん** → デマや誹謗中傷の書き込み、**マルウェアサイトへの誘導する**仕組みの書き込み、  
ファイル削除によるサーバの機能停止、任意のサーバスクリプトの実行

## 1. 対策

外部から**ファイル名を指定できる**仕様はさける . . . ファイル名**固定**、セッション変数に保持、**ファイル名は直接指定**しない。

ファイル名に**ディレクトリ名が含まれない**ようにする . . . 「**../**」など含まれないようにする。？チェック

＜例＞basename関数を使用する。

ファイル名を**英数字に限定**する . . . 「**../**」などが入力されないようにする。

## 2. 攻撃手法

**ヌルバイト(文字コード0、文字列の終端を表す)**を利用する。

## 3. 原因

ファイル名が**外部から指定**することができる。

ファイル名として、**絶対パスや相対パスの形で異なるディレクトリを指定**できる。

**組み立てたファイル名**に対するアクセス**可否をチェック**していない。

## 4. 意図しないファイルの公開による影響

**秘密ファイルが閲覧ができる**ようになってしまう。

### ▼ 原因

**公開ディレクトリ**にファイルが置かれている

ファイルに対する**URLを知る**方法がある。。

ファイルに対する**URLアクセス制限**がかかっていない。

### ▼ URLを確認するための手段

**ディレクトリリンスニング(ファイルの一覧を表示する機能)**が有効

ファイル名が日付、ユーザ名、連番など**推測ができる**。

user.datなど**ありがちな名前**。

**エラーメッセージ**などによりファイルのパス名がわかる。




外部サイトからリンクされるなどして、検索エンジンに登録される。

## ▼ 対策

公開ディレクトリに公開ファイルしないは置かない、ディレクトリリスニングを無効にする。

＊**htaccess**などによる制限

## Index of /ecc/uchiyama

Name	Last modified	Size	Description
 <a href="#">Parent Directory</a>		-	
 <a href="#">test.php</a>	2023-07-11 16:39	165	
 <a href="#">test.py</a>	2023-07-11 16:23	16	

## OSコマンドインジェクション

PHPなどの言語には**OSのコマンド**を呼び出し、実行できる**関数**がある。

↓ 関数を利用

OSコマンドを不正に実行できる。

### 1. 攻撃手法

パイプやリダイレクトなどの**シェル**スクリプトの機能を**利用**する。

### 2. 原因

OSコマンドを呼び出すときに、シェルのメタ文字がエスケープされていない。

シェル機能を呼び出せる関数を使用している。

#### ▼ シェル

複数のコマンドを実行できる構文がある（`;`、`&`、`&&`、`||`、```、`|`など）。元のコマンドに追加して別のコマンドを実行させることができる。

### 3. 対策

OSコマンドの呼び出しを使用しない実装方法を選ぶ。

シェル呼び出し機能がある関数避ける。

外部から入力された文字列をコマンドラインのパラメータに渡さない。

OSコマンドに渡すパラメータを安全な関数によりエスケープする。

#### ＊保険的対策

パラメータの検証・・・文字数を制限する。

アプリの稼働するユーザ権限を最小にする・・・Webアプリケーションの権限をを最小にする。

WebサーバのOS、ミドルウェアのバッチ適用。