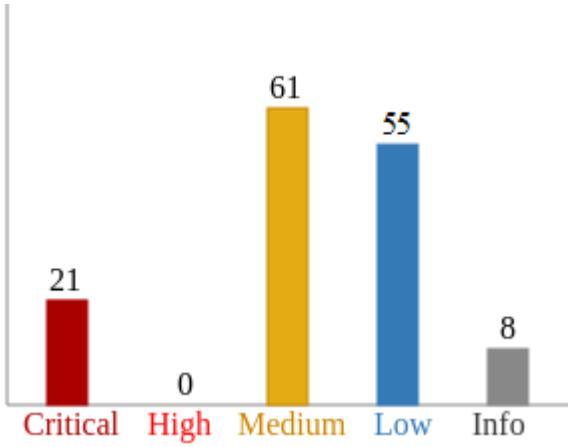
Web アプリケーション スキャンレポート

サンプルサイト
(https://example.jp/todo/)

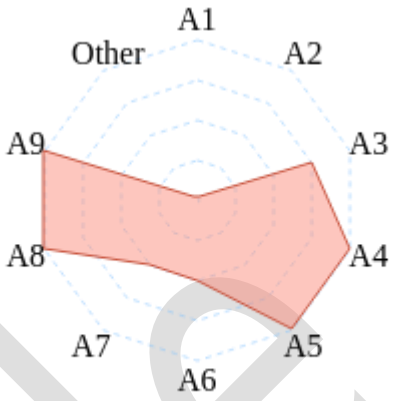
目次

スキャンサマリー

全体評価　Critical



深刻度ごとのサマリー



OWASP TOP 10 カテゴリー

脆弱性の深刻度は CVSSv3 (https://www.ipa.go.jp/security/vuln/CVSSv3.html) に基づき以下の基準で設定しています。

深刻度	CVSSv3 基本値	脆弱性に対して想定される脅威
Critical	9.0～10.0	・リモートからシステムを完全に制御されるような脅威 ・大部分の情報が漏えいするような脅威 ・大部分の情報が改ざんされるような脅威
High	7.0～8.9	
Medium	4.0～6.9	・一部の情報が漏えいするような脅威 ・一部の情報が改ざんされるような脅威 ・サービス停止に繋がるような脅威 ・その他、Critical/High に該当するが再現性が低いもの
Low	0.1～3.9	・攻撃するために複雑な条件を必要とする脅威 ・その他、Medium に該当するが再現性が低いもの
Info	0	

基本情報

サイトタイプ	PC 向けサイト
トップ URL	https://example.jp/todo/
画面数	102 (スキャン対象: 102)
開始時刻	yyyy/MM/dd HH:mm
終了時刻	yyyy/MM/dd HH:mm
巡回ドメイン	なし

脆弱性 サマリー

脆弱性 カテゴリー	件数
Critical	
SQL インジェクション	: 15 件
PHP ファイルインクルージョン	: 5 件
認証の迂回（SQL インジェクション）	: 1 件
High	: 0 件
Medium	
クリックジャッキング	: 23 件
クロスサイトスクリプティング	: 38 件
Low	
セッション ID の表示（URL）	: 33 件
デバッグ情報の表示（レスポンスボディ）	: 21 件
不要ファイルの公開	: 1 件
Info	
X-Content-Type-Options ヘッダの不備	: 1 件
Strict-Transport-Security ヘッダの不備	: 1 件
X-XSS-Protection ヘッダの不備	: 1 件
製品情報の表示（レスポンスヘッダ）	: 1 件
製品情報の表示（レスポンスボディ）	: 1 件
Set-Cookie ヘッダの HttpOnly 属性の不備	: 2 件
Set-Cookie ヘッダの SameSite 属性の不備	: 1 件

スキャン結果詳細

Critical

SQL インジェクション

深刻度

Critical

CVSS Score: 9.8

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

概要

危険な文字列を SQL 文に挿入できます。そのため、攻撃者が任意の SQL 文を実行できる危険性があります。

OWASP TOP 10 カテゴリー

A1:2017-インジェクション

ASVS4.0 カテゴリー

5.1.2,5.1.3,5.1.4,5.3.1,5.3.4,5.3.5,13.2.2,13.3.1

解説・対策方法

SQL インジェクションとは、攻撃者が細工した入力値を送信することで、開発者の想定していない SQL 文を実行できてしまう脆弱性です。この脆弱性は、利用者の送信した値が適切に前処理されずに SQL 文の一部として利用されることが原因で発生します。

この脆弱性を悪用することで、データベースの情報漏洩や情報改ざんなど、開発者の想定していない処理を実行されてしまう危険性があります。

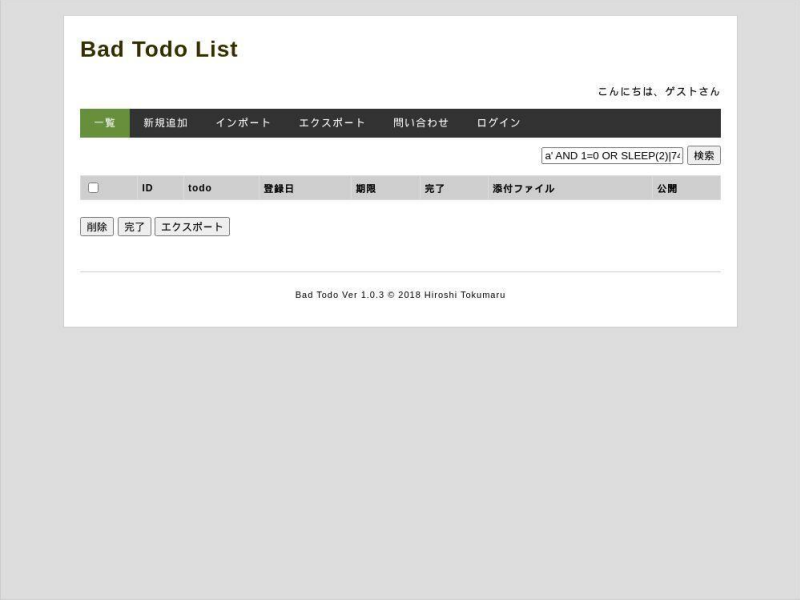
対策方法としては、想定していない値が入力された場合に処理を中断することや、SQL 文で特殊な意味を持つ文字を無害化することが挙げられます。後者を実現する一般的な方法としては、パラメータ化クエリやプリペアードステートメントの利用が挙げられます。

参考情報

- 安全なウェブサイトの作り方 - 1.1 SQL インジェクション：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_1.html)

- SQL Injection Prevention - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html)

スクリーンショット



脆弱性が見つかった箇所

画面
2726.一覧
(https://example.jp/todo/todolist.php?TODOSESSID=h43iejuuqu85bi5fj9e5frt0n3&key=a)
リクエスト URL
https://example.jp/todo/todolist.php?TODOSESSID=h43iejuuqu85bi5fj9e5frt0n3&key=a (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
GET パラメータ	key	a	a' AND 1=0 OR SLEEP(2)|74880364|(SELECT 0 UNION SELECT 1)='	処理に時間がかかる SQL 文を含むパラメータ値を送信したところ、応答時間が大きく変化しました。

画面
2730.一覧
(https://example.jp/todo/todolist.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96&key=a)
リクエスト URL
https://example.jp/todo/todolist.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96&key=a (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
GET パラメータ	key	a	a' AND 1=0 OR SLEEP(2)|70563351|(SELECT 0 UNION SELECT 1)='	処理に時間がかかる SQL 文を含むパラメータ値を送信したところ、応答時間が大きく変化しました。

PHP ファイルインクルージョン

深刻度

Critical

CVSS Score: 9.8

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

概要

外部ファイルをスクリプト内に挿入できます。そのため、攻撃者が任意のコードを実行できる危険性があります。

OWASP TOP 10 カテゴリー

Other:その他

ASVS4.0 カテゴリー

5.1.2,5.1.3,5.1.4,5.3.9,12.3.3,12.6.1,13.1.1,13.2.2

解説・対策方法

ファイルインクルージョンとは、攻撃者が入力値として送信したパスや URL をアプリケーションが開き、その内容を実行中のスクリプトに挿入して実行してしまう脆弱性です。

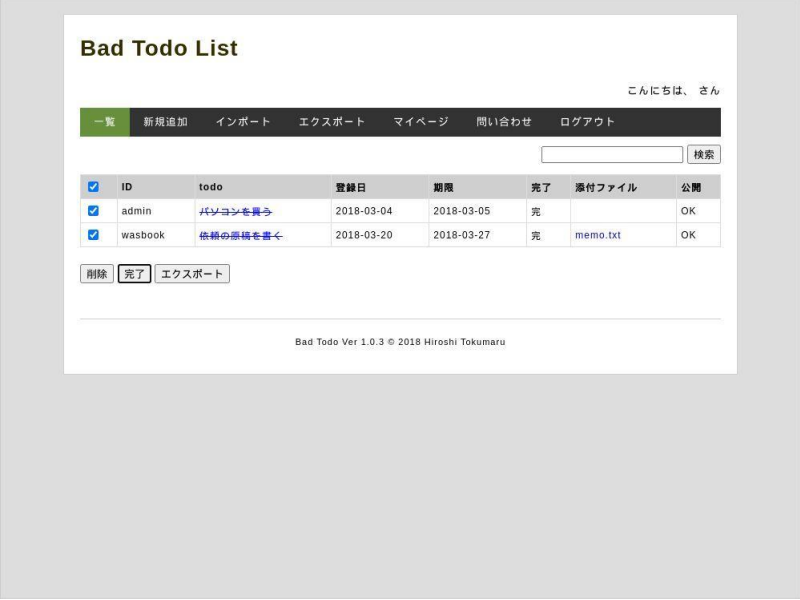
この脆弱性により、攻撃者が任意のコードを実行できる危険性があります。そのため、情報漏えいや情報改ざん、サービスの破壊などの被害に繋がる危険性があります。

対策方法としては、想定していない値が入力された場合に処理を中断することが挙げられます。

参考情報

- CWE - CWE-98: Improper Control of Filename for Include/Require Statement in PHP Program ('PHP Remote File Inclusion') (4.5) (https://cwe.mitre.org/data/definitions/98.html)

スクリーンショット



↓

aeyescan remote file inclusion

脆弱性が見つかった箇所

画面
2731.ログアウトしています (https://example.jp/todo/editlist.php)
リクエスト URL
https://example.jp/todo/editlist.php (POST)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
POST パラメータ	process	donelist	http://ai0.jp/rfi/donelist	「http://ai0.jp/rfi/」の内容を実行した結果と思われる文字列「aeyescan remote file inclusion」がレスポンスボディに含まれていました。

画面
2734.ログアウトしています (https://example.jp/todo/editlist.php)
リクエスト URL
https://example.jp/todo/editlist.php (POST)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
POST パラメータ	process	exportlist	http://ai0.jp/rfi/exportlist	「http://ai0.jp/rfi/」の内容を実行した結果と思われる文字列「aeyescan remote file inclusion」がレスポンスボディに含まれていました。

認証の迂回（SQL インジェクション）

深刻度

Critical

CVSS Score: 9.8

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

概要

ログイン機能に SQL インジェクションの脆弱性が存在します。この脆弱性を悪用することで、認証情報を知らない第三者によってログインされてしまう危険性があります。

OWASP TOP 10 カテゴリー

A2:2017-認証の不備

ASVS4.0 カテゴリー

5.1.3,5.1.4,5.3.1,5.3.4,5.3.5,13.2.2

解説・対策方法

SQL インジェクションとは、攻撃者が細工した入力値を送信することで、開発者の想定していない SQL 文を実行できてしまう脆弱性です。この脆弱性は、利用者の送信した値が適切に前処理されずに SQL 文の一部として利用されることが原因で発生します。

この脆弱性を悪用することで、データベースの情報漏洩や情報改ざんなど、開発者の想定していない処理を実行されてしまう危険性があります。

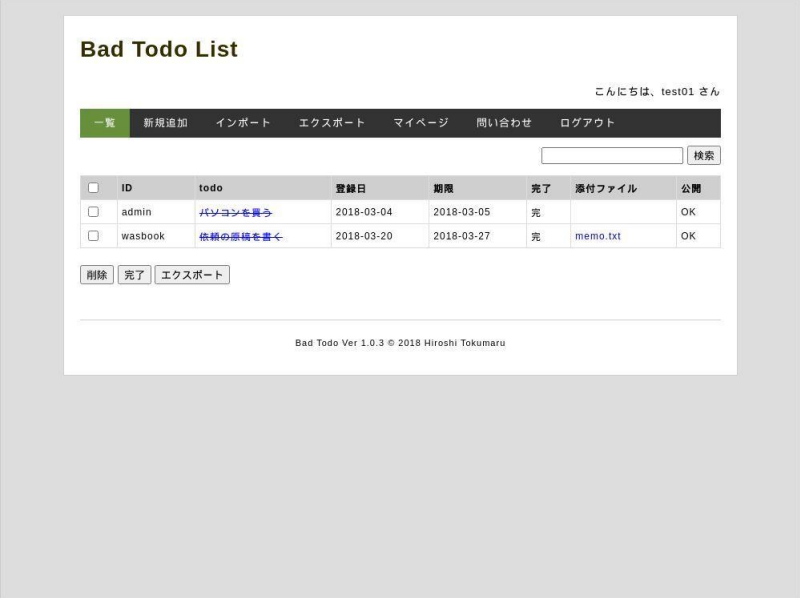
対策方法としては、想定していない値が入力された場合に処理を中断することや、SQL 文で特殊な意味を持つ文字を無害化することが挙げられます。後者を実現する一般的な方法としては、パラメータ化クエリやプリペアードステートメントの利用が挙げられます。

参考情報

- 安全なウェブサイトの作り方 - 1.1 SQL インジェクション：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_1.html)
- SQL Injection Prevention - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html)
- Authentication - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html)

スクリーンショット

↓



脆弱性が見つかった箇所

画面
2760.一覧 (https://example.jp/todo/todolist.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)
リクエスト URL
https://example.jp/todo/logindo.php? (POST)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
POST パラメータ	userid	test01	test01'--	パラメータ値に「'-- 」を付けたところ、ログインに成功しているように見えました。

Medium

クリックジャッキング

深刻度

Medium

CVSS Score: 4.3

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:N/I:L/A:N

概要

フォームが存在するページを外部サイトに埋め込むことができます。そのため、利用者がなりすましの被害を受ける危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

14.4.7

解説・対策方法

クリックジャッキングとは、フォームが存在するページを外部サイトに埋め込むことができる脆弱性です。この脆弱性は、外部サイトに埋め込まれることを対象ページが拒否しないことが原因で発生します。

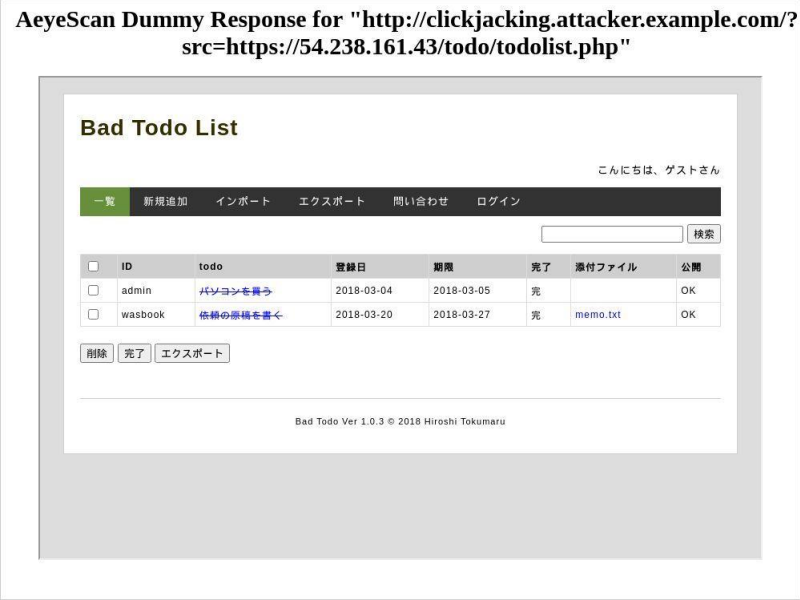
この脆弱性は、攻撃者が用意した罠ページに透明な iframe タグで対象ページを読み込み、その iframe タグの下に無害に見えるボタンを重ねて配置するなどの方法で悪用されます。この場合、無害に見えるボタンを被害者にクリックさせることで、実際には、そのボタンの上にある対象ページのフォーム投稿ボタンをクリックさせることができます。これにより、あたかも被害者が自分の意思で対象ページのフォームを投稿したかのように見せかけることができます。そのため、被害者はなりすましの被害を受ける危険性があります。

対策方法としては、X-Frame-Options レスポンスヘッダを追加することや、Content-Security-Policy ヘッダを追加して frame-ancestors ディレクティブを設定することが挙げられます。

参考情報

- 安全なウェブサイトの作り方 - 1.9 クリックジャッキング：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_9.html)
- Clickjacking Defense - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html)

スクリーンショット



脆弱性が見つかった箇所

画面
2716.一覧 (https://example.jp/todo/todolist.php)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	外部サイトに iframe タグで「https://example.jp/todo/todolist.php」を埋め込んだところ、「完了」ボタンを含むフォームが表示されました。

画面
2717.問い合わせ (https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	外部サイトに iframe タグで「https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96」を埋め込んだところ、「送信」ボタンを含むフォームが表示されました。

クロスサイトスクリプティング

深刻度

Medium

CVSS Score: 6.1

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N

概要

HTML や JavaScript をレスポンスの HTML 内に挿入できます。そのため、攻撃者が被害者のブラウザ上で任意の HTML や JavaScript を実行できる危険性があります。

OWASP TOP 10 カテゴリー

A7:2017-クロスサイトスクリプティング(XSS)

ASVS4.0 カテゴリー

5.1.2,5.1.3,5.1.4,5.2.1,5.2.8,5.3.1,5.3.3,5.3.6,13.2.2

解説・対策方法

クロスサイトスクリプティングとは、利用者のブラウザ上で、開発者の想定していない HTML や JavaScript を実行させることができる脆弱性です。この脆弱性は、入力値を適切に処理せずに、レスポンスに出力してしまうことが原因で発生します。

この脆弱性には、反射型と格納型の 2 つの種類があります。前者は、細工した入力値を送信した時にのみ、開発者の想定していない HTML や JavaScript を実行させることができます。これは、攻撃者が細工して用意した URL に被害者をアクセスさせるなどして悪用されます。後者は、細工した入力値を送信した時以降、開発者の想定していない HTML や JavaScript を実行させることができます。これは、攻撃者があらかじめ細工した入力値を送信しておき、脆弱性が発生するページに被害者を誘導するなどして悪用されます。

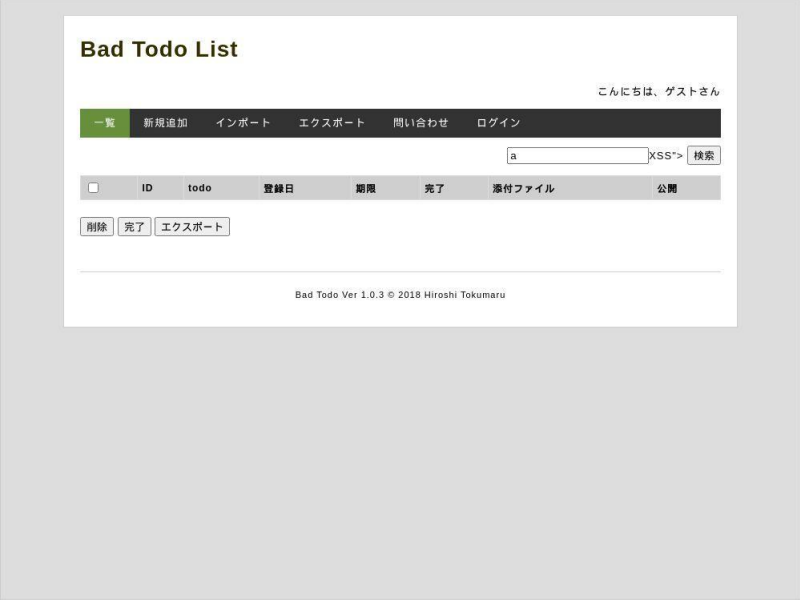
この脆弱性により、攻撃者が被害者のブラウザ上で任意の HTML や JavaScript を実行できる危険性があります。JavaScript はページ上の情報を取得して外部に送信したり、ページ上のボタンなどを操作することができるため、情報漏えいや情報改ざんなどの被害に繋がる危険性があります。

対策方法としては、想定していない値が入力された場合に処理を中断することや、HTML で特殊な意味を持つ文字を無害化することが挙げられます。後者としては HTML 文字参照に変換する方法が一般的です。具体的には「<」を「<」に、「>」を「>」、「&」を「&」、「"」を「"」、「'」を「'」に置換します。ただし、onclick などのイベントハンドラに出力される場合は、HTML 文字参照に変換する前に、「\」を「\\」に、「"」を「\"」に、「'」を「\'」に、改行文字を「\n」に置換するなどして、JavaScript で特殊な意味を持つ文字を無害化する必要があります。

参考情報

- 安全なウェブサイトの作り方 - 1.5 クロスサイト・スクリプティング：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_5.html)
- Cross Site Scripting Prevention - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html)

スクリーンショット



脆弱性が見つかった箇所

画面
2726.一覧
(https://example.jp/todo/todolist.php?TODOSESSID=h43iejuuqu85bi5fj9e5frt0n3&key=a)
リクエスト URL
https://example.jp/todo/todolist.php?TODOSESSID=h43iejuuqu85bi5fj9e5frt0n3&key=a (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
GET パラメータ	key	a	a"/oncut=confirm(90086488)>XSS	パラメータ操作後に「a」を切り取りしたところ、「90086488」を含んだダイアログが表示されました。

画面
2727.ログアウトしています (https://example.jp/todo/editlist.php)
リクエスト URL
https://example.jp/todo/editlist.php (POST)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
URL	—	https://example.jp/todo/editlist.php	https://example.jp/todo/editlist.php/'%22%3E%...)%3E%3C%3Ca%7Casvg/onload%3Dconfirm(96733324)%3E	「96733324」を含んだダイアログが表示されました。

Low

セッション ID の表示（URL）

深刻度

Low

CVSS Score: 3.1

CVSS Vector: CVSS:3.0/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N

概要

セッション ID が URL に含まれています。そのため、利用者が目を離している間に、攻撃者が画面を盗み見たりブラウザを操作することで、攻撃者がセッション ID を取得できる危険性があります。また、Referer リクエストヘッダを介してアクセスログにセッション ID が記録されてしまう危険性があります。

OWASP TOP 10 カテゴリー

A3:2017-機微な情報の露出

ASVS4.0 カテゴリー

3.1.1,8.3.1,13.1.3

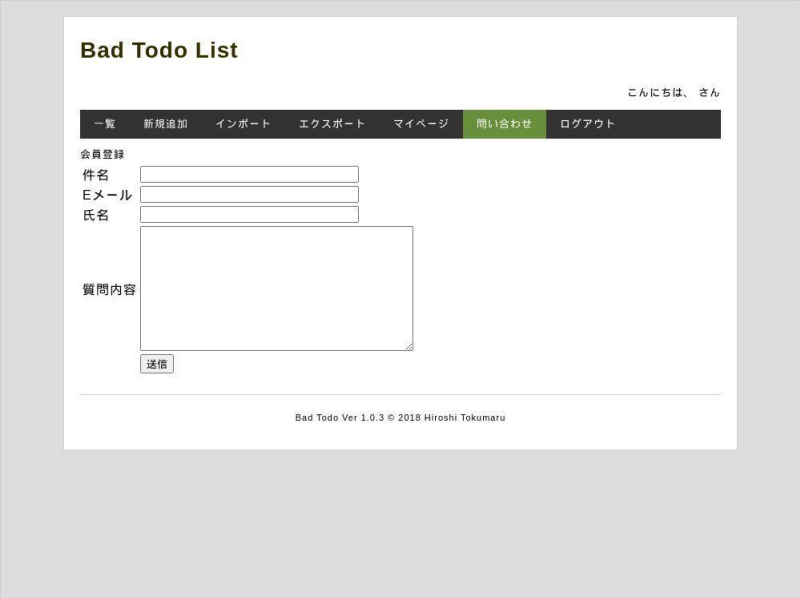
解説・対策方法

セッション ID が URL に含まれています。
そのため、利用者が目を離している間に、攻撃者が画面を盗み見たりブラウザを操作することで、攻撃者がセッション ID を取得できる危険性があります。また、Referer リクエストヘッダを介してアクセスログにセッション ID が記録されてしまう危険性があります。
対策方法としては、セッション ID を URL の一部として表示しないことが挙げられます。

参考情報

- 安全なウェブサイトの作り方 - 1.4 セッション管理の不備：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_4.html)
- Session Management - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html)

スクリーンショット



脆弱性が見つかった箇所

画面
2717.問い合わせ (https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)
リクエスト URL
https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96 (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「TODOSESSID=dh3ongu8lgofe8mb4brcol1l96」が URL に含まれていました。

画面
2718.一覧
(https://example.jp/todo/todolist.php?rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)
リクエスト URL
https://example.jp/todo/todolist.php?rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96 (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「TODOSESSID=dh3ongu8lgofe8mb4brcol1l96」が URL に含まれていました。

画面
2719.一覧 (https://example.jp/todo/todolist.php)
リクエスト URL
https://example.jp/todo/index.html?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96 (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「TODOSESSID=dh3ongu8lgofe8mb4brcol1l96」が URL に含まれていました。

画面
2720.ログアウトしています
(https://example.jp/todo/export.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)
リクエスト URL
https://example.jp/todo/export.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96 (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「TODOSESSID=dh3ongu8lgofe8mb4brcol1l96」が URL に含まれていました。

デバッグ情報の表示（レスポンスボディ）

深刻度

Low

CVSS Score: 3.7

CVSS Vector: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

概要

デバッグ情報がレスポンスボディに含まれています。そのため、アプリケーションに対する攻撃の難易度が下がる危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

14.3.1,14.3.2

解説・対策方法

デバッグ情報がレスポンスボディに含まれています。

この脆弱性により、攻撃者はアプリケーションに関する情報を取得できるため、アプリケーションに対する攻撃の難易度が下がる危険性があります。

対策方法としては、デバッグ情報を出力しないことが挙げられます。

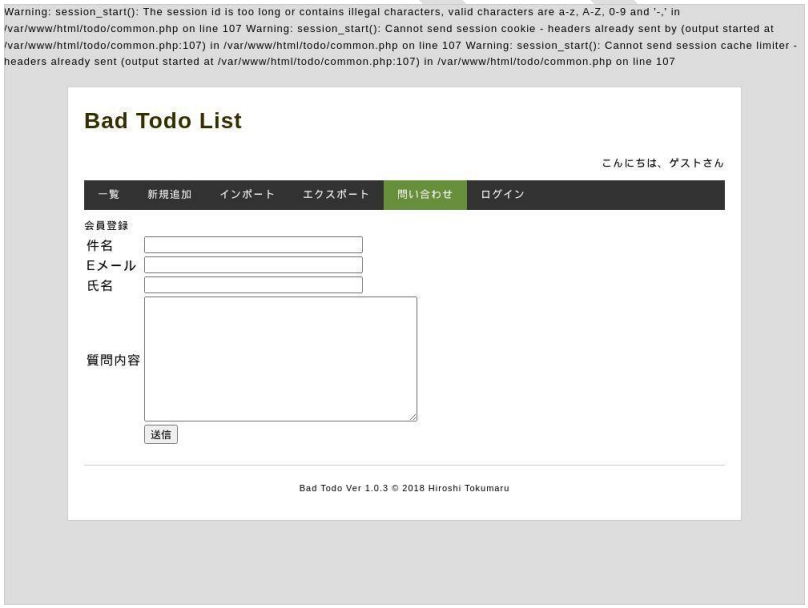
補足

ツール仕様により、この脆弱性はデバッグ情報の内容が同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。

参考情報

- Error Handling - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/Error_Handling_Cheat_Sheet.html)

スクリーンショット



脆弱性が見つかった箇所

画面

2717.問い合わせ (https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)

リクエスト URL
https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96 (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
GET パラメータ	TODOSESSID	dh3ongu8lgofe8mb4brcol1l96	../../../../../../../../../../etc/passwd	「at /var/www/html/todo/common.php:107」がレスポンスボディに含まれていました。 「common.php on line 107」がレスポンスボディに含まれていました。

画面
2724.Todo 詳細
(https://example.jp/todo/todo.php?item=1&rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)
リクエスト URL
https://example.jp/todo/todo.php?item=1&rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96 (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
GET パラメータ	item	item	item'"></script></textarea><svg/onload=confirm(3...irm(38595123)><<a|asvg/onload=confirm(38595123)>	「todo.php on line 3」がレスポンスボディに含まれていました。
GET パラメータ	item	item	item[#this.getClass().forName('java.lang.Thread').sleep(5000)][65509621]	「todo.php on line 11」がレスポンスボディに含まれていました。

不要ファイルの公開

深刻度

Low

CVSS Score: 3.7
CVSS Vector: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N

概要

不要なファイルが公開されています。そのため、ファイルを介して機微情報が漏洩する危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

4.3.2,12.5.1,14.2.2

解説・対策方法

不要なファイルが公開されています。
そのため、ファイルを介して機微情報が漏洩する危険性があります。
対策方法としては、不要なファイルを削除することや、Web サーバの設定でアクセス禁止を設定することが挙げられます。

補足

この脆弱性は、サイト全体に対して 1 回だけ検査します。

参考情報

- 4.2 Configuration and Deployment Management Testing | WSTG - Latest | OWASP (https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/02-Configuration_and_Deployment_Management_Testing/README)

スクリーンショット

脆弱性が見つかった箇所

画面
2716.一覧 (https://example.jp/todo/todolist.php)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「https://example.jp/todo/.user.ini」に直接アクセスしたところ、ファイルの内容が表示されました。

Info

X-Content-Type-Options ヘッダの不備

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

X-Content-Type-Options レスポンスヘッダに不備があります。そのため、ブラウザのセキュリティ機構が有効に機能しない危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

14.4.4

解説・対策方法

X-Content-Type-Options レスポンスヘッダは、MIME スニッフィングと呼ばれる機能を無効化するためのヘッダです。MIME スニッフィングはクロスサイトスクリプティングなどの脆弱性を発生させやすいため、無効化することでセキュリティを高めることができます。なお、X-Content-Type-Options レスポンスヘッダの値には「nosniff」のみを使用できます。

X-Content-Type-Options レスポンスヘッダがない場合、または値が「nosniff」でない場合、MIME スニッフィングが無効化されず、クロスサイトスクリプティングなどの攻撃の難易度が下がる危険性があります。

対策方法としては、X-Content-Type-Options レスポンスヘッダを追加し、値に nosniff を設定することが挙げられます。

補足

ツール仕様により、この脆弱性は登録ドメインが同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。

参考情報

- OWASP Secure Headers Project (https://owasp.org/www-project-secure-headers/)

レスポンスヘッダ

```
HTTP/1.1 200 OK
Accept-Ranges: bytes
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 90
Content-Type: text/html; charset=UTF-8
Date: Tue, 20 Dec 2022 06:39:20 GMT
ETag: "4d-56c2a2de59480-gzip"
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
Server: nginx/1.23.2
Vary: Accept-Encoding
X-UA-Compatible: IE=edge
```

脆弱性が見つかった箇所

画面

2716.一覧 (https://example.jp/todo/todolist.php)

リクエスト URL
https://example.jp/todo/ (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	X-Content-Type-Options レスポンスヘッダがありませんでした。

Strict-Transport-Security ヘッダの不備

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

Strict-Transport-Security レスポンスヘッダに不備があります。そのため、ブラウザのセキュリティ機構が有効に機能しない危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

14.4.5

解説・対策方法

Strict-Transport-Security レスポンスヘッダは、対象サイトへのアクセスを HTTPS 通信にするためのヘッダです。このヘッダが設定されると、一定期間、対象サイトに HTTP 通信で接続する時にブラウザが自動的に HTTPS 通信に変換します。

Strict-Transport-Security レスポンスヘッダがない場合、または max-age ディレクティブに 0 が指定されている場合、HTTPS 通信への変換が行われません。そのため、通信経路上の攻撃者による盗聴の難易度が下がります。

対策方法としては、Strict-Transport-Security レスポンスヘッダを追加し、max-age ディレクティブに必要十分な時間を設定することが挙げられます。

補足

ツール仕様により、この脆弱性は登録ドメインが同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。

参考情報

- HTTP Strict Transport Security - OWASP Cheat Sheet Series (https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Strict_Transport_Security_Cheat_Sheet.html)

レスポンスヘッダ

```
HTTP/1.1 200 OK
Accept-Ranges: bytes
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 90
Content-Type: text/html; charset=UTF-8
Date: Tue, 20 Dec 2022 06:39:20 GMT
ETag: "4d-56c2a2de59480-gzip"
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
Server: nginx/1.23.2
Vary: Accept-Encoding
X-UA-Compatible: IE=edge
```

脆弱性が見つかった箇所

画面

2716.一覧 (https://example.jp/todo/todolist.php)

リクエスト URL

https://example.jp/todo/ (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	Strict-Transport-Security レスポンスヘッダがありませんでした。

X-XSS-Protection ヘッダの不備

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

X-XSS-Protection レスポンスヘッダに不備があります。そのため、ブラウザのセキュリティ機構が有効に機能しない危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

無し

解説・対策方法

X-XSS-Protection レスポンスヘッダは、クロスサイトスクリプティングに対する防御機能を有効化するためのヘッダです。

X-XSS-Protection レスポンスヘッダがない場合、または値が 1 ではない場合、クロスサイトスクリプティングに対する防御機能が有効化されません。そのため、クロスサイトスクリプティングの被害が拡大する危険性があります。

対策方法としては、X-XSS-Protection レスポンスヘッダを追加し、値に 1 を設定することが挙げられます。

補足

ツール仕様により、この脆弱性は登録ドメインが同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。

参考情報

- 安全なウェブサイトの作り方 - 1.5 クロスサイト・スクリプティング：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_5.html)
- OWASP Secure Headers Project (https://owasp.org/www-project-secure-headers/)

レスポンスヘッダ

```
HTTP/1.1 200 OK
Accept-Ranges: bytes
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 90
Content-Type: text/html; charset=UTF-8
Date: Tue, 20 Dec 2022 06:39:20 GMT
ETag: "4d-56c2a2de59480-gzip"
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
Server: nginx/1.23.2
Vary: Accept-Encoding
X-UA-Compatible: IE=edge
```

脆弱性が見つかった箇所

画面

2716.一覧 (https://example.jp/todo/todolist.php)

リクエスト URL

https://example.jp/todo/ (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	X-XSS-Protection レスポンスヘッダがありませんでした。

製品情報の表示（レスポンスヘッダ）

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

製品情報がレスポンスヘッダに含まれています。そのため、アプリケーションに対する攻撃の難易度が下がる危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

14.3.3

解説・対策方法

製品情報がレスポンスヘッダに含まれています。

この脆弱性により、攻撃者はアプリケーションに関する情報を取得できるため、アプリケーションに対する攻撃の難易度が下がる危険性があります。

対策方法としては、製品情報を出力しないことが挙げられます。

補足

ツール仕様により、この脆弱性は登録ドメインが同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。

参考情報

- 4.1 Information Gathering | WSTG - Latest | OWASP (https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/01-Information_Gathering/)

レスポンスヘッダ

```
HTTP/1.1 200 OK
Accept-Ranges: bytes
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 90
Content-Type: text/html; charset=UTF-8
Date: Tue, 20 Dec 2022 06:39:20 GMT
ETag: "4d-56c2a2de59480-gzip"
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
Server: nginx/1.23.2
Vary: Accept-Encoding
X-UA-Compatible: IE=edge
```

脆弱性が見つかった箇所

画面

2716.一覧 (https://example.jp/todo/todolist.php)

リクエスト URL

https://example.jp/todo/ (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由

ー	ー	ー	ー	「Server: nginx/1.23.2」がレスポンスヘッダに含まれていました。

製品情報の表示（レスポンスボディ）

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

製品情報がレスポンスボディに含まれています。そのため、アプリケーションに対する攻撃の難易度が下がる危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

14.3.3

解説・対策方法

製品情報がレスポンスボディに含まれています。

この脆弱性により、攻撃者はアプリケーションに関する情報を取得できるため、アプリケーションに対する攻撃の難易度が下がる危険性があります。

対策方法としては、製品情報を出力しないことが挙げられます。

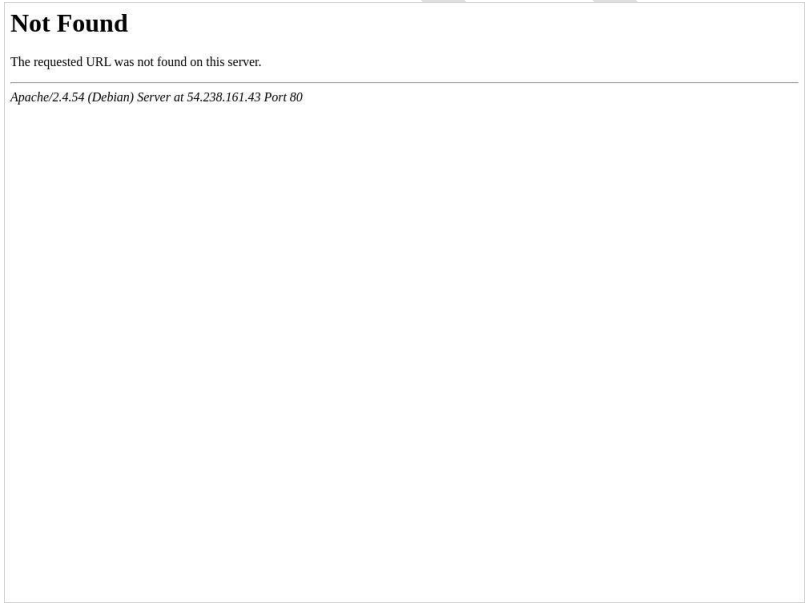
補足

ツール仕様により、この脆弱性は登録ドメインが同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。

参考情報

- 4.1 Information Gathering | WSTG - Latest | OWASP (https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/01-Information_Gathering/)

スクリーンショット



脆弱性が見つかった箇所

画面

2716.一覧 (https://example.jp/todo/todolist.php)

リクエスト URL
https://example.jp/todo/ (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
URL	ー	https://example.jp/todo/	https://example.jp/todo//'%7C%7CDBMS_PIPE.RECEIVE_MESSAGE(CHR(0)%2C5)/0%7C%7C'63303356	「Apache/2.4.54」がレスポンスボディに含まれていました。

Set-Cookie ヘッダの HttpOnly 属性の不備

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

Set-Cookie レスポンスヘッダに HttpOnly 属性がありません。そのため、クロスサイトスクリプティングの脆弱性がある場合に、クロスサイトスクリプティングの被害が拡大する危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

3.2.3,3.4.2

解説・対策方法

Set-Cookie レスポンスヘッダの HttpOnly 属性は、クロスサイトスクリプティングの被害を軽減するための属性です。具体的には JavaScript がクッキーにアクセスできないようにします。

この属性がない場合、クロスサイトスクリプティングの脆弱性がある場合に、攻撃者は JavaScript によってクッキーを取得することができます。そのため、クロスサイトスクリプティングの被害が拡大する危険性があります。

対策方法としては、HttpOnly 属性を追加することが挙げられます。

補足

ツール仕様により、この脆弱性はクッキー名が同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。また、最大 100 件を表示します。

参考情報

- 安全なウェブサイトの作り方 - 1.5 クロスサイト・スクリプティング：IPA 独立行政法人 情報処理推進機構 (https://www.ipa.go.jp/security/vuln/websecurity-HTML-1_5.html)

- Testing for Cookies Attributes | WSTG - Latest | OWASP (https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/06-Session_Management_Testing/02-Testing_for_Cookies_Attributes)

レスポンスヘッダ

```
HTTP/1.1 200 OK
Cache-Control: public, max-age=60
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 1302
Content-Type: text/html; charset=UTF-8
Date: Tue, 20 Dec 2022 06:39:48 GMT
Expires: Tue, 20 Dec 2022 06:40:48 GMT
Last-Modified: Mon, 28 Nov 2022 04:22:44 GMT
Server: nginx/1.23.2
Set-Cookie: TODOSESSID=7lff9dc91ub7l0p19a6luq2kc1; path=/; Secure
Vary: Accept-Encoding
X-Powered-By: PHP/5.3.3
X-UA-Compatible: IE=edge
```

脆弱性が見つかった箇所

画面
2716.一覧 (https://example.jp/todo/todolist.php)
リクエスト URL
https://example.jp/todo/todolist.php (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「Set-Cookie: TODOSESSID=7lff9dc91ub7l0p19a6luq2kc1; path=/; Secure」がレスポンスヘッダに指定されており、HttpOnly属性が含まれていませんでした。

画面
2760.一覧 (https://example.jp/todo/todolist.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)
リクエスト URL
https://example.jp/todo/logindo.php? (POST)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「Set-Cookie: USER=O%3A4%3A%22User%22%3A3%3A%7Bs%3...11%3A%22%00User%00super%22%3Bi%3A0%3B%7D; path=/; Secure」がレスポンスヘッダに指定されており、HttpOnly属性が含まれていませんでした。

Set-Cookie ヘッダの SameSite 属性の不備

深刻度

Info

CVSS Score: 0.0

CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:N

概要

セッション ID をセットする Set-Cookie レスポンスヘッダに SameSite 属性がありません。そのため、クロスサイトリクエストフォージェリの難易度が下がる危険性があります。

OWASP TOP 10 カテゴリー

A6:2017-不適切なセキュリティ設定

ASVS4.0 カテゴリー

3.4.3

解説・対策方法

Set-Cookie レスポンスヘッダの SameSite 属性は、外部サイトにあるリンクやフォームから自サイトにアクセスする場合などにクッキーを送信するか否かを設定する属性です。この属性の値には「None」「Lax」「Strict」の 3 種類があり、順にクッキー送信の抑制が強くなります。

この属性がない場合、デフォルト値である「Lax」の設定が利用されます。「Lax」の設定は、外部サイトにあるリンクや GET メソッドのフォームから自サイトにアクセスする場合などにクッキーを送信します。そのため、クロスサイトリクエストフォージェリの難易度が下がる危険性があります。

対策方法としては、自サイトが外部サイトからのアクセスを想定していない場合は、セッション ID をセットする Set-Cookie レスポンスヘッダに SameSite 属性を追加し、その値に「Strict」を設定することが挙げられます。

補足

ツール仕様により、この脆弱性はクッキー名が同じであるものを 1 件にまとめて表示します。そのため、他画面にも同じ脆弱性が存在する可能性があります。改修の際は他画面も確認してください。また、最大 100 件を表示します。

参考情報

- Testing for Cookies Attributes | WSTG - Latest | OWASP (https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/06-Session_Management_Testing/02-Testing_for_Cookies_Attributes)

レスポンスヘッダ

```
HTTP/1.1 200 OK
Cache-Control: public, max-age=60
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 1302
Content-Type: text/html; charset=UTF-8
Date: Tue, 20 Dec 2022 06:39:48 GMT
Expires: Tue, 20 Dec 2022 06:40:48 GMT
Last-Modified: Mon, 28 Nov 2022 04:22:44 GMT
Server: nginx/1.23.2
Set-Cookie: TODOSESSID=7lff9dc91ub7l0p19a6luq2kc1; path=/; Secure
Vary: Accept-Encoding
X-Powered-By: PHP/5.3.3
X-UA-Compatible: IE=edge
```

脆弱性が見つかった箇所

画面

2716.一覧 (https://example.jp/todo/todolist.php)

リクエスト URL

https://example.jp/todo/todolist.php (GET)

パラメータ情報

タイプ	パラメータ名	正常値	操作値	検知理由
ー	ー	ー	ー	「Set-Cookie: TODOSESSID=7lff9dc91ub7l0p19a6luq2kc1; path=/; Secure」でセッション ID をセットしており、SameSite 属性が含まれていませんでした。

各種ガイドライン対応状況

OWASP TOP 10 サマリー

OWASP TOP 10 カテゴリー	**件数**
A1:2017-インジェクション	
SQL インジェクション	: 15 件
A2:2017-認証の不備	
認証の迂回（SQL インジェクション）	: 1 件
A3:2017-機微な情報の露出	
セッション ID の表示（URL）	: 33 件
A4:2017-XML 外部エンティティ参照(XXE)	: 0 件
A5:2017-アクセス制御の不備	: 0 件
A6:2017-不適切なセキュリティ設定	
X-Content-Type-Options ヘッダの不備	: 1 件
Strict-Transport-Security ヘッダの不備	: 1 件
X-XSS-Protection ヘッダの不備	: 1 件
製品情報の表示（レスポンスヘッダ）	: 1 件
製品情報の表示（レスポンスボディ）	: 1 件
Set-Cookie ヘッダの HttpOnly 属性の不備	: 2 件
Set-Cookie ヘッダの SameSite 属性の不備	: 1 件
クリックジャッキング	: 23 件
デバッグ情報の表示（レスポンスボディ）	: 21 件
不要ファイルの公開	: 1 件
A7:2017-クロスサイトスクリプティング(XSS)	
クロスサイトスクリプティング	: 38 件
A8:2017-安全でないデシリアライゼーション	: 0 件
A9:2017-既知の脆弱性のあるコンポーネントの使用	: 0 件
Other:その他	
PHP ファイルインクルージョン	: 5 件

IPA「安全なウェブサイトの作り方」対応状況サマリー

脆弱性カテゴリー	対応状況
1) SQL インジェクション	×
2) OS コマンド・インジェクション	○
3) パス名パラメータの未チェック／ディレクトリ・トラバーサル	○
4) セッション管理の不備	×
5) クロスサイト・スクリプティング	×
6) CSRF（クロスサイト・リクエスト・フォージェリ）	○
7) HTTP ヘッダ・インジェクション	○
8) メールヘッダ・インジェクション	○
9) クリックジャッキング	×
10) バッファオーバーフロー	○
11) アクセス制御や認可制御の欠落	○

OWASP ASVS4.0 レベル 1 対応状況サマリー

項番	内容	対応状況
2.1.1	ユーザが設定するパスワードは、最低 12 文字となっている。(C6)	○
2.1.2	64 文字以上のパスワードが使用できる。(C6)	○
2.1.3	パスワードにスペースを含めることができ、切り捨てが行われない。任意で、連続した複数のスペースは 1 つにまとめてもよい。(C6)	○
2.1.4	パスワードに Unicode 文字が使用できる。単一の Unicode 符号点は文字と見なされるため、12 文字の絵文字や 64 文字の漢字が有効に使用できる必要があります。	○
2.1.5	ユーザは自身のパスワードを変更できる。	-
2.1.6	パスワード変更機能には、ユーザの現在のパスワードと新しいパスワードが必要とされる。	○
2.1.7	アカウント登録、ログインおよびパスワード変更中に送信されるパスワードが、ローカル（システムのパスワードポリシーに一致する上位 1,000 または 10,000 個の最も一般的なパスワードなど）または外部 API を使用して、侵害されたパスワードと照合される。API を使用する場合は、平文のパスワードが送信されたりパスワードの侵害状況を確認する際に使用されたりしないように、ゼロ知識証明またはその他の仕組みを使用する必要があります。パスワードが侵害された場合、アプリケーションはユーザに新しい侵害されていないパスワードの設定を要求する必要があります。(C6)	○
2.1.8	ユーザがより強力なパスワードを設定できるように、パスワード強度メータが用意されている。	-
2.1.9	使用可能な文字の種類を制限するパスワード規則がない。大文字、小文字、数字、特殊文字を要求する必要はない。(C6)	○
2.1.10	定期的なクレデンシャル変更またはパスワード履歴に関する要件がない。	-
2.1.11	パスワード入力に対して、ペースト、ブラウザのパスワードヘルパー、および外部パスワードマネージャが使用できる。	○
2.1.12	ユーザがマスクされたパスワード全体を一時的に表示するか、またはネイティブ機能としてこれを備えていないプラットフォームではパスワードの最後に入力した文字を一時的に表示するかを選択できる。	-
2.2.1	耐自動化コントロールが流出したクレデンシャルテスト攻撃、ブルートフォース攻撃、およびアカウントロックアウト攻撃の軽減に効果的となっている。このようなコントロールには最も一般的な流出パスワードのブロック、ソフトロックアウト、レート制限、CAPTCHA、試行間の遅延増加、IP アドレスの制限、または場所、デバイスへの最初のログイン、アカウントロック解除の最近の試行などのリスクベースの制限があります。一つのアカウントで一時間あたり 100 回以上の試行失敗が可能ではない。	-
2.2.2	弱いオーセンティケータ(SMS や電子メールなど) の使用がよりセキュアな認証方式の代わりとしてではなく、二次検証とトランザクション承認に限定されている。より強い方式が弱い方式の前に提示されていること、ユーザがリスクを承知していること、またはアカウント侵害のリスクを制限するために適切な対策が講じられている。	-
2.2.3	クレデンシャルのリセット、電子メールやアドレスの変更、不明な場所や危険な場所からのログインなどの認証詳細の更新後に、セキュアな通知がユーザに送信される。SMS や電子メールではなく、プッシュ通知の使用が推奨されますが、プッシュ通知がない場合、通知に機密情報が開示されていない限り SMS や電子メールは受け入れられます。	-

2.3.1	システムが生成した初期パスワードまたはアクティベーションコードは、セキュアでランダムに生成されており、6 文字以上であり、文字と数字を含むことができ、短期間で有効期限が切れる。これらの初期の秘密情報が長期間有効なパスワードになることを許可してはいけない。	-
2.5.1	システムによって生成された初期アクティベーションまたはリカバリ用の秘密情報が平文でユーザに送信されていない。(C6)	-
2.5.2	パスワードのヒントや知識ベースの認証（いわゆる「秘密の質問」）が存在しない。	○
2.5.3	パスワードクレデンシャルのリカバリによって現在のパスワードが明らかにならない。(C6)	○
2.5.4	共有アカウントまたはデフォルトアカウントが存在しない（例えば"root", "admin", "sa"）。	○
2.5.5	認証要素が変更または置き換えられた場合、ユーザにこのイベントが通知される。	-
2.5.6	パスワードを忘れた場合や他のリカバリパスは、TOTP またはその他のソフトトークン、モバイルパス、または別のオフラインリカバリ機構などのセキュアなリカバリ機構を使用する。(C6)	-
2.7.1	SMS や PSTN などの平文経路外（NIST で制限されている）オーセンティケータがデフォルトで提供されておらず、プッシュ通知などの強力な代替が最初に提供されている。	-
2.7.2	経路外検証者が 10 分後に、帯域外認証リクエスト、コード、またはトークンが期限切れにさせる。	-
2.7.3	経路外検証者の認証リクエストやコード、またはトークンが 1 回しか使用できず、元の認証リクエストに対してのみ使用可能となっている。	-
2.7.4	経路外オーセンティケータおよび検証者はセキュアで独立したチャネルを介して通信する。	-
2.8.1	時間ベースの OTP は期限切れまでの有効期間が定義されている。	-
3.1.1	アプリケーションが URL パラメータやエラーメッセージ内にセッショントークンを漏洩しない。	×
3.2.1	アプリケーションがユーザ認証時に新しいセッショントークンを生成する。(C6)	○
3.2.2	セッショントークンに少なくとも 64 ビットのエントロピーがある。(C6)	○
3.2.3	適切に保護された Cookie(セクション 3.4 を参照) や HTML 5 セッションストレージなどのセキュアな方法を使用して、アプリケーションがブラウザにセッショントークンのみを保存する。	×
3.3.1	「戻る」ボタンや下流の依拠当事者(Relying Parties)が、依拠当事者(Relying Parties)間を含め認証済みセッションを再開しないように、ログアウトおよび有効期限によってセッショントークンが無効になる。(C6)	○
3.3.2	オーセンティケータがユーザにログインしたままでいることを許可する場合、アクティブに使用されているときと、アイドル期間後の両方で、定期的に再認証が行われる。(C6)	-
3.4.1	クッキーベースのセッショントークンに Secure 属性が設定されている。(C6)	○
3.4.2	クッキーベースのセッショントークンに HttpOnly 属性が設定されている。(C6)	×
3.4.3	クロスサイトリクエストフォージェリ攻撃を抑制するために、クッキーベースのセッショントークンが SameSite 属性を利用している。(C6)	×
3.4.4	セッションクッキーの機密性を確保するために、クッキーベースのセッショントークンが"__Host-"プレフィックス（参考文献を参照）を使用している。	-
3.4.5	セッションクッキーを上書きまたは開示する可能性があるセッションクッキーを設定または使用する他のアプリケーションを持つドメイン名配下でアプリケーションが公開されている場合は、最も正確なパスを使用	-

	してクッキーベースのセッショントークンに path 属性を設定している。(C6)	
3.7.1	機密性の高いトランザクションやアカウントの変更を許可する前に、アプリケーションが完全で有効なログインセッションを確保していること、または再認証や二次検証を必要としている。	○
4.1.1	特に、クライアント側のアクセス制御が存在し、それが迂回される可能性がある場合には、アプリケーションは信頼できるサービスレイヤに対してアクセス制御ルールを適用する。	○
4.1.2	アクセス制御で使用されるすべてのユーザ属性とデータ属性およびポリシー情報は、特に認可されていない限りエンドユーザーによって操作されない。	○
4.1.3	最小権限の原則が導入されている。ユーザは認可されているものについてのみ、機能やデータファイル、URL、コントローラ、サービス、他のリソースにアクセスできる。これは、なりすましや権限昇格に対する防御となる。(C7)	○
4.1.4	デフォルト拒否の原則が存在する。これにより新規のユーザやロールは最小限の権限または権限なしで開始し、アクセスが明示的に割り当てられるまで新しい機能へのアクセスができません。(C7)	○
4.1.5	例外が発生した場合も含めて、アクセス制御がセキュアに失敗する。(C10)	○
4.2.1	機密データおよび API が、他人のレコードの作成や更新、全員のレコードの閲覧、すべてのレコードの削除など、レコードの作成、読み取り、更新、削除を目的とするダイレクトオブジェクト攻撃に対して保護されている。	○
4.2.2	認証済みの機能を保護するために、アプリケーションやフレームワークが強力な CSRF 対策メカニズムを実施していること、および有効な自動化対策や CSRF 対策が未認証機能の保護を実施している。	○
4.3.1	管理インタフェースが、不正使用を防ぐために、適切な多要素認証を使用している。	-
4.3.2	ディレクトリリスティグは、意図して許可されない限り、無効となっている。また、ファイルやディレクトリのメタデータ (Thumbs.db、.DS_Store、.git、.svn フォルダなど) の検出や開示が許可されていない。	×
5.1.1	アプリケーションが HTTP 変数汚染攻撃に対する防御策を備えている。特にアプリケーションフレームワークが、リクエストパラメータのソース(GET、POST、Cookie、ヘッダ、環境など)を区別しない場合はこの防御が必要。	-
5.1.2	フレームワークが大量のパラメータ割り当て攻撃から保護している、またはフィールドを非公開にするなど安全でないパラメータの代入を防ぐための対策が行われている。(C5)	○
5.1.3	すべての入力データがバリデーションされている。入力データには、HTML のフォームのフィールド、REST 呼び出し、クエリパラメータ、HTTP ヘッダ、Cookie、バッチファイル、RSS フィードなども含む。バリデーションはホワイトリスト方式で行う。(C5)	×
5.1.4	構造化データが強く型付けされており、使用可能な文字、長さ、パターン等の定義されたスキーマに基づいてバリデーションされる。（例：クレジットカード番号や電話番号などのバリデーションや、地区名と郵便番号等 2 つの関連するフィールドのデータが妥当なことのバリデーション）(C5)	×
5.1.5	URL のリダイレクト先と転送先がホワイトリストに登録された宛先のみ許可されている、または信頼できない可能性のあるコンテンツにリダイレクトするときに警告を表示する。	○
5.2.1	WYSIWYG エディタ等から取得した信頼できない HTML 入力が、HTML	×

	サニタイザライブラリもしくはフレームワークの機能によって適切に無害化され、入力バリデーションやエンコードにより適切に処理されている。(C5)	
5.2.2	非構造化データが無害化され、使用可能な文字や長さなど一般的な対策が適用されている。	○
5.2.3	SMTP または IMAP インジェクションから保護するため、アプリケーションがメールシステムに渡す前にユーザ入力を無害化する。	○
5.2.4	eval()もしくは動的コード実行機能を使用しない。代替方法がない場合は、実行前に含まれるユーザ入力の無害化もしくはサンドボックス化する。	○
5.2.5	テンプレートインジェクション攻撃に対して、ユーザ入力の無害化もしくはサンドボックス化によってアプリケーションが保護されている。	○
5.2.6	信頼できないデータやファイル名や URL 入力フィールドなど HTTP ファイルメタデータのバリデーションまたは無害化や、プロトコル、ドメイン、パス、ポートにホワイトリストを使用することで、アプリケーションが SSRF 攻撃から保護されている。	○
5.2.7	ユーザの指定した SVG スクリプト化可能コンテンツ（特に XSS に関連するインラインスクリプトや foreignObject）に対して、アプリケーションが無害化またはサンドボックス化している。	○
5.2.8	ユーザ指定の Markdown、CSS、XSL スタイルシート、BBCode のようなスクリプト可能もしくは式のテンプレート言語コンテンツに対して、アプリケーションが無害化、無効化またはサンドボックス化されている。	×
5.3.1	出力エンコーディングがインタプリタとコンテキストが要求するものに関連している。例えば特に信頼できない入力("ねこ" や"O'Hara" など、Unicode 文字やアポストロフィを含む名前など) に対して、HTML 値、HTML 属性、JavaScript、URL パラメータ、HTML ヘッダ、SMTP、その他コンテキストが必要とするものに対して個別のエンコードを使用する。(C4)	×
5.3.2	どの Unicode 文字でも有効かつ安全に処理されるように、出力エンコーディングがユーザが選択した文字コードセット、ロケールが保持されている。(C4)	○
5.3.3	HTML や他の Web クライアントコード中に存在するすべての文字列変数が、コンテキストに応じて適切に手動でエンコードされる、もしくはコンテキストに応じて自動的にエンコードを行うテンプレートを使用しており、アプリケーションが、反射型、格納型、および DOM ベースクロスサイトスクリプティング(XSS) 攻撃の影響を受けない。(C4)	×
5.3.4	データ選択またはデータベースクエリ（例、SQL、HQL、ORM、NoSQL）がクエリのパラメータ化、ORM、エンティティフレームワークもしくは他の方法により保護されており、データベースインジェクション攻撃の影響を受けない。(C3)	×
5.3.5	パラメータ化もしくはより安全な機構が存在しない場合、SQL インジェクションから保護するための SQL エスケープの使用など、コンテキスト固有の出力エンコーディングによりインジェクション攻撃から保護されている。(C3, C4)	×
5.3.6	eval 攻撃、リモート JavaScript インクルード、CSP バイパス、DOM XSS、JavaScript の式の評価を含む JavaScript もしくは JSON インジェクション攻撃からアプリケーションが保護されている。(C4)	×
5.3.7	アプリケーションが LDAP インジェクションの影響を受けない、またはセキュリティ管理策によって LDAP インジェクションが防止される。(C4)	○
5.3.8	OS コマンドインジェクションに対して保護していること、およびオペレーティングシステムコールがパラメータ化された OS クエリを使用、もしくはコンテキストコマンドライン出力エンコーディングを使用する。	○

	(C4)	
5.3.9	アプリケーションが、リモートファイルインクルード(RFI) やローカルファイルインクルード(LFI) の影響を受けない。	×
5.3.10	アプリケーションが XPath インジェクション攻撃や XML インジェクション攻撃から保護されている。(C4)	○
5.5.1	シリアライズされたオブジェクトが整合性チェックを使用していること、または悪意のあるオブジェクトの作成やデータの改ざんを防ぐために暗号化されている。(C5)	○
5.5.2	アプリケーションが XML パーサを可能な限り最も制限の厳しい構成のみを使用するように正しく制限し、外部エンティティの解決などの危険な機能を無効にして XXE を防ぐようにしている。	○g
5.5.3	信頼できないデータのデシリアライズが回避されている、またはカスタムコードとサードパーティのライブラリ（JSON、XML、YAML パーサなど）の両方で保護されている。	○
5.5.4	ブラウザもしくは JavaScript ベースのバックエンドで JSON をパースするときは JSON.parse を使用する。eval() を使用しない。	○
6.2.1	すべての暗号化モジュールにおいて、処理に失敗した場合の安全対策が施されており、オラクルパディング攻撃を許さない方法でエラーが処理される。	○
7.1.1	アプリケーションがクレデンシャルまたは支払い情報をログに保存していない。セッショントークンは、不可逆的なハッシュ形式でのみログに保存する。(C9, C10)	-
7.1.2	現地のプライバシー法または関連するセキュリティポリシーで定義されているその他のセンシティブなデータをログに記録していない。(C9)	-
7.4.1	予期しないエラーまたはセキュリティ上重要なエラーが発生したときに、サポート担当者が調査に使用できる一意の ID とともに一般的なメッセージが表示される。(C10)	-
8.2.1	最新のブラウザで機密データがキャッシュされないように、アプリケーションは十分なキャッシュ防止ヘッダを設定する。	-
8.2.2	クライアントの記憶域（HTML5 のローカルストレージ、セッションストレージ、IndexedDB、通常の Cookie、FlashCookie など）に保存されるデータにセンシティブなデータや PII が含まれていない。	○
8.2.3	セッションの終了後、ブラウザの DOM など認証されたデータがクライアントの記憶域から消去される。	○
8.3.1	センシティブなデータが HTTP メッセージボディまたはヘッダでサーバに送信される。どんな HTTP verb のクエリストリングパラメータにもセンシティブなデータが含まれない。	×
8.3.2	ユーザが自身のデータをオンデマンドで、削除またはエクスポートできる。	-
8.3.3	個人情報の収集および利用に関する明確なポリシーが、ユーザに提供されている。個人情報が何らかの方法で使用される前に、オプトイン方式で個人情報利用に関する同意を得ている。	-
8.3.4	アプリケーションにより作成および処理される、すべてのセンシティブなデータを特定している。センシティブなデータを処理する方法に関するポリシーが、策定されている。(C8)	-
9.1.1	安全な TLS がすべてのクライアント接続に使用されており、安全でないプロトコルや非暗号化プロトコルにフォールバックしない。(C8)	×
9.1.2	オンラインまたは最新の TLS テストツールを使用して、強力なアルゴリズム、暗号、プロトコルのみが有効であり、最も強力なアルゴリズムと暗号が優先的に設定されている。	×
9.1.3	古いバージョンの SSL や TLS プロトコル（SSLv2、SSLv3、TLS 1.0、TLS 1.1）、アルゴリズム、暗号および構成が無効になっている。最新バージョンの TLS を優先する暗号スイートに設定する。	×

10.3.1	アプリケーションにクライアントまたはサーバの自動更新機能がある場合は、安全なチャネルを経由して、デジタル署名がされていることを確認する。インストールまたは実行する前に、アップデートするコードのデジタル署名を検証する必要がある。	-
10.3.2	アプリケーションで、コード署名やサブリソースの完全性などの保護が使用されている。加えて、信頼できない場所やインターネットから取得したモジュール、プラグイン、コード、ライブラリなどをロードまたは実行しない。	○
10.3.3	期限切れドメイン名、期限切れ DNS ポインタまたは CNAME、パブリックソースコードリポジトリでの期限切れプロジェクト、一時的なクラウド API、サーバレス機能、ストレージバケット（autogen-bucket-id.cloud.example.com）など、DNS エントリまたは DNS サブエントリに依存している場合、アプリケーションがサブドメイン奪取（subdomain takeover）から保護されている。アプリケーションの保護は使用する DNS 名の有効期限または変更を定期的にチェックすることを含めます。	○
11.1.1	アプリケーションが同じユーザのビジネスロジックフローを手順通り、省略せずに処理する。	○
11.1.2	アプリケーションは、すべてのステップが人間の実際に処理する時間で処理されたビジネスロジックフローのみ処理する。（送信されるのが早すぎるトランザクションは処理されない。）	×
11.1.3	アプリケーションに適切な制限が実装されており、特定のビジネス活動やトランザクションに対し、ユーザごとに適用されている。	○
11.1.4	データの流出、ビジネスロジック要求、過剰なファイルのアップロード、DoS 攻撃(denial of service attacks)を検出および保護するために、アプリケーションに充分な自動攻撃を検知し防止する制限がある。	-
11.1.5	アプリケーションは脅威モデリング(threat modelling)または類似の方法を使用して特定されたビジネスリスクから保護するために、ビジネスロジックの制限またはバリデーションがある。	○
12.1.1	ストレージを圧迫させたり、DoS 攻撃を引き起こしたりする可能性のある大きなファイルをアプリケーションが受け付けない。	-
12.3.1	パストラバーサルから保護するために、ユーザが送信したファイル名のメタデータがシステムまたはフレームワークファイルや URL API で直接使用されていない。	○
12.3.2	ローカルファイル（LFI）の漏えい、作成、更新、または削除を防止するために、ユーザが送信したファイル名のメタデータをバリデートもしくは無視する。	○
12.3.3	SSRF にも繋がる可能性があるリモートファイル(RFI)の漏えいまたは実行を防ぐために、ユーザが送信したファイル名のメタデータをバリデートもしくは無視する。	×
12.3.4	アプリケーションを反射型ファイルダウンロード（RFD）から保護するため、ユーザが送信したファイル名（JSON、JSONP または URL パラメータの中にある）をバリデートまたは無視し、レスポンスの Content-Type ヘッダは text/plain に設定、および Content-Disposition ヘッダは固定ファイル名を指定する。	○
12.3.5	OS コマンドインジェクションから保護するために、信頼できないファイルメタデータをシステム API またはライブラリで直接使用しない。	○
12.4.1	信頼できない場所から取得したファイルは、Web ルート以外にパーミッションを制限した上で保存し、可能な限り厳密なバリデーションを行う。	○
12.4.2	信頼できない場所から取得したファイルが、アプリケーションが想定する種類であることを検証し、既知の悪性コンテンツがアップロードされるのを防止するためにアンチウイルスソフトで検査する。	×
12.5.1	意図しない情報やソースコードの漏えいを防ぐために、特定のファイル	×

	拡張子を持つファイルのみを処理するように Web 層が構成されている。例えば、バックアップファイル（.bak など）、一時作業ファイル（.swp など）、圧縮ファイル（.zip、.tar.gz など）およびエディタで一般的に使用されるその他の拡張子は、必要ない限り処理しない。	
12.5.2	アップロードされたファイルへの直接のリクエストが HTML/JavaScript コンテンツとして実行されない。	○
12.6.1	Web サーバまたはアプリケーションサーバが、リクエストを送信したりデータ/ファイルをロードしたりできるように、リソースまたはシステムにホワイトリストが構成されている。	×
13.1.1	SSRF 攻撃や RFI 攻撃で使用される可能性があるような、異なる URI またはファイルのパーサを悪用する攻撃を回避するために、すべてのアプリケーションコンポーネントが同じエンコードおよびパーサを使用する。	×
13.1.2	Web サービスアプリケーション内の管理機能にアクセスできるのは、アクセスが許可された管理者のみとなっている。	○
13.1.3	API URL が API Key やセッショントークンなどの機密情報を公開していない。	×
13.2.1	保護された API またはリソースに対して通常ユーザが DELETE または PUT を使用するのを防ぐなど、有効化されている RESTful HTTP メソッドが、ユーザまたはアクションにとって妥当となっている。	○
13.2.2	JSON スキーマバリデーションが設定され、入力を受け付ける前に確認されている。	×
13.2.3	Cookie を使用する RESTful Web サービスが、次のうち少なくとも 1 つ以上を使って、クロスサイトリクエストフォージェリから保護されている。三重または二重送信クッキーパターン（参考文献）、CSRF ノンス、ORIGIN リクエストヘッダのチェック。	×
13.3.1	適切に形成された XML 文書を確保するために、XSD スキーマバリデーションに続いて、そのデータの処理が行われる前に各入力フィールドのバリデーションが実施されている。	×
14.2.1	すべてのコンポーネントが最新となっている。できればビルド時またはコンパイル時にディペンデンシチェッカを使用する。(C2)	×
14.2.2	サンプルアプリケーション、プラットフォームのドキュメント、デフォルトまたはサンプルのユーザなど、不要な機能、ドキュメント、サンプル、コンフィギュレーションがすべて削除されている。	×
14.2.3	JavaScript ライブラリ、CSS スタイルシート、Web フォントなどのアプリケーション資産がコンテンツ配信ネットワーク（CDN）または外部プロバイダで外部的にホストされている場合、資産の整合性を検証するためにサブリソース完全性（SRI）が使用されている。	○
14.3.1	Web またはアプリケーションサーバとフレームワークのエラーメッセージが、意図しないセキュリティ情報の開示を排除するために、ユーザが実行可能なカスタマイズされた応答を返すように構成されている。	×
14.3.2	Web またはアプリケーションサーバとアプリケーションフレームワークのデバッグモードが運用環境で無効になっていることを確認して、デバッグ機能、開発者コンソール、および意図しないセキュリティ開示を排除する。	×
14.3.3	HTTP ヘッダまたは HTTP レスポンスの一部がシステムコンポーネントの詳細なバージョン情報を公開していない。	×
14.4.1	すべての HTTP レスポンスに、安全な文字セット（例：UTF-8、ISO 8859-1）を指定するコンテントタイプヘッダが含まれている。	×
14.4.2	すべての API レスポンスに Content-Disposition:attachment; filename="api.json"が含まれている（または他のコンテントタイプの適切なファイル名）。	○
14.4.3	HTML、DOM、JSON、JavaScript インジェクションの脆弱性などの XSS	×

	攻撃の影響を軽減するのに役立つ Content Security Policy(CSPv2) が配置されている。	
14.4.4	すべてのレスポンスに X-Content-Type-Options：nosniff が含まれている。	×
14.4.5	HTTP Strict Transport Security ヘッダがすべてのレスポンスとすべてのサブドメインに含まれている。例えば、Strict-Transport-Security：max-age = 15724800;	×
14.4.6	「no-referrer」や「same-origin」のような、適切な「Referrer-Policy」ヘッダが含まれている。	-
14.4.7	サードパーティのサイトに埋め込むべきではないサイトで、適切なヘッダが（X-Frame-Options または Content-Security-Policy：frame-ancestors）が、サイトで使用されている。	×
14.5.1	アプリケーションサーバが、pre-flight OPTIONS を含む、アプリケーションまたは API で使用されている HTTP メソッドのみを受け入れる。	○
14.5.2	提供された Origin ヘッダは、攻撃者によって簡単に変更できるため、認証やアクセス制御の判断に使用されていない。	○
14.5.3	オリジン間リソース共有（CORS）の Access-Control-Allow-Origin ヘッダが信頼できるドメインの厳密なホワイトリストを使用して照合し、「null」オリジンをサポートしていない。	○

画面毎のスキャン結果サマリー

画面タイトル (URL)	件数
2716.一覧 (https://example.jp/todo/todolist.php)	18 件
2717.問い合わせ (https://example.jp/todo/inquery.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	3 件
2718.一覧 (https://example.jp/todo/todolist.php?rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	2 件
2719.一覧 (https://example.jp/todo/todolist.php)	2 件
2720.ログアウトしています (https://example.jp/todo/export.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	1 件
2721.ログアウトしています (https://example.jp/todo/newtodo.php?rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	1 件
2722.ログアウトしています (https://example.jp/todo/import.php?TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	1 件
2723.ログイン (https://example.jp/todo/login.php?url=todolist.php&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	2 件
2724.Todo 詳細 (https://example.jp/todo/todo.php?item=1&rnd=63a1459fa8b26&TODOSESSID=dh3ongu8lgofe8mb4brcol1l96)	3 件
2726.一覧 (https://example.jp/todo/todolist.php?TODOSESSID=h43iejuuqu85bi5fj9e5frt0n3&key=a)	4 件
2727.ログアウトしています (https://example.jp/todo/editlist.php)	2 件
2728.問い合わせ (https://example.jp/todo/inquerydo.php)	4 件

※サンプルのため対象情報は省略しています。

スキャン項目一覧

主なスキャン項目を以下に列挙します。

スキャンルール	スキャン項目	概要
Web アプリケーションスキャン	A1:2017-インジェクション	SQL インジェクション、NoSQL インジェクション、OS コマンドインジェクション、LDAP インジェクションといったインジェクションに関する脆弱性は、コマンドやクエリの一部として信頼されないデータが送信される場合に発生します。攻撃コードはインタープリタを騙し、意図しないコマンドの実行や、権限を有していないデータへのアクセスを引き起こします。
	A2:2017-認証の不備	認証やセッション管理に関連するアプリケーションの機能は、不適切に実装されていることがあります。不適切な実装により攻撃者は、パスワード、鍵、セッショントークンを侵害したり、他の実装上の欠陥により、一時的または永続的に他のユーザーの認証情報を取得します。
	A3:2017-機微な情報の露出	多くのウェブアプリケーションや API では、財務情報、健康情報や個人情報といった機微な情報を適切に保護していません。攻撃者は、このように適切に保護されていないデータを窃取または改ざんして、クレジットカード詐欺、個人情報の窃取やその他の犯罪を行う可能性があります。機微な情報は特別な措置を講じないでいると損なわれることでしょう。保存や送信する時に暗号化を施すことや、ブラウザ経由でやり取りを行う際には安全対策を講じることなどが必要です。
	A4:2017-XML 外部エンティティ参照(XXE)	多くの古くて構成の悪い XML プロセッサーにおいては、XML 文書内の外部エンティティ参照を指定することができます。 外部エンティティは、ファイル URI ハンドラ、内部ファイル共有、内部ポートス

		キャン、リモートコード実行、DoS（サービス拒否）攻撃により、内部ファイルを漏えいさせます。
	A5:2017-アクセス制御の不備	権限があるもののみが許可されていることに関する制御が適切に実装されていないことがあります。攻撃者は、このタイプの脆弱性を悪用して、他のユーザのアカウントへのアクセス、機密ファイルの表示、他のユーザのデータの変更、アクセス権の変更など、権限のない機能やデータにアクセスします。
	A6:2017-不適切なセキュリティ設定	不適切なセキュリティの設定は、最も一般的に見られる問題です。これは通常、安全でないデフォルト設定、不完全またはアドホックな設定、公開されたクラウドストレージ、不適切な設定の HTTP ヘッダ、機微な情報を含む冗長なエラーメッセージによりもたらされます。 すべてのオペレーティングシステム、フレームワーク、ライブラリ、アプリケーションを安全に設定するだけでなく、それらに適切なタイミングでパッチを当てることやアップグレードをすることが求められます。
	A7:2017-クロスサイトスクリプティング(XSS)	XSS の脆弱性は、適切なバリデーションやエスケープ処理を行っていない場合や、HTML や JavaScript を生成できるブラウザ API を用いているユーザ入力データで既存の Web ページを更新する場合に発生します。 XSS により攻撃者は、被害者のブラウザでスクリプトを実行してユーザーセッションを乗っ取ったり、Web サイトを改ざんしたり、悪意のあるサイトにユーザーをリダイレクトします。
	A8:2017-安全でないデシリアライゼーション	安全でないデシリアライゼーションは、リモートからのコード実行を誘発します。デシリアライゼーションの欠陥によるリモートからのコード実行に至らない場合でさえ、リプレイ攻撃やインジェクショ

		ン攻撃、権限昇格といった攻撃にこの脆弱性を用います。
	A9:2017-既知の脆弱性のあるコンポーネントの使用	ライブラリ、フレームワークやその他ソフトウェアモジュールといったコンポーネントは、アプリケーションと同等の権限で動いています。脆弱性のあるコンポーネントが悪用されると、深刻な情報損失やサーバの乗っ取りにつながります。既知の脆弱性があるコンポーネントを利用しているアプリケーションや API は、アプリケーションの防御を損ない、様々な攻撃や悪影響を受けることになります。
	Other:その他	OWASP TOP 10 にカテゴライズされない脆弱性です。