NeoShield Quantum X

フリーランス開発者と Web 制作会社のための Web サイトセキュリティレポート

顧客向けレポート

Webサイトセキュリティ改善レポート

www.client-shop.example

公開されているセキュリティ設定の受動的な診断



観測された設定スコア

カバー率 100% · 確度 中

提出先	**株式会社サンプル小売（デモ）**
作成者	**貴社名** · NeoShield Quantum X
Webサイト	https://www.client-shop.example/
レポートID	DEMO-2026-0001
診断日	2026-10-01 · **パッシブ診断**
再確認の診断日	2026-10-08
状況	**確認済みの指摘 4 件（高・重大 0 件）、状況に応じて確認する観測 0 件**

デモ用データを使用したサンプルレポートです。

取扱注意

本レポートは上記Webサイトのセキュリティ設定について記載したものです。Webサイトの管理責任者にのみ共有してください。本レポートはペネトレーションテストではなく、Webサイトが安全であることを証明するものではありません。

目次

エグゼクティブサマリー 3
スコアの算出方法 4
指摘の一覧 5
検出項目、証拠、改善方法 6
合格したチェック 14
修正ロードマップ 14
付録 A：参照規格の対応表 15
付録 B：用語の定義 15
付録 C：範囲、許可、方法 15
付録 D：制約 16
付録 E：レポートの完全性 16

各セクション名はクリックできます。同じセクションは PDF のしおりからも開けます。

エグゼクティブサマリー

観測された設定スコア	カバー率	状況確認が必要	直ちに対応
55/100 · D	100%	0	確認済みなし

該当チェック 17 件中 17 件完了 · 確認済みの指摘 4 件 · 修正確認済み 2 件 · 確度：中 · パッシブ診断

このスコアは、この診断で完了した、公開されている設定のチェックだけを対象にしています。アプリケーションの脆弱性や全体的なビジネスリスクを測るものではありません。
確認済みの指摘は直接観測されたもので、スコアを下げます。状況に応じて確認する観測は、Web サイト、DNS、メールの使い方に左右されるもので、スコアは下げません。

このレポートの全指摘の重要度



対応・確認が必要な項目（領域別）



ビジネスにとっての意味

Webサイトは有効な証明書による HTTPS で利用できますが、ブラウザのセキュリティ対策がいくつか不足しています。未対応の主な課題は、多層防御の対策である Content-Security-Policy が設定されていないことと、セッション Cookie に Secure と HttpOnly の属性がないことです。2件は修正され、2026-10-08 の確認スキャンで確認されました。残る4件は、設定例とともに優先度順に記載しています。 この診断は、記載の時点で公開されていた設定だけを対象にしています。認証が必要な機能、ソースコード、内部インフラ、業務ロジック、アプリケーションの脆弱性全般は評価していません。

技術的な概要

公開されている HTTP レスポンスヘッダー、TLS 設定、Cookie、リダイレクト、メール認証の DNS レコード、よく公開されてしまうパスを対象とした受動的な診断です。検出項目6件（高1件、中3件、低2件）、修正確認済み2件、合格したチェック11件。DKIM は未評価です（セレクターの指定なし）。

確認が必要な項目

1	中	Content-Security-Policy ヘッダーが設定されていない	30日以内に修正
2	中	セッション Cookie に Secure と HttpOnly 属性がない	30日以内に修正
3	中	クリックジャッキング対策がない	30日以内に修正
4	低	サーバーソフトウェアのバージョンが表示されている	90日以内に確認

修正確認後：100点中73点、評価 C

スコアの算出方法

スコアは 100 から始まります。確認済みの未解決の指摘ごとに、重要度に応じた固定点（重大 25、高 15、中 8、低 3）を差し引きます。同じ原因は1回だけ差し引きます。要確認の項目、情報、実行できなかったチェックは差し引きません。

検出項目	重要度	点数
Strict-Transport-Security ヘッダーがない	高	−15
Content-Security-Policy ヘッダーが設定されていない	中	−8
セッション Cookie に Secure と HttpOnly 属性がない	中	−8
クリックジャッキング対策がない	中	−8
X-Content-Type-Options ヘッダーがない	低	−3
サーバーソフトウェアのバージョンが表示されている	低	−3
差し引き合計		−45

100 − 45 = 55/100、評価 D。

評価：A 90〜100、B 80〜89、C 70〜79、D 55〜69、F 55未満。

カバー率と確度

カバー率	100% · 該当チェック 17 件中 17 件完了
確度	中 · 確認済みの指摘の一部に解釈が必要
未完了	DKIM（セレクターの指定なし）
対象外	1 件（この Web サイトには該当しません）

カバー率は、予定した診断のうち結果が得られた割合です。実行できなかったチェックは、完了させるか手作業で確認できるよう一覧にしています。合格としては数えません。

指摘の一覧

番号	検出項目	重要度	優先度	分類	対応状況
F-01	Content-Security-Policy ヘッダーが設定されていない	中	P2	確認済みの指摘	引き続き観測
F-03	セッション Cookie に Secure と HttpOnly 属性がない	中	P2	確認済みの指摘	引き続き観測
F-04	クリックジャッキング対策がない	中	P2	確認済みの指摘	引き続き観測
F-06	サーバーソフトウェアのバージョンが表示されている	低	P3	確認済みの指摘	引き続き観測
F-02	Strict-Transport-Security ヘッダーがない	高	P1	確認済みの指摘	解決
F-05	X-Content-Type-Options ヘッダーがない	低	P3	確認済みの指摘	解決

優先度：P0 は直ちに対応、P1 は7日以内に修正、P2 は30日以内、P3 は90日以内に確認、P4 は任意の強化項目。スコアを下げるのは確認済みの指摘だけです。付録 B を参照してください。

検出項目、証拠、改善方法

F-01 Content-Security-Policy ヘッダーが設定されていない 中

検出項目ID	DEMO-2026-0001-F-01 · ??? csp
分類	確認済みの指摘: Web サイトの応答や DNS で直接観測されたものです。スコアを下げます。
対応状況	引き続き観測
修正の優先度	P2 · 30 日以内に修正
スコアへの影響	−8 点
確度	高
対象URL	https://www.client-shop.example/
検出方法	HTTP レスポンスヘッダーの受動的な確認（GET /、リダイレクト追従）。
確認日時	2026-10-01T09:14:05Z
証拠	GET / は HTTP/2 200 を返し、Content-Security-Policy ヘッダーも Content-Security-Policy-Report-Only ヘッダーもありませんでした。
影響の理由	Content Security Policy がない場合、コンテンツインジェクションやクロスサイトスクリプティングの脆弱性が悪用されたときに、その影響を抑えるためにブラウザが適用できる制限が少なくなります。
顧客向けの説明	どのプログラムの実行を許すかを、サイトがブラウザーに伝えていません。万一、攻撃者がページにプログラムを紛れ込ませた場合、訪問者のブラウザーでの実行を止めるものがありません。厳しいポリシーはサイトの一部を壊すことがあるため、導入にはテストが必要です。
ビジネスへの影響	プラグインなどを通じてスクリプトインジェクションの欠陥が生じた場合に、注入されたコードが読み込んだり送信したりできる内容を制限する、ブラウザ側の対策がありません。
技術的な影響	ブラウザによる script-src、object-src、base-uri、frame-ancestors の制限が適用されていません。
制約事項	Content Security Policy がないことだけで、Webサイトにクロスサイトスクリプティングの脆弱性があると証明されるわけではありません。CSP は多層防御のための対策です。
改善方法	まずレポート専用のポリシーで始め、レポートを確認してから適用します。
確認の基準	再診断で、スクリプトの読み込み元を制限する強制モードの Content-Security-Policy ヘッダーが確認されること。レポート専用ポリシーの場合は一部解決として記録されます。
確認状況	最新の診断でも残存
人による確認	修正を計画する前に必要です。
参照規格	OWASP A05:2021 · CWE-693 · NIST SP 800-53 SC-18

プラットフォーム別の設定例

Nginx

```
# server { } block (or http { })
add_header Content-Security-Policy-Report-Only "default-src 'self'; object-src 'none'; base-uri 'self';
frame-ancestors 'self'" always;
```

まず Report-Only モードで始め、ページで実際に使っている読み込み元を追加してから、ヘッダー名を Content-Security-Policy に変えて適用します。 反映：nginx -t && systemctl reload nginx。location ブロック内の add_header は、server レベルで設定したヘッダーを置き換えます。

Apache HTTP Server

```
# Requires mod_headers (a2enmod headers)
Header always set Content-Security-Policy-Report-Only "default-src 'self'; object-src 'none'; base-uri 'self';
 frame-ancestors 'self'"
```

まず Report-Only モードで始め、ページで実際に使っている読み込み元を追加してから、ヘッダー名を Content-Security-Policy に変えて適用します。 VirtualHost または .htaccess に記述し、apachectl configtest を実行してから再読み込みします。

Microsoft IIS

```
<?xml version="1.0" encoding="utf-8"?>
<!-- web.config: merge into the existing <configuration> element -->
<configuration>
  <system.webServer>
    <httpProtocol>
      <customHeaders>
        <add name="Content-Security-Policy-Report-Only" value="default-src 'self'; object-src 'none'; base-uri
 'self'; frame-ancestors 'self'" />
      </customHeaders>
    </httpProtocol>
  </system.webServer>
</configuration>
```

まず Report-Only モードで始め、ページで実際に使っている読み込み元を追加してから、ヘッダー名を Content-Security-Policy に変えて適用します。 この web.config を持つサイトまたはアプリケーションに適用されます。

Cloudflare

```
Rules > Transform Rules > Modify Response Header > Create rule
1. Apply to: All incoming requests (or this hostname)
2. Action: Set static
3. Header name: Content-Security-Policy-Report-Only
4. Value: default-src 'self'; object-src 'none'; base-uri 'self'; frame-ancestors 'self'
5. Deploy
```

まず Report-Only モードで始め、ページで実際に使っている読み込み元を追加してから、ヘッダー名を Content-Security-Policy に変えて適用します。 Cloudflare は、プロキシ経由（オレンジクラウド）の通信に対してのみエッジでヘッダーを設定します。

WordPress

```
<?php
// In a small site-specific plugin (or the child theme's functions.php)
add_action( 'send_headers', function () {
    header(
        "Content-Security-Policy-Report-Only: "
        . "default-src 'self'; "
        . "object-src 'none'; "
        . "base-uri 'self'; "
        . "frame-ancestors 'self'"
    );
} );
```

まず Report-Only モードで始め、ページで実際に使っている読み込み元を追加してから、ヘッダー名を Content-Security-Policy に変えて適用します。 ヘッダーは Web サーバーまたは CDN で設定するのが望ましく、これは WordPress しか操作できない場合の方法です。

AWS CloudFront

```
CloudFront > Policies > Response headers > Create response headers policy
1. Security headers or Custom headers: add Content-Security-Policy-Report-Only = default-src 'self';
object-src 'none'; base-uri 'self'; frame-ancestors 'self'
2. Origin override: enabled
3. Attach the policy to the distribution's cache behavior(s)
```

まず Report-Only モードで始め、ページで実際に使っている読み込み元を追加してから、ヘッダー名を Content-Security-Policy に変えて適用します。 変更は数分以内にすべてのエッジロケーションに反映されます。

> 設定の変更は、本番環境に反映する前にステージング環境で検証してください。ポリシーに誤りがあると、スクリプト、サインイン、購入手続き、埋め込みサービス、アクセス解析、外部サービスとの連携が動作しなくなる場合があります。

参考資料

MDN: Content-Security-Policy
OWASP Content Security Policy Cheat Sheet
developer.mozilla.org
owasp.org

F-03 セッション Cookie に Secure と HttpOnly 属性がない

中

検出項目ID	DEMO-2026-0001-F-03 · ??? cookies
分類	**確認済みの指摘**: Web **サイトの応答や** DNS **で直接観測されたものです。スコアを下げます。**
対応状況	**引き続き観測**
修正の優先度	P2 · 30 **日以内に修正**
スコアへの影響	**ー8 点**
確度	**高**
対象URL	https://www.client-shop.example/
検出方法	**公開ページの** Set-Cookie **ヘッダーの受動的な確認。**
確認日時	2026-10-01T09:14:05Z
証拠	Set-Cookie: PHPSESSID=[redacted]; path=/
影響の理由	Secure **がないと** Cookie **が暗号化されていない** HTTP **で送信されることがあり、**HttpOnly **がないとページ内のスクリプトから読み取れます。**
顧客向けの説明	**サイトのログイン用** Cookie **に標準的な保護が設定されていないため、ログイン中のセッションを盗まれたり、他のサイトから悪用されたりしやすくなっています。修正はサイトまたはプラットフォームの設定変更です。**
ビジネスへの影響	**ログイン中のお客様のセッション盗用やアカウント乗っ取りのリスクが高まります。**
技術的な影響	**セッション** ID **がページ内のスクリプトから読み取れ、平文の** HTTP **で送信される可能性があります。**
制約事項	**受動的な診断で確認できるのは、公開ページで設定される** Cookie **だけです。**
改善方法	**セッション** Cookie **に** Secure**、**HttpOnly**、**SameSite=Lax **を設定します。**
確認の基準	**再診断で、セッション** Cookie **に** Secure **と** HttpOnly **の属性が確認されること。**
確認状況	**最新の診断でも残存**
人による確認	**修正を計画する前に必要です。**
参照規格	CWE-614 · CWE-1004 · CWE-1275

プラットフォーム別の設定例

Nginx

```
# When Nginx proxies an application (nginx 1.19.3+)
proxy_cookie_flags ~ secure httponly samesite=lax;
```

プロキシ先のアプリケーションが設定する Cookie **に適用されます。アプリケーション自体で属性を修正するのが望ましい方法です。**

Apache HTTP Server

```
# Requires mod_headers. Prefer fixing the application; use this as a stopgap.
Header edit Set-Cookie "(?i)^((?:(?!;\s?HttpOnly).)+)$" "$1; HttpOnly"
Header edit Set-Cookie "(?i)^((?:(?!;\s?Secure).)+)$" "$1; Secure"
```

各ルールは、属性がない場合にのみ追加します。アプリケーションが設定する Cookie **に適用されるよう、**"always" **ではなく** "Header edit" **を使ってください。ログインが引き続き動作することを確認してください。**

Microsoft IIS

```
<?xml version="1.0" encoding="utf-8"?>
<!-- web.config (ASP.NET): merge into the existing <configuration> element -->
<configuration>
  <system.web>
    <httpCookies requireSSL="true" httpOnlyCookies="true" sameSite="Lax" />
  </system.web>
</configuration>
```

sameSite 属性には .NET Framework 4.7.2 以降が必要です。

WordPress

```
; php.ini (or .user.ini) for the PHP session cookie
session.cookie_secure = 1
session.cookie_httponly = 1
session.cookie_samesite = "Lax"
```

WordPress 自身のログイン Cookie は、HTTPS のサイトではすでに HttpOnly と Secure が付いています。独自の Cookie を設定するプラグインは更新が必要な場合があります。

設定の変更は、本番環境に反映する前にステージング環境で検証してください。ポリシーに誤りがあると、スクリプト、サインイン、購入手続き、埋め込みサービス、アクセス解析、外部サービスとの連携が動作しなくなる場合があります。

参考資料

MDN: Set-Cookie
OWASP Session Management Cheat Sheet
developer.mozilla.org

F-04 クリックジャッキング対策がない 中

検出項目ID	DEMO-2026-0001-F-04 · ??? xfo
分類	確認済みの指摘: Web サイトの応答や DNS で直接観測されたものです。スコアを下げます。
対応状況	引き続き観測
修正の優先度	P2 · 30 日以内に修正
スコアへの影響	−8 点
確度	高
対象URL	https://www.client-shop.example/
検出方法	HTTP レスポンスヘッダーの受動的な確認。
確認日時	2026-10-01T09:14:05Z
証拠	X-Frame-Options も CSP の frame-ancestors ディレクティブも返されませんでした。
影響の理由	ほかのWebサイトが、これらのページを見えないフレームに読み込み、訪問者をだましてクリックさせることができます。
顧客向けの説明	別の Web サイトがこのサイトを自分のページの中に隠して表示し、「確定」や「削除」など、見えないボタンを押させることができます。サーバー設定1つで防げます。
ビジネスへの影響	訪問者が、アカウント設定の変更など意図しない操作をさせられるおそれがあります。
技術的な影響	ページを別オリジンのフレームに埋め込むことができます。
改善方法	X-Frame-Options: SAMEORIGIN、または CSP の frame-ancestors 'self' を送信します。
確認の基準	再診断で、X-Frame-Options（DENY または SAMEORIGIN）または CSP の frame-ancestors ディレクティブが確認されること。
確認状況	最新の診断でも残存

人による確認	**修正を計画する前に必要です。**
参照規格	CWE-1021

プラットフォーム別の設定例

Nginx

```
# server { } block (or http { })
add_header X-Frame-Options "SAMEORIGIN" always;
```

すでに Content-Security-Policy を送信している場合は、frame-ancestors 'self' を追加すると、最新のブラウザで同じ保護が得られます。 反映：nginx -t && systemctl reload nginx。location ブロック内の add_header は、server レベルで設定したヘッダーを置き換えます。

Apache HTTP Server

```
# Requires mod_headers (a2enmod headers)
Header always set X-Frame-Options "SAMEORIGIN"
```

すでに Content-Security-Policy を送信している場合は、frame-ancestors 'self' を追加すると、最新のブラウザで同じ保護が得られます。 VirtualHost または .htaccess に記述し、apachectl configtest を実行してから再読み込みします。

Microsoft IIS

```
<?xml version="1.0" encoding="utf-8"?>
<!-- web.config: merge into the existing <configuration> element -->
<configuration>
  <system.webServer>
    <httpProtocol>
      <customHeaders>
        <add name="X-Frame-Options" value="SAMEORIGIN" />
      </customHeaders>
    </httpProtocol>
  </system.webServer>
</configuration>
```

すでに Content-Security-Policy を送信している場合は、frame-ancestors 'self' を追加すると、最新のブラウザで同じ保護が得られます。 この web.config を持つサイトまたはアプリケーションに適用されます。

Cloudflare

```
Rules > Transform Rules > Modify Response Header > Create rule
1. Apply to: All incoming requests (or this hostname)
2. Action: Set static
3. Header name: X-Frame-Options
4. Value: SAMEORIGIN
5. Deploy
```

すでに Content-Security-Policy を送信している場合は、frame-ancestors 'self' を追加すると、最新のブラウザで同じ保護が得られます。 Cloudflare は、プロキシ経由（オレンジクラウド）の通信に対してのみエッジでヘッダーを設定します。

WordPress

```
<?php
// In a small site-specific plugin (or the child theme's functions.php)
add_action( 'send_headers', function () {
    header(
        "X-Frame-Options: "
        . "SAMEORIGIN"
    );
} );
```

すでに Content-Security-Policy を送信している場合は、frame-ancestors 'self' を追加すると、最新のブラウザで同じ保護が得られます。 ヘッダーは Web サーバーまたは CDN で設定するのが望ましく、これは WordPress しか操作できない場合の方法です。

AWS CloudFront

```
CloudFront > Policies > Response headers > Create response headers policy
1. Security headers > X-Frame-Options: SAMEORIGIN
2. Origin override: enabled
3. Attach the policy to the distribution's cache behavior(s)
```

すでに Content-Security-Policy を送信している場合は、frame-ancestors 'self' を追加すると、最新のブラウザで同じ保護が得られます。 変更は数分以内にすべてのエッジロケーションに反映されます。

設定の変更は、本番環境に反映する前にステージング環境で検証してください。ポリシーに誤りがあると、スクリプト、サインイン、購入手続き、埋め込みサービス、アクセス解析、外部サービスとの連携が動作しなくなる場合があります。

参考資料

MDN: X-Frame-Options
OWASP Clickjacking Defense Cheat Sheet
developer.mozilla.org

F-06 サーバーソフトウェアのバージョンが表示されている 低

検出項目ID	DEMO-2026-0001-F-06 · ??? server_banner
分類	**確認済みの指摘**: Web **サイトの応答や** DNS **で直接観測されたものです。スコアを下げます。**
対応状況	**引き続き観測**
修正の優先度	P3 · 90 **日以内に確認**
スコアへの影響	**ー3 点**
確度	**中**
対象URL	https://www.client-shop.example/
検出方法	Server **と** X-Powered-By **レスポンスヘッダーの受動的な確認。**
確認日時	2026-10-01T09:14:05Z
証拠	Server: Apache/2.4.41 (Ubuntu) X-Powered-By: PHP/7.4.3
影響の理由	**正確なバージョン番号があると、既知の脆弱性とサイトを照合しやすくなります。**
ビジネスへの影響	**新しい脆弱性が公表されたときに、攻撃者がこのサイトを優先的に狙う手がかりになります。**
技術的な影響	**ソフトウェア名とバージョンがすべての訪問者に公開されています。**
制約事項	**バージョン表示は正確でない場合があります。ディストリビューションでは、バージョン表記を変えずにセキュリティ修正を取り込むことがよくあります。**
改善方法	Web **サーバーと** PHP **の設定でバージョン情報を隠し、両方を最新に保ちます。**
確認の基準	**再診断で、**Server **と** X-Powered-By **ヘッダーにバージョン番号が確認されないこと。**
確認状況	**最新の診断でも残存**
人による確認	**修正を計画する前に必要です。**
参照規格	CWE-200

プラットフォーム別の設定例

Nginx

```
# http { } block
server_tokens off;
```

バージョン番号は隠れますが、Server **ヘッダーには** "nginx" **と表示されたままです。**

Apache HTTP Server

```
# Main server configuration (not .htaccess)
ServerTokens Prod
ServerSignature Off
```

Server **ヘッダーが** "Apache" **だけになります。**X-Powered-By **は** php.ini **で** expose_php = Off **を設定します。**

Microsoft IIS

```
<?xml version="1.0" encoding="utf-8"?>
<!-- web.config, IIS 10 (version 1709+): merge into the existing <configuration> element -->
<configuration>
  <system.webServer>
    <security>
      <requestFiltering removeServerHeader="true" />
    </security>
    <httpProtocol>
      <customHeaders>
        <remove name="X-Powered-By" />
      </customHeaders>
    </httpProtocol>
  </system.webServer>
</configuration>
```

removeServerHeader は IIS 10 バージョン 1709 以降で利用できます。

WordPress

```
; php.ini (or .user.ini): removes the X-Powered-By: PHP/x.y header
expose_php = Off
```

Server ヘッダー自体は、WordPress ではなく Web サーバー（Nginx または Apache）が制御します。

設定の変更は、本番環境に反映する前にステージング環境で検証してください。ポリシーに誤りがあると、スクリプト、サインイン、購入手続き、埋め込みサービス、アクセス解析、外部サービスとの連携が動作しなくなる場合があります。

参考資料

OWASP: Fingerprint Web Server
owasp.org

F-02 Strict-Transport-Security ヘッダーがない

検出項目ID	DEMO-2026-0001-F-02 · ??? hsts
分類	**確認済みの指摘**: Web サイトの応答や DNS で直接観測されたものです。スコアを下げます。
対応状況	**解決**
修正の優先度	P1 · 7 日以内に修正
スコアへの影響	−15 点
確度	高
対象URL	https://www.client-shop.example/
検出方法	HTTPS での HTTP レスポンスヘッダーの受動的な確認。
確認日時	2026-10-01T09:14:05Z
証拠	HTTPS は利用できますが、応答に Strict-Transport-Security ヘッダーがありません。
影響の理由	HSTS は、このサイトでは常に HTTPS を使うようブラウザに指示し、最初のリクエストが HTTP に格下げされる余地をなくします。
顧客向けの説明	ブラウザーに対して、常に安全な版のサイトを使うよう指示していません。公衆 Wi-Fi などでは、訪問者が気づかないうちに安全でないページに誘導される可能性があります。修正はサーバー設定1行です。
ビジネスへの影響	信頼できないネットワークの訪問者が、安全でないサイトに誘導されるおそれがあります。
技術的な影響	サイトへの最初の平文 HTTP リクエストが、ブラウザによって HTTPS に切り替えられません。

改善方法	すべてのサブドメインで HTTPS が安定したら、max-age を1年以上にした Strict-Transport-Security を送信します。
確認の基準	再診断で、HTTPS レスポンスに max-age が 31536000 以上の Strict-Transport-Security が確認されること。
確認状況	修正確認済み
人による確認	修正を計画する前に必要です。
参照規格	CWE-319 · NIST SP 800-53 SC-8
参考資料	MDN: Strict-Transport-Security OWASP HTTP Security Response Headers developer.mozilla.org

F-05 X-Content-Type-Options ヘッダーがない

低

検出項目ID	DEMO-2026-0001-F-05 · ??? xcto
分類	確認済みの指摘: Web サイトの応答や DNS で直接観測されたものです。スコアを下げます。
対応状況	解決
修正の優先度	P3 · 90 日以内に確認
スコアへの影響	−3 点
確度	高
対象URL	https://www.client-shop.example/
検出方法	HTTP レスポンスヘッダーの受動的な確認。
確認日時	2026-10-01T09:14:05Z
証拠	応答に X-Content-Type-Options ヘッダーがありません。
影響の理由	ブラウザがファイルの種類を推測し、アップロードされたテキストをスクリプトとして扱うおそれがあります。
ビジネスへの影響	単独での影響は小さいものの、コンテンツタイプの取り違えに対する防御が一層なくなります。
技術的な影響	MIME タイプの推測（スニッフィング）が無効化されていません。
改善方法	X-Content-Type-Options: nosniff を送信します。
確認の基準	再診断で X-Content-Type-Options: nosniff が確認されること。
確認状況	修正確認済み
人による確認	修正を計画する前に必要です。
参照規格	CWE-16
参考資料	MDN: X-Content-Type-Options developer.mozilla.org

合格したチェック

- 信頼されたチェーンを持つ有効な TLS 証明書
- 証明書の有効期限に余裕あり（残り74日）
- HTTP は HTTPS にリダイレクト
- 診断したページに混在コンテンツなし
- Referrer-Policy あり
- Permissions-Policy あり
- 重複または矛盾するセキュリティヘッダーなし
- よく公開されてしまうファイル（.git、.env、バックアップ）は見つかりませんでした
- security.txt を公開
- SPF レコードあり
- DMARC レコードあり

修正ロードマップ

30日以内に修正 (3)

F-01 Content-Security-Policy ヘッダーが設定されていない
F-03 セッション Cookie に Secure と HttpOnly 属性がない
F-04 クリックジャッキング対策がない

90日以内に確認 (1)

F-06 サーバーソフトウェアのバージョンが表示されている

再確認

変更のたびに Verify Fix を実行してください。NeoShield が同じ Web サイトを再スキャンし、各指摘を「解決」「一部解決」「未解決」「確認できず」として記録します。更新されたレポートを作業の記録として保管してください。

これらの期間は一般的な目安です。Webサイトのリスク、顧客との契約、コンプライアンス上の義務に合わせて調整してください。

付録 A：参照規格の対応表

各指摘は、下記の公開された弱点・プロトコル・管理策の参照に対応しています。参照は各チェックの根拠を示すもので、Web サイトや NeoShield がそれらの認証を受けていることを意味しません。

番号	検出項目	参照
F-01	Content-Security-Policy ヘッダーが設定されていない	OWASP A05:2021, CWE-693, NIST SP 800-53 SC-18
F-03	セッション Cookie に Secure と HttpOnly 属性がない	CWE-614, CWE-1004, CWE-1275
F-04	クリックジャッキング対策がない	CWE-1021
F-06	サーバーソフトウェアのバージョンが表示されている	CWE-200
F-02	Strict-Transport-Security ヘッダーがない	CWE-319, NIST SP 800-53 SC-8
F-05	X-Content-Type-Options ヘッダーがない	CWE-16

付録 B：用語の定義

P0	直ちに対応. サイトやデータを直接危険にさらします。
P1	7日以内に修正. HTTPS の強制など、すべての訪問者に関わる重要な保護が欠けています。
P2	30日以内に修正. CSP や Cookie の属性など、他の欠陥があるときに重要になる防御が弱くなっています。
P3	90日以内に確認. 直接の影響が限られる強化項目または情報開示です。
P4	任意の強化項目. 必須ではありません。都合のよいときに検討してください。
確認済みの指摘	Web サイトの応答や DNS で直接観測されたものです。スコアを下げます。
状況に応じて確認する観測	Web サイト、DNS、メールの使い方に左右されます。確認してください。スコアは下げません。
情報としての観測	認識のために記録しています。対応は不要です。
判定できなかったチェック	チェックを完了できませんでした。合格としては数えません。
解決	同じ Web サイトのその後のスキャンで、指摘が観測されなくなったものです。
引き続き観測	その後のスキャンでも指摘が観測されました。
カバー率	該当するチェックのうち結果が得られた割合です。カバー率が低いと評価に上限が付きます。
確度	根拠が結果をどれだけ直接裏付けるか：HTTP や DNS の直接の根拠なら高、解釈が必要なら中、応答が不完全またはモジュールが実行できなかった場合は低。カバー率だけからは決めていません。

付録 C：範囲、許可、方法

- 対象：https://www.client-shop.example/（公開Webサイト1件、ルートドメインと www）
- 方法：公開ページへの非侵入的なリクエストと DNS の照会
- 許可：診断前にアカウント所有者が確認済み
- 確認項目：HTTP セキュリティヘッダー、TLS と証明書、Cookie、HTTP から HTTPS へのリダイレクト、SPF、DMARC、security.txt、よく公開されてしまうファイル

付録 D：制約

- これは受動的な設定診断であり、ペネトレーションテストではありません。
- ログイン後の領域、アプリケーションのロジック、インジェクションの不具合、業務ロジックの問題は診断していません。

- 結果は診断時に確認した応答に基づきます。CDN やロードバランサーにより、ほかの訪問者には異なるヘッダーが返される場合があります。
- 設定例は、本番に反映する前にステージング環境で検証してください。

付録 E：レポートの完全性

レポートID	DEMO-2026-0001
スキャン日時	2026-10-01
作成日時	2026-11-04 19:27 UTC
診断エンジンのバージョン	demo
スコアリングモデル	2026.07-b
SHA-256	d90968d5395ff2de9fdd7cfed52b3b7c2ab9fad8061f5b0866d9efe129514c65

SHA-256 の値は、この PDF の元になった保存済みスキャンデータの指紋です。NeoShield は、コピーが保存済みレポートと一致するかの確認に使えます。

NeoShield Quantum X で作成。指摘は、診断時に公開されていた応答から自動的に生成されたもので、NeoShield のスタッフは確認していません。変更を行う前に、このレポートを送る方が確認してください。このレポートはペネトレーションテスト、セキュリティ認証、保証のいずれでもありません。