TLP:CLEAR【全分野】

別紙

2026 年 10 月 21 日

内閣官房　国家サイバー統括室
対処調整・官民連携等ユニット

PHP の深刻な脆弱性(CVE-2026-109618、2026-109619)について(注意喚起)

1. 対象ソフトウェア

PHP における深刻度 Critical(CVSS v4.0: 9.2)の重大な脆弱性 2 件(CVE-2026-109618、CVE-2026-109619) が 2026 年 10 月 21 日 23 時（日本時間）に公開される予定です（参考 URL①、予定前倒しの可能性有り)。Hacker News フォーラムでのリーク情報によれば､PHP 5 系以降のすべての PHP バージョンに影響する可能性があります(参考 URL②)。

2. 脆弱性悪用による影響等

現時点では、技術的な詳細は公表されておりませんが、Hacker News フォーラムでのリーク情報や、watchTowr の情報によれば、本脆弱性はリモートコード実行の脆弱性であり、任意のコマンドを実行される可能性があります（参考 URL②、③)。

3. 悪用状況

現時点での悪用事例は確認されておりませんが､昨今の不正アクセス等による情報漏えい等の事案の発生状況を踏まえると、当該脆弱性への適切な対応がなされない場合、短時間でウェブシステム等への不正アクセス等が生じることが懸念されます。

4. 対応

現時点では修正版は提供されていませんが､開発元は本脆弱性に対応するセキュリティアップデートを 2026 年 10 月 21 日 23 時（日本時間）に公開予定であると公表しています。公表後に直ぐに攻撃が開始される可能性もあるため、PHP を利用している組織は利用状況の確認や影響調査を実施し､修正版の公開後に可及的速やかに適用できるよう、事前に準備を進めることを強く推奨します。

また、インシデント発生時には所管省庁等へのご報告及びご連絡をお願いします。

参考 URL

① Upcoming Critical PHP Security Release (2026/10/20)
https://www.php.net/archive/2026.php#2026-10-20-1

② Security Advisory 2026-018: Critical Vulnerabilities in PHP (2026/10/20)
https://cert.europa.eu/publications/security-advisories/2026-018/

③ PHP warns admins of two critical flaws, emergency patches coming tomorrow (2026/10/20)
https://www.bleepingcomputer.com/news/security/php-warns-admins-of-two-critical-flaws-emergency-patches-coming-tomorrow/

TLP:AMBER+STRICT【全分野】

別紙

2026 年 10 月 21 日

内閣官房　国家サイバー統括室
対処調整・官民連携等ユニット

（第 2 報）PHP の深刻な脆弱性(CVE-2026-109618、2026-109619)について（注意喚起）

1. 対象ソフトウェア

PHP における深刻度 Critical(CVSS v4.0: 9.2)の重大な脆弱性 2 件(CVE-2026-109618、CVE-2026-109619）が 2026 年 10 月 21 日 23 時（日本時間）に公開される予定です（参考 URL①、予定前倒しの可能性有り)。Hacker News フォーラムでのリーク情報によれば､PHP 5 系以降のすべての PHP バージョンに影響する可能性があります(参考 URL②)。

2. 脆弱性悪用による影響等

現時点では、技術的な詳細は公表されておりませんが、Hacker News フォーラムでのリーク情報や、watchTowr の情報によれば、本脆弱性はリモートコード実行の脆弱性であり、任意のコマンドを実行される可能性があります（参考 URL②、③)。

3. 悪用状況

現時点での悪用事例は確認されておりませんが､昨今の不正アクセス等による情報漏えい等の事案の発生状況を踏まえると、当該脆弱性への適切な対応がなされない場合、短時間でウェブシステム等への不正アクセス等が生じることが懸念されます。

4. 対応

現時点では修正版は提供されていませんが､開発元は本脆弱性に対応するセキュリティアップデートを 2026 年 10 月 21 日 23 時（日本時間）に公開予定であると公表しています。公表後に直ぐに攻撃が開始される可能性もあるため、PHP を利用している組織は利用状況の確認や影響調査を実施し､修正版の公開後に可及的速やかに適用できるよう、事前に準備を進めることを強く推奨します。

なお、迅速に修正パッチを適用することが困難な場合は、例えばウェブシステム等を一時的に停止させることもリスクを低減させるための有効策の一つであると考えられます。

また、インシデント発生時には所管省庁等へのご報告及びご連絡をお願いします。

5. 緩和策（追加）

本内容は､あくまでベンダー公式の修正パッチを適用するまでのリスク緩和を目的とした措置であり、脆弱性を悪用した攻撃自体の回避を可能にするものではありません。

開発元から提供される修正済みビルドへの可及的速やかなアップグレードが唯一の恒久的な解決策となります。

また、ここに示す各種設定変更に伴い、正常なシステムの動作の停止や、サーバーリソースの急変動といった潜在的な副作用を伴う可能性があります。各組織の検証環境・試験環境において必ず事前検証し、業務要件との整合性を確認したうえで、適用を判断してください。

(1) Web サーバー/WAF レベルにおける multipart/form-data リクエストの遮断

- 実施内容：Web サーバーまたは WAF において、Content-Type: multipart/form-data を含む HTTP リクエストを一括して拒否する。
- 設定対象：Web サーバーの場合は、nginx（nginx.conf）や Apache（httpd.conf および関連する設定ファイル）等のサイト設定ファイルが該当します（参考 URL④⑤）。
- 留意点：本手法は、読み取り専用サイトのような multipart/form-data 形式を必要としない場合には有効ですが、ユーザーからのファイルアップロード機能や、Web フォームを実装しているシステムでは、当該機能が利用不能となります。また、フィルタリングに際し、影響を受ける PHP アプリケーションへのアクセス経路をすべてカバーする必要があるため、利用可能な場合は WAF ルールの適用が推奨されます。

(2) PHP ランタイムレベルでの設定変更

- 実施内容： PHP 自身の内部動作パラメータを次の表に示すものに調整し、脆弱性の発現条件を抑止する。
- 設定対象：php.ini、ロードされる conf.d/*.ini（参考 URL⑥）
- 留意点：Web サーバーや、PHP サービスのリロードや再起動を通して、設定変更が反映されていることや、プール設定やディレクトリ単位のオーバーライド設定によって意図せず上書きされていないかを確認してください。また、パッチ適用後は、ワークロードに合わせた値にチューニングしてください。

設定値	説明
memory_limit = -1	メモリ使用量を無制限にすることで、メモリ制限によって引き起こされる特定の攻撃を一時的に抑制することができます。ただし、OS によるメモリ不足によるプロセス強制終了やサービスの不安定化を引き起こす可能性があります。パッチ適用後には設定値を元に戻す必要があります。
post_max_size = 8M	アプリケーションが受け付ける POST リクエスト全体の最大値を制限します。8M は PHP ビルトインのデフォルト値であり、アプリケーションに応じて数値を調整してください。既

	にこれより小さい値が設定されている場合は、より厳しい現行値を維持してください。
max_input_nesting_level = 4	PHP がリクエストから解析する入力配列のネスト（入れ子）深度を浅く制限し、過度に入り組んだデータ構造によるパーサーの異常動作を抑制します。
session.upload_progress.enabled = 0	ファイルアップロード時の進捗管理機能を無効化します。脆弱性の悪用経路の可能性がある内部処理を停止させます。
fatal_error_backtraces = 0	致命的エラー発生時における詳細なバックトレース情報の出力を停止し、攻撃者に対する内部メモリアドレスやスタック構造の漏洩を防止します。
expose_php = 0	HTTP レスポンスヘッダー（X-Powered-By）から PHP のバージョン情報の出力を抑止し、攻撃者による自動スキャン時のフィンガープリンティングを困難にします。本設定はパッチ適用後においても適用することが推奨されます。

(3) PMP-FPM レベルでの保護構成

- 実施内容：PHP-FPM (FastCGI Process Manager)において、プロセスの健全性を維持し、状態異常の長期滞留を防ぐ。
- 設定対象：環境に応じたアクティブなプール設定ファイル（参考 URL⑦）
  - 例：/etc/php-fpm.conf、/etc/php-fpm.d/www.conf 等

設定値	説明
pm.max_requests = 50	各 FPM ワーカープロセスが処理する最大リクエスト数を 50 件程度に短縮します。ワーカープロセスを頻繁に自動破棄・再生成させることで、仮にメモリ破壊や不正な状態がプロセス内に残留した場合でも、次以降のリクエストへ被害が連鎖・蓄積する機会を最小限に抑えます。
apparmor_hat の使用	OS 側で AppArmor を利用している場合、プール設定内で適切なサブプロファイル名（Hat 名）を指定し、Web サーバー権限と PHP ワーカープロセスの権限分離を確立します。

(4) ホストメモリの制限（systemd）

- 実施内容：設定対象のファイルにおいて、[Service]セクションで MemoryMax に検証済みの有限値を設定することで、PHP プロセスが異常なメモリ消費に陥った際、ホスト OS 全体のクラッシュを防止する。
- 設定対象：systemd で管理されている PHP-FPM サービスに対するドロップイン設定ファイル「/etc/systemd/system/<service_name>.service.d/limits.conf」（参考 URL⑧）
- 留意点：

- cgroup v2 において、MemoryMax はユニット全体のプロセスが使用する上限値となります。全ワーカープロセスの合計消費量を考慮したサイジングを行ってください。
- Docker や Kubernetes などのコンテナ環境を使用している場合は、プラットフォームがサポートするメモリ制限設定を構成し、その適用範囲を確認してください。
- メモリ不足が発生した場合、PHP プロセスが強制終了され、サービスが中断される可能性があります。どのプロセスが終了対象となるかや、サービス復旧プロセスを事前に確認してください。

(5) AppArmor プロファイル / SELinux ポリシー

- 実施内容：AppArmor または SELinux を利用している場合において、PHP への関連付けを通して、攻撃者の行動を最小権限のサンドボックス内に封じ込め、OS コマンド実行や横展開を制限する。
- AppArmor による制御の場合：PHP に関連付けられたプロファイルを /etc/apparmor.d/<profile> （参考 URL⑨）で編集・ロードしてください。また、ログ記録のみの complain モードではなく、enforce モードを有効にしてください。
- SELinux による制御の場合：ベンダー公式のサポートポリシー（参考 URL⑩）に則り、PHP プロセスが制限されたドメイン内で動作していることを確認し、Enforcing モードで運用してください。

参考 URL
① Upcoming Critical PHP Security Release (2026/10/20)
https://www.php.net/archive/2026.php#2026-10-20-1
② Security Advisory 2026-018: Critical Vulnerabilities in PHP (2026/10/20)
https://cert.europa.eu/publications/security-advisories/2026-018/
③ PHP warns admins of two critical flaws, emergency patches coming tomorrow (2026/10/20)
https://www.bleepingcomputer.com/news/security/php-warns-admins-of-two-critical-flaws-emergency-patches-coming-tomorrow/
④ nginx documentation, Beginner's Guide - Configuration File's Structure
https://nginx.org/en/docs/beginners_guide.html#conf_structure
⑤ Apache HTTP Server Project, 設定ファイル
https://httpd.apache.org/docs/2.4/configuring.html
⑥ The PHP Documentation Group, Runtime Configuration - The configuration file
https://www.php.net/manual/en/configuration.file.php
⑦ The PHP Documentation Group, FastCGI Process Manager (FPM) - Configuration
https://www.php.net/manual/en/install.fpm.configuration.php
⑧ freedesktop.org, systemd.unit
https://www.freedesktop.org/software/systemd/man/257/systemd.unit.html
⑨ AppArmor Documentation, Getting Started - Confine an application
https://www.apparmor.net/getting-started/confine-your-application/
⑩ RedHat Documentation, Chapter 8. Writing a custom SELinux policy
https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/8/html/using_selinux/writing-a-custom-selinux-policy_using-selinux

TLP:AMBER+STRICT【全分野】

別紙

2026 年 10 月 21 日

内閣官房　国家サイバー統括室
対処調整・官民連携等ユニット

（第 3 報）PHP の深刻な脆弱性(CVE-2026-109618、2026-109619)について（注意喚起）

1. 対象ソフトウェア

事前の予告通り（参考 URL①、②）、PHP における深刻度 Critical(CVSS v4.0: 9.2)の重大な脆弱性 2 件（CVE-2026-109618、CVE-2026-109619）が公開されました。次のバージョンが影響を受けます（参考 URL③）。

- 8.5.12 未満
- 8.4.27 未満
- 8.3.36 未満
- 8.2.35 未満

なお、PHP 8.1 以前はサポート対象外となるため、サポートされているバージョンへのアップグレードを推奨します。ただし、ディストリビューションや LTS ベンダーが対応している場合もあるため、ベンダーのパッケージを確認してください。

2. 脆弱性悪用による影響等

対象ソフトウェアにはリモートコード実行の脆弱性があり、認証されていないリモートの攻撃者によって、対象システムにおいて任意のコマンドを実行される可能性があります。

3. 悪用状況

現時点での悪用事例は確認されておりませんが、昨今の不正アクセス等による情報漏えい等の事案の発生状況を踏まえると、当該脆弱性への適切な対応がなされない場合、短時間でウェブシステム等への不正アクセス等が生じることが懸念されます。

4. 対応

開発元より、本脆弱性に対応するセキュリティアップデートが公開されています。対象ソフトウェアを利用している組織は脆弱性が修正されたバージョンにアップデートするとともに、侵害の有無を確認することを強く推奨します（参考 URL③）。

なお、迅速に修正パッチを適用することが困難な場合は、例えばウェブシステム等を一時的に停止させることもリスクを低減させるための有効策の一つであると考えられます。

また、インシデント発生時には所管省庁等へのご報告及びご連絡をお願いします。

5. 緩和策

本内容は、あくまでベンダー公式の修正パッチを適用するまでのリスク緩和を目的とした措置であり、脆弱性を悪用した攻撃自体の回避を可能にするものではありません。開発元から提供される修正済みビルドへの可及的速やかなアップデートが唯一の恒久的な解決策となります。

また、ここに示す各種設定変更に伴い、正常なシステムの動作の停止や、サーバーリソースの急変動といった潜在的な副作用を伴う可能性があります。各組織の検証環境・試験環境において必ず事前検証し、業務要件との整合性を確認したうえで、適用を判断してください。

(1) Web サーバー/WAF レベルにおける multipart/form-data リクエストの遮断

- 実施内容：Web サーバーまたは WAF において、Content-Type: multipart/form-data を含む HTTP リクエストを一括して拒否する。
- 設定対象：Web サーバーの場合は、nginx（nginx.conf）や Apache（httpd.conf および関連する設定ファイル）等のサイト設定ファイルが該当します（参考 URL④⑤）。
- 留意点：本手法は、読み取り専用サイトのような multipart/form-data 形式を必要としない場合には有効ですが、ユーザーからのファイルアップロード機能や、Web フォームを実装しているシステムでは、当該機能が利用不能となります。また、フィルタリングに際し、影響を受ける PHP アプリケーションへのアクセス経路をすべてカバーする必要があるため、利用可能な場合は WAF ルールの適用が推奨されます。

(2) PHP ランタイムレベルでの設定変更

- 実施内容： PHP 自身の内部動作パラメータを次の表に示すものに調整し、脆弱性の発現条件を抑止する。
- 設定対象：php.ini、ロードされる conf.d/*.ini（参考 URL⑥）
- 留意点：Web サーバーや、PHP サービスのリロードや再起動を通して、設定変更が反映されていることや、プール設定やディレクトリ単位のオーバーライド設定によって意図せず上書きされていないかを確認してください。また、パッチ適用後は、ワークロードに合わせた値にチューニングしてください。

設定値	説明
memory_limit = -1	メモリ使用量を無制限にすることで、メモリ制限によって引き起こされる特定の攻撃を一時的に抑制することができます。ただし、OS によるメモリ不足によるプロセス強制終了やサービスの不安定化を引き起こす可能性があります。パッチ適用後には設定値を元に戻す必要があります。

post_max_size = 8M	アプリケーションが受け付ける POST リクエスト全体の最大値を制限します。8M は PHP ビルトインのデフォルト値であり、アプリケーションに応じて数値を調整してください。既にこれより小さい値が設定されている場合は、より厳しい現行値を維持してください。
max_input_nesting_level = 4	PHP がリクエストから解析する入力配列のネスト（入れ子）深度を浅く制限し、過度に入り組んだデータ構造によるパーサーの異常動作を抑制します。
session.upload_progress.enabled = 0	ファイルアップロード時の進捗管理機能を無効化します。脆弱性の悪用経路の可能性がある内部処理を停止させます。
fatal_error_backtraces = 0	致命的エラー発生時における詳細なバックトレース情報の出力を停止し、攻撃者に対する内部メモリアドレスやスタック構造の漏洩を防止します。
expose_php = 0	HTTP レスポンスヘッダー（X-Powered-By）から PHP のバージョン情報の出力を抑止し、攻撃者による自動スキャン時のフィンガープリンティングを困難にします。本設定はパッチ適用後においても適用することが推奨されます。

(3) PMP-FPM レベルでの保護構成

- 実施内容：PHP-FPM (FastCGI Process Manager)において、プロセスの健全性を維持し、状態異常の長期滞留を防ぐ。
- 設定対象：環境に応じたアクティブなプール設定ファイル（参考 URL⑦）
  - 例：/etc/php-fpm.conf、/etc/php-fpm.d/www.conf 等

設定値	説明
pm.max_requests = 50	各 FPM ワーカープロセスが処理する最大リクエスト数を 50 件程度に短縮します。ワーカープロセスを頻繁に自動破棄・再生成させることで、仮にメモリ破壊や不正な状態がプロセス内に残留した場合でも、次以降のリクエストへ被害が連鎖・蓄積する機会を最小限に抑えます。
apparmor_hat の使用	OS 側で AppArmor を利用している場合、プール設定内で適切なサブプロファイル名（Hat 名）を指定し、Web サーバー権限と PHP ワーカープロセスの権限分離を確立します。

(4) ホストメモリの制限（systemd）

- 実施内容：設定対象のファイルにおいて、[Service]セクションで MemoryMax に検証済みの有限値を設定することで、PHP プロセスが異常なメモリ消費に陥った際、ホスト OS 全体のクラッシュを防止する。

- 設定対象：systemd で管理されている PHP-FPM サービスに対するドロップイン設定ファイル「/etc/systemd/system/<service_name>.service.d/limits.conf」（参考 URL⑧）
- 留意点：
  - cgroup v2 において、MemoryMax はユニット全体のプロセスが使用する上限値となります。全ワーカープロセスの合計消費量を考慮したサイジングを行ってください。
  - Docker や Kubernetes などのコンテナ環境を使用している場合は、プラットフォームがサポートするメモリ制限設定を構成し、その適用範囲を確認してください。
  - メモリ不足が発生した場合、PHP プロセスが強制終了され、サービスが中断される可能性があります。どのプロセスが終了対象となるかや、サービス復旧プロセスを事前に確認してください。

(5) AppArmor プロファイル / SELinux ポリシー

- 実施内容：AppArmor または SELinux を利用している場合において、PHP への関連付けを通して、攻撃者の行動を最小権限のサンドボックス内に封じ込め、OS コマンド実行や横展開を制限する。
- AppArmor による制御の場合：PHP に関連付けられたプロファイルを /etc/apparmor.d/<profile> （参考 URL⑨）で編集・ロードしてください。また、ログ記録のみの complain モードではなく、enforce モードを有効にしてください。
- SELinux による制御の場合：ベンダー公式のサポートポリシー（参考 URL⑩）に則り、PHP プロセスが制限されたドメイン内で動作していることを確認し、Enforcing モードで運用してください。

参考 URL
① Security Advisory 2026-018: Critical Vulnerabilities in PHP (2026/10/20)
https://cert.europa.eu/publications/security-advisories/2026-018/
② PHP warns admins of two critical flaws, emergency patches coming tomorrow (2026/10/20)
https://www.bleepingcomputer.com/news/security/php-warns-admins-of-two-critical-flaws-emergency-patches-coming-tomorrow/
③ Security advisory: Critical remote code execution vulnerabilities, update now (2026/10/21)
https://www.php.net/index.php#2026-10-21-5
④ nginx documentation, Beginner's Guide - Configuration File's Structure
https://nginx.org/en/docs/beginners_guide.html#conf_structure
⑤ Apache HTTP Server Project, 設定ファイル
https://httpd.apache.org/docs/2.4/configuring.html
⑥ The PHP Documentation Group, Runtime Configuration - The configuration file
https://www.php.net/manual/en/configuration.file.php
⑦ The PHP Documentation Group, FastCGI Process Manager (FPM) - Configuration
https://www.php.net/manual/en/install.fpm.configuration.php
⑧ freedesktop.org, systemd.unit
https://www.freedesktop.org/software/systemd/man/257/systemd.unit.html
⑨ AppArmor Documentation, Getting Started - Confine an application
https://www.apparmor.net/getting-started/confine-your-application/
⑩ RedHat Documentation, Chapter 8. Writing a custom SELinux policy
https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/8/html/using_selinux/writing-a-custom-selinux-policy_using-selinux