wpsec.it

SICUREZZA WORDPRESS

Report forense di incidente

Compromissione WordPress - analisi, bonifica e messa in sicurezza

ASSET ANALIZZATO

demo-ristorante.example

STATO FINALE

RISOLTO, VERIFICATO PULITO

SEVERITÀ

SEVERITÀ ALTA

DATA INTERVENTO

23 maggio 2026

wpsec.it

Squadra sicurezza WPsec.it - info@wpsec.it - https://wpsec.it

CONTROLLO DOCUMENTO

Informazioni del report

Nota sull'esempio pubblico.
Questo documento è un esempio pubblico sanitizzato. Dominio, percorsi, utenze, date e indicatori sono fittizi o generalizzati. Non contiene dati cliente, IP reali, registri grezzi, impronte digitali, password, token, chiavi API o file malevoli.

TIPO DOCUMENTO

Analisi forense di incidente WordPress e registro delle azioni correttive

CLASSIFICAZIONE

Esempio pubblico sanitizzato - dominio e dati fittizi

AMBIENTE

Ambiente WordPress su hosting gestito

STATO FINALE

Risolto - malware rimosso, persistenza eliminata, sito messo in sicurezza

MODELLO

Modello report forense WPsec v1.0.0

GENERATO IL

23/05/2026

Il report reale consegnato al cliente può contenere più dettagli tecnici, allegati, evidenze riservate e sezioni specifiche per conformità. Questo esempio è volutamente sanitizzato per essere pubblicabile.

1. Sintesi direzionale

Il sito `demo-ristorante.example` è stato compromesso e utilizzato per servire pagine spam condizionate agli utenti provenienti da motori di ricerca. Il proprietario vedeva una pagina iniziale apparentemente normale, mentre i sistemi di scansione dei motori di ricerca e una parte del traffico organico venivano esposti a contenuti non autorizzati.

L’analisi ha identificato tre elementi principali:

- un componente obbligatorio malevolo usato come persistenza;
- file PHP malevoli in una cartella scrivibile;
- utenze amministrative non autorizzate.

Il vettore più probabile è lo sfruttamento di un componente non aggiornato con funzionalità di caricamento file. Le evidenze disponibili non permettono di affermare con certezza il primo accesso, perché i registri non coprono l’intera finestra di intrusione. La persistenza osservata, invece, è stata confermata e rimossa.

Stato finale: malware rimosso, utenze non autorizzate eliminate, credenziali operative ruotate, componente vulnerabile sostituito, misure di sicurezza applicate e sito verificato pulito.

2. Perimetro e metodologia

Perimetro. L’intervento ha riguardato l’installazione WordPress principale, la base dati associata, i registri disponibili, le utenze amministrative e i componenti attivi nel perimetro dell’hosting.

Metodo di lavoro.

1. Acquisizione del contesto e dei sintomi.
2. Copia di sicurezza preliminare di file e base dati.
3. Ispezione della struttura file WordPress.
4. Analisi della base dati su utenti, opzioni ed eventi programmati.
5. Lettura dei registri disponibili.
6. Mappatura della persistenza.
7. Bonifica, messa in sicurezza e riduzione della superficie d’attacco.
8. Verifica post-bonifica.

Principio di evidenza. Le conclusioni sono classificate come confermate, probabili o non determinabili. Quando una prova diretta non è disponibile, il report esplicita il limite.

3. Evidenze tecniche

3.1 Modello di compromissione

Livello	Scopo	Meccanismo
Caricamento iniziale	Inserire file malevoli in area scrivibile	Punto di caricamento esposto da componente vulnerabile
Accesso nascosto	Eseguire codice lato server	File PHP offuscato nella cartella caricamenti
Persistenza	Sopravvivere a pulizie parziali	Componente obbligatorio e utenza amministrativa non autorizzata
Monetizzazione	Manipolare traffico organico	Pagine spam e reindirizzamenti condizionati

3.2 Artefatti malevoli identificati e rimossi

#	Percorso	Classificazione	Funzione
1	`wp-content/uploads/2026/05/cache.php`	Accesso nascosto PHP	Esecuzione e reiniezione di file malevoli
2	`wp-content/mu-plugins/site-health-cache.php`	Modulo di persistenza	Ricreazione accesso privilegiato
3	`wp-content/uploads/2026/05/config-spam.dat`	Configurazione codificata	Configurazione spam e reindirizzamenti

3.3 Utenze non autorizzate rimosse

Nome utente	Ruolo	Evidenza	Azione
`support_wp`	Amministratore	Data creazione fuori finestra operativa	Rimosso
`copia_supporto`	Amministratore	Nessuna corrispondenza con utenze autorizzate	Rimosso

3.4 Elementi esaminati e confermati leciti

Elemento	Esito
Nucleo WordPress	Verificato e riallineato
Tema attivo	Nessun modello malevolo residuo

Elemento	Esito
Componenti licenziati	Aggiornati o sostituiti dove necessario
Pianificatore WordPress	Nessuna attività malevola residua

4. Causa probabile e vettore d’ingresso

Vettore iniziale: probabile. Il primo accesso non può essere provato con certezza perché i registri non coprono l’intera finestra dell’intrusione. Le richieste anomale disponibili puntano però a un punto di ingresso del componente galleria, installato in versione vulnerabile.

Causa della reinfezione: confermata. La reinfezione era sostenuta dal componente obbligatorio malevolo e dalle utenze amministrative non autorizzate. Una pulizia limitata ai file visibili nella cartella caricamenti non avrebbe eliminato la persistenza.

Fattori contribuenti.

- Componente non aggiornato.
- Assenza di blocco dell’esecuzione PHP nella cartella caricamenti.
- Utenze amministrative senza 2FA.
- Registri con conservazione insufficiente per coprire tutta la finestra d’attacco.

5. Timeline dell’attacco

Data / ora	Evento	Evidenza
20/05 02:14	Scansione automatizzata su punto di ingresso del componente	Registro accessi disponibile
20/05 02:31	Caricamento di file malevolo nella cartella caricamenti	Data del file system
20/05 02:40	Creazione utenza amministrativa non autorizzata	Tabelle `wp_users` e `wp_usermeta`
21/05 09:18	Prime pagine spam osservate	Search Console e URL indicizzate
23/05 11:20	Bonifica, messa in sicurezza e verifica finale	Inventario file, scansione e controlli manuali

6. Contenimento e perimetro d’impatto

Le evidenze disponibili indicano una compromissione confinata all’installazione WordPress. Non sono emerse prove di compromissione a livello server o accesso ad altre utenze hosting nello stesso ambiente.

La base dati è stata ispezionata per utenti, opzioni, reindirizzamenti ed eventi programmati. Le cartelle scrivibili sono state controllate per file eseguibili malevoli. Le credenziali operative sono state ruotate perché la presenza di utenze amministrative non autorizzate rende prudente assumere possibile esposizione.

7. Azioni eseguite

Conservazione evidenze. Prima delle modifiche è stata conservata una copia operativa di file e base dati.

Rimozione malware. I file malevoli confermati sono stati rimossi o messi in quarantena. Le cartelle scrivibili sono state rese non eseguibili dove consentito dall’hosting.

Pulizia utenze. Le utenze non autorizzate sono state eliminate. Le utenze legittime sono state verificate e ridotte secondo necessità.

Riduzione superficie d’attacco. Il componente vulnerabile è stato sostituito con una versione aggiornata e licenziata. Componenti inutilizzati sono stati rimossi.

Messa in sicurezza. Sono stati ruotati salt, password operative e accessi amministrativi. Sono state applicate regole di blocco per PHP nella cartella caricamenti, disabilitazione dell’editor file e controlli sui permessi.

8. Verifica post-bonifica

Controllo	Atteso	Risultato
Controllo malware manuale	Nessun artefatto attivo	Pulito
URL degli accessi nascosti noti	404 / 403	403 / 404
Inventario amministratori	Solo utenze legittime	Confermato
Disponibilità pagina iniziale	HTTP 200	200
Test reindirizzamento condizionato	Nessun reindirizzamento	Nessun reindirizzamento osservato
Esecuzione PHP nella cartella caricamenti	Bloccata	403 accesso negato

9. Postura di sicurezza attuale

Alla chiusura dell’intervento il sito risulta:

- privo degli artefatti malevoli confermati;
- privo delle utenze amministrative non autorizzate identificate;
- messo in sicurezza contro il vettore di reinfezione osservato;
- operativo sulle pagine pubbliche e sui flussi principali;
- pronto per monitoraggio e azioni successive.

La postura è migliorata, ma non equivale a immunità futura. WordPress resta un ambiente dinamico: componenti, temi, credenziali e hosting devono rimanere sotto controllo.

10. Indicatori di compromissione

File malevoli

- `wp-content/uploads/2026/05/cache.php`
- `wp-content/mu-plugins/site-health-cache.php`
- `wp-content/uploads/2026/05/config-spam.dat`

Utenze non autorizzate

- `support_wp`
- `copia_supporto`

Schema URL malevoli

- `/wp-content/uploads/2026/05/cache.php?verify=*`
- `/?origine=motore-ricerca&reindirizza=*`

11. Rischio residuo e azioni successive richieste

P1 - Rotazione credenziali

- Ruotare password hosting, SFTP, base dati e amministratore WordPress.
- Ruotare eventuali chiavi API collegate a componenti terzi se accessibili dal pannello di gestione.

P1 - Controlli accesso

- Attivare 2FA su tutte le utenze amministrative.
- Ridurre gli amministratori permanenti al minimo necessario.

P2 - Monitoraggio

- Verificare Search Console per avvisi di sicurezza e pagine spam residue.
- Attivare copie di sicurezza esterne versionate e test periodico di ripristino.
- Attivare controllo integrità file e avvisi su modifiche sospette.

12. Appendice - Dati dell’intervento

- **Copie conservate:** copia pre-intervento conservata secondo procedura WPsec.
- **Strumenti usati:** analisi manuale file system/base dati, WP-CLI, revisione registri accessi, strumento di scansione esterno, controlli HTTP.
- **Stato WordPress:** nucleo riallineato, componente vulnerabile sostituito, utenze amministrative verificate.
- **Nota dati:** l’esempio pubblico usa dati fittizi e non rappresenta un cliente reale.

Contatti

wpsec.it - Sicurezza WordPress
Analista: Squadra sicurezza WPsec.it
Sito: https://wpsec.it
Email: info@wpsec.it

Preparato da wpsec.it. Il report è basato su evidenze: le conclusioni non supportate da prova diretta vengono indicate come probabilità o limite dell’analisi.