Advisory: CVE-2020-29047 - Unauthenticated RCE via Arbitrary Object Deserialisation in WordPress Hotel Booking Plugin

Research / Security Alerts / Posted March 03, 2021

It is possible to gain **Unauthenticated** Remote Code Execution (RCE) on any **WordPress** instance that is using this plugin due to the unsafe use of *maybe_unserialize* for the parsing of unsanitised user input, via the cookie *thimpress_hotel_booking_1* used within *includes/class-wphb-sessions.php*

CVE: CVE-2020-29047
Severity: **HIGH**
Vulnerability Type: **CWE-502**: Deserialization of Untrusted Data
Requires Authentication: **No**

Timeline

Discovered: **2020-11-17** – Nick Blundell, AppCheck Ltd
Contacted Vendor: 2020-11-17
Reported to Vendor: 2020-11-18
Fixed: **2020-12-08**

Affected Software

Name: **WP Hotel Booking**
URL: https://wordpress.org/plugins/wp-hotel-booking/
Vendor: Thimpress (https://thimpress.com/)
Vulnerable versions: **< 1.10.3**
Google Dork: inurl:"/wp-content/plugins/wp-hotel-booking/"

Affected Components

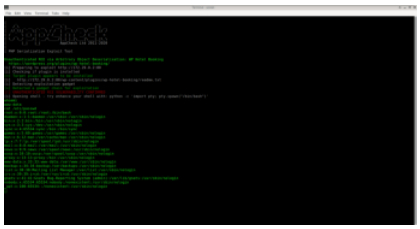
The following code **deserialises** the cookie value sent from the user, such that **arbitrary code** may be injected:

```
// File: includes/class-wphb-sessions.php

public function load() {
    if ( isset( $_SESSION[ $this->prefix ] ) ) {
        return $_SESSION[ $this->prefix ];
    } else if ( $this->remember && isset( $_COOKIE[ $this->prefix ] ) ) {
        return $_SESSION[ $this->prefix ] = maybe_unserialize( $_COOKIE[ $this->prefix ] );
    } //                                   ^^^^^^^^^^^^^^^^^
    return array();
}
```

Exploitation Demo

Exploitation of PHP serialisation vulnerabilities involves leveraging a collection of *gadget* classes that are already present within the vulnerable application, such as third party libraries, in such a way that arbitrary code execution (or some other malicious action) is executed when that *chain* of gadget classes is *deserialised*. See references below for more details on this class of vulnerability and its exploitation.



CVE-2020-29047 Resolution

Please update to the latest version of the plugin.

References

- https://owasp.org/www-community/vulnerabilities/PHP_Object_Injection
- https://owasp.org/www-project-top-ten/2017/A8_2017-Insecure_Deserialization
- http://plugins.svn.wordpress.org/wp-hotel-booking/trunk/includes/class-wphb-sessions.php
- https://notsosecure.com/remote-code-execution-via-php-unserialize/
- https://nitesculucian.github.io/2018/10/05/php-object-injection-cheat-sheet/
- https://www.php.net/manual/en/language.oop5.magic.php
- https://vickieli.medium.com/diving-into-unserialize-pop-chains-35bc1141b69a

About AppCheck

AppCheck is a software security vendor based in the UK, that offers a leading security scanning platform that automates the discovery of security flaws within organisations websites, applications, network, and cloud infrastructure.

As always, if you require any more information on this topic or want to see what unexpected vulnerabilities AppCheck can pick up in your website and applications then please get in contact with us: info@appcheck-ng.com

Get started with Appcheck

This website uses cookies to improve user experience. By using our website you consent to the storage of cookies from this website.

I agree

POPULAR

DNS Security

The New OpenSSL Critical Vulnerability - Early Information and Detections

File Upload Vulnerabilities

RECENT ARTICLES

DNS Security

The New OpenSSL Critical Vulnerability - Early Information and Detections

File Upload Vulnerabilities

RELATED

File Upload Vulnerabilities

What is Open-Source Intelligence (OSINT)?

World's Strangest Hacks

AppCheck Ltd. is a company registered in England and Wales with company number 06888174