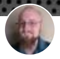

Ram Gall

March 26, 2020

Vulnerabilities Patched in IMPress for IDX Broker

On February 28, 2020, the Wordfence Threat Intelligence team became aware of a newly patched stored Cross-Site Scripting (XSS) vulnerability in IMPress for IDX Broker, a WordPress plugin with over 10,000 installations. Although all Wordfence users, including those still using the free version of Wordfence, were already protected from this vulnerability by the Web Application Firewall's built-in XSS protection, we investigated the plugin further and discovered an additional stored XSS vulnerability. We also found a flaw that would allow an authenticated attacker with minimal, subscriber-level permissions to permanently delete any page or post on the site, in addition to creating pages with arbitrary titles.

We initially reached out to the plugin's vendor the same day, on February 28, 2020, but received no response over an extended period of time. On March 19, 2020, after notifying the WordPress plugin team, we received a response from the plugin's developer, at which time we sent the full disclosure details. A fully patched version was released on March 23, 2020, and we recommend updating to the latest version, 2.6.2, immediately.

Wordfence Premium users received a new firewall rule on March 2nd to protect against exploits targeting these vulnerabilities. Free Wordfence users will receive this rule on April 1, 2020.

Description: Authenticated Stored Cross-Site Scripting(XSS)
Affected Plugin: IMPress for IDX Broker
Plugin Slug: idx-broker-platinum
Affected Versions: <= 2.6.1
CVE ID: CVE-2020-11512
CVSS Score: 7.4 (high)
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:C/C:L/I:L/A:L
Fully Patched Version: 2.6.2

The IMPress for IDX Broker plugin contains a captcha feature to prevent spam submissions. Since it uses Google's ReCAPTCHA service, it requires an API key. Unfortunately, the AJAX action the plugin registered to update this API key did not use capability checks or nonce checks.

This made it possible for a logged-in attacker with minimal permissions, such as a subscriber, to send a request to wp-admin/admin-ajax.php with the `action` parameter set to `idx_update_recaptcha_key` and the `idx_recaptcha_site_key` parameter set to a malicious JavaScript, which could then be executed in an administrator's browser the next time they visited the plugin's settings panel.

As with most attacks taking advantage of stored XSS in admin areas, this could be used to make use of the administrator's session in order to create a new, malicious administrative user.

The AJAX action:

```
add_action( 'wp_ajax_idx_update_recaptcha_key', array( $this, 'idx_update_recaptcha_key' ) );
```

The vulnerable function:

```
public function idx_update_recaptcha_key() {
    if ( $_POST['idx_recaptcha_site_key'] ) {
        update_option( 'idx_recaptcha_site_key', $_POST['idx_recaptcha_site_key'], false );
        echo 1;
    } else {
        delete_option( 'idx_recaptcha_site_key' );
        echo 'error';
    }
    die();
}
```

Description: Authenticated Post Creation, Modification, and Deletion
Affected Plugin: IMPress for IDX Broker
Plugin Slug: idx-broker-platinum
Affected Versions: <= 2.6.1
CVE ID: CVE-2020-9514
CVSS score: 8.1(high)
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:H
Fully Patched Version: 2.6.2

One of the features included with the IDX Broker plugin is the ability to create and delete "dynamic pages," intended to ensure that any IDX pages match the site's style and branding.

The plugin registers 2 AJAX actions that are used to do this:

```
add_action( 'wp_ajax_create_dynamic_page', array( $this, 'idx_ajax_create_dynamic_page' ) );
```

```
add_action( 'wp_ajax_delete_dynamic_page', array( $this, 'idx_ajax_delete_dynamic_page' ) );
```

Once again, neither of the functions called by these AJAX actions used capability checks or nonce checks. As such it was possible for an authenticated attacker with minimal, subscriber-level, permissions to send a request to wp-admin/admin-ajax.php with the `action` parameter set to `create_dynamic_page` and the `post_title` parameter set to any arbitrary value. In return, a new dynamic page with that title would be created.

If a `wrapper_page_id` parameter was included and set to the ID of an existing post or page, that post or page would be replaced with a blank wrapper page:

```php
public function idx_ajax_create_dynamic_page() {



    $new_post   = array(
        'post_title'   => $post_title,
        'post_name'    => $post_title,
        'post_content' => $post_content,
        'post_type'    => 'idx-wrapper',
        'post_status'  => 'publish',
    );
    if ( $_POST['wrapper_page_id'] ) {
        $new_post['ID'] = $_POST['wrapper_page_id'];
    }
    $wrapper_page_id = wp_insert_post( $new_post );
    update_option( 'idx_broker_dynamic_wrapper_page_name', $post_title, false );
    update_option( 'idx_broker_dynamic_wrapper_page_id', $wrapper_page_id, false );
    $wrapper_page_url = get_permalink( $wrapper_page_id );
    $this->idx_api->set_wrapper( 'global', $wrapper_page_url );
    update_post_meta( $wrapper_page_id, 'idx-wrapper-page', 'global' );

    die(
        json_encode(
            array(
                'wrapper_page_id'   => $wrapper_page_id,
                'wrapper_page_name' => $post_title,
            )
        )
    );
}
```

Alternatively, if the attacker set the `action` parameter to `delete_dynamic_page` and sent a `wrapper_page_id` parameter with the ID of an existing post or page, then that post or page would be permanently deleted:

```php
public function idx_ajax_delete_dynamic_page() {
    if ( $_POST['wrapper_page_id'] ) {
        wp_delete_post( $_POST['wrapper_page_id'], true );
        wp_trash_post( $_POST['wrapper_page_id'] );
    }
    die();
}
```

Disclosure Timeline

February 28, 2020 – Our Threat Intelligence team discovers and analyzes vulnerabilities in the IMPress for IDX Broker plugin while reviewing a recently patched vulnerability. We attempt to make contact with the plugin vendor.
March 2, 2020 – Firewall rule released for Wordfence Premium users.
March 19, 2020 – After followup with WordPress.org plugin team, plugin vendor confirms appropriate mailbox, and we provide them with full disclosure.
March 23, 2020 – Fully patched version becomes available.
April 1, 2020 – Firewall rule becomes available to Wordfence free users.

Conclusion

In today's post, we detailed several vulnerabilities including stored XSS and Post creation, modification, and deletion found in the IMPress for IDX Broker plugin. These flaws have been patched in version 2.6.2, and we recommend that users update to the latest version available immediately. Sites running Wordfence Premium have been protected from attacks against this vulnerability since March 2, 2020. Sites running the free version of Wordfence received the firewall rule update on April 1, 2020.
Did you enjoy this post? Share it!

Comments

2 Comments

Kadigan *
March 27, 2020
12:31 am

Is it possible to list registered AJAX actions, and then perform a scan of these functions (the files containing them) to see if they at least perform the basic nonce check? This is getting ridiculous!

Similarly, is it possible to enforce at least minimal capability checking to get into the admin area, or something? Like, if a plugin didn't do that, it'd be impossible to interact with it at all... (except enabling/disabling, of course)

Ram Gall *
March 27, 2020
7:08 am

Hi Kadigan!

When manually hunting for vulnerabilities, we regularly search through the files in a plugin for registered AJAX actions and check the associated functions. An automated scan for this type of thing probably wouldn't be practical, however, since many plugins use their own custom nonce and capability checking functions. Additionally, many AJAX actions are actually intended to be available to all site visitors, or all logged in users - even when manually reviewing a plugin, it's sometimes difficult to tell which user roles should be allowed to access certain functions. Finally, some plugins use non-standard methods to register AJAX actions - a manual review can find these, but it would be difficult scan for them automatically. As far as capability checking to get into the admin area is concerned, we've found that most plugins do a fairly good job of capability checking when it comes to actually displaying or loading menu pages containing sensitive functionality - subscribers typically only have access to update their own profiles.

Stay Updated

Sign up for news and updates from our panel of experienced security professionals.

you@example.com

☐ By checking this box I agree to the terms of service and privacy policy *

SIGN UP