Ram Gall

October 5, 2020

High Severity Vulnerabilities in Post Grid and Team Showcase Plugins

On September 14, 2020, our Threat Intelligence team discovered two high severity vulnerabilities in Post Grid, a WordPress plugin with over 60,000 installations. While investigating one of these vulnerabilities, we discovered that almost identical vulnerabilities were also present in Team Showcase, a separate plugin by the same author with over 6,000 installations.

We initially reached out to the plugin's developer, PickPlugins, on September 16, 2020 and provided full disclosure the next day. Patches for both plugins were made available only a few hours after we provided disclosure on September 17, 2020.

Wordfence Premium users received a firewall rule protecting both plugins from both vulnerabilities on September 16, 2020. Sites still running the free version of Wordfence will receive this rule after 30 days, on October 16, 2020.

Description: Stored Cross-Site Scripting (XSS)
Affected Products: Post Grid, Team Showcase
Plugin slug: post-grid,team
Affected Versions: Post Grid < 2.0.73 and Team Showcase < 1.22.16
CVE ID: CVE-2020-35936 and CVE-2020-35937
CVSS Score: 7.5 (High)
CVSS Vector: CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H
Fully Patched Version: Post Grid 2.0.73 and Team Showcase 1.22.16

Post Grid is a popular WordPress plugin that allows users to display their posts in a grid layout, while Team Showcase is designed to showcase an organization's team members. Both of these plugins allowed the import of custom layouts, and contained nearly identical functions in order to import these layouts. Post Grid no longer actually made use of the vulnerable import function, though the vulnerable code was still present.

In both cases, a logged-in attacker with minimal permissions such as subscriber could trigger the functions by sending an AJAX request, with the `action` set to `post_grid_import_xml_layouts` for the Post Grid plugin or `team_import_xml_layouts` for the Team Showcase plugin, with each action triggering a function with the same name.

Additionally, in the Post Grid plugin, the `post_grid_import_xml_layouts` function could also be triggered via a shortcode. By default, this meant that only authenticated users would be able to to activate it. Any 3rd party plugin allowing unauthenticated shortcode execution, however, would extend the vulnerability to unauthenticated attackers.

```
add_shortcode('post_grid_import_xml_layouts', 'post_grid_import_xml_layouts');
```

```
    $user_id = get_current_user_id();
    $source = isset($_POST['source']) ? sanitize_text_field($_POST['source']) : '';
    $skip = isset($_POST['skip']) ? sanitize_text_field($_POST['skip']) : '';


    if($skip == 'yes'){

        if(strpos($source, 'post-grid-pro')){
            $post_grid_info['import_pro_layouts'] = 'done';
        }else{
            $post_grid_info['import_layouts'] = 'done';
        }

        $response['skip_success'] = __('Import skipped','post-grid');
        update_option('post_grid_info', $post_grid_info);

        echo json_encode($response);
        die();
    }

    if(!empty($source)){
        $json_obj = file_get_contents($source);
    }else{
        $json_obj = '';
    }



    //$xml_json = json_encode($html_obj);
    $xml_arr = json_decode($json_obj, true);


    $items = isset($xml_arr['rss']['channel']['item']) ? $xml_arr['rss']['channel']['item'] : array();

    if(!empty($items))
    foreach ($items as $item){

        $post_title = isset($item['title']) ? $item['title'] : '';
        $postmeta = isset($item['postmeta']) ? $item['postmeta'] : array();

        $post_id = wp_insert_post(
            array(
                'post_title'    => $post_title,
                'post_content'  => '',
                'post_status'   => 'publish',
                'post_type'     => 'post_grid_layout',
                'post_author'   => $user_id,
            )
        );

//            echo '<br>';
//            echo $post_title. ' Created';
//            echo '<br>';


        foreach ($postmeta as $meta){

            $meta_key = isset($meta['meta_key']['__cdata']) ? $meta['meta_key']['__cdata'] : '';
            $meta_value = isset($meta['meta_value']['__cdata']) ? $meta['meta_value']['__cdata'] : '';

//            echo '<br>';
//            //var_dump(unserialize($meta_value));
//            echo '<br>';


            if($meta_key == 'layout_options' || $meta_key == 'layout_elements_data' || $meta_key == 'custom_scripts'
                print_r($meta_value);

                update_post_meta($post_id, $meta_key, unserialize($meta_value));
            }

        }



    }


    $response['success'] = __('Import done','post-grid');


    if(strpos($source, 'post-grid-pro')){
        $post_grid_info['import_pro_layouts'] = 'done';
    }else{
        $post_grid_info['import_layouts'] = 'done';
    }

    update_option('post_grid_info', $post_grid_info);


    echo json_encode($response);
    die();

}
add_action('wp_ajax_post_grid_import_xml_layouts', 'post_grid_import_xml_layouts');
```

Regardless of how the vulnerable function was triggered, an attacker could supply a `source` parameter pointing to a crafted malicious payload hosted elsewhere. The function would then open the file containing the payload, decode it, and create a new page layout based on its contents. The created layout included a `custom_scripts` section, and an attacker could add malicious JavaScript to the `custom_css` portion of this section. This would then be executed whenever an administrative user edited the layout or a visitor visited a page based on the layout.

Any malicious JavaScript added in this manner could be used to take over a site by adding a malicious administrator, adding a backdoor to plugin or theme files, or stealing the administrator's session information.

Description: PHP Object Injection
Affected Products: Post Grid, Team Showcase
Plugin slug: post-grid,team
Affected Versions: Post Grid < 2.0.73 and Team Showcase < 1.22.16
CVE ID: CVE-2020-35938 and CVE-2020-35939
CVSS Score: 7.5 (High)
CVSS Vector: CVSS:3.0/AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H
Fully Patched Version: Post Grid 2.0.73 and Team Showcase 1.22.16

The `post_grid_import_xml_layouts` and `team_import_xml_layouts` functions could also be used for PHP Object Injection using the same mechanism as the XSS attack. This was possible because the vulnerable functions unserialized the payload supplied in the `source` parameter.

As such an attacker could craft a string that would be unserialized into an active PHP Object. Although neither plugin utilized any vulnerable magic methods, if another plugin using a vulnerable magic method was installed, Object Injection could be used by an attacker. Doing so would allow a malicious actor to execute arbitrary code, delete or write files, or perform any number of other actions which could lead to site takeover.

As with the XSS vulnerability, the PHP Object injection vulnerability would typically require the attacker to have an account with at least subscriber level privileges. However, sites using a plugin or theme that allowed unauthenticated visitors to execute arbitrary shortcodes would be vulnerable to unauthenticated attackers.

Timeline

September 14, 2020 – Our Threat Intelligence team finds two vulnerabilities in the Post Grid plugin.
September 16, 2020 – We discover identical vulnerabilities in the Team Showcase plugin. We release a firewall rule for Wordfence Premium customers and reach out to PickPlugins, the developer for both plugins.
September 17, 2020 – PickPlugins responds, and we provide full disclosure. PickPlugins releases fixes for both plugins.
October 16, 2020 – The firewall rule becomes available to free Wordfence users.

Conclusion

In today's post, we detailed two high-severity vulnerabilities present in both the Post Grid plugin and the Team Showcase plugin, including a stored Cross-Site Scripting(XSS) vulnerability and a PHP Object Injection vulnerability.

Wordfence Premium users have been protected from attacks against both plugins since September 16, 2020. Sites still running the free version of Wordfence will receive the firewall rule on October 16, 2020.

If your site is running either of these plugins it is critical that you update to the latest version as soon as possible. At the time of this writing, the latest version of Post Grid is 2.2.72 and the latest version of Team Showcase is 1.22.16. If you

Special thanks to the plugin's developer, PickPlugins, for their rapid response in patching these vulnerabilities.

Did you enjoy this post? Share it!

Comments

No Comments