nfo CMS <= 8.1 (weixinreply.class.php) PHP Code ction Vulnerability

re Link:

w.metinfo.cn

ed Versions:

9, 8.0, and 8.1.

ability Description:

able code is located into the **/app/system/weixin/include/class/weixinreply.class.php** script.

, within the `weixinreply::wxAdminLogin()` method:

```
public function wxAdminLogin($data = array(),$code = '')
{
    global $_M;
    $weixinapi = load::mod_class('weixin/weixinapi','new');
    $login_code = cache::get("weixin/".$code);
     if ($login_code) {
        cache::put("weixin/".$login_code,$data['FromUserName']);
     }
    return;
}
```

passed through the `EventKey` and `FromUserName` XML tags from the HTTP request body when dispatching *weixin* API ly sanitized before being used in a call to the `cache::get()` and `cache::put()` methods respectively.

, the `$code` parameter may include Path Traversal sequences, making the `cache::get()` method into including arbitrar e abused to set the `$login_code` variable to the **"Array"** string by including an arbitrary cache file. Subsequently, the `cac` l write the `FromUserName` parameter into the **/cache/weixin/Array.php** file, embedding it within double quotes:

```
public static function put($file, $data, $type = 'php')
{
    global $_M;

    load::sys_func('file');
    $save = PATH_CACHE . $file . '.' . $type;
    makefile($save);
    #$data = str_replace(array("\"", "\\"), array("\\\"", "\\\\"), $data);
    if (!is_array($data)) {
        file_put_contents($save, "<?php\ndefined('IN_MET') or exit('No permission');\n\$cache=\"{$data}\"
    } else {
        $info = var_export($data, true);
        $info = "<?php\ndefined('IN_MET') or exit('No permission');\n\$cache = {$info};\n?>";
        file_put_contents($save, $info);
    }
}
```

e exploited by remote, unauthenticated attackers to inject and execute arbitrary PHP code by abusing PHP's complex cu unauthenticated Remote Code Execution (RCE).

n MetInfo is running on non-Windows servers, successful exploitation of this vulnerability requires the **/cache/weixin/** c
n is created when installing and configuring the official WeChat plugin.

of Concept:

mainsecurity.com/pocs/CVE-2026-29014.php

on:

solution is currently available.

sure Timeline:

6] – Vendor contacted through several `@metinfo.cn` and `@mituo.cn` email addresses, no response

6] – Tried to reach out to the vendor again, no response

6] – Tried to reach out to the vendor once again, no response

6] – Tried to reach out to the vendor through Weibo, no response

6] – CVE identifier requested

6] – CVE identifier assigned

6] – Public disclosure

ference:

29014 has been assigned to this vulnerability.

s:

ty discovered by Egidio Romano.