Title: WAF (Cloudflare) Bypass - and direct server access to vulnerable SSH server

Vector: https://rarible.com

Vulnerability Description:

When a company implements a WAF on their site, in this case one such as cloudflare, the original IP address is hidden, and in effect so is the access to the website itself VIA ftp, and ssh.

This can also help protect a website from denial of service attacks, and preserve site mobility and security.

Bypassing this vector, and gaining access to SSH and HTTP related services that are not behind the firewall of the service, is called WAF bypass.

Findings:

When doing some OSINT and server recon of the Rarible website, managed to come across an IP address directly linked to the website:

"65.108.241.68"

Upon further analysis and enumeration of this website, I noticed it had 3 ports open: 80, 443, 22.

When Inspecting the headers received when querying this IP address via port 80, got the following response:

```
{'Server': 'nginx'
 'Date': 'Wed
 05 Apr 2023 15:32:03 GMT'
 'Content-Type': 'text/html; charset=utf-8'
 'Content-Length': '2127'
 'Connection': 'keep-alive'
 'X-Powered-By': 'Express'
 'Access-Control-Allow-Origin': 'https://rarible.com'
 'Vary': 'Origin'
 'ETag': 'W/"84f-BbU2Uum4GxlhylVLkOO4dB24nMk"'
 'X-Response-Time': '15.72ms'
 'Server-Timing': 'renderServerSideApp;dur=15.718955'}
```

Notice, in the access control headers, specifically lists '**https://rarible.com**' as an allowed origin for traffic.
Upon further looking at the other port listed as a web service, port 443 on the IP address, came across an SSL certificate:

SSL Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            22:81:7c:0b:13:e1:79:eb:8d:00:5d:9a:0f:a0:f7:5e:e1:d2:7c:6f
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN=wlbl-proxy1.ext.rarible.com
        Validity
            Not Before: Nov 24 07:35:59 2022 GMT
            Not After : Nov 21 07:35:59 2032 GMT
        Subject: CN=wlbl-proxy1.ext.rarible.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:c9:5d:fa:7d:06:f3:e0:38:37:62:ba:ca:fa:36:
                    59:03:95:8f:84:b5:43:8b:d0:61:4d:37:b3:43:15:
                    53:65:2d:77:94:65:9c:48:b7:29:e7:dc:08:ae:c5:
                    65:5b:26:d3:c6:50:2c:2d:2f:12:94:8e:5a:23:1c:
                    32:6b:3f:d8:0a:e6:66:63:70:7f:c4:5c:f4:9e:32:
                    f2:d7:46:f3:8a:6c:9e:3f:b2:76:82:b6:7b:d7:4d:
                    6c:35:ac:d7:de:5f:e2:9d:41:6a:24:6c:bc:34:03:
                    56:06:67:56:6e:ee:33:ef:c7:7b:51:60:23:f7:9d:
                    06:63:0f:2f:b2:25:ac:0f:03:51:67:c8:72:47:1b:
                    00:8c:2c:b1:87:22:73:99:64:72:e8:98:de:30:ea:
                    9f:83:ae:87:02:fe:49:69:c8:92:39:4b:c2:e7:fd:
                    40:ab:85:ce:97:1e:31:d5:66:5a:85:2c:76:9d:e2:
                    65:57:4b:bc:3f:8e:1d:11:23:63:13:79:20:32:a7:
                    17:36:4b:7c:28:3c:0d:a1:05:94:b7:ec:42:cf:c7:
                    bc:4c:ff:bf:1c:44:b8:aa:a7:af:ff:85:cd:8c:c4:
                    0d:c3:65:7a:42:7d:d4:b2:48:43:d8:ee:32:d1:11:
                    d2:97:61:fc:86:02:1a:a1:d5:e7:bb:57:df:60:07:
                    8e:95
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Alternative Name:
                DNS:wlbl-proxy1.ext.rarible.com
    Signature Algorithm: sha256WithRSAEncryption
```

On it, you can see highlighted, 2 proxy like servers, under the direct domain of "**rarible.com**"

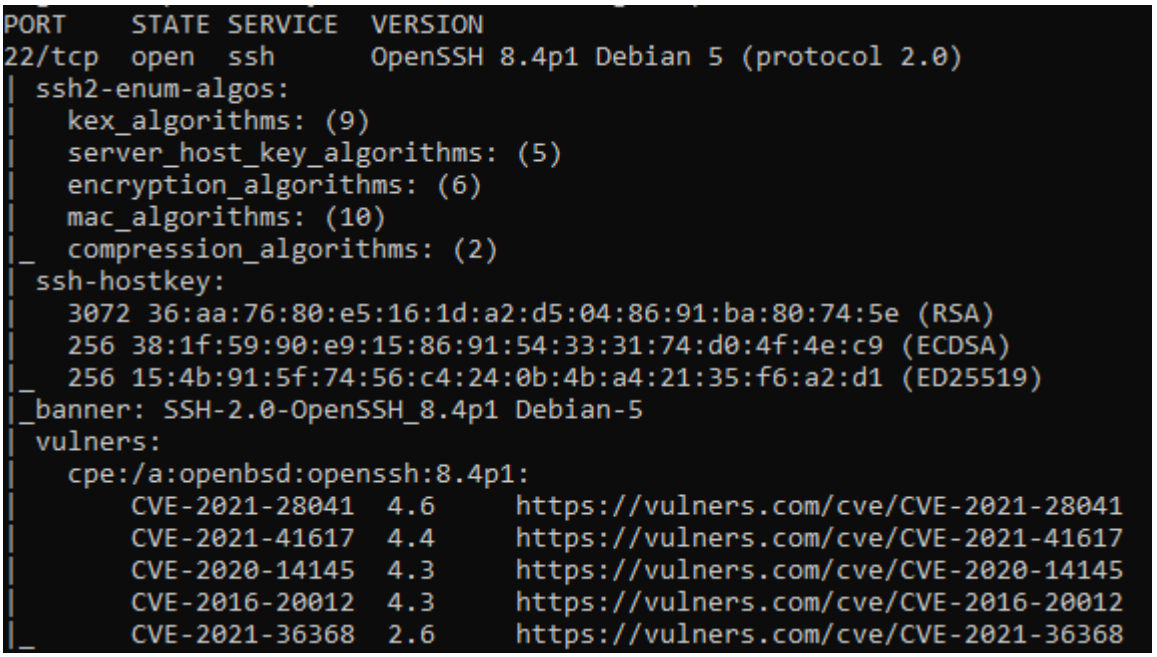
Clearly indicating that this IP address belongs to this domain, and hosts a web server on the domain rarible.com

Upon some vulnerability analysis and further OSINT of the last found port: 22

Noticed it was running an out of date and vulnerabile version of the **"OpenSSH Debian"** service- more specifically:

"OpenSSH 8.4p1 Debian 5 (protocol 2.0)"

With several listed security vulnerabilities:



Conclusion:

With the correct skillset, and time an attacker may be able to use the found, unprotected information to:

- Brute force or exploit the vulnerable discovered SSH server, and use it to take over the website

- Launch a denial of service on the web application, due to it not being in the protection of cloudflare anymore

- Gather further information, or use previously exploited system weaknesses or vulnerabilities to pose a risk to the service