

Forum: [Hack Forums](#)

Thread: [Grabbing the backend of a cloudflare site](#)

Sennheiser - January 22, 2015, 10:06 pm

Anyone able to pull a CF ip?

I tried using

<http://www.crimeflare.com/>

and

<http://cfres.unkn0wndevs.pw/>

IP Lookup - January 22, 2015, 10:14 pm

I can find the back end of any site hiding behind Cloudflare.

Try me.

SIM-1241186 - January 23, 2015, 3:55 pm

<http://www.securityidiots.com/Web-Pentest/Information-Gathering/Cloudflare-Bypass/Part-1-Understanding-Cloudflare-Security.html>

<http://www.securityidiots.com/Web-Pentest/Information-Gathering/Cloudflare-Bypass/Part-2-Cloudflare-Security-Bypass.html>

<http://www.securityidiots.com/Web-Pentest/Information-Gathering/Cloudflare-Bypass/Part-3-Cloudflare-Security-Bypass.html>

<http://www.securityidiots.com/Web-Pentest/Information-Gathering/Cloudflare-Bypass/Part-4-Cloudflare-Security-Bypass.html>

Sennheiser - January 24, 2015, 9:46 am

still need this,

Unable to grab it using any of the methods

Tony S - January 24, 2015, 10:32 am

(Jan 24, 2015 10:46 AM)Sennheiser Wrote: [\[CITE\]](#) still need this,

Unable to grab it using any of the methods

Send an abuse report containing pretty much anything to CloudFlare. For the most part, as far as I know, their responses are automated and every response will follow the same format.

Inside the complaint response, you'll find a snippet of a WHOIS ran on the IP address that whatever domain you reported is pointing to. Inside of this WHOIS snippet will be the [NetName](#) (ex: LVL-T-GOGL-8-8-8 or OVH-ARIN-3) that the IP address is currently assigned to. IP addresses aren't normally allocated singularly, instead ranges are typically allocated to organizations.

Let's say the abuse complaint has came back and said that the NetName of the IP address that the domain is pointing to is OVH-ARIN-3. [According to ARIN](#), that NetName has been allocated the IP

range 142.4.192.0 - 142.4.223.255. That means that the IP address that the domain is pointing to is inside this range - which means that we've narrowed it down to 33×256 (7936) separate IP addresses.

Scanning the range is trivial, infact it won't take long whatsoever.

Let's say the domain is foo.com. Visit foo.com in your browser and take a look at the pages source - try to find a unique string (ex: <title>foo.com - All of your foo needs!</title>). Now, for every IP address in that range, you'll just need to run a command similar to -

```
$ curl -H "Host: foo.com" 142.4.<octet>.<octet> | grep '<title>foo.com - All of your foo needs!</title>'
```

And if anything is printed then you know that you have found the correct IP address.