Shodan - The Complete Guide, Featured on TryHackMe

January 1, 2021 · 10 min · Bee

Shodan.io is a search engine for the Internet of Things.

Ever wondered how you can find publicly accessible CCTV cameras? What about finding out how many Pi-Holes are publicly accessible?

Or whether your office coffee machine is on the internet?

Shodan.io is the answer!

Shodan scans the whole internet and indexes the services run on each IP address.

Note: if you are following along, you'll need a premium Shodan account.

Finding services

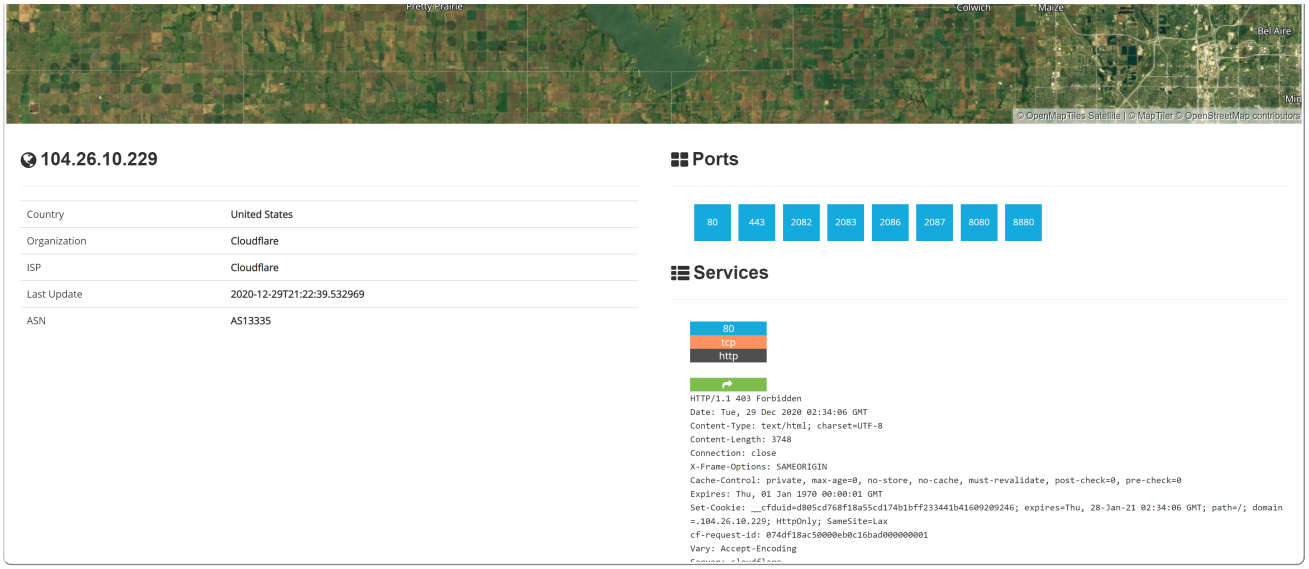
Let's say we are performing a pentest on a company, and we want to find out what services one of their servers run.

We need to grab their IP address. We can do this using `ping` .

We can ping tryhackme.com and the ping response will tell us their IP address.

```
Pinging tryhackme.com [142.93.194.248] with 32 bytes of data:
```

Then once we do this, we put the IP address into Shodan to get:



We can see that TryHackMe runs on Cloudflare in the United States and they have many ports open.

Cloudflare acts as a proxy between TryHackMe and their real servers. If we were pentesting a large company, this isn't helpful. We need some way to get their IP addresses.

We can do this using Autonomous System Numbers.

Autonomous System Numbers

An autonomous system number (ASN) is a global identifier of a range of IP addresses. If you are an enormous company like Google you will likely have your own ASN for all of the IP addresses you own.

We can put the IP address into an ASN lookup tool such as https://www.ultratools.com/tools/asnInfo ,

Which tells us they have the ASN AS14061.

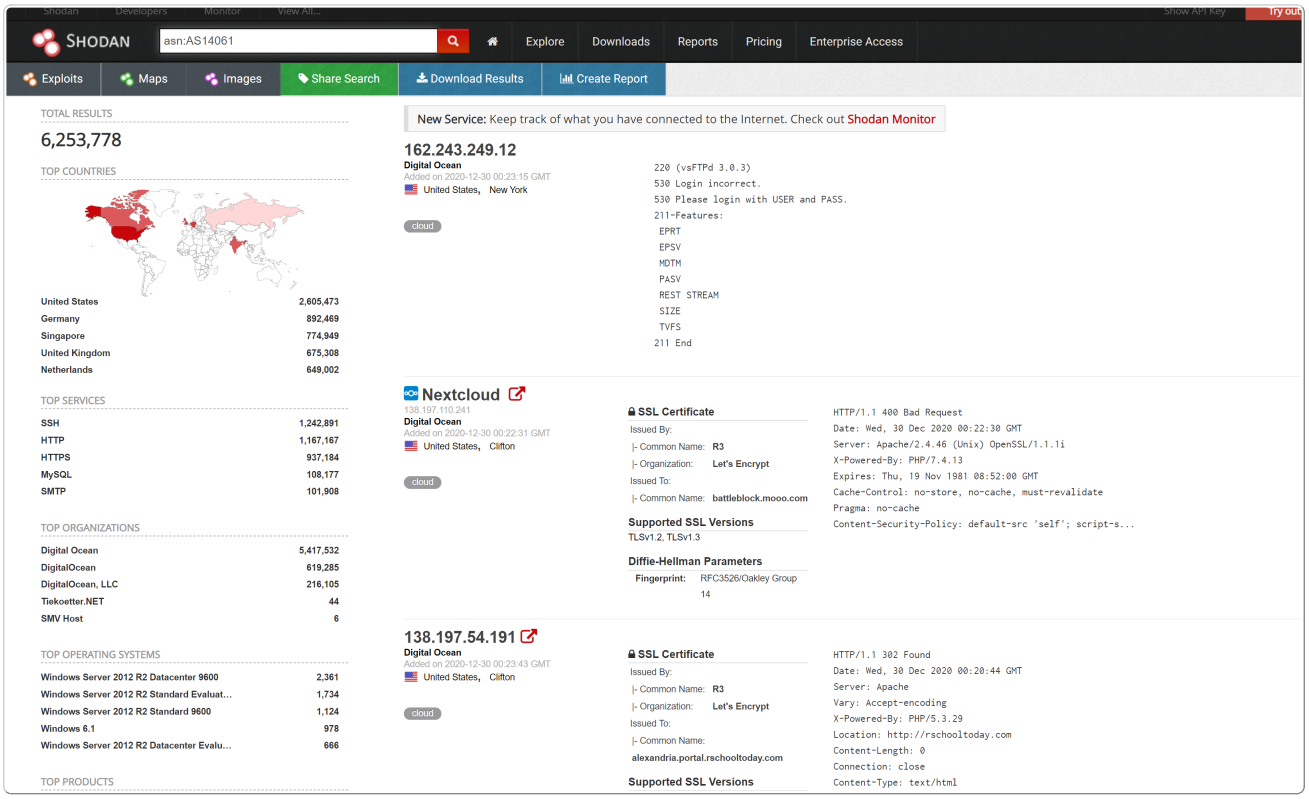
Tryhackme isn't a mega large corporation, so they don't own their own ASN. When we google AS14061 we can see it is a DigitalOcean ASN number.

On Shodan.io, we can search using the ASN filter. The filter is `ASN:[number]` where number is the number we got from earlier, which is AS14061.

Doing this, we can see a whole range 6.2 million websites, in fact) that are on this

one single ASN!

https://www.shodan.io/search?query=asn%3AAS14061



Knowing the ASN is helpful, because we can search Shodan for things such as coffee makers or vulnerable computers within our ASN, which we know (if we are a large company) is on our network.

Getting started

Time to dig in! If you get stuck, look at the previous task for some help! :)

Banners

To get the most out of Shodan, it's important to understand the search query syntax.

Devices run services, and Shodan stores informatiion about them. The information is stored in a *banner*. It's the most fundamental part of Shodan.

An example banner looks like:

```
{
```

```
        "data": "Moxa Nport Device",
        "Status": "Authentication disabled",
        "Name": "NP5232I_4728",
        "MAC": "00:90:e8:47:10:2d",
        "ip_str": "46.252.132.235",
        "port": 4800,
        "org": "Starhub Mobile",
        "location": {
              "country_code": "SG"
        }

}
```


We're looking at the output of a single port, which includes information about the IP and authentication details.

You don't really see this outside of the API, so we won't delve into it.

Filters

On the Shodan.io homepage, we can click on "explore" to view the most up voted search queries. The most popular one is webcams.

https://www.shodan.io/explore

Note: this is a grey area. It is legal to view a publicly accessible webcam, it is illegal to try to break into a password protected one. Use your brain and research the laws of your country!

One of the other most up voted searches is a search for MYSQL databases.

https://www.shodan.io/search?query=product%3AMySQL

If we look at the search, we can see it is another filter.

product:MySQL

Knowing this, we can actually combine 2 searches into 1.

On TryHackMe's ASN, let's try to find some MYSQL servers.

We use this search query

asn:AS14061 product:MySQL

And ta-da! We have MYSQL servers on the TryHackMe ASN (which is really the DigitalOcean ASN).

https://www.shodan.io/search?query=asn%3AAS14061+product%3AMySQL

Shodan has many powerful filters. My favourite one is the vuln filter, which let's us search for IP addresses vulnerable to an exploit.

Let's say we want to find IP addresses vulnerable to Eternal Blue:

vuln:ms17-010

However, this is only available for academic or business users, to prevent script kiddies from abusing this!

Here are some nice filters we can use on Shodan:

City Country Geo (coordinates) Hostname net (based on IP / CIDR) os (find operating systems) port before/after (timeframes)

API

Shodan.io has an API! It requires an account, so I won't talk about it here.

If you want to explore the Shodan API, I've written a blog post about finding Pi-Holes with it here:

https://github.com/beesecurity/How-I-Hacked-Your-Pi-Hole/blob/master/README.md

The API lets us programmatically search Shodan and receive a list of IP addresses in return. If we are a company, we can write a script to check over our IP addresses to see if any of them are vulnerable.

PS: You can automatically filter on Shodan by clicking the things in the left hand side bar!

Shodan Monitor

Shodan Monitor is an application for monitoring your devices in your own network. In their words:

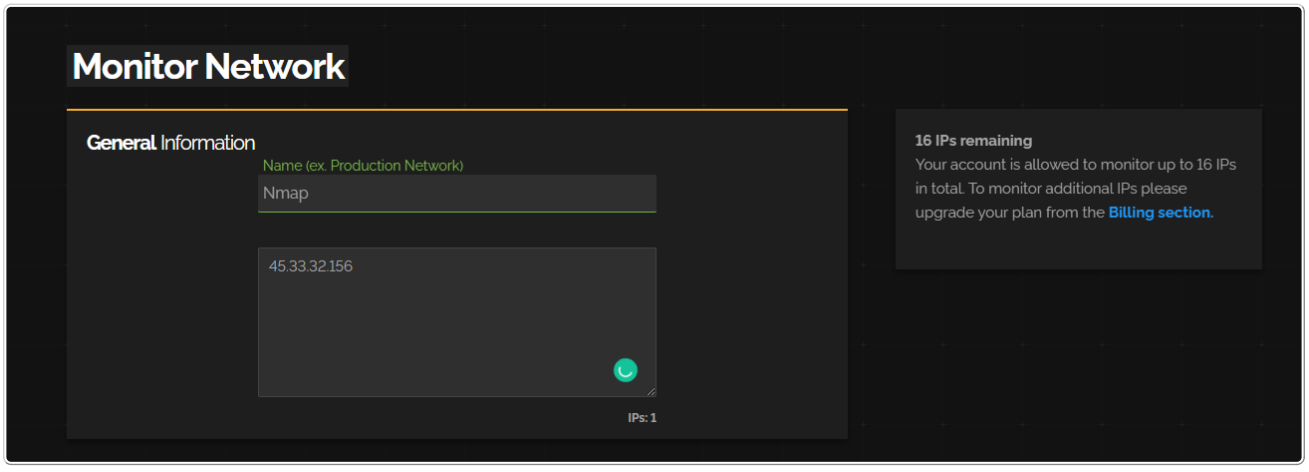
> Keep track of the devices that you have exposed to the Internet. Setup notifications, launch scans and gain complete visibility into what you have connected.

Previously we had to do this using their API, but now we have this fancy application.

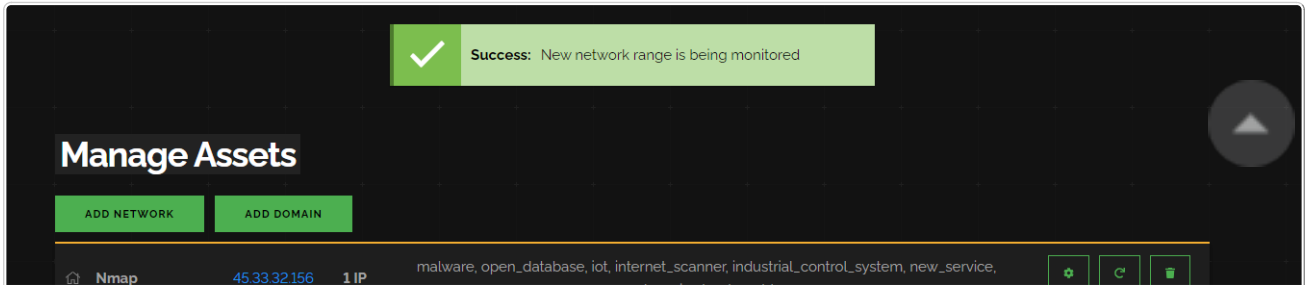
Access the dashboard via this link:

https://monitor.shodan.io/dashboard

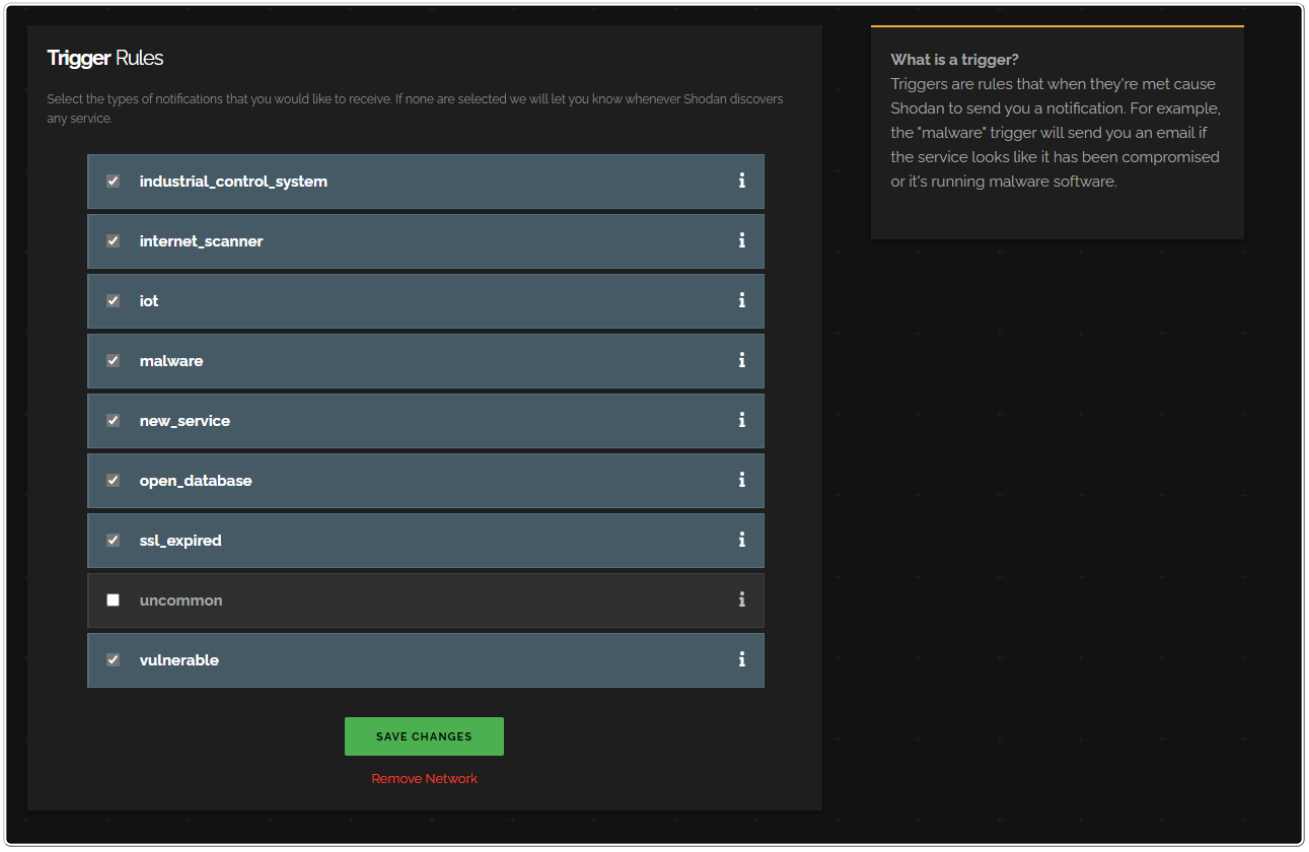
You'll see it's asking for an IP range.



Once we add a network, we can see it in our dashboard.

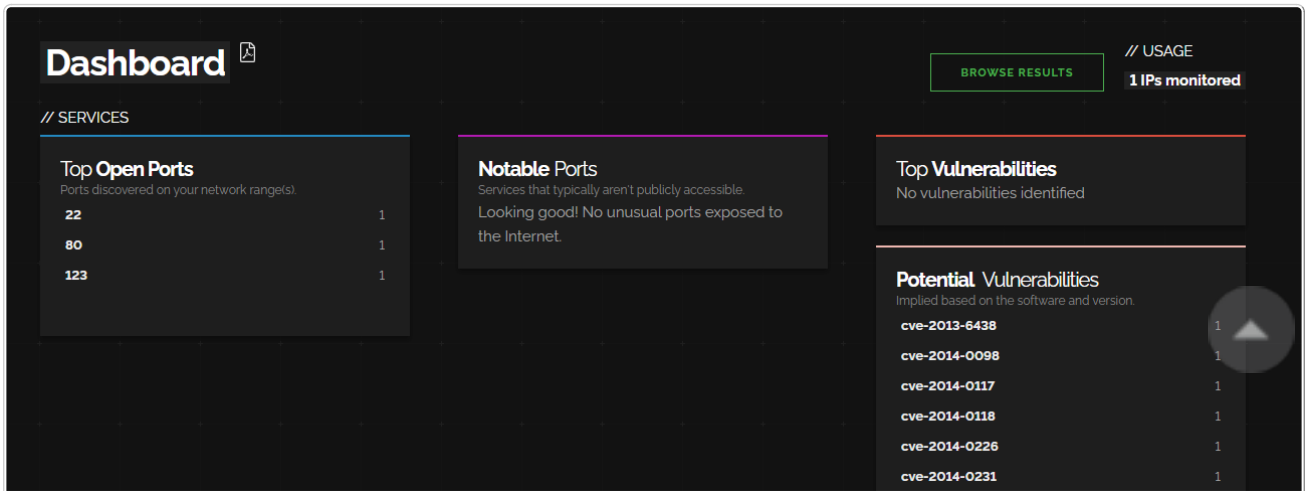


If we click on the settings cog, we can see that we have a range of “scans” Shodan performs against our network.



Anytime Shodan detects a security vulnrability in one of these categories, it will email us.

If we go to the dashboard again we can see it lays some things out for us.



Most notably:

- Top Open Ports (most common)
- Top Vulenrabilites (stuff we need to deal with right away)
- Notable Ports (unusual ports that are open)
- Potential Vulenrabilites
- Nootable IPs (things we should investigate in more depth).

The interesting part is that you can actually monitor other peoples networks using this. For bug bounties you can save a list of IPs and Shodan will email you if it finds any problems.

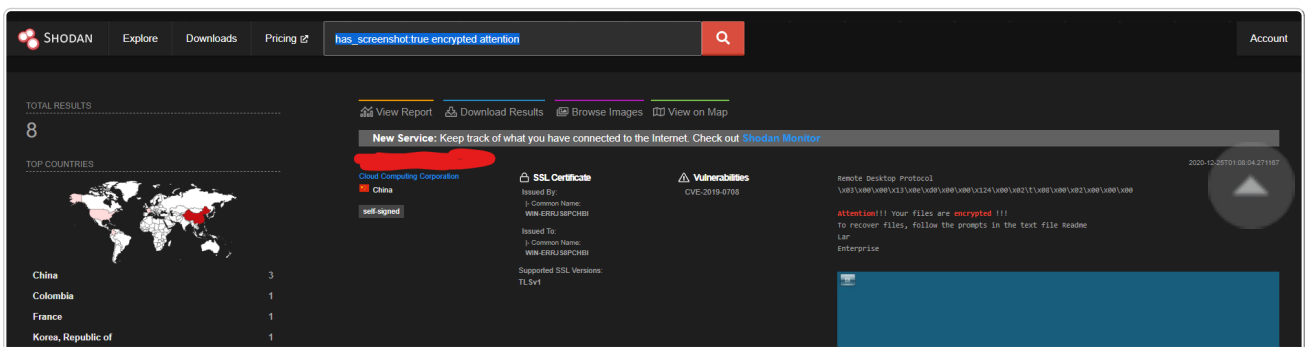
Shodan Dorking

Shodan has some lovely webpages with Dorks that allow us to find things. Their search example webpages features some.

Some fun ones include:

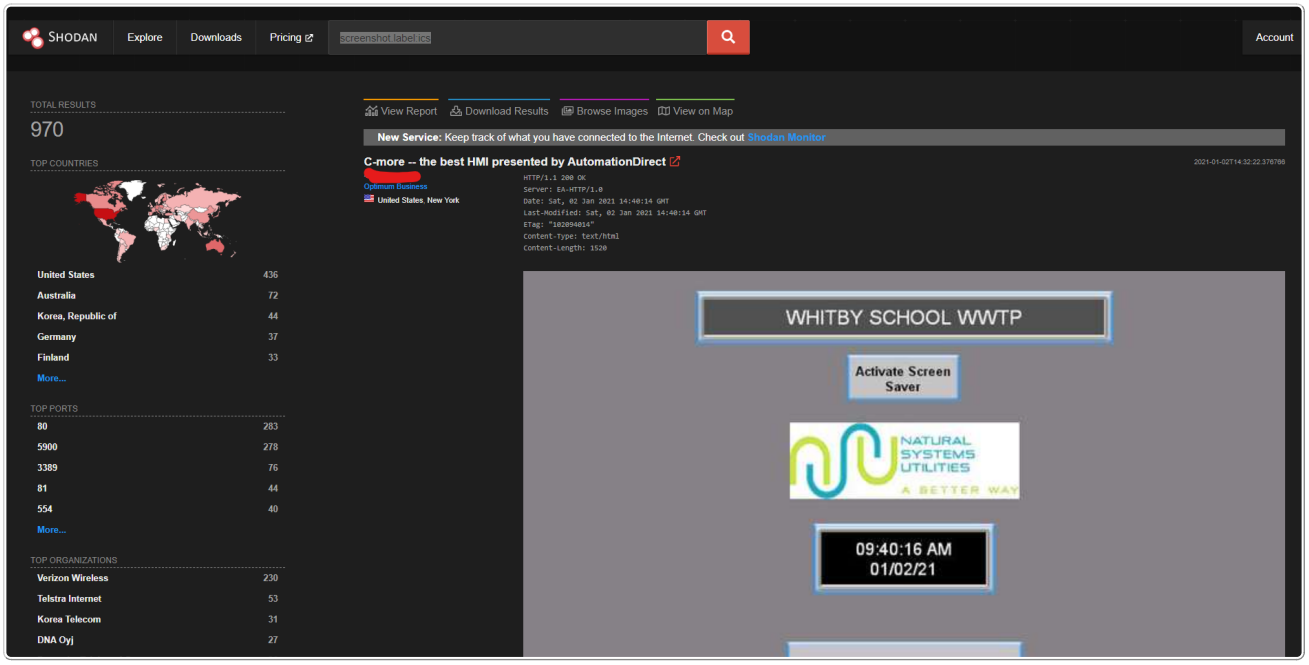
```
has_screenshot:true encrypted attention
```

Which uses optical character recongition and remote desktop to find machines compromised by ransomware on the internet.




```
screenshot.label:ics
```

Using Machine Learning, Shodan can identify industrial control systems which are connected to the internet.




```
vuln:CVE-2014-0160
```

Internet connected machines vulenrable to heartbleed. Note: CVE search is only allowed to academic or business subscribers.

Solar Winds Supply Chain Attack by using Favicons:

```
http.favicon.hash:-1776962843
```

You can find more Shodan Dorks on GitHub or in Shodan's Explore Page

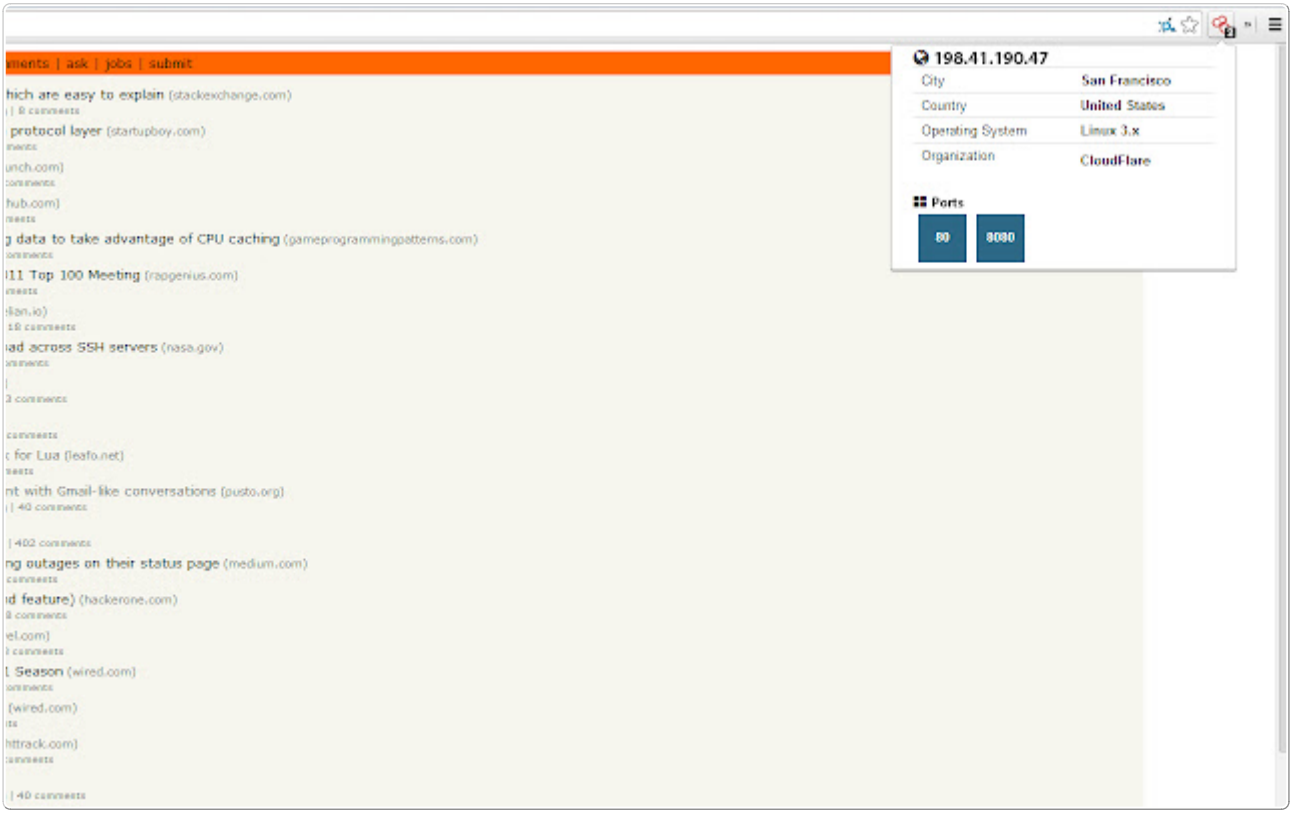
Shodan Extension

Shodan also has an extension.

https://chrome.google.com/webstore/detail/shodan/jjalcfnidlmpjhdfepjhjbhnhkbgleap

When installed, you can click on it and it'll tell you the IP address of the webserver running, what ports are open, where it's based and if it has any security issues.

I imagine this is a good extension for any people interested in bug bounties, being quickly able to tell if a system looks vulnerable or not based on the Shodan output.



How I Hacked Your Pi-Hole

There are 5308 publically available Pi-Holes according to Shodan.io. This article will demonstrate how bad of an idea this is.

If you've spent any time over at r/pihole, you'll know they always talk about not having publically available Pi-Holes. But, why? What's the harm?

Out of the 5308 Pi-Holes publically available, I found 100 of them are vulnerable.

Vulnerable means:

1. You can access /admin/index.php?login. This is because not all Pi-Holes listed by Shodan work in the way we expect them to.

However, in my research I came across something interesting:

Passwordless Pi-Holes

I came across many passwordless Pi-Holes. 578 passwordless Pi-Holes to be correct. These aren't considered "vulnerable", because they are by default open. It's incredibly easy to find these, if you click around on Shodan enough you'll find them. You can also search the Shodan API (like I did below) and try to access a page, `/admin/queries.php` and see if it doesn't prompt for a password. Some of these Pi-Holes are meant to be public. You can tell because their domain name includes "public-pihole".

It's hard to differentiate between Pi-Holes meant to be publically accessible and those that aren't, so I haven't explored these much. Just know that these exist and to not make a publically facing Pi-Hole without a password for your personal use.

Finding these Pi-Holes

Shodan.io is a service that scans the web. It finds IoT or other devices like Pi-Hole. Using the Shodan API, we can programatically explore these Pi-Holes. Or, you can click here and explore them manually.

Finding these Pi-Holes took a minimal amount of code. I was surprised to find my Pi-Hole on this list. You need a Shodan membership. Also, don't attack Pi-Holes you don't own.

```
from shodan import Shodan
import requests
api = Shodan('API_KEY')
```

```
def url_ok(url):
    r = requests.head("http://" + url)
    return r.status_code == 200

def check_page(url):
    r = requests.get("http://" + url + "/admin/")
    return "Pi-hole" in r.text

def pruneIPS(vulnerableIPs):
    for i in vulnerableIPs:
        if not url_ok(i):
            if not check_page(i):
                vulnerableIPs.remove(i)
    return vulnerableIPs

result = api.search("pi-hole")

VulnerableIP = []
for service in result['matches']:
    VulnerableIP.append(service['ip_str'])
```

Hacking the Pi-Hole

Pi-Hole doesn't block bruteforcing. You can enter 200,000 incorrect passwords and Pi-Hole wouldn't care. How easy is it to create a brute-force attack? With Hydra - very easy. Hydra is a brute-forcing tool that uses a dictionary to attack a target. The payload for brute-forcing a Pi-Hole is:

```
hydra -l '' -P /usr/share/wordlists/rockyou.txt 192.168.0.1 http-post-form "
```

Our wordlist is the infamous rockyou.txt. We use the http-post-form module and enter some information. The form to post, the name of the variable for the password ("pw"), and what will be on the page when a failed login attempt happens (the forgotten password box).

Since Pi-Hole doesn't block brute-force attacks, it makes it trivial to brute-force most of the Pi-Holes assuming password length is low (under 9 chars optimally, although 9 chars can be done in 19 hours https://howsecureismypassword.net/). It's worth mentioning that Pi-Hole's default password is very secure, but the lack

of any timeouts sucks. Not to mention that most people change their password to something more human.

To recap, it is possible to get a list of all the vulnerable Pi-Holes and to brute-force their passwords with a dictionary attack.

I will not show code for this, because it is illegal. However, you can see the 2 separate parts in action above. Please test your own Pi-Hole. Do not test a Pi-Hole you do not own.

What can an attacker see?

Once the attacker has gained access to your Pi-Hole, they can see every website you visit. Your hosts' files, your clients and their IP addresses. Your top domains and your top blocked domains.

Now, they have a pretty good idea of who you are. They know all your devices. They know your password (which, most people would reuse on at least one of their devices). And they know you, pretty well actually. Considering your entire internet history is there. **There is no point in using a Pi-Hole if all of your DNS information is as easily accessible as this.**

DNS Amplification Attacks

An attacker could use your DNS server to perform a DDoS attack. This is very common and is called a DNS amplification attack. This is likely illegal for you to allow this, depending on the country. Another reason to not use a publically facing Pi-Hole.

How to protect yourself

Search your IP address on Shodan.io. Read the output. Shodan will tell you what ports are open, and will let you know whether it thinks a Pi-Hole service is

running.

1. Use a VPN to connect to your Pi-Hole: https://www.reddit.com/r/pihole/comments/bl4ka8/guide_pihole_on_the_go_with_wireguard/
2. Don't have a publically facing Pi-Hole
3. Choose a very strong password
4. Turn your Pi-Hole into a no-logs Pi-Hole

Fun Facts

Statistically, most people with exploitable Pi-Holes used Deutsche Telekom AG. However, most publically accessible Pi-Holes are hosted on:

1. Digital Ocean
2. OVH SAS
3. Google Cloud
4. Deutsche Telekom AG

Infosec