Scanning the Internet for Vulnerable Devices

Anna Sim
UT Austin

Sean Wang
UT Austin

ABSTRACT

Shodan is a popular search engine used to detect Internet-facing devices by performing port scans on a network. By using this tool, we are able to collect data from IPs across the Internet, focusing on select vulnerabilities. In this paper, we will discuss how Shodan is used to find numbers of remote machines on the Internet that are vulnerable from selected CVEs. The data is then analyzed to see what the detected machines have in common. After recording data from Shodan, we then created our own scanner to scan the Internet for vulnerable machines on the same ports as those of the identified devices in the CVEs. The results are then compared to those of Shodan's scan.

1. INTRODUCTION

With the growth and popularization of the Internet of Things (IoT), more and more devices are becoming connected to each other through the Internet. This allows many devices to now have the convenience and accessibility of being connected to other devices through the Internet, such as printing to a shared printer connected to the Internet or controlling security cameras remotely.

However, a key problem that comes hand in hand with these advantages is that any vulnerabilities across millions of endpoints could allow unwanted and unauthorized guests to control the device remotely. Devices without proper security in place are vulnerable to attacks such as Denial of Service or a remote control protocol. Attackers can abuse these gaps in security to gain access to information the victim might have or hijack the device and use it for their own purposes. If a device on a subnet were left unsecured and gets hijacked, attackers could even use it as a backdoor to other devices connected to the same subnet. There are plenty of tools that attackers can use to discover potentially useful information to perform these attacks, such as zmap [7], nmap [9], and Shodan [1]. Both zmap [7] and nmap [9] are used to scan the Internet, but the latter can give much more detailed information, such as the device's name, OS, and even hardware information through OS fingerprinting and banner grabbing. However, the search engine tool, Shodan, can provide a way for attackers to filter IPs for certain vulnerabilities as well and it works by searching and indexing any connected device. These devices range from home routers to traffic lights, and Shodan can produce useful information through it's banner grabbing capabilities, similar to nmap. The service banner contains various information for an IoT device, including geographic location, IP address, software version, make and model, etc. for specific ports. Currently, there are many computer systems, including traffic lights, security cameras, or industrial control systems, that have little to no security, leaving them vulnerable to attackers[6].

The approach taken in this paper is to determine some Common Vulnerabilities and Exposures (CVEs) that can be detected remotely and then utilize Shodan to see how many machines have those vulnerabilities and what similarities those machines have. Shodan is a quick and easy way to explore the large spanned IoT and detect key vulnerabilities in Internet-facing devices. Although many people believe Shodan is a privacy concern due to its detailed device-level insights, it is completely legal and serves to only collect already available data to the public. After scanning the ports through Shodan, our next step is creating a custom scanning tool that will be used to see if we obtain similar results to that of Shodan's scan.

Overall, the setup is fairly straightforward. The data returned by Shodan's search is split up into several pieces based on the different search terms used and by the CVE that the search terms target. On the other hand, the methods for a custom scanner for specific CVEs depend on the specific CVEs chosen, but have a similar pipeline: scan for IPs that respond on a specific port, establish socket connections with those IPs, grab banners from devices we can connect to, and search the banners for information of interest. In this case, we look at CVE-2014-2256, CVE-2019-0708, and CVE-2018-0101.

Evaluating the performance of our custom tool, we see that although our results are not as numerous and impressive as the Shodan searches, it is still possible to find these vulnerabilities in some devices with relative ease.

The main contributions of this paper are as follows:

- We explore more about the vulnerabilities being searched for and provide more detail on how these can be exploited for some specific systems.
- We show how these vulnerabilities can be detected using Shodan and what information can be gathered

from these scans. We analyze this information and discuss how this could be used maliciously.

- We create our own scanner and compare its results to Shodan to get insight on how information of specific systems is exposed to the Internet.

2. PRACTICAL IMPLICATIONS

The number of devices connected to the Internet is growing exponentially, increasing the chances that malicious actors will be able to find vulnerable Internet-connected devices. These devices can expose a wide variety of remotely accessible services and are easily identified through search engines, like Shodan, designed for IoT devices. This poses as a privacy concern for many, since Shodan's scans can reveal a lot of private information about a device.

We created a tool to scan the Internet because it is useful in finding specific information pertaining to vulnerabilities in a device that malicious actors could use to carry out an attack. The user is then able to take precautionary security steps to prevent any attackers from maliciously interfering with the device's normal functionality, such as a Denial of Service attack or using it to attack other devices on the same network that may be more secure to out of network attacks. Thus, by performing regular scans on a specific subnet or even the whole Internet, a user would able to keep up to date if any new devices connected to the network have similar vulnerabilities.

3. METHODS

Since there are three different CVEs to look at, the methods used to scan for devices vulnerable to those will be slightly different, both with Shodan and our custom scanning tool. The CVEs listed below were chosen due to their commonality, since this would mean that there would be more information to work with. Additionally, that gives the custom scanner more opportunities in the situation that it does not perform as well as Shodan, so that a better comparison can be made between the two scanning methods.

For CVE-2014-2256, the vulnerability is described as an issue with Siemens SIMATIC S7-1200 CPU PLC devices that have an older firmware that allows remote attackers to execute a Denial of Service (DoS) attack[2]. On Shodan, the exact search term used to find these vulnerable devices is: `siemens port:"102"`. The DoS attack is caused by attackers sending ISO-TSAP, or TCP, packets to specifically port 102, since this port is left open by the firmware. Then, for the custom tool, we use `zmap` [7] to scan on port 102 and get a list of devices probed. Then, we filter for devices by establishing a socket connection for each probed IP and grabbing a banner from the device to get more information. We then search the information for "Siemens" to filter the list of IPs for vulnerable devices.

CVE-2019-0708 is described as a remote code execution vulnerability that exists in Microsoft's Remote Desktop Protocol in Windows[4]. In Shodan, we need several searches for devices with port 3389 open and running versions of Windows that could potentially be vulnerable. This means our searches combine the term `port:"3389"` with one of the following: `os:"Windows 7 or 8"`, `os:"Windows XP"`, `2003`, or `2008`. In this case, the last two terms refer to versions of Windows Server. Our custom tool for this vulnerability follows a similar pipeline to the previous vulnerability on Siemens devices, utilizing `zmap` [7] to gather IPs probed on port 3389. However, in this case, after connecting to each IP and attempting to get a banner from each one, we search the banner information for any of the following terms: `Windows 7`, `Windows 8`, `Windows XP`, or `Windows Server`.

Finally, CVE-2018-0101 is a vulnerability in Cisco's Adaptive Security Appliance (ASA) Software, specifically the Secure Sockets Layer (SSL) VPN functionality. This vulnerability could allow remote execution of a code and is possible since enabling the webvpn feature will cause an attempted double free on a region of memory on the device. This is detectable by checking to see if the Cisco ASA device has the webvpn feature enabled. The National Vulnerability Database provides a list of specific devices affected by this vulnerability[3]. In Shodan, we use the search term `"Set-Cookie: webvpn=;" ssl:"ASA Temporary"`, since the ASA Software sets the webvpn cookie. Using just the webvpn filter may also capture other devices where some other web applications might have set the cookie as well, but the ASA Temporary SSL certificate filter gives a narrower set of search results of devices vulnerable to this CVE. Our custom tool to scan for this vulnerability has a similar pipeline, probing for IPs on port 443 instead. Additionally, rather than grab a banner from the device, we want to look for the term "ASA Temporary" in the SSL Certificate. Using the Python module for OpenSSL **??**, we attempted to obtain SSL Certificates from each IP that was probed and for the successful attempts, we then search the certificate for the term.

4. EXPERIMENTAL RESULTS

Our experimental results were still interesting despite some of the issues we ran into. Often, running the `zmap` **??** scan would cause our home router to shutdown during long scans, so to make up for that we did several shorter scans to increase our sample size, keeping only unique IPs. We also approached scanning by creating a tool to establish connections to each individual IP, but even with multithreading it could not reach the speed of `zmap` **??**. In the end, we decided to include that method in the pipeline as an additional filter to make sure we have a list of IPs that we can create connections to.

For CVE-2014-2256, the Shodan search using the term `siemens port:"102"` returned 1,499 results, with the top five countries the results are from being Germany, Italy, United States, Spain, and United Kingdom, as shown in Figure 1. Our custom tool, on the other hand, discovered 2957 internet facing devices on port 102 after a few zmap scans of 10000 seconds each. Then, after grabbing banners for each IP and filtering the banner information, we see that 2500 are open and 59 are vulnerable. Compared to the Shodan search, this is not a lot of results, but shows that even Shodan is not necessarily needed to find this vulnerability in a device. A direct comparison between the number of results found for both Shodan and our custom tool, along with ratios of vulnerable devices detected to devices probed, is given in Table 1.

For CVE-2019-0708, the Shodan search for `port:"3389"` returned 4,977,976 results, but would need to be narrowed down based on the operating system. Adding the term `os:"Windows 7 or 8"` to the original search term returned 14,012 results, shown in Figure 2. Adding `os:"Windows XP"` to the original search term returned 2,174 results. Adding

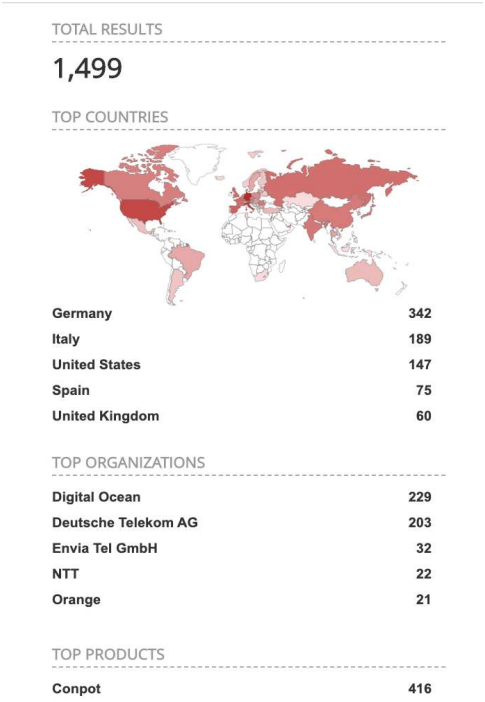

Figure 1: Shodan search results for CVE-2014-2256

	Shodan	Custom Tool
Scan on Port 102	21,843	2,500
Vulnerable Devices Detected	1,401	59
Vulnerable to Detected Ratio	0.0641	0.0236

Table 1: Comparison of Shodan results and our custom tool's results for CVE-2014-2256.

2003 to the original search term returned 35,560 results. Adding 2008 to the original search term returned 36,802 results. Our custom scanning script, on the other hand, discovered 8,796 internet facing devices on port 3389 after running a zmap [7] scan for 10000 seconds. Then, after each layer of filtering, we had 8,796 IPs that we could connect to and 732 vulnerable devices out of those. We found considerably more vulnerable devices with our custom tool for this CVE, likely due to the fact that there are just more devices with this vulnerability overall. Again, a comparison is given in tabular form in Table 2 with the vulnerability ratio listed at the bottom.

Finally, for CVE-2018-0101, the Shodan search for `"Set-Cookie: webvpn=;" ssl:"ASA Temporary"` returned 42,246results. The top five countries these results were from are United States, United Kingdom, Germany, Canada, and Italy, with the exact numbers shown in Figure 3. A more generic search for `port:"443"` gave 42,311,357 results, but includes devices used for purposes besides a VPN server. Our custom tool found 9,016IPs after scanning for 10000 seconds, but found that, out of those, 104 had "ASA Temporary" as the common name in their SSL Certificates. A side by side comparison of the number of results from each tool is shown in Table 3.

Figure 2: Shodan search results for CVE-2019-0708, where the device OS is Windows 7 or 8

	Shodan	Custom Tool
Scan on Port 3389	4,977,976	8,796
Vulnerable Devices Detected (by OS for Shodan)		
Windows 7 or 8	14,012	
Windows XP	2,174	
Windows Server 2003	35,560	
Windows Server 2008	36,802	
Total	88,548	732
Vulnerable to Detected Ratio	0.01779	0.08322

Table 2: Comparison of Shodan results and our custom tool's results for CVE-2019-0708.

	Shodan	Custom Tool
Scan on Port 443	40,311,357	9,016
Vulnerable Devices Detected	42,246	104
Vulnerable to Detected Ratio	0.00104799	0.011535

Table 3: Comparison of Shodan results and our custom tool's results for CVE-2018-0101.

5. RELATED WORK

Bodenheim [6] discusses the metrics on whether or not Shodan is being used to target industrial control system devices (ICS), and addresses the concern of whether this method and channel of attack are widely used when targetting a specific set of devices. The research done in this paper evaluates Shodan's impact on Internet-connected ICS device security by using a series of Internet-facing ICS honeypots. These honeypots were designed to be representative of ICS devices that are currently found through Shodan. Thus, allowing him to analyze network activity by measuring transmission control protocol (TCP) connections, total number of TCP packets, and the number of distinct IP addresses interacting with each honeypot. Overall, they found Shodan does not impact Internet-facing ICS device security,

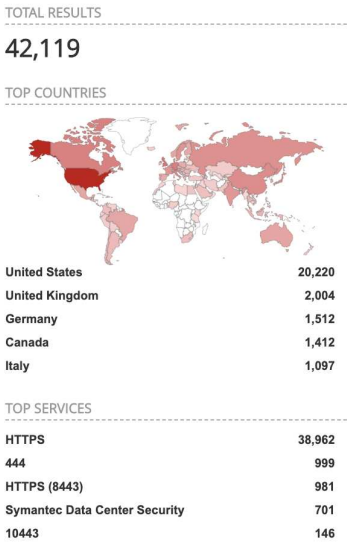

Figure 3: Shodan search results for CVE-2018-0101

but is identified as a passive reconnaissance tool.

In another paper, Bodenheim et. al.[5], analyzes Shodan's detection ability on a specific programmable logic controller (PLC) and suggests a potential solution to mitigate its visibility to Shodan. Four PLCs were configured to be Internet-connecting and deployed to evaulate Shodan's indexing and querying proficiencies. All the PLCs were exposed to two ports, Port 80 and Port 44818, while two of these were designed to have altered service banners in order to prevent Shodan query discovery. The results show that Shodan was able to successfully index and identify all the deployed PLCs within 19 days, thus showing the expansive capabilities of Shodan.

Ercolani et. al.[8] discusses how Shodan actually scans the Internet. Shodan works by scanning the Internet for open ports on IP addresses and determines runnings services on chosen ports. In order to visualize these open ports, they create visualizations that use both IP addresses and open ports as nodes. The team focuses on identifying Supervisory Control and Data Acquisition (SCADA) and Industrial Control System (ICS) devices due to the infrastructural support of these devices. These Shodan visualizations pose a potential solution in understanding what devices are running on a network.

6. CONCLUSIONS

Overall, the test results were decent, considering the bandwidth limitations that we had for scanning the internet. In fact, the ratios show that with a bit of luck, even if scanning the whole internet is not possible, vulnerable devices in the IPs probed are not too hard to come across. We did expect that there would be much less results from our custom tool compared to Shodan, which is evident when looking at the raw numbers from both port scanning and further filtering, like specifying the device manufacturer or the device OS. However, it is important to take note of the feasibility of writing a tool to detect certain vulnerabilities. For example, given a much smaller network rather than the whole internet, probing devices would not take as long and our custom tool would be much faster. In other words, detection of these vulnerabilities in one's own devices is not a big challenge and a solution should be sought after quickly for such vulnerabilities. Whether the solution is easy or difficult depends much more on the exact vulnerability and what the solution is is a whole different problem.

7. REFERENCES

[1] Shodan.
[2] Cve-2014-2256 detail, March 2014.
[3] Cve-2018-0101 detail, October 2019.
[4] Cve-2019-0708 detail, July 2019.
[5] R. Bodenheim, J. Butts, S. Dunlap, and B. Mullins. Evaluation of the ability of the shodan search engine to identify internet-facing industrial control devices. Technical report, Air Force Institute of Technology, Jan. 2014.
[6] R. C. Bodenheim. Impact of the shodan computer search engine on internet-facing industrial control system devices. Technical report, Air Force Institute of Technology, Mar. 2014.
[7] Z. Durumeric, E. Wustrow, and J. A. Halderman. Zmap: Fast internet-wide scanning and its security applications. In *Proceedings of the 22Nd USENIX Conference on Security*, SEC'13, pages 605–620, Berkeley, CA, USA, 2013. USENIX Association.
[8] V. J. Ercolani, M. W. Patton, and H. Chen. Shodan visualized. 2016.
[9] G. F. Lyon. *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*. Insecure, USA, 2009.