Article can be accessed online at http://www.publishingindia.com

Analyzing Threat Perception and Geo-Location of Internet-Facing Systems

Pawandeep Singh*

Abstract

'There is no security on this earth; there is only opportunity.'

-Douglas MacArthur.

For understanding security, it is vital to know about the perception of threat. Hence, to analyze the security of information systems, we must be aware about the exposure of different systems as well as the tools available with an attacker to attack the system.

In this paper, the research focused on gaining information about Cyber-security vulnerabilities, detecting vulnerable systems using known methods and then geo-location of systems for the visualization of the same. Three cases were studied to cover different type of systems so that it can be extrapolated from there on for generalization of the process of obtaining potentially vulnerable systems using this methodology suggested.

We were successful in obtaining the list of potentially vulnerable systems and also estimated the ability of the tools used i.e. SHODAN and IP to geo-location websites and could conclude that these tools possess the potential of reducing the resources required to exploit a system, but, are vital to aid the defenders in identifying vulnerable systems.

Keywords: Cyber-Security, Information-Security, Vulnerability, Geo-location, SHODAN

1. Introduction

The modern societies and economies are wired together by networks, cables and the IP addresses of our computers. Due to this vital role of Complex Critical Communication and Information Systems in our development, their security cannot be neglected. This has indeed affected the security environment of the 21st Century and is a matter of concern for authorities at all levels. The North Atlantic Treaty Organization (NATO) made a special Cyber Defense policy in June 2011 for the same defining the Cyber Defense Governance at different levels. The security of Information Systems is always challenged by various types of vulnerabilities which in turn lead to different kinds of exploits on the systems with such vulnerability. *Our vision is to understand the perception of threat to these systems and make deductions about their security.*

The main motivation behind this work was the Stuxnet attack on the Siemens SCADA (Supervisory Control And Data Acquisition) systems in Iran's Bushehr Nuclear Power Plant in 2009. Where the attackers possessed the Stuxnet code and desired the location and information about the potentially vulnerable systems, it is known that they succeeded eventually. This points out that the aspect of finding the potentially vulnerable systems is vital to any form of cyber attack. Further, various news bulletins pointed out that SHODAN, a system search engine, could be used to pinpoint similar Industrial Control Systems. Hence, the purpose of this paper is:

- To determine how much of information is available about Internet Facing systems and whether that information can be used to attack/exploit the systems.
- To verify the ability and precision of tools like SHODAN that are legally available to the users and identify what extent of threat they might possess in aiding a novice attacker.

We begin by the preliminary concepts about Cyber Security and some technical terms involved in this report.

* Student, ISERC, Visva-Bharati University, Santiniketan, West Bengal, India. e-mail: pawan.s@live.co.uk

Some preliminary concepts involved in this research are the following:

I. Vulnerability: The term 'vulnerability' in the area of Information-Security is defined as a shortcoming in the software that can directly be used by the hacker to exploit the system.

It may affect the Integrity, Confidentiality or Availability of the system. *(CVE terminology, n.d.r)*

Vulnerability of a system can be exploited by:

a. An insider who has certain access over the system. It might involve physical access or network access.

b. An outsider who has no access over the system but has Internet Access as the system into consideration.

Both types of attackers can prove fatal but the former issue, i.e., the attack/exploit by an insider rests less on the technology of the attack and more on the social engineering aspects which would involve the motivation behind the attack and the limitation of system access to certain users (potential attackers) depending on the level of trust of the Administrator. The latter issue, however, involves the analysis of the vulnerability of the system and programming skill of the attacker to make such an attempt. Hence, the report would focus on the latter issue.

So, we would consider the vulnerabilities of systems connected to the internet and analyze the means available to a remote attacker to exploit them.

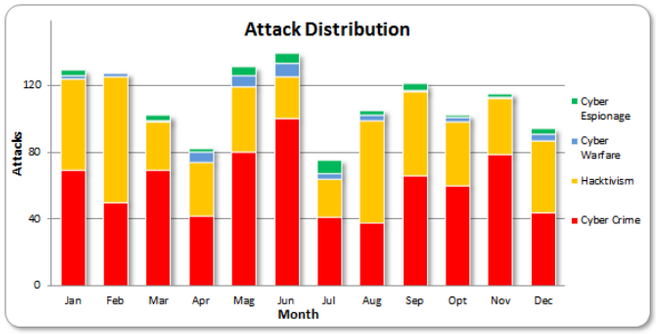
II. Attack Vector: An attack vector is a path or means by which a hacker (or cracker) can gain access to a computer or network server in order to deliver a payload or malicious outcome. (Rouse, 2004) Depending on the vulnerabilities the access vector may be local system access, network access, or adjacent network access.

An example of physical access as the attack vector for an exploit is Magic Lantern, one of the first malwares that was created by FBI*(Sullivan, 2001)*, which was keystroke recording software that needed to be installed and run on the device.

Hence, for the purpose of this report, our concern is on vulnerabilities with attack vector to be remote access i.e., the vulnerability can be exploited via the internet.

III. Vulnerability Statistics: We would now move on to the analysis of some past statistics available on the internet for Cyber-Attacks. Firstly, we notice the cyber-attack distribution of 2012 to obtain the different types of attacks along with their frequency. This is illustrated in Figure 1.

Figure 1: Attack Distribution-2012 (Hackmageddon, 2012)



The terminology used in Figure 1:

1. Cyber Espionage – It involves the unauthorized probing to test a target computer's configuration or evaluate its system defenses, or the unauthorized viewing and copying of data files. (Wilson, 2008)
2. Cyber Warfare – Actions by a nation-state to penetrate another nation's computers or networks for the purposes of causing damage or disruption (Clarke, 2010)
3. Hacktivism – The use of computers and computer networks to promote political ends. (Krapp, 2005)
4. Cyber crime – Offences that are committed against individuals or groups of individuals with a criminal motive to intentionally harm the reputation of the victim or cause physical or mental harm to the victim directly or indirectly, using modern telecommunication networks such as Internet (Chat rooms, emails, notice boards and groups) and mobile phones. (Halder & Jaishanker, 2011)

Different types of attacks as shown above require different types of Attack Techniques which in turn require certain programming skills to be possessed by the attacker. The more skilled the attacker, the higher is his programming skill which in turn would lead to more sophisticated attacks.

A figure describing the Distribution of Attack Techniques in May 2013 is given in Figure 2.

Figure 2. Attack Techniques May 2013 (Hackmageddon, 2013a)

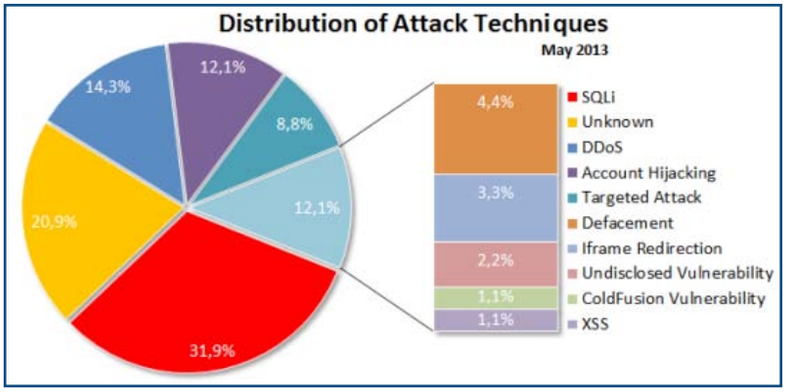


Figure 3. Target Distribution- May 2013 (Hackmageddon, 2013b)

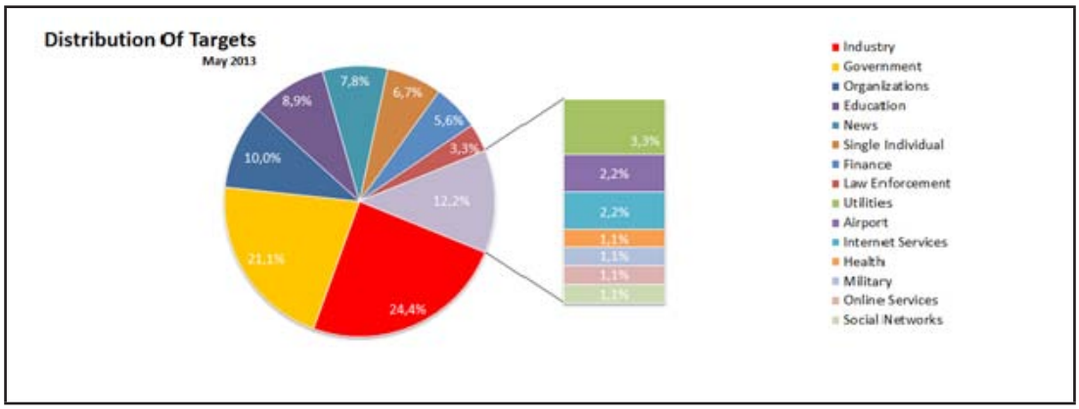


Figure 2 shows the different attack techniques involve exploitation of different vulnerabilities. The most common attack for the month was SQL injection attack at 31.9%. A large number of attacks i.e. 20.9%, had unknown techniques used in them. This shows that there is a lot of innovation being done by the attackers for cyber-attacks hence there is a need for defenders to stay aware and updated about new techniques.

Another distribution of recorded data for May 2013 is on the Targets attacked, as given in Figure 3.

The above data in Figure 3 shows that most of the attacks took place on Industrial and Governmental systems where maximum gain for the attacker was possible. Hence a clear motive of almost every cyber attack is gaining valuable information or to exploit a system in such a way that can be translated into financial gains.

This distribution motivates us to study the plausible attacks or the attack surfaces for Industrial Control Systems in detail. We would consider the example of systems vulnerable to Stuxnet attack i.e. Siemens' SCADA systems.

Also, the combined statistics show the typical characteristic of any cyber (or cryptanalytic) attack that the amount of resource used in the attack must be less than the amount of gain through it.

Hence, if the resource used in a part of the attack is somehow reduced, then it would imply the possibility of increase in sophistication of the attack which may lead to exploit even more secure systems.

Attacker's scenario: To comment about the existing security of information systems, we must first have the knowledge about an average attacker and his methodology of attack. It is considered for this report that the attacker has an average programming skill. Using it, he may proceed in two different processes:

1. He may plan to attack on a desired organization. For this, he should first be aware about a subset of systems on which his attack would be feasible and obtain their IP addresses. He could get the IP address of the server that is hosting the organization website through the popular DOS command 'nslookup'. After obtaining the IP Addresses, he could obtain the system's information through SHODAN, a unique search engine. If he is lucky enough to obtain a vulnerable component in the system which he can exploit using his programming skill then he is through with his desired organization attack.
2. He may plan an attack by analyzing his programming skill and obtain the information about the vulnerabilities and the methods to exploit them using Exploit-db and then search for vulnerable systems in SHODAN using specific filters and entering required keywords. On obtaining the search results for potentially vulnerable systems he might proceed with the attack on a 'favorable' system. Favorable here indicating the system which could provide him the maximum gain.

The former case involves a targeted attack where the attacker has focused attention on the particular organization. The latter case however, involves building over relatively less information. Hence we focus on the latter case for this report. This would give us the maximum amount of information available through different tools on the internet. Also, proceeding by the latter case we can get an estimate of the real situation in cyber security and the ease of the attacker to get the information of potentially vulnerable systems.

Hence, our research focuses on gaining information about Cyber-security vulnerabilities, detecting vulnerable systems using known methods and then geo-location of systems for the visualization of the same.

2. Methodology

The methodology incorporated for the analysis of attack surfaces to Internet facing systems was by starting with some information security vulnerability and sequentially, obtaining means to exploit it, searching for potentially vulnerable systems and for presentation purposes, obtaining their geo-locations and plotting them on the Google map.

This was accomplished by creating a python based tool using SHODAN's WebAPI, IP Address Labs API and pygmaps, the Google Maps' python extension. The python code is also added in the APPENDIX section of the report. This program works as follows:-

- A search query is entered containing all the keywords/SHODAN search filters required.
- Then the program searches SHODAN using WebAPI which returns various IP addresses of matching internet connected systems.
- Next, it uses IP Address Labs API for obtaining geographical coordinates of the given IP address
- Finally, it plots the geo-locations on the Google Map which pops up after all the results are obtained.
- Also, a time comparison mode has been incorporated for detailed results color-coding whether the system is consistent, has been newly added or is no longer present in the current results with respect to the reference file of IP addresses obtained previously.
- It also gives an option for the user to save the file of IP Addresses which might be used as a reference file later on.

The detailed steps involved in the research methodology are as follows:

3. Vulnerability/Exploit Search

Our first step was to select an exploitable vulnerability, say Ψ, and explore the type of systems affected and means to exploit it. We were able to get both of these by the freely accessible Vulnerability Databases on the internet like Exploit-db, CVE-details, Metaspoilt, etc. *Exploit-db* was used for this research because of its compatibility with SHODAN. And further, CVE Details and OSVDB (Open Source Vulnerability Data Base) were also consulted for other details about vulnerabilities such as their CVSS scores.

Exploit-db provides the user with the remotely exploitable vulnerabilities, the list of systems that they affect and the required operating environment for the exploit as well as the code and/or the payload associated with the vulnerability freely available to download. In view of a potential attacker, it is a head-start due to availability of the information about the vulnerability as well as the materials required for a successful exploit. This

Snapshot 1. (.xml) Generated by IP Address Labs

services.ipaddresslabs.com/iplocation/locateip?key=demo&ip=216.131.72.76

This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<response>
  ▼<query_status>
     <query_status_code>OK</query_status_code>
     <query_status_description>Query successfully performed.</query_status_description>
   </query_status>
   <ip_address>216.131.72.76</ip_address>
  ▼<geolocation_data>
     <continent_code>NA</continent_code>
     <continent_name>North America</continent_name>
     <country_code_iso3166alpha2>US</country_code_iso3166alpha2>
     <country_code_iso3166alpha3>USA</country_code_iso3166alpha3>
     <country_code_iso3166numeric>840</country_code_iso3166numeric>
     <country_code_fips10-4>US</country_code_fips10-4>
     <country_name>United States</country_name>
     <region_code>CA</region_code>
     <region_name>California</region_name>
     <city>San Francisco</city>
     <postal_code>-</postal_code>
     <metro_code>807</metro_code>
     <area_code>415</area_code>
     <latitude>37.7749</latitude>
     <longitude>-122.4194</longitude>
     <isp>Reliablehosting.com</isp>
     <organization>Black Oak Computers Inc - San Francisco</organization>
   </geolocation_data>
</response>
```


information gives us what kind of systems to search for. Also, based on other details such as the Operating System required for the exploit, we can incorporate required search filters for the same.

Hence we have all the parameters required for a system to be vulnerable to Ψ and we have to find maximum possible systems of this type.

4. Search for Systems

Now we search for exploitable systems using our program. For an optimal search containing the maximum number of vulnerable systems and containing only the systems of the required attributes (i.e. avoiding false positives), we must have the right search keyword. We can get this by trying the words obtained by the Exploit-db result or by iterating searches in SHODAN to modify the keywords according to the results obtained. We can further use SHODAN's filters like country, city, os, etc. to get the optimal results.

Once the required keyword is acquired and searched for along with the respective filters, the program would obtain the results from SHODAN which are systems that are potentially vulnerable to Ψ. As explained in the beginning of this section, we used our python program and searched SHODAN to acquire the IP Addresses of the potentially variable systems.

5. Shodan's Results

More information about the vulnerable systems was obtained through the results given out by the program. The results returned by SHODAN apart from IP Addresses are in the form of banners. A sample banner is given below:

74.41.163.238

Allen-Bradley 1747-L541E SLC-5/05 Series A Revision 3 1747enet 1.27

The above banner indicates that the IP Address is of an Allen Bradley device the series version and revision are mentioned in the banner. Using information available in the banners we can verify whether the search returns valid returns or not.

6. Geo-Location and Google Maps

After obtaining the IP Addresses, the program passed the IP Address to IP Address labs and returned the geographical coordinates corresponding to each IP Address to get an idea about the spread of the systems. Every geo-location providing tool available on the internet provides country precision in geo-location. Further, a slight higher city precision was observed in www.ipinfodb.com but it did not return results in geographical coordinates, instead it gave city names. Hence, IP address labs was used to translate IP addresses to geographical coordinates. Government agencies are much more equipped in tracking location of systems, some techniques can even track the system to the particular hotel room from which internet is accessed. But tools open to the pubic give very little detail about system location via IP Addresses.

IP address labs generates an .xml file for every query and hence makes it simple for the program to extract whatever information is required for the user.

A sample of the .xml file generated for a query is displayed in Snapshot 1:

The information about the ISP as well as the using organization can be obtained if available. Unfortunately, this information is not completely dynamic but involves the fields available with the ISP. There might be some inaccuracy associated with it.

But for visualization and presentation purposes, this information is sufficient enough.

And, for the conveying of this information in a presentable manner, a Google Map was generated by plotting of the geographical coordinates obtained using pygmaps, the python Google map generating tool. This improved the visualization aspect of the data obtained.

7. Comparison Over Time

Another feature is incorporated in the program by which a comparison of the systems over time can be made. This includes comparing the results obtained by a reference file of the search results previously stored in the folder or generation of a reference file if not.

The results hence could be color-coded in terms of consistent systems, new systems and removed systems.

Map 1. Program Run in Individual Mode, keyword Entered was 'CIMPLICITY'

Map 2. Program run in Comparison mode, Keyword Entered was 'Modbus+Bridge' with some Manipulation in Reference File for Proper Demonstration of Comparison Mode.

8. Sample Outputs

The Google maps obtained by the above methodology are like the following:

a. In Individual Mode (Map 1):

b. In Comparison Mode (Map 2):

9. Limitations Encountered and Overcome

i. There are search limitations for free license in SHODAN.

The free license was used for this demonstration in SHODAN but the IP Addresses obtained were at most 100 per search keyword in Web API and the filters allowed were only os, hostname, port and net.

The search filters were sufficient for our study and a maximum of 100 results showed us a pattern about the spread of the system. But SHODAN's limitations can be easily overcome if the user (or attacker) has a paid license.

ii. Limitations in determining Geo-Location

The process by which the websites translate IP Addresses to geo-location involves returning details filled by the buyer of the IP Address. These are not very precise in general. Also, sometimes it returns the information about the ISP rather than the actual user. Few Internet Service Providers (ISPs) however offer internet access to several countries through satellites. And geo-location is not possible in such cases. So, we plot these at (latitude, longitude) = (0, 0) which is off-coast Africa with a different color so that they can be identified separately.

Note that the above limitations are not inherent in the research methodology but are just resource limitations. More number of IP Addresses can be obtained from Shodan by using a paid license and city-precise geo-location does not affect the presentation much. Instead, it avoids cluttering on the map.

10. Results

We consider three cases for our searches to cover different type of systems so that it can be extrapolated from there on for generalization of the process of obtaining potentially vulnerable systems using this methodology suggested.

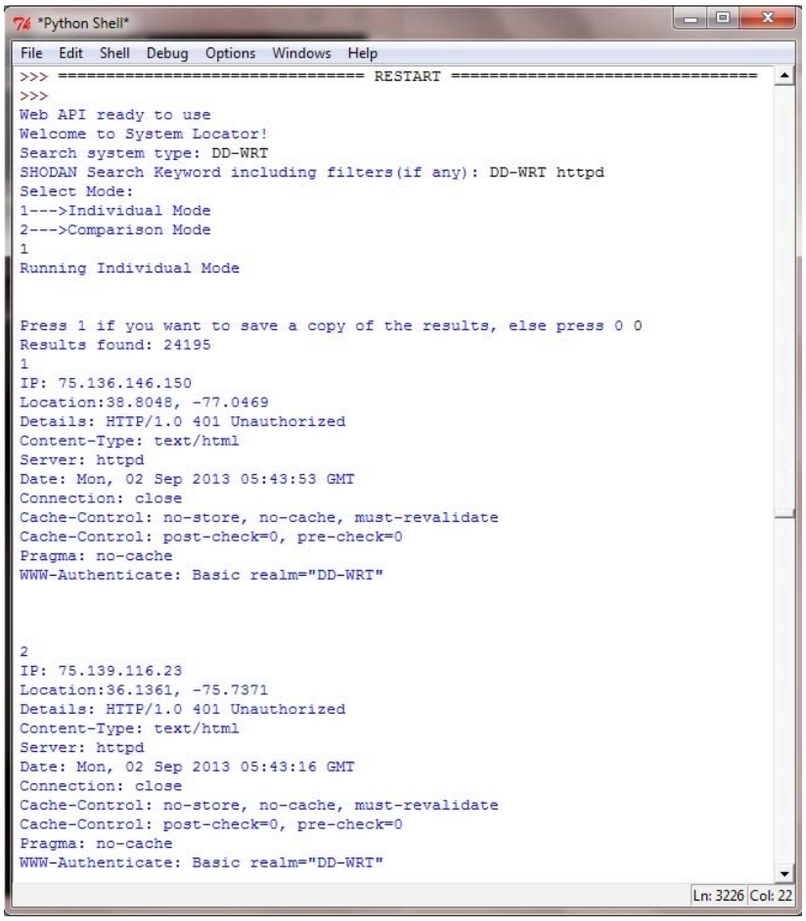
10.1. (Case I) DD-WRT Router Firmware

The DD-WRT router firmware (Linux-based) is used with various router making companies such as D-Link, Buffalo and Linksys among others with many of their

Snapshot 2. Exploit Results obtained in System Search Program

```
Exploit Results found: 5
15842: DD-WRT Information Disclosure Vulnerability

16856: DD-WRT HTTP Daemon Arbitrary Command Execution

9209: DD-WRT (httpd service) Remote Command Execution Vulnerability

10030: DD-WRT HTTP  v24-SP1 Command Injection Vulnerability

7389: DD-WRT v24-sp1 (XSRF) Cross Site Reference Forgery Exploit

Modules found: 1
exploit : DD-WRT HTTP Daemon Arbitrary Command Execution
enter any no. to exit
```

Snapshot 3. Results for DD-WRT httpd



router versions. There are 5 Exploit-db exploits found in DD-WRT and 1 Metaspoilt Module too, hence the attacker can easily download the code and payload (if required) to exploit these vulnerabilities using Exploit-db or Metaspoilt. Snapshot 2 is a screen-shot of the corresponding Exploit Results from the program:-

Out of the above stated Vulnerabilities, the Remote Command Execution Vulnerability (Exploit-db ID 9209) (CVSS Score – 6.8) was selected for further analysis.

The search keyword used for DD-WRT router firmware was 'DD-WRT httpd' and the following results were obtained:

Map 3. Google Map corresponding to 'DD-WRT httpd'

The total number of connections found out was 24,195.

Out of these, 100 IP Addresses and their corresponding geo-location was returned by the program.

Snapshot 3 shows the Program's returned results:-

Map 3 is the Google Map output for the above mentioned search:

Case II) Nginx Servers

In this case, we look into Servers and their vulnerabilities. A common vendor with vulnerabilities in Servers is Nginx. nginx (pronounced "engine x") is an open source web server and a reverse proxy server for HTTP, SMTP, POP3, and IMAP protocols.

We first obtain the exploit results for 'nginx' and get the following:

Exploit Results found: 6

9829: nginx 0.7.61 WebDAV directory traversal

12804: nginx [engine x] http server <= 0.6.36 Path Traversal

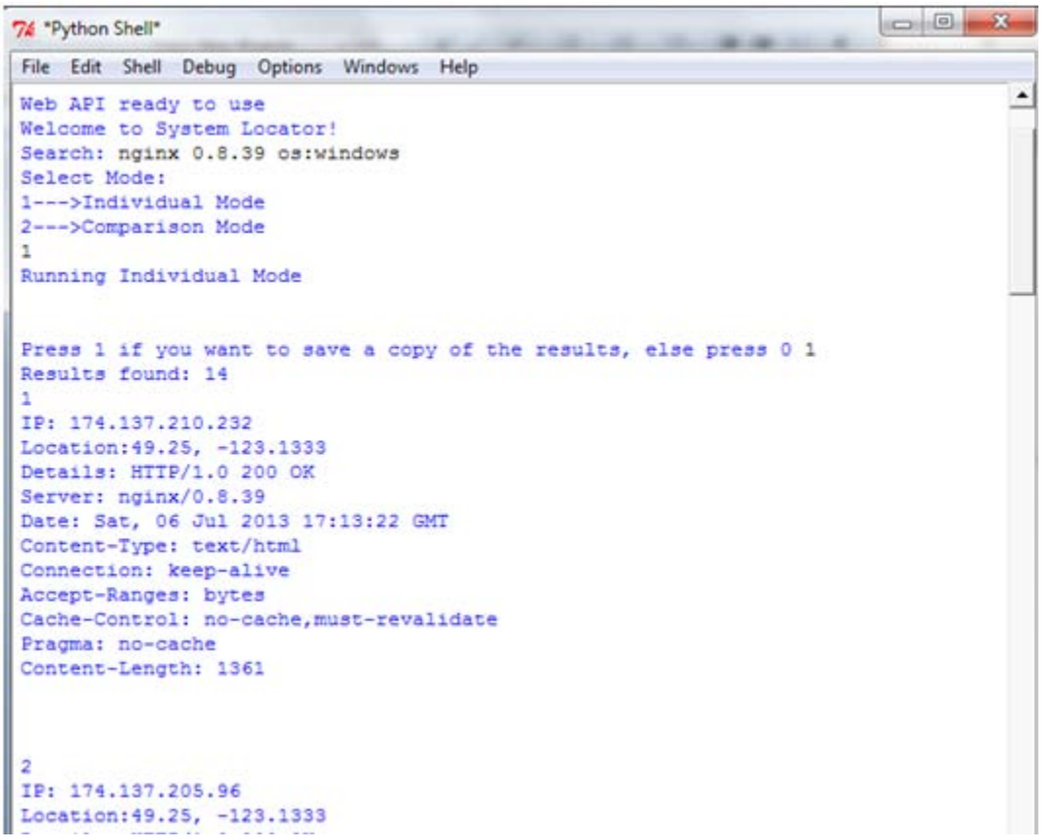
13818: Nginx 0.8.36 Source Disclosure and DoS Vulnerabilities

14830: nginx v0.6.38 Heap Corruption Exploit

13822: Nginx <= 0.7.65 / 0.8.39 (dev) Source Disclosure / Download Vulnerability

Snapshot 4. Exploit-db page for Exploit ID: 13822

Snapshot 5. Results for 'nginx 0.8.39 os:windows



9901: nginx 0.7.0-0.7.61, 0.6.0-0.6.38, 0.5.0-0.5.37, 0.4.0-0.4.14 PoC

As an example, we consider the vulnerability with Expoit ID 13822. Snapshot 4 is a screen-shot of its Exploit-db page:

Snapshot 4 shows that both the Exploit code and Vulnerable App is available for download. Also, Exploit-db page mentions that this vulnerability is present in windows systems running the server application.

Now, from the Exploit definition we know that all the versions before 0.8.39 running on windows operating system are vulnerable to this exploit. So, the filter os:windows is added to the search keyword 'nginx 0.8.39'

Hence we search SHODAN for Nginx 0.8.39 os:windows and obtain the results as shown in Snapshot 5:

Hence we get 14 potentially vulnerable systems for nginx 0.8.39 on windows.

Map 5 is the Google Map that was obtained for this Search Query [Nginx 0.8.39 os: windows].

Case III) Industrial Control Systems

The previous cases are dependent on the type of vulnerabilities that the attacker comes across. But, this case is included as the common observation that Industrial Control Systems are some of the most sought after systems for attack by attackers (Figure 3, Target Distribution). Another motivation for the inclusion of ICS in this case was a previously published MPhil dissertation of the University of Cambridge by Mr. Eireann P. Leverett emphasizing the importance of security of Industrial Control Systems.

Historic events like the Stuxnet attack on the Siemens' SCADA systems in Iran's Nuclear Facility on June 2010 have shown the need for high level of protection for these systems.

In this section, we would search for Internet facing Seimens' SCADA systems taking the case where an attacker might possess a deadly malware (similar to Stuxnet) that he wishes to plant on a Siemens' SCADA system. We would only try to locate such systems for considering this scenario.

Map 5. Google Map for 'Nginx 0.8.39 os:windows'

Map 5: Google Map for 'Nginx 0.8.39 os:windows'

Note. Place marks close to each other on the obtained Google Maps can be distinguished by zooming into the figure if the returned geo-location corresponds to different geographical coordinates.

Here, in the obtained Google map for Windows running Nginx 0.8.39 servers, some place marks appeared to be overlapping but they were present in different cities altogether when zoomed upon.

The search keyword for this case was obtained by simple internet searches using the knowledge that the Siemens' SCADA systems with the operating environment WinCC and PCS-7 are vulnerable to Stuxnet Attack. A search for 'wincc' and 'PCS-7' in Shodan's Exploits gave us the terms 'Simatic HMI' and 'Simatic S7' which were helpful to locate these devices.

After obtaining the search keywords as the above method suggests:

We search for Stuxnet vulnerable systems using the keywords:

1) Simatic+HMI
2) Simatic+NET
3) Simatic+S7

11. Simatic+HMI

These are Human Machine Interface systems by Siemens which are vulnerable to Stuxnet Attacks.

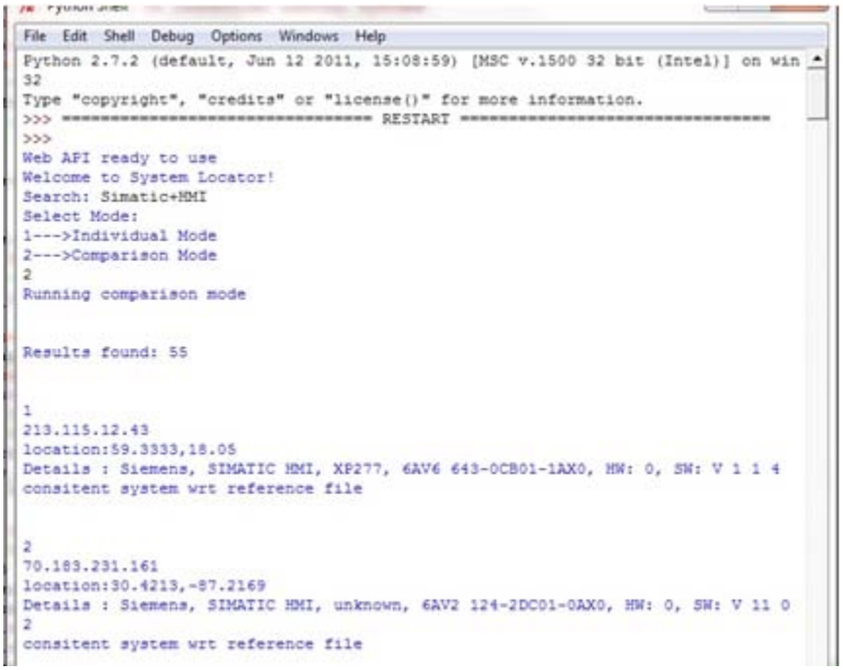
There were 55 results obtained for this case.

Snapshot 6 is the screen shot for the obtained systems:

Also, since the results were only 55. We get a much clearer picture of the geographical spread of the systems in the Google Map obtained.

Map 6 is the Google Map obtained for these Simatic Systems:

Snapshot 6. Results for 'Simatic+HMI'



Map 6. Google Map corresponding to 'Simatic+HMI'

Note: Map 6 is color coded for the comparison mode hence the color of the system place-marks is green as they are all consistent with the reference file for the search query.

1. Simatic+NET

These are again Simatic Systems which are, or at some point of time were, vulnerable to stuxnet attack.

The total results obtained for these systems are 107.

Snapshot 7 is a screen-shot of the results obtained:

Map 7 is the Google Map generated from this program for keyword 'Simatic+NET'.

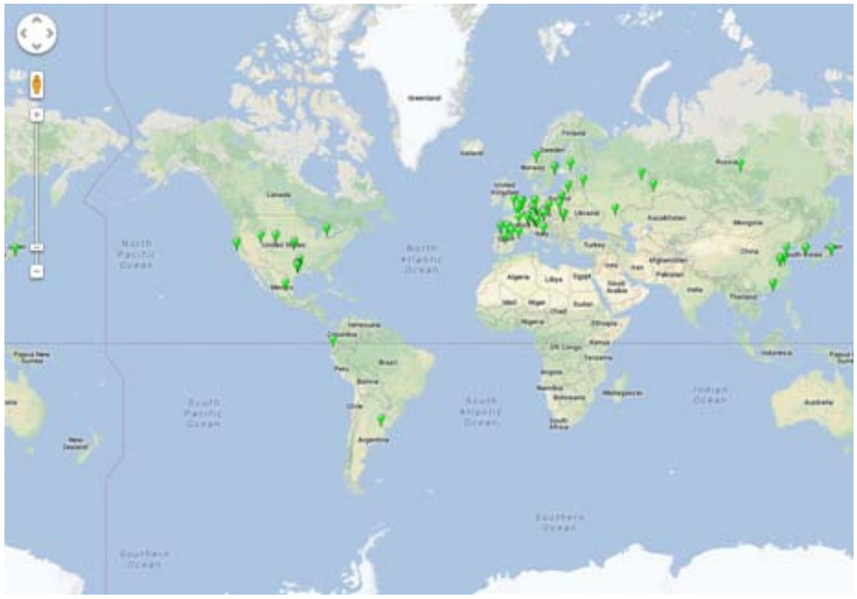
2. Simatic+S7

Snapshot 7. Results for 'Simatic+NET'

```
File  Edit  Shell  Debug  Options  Windows  Help
Welcome to System Locator!
Search: Simatic+NET
Select Mode:
1--->Individual Mode
2--->Comparison Mode
2
Running comparison mode


Results found: 107


1
166.130.49.201
location:40.2969,-111.6946
Details : Siemens, SIMATIC NET, CP343-1, 6GK7 343-1CX10-0XE0, HW: Version 2, FW:
 Version V2.0.16, VPV2517056
consitent system wrt reference file


2
89.26.19.201
location:47.8,13.0333
Details : Siemens, SIMATIC NET, CP 343-1 Lean, 6GK7 343-1CX10-0XE0, HW: Version
6, FW: Version V2.6.0, VPC8545855
consitent system wrt reference file
```

Notice the peculiar banners in Snapshot 7 returned for Simatic ICS systems.

Map 7. Google Map corresponding to 'Simatic+NET'



Again the program was run in 'Comparison Mode' marking consistent systems green in Map 7.

This keyword denotes the Siemens' PCS-7 systems which are also vulnerable to a Stuxnet attack and contain other vulnerabilities as well.

111 systems were found with this search keyword and the screen-shot for the results is displayed as Snapshot 8.

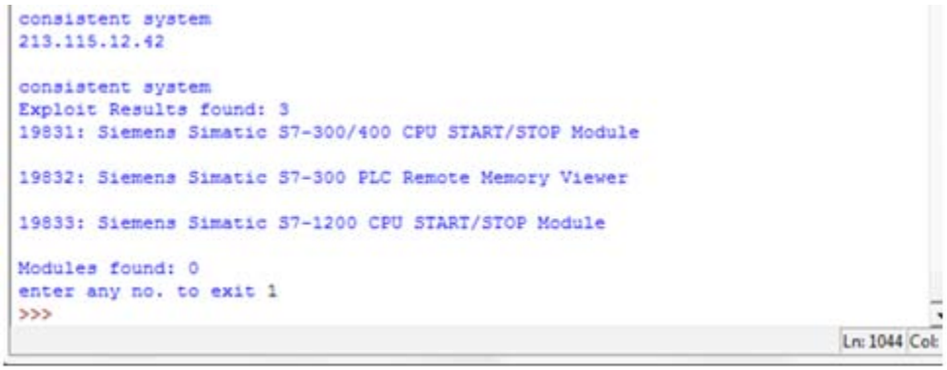
The Google Map obtained by plotting these systems is given as Map 8.

One system is plotted at (0, 0) this is because the geo-location for this IP was not determined by IP Address Labs which was because the corresponding ISP offers internet access to several countries through satellites.

Snapshot 8. Results for 'Simatic+S7'

The systems display a unique banner message and hence are easy to search for.

Snapshot 9. Further exploit results for Siemens' PCS-7 or 'Simatic+S7'



Snapshot 9 illustrates that other exploit results are also displayed in the results.

12. Discussions

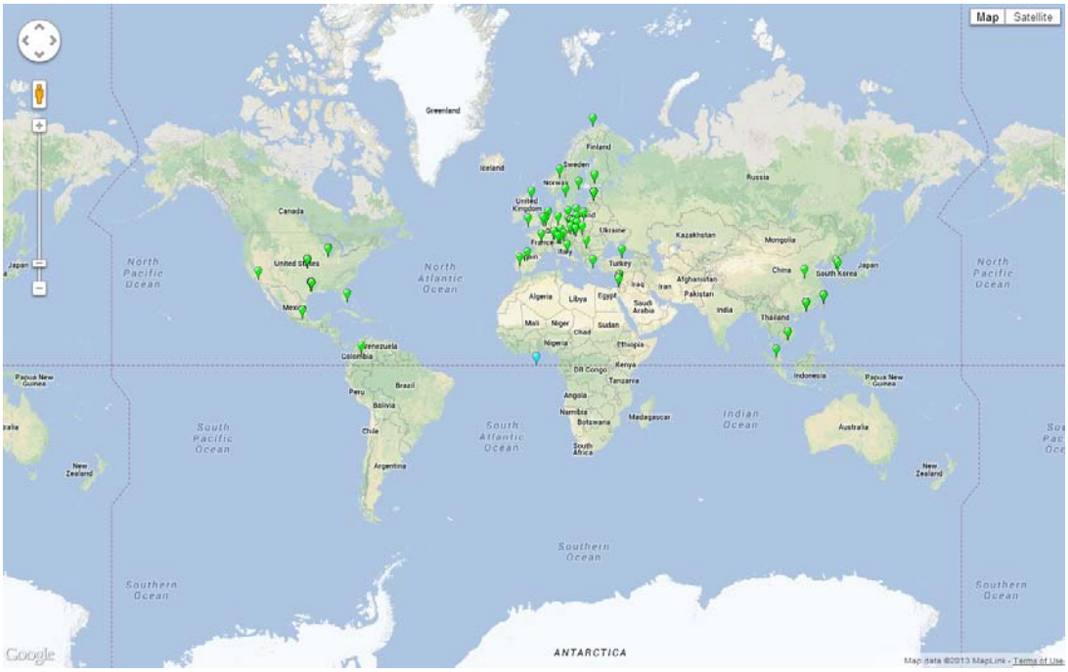
This section of the paper analyses the results obtained by the System Search Program to further make deductions about the present Internet Security strategy and cyber safety protocols across the Globe. Clearly, the results demonstrate that a considerable amount of information about an internet facing system is available on using the tools previously described. The information includes: the hardware components, the operating environment and the location. Using this information, vulnerable systems may be targeted.

Through this methodology, a search for different types of systems can be made. Though illustrations were for three cases, the method can be extrapolated to different system types and several vulnerabilities can be focused on.

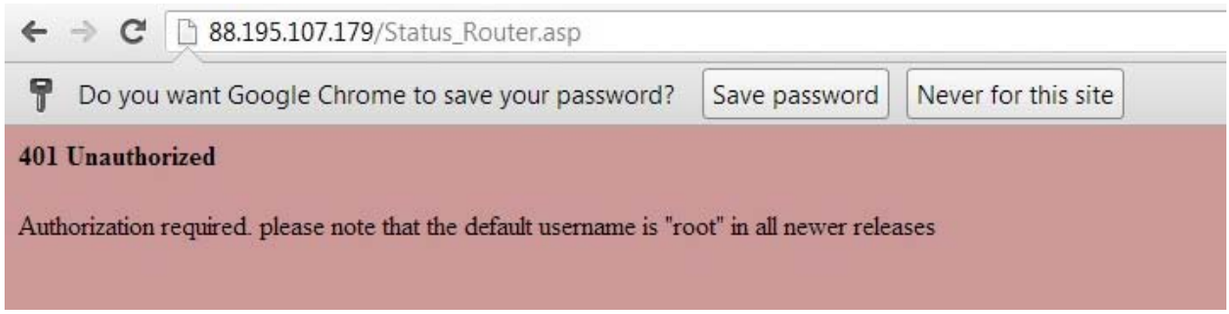
1. DD-WRT Router Firmware

DD-WRT on httpd servers exhibits a high risk vulnerability and also a large number of systems (24,195) are present. Though the vulnerability can be fixed using a patch, one of the main problems with these systems is that remote GUI is enabled in them by default and some users do not change the default usernames and passwords and this can be exploited by a remote attacker to manipulate the settings of the device. We were able to gain access to one such system using the default credentials. This can be seen in Snapshot 10 and 11.

Map 8. Google Map corresponding to 'Simatic+S7'



Snapshot 10. Accessing a DD-WRT router over the internet.



As shown in snapshot 10, the default username, *i.e. root*, was also provided by the Router's remote GUI and obtaining the corresponding password *("admin")* is merely a Google search away.

Snapshot 11 shows how all the Router's settings are available for any alteration and the connected devices are also exposed along with their MAC Addresses. This was the result of a curious exercise for checking the vulnerability of the device and no changes were made in the router settings. The IP Address has been hidden for the security of the device and we hope the security settings of the same are altered.

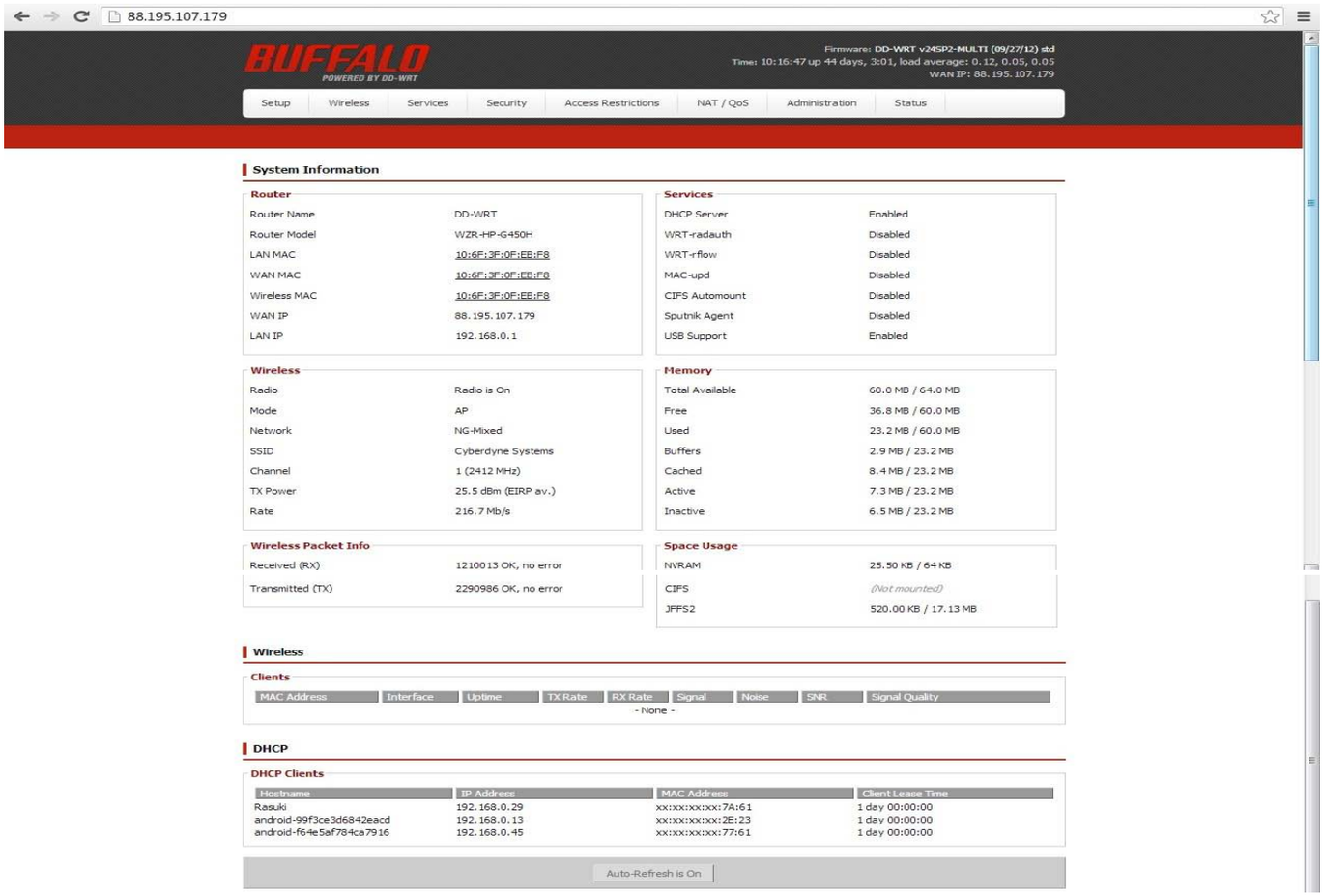
The temporary fix offered by DD-WRT on this vulnerability was not found to be sufficiently effective. The DD-WRT website reported:

'Note: The exploit can only be used directly from outside your network over the internet if you have enabled remote Web GUI management in the Administration tab. As immediate action please disable the remote Web GUI management. But that limitation could be easily overridden by a Cross-Site Request Forgery (CSFR) where a malicious website could inject the exploit from inside the browser.' (*DD-WRT* website) The permanent fixes and upgrades were made available later.

2. Nginx Servers

The easy detection of servers through SHODAN is because the results return a banner for server type. Snapshot 5 in Results section shows how banners with server information are provided in results.

Snapshot 11. Authorized access to the DD-WRT router using the default password ('admin').



The highlighted area in Snapshot 5 illustrates how searching for a particular kind and/or version of a server is convenient.

Though Apache servers were our first preference but a large number of different versions of Apache servers were obtained and version precision in searches was not obtained. Microsoft's IIS Servers were easy to detect but vulnerabilities in IIS can be rectified through patches and hence vulnerable systems could not be separated out. Hence, Nginx servers were chosen and the required results were obtained.

The Nginx Servers running Windows OS were easily detected using SHODAN's OS filter but very few results (14 systems) were obtained. The information available for removal of this vulnerability on the official site was through up gradation of the servers to version 0.8.40 onwards. But still there were systems found running the vulnerable versions.

Hence users should be aware about the updates and patches available for their products.

An interesting fact observed in the Nginx Servers was the precision of the searched versions. Our search for the version 0.8.39 gave us all the required results and no false positives were returned. The removal of the source disclosure vulnerability (Exploit ID: 13822) requires an update of the version of servers which is available for free. This is not the case for the obtained systems.

3. Industrial Control Systems

The Stuxnet Affected Siemens' Systems were searched and separated because of their distinct search keywords which were obtained as mentioned in the Methodology section. All of these systems are expected to be fixed especially because of the attention that Stuxnet grabbed by the attack on Iran's nuclear facilities.

The methodology can be generalized for any form of Industrial Control Systems. And due to the high risk

that these systems face, it is always advisable not to connect them to the internet, but the results obtained in the previous section indicate that not many users are implementing this. This may be because these systems are often maintained via the internet from distant areas.

Appendix 2 further provides search results for various known ICS compared with previously obtained results (Leverett, 2011).

Industrial systems are attacked most frequently and therefore they need a higher level of protection. Also, the vendor's response to vulnerabilities is a crucial factor and Siemens' response on Stuxnet has been a roller coaster ride, consisting of a blame game followed by a dedicated website about the issue *(Peterson, 2010)*. Though there are anti-viruses and other malware detection software for detection and removal of Stuxnet, another malware could attack Siemens' SCADA or other ICS.

13. Conclusion

The scenario we have observed so far clearly shows the extent of information available about a system by these tools and how potentially exploitable systems can be obtained corresponding to a given vulnerability. Does it imply that these tools themselves pose a threat to security of Information systems? Why are these tools legal and available for everyone to use?

As far as vulnerability databases are concerned, they are vital to the network defenders. Whenever vulnerability is detected in a system, if it is detected by a vulnerability database or a Cyber-Security Researcher or a concerned user, the product vendor is the 1st party to be notified along with agencies like CERT. For the vendor to issue temporary and subsequently permanent fixes. Only after these releases, it is posted in a vulnerability database. And adept attackers do not use these databases. Instead, they innovate on attack techniques.

SHODAN has been a controversial topic because it is the first search engine of its kind and has shown potential in scanning the internet. Many press releases claim that it considerably reduces the resources required for an attack and hence poses threat to valuable systems. But other illegal tools like botnets are present for an attacker to scan the internet, which also provide him with anonymity. There arises a concern that the ease in availability of such tools might initiate the script-kiddies to try out some exploits. However, there exist other complexities involved in making an exploit. A further exercise would be setting up the apparatus to attack one system from the other to analyze what are the crucial points for a real exploit.

Further, it is noteworthy that global deductions about the search results, is not possible as the search language is only English whereas the banners may contain other languages as well. Nevertheless, regional filtering allows us to make some deductions about systems, at the regional level. This does make countries like China and Japan less exposed to SHODAN. But similar tools to SHODAN can be developed to overcome this. However, strategically this might be an advantage to them. Also, we are yet unsure about the extent of systems visible to SHODAN. There might be systems of the required type behind a secure firewall whose TCP/IP banners cannot be retrieved by SHODAN. Hence comparison of the number of systems corresponding to different vulnerabilities is not very accurate; but a rough estimate obtained may reflect the entire system population.

Extension of this work can be done in terms of other information provided by IP to geo-location convertors based on ISP's and geo-location precision corresponding to different countries. This can be done by taking the IP addresses of the hosting servers of the top sites used by these countries and tracking their geo-location and the extent to which geo-location precision is attained. Further, comparison of different geo-location tools and analysis of the ambiguities obtained may provide different ISP policies of different countries giving the country's cyber security situation.

References

[1] Clarke, R. A. (2010). *Cyber War: The Next Threat to National Security and What to do about it*. New York: HarperCollins Publishers.

[2] Common Vulnerabilities and Exposures (CVE) Terminology. (2014). Retrieved from http://cve.mitre.org/about/terminology.html (accessed on January 1, 2014).

[3] DD-WRT remote command vulnerability page. (2014). Retrieved from http://www.dd-wrt.com/site/content/dd-wrt-httpd-vulnerability-milw0rmcom-report (accessed on January 1, 2014)

[4] Hackmageddon. (2012). Cyber Attack Statistics. Retrieved from http://hackmageddon.com/2012-

cyber-attacks-statistics-master-index/ (accessed on July 20, 2013)

[5] http://paulsparrows.files.wordpress.com/2012/06/2012 -attack-distribution.png]

[6] Hackmageddon. (2013). Cyber Attack Statistics. Retrieved from http://hackmageddon.com/2013/06/09/may-2013-cyber-attacks-statistics/ (accessed July 20, 2013)

[7] http://paulsparrows.files.wordpress.com/2013/06/attacks-may-2013.png]

[8] http://paulsparrows.files.wordpress.com/2013/06/targets-may-2013.png]

[9] Halder, D. & Jaishanker, K. (2011). *Cyber Crime and the Victimization of Women: Laws, Rights, and Regulations*. Hershey, PA, USA: *IGI Global.*

[10] Krapp, P. (2005). *Terror and Play, or What Was Hacktivism*? Grey Room MIT Press.

[11] Leverett, E. P. (2011). MPhil in Advanced Computer Science Dissertation, University of Cambridge. Retrieved from www.cl.cam.ac.uk/~fms27/papers/2011-Leverett-industrial.pdf

[12] NATO and Cyber Defense. (2014). Retrieved from http://www.nato.int/cps/en/natolive/topics_78170.htm? (accessed on January 1, 2014).

[13] Peterson, D. (2010). Siemens' Roller Coaster Response to Stuxnet": Digital Bond Blog. Retrieved from http://www.digitalbond.com/blog/2010/08/12/siemens-roller-coaster-response-to-stuxnet/ (accessed on January 1, 2014)

[14] Rouse, M. (2012). Definition Attack Vector. Retrieved from http://searchsecurity.techtarget.com/definition/attack-vector (accessed on January 6, 2014)

[15] Sullivan, B. (2001). FBI Software Cracks Encryption Wall. MSNBC. Retrieved from http://www.nbcnews.com/id/3341694/ns/technology_and_science-security/t/fbi-software-cracks-encryption-wall/#.Usp7SfQW2So (accessed on January 6, 2014)

[16] Wilson, C. (2008). Congresional Research Service, RL32114, Botnets, Cybercrime, and Cyber Terrorism: Vulnerabilities and Policy Issues for Congress 12 (2008). Retrieved from www.fas.org/sgp/crs/terror/RL32114.pdf