Brainwave: A Multidisciplinary Journal (ISSN: 2582-659X), Vol. 2, Special Issue No. 1, May 2021, pp. 147-152, © Brainware University

A Smart Network and Port Scanning Tool

Anurag Mondal[1*], Liza Rozario[1], Partha Pratim Dasgupta[2]
[1]*Student of BCA, Dept. of Computational Science, Brainware University, West Bengal, India*
[2]*Dept. of Computational Science, Brainware University, West Bengal, India*
**anorak318@gmail.com*

**Corresponding Author*

Abstract

In computer networking, a port is an endpoint for communication. At the software level, a port is a conceptual construct within an operating system that defines a particular process or form of network service. The Transmission Control Protocol (TCP) and the User Datagram Protocol are the most common transport protocols that use port numbers (UDP). Port scanning refers to device port monitoring, which is performed by us when we want to connect with a certain computer. To find communication points inside unique device ports, we perform port-scanning techniques.To be able to check whether a service is currently running on a port or how many services are running or to scan number of hosts alive in a network, where we need to ping every single IP of the network and wait for the response, seems hectic and for this particular reason, a network automation tool will come very handy for the user and as a solution to this and a lot more, we bring DEDMAP - a Simple but Powerful, Clever and Flexible Cross-Platform Port Scanning tool made with ease to use and convenience in mind.

Keywords: Networking; TCP; UDP; Automation; Dedmap;

1. Introduction

An interconnection of hosts (devices) to help them communicate with one another is called Networking. There are devices available to make networking possible such as Router, Hubs, Switch, and Bridge.

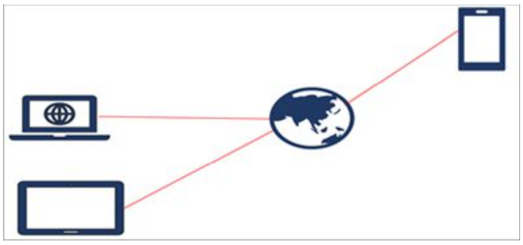

Figure 1: Depiction of Networking among devices

The sole reason for systems administration is to assist gadgets with conveying other. Be that as it may, how? How could gadgets speak with one another?

An extremely normal example is when we go to somebody's home, the first thing we do is to thump on the door to check if somebody is inside. In the realm of innovation, Ports are to some degree a comparable to doors. A port is where data is sent or gotten on a PC. In the event that somebody needs to send information to a PC, they need to send it through the assigned port.

Presently to decide if a port is open and tuning in (getting message) one requirements to speak with a port and check whether the administrations are sending back any reaction. Ports are software oriented and supervised by the operating system of a PC. Each port is associated with a specific help. They allow PCs to differentiate between different kinds of traffic. Ports are standardized across all gadgets affiliated with an organization. For particular conventions, most ports are saved; for example, all Hypertext Transfer Protocol (HTTP) messages go to port 80. Although IP offers the ability for messages to go to and from explicit gadgets, port numbers allow unique administrations or applications to be based within those gadgets. These open ports can be exploited by aggressors by code vulnerabilities or malevolent administration. That is the reason monitoring the unused ports and shutting them is viewed as the best practice to stop these assaults.[1]

2. Background Study

There are numerous tools available in the market to help with port scanning all their unique features. The top 4 best[4] ones are:

(1) Nmap[3],
(2) Unicornscan,
(3) Angry IP Scan and
(4) Netcat.

There are many more tools available for scanning ports and live hosts on a network but the ones mentioned above are the most significant ones. Upon doing further research on them we found their significance as below:

2.1 Nmap:

A free, open-source tool for vulnerability scanning and network discovery is Nmap, short for Network Mapper. Over the years, Nmap has grown and is highly versatile, it is a port-scan tool at its core, collecting information by sending raw packets to device ports. It listens for answers and decides whether ports are open, closed or filtered by, for example, a firewall in some way. Port discovery or enumeration requires other terminology used for port scanning. [4]

2.2 .Unicornscan:

Unicornscan is a modern engine for collecting and correlating knowledge designed for and by members of the safety research and testing community. It was designed to provide a Scalable, Accurate, Versatile, and Powerful Engine. It is published under the terms of the GPL license for the community to use.[5]

2.3. Angry IP Scanner:

A really fast IP address and port scanner is the Angry IP scanner. It can scan IP addresses as well as any of their ports in any range. It is lightweight and cross-platform. It can be freely copied and used anywhere, without any installation needed. The angry IP scanner simply pings each IP address to check if it is alive, then optionally resolves its host name, decides the MAC address, scans ports, etc. With plugins, the sum of collected data about each host can be expanded [6][7].

2.4. NETCAT

Netcat is a command line tool responsible for network data reading and writing. Netcat uses the network protocols TCP/IP and UDP to exchange data. The tool originally originated in the Unix world, but is now available on all platforms. The "Swiss army knife for TCP/IP" is also called Netcat. It enables us, for example, to diagnose faults and issues that jeopardize a network's functionality and protection. Netcat may also perform port scans, data streaming, or quick data transfers.[8][9] [10][11]

3. Problem Identification

The whole networking concept makes no sense if devices cannot communicate with each other. Devices communicate through ports so, to communicate, one needs to get in touch with the designated port of the target device. But pinging all computers on a network at once is not feasible. Nor is pinging all ports of a PC physically time-effective. We need a device that naturally does the work for us and is easy to use and quick.

There are numerous tools out in the market but everyone has their own cons. They do not have the required features all in one place which is why we worked up one, keeping convenience and ease in mind specially.

Our tool is named DEDMAP and the logo is given below:

Figure 2: Logo of DEDMAP

4. Proposed Model

DEDMAP is a Simple but Powerful, Clever and Flexible Cross-Platform Port Scanning tool made with ease to use and convenience in mind. Both TCP and UDP protocols have 0 to 65535 ports. The following three ranges [11][12] can be split between these 65535 ports:

System or reserved ports: from 0 to 1023
User or registered ports: from 1024 to 49151
Dynamic or private ports: from 49151 to 65535

The features of DEDMAP are enlisted below:

- It tries to scan a target IP or range of IP’s and find services that are running and listening on some ports.
- It can also scan a range of hosts to find live hosts.
- It performs both DNS and rDNS lookup.

- It performs sweep scan.
- Full Support for Android devices (via termux)

The proposed working of DEDMAP is depicted in the picture below:

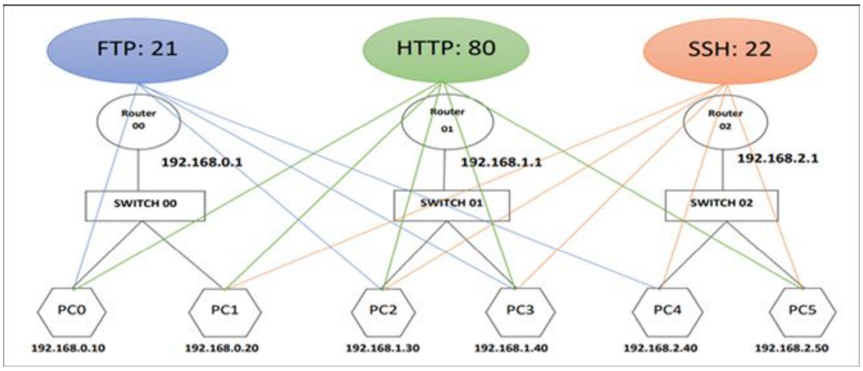

Figure 3: Depiction of Working of DEDMAP

Our tool DEDMAP is focused on automatically discovering open ports of specified targets. Consider the block diagram above, suppose user on PC0 wishes to check open ports on PC2. When the PC0 user types the command to check open ports on the target IP i.e.192.168.1.30 the output returns the port number 21,80,22 are open on the target as shown in the picture..

5. Material

Python3 [14]
Pip [15]
pyfiglet [16]
funcy
numpy[17]
colorama [18]

6. Future Scope

With these features is can be really helpful for network automation. We have plans to make DEDMAP available for chat applications like WhatsApp and Telegram as well in near future which will allow a user to use all the features of DEDMAP in a mobile phone right from a chat application.
We will also be adding a small honeypot feature in DEDMAP which will be very useful for monitoring a server for incoming traffic which can be further used as an IDS system. We are yet to fix UDP scanning. We will use multithreading[19] to drastically improve the performance of the tool. We will test the tool and add support for windows.

7. Limitations

DEDMAP is slow as no multithreading is used in the program. UDP does not work properly as of now. The user must maintain the sequence: "dedmap [--option(s)] [target(s)]". The tool supports IP range only in the last octet .i.e 1.1.1.(1-200) . This is also a safety measure to prevent the user from scanning the ENTIRE INTERNET (1-255.1-255.1-255.1-255) and blowing up his/her NIC, RAM, CPU and HARDDISK.

8. Conclusion

DEDMAP is expected to have lots of bugs as it is at a very early stage. It has not been tested in Windows yet and will not work most probably. Some of the example (commands) that you can try out with our tool is as follows:

- dedmap 192.168.0.10
- dedmaplocalhost
- dedmap –d google.com yahoo.com facebook.com localhost[20]
- dedmap google.com
- dedmap google.com yahoo.com
- dedmap 192.168.0.20 192.168.1.20 192.168.1.30
- dedmap 192.168.0.1-100 google.com (Perform a tcp scan on all the hosts without pinging to bypass firewall icmp block)
- dedmap –p 20 192.168.1.30
- dedmap –p 20,21,22 192.168.3.40
- dedmap –smlan –p 21 192.168.0.1-255 (Perform a tcp port scan in lan mode on all the live hosts)
- dedmap –s 192.168.3.50-255
- dedmap –sr 192.168.3.40-255 (Perform a reverse dns lookup on all the live targets in the network)
- dedmap –st 192.168.0.1-255 (To scan only the hosts which are alive in the network)
- dedmap –w report.txt 127.0.0.1

We would also like to add that this tool is made for educational purpose only. Use it with/on systems or networks you own or have permission from the owner. We shall not be held responsible for whatsoever you do with this tool.

With our tool, we hope to ease out communication with ports, detecting alive hosts in a network and many more activities for the user.

References

1. Forouzan, A.B., 2007. Data communications & networking (sie).Tata McGraw-Hill Education.
2. Borges,E. , MAY 22 2018 Top 5 Best Port Scanners.
3. Lyon, G.F., 2008. Nmap network scanning: The official Nmap project guide to network discovery and security scanning. Insecure.Com LLC (US).

4. Orebaugh, A. and Pinkard, B., 2011. Nmap in the enterprise: your guide to network scanning. Elsevier.

5. Lee, R.E. and Louis, J.C., 2008. Unicornscan.

6. Joshi, R.C. and Pilli, E.S., 2016. Network forensic tools.In Fundamentals of Network Forensics (pp. 71-93).Springer, London.

7. Avasthi, D., 2012. Network Forensic Analysis with Efficient Preservation for SYN Attack. International Journal of Computer Applications, 46(24), pp.17-22.

8. Kanclirz, J. ed., 2008. Netcat power tools.Elsevier.

9. Pinart, C. and Junyent, G., 2005, March. NetCat: Cross-plane approach for dynamic, distributed service provisioning in a GMPLS enabled optical testbed. In National Fiber Optic Engineers Conference (p. NThJ2).Optical Society of America.

10. Giacobbi, G., 2006. The gnu netcat.

11. Giacobbi, G., 2005. The GNU Netcat–Official homepage.

12. Stevens, W.R. and Narten, T., 1990. UNIX network programming. ACM SIGCOMM Computer Communication Review, 20(2), pp.8-9.

13. Imdad, U., Habib, M.A., Ahmad, M., Mahmood, N. and Ashraf, R., 2017, July. Auto Configuration Based Enhanced and Secure Domain Naming Service for IPV-6 Internet of Things. In Proceedings of the International Conference on Future Networks and Distributed Systems (pp. 1-5).

14. Pilgrim, M. and Willison, S., 2009. Dive Into Python 3 (Vol. 2). Apress.

15. Cruz, R.M., Hafemann, L.G., Sabourin, R. and Cavalcanti, G.D., 2020. DESlib: A Dynamic ensemble selection library in Python. Journal of Machine Learning Research, 21(8), pp.1-5.

16. Miller, P. and Bryce, C., 2016. Learning Python for Forensics.Packt Publishing Ltd.

17. Oliphant, T.E., 2006. A guide to NumPy (Vol. 1, p. 85). USA: Trelgol Publishing.

18. MacLean, C., 2017. Python usage metrics on Blue Waters. Cray User Group.

19. Akkary, H. and Driscoll, M.A., 1998, December. A dynamic multithreading processor.In Proceedings. 31st Annual ACM/IEEE International Symposium on Microarchitecture (pp. 226-236). IEEE.

20. Harwood, A. and Jacobs, T., 2005. Localhost: A browsable peer-to-peer file sharing system.