iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** V **Month of publication:** May 2020

DOI: http://doi.org/10.22214/ijraset.2020.5505

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

A Survey on different Port Scanning Methods and the Tools used to perform them

Abhinav Upadhya[1], B. K. Srinivas[2]
[1, 2]*Department of Information Science and Engineering, RV College of Engineering, R V Vidyaniketan Post, Mysore Road, Bengaluru – 560 059, Karnataka, India*

Abstract: Port Scanning is a method which is used in the vulnerability Detection of a host Machine. *Port scanning is a phase in footprinting and scanning; this comes in reconnaissance which is considered as the first stage of a computer attack. However there are different methods that can be used in different circumstances to perform a port scanning. But each method has its own cons and pros. This survey paper presents few such methods that was developed long ago but are in constant research and the tools that employ these methods. The paper presents different methods and compares them.*

Keywords: TCP Scan, Inverse Mapping Scanning, Slow Scan, Half Open Scan, FIN Scan, Xmas Tree Scan, Null Scan, UDP Scan, Dumb Scanning, Fragmentation, FTP bounce Sharing, Connect Method(), SYN, ACK, RST, RFC 959.

I. INTRODUCTION

Port Scanning refers to the process of sending packets to specific ports on a host and analyzing the responses to learn details about its running services or locate potential vulnerabilities. Port Scanning is one of the most popular techniques that attackers use to discover services that can be exploited to break into systems. Port Scanning is often the first step of reconnaissance used by hackers when trying to infiltrate a network or steal/destroy sensitive data.

By port scanning, the attacker can find the following information about the targeted systems: what services are running, what users own those services, whether anonymous logins are supported, and whether certain network services require authentication. Port scanning is a technique used by attackers, curious individuals, and administrators to collect information from computers connected to a network. Port scanning is a technique used by attackers, curious individuals, and administrators to collect information from computers connected to a network. System and network administrators use port scans to identify open ports to a system so that they may be able to limit access to those ports, or shut them off entirely. Attackers use port scanning in the same way that administrators do, but with malicious intent. All systems that are connected to a LAN or the Internet via a modem run services that listen to well-known and not so well-known ports.

This paper is structured as follows. Section II discusses the different methods on Port Scanning mainly derived from [1]. Section III discusses the Prerequisites of Port Scanning method. Section IV Discusses the Importance/Need of Port Scanners mainly from [3]. Section VI talks about the widely used methodology and why it is widely used.

II. DIFFERENT METHODS

A. Non Stealth Scanning (TCP Connect)

The Non Stealth Scanning makes use of the TCP Connect method which does a complete 3 way handshake with the host machine. When a client wants to connect with a server, it first sends a TCP packet with the SYN (Synchronize Sequence Number) flag set. The server then sends back a TCP packet with the SYN and ACK (Acknowledge) flags set if the port is open on the server. A RST (Reset) packet is sent to the client if the port is closed. If the port is open and the server sends back the SYN|ACK packet, the client computer then sends an ACK back to the server.

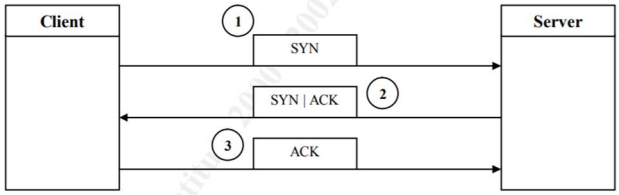


Fig 1. Three – Way Handshake.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

A non-stealth port scanner utilizes the TCP connect() method of connecting to the destination host. The connect() is a system call provided by the operating system to open a connection to a remote host. The Advantage of using this scanning method is that it requires no special privilege. But the Disadvantage of using this method is that the scanning activity is very visible to the administrator.
There are many tools that can perform this scan namely, nmap, TCP Port Scanner, hping2.

B. Inverse Mapping Scan

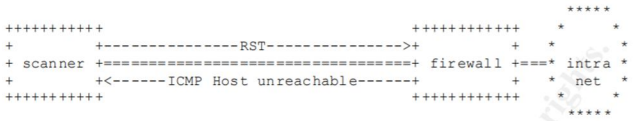


Fig 2. Inverse Mapping Scan

The idea behind this method is that "intruders send packets that normally would go unnoticed or cause no unusual behaviour to a list of addresses". Attackers use specially crafted packets with customized flags, which in this case included RST (Reset) and SYN-ACK packets and DNS response packets. This type of scan did not find out information about the ports specifically that were open, but rather tested the host to see if it would respond. A computer if exists and is connected to the network would respond to the request, while a non-existent computer would generate an ICMP host unreachable error message.
TCP resets are often used for these attacks, since TCP resets are ubiquitous in the Internet, and few current IDS systems bother to log them. This will change in the future, as IDS systems evolve. Resets aimed at non-existing hosts will certainly become one of the targets of analysis, for in normal use they are only seen as results of error conditions. Thus an abnormally large amount of resets aimed at non-existing hosts is a clear indication of a scan in progress. The problem for the attacker is that in order to gain information from the scan, he has to provide at least one genuine source address where he can study the returned packets. Many scanning tools, e.g. nmap, provide a way of sending decoy packets from several forged source addresses to confuse the IDS systems. But always there is one genuine address among the rest, making the tracing of the attacker possible if not probable.
This method is best used when the attacker wishes to know about the existence of a host Machine. Much Effort has been put into improving the method. There are several tools that can be used to perform an inverse mapping scan namely, nmap, vscan.

C. Slow Scan

This is a low-tech solution to the problem of being logged or noticed by the remote system. A "normal" scan will go through thousands of ports within a short time frame, usually under a minute. By waiting for a given amount of time between scans for individual ports, logging programs can be defeated. The downside to this type of stealth is the time factor involved. To be stealthy enough to be undetected by an intrusion detection system or a system administrator can take a very long time. There is nothing fancy about this method, but it does prove that unless a history is kept of all the attempts to each port, detection becomes very difficult. This can be used when the time factor involved is not an issue for the attacker. There are many tools that can perform a slow port scanning namely, nmap, angry IP Scanner, unicorn scanner, netcat.

D. SYN Scan(Half Open Scan)

The title came from the method used to connect to ports on a host. While the TCP connect() method uses the full 3-way handshake to connect to a port on a host, the SYN scan uses a modified handshake which only includes a 2-way communication channel. The SYN scan begins exactly the same way that the TCP connect() method does by having the client send a packet with the SYN flag set. Similarly, the server then sends back a SYN|ACK packet to the client if the port is open. If the port is not open, a RST (Reset) packet is sent to the client. This is where the SYN scan and the TCP connect() method differ: a final ACK packet is never sent back to the server acknowledging that the client has received the SYN|ACK packet from the server. Instead, a RST packet is sent to the server in order to destroy the connection.

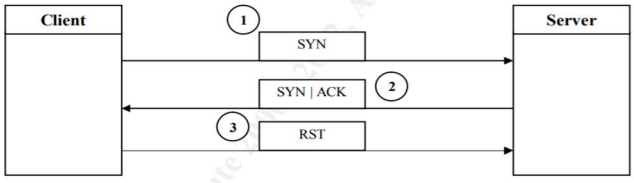


Fig 3. Incomplete 3 – way handshake.

SYN scanning is the most common type of port scanning that is used because of its enormous advantages and few drawbacks. As a result, novice attackers tend to overly rely on the SYN scan while performing system reconnaissance. As a scanning method, the primary advantages of SYN scanning are its universality and speed. RFC 793 defines the required behavior of any TCP/IP device in that an incoming connection request begins with a SYN packet, which in turn must be followed by a SYN/ACK packet from the receiving service. For this reason, like TCP Connect scanning, SYN scanning works against any TCP stack. Unlike TCP Connect scanning, it is possible to scan thousands of ports per second using this method.

The scanning rate is extremely fast because no time is wasted completing the handshake or tearing down the connection. TCP SYN scanning can also immediately detect 3 of the 4 important types of port status: open, closed, and filtered. When a SYN is sent to an open port and unfiltered port, a SYN/ACK will be generated. This technique allows an attacker to scan through stateful firewalls due to the common configuration that TCP SYN segments for a new connection will be allowed for almost any port. When a SYN packet is sent to a closed port a RST is generated, indicating the port is closed. When SYN scanning to a particular port generates no response, unreachable errors, the port is filtered.

There are many tools that can be used to perform a SYN Scan namely nmap, strobe, scanrand component of the Paketto Keiretsu suite, TCP Port Scanning.

E. FIN Scan

An adversary uses a TCP FIN scan to determine if ports are closed on the target machine. This scan type is accomplished by sending TCP segments with the FIN bit set in the packet header. The RFC 793 expected behavior is that any TCP segment with an out-of-state Flag sent to an open port is discarded, whereas segments with out-of-state flags sent to closed ports should be handled with a RST in response. This behavior should allow the adversary to scan for closed ports by sending certain types of rule-breaking packets (out of sync or disallowed by the TCB) and detect closed ports via RST packets.

The FIN (Finish) scan is an answer to the possible logging capabilities of the SYN scan. Some packet loggers and firewalls are configured to detect SYN packets to restricted ports. In the FIN scan, a packet is sent with just the FIN flag set. If the port is closed, the host sends back a RST flag, whereas an open port simply ignores the packet and nothing is returned to the client. This is required behaviour as set out in the RFC for Transmission Control Protocol. [6] It is through exploiting the requirement that TCP has for ensuring packets arrive at their destination that attackers can probe open ports and possibly evade detection. Because a firewall or packet logger may be setup to detect SYN packets, a FIN packet would slip through unnoticed. There are many tools that can perform a FIN Scan namely nmap, hping2, NetScan to name a few.

F. Xmas Tree Scan

Xmas scans derive their name from the set of flags that are turned on within a packet. These scans are designed to manipulate the PSH, URG and FIN flags of the TCP header.

Like the FIN scan, the Xmas tree scan employs the use of invalid packet header flags to elicit a response from a host regarding open ports. There are a few different methods that have been applied that all use the Xmas tree scan name. Nmap executes the Xmas tree scan using 3 packet header flags, which are the FIN, URG (Urgent), and PSH (Push) flags. This type of scan is very similar to the FIN scan, with 2 extra flags set.

Other Xmas tree scanners set all TCP header flags to be on, which is most likely where the name is from. Like FIN scan, a closed port will return a RST packet, whereas an open port will ignore the packet.

There are many tools that can perform a Xmas Tree Scan namely nmap, hping2, netscan to name a few.

G. Null Scan

The Null scan produces a reaction to the FIN and Xmas tree scans, but differs in packet header flags. Instead of turning on flags in the header that would cause the packet to be received by the host as an invalid packet, the Null scan turns off all header flags. This again causes a RST packet to be sent to the client if a port is closed, but is ignored if the port is open. Microsoft operating systems in addition to a number of others have ignored the RFC for TCP and have implemented it somewhat differently than the standard. Null Scan is a type of scan that is used to identify ports.

A Null Scan is a series of TCP packets that contain a sequence number of 0 and no set flags. In a production environment, there will never be a TCP packet that doesn't contain a flag. Because the Null Scan does not contain any set flags, it can sometimes penetrate firewalls and edge routers that filter incoming packets with particular flags.

Works only on unix based systems. There are many tools that can be used to perform Null Scan namely, nmap, hping2, netscan to name a few.

H. UDP Scan

UDP is a much simpler of a protocol than TCP is, given that it is not connection oriented, that is UDP does not concern itself with ensuring that packets arrive at their destination successfully. There are a number of vulnerabilities that can be scanned for by using UDP scanning. Programs can easily open high UDP ports without the user's knowledge. These are mainly undocumented, thus tracking them down is made much easier with the application of UDP scanning. Currently there is only one known method for UDP scanning, which entails sending a 0 byte UDP packet to each port on the host machine. If a port is closed, an ICMP port unreachable error will be returned, otherwise it can be inferred that the port is open.

Different types of ICMP messages can indicate a filtered port. UDP scanning is slower than TCP scanning. The protocol characteristics of UDP make port scanning inherently more difficult than with TCP, as well as dependent upon ICMP for accurate scanning. Due to ambiguities that can arise between open ports and filtered ports, UDP scanning results often require a high degree of interpretation and further testing to refine. In general, UDP scanning results are less reliable or accurate than TCP-based scanning. There are many tools that can perform a UDP Scan namely nmap, FounderStone's superscan.

I. Idle Scan(Dumb Scan)

The dumb scan method of stealth scanning involves the use of a third party computer that receives very little or no network traffic. This third party is also known as a dumb host. Typically, attackers search for these computers on cable modem subnets looking for Windows-based computers that have been left on at night. The dumb host method of stealth scanning requires a utility to generate customized TCP packets, and a ping utility. Firstly, the attacker sends a repetitive ICMP ping to the dumb host with an ID number of +1. Secondly, the attacker sends a spoofed SYN packet to the host with the dumb host's IP address in place of his/her own. The destination port is set to the port that the attacker wishes to scan. Because the host receives the TCP packet with the IP of the dumb host, any reply to a connection request will be sent back to the dumb host. The continuous pinging of the dumb host reveals whether the port is open on the host or not. Typically, if the port is open, the ID number will increase, whereas if the port is closed the ID will most likely remain at +1.

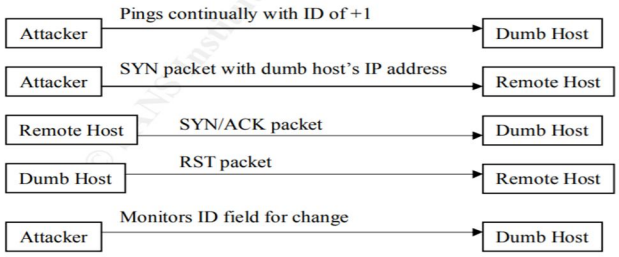


Fig 4. Procedure for dumb scan.

The dumb scan is very effective, and very stealthy. By utilizing a third party, connection attempts are concealed and most logging capabilities by an intrusion detection system are thwarted. This is due to the fact that no information is communicated directly from the remote host to the attacker's computer.

A unique advantage of idle scan is that it can be used to defeat certain packet filtering firewalls and routers. IP source address filtering is a common (though weak) security mechanism for limiting machines that may connect to a sensitive host or network. For example, a company database server might only allow connections from the public web server that accesses it. Or a home user might only allow SSH (interactive login) connections from his work machines.

Idle scanning can sometimes be used to map out these trust relationships. The key factor is that idle scan results list open ports from the zombie host's perspective. A normal scan against the aforementioned database server might show no ports open, but performing an idle scan while using the web server's IP as the zombie could expose the trust relationship by showing the database-related service ports as open. A disadvantage to idle scanning is that it takes far longer than most other scan types. Another issue is that you must be able to spoof packets as if they are coming from the zombie and have them reach the target machine. Many ISPs (particularly dialup and residential broadband providers) now implement egress filtering to prevent this sort of packet spoofing. Higher end providers (such as colocation and T1 services) are much less likely to do this. If this filtering is in effect, Nmap will print a quick error message for every zombie you try. If changing ISPs is not an option, you might try using another IP on the same ISP network. Sometimes the filtering only blocks spoofing of IP addresses that are *outside* the range used by customers. Another challenge with idle scan is that you must find a working zombie host, as described in the next section.

This is the ultimate stealth scan there is.

There are many tools that does the dumb scan namely, nmap, hping2, vscan.

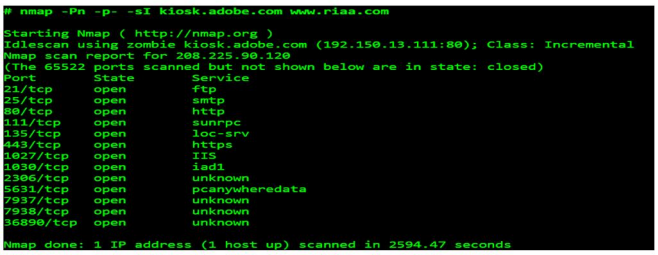


Fig 5. An idle scan against riaa using nmap.

J. Fragmentation

The main idea behind fragmentation is to use very small, broken up IP packets. If the TCP header is broken up into many smaller pieces, it is much more difficult for packet filters, intrusion detection systems, and system administrators to detect what the attacker is doing. Firewalls and packet filters that queue up all IP packets will most likely to detect this kind of a probe, since all of the fragmented packets would be collected and analyzed before letting them pass. These packets are very small and usually of different size. Many programs cannot cope with packets of such small size and shape. The point of this type of protection is not so much to protect the identity of the attacker, but more to conceal the intention of the packets being transmitted to the server.

Advantage of this scanning method is that this can be used to evade many firewalls and ips systems. The Disadvantage of using this method is that scanning can have a negative impact on the target devices and other devices which is on the path.

There are many tools that can be used to perform fragmentation scanning namely, fragtest utility, fragRoute utility, nmap to name a few.

K. FTP Bounce Scanning

Bounce scanning is another technique, which allows an attacker to camouflage his/her scanning activities. Essentially, attackers "bounce" their scans through services running on other computers that allow commands to pass through, in effect covering their tracks. An interesting feature of the FTP protocol (RFC 959) is support for so-called proxy FTP connections. This allows a user to connect to one FTP server, then ask that files be sent to a third-party server. Such a feature is ripe for abuse on many levels, so most servers have ceased supporting it. One of the abuses this feature allows is causing the FTP server to port scan other hosts. Simply ask the FTP server to send a file to each interesting port of a target host in turn. The error message will describe whether the port is open or not. If the port is listening on the host, the transfer process will be successful indicating an open port, but if the port is closed an error will be generated. This method is useful for scanning behind firewalls and concealing the identity of the attacker, but it is also slow and somewhat tedious. There are tools that can be used to perform this scan namely, nmap, proxy bounce sharing to name a few.

III. PREREQUISITES REQUIRED

All forms of port scanning rely on the assumption that the targeted host is compliant with RFC 793 - Transmission Control Protocol. Although this is the case most of the time, there is still a chance a host might send back strange packets or even generate false positives when the TCP/IP stack of the host is non-RFC-compliant or has been altered. This is especially true for less common scan techniques that are OS-dependent (FIN scanning).

IV. NEED FOR PORT SCANNING

Attackers and administrators both use port scanning method for both attacking and for securing a host machine.

The attackers often use port scanning as a preliminary step when targeting networks. They use the scan to scope out the security levels of various organizations and determine who has a strong firewall and who may have a vulnerable server or network. A number of TCP protocol techniques actually make it possible for attackers to conceal their network location and use "decoy traffic" to perform port scans without revealing any network address to the target. They probe networks and systems to see how each port will react - open, closed, or filtered. Open and closed responses alert hackers that your network is in fact on the receiving end of the scan. These cyber criminals can then determine the level of security and what type of operating system your business has. Port scanning is an old technique that requires security changes and up-to-date threat intelligence as protocols and security tools are evolving daily. Port scan alerts and firewalls are necessary to monitor traffic to your ports and ensure malicious traffic doesn't detect your network.

Security techs can routinely conduct port scanning for network inventory and to expose possible security vulnerabilities.

V. CONCLUSION

The table below gives a fair idea regarding the methods and their use cases and the tools that are used to perform them.

SL NO.	Method	Details	Circumstances where the method is used	Tools that perform them
1.	Non Stealth(TCP Connect)	Uses tcp 3-way handshake to make connection-n	where no special privilege is required .	Nmap,hping2.
2.	Inverse Mapping	Use specially crafted packets which includes RST (Reset) and SYN-ACK packets and DNS response packets.	Used to find the existence of a host in a network	Nmap , vscan
3.	Slow Scan	By giving time between two scans the detection becomes difficult	Best used when time isn't a big factor. Also unless track records for each port scan is kept, detection becomes very difficult.	Nmap, angry IP Scanner, unicorn scanner, netcat.
4.	SYN scan	SYN scan uses modified handshake which only includes a 2-way communication channel.	Unlike TCP Connect scanning, it is possible to scan thousands of ports per second using this method.	Nmap, strobe , TCP port Scanner
5.	FIN Scan	In the FIN scan, a packet is sent with just the FIN flag set. If the port is closed, the host sends back a RST flag, whereas an open port simply ignores the packet and nothing is returned to the client.	FIN scanning results must always be interpreted as part of a larger scanning strategy	Nmap, netScan, hping2.
6.	Xmas Tree Scan	very similar to the FIN scan, with 2 extra flags set.	Fast when compared to other scans.	Nmap, hping2, netScan.
7.	Null Scan	The Null scan is similar to FIN and Xmas tree scans, but differs in packet header flags. Instead of turning on flags in the header that would cause the packet to be received by the host as an invalid packet, the Null scan turns off all header flags.	Works only for unix based systems.	nmap,netScan,hping2.
8.	UDP Scan	involves sending a UDP datagram to the target port and looking for evidence that the port is closed.	Programs can easily open high UDP ports without the user's knowled-ge.	nmap, Foundstone's SuperScan, Scanudp utility developed by Fryxar
9.	Idle Scan	involves the use of a third party computer that receives very little or no network traffic. This third party is also known as a dumb host	Can be used to defeat certain packet filtering firewalls and routers. IP source address filtering is a common (though weak) security mechanism for limiting machines that may connect to a sensitive host or network.	hping2, vscan, nmap
10.	Fragmentation	is to use very small, broken up IP packets	for evading the firewalls and other packet filtering devices. This type of scanning can consume the processing power of the victim host or the devices which are in the front of the victim IP addresses.	fragtest utility, fragRoute utility,nmap.
11	FTP Bounce Scanning	The FTP server sends a file to each interesting port of a target host. The error message will describe whether the port is open or not.	Nearly all port program are configured to refuse port commands, but there are servers still that are vulnerable to this attack.	nmap,proxy bounce scanning.

VI. WIDELY USED METHODOLOGY

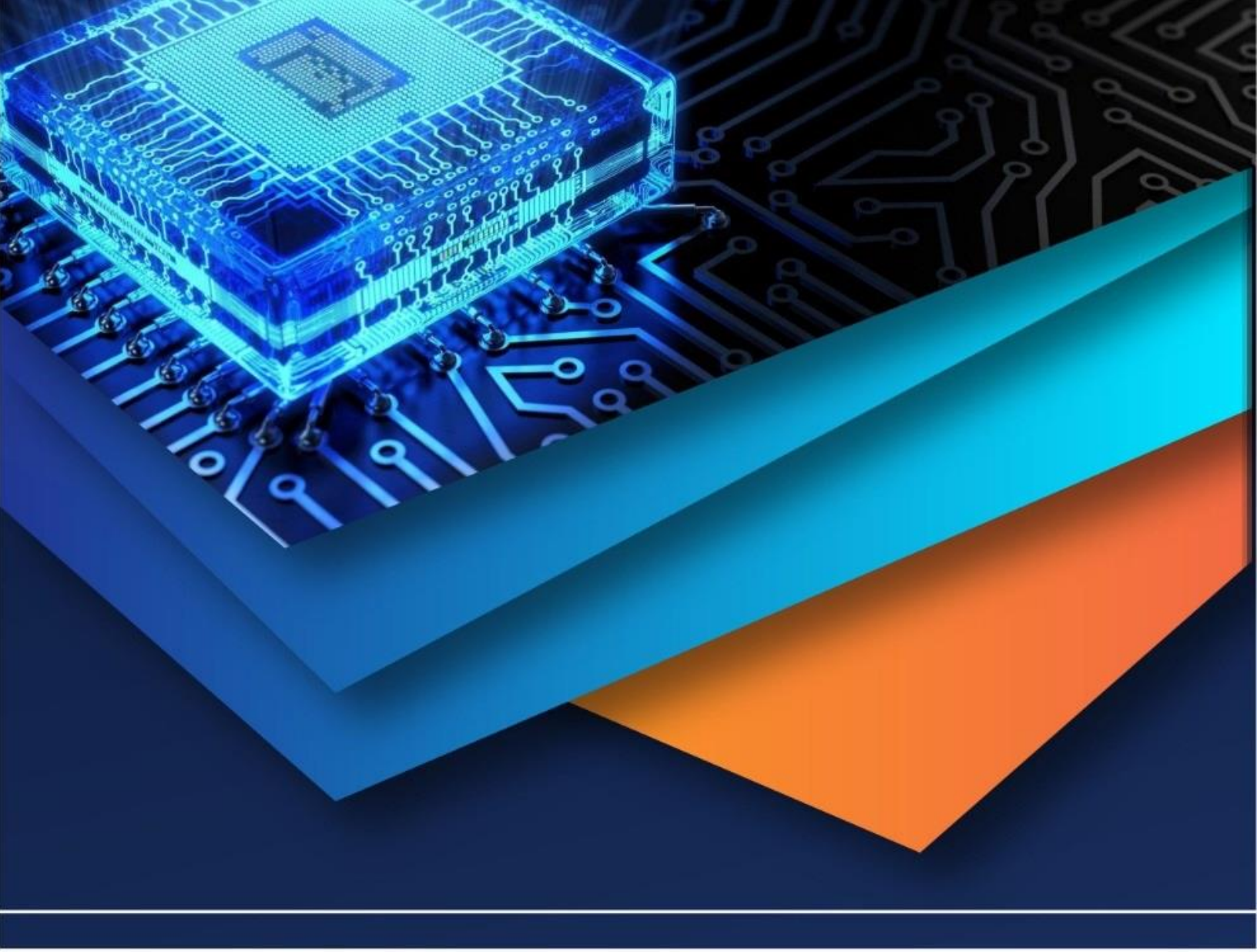
TCP SYN Scan is the most popular port scanning method used .Because it can be performed quickly, scanning thousands of ports per second on a fast network not hampered by intrusive firewalls. SYN scan is relatively unobtrusive and stealthy, since it never completes TCP connections. It also works against any compliant TCP stack rather than depending on idiosyncrasies of specific platforms as Nmap's FIN/NULL/Xmas, and idle scans do. It also allows clear, reliable differentiation between open, closed, and filtered states.

VII. ACKNOWLEDGEMENT

A lot of analysis and reading was done prior to the curation of the content of the given paper. This would definitely not be possible under the guidance of our mentor, Prof. B.K. Srinivas, who constantly guided us and gave us a direction in the due course of the given paper. We would also like to thank our Head of Department, Dr. B.M. Sagar, who gave us this opportunity to work on this paper.

REFERENCES

[1] https://www.giac.org/paper/gsec/1985/stealth-port-scanning-methods/103446
[2] http://www.ajer.org/papers/v5(06)/G050603842.pdf.
[3] https://capec.mitre.org/data/definitions/302.html.
[4] Mehiar Dabbagh, Ali J. Ghandour, Kassem Fawaz, Wassim El Hajj, Hazem Hajj, “Slow port scanning detection”, IEEE, 5-8 Dec. 2011 doi: 10.1109/ISIAS.2011.6122824 .
[5] Slow port scanning detectionH.U. Baig and F. Kamran, "Detection of Port and Network Scan Using Time Independent Feature Set," Intelligence and Security Informatics, 2007 IEEE, pp.180-184, 23-24 May 2007 doi: 10.1109/ISI.2007.379554.
[6] J. Kim and J. Lee, "A slow port scan attack detection mechanism based on fuzzy logic and a stepwise policy," Intelligent Environments, 2008 IET 4th International Conference, pp.1-5, 21-22 July 2008.
[7] S. Jahr, "Slow portscanning detection," Internet: http://www.ztian.org/docs/slow-portscanning-detection.pdf, Nov. 2005 [Mar. 22, 2010].
[8] 2014Tariq Ahamad Ahanger, Port Scan – A Security Concern, International Journal of Engineering and Innovative Technology(IJEIT),ISSN-2277-3754, Volume 3 Issue 10 April.
[9] Nmap Network Scanning Guide – Gordon Lyon.
[10] Ensafi, R., Park, J. C., Kapur, D., and Crandall, J. R. (2010) Idle port scanning and non-interference analysis of network protocol stacks using model checking. Proceedings of USENIX Security’10, Washington, DC, USA, pp. 257–272. USENIX Association.
[11] Staniford-Chen, S., Cheung, S., Crawford, R., Dilger, M., Frank, J., Hoagland, J., Levitt, K., Wee, C., Yip, R., and Zerkle, D. (1996) Grids: a graph based intrusion detection system for large networks. Proceedings of 19th NISS’96, Baltimore, MD, USA, October, pp. 361–370. NIST.
[12] Green, J., Marchette, D., Northcutt, S., and Ralph, B. (1999) Analysis techniques for detecting coordinated attacks and probes. Proceedings of WIDNM’99, Santa Clara, California, USA, April, 9-12, pp. 1–9. USENIX Association.
[13] Robertson, S., Siegel, E. V., Miller, M., and Stolfo, S. J. (2003) Surveillance detection in high bandwidth environments. Proceedings of DARPA DISCEX III’03, Washington, DC, April, 22-24, pp. 130–139. IEEE Computer Society.
[14] Heberlein, T., Dias, G., Levitt, K., Mukherjee, B., Wood, J., and Wolber, D. (1990) A network security monitor. Proceedings of RISP’90, Oakland, California, USA, May, 7-9, pp. 296–304. IEEE Computer Society.
[15] Fyodor (1997) The art of port scanning. Phrack Mag., Article 11, 7.
[16] QoSient. Argus. http://www.qosient.com/argus/.
[17] Leckie, C. and Kotagiri, R. (2002) A probabilistic approach to detecting network scans. Proceedings of NOMS’02, Florence, Italy, April, 15-19, pp. 359–372. IEEE Computer Society.
[18] Kim, H., Kim, S., Kouritzin, M. A., and Sun, W. (2004) Detecting network portscans through anomaly detection. Proc. SPIE 5429, Orlando, FL, USA, April,12, pp. 254–263.
[19] Kato, N., Nitou, H., Ohta, K., Mansfield, G., and Nemoto, Y. (1999) A real-time intrusion detection system (ids) for large scale networks and its evaluations. IEICE Trans. Commun., E82-B, 1817–1825.
[20] Ertoz, L., Eilertson, E., Lazarevic, A., Tan, P.-N., Dokas, P., Kumar, V., and Srivastava, J. (2003) Detection of novel network attacks using data mining. Proceedings of ICDM WDMCS’03, Melbourne, Florida, USA, November, 19, pp. 30–39.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY