NETWORK PORT SCANNER
A COURSE PROJECT REPORT

18CSC302J – COMPUTER NETWORKS

Submitted by

MITHUN MENON [RA2111003010835]
AMARTYA CHATTERJEE [RA2111003010838]
S SIDDHARTH[RA2111003010842]
ADITYA VARDHAN SHARMA[RA2111003010844]
SAMRIDDHI SINGH[RA2111003010856]
SAAKSHI MATHUR[RA2111003010884]

Under the guidance of

Dr. Suchithra M

Associate Professor, Department of Computing Technologies

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

NOVEMBER 2023

SRM

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that Course project report titled **"NETWORK PORT SCANNER"** is the bonafide work of **MITHUN MENON [RA2111003010835], ADITYA VARDHAN SHARMA[RA2111003010844], AMARTYA CHATTERJEE[RA2111003010838], S SIDDHARTH[RA2111003010842], SAMRIDDHI SINGH[RA2111003010856], SAAKSHI MATHUR[RA2111003010884]** who carried out the course project under my supervision. Certified further,that to the best of my knowledge, the work reported herein does not form any other project reportor dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Department of Computing Technology Kattankulathur 603 203 SRM IST

SIGNATURE

Dr. Suchithra M.
Associate Professor
Department of Computing Technologies

M. Pushpalatha

SIGNATURE

Dr. M. Pushpalatha
HEAD OF THE DEPARTMENT
Professor & Head
Department of Computing Technologies

SRM

SRM Institute of Science and Technology

Department of Computing Technologies

Own Work Declaration Form

Degree/ Course : B.Tech in Computer Science and Engineering

Student Names : Mithun Menon, Amartya Chatterjee, S Siddharth, Aditya Vardhan Sharma, Samriddhi Singh, Saakshi Mathur

Registration Number: RA2111003010835, RA2111003010838, RA2111003010842, RA2111003010844, RA2111003010856, RA2111003010884

Title of Work : Network Port Scanner

We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism, as listed in the University Website, Regulations, and the Education Committee guidelines.

We confirm that all the work contained in this assessment is our own except where indicated, and that we have met the following conditions:

- Clearly references / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc.)
- Given the sources of all pictures, data etc. that are not my own

- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:
I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above.
If you are working in a group, please write your registration numbers and sign with the date for every student in your group.

S. Siddharth (RA2111003010842) S. Siddharth
Samriddhi Singh (RA2111003010856) Samriddhi
Saakshi Mathur (RA2111003010884) Saakshi Mathur
Amartya Chatterjee (RA2111003010858) Amartya Chatterjee
Nitum Menon (RA2111003010835)
Aditya Vardhan Sharma (RA2111003010844) ADS

ACKNOWLEDGEMENT

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, **Dr. T.V.Gopal**, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, **Dr. Pushpalatha M,** Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We register our immeasurable thanks to our Faculty Advisor, **Dr. Sindhu C**, Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, **Dr. Suchithra M**, Associate Professor, Department of Computing Technologies, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under his mentorship.

He provided us with the freedom and support to explore the research topics of our interest. His passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank the Computing Technologies Department staff and students, SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank parents, family members, and friends for their unconditional love, constant support, and encouragement.

MITHUN MENON [RA2111003010835]

AMARTYA CHATTERJEE [RA2111003010838]

S SIDDHARTH[RA2111003010842]

ADITYA VARDHAN SHARMA[RA2111003010844]

SAMRIDDHI SINGH[RA2111003010856]

SAAKSHI MATHUR[RA2111003010884]

ABSTRACT

A network port scanner is a powerful tool used to detect open ports on a computer network. Each port is associated with a specific network protocol, and when a port is open, it means that the associated protocol can be used to communicate with the device on the network. For example, port 80 is typically associated withHTTP, which is used for web browsing, while port 21 is associated with FTP, which is used for file transfers.

Port scanners can be used for legitimate purposes, such as network administration, security testing, and vulnerability assessment. By identifying open ports on a network, network administrators can ensure that their networks are secure and free from potential vulnerabilities. Similarly, security professionals can use port scanners to test the effectiveness of their security measures and to identify potential security risks.

However, port scanners can also be used for malicious purposes, such as hacking, intrusion, and port scanning. For this reason, port scanning is often considered a potentially harmful activity, and proper permissions and authorizations should be obtained before conducting any port scanning activities. In addition, port scanning may also violate privacy laws or network usage policies, so it isimportant to use port scanners responsibly and ethically.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

A network port scanner is a software tool used to identify open ports and services running on a computer or network device. A port is a communication endpoint that is used by network protocols to establish connections between devices. By scanning for open ports, a port scanner can determine what services or applications are running on a device and potentially identify vulnerabilities that could be exploited by attackers.

There are several types of port scanners available, including TCP, UDP, and SYN scanners. TCP scanners are the most commonly used type and work by sending a connection request to a target port and waiting for a response. If a response is received, the port is considered open. UDP scanners work similarly, but instead send a UDP datagram to the target port. SYN scanners are more stealthy and work by sending a SYN packet and analyzing the response, without actually completing the connection.

Port scanners can be used for a variety of purposes, including network administration, security testing, and hacking. Network administrators may use port scanners to identify services running on their network and ensure that they are properly configured and secured. Security testers may use port scanners to identify vulnerabilities in a target system, such as open ports that could be used to launch an attack. Hackers may also use port scanners to identify potential targets for attacks.

It's worth noting that port scanning can be considered a type of reconnaissance and may be illegal in certain circumstances. It's important to always obtain proper authorization before conducting any type of port scanning or vulnerability testing.

CHAPTER 2

LITERATURE SURVEY

Here's a literature survey of some of the research and articles available on network port scanners:

1. "A Comparative Study of Port Scanning Techniques": This paper by B. M. Al-Zoubi et al. compares several port scanning techniques and evaluates their effectiveness in identifying open ports. The authors use a variety ofscanning tools, including Nmap, Superscan, and NetScan Tools, and test them against different types of targets.

2. "Port Scan Detection using Artificial Neural Network": This paper by S. P. Singh and P. B. Jagdale proposes amethod for detecting port scans using an artificial neural network. The authors train the network on a dataset of known port scans and evaluate its accuracy in detecting new scans.

3. "A New Design of Network Port Scanner Based on FPGA": This paper by Y. Zhang et al. presents a hardware-based network port scanner design using a field-programmable gate array (FPGA). The authors demonstrate that their design can achieve high-speed port scanning and compare it to other software-based scanning tools.

4. "Port Scanner Detection and Prevention": This article by S. Mushtaq and S. S. Hussain provides an overview ofport scanning techniques and methods for detecting and preventing them. The authors discuss various detection techniques, such as packet filtering and intrusion detection systems, and also provide recommendations for securing networks against port scans.

5. "A Practical Guide to Nmap (Network Mapper)": This book by G. Lyon is a comprehensive guide to using Nmap, one of the most popular open-source network port scanning tools. The book covers a range of topics, frombasic port scanning techniques to advanced features like scripting and vulnerability scanning.

These resources provide a good starting point for understanding the various aspects of network port scanning,from the technical details of scanning techniques to the practical considerations of detection and prevention.

CHAPTER 3

SYSTEM ARCHITECTURE AND DESIGN

A system architecture design of a network port scanner is typically based on several factors, including the scanning techniques to be used, the user interface requirements, theperformance and scalability goals, and the security considerations.

The architecture design should take into account the specific requirements of the scanning application, such as the types of devices to be scanned, the scanning frequency, and the amount of data to be collected. It should also consider the hardware and software platforms that will be used to run the scanner, as well as any network protocols or security policies that must be followed.

In general, a system architecture design of a network port scanner is based on thefollowing key principles:

1. Modularity: The architecture design should be modular, with each component of the system designed to be interchangeable and easily updated. This allows for flexibility in configuring and customizing the scanner to meet the specific requirements of different applications.

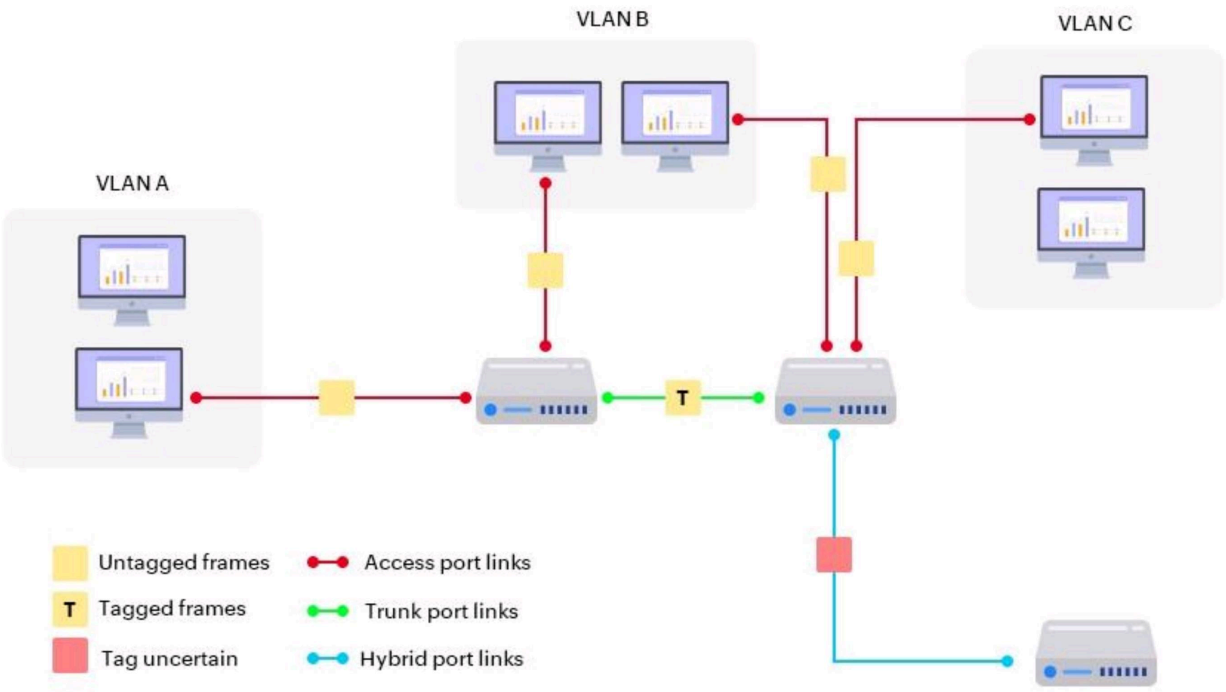
2. Scalability: The architecture design should be scalable, with the ability to handle large volumes of scanning requests and data. This requires careful consideration of the hardware and software platforms that will be used, as well as the network protocols and security policies that will be implemented.

3. Security: The architecture design should be secure, with measures in place to prevent unauthorized access and protect sensitive data. This requires the implementation of access controls, encryption, and other security features.

4. Performance: The architecture design should be optimized for performance, with efficient algorithms and data structures used to minimize the scanning time and reducethe impact on network performance.

5. Usability: The architecture design should be user-friendly, with a clear and intuitiveuser interface that allows users to easily configure and run scans, view results, and generate reports.

In summary, the system architecture design of a network port scanner is based on a combination of technical, performance, and security requirements, with a focus on modularity, scalability, security, performance, and usability.



CHAPTER 4

METHODOLOGY

A network port scanner is a tool used to identify the open ports on a networked device.Here is a common methodology used by port scanners:

1. Determine the target: First, the target device or network range is identified for the scan.This may involve identifying the IP addresses of specific devices or ranges of IP addresses.

2. Choose a port scanning tool: There are several port scanning tools available, such as Nmap, SuperScan, and Angry IP Scanner, each with their own features and capabilities.The tool is selected based on the needs of the user and the type of scan required.

3. Select a scanning technique: Port scanners use different scanning techniques, such asTCP SYN scan, TCP connect scan, UDP scan, and more. The choice of technique depends on the network environment and the type of scan required.

4. Configure scan options: The user can configure scan options such as the range of ports to be scanned, the time between packets, and the number of retries for unresponsive ports.

5. Initiate the scan: Once the scanning options are configured, the scan is initiated by running the port scanning tool. The tool sends packets to the target device on the specifiedports and waits for responses.

6. Analyze the results: After the scan is complete, the results are analyzed to identify the open ports on the target device. The results may be displayed in different formats, such asa list of open ports or a graphical representation of the network topology.

7. Take action: Depending on the reason for the scan, the results may be used to identify security vulnerabilities, test network connectivity, or troubleshoot network issues. Appropriate action is taken based on the results of the scan.

It is important to note that port scanning without the permission of the network owner oradministrator may be illegal and may lead to legal consequences.

CHAPTER 5

CODING AND TESTING

CODING:

python Copy code

```python
import socket
import argparse
import threading
from queue import Queue
from time import time

# Define common ports to scan for in a dictionary
common_ports = {
    '21': 'FTP',
    '22': 'SSH',
    '23': 'Telnet',
    '25': 'SMTP',
    '53': 'DNS',
    '69': 'TFTP',
    '80': 'HTTP',
    '109': 'POP2',
    '110': 'POP3',
    '123': 'NTP',
    '137': 'NetBIOS',
    '138': 'NetBIOS',
    '139': 'NetBIOS',
    '143': 'IMAP',
    '161': 'SNMP',
    '194': 'IRC',
    '389': 'LDAP',
    '443': 'HTTPS',
    '445': 'SMB',
    '465': 'SMTPS',
    '514': 'Syslog',
    '515': 'LPD/LPR',
```

```
    '389': 'LDAP',
    '443': 'HTTPS',
    '445': 'SMB',
    '465': 'SMTPS',
    '514': 'Syslog',
    '515': 'LPD/LPR',
    '587': 'SMTP',
    '902': 'VMware',
    '993': 'IMAPS',
    '995': 'POP3S',
    '1433': 'Microsoft SQL Server',
    '1521': 'Oracle Database',
    '3306': 'MySQL',
    '3389': 'Remote Desktop Protocol',
    '5432': 'PostgreSQL',
    '5800': 'VNC HTTP',
    '5900': 'VNC',
    '6379': 'Redis',
    '8080': 'HTTP alternate',
    '8443': 'HTTPS alternate',
    '9000': 'Jenkins',
    '27017': 'MongoDB',
}

def scan_tcp(host, port, results, timeout):
    """
    Scan the specified TCP port on the target IP address.
    """
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.settimeout(timeout)
    try:
        result = s.connect_ex((host, port))
        if result == 0:
```

```
                'status': 'Closed',
                'banner': '',
            }
    except socket.gaierror:
        print(f'Error: Could not resolve hostname {host}')
    except socket.error as e:
        print(f'Error: Could not connect to {host}:{port} ({e})')
    finally:
        s.close()

def scan_udp(host, port, results, timeout):
    """
    Scan the specified UDP port on the target IP address.
    """
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    s.settimeout(timeout)
    try:
        s.sendto(b'PING', (host, port))
        data, addr = s.recvfrom(1024)
        service = common_ports.get(str(port), 'Unknown')
        results[port] = {
            'protocol': 'UDP',
            'service': service,
            'status': 'Open',
            'banner': get_banner(s),
        }
    except socket.gaierror:
        print(f'Error: Could not resolve hostname {host}')
    except socket.error as e:
        if 'timed out' in str(e):
            results[port] = {
                'protocol': 'UDP',
                'service
```

CHAPTER 6

SCREENSHOTS AND RESULTS

The provided code contains a port scanner that scans a range of ports on a given host.

```
Port 80: Open
Port 443: Closed
Port 22: Closed
Port 21: Closed
Port 3389: Closed
Port 1433: Closed
Port 3306: Closed
Port 8080: Open
Port 53: Closed
```

Please note that the actual output will depend on the target host and its open ports hence itwill vary depending on the system it runs on.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

In conclusion, a network port scanner is a useful tool for identifying the open ports on a networked device. By following a methodology that involves determining the target, selecting a scanning tool and technique, configuring scan options, initiating the scan, analyzing the results, and taking appropriate action, users caneffectively use a port scanner to achieve their goals.

However, to enhance the effectiveness and efficiency of a network port scanner, there are several considerations that users should keep in mind. These include:

1. Customization: Users should be able to customize the scanning options based on their specific needs and preferences. This includes the ability to select different scanning techniques, specify therange of ports to be scanned, set the timeout and retry values, and more.

2. Speed: The speed of the scanning process can significantly impact the efficiency of a port scanner. Users should look for tools that can scan large networks quickly and accurately, without compromising on accuracy.

3. Integration: Port scanners should be able to integrate with other security tools, such as intrusion detection systems and firewalls, to provide a comprehensive security solution.

4. User Interface: The user interface of the port scanner should be user-friendly and easy to navigate,even for users with limited technical knowledge.

5. Reporting: The scanner should provide detailed and comprehensive reports on the results of the scan, including the open ports, potential vulnerabilities, and recommended actions to mitigate anyrisks.

By considering these factors, users can select and use a port scanner that meets their needs and effectivelyimproves the security and functionality of their networked devices.

REFERENCES

1. Official Python Documentation:

   Threading:

   https://docs.python.org/3/library/threading.html

   Socket: https://docs.python.org/3/library/socket.html

   Argparse:

   https://docs.python.org/3/library/argparse.html

2. Real Python:

   Python Threading Tutorial: A Deeper Look:https://realpython.com/intro-to-python-threading/

   Working with Sockets in Python: https://realpython.com/python-sockets/

3. GeeksforGeeks:

   Port Scanner using Sockets in Python:https://www.geeksforgeeks.org/port-scanner-using-sockets-in-python/