COMP 7003
Introduction to Information and Network Security

Assignment-03

Design

Anmol Mittal
A01397754
February 16th, 2025
Course Reference Number (CRN): 91662

Purpose

The program is a SYN scanner that sends SYN packets to a specified target or range of targets, determines the status of each port based on the response, and categorizes the ports as open, closed, or filtered. It supports scanning single IPs, IP ranges, and subnets, and allows filtering results based on port status. The program accepts the command line argument as follows:

- sudo python3 main.py -t <target> -p <ports> --show <filter>
  -t <target>: Specifies the target(s) to scan
  -p <ports>: Specific ports or port ranges to scan
  --show <filter>: Filters the final output based on port states (open, closed, filtered)

Data Types

Arguments

Arguments	**Description**
-t <target>	specifies the target(s) to scan
-p <ports>	specific ports or port ranges to scan
--show <filter>	Filters the final output based on port states (open, closed, filtered)

Functions

Function	**Description**
main	Parses command-line arguments and starts the scan.
parse_arguments	Parses user input and extracts targets, ports, and filters.
get_local_subnet	Determines the local subnet if no target is provided.
is_valid_ip	Validates if the provided IP address is correctly formatted.
is_host_online	Sends ARP requests to check if a host is online.
syn_scan	Sends a SYN packet and interprets the response (open/closed/filtered).
scan_target	Iterates through ports for a target and categorizes their status.
format_results	Formats scan results for output.
print_section	Displays categorized scan results.

States

State	Description
START	Parse arguments and determine target(s) and ports.
SCANNING	Send SYN packets to each target and analyze responses.
ANALYZING	Categorize responses into open, closed, or filtered.
REPORTING	Format and display the results based on user filters.
FINISHED	Scan complete, program exits.

State Table

From State	To State	Function
START	SCANNING	parse_arguments
SCANNING	ANALYZING	scan_target
ANALYZING	REPORTING	format_results
REPORTING	FINISHED	print_section

Pseudocode

- is_valid_ip

Parameters

Parameter	Type	Description
ip	String	IPv4 address to validate.

Return

Value	Reason
bool	True if valid IPv4, False otherwise.

Pseudo Code

TRY to convert ip to packed binary format
IF successful: RETURN True
CATCH any errors: RETURN False

- get_local_subnet

Parameters

Parameter	Type	Description
None	-	-

Return

Value	Reason
string	Detected subnet in CIDR notation (fallback: 192.168.0.0/24).

Pseudo Code

Professor Provided Function

- is_host_online

Parameters

Parameter	Type	Description
target	String	IP address to check.

Return

Value	Reason
None	Captures packets until the stop condition is met.

Pseudo Code

Professor Provided Function

- syn_scan

Parameters

Parameter	Type	Description
target	String	IP address to scan.
port	int	TCP port to check.

Return

Value	Reason
string	"open", "closed", or "filtered" status.

Pseudo Code

```
IF scanning localhost:
  CREATE TCP socket
  TRY to connect to port
  SUCCESS: RETURN "open"
  FAILURE: RETURN "closed"
ELSE:
  CRAFT SYN packet
  SEND and wait for response
  IF no response: RETURN "filtered"
  ANALYZE TCP flags:
    SYN-ACK: Send RST, RETURN "open"
    RST: RETURN "closed"
    OTHER: RETURN "filtered"
```

- scan_target

Parameters

Parameter	Type	Description
target	String	IP address to scan.
port	list [int]	TCP port to check.
open_hosts	list[tuple]	List to store (IP, port) for open ports.
closed_hosts	list[tuple]	List to store (IP, port) for closed ports.
filtered_hosts	list[tuple]	List to store (IP, port) for filtered ports.

Return

Value	Reason
None	-

Pseudo Code

```
PRINT scanning header
CHECK if host is online
IF unreachable: PRINT and exit
FOR EACH port in port list:
  RUN syn_scan
  RECORD result in appropriate list
  PRINT status update
```

- parse_arguments

Parameters

Parameter	Type	Description
None	-	-

Return

Value	Reason
tuple	(targets, ports, show_filter) parsed from CLI.

Pseudo Code

```
SETUP argument parser
PARSE CLI inputs
IF no target specified:
  GET local subnet
PROCESS target input:
  CIDR: Expand to IP list
  RANGE: Validate and expand
  SINGLE: Validate IP
PROCESS ports input:
  COMMA/RANGE: Validate and expand
  DEFAULT: All ports (1-65535)
```

RETURN targets, ports, filter

- format_results:

Parameters

Parameter	Type	Description
hosts & port	list[tuple]	List of (IP, port) tuples.

Return

Value	Reason
list[str]	Formatted "IP:port " strings.

Pseudo Code

CREATE empty list
FOR EACH (ip, port) in hosts:
  FORMAT as "ip:port"
  ADD to list
RETURN formatted list

- print_section

Parameters

Parameter	Type	Description
title	string	Section header text.
entries	list[str]	List of results to display.

Return

Value	Reason
None	-

Pseudo Code

PRINT section header
IF entries exist:
  FOR EACH entry: PRINT as bullet point
ELSE: PRINT "None found"