

# APLICAȚIE

## TIP PORT SCANNER



Îndrumător:  
Slt. Adina VAMAN

Realizat de:  
Sd. Sg. Sorin-Ionuț MIHALI  
Sd. Sg. Maria-Emilia GRIGORE  
Grupa C113C

## Cuprins

<b>Capitol 1 - Introducere .....</b>	<b>3</b>
1.1. Scopul proiectului.....	3
1.2. Lista definițiilor .....	3
1.3. Structura documentului.....	4
 <b>Capitolul 2 – Arhitectură și componente SW .....</b>	<b>5</b>
2.1. Descrierea produsului SW.....	5
2.2. Detalierea platformei SW/HW .....	5
2.3. Actori .....	5
2.4. Arhitectura internă .....	6
 <b>Capitolul 3 – Interfața grafică .....</b>	<b>7</b>
3.1. Arbore funcționalități .....	7
3.2. Descriere funcționalități .....	8
 <b>Capitolul 4 – Testare funcționalități .....</b>	<b>9</b>

## **Capitolul 1 – Introducere**

### **1.1 Scopul proiectului**

Crearea unui produs software care să permită utilizatorului scanarea porturilor de rețea, pentru vizualizarea traficului(transmisie-recepție), dar și a stadiului în care acestea se află(open/not open).

### **1.2 Lista definițiilor**

Scanarea porturilor dintr-o rețea este o metodă prin care putem să determinăm care porturi sunt deschise și ar putea permite primirea/trimiterea de pachete în rețeaua respectivă. Această metodă presupune, de asemenea, și trimiterea de pachete pe anumite porturi pentru a vedea răspunsurile primite și a identifica, astfel, vulnerabilitățile existente (exemplu: acces neautorizat). Protocoalele folosite pentru scanarea de porturi sunt TCP si UDP.

Porturi hardware: mufe pentru periferice

Porturi software: un segment/parte/bucată de cod software căreia i-a fost asignat un alt sistem de operare sau componentă hardware pe care să lucreze față de cel/cea original/ă.

Porturi de rețea: valoare numerică asociată cu un protocol care facilitează comunicarea pentru un serviciu/funcție

### **1.3 Structura documentului**

Documentul este împărțit în trei capitole. Capitolul 1 reprezintă introducerea. Capitolul 2 prezintă actorii (tipurile de utilizatori ai aplicației) și arborele de fișiere al aplicației. Capitolul 3 prezintă interfața grafică și funcționalitățile pe care aceasta le-o oferă utilizatorului. Capitolul 4 cuprinde exemple de testare a funcționalităților aplicației.

## Capitolul 2 – Arhitectură și componente SW

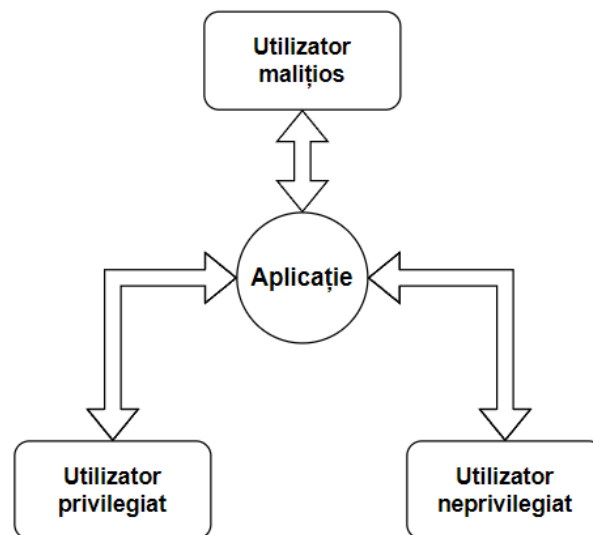
### 2.1. Descrierea produsului software

Aplicația va fi dezvoltată în limbajul de programare C/C++, iar interfața grafică va fi realizată cu ajutorul framework-ului Qt.

### 2.2. Detalierea platformei SW/HW

Produsul software este dezvoltat pentru dispozitivele pe care rulează sistemul de operare Linux/Windows. Vom utiliza platforma Qt Creator pentru crearea interfeței prietenoase cu utilizatorul și mediul de dezvoltare Microsoft Visual Studio Code.

### 2.3. Actori



# Academia Tehnică Militară “Ferdinand I”

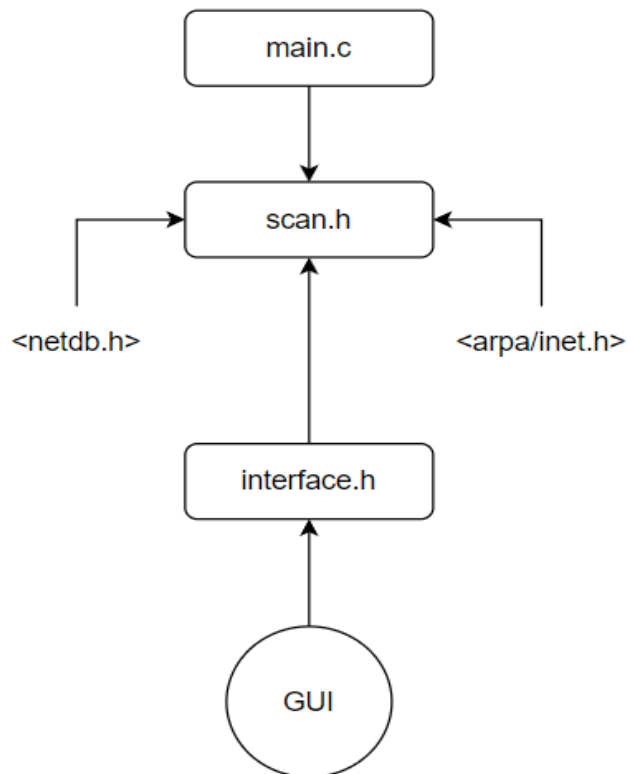
## Proiect – Proiectarea Sistemelor de Operare

Utilizator neprivilegiat- nu poate accesa opțiunea ”File”.

Utilizator privilegiat- poate accesa toate opțiunile oferite de aplicație.

Utilizator malițios- poate citi trafic de pe porturile hosturilor cu adresa IP privată.

### 2.4. Arhitectura internă



Dacă utilizatorul introduce un hostname, acesta va fi translatat într-o adresă IP, utilizând header-ul *netdb.h*, structura *hostent* și funcționalitățile definite.

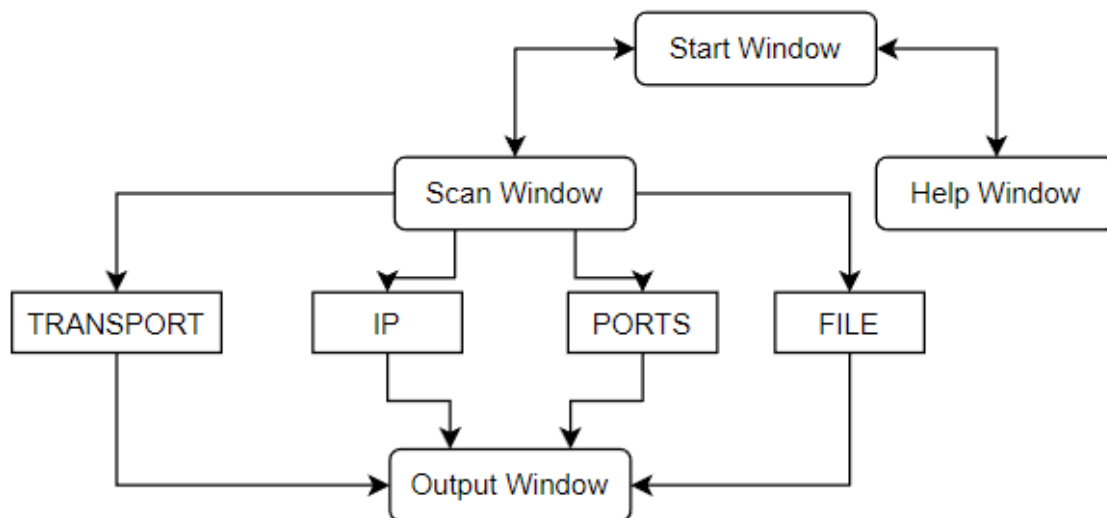
De asemenea, adresele IP vor putea fi citite dintr-un fișier dat ca argument al programului.

Din câmpul ”Ports” se va citi un port sau un interval de porturi pentru care se va verifica conectivitatea, folosind header-ul `arpa/inet.h`, structura `sockaddr_in` și funcționalitățile sale.

Dacă opțiunea ”Transport” (UDP/TCP) nu este selectată, scanarea implicită se va face pe ambele tipuri de porturi.

## Capitolul 3 – Interfața grafică

### 3.1. Arbore funcționalități aplicație



### 3.2. Descriere funcționalități

După rularea programului, se va deschide o interfață grafică, prietenoasă cu utilizatorul. Prima fereastră deschisă este ”Start Window”, oferind două funcționalități: trecerea în fereastra de scanare și un meniu help.

Buton ”Scan”- pornește aplicația care oferă utilizatorului toate funcționalitățile.  
Buton ”Help”- oferă detaliile de utilizare ale aplicației.

Fereastră ”Scan Window”:

Implicit, aplicația va scana toate porturile de la 1 la 1024.

Opțiunea ”Ports”: scanarea se va efectua pe un port/listă de porturi.

Opțiunea ”IP”: permite utilizatorului să scaneze o adresă IP individuală.

Opțiunea ”File”: se dă ca argument un nume de fișier care conține pe fiecare linie câte o adresă IP, pe care se va face scanarea.

\*nota: în cazul în care nu e specificată opțiunea ”IP” și opțiunea ”File” sau dacă fișierul dat ca argument la opțiunea ”File” este gol, va apărea o eroare și utilizatorul va avea posibilitatea de a completa parametrii din nou.

Opțiunea ”Transport”: implicit, se scanează atât porturile UDP, cât și TCP. Utilizând această opțiune, putem alege să facem scanarea doar pe unul dintre aceste tipuri.

Opțiunea ”Help”: oferă utilizatorului un man-page.

Fereastră ”Output Window”:

După scanare, în această fereastră va apărea rezultatul sau un mesaj de eroare, în cazul în care nu s-a putut realiza sau au intervenit alte erori pe parcursul execuției.



## Capitolul 4 – Testare funcționalități

### Test 1- Verificare conexiune https

*Pentru input:*

Hostname: google.com

Start port: 75

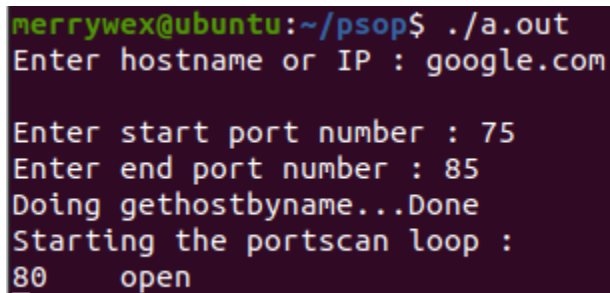
End port: 85

*Output:*

Translating hostname...Done

Scanning...

80    open



```
merrywex@ubuntu:~/psop$ ./a.out
Enter hostname or IP : google.com

Enter start port number : 75
Enter end port number : 85
Doing gethostbyname...Done
Starting the portscan loop :
80    open
█
```

### Test 2- Verificare conexiune https, cu hostname invalid

*Pentru input:*

Hostname: fhgchc

Start port: 0

End port: 0

*Output:*

Host name lookup failure

Academia Tehnică Militară “Ferdinand I”  
Proiect – Proiectarea Sistemelor de Operare

```
merrywex@ubuntu:~/psop$ ./a.out
Enter hostname or IP : fhgchc

Enter start port number : 0
Enter end port number : 0
fhgchc: Host name lookup failure
```

**Test 3- Verificare conexiune https, cu porturi invalide**

*Pentru input:*

Hostname: google.com

Start port: -8

End port: -5

*Output:*

Doing gethostbyname...Done

Starting the portscan loop :

```
merrywex@ubuntu:~/psop$ ./a.out
Enter hostname or IP : google.com

Enter start port number : -8
Enter end port number : -5
Doing gethostbyname...Done
Starting the portscan loop :
merrywex@ubuntu:~/psop$
```