

### Etapa 1

#### 1. Analiza codului sursă

Mirai este compus din cinci componente esențiale, fiecare jucând un rol distinct în funcționarea și expansiunea botnetului. Codul sursă, conform structurii din GitHub, este organizat în patru directoare principale.

#### Componentele principale ale Botnetului Mirai

- **Virusul:** componenta responsabilă de atacurile DDoS, implementează zece vectori de atac (DNS, UDP Plain, VSE, TCP ACK, TCP SUN, HTTP, etc).
- **Serverul de Comandă și Control (C&C):** centralizează activitatea botnetului, gestionând comenzile transmise către boturi pentru a lansa atacuri DDoS. Serverul C&C este implementat în subfolderul **Mirai/Cnc** și este scris în Go.
- **Botul:** este constituit din dispozitive IoT compromise prin atacuri de tip dictionary sau brute-force, care trimit datele de autentificare și alte informații către serverul C&C. **Mirai/Bot** conține fișierele necesare pentru executarea virusului și monitorizarea activității fiecărui bot.
- **Serverul de Raportare:** menține evidența dispozitivelor compromise și colectează datele trimise de acestea pentru monitorizarea botnetului. Fișierele pentru raportare se află în subfolderul **Mirai/Tools**, care include unelte pentru criptarea stringurilor și funcționalitățile de raportare.
- **Serverul de Încărcare (Loader):** Responsabil pentru încărcarea și execuția malware-ului pe dispozitive noi descoperite ca vulnerabile. Acesta este implementat în folderul **Loader**.

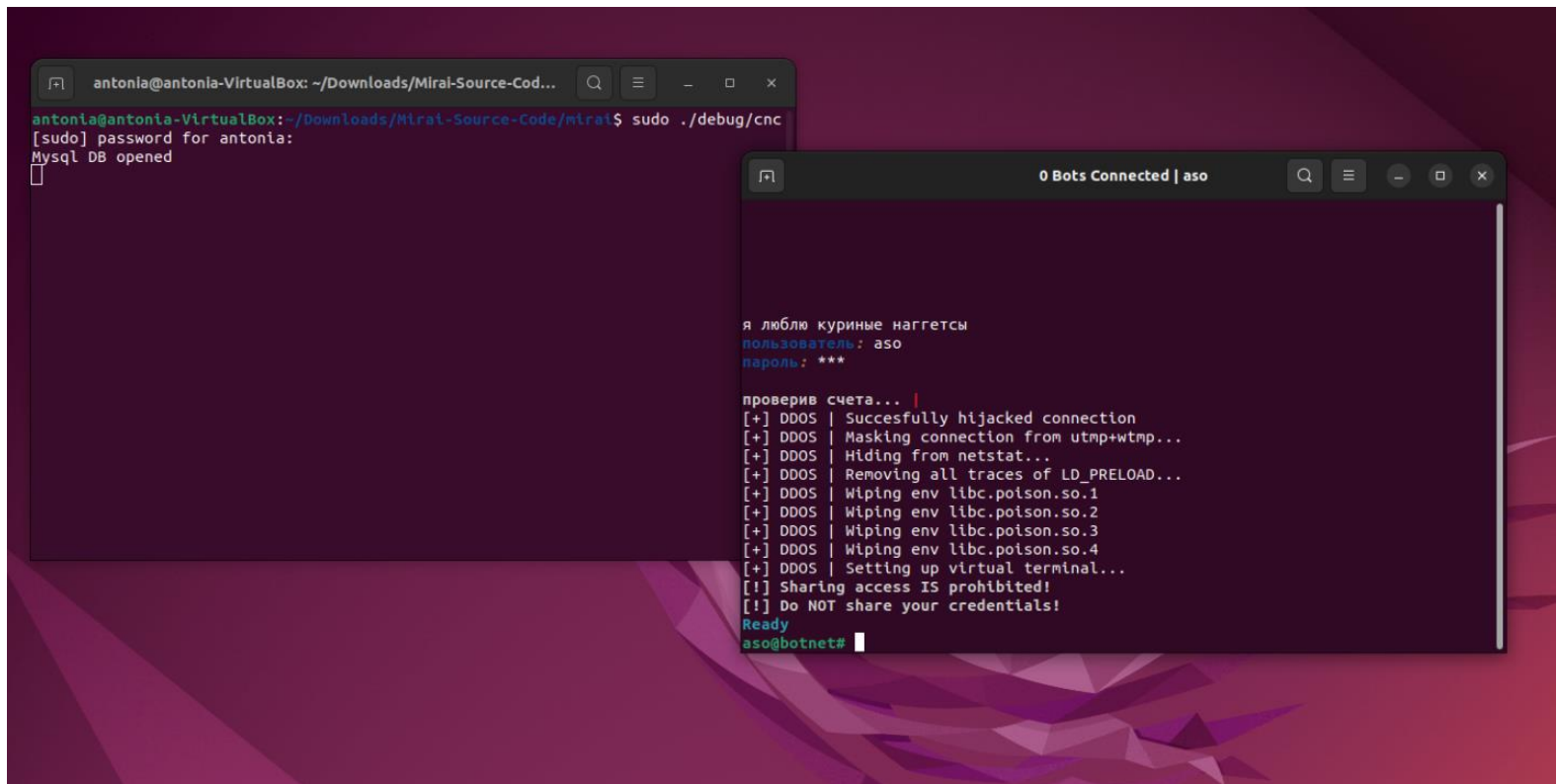
#### 2. Configurare și instalare

Am rulat scriptul furnizat în cadrul laboratorului, care a realizat instalarea pachetelor necesare: git (pentru clonarea codului sursă Mirai de pe GitHub), build-essential (set de unelte de compilare), golang-go (pentru compilarea componentelor scrise în Go), mariadb-server și mariadb-client (pentru configurarea bazei de date necesare funcționării Mirai). S-a creat un utilizator dedicat (asodb) și o bază de date (mirai) cu tabelele history, users, și whitelist, necesare gestionării istoricului și datelor utilizatorilor.

### 3. Testare

Pentru testarea aplicației, am pornit serverul Command & Control (C&C) utilizând comanda **`sudo ./debug/cnc`**. Apoi, am încercat conectarea la acesta prin comanda **`sudo telnet 127.0.0.1`**, autentificându-mă cu **`aso`** ca utilizator și **`aso`** ca parolă.

Aceste comenzi au permis verificarea funcționării corecte a serverului C&C și stabilirea unei conexiuni Telnet, confirmând capacitatea aplicației de a gestiona conexiuni și de a procesa autentificarea.



```
antonia@antonia-VirtualBox: ~/Downloads/Miral-Source-Cod...  
antonia@antonia-VirtualBox:~/Downloads/Miral-Source-Code/miral$ sudo ./debug/cnc  
[sudo] password for antonia:  
Mysql DB opened  
[  
  
0 Bots Connected | aso  
  
я люблю куриные наггетсы  
пользователь: aso  
пароль: ***  
  
проверив счета... |  
[+] DDOS | Successfully hijacked connection  
[+] DDOS | Masking connection from utmp+wtmp...  
[+] DDOS | Hiding from netstat...  
[+] DDOS | Removing all traces of LD_PRELOAD...  
[+] DDOS | Wiping env libc.poisn.so.1  
[+] DDOS | Wiping env libc.poisn.so.2  
[+] DDOS | Wiping env libc.poisn.so.3  
[+] DDOS | Wiping env libc.poisn.so.4  
[+] DDOS | Setting up virtual terminal...  
[!] Sharing access IS prohibited!  
[!] Do NOT share your credentials!  
Ready  
aso@botnet#
```