## **Introduzione**

**Packet Sniffer** è un programma sviluppato per monitorare e analizzare il traffico di rete. Consente agli utenti di catturare pacchetti di rete da un indirizzo IP specificato, o da tutta la rete, monitorare specifici protocolli di rete come ARP, DNS, TCP, UDP e RAW, e salvare i dati in un file CSV per una successiva analisi. L'interfaccia grafica del programma è progettata per essere semplice e intuitiva, anche per gli utenti non esperti.

## **Funzionalità**

### **Login**

Al primo avvio del programma, l'utente viene presentata una finestra di login. Solo gli utenti che inseriscono le credenziali corrette possono accedere all'applicazione.

* **Username**: admin
* **Password**: securepassword

### **Sniffing dei Pacchetti**

L'utente può configurare il programma per monitorare specifici indirizzi IP o tutta la rete, con la possibilità di scegliere un intervallo di porte o una singola porta da monitorare. Il programma cattura pacchetti di rete in tempo reale.

* **Supporto per i seguenti protocolli**:
  + ARP
  + DNS
  + TCP
  + UDP
  + RAW (dati grezzi)

### **Visualizzazione in Tempo Reale**

Ogni pacchetto catturato viene visualizzato in tempo reale nella finestra principale dell'applicazione, con informazioni dettagliate come:

* Timestamp
* IP di origine e destinazione
* Porta
* Dati (se presenti)

### **Esportazione in CSV**

I pacchetti catturati possono essere esportati in un file CSV. Questa funzionalità permette all'utente di archiviare i dati per una successiva analisi, creare report o fare diagnosi più approfondite.

* **Dati esportati**:
  + Timestamp
  + IP di origine
  + Porta
  + Dati (in formato esadecimale)

## **Tecnologie Utilizzate**

* **Scapy**: Una libreria potente di Python per l'analisi e la manipolazione di pacchetti di rete. Utilizzata per sniffare i pacchetti e decodificarli.
* **Tkinter**: Utilizzato per creare l'interfaccia grafica del programma, inclusi i campi di input, pulsanti e la visualizzazione dei pacchetti in tempo reale.
* **CSV**: Il formato di esportazione dei dati catturati. Viene utilizzato per creare file che possono essere analizzati successivamente.
* **Logging**: La libreria di logging di Python è utilizzata per registrare eventi significativi, come errori o attività di cattura pacchetti.

## **Installazione**

### **Requisiti**

* **Python**: La versione consigliata di Python è la 3.x.
* **Librerie necessarie**:
  + Scapy
  + Tkinter
  + CSV (preinstallato con Python)
  + Logging (preinstallato con Python)

### **Installazione**

1. Assicurati di avere Python 3.x installato.

Installa Scapy utilizzando il comando:  
bash  
Copia codice  
pip install scapy

Se non hai Tkinter, installalo con:  
bash  
Copia codice  
pip install tk

1. Scarica il codice del programma e salvalo nella tua directory di lavoro.

Avvia il programma con il comando:  
bash  
Copia codice  
python packet\_sniffer.py

## **Flusso di Lavoro**

1. **Login**:
   * L'utente deve fornire le credenziali per accedere al programma.
2. **Configurazione**:
   * L'utente inserisce l'indirizzo IP di destinazione e specifica l'intervallo di porte da monitorare.
3. **Avvio dello Sniffing**:
   * Una volta configurato, l'utente clicca su "Start" per iniziare a catturare i pacchetti di rete.
4. **Monitoraggio in Tempo Reale**:
   * I pacchetti vengono visualizzati nella finestra principale. Ogni pacchetto è analizzato e le informazioni vengono mostrate all'utente.
5. **Esportazione dei Dati**:
   * Una volta fermato lo sniffing, l'utente può esportare i dati catturati in un file CSV per ulteriori analisi.

## **Esempio di Utilizzo**

1. Avvia il programma e accedi con le credenziali di default: **Username**: admin, **Password**: securepassword.
2. Inserisci un indirizzo IP di destinazione (o lascia "0.0.0.0" per monitorare tutta la rete).
3. Specifica un intervallo di porte o una porta singola (ad esempio, 80 per HTTP).
4. Premi il pulsante "Start" per avviare la cattura dei pacchetti.
5. I pacchetti catturati vengono visualizzati in tempo reale nella finestra principale.
6. Al termine, premi "Stop" e poi "Esporta in CSV" per salvare i dati catturati.

## **Benefici per l'Utente**

* **Monitoraggio del Traffico di Rete**: Utile per analizzare il traffico e identificare problemi di rete o comportamenti sospetti.
* **Facilità di Uso**: L'interfaccia grafica è semplice, rendendo il programma accessibile anche a chi non ha esperienza con il networking.
* **Analisi Dettagliata**: La possibilità di visualizzare pacchetti specifici come ARP, DNS, TCP, UDP e RAW fornisce una comprensione approfondita del traffico di rete.
* **Esportazione Dati**: I dati possono essere esportati in formato CSV per una successiva analisi offline.

## **Possibili Miglioramenti**

* **Supporto per Altri Protocolli**: Implementazione di ulteriori protocolli di rete, come ICMP o SNMP.
* **Filtri Avanzati**: Aggiungere funzionalità per filtrare pacchetti in base a parametri più specifici, come il contenuto del payload o l'indirizzo MAC.
* **Visualizzazione Grafica**: Integrare strumenti di visualizzazione per rappresentare graficamente i dati catturati, come ad esempio un grafico delle connessioni TCP attive.
* **HTTPS**: La decodifica dei pacchetti HTTPS è complessa e richiede l'accesso ai dati cifrati. Senza un certificato privato o strumenti specializzati, non è possibile decodificare HTTPS a meno che non si abbia un MITM (Man-In-The-Middle) setup.

## **Conclusioni**

**Packet Sniffer** è uno strumento potente e versatile per il monitoraggio del traffico di rete. Con la sua interfaccia semplice e le funzionalità avanzate, è utile per professionisti della sicurezza di rete e per chiunque desideri comprendere meglio come funziona la rete. Con ulteriori sviluppi, come il supporto per più protocolli e filtri avanzati, il programma potrebbe diventare ancora più completo e personalizzabile.