Progetto Gestione di Reti A.A. 2024/2025

Nome: Diego

Cognome: Milletti

E-mail: d.milletti@studenti.unipi.it

Introduzione

Il presente documento descrive il funzionamento di un progetto sviluppato per il rilevamento di anomalie nella dimensione della finestra TCP nei flussi di rete. Lo strumento, realizzato in linguaggio C, analizza file ".pcap" ed è in grado di individuare comportamenti anomali legati alla trasmissione TCP. I risultati delle analisi possono essere visualizzati direttamente sul terminale oppure esportati in un report testuale in formato ".txt".

Prerequisiti

- Avere installato il compilatore **gcc (sudo apt-get install gcc)**
- Aver installato la libreria **libpcap (sudo apt-get install libpcap-dev)**

Descrizione

Lo scopo principale del programma è analizzare i flussi TCP in modo **bidirezionale**, monitorando in particolare la **dimensione della finestra TCP** in ciascuna direzione. A tal fine sono state definite specifiche strutture dati:

- **FlowKey**: struttura utilizzata per identificare univocamente un flusso TCP. Contiene gli indirizzi IP e le porte di origine e destinazione, oltre a un intero is_ipv6 che permette di distinguere tra flussi IPv4 e IPv6.
- **FlowData**: struttura che rappresenta un intero flusso TCP. Include un FlowKey, un contatore di pacchetti (packet_count), un contatore delle finestre TCP pari a zero (zero_window_count) e una struttura win_stats per raccogliere i valori assunti dalla finestra TCP durante la comunicazione.

Il progetto è stato sviluppato a partire dal programma pcount.c, integrando diverse funzionalità provenienti dal software **nDPI** al fine di ottimizzare l'uso della memoria. In particolare, è stato utilizzato il puntatore "struct ndpi_analyze_struct *win_stats" per memorizzare i valori della finestra TCP per ogni pacchetto. Questo ha permesso di calcolare alcune statistiche significative, come la **deviazione standard e il coefficiente di variazione**, utile per l'individuazione di alcune anomalie.

Tra i criteri di rilevamento, il programma segnala anche la presenza di **finestre TCP di dimensione uguale a zero**. Se almeno un'occorrenza viene rilevata durante l'analisi, verrà generato un avviso, visualizzato sul terminale oppure scritto nel report in formato ".txt" (se è stato specificato il parametro "-r").

Per una corretta gestione dei pacchetti e flussi TCP, è stata modificata la funzione "dummyprocesssPacket" e sono state aggiunte funzioni ausiliari come:

- **Comparekeys**: confronta due strutture "Flowkey" per verificare se appartengono allo stesso flusso (nella stessa direzione).
- **Match_flow_bidir**: determina se due "Flowkey" rappresentano le due direzioni dello stesso flusso bidirezionale.

- **Get_or_create_flow**: crea dinamicamente un nuovo flusso se non ancora presente, oppure restituisce il flusso esistente alla “Flowkey” in ingresso.
- **Cleanup_flows**: libera tutta la memoria allocata dinamicamente.

Infine, per effettuare la vera e propria analisi, è stata implementata la funzione **flow_analysis,** che esamina la dimensione della finestra TCP per ciascun flusso in entrambe le direzioni. La funzione rileva eventuali anomalie e stampa diverse informazioni utili, come il numero totale di pacchetti, la dimensione minima, massima e media della finestra TCP.

Come accennato in precedenza, per la segnalazione di anomalie viene utilizzata **la deviazione standard**. Dopo aver fatto diverse catture e analisi, **ho individuato una soglia** oltre la quale la deviazione è considerata anomala, ma **solo se supera anche il 70% della media (ovvero il coefficiente di variazione)** e se il numero totale di pacchetti di tale direzione del flusso analizzati **sono maggiori di 15.** Questo triplo controllo è stato introdotto per ridurre i falsi positivi e garantire un’analisi più affidabile e coerente con il comportamento reale dei flussi TCP.

Istruzioni per l’uso

1. **Compilazione**:
   Utilizzare il comando **make** per compilare automaticamente il codice.

2. **Esecuzione**:
   Eseguire il programma usando il comando:
   “**sudo ./pcount -i <file.pcap> -v 1”**
   Il filtro TCP verrà applicato di default.
   Per analizzare grandi file .pcap, è consigliato salvare l’output su un file report, aggiungendo il parametro “**-r <file_report.txt>”.**

3. **Pulizia**:
   Utilizzare il comando **make clean** per rimuovere i file eseguibili generati.

Test

Ho effettuato vari test per controllare il corretto funzionamento del programma:

- È stato utilizzato il tool **Valgrind** per controllare l’eventuale presenza di perdite di memoria. Lo screenshot riportato di seguito mostra che l’unica perdita rilevata è dovuta a una funzione interna della libreria **libpcap**, non dipendente dal programma sviluppato.

```
------------------------------------------------------------
==112723==
==112723== HEAP SUMMARY:
==112723==     in use at exit: 96 bytes in 1 blocks
==112723==   total heap usage: 59 allocs, 58 frees, 42,865 bytes allocated
==112723==
==112723== 96 bytes in 1 blocks are definitely lost in loss record 1 of 1
==112723==    at 0x484DA83: calloc (in /usr/libexec/valgrind/vgpreload_memcheck-amd64-linux.so)
==112723==    by 0x48895BD: ??? (in /usr/lib/x86_64-linux-gnu/libpcap.so.1.10.1)
==112723==    by 0x487BB93: pcap_compile (in /usr/lib/x86_64-linux-gnu/libpcap.so.1.10.1)
==112723==    by 0x10D2DC: main (pcount.c:795)
==112723==
==112723== LEAK SUMMARY:
==112723==    definitely lost: 96 bytes in 1 blocks
==112723==    indirectly lost: 0 bytes in 0 blocks
==112723==      possibly lost: 0 bytes in 0 blocks
==112723==    still reachable: 0 bytes in 0 blocks
==112723==         suppressed: 0 bytes in 0 blocks
==112723==
==112723== For lists of detected and suppressed errors, rerun with: -s
==112723== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
==112723== could not unlink /tmp/vgdb-pipe-from-vgdb-to-112723-by-root-on-???
==112723== could not unlink /tmp/vgdb-pipe-to-vgdb-from-112723-by-root-on-???
==112723== could not unlink /tmp/vgdb-pipe-shared-mem-vgdb-112723-by-root-on-???
diego@Diego:/mnt/c/Users/dmill/OneDrive/Desktop/Progetto_GR$
```

- Lo screenshot seguente rappresenta l'analisi senza anomalie di un flusso TCP presente nel file **gnutella.pcap** (incluso nel progetto)

```
------------------------------------------------------------------

╔══════════════════════════════════════════════════════════════╗
║ Flusso 10.0.2.15:50300 ⇄ 188.61.52.183:11852                 ║
╚══════════════════════════════════════════════════════════════╝

→ Direzione 10.0.2.15 → 188.61.52.183
  Pacchetti: 66
  Finestra [min: 62805, max: 64240, media: 63500.65]

→ Direzione 188.61.52.183 → 10.0.2.15
  Pacchetti: 69
  Finestra [min: 65535, max: 65535, media: 65535.00]
------------------------------------------------------------------
```

- Nel file **443-firefox.pcap** (anch'esso incluso), l'analisi ha correttamente individuato un'anomalia: sono state rilevate 4 finestre TCP con dimensione uguale a zero, segnalate a terminale.

```
======= ANALISI FLUSSI TCP =======
╔═══════════════════════════════════════════════════════════╗
Flusso 192.168.1.13:53096 ⇄ 178.62.197.130:443
╚═══════════════════════════════════════════════════════════╝
→ Direzione 192.168.1.13 → 178.62.197.130
  Pacchetti: 316
  Finestra [min: 0, max: 65535, media: 4731.26]
  ⚠ Anomalia ⚠ : dimensione finestra TCP uguale a zero (4 volte)

→ Direzione 178.62.197.130 → 192.168.1.13
  Pacchetti: 351
  Finestra [min: 500, max: 65160, media: 685.42]
------------------------------------------------------------
diego@Diego:/mnt/c/Users/dmill/OneDrive/Desktop/Progetto_GR$
```

- Questo screenshot mostra l'analisi di un flusso IPv6 contenuto nel file **http_ipv6.pcap** (anch'esso allegato). In entrambe le direzioni del flusso sono stati analizzati meno di 15 pacchetti, motivo per cui il programma segnala che i dati non sono sufficienti per un'analisi affidabile. Ho ritenuto utile stampare comunque queste informazioni per evidenziare i limiti dei dati disponibili e mantenere la trasparenza nell'output.

```
------------------------------------------------------------
╔═══════════════════════════════════════════════════════════╗
Flusso 2a00:d40:1:3:7aac:c0ff:fea7:d4c:37506 ⇄ 2a03:b0c0:3:d0::70:1001:443
╚═══════════════════════════════════════════════════════════╝
→ Direzione 2a00:d40:1:3:7aac:c0ff:fea7:d4c → 2a03:b0c0:3:d0::70:1001
[!] Troppi pochi pacchetti per un'analisi affidabile [!]
  Pacchetti: 14
  Finestra [min: 225, max: 28800, media: 2363.64]

→ Direzione 2a03:b0c0:3:d0::70:1001 → 2a00:d40:1:3:7aac:c0ff:fea7:d4c
[!] Troppi pochi pacchetti per un'analisi affidabile [!]
  Pacchetti: 12
  Finestra [min: 116, max: 28560, media: 2491.83]
------------------------------------------------------------
```

- Gli ultimi 3 screenshot si riferiscono all'analisi di un file ".pcap" di grandi dimensioni, **Thursday-WorkingHours.pcap** proveniente dal dataset CIC-IDS 2017.
  In questo caso, sono state rilevate diverse anomalie, tra cui:
  - Elevata deviazione standard della finestra TCP,
  - Coefficiente di variazione elevato,
  - Presenza di molte finestre TCP con valore zero.

  Il file **Thursday-WorkingHours.pcap** non è incluso nel progetto per motivi di spazio, ma è disponibile al seguente link:
  https://www.unb.ca/cic/datasets/ids-2017.html

```
------------------------------------------------------------
╔══════════════════════════════════════════════════════════╗
║Flusso 192.168.10.17:47724 ⇄ 61.184.116.79:443            ║
╚══════════════════════════════════════════════════════════╝
→ Direzione 192.168.10.17 → 61.184.116.79
  Pacchetti: 84
  Finestra [min: 0, max: 29200, media: 2497.10]
  ⚠ Anomalia ⚠ : dimensione finestra TCP uguale a zero (59 volte)

→ Direzione 61.184.116.79 → 192.168.10.17
  Pacchetti: 71
  Finestra [min: 123, max: 14600, media: 537.68]
------------------------------------------------------------
```

```
------------------------------------------------------------
╔══════════════════════════════════════════════════════════╗
║Flusso 192.168.10.17:40665 ⇄ 106.122.252.16:443           ║
╚══════════════════════════════════════════════════════════╝
→ Direzione 192.168.10.17 → 106.122.252.16
  Pacchetti: 29
  Finestra [min: 229, max: 29200, media: 1334.62]

→ Direzione 106.122.252.16 → 192.168.10.17
  Pacchetti: 29
  Finestra [min: 0, max: 65535, media: 8017.07]
  ⚠ Anomalia ⚠ : dimensione finestra TCP uguale a zero (1 volte)
  ⚠ Anomalia ⚠ : Forti oscillazioni della finestra TCP (dev_std=19539.66)
------------------------------------------------------------
```

```
------------------------------------------------------------
╔══════════════════════════════════════════════════════════╗
║Flusso 192.168.10.14:60915 ⇄ 192.82.242.28:80             ║
╚══════════════════════════════════════════════════════════╝
→ Direzione 192.168.10.14 → 192.82.242.28
  Pacchetti: 22
  Finestra [min: 8192, max: 64240, media: 61570.36]

→ Direzione 192.82.242.28 → 192.168.10.14
  Pacchetti: 28
  Finestra [min: 8190, max: 64240, media: 25858.93]
  ⚠ Anomalia ⚠ : Forti oscillazioni della finestra TCP (dev_std=21465.44)
------------------------------------------------------------
```