Gestione di Reti a.a 2024/2025

Relazione progetto

Candidati: Ferrante Luca [n° 654430]
Data consegna: 06/06/2025

Descrizione

Questo tool è progettato per recuperare informazioni su dispositivi di rete remoti tramite il protocollo SNMP (Simple Network Management Protocol), con l'ausilio di Shodan per l'individuazione preliminare degli host esposti e accessibili.

1. **Scansione e Identificazione Dispositivi SNMP**:
    - Attraverso shodan.io è possibile identificare dispositivi remoti accessibili pubblicamente che espongono il servizio SNMP.
    - Una volta individuato un target e passato come parametro al tool, viene stabilita una connessione SNMP per l'interrogazione dei dati di traffico.
2. **Raccolta Dati SNMP**:
    - Estrae informazioni da OID rilevanti (nel mio caso ifOutOctets).
    - I dati raccolti vengono normalizzati e strutturati per l'analisi (gestendo anche un possibile overflow dei contatori a 32 bit).
3. **Algoritmo di Similarità del Traffico**:
    - Applica un algoritmo di similarità per confrontare i pattern di traffico tra le diverse porte.
    - Identifica le interfacce che mostrano comportamenti simili nel tempo.
4. **Output**:
    - Fornisce un report con:
        - Lista delle interfacce monitorate.
        - Coppie di porte con traffico simile.

Struttura e logica progetto

Il mio progetto è basato su un file principale: "***port_similarity.c***" che contiene tutta la logica del tool. Ho usufruito di tre librerie per implementare il tutto:

- ➔ ***net-snmp***: usata per inizializzare la sessione SNMP e costruire gli SNMP PDU per fare polling al SNMP agent e recuperare le informazioni su *ifOutOctets*;
- ➔ ***rrdtool***: usato per immagazzinare le informazioni raccolte in un database a sequenze temporali. Invece di creare un unico file .rrd contenente molteplici Data Source (DS) — uno per ciascuna interfaccia rilevata tramite SNMP sull'agent remoto — ho preferito adottare un approccio modulare, generando un file .rrd separato per ogni interfaccia. Questo mi ha consentito una gestione più semplice e una maggiore flessibilità nell'analisi e visualizzazione dei dati relativi alle singole porte.
- ➔ ***ndpi***: usata principalmente per applicare l'algoritmo di similarità tra bin (distanza di Euclide) (https://github.com/ntop/nDPI).

È presente anche un altro file "***gentest.sh***" che rappresenta un piccolo script per generare 5 interfacce di test.

Come funziona?

```
Usage: port_similarity -h <hostname> | -l | -c <basedir> [-a <alpha>][-t <threshold>]
                                [-e <end>][-s <start>][-S <step][-v][-z]
-e <end>       | RRD end time. Default now
-s <start>     | RRD start time. Default now-1d
-h <hostname>  | Hostname or IP address of the agent
-c <basedir>   | Analyze similarity in RRD files in specified basedir (without SNMP polling)-t <threshold> | Similarity threshol
d. Default 100 (0 == alike)
-l             | Use localhost instead of specifying hostname (127.0.0.1)
-S <step>      | Set RRD step. Valid range >0. Default 60
-v             | Verbose
-z             | Skip zero RRDs during comparison


Example: port_similarity -l


Goal: find similar port on localhost SNMP manager
```

A seconda delle opzioni specificate è possibile avere comportamenti diversi del programma. I principali sono:

- *-l* => con cui è possibile interrogare un SNMP agent installato sulla propria macchina;
- *-h* => con cui è possibile interrogare un SNMP agent installato su una macchina remota;
- *-c* => con cui è possibile passare una directory contenente dei file .rrd e applicare direttamente l'analisi di similarità (applica *rrd_similarity* di nDPI).

Per le prime due opzioni, all'avvio, il tool avvierà un thread che farà **polling** al SNMP agent scelto. Per avere dati sostanziosi è consigliato aspettare 20/30 minuti in modo tale da poter popolare i file rrd. Per interrompere il polling basta mandare un segnale (CTRL-C) e successivamente viene applicato l'algoritmo di similarità sui valori registrati e viene stampato un report (più o meno dettagliato a seconda delle opzioni passati all'avvio).

```
Found 0 (0.000 %) similar RRDs / 78 zero alike RRDs [num_rrds: 22]
rrds/200.58.174.254/1.rrd       510584.3        2977192.5
rrds/200.58.174.254/2.rrd       0.0     0.0
rrds/200.58.174.254/3.rrd       29497450.0      27115122.0
rrds/200.58.174.254/4.rrd       0.0     0.0
rrds/200.58.174.254/5.rrd       1252371.9       7302520.5
rrds/200.58.174.254/6.rrd       0.0     0.0
rrds/200.58.174.254/7.rrd       0.0     0.0
rrds/200.58.174.254/8.rrd       0.0     0.0
rrds/200.58.174.254/9.rrd       24892484.0      26960858.0
rrds/200.58.174.254/10.rrd      1077047.4       6280211.0
rrds/200.58.174.254/11.rrd      24583528.0      27689696.0
rrds/200.58.174.254/12.rrd      210677.4        1228449.9
rrds/200.58.174.254/13.rrd      278435.6        1623544.6
rrds/200.58.174.254/14.rrd      0.0     0.0
rrds/200.58.174.254/15.rrd      1127404.4       6573840.5
rrds/200.58.174.254/16.rrd      0.0     0.0
rrds/200.58.174.254/17.rrd      0.0     0.0
rrds/200.58.174.254/18.rrd      0.0     0.0
rrds/200.58.174.254/19.rrd      0.0     0.0
rrds/200.58.174.254/20.rrd      0.0     0.0
rrds/200.58.174.254/21.rrd      0.0     0.0
rrds/200.58.174.254/22.rrd      0.0     0.0
```

Istruzioni compilazione e esecuzione

Attraverso il Makefile che ho creato è possibile compilare il tutto attraverso il comando ***make***: viene prima compilata la libreria ndpi, poi vengono create le cartelle */test/* e */rrds/* (rispettivamente la prima contiene i file rrd per il test, la seconda contiene i file rrd degli host contattati) e infine viene compilato il tool.

Per quanto riguarda l'esecuzione:

- ***make run-remote***: permette di eseguire il tool con un SNMP agent già individuato su shodan.io;
- ***make run-local***: permette di eseguire il tool con un SNMP agent installato sulla propria macchina;
- ***make run-test***: permette di eseguire il tool (solo algoritmo similarità) sulla cartella *test/*. (**NOTA**: bisogna prima eseguire *./gentest.sh* per generare i file rrd in quella cartella)

Per un eventuale pulizia dei file .o e alto basta digitare ***make clean***.