MUD generator from pcap

progetto per il corso “Gestione di Rete”
A.A. 2018/2019

Lorenzo Boccolini, matricola 544098

Indice

1 Introduzione

“MUD generator from pcap” è un tool sviluppato in Python che consente di generare un file MUD (https://developer.cisco.com/site/mud/), da un determinato file pcap preso in input.

Il programma consente di specificare l’indirizzo ip rispetto al quale analizzare i pacchetti nel file pcap (così da poter realizzare il file MUD da un host diverso rispetto all’ip specificato); se nessun indirizzo è specificato, viene utilizzato l’indirizzo ip corrente del local host.

Il file generato contiene gli indirizzi,i numeri di porta e il protocollo (TCP / UDP) di tutti gli host che hanno contattato o sono stati contattati dal dispositivo in questione, separati per traffico entrante o uscente.
In questo modo viene creato “*un profilo*” del dispositivo in questione, utile al fine di limitarne il traffico ai soli host specificati nel file MUD generato, in un’ottica di fornire un meccanismo di sicurezza contro comunicazioni indesiderate per il dispositivo.

2 Struttura del codice

Il tool è organizzato secondo le seguenti 3 classi.

Address.py

La classe “Address.py”, come suggerisce il nome, è usata per rappresentare ogni indirizzo estratto nella fase di analisi del file pcap.
Ogni indirizzo viene rappresentato logicamente come una tripla:

- indirizzo ip
- protocollo
- numeri di porta associati

La classe fornisce una serie di metodi utili al confronto e alla gestione degli indirizzi.

AddressTree.py

La classe “AddressTree.py” rappresenta una struttura ad albero, per la memorizzazione dei vari indirizzi estratti dal file pcap.
La struttura ad albero binario rende le operazioni di inserimento e ricerca degli indirizzi molto efficienti in termini di tempo.

Creator.py

La classe “Creator.py” è la classe principale del tool.
Attraverso la libreria “pyshark”, elabora i pacchetti contenuti nel file pcap in input, separando i pacchetti ricevuti e inviati dall’indirizzo ip specificato.
L’analisi dei pacchetti si limita al traffico TCP e UDP, applicando un *display_filter* al file pcap.

Nell’analisi del file pcap, viene impostato il flag “*keep_packets*” a false, che consente di tenere in memoria un solo pacchetto alla volta, portando vantaggi significativi in termini di tempo risparmiato nel caso in cui il file pcap da esaminare sia molto grande.

Vengono creati due alberi (istanze della classe *AddressTree),* per mantenere separati gli indirizzi che hanno contattato il local host da quelli che sono stati contattati da quest’ultimo.
Per ogni pacchetto TCP o UDP che ha come mittente o destinatario l’indirizzo in questione, viene creata un’istanza della classe *Address* e aggiunta all’albero degli indirizzi corrispondente.

Per ogni indirizzo viene poi effettuato un tentativo di risoluzione,tramite un'interrogazione al dns, per provare a rimpiazzare l'indirizzo numerico con il suo nome simbolico.

Infine vengono create due liste di indirizzi tramite la funzione "*createACL*" e aggiunte (secondo il formato del MUD) al file prodotto in output in formato json, che conterrà quindi tutti gli host che hanno comunicato con il local host, ognuno con protocollo e numero di porta associato.
Nel file json prodotto vengono inoltre inserite altre informazioni, secondo il modello del MUD (url al file MUD, data e ora di ultima modifica, nome/indirizzo del local host).

Il file MUD generato è salvato nel path specificato come secondo argomento dell'esecuzione del tool, con nome composto come segue:

nome/indirizzo host + **data e ora generazione del file** + "MUD.json" .

Apponendo data e ora al nome dell'host, si possono quindi generare più file MUD differenti per uno stesso host, nella stessa directory, senza avere conflitti sul nome con cui il file viene salvato.

3 Istruzioni per l'esecuzione

Il tool va eseguito da terminale attraverso il seguente comando:

python3 Creator.py pcapfile destination_directory [ip_address]

dove:

- *pcapfile* rappresenta il path al file pcap da utilizzare per generare il MUD;
- *destination_directory* rappresenta il path in cui salvare il file MUD generato;
- *ip_address* (parametro facoltativo) fornisce l'indirizzo ip (v4) dell'host rispetto al quale generare il file MUD.

Nel caso in cui il parametro *ip_address* non venga specificato, il tool provvederà a recuperare automaticamente l'indirizzo ip del local host e ad usare tale indirizzo per generare il file MUD.

Il tool genererà eccezioni (stampando messaggi esplicativi di errore) qualora uno dei path passati come argomento risultasse errato.

Qualora l'utente avviasse il tool con un comando mal formato, o specificasse come secondo parametro l'opzione "-h" (o "--help"), verrà stampato un messaggio indicante la corretta composizione del comando da eseguire.

4 Requisiti e dipendenze

Il tool è stato interamente sviluppato e testato su *Ubuntu 18.04.3 LTS,* utilizzando la *versione 3.6.8* di *Python* .

Per il corretto funzionamento del tool, sono necessari i seguenti moduli:

- *sys , json, socket , datetime* : pacchetti di base, inclusi nella Python Standard Library;
- *pyshark* : `pip3 install pyshark`
- *dns.resolver* :
```
git clone https://github.com/rthalley/dnspython
cd dnspython/
python setup.py install
```