Guido Ricci, settembre 2014.

Progetto di Gestione di Rete

1. Introduzione

SecRep, abbreviazione di security report, è un chisel per sysdig che si pone l'obbiettivo di facilitare la rilevazione di eseguibili malintenzionati. Per fare ciò secRep, durante l'esecuzione, intercetta tutte le operazioni che il processo, o l'eseguibile, interessato effettua su files e socket.
Durante l'esecuzione secRep mostra le liste(lru) dei files e socket utilizzati di recente, mentre al termine dell'esecuzione mostra le liste complete dei files e socket ordinate dal più al meno utilizzato.

2. Implementazione

secRep è, necessariamente, uno script lua che necessita di due parametri per essere eseguito. Il primo parametro “p” è di tipo stringa e deve essere un nome di eseguibile(senza path), o un PID, in tal caso lo script provvederà a convertire il valore da stringa a numero. Il secondo parametro “log” è di tipo stringa, se assume valore “y” o “yes” o “s” o “si” lo script produrrà un file di log degli eventi rilevanti.
Sono implementate le seguenti callback:
- on_set_arg: acquisizione e controllo della correttezza del parametro;
- on_init: notifica a sysdig dei campi necessari, e impostazione del filtro eventi;
- on_event: parsing degli eventi, salvataggio dei dati in apposite strutture, e breve output;
- on_capture_end : output completo.

Divisione in files

Per facilitarne la lettura e la comprensione lo script è stato frammentato in tre files:
secRep.lua: implementazione delle callback utilizzando le funzioni e varibili definite negli altri due files;
srOut.lua: implementazione delle funzioni dedicate alla stampa su standard output;
srLogic.lua: implementazione delle funzioni di gestione degli eventi e di archiviazione dei dati rilevanti.

Strutture dati

Lo script si appoggia su due tabelle lua: out, e logic.
La tabella out altro non contiene che le funzioni per la gestione dell'output.
La tabella logic è il core dello script in quanto ha al suo interno:

- fileTab: tabella che contiene records relativi ai files utilizzati dal processo in esame, dove tali records hanno come formato
  {key, fileType, openCounter, readCounter, writeCounter}
  nota: key è il nome del file completo di path;
- socketTab: tabella che contiene records relativi ai socket(TCP e UDP) utilizzati dal processo in esame, dove tali records hanno come formato
  {key, cSocket, protocol, role, readCounter, writeCounter}
  nota: key e cSocket sono rispettivamente la rappresentazione sotto forma di stringa dei socket del server e del client nel formato ip:porta;

- FileLruTab: tabella gestita come una lista lru che contiene i riferimenti ai logic.limit (default 6) records in fileTab dei files utilizzati di recente, usata nell'output breve;
- SocketLruTab: tabella gestita come una lista lru che contiene i riferimenti ai logic.limit (default 6) records in socketTab dei socket utilizzati di recente, usata nell'output breve;

Nota: fileLruTab e socketLruTab sono delle liste lru e non delle liste che contengono i logic.limit elementi di massimo utilizzo perchè rappresentano un buon compromesso in termini di efficienza. Inoltre i files/socket più utilizzati tenderanno a rimanere nelle liste lru.

Output breve e output completo su standard output

Lo script presenta due modi di output:

- breve: stampa dei record indicati nelle liste lru, effettuata al più una volta al secondo se ci sono nuovi eventi (durante cattura/parsing).
- completo: stampa di tutto il contenuto di fileTab e socketTab in ordine di utilizzo decrescente, effettuata come ultima operazione prima del termine dello script

Descrizione dei modi e delle scelte implementative

- liste lru e principio di località e riuso
- stampa liste ogni secondo
- scelta di scrivere più files lua, creando delle “librerie”
- scelta di scrivere un filtro con al suo interno le sysCall relative ai socket e non utilizzarle.

Output su files

File di report:

- file di nome secRep_[ora][data].txt collocato nella home dell'utilizzatore, stesso contenuto dell'output completo.

File di log:

- file di nome secRepLog_[ora][data].txt collocato nella home dell'utilizzatore, scRep produce questo file se e solo se il secondo parametro assume valore “y” o “yes” o “s” o “si”,
  contiene ogni singolo evento rilevante.