DNSmonitor

Simone Anile
simone.anile@gmail.com
Corso di Gestione di Rete
Anno Accademico 2016/2017
Dipartimento di Informatica
Università di Pisa

Indice

1. Introduzione
    1.1. Il Domain Name System 3
    1.2. Lo scopo del progetto 3
2. Funzionamento
    2.1. Funzionamento di DNSmonitor 3
        2.1.1. Configurazione DNS 3
    2.2. Monitoraggio 4
    2.3. Terminazione 4
3. Compilazione ed esecuzione
    3.1. Dipendenze e compilazione 4
    3.2. Esecuzione 5
    3.3. Parametri di avvio 6
4. Riferimenti 6

1. Introduzione

1.1. Il Domain Name System

Il DNS[1][1bis] è un sistema che ci permette di risolvere indirizzi simbolici (p.es. www.google.it) in indirizzi IP (p.es. 216.58.205.35). Quando inseriamo un indirizzo di una pagina web su un browser, questo fa una richiesta ad un server che restituirà l'indirizzo reale della pagina.
Esistono vari server DNS (p.es. Google Public DNS[2], Norton™ ConnectSafe[3]), alcuni di questi svolgono un servizio ulteriore bloccando alcuni siti nel caso contengano malware o siano dannosi per i più piccoli (per esempio OpenDNS FamilyShield[4] può essere usato per bloccare contenuti pornografici).

1.2. Lo scopo del progetto

Il programma per sistemi Linux (DNSmonitor) creato nell'ambito di questo progetto consente di monitorare la rete locale ed essere avvisati in caso in comportamenti anomali di alcuni host. Questi comportamenti possono essere sia richieste a pagine bloccate dal server DNS sia richieste ripetute a siti non esistenti. Se il programma viene avviato con una corretta configurazione per l'invio delle mail si può essere avvisati così da poter intervenire in caso di anomalie.

2. Funzionamento

2.1. Funzionamento di DNSmonitor

DNSmonitor ha due fasi principali, una di configurazione e l'altra di monitoraggio. Nella fase iniziale vengono letti gli eventuali parametri passati da riga di comando, si inizializza la cattura dei dati dalla rete (si usa la libreria pcap[5][6]), si prepara il file di log e si controlla il comportamento del DNS impostato. Se non ci sono stati problemi si passa poi alla seconda fase.

2.1.1. Configurazione DNS

Il programma deve capire come si comporta il server DNS configurato. Usando la funzione **gethostbyname**[7], per fare richieste particolari e capire il comportamento con vari indirizzi.
La funzione principale di questa fase è **dns_page** (dal file pagine.c). Vengono effettuate richieste per 4 tipi di pagine (sono anche i tipi riconosciuti dal programma):

- Sconosciute: generalmente indirizzi che non esistono;
- Errate: probabilmente indirizzi con errori di battitura;
- Malware: indirizzi a server con contenuti pericolosi;
- Pornografia: indirizzi a server con contenuto pornografico.

Gli indirizzi per le prime due categorie vengono selezionati con un algoritmo generatore (Domain generation algorithm[8]). Un esempio di risultato è:

- Sconosciute: sehccrlyfadifehn
- Errate: sehccrlyfadifehn.sehccrlyfadifehn

Gli indirizzi di Malware e Pornografia usati sono stati inseriti all'inizio del file "pagine.h". La funzione restituisce una lista con gli indirizzi delle pagine dati dal server DNS come risposta. Tra questi ci può essere anche il valore NX (errore 3 DNS, NXDOMAIN) se il server ha saputo dare alcuna risposta.

2.2. **Monitoraggio**

La seconda fase è quella che controlla i frame catturati e gestisce l'invio degli avvisi. Per dover lavorare su meno dati è stato applicato un filtro alla cattura per ricevere dati solo se sono stati inviati con il protocollo IP (Livello di rete), quello UDP (Livello di trasporto) e dalla porta 53 (DNS). Con la funzione **dns** (dal file dnsa.c) si elaborano tutti i dati ricevuti estraendo le informazioni del DNS. Si considerano solo le risposte e solo quelle che contengono un indirizzo IP (risposte di tipo A). Viene poi confrontato con quelli presenti nella lista ottenuta durante la configurazione e se c'è qualche corrispondenza si aggiunge, con un thread separato, ad una lista di host "sotto controllo" che tiene conto ti quante volte quel dispositivo ha ricevuto una risposta con una pagina di un certo tipo. Un altro thread, attivo dall'avvio della cattura, scorre la lista e controlla se c'è chi ha fatto troppe richieste nell'ultimo periodo di tempo. Una volta terminata la lista aspetta 1 secondo e ricomincia. Il periodo di tempo e il numero di richieste per mandare l'avviso si possono impostare passando gli appositi valori da riga di comando all'avvio del programma (……). In caso di attività sospetta viene scritto l'evento sul file di log e, se configurato correttamente, si invia una e-mail.

2.3. **Terminazione**

In caso di errori durante la configurazione il programma termina, altrimenti (durante il monitoraggio) viene praticamente ignorato quello su cui si sta lavorando. In ogni caso si aggiorna il file di log. La terminazione può avvenire anche per problemi con la funzione che consente di effettuare la cattura, se il problema non riguarda la mancanza di connettività (una volta che questa era già attiva), in quel caso si ritenta dopo 5 secondi.
Per poter chiudere in modo pulito il programma è stato inserito un gestore che si attiva alla ricezione del segnale 2 (CTRL+C), il quale ferma la cattura e il thread di monitoraggio.

3. Compilazione ed esecuzione

3.1. **Dipendenze e compilazione**

Il programma per funzionare ha bisogno dei seguenti pacchetti (i comandi potrebbero essere differenti):

- gcc[9] (compilatore): sudo apt-get install gcc
- make[10] (per l'utilizzo del Makefile e degli script): sudo apt-get install make
- libpcap (per la cattura): sudo apt-get install libpcap-dev
- sendemail[11] (per l'invio delle e-mail): sudo apt-get install sendemail

Il programma può essere compilato usando il compilatore:

```
gcc -o dnsa dnsa.c pagine.c utils.c -lpthread -lpcap
```

oppure con il Makefile:

```
make
```

È possibile anche usare uno degli script descritti nel prossimo paragrafo.

3.2. Esecuzione

Per eseguire il programma, se è stato compilato con il primo metodo o con il secondo:

```
sudo ./dnsa
```

per conoscere i parametri disponibili (da usare obbligatoriamente nel caso si vogliano attivare le e-mail):

```
sudo ./dnsa -h
```

Se invece si vuole usare uno script si deve prima ottenere i permessi per eseguirli:

```
chmod +x *.sh
```

e poi avviare quello scelto:

- start.sh: avvio con selezione automatica dell'interfaccia e senza configurazione e-mail;
- startv.sh: come il precedente ma con la modalità verbose;
- startm.sh: interfaccia automatica, modalità verbose e configurazione e-mail;

Dato che viene usato un file di log è necessario che nella stessa posizione dell'eseguibile sia presente una cartella chiamata "logs" (senza virgolette) e che sia possibile scriverci dentro dal programma. Se invece si usa uno script verrà creata automaticamente (se non fosse già presente), i permessi di scrittura sono comunque richiesti.

Il programma è stato testato principalmente su un Raspberry[12] Pi 3 con sistema operativo Raspbian.

Esempio di avvio con lo script startv.sh:

```
pi@raspberrypi:~/MEGA/ProgettoGR $ ./startv.sh
\rm -f *~ dnsa
gcc dnsa.c -o dnsa utils.c pagine.c -lpcap -lpthread
*************************
*  _____  _   _  _____  *
* |  __ \| \ | |/ ____| *
* | |  | |  \| | (___   *
* | |  | | |\___ \  *
* | |__| | |\  |___) | *
* |_____/|_| \_|_____/  *
*       monitor         *
*************************

1498911712 --- Avvio.
1498911713 --- Catturo da wlan0.
1498911713 --- Selezione interfaccia ✓.
1498911713 --- Configurazione...
1498911713 --- Unknown page ✓.
1498911713 --- Wrong page ✓.
1498911713 --- Malware page ✓.
1498911713 --- Pornography page ✓.
1498911713 --- |->[ NX w ]->[ NX u ]<-|
1498911713 --- Avvio pcap_loop.

1498911713 --- [ 50:C7:BF:3C:5E:88 -> B8:27:EB:9C:39:4E ]
               [ 192.168.1.1:53 -> 192.168.1.2:56007 ]
               [ id 0xf587 ] A: NXDOMAIN
1498911713 -A- Richiesta non consentita 192.168.1.2 wu.

1498911713 --- [ 50:C7:BF:3C:5E:88 -> B8:27:EB:9C:39:4E ]
               [ 192.168.1.1:53 -> 192.168.1.2:59252 ]
               [ id 0x7145 ] A: NXDOMAIN
1498911713 -A- Richiesta non consentita 192.168.1.2 wu.

1498911713 --- [ 50:C7:BF:3C:5E:88 -> B8:27:EB:9C:39:4E ]
               [ 192.168.1.1:53 -> 192.168.1.2:51452 ]
               [ id 0x38a1 ] A: NXDOMAIN
1498911713 -A- Richiesta non consentita 192.168.1.2 wu.

1498911713 --- [ 50:C7:BF:3C:5E:88 -> B8:27:EB:9C:39:4E ]
               [ 192.168.1.1:53 -> 192.168.1.2:34982 ]
               [ id 0x266a ] A: 192.64.147.177
^C
***
Ricevuto segnale 2
***
1498911718 --- Chiusura
```

3.3. Parametri di avvio

Come detto precedentemente per scoprire i paratmetri che si possono utilizzare:

```
sudo ./dnsa -h
```

si otterrà la seguente lista:

```
Parametri:
-h                  Stampa help
-v                  Modalita' verbose]
-i <interfaccia>    Interfaccia (per la cattura)*

Parametri per gli allarmi:
-a <valore>         Numero di eventi per mandare un allarme (>0)
-r <valore>         Numero di ripetizioni prima di azzerare
                    il contatore per gli allarmi (>=0)

Parametri per l'invio di mail di allarme:
-t <indirizzo>      Indirizzo mail destinatario
-f <indirizzo>      Indirizzo mail mittente
-o <oggetto>        Oggetto mail
-s <server>         Server mail in uscita
-u <utente>         Utente mail
-p <password>       Password mail

*Interfacce disponibili (-i):
 0. wlan0
 1. any
 2. lo
 3. eth0
...
```

4. Riferimenti

[1]	[it]	DNS (Wikipedia):	https://it.wikipedia.org/wiki/Domain_Name_System
[1bis]	[en]	DNS (Wikipedia):	https://en.wikipedia.org/wiki/Domain_Name_System
[2]	[en]	Google Public DNS:	https://developers.google.com/speed/public-dns/
[3]	[en]	Norton™ ConnectSafe:	https://connectsafe.norton.com/
[4]	[en]	OpenDNS FamilyShield:	https://www.opendns.com/setupguide/?url=familyshield
[5]	[en]	libpcap:	http://www.tcpdump.org/
[6]	[en]	libpcap (Wikipedia):	https://en.wikipedia.org/wiki/Pcap
[7]	[en]	gethostbyname:	https://linux.die.net/man/3/gethostbyname
[8]	[en]	DGA:	https://en.wikipedia.org/wiki/Domain_generation_algorithm
[9]	[en]	gcc:	https://www.gnu.org/software/gcc/
[10]	[en]	make:	https://www.gnu.org/software/make/
[11]	[en]	sendemail:	http://caspian.dotconf.net/menu/Software/SendEmail/
[12]	[en]	Raspberry:	https://www.raspberrypi.org/