Relazione Reti Geografiche

Packet Capture on General Purpose Systems

Egidio Mario 0522500969

Fasolino Lorenzo 0512105335

Sommario

[Definizione Esperimento 3](#_Toc54557591)

[Strategia utilizzata 4](#_Toc54557592)

[Packet Capture 5](#_Toc54557593)

[Analisi pacchetti 6](#_Toc54557594)

[Risultati 7](#_Toc54557595)

[Valori 7](#_Toc54557596)

[Analisi Pacchetti HTTP 7](#_Toc54557597)

[Siti terze parti 8](#_Toc54557598)

[Tempo di esecuzione 8](#_Toc54557599)

[Spazio occupato 9](#_Toc54557600)

[Conclusioni 10](#_Toc54557601)

# **Definizione Esperimento**

Setting

1. Installare wireshark (https://www.wireshark.org/) 2. installare Selenium (https://www.selenium.dev/)

Metodolgia

1. Definire il workload (set dei primi 50 siti Web piu` popolari, <https://www.alexa.com/topsites>)
2. Attivare wireshark e richiedere le pagine attraverso l’utilizzo di Selenium
3. Scrivere uno script che acceda al log di wireshark e calcoli:
   1. Il numero di pacchetti https
   2. il numero di siti terzi contattati

L’esperimento verrà ripetuto 5 volte aumentando di volta in volta il tempo di capture.

1. Exp1. 5 minuti di packet capture
2. Exp2. 10 minuti di packet capture
3. Exp3. 15 minuti di packet capture
4. Exp4. 20 minuti di packet capture
5. Exp5. 25 minuti di packet capture

Analisi dei risultati

1. Quanto spazio su disco serve per immagazzinare i dati dei 5 esperimenti?
2. Quanto tempo impiega lo script per ciascuno dei 5 esperimenti?

# **Strategia utilizzata**

L’esperimento consiste nel catturare ed analizzare i pacchetti http che viaggiano dopo aver invocato i 50 siti più visitati al mondo.

Come prima cosa è stato definito il workload ovvero il set dei primi 50 siti Web più popolari, (<https://www.alexa.com/topsites>).

Il passo successivo è stato quello di installare Wireshark (https://www.wireshark.org/) e Selenium (<https://www.selenium.dev/>). Dopodiché è stato realizzato uno script in Python per il lancio automatizzato dei siti web. Lo script è stato ripetuto 5 volte:

1. Exp1. 5 minuti di packet capture

2. Exp2. 10 minuti di packet capture

3. Exp3. 15 minuti di packet capture

4. Exp4. 20 minuti di packet capture

5. Exp5. 25 minuti di packet capture

Ad ogni esperimento sono stati catturati i pacchetti con l’utilizzo di Wireshark.

Dopo averli catturati sono stati esportati come file in formato JSON.

Poi è stato realizzato uno script (sempre in Python) che prende in input i file JSON esportati da Wireshark e calcola:

1. Il numero di pacchetti https
2. Il numero di siti terzi contattati
3. Quanto tempo impiega lo script per ciascuno dei 5 esperimenti

Infine, è stato calcolato quanto spazio su disco serve per immagazzinare i dati dei 5 esperimenti.

# **Packet Capture**

Per effettuare le richieste in modo automatizzato per un determinato periodo di tempo dei 50 siti più visitati al mondo è stato realizzato uno script in Python.

Questo script utilizza Selenium per effettuare il test.

Lo script riceve in input il tempo di navigazione per poi invocare i 50 siti in loop fino allo scadere del tempo.

All’avvio dello script è stato avviato ogni volta Wireshark per effettuare il packet capture. Successivamente ad ogni test sono stati esportati i log da Wireshark in formato JSON.

Durante i primi test sono stati riscontrati dei problemi con i seguenti siti:

* okezon.com
* microsoftonline.com

Quindi sono stati sostituiti per i test da:

* twitter.com
* instructure.com

# **Analisi pacchetti**

Per analizzare i pacchetti JSON esportati è stato realizzato uno script in Python che:

* prende in inputi il file da analizzare
* analizza quanti sono i pacchetti http totali
* analizza quanti sono i pacchetti https
* analizza quanti sono i pacchetti http (pacchetti http che non usano ssl)
* analizza quanti sono i pacchetti http2
* analizza il numero di siti di terze parti invocati
* calcola il tempo di esecuzione dello script

Di seguito viene riportato un esempio di output dello script invocato sul file JSON riferito al test di durata 5 minuti.

**5min\_esito.txt**

>> I pacchetti in totale sono 3231

>> I pacchetti https sono 2971

>> I pacchetti http2 sono 2483

>> I pacchetti http che non usano ssl sono 257 (260)

>> I siti di interesse visitati sono 50

>> I siti di terze parti visitati sono 151

>> Lo script ha impiegato 0.524603061 secondi per essere eseguito

**5min\_elencoSitiTerziChiamati.txt**

|  |  |  |
| --- | --- | --- |
| 1) www.completion.amazon.com  2) www.phs.tanx.com  3) www.sspapi.zenyou.71360.com  4) www.img.t.sinajs.cn  5) www.mat1.gtimg.com  6) www.gtms03.alicdn.com  7) www.wl.jd.com  8) www.n0.sinaimg.cn  9) www.static.360buyimg.com  10) www.s.amazon-adsystem.com | 11) www.s1.bdstatic.com  12) www.img20.360buyimg.com  13) www.login.live.com  14) www.s.360.cn  15) www.tracert.alipay.com  16) www.imgs.xinhuanet.com  17) www.pcookie.tmall.com  18) www.img13.360buyimg.com  19) www.ssum-sec.casalemedia.com  20) www.i.go.sohu.com | **…** |

# **Risultati**

## Valori

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Eseguiti | **Pacchetti totali** | **Pacchetti https** | **Pacchetti http** | **Pacchetti http2** | **Siti terze parti** | **Tempo esecuzione (secondi)** | **Memoria occupata (MB)** |
| **5 min** | 3231 | 2971 | 257 | 2483 | 151 | 0,52 | 42,5 |
| **10 min** | 6447 | 5931 | 513 | 4906 | 220 | 1,08 | 90,2 |
| **15 min** | 10876 | 10209 | 664 | 8570 | 241 | 1,95 | 160,7 |
| **20 min** | 13099 | 12326 | 772 | 10655 | 269 | 2,35 | 178,6 |
| **25 min** | 18390 | 17297 | 1097 | 14547 | 275 | 3,66 | 263 |

## Analisi Pacchetti HTTP

## Siti terze parti

Indica il numero di siti di terze parti invocati a partire dalle richieste dei 50 siti più visitati al mondo.

## Tempo di esecuzione

Indica il tempo impiegato dallo script per analizzare i file JSON esportati da Wireshark.

## Spazio occupato

Indica lo spazio su disco occupato dai file JSON esportati da Wireshark.

# **Conclusioni**

In conclusione, al crescere della durata dell’esperimento (quindi passando da esperimenti di 5min a esperimenti di 25min) cresce il numero di pacchetti, quindi la dimensione dei log di Wireshark esportati in JSON e anche il tempo per eseguire lo script su questi ultimi.

Inoltre, anche l’insieme dei siti di terze parti cresce al crescere del tempo dell’esperimento. (un sito di terze parti presente 2 o più volte viene conteggiato nel totale una sola volta)

In generale il numero di pacchetti https è approssimativamente il 93,65%.

Mentre il numero di pacchetti non https (pacchetti http che non usano ssl) è approssimativamente il 6,35%.

In media su 50 siti richiesti vengono fatte 231 richieste dirette a siti di terze parti.

**Link al codice sorgente:**

**https://github.com/LorenzoFasolino/AutoRequester.git**