International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 4, Issue 8, August – 2020, Pages: 323-328

APT Behavior Audit A Technical Study for Anomalies Sampling Techniques

Mourad M.H Henchiri

University of Nizwa, Oman, CEMIS, IS dept.
mourad@unizwa.edu.om

Abstract*—The discovery of essential entities behind an APT move on a platform or which play a preponderant role in the dynamics of a network flow is a crucial objective in the analysis of complex systems' security. To achieve this objective, we propose the use of flow variability and travel time measures as well as dynamic path congestion factors to enrich centrality measures in complex networks. Integrating network dynamics requires the collection and management of big data from various sources, especially from recorded datasets used upon historical APTs[1, 2, 6]. The concept of adequate data flow sampling has been integrated into the proposed smart environment. We also present the big data architecture, as well as the orchestration of software components, both for data capture, generation of congestion events, calculation of time centralities, analytics and visualization. The built solution constitutes a fundamental building block in macro-regulation and traffic monitoring in work and business environments.*

Keywords: APT, time measure, flow variability, specific software, OSI model, DFMS, capturing, sampling.

I. Introduction

Often considered one of the crucial security measures, the study of APT mobility seen in various platforms across history. This mobility was the building block behind the analysis of the spread of obfuscated backdoors, also, those analysis showed the impact of the APT mobility upon the size of operator networks; more traffic causes more chances for obfuscation and anomalies detection failure. This is a phenomena that has to be explored, and in order to carry out a scientific analysis of the phenomena occurring in our environment, we carry out all kinds of measurements[2]. Our knowledge progresses as the data collected validates certain hypotheses to the detriment of others. The tools and processes for measuring, managing and analyzing data therefore form the basis from which the understanding of our environment develops. collected, managed, analyzed, following an often standardized process. The development of the tools necessary for the implementation of a data processing process relied on different models; data pull or data push model:

Data Pull model: We need to seek for the data

Data Push model: Data gets manifested by itself.

Thus, often such a flow metaphor naturally is presented to manifest the data movement. At the time where the data pull model does not prevent nor block the consideration of data flows. Usually, with such a model, the data will have to survive path constraints as they will have to wait, at one point or another, for a utility to be consumed. In this research we are also highlighting the advances in electronics and computing that have enriched the practice of data collection and management. The constant is the increase in processing capacities, both in terms of acquisition, storage and access to data. But when information has to be extracted instantly from continuously collected data, going through a traditional DBMS can be a bottleneck. Where our main focus is on data analysis and anomalies detection based on variables study, and in order to cope with this eventuality, a new systems era has recently appeared: DFMS (data flow management systems).

This is to be considered and solved when exploring a data flow capturing and sampling system as an internally organized whole where the elements are intimately linked, which function as a single unit in relation to its environment or to other systems [4]. This makes us infront of a complexity, where data transmission is a systematic process, that should be trespassed.

II. CAPTURING TECHNIQUES

The first operation is the capture of frames. However, there are several possible cases:

- Capture over a period of time with rotation (use of a circular "buffer")
- Capture in real time manually (the user starts and stops the capture)
- Capture with automatic limitations (in time or over size …)

On the other hand, it can be useful to limit the data captured to those that are involved during the analysis:

- Filtering by protocols
- Filtering by addresses
- Limiting the size of captured packets

- Data packet size limiting:

In theory, nothing limits the size of an IP packet: you can put as much data as you want behind the header. But in practice, the size of a package is not infinite. The various network devices do not have infinite memory and have a fixed limit, which must be taken into account.

- Automatic Stop On Threshold

With referring to all available data capturing utilities, in our case of study is the wireshark, it is possible to limit the capture on 3 criteria: number of packets, size of the capture and delay in time. These three criteria can be combined. This automatic shutdown helps reduce analysis work later and does not overwrite an important event [3].

- Circular captures

This is the most interesting model, especially if it has sufficient disk space. Indeed, network problems are often fleeting and when an incident occurs, the time of activating a capture does not allow the origin of the problem to be found. On the other hand, a linear capture allows you to go back through the history of captured frames but the manipulation of a single file is often large and difficult in size. Circular capture solves these problems:

- ➔ An analysis tool would make a round capturing life cycle;
- ➔ As per the wireshark, we were able to generate a list of files of one(1) hour of captured data each, and we were able to ring the buffer used for a 24 hours life cycle(circle).

This way, circular captures never escape data that existed on the network. and this on the intention of analysing and using each of the one hour captured files within the next 24 hours from the capture time. otherwise it would be rewritten by the wireshark's new captured file.

Circular capturing is an added value to the anomalies detection industry when specifying the risk of losing a network behavior cause. It is to recule in time within the captured files and then it would be feasible to detect the anomaly cause.

- Filtering captures

Numerous are the tools that can be used for filtering the data flow, inclusive firewalls, IDSs, IPSs… Wireshark enhances the stream filtering, and if the stream to be monitored is correctly identified (server, network range, port numbers), it is possible to record only the frames that correspond to it. For that, Wireshark allows you to apply a filter on the packets to be recorded.

Here, stating the status of the wireshark network analyzer, and this is commonly seen amongst a wide range of professional network analyzers from different providers, error detection is a visualized process and for this, errors has to be eliminated from the analyzed packets and behavior study has to avoid errors consideration.

How to identify the origin of a network anomaly?

Every network administrator has heard this complaint: "the network is slow, I can't even have access to my applications to do my job!"

How can you identify if there is a network slowdown, or if the degradation comes from another base (application server, database, workstation)?

What network diagnostic tool can I use to determine if there is a network slowdown? Vast question… Your first target (as a network administrator) will be to look at the other bases where there is no congestion and where the latency times and retransmission rates are correct… and the one where there is material to investigate , in other parts of the application chain. How to proceed ?

So, you have Wireshark present on your PC; Let's go !

Let's select our first complainant: we'll call him Bob. Let's follow these steps[4]:

Let's place a capture point either using TAPs or a port mirroring - (see our article on the subject) - on Bob's network connection, but also on the servers side. This means that you will need two appliances to capture the traffic, one for the stream to disk and one for the packet recording.

There, you realize that the degradations are intermittent: you are going to need to record the traffic over a large number of days… and ask Bob to carefully note the precise times when each degradation occurs (which he probably won't do;).

After 2 or 3 days of traffic capture, the final battle arrives: analyzing 2GB of network trace files with Wireshark… Is it feasible? Not really…

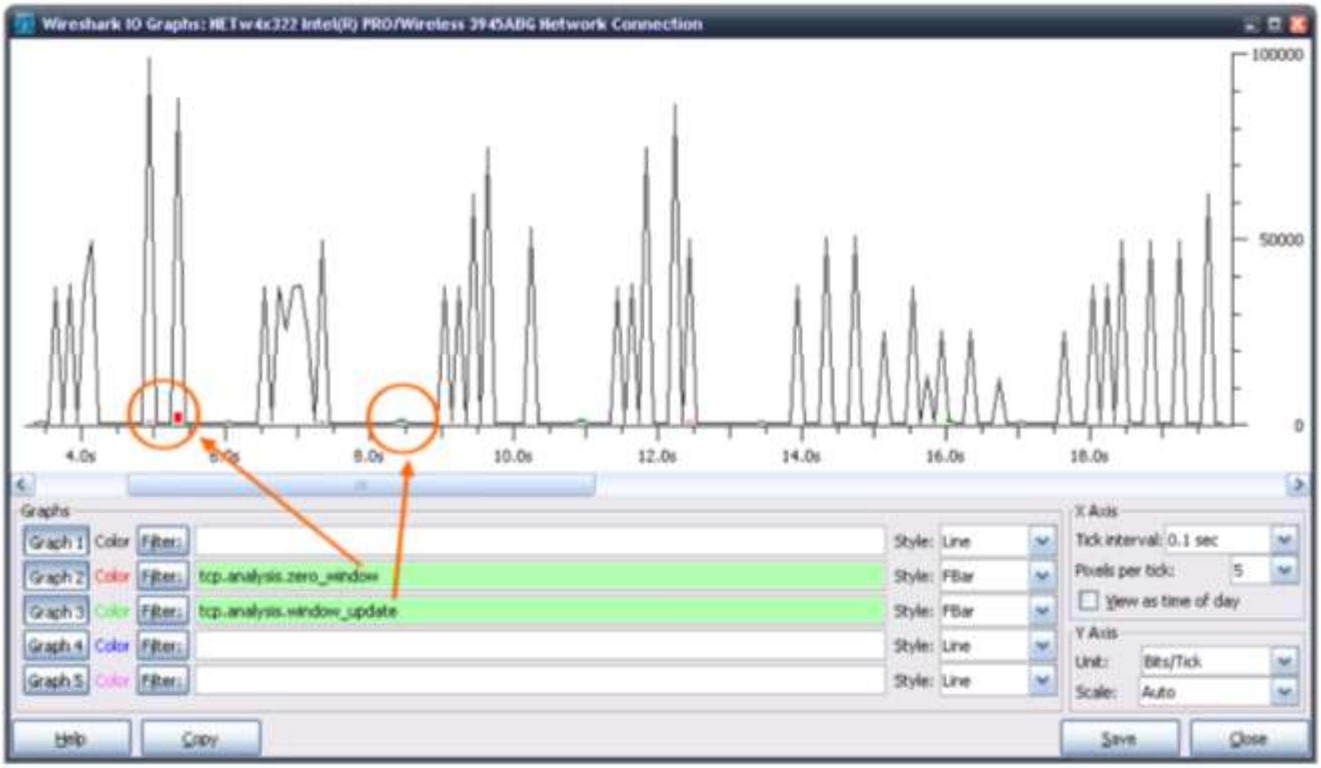
Let's break these results down into chunks and do some manual response time calculations [5].

After a (long… and relatively partial) analysis of the PCAP files… you spent 3 days, you found no sign of network problem, bandwidth saturation or packet loss… but a large number of application transactions carried out in tens of seconds!

We will record this information in a report and send it to the team in charge of the applications, knowing that they are not big fanatics of reading network traces and moreover that they will receive this information with a certain level of confidence. suspicion.

Be prepared, however, to consider these same steps for every user complaint and for every new incident.

To learn how, in 4 simple steps, to diagnose performance degradation of networks and applications, download our "Performance Diagnosis Guide".



Graph 1 Color Filter: Style: Line
Graph 2 Color Filter: tcp.analysis.zero_window Style: FBar
Graph 3 Color Filter: tcp.analysis.window_update Style: FBar
Graph 4 Color Filter: Style: Line
Graph 5 Color Filter: Style: Line

Tick interval: 0.1 sec
Pixels per tick: 5
□ View as time of day
Unit: Bits/Tick
Scale: Auto

Figure 1: Errors visualization

III. ANALYSIS TECHNIQUES

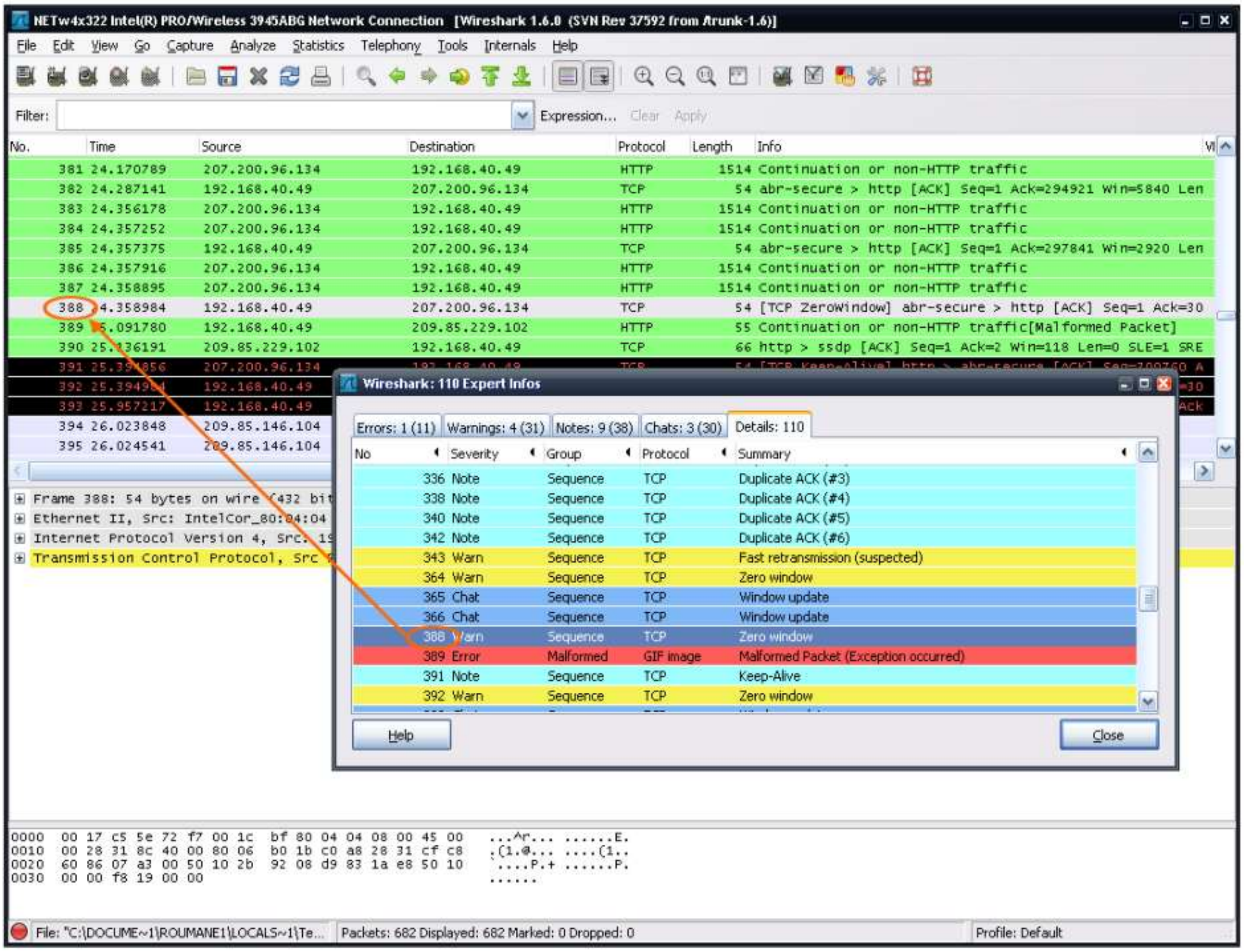
Analyzing behavior based on data sampling is the most complex part from the behavioral study process, but if the capture options are used judiciously, this work of analysis will not take too long.

- Reading frames:

according to the wireshark specificity there are variety of options to read and understand the data flow:

- Abstract summary
- protocols arborescence
- Data visualization

A constraint that persists usually at every analysis phase is the error avoidance;
In first view, the rapid and expert analysis functionalities give the chance to the behavior study to avoid error circumstances and genuine warnings in addition to the notes and chats happening within the communicating processes for a genuine communication circuits realisation and maintain, which is the role of the session layer from the OSI standardization model.



NETw4x322 Intel(R) PRO/Wireless 3945ABG Network Connection [Wireshark 1.6.0 (SVN Rev 37592 from /trunk-1.6)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
381	24.170789	207.200.96.134	192.168.40.49	HTTP	1514	Continuation or non-HTTP traffic
382	24.287141	192.168.40.49	207.200.96.134	TCP	54	abr-secure > http [ACK] Seq=1 Ack=294921 Win=5840 Len
383	24.356178	207.200.96.134	192.168.40.49	HTTP	1514	Continuation or non-HTTP traffic
384	24.357252	207.200.96.134	192.168.40.49	HTTP	1514	Continuation or non-HTTP traffic
385	24.357375	192.168.40.49	207.200.96.134	TCP	54	abr-secure > http [ACK] Seq=1 Ack=297841 Win=2920 Len
386	24.357916	207.200.96.134	192.168.40.49	HTTP	1514	Continuation or non-HTTP traffic
387	24.358895	207.200.96.134	192.168.40.49	HTTP	1514	Continuation or non-HTTP traffic
388	24.358984	192.168.40.49	207.200.96.134	TCP	54	[TCP ZeroWindow] abr-secure > http [ACK] Seq=1 Ack=30
389	25.091780	192.168.40.49	209.85.229.102	HTTP	55	Continuation or non-HTTP traffic[Malformed Packet]
390	25.136191	209.85.229.102	192.168.40.49	TCP	66	http > ssdp [ACK] Seq=1 Ack=2 Win=118 Len=0 SLE=1 SRE
391	25.394856	207.200.96.134	192.168.40.49	TCP	54	[TCP Keep-Alive] http > abr-secure [ACK] Seq=300760 A
392	25.394984	192.168.40.49				=30
393	25.957217	192.168.40.49				Ack
394	26.023848	209.85.146.104				
395	26.024541	209.85.146.104				

```
⊞ Frame 388: 54 bytes on wire (432 bit
⊞ Ethernet II, Src: IntelCor_80:04:04
⊞ Internet Protocol Version 4, Src: 19
⊞ Transmission Control Protocol, Src P
```

Wireshark: 110 Expert Infos

No	Severity	Group	Protocol	Summary
336	Note	Sequence	TCP	Duplicate ACK (#3)
338	Note	Sequence	TCP	Duplicate ACK (#4)
340	Note	Sequence	TCP	Duplicate ACK (#5)
342	Note	Sequence	TCP	Duplicate ACK (#6)
343	Warn	Sequence	TCP	Fast retransmission (suspected)
364	Warn	Sequence	TCP	Zero window
365	Chat	Sequence	TCP	Window update
366	Chat	Sequence	TCP	Window update
388	Warn	Sequence	TCP	Zero window
389	Error	Malformed	GIF image	Malformed Packet (Exception occurred)
391	Note	Sequence	TCP	Keep-Alive
392	Warn	Sequence	TCP	Zero window

```
0000  00 17 c5 5e 72 f7 00 1c  bf 80 04 04 08 00 45 00   ...^r... ......E.
0010  00 28 31 8c 40 00 80 06  b0 1b c0 a8 28 31 cf c8   .(1.@... ....(1..
0020  60 86 07 a3 00 50 10 2b  92 08 d9 83 1a e8 50 10   `....P.+ ......P.
0030  00 00 f8 19 00 00                                  ......
```

File: "C:\DOCUME~1\ROUMANE1\LOCALS~1\Te... Packets: 682 Displayed: 682 Marked: 0 Dropped: 0 Profile: Default

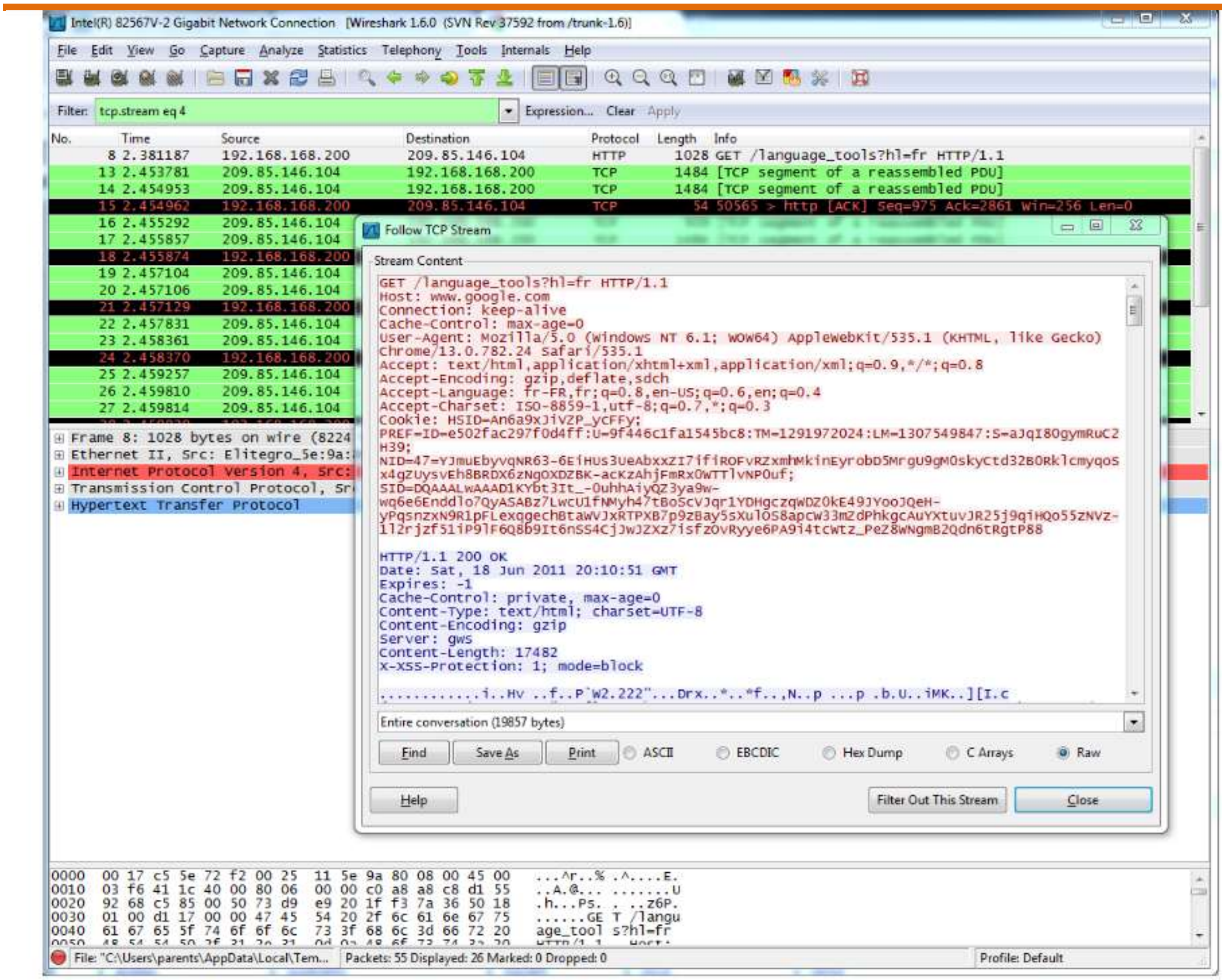
Figure 2: Network flow errors categories

From another side, a normal analysis exposes the captured data stores with their properties; timing and location. using this normal analysis several data might be read from the first draft [7]:

- protocols hierarchy
- Packets length
- conversations (data transmission) details

IV. TEXTS ANALYSIS

reconstructing the data flow is an option that must be implemented in order to understand the data in question. Wireshark has the capability to help and display the captured text in HTML format, where clearly we can differentiate the flow of the requests and the flow of the responses from the services' providers [8].


Intel(R) 82567V-2 Gigabit Network Connection [Wireshark 1.6.0 (SVN Rev 37592 from /trunk-1.6)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply
Follow TCP Stream
Stream Content
Find Save As Print
Help
Filter Out This Stream
Close

Filter: tcp.stream eq 4

```
No.    Time      Source           Destination      Protocol Length Info
     8 2.381187  192.168.168.200  209.85.146.104   HTTP     1028 GET /language_tools?hl=fr HTTP/1.1
    13 2.453781  209.85.146.104   192.168.168.200  TCP      1484 [TCP segment of a reassembled PDU]
    14 2.454953  209.85.146.104   192.168.168.200  TCP      1484 [TCP segment of a reassembled PDU]
    15 2.454962  192.168.168.200  209.85.146.104   TCP        54 50565 > http [ACK] Seq=975 Ack=2861 Win=256 Len=0
    16 2.455292  209.85.146.104
    17 2.455857  209.85.146.104
    18 2.455874  192.168.168.200
    19 2.457104  209.85.146.104
    20 2.457106  209.85.146.104
    21 2.457129  192.168.168.200
    22 2.457831  209.85.146.104
    23 2.458361  209.85.146.104
    24 2.458370  192.168.168.200
    25 2.459257  209.85.146.104
    26 2.459810  209.85.146.104
    27 2.459814  209.85.146.104
```

```
⊞ Frame 8: 1028 bytes on wire (8224
⊞ Ethernet II, Src: Elitegro_5e:9a:
⊞ Internet Protocol Version 4, Src:
⊞ Transmission Control Protocol, Sr
⊞ Hypertext Transfer Protocol
```

```
GET /language_tools?hl=fr HTTP/1.1
Host: www.google.com
Connection: keep-alive
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.1 (KHTML, like Gecko)
Chrome/13.0.782.24 Safari/535.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,sdch
Accept-Language: fr-FR,fr;q=0.8,en-US;q=0.6,en;q=0.4
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3
Cookie: HSID=An6a9xJiVZP_yCFFy;
PREF=ID=e502fac297f0d4ff:U=9f446c1fa1545bc8:TM=1291972024:LM=1307549847:S=aJqI80gymRuC2
H39;
NID=47=YJmuEbyvqNR63-6EiHUs3UeAbxxZI7ifiROFvRZxmhMkinEyrobD5MrgU9gM0skyCtd32B0Rklcmyqos
x4gZUysvEh8BRDX6zNgOXDZBK-acKzAhjFmRxOWTTlvNPOuf;
SID=DQAAALwAAAD1KYbt3It_-0uhhAiyQZ3ya9w-
wq6e6Enddlo7QyASABz7LwcU1fNMyh47tBoScVJqr1YDHgczqWDZ0kE49JYooJQeH-
yPqsnzxN9R1pFLexqgechBtaWVJxRTPXB7p9zBay5sXulOS8apcW33mZdPhkgcAuYXtuvJR25j9qiHQo55zNVz-
1l2rjzf51iP9lF6Q8b9It6nSS4CjJwJZXz7isfzOvRyye6PA9i4tCWtz_PeZ8WNgmB2Qdn6tRgtP88

HTTP/1.1 200 OK
Date: Sat, 18 Jun 2011 20:10:51 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=UTF-8
Content-Encoding: gzip
Server: gws
Content-Length: 17482
X-XSS-Protection: 1; mode=block

...........i..Hv ..f..P`W2.222"...Drx..*..*f..,N..p ...p .b.U..iMK..][I.c
```

Entire conversation (19857 bytes)

○ ASCII ○ EBCDIC ○ Hex Dump ○ C Arrays ● Raw

```
0000  00 17 c5 5e 72 f2 00 25  11 5e 9a 80 08 00 45 00   ...^r..% .^....E.
0010  03 f6 41 1c 40 00 80 06  00 00 c0 a8 a8 c8 d1 55   ..A.@... .......U
0020  92 68 c5 85 00 50 73 d9  e9 20 1f f3 7a 36 50 18   .h...Ps. . ..z6P.
0030  01 00 d1 17 00 00 47 45  54 20 2f 6c 61 6e 67 75   ......GE T /langu
0040  61 67 65 5f 74 6f 6f 6c  73 3f 68 6c 3d 66 72 20   age_tool s?hl=fr
```

File: "C:\Users\parents\AppData\Local\Tem... Packets: 55 Displayed: 26 Marked: 0 Dropped: 0 Profile: Default

Figure 3: Network flow text analysis

V. AUDIT

Analysing captured data is not a worker agent functionality, yet, it is the role of the user agent, to be well skilled with less knowledge about network protocols.
An audit in this concern might start with a protocol used within the captured sample of flow.
Then, an analysis would lead to, particularly, analyse the TCP and UDP protocols, which are generalized amongst different communication solutions [7].

VI. CONCLUSIONS AND FUTURE WORK

Presenting a study of the factors that make a network data flow more readable and more accurate to deploy for an anomaly detection is a gathering of a multitude of steps that must figure sequentially. Besides, to the constraints to consider; the actor skills is a must requirement, a deep understanding of the protocols architecture and behavior is a must too. All are to avoid erroneous reporting and thus failure in analyzing flows behavior.

REFERENCES

[1] N.Brownlee,"RTFM:ApplicabilityStatement,"RFC2721(Informational), Internet Engineering Task Force, October 1999.[Online]. Available: http://www.ietf.org/rfc/rfc2721.tx

[2] L. Deri, E. Chou, Z. Cherian, K. Karmarkar, and M. Patterson, "In-creasing Data Center Network Visibility with Cisco NetFlow-Lite," inProceedings of the 7th

International Conference on Network and ServiceManagement, CNSM'11, 2011, pp. 1–6.

[3] S. Alcock, P. Lorier, and R. Nelson, "Libtrace: A Packet Captureand Analysis Library,"SIGCOMM Computer Communication Review,vol. 42, no. 2, pp. 42–48, 2012.

[4] F. Fusco and L. Deri, "High Speed Network Traffic Analysis withCommodity Multi-core Systems," inProceedings of the 10th ACMSIGCOMM conference on Internet measurement, IMC'10, 2010, pp.218–224.

[5] B. Trammell and B. Claise, "Guidelines for Authors and Reviewers ofIP Flow Information Export (IPFIX) Information Elements," RFC 7013(Best Current Practice), Internet Engineering Task Force, September2013. [Online]. Available: http://www.ietf.org/rfc/rfc7013.txt

[6] P. Velan and R. Krejˇc´ı, "Flow Information Storage Assessment UsingIPFIXcol," inDependable Networks and Services. Proceedings of the 6thInternational Conference on Autonomous Infrastructure, Managementand Security, AIMS'12, ser. Lecture Notes in Computer Science, vol.7279. Springer Berlin Heidelberg, 2012, pp. 155–158.

[7] B. Li, J. Springer, G. Bebis, and M. H. Gunes, "A Survey of NetworkFlow Applications,"Journal of Network and Computer Applications,vol. 36, no. 2, pp. 567–581, 2013.

[8] L. Hellemons, L. Hendriks, R. Hofstede, A. Sperotto, R. Sadre, andA. Pras, "SSHCure: A Flow-Based SSH Intrusion Detection System," inDependable Networks and Services. Proceedings of the 6th InternationalConference on Autonomous Infrastructure, Management and Security,AIMS'12, ser. Lecture Notes in Computer Science, vol. 7279. SpringerBerlin Heidelberg, 2012, pp. 86–97.