Al-Yemenia University Journal
مجلة الجامعة اليمنية

# Intelligent Machine Learning mechanism for Advanced Persistent Threats (APT) early detection

**Al-Marhabi Zaid Ali**
*Head of Management Information System, Faculty of Managements and Economics*
*Al-Yemenia University, Yemen.*
*marhabi2000@gmail.com*

**Al-Hamdi Ayeda G**
*Computer Science Department, Faculty of Applied Sciences, Hajjah University,Yemen*
*ayeda_gu@yahoo.com*

**Habeb Abduljlil A.**
*College of Computer Science and Electronic Engineering*
*Hunan University, Changsha, China*
*abdualgalilhabeb@yahoo.com*

**Abstract**. Advanced Persistent Threats (APTs) are the major risk to the security of the online systems , therefore, its detection is very important. User and Entity Behavior Analytics (UEBA) mechanism detects . APTs by utilizing the machine learning algorithms, APTs are electronic attacks aimed at a particular place, usually governmental or private. Typically, the objective of these cyber-attacks is to steal valuable information from their database. The attack by APTs is a significant issue for the security of information and global networks. APT attacks may be combined with shareware or other software for downloading. Many kinds of APTs do not have difficulty passing the system firewall, their malicious behavior is hidden and they avoid all traditional detection methods with advanced evasion techniques.
Advanced Persistent Threats (APTs) are type of attacks that are very dangerous and they cause a lot of damages in the cyberspace, the main objective of this paper is to design and implement detection and prediction mechanisms of Advanced Persistent Threats (APTs) using cybersecurity and machine learning. Thus our research paper attempts to find a mechanism to identify the attacks that can be classified as APT attacks.

**Keywords**: Advanced Persistent Threats (APTs), Machine Learning, UEBA, Cyber Security Attack.

## 1. Introduction

In every Information Technology (IT) system, when any user accesses the system, a log is kept. UEBA does more than just relying on general logs ,it makes the IT system more secure. The artificial intelligence system is implemented to record the normal behavior of the users getting access to the IT system [2]. There is always a certain pattern based on which every user accesses the system and if anything abnormal is observed, UEBA quickly gives a security breach alert. Most of the APT malware, when enters into the system, leaves a little signature in the network traffic [3].

An advanced persistent threats (APTs) are computer network cyber-attacks in which an unauthorized and unlicensed person gets access to a computer network or systems and remains inside undetected for a long time [1]; The real intention of advanced persistent threats (APTs) attack is to steal and hijack data rather than to cause real damage to the computer network or organizations [1].

The attackers can penetrate the networks, systems, and organizations through advanced planning, stealthy techniques, and multiple attacks vectors (Such as internet, emails and deception) to have escalated privileges in the victim's environment to ultimately steal data.

**Advanced:** full spectrum of computer intrusion technologies [2].

The advanced term indicates that we are dealing with a threat that makes use of the full spectrum of attack technologies available [2].

**Persistent:** structured series of attacks with a long term goal, continuous monitoring and interaction[2].

The persistent term introduces a time factor, that means carried on in a structured manner that implies continuous and prolonged monitoring of the

target, so we can say advanced persistent threats are not a "Hit – and – Go" type of attacks, but a "Low – and – Slow" type[2].

**Threats:** criminal operators coordinate attack action [2].

The threats term used to indicate that we are dealing with a targeted attack with specific objectives that most likely involve a strong human intervention, as opposed to scripted attack or malwares [2].

The first use of the term "Advanced Persistent Threats (APTs)" was from the United State government July 2005, describing a modern, deceptive and physical form of attacks that targeted specific employees and tricked them into downloading files using browsers, accessing a website infected with Trojan horse software or opening email attachments having malwares[3].

**Life Cycle of APTs**

Security specialists that studied advanced persistent threats (APTs) have identified " APTs typically progress through a set of specific phases". We indicate those phases as the advanced persistent threats (APTs) life cycle[4][5].


***Figure 1.*** *Diagram depicting the life cycle staged approach of advanced persistent threats (APTs)[6].*

There are several versions of the advanced persistent threats (APTs) life cycle, at different levels of details. On the right, you see for example a well-known representation of the APTs life cycle. On the high-level, the steps in the pictures can be summarized in the following phases[4][5].

## 2. Related Works

This mechanism is developed to detect and mitigate the APTs, and provide a high level of security to the data. It consists of a total of eight successive phases in which it manages to detect the APT malware and making the crucial data inaccessible to malware. One of the advantages of this mechanism over others is that it can handle internal and external attacks[1].

The organizations have to adopt this mechanism without any external requirement. It is evident from research that, this method is very effective in detecting and mitigating APTs[2].

It simply works on the methodology of the military in which after detecting an intruder, the mechanism attacks it to make it ineffective[3]. Almost every type of attacks can be detected by the mechanism. It can easily handle the brute force attacks which require access to the data by forcing the data management system[4].But IKC blocks and discard it completely from the system without letting it to reach crucial data.

The stages involved in this mechanism are as follows:

- The first stage (observation stage) is known as reconnaissance in which the mechanism initiates the detection process. It analyzes the system externally at first and then moves in. This makes the mechanism detects the APT and even its mode of attack in some cases when the hackers have not used any complex method. The cyberattacks seek information and data that might give feedback about weak points in the system. By using

reconnaissance tools, the cyberattacks can do deep scan in corporate networks to search for entry points and vulnerabilities to be exploited. [5].

- The second stage is known as the intrusion in which the hackers get into the system and the data and transfer malware to the system (such as: adware, spyware and ransomware) [4]. Unfortunately, this malware could be delivered using different ways (such as: phishing email, hacker prone WIFI, and it might be a compromised website) [5]. IKC mechanism has the best opportunity to block the intruder at this stage.
- The Third stage is exploitation and this stage is aimed to mitigate the effect of malware and to make the system more secure against APT [2].
- In the fourth stage, the mechanism reduces access to crucial data, and this prevents hackers from increasing the privilege for accessing crucial data. This stage is known as the privilege escalation stage [5].
- After getting into the system the next action of the hackers is to get access to connected systems so that the effect can be huge. IKC mechanism allows the professionals to prevent the access of hackers [1]. Therefore, this stage is called the lateral movement stage.
- This stage makes the mechanism more secure as it does not allow the intruder to cover his track [4]. It provides the opportunity for the information security managers to look into the tracks and know more about the hacker.
- Another important tactic which the hackers normally do is the complete disruption of services [3]. This is very dangerous for the system and therefore, IKC implements well-defined security protocols to prevent hackers from doing that.
- If the hacker gets to this stage, then the IKC mechanism completely blocks the access to data so that it cannot be taken through the network [2].

The mechanism works according to the stages involved in APTs and therefore it makes the system more secure. The management team has more

leverage in controlling the hackers in getting access to the data. The mechanism is usually implemented along with UEBA [5]. This maximizes its performance and provides more security to the IT system.

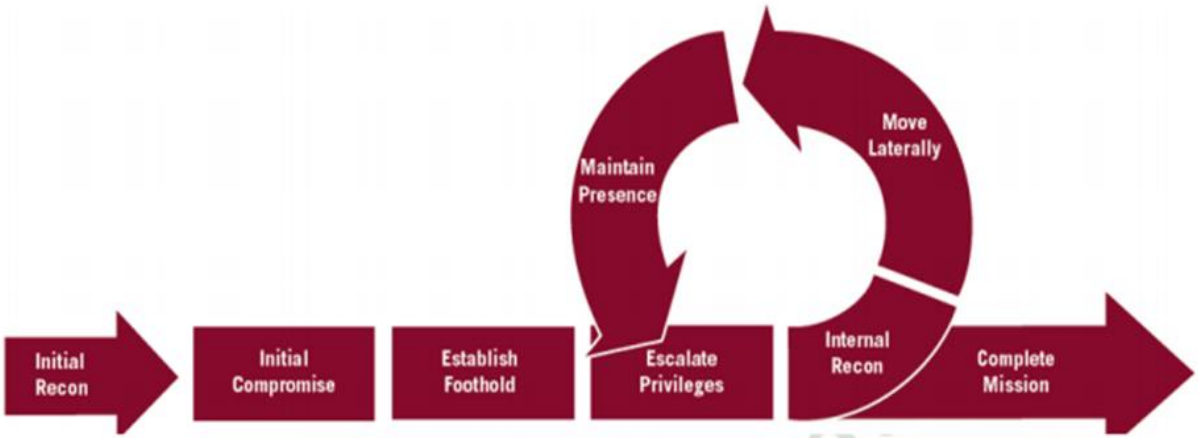



***Figure 2.*** *A typical kill chain-based mechanism to detect APT*

### 2.1 Process of APT attack.

APT attacks require careful planning and execution of everything, the series of steps taken by the assailants are similar, however, as the term was drafted in 2006 by the United States Air Force (USAF).

Earlier studies identified four steps[37], five steps[36] and [38], six steps[39], seven steps[40] and eight steps[41], However, each APT has a common feature that the same stages are passed to achieve its goal.

APTs are electronic attacks aimed at a particular place, usually governmental or private. Typically, the objective of these cyber-attacks is to steal valuable information from their database[33]. The attack by APTs is a significant issue for the security of information and global networks [34]. APT attacks may be combined with shareware or other software for downloading. Many kinds of APTs do not have difficulty passing the system firewall. Their malicious behavior is hidden and they avoid all traditional detection methods with advanced evasion techniques [35],

The APT attack process can be summarized as described in Figure 3.

1. Collection of intelligence: infiltrate the network of the company.
2. Initial operation: malware installation.
3. Command and control: Search for other vulnerabilities and additional controls infiltrated by malware, Fixed vulnerabilities are identified for the continuation of the attack, as well as the access to important information, passwords, and e-mail addresses.
4. Scaling privilege: after reaching the target, the hacker will clean up the effect and at any time leave a few gaps in his return.
5. Data exfiltration: export the data to the hacker.

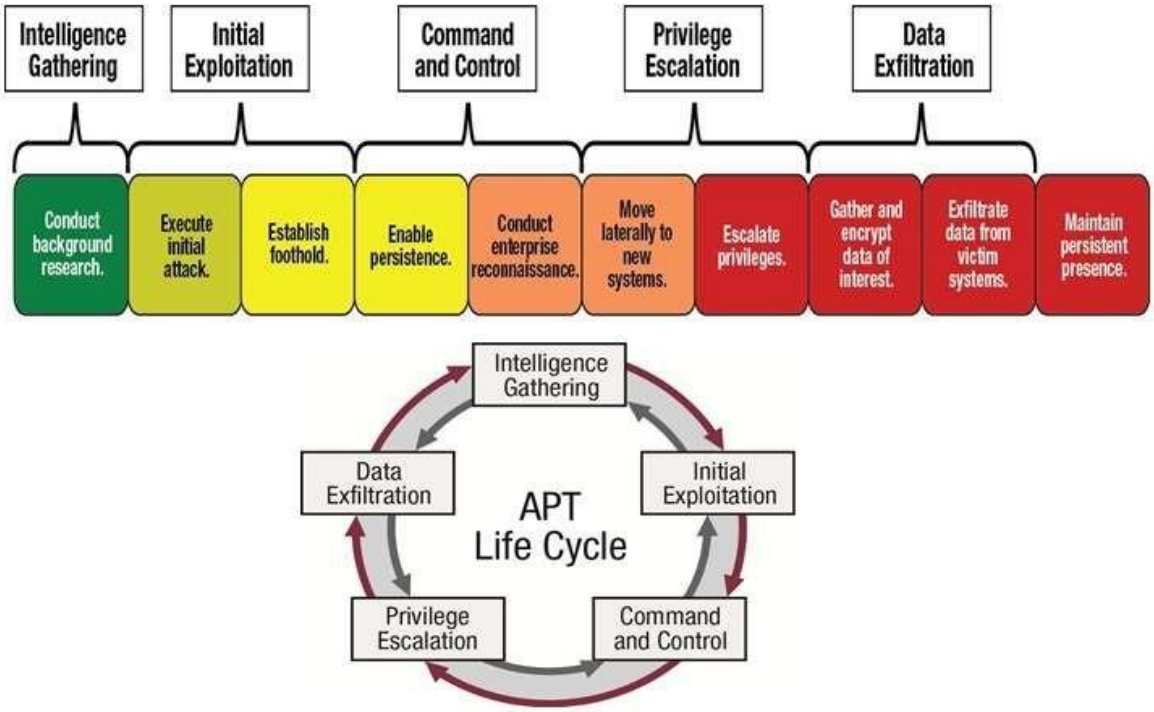



***Figure 3.*** *APT attack process.*

## 3. Proposed Mechanism

Here the first step is to get the dataset for areal data learning. After downloading the real Dataset which was about 700MB from (gitlab) website, Figure 4 shows one of the original dataset files obtained.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Flow ID | Src IP | Src Port | Dst IP | Dst Port | Protocol | Timestam | Flow Dura | Tot Fwd P | Tot Bwd P | TotLen Fw | TotLen Bw | Fwd Pkt Le | Fwd Pkt Le | Fwd Pkt Le | Fwd Pkt Le | Bwd Pkt Le | Bwd Pkt Le | Bwd Pkt Le | Bwd Pkt Le | Flov |
| 2 | 255.255.25 | 0.0.0.0 | 68 | 255.255.25 | 67 | 17 | 15/07/201 | 1.2E+08 | 88 | 1 | 25515 | 289 | 296 | 288 | 289.9432 | 2.42287 | 289 | 289 | 289 | 0 | 21 |
| 3 | 0.87.248.2 | 0.87.248.2 | 0 | 3.0.0.0 | 0 | 0 | 15/07/201 | 1.2E+08 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 4 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 110 | 6 | 15/07/201 | 12 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 5 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 993 | 6 | 15/07/201 | 12 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 6 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 113 | 6 | 15/07/201 | 99 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 7 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 80 | 6 | 15/07/201 | 690 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 8 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 445 | 6 | 15/07/201 | 659 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 9 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 21 | 6 | 15/07/201 | 489 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 10 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 3306 | 6 | 15/07/201 | 612 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 11 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 199 | 6 | 15/07/201 | 10 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 12 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 5900 | 6 | 15/07/201 | 14 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 13 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 1720 | 6 | 15/07/201 | 12 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 14 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 1271 | 6 | 15/07/201 | 15 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 15 | 192.168.3. | 192.168.3. | 34479 | 192.168.3. | 2007 | 6 | 15/07/201 | 13 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

***Figure 4.*** *one of the main dataset file for APT attack.*

In order to be able to mine into this huge amount of data, which is dataset for about three days (we will explain this in the next section).

### 3.1 Identify important columns in data

After referring to the scientific paper under the title(Analysis of high volumes of network traffic for Advanced Persistent Threads detection) which was published in the year 2016, and this research is based on (from research) and on part (5) feature extraction and normalization specifically. We were able to find out the important columns that lead us to the exploration of dataset.

This scientific paper summarizes it in equation (1):

$$x_t(h) = (x_t^1(h), x_t^2(h), \ldots, x_t^N(h)) \qquad (1)$$

And from this, we start digging in the file called (data_custom_normal.csv) and from who wonderful explanations to us we reached that the file contains many elements that concern us, and what we are interested in are: Flow ID, Src IP, Dst IP, and Time Stamp, TotLen Fwd Pkts.

As we note, the number of columns is too many, and after checking them, we found that the important columns are arranged as follows:

Flow ID : this attribute for counting number of flows.

Src Ip: for identifying local(home) host IP Address.

Dst IP: for identifying external host IP Address.

Timestamp: this row mention the time of communication between two(local and external) hosts.

TotLen Fwd Pkts: this row to count the number of bytes transmitted between two hosts.

| | A | B | C | H | I | M |
|---|---|---|---|---|---|---|
| | Flow ID | Src IP | Dst IP | Group | Timestamp | TotLen Fwd Pkts |
| 2 | 192.168.3.1-192.168.3.30-51427-22-6 | 192.168.3.30 | 192.168.0.1 | 0 | 15/07/2019 01:55:20 PM | 12352 |
| 3 | 8.0.6.4-8.6.0.1-0-0-0 | 8.6.0.1 | 8.0.6.4 | 0 | 15/07/2019 01:55:21 PM | 0 |
| 4 | 255.255.255.255-0.0.0.0-67-68-17 | 0.0.0.0 | 255.255.255.255 | 0 | 15/07/2019 01:55:22 PM | 25515 |
| 5 | 192.168.3.10-239.2.11.71-53569-8662-17 | 192.168.3.10 | 239.2.11.71 | 0 | 15/07/2019 01:55:22 PM | 1072 |
| 6 | 192.168.3.30-192.168.3.31-40504-9200-6 | 192.168.3.30 | 192.168.3.31 | 0 | 15/07/2019 01:55:23 PM | 23499 |
| 7 | 0.87.248.248-3.0.0.0-0-0-0 | 0.87.248.248 | 3.0.0.0 | 0 | 15/07/2019 01:55:58 PM | 0 |
| 8 | 192.168.3.1-192.168.3.30-34479-110-6 | 192.168.3.1 | 192.168.3.30 | 0 | 15/07/2019 01:57:12 PM | 0 |
| 9 | 192.168.3.1-192.168.3.30-34479-993-6 | 192.168.3.1 | 192.168.3.30 | 0 | 15/07/2019 01:57:12 PM | 0 |
| 10 | 192.168.3.1-192.168.3.30-34479-113-6 | 192.168.3.1 | 192.168.3.30 | 0 | 15/07/2019 01:57:12 PM | 0 |
| 11 | 192.168.3.1-192.168.3.30-34479-80-6 | 192.168.3.1 | 192.168.3.30 | 0 | 15/07/2019 01:57:12 PM | 0 |
| 12 | 192.168.3.1-192.168.3.30-34479-445-6 | 192.168.3.1 | 192.168.3.30 | 0 | 15/07/2019 01:57:12 PM | 0 |

***Figure 5.*** *five important attributes in original Dataset*

Now that the previous step is to reduce the number of important columns, we have only five important columns. Figure 5 shows those columns.

### 3.2 Framework overview

This section gives an overview of the proposed frameworks for the classification of internal hosts that are likely participating in APT data exfiltration. The final output is an ordered list of hosts that may have conducted suspicious network activities in connection with APTs. Security analysts should also concentrate on the manual review of the top-k suspected hosts only.

APT detection's key objective is to address the problems faced by a creative approach with the following characteristics:

- Unlike many APT detective solutions that require a large number of host-based logs to be analyzed. We concentrate on traffic formation easily obtainable from network samples in the observed network environment;
- To handle large traffic volumes effectively, we ex-tract and analyze flow records; this design choice produces better results in terms of both storage occupation and research costs;
- In contrast to most of the current methods for detecting anomalies that rely primarily on the statistics of the network, we concentrate on individual hosts and comparative statistics to classify hosts performing questionable activities (i) in terms of past conduct and (ii) other internal organization's hosts;
- We provide a collection of features to recognize hosts that may participate in data exfiltration.
- • Furthermore, deep packet inspection does not count on the proposed rating method and can therefore work with encrypted traffic.

We are talking about an enterprise network scenario that comprises a few thousand hosts. Figure 6 reports the main phases of the proposed framework:

(1) flow collection and storage;
(2) feature extraction;
(3) feature normalization;
(4) computation of suspiciousness scores;
(5) (5) ranking.

We note that the second, third and fourth stages can be carried out independently for each internal host and therefore, by running these stages in parallel, the approach proposed can scale to very large networks comprised of over a hundred thousand host groups.

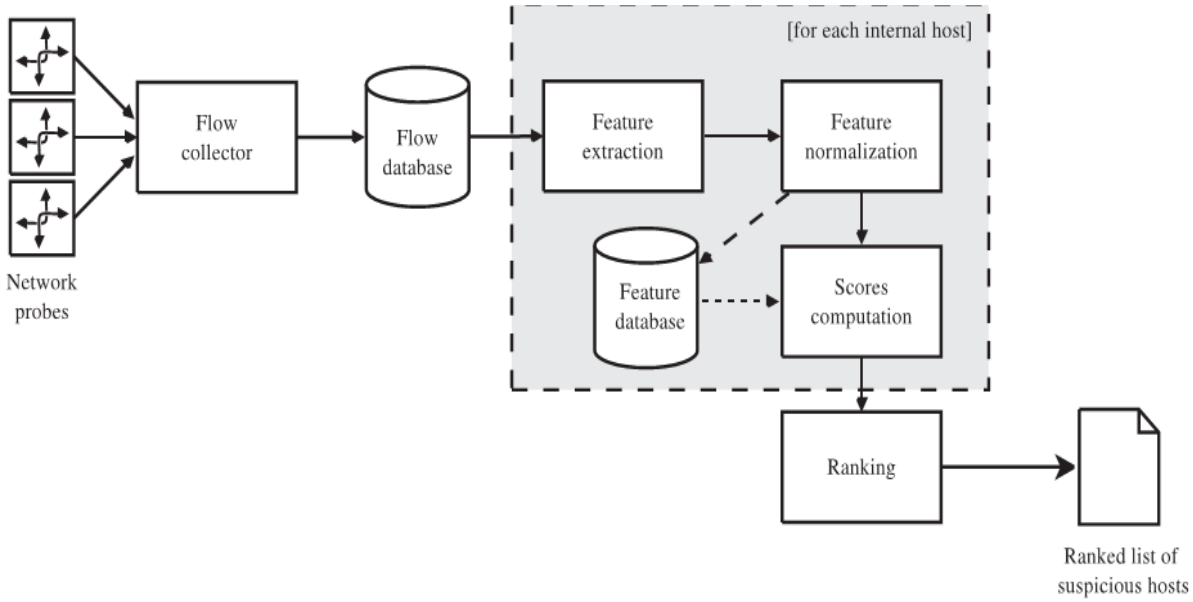



***Figure 6.*** *proposed Framework*

- Since it is impossible to analyze and store raw traffic data, the first step is aimed at collecting flow records. Each record contains some information extracted from the IP header, such as addresses and ports of source and destination (e.g., TCP, UDP, ICMP). In terms of its efficiency, the adoption of flow records instead of raw traffic data offers several advantages.
- flow records, also obtained over longer periods of time (e.g. years) can be easily processed and compressed.
- Flow records processing is more computer-feasible than analysis of enormous quantities of raw data.

The historical window represents the host's past actions in the network. The movement is often taken into account in relation to the recent past, so that unusual directions are considered suspected.

In the fifth step, the ranking algorithm gives each host a suspicion score that can be compared to all other hosts. The degree of suspicion is evaluated in particular as a linear combination of:

- normalized distance of an internal host with respect to the centroid of the feature space.
- percentage increment of the magnitude of the movement.
- difference in direction of movements in the observed network environment with respect to movements of all internal hosts.

As a result, an analyzer will concentrate exclusively on the top k suspicious hosts, classified as suspect hosts.

The information for each of these stages, along with various examples and evaluations on flow data from a real network of approximately 10K hosts, will be discussed in the following sections.

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | Time | Src Ip | Number of Bytes Uploaded | Number of Flows | Number of External Ips |
| 3305 | 132 | 0.0.0.0 | 0 | 0 | 0 |
| 3306 | 132 | 0.87.248.248 | 0 | 0 | 0 |
| 3307 | 132 | 151.101.1.194 | 0 | 0 | 0 |
| 3308 | 132 | 172.217.11.173 | 0 | 0 | 0 |
| 3309 | 132 | 172.217.11.69 | 0 | 0 | 0 |
| 3310 | 132 | 172.217.14.100 | 0 | 0 | 0 |
| 3311 | 132 | 172.217.4.174 | 0 | 0 | 0 |
| 3312 | 132 | 192.112.36.4 | 0 | 0 | 0 |
| 3313 | 132 | 192.168.3.1 | 0 | 0 | 0 |
| 3314 | 132 | 192.168.3.10 | 8564 | 6 | 1 |
| 3315 | 132 | 192.168.3.29 | 0 | 0 | 0 |
| 3316 | 132 | 192.168.3.30 | 36390 | 6 | 1 |
| 3317 | 132 | 192.168.3.31 | 0 | 0 | 0 |
| 3318 | 132 | 192.168.3.34 | 0 | 0 | 0 |
| 3319 | 132 | 192.30.253.112 | 0 | 0 | 0 |
| 3320 | 132 | 198.143.164.252 | 0 | 0 | 0 |
| 3321 | 132 | 198.97.190.53 | 0 | 0 | 0 |
| 3322 | 132 | 202.12.27.33 | 0 | 0 | 0 |
| 3323 | 132 | 216.58.217.206 | 0 | 0 | 0 |
| 3324 | 132 | 34.206.81.156 | 0 | 0 | 0 |
| 3325 | 132 | 54.81.84.187 | 0 | 0 | 0 |
| 3326 | 132 | 8.6.0.1 | 0 | 0 | 0 |
| 3327 | 132 | 91.189.91.23 | 0 | 0 | 0 |
| 3328 | 132 | 91.189.91.26 | 0 | 0 | 0 |
| 3329 | 132 | 91.189.92.20 | | 0 | 0 |

***Figure 7.*** *mining result for the last time slot*

With the new partition process, we will have to re-divide the time in the first file from 20 minutes to only 12 minutes, so the total time periods in the first file will be 83 time periods (0 -82ts). The second file will be about 50 periods of time. After the partition process in both files, we got a total of 132 periods of time and this amount of period is somewhat satisfactory. Figure 7 shows the first excavation(mining) in one of the time periods (here we chose the last time period).

In the previous figure, we make a shadow for the time period 132. Where the first column shows the time period and the second column shows the local address, and the third column shows the total transmitted data(bytes)after collecting the active addresses in all time periods. As we note that in this period we have we have only two active titles: 192,168.3.10 and 192,168.3.30, where the first transmits 8564 Bytes and the second 36,390 Bytes. The first done six messaging times as well as the second, which is explained by the fourth column, but the fifth and final column shows that they are both Connected to only one device during this period (and for the most time periods these two addresses are the most active addresses).

### 3.3 Feature extraction and normalization

In this segment, we present and a range of features for which we can extract data exfiltrations for each internal host. Then we explore how to standardize and compare the components of each vector function.

- Feature extraction

We extract a set of options for each internal host in the observed network environment to detect data exfiltration through the analysis of suspect and rare movements in the space of the feature.

Let us define $H_I$ and $H_E$ as the sets of *internal* and *external* hosts, respectively. For each internal host $h \in H_I$, a *feature vector* $\mathbf{x}_t(h)$ is computed that is defined as the following ordered tuple:

$$\boldsymbol{x}_t(h) = (x_t^1(h), x_t^2(h), \ldots\ldots, x_t^N(h)) \quad (2)$$

where $x_t^i(h)$, with regard to internal host h, corresponds to the value of the i -th portion of the vector function t. Each sample time (T), in particular, is calculated with the feature vector values $x_t^1(h)$. This makes it possible to create a time set for the internal host h ∈ $H_i$ with the vector values. In the next steps, these sequences shall be used to evaluate the host's movements over time. For the sake of simplicity, unless specified, in the remainder of the paper we use a simplified notation in which we omit $h$ , that is: the feature vector of internal host $h$ is referred to as $\mathbf{x}_t\ (h) = \mathbf{x}_t$ , with components $x_t^i(h) = x_t^i$.

We refer in this report to time granularity T = 1 day without loss of generality as we have checked that, for various reasons, this is agood granularity, of which:

APTs include activities which could continue for days or months [1], which means that study could cause unnecessary fine time pellets (e.g. minutes);

• The security analyst can quickly concentrate on the suspicious hosts for manual analysis by providing a ranking list of suspicious hosts once a day.

Additional time granularities may also be considered based on the characteristics of the observed network environment and potential safety analyst domain expertise.

We now propose a series of features aimed at identifying hosts engaged in suspicious network activities which may require data exfiltration of an APT:

(1) ***numbytes****: Amount of megabytes of external addresses uploaded from internal hosts (i.e. potential exfiltration points);*

(2) ***numflows****: Number of flows to external hosts initiated by internal hosts (typically connections);*

(3) ***numdst****: Number of external IP addresses for an internal host link.*

For each of these features we motivate the choice now.

**numbytes** enables us, as they may correspond to data exfiltrations, to monitor deviations of uploaded bytes. For instance, if a host shows ten times more bytes, it can participate in activities related to APT. The number of bytes that internal hosts load in megabytes is shown in this paper**.**

**Numflows** are used to track internal hosts' data transfers. Exfiltrations by internal hosts are initiated for two key reasons: The outgoing connections are not blocked by most fire-styles; (ii) external hosts are suspected of initiating connections and can easily be found by conventional intrusion detection systems based on the signature.

**Numberdst** allows anomalous activities to be identified involving changes in the number of different destinations that each internal host contacts. For instance, if the number of external IPs contacted by an internal host is stable in a given time window, whereas the number of bytes or connections uploaded is increasing significantly, it may be a data exfiltration or activity associated with APT.

We know that numflows and numdst are linked to a certain extent, if there is a significant rise in the number of flows initiated by an internal host, the number of destinations is expected to increase. On the contrary, the number of flows is lowering. Although the learning machine usually recommends that uncorrelated characteristics be used to maximize information contained in a feature vector. We note that we are not interested in classification ,but in the identification of suspicious motions in the feature space.

Therefore, we willingly choose two correlated characteristics to catch the presence of internal hosts moving towards the feature space that violates the predicted connection from numflow to numdst;

- Feature Normalization

In this paragraph, after we have completed the division of the dataset into about 133 periods of time, as in the previous paragraph;

Here first we have to calculate Quartile Weighted Median (QWM) by equation (1) which shows that it is the third quarter for each time period with twice the second quarter of the same period with the third quarter of that period all divided by 4, but we have to pay attention that we have to calculate QWM for the three columns i.e.

We must normalize the distributions of the internal hosts to make a reasonable compare of their movements and location. A normalization of the range is one of the most common techniques. It maps a distribution between 0 and 1 by normalizing the maximum and minimum value. However, in the considered sense, we have shown that each function has a different distribution of heavy tails. Of-scale values are also often used and normalization of the range would provide bad results, as most values would be close to 0.

We adopt the QWM metric for two parties to overcome this challenge, which is described as the QWM metric :

$$QWM(D) = \frac{Q_{75}(D)+2*Q_{50}(D)+Q_{25}(D)}{4} \quad (3)$$

Where $Q_k(D)$ corresponds to the k quantile of the dataset D. We observe that this measure takes into account both the median of the values (Q .50), and the variance of the data, that is represented by the first and third quartiles. This makes the QWM robust and independent of the distribution of the data, and also suitable to normalize values of heavy-tailed distributions to similar central tendencies.

QWM account for the total data sent by byte, and QWM for the number of times the messaging and QWM for the number of external addresses Figure 8 QWM calculation for the first time period (period 0)

| | A | B | C | D | E | H | I | J |
|---|---|---|---|---|---|---|---|---|
| 1 | Time | Src Ip | Number of Bytes Uploaded | Number of Flows | Number of External Ips | QWMnumbytes | QWMnumflows | QWMnumdst |
| 2 | 0 | 0.0.0.0 | 0 | 0 | 0 | 45444.5 | 251.625 | 3.8125 |
| 3 | 0 | 0.87.248.248 | 0 | 0 | 0 | | | |
| 4 | 0 | 151.101.1.194 | 0 | 0 | 0 | | | |
| 5 | 0 | 172.217.11.173 | 0 | 0 | 0 | | | |
| 6 | 0 | 172.217.11.69 | 0 | 0 | 0 | | | |
| 7 | 0 | 172.217.14.100 | 0 | 0 | 0 | | | |
| 8 | 0 | 172.217.4.174 | 0 | 0 | 0 | | | |
| 9 | 0 | 192.112.36.4 | 0 | 0 | 0 | | | |
| 10 | 0 | 192.168.3.1 | 0 | 1085 | 1 | | | |
| 11 | 0 | 192.168.3.10 | 8620 | 6 | 1 | | | |
| 12 | 0 | 192.168.3.29 | 0 | 0 | 0 | | | |
| 13 | 0 | 192.168.3.30 | 99779 | 88 | 8 | | | |
| 14 | 0 | 192.168.3.31 | 0 | 0 | 0 | | | |
| 15 | 0 | 192.168.3.34 | 125857 | 51 | 0 | | | |
| 16 | 0 | 192.30.253.112 | 0 | 0 | 0 | | | |
| 17 | 0 | 198.143.164.252 | 0 | 0 | 0 | | | |
| 18 | 0 | 198.97.190.53 | 0 | 0 | 0 | | | |
| 19 | 0 | 202.12.27.33 | 0 | 0 | 0 | | | |
| 20 | 0 | 216.58.217.206 | 0 | 0 | 0 | | | |
| 21 | 0 | 34.206.81.156 | 0 | 0 | 0 | | | |
| 22 | 0 | 54.81.84.187 | 0 | 0 | 0 | | | |
| 23 | 0 | 8.6.0.1 | 0 | 0 | 0 | | | |
| 24 | 0 | 91.189.91.23 | 0 | 0 | 0 | | | |
| 25 | 0 | 91.189.91.26 | 0 | 0 | 0 | | | |
| 26 | 0 | 91.189.92.20 | 0 | 0 | 0 | | | |

***Figure 8.*** *Quartile Weighted Median calculation*

QWM calculation results for the amount of data sent can be written as follows if we use Excel:

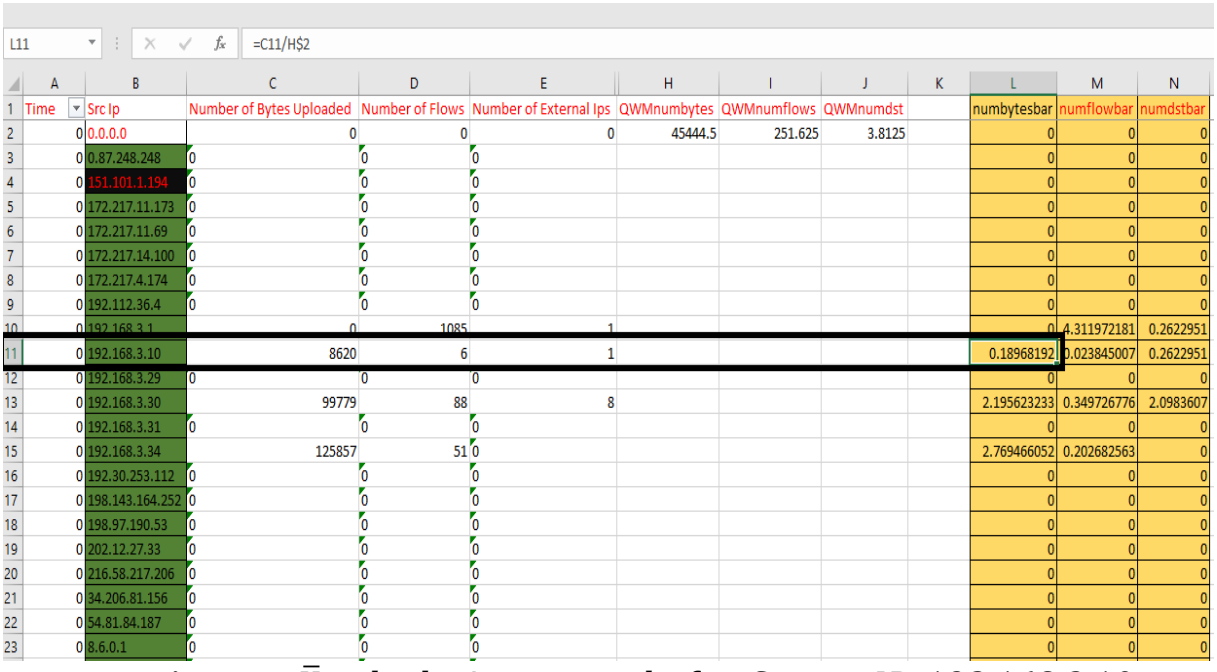
=(QUARTILE.EXC(C2:C26,1)+ 2*QUARTILE.EXC(C2:C26,2)+ QUARTILE.EXC(C2:C26,3)/4)

Second: QWM calculation was the first step to calculate the value of ($\bar{X}$), as described in the equation (4). The $\bar{X}$ is the value of X in all three columns (the third column of the total data and the fourth for the number of messaging times and the fifth for the number of addresses contacted) divided by the value of QWM for the same period of time.

$$\bar{X}_t^i = \frac{x_t^i}{QWM(\boldsymbol{X}_t^i)} \qquad (4)$$

Where $X_t$ is the set of feature vectors of all internal hosts h ∈ $H_I$, and $x_t^i$ is the set of the (i-th) components. We observe that normalization is performed

with respect to the whole population to fairly compare the behaviours of all internal hosts.



L11 =C11/H$2

| | A | B | C | D | E | H | I | J | K | L | M | N |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Time | Src Ip | Number of Bytes Uploaded | Number of Flows | Number of External Ips | QWMnumbytes | QWMnumflows | QWMnumdst | | numbytesbar | numflowbar | numdstbar |
| 2 | 0 | 0.0.0.0 | 0 | 0 | 0 | 45444.5 | 251.625 | 3.8125 | | 0 | 0 | 0 |
| 3 | 0 | 0.87.248.248 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 4 | 0 | 151.101.1.194 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 5 | 0 | 172.217.11.173 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 6 | 0 | 172.217.11.69 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 7 | 0 | 172.217.14.100 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 8 | 0 | 172.217.4.174 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 9 | 0 | 192.112.36.4 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 10 | 0 | 192.168.3.1 | 0 | 1085 | 1 | | | | | 0 | 4.311972181 | 0.2622951 |
| 11 | 0 | 192.168.3.10 | 8620 | 6 | 1 | | | | | 0.18968192 | 0.023845007 | 0.2622951 |
| 12 | 0 | 192.168.3.29 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 13 | 0 | 192.168.3.30 | 99779 | 88 | 8 | | | | | 2.195623233 | 0.349726776 | 2.0983607 |
| 14 | 0 | 192.168.3.31 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 15 | 0 | 192.168.3.34 | 125857 | 51 | 0 | | | | | 2.769466052 | 0.202682563 | 0 |
| 16 | 0 | 192.30.253.112 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 17 | 0 | 198.143.164.252 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 18 | 0 | 198.97.190.53 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 19 | 0 | 202.12.27.33 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 20 | 0 | 216.58.217.206 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 21 | 0 | 34.206.81.156 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 22 | 0 | 34.81.84.187 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |
| 23 | 0 | 8.6.0.1 | 0 | 0 | 0 | | | | | 0 | 0 | 0 |

***Figure 9.*** *$\bar{X}$ calculating example for Source IP*: 192.168.3.10

In the previous figure 9, we deliberately specified line 11, which is for the $\bar{X}$ value calculation for local address No. 192,168.3.10, This is calculated in the column entitled numbytesbar, which is the value of the C11 field divided by the value of QWM numbytes for the same time period and is in the H2 field specifically and the equation will be as follows = C11/H$2.

Thus, the $\bar{X}$ value will be calculated for all third, fourth and fifth columns in numbytesbar, numflowbar and numdstbar respectively.

In the previous part we make a dataset preparation for mining all collected dataset, this step is very important for correct suspiciousness finding. Let us consider the normalized *feature vector* of the internal host *h* at time *t*:

$$\boldsymbol{x}_t = (x_t^1, x_t^2, x_t^3) \qquad \textbf{(5)}$$

where $x_t^i$ are *numbytes*( $x_t^1$ ), *numdst* ( $x_t^2$ ) and *numflows* ( $x_t^3$ ).

At each time *t* we compute for each internal host three *suspiciousness scores* :

- $s_t^1$: *distance* from the centroid of the feature space.
- $s_t^2$: *magnitude* of the movement in the feature space.
- $s_t^3$: *unlikelihood* of movement *direction*.

## A. S1 Counting: Distance from feature space centroid

We know, according to the scientific paper, which is the main reference in our research, that S1 is distance from the centroid of the feature space, which in turn, before the calculation of S1 We must calculate the value of c (x) first and this can be calculated as in the equation (1). Then we also have to calculate D1, D2 and D3 for numbytesbar, numflowbar and numdstbar respectively.

First, we compute a score $s_t^1$ that depends on the position of the host *h* at time *t* with respect to all the other internal hosts. The aim is to determine whether a host is located in a multidimensional feature space anomalous area in time t.

Let us consider $\mathbf{X}_t$ as the set of all positions of internal hosts in the feature space at time *t*, that is:

$$\mathbf{X}_t = \{ \mathbf{x}_t (h) : h \in H_I \} \quad (6)$$

We then define $\mathbf{c}(\mathbf{X}_t)$ as the *centroid* of the feature space at time *t* . In particular, it is computed as:

$$\boldsymbol{c}(\boldsymbol{X_t}) = \left(\frac{\sum_h x_t^1(h)}{|X_t|}, \frac{\sum_h x_t^2(h)}{|X_t|}, \frac{\sum_h x_t^3(h)}{|X_t|}\right) \quad (7)$$

So for the (c(x) calculation for each time period for each of the three columns mentioned, we make a new Excel file and we place the three values

instead of the original X values, i.e. numbytesbar, which we put instead of Number of Bytes Uploaded, And numflowbar we put it instead of Number of Flows, and finally numdstbar instead of Number of External Ips. So that we can calculate the value of (c (x) by collecting the total data in each time period and then dividing the output by the number of fields in that column, as illustrated in figure 10.

| 1 | Time | Src Ip | numbytesbar | numflowbar | numdstbar | | D1 | D2 | D3 |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 0 | 0.0.0.0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 3 | 0 | 0.87.248.248 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 4 | 0 | 151.101.1.194 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 5 | 0 | 172.217.11.173 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 6 | 0 | 172.217.11.69 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 7 | 0 | 172.217.14.100 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 8 | 0 | 172.217.4.174 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 9 | 0 | 192.112.36.4 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 10 | 0 | 192.168.3.1 | 0 | 4.311972181 | 0.262295082 | | 0 | 486.3279959 | 6.25 |
| 11 | 0 | 192.168.3.10 | 0.18968192 | 0.023845007 | 0.262295082 | | 0.8462781 | 0.0148721 | 6.25 |
| 12 | 0 | 192.168.3.29 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 13 | 0 | 192.168.3.30 | 2.195623233 | 0.349726776 | 2.098360656 | | 113.3905508 | 3.199153943 | 400 |
| 14 | 0 | 192.168.3.31 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 15 | 0 | 192.168.3.34 | 2.769466052 | 0.202682563 | 0 | | 180.4069738 | 1.074509221 | 0 |
| 16 | 0 | 192.30.253.112 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 17 | 0 | 198.143.164.252 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 18 | 0 | 198.97.190.53 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 19 | 0 | 202.12.27.33 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 20 | 0 | 216.58.217.206 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 21 | 0 | 34.206.81.156 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 22 | 0 | 54.81.84.187 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 23 | 0 | 8.6.0.1 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 24 | 0 | 91.189.91.23 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 25 | 0 | 91.189.91.26 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 26 | 0 | 91.189.92.20 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 27 | sum | sum | 5.154771204 | 4.888226528 | 2.62295082 | | | | |
| 28 | count | count | 25 | 25 | 25 | | | | |
| 29 | C | C | 0.206190848 | 0.195529061 | 0.104918033 | | | | |

***Figure 10.*** *c(x) calculation for each time period*

As we note that in line 27 we calculated the total byte sent to each address in the first time period (no. 0) i.e. the total data from numbytes bar field from line 2 to line 29, and line 28 we calculated the number of lines or addresses, which is almost constant in all time periods of 25. Now we can easily calculate the c(x) value, which is Centroid of the feature space at time t, which is the division of line 27 on line 28. After calculating the value of (c (x) we can now calculate the value of D, which, as in the equation, is (the value of $X^{-}$ for that field divided by the value of C for that time period)2, for example in the previous form if we

choose the local address: 192.168.3.10, which is in line 11 specifically we note that the value of D1 Is (0.8462781) in excel program we can write the equation like this = (C11/C$29)^2 means the value of line 11 on the value of C for the same line. We note that in the value of C we marked $C$29 so that the equation is generalized for all elements of the single time interpretation. In the same way, both D2 and 3D will be calculated. Finally, we can calculate the value of S1 by equation (4) which means the square root of the values of both D1, D2 and D3, as shown in Figure 10.2.

$$s_t^1 = d_t\big(x_t(h), \boldsymbol{c}(\boldsymbol{X}_t)\big) = \sqrt{\sum_{i=1}^{3}(x_t^i(h) - c^i(\boldsymbol{X}_t))^2} \qquad (8)$$

P2 =SQRT(G2+H2+I2)

| | C | D | E | F | G | H | I | O | P |
|---|---|---|---|---|---|---|---|---|---|
| 1 | numbytesbar | numflowbar | numdstbar | | D1 | D2 | D3 | | S1 |
| 2 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 3 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 4 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 5 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 6 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 7 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 8 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 9 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 10 | 0 | 4.311972181 | 0.262295082 | | 0 | 486.3279959 | 6.25 | | 22.19409822 |
| 11 | 0.18968192 | 0.023845007 | 0.262295082 | | 0.8462781 | 0.0148721 | 6.25 | | 2.666673996 |
| 12 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 13 | 2.195623233 | 0.349726776 | 2.098360656 | | 113.3905508 | 3.199153943 | 400 | | 22.72860983 |
| 14 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 15 | 2.769466052 | 0.202682563 | 0 | | 180.4069738 | 1.074509221 | 0 | | 13.47150634 |
| 16 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 17 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 18 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 19 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 20 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 21 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 22 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 23 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 24 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 25 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 26 | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 |
| 27 | 5.154771204 | 4.888226528 | 2.62295082 | | | | | | |

***Figure 11.*** *Calculating result for D1, D2, D3 and $S^1$*

As seen in the previous figure, the value of the $S_1$ is calculated by excel for each row in the following mathematical form: =SQRT (G2+H2+I2) With the row number changed for all the values

## B. S2 Counting: Magnitude of movement in the feature space.

In the previous section we have calculated $S_1$, now in this paragraph we will discuss how to calculate the value of $S_2$ which is magnitude of movement in the feature space, but before we can calculate it according to the equation( we must first calculate the value (BW for all three columns $\bar{X}$) and then calculate (Mt also for all three columns $\bar{X}$

The purpose is to identify a distance metric that is suitable for measuring suspiciousness of movements of internal hosts in the feature space.

The movement of x t from t − 1 to t is represented as a distance vector in a Euclidean space:

$$x_t - x_{t-1} = (x_t^1 - x_{t-1}^1, x_t^2 - x_{t-1}^2, x_t^3 - x_{t-1}^3) \quad (9)$$

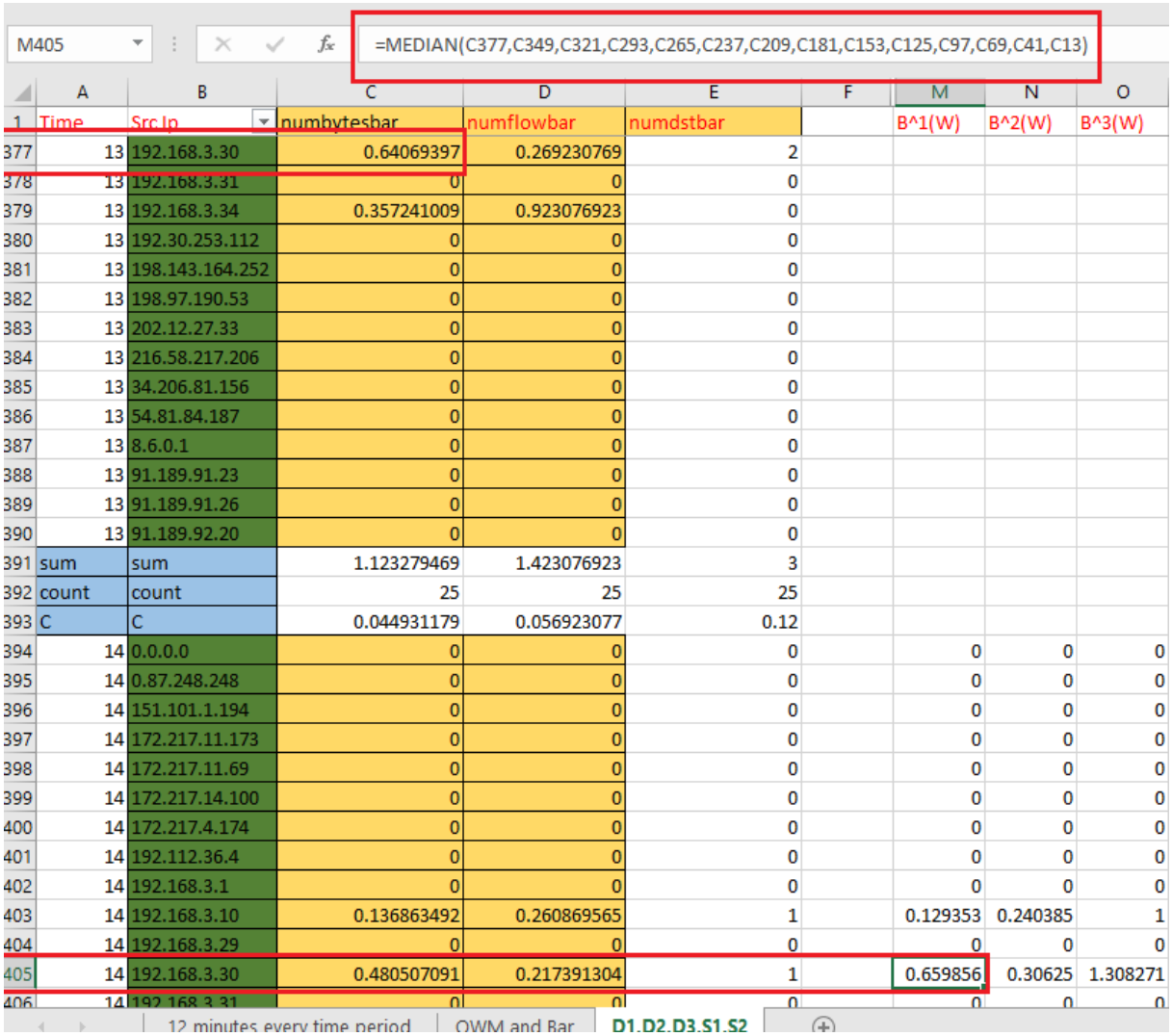
One disadvantage of this representation is that the magnitude of the distance vector is probably higher for internal hosts far removed from the source of the feature space. For example, when a host typically uploads about 10 GB of data per day, with a default 1 GB variance, then their Distance vector will always be more than a host that uploads usually a few MB per day and then suddenly uploads 100 MB.

Now as we see in figure 12 we start with a B(w) calculation, as we see in the equation (10) B(w). is B(w)for each local address is the average value of its $\bar{X}$ in the same time period and $\bar{X}$ also has in the previous time period.

$$B_{t-1}(W) = (Q_{50}(U_j x_j^1), Q_{50}(U_j x_j^2), Q_{50}(U_j x_j^3)) \quad (10)$$

where j ∈ {t − W − 1 , . . . , t − 1 } , and each i -th component of $\beta_{t-1}$ (W ) corresponds $x_t^1$ of feature to vector the median $x_t$.

Here in order to get more useful data the average is suggested to be calculated not for one previous time period, but for 14 previous time periods.



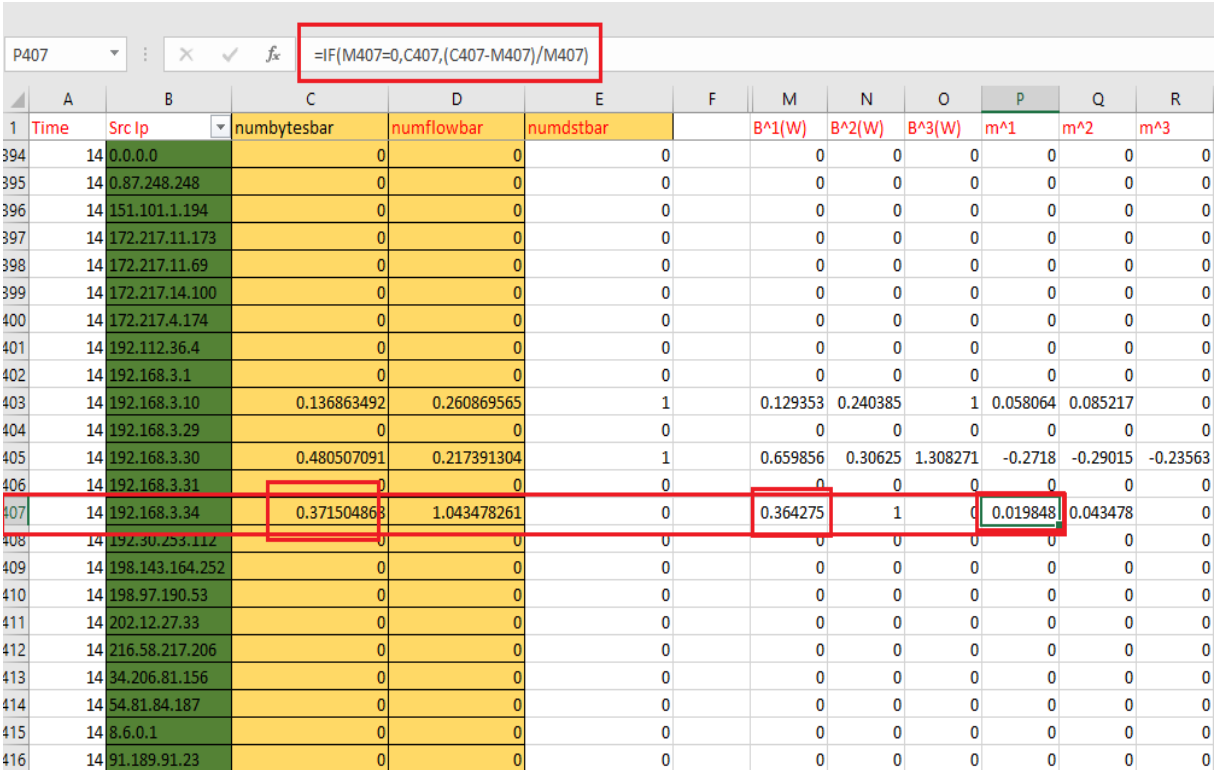

| | A | B | C | D | E | F | M | N | O |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Time | Src Ip | numbytesbar | numflowbar | numdstbar | | B^1(W) | B^2(W) | B^3(W) |
| 377 | 13 | 192.168.3.30 | 0.64069397 | 0.269230769 | 2 | | | | |
| 378 | 13 | 192.168.3.31 | 0 | 0 | 0 | | | | |
| 379 | 13 | 192.168.3.34 | 0.357241009 | 0.923076923 | 0 | | | | |
| 380 | 13 | 192.30.253.112 | 0 | 0 | 0 | | | | |
| 381 | 13 | 198.143.164.252 | 0 | 0 | 0 | | | | |
| 382 | 13 | 198.97.190.53 | 0 | 0 | 0 | | | | |
| 383 | 13 | 202.12.27.33 | 0 | 0 | 0 | | | | |
| 384 | 13 | 216.58.217.206 | 0 | 0 | 0 | | | | |
| 385 | 13 | 34.206.81.156 | 0 | 0 | 0 | | | | |
| 386 | 13 | 54.81.84.187 | 0 | 0 | 0 | | | | |
| 387 | 13 | 8.6.0.1 | 0 | 0 | 0 | | | | |
| 388 | 13 | 91.189.91.23 | 0 | 0 | 0 | | | | |
| 389 | 13 | 91.189.91.26 | 0 | 0 | 0 | | | | |
| 390 | 13 | 91.189.92.20 | 0 | 0 | 0 | | | | |
| 391 | sum | sum | 1.123279469 | 1.423076923 | 3 | | | | |
| 392 | count | count | 25 | 25 | 25 | | | | |
| 393 | C | C | 0.044931179 | 0.056923077 | 0.12 | | | | |
| 394 | 14 | 0.0.0.0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 395 | 14 | 0.87.248.248 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 396 | 14 | 151.101.1.194 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 397 | 14 | 172.217.11.173 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 398 | 14 | 172.217.11.69 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 399 | 14 | 172.217.14.100 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 400 | 14 | 172.217.4.174 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 401 | 14 | 192.112.36.4 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 402 | 14 | 192.168.3.1 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 403 | 14 | 192.168.3.10 | 0.136863492 | 0.260869565 | 1 | | 0.129353 | 0.240385 | 1 |
| 404 | 14 | 192.168.3.29 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 405 | 14 | 192.168.3.30 | 0.480507091 | 0.217391304 | 1 | | 0.659856 | 0.30625 | 1.308271 |
| 406 | 14 | 192.168.3.31 | 0 | 0 | 0 | | 0 | 0 | 0 |



***Figure 12.*** *S2 Calculation (last step)*

We note that the BW value is the average of $\bar{X}$ values for 14 previous time periods, For example, we are in time period 14 and at 192.168.3.30 specifically, we calculated the average $\bar{X}$ values in paragraph 13, 12, 11,...., 0 and so, as usual, the B(w)value must be calculated for all the $\bar{X}$ values, note: in the previous form we have identified all important fields and formulas in red to be easily understood. As B(w)has been calculated, we also calculate the value of Mt and can be calculated by the equation (11) for each of the three $\bar{X}$ values, which according to the equation is ($\bar{X}$ value for the same local address - B(w)value/ B(w) value). After we calculate the value of $\bar{X}$ and also calculate B(w)we will now be able to calculate mt value ,As in the figure 13.

$$m_t = \frac{x_t - B_{t-1}(w)}{B_{t-1}(w)} = \left(\frac{X_t^1 - B_{t-1}^1(w)}{B_{t-1}(w)}, \frac{X_t^2 - B_{t-1}^2(w)}{B_{t-1}(w)}, \frac{X_t^3 - B_{t-1}^3(w)}{B_{t-1}(w)}\right) \quad (11)$$




| | A | B | C | D | E | F | M | N | O | P | Q | R |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Time | Src Ip | numbytesbar | numflowbar | numdstbar | | B^1(W) | B^2(W) | B^3(W) | m^1 | m^2 | m^3 |
| 394 | 14 | 0.0.0.0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 395 | 14 | 0.87.248.248 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 396 | 14 | 151.101.1.194 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 397 | 14 | 172.217.11.173 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 398 | 14 | 172.217.11.69 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 399 | 14 | 172.217.14.100 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 400 | 14 | 172.217.4.174 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 401 | 14 | 192.112.36.4 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 402 | 14 | 192.168.3.1 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 403 | 14 | 192.168.3.10 | 0.136863492 | 0.260869565 | 1 | | 0.129353 | 0.240385 | 1 | 0.058064 | 0.085217 | 0 |
| 404 | 14 | 192.168.3.29 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 405 | 14 | 192.168.3.30 | 0.480507091 | 0.217391304 | 1 | | 0.659856 | 0.30625 | 1.308271 | -0.2718 | -0.29015 | -0.23563 |
| 406 | 14 | 192.168.3.31 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 407 | 14 | 192.168.3.34 | 0.371504863 | 1.043478261 | 0 | | 0.364275 | 1 | 0 | 0.019848 | 0.043478 | 0 |
| 408 | 14 | 192.30.253.112 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 409 | 14 | 198.143.164.252 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 410 | 14 | 198.97.190.53 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 411 | 14 | 202.12.27.33 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 412 | 14 | 216.58.217.206 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 413 | 14 | 34.206.81.156 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 414 | 14 | 54.81.84.187 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 415 | 14 | 8.6.0.1 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |
| 416 | 14 | 91.189.91.23 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |

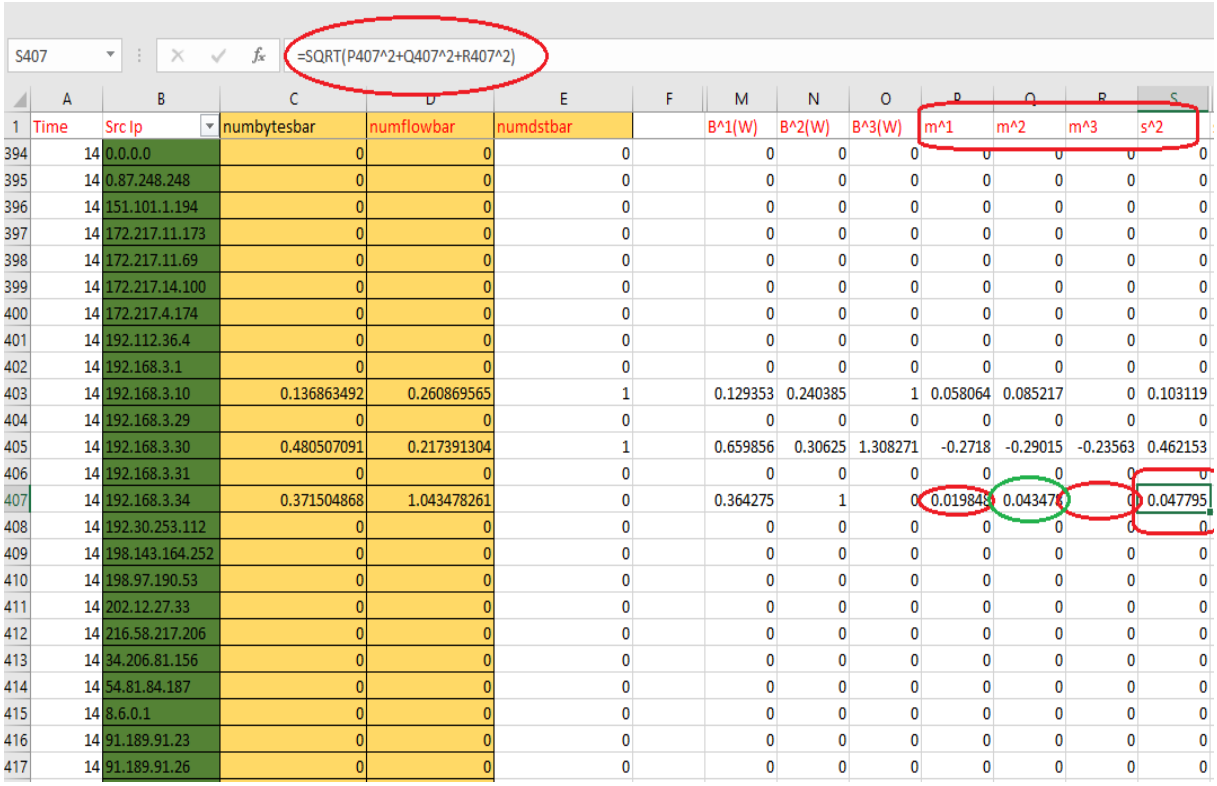
***Figure 13.*** *result of $B^1(w)$, $B^2(w)$, $B^3(w)$ and $M^1(t)$, $M^2(t)$, $M^3(t)$*

In the previous form at the top of the format specifically, we observe the mathematical formula of how Mt is calculated, and we added the IF function in order to get us the correct results. If the denominator is 0 because we know that it is not correct to divide by 0, and we said in this function if the value of the denominator that is B(w)is 0, the result of M(t) in that field is the value of $\bar{X}$ itself for the same row. We selected the address: 192,168.3.34 as an example in the previous format.

Note We have to calculate M(t) value for all three columns so we have $M^1(t)$, $M^2(t)$ *and* $M^3(t)$

Finally, we can calculate the value of the S2 as equation (12), which, according to the previous equation, is the square root of $M^1(t)$, $M^2(t)$, $M^3(t)$, as illustrated in figure 14.

$$s_t^2 = \|\boldsymbol{m}_t\| = \sqrt{\sum_{i=1}^{3}\left(\frac{x_t^i - B_{t-1}^i(w)}{B_{t-1}^i(w)}\right)^2} \qquad (12)$$



S407 =SQRT(P407^2+Q407^2+R407^2)

| | A | B | C | D | E | F | M | N | O | P | Q | R | S |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Time | Src Ip | numbytesbar | numflowbar | numdstbar | | B^1(W) | B^2(W) | B^3(W) | m^1 | m^2 | m^3 | s^2 |
| 394 | 14 | 0.0.0.0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 395 | 14 | 0.87.248.248 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 396 | 14 | 151.101.1.194 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 397 | 14 | 172.217.11.173 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 398 | 14 | 172.217.11.69 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 399 | 14 | 172.217.14.100 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 400 | 14 | 172.217.4.174 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 401 | 14 | 192.112.36.4 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 402 | 14 | 192.168.3.1 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 403 | 14 | 192.168.3.10 | 0.136863492 | 0.260869565 | 1 | | 0.129353 | 0.240385 | 1 | 0.058064 | 0.085217 | 0 | 0.103119 |
| 404 | 14 | 192.168.3.29 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 405 | 14 | 192.168.3.30 | 0.480507091 | 0.217391304 | 1 | | 0.659856 | 0.30625 | 1.308271 | -0.2718 | -0.29015 | -0.23563 | 0.462153 |
| 406 | 14 | 192.168.3.31 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 407 | 14 | 192.168.3.34 | 0.371504868 | 1.043478261 | 0 | | 0.364275 | 1 | 0 | 0.019845 | 0.043478 | 0 | 0.047795 |
| 408 | 14 | 192.30.253.112 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 409 | 14 | 198.143.164.252 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 410 | 14 | 198.97.190.53 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 411 | 14 | 202.12.27.33 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 412 | 14 | 216.58.217.206 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 413 | 14 | 34.206.81.156 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 414 | 14 | 54.81.84.187 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 415 | 14 | 8.6.0.1 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 416 | 14 | 91.189.91.23 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 417 | 14 | 91.189.91.26 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

***Figure 14.*** *excel calculation for* $S^2$

In the previous figure we deliberately worked out a determination on the three M(t) values as well as the output of the S2 value, and at the top of the picture, we can observe the mathematical formula in the Excel program for how to calculate the value of the S2.

## C. S3 Likelihood of movement direction in the feature space

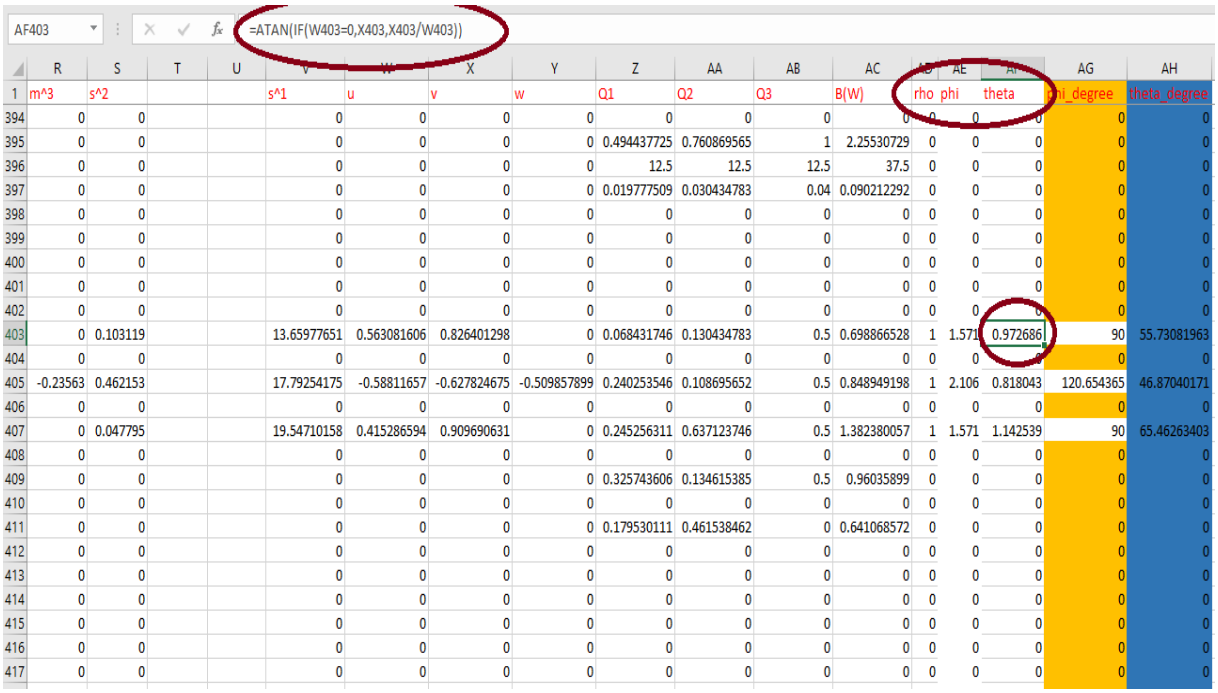
The magnitude of m t alone is not enough to declare and describe the suspicion of the movement of the host in the feature space. This score is designed to take the direction of the motion vector into consideration. Increased suspicion should be given to unusual instructions (i.e. low probability directions). For instance, a movement in a direction that increases the number of flows while the number of destinations falls unusually.

The direction of $\mathbf{m}_t$ related to an internal host is represented by the *unit vector* $\widehat{\boldsymbol{m}_t}$ that is defined as the following ratio:

$$\widehat{\boldsymbol{m}_t} = \frac{\boldsymbol{m}_t}{\|\boldsymbol{m}_t\|} = (\boldsymbol{u}_t, \boldsymbol{v}_t, \boldsymbol{w}_t) \qquad (13)$$

where $u_t$ , $v_t$ and $w_t$ are the components of the unit vector $\widehat{\boldsymbol{m}_t}$ .

here in Figure 15 we can see ρ, ϕ and θ calculation after calculating $\boldsymbol{u_t, v_t}$ ***and*** $\boldsymbol{w_t}$ in the top of the figure we can see the excel calculation formula for ρ, ϕ and θ.




| | R | S | T | U | V | W | X | Y | Z | AA | AB | AC | AD | AE | AF | AG | AH |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | m^3 | s^2 | | | s^1 | u | v | w | Q1 | Q2 | Q3 | B(W) | rho | phi | theta | phi_degree | theta_degree |
| 394 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 395 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0.494437725 | 0.760869565 | 1 | 2.25530729 | 0 | 0 | 0 | 0 | 0 |
| 396 | 0 | 0 | | | 0 | 0 | 0 | 0 | 12.5 | 12.5 | 12.5 | 37.5 | 0 | 0 | 0 | 0 | 0 |
| 397 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0.019777509 | 0.030434783 | 0.04 | 0.090212292 | 0 | 0 | 0 | 0 | 0 |
| 398 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 399 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 400 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 401 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 402 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 403 | 0 | 0.103119 | | | 13.65977651 | 0.563081606 | 0.826401298 | 0 | 0.068431746 | 0.130434783 | 0.5 | 0.698866528 | 1 | 1.571 | 0.972686 | 90 | 55.73081963 |
| 404 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 405 | -0.23563 | 0.462153 | | | 17.79254175 | -0.58811657 | -0.627824675 | -0.509857899 | 0.240253546 | 0.108695652 | 0.5 | 0.848949198 | 1 | 2.106 | 0.818043 | 120.654365 | 46.87040171 |
| 406 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 407 | 0 | 0.047795 | | | 19.54710158 | 0.415286594 | 0.909690631 | 0 | 0.245256311 | 0.637123746 | 0.5 | 1.382380057 | 1 | 1.571 | 1.142539 | 90 | 65.46263403 |
| 408 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 409 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0.325743606 | 0.134615385 | 0.5 | 0.96035899 | 0 | 0 | 0 | 0 | 0 |
| 410 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 411 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0.179530111 | 0.461538462 | 0 | 0.641068572 | 0 | 0 | 0 | 0 | 0 |
| 412 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 413 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 414 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 415 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 416 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 417 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

***Figure 15.*** *Calculating ρ, ϕ and θ.*

We note, however, that the unit vector distribution is not universal and some areas are significantly more populated than others. This means that motions are more frequent in some directions than motions in others.

It is convenient to use Spherical coordinates, derived from the components $\boldsymbol{u_t, v_t}$ and $\boldsymbol{w_t}$ through the following equations, in order to understand the distribution of unit vectors in internal hosts:

where $\rho \geq 0$ is the length (magnitude) of the vector, $0° \leq \phi \leq 180°$ and $-180° \leq \theta \leq 180°$ are two angles that describe the direction of the movement in the

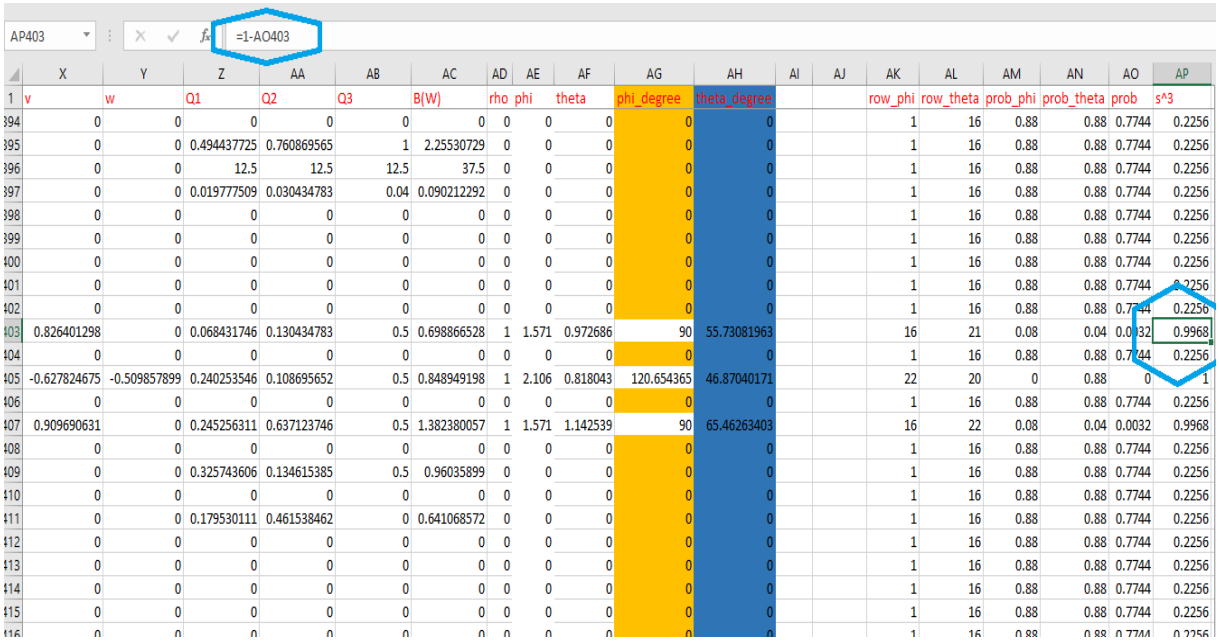
feature space. Since all unit vectors $\widehat{\boldsymbol{m}_t}$ have ρ= 1 by definition, only two variables (ϕ and θ) are required to represent the direction of the unit vector.

This space region captures movements for which all three features increase. In particular $u_t$ (corresponding to the number of up- loaded bytes) increases more than $v_t$ (number of destinations) and $w_t$ (number of flows), while $v_t$ and $w_t$ grow proportionally with respect to each other. We can conclude that a considerable increase in the number of uploaded bytes with respect to the previous his- tory is quite common, while strong increases in the number of destinations or in the number of flows are less frequent.

The third score $s_t^3$ is defined as:

$$s_t^3 = 1 - \Pr(\widehat{\boldsymbol{m}_t}) \qquad (14)$$

where $Pr(\widehat{\boldsymbol{m}_t})$ represents the probability of a certain direction in the feature space, computed as the value of a bin divided by the total number of internal hosts.




| | X | Y | Z | AA | AB | AC | AD | AE | AF | AG | AH | AI | AJ | AK | AL | AM | AN | AO | AP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | v | w | Q1 | Q2 | Q3 | B(W) | rho | phi | theta | phi_degree | theta_degree | | | row_phi | row_theta | prob_phi | prob_theta | prob | s^3 |
| 394 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 395 | 0 | 0 | 0.494437725 | 0.760869565 | 1 | 2.25530729 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 396 | 0 | 0 | 12.5 | 12.5 | 12.5 | 37.5 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 397 | 0 | 0 | 0.019777509 | 0.030434783 | 0.04 | 0.090212292 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 398 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 399 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 400 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 401 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 402 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 403 | 0.826401298 | 0 | 0.068431746 | 0.130434783 | 0.5 | 0.698866528 | 1 | 1.571 | 0.972686 | 90 | 55.73081963 | | | 16 | 21 | 0.08 | 0.04 | 0.0032 | 0.9968 |
| 404 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 405 | -0.627824675 | -0.509857899 | 0.240253546 | 0.108695652 | 0.5 | 0.848949198 | 1 | 2.106 | 0.818043 | 120.654365 | 46.87040171 | | | 22 | 20 | 0 | 0.88 | 0 | 1 |
| 406 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 407 | 0.909690631 | 0 | 0.245256311 | 0.637123746 | 0.5 | 1.382380057 | 1 | 1.571 | 1.142539 | 90 | 65.46263403 | | | 16 | 22 | 0.08 | 0.04 | 0.0032 | 0.9968 |
| 408 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 409 | 0 | 0 | 0.325743606 | 0.134615385 | 0.5 | 0.96035899 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 410 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 411 | 0 | 0 | 0.179530111 | 0.461538462 | 0 | 0.641068572 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 412 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 413 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 414 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 415 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |
| 416 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 |

***Figure 16.*** *Calculation of* $s_t^3$.

According to equation 14 the excel formula shown as illustrated in Figure 16 Hence, $1 - P\ r(\widehat{\boldsymbol{m}_t})$ is its *complement probability*, that represents the

*unlikelihood* of moving in a certain direction in the feature space. The higher $s_t^3$, the more suspicious is the direction followed by the host.

## D. Computation of the final score

The final step of our framework for APT detection is to compute the *final score* for each host of the internal network by combining the three suspiciousness scores described in previous sections.

The final *score* $S_t$ is computed as a *linear combination* of scores $s_t^1, s_t^2$ and $s_t^3$ . In particular, we adopt the following formula:

$$S_t = \sum_{j=1}^{3} (\delta_t^j, s_t^j) \tag{15}$$

Where $\delta_t^j$ is a normalization weight associated with the $j$ -th score, Since the three scores are characterized by different bounds, scales and distributions, we normalize them by defining $\delta_t^j$ through the *QWM* metric:

$$\delta_t^j = \frac{\sum_{k,k\neq j} QWM(s_t^j)}{\sum_k QWM(s_t^j)}, K \in \{1,2,3\} \tag{16}$$

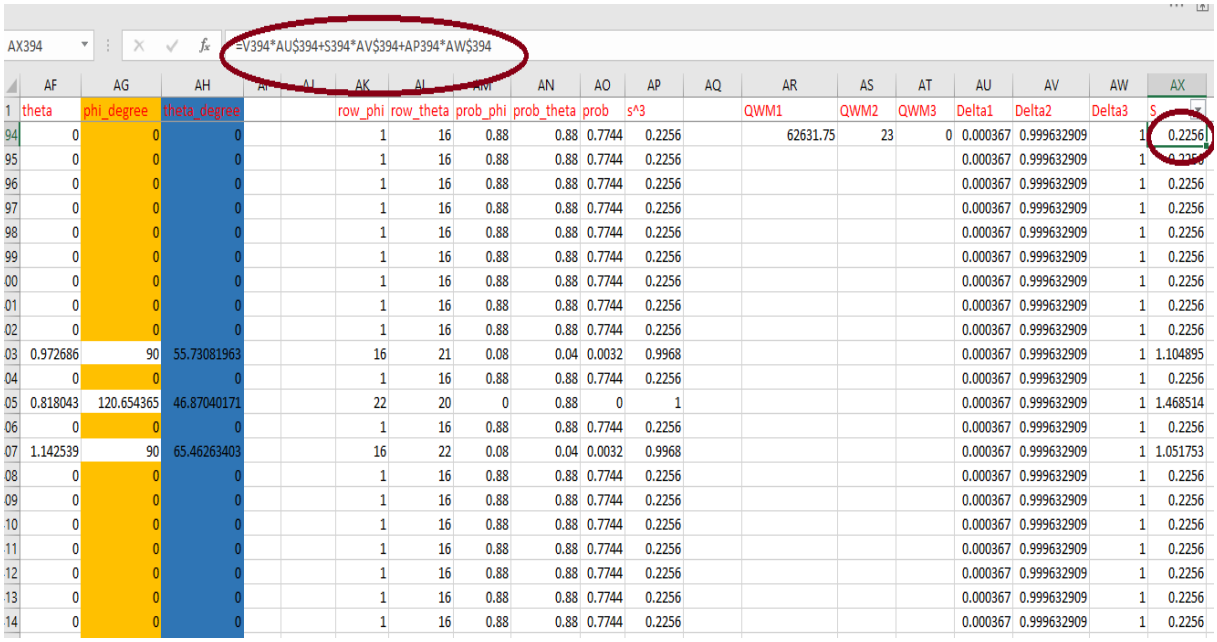
AU394 =((AS$394+AT$394)/(AR$394+AS$394+AT$394))

| | AF | AG | AH | AI | AK | AL | AM | AN | AO | AP | AQ | AR | AS | AT | AU | AV | AW |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | theta | phi_degree | theta_degree | | row_phi | row_theta | prob_phi | prob_theta | prob | s^3 | | QWM1 | QWM2 | QWM3 | Delta1 | Delta2 | Delta3 |
| 394 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | 62631.75 | 23 | 0 | 0.000367 | 0.999632909 | |
| 395 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 396 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 397 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 398 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 399 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 400 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 401 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 402 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 403 | 0.972686 | 90 | 55.73081963 | | 16 | 21 | 0.08 | 0.04 | 0.0032 | 0.9968 | | | | | 0.000367 | 0.999632909 | 1 |
| 404 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 405 | 0.818043 | 120.654365 | 46.87040171 | | 22 | 20 | 0 | 0.88 | 0 | 1 | | | | | 0.000367 | 0.999632909 | 1 |
| 406 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 407 | 1.142539 | 90 | 65.46263403 | | 16 | 22 | 0.08 | 0.04 | 0.0032 | 0.9968 | | | | | 0.000367 | 0.999632909 | 1 |
| 408 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 409 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 410 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 411 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 412 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 413 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 414 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 415 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |
| 416 | 0 | 0 | 0 | | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 |

***Figure 17.*** *Calculation of $\delta_t^j$ and QWM for all S's*

We note that the calculation of $\delta_t^j$ after calculating *QWM* values for $s_1$, $s_2$ and $s_3$ related to the different scores are normalized with respect to the sum of all *QWM* s as illustrated in Figure 17.

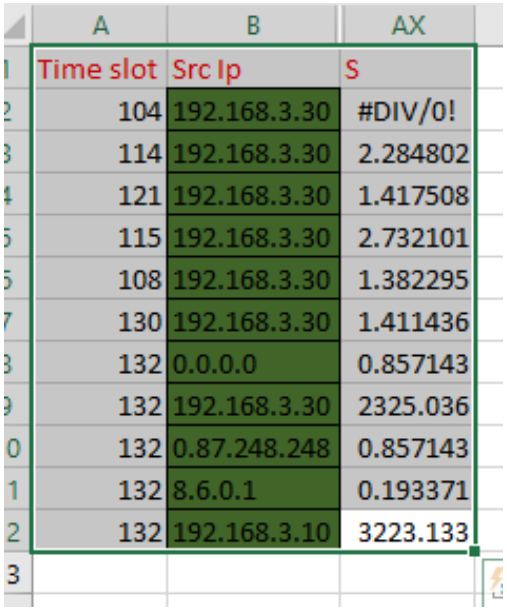
The last output of our framework is a descending list of internal hosts with a final result in $S_t$. Security analysts can use this list to prioritize a manual and time-based screening of suspicious internal hosts' net and system activities, here in Figure 18 is last score for final S after calculating $\delta_t^j$ and QWM.


AX394 =V394*AU$394+S394*AV$394+AP394*AW$394

| | AF | AG | AH | AK | AL | AM | AN | AO | AP | AQ | AR | AS | AT | AU | AV | AW | AX |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | theta | phi_degree | theta_degree | row_phi | row_theta | prob_phi | prob_theta | prob | s^3 | | QWM1 | QWM2 | QWM3 | Delta1 | Delta2 | Delta3 | S |
| 94 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | 62631.75 | 23 | 0 | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 95 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 96 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 97 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 98 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 99 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 00 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 01 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 02 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 03 | 0.972686 | 90 | 55.73081963 | 16 | 21 | 0.08 | 0.04 | 0.0032 | 0.9968 | | | | | 0.000367 | 0.999632909 | 1 | 1.104895 |
| 04 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 05 | 0.818043 | 120.654365 | 46.87040171 | 22 | 20 | 0 | 0.88 | 0 | 1 | | | | | 0.000367 | 0.999632909 | 1 | 1.468514 |
| 06 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 07 | 1.142539 | 90 | 65.46263403 | 16 | 22 | 0.08 | 0.04 | 0.0032 | 0.9968 | | | | | 0.000367 | 0.999632909 | 1 | 1.051753 |
| 08 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 09 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 10 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 11 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 12 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 13 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |
| 14 | 0 | 0 | 0 | 1 | 16 | 0.88 | 0.88 | 0.7744 | 0.2256 | | | | | 0.000367 | 0.999632909 | 1 | 0.2256 |

***Figure 18.*** *the score of final S*

Now we got the last result which is the final S results, according to this result any Source IP got high score of final S means this IP under APT attack or suffer from data exfiltration. Example in the last Figure 19 which contain the conclusion of all source IP in the exact time slot has high data exfiltration.



| | A | B | AX |
|---|---|---|---|
| 1 | Time slot | Src Ip | S |
| 2 | 104 | 192.168.3.30 | #DIV/0! |
| 3 | 114 | 192.168.3.30 | 2.284802 |
| 4 | 121 | 192.168.3.30 | 1.417508 |
| 5 | 115 | 192.168.3.30 | 2.732101 |
| 6 | 108 | 192.168.3.30 | 1.382295 |
| 7 | 130 | 192.168.3.30 | 1.411436 |
| 8 | 132 | 0.0.0.0 | 0.857143 |
| 9 | 132 | 192.168.3.30 | 2325.036 |
| 10 | 132 | 0.87.248.248 | 0.857143 |
| 11 | 132 | 8.6.0.1 | 0.193371 |
| 12 | 132 | 192.168.3.10 | 3223.133 |
| 13 | | | |

***Figure 19.*** *Example of data exfiltration according to time slot.*

## 4. Conclusions

The most important contribution to our paper is an attempt to find a mechanism to identify the attacks that can be classified as APT attacks. In our proposed mechanism we start searching read dataset then we start learning this dataset and trying to choose the important patterns that we can depend on for mining. Then we will start our mining after a good preparation because we know well the correct preparation of the data contributes to obtaining useful patterns at the end of the exploration process. Then we continue the calculation steps for $s_1$, $s_2$, $s_3$ where $s_1$ defines distance from the centroid of the feature space and $s_2$ defines the magnitude of the movement in the feature space finally $s_3$ defines unlikelihood of movement direction.

Th last part of this proposed framework contains  most important results of our project work. That is certainly the value of total S, through which we will be able to identify the internal host that is exposed to attacks or suffers from data exfiltration.

## 5. Reference

[1]- Eric Cole. Advanced Persistent Threat: Understanding the Danger and How to Protect Your Organization. Newnes, 2012. 320 pages. ISBN-1597499552.

[2]- Tyler Wrightson. Advanced Persistent Threat Hacking: The Art and Science of Hacking Any Organization. December 15th, 2014 by McGraw-Hill Education. ISBN- 978-0071828369.

[3]- The Most Famous Advanced Persistent Threats in History. itbusinessedge.com/slideshows/the-most-famous-advanced-persistent-threats-in-history-24.html

[4]- Advanced Persistent Threats: Learn the ABCs of APTs. https://www.secureworks.com/blog/advanced-persistent-threats-apt-a

[5]- Advanced Persistent Threat (APT) Lifecycle. https://www.varonis.com/blog/advanced-persistent-threat/y.

[6]- Diagram depicting the life cycle staged approach of advanced persistent threats (APTs). https://commons.wikimedia.org/wiki/File:Advanced_persistent_threat_life cycle.svg

[7]- Top Four Mitigation Strategies to Protect Your ICT Systems. https://www.ncsc.govt.nz/assets/NCSC-Documents/NCSC-Top-4-Mitigation-Strategies-To-Protect-Your-ICT-System.pdf

[8]- Cyber Attack on Fortune 500 Companies Infiltrates Employee Conversations To Succeed. https://www.sosdailynews.com/news.jspx?articleid=EC1E812DD9E18D742998BCBDD0FF59AD

[9]- Are Fortune 500 companies failing at cyber security. securitybrief.co.nz/story/are-fortune-500-companies-failing-at-cyber-security

[10]- Marchetti, M. 2016. Analysis of high volumes of network traffic for advanced persistent threat detection. Computer Networks, 109: p. 127-141.

[11]- Shashanka, M., Shen, M. Y., & Wang, J. (2016). User and entity behavior analytics for enterprise security. Proceedings - 2016 IEEE International Conference on Big Data, Big Data 2016. https://doi.org/10.1109/BigData.2016.7840805.

[12]- Chen, X., Zeng, X., Wang, W., & Shao, G. (2017). Big Data Analytics for Network Security and Intelligence. Gongcheng Kexue Yu Jishu/Advanced Engineering Science. https://doi.org/10.15961/j.jsuese.201600352

[13]- Salitin, M. A., & Zolait, A. H. (2018). The role of user entity behavior analytics to detect network attacks in real time. 2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2018. https://doi.org/10.1109/3ICT.2018.8855782.

[14]- Brook, C. (2020). What is User and Entity Behavior Analytics? A Definition of UEBA, Benefits, How It Works, and More | Digital Guardian. Data Protection 101. https://digitalguardian.com/blog/what-user-and-entity-behavior-analytics definition-ueba-benefits-how-it-works-and-more.

[15]- Kongsgard, K. W., Nordbotten, N. A., Mancini, F., & Engelstad, P. E. (2017). An internal/insider threat score for data loss prevention and detection. IWSPA 2017 - Proceedings of the 3rd ACM International Workshop on Security and Privacy Analytics, Co-Located with CODASPY 2017. https://doi.org/10.1145/3041008.3041011.

[16]- Lepide. (2020). How do SIEM Solutions Help Mitigate Advanced Persistent Threats (APT)? Data Security. https://www.lepide.com/blog/how-do-siem-solutions-help-mitigate-advanced-persistent-threats-apt/.

[17]- Friedberg, I., Skopik, F., Settanni, G., & Fiedler, R. (2015). Combating advanced persistent threats: From network event correlation to incident detection. Computers and Security. https://doi.org/10.1016/j.cose.2014.09.006

[18]- Mirza, N. A. S., Abbas, H., Khan, F. A., & Al Muhtadi, J. (2015). Anticipating Advanced Persistent Threat (APT) countermeasures using collaborative security mechanisms. Proceedings - 2014 International Symposium on Biometrics and Security Technologies, ISBAST 2014. https://doi.org/10.1109/ISBAST.2014.7013108

[19]- Schultz, E. E. (2010). Security Information and Event Management (SIEM). In Encyclopedia of Information Assurance. https://doi.org/10.1081/e-eia-120046525

[20]- Cardenas, A. A., Manadhata, P. K., & Rajan, S. P. (2013). Big data analytics for security. IEEE Security and Privacy. https://doi.org/10.1109/MSP.2013.138

[21]- Yadav, T., & Rao, A. M. (2015). Technical aspects of cyber kill chain. Communications in Computer and Information Science. https://doi.org/10.1007/978-3-319-22915-7_40

[22]- Dalziel, H. (2015). Cyber Kill Chain. In Securing Social Media in the Enterprise. https://doi.org/10.1016/b978-0-12-804180-2.00002-6.

[23]- Hutchins, E. (2018). Cyber Kill Chain® | Lockheed Martin. Lockheed Martin.

[24]- Hutchins, E., Cloppert, M., & Amin, R. (2011). Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. 6th International Conference on Information Warfare and Security, ICIW 2011.

[25]- Varonis. (2020). What is The Cyber Kill Chain and How to Use it Effectively | Varonis. Data Security. https://www.varonis.com/blog/cyber-kill-chain/

[26]- Zaid A. Ali Al-Marhabi, LiRen Fa, FanZi Zeng, Ayeda G. Ali Al-Hamdi, "The Design and Evaluation of a Hybrid Compression Technique (HCT) for Wireless Sensor Network", JDCTA: International Journal of Digital Content Technology and its Applications, Vol. 5, No. 5, pp. 201 ~ 207, 2011.

[27]- Jasek, R., Kolarik, M. and Vymola, T. 2013. APT detection system using honeypots. in Proceedings of the 13th International Conference on Applied Informatics and Communications (AIC'13), WSEAS Press.

[28]- Welch, V. 2012. Security at the Cyber Border: Exploring Cybersecurity for International Research Network Connections.

[29]- Saud, Z., & Islam, M. H. (2015). Towards proactive detection of advanced persistent threat (APT) attacks using honeypots. ACM International Conference Proceeding Series. https://doi.org/10.1145/2799979.2800042.

[30]- Balzarotti, D. 2010. Efficient detection of split personalities in malware. in Network and Distributed System Security Symposium (NDSS).

[31]- Radmand, A. 2009. A ghost in software, [cited 2013 sep 21]; Course]. Available from: http://cs.columbusstate.edu/cae-ia/StudentPapers/radmand.azadeh.pdf.

[32]- Hamed, T., Ernst, J.B. and Kremer, S.C. 2018. A Survey and Taxonomy on Data and Preprocessing Techniques of Intrusion Detection Systems, in Computer and Network Security Essentials. 2018, Springer. p. 113-134.

[33]- Idika, N. and A.P. Mathur, A.P. 2007. A survey of malware detection techniques. Purdue University, p. 48.

[34]- Alkhalidi Sadam, WangDong and Al-Marhabi Zaid , " Sector-Based Charging Schedule in Rechargeable Wireless Sensor Networks" KSII Transactions on Internet and Information Systems, vol. 11, no. 5, pp. 2310-2345, 2017.

[35]- Mohammed Ali Mohammed MOQBEL, Wangdong, Al-marhabi Zaid Ali, "MIMO Channel Estimation Using the LS and MMSE Algorithm", IOSR

Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 12, Issue 1, Ver. II (Jan.-Feb. 2017), PP 13-22.

[36]- Alkhalidi, S.M.; Wang, D.; Al-Marhabi, Z.A. Adopting Sector-Based Replacement (SBR) and Utilizing Air-R to Achieve R-WSN Sustainability. Information 2017, 8, 70.

[37]- Maarof, M.A. and Osman, A.H. 2012. Malware Detection Based on Hybrid Signature Behaviour Application Programming Interface Call Graph. American Journal of Applied Sciences.

[38]- Kaur, G., Khurana, M., & Sethi, M. (2011). Intrusion detection system using honeypots and swarm intelligence. Proceedings of the International Conference on Advances in Computing and Artificial Intelligence, ACAI 2011. https://doi.org/10.1145/2007052.2007060

[39]- Sharma, P.K. 2017. DFA-AD: a distributed framework architecture for the detection of advanced persistent threats. Cluster Computing, 20(1): p. 597-609.

[40]- Chakkaravarthy, S.S., Vaidehi, V. and Rajesh, P. 2018. Hybrid Analysis Technique to detect Advanced Persistent Threats. International Journal of Intelligent Information Technologies (IJIIT), 14(2): p. 59-76.

[41]- Haq, T., Zhai, J. and Pidathala, V.K. 2017. Advanced persistent threat (APT) detection center. 2017, Google Patents.

[42]- Alshamrani, A. 2019. A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities. IEEE Communications Surveys & Tutorials.

[43]- Kreibich, C., & Crowcroft, J. (2004). Honeycomb - Creating intrusion detection signatures using honeypots. Computer Communication Review. https://doi.org/10.1145/972374.972384

[44]- Zaid A. Ali Al-Marhabi, LiRen Fa, FanZi Zeng, Maan Younus Abdullah Alfathi, "HCT Plus based on diminishing WSN Energy Consumption",

JDCTA: International Journal of Digital Content Technology and its Applications, Vol. 6, No. 3, pp. 45 ~ 53, 2012.

[45]- Zaid A. Ali Al-Marhabi, LiRen Fa, FanZi Zeng, Maan Younus Abdullah Alfathi, Alhamidi Radman , "Achieving WSN Performance and Forest Monitoring System With WSC", IJACT: International Journal of Advancements in Computing Technology, Vol. 4, No. 12, pp. 77 ~ 84, 2012.