****

**ADANA ALPARSLAN TÜRKEŞ SCIENCE AND TECHNOLOGY UNIVERSITY**

**FACULTY OF ENGINEERING**

**DEPARTMENT OF COMPUTER ENGINEERING**

**CEN 427 CYBER SECURITY**

**HARDWARE PERFOURMANCE COUNTER FOR MALWARE DETECTION**

**YEKTA BUYUKKAYA 180101007**

**KADIR GUNEY 180101006**

**ISMAIL DEREEVLI 170101024**

**ADANA 2021-2022**

**ABSTRACT**

As the world is technologically advancing, the growth of malware has grown increasingly. Hardware based malware detection systems has turned out to be efficient response to decrease the exploitability as they are less available to access for misuse.

Performance counters detects malware projects and the applications can be further characterized with high precision. In this project we are aimed to detect malware using hardware performance counters and other hardware attributes. Run the virus on linux than get the the metrics using Perf. We trained the malware data on Python and created working Demo .

**INTRODUCTION**

Malware is a general term used to refer to various forms of hostile or malicious programs, including viruses,worms, Trojan horses, spyware, and so on.Due to the rapid growth in the number and the use of stealth techniques, malicious programs have become one of the most common security threats on the Internet.Development of malware has been a really significant issue thanks to daily innovations.McAfee reports that there are 75 million malware cases in 2017.Thus, anti-malware systems are to be implemented in abundance to secure the systems from these threats.

Recently, the information available in hardware has been used for behavioral malware detection.The basic idea - such as the idea presented in this report - is to use Hardware Performance Counters to build models of benign and malicious programs.It has been argued that if the hardware-level detection is desirable or not because it is harder for a malicious program to defeat hardware-based malware detector.

In this report, we use Hardware Performance Counters for malware detection and to use the information inside Hardware Performance Counters, we use “perf” library available on Linux-based operating systems.We run viruses on Ubuntu that we collected, get their HPC data using “perf” and apply Random-Forest and Decision Tree algorithms to test if HPCs can detect them as malware or not.

**DATA COLLECTION**

We had a hard time finding malware on linux. After a long search on the internet we found github link <https://github.com/MalwareSamples/Linux-Malware-Samples>. And also utilize malware bazaar for .elf extension. Download these malwares on Ubuntu. Decided to run on VM because of the malware can effect our computers but the Perf library does not support some HPC metrics as it comprehends that it’s in an Virtualize Environment also because the virtual machine only simulates HPC data we figured it would not be useful. So we decided run it these malwares old, don’t use computer

The benign programs were common Linux programs, such as cat, cp, ls, top, mozilla,libra office,judge ls while malicious programs were common Linuxbased viruses, Trojans,rootkit,worm,virus and backdoors, such as Clifax.A, Svat.B, Mirkforce.A, and Small.R. Moreover

We have 5 different virus type:

***Worm***: Worm is similar to the virus malware where it replicates the malware files and will cause the same damage as virus. one of the dangerous malwares compared to all the types and it will enter through the weakness of the system and it takes advantage of the files exchange feature.

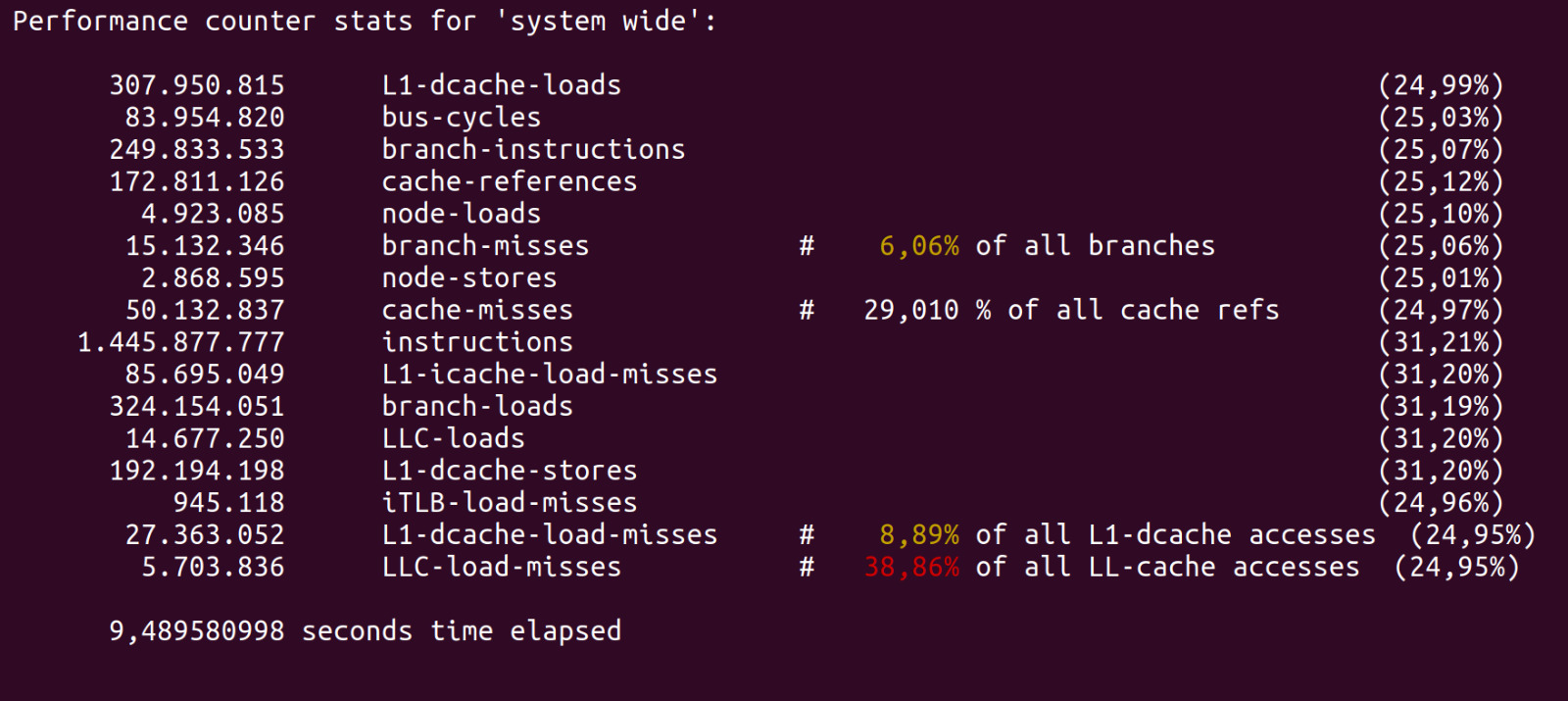
***Trojans:*** Trojans malware looks as a legitimate software and it tricks the user to load and to execute the files, the malware name has been derived from the wooden horse that Greeks used to destroy the troy. Once the trojan malware is activated it can achieve many numbers of attacks on the host system, from popping up windows multiple times to damaging the host by deleting files, stealing the data.

***Rootkit:***  It is a program or a collection of software tools which will give access and control of the computer to the threat actor. There have been many uses for this type of software to give remote end user support, often rootkits open a backdoor for the victim systems and introduces malicious software.

***Backdoor***: It is a method of avoiding the authentication, providing a secure remote access to the computer and access to the plaintext while the attempts to be undetected. It will be in the form of installing program or can be notification to the existing program

***Virus:*** Virus is something similar to the biology term, which we don’t need. It is a small set of instructions or a script which affects the systems health or the computer. It will spread from one computer to other computer and leaves malicious files as it spreads.

Running these malwares we collect 16 attributes.



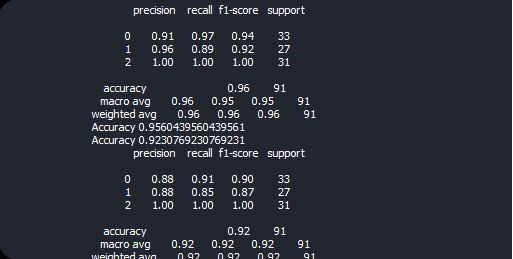
Than we added one more class which “malware type” it is and so we have 17 attiribute. We collected data 100 for each virus type and we collected their datas in 10 seconds intervals.

sudo perf stat -e L1-dcache-loads,bus-cycles,branch-instructions,cache-references,node-loads,branch-misses,node-stores,cache-misses,instructions,L1-icache-load-misses,branch-loads,LLC-loads,L1-dcache-stores,iTLB-load-misses,L1-dcache-load-misses,LLC-load-misses -a sleep 10

Using that code with via Perf library we collected the datas and saved in a file called “worm-dataset-final.xlsx”. Also we have capture a video how to do this.

**TRAIN DATA**

We trained our data on Python. Used Random Forest and Decision Tree. Before the train data apply the Label encoder for the get data numeric and use outliers method for clear outliers variables. So we can get high ratio.



Pictures shows the values from our Demo. First algorithm is Random Forest values and second one Decision Tree. We got precision,recall,f-1, and accuracy measurement. Choose virus type rootkit and trojan for train. Brief look the data measurement. We had 0.96 for Random Forest Algorithm and 0.95 Decision Tree Algorithm. This is highly ratio for accuracy we can count on it true.