Next-Generation Antivirus endowed with Web-Server SandBox Applied to Audit Fileless Attack

Sidney Lima (sidney.lima@ufpe.br)
UFPE: Universidade Federal de Pernambuco https://orcid.org/0000-0002-4350-9689
Sthéfano Silva
UPE: Universidade de Pernambuco
Ricardo Pinheiro
UPE: Universidade de Pernambuco
Danilo Souza
UPE: Universidade de Pernambuco
Petrônio Lopes
UPE: Universidade de Pernambuco
Rafael Lima
UPE: Universidade de Pernambuco
Jemerson Oliveira
UPE: Universidade de Pernambuco
Thyago Monteiro
UPE: Universidade de Pernambuco
Sérgio Fernandes
UPE: Universidade de Pernambuco
Edison Albuquerque
UPE: Universidade de Pernambuco
Washington Silva
UFPE: Universidade Federal de Pernambuco
Wellington Santos
UFPE: Universidade Federal de Pernambuco

Research Article

Keywords: Malware, Fileless Attack, php,Dynamic Runtime Behaviors,Artificial Neural Network,Computer Forensics

Posted Date: February 23rd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-390916/v1

Next-Generation Antivirus endowed with Web-Server SandBox Applied to Audit Fileless Attack

Mr. Sidney M. L. Lima[a,*,1] (Researcher), Sthéfano H. M. T. Silva[b], Ricardo P. Pinheiro[b], Danilo M. Souza[b], Petrônio G. Lopes[b], Rafael D. T. de Lima[b], Jemerson R. de Oliveira[b], Thyago de A. Monteiro[b], Sérgio M. M. Fernandes[b], Edison de Q. Albuquerque[b], Washington W. A. da Silva[c] and Wellington P.dos Santos[c]

[a] *sidney.lima@ufpe.br, Electronics and Systems Department, Federal University of Pernambuco, Recife, Brazil*
[b] *{rpp3,smll, dms2, shmts, pgl, rdtl, jro, tam, smurilo, edison}@ecomp.poli.br, Department of Computing, University of Pernambuco, Recife, Brazil*
[c] *{washington.silva,wellington.silva}@ufpe.br, Biomedical Engineering Department, Federal University of Pernambuco, Recife, Brazil*

ARTICLE INFO

Keywords:
Malware
Fileless Attack
php
Dynamic Runtime Behaviors
Artificial Neural Network
Computer Forensics

ABSTRACT

Background and Objective: Almost all malwares running on web-server are php codes. Then, the present paper creates a NGAV (Next Generation Antivirus) expert in auditing threats web-based, specifically from php files, in real time.

Methods: In our methodology, the malicious behaviors, of the personal computer, serve as input attributes of the statistical learning machines. In all, our dynamic feature extraction monitors 11,777 behaviors that the web fileless attack can do when launched directly from a malicious web-server to a listening service in a personal computer.

Results: Our NGAV achieves an average 99.95% accuracy in the distinction between benign and malware web scripts. Distinct initial conditions and kernels of neural networks classifiers are investigated in order to maximize the accuracy of our NGAV.

Conclusions: Our NGAV can supply the limitations of the commercial antiviruses as for the detection of Web fileless attack. In opposition of analysis of individual events, our engine employs authorial Web-server Sandbox, machine learning, and artificial intelligence in order to identify malicious Web-sites.

1. Introduction

The internet has been characterized as the main way of communication in contemporary society. The internet is notable for the convergence of all previously existing media. Through the World Wide Web, it is possible to watch television, listen to the radio, read the newspaper and have access to any other form of information among different peoples, languages and cultures.

With the popularization of the Internet, students create their own virtual study environment, provide their own content and can interact, actively and constantly, in the search for knowledge. The World Wide Web impels the creativity, technological abilities, and makes possible the opening of different visions, as well as the increase of communication and learning abilities.

As a side effect, the increasing popularization of the Internet propitiates that malware [1] production continues to growing in a fast way still during some years seen that internet is a large middle of propagation of malicious applications. Only in 2016, were created more than 7.1 million malwares, an increase of 47.3% compared to the year 2015. [9]. It is emphasized that most of the disorders caused by malwares are irreversible.

Therefore, more and more is being investing in digital security through new technologies in antivirus, firewall and biometrics. It is estimated that antivirus services are present in 95% of personal computers, in addition to 84% of Internet users having firewall services enabled and 82% have Automatic Updates enabled on Microsoft Operating System [16].

Therefore, more and more people are investing in digital security through new technologies such as antivirus, firewall and biometric technology. It is estimated that in addition to 84% of Internet users who have enabled firewall services and 82% of Internet users who have enabled automatic updates of the Microsoft operating system, 95% of personal computers have antivirus services [16].

Despite the massive presence of cyber-surveillance mechanisms in almost all personal computers, cyber-attacks have been causing billionaires damages on an increasingly larger scales [16]. One of the reasons for this failure is because once vulnerability is solved attackers try to come up with another tactic [24].

Currently, instead of conventional infections, through portable executable files (PE files), modern cyber attacks employ fileless attacks. Technically, fileless server-side attacks are launched directly from a malicious web-server to a listening service in an endpoint (personal computer) [6]. According to monitoring of Skybox Security during 2017, of the 55 new vulnerabilities exploited, only 24 percent were client–side vulnerabilities and 76 percent were server–side. The decline in client–side exploits reflects the trend in the decline in client–targeting exploit kits [22].

Symantec Research estimates that the main forms of cyber-infections, over the internet, are actually regular Web sites

*Principal corresponding author
sidney.lima@ufpe.br (S.M.L. Lima)
ORCID(s): 0000-0002-4350-9689 (S.M.L. Lima)
[1] malware (Malicious + Software)

that have been compromised or infected with malicious code [26]. The full list of most dangerous web-site attack categories can be seen in Figure 1. It is interesting to note that web-sites hosting adult/pornographic content are not in the top five, but ranked tenth. Moreover, religious and ideological sites were found to have triple the average number of threats per infectious site than adult/pornographic sites [26]. It is concluded that, regardless of their behavior, an Internet user is not safe of infections - since conventional advice is no longer useful as, for example, not accessing pornography sites aimed at avoiding cyber-invasions.

Once installed on compromised sites, malicious scripts dynamically launch drive-by attacks through web browsers in client-side (personal computers). Almost all malware running on server-side are php codes (a server-side scripting language commonly used on web-sites) [24].

In synthesis, server-side attacks, through php malwares, have the ability to deceive web-hosting providers and other cyber-surveillance mechanisms [24]. Then, the present paper investigates (i) 86 commercial antiviruses for the expertise of malicious phps. Malwares detection ranged from 0% to 78.50%, depending on the antivirus. It is emphasized that in our study, the analyzed malwares have their malicious performances documented by the incident responders. Even so, more than a half part of the commercial antivirus evaluated had no knowledge about the stocks of the malware files investigated.

In order to validate our authorial antivirus, the proposed paper develops (ii) a controlled environment named Web-Server Next Generation Sandbox. In our environment, the Server-side and the Client-side are developed in order to virtualize the malicious web-server and the personal computer, respectively. Then, the malicious behaviors, originating from the web fileless attack, serve as input attributes of the statistical learning machines. Our feature extraction monitors 11,777 behaviors that the fileless attack can do when launched directly from a malicious web-server to a listening service in a personal computer. Our NGAV (Next Generation Antivirus) solution can (re)construct a chain of events, visualize the actions that an actual attacker might take, as opposed to looking at individual, discreet events.

Our NGAV (iii) achieves an average performance of 99.95% in the distinction between benign and malware. So, the present paper demonstrates that artificial intelligence is a good alternative for commercial antivirus manufacturers. The limitations of cyber-security mechanisms can be procued by our antivirus expert in auditing web filelless attacks. Our engine employs advanced data science, machine learning, and artificial intelligence in order to identify malicious behavior from Server-side.

This work is organized as follows: in section 2 we present the limitations of commercial antiviruses. In section 3, we discuss the state-of-the-art regarding artificial intelligence antiviruses; in section 4, we present the proposed methodology; in section 5, we make a comparison between the authorial network and classic ones; in section 6, we show the results and some discussions. Finally, in section 7, we make the general conclusions and discuss the perspectives of our work.

2. Commercial Antiviruses Limitation

Despite being questioned more than a decade ago, the modus operandi of the antiviruses is based on signatures when the suspicious file is queried in blacklisted databases [10][20]. Therefore, it is enough that the hash of the investigated file is not in the blacklist of the antivirus so that the malware is not detected. The hash functions as a unique identifier for a specific file. So, understanding the limitations of commercial antiviruses, developing and distributing variants of malicious applications is not a difficult task. To do this, basically minor changes to the original malware, such as repetitive loops and conditional deviations, can be made to the original malware using routines that actually have no utility to the program, without instructions within its scope.

However, useless changes can make the hash value of the modified malware different from the hash value of the original malware. Therefore, antivirus software classified as original malware will not detect malware that increments by empty routines. It is worth noting that there are exploits that automatically create and distribute the original malware variants. We came to the conclusion that signature-based antivirus is not effective when attacked by variants of the same malware [10] [20].

Through the VirusTotal platform, the proposed paper investigates 86 commercial antivirus with their respective results presented in Table 1. We have utilized 200 malicious phps obtained from the PAEMAL dataset [17]. The aim of this paper is to verify the amount of virtual threats catalogued by the antiviruses. The motivation is that the acquisition of new virtual plagues plays an important role in combating malicious applications. Therefore, the larger the malware database, named blacklist, best tends to be the defense provided by the antiviruses.

Initially, the malware is sent to a server belonging to the VirusTotal platform. After that, php files will be analyzed through 86 commercial antiviruses associated with VirusTotal. Soon, the antivirus provides diagnostic information on the php files submitted to the platform. VirusTotal allows three different types of diagnostics to be issued; malware, benign, and omission.

As for the first possibility of VirusTotal, the antivirus detects the malignancy of the suspicious file. In the proposed experimental environment, all submitted files are malware documented by the incident responders. Soon, the antivirus hits when it detects the malignancy of the investigated file. Malware detection indicates that the antivirus provides a robust service against cyber-intrusions.

In the second possibility, antivirus software proves the benignity of the file under investigation. Therefore, when the antivirus software declares the benignness of the file, since all samples are malicious, it is a false negative. This means that the file under investigation is malicious software, but the antivirus software has proved its benign in the wrong way.

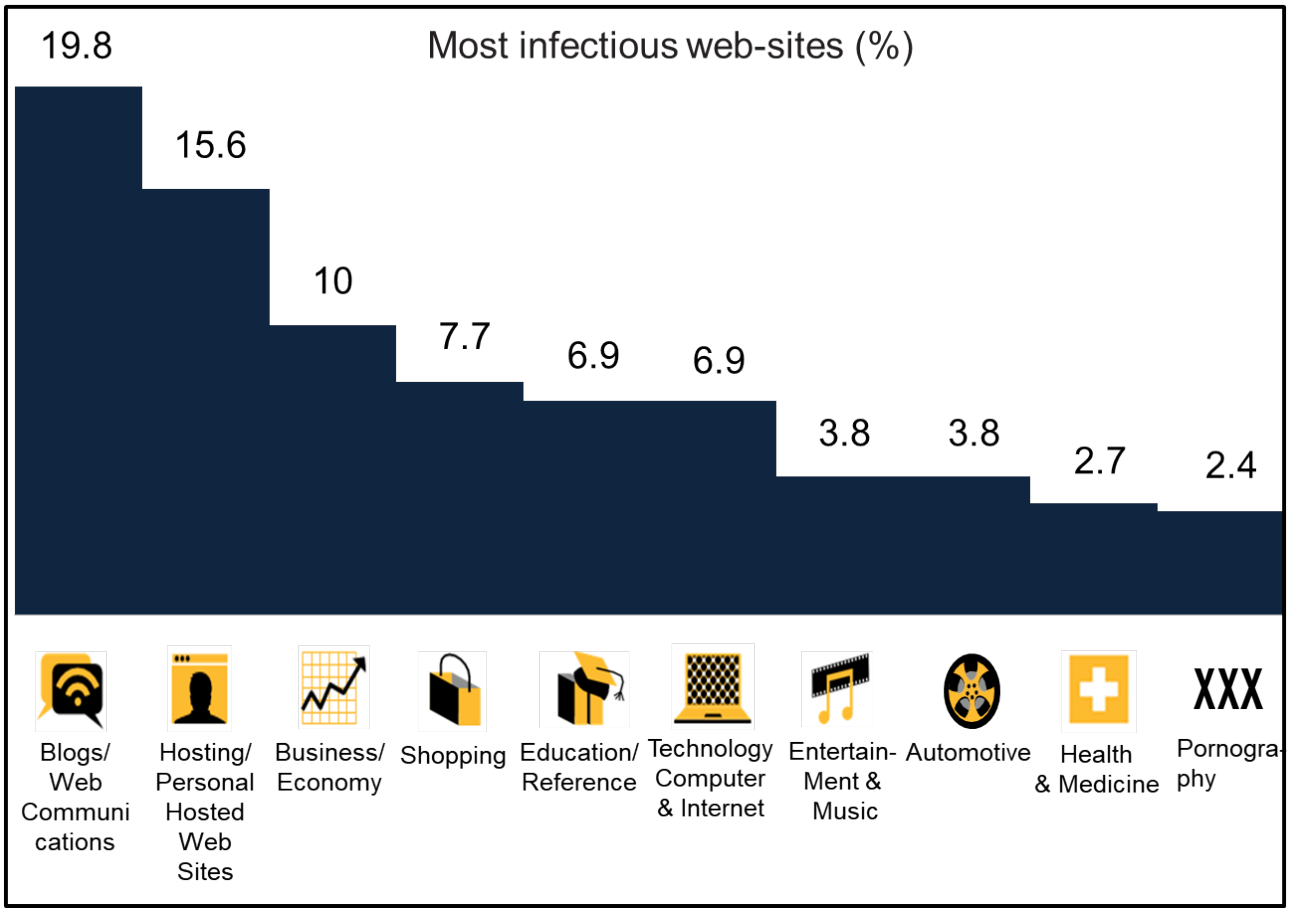

Figure 1: Most dangerous web-site attack categories according Symantec [26].

In the third possibility, antivirus software will not diagnose on suspicious files. Omission means that the investigated file has never been evaluated by an antivirus program, so the robustness of real-time evaluation of it is low. The antivirus software did not perform a diagnosis, which shows its limitations for large-scale services.

Table 1 shows the results achieved by the 86 commercial antiviruses evaluated. Ikarus antivirus obtained the best performance being able to detect 78.50% of the malware investigated. A major adversity is the fact that the antivirus manufacturers do not share their respective blacklists of malwares due to commercial disputes.

Through the analysis of Table 1, the proposed paper points to an aggravating factor of this adversity; the same antivirus manufacturer does not even share its databases among its distinct antiviruses. Note, for example, that Avast and AVG antiviruses belong to the same company. Their blacklists are not shared with each other. Therefore, the commercial strategies, of the same company, hinder the chartering of malwares. It is complemented that antivirus manufacturers are not necessarily concerned with avoiding cyber-invasions, but in optimizing their commercial incomes.

Malware detection ranged from 0% to 78.50%, depending on the antivirus investigated. On average, the 86 antiviruses were able to detect 16.82% of malware evaluated, with a standard deviation of 21.88 %. The high standard deviation indicates that the detection of malicious samples may change suddenly depending on the antivirus software selected. We concluded that the protection against network intrusion is due to the choice of powerful anti-virus software with a powerful latest blacklist.

On average, the antiviruses attested false negatives in 49.49% of the cases, with standard deviation of 38.32%. Atone to the benignity of a malware may imply unrecoverable losses. A person or institution, for instance, would rely on a certain malicious application when, in fact, it is a malware. Also as an unfavorable aspect, about 57% did not issue an opinion on any of the 200 malicious samples. On average, the antiviruses were omitted in 33.68% of the cases, with a standard deviation of 45.61%. The omission of the diagnosis points to the limitation of the antivirus regarding the detection of malwares in real time.

It includes as adversity, in the fight against malicious applications, the fact that commercial antivirus does not have a standard in the classification of malwares as seen in Table 2. We chose 3 of the 998 malwares to exemplify the miscellaneous of rankings given by commercial antivirus. Because there is no default, the antivirus will give them the names

Table 1
Results of commercial antiviruses. Expanded results of 86 worldwide commercial antiviruses are in the authorial repository [17].

Antivirus	Detection (%)	False Negative (%)	Omission (%)
Ikarus	78.50	16.00	5.50
GData	59.00	39.50	1.50
AegisLab	55.50	42.50	2.00
Avast	54.50	45.50	0.00
MAX	54.50	42.50	3.00
AVG	54.00	46,00	0.00
Kaspersky	50.50	48.00	1.50
ZoneAlarm	50.50	48.00	1.50
Avira	49.00	50.00	1.00
MicroWorld-eScan	47.00	53.00	0.00
BitDefender	47.00	51.50	1.50
Zoner	0.00	99.50	0.50
CrowdStrike	0.00	0.00	100.00
Alibaba	0.00	1.50	98.50
Agnitum	0.00	0.50	99.50
ByteHero	0.00	0.50	99.50
Norman	0.00	0.00	100.00
ahnlab	0.00	0.00	100.00
AntiVir	0.00	0.00	100.00
Commtouch	0.00	0.00	100.00
VirusBuster	0.00	0.00	100.00

they want. For example, McAfee-GW-Edition can identify a php malware such as "HEUR_HTJS. HDJSFN "and McAfee, belonging to the same company, identify it as "JS. Blacole. H ".

Therefore, the lack of a pattern hinders the cyber-surveillance strategies as each category of malware should have different treatments (vaccines). We concluded that it is not feasible to learn the supervised machine aiming to recognize the standard of php malware categories. Due to this confusing labelling of MultiClass Classification, provided by the experts (antiviruses) as seen in Table 2, it is statistically improbable that some machine learning technique acquires generalization capability.

3. State-of-the-art

A large difficulty in combating malicious phps, is that web-browsing and other web-based applications are real-time by nature [18]. Traditional signature-based detection engines often miss a large number of today's threats. While signature detection is great for known malware, detecting new forms with signature profiles is extremely difficult [18].

Given the limitations of commercial antiviruses, organizations seek to supply the shortcomings of traditional antiviruses through cyber-security mechanisms named NGAVs (Next Generation Antivirus). NGAVs solutions seek to recognize patterns of malware behaviors through advanced data science, machine learning, and neural networks [10][20]. The recommendation of the incident researchers is that the NGAVs add multiple layers of threat intelligence and advanced analytics [23].

LIMA, *et al.* (2021) create a NGAV able to detect PE files malwares with an average accuracy of 98.32% [11]. The executable is submitted to a disassembling process. Then, the executable can be studied, and therefore, it is possible to investigate the malicious intent of the file. Analysis made by LIMA, *et al.* (2021) extracts 630 features of each executable. These features are the input neurons of artificial neural networks. The classification of neural networks aims to group executables of 32-bit architectures into two classes: benign and malware. Antivirus made by LIMA, *et al.* (2021) employs shallow neural networks.

On the other side, deep nets-based antiviruses have also achieved excellent accuracies. SU, J. *et al.* (2018) achieve an average accuracy of 94.00% in order to detect IoT (Internet of Things) malwares [25]. The deep network structure has 6 layers. There are 3 layers with learnable weights: 2 convolutional layers, and 1 fully connected layer. The network is trained with 5000 iterations with a training batch size of 32 and learning rate 0.0001.

MANIATH, S. *et al.* (2017) create antivirus in order to detect ransomware by employing LSTM (Long-Short Term Memory) deep networks [15]. The training network consists of 3 layers with 64 LSTM nodes in each layer. The deep network is trained with training for 500 epochs with a batch size of 64. MANIATH, S. *et al.* (2016) achieve an average accuracy of 96.67%.

The disadvantage of the deep net is the long training time. As an aggravating factor, deep networks have lower parallel capabilities because these layers are continuous. Therefore, a layer can be executed only after the upper layer has completed its work. In applications that require frequent training (learning) as anti-virus software, this fact can be-

Table 2
Result of the submission of three malware to VirusTotal. Expanded results of 86 worldwide commercial antiviruses are in the authorial repository [17].

Antivirus	VirusShare_A	VirusShare_B	VirusShare_C
K7GW	Benign	Benign	Benign
K7AntiVirus	Exploit	Benign	Benign
Avast	Benign	Suspicious_GEN.F47V0405	Suspicious_GEN.F47V0614
AVG	JS:Decode-DB	php:Multicom-A	JS:Agent-DWO
AegisLab	Trojan.Script.Agent.dtkph	Benign	Benign
Avast	Benign	Suspicious_GEN.F47V0405	Suspicious_GEN.F47V0614
MAX	EXP/Blacole.EB.4	malware	malware
AVG	JS:Decode-DB	php:Multicom-A	JS:Agent-DWO
Kaspersky	Win.Trojan.Iframe-68	Backdoor.IRCBot.ADDS	JS:Trojan.JS.Agent.SJP
ZoneAlarm	Benign	Benign	Benign
Avira	Benign	Benign	Benign
MicroWorld-eScan	Benign	Benign	JS.TrojanjQuery.8C8B
BitDefender	Trojan.JS.Iframe.wq	Benign	HEUR:Trojan.Script.Generic
Ad-Aware	Benign	Benign	Benign
Emsisoft	Trojan.JS.IFrame.ANM	Benign	HTML/Phishing.m
ALYac	JS/BlacoleRef.E	JS.Redirector.AX	Benign
Baidu	Trojan.JS.IFrame.ANM	Benign	Benign
Bkav	Omission	Omission	Omission
McAfee-GW-Edition	HEUR_HTJS.HDJSFN	Benign	Benign
Arcabit	Exploit	TrojanDownloader :php/RunShell.A	Benign
McAfee	JS.Blacole.H	php/SillyDlScript.HFI	Benign
Antiy-AVL	Malware	php/Downloader.A	HTML/Infected.WebPage.Gen2

come an obstacle, because on average 8 new malwares are created every second [9]. In summary, there should be no difference in the learning time of antivirus software compared with the rate of new malware generation worldwide.

One of our specific goals is to prove the efficiency of antivirus with shallow neural networks compared to deep learning. The shallow neural network can obtain the same performance as the next-generation deep learning model after proper parameter setting and training [4]. In addition to training time, our shallow neural network-based antivirus can also provide statistically higher accuracy than the deep network-based antivirus with reduced computational cost.

Deep neural networks, especially convolutional networks work based on linear filter convolution. Although it has an important role in computer applications, filter convolution is limited to applications when forming vector flow gradients. Consider, for example, biomedical images from mammography equipment. The image is full of phenomena that interfere with the breast [12]. Then, the convolution of the filter is important in order to eliminate noise, and therefore, discard small irregularities in findings corresponding to potential cancers. The convolution technique as a Gaussian filter is very important for reducing noise in biomedical images.

As a counter-example, consider the repository illustrated in Table 3. The features are completely disconnected from each other despite belonging to the same neighborhood. An application suspected of trying to check Wi-fi data has no correlation with accessing the victim's image gallery or browser. Then, when applying the linear convolution of filters in the repository, illustrated in Table 3, accessing browser, containing the value 0, would be treated as noise. The explanation is that its neighborhood has positive values. In synthesis, the suspect application would be accused of accessing the victim's browser even the extraction of features having audited the inverse. Then, convolutional techniques suffers a disadvantage when applied to malware pattern recognition.

Despite the difficulties, deep nets-based antiviruses are able to obtain average accuracies superior to 90% in malware detection [15][25]. The suspect executable goes through a reverse engineering process aiming to revert the binary file in its assembly code. This methodology, named static analysis, can overcome the limitations of traditional signature-based detection engines.

However, static analysis can be easily bypassed by a web fileless attack. In synthesis, static feature approaches are invalid in order to combat fileless attack seen there is no way, for a personal computer, to audit source codes stored and executed on a remote web server

The inability of the static feature approach to accurately detect fileless attack shifted the focus of malware research to dynamic approach. Then, instead the impracticable static analysis, the extraction of features of our NGAV concerns the traces of calls performed by all processes spawned by the malware, files being created, deleted and downloaded by the malware during its execution, memory dumps of the malware processes, and network traffic trace in PCAP format.

In all, our dynamic feature extraction monitors 11,777 behaviors that a web fileless attack can do when launched directly from a malicious web-server to a listening service in a personal computer. As experiments, the authorial antivirus

has its accuracy compared to state-of-the-art antiviruses. In order to avoid unfair comparisons, feature extraction stage is standardized with our 11,777 behaviors in evaluated state-of-the-art antiviruses. Our antivirus can combine high accuracy with reduced learning time.

Table 3
Example of a statistical repository based on malware detection.

Features		
Check Wi-fi data	Access the Browser	Access Image Gallery
1	0	1

4. Materials and Methods

There are a variety of reasons why web-based malware presents such a challenge for traditional antivirus products [18]. One of these concerns the acquisition of the malware php file since it would be necessary to allow the web-hosting provider from where php was run remotely. However, in digital forensic practice web-hosting business often work in a disintegrated way and does not share information with cybersecurity companies [24]. Therefore, the action strategies of web hosting companies hinder and slow the coping with phps malwares.

It is increasing the use of malicious php scripts designed to make web-servers able to perform nefarious activities [24]. As adversity, web server-side attacks are already exceptionally difficult to catalog [24]. Due to the low-margin nature of the hosting business, when some hosting providers discover an infectious server, they often simply rebuild a new virtual server instance, rather than diagnosing what happened [24]. Since neither they nor their security partners understand what happened, the new instances often become rapidly infected as well [24].

So the proposed paper claims that it is necessary to integrate web-hosting providers and cyber-surveillance companies targeting Server-side malware sharing. The lack of information sharing is one of the main challenges in order to combat malwares. The level of insight that cyber-defenders have of cyber criminals' activities is considerably limited, and the identification of evolving tactics usually occurs after malicious campaigns start [9].

On the other hand, cyber-criminals have the privilege of accessing researches conducted by communities of enthusiasts who can download and use open source malwares [9]. This discrepancy in access to information, between cyber-defenders and cyber-criminals is on ascension and is technically named of asymmetrical cyberwarfare [9].

Then, the present paper employs the PAEMAL (php Analysis Environment Applied to Malware Machine Learning), a dataset which allows the classification of php files between malicious and benign. PAEMAL is composed of 200 php malware files and 1000 other benign php files. In regard to virtual plagues, PAEMAL extracted malicious php files from VirusShare [1]. In order to catalog the 200 samples of php malwares, it was necessary to acquire and analyze, by authorial script, about 1.3 million malwares from the reports updated by VirusShare daily.

Regarding the benign php files, the catalog was given from native scripts of open source tools such as phpMyAdmin [2]. It is emphasized that all benign files were submitted to the VirusTotal audit. Therefore, the samples of benign php files, contained in the PAEMAL, had their benevolence attested by the world's leading commercial antiviruses companies. The results obtained corresponding to the analyses of the benign php files and malwares, resulting from the audit of VirusTotal, are available for consultation in the virtual address of PAEMAL [17].

If there was no treatment in PAEMAL, there would be a tendency of higher hits in the majority class (benign) and high error rate in the minority class (malware). The explanation is because the number of benign and malware samples are unequal: 200 and 1000, respectively. Therefore, when employing unbalanced databases, the accuracy rates of the classifiers can be favored if they are tendentious in relation to the majority class. Aiming not to favor biased classifiers, the present work employs a strategy inspired by biomedical engineering works. In the health area, the presence of an abnormality (e.g. cancer) occurs every thousand diagnoses of healthy patients.

Then, the biomedical strategy concerns to repeating the training according to the ratio between the majority and minority classes (200:1000 = 5 iterations) [28]. In our paper, for each five iterations, a distinct package of 200 samples of the major class (benign) is presented to the 200 samples of the minority class (malware). In this way, the non-favoring of tendentious classifiers is guaranteed, allied to maintain the diversity of the different samples, from the majority class (benign), contained in the dataset [28].

In clinical practice, the absorption of a malignant sample (e.g., cancer) leads to a false negative. It is worth noting that the patient's chances of recovery are associated with early detection of the cancer. Then, the proposed paper is inspired by the state-of-the-art methodological care of biomedical engineering in order to reserve relevant amounts of benign and malware specimens in separate packages for training and testing. Therefore, assuming a sample reserved for testing with little or no instance of the malware class, then the classification, tendentious to the benign class, would have its favored hit rate. Thus, the proposed paper presents the methodological care to select equally, randomly, benign and malware samples destined for training and testing.

The purpose of PAEMAL dataset creation is to give a full possibility that the proposed methodology being replicated by third parties in future works. Therefore, PAEMAL freely makes available all its benign and malware samples:

[1] VirusShare is a repository of malware samples to provide security researchers, incident responders, forensic analysts, and the morbidly curious access to samples of live malicious code. Available at: https://virusshare.com/. Accessed on Nov. 2020.

[2] phpMyAdmin: open source administration tool for MySQL. Available in: https://www.phpmyadmin.net/. Accessed on Nov. 2020.

- Virustotal audits,
- Dynamic analysis made by our Web-Server Next Generation Sandbox,

In its virtual address, PAEMAL also provides its 1000 benign php files. In addition, our dataset displays the relationship of all other 200 php files, this time, malwares. Then, there is the possibility of acquiring all malwares, employed by PAEMAL, through the establishment of agreement and submission to the terms of use of ViruShare. It is concluded that our PAEMAL database enables transparency and impartiality to research, in addition to demonstrating the truthfulness of the results achieved. Then, PAEMAL is expected to serve as a basis for the creation of new scientific works targeting new Web-Server Next Generation Antiviruses.

5. Proposed Methodology

Figure 3 shows the diagram of the methodology proposed in block diagram. Initially, an web application is created employing a suspicious php script on Server-side. Then, the client requests the suspected Web page from Server-side. From there, malicious behaviors, originating from web-fileless attack, are audited in Windows 7 by our Web-Server Next Generation Sandbox. In following stage, the dynamic features of php files are stored in the format of machine learning repository. As method of features mining, some behaviors audited by our Sandbox are despised. The adopted criterion of mining refers to the elimination of features which concern a single php file, for example, process IDs, process names, md5, sha, among others.

PAEMAL presents 200 and 1000 malignant and benign php files, respectively. If there was no treatment in PAEMAL, there would be a tendency for greater accuracy hits in the majority (benign) class and high error rate in the minority class (malware). Thus, the proposed methodology employs a strategy inspired in the biomedical engineering state-of-the-art [28]. For five iterations, a distinct package of 200 copies of the major class (benign) is presented to the 200 copies of the minority class (malwares). After the database balancing, the suspicious behaviors of the php files serve as input attributes of the artificial neural networks employed as classifiers. The goal is to group the php files into two classes; benign and malwares.

In each combination (200 benign: 200 malwares) from the dataset balancing, the k-fold cross validation method is used, where k=10. In the first iteration, the first part is destined to the test set, while the others are reserved for the training. This alternation occurs for ten iterations until all ten parts have been applied to the test phase. The accuracy of the classifier is the arithmetic mean of the hit rate obtained in the ten iterations.

5.1. Web-Server Next Generation Sandbox

Sandboxes are excellent controlled environments in order to audit suspicious files [3]. The actions audited by the Sandbox refer to alterations in the OS registry, traces of calls performed by all processes spawned by the malware file, files being created, deleted and downloaded by the malware during its execution, memory dumps of the malware processes, and network traffic trace.

There are also Sandboxes which only accept auditing through their web-sites, in this case, it is enough that the user uploads the suspicious file, obviously, present in his/her computer. Although they play a key role in digital forensics, the Sandboxes employed by the state-of-the-art do not present mechanisms to audit a fileless attack.

In order to validate our NGAV, the present work develops a controlled environment named Web-Server Next Generation Sandbox. Our goal is to monitor the harmful effects of web-fileless attacks launched directly from malicious web-servers to listening services in personal computers. In our environment, we have developed the Server-side and the Client-side in order to virtualize the malicious Server-side and the personal computer, respectively. Our Server-side consists of:

- Linux OS: Linux is the underlying operating system running a large percentage of the Internet's web servers including many of the world's most important, highest volume, always-connected websites [24].
- Php interpreter: Security Threat Report has been identifying that the large majority of web-exploits are malicious php scripts designed to make Linux servers operate as nodes in nefarious activities [24]. Php scripts are dynamic web codes executed on the Server-side, unlike HTML that is static and it is executed on the Client-side (browser). Php is invoked on the server by the client and has its source code executed remotely. Therefore, its result, including HTML, is forwarded to the client for its use.
- MySQL Server: php works as a complement of the Web server adding new functionalities, much used for database querying in MySQL language, and widely used in the persistence of financial transactions and personal data of internet users. MySQL Servers are employed in a vast amount of web applications like e-commerce, social networks, blogs and human resource control portals.
- Apache HTTP Server: Apache Server is responsible for executing the HTTP protocol, the basis for communication on the World Wide Web. HTTP uses the Client-server model based on the request and response paradigm. Initially, the Client-side tries to establish a connection with Server-side by sending a request. The server then responds with the requested content if the connection succeeds, in addition to the other technical information on the server, such as the protocol version. After sending the response by the server, the established connection is closed.

In the Client-side, the Windows 7 SP1x86 Operating System is used endowed with the following facilities:

[3] Cuckoo: Automated Malware Analysis. Available in: https://cuckoosandbox.org/. Accessed on Nov. 2020.

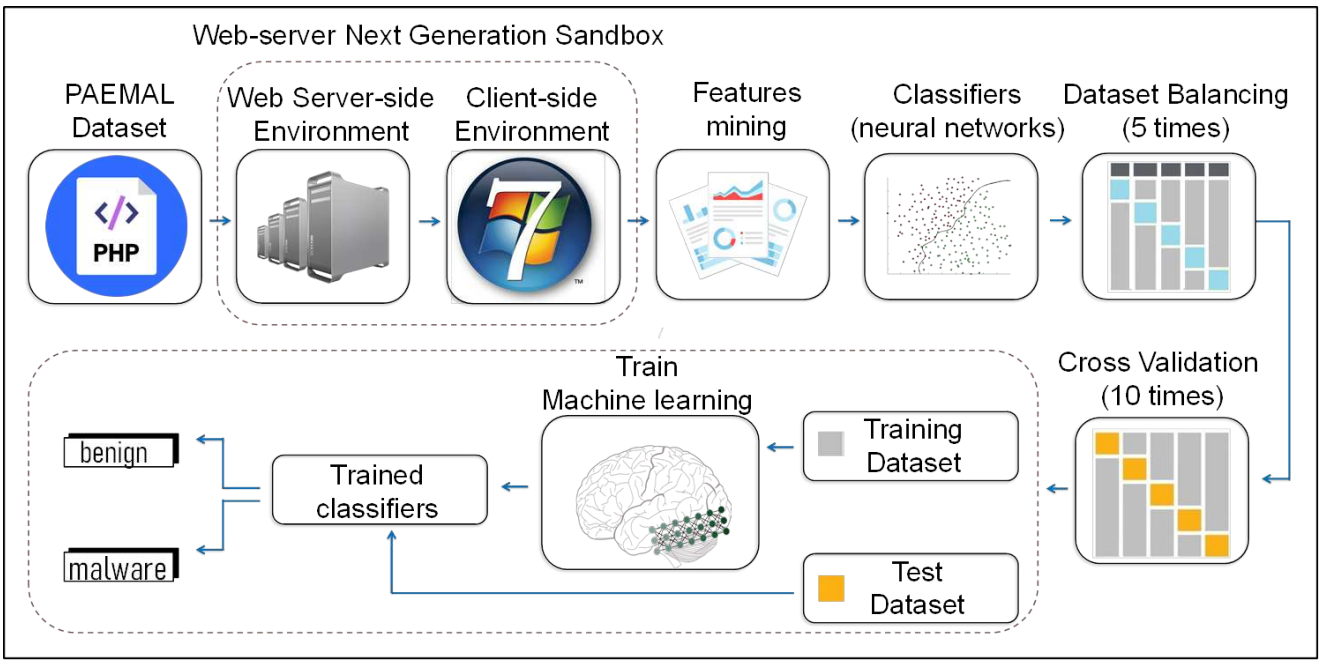

Figure 2: Diagram of the proposed methodology.

- Java Virtual Machines: fileless exploits, commonly, aim to corrupt the JVM and consequently affect the Java Security Manager [8]. The reason is that Java Security Manager is a class that manages the external boundary of the JVM. Java Security Manager controls how Java application, executing within the JVM, can interact with resources outside the JVM (OS level, e.g.: Windows 7) [8].

- Edge default browser: even after the malignant webpage has been closed, its malefactions can persist due to web-browser infection. From the corruption of files linked to the execution of the web-browser, the victim starts to suffer nefarious activities such as web-browser redirection, automatic downloads of other malwares and robbery/ hijacking of social networking passwords [5].

- Adobe Reader: similarly, even after the malign webpage has been closed, there may be persistent malefactions in non-browser environments such as Adobe Reader. The explanation is that malicious scripts can be embedded in various file types such as pdf among others. In this way, hidden malware can be executed automatically when the document is opened by the victim [5].

- Microsoft Office: besides pdf files, malwares can also be embedded in various types of files such as rtf, doc, ppt, among others. Thus, the malware can be executed automatically when the document is opened by the victim through Microsoft Office. As strategy anti-forensic, the first stage script extracts another script into randomly named files on disk and creates a scheduled task to start it a minute later [27].

In our Web-Server Next Generation Sandbox, Client-side queries server-side through the standard URL [4] regarding the http protocol, local domain, and port 80 for TCP [5] in the transport layer. Then the suspicious php file is invoked on the server and has its code executed and therefore its result is forwarded to the Client-side for its use. From there, the malicious behaviors, originating from the Web-fileless attack, are audited in Windows 7 by our Sandbox. Forensics, conducted by our sandbox, employs authorial scripts, Cuckoo Sandbox scripts, Oracle software and VirtualBox virtual machine.

On average, our dynamic feature extraction monitors 11.777 behaviors that the fileless attackcan do when launched directly from a malicious web-server to a listening service in a personal computer. Our NGAV solution can actually (re)construct a chain of events, visualizing what the actual attacker might be up to, as opposed to looking at individual, discreet events. In the next sub-section, the features cataloged by our Web-Server Next Generation Sandbox will be described.

5.2. Dynamic Feature Mining

The features of php-type files originate through the dynamic analysis of suspicious files. Then, in our methodology, malware is executed in order to infect the Windows 7 audited, in real time (dynamic), by our Web-Server Next Generation Sandbox. The amount of dynamic features depends on the iterations from the database balancing. For five iterations, a distinct packet of 200 samples of the (benign)

[4] URL: Uniform Resource Locator
[5] TCP: Transmission Control Protocol

major class is presented to the 200 samples of the minority class (malware). Then, in the five iterations are audited 11767, 11786, 11802, 11764, and 11767 suspicious behaviors, respectively.

In our Web-Server Next Generation Sandbox, the number of features depends on the behavior of the audited files. On average, 11,777 features are generated regarding the monitoring of the suspect file in the proposed controlled environment. Next, the groups of features related to the controlled monitoring of the files investigated are detailed.

- Features related to Code Injection, a technique used by an attacker to introduce code into vulnerable programs and change their behavior. The auditory checks whether the tested server tries to:
  - execute a process and inject code while it is uncompressed;
  - injecting code into a remote process using one of the following functions: CreateRemoteThread or NtQueueApcThread.
- Features related to Keyloggers, programs that record all user-entered keyboard entries. Primary purpose is illegally capturing passwords and other confidential information.
- Features related to the search for other possibly installed programs. The goal is to verify if the audited server searches for:
  - discover where the browser is installed, if there is one in the system.
  - discover if there is any sniffer or a installed network packet analyzer.
- Features related to disable Windows components. Our NGAV checks if the tested server tries to disable any of the windows programs: CMD.exe, Device Manager, or Registry Editor, by manipulating the Windows Regedit.
- Features related to memory dump, process in which the contents of RAM memory is copied for diagnostic purposes. The proposed digital forensics audits if the server tries to find malicious URL's in memory dump processing.
- Features related to crypto-coin mining. Our NGAV verifies if the tested server tries to connect to mining pools, the goal is to generate virtual currencies without the cognition (and not benefiting) the computer owner.
- Features related to system modifications. Our NGAV verifies if the tested server tries to create or modify system certificates, security center warnings, user account control behaviors, desktop wallpaper, or values in the ADS (Alternate Data Stream).
- Features related to Microsoft Office. It checks if the server tested tries to:
  - create a suspicious VBA object
  - run Microsoft Office processes inserted in a command line interface packed object.
- Features related to packing and obfuscation. The proposed digital forensic verifies that the tested server:
  - has packet or encrypted information indicative of packing
  - creates a slightly modified copy of itself (polymorphic packing);
  - is compressed using UPX (Ultimate Packer for Executables) or VMProtect (software used in order to obfuscate code and virtualize programs).
- Features related to persistence, functionality of backup information in a system, without the need to register them before. Our Sandbox audit if suspicious server tries to use JavaScript in a registry key value in regedit.
- create an ADS (Alternate Data Stream), NTFS feature that contains information to find a specific file by author or title, used maliciously because as the information that is present in it does not change the features of the file associated with it, transforming them into an ideal option for the construction of rootkits, because they are hidden (esteganography).
- Feature related to POS (Point Of Sale), type of attack that aims to obtain the information of credit and debit cards of victims.
- Features related to powershell code injectors. Our Sandbox checks if the tested server attempts to create a suspicious powershell process;
- Features related to processes. Checks if the tested server:
  - is interested in some specific process in execution;
  - repeatedly searches for a process not found;
  - tries to fail some process.
- Features related to ransomwares, cyber-attacks that turn the computer data inaccessible, requiring payment in order to restore the user access.
- Features related to Windows 7 OS (Regedit):
  - Changes in associations between file extensions and software installed on the machine;
  - Changes to the current user information;
  - Driver corruption;
  - Changes to the Windows appearance settings and settings made by users, such as wallpaper, screensaver, and themes;

  - Changes to Hardware Settings.

- Features related to Trojans (malicious program that enters a computer masked as another program, legitimate) of remote access, or RAT (Remote Access Trojans).

- Features related to network traffic trace Windows 7 OS in PCAP format.

- Features related to DNS servers (Domain Name System, servers responsible for the translation of URL addresses in IP).

5.3. Neural Networks for Malware Pattern Recognition

As for malware pattern recognition, an essential task relates to assigning a class (label) to each file investigated from its features. So, based on a file setting, named training set, it is possible to formulate a hypothesis about the different classes linked to our NGAV. Therefore, it is up to the classifier, to estimate the class of an unprecedented file by comparing the features of its audited behavior in real time and those captured during the training stage. The present work employs artificial neural networks as classifiers.

In order to select the best configuration of the neural network architecture, diverse learning functions and initial configurations that require a bigger volume of computations are computed, such as doubling the number of neurons in the hidden layer. The neural network architectures have an input layer containing a number of neurons relative to the feature extraction vector of the fileless attack.

In order to select the best configuration of the neural network architecture, different learning functions and initial configurations that require a lot of computation are used, such as doubling the number of neurons in the hidden layer. The neural network architecture has an input layer, which contains many neurons relative to the feature extraction vector of the fileless attack. In arrange to select the finest arrangement of the neural organize design, diverse learning capacities and introductory setups that require a bigger volume of computations are utilized, such as multiplying the number of neurons within the covered up layer. The neural arrange designs have an input layer containing a number of neurons relative to the include extraction vector of the fileless assault.

As mentioned before, in the five iterations, from the database balancing, 11767, 11786, 11802, 11764 and 11767 suspicious behaviors are audited, respectively. Thus, the input layer of nets contains as many neurons as the audited behaviors in each of the iterations. They relate to the dynamic features coming from the fileless attack. The output layer has two neurons, corresponding to benign and malware classes.

Neural networks are computational intelligence models used to solve problems of pattern recognition having as main characteristic the power of generalization in front of data not presented to the network. In most of the neural networks, such as MLP (Multilayer Perceptron) [29], knowledge about network parameters is needed to obtain maximum performance in order to solve the problem. A common concern in this network type is avoid getting stuck in minor locations [7], being necessary to add network control methods to get rid of these regions. Another common characteristic in this type of network is the high training time required to make the network able to perform classifications correctly.

The ELM (Extreme Learning Machine) network has as main characteristic the training speed and data prediction when compared MLP neural networks. ELMs (Extreme Learning Machines) are powerful and flexible kernel-based learning machines whose main characteristics are fast training and robust classification performance [7]. The ELM network is a single hidden layer network, not recurrent, based on an analytical method to estimate the network output weights, in any random initialization of input weights.

The ELMs have been widely applied in several areas such as Biomedical Engineering [14][13][12] [19][1][2][3]. ELMs networks can greatly contribute to the advancement of digital security of devices. The proposed paper applies the ELMs in the area of information security specifically in the recognition of malware patterns.

Mathematically, in ELM neural network the input attributes x_{ik} correspond to the set $\left\{x_{it} \in \mathbb{R};\ i = 1, ...n;\ t = 1, ...v\right\}$. Therefore, there are n features extracted from the application and v training data vectors. The hidden layer h_j, consisting of m neurons, is represented by the set $\left\{h_j \in \mathbb{R};\ j \in N^*;\ j = 1, ...m\right\}$. The ELM training process is fast because it is composed of only a few steps. Initially, the input weights w_{ji} and bias $bias\ b_{jt}$ are defined in a random generation. Given an activation function f:R→R, the learning process is divided into three steps:

- Random generation of weight w_{ji}, corresponding to the weights between the input and the hidden layers, and bias $bias\ b_{jt}$.

- Calculate the matrix H, which corresponds to the output of the neurons of the hidden layer.

- Calculate the matrix of the output weights $= H \dagger Y$, where $H\dagger$ is the generalized Moore-Penrose inverse matrix of the matrix H, and Y corresponds to the matrix of desired outputs s.

The output of the hidden layer neurons, corresponding to the matrix H, is calculated through the kernel K, dataset inputs and weights between the input and the hidden layers, as shown in Eq. (1).

The proposed work resulted in an antivirus composed of ELMs neural networks seeking the preventive detection of malwares. Instead of using conventional kernels, authoring kernels will be used for ELMs. We employ mELMs (morphological ELMs), ELMs with hidden layer cores based on the morphological operators of Erosion and Dilation image processing. Kernels are mathematical functions employed as a method for learning neural networks. This learning method enables the creation of non-linear data mapping. Thus,

there is no need to increase the number of adjustable parameters, as in the learning rate used in networks with backward propagation.

There are two fundamental morphological operations, Erosion and Dilation. The theory of Mathematical Morphology can be considered constructive, because all operations are built based on Erosion and Dilatation. Mathematically, Erosion and Dilation are defined according to Eq. (1) and Eq. (2), respectively:

$$\varepsilon_g(f)(u) = \bigcap_{v \in S} f(v) \vee \overline{g}(u - v) \tag{1}$$

$$\varepsilon_g(f)(u) = \bigcup_{v \in S} f(v) \wedge g(u - v) \tag{2}$$

Where $f : S \to [0,1]$ and $g : S \to [0,1]$ normalized images in the form of a matrix named S format, where $S \in \mathbb{N}^2$. pixel is defined by the Cartesian pair $(u, f(u))$, where u is the position associated with the value $f(u)$. v is the matrix of $f(u)$, covered by g. The operators and are associated with the maximum operation, while and are associated with the minimum operation. g is the structuring element for both Erosion and Dilation [21]. $\overline{g}$ is the negation of g.

In Eq. (1) initially occurs the negation of the structuring element $\overline{g}$. Then, it happens the operation of maximum $\vee$ denoted by $f(v) \vee \overline{g}(u - v)$, where $f(v)$ refers to the original image matrix f covered (matched) by $\overline{g}$. $f(v)$ is technically named the active region of the image. Finally, the value $\epsilon_g(f)(u)$, in the position u, of the eroded image receives the minimum value between the maximums, via the operator. $\epsilon_g(f)(u)$ gets the value 0 associated with absolute black. Erosion overlays $\overline{g}$ to the original image f. The goal is that areas similar to $\overline{g}$ expand [21]. By associating 1's to absolute white and 0's to absolute black, Erosion increases the darker areas and eliminates the regions with greater intensity [21].

Eq. (2) shows the performance of the morphological dilation operation. Due to mathematical precedence, the minimum $\wedge$ operation denoted by $f(v) \wedge g(u - v)$, occurs, where $f(v)$ refers to the original image matrix f covered (matched) by g. Therefore, the value $\delta_g(f)(u)$, at the u, position, of the expanded image receives the maximum value between the minimums, through the $\cup$ operator. Dilatation superimposes the structuring element g on the original image f. The goal is that areas similar to g expand. By associating 1's with absolute white and 0's with absolute black, the dilation increases the areas with more intense tonality and eliminates the dark regions [21].

Our antivirus employs mELMs (Morphological Extreme Learning Machines). They are inspired by mathematical morphology based on non-linear operators of Erosion and Dilatation. According to Eq. (1) concerning the erosion image operator, the Erosion ELM kernel can be defined according to Eq. (3), where $\{i \in \mathbb{N}^*, i = 1, \ldots, n\}$, $\{j \in \mathbb{N}^*, j = 1, \ldots, m\}$, $\{t \in \mathbb{N}^*, t = 1, ..v\}$. So there are n neurons in the entry layer (without the bias), m neurons in the hidden layer, and v training data vectors.

$$K_\epsilon(t,i) = (x_{it} \vee \overline{w}_{ji}) + b_{jt} \tag{3}$$

Similar to the Erosion kernel, Eq. (4) defines the Dilation kernel inspired by Eq. (2) and referring to the morphological operator of Dilation.

$$K_\delta(t,i) = (x_{it} \wedge w_{ji}) + b_{jt} \tag{4}$$

In order to achieve good performance in ELMs, it is necessary to choose a kernel that is able to optimize the decision boundary for the presented problem as seen in Figure 3 (a). A Linear kernel gets great results when used to solve a linearly separable problem. However, when used to solve nonlinearly separable problems as shown in Figure 3 (b) for a sigmoid distribution, it does not perform satisfactorily.

Figure 3 (c), Figure 3 (d) show the performance of the mELM kernel Erosion and Dilation, with the respective accuracies of 95.05% and 99.50%. It is possible to notice when analyzing the figures that the mELMs have the capacity to accurately map the different distributions referring to different problems.

The effectiveness of our morphological neural networks is due to their ability to adapt to any type of distribution, since their mapping does not obey any conventional geometric figure. The mapping of decision border is made by their own training data, the very position in n-dimensional space that will determine whether that surrounding region belongs to class 1 or class 2. The n represents the number of neurons in the input layer. Therefore, our mELM kernel is able to naturally detect and model the n-dimensional regions divided into different classes by using Math Morphology.

6. Results of ELM Networks

We employ seven different kernel types for the ELMs neural networks. In the state-of-the-art, seven of these kernels are described by HUANG, et al, (2012), they are; Wavelets Transform, Sigmoid, Sine, Hard Limit and Tribas (Triangular Base Function) [7]. In addition, two authorial kernels are employed: Dilation and Erosion.

The Wavelets kernel have no hidden layer [7]. The calculations are based on the transformation of the input data and can work nearly to kernels containing architectures with hidden layers [7]. A good generalization capability of these kernels depends on an adjusted choice of parameters (C, γ) [7]. The cost parameter C refers to a reasonable equilibrium point between the hyperplane margin width and the classification error minimization in relation to the training set. The kernel parameter γ controls the decision boundary in function of classes [7]. There is no universal method in the sense of choosing the parameters (C, γ).

The best combination (C, γ) depends on the data set employed [7]. In the proposed paper, there is the investigation of the parameters (C, γ) inspired by the method proposed by HUANG, et al (2012), which consists of training

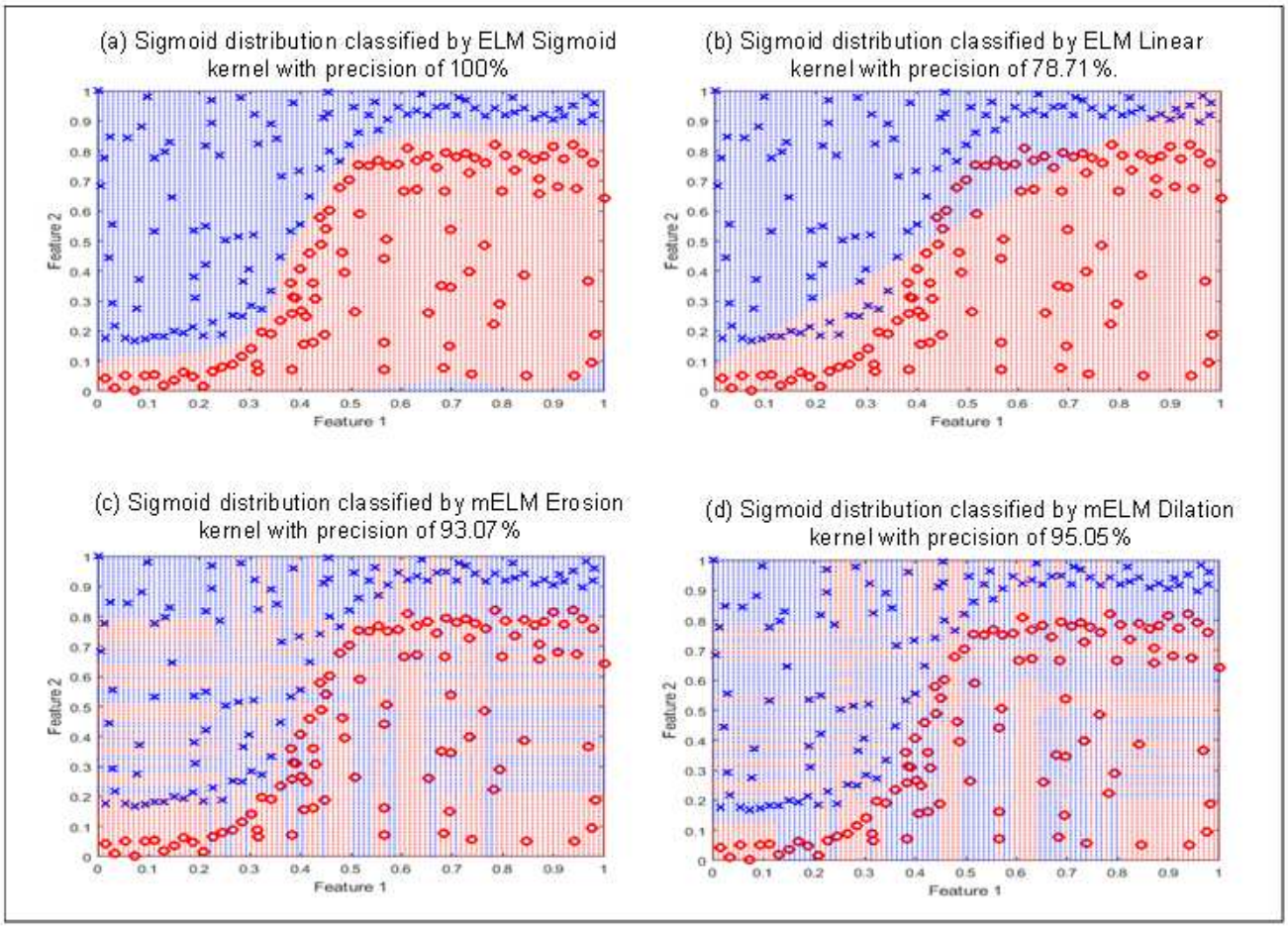

Figure 3: (a) Successful performance of the kernel compatible with dataset. (b) Inaccurate classification of the Linear kernel in a non-linearly separable distribution. (c - d) Successful performances by Dilation and Erosion kernels.

increasing sequences of C and γ, mathematically, 2^n, where $n = \{-24, -10, 0, 10, 25\}$ [7]. The hypothesis is to verify if these parameters with values different from the standards; $(C = 1, \gamma = 1)$, generate better results. In the Linear kernel, there is only the investigation of the cost parameter C, it is not possible to explore the kernel parameter γ [7].

Table 4 details the results obtained by the ELMs neural networks with Wavelets kernel. For five times, a separate package of benign samples (majority class) is presented to the package of malware samples (minority class). In each of these 5 times, we investigate 10 folds in relation to cross-validation of the k-fold method, where k=10. Then, 5*10 total 50 distinct executions in each row on Table 4. In relation to precision in the test phase, the maximum average performance was 57.05% in the distinction between benign and malware samples through the Wavelets kernel with the parameters $(C, \gamma) = (2^{10}, 2^0)$. In Table 4, there are only the best and worst case descriptions, in that order.

The Sigmoid, Sine, Hard Limit, Tribas, Dilation, and Erosion kernels employ hidden-layer architectures. Then, there is the investigation in relation to amount of neurons in the hidden layer of these kernels. The hypothesis is to verify if architectures that require a higher volume of calculations, such as doubling the number of neurons in the hidden layer, are able to generate better accuracies rates compared to architectures that require a smaller amount of calculations. There is the evaluation of two architectures; they employ 100 and 500 neurons in their respective hidden layers. These architectures have excellent accuracy in the application of ELM networks in Biomedical Engineering [12].

Table 5 details the results obtained by the ELMs neural networks with the Sigmoid, Sine, Hard Limit, Tribas (Triangular Basis Function), Dilation, and Erosion kernels. For five times, a separate package of benign samples (majority class) is presented to the package of malware samples (minority class). In each of these 5 times, we investigate 10 folds referring to the k-fold method, where k=10. Then, 5*10 total 50 distinct executions in each row of Table 5. Regarding precision, the maximum average performance was 99.95% with standard deviation of 0.05 through the Dilation kernel endowed with 500 neurons in its hidden layer. Then, our Dilation kernel suffers abrupt changes due to the initial conditions.

Table 4
Result of ELM Networks. The (C, γ) parameters vary according to the set $\{2^{-24}, 2^{-10}, 2^{0}, 2^{10}, 2^{25}\}$. There are only the best and worst case descriptions.

kernel	(C, γ)	Train rate (%)	Test rate (%)	Train time (sec.)	Test time (sec.)
Wavelets	$(2^{10}, 2^{0})$	**100.00 ± 0.00**	**57.05 ± 2.66**	0.52 ± 0.07	0.16 ± 0.04
	$(2^{25}, 2^{25})$	**100.00 ± 0.00**	47.15 ± 8.50	0.51 ± 0.07	0.15 ± 0.02

Table 5
Result of ELM Networks. The number of neurons in the hidden layer varies according to the set 100, 500.

kernel	neurons	Train rate (%)	Test rate (%)	Train time (sec.)	Test time (sec.)
Sigmoid	100	50.00 ± 0.00	50.00 ± 0.00	0.36 ± 0.05	0.01 ± 0.00
	500	50.00 ± 0.00	50.00 ± 0.00	0.76 ± 0.06	0.06 ± 0.02
Sine	100	73.84 ± 1.75	51.25 ± 8.29	0.34 ± 0.05	0.01 ± 0.00
	500	**100.00 ± 0.00**	50.40 ± 7.68	0.80 ± 0.06	0.06 ± 0.03
Hard limit	100	50.01 ± 0.00	49.92 ± 0.00	0.28 ± 0.03	0.00 ± 0.00
	500	50.01 ± 0.00	49.92 ± 0.00	1.36 ± 0.22	0.02 ± 0.01
Tribas	100	50.00 ± 0.00	50.00 ± 0.00	0.40 ± 0.08	0.01 ± 0.01
	500	50.00 ± 0.00	50.00 ± 0.00	0.80 ± 0.10	0.06 ± 0.02
Dilation	500	**100.00 ± 0.00**	**99.95 ± 0.05**	23.05 ± 1.80	2.24 ± 0.13
	100	**100.00 ± 0.00**	99.35 ± 0.66	4.23 ± 0.57	0.43 ± 0.05
Erosion	500	**100.00 ± 0.00**	99.65 ± 0.35	20.79 ± 0.81	2.30 ± 0.27
	100	85.78 ± 13.68	76.50 ± 15.12	3.64 ± 0.22	0.36 ± 0.03

7. Results in relation to the State-of-the-Art

In this section, the proposed antivirus is compared to state-of-the-art antiviruses. In order to avoid unfair comparisons, the feature extraction stage is standardized by monitoring 7,690 behaviors that the suspicious JavaScript file can do when executed purposely. In order to avoid unfair comparisons, the feature extraction stage is standardized by monitoring the behaviors that web fileless attack can do when launched directly from a malicious web-server to a listening service in a personal computer. In the classification stage, our antivirus is endowed with the mELM Dilation kernel and contains 500 neurons in its hidden layer. As experiments, the authorial antivirus is compared to antiviruses based on deep as based on shallow networks.

With regard to shallow net-based, the antivirus made by LIMA, *et al.* (2021) is replicated. Whereas the antivirus made by LIMA, *et al.* (2021) employs neural networks based on data backpropagation [11]. LIMA, *et al.* (2021) investigate eleven distinct learning functions in order to optimize the accuracy of their antivirus. For each learning function, LIMA, *et al.* (2021) explore 4 hidden layer architectures [11]. Our authorial antivirus is also compared to antivirus based on deep neural network. The antiviruses made by SU, J. *et al.* (2018) and MANIATH, S. *et al.* (2017) are replicated.

Figure 4 and Figure 5 are graphical representations of the results described in Table 6. Figure 4 (a) presents the boxplots, from the training stage, in relation to authorial antivirus and state-of-the-art. The authorial antivirus obtained an average performance of 100.00% with a standard deviation of 0.00%. The antivirus made by LIMA, *et al.* (2021) obtained average accuracy of 49.70% and 100.00%, in its worst and best scenarios, respectively. These results were obtained using the learning functions "Resilient backpropagation (Rprop)" and "Fletcher-Powell conjugate gradient backpropagation" with 100 neurons in their hidden layers, respectively. The antivirus made by SU, J. *et al.* (2018) obtains a training accuracy of 99.51% on average. The average accuracy, resulting from the training, was 89.99% through the antivirus made MANIATH, S. *et al.* (2017).

Figure 4 (b) shows the boxplots for the best accuracy in the test phase. The authorial antivirus obtained an average accuracy of 99.95%. Antivirus made by SU, J. *et al.* (2016) obtains an accuracy of 99.50% on average. Antivirus made by MANIATH, S. *et al.* (2017) achieved an average performance of 89.90%. The antivirus made by LIMA, *et al.* (2021) achieved a mean accuracy of 49.67% and 99.96%, in its worst and best scenarios, respectively. Therefore, it is corroborated that neural networks based on backpropagation can suffer major variations, in their accuracies, depending on their configuration parameters. Then, the decision made by LIMA, *et al.* (2021) was salutary. This state-of-the-art antivirus explores different learning functions, gradients and architectures in order to optimize the accuracy of its neural networks based on data backpropagation.

Figure 5 (a) and Figure 5 (b) present the boxplots referring to the times spent during the training and test phases, respectively. In relation to training time, deep network-based antiviruses are slower since there is the use of deep network recurrent structure. In opposite, the authorial antivirus consumes only 23.05 seconds in order to conclude, on average,

its training. Although learning based on backpropagation, the work made by LIMA, *et al.* (2021) concludes its training in the order of seconds. Regarding the time consumed during the test phase, all techniques consumed very close times without great discrepancies.

Table 7 shows the confusion matrices of the techniques presented in Table 6 in percentage terms. The confusion matrix is important in order to verify the quality of supervised learning. In Table 7, B. and M. are abbreviations of Benign and Malware. The desired classes are arranged on the vertical label while the obtained classes are on the horizontal label. On confusion matrix, the main diagonal is occupied by cases whenever obtained class coincides with expected class, named true positives cases. Then, a good classifier has main diagonal occupied by high values and other elements have low values.

Table 7 shows main diagonals emphasized in bold. Our antivirus, in the test phase, mistakenly classified on average 0.10% of cases as benign when they were malware cases (false negative). Following the same reasoning, there was a mean classification of 0.00% of cases said to be malware when they were benign applications (false positive).

In digital forensics, a false positive would imply a benign application wrongly condemned. On other side, a false negative can result in an undetected malware. It is worth mentioning that malware can cause irreversible and irrecoverable harm to the entire world wide web. In synthesis, a false negative can result in the loss of the victim's dignity, finances and mental health. It is emphasized that authorial antivirus presents the lowest average percentage of false negatives with only 0.10%.

Still regarding Table 7, sensitivity and specificity refer to the ability of the antivirus to identify malware and benign applications, respectively. The proposed work presents the confusion matrix in percentage terms in order to facilitate the interpretation of sensitivity and specificity. In synthesis, the sensitivity and specificity are presented in the confusion matrix itself, described in Table 7. For example, the proposed antivirus averages 100.00% with respect to both sensitivity and true positives. Following the same reasoning, authorial antivirus obtains, on average, 99.90% for both specificity and true negatives.

Table 9 shows the parametric t-students and non-parametric Wilcoxon hypothesis tests between our antivirus and the state-of-the-art. It is possible to conclude that our authorial antivirus is statistically equivalent in comparison to best configuration of the antivirus made by LIMA, *et al.* (2021). The explanation is that in both the parametric t-students and the non-parametric Wilcoxon tests, the null hypothesis were accepted. Therefore, our authorial antivirus and the best scenario of antivirus made by LIMA, *et al.* (2021) are statistically equivalent.

Authorial antivirus demonstrated a major advantage when compared to the state-of-the-art. Our antivirus was able to achieve the best average accuracy with 99.95% accuracy accompanied by a training time corresponding to 23.05 seconds. The relationship between percentage accuracy and training time in reverse order is employed in Biomedical Engineering [12]. It is admitted that the establishment of this relationship assumes an important role in Information Security since 8 (eight) new malwares are released per second [9]. So, paradoxically, a newly launched antivirus may already be obsolete and require new training through a newly discovered vulnerability. In syntheses, the learning time of an antivirus should not be discrepant in comparison to the rate of new malware creation worldwide.

8. Conclusion

With the growth of the World Wide Web, the estimative is that malware propagation will continue to grow for a few years as the Internet is the primary means of cyber-infection [16]. Despite the presence, almost totalitarian, of antivirus on personal computers, malwares have caused billionaire losses on increasingly larger scales. One explanation is that cyber-attacks are systematically renewed [24].

Among modern cyber-invasions, we highlight the fileless attacks mainly through malicious web-servers. Currently, the user is infected through attractive web-sites endowed with Social Engineering, with ideological and religious content, unable to raise any kind of suspicion under their real intentions [26].

In this paper, we evaluated 86 conventional antiviruses against cyber-threats in php files since almost all malwares run on web-server are php codes [24]. The detection variation of php malwares ranged from 0% to 78.50% depending on the antivirus. Commercial antiviruses, on average, managed to detect 16.82% of the threats. On average, conventional antiviruses reported false negatives and were omitted in 49.49% to 33.68% of cases, respectively.

The current work used VirusTotal as a system to automatically submit malicious code to commercial antiviruses. VírusTotal limits file submissions to full product platforms only. This implies that it was not possible to make comparisons between the fulls and free versions of the major worldwide antiviruses. It is assumed that the results of the free versions are substantially lower compared to the full licensed versions.

On average, 57% of commercial antiviruses, in full mode, did not detect any malware php file. It is important to note that the malicious samples evaluated are public domain and catalogued by incident responders. We understood that there are failures in services provided by commercial antiviruses in relation to protection against cyber-threats on a large scale and in real-time.

In order to supply the limitations of commercial antiviruses, state-of-the-art employs the analysis of the source code of the suspicious file, named static feature approach. Then, the algorithm can be studied and, therefore, it is possible to investigate the malicious intention of the file before even of it to be executed by the user [11][25][15]. However, static analysis is impractical with fileless attack since the file is executed remotely and, therefore, its source code is not present on the personal computer. Then, instead of the unworkable static analysis, the extraction of features from our

Table 6
Comparison among the authorial antivirus and the state-of-the-art.

Technique	Train rate (%)	Test rate (%)	Train time (sec.)	Test time (sec.)
Authorial Antivirus	**100.00 ± 0.00**	99.95 ± 0.05	23.05 ± 1.80	2.24 ± 0.13
Antivirus made by LIMA, *et al.* (2020), worst conf. [11]	49.70 ± 1.56	49.67 ± 1.39	10.11 ± 16.85	0.39 ± 0.08
Antivirus made by LIMA, *et al.* (2020), best conf. [11]	**100.00 ± 0.00**	**99.96 ± 0.04**	756.45 ± 256.57	0.38 ± 0.06
Antivirus made by SU, J. *et al.* (2016) [25]	99.51 ± 0.12	99.50 ± 1.01	1683.36 ± 330.09	1.70 ± 0.55
Antivirus made by MANIATH, S. *et al.* (2017) [15]	89.99 ± 20.20	89.90 ± 20.16	1523.54 ± 307.00	0.07 ± 0.01

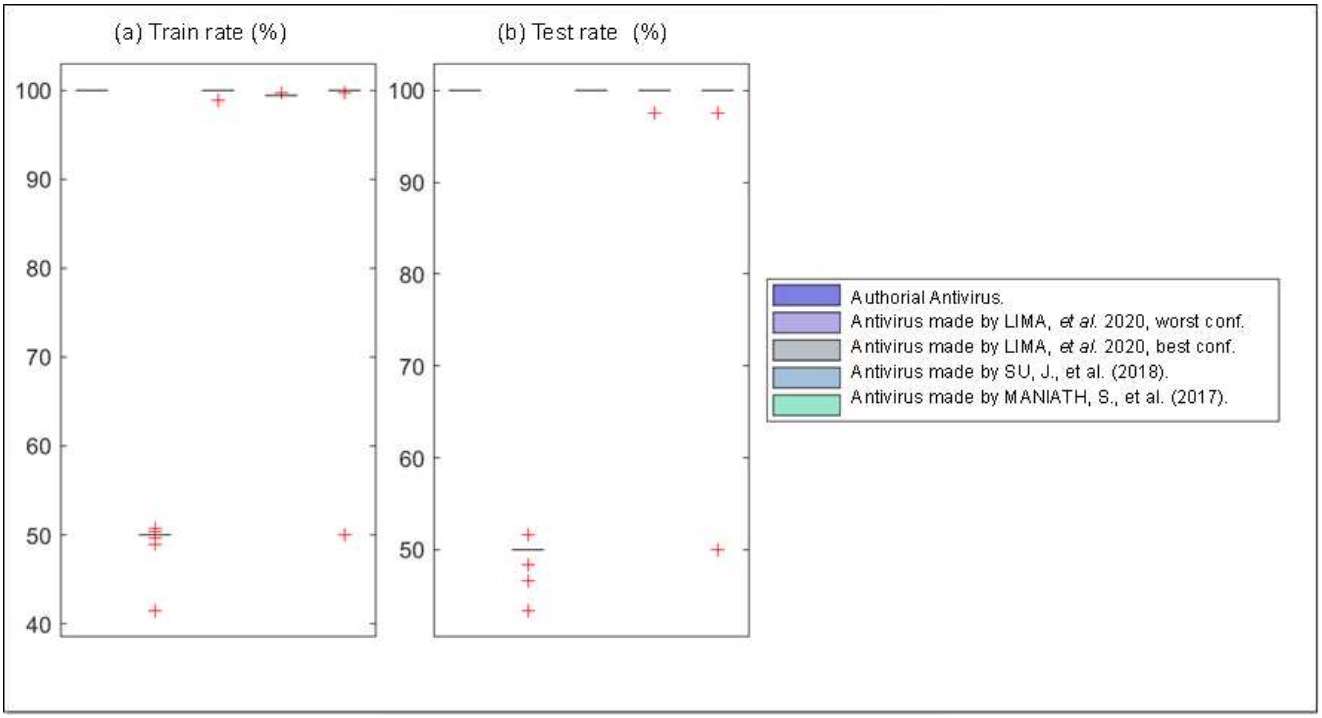

Figure 4: Boxplots referring to the accuracies of the authorial antivirus and the state-of-the-art.

NGAV concerns the audit of the anomalous behavior on the victim's computer due to fileless attack. In average, our dynamic extraction of features monitors 11,777 behaviors that the fileless attack can to do when launched directly from a malicious web-server to a listening service in a personal computer. Our NGAV solution can actually (re)construct a chain of events, visualizing what the actual attacker might be up to, as opposed to looking at individual, discreet events.

Table 7
Confusion matrix of the authorial antivirus and the state-of-the-art. (%)

Technique		Train		Test	
		M.	B.	M.	B.
Authorial Antivirus	M.	**100.00 ± 0.00**	0.00 ± 0.00	**100.00 ± 0.00**	0.00 ± 0.00
	B.	0.00 ± 0.00	**100.00 ± 0.00**	0.10 ± 0.07	**99.90 ± 0.07**
Antivirus made by LIMA, *et al.* (2020), worst configuration [11]	M.	**44.14 ± 49.74**	55.86 ± 49.74	**44.11 ± 49.77**	55.89 ± 49.77
	B.	44.74 ± 49.33	**55.26 ± 49.33**	44.78 ± 49.37	**55.22 ± 49.37**
Antivirus made by LIMA, *et al.* (2020), best configuration [11]	M.	**100.00 ± 0.00**	0.00 ± 0.00	**99.93 ± 0.07**	0.07 ± 0.07
	B.	0.00 ± 0.00	**100.00 ± 0.00**	0.00 ± 0.00	**100.00 ± 0.00**
Antivirus made by SU, J. *et al.* (2018) [25]	M.	**100.00 ± 0.00**	0.00 ± 0.00	**100.00 ± 0.00**	0.00 ± 0.00
	B.	0.97 ± 0.24	**99.03 ± 0.24**	0.95 ± 0.95	**99.05 ± 0.95**
Antivirus made by MANIATH, S. *et al.* (2016) [15]	M.	**90.00 ± 20.20**	10.00 ± 20.20	**90.00 ± 20.20**	10.00 ± 20.20
	B.	0.02 ± 0.11	**79.98 ± 40.40**	0.19 ± 0.94	**79.81 ± 40.32**

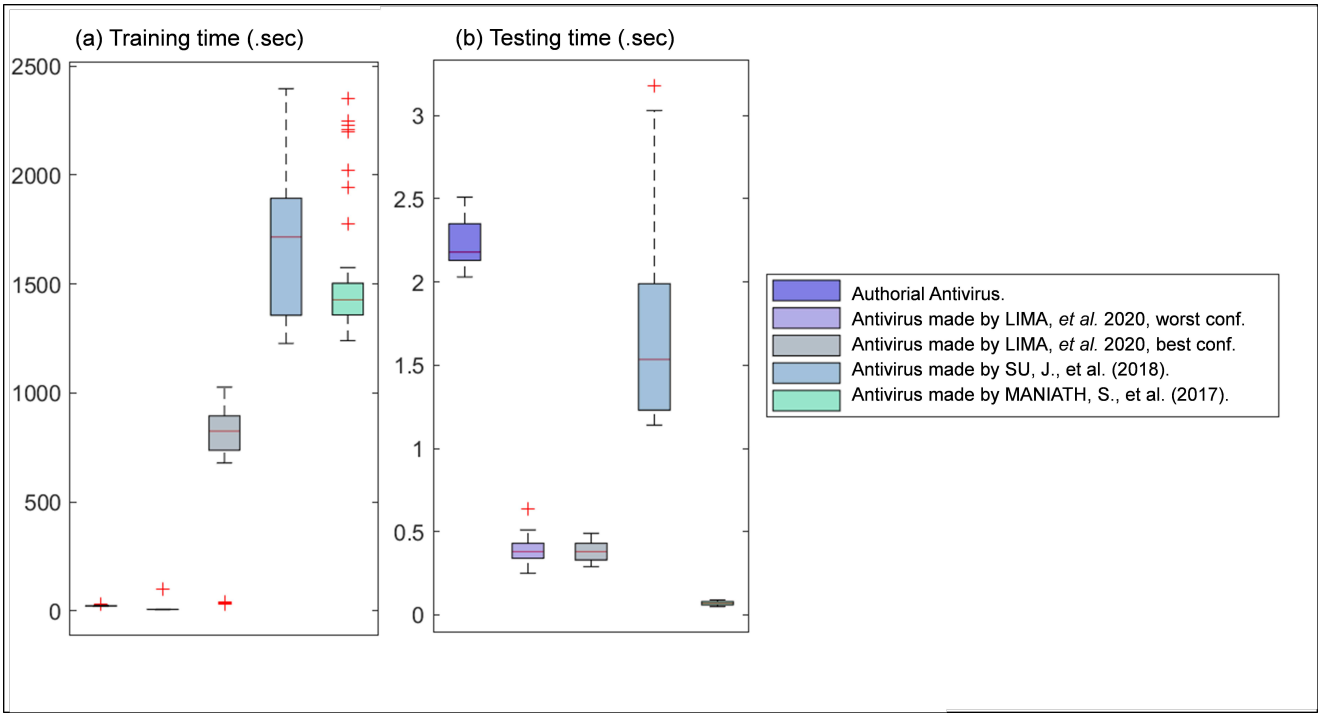

Figure 5: Boxplots regarding the processing times of the authorial antivirus and the state-of-the-art.

Table 8
T-students and Wilcoxon hypothesis test of the authorial antivirus and the state-of-the-art.

	t-students (parametric test)		Wilcoxon (non-parametric test)	
Comparison	Hypothesis	*p*-value	Hypothesis.	*p*-value
Authorial Antivirus *vs* Antivirus made by LIMA, *et al.* (2020), worst conf.	1	2.62632e-234	1	2.09518e-63
Authorial Antivirus *vs* Antivirus made by LIMA, *et al.* (2020), best conf.	0	0	0	0
Authorial Antivirus *vs* Antivirus made by SU, *et al.* (2018)	1	8.57969e-09	1	8.28322e-09
Authorial Antivirus *vs* Antivirus made by MANIATH, *et al.* (2017)	1	5.89733e-09	1	1.80676e-10

In this paper, ELM learning machines are applied in digital forensics specifically in the recognition of patterns of malwares php. Then, malicious behaviors derived from of suspicious php executed in web server, is employed as input attributes of statistical learning machines used as classifiers in the distinction between benign and malware phps. Regarding the precision, our authorial antivirus reached an average accuracy of 99.95%.

Our NGAV can be extended to provide cyber-protection to local networks. Then, the future goal is that our NGAV can be executed in personal computers and in proxy server which is the intermediary between the World Wide Web and the local network. It will be promise of NGAV, executed in the proxy, monitoring the network traffic trace in PCAP format. In this way, it will be minimized the workload of our NGAV executed also in the personal computers. For this, it is necessary to create a new Web-Server Next Generation Sandbox endowed with an architecture composed of a web-server, a proxy server and multiple personal computers.

The future goal of our NVAG is to supply the limitations of the proxies' defense mechanisms which are based on blacklists as well as commercial antiviruses. Therefore, it will not be more necessary to wait for the local network to be infected and, in sequence, also wait for denounces of anomalous behavior and then take providences in order to detect a new malicious web-server.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

Research Involving Human Participants and/or Animals The authors declare that no human participants were involved in this research.

Informed Consent This research did not include healthcare intervention of human participants.

9. Authorship contributions

Washington Silva and Wellington Santos conceived the presented idea of morphological extreme learning machine. Wellington Santos developed the theoretical formalism and Washington Silva performed the implementation. Ricardo Pinheiro, Danilo Souza, and Sthéfano Silva carried out the experiment. Petrônio Lopes, Rafael Lima, Jemerson Oliveira, Thyago Monteiro constructed the dataset and validated the samples,

Sidney Lima wrote the manuscript with support from Sérgio Fernandes and Edison Albuquerque. All authors discussed the results and contributed to the final manuscript and the interpretation of the results. All authors provided critical feedback and helped shape the research, analysis and manuscript.

References

[1] AZEVEDO, W.W. *et al.*., 2015a. Fuzzy morphological extreme learning machines to detect and classify masses in mammograms. In: 2015 IEEE International Conference on Fuzzy Systems (FUZZIEEE), Istanbul. doi:https://doi.org/10.1109/FUZZ-IEEE.2015.7337975.

[2] AZEVEDO, W.W. *et al.*., 2015b. Morphological extreme learning machines applied to detect and classify masses in mammograms. In: 2015 International Joint Conference on Neural Networks (IJCNN), Killarney. doi:https://doi.org/10.1109/IJCNN.2015.7280774.

[3] AZEVEDO, W.W. *et al.*., 2020. Morphological extreme learning machines applied to the detection and classification of mammary lesions. In: Tapan K Gandhi; Siddhartha Bhattacharyya; Sourav De; Debanjan Konar; Sandip Dey. (Org.). Advanced Machine Vision Paradigms for Medical Image Analysis. 1ed.Londres: Elsevier Science. , 1–30doi:https://doi.org/10.1016/B978-0-12-819295-5.00003-2.

[4] BA, L., CAURANA, R., 2014. Do deep nets really need to be deep? Advances in Neural Information Processing Systems , 2654–2662.

[5] CISCO, 2018. CISCO 2018 Annual Cybersecurity Report. Accessed on Dec. 2020. URL: https://www.cisco.com/c/dam/m/hu_hu/campaigns/security-hub/pdf/acr-2018.pdf.

[6] CONRAD, E., MISENAR, S., FELDMAN, J., 2017. Eleventh Hour CISSP (Certified Information Systems Security Professional). Syngress Publishing.

[7] HUANG, G.B. *et al.*., 2012. Extreme learning machine for regression and multiclass classification. IEEE Transactions on Systems, Man, and Cybernetics 42(2), 513–519. doi:https://doi.org/10.1109/TSMCB.2011.2168604.

[8] IBM, 2014. Explore the latest security trends—from malware delivery to mobile device risks—based on 2013 year-end data and ongoing research. Accessed on Dec. 2020.

[9] Intel, 2018. McAfee Labs. Accessed on Feb 2020. URL: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-mar-2018.pdf.

[10] LIMA, S., 2020. Limitation of COTS antiviruses: issues, controversies, and problems of COTS antiviruses. In: Cruz-Cunha, M.M., Mateus-Coelho, N.R. (eds.) Handbook of Research on Cyber Crime and Information Privacy, vol. 1, 1st edn. IGI Global, Hershey. doi:http://dx.doi.org/10.4018/978-1-7998-5728-0.ch020.

[11] LIMA, S., SILVA, H., LUZ, J. *et al.*., 2020a. Artificial intelligence-based antivirus in order to detect malware preventively. Progress in Artificial Intelligence doi:https://doi.org/10.1007/s13748-020-00220-4.

[12] LIMA, S., SILVA-FILHO, A.G., SANTOS, W.P., 2016. Detection and classification of masses in mammographic images in a multi-kernel approach. Computer Methods and Programs in Biomedicine 134, 11–29. doi:https://doi.org/10.1016/j.cmpb.2016.04.029.

[13] LIMA, S.M.L., SILVA-FILHO, SANTOS, W.P., 2020b. Morphological Decomposition to Detect and Classify Lesions in Mammograms.In: Wellington Pinheiro dos Santos; Maíra Araújo de Santana; Washington Wagner Azevedo da Silva. (Org.). Understanding a Cancer Diagnosis. https://novapublishers.com/shop/understanding-a-cancer-diagnosis/.

[14] LIMA, S.M.L., SILVA-FILHO, A.G., DOS SANTOS, W.P., 2014. A methodology for classification of lesions in mammographies using zernike moments, elm and svm neural networks in a multi-kernel approach. In: 2014 IEEE International Conference on Systems, Man and Cybernetics SMC, San Diego doi:https://doi.org/10.1109/SMC.2014.6974041.

[15] MANIATH, S., ASHOK, A., 2017. Deep learning lstm based ransomware detection. Recent Developments in Control, Automation Power Engineering doi:https://doi.org/10.1109/RDCAPE.2017.8358312.

[16] Microsoft, 2013. Trustworthy Computing. Microsoft Computing Safety Index (MCSI) Worldwide Results Summary. Technical report.

[17] PAEMAL, 2020. PAEMAL(PHP Analysis Environment Applied to Malware Machine Learning). Accessed on Dec. 2020. URL: https://github.com/rewema/PAEMAL.

[18] PALOALTO, 2013. PALOALTO 2013 The network security company. The Modern Malware Review. Analysis of New and Evasive Malware in Live Enterprise Networks. volume 1st Edition.

[19] PEREIRA, J.M.S. *et al.*., 2020. Method for Classification of Breast Lesions in Thermographic Images Using ELM Classifiers. In: Wellington Pinheiro dos Santos; Maíra Araújo de Santana; Washington Wagner Azevedo da Silva. (Org.). Understanding a Cancer Diagnosis. https://novapublishers.com/shop/understanding-a-cancer-diagnosis/.

[20] SANS, 2017. SANS Institute InfoSec Reading Room. Out with The Old, In with The New: Replacing Traditional Antivirus. Accessed on Feb 2020. URL: https://www.sans.org/reading-room/whitepapers/analyst/old-new-replacing-traditional-antivirus-37377.

[21] SANTOS, W.P., 2011. Mathematical Morphology In Digital Document Analysis and Processing. volume 8. New York: Nova Science.

[22] Skybox, 2018. Skybox Security Vulnerability and Threat Trends Report 2018. Analysis of current vulnerabilities, exploits and threats in play. Accessed on Dec. 2020. URL: https://lp.skyboxsecurity.com/rs/skyboxsecurity/images/Skybox_Report_Vulnerability_Threat_Trends_18.pdf.

[23] Skycure, 2016. Skycure Mobile Threat Defense. Mobile Threat Intelligence Report Q1 2016.. Accessed on Dec. 2020. URL: https://www.symantec.com/content/dam/symantec/docs/reports/skycure-mobile-threat-intelligence-report-q1-2016-en.pdf.

[24] Sophos, 2014. Sophos Security made simple. Security Threat Report 2014. Smarter, Shadier, Stealthier Malware. Accessed on Dec. 2020. URL: https://www.sophos.com/en-us/medialibrary/pdfs/other/sophos-security-threat-report-2014.pdf.

[25] SU, J., VASCONCELLOS, D., t., 2018. Lightweight classification of iot malware based on image recognition. 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC) doi:https://doi.org/10.1109/COMPSAC.2018.10315.

[26] Symantec, 2012. Symantec Reports. Internet Security Threat Report: 2001 Trends.. volume 17. Published April 2012. Symantec Corporation.

[27] Symantec, 2017. Symantec Reports. Internet Security Threat Report: Living off the land and fileless attack techniques. An ISTR Special Report.

[28] WANG, Y.; QIU, Y.T.T.M.K.L.H.Z.B., 2017. A two-step convolutional neural network based computer-aided detection scheme for automatically segmenting adipose tissue volume depicting on ct images. Computer Methods and Programs Biomedicine 144, 97–104. doi:10.1016/j.cmpb.2017.03.017.

[29] XIANG, C., DING, S.Q., LEE, T.H., 2005. Geometrical interpretation and architecture selection of mlp. The IEEE Transactions on Neural Networks and Learning Systems 16, 84–96. doi:https://doi.org/10.1109/TNN.2004.836197.