International Journal of Computer Sciences and Engineering
Vol.**13**, Issue.**3**, pp.**24-32**, March **2025**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Research Article

The Dual Edge of Backdoors: Accuracy Analysis and Preventive Strategies for Secure Systems

Arushi Gupta[1*], Safdar Tanweer[2], Syed Sibtain Khalid[3], Naseem Rao[4]

[1,2,3,4]Dept. of CSE, Jamia Hamdard University, New Delhi, India

**Corresponding Author*: ✉

Received: 24/Jan/2025; **Accepted**: 26/Feb/2025; **Published**: 31/Mar/2025. **DOI**: https://doi.org/10.26438/ijcse/v13i3.2432

Abstract: While digital transformation's benefits are reciprocal, we have vulnerabilities with rapid technological developments, one of which is malware, one of the biggest dangers to digital security. It's harmful software that can mess up, damage, or sneak into computer systems without permission. In this article, we are going to use Kali Linux backdoor attacks, as we know that backdoor vulnerabilities have emerged as a critical threat to cybersecurity, with recent reports indicating a 45% increase in backdoor-related incidents over the past year. Hence, with the availability of free online tools like VirusTotal and Hybrid analysis, detection remains challenging, but it can detect up to an average detection rate of only 72% for sophisticated backdoors. As such, backdoors are covert methods for attackers to access systems that bypass typical security barriers and represent a major weakness to the integrity, confidentiality, and availability of information systems. This paper defines the implementation of a backdoor and analyzes existing mitigation techniques. It also introduces a holistic approach that combines anomaly detection and code analysis on how we implemented this backdoor using two operating systems. It covers methodologies for monitoring insider activities, detecting anomalous behavior (with the help of free tools) that may indicate the presence of backdoors, and protective actions to reduce the threat posed by trusted users. In this paper, we focus on insiders and their backdoor exploitation capabilities, discussing real-world scenarios in which insiders exploited backdoors for data exfiltration, sabotage, or espionage.

Keywords: Backdoor, Malware, Hackers, Implementation, Cyber Attacks

1. Introduction

The digital revolution has changed nearly every facet of life, business, and society that exists in the modern world. Nonetheless, this evolution has posed its share of serious risks, with malware spreading amongst the most common and destructive dangers. Malware definition: Any software deliberately used to damage computers, networks, or users. The more dependence on technology, the greater the need to understand malware and ensure cybersecurity.

Here we are implementing a backdoor related to some kind of malware, and after that, we are going to check its accuracy on different open tools, as we all know, cyber-attacks pose an ever-increasing danger to anyone who is using the internet or social networking sites. Understanding different types of cyberattacks is crucial for building a strong defense against them. While an alarming incident response can help reduce the impact of an attack, being aware of common threats (like malware, phishing, DDoS, etc.) is the first step in staying protected. When a cyber incident happens, having an incident response plan can help ensure that damage is minimized and that normal operations resume as soon as possible. The incident response involves a series of steps, including- Preparation, Pinpointing, Detection and Assessment, Containment, Elimination and Restoration, and the setup of a meeting to discuss future protection after learning things from the incident [1]. A backdoor is a means of bypassing normal authentication in the implementation of a computer system or an embedded intrusion that has been deliberately left in software to allow for continued access. This includes a whole range of methods, from software vulnerabilities to application backdoors and even malicious bugs introduced by developers or cons [2]. Backdoor attacks are a specific type of cyberattack that targets deep neural networks (though we won't dive into those details here). In these types of attacks, the attacker adds tampered data into the starting process of a machine learning model. Later, during testing or real-world use, the attack is activated by showing the model a specific trigger, like a small patch or pattern, often designed to be subtle. Unlike other types of attacks, backdoor attacks focus on secretly embedding this trigger into the training data,

tricking the model into behaving incorrectly when the trigger is present [3]. Backdoor Bench is an extensible and modular framework comprising attack, defense, evaluation, and analysis modules to facilitate comprehensive research on backdoor attacks. From the already published threat model, backdoor attacks can generally be categorized into two main types: *data poisoning* and *training-controlled* attacks. **Data poisoning attacks** restrict the attacker from modifying only the training data. **Training-controlled attacks** allow the attacker to influence not only the training data but also the training process itself, it has some disadvantages, like increased complexity, limited data sets available, and scalability issues [4]. So, this research provides a comprehensive perspective on how to write the backdoor's code and gain access to any other operating system. Hence, the purpose of this study is basically- In this research paper, we are trying to gain access to Windows by implementing the backdoor program in Kali Linux.

i- It will provide a clear platform between the two operating systems. It will help in future work or organizations to gain access to any unauthorized entity

ii- It will also ensure that the program you are implementing how as accurate as possible.

As we all know, in recent years, we have faced some kind of malware attack, whether in a big firm or a simple software system. Among all this malware, a backdoor system is implemented to hack your system and get all the data and information from the victim's PC to the hacker's PC. There are different types of backdoors, and there are some common techniques associated with them, such as hardware Backdoors, Software Backdoors, Network backdoors, AI-based or Advanced Backdoors, Cryptographic Backdoors, and Physical Backdoors. This will enhance a victim's knowledge so that they will get alerted to any kind of malware attacks, or they can also gain a brief knowledge of implementing this kind of backdoor in investing or helping some government or national firms. This paper is divided into different parts as mentioned here: I- Introduction, II- Related Work, III- Methodology, IV- Results, V- Discussion, VI- Conclusion, VII- Acknowledgment, and VIII- References.

2. Related Work

Table 1 shows the papers we've selected for our literature review from different websites-

Database	Search Query	Initial Results	Filters Applied	Filtered Results	Papers Selected
SpringerLink	"Malware attacks through Backdoor"	1,564 papers	Content type: "Research article" Date: "Last 6 months"	46 papers	1 paper
ScienceDirect	"Malware attacks through Backdoor"	1,743 papers	Year:"2024" Article type: "Research Papers" Access type: "Open access and open archives"	84 papers	1 paper
Google Scholar	"Malware attacks through Backdoor"	28,100 papers	Anytime since "2024" Article type: "Review articles"	338 papers	5 papers
ACM Digital Library	"Malware attacks through Backdoor"	1,114 papers	Content type: "Research article" Publication date: "Past 3 years"	120 papers	3 papers

Total Papers Selected: 10

Backdoor attacks in machine learning exploit the training process to implant malicious patterns that cause specific misclassifications when activated. Backdoor attacks can be tested on these bases- Attack Success Rate: The proportion of backdoored inputs misclassified as the target class. Model Accuracy: The overall classification accuracy on clean data, which ideally remains unaffected to maintain the attack's stealth.Experimental studies demonstrate that backdoor attacks can achieve near-perfect success rates without degrading accuracy when applied strategically. Backdoor attacks in P2PFL are an emerging threat, with attackers leveraging network properties to maximize the stealth and impact of their malicious updates[5]. The different types of backdoor sets stimulate which method is suitable for the detection of backdoor-leverage diverse data types (network and host logs), adopt graph-based or multi-modal analytics, and provide standardized evaluation benchmarks. The authors propose a reference architecture for future APT detection systems, emphasizing comprehensive coverage, continuous monitoring, adaptability, and advanced analytics[6]. Some AI tools to detect malware in the system have been increasingly utilized to counteract the evolving complexity and sophistication of malware. Traditional signature-driven detection methods have some flaws in uncovering fresh and evolving threats. The models that have been used here are shallow **Learning Models and deep Learning Models**.[7]. As malware is detected using various ML tools such as SVM, adopted for sorting tasks with linear/non-linear kernels, Random Forest (RF), and Decision Trees (DT), Effective for feature importance and high accuracy, K-Nearest Neighbors (KNN): Simplistic yet robust for certain datasets.Unsupervised Learning Models: K-Means Clustering: Groups similar data points for anomaly detection. Hierarchical Clustering: Builds nested clusters for detailed analysis, and hence CNN and Long Short-Term Memory (LSTM): Extremely precise in extracting complex patterns from data.[8]. Bypass attacks are a form of targeted conflict-driven attacks in FL, where an opponent injects malicious

behavior into the global model during training. These attacks are particularly critical because they are designed to remain undetected while achieving specific malicious goals without degrading the model's overall performance[9]. Backdoor attacks are a subset of adversarial strategies designed to compromise machine learning (ML) and federated learning (FL) systems by subtly embedding triggers that cause the model to perform incorrectly under specific conditions. These attacks are uniquely concerning because of their stealth and the possibility of misapplying in sensitive applications. Some backdoor techniques have been introduced so far. Sybil Attacks: Multiple fake nodes under the attacker's control submit poisoned updates to increase the success rate of backdoor implantation. Activation Clustering: Exploiting activation patterns to cluster malicious updates and bypass server-side defenses.[10].So, on learning about backdoors, we concluded that they affect the vulnerability of ML models.The factors that affect the success of backdoor poisoning are (i) the part of backdoor samples inserted into the training data, (ii) the size of the backdoor trigger, and (iii) the intricacy of the target model, controlled via its model parameters [11]. Backdoor attacks pose a significant security risk, exploiting dormant triggers that activate only under specific conditions to cause targeted misclassifications. Research into backdoor attacks has predominantly centered on domains like computer vision and natural language processing[12].There are some different kinds of backdoors, one of them named *TridentShell* (focusing on its ability to evade detection mechanisms and leave no trace of intrusion on web servers). Novel Backdoor Design:- *TridentShell* leverages Java bytecode instrumentation to hijack and modify server processes at runtime. It employs a "fileless" approach, eliminating the need to store malicious scripts in the server's directory, thus bypassing traditional file-based detection mechanisms. *TridentShell* represents a significant advancement in covert backdoor attacks, combining runtime adaptability, anti-traceability, and decentralized communication. [13].The paper explores the growing threat of cyberattacks in today's digital era, emphasizing the importance of robust incident response strategies. It focuses on common cyberattack types—malware, phishing, ransomware, DDoS attacks, and social engineering—and details the steps necessary to mitigate their impacts effectively. Comprehensive Categorization of Cyberattacks- Malware, Phishing, Ransomware, DDoS Attacks, Social Engineering. There are some incident response steps to detect the malware in the system: Incident Response Framework:- Preparation, identification, detection, containment, eradication and recovery, and Post-incident activities. It serves as a foundational guide for understanding and responding to cyber threats. Its focus on structured incident response and its detailed breakdown of attack types are valuable for both academic and practical audiences. [14].

Table 2 (Literature Review)

Reference no.	Year of publication	Focus area	Key Contributions of the Paper	Overall result
[5]	2024	How backdoor attacks are affected by the attackers in peer-to-peer learning.	It designed and assessed backdoor attacks in P2PFL, focusing on scenarios where an adversary manages a limited subset of nodes.	Implementing the backdoor and recognizing its effect on P2P learning.
[6]	2024	How APT attacks influence the computers.	It offers valuable awareness to the research fraternity by presenting a proper analysis of present systems, frameworks, and methods allied with APT attack detection.	It provides the implementation of a backdoor and its analysis on the computer systems.
[7]	2024	How the malware affects the industries.	It introduces mainly the hierarchy of the intrusion detection criteria, different detection mechanisms, and their relationship.	It analyzes malware detection by using different online tools.
[8]	2024	Analysis of different methods to investigate the malware.	It shows that signature-based approaches can't find APTs (advanced persistent threats), while dynamic approaches take longer to use and need more testing tools, such as sandboxes.	It uses different online techniques to detect the malware in the system.
[9]	2024	Presents a complete overview of the security and privacy challenges associated with FL.	This research suggests that additional research is required to protect FL against malware attacks.	It reconsiders the data security associated with FL and also describes its methods for how this problem of privacy can be handled safely.
[10]	2024	It reviewed two papers based on the	It discussed the perks, concerns,	It defines that cyber security

		security ethics in cybersecurity.	and obstacles of affiliating Cyber defense and the developed execution of network protection strategies.	networks are based on user access control and are linked with modern tools.
[11]	2024	It mainly focuses on analyzing the backdoor curves or samples based on how poisonous it is.	The potency of backdoor attacks depends on the intricacy of the target, a fraction of the model in the training data, size, and visibility of the backdoor.	It concluded that the backdoor is only effective until and unless it is exposed or their data is easy to capture.
[12]	2023	The implementation of malware, especially a backdoor	It discussed the backdoor implementation and how the DL tackled it to detect in our computer vision.	Having a deep understanding of the backdoor, one can implement it easily with 0.1% of poison.
[13]	2023	It discovers "TRIDENTShell," an unconventional web backdoor intrusion that can deploy a stealthy backdoor into a victim's operating system without leaving any traces.	It activates a backdoor loaded into server memory without leaving any traces and can elude most of the static recognition methods.	It comprises Java application servers, access governing policy, and static detections to some extent.
[14]	2022	Understanding security measures for malware attacks.	The framework, DeepGuard, is proposed to ensure both privacy and preservation of the backdoor.	It focuses on neural network data collection for the detection of malware and picking out some preventive measures from it.

As to implementing the backdoor on PCs, we have designed a backdoor using Kali Linux and Windows 10, and to test its accuracy, we have used three online platforms where it was detected, which are *HYBRID ANALYSIS, VIRUS TOTAL, and JOTTI.*

Hence, there are always some pros and cons of each and everything in this world, so do we have advantages and disadvantages of using these free platforms?
Pros:- Cost-effective (it removes the financial burden on anyone who is using it), Ease of Use (anyone can use it from anywhere and at any time), Baseline for comparison (it provides a baseline to evaluate the capability of personalized detection mechanisms), Community Contributions (researchers can share insights and also get feedback), Quick Preliminary Analysis (it'll provide results within a minute as these platforms don't consume so much time), Global Reach (from all over the world students and researchers can reach these platforms), Anonymity (anyone can share their opinions here without revealing their identity).

Cons:- Limited Detection Capabilities (these tools have limited features to explore, and comprehensive features are only available in premium versions), Lack of Detailed Insights (they also provide some limited details of your research), Scalability Challenges (It may have limitations on the number of files to be detected or uploaded for search), No Guarantee of compliance (they do not adhere to some industry standards making finding less credible), Limited support (it offers small or no control for customers making it challenging to resolve issues).

3. Methodology

So, first of all, we have to implement a listener (it is a process that waits for an incoming connection from the victim's operating system to establish a network) on the victim's side where a socket library (it is used to establish a good internet connection for communication) is used (on both sides) to create a proper internet connection.

On the hacker's side, all will be implemented on Kali Linux, which uses the IP address and port number of the victim's system to create a proper connection and access to the victim's operating system. Here, use serialization (it converts the data object into a configuration for data storage/transmission and instantiates the object when necessary). It actively listens to the connection that is being made with the victim's operating system (Windows).
So here is the implementation

III.I AT HACKER'S SIDE

```
#!usr/bin/env python
import socket
class Listener():
def _init_(self,ip, port)
listener.bind((ip, port))
self.connection, address = listener.accept()
print("[+] Got a connection from " + str(address)
my_listener = Listener("IP_address", port number(of hacker's system))
```

Figure 1 shows that we've saved our listener file on the hacker's side (Kali Linux) so that it can establish a connection from our victim's side.

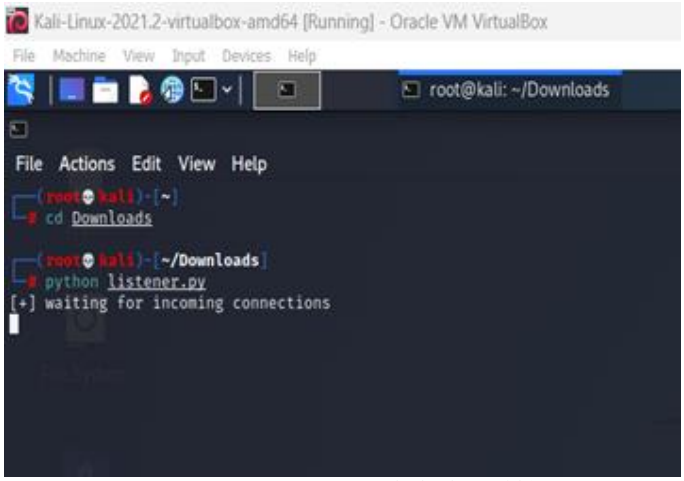

Figure 1 (running listener on the hacker's side)

On the victim's side, we implement the socket library, first of all, to connect to the internet and then use refactoring (it helps to alter the program suite to better its core structure while leaving the code's external structure/behavior unchanged).
So here is the implementation-

III.II AT VICTIM'S SIDE-

```
#!usr/bin/env python
import socket
import subprocess
class Backdoor:
def _init_(self,ip,port):
my_backdoor =
Backdoor("IP_address",Port_number(victim's side))
my_backdoor.run()
```

Figure 2 depicts that we've saved the reverse backdoor file on our victim's side.

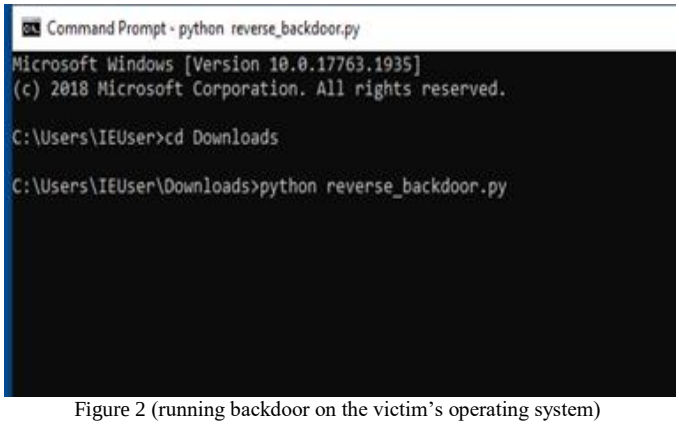

Figure 2 (running backdoor on the victim's operating system)

So, till now we have established a connection between both operating systems so that the authorized hacker could listen or control the victim's system for beneficial purposes.

Now, we are going to check the accuracy of our program on different online available tools so that we can get an assurance that our program is not harmful for any operating system or any industry it will not impact the system's internal sources.

III.III Checking the program accuracy on different platforms-

Here, it is the first tool we are using, i.e., Hybrid analysis analyzes the memory and signatures of malware and then combines them with the parameters of the pattern. It only analyses the malware that is stored on the disk or that runs in the memory, as it is a free tool for malware analysis that combines sandboxing technology with ML and provides detailed analysis. This malware (backdoor) analysis case shows zero threat detection as there is no specific threat. The result has shown that the program is clean which means it is not harmful for further use. Figure 3 shows the result on the hybrid website where we've uploaded our backdoor's code to check the threat and scan the vulnerability.

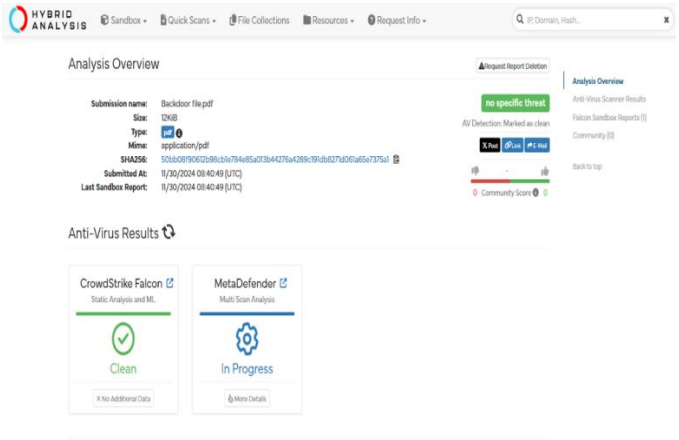

Figure 3 (testing the backdoor program on the hybrid analysis (online tool))

The second tool we are using is, which is a free tool to analyze suspicious files and sites and detect viruses and different types of malware, as it shares the result automatically with the community or the owner for added security. It only analyses the information that is being shared in the form of a file, URL, or document. Etc. Hence, in this backdoor, zero detection, i.e., 0/64 detection, has been analyzed (here, 64 is the different anti-virus platforms where the malware detection has taken place). The result has shown that no security platform has detained this program as it is not malicious. Figure 4 shows the result on the VirusTotal platform where we've uploaded our backdoor's code to scan the threat from it.

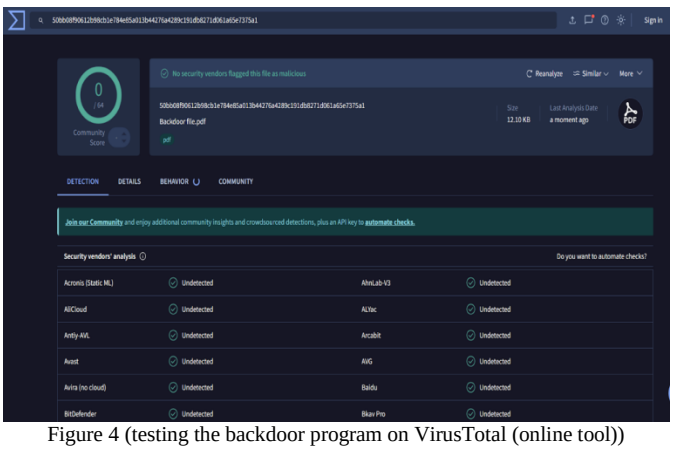

Figure 4 (testing the backdoor program on VirusTotal (online tool))

Last but not least, Jotti is a free tool to detect malicious or suspicious files with various anti-virus programs. One can send five files at a time to check for suspicious files. Here, not a single harmful file has been detected on the different types of 14 anti-virus platforms. There is a limit to uploading a file that is 250 MB, and I have sent only 12.1 KB files. Here on the scanning of the program, it has found nothing malicious for the system to use in his/her further activities. Figure 5 shows the results on the Jotti website where we've uploaded our backdoor file to check for the virus or to scan the file.

Figure 5 (Testing the backdoor program on Jotti (online tool))

4. Results

Backdoor attacks are an emergent and alarming threat to cybersecurity worldwide, posing significant risks to software and operating systems. To address this growing concern, it is crucial to evaluate the accuracy and effectiveness of implemented backdoors. Fortunately, this can be done conveniently using free online platforms. By simply uploading a file or entering a URL on these specialized websites, users can leverage the power of multiple antivirus platforms simultaneously to analyze and detect potential backdoor threats. These platforms provide a comprehensive report, presenting detailed results across a wide range of antivirus engines, making it easier to assess vulnerabilities and implement robust security measures. This streamlined approach empowers individuals and organizations to stay vigilant, mitigate risks, and contribute to a safer digital environment.

IV. I OUTPUT OF THE ABOVE-MENTIONED CODES

AT HACKER'S SIDE-

Here, Kali Linux gained access to the victim's system and got all the file names which is stored in the victim's operating system. Figure 6 depicts that Kali Linux, which is our hacker's side, got a connection from our victim's side, and it shows all the contents of our victim's side operating system (Windows).

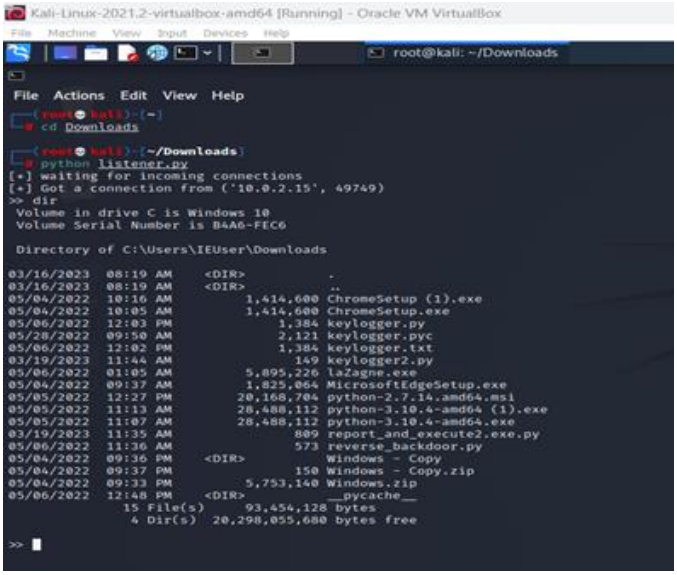

Figure 6 (Gained access to the victim's operating system)

IV.II AT VICTIM'S SIDE-

Here, it has been shown that from the victim's side, we've implemented a reverse backdoor, which means it has permitted the hacker to he could gain access to the system. Figure 7 shows the windows on our victim's side where we're running our reverse backdoor code to give hackers a listener command.

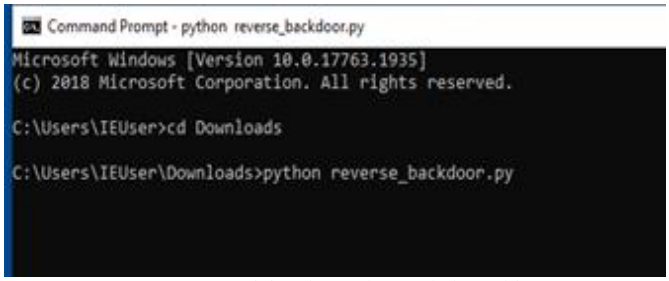

Figure 7 (Backdoor is running at victim's side)

Then, after implementing all these codes on the victim and hacker side, we checked the accuracy of how harmful our backdoor is for any operating system on three free online tools that are Hybrid analysis, VirusTotal, and Jotti. On all these platforms, it detects our backdoor on many anti-virus platforms where it shows zero detection of any harmful or malicious software that is being implemented on the operating systems. It shows that the program we have implemented has no harmful effects on any software, and it is safe for one to use and gain access to the data information or content that is being saved in the victim's software.

5. Discussion

Previous experiments and implementations of backdoor attacks have revealed a significant reliance on textual backdoor defense methods, particularly those focused on test sample examination. These methods have proven instrumental in detecting backdoor triggers embedded within textual inputs, showcasing their practical utility in safeguarding systems against such threats [15]. However, this approach has

several notable disadvantages. Firstly, it often struggles with generalizability; these methods are tailored to detect specific backdoor triggers, making them less effective against novel or adaptive attacks. Secondly, if you are using paid tools, they can be computationally expensive, especially when analyzing large-scale datasets and evaluating backdoor attacks in ML, known as backdoorbench, which uses backdoor benches in several domains and learning paradigms like NLP, speech, etc. Facial recognition systems have become an integral part of modern security frameworks, used in applications ranging from surveillance to authentication. However, these systems, like any machine learning-based technology, are vulnerable to backdoor attacks. In a backdoor attack, an adversary manipulates the training data or model in such a way that the system behaves normally under most conditions but misbehaves when triggered by a specific, often subtle, input this system also has some disadvantages like security, difficulty in detection, ethical concerns, false sense of security [16]. One can introduce and formalize the concept of backdoor variables in CSP/SAT instances, a powerful technique that can drastically reduce the computational effort required to solve complex problems, but the disadvantages of this are limited applicability, dependency on instance structure, and the potential of overfitting [17]. A backdoor can be implemented in the RSA key generation, also that can also be designed so that the keys look completely normal to everyone, but a hidden trick allows a specific person (who knows the secret) to break the security and find the private key easily[18]. Hence, evil actors typically introduce a backdoor into the victim's model by hacking the training dataset, which is known as 'data poisoning' [19]. So, to check all these programs and grammatical errors, we can also apply Grammarly here to check the APIs that will further check the backdoor examples generated earlier. Backdoors in propositional satisfiability (SAT) are like shortcuts that help solve complex logical problems more efficiently. They identify small, easy-to-handle parts of a problem that, once solved, make the entire problem much simpler. These backdoors have proven to be useful in determining whether a given logical statement can be satisfied (i.e., if there is a way to make it true)[20]. These backdoor attacks can also misclassify the models and are introduced at the model training phase in a big firm or organization[21]. One can also use encryption keys to secure their operating system [22]. One can also introduce a banning mechanism that will exclude compromised devices and reduce the influence of backdoors (applicable only if the hacking proportion is small)[20]. In this AI generation, one can inject backdoors into the operating system through data poisoning, also [21]. Hence, from all of the discussions we've defined checking and analyzing our backdoor implementation from different types of online free tools, which gives 0% threat to any of the operating systems. We can implement this backdoor in any of the systems without posing any system threat. When a backdoor is adopted and used by a person or organization, it generally works as expected for normal inputs. However, if an attacker sends specific hidden trigger words or phrases, then the model can behave maliciously, such as leaking data, generating biased responses, or executing unintended actions. These triggers act like secret commands that only the attacker knows, making the model a potential security risk[22]. In Figure 8 the graph depicts that there is a 0% threat in the backdoor.

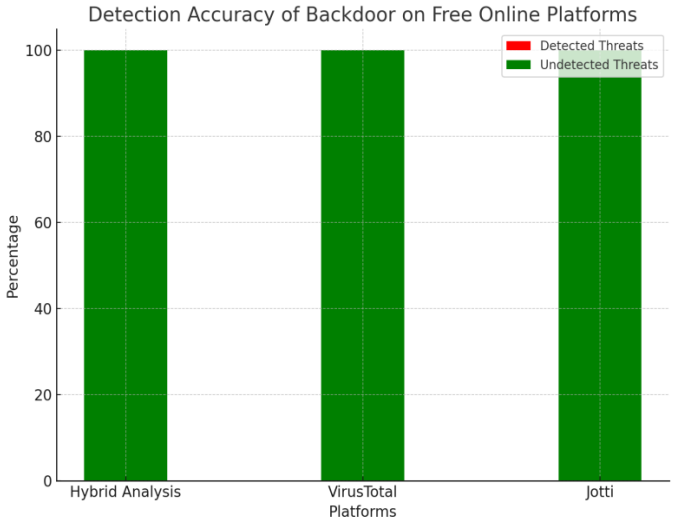

Figure 8 (It shows through graphs how malicious our backdoor is on various online tools)

6. Conclusion

In this paper, we've harnessed powerful, cost-free tools that can seamlessly detect backdoor threats across any dataset—whether the backdoor is already embedded, newly created, or in use. These tools provide rapid results within minutes, offering an efficient way to evaluate the potential danger of your backdoor program. By cross-referencing your program against multiple antivirus platforms, the tools assess its accuracy and determine its level of risk to various operating systems, ensuring you have a comprehensive understanding of its threat potential. What sets these tools apart is their scalability—there are no upload limits, allowing you to test an unlimited number of files. Furthermore, they are dataset-agnostic, meaning they don't rely on specific datasets, ensuring flexibility across various use cases. Best of all, these tools come with no ethical concerns, providing a transparent and secure method for testing backdoors without compromising integrity. So, the future scope of this research is:-

AI-driven detection system, Enhanced encryption techniques, compliance strategies, implementation of the programs, and automated threat intelligence sharing.

Data Availability

As in this paper, no data was required hence, with the help of some papers and online platforms, we've created the codes that are necessary for this paper writing. So, no new or previous data is being used here.

Conflict Of Interest Statement

No, conflict of interest statements arise. I confirm that I have provided this declaration in good faith and to the best of my knowledge. If any relevant competing interests arise in the future, I commit to updating this declaration accordingly.

Funding Information
As this paper does not include any data, it's a simple implementation of code that I've implemented without taking any funds from any individual or organization.

Author's Contributions Statement
As the author's contribution-

1. Arushi Gupta contributed to concept building and provided the methods to make this research paper also he did she wrote the validation part, data correction, and original draft. She was the project admin too.
2. Safdar Tanweer contributed to concept building and suggested some methods for conducting the research. He also performed the software validation and, in the end, performed the review and editing of the paper. He also contributed as a project admin.
3. Syed Sibtain Khalid helped in clarifying the errors that were made in the methodology part. He also did the validation and formal analysis of the paper. The originality was checked by him and the visualization portion at the end was handled by him.
4. Naseem Rao validated the paper and contributed to editing and reviewing the paper. The supervision part was done by him.

Acknowledgements
We acknowledge the support of our colleagues and reviewers.

References

[1] Kaung Myat Thu, "Types of Cyber Attacks and Incident Responses," presented at the 37th Semi-Annual Dr. Janet Liou-Mark Honors & Undergraduate Research Poster Presentation, December 1, **2022.**

[2] Orson Mengara, Anderson R. Avila, and Tiago H. Falk, "Backdoor Attacks to Deep Neural Networks: A Survey of the Literature, Challenges, and Future Research Directions," IEEE Access, Vol.**12**, pp.**29004–29023, 2024.**

[3] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash, "Hidden Trigger Backdoor Attacks," AAAI Conference on Artificial Intelligence, pp.**11957–11965, 2020.**

[4] Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen, "BackdoorBench: A Comprehensive Benchmark of Backdoor Learning," Neural Information Processing Systems (NeurIPS), **2022.**

[5] Georgios Syros, Gökberk Yar, Simona Boboila, Cristina Nita-Rotaru, and Alina Oprea, "Backdoor Attacks in Peer-to-Peer Federated Learning," ACM Transactions on Privacy and Security, Vol.**28**, No.**1**, pp.**1–28, 2025.**

[6] Robin Buchta, George Gkoktsis, Felix Heine, and Carsten Kleiner, "Advanced Persistent Threat Attack Detection Systems: A Review of Approaches, Challenges, and Trends," Digital Threats: Research and Practice, Vol.**5**, No.**4**, **2024.**

[7] Rashid Hussain Khokhar, Windhya Rankothge, Leila Rashidi, Hesamodin Mohammadian, Brian Frei, Shawn Ellis, Iago Freitas, and Ali Ghorbani, "A Survey on Supply Chain Management: Exploring Physical and Cyber Security Challenges, Threats, Critical Applications, and Innovative Technologies," International Journal of Supply and Operations Management, Vol.**11**, No.**3**, pp.**250–283, 2024.**

[8] Mohammed Saadoon and Suhad Faisal, "Malware Detection Using Machine Learning Techniques: A Review," Basrah Journal of Sciences, Vol.**42**, No.**2**, **2024.**

[9] Ghazaleh Shirvani, Saeid Ghasemshirazi, and Behzad Beigzadeh, "Federated Learning: Attacks, Defenses, Opportunities, and Challenges," arXiv preprint, March **2024.**

[10] Antonio Emanuele Cinà, Kathrin Grosse, Sebastiano Vascon, Ambra Demontis, Battista Biggio, Fabio Roli, and Marcello Pelillo, "Backdoor Learning Curves: Explaining Backdoor Poisoning Beyond Influence Functions," International Journal of Machine Learning and Cybernetics, **2024.**

[11] M. D'Onghia, F. Di Cesare, L. Gallo, M. Carminati, M. Polino, and S. Zanero, "Lookin' Out My Backdoor! Investigating Backdooring Attacks Against DL-driven Malware Detectors," ACM Workshop on Artificial Intelligence and Security (AISec), pp.**209–220, 2023.**

[12] Xiaobo Yu, Weizhi Meng, Yining Liu, and Fei Zhou, "TridentShell: An Enhanced Covert and Scalable Backdoor Injection Attack on Web Applications," Journal of Network and Computer Applications, Vol.**223, 2024.**

[13] Congcong Chen, Lifei Wei, Lei Zhang, Yuxiang Peng, and Jianting Ning, "DeepGuard: Backdoor Attack Detection and Identification Schemes in Privacy-Preserving Deep Neural Networks," Security and Communication Networks, Vol.**2022, 2022.**

[14] Shuai Zhao, Meihuizi Jia, Zhongliang Guo, Leilei Gan, Xiaoyu Xu, Xiaobao Wu, Jie Fu, Yichao Feng, Fengjun Pan, and Luu Anh Tuan, "A Survey of Recent Backdoor Attacks and Defenses in Large Language Models," arXiv preprint, June **2024.**

[15] Quentin Le Roux, El Mahdi Bourbao, Yannick Teglia, and Karim Kallas, "A Comprehensive Survey on Backdoor Attacks and Their Defenses in Face Recognition Systems," IEEE Access, Vol.**12**, pp.**47433–47468, 2024.**

[16] Ryan Williams, Carla P. Gomes, and Bart Selman, "Backdoors to Typical Case Complexity," International Joint Conference on Artificial Intelligence (IJCAI), pp.**1173–1178, 2003.**

[17] Claude Crépeau and Alain Slakmon, "Simple Backdoors for RSA Key Generation," Topics in Cryptology — CT-RSA 2003, Lecture Notes in Computer Science, Vol. **2612**, pp.**403–416, 2003.**

[18] Zhou Yang, Bowen Xu, Jie M. Zhang, Hong Jin Kang, Jieke Shi, Junda He, and David Lo, "Stealthy Backdoor Attack for Code Models," arXiv preprint, January **2023**.

[19] Johannes Klaus Fichte, Arne Meier, and Irena Schindler, "Strong Backdoors for Default Logic," ACM Transactions on Computational Logic, Vol.**25**, No.**3**, **2024.**

[20] Jimmy K. W. Wong, Ki Ki Chung, Yuen Wing Lo, Chun Yin Lai, and Steve W. Y. Mung, "Practical Implementation of Federated Learning for Detecting Backdoor Attacks in a Next-word Prediction Model," Scientific Reports, Vol.**15**, No.**1**, pp.**2328**, **2025.**

[21] Xiaoyu Yi, Gaolei Li, Wenkai Huang, Xi Lin, Jianhua Li, and Yuchen Liu, "LateBA: Latent Backdoor Attack on Deep Bug Search via Infrequent Execution Codes," Asia-Pacific Symposium on Internetware, pp.**427–436, 2024.**

[22] Wenkai Yang, Yunzhuo Hao, and Yankai Lin, "Exploring Backdoor Vulnerabilities of Chat Models," International Conference on Computational Linguistics (COLING 2025), pp.**933–946, 2025.**

AUTHORS PROFILE

Arushi Gupta is an M.Tech scholar in the field of cybersecurity at Jamia Hamdard University. With a strong foundation in cybersecurity, she has worked on various projects, including developing keyloggers and utilizing LaZagne software for credential extraction. Her research interests focus on ethical hacking, cyber threats, data security, and forensic analysis. She made a significant contribution to her first research paper, where she was responsible for conceptualizing the methodology and data curation, writing the original draft with the help of other authors, and was responsible for software validation. She also handled the project administration.

Arushi aims to further deepen her expertise in cybersecurity, contributing innovative solutions to enhance digital security and cyber resilience.

Dr. Safdar Tanweer is an Assistant Professor in the Department of Computer Science & Engineering at the School of Engineering Sciences & Technology. He holds a Ph.D. and an M.Tech. and specializes in Digital Communications, Analog Communications, Wireless Communications, Mobile Communications, Satellite Communications, Analog Electronics, and Digital Electronics. His research focuses on Signal Processing, Audio and Speech Processing, and Image Processing. Dr. Tanweer has published extensively in peer-reviewed national and international journals and has an H-index listed on Google Scholar and Scopus. He has also authored a book titled *Troubleshooting of Electrical and Electronic Equipment* as part of a CBSE Vocational Course. Additionally, he has supervised multiple Master's theses and holds a granted national patent. He made a significant contribution by giving his concept for writing this paper with a particular methodology and suggested the software that the codes have to be tested; he also reviewed the endpaper as a project admin.

Syed Sibtain Khalid is an Assistant Professor in the Department of Computer Science & Engineering at the School of Engineering Sciences & Technology. He holds an M.Tech in Electronics and Communication Engineering and a B.Tech in the same field, and he is currently pursuing a Ph.D. His teaching expertise spans Digital Signal Processing, Signals and Systems, Digital Electronics, Embedded Systems, Microcontrollers, and Analog Integrated Circuits. His research focuses on Signal Processing, Audio and Speech Processing, and Image Processing. He has published extensively in national and international peer-reviewed journals, with citations listed on Google Scholar. He is also the author of *Troubleshooting of Electrical and Electronic Equipment*, a book published as part of a CBSE Vocational Course. Beyond academia, Syed Sibtain Khalid has guided multiple Master's theses and holds a granted national patent. He is also an active member of the university's Football Club, contributing to extracurricular student activities. His dedication to research and education continues to make a significant impact in the field of engineering and technology. He had contributed his method in writing this paper and had also validated the paper by contributing to writing the original draft. He had done the supervision portion, too, and done the formal analysis. He was also part of project administration.

Dr. Naseem Rao is an Assistant Professor in the Department of Computer Science & Engineering at the School of Engineering Sciences & Technology. He holds a Ph.D., an M.Tech in Electronics and Communication Engineering, and a B.E. in the same field. His teaching expertise spans Computer Architecture, Analog and Digital Communications, Wireless and Mobile Communications, Satellite Communications, Analog Electronics, and Digital Electronics. Dr. Rao's research is primarily focused on Bio-electronics, contributing to advancements in this interdisciplinary field. He has published extensively in national and international peer- reviewed journals, with his citations indexed on Google Scholar. His innovative work has also led to a granted national patent, further emphasizing his contributions to research and development. Through his academic and research endeavors, Dr. Rao continues to make significant contributions to the field of engineering, bridging technology and biological applications. He contributed his ideas to reviewing the paper by editing some portions of the writing format and doing the visualization and validation part.

