information

MDPI

Article

Evading Cyber-Attacks on Hadoop Ecosystem: A Novel Machine Learning-Based Security-Centric Approach towards Big Data Cloud

Neeraj A. Sharma [1], Kunal Kumar [1], Tanzim Khorshed [2], A B M Shawkat Ali [1], Haris M. Khalid [3,4,*], S. M. Muyeen [5] and Linju Jose [6]

[1] Department of Computer Science and Mathematics, School of Science and Technology, The University of Fiji, Lautoka 5276, Fiji; neerajs@unifiji.ac.fj (N.A.S.); kunalk@unifiji.ac.fj (K.K.); shawkata@unifiji.ac.fj (A.B.M.S.A.)
[2] RedHat, Perth, WA 6000, Australia; t.khorshed@gmail.com
[3] College of Engineering and Information Technology, University of Dubai, Academic City, Dubai 14143, United Arab Emirates
[4] Department of Electrical and Electronic Engineering Science, University of Johannesburg, Aukland Park 2006, South Africa
[5] Department of Electrical Engineering, Qatar University, Doha 2713, Qatar; sm.muyeen@qu.edu.qa
[6] Department of Electrical and Electronics Engineering, Higher Colleges of Technology, Sharjah 7947, United Arab Emirates; linju.js@ieee.org
* Correspondence: harism.khalid@ieee.org; Tel.: +971-569750165

Abstract: The growing industry and its complex and large information sets require Big Data (BD) technology and its open-source frameworks (Apache Hadoop) to (1) collect, (2) analyze, and (3) process the information. This information usually ranges in size from gigabytes to petabytes of data. However, processing this data involves web consoles and communication channels which are prone to intrusion from hackers. To resolve this issue, a novel machine learning (ML)-based security-centric approach has been proposed to evade cyber-attacks on the Hadoop ecosystem while considering the complexity of Big Data in Cloud (BDC). An Apache Hadoop-based management interface "Ambari" was implemented to address the variation and distinguish between attacks and activities. The analyzed experimental results show that the proposed scheme effectively (1) blocked the interface communication and retrieved the performance measured data from (2) the Ambari-based virtual machine (VM) and (3) BDC hypervisor. Moreover, the proposed architecture was able to provide a reduction in false alarms as well as cyber-attack detection.

Keywords: Ambari; Big Data; Big Data in Cloud; classification; cloud computing; cyber-attack; cyber security; cyber threats; gaps; Hadoop; internet-of-things; machine learning; trust; virtualization; virtual machine

check for updates

Citation: Sharma, N.A.; Kumar, K.; Khorshed, T.; Ali, A.B.M.S.; Khalid, H.M.; Muyeen, S.M.; Jose, L. Evading Cyber-Attacks on Hadoop Ecosystem: A Novel Machine Learning-Based Security-Centric Approach towards Big Data Cloud. *Information* **2024**, *15*, 558. https://doi.org/10.3390/info15090558

Academic Editor: Haridimos Kondylakis

Received: 23 August 2024
Accepted: 5 September 2024
Published: 10 September 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

1.1. Big Data (BD)—Features and Prospects

BD defines a data type that has the features of (1) a higher volume, (2) rapid velocity, and (3) greater diversity and variety. The higher volume contains threads of both structured and unstructured data in the information technology (IT) domain. However, a bigger challenge is to analyze the BD which has variable streaming speeds and an exceeded processing capacity. And this is a continuous challenge due to the ongoing need of time and demand from the computer users towards a (1) faster and (2) higher velocity-driven integration of (1) telecommunications, (2) IT, and (3) computers as communication technologies.

1.2. BD—Continuous Demand as Well as Challenges

Due to technology integration, the demand for BD is increasing and becoming larger [1–5]. This is primarily due to the evolution of industry and smarter versions of

operating systems. It is deployed in all the communication variants of learning methods [6], intelligent approaches [7], cyber–physical infrastructures [8–14], new V2G technology systems, photovoltaic interactions, renewable energy integration, etc. Moreover, all the information-driven businesses and industries are heavily reliant on BD to have that competitive edge of technology. For example, recently, a prediction was made by Cisco Systems about internet data traffic to reach to 4.8 zeta-bytes annually [15]. However, the accumulated amount of volume for the generated data as well as its high speed and versatility in variation has opened several challenges for existing IT security companies. And the migration from conventional data collection to BD, VMs, and cloud infrastructure has been a bottleneck towards technology interactions [16]. Even the open-source contributors like Hadoop are finding it difficult to fill the security gaps of BD, which are eventually prone and vulnerable to malicious actors, hackers, and cyber criminals [17].

1.3. BD in Cloud (BDC)—Concept, Complex Procedure, Major Gaps, and Scope of This Work

The concept of BDC blends BD with the service Cloud. This new concept has been floated in the technology market to utilize the highs of (1) cloud computing, (2) its resources, and (3) services. This concept brings the luxury of BD being able to focus on a higher level of dynamics and challenges and allows Cloud to take care of the computing infrastructures and its outreach to all sizes of enterprises [17–27].

The compilation of BD is a complex procedure. The IT revolution has brought several challenges and thus the research scope is built on those challenges and gaps. With big opportunities arise big challenges. For a fully operational and potential-driven BD, several loopholes and gaps are identified by the experts. Figure 1 shows the information flow of the BD platform. The data with a (1) higher volume, (2) velocity, and (3) verifying level are fused into the BD platform. With the interface of hardware, open-source libraries, and distributed storage, the BD promises to make a computational analysis for (1) patterns, (2) signatures, and (3) flows in the information-based data. However, this computational solution has left several gaps and loopholes which need to be addressed. These gaps are (1) challenges for BD, (2) concerns of privacy, (3) risks for BD, and (4) threats to BD. The BD challenges could be further classified as: (1) access to data, (2) overkill, (3) skill shortages, (4) costs of cluster, (5) the development gap, and (6) the processing of data during real-time streaming. The privacy concerns could be further explained as: (1) government monitoring, (2) the re-identification of identity, (3) regulations of policy, 4) creepy factor, (5) a lack of transparency, (6) privacy policies, (7) the future utilization of BD, and (8) large datasets with multiple entities. The BD risks involve (1) a non-reliable commodity hardware, (2) a tender and insecure computation, (3) validation and filtering procedures for input, (4) no granular access control, and (5) a security system interface with the orthodox system. The BD threats propagate (1) the number of open ports, (2) temptation of attackers towards more data, (3) enhanced volume, velocity, and diversity, (4) undetectable malicious activities, (5) false positives, (6) security concerns due to large logs, (7) and latency of data.

The gaps and variants of BD require adequate information to address them. However, it was revealed that the cloud technology providers are hesitant to share data and information. This is due to security and privacy reasons [28,29]. To address this concern, this work included a method which can detect cyber-attacks as a BDC customer despite limited data information and resources.

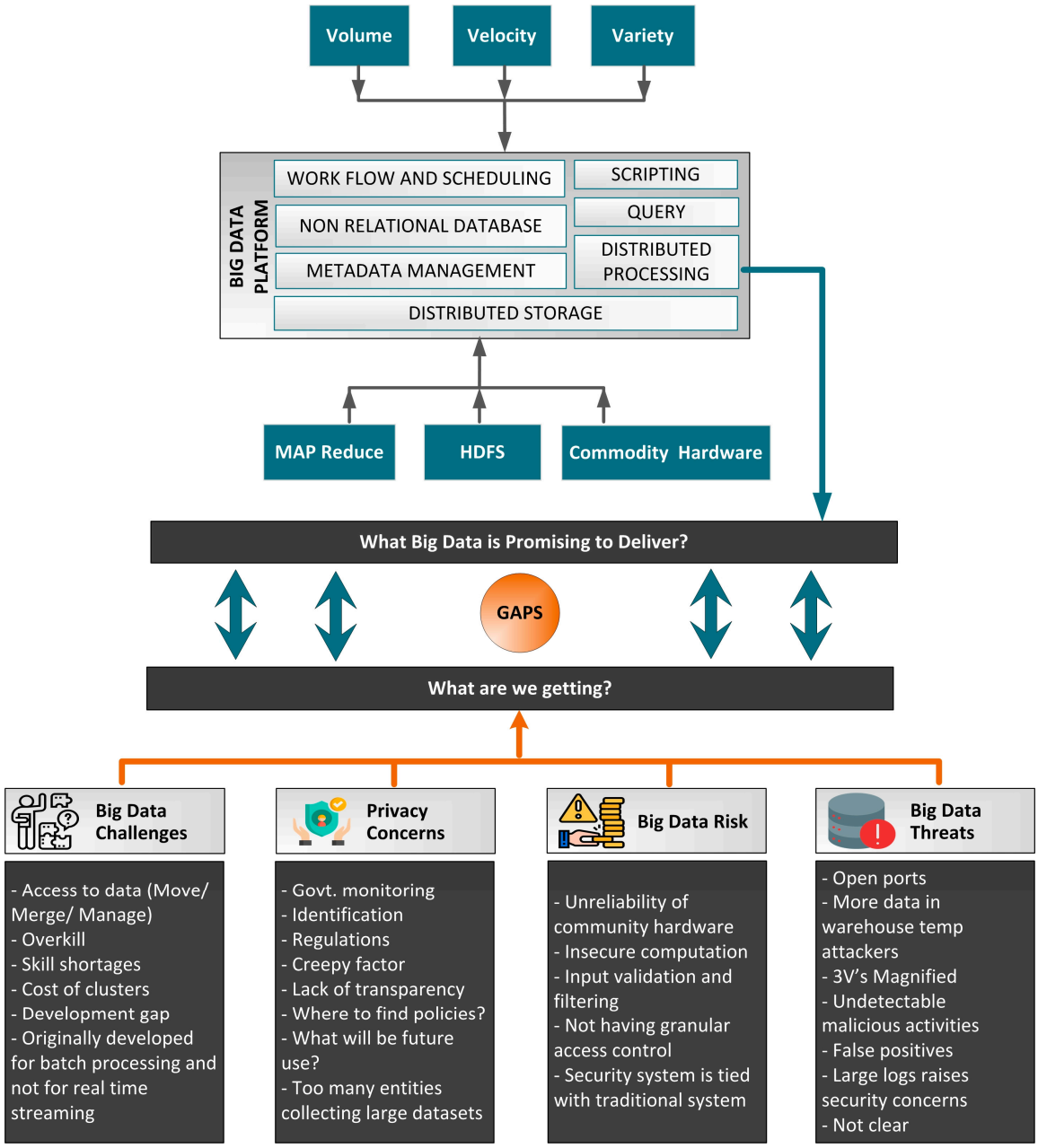

Figure 1. BD gaps and loopholes. Here, MAPreduce and HDFS are the acronyms for big data analysis model that processes data sets using a parallel algorithm on computer clusters and Hadoop Distributed File System.

1.4. Literature and Preceding Affined Review on Cyber Attacks in BD and Hadoop Ecosystems

Numerous studies in the literature have proposed discussions on cyber-attacks in BD and Hadoop ecosystems. This has also been presented in this section as well as canvassed in Table 1.

Ref. [30] discusses the integration of cyber security into the BD ecosystem. It expresses models of attacks while utilizing approaches like finite state machine and national institute standards and technology-based risk management frameworks. It expresses the architecture of security which involves a variation of low-cost BD processing to an industrial supported layered security. The processing has also been made operational towards sensitive data in multi-tenanted environments. The limitation of this work is a complete layered security solution for secure operations and processing mission critical data. The authors of [31] discuss cyber forensics and proposed a Hadoop analytical framework for an improved

accuracy and detection ratio while avoiding polynomial time complexity. A Hadoop-based distributed file system was introduced. The limitation of the work was to test the model on a more dynamic test case. The authors of [32] proposed a cloud computing framework with a data storage and task scheduling module for cyber security management. The proposed framework also consists of end-user devices and a monitoring center. The limitation of the work is its implementation towards scalability and diverse applications. The authors of [33] proposed a blockchain technology-driven solution towards the incident response process in BD systems. The limitation of this work is its validation of the proposed solution on different attack scenarios and the optimal parametrization of the algorithm. The authors of [34] discuss the cybersecurity prospects in smart grids, smart cities, and possible associated solutions. They also canvass the blockchain and IoT technology and their involvement in these infrastructures. The work lacks focus on a particular solution which can address the cyber security related in BD systems. The work in [35] proposes a novel framework which utilizes the attack probability score to detect cyber-attacks in BD systems. The probability score utilizes data-flow sacks for a better execution time. The limitation of the work is an architecture comprising of this score system since the proposed scheme is built on a virtual software-based cluster. The authors of [36] discuss the vulnerabilities which are encountered in the Apache Hadoop framework. They also address the challenges of an open-source framework. The limitation of this work is the lack of a probable solution towards these vulnerabilities to improve the immunity of the Apache Hadoop framework.

Table 1. Preceding affined works: contribution and limitations. Here BD is the acronym of big data.

Ref.	Contribution	Limitation
[30]	Proposed finite state machine and technology-based risk assessment frameworks.	Lacks a complete layered security solution.
[31]	Presented a Hadoop analytic framework for improved accuracy and detection ratio.	Requires a versatile test case and real-world potential challenges.
[32]	Suggested a cloud computing framework with data storage and task scheduling module.	Lacks scalability and diverse test case applications.
[33]	Proposed an incident response process based on block technology.	Lacks attack scenarios and optimal parameterization.
[34]	Discussed cyber security prospects in smart grids, smart cities, and associated solutions.	Deficits a particular solution towards cyber security in BD systems.
[35]	Proposed a framework which utilizes attack probability score to detect cyber-attacks in BD systems.	Lacks architecture which comprises of virtual software-based cluster.
[36]	Discussed vulnerabilities in Apache Hadoop framework.	Lacks a solution to enhance the immunity of Hadoop framework.

1.5. BDC—Purpose, Focus, and Main Contribution of This Article

The ideology of BDC is generated to accumulate and analyze the Cloud data. This involves (1) the major components of BD technology which are ideally utilized to store and analyze data-driven information in the Cloud, and (2) on-demand resource sharing technologies of Cloud computing [29,37–42]. The statistical analytics software (SAS, 9.4M7) extends the proportions of BDC towards versatility and convolution [19]. However, this blend has generated some security gaps which were inherited from the fusion. And this was further enhanced by the BD technologies. The focus of this work is to address those security gaps and propose a route of detecting cyber-attacks in the Hadoop-driven BDC.

The main contribution of this paper is as follows:

- A comprehensive review of BDC while understanding its definition, framework, main aspects, and research routes.
- Security challenge-driven investigative survey.

- Protecting BDC using a layered approach while identifying the cyber-attack types using ML techniques. These techniques also involve statistical leaning as well as rule-based learning theory.
- Implement adequate measures towards security with Hadoop and BD technologies while informing researchers, data scientists, and service providers on valuable data protection.

The graphical concept of BDC can be seen in Figure 2 where BD technologies and Cloud computing are merging to give BDC. Cloud computing provides the services of (1) resources shared using virtualization technologies, (2) being flexible, (3) on-demand and instant service, and (4) billing as utility. The BD Technologies provide services of (1) an extra-large dataset that is unable to be analyzed via traditional and conventional computing techniques, (2) big challenges and big opportunities for data handling, (3) an increased data processing velocity towards organizations, and (4) an interaction with open service providers like Hadoop, MapReduce, GridGrain, Storm, etc. The BDC focused on the 3Vs of volume, velocity, and variety along with performance and scalability. However, there was no security perspective considered which brought (1) inherited security issues, (2) Cloud computing architecture-based security issues, and (3) security issues by BD technologies.

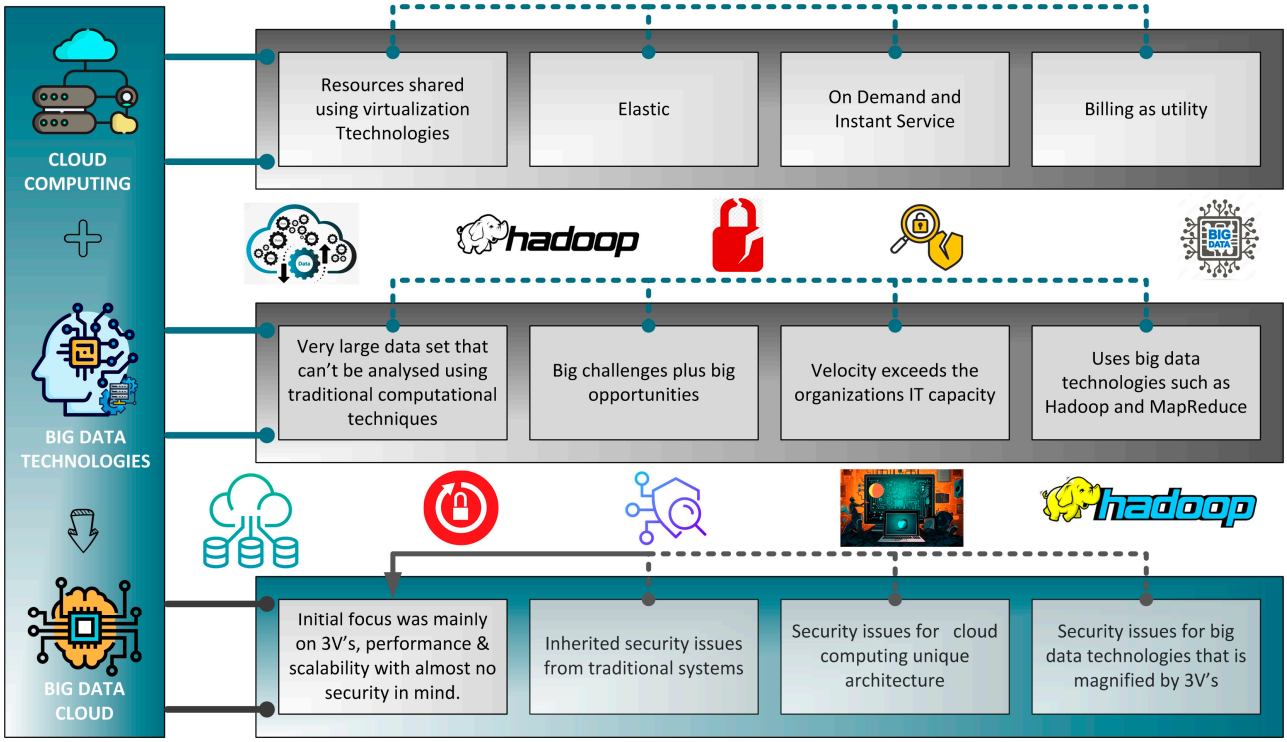

Figure 2. Graphical abstract of BDC and security vulnerabilities.

1.6. Formation of the Remaining Paper

The formation of the remaining portion of the paper is as follows: The BD and its ingredients and components are described in Section 2. The experimental design and performance data collection are illustrated in Section 3. The experimental results and performance comparison are discussed in Section 4. Finally, some conclusions and future work discussion are drawn in Section 5.

2. BDC—Main Ingredients and Components

This section comprises of (1) ingredients and the basis, (2) BDC security and research gaps, (3) vulnerabilities and impacts of the Hadoop ecosystem, and (4) steps towards BD security.

2.1. Ingredients and Basis

The main ingredients of the BDC are required to be understood to address the security issues produced from the features and unique architecture of the existing systems. The basis is usually defined in a cloud layer and BD technologies, as can be seen in Figure 3.

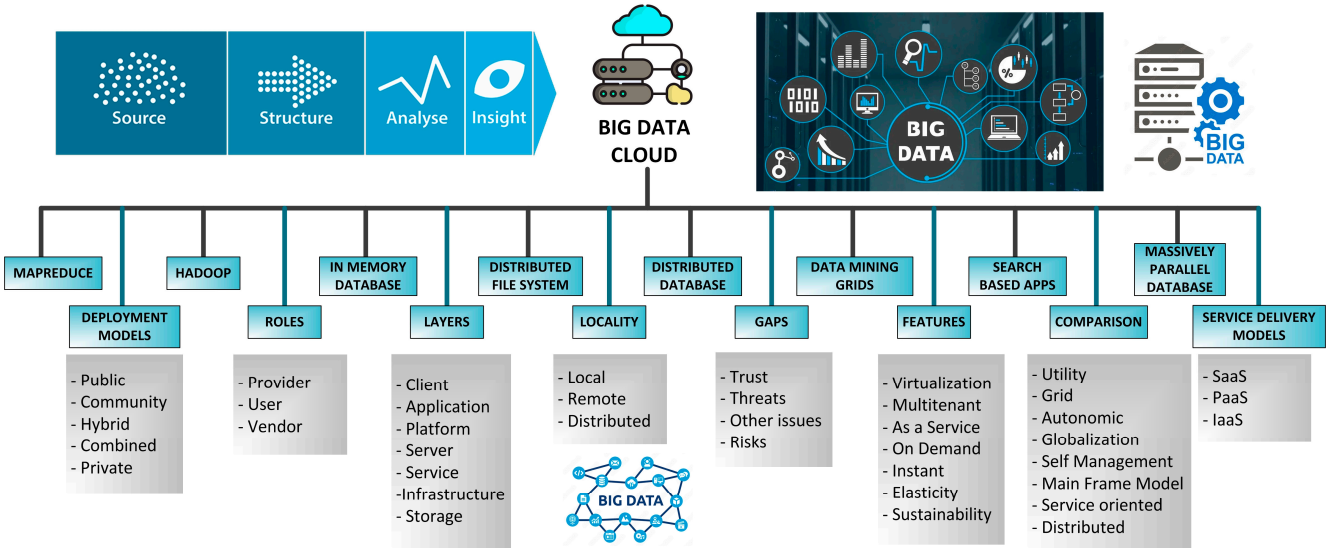

Figure 3. BDC—ingredients and basis. In this figure, SaaS, PaaS, and IaaS are the acronyms of software as a service, platform as a service, and infrastructure as a service, respectively.

2.2. BDC Security and Research Gaps

The literature finds many examples of work addressing BD and security issues of computing. However, there was no work addressing the blend of BDC security. Since researchers were exploring ways of solving security challenges using BD technologies, this was a unique route which was undiscovered and could bring directions of solutions if gaps were highlighted along with experiments [43,44].

2.3. Hadoop Ecosystem—Vulnerabilities and Impacts

The BDC is heavily reliant on BD technologies like Hadoop ecosystems. A possible situation of disabled components of Hadoop ecosystems could make the BDC more vulnerable to intrusion. An experimental setup was run in a research center to address this vulnerability with BDC. A set of denial-of-service (DoS) attacks to shut down or disrupt the network were run on Ambari, which is a Hadoop management interface [45]. This was eventually disabled and paralyzed with perfection. This resulted in halting all the communication interfaces between the management interface Ambari and the rest of the network of Hadoop. The test case was developed to evaluate the proposed security solutions and their operational feasibility towards BDC [28,29,37,38,40–42]. The test was successfully executed, and cyber-attack detection was made. The execution was then verified by the performance data generated by a virtual machine manager (VMM).

2.4. Hadoop Ecosystem and Its Steps towards BD Security

This section involves steps towards BD security which includes (1) an open-source computer library, (2) Hadoop cluster for fault tolerance, (3) Hadoop Yahoo developers' network, and BD technology aspects of security.

2.4.1. An Open-Source Computer Library

Hadoop is an open-source computer library focusing on the short falls and discrepancies of BD while enhancing reliable and scalable computing [17]. This also supports the applications being operated through BD. The operation of Hadoop is licensed by Apache.

2.4.2. Hadoop Cluster–Fault Tolerance

The open-source computer library Hadoop achieves reliability by processing the information across all the networks and multiple hosts. It thereby replicates this processed information and generates back-ups. This routine provides immunity and tolerance towards faults. And hence, it does not require any external hands like RAID technology.

2.4.3. Hadoop–Yahoo Developers Network

The multiple node-based clustered network is also operational at its full scale with the "Yahoo Developers Network". A Hadoop cluster is functional in Yahoo with 4500 nodes, 40,000 servers, and more than 1,000,000 CPUs [17,24]. This also involves: (1) ad systems, (2) a web search, and (3) scaling tests. Here, the scaling test is particularly endorsed to large-scale testing for Apache Hadoop. Figure 4 shows a comprehensive infrastructure of the Hadoop Ecosystem. It elaborates the sub features of (1) the open-source network, (2) operational services, (3) core services, (4) architecture, (5) processing part, (6) data services, (7) platform services, (8) storage part, (9) gaps, (10) challenges, and (11) security and privacy aspects.

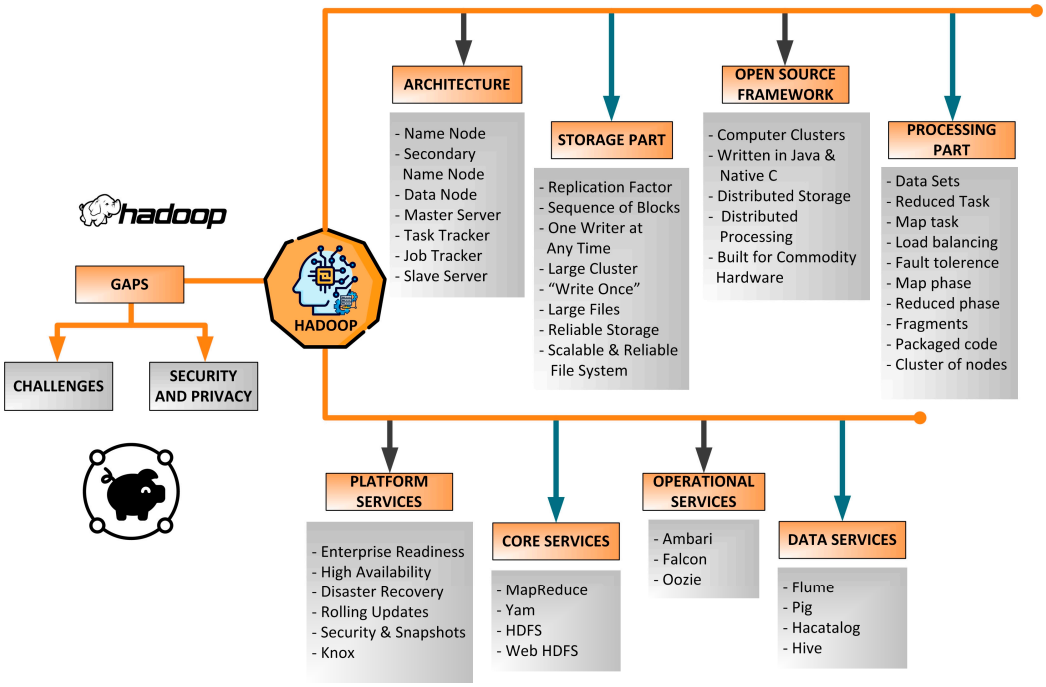

Figure 4. Hadoop Ecosystem—an infrastructure. Here, HDFS is the acronym of Hadoop Distributed File System.

2.4.4. BD Technologies—Aspects of Security

BD technologies were providing solutions towards data scalability, its integration, and computing performance. The security aspect and its possible breach were taken care of by the traditional techniques and tools. This opened a channel of vulnerabilities for malicious actors and hackers to access these knowledge-based data, which could be very worthy and vital for shaping future strategies. This could also open windows for arrays of (1) manipulations, (2) service denials, (3) disclosures, (4) alterations, etc., and the IoT could be converted into a weapon of cyber war.

3. Experimental Design and Performance Data Collection

A test case for an infected interface was prepared to represent Hadoop web management. The test case represented the Ambari web with BDC of the Hadoop ecosystem [40]. A cloud-based resident Hadoop Ecosystem was created in the BDC. This section consists of

discussions on (1) the computing processor, (2) operating system of designed attacks, (3) data collection definition and components in the role, (4) Hadoop VM-based cyber-attacks and activities, and (5) algorithm description and pseudo code.

3.1. Computing Processor

A computing processor with the following specs has been chosen to develop the test case, a dual-core Intel Pentium processor G3220, 8 GB of RAM, Intel® Virtualization Technology (VT-x). Five virtual machines (VM) were chosen for the experiment, as shown in Figure 5. Out of five VMs, four machines are the attackers/hackers, and one machine was the infected/victim.

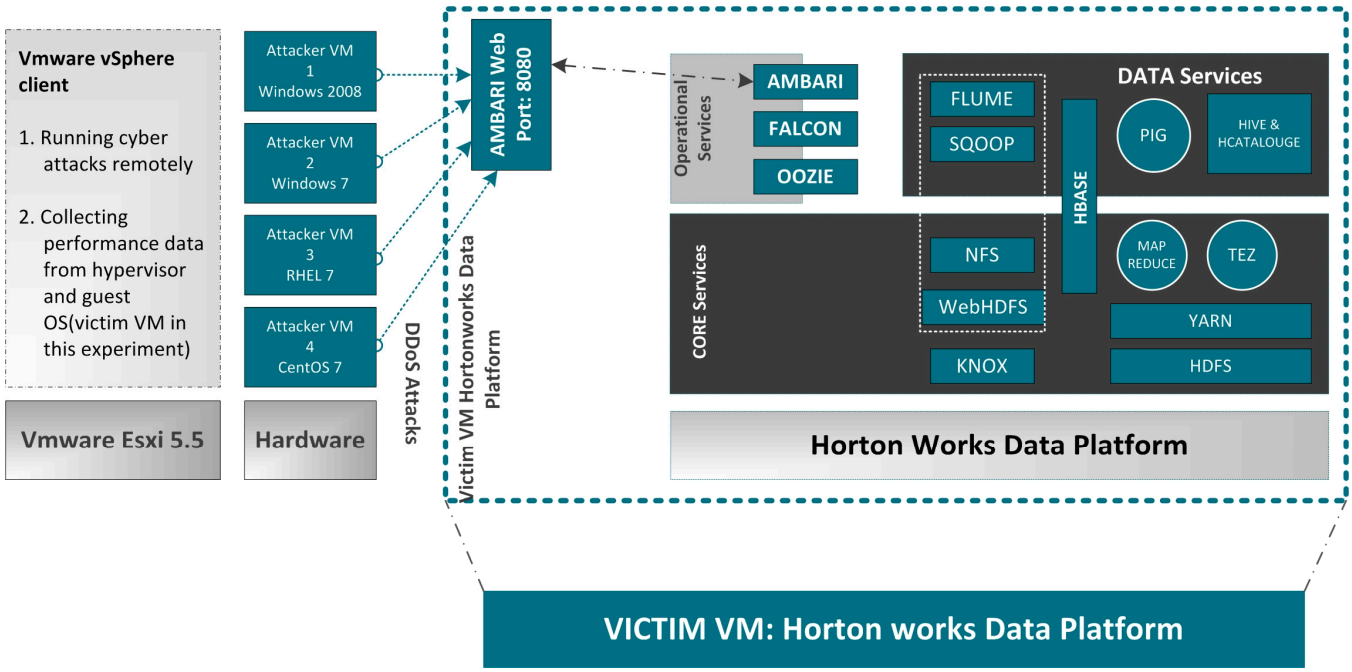

Figure 5. Experimental design.

3.2. Operating Systems of the Designed Attacks

The designed attacks were simulated on operating systems driven on VM platform [46–50] as follows: (1) occurrence of two attacker VMs using Windows-based operating system (Windows Server 2008 [51], Windows 7 [52]) (a windows-based operating system), (2) two attacker VMs were driven by Linux operating systems (RHEL7 [53], CentOS [54]), (3) the VM for the infected/victim was based on Hypervisor (Type 1, VMware ESXi 5.5 [49,50], 64-bit) and Guest OS. Note, all VMs were built on the HDP VM platform [46–48].

3.3. Performance Data Collection—Definition and Components in the Role

A performance data collection is defined as the process in which the data related to the performance measurement is gathered from the network devices and is further stored in the database. In the setup of this test case, if the definition of the performance data collection is applied, it shows the Hadoop-driven VM-collected performance data related to the performance measurement of (1) the CPU, (2) disk, (3) memory space, and (4) network utilization. This performance data was stored in VMM and HDP.

3.4. Hadoop VM-Based Cyber Attacks and Activities

A decent choice for the attack classification method was made using cloud computing architecture and its implementation in test cases [33–35]. The classification method aims here to see the impact of cyber-attacks on (1) the CPU, (2) network, (3) memory, (4) data storage, and (5) physical disk usage. Five variants of cyber-attacks were considered on

the Hadoop VM. A comprehensive explanation on attack methods' details and real-time execution can be explored in the cloud computing book [34]. The aim here is to analyze the performance of the following parameters towards the cyber-attacks: (1) the CPU, (2) network, (3) database, (4) memory, and (5) physical disk usage. Five different types of cyber-attacks and activities on Hadoop VM ports were considered to make the users aware of infected VMs. It was also ensured that HDP is performing its normal and active operation. These attacks are (1) ordinary operations and activities of Hadoop, (2) an Ambari port-based XPOIC attack [55], (3) Ambari port-based LOIC attack [56], (4) Ambari port 8080-based RTDoS attack [57], and (5) Hadoop port 80-based LOIC attack. Figure 6 shows the Ambari-based web interface in the pre-attack mode.

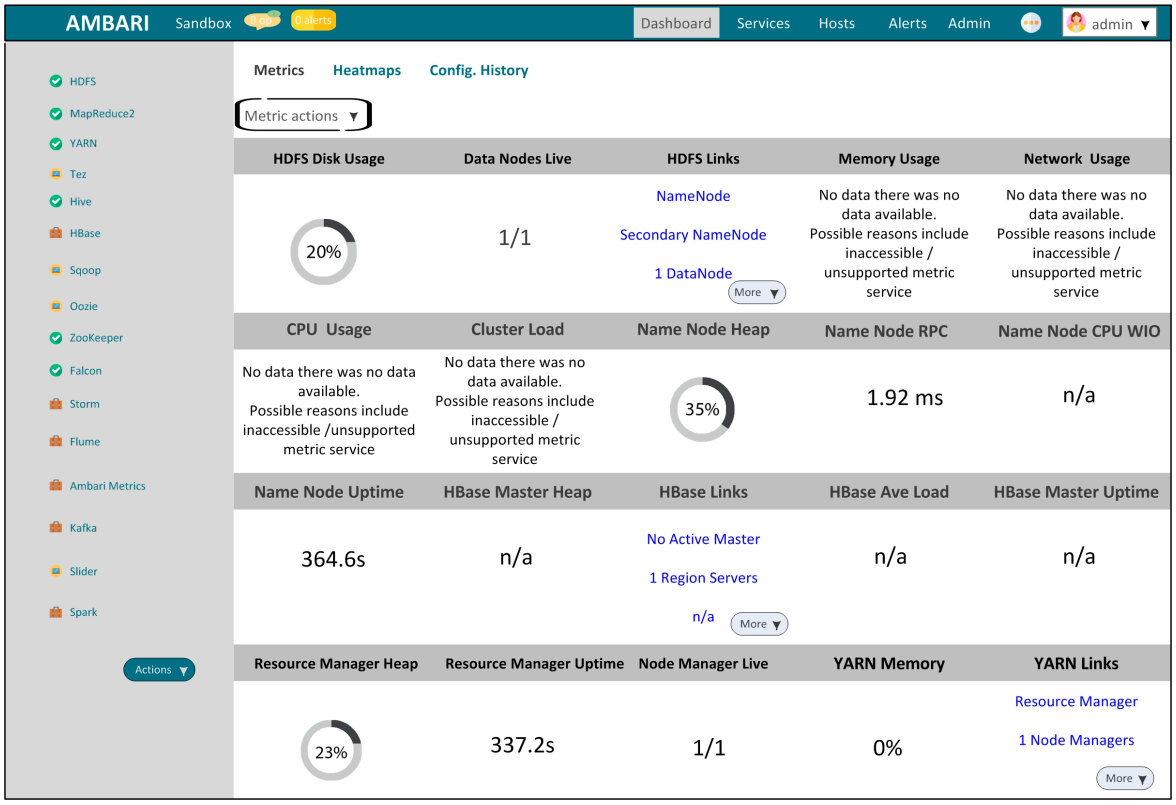

Figure 6. Ambari-based web interface in pre-attack.

An attack was simulated in 8080 ports. This simulation was successful since all the communication between Ambari and the remaining Hadoop network was truncated. Figure 7 shows the Ambari-based web interface during the phase of an attack.

At the receiving end, HDP VM received IP 192.168.186.129 from the internal network DHCP server. This required an adaptive IP address which could generate the same ID throughout the process. Therefore, a Hartonworks Sandbox with HDP was provided with the same resources. Figure 8 shows the attack being performed on the VM with IP 192.168.186.129, port 8080, and generated attack with Java LOIC [58,59]. Note that during this attack simulation, only the VM of HDP was kept activated.

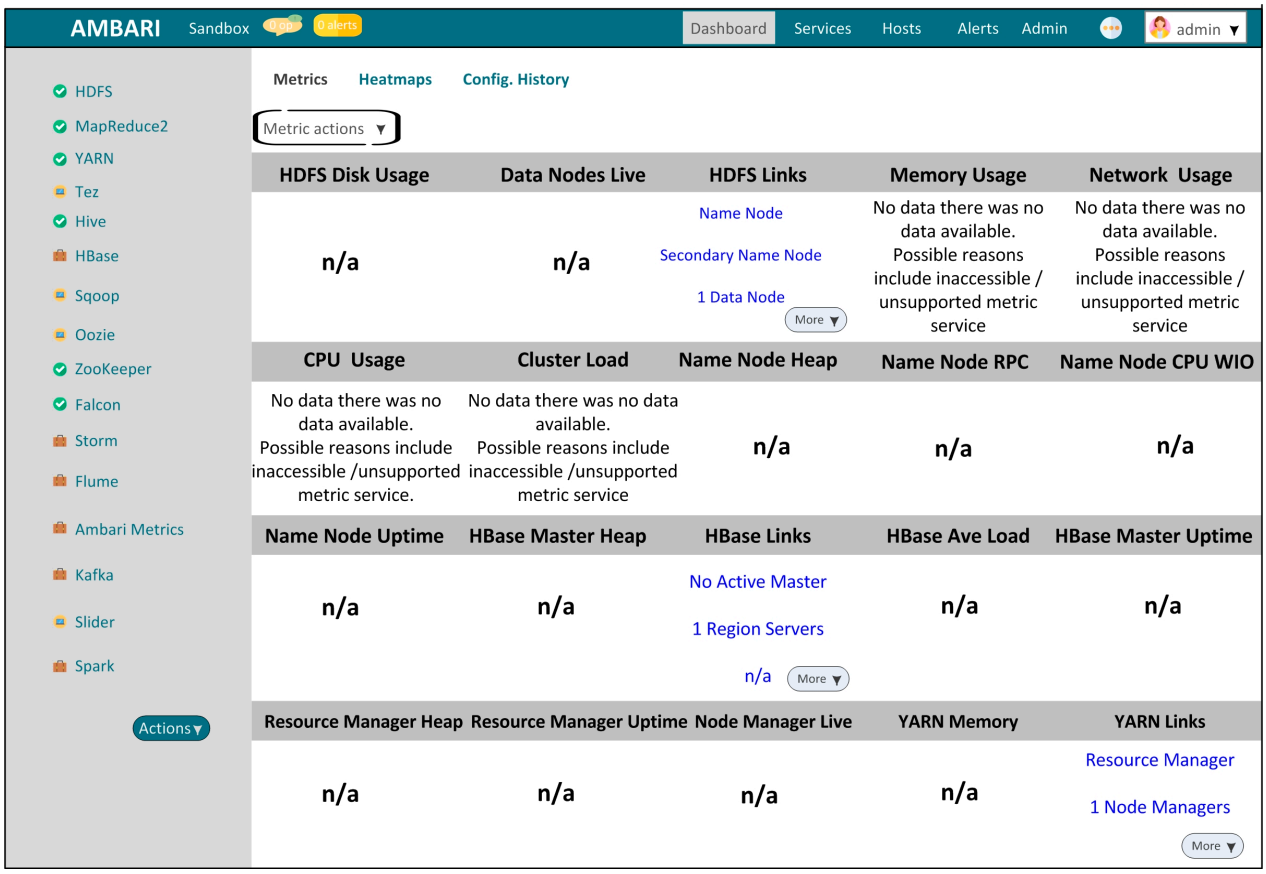

Figure 7. Ambari-based web interfaced during an attack.

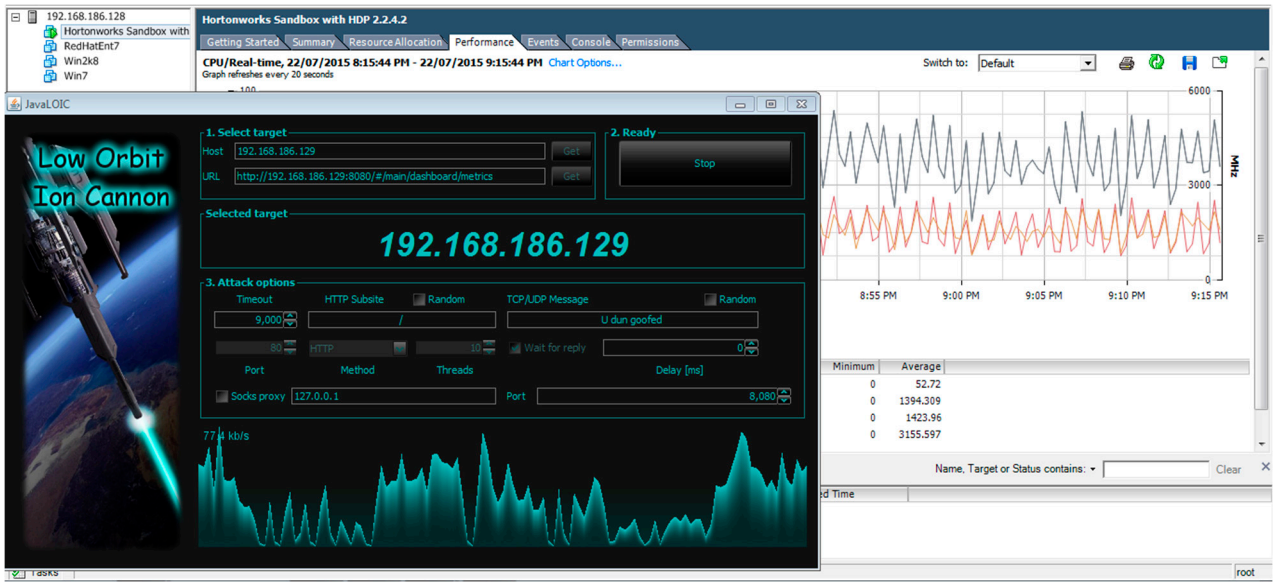

Figure 8. Attack performed on VM port 8080 with Java LOIC.

Figure 9 shows the data store-driven performance graph of Hortonworks sandbox VM. This performance graph reflects the occurrence of the Java LOIC attack. It can be observed that there are yellow spikes which define a sudden increase in the reading latency. Similarly, it can be observed that there are purple spikes which define a sudden increase in the writing latency. These latencies were observed during the 8:49:00 p.m.–9:02:00 p.m. time frame.

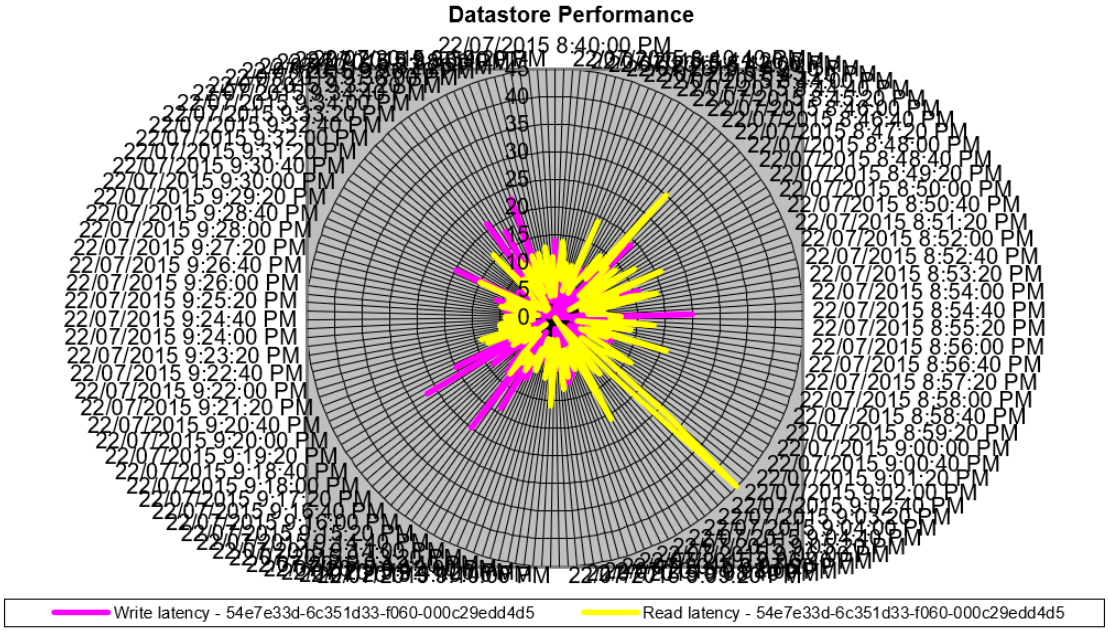

Figure 9. Hadoop VM performance graph—Generated attack using Java LOIC [28].

Figure 10 shows the Hadoop VM attack. This attack is made on default HTTP port 80. The attack was made using the RTDoS tool by Rixer [60–62]. Similarly, Figure 11 shows the graph of the CPU performance of the Hadoop VM. This performance was recorded during an attack using RTDoS by Rixer. It was observed that there is irregular CPU usage.

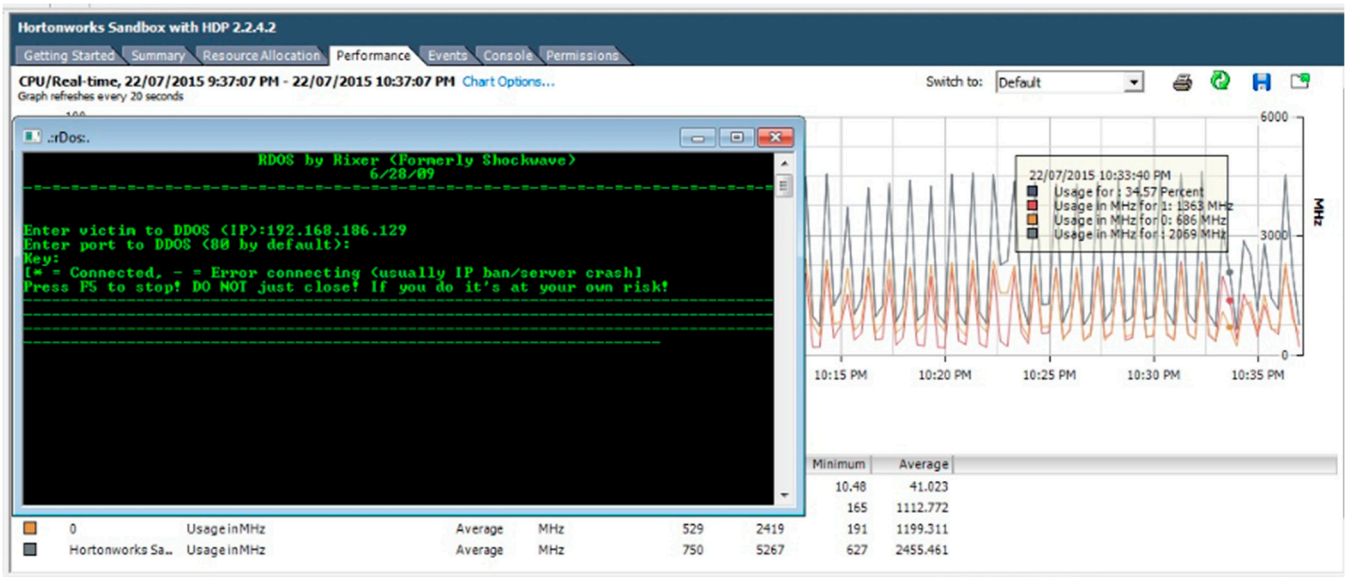

Figure 10. Hadoop VM attack—Running RTDOS (Rixer) on default HTTP port 80.

This irregularity was observed during the time frames of 10.56 p.m. to 11.08 p.m. The trends show the adequacy of the experiments in the CPU performance chart.

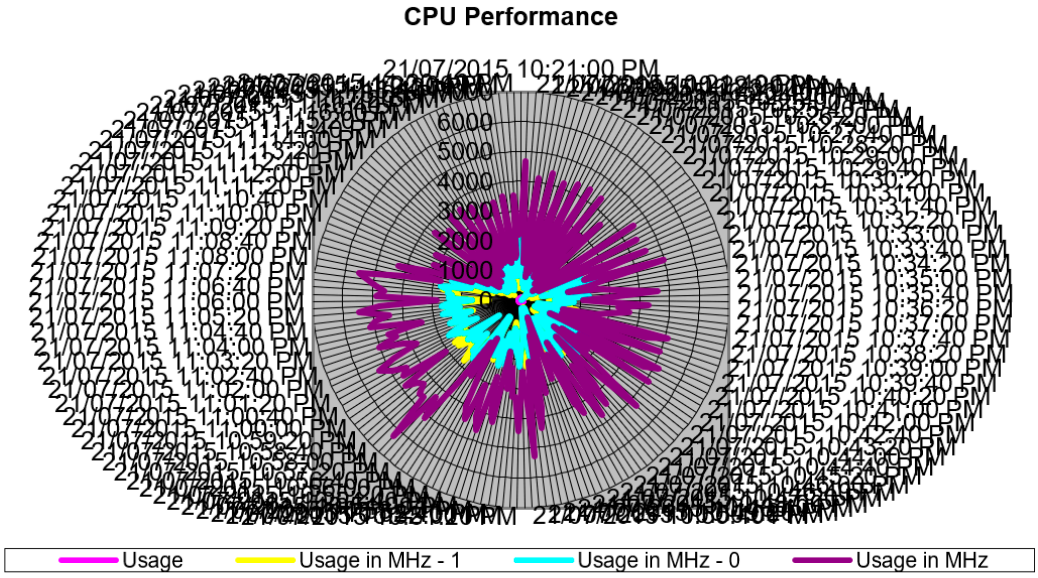

Figure 11. Hadoop VM during RTDoS attack (Rixer)—CPU performance and trends [24].

3.5. Algorithm Description—Classification of Real-Time Cyber Attacks and PART Algorithm Pseudo Code

A supervised learning-based algorithm was considered here to classify between real-time cyber-attacks and the other activities of the same course and signature. The function of training the input data with artificial intelligence (AI) was captured to map new trends and new attack examples while identifying the anomalies and variations of unseen data.

In this pursuit, several mainstream techniques have been considered towards tackling the real-time cyber-attacks These state-of-the-art techniques are: (1) Decision tree C4.5 [63], (2) Naïve Bayes [64], and SVM [65]. All these algorithms are described well in [66]. Out of all these algorithms, the projective adaptive resonance theory (PART)-based algorithm was considered [67]. This is due to its predictive range of cognitive and neural theories.

The PART algorithm involves initialization and simulation steps. The algorithm is structured as follows. Here m is defined as the number of nodes. These nodes are defined in the F_1 layer as the number of dimensions of the input data. F_2 is defined as the noncommitted node. v_i is the committed F_2 node. v_j is the committed F_2 node. σ is the distance vigilant parameter. L, α, and θ are the internal parameters. ρ_o, ρ_h, and σ are the external input parameters. T_j is the bottom-up filter input. S is the dataset. h_{ij} is defined as the selective output signaling mechanism. r_j is the matching degree. C_j is the data cluster. D_j is the associated dimension set and O is the outlier.

The initialization and simulation steps are represented as follows. In the first step, the external inputs ρ and ρo are equated. In the second step, the committed F_2 node is set as being non-committed. In the third step, the selective output signaling mechanism h_{ij} is computed for the F_1 and F_2 nodes. In the fourth step, the bottom-up filter input T_i for the committed v_i nodes is computed. In the fifth step, the matching degree r_j towards the winning node v_j is computed. All the steps are repeated to (1) satisfy the stopping condition, (2) form stable clusters, (3) compute the associated dimension, and (4) eliminate the outliers.

The pseudo code on the same has been expressed in Algorithm 1 as follows.

Algorithm 1. Pseudocode for PART algorithm.

Algorithm: Projective Adaptive Resonance Theory

Data: m, F_1, F_2, F_2, σ, L, α, and θ, ϱ_o, ϱ_h, σ, T_j.
Result: h_{ij}
if initialization/simulated is made.
    If ϱ and ϱ_O are equated,
        The PART loop is initialized.
    else
        Error Towards Classification.
    end
    if committed F_2 is set,
        Declared as being non-committed.
    else
        Error Towards Classification.
    end
    if h_{ij} is computed,
        Computation made for F_1 and F_2 nodes.
    else
        Error Towards Classification.
    end
    if T_i is computed.
        Computation made for committed v_j nodes.
    else
        Error Towards Classification.
    end
    if r_j is computed.
        Computation made for committed v_i node.
    else
        Error Towards Classification.
    end
end
Return: (satisfactory stopping condition, stable clusters, computing associated dimensions, eliminate outliers.**)**
Here,
m–number of nodes.
F_1 layer–the number of dimensions of input data.
F_2 layer–the noncommitted node.
v_i–committed F_2 node.
v_j–committed F_2 node.
σ–distance vigilant parameter.
L, α, θ–internal parameters.
ϱ_o, ϱ_h, σ–external input parameters
T_j–bottom-up filter input.
S–dataset
h_{ij}–selective output signaling mechanism.
r_j–matching degree.
C_j–data cluster.
D_j–associated dimension set.
O–outlier.

4. Experimental Results and Performance Comparison

This section talks about the experimentally driven results and performance comparison. It addresses the algorithm performance with the attack classification. It comprises (1) the performance for the attack classification, (2) confusion matrix-based performance analysis, (3) enhancing the performance analysis, and (4) a performance comparison with related works.

4.1. Performance for Attack Classification

Table 2 shows that a set of six algorithms have been tested. And for these algorithms, the classification accuracy has been evaluated. It can be observed that the Naïve Bayes algorithm was far behind in its accuracy percentage with 58.29% accuracy. The SMO has shown some better results with 86.87% classification accuracy. The J48, REPTree, Decision Tree, and PART algorithms were standout with accuracies of 90% and above. Among these algorithms, the PART algorithm was the most prominent one with the highest accuracy percentage and is the first choice for the Hadoop ecosystem.

Table 2. Attack Classification—performance comparison of algorithms. Here, SMO, J48, and REPTree are the acronyms of sequential minimal optimization, weak classifiers trees J48, and reduced error pruning tree, respectively.

Algorithm Names	SMO	J48	REPTree	Naïve Bayes	Decision Tree	PART
Attack Classification Accuracy	86.87%	90.78%	90.78%	58.29%	90.32%	91.99%

Note that the classification accuracy was captured as an average performance towards the problem.

4.2. Confusion Matrix-Based Performance Analysis

Since the classification accuracy showed an averaged form of accuracy, the confusion matrix was considered to study the performance measures of the techniques deployed in the form of (1) actual and (2) predicted classification tasks [64].

Details can be seen in Table 3 of the same, where the parameters like (1) normal activities, (2) a Ransom Denial-of-Service (RDoS) attack (Ambari-based port 8080), (3) low orbit ion cannon (LOIC) attack (Ambari-based port 8080), (4) x-orbit ion cannon (XOIC) attack (Ambari-based port 8080), (5) Java LOIC attack (Ambari-based port 8080), and (6) LOIC attack (Hadoop-based port 80) were discussed. The parameters were compared with the cross-validation technique which proposed to classify the original samples into a testing and training set to evaluate the system. The classification is usually equal to the size of 10 subsamples.

Table 3. Performance analysis—comparison of normal activities, RTDoS attack on Ambari port 8080, LOIC attack on Ambari port 8080, XPOIC attack on Ambari port 8080, Java LOIC attack on Ambari port 8080, and LOIC attack on Hadoop's port 80. Here, RDoS, LIOC, and XOIC are the acronyms of ransom denial-of-service, low-orbit ion cannon, and x-orbit ion cannon, respectively.

Attack on Activities	Normal Activities	RDoS Attack	LOIC Attack	XOIC Attack	JavaLOIC Attack	LOIC Attack	Attack Classification Performance
Normal Activities	613	13	3	4	9	10	91.9926%
RDoS Attack	22	21	0	0	0	0	
LOIC Attack	7	0	39	0	0	1	
XOIC Attack	1	0	0	23	0	0	
JavaLOIC Attack	2	0	0	0	147	1	
LOIC Attack	7	0	0	0	0	145	

Table 3 analyzes the performance analysis as follows. The first row shows the attack considered for Normal Activities. It shows here that out of 652 instances, there are (1) 613 accurately classified instances, (2) 13 instances mistakenly classified as RTDoS, (3) 3 instances mistakenly classified as LOIC, (4) 4 instances mistakenly classified as XPOIC, (5) 9 instances

mistakenly classified as Java LOIC, and (6) 10 instances mistakenly classified as LOIC for port 80. Similarly, the second row shows the attack considered for the RTDoS Attack on Ambari port 8080. It shows here that out of 43 instances, there are (1) 22 instances classified mistakenly as Normal Activities, (2) 21 accurately classifieds instances of an RTDoS attack on Ambari-based port 8080, and (3) zero classified instances for the remaining activities. The third row shows the attack considered for the LOIC Attack on Ambari port 8080. It shows here that out of 47 instances, there are (1) 7 instances mistakenly classified as Normal Activities, (2) 1 instant mistakenly classified as an LOIC attack on Hadoop port 80, and (3) 39 accurately classifieds instances of an LOIC attack on Ambari-based port 8080. The fourth row shows the attack considered for the XPOIC attack on Ambari port 8080. It shows here that out of 24 instances, there are (1) 1 accurately classified instance of normal activity and (2) 23 accurately classified instances of an XPOIC attack on Ambari-based port 8080. The fifth row shows the attack considered for Java LOIC for Ambari port 8080. It shows here that out of 150 instances, there are (1) 2 instances mistakenly classified as normal activities, (2) 1 instant mistakenly classified as LOIC Attack on Hadoop port 80, and (3) 147 accurately classified instances of Java LOIC attack on Ambari-based port 8080. The sixth row shows the attack considered for LOIC on Hadoop port 80. It shows here that out of 152 instances, there are (1) 7 instances mistakenly classified as normal activities and (2) 145 accurately classified instances of LOIC attack on Hadoop port 80.

4.3. Enhancing the Performance Analysis

In this section, routes were considered to enhance the performance analysis. Figure 12 shows a detailed structure of the graphical representation of the proposed scheme where testing and training procedures have been visualized comprehensively.

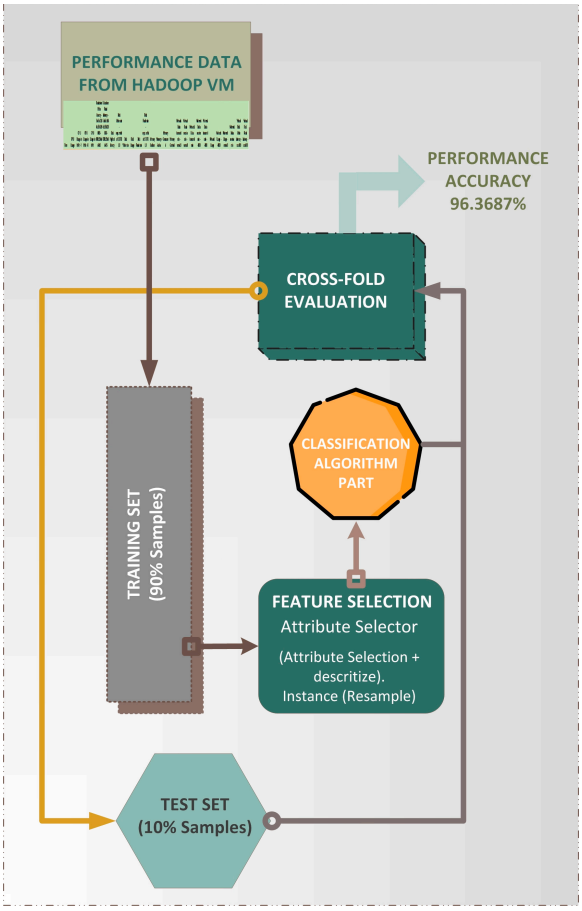

Figure 12. Graphical presentation—an ML-driven workflow.

In general, during the performance analysis, it has been observed that there are some parameters which affect the performance. For example, field error rates are one of the parameters which can be controlled if abrupt action can be taken to keep it to around 5% [65]. Similarly, the false-positive protein interactions are also one of the parameters in the experimental data which could affect the overall performance analysis [58,68,69].

To address these variants, the literature has proposed discretization to remove the noise data and further improve the classification performance [70–72]. This also involves methods such as (1) global discretization [73], (2) an ML of a Bayesian network [74], and (3) association rule construction [75]. It has been observed that filtering and discretization have further improved performance. They remove the outliers and irrelevant instances and process the same data with data mining processes.

The improved performance can be seen in Table 4 where the PART algorithm has further worked on the variants of (1) noise patterns, (2) computational accuracy, and (3) overall efficiency.

Table 4. Performance analysis after adapting the filtering approach—comparison of normal activities, RTDoS attack on Ambari port 8080, LOIC attack on Ambari port 8080, XPOIC attack on Ambari port 8080, Java LOIC attack on Ambari port 8080, and LOIC attack on Hadoop port 80.

Attack on Activities	Normal Activities	RDoS Attack	LOIC Attack	XOIC Attack	JavaLOIC Attack	LOIC Attack	Attack Classification Performance
Normal Activities	633	7	1	0	7	9	96.3687%
RDoS Attack	9	31	0	0	0	0	
LOIC Attack	0	0	41	0	0	0	
XOIC Attack	0	0	0	30	0	0	
JavaLOIC Attack	1	0	0	0	138	0	
LOIC Attack	2	0	0	0	0	162	

4.4. Performance Comaprison and Related Works

This section talks about the performance comparison with the related works in the domain of cyber-attack detection systems. The focus parameters of performance for the researchers are (1) gaining high accuracy and (2) minimizing the false alarms [76]. This also branches the benefits of cloud architectures where anomalies and attacks can be detected internally in the architecture and DDoS attacks can be distinguished [23,37,38]. Note that though the utilization of the datasets and deployed methods are different in the comparison analysis, the focus was to compare the accuracy of cyber-attack detection. The percentage-based comparative analysis was made in Figure 13. The comparison was made with (1) Pietraszek and Tanner utilizing data mining and the ML scheme [77], (2) Hoang from [78] utilizing fuzzy-inference-based system, (3) Tjhai et al. from [79] using SOM NN and the K-means algorithm-based scheme, (4) Spathoulas and Katsikas from [80], which utilized the ML-based technique, and (5) Al-Mamory and Zhang from [81] with data mining. All these techniques claimed a percentile of success in attack detection and false alarm reduction. However, refs. [77–81] have achieved accuracy of 52%, 45%, 50%, 75%, and 82%, respectively. On the other hand, the percentage performance accuracy of the proposed research is 96.3687%, which is the highest accuracy built on ML algorithms for cyber-attack detection and reducing false alarms. This is because of its features of (1) hypervisor-generated performance data for the cloud architecture and (2) PART algorithm-driven higher detection accuracy. A detailed comparison was also given in Table 5 with comparison parameters of the (1) access of cloud customers to datasets, (2) method, (3) feasibility towards cloud systems, (4) feasibility towards pre-cloud systems, (5) architectural complexities, and (6) attack detection in Hadoop. The proposed research was distinct in all the comparison metrics.

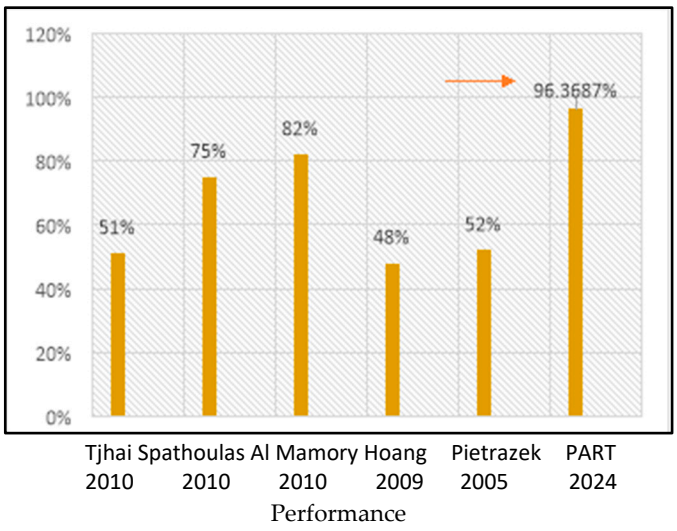

Figure 13. Percentage-based comparative analysis. From left to right, the comparison is made between references [77–81], and proposed PART algorithm respectively.

Table 5. Performance comparison with mainstream techniques. Here, DMC, ML, HMM, SOM, NN, KMA, and PART are the acronyms of data mining clustering, machine learning, hidden Markov model, self-organizing map, neural networks, k-means clustering, and projective adaptive resonance theory, respectively.

References	Pietrazek et al. [77]	Hoang et al. [78]	Tjhai et al. [79]	Spathoulas et al. [80]	Al-Mamory et al. [81]	Proposed Research
Year	2005	2009	2010	2010	2010	2023
Cloud Customer—Dataset Early Access?	No	No	No	No	No	Yes
Method	DMC and ML	HMM	SOM NN and KMA	Filtering Algorithms	Filtering Algorithms	PART Algorithm with Filtering
Cloud System Feasibility?	Yes	Yes	Yes	Yes	Yes	Yes
Pre-Cloud System Feasibility?	Yes	Yes	Yes	Yes	Yes	No
Architectural Complexity Utilization for Experimental Advantage?	No	No	No	No	No	Yes
Hadoop-Driven Attack Detection	Not Tested	Not Tested	Not Tested	Not Tested	Not Tested	Yes

5. Conclusions and Future Work

This work has proposed the options of evading cyber-attacks on the Hadoop ecosystem. It addressed the novelty from the perspective of detecting the malicious attacks and anomalies for an affected interface of Hadoop and an affected communicated network. In this pursuit, usually, the efficiency of the algorithm is determined by the amount of access and observability the proposed scheme can have towards the computing architecture. This was attained by the PART algorithm to achieve the highest accuracy of 96.387%. Though there was a limited percentage of attacks maintained to incur in the Hadoop ecosystem, it is assumed that a similar success rate would be achieved in other parts of the loops and ports as well.

The future work aims towards a multi-layered detection system to handle the BD complexity in a more comprehensive way. In this multi-layered system, each layer will be defined for a specific role. One of the primary layers should be assigned as the security layer to monitor all the processes of Hadoop. The other layer should be allocated towards the

performance data. This layer ensures the data generated towards the hypervisor. Another layer should be log-specific to portray the conventional attack detection methods. The fusion of such a multi-layered concept would allow one to streamline communication with the Hadoop ecosystem even during the phases of critical halt or affected situations. This would surely enhance the accuracy percentage levels too.

Author Contributions: N.A.S.: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. K.K.: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. T.K.: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. A.B.M.S.A.: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. H.M.K.: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. S.M.M.: Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. L.J.: Analyzed and interpreted the data; Contributed reagents; Materials; Analysis tools or Data; Wrote the Paper. All authors have read and agreed to the published version of the manuscript.

Funding: The publication of this article was not funded by any authorities.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be made available on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Han, F.; Ren, J. Analyzing Big Data Professionals: Cultivating Holistic Skills through University Education and Market Demands. *IEEE Access* **2024**, *12*, 23568–23577. [CrossRef]
2. Ahmadi, S. A Comprehensive Study on Integration of Big Data and AI in Financial Industry and its Effect on Pre-sent and Future Opportunities. *Int. J. Curr. Sci. Res. Rev.* **2024**, *7*, 66–74. [CrossRef]
3. Kamyab, H.; Khademi, T.; Chelliapan, S.; Kamarposhti, M.S.; Rezania, S.; Yusuf, M.; Farajnezhad, M.; Abbas, M.; Jeon, B.H.; Ahn, Y. The latest innovative avenues for the utilization of artificial Intelligence and big data analytics in water resource management. *Results Eng.* **2023**, *20*, 101566. [CrossRef]
4. Acciarini, C.; Cappa, F.; Boccardelli, P.; Oriani, R. How can organizations leverage big data to innovate their business models? A systematic literature review. *Technovation* **2023**, *123*, 102713. [CrossRef]
5. Gao, Q.; Cheng, C.; Sun, G. Big data application, factor allocation, and green innovation in Chinese manufacturing enterprises. *Technol. Forecast. Soc. Chang.* **2023**, *192*, 122567. [CrossRef]
6. Inayat, U.; Zia, M.F.; Mahmood, S.; Khalid, H.M.; Benbouzid, M. Learning-based methods for cyber-attacks detection in IoT systems: A survey on methods, analysis, and future prospects. *Electronics* **2022**, *11*, 1502. [CrossRef]
7. Said, Z.; Sharma, P.; Nhuong Bora, B.J.; Lichtfouse, E.; Khalid, H.M.; Luque, R.; Nguyen, X.P.; Hoang, A.T. Intelligent approaches for sustainable management and valorisation of food waste. *Bioresour. Technol.* **2023**, *377*, 128952. [CrossRef]
8. Mahmoud, M.S.; Khalid, H.M.; Hamdan, M. *Cyber-Physical Infrastructures in Power Systems: Architectures and Vulnerabilities. S and T Books*; Academic Press: Cambridge, MA, USA, 2021; pp. 1–496.
9. Khalid, H.M.; Qasaymeh, M.M.; Muyeen, S.M.; El Moursi, M.S.; Foley, A.M.; Sweidan, T.; Sanjeevikumar, P. WAMS operations in power grids: A track fusion-based mixture density estimation driven grid resilient approach towards cyber-attacks. *IEEE Syst. J.* **2023**, *17*, 3950–3961. [CrossRef]
10. Khalid, H.M.; Flitti, F.; Mahmoud, M.S.; Hamdan, M.; Muyeen, S.M.; Dong, Z.Y. WAMS operations in modern power grids: A median regression function-based state estimation approach towards cyber-attacks. *Sustain. Energy Grid Netw.* **2023**, *34*, 101009. [CrossRef]
11. Yazdinejad, A.; Dehghantanha, A.; Karimipour, H.; Srivastava, G.; Parizi, R.M. A robust privacy-preserving federated learning model against model poisoning attacks. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 6693–6708. [CrossRef]
12. Sakhnini, J.; Karimipour, H.; Dehghantanha, A.; Yazdinejad, A.; Gadekallu, T.R.; Victor, N.; Islam, A. A generalizable deep neural network method for detecting attacks in industrial cyber-physical systems. *IEEE Syst. J.* **2023**, *17*, 5152–5160. [CrossRef]
13. Yazdinejad, A.; Dehghantanha, A.; Srivastava, G.; Karimipour, H.; Parizi, R.M. Hybrid privacy preserving federated learning against irregular users in next-generation internet of things. *J. Syst. Archit.* **2024**, *148*, 103088. [CrossRef]

14. Yazdinejad, A.; Dehghantanha, A.; Srivastava, G. AP2FL: Auditable privacy-preserving federated learning framework for electronics in healthcare. *IEEE Trans. Consum. Electron.* **2023**, *70*, 2527–2535. [CrossRef]
15. Kompton, K. Cisco's Global Cloud Index Study: Acceleration of the Multi-Cloud Era. 2018. Available online: https://blogs.cisco.com/news/acceleration-of-multicloud-era (accessed on 23 August 2024).
16. Cyber Security News. Top 10 Big Data Security and Privacy Challenges Report Released. 2013. Available online: https://www.securitymagazine.com/articles/84461-top-10-big-data-security-and-privacy-challenges-report-released (accessed on 23 August 2024).
17. Hadoop Wiki. Available online: https://www.projectpro.io/hadoop-wiki (accessed on 23 August 2024).
18. Berndt, R.; Tuemmler, C.; Kehl, C.; Aehnelt, M.; Grasser, T.; Franek, A.; Ullrich, T. Open problems in 3D model and data management. In Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, Valletta, Malta, 27–29 February 2020; 1, pp. 347–354.
19. Favaretto, M.; Clercq, E.D.; Schneble, C.O. What is your definition of Big Data? Researchers' understanding of the phenomenon of the decade. *PLoS ONE* **2020**, *15*, e0228987. [CrossRef] [PubMed]
20. Pamidala, S. Implementing a Big Data Platform on IBM Cloud. 2018. Available online: https://www.ibm.com/cloud/blog/implementing-big-data-platform-cloud (accessed on 23 August 2024).
21. Abaker, I.; Hashem, T.; Yaqoob, I.; Anuar, N.B.; Mokhtar, S.; Gani, A.; Khan, S.U. The rise of "big data" on cloud computing: Review and open research issues. *Inf. Syst.* **2015**, *47*, 98–115.
22. Lucidworks. Starfish: A Hadoop Performance Tuning Tool. 2013. Available online: https://lucidworks.com/post/starfish-a-hadoop-performance-tuning-tool/ (accessed on 23 August 2024).
23. Berisha, B.; Mëziu, E.; Shabani, I. Big data analytics in Cloud computing: An overview. *J. Cloud Comput. Adv. Syst. Appl.* **2022**, *11*, 24. [CrossRef]
24. Cnudde, P. Peter Cnudde on How Yahoo Uses Hadoop, Deep Learning and Big Data Platform. 2016. Available online: https://www.infoq.com/articles/peter-cnudde-yahoo-big-data/ (accessed on 23 August 2024).
25. Talari, G.; Cummins, E.; McNamara, C.; Brien, J.O. State of the art review of Big Data and web-based decision support systems (DSS) for food safety risk assessment with respect to climate change. *Trends Food Sci. Technol.* **2022**, *126*, 192–204. [CrossRef]
26. Intel. Speed Big Data Analytics on the Cloud with an in-Memory Data Accelerator. 2019. Available online: https://www.intel.com/content/www/us/en/developer/articles/technical/speed-big-data-analytics-on-the-cloud-with-an-in-memory-data-accelerator.html (accessed on 23 August 2024).
27. Microsoft. Mapping Data Flows Performance and Tuning Guide. Microsoft Learn AI Skills Challenge. 2023. Available online: https://learn.microsoft.com/en-us/azure/data-factory/concepts-data-flow-performance (accessed on 23 August 2024).
28. Khorshed, M.T. Combating Cyber-Attacks in Cloud Computing Using Machine Learning Techniques. Master's Thesis, Deakin University, Geelong, Australia, 2016. Computer Science. Available online: https://dro.deakin.edu.au (accessed on 23 August 2024).
29. Khorshed, M.T.; Ali, A.; Wasimi, S.A. A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing. *Future Gener. Comput. Syst.* **2012**, *28*, 833–851. [CrossRef]
30. Tall, M.; Zou, C.C.; Wang, J. Integrating cybersecurity into a big data ecosystem. In Proceedings of the IEEE Military Communications Conference, San Diego, CA, USA, 29 November–2 December 2021; pp. 69–76. [CrossRef]
31. Chhabra, G.S.; Singh, V.; Singh, M. Hadoop-based analytic framework for cyber forensics. *Int. J. Commun. Syst.* **2018**, *31*, e3772. [CrossRef]
32. Xu, G.; Yu, W.; Chen, Z.; Zhang, H.; Moulema, P.; Fu, X.; Lu, C. A cloud computing based system for cyber security management. *Int. J. Parallel Emergent Distrib. Syst.* **2014**, *30*, 29–45. [CrossRef]
33. Moreno, J.; Serrano, M.A.; Fernandez, E.B.; Fernández-Medina, E. Improving incident response in big data ecosystems by using blockchain technologies. *Appl. Sci.* **2020**, *20*, 724. [CrossRef]
34. Sadik, M.; Ahmed, L.; Sikos, F.; Islam, A.K.M.N. Towards a sustainable cybersecurity ecosystem. *Computers* **2020**, *9*, 74. [CrossRef]
35. Aditham, S.; Ranganathan, N. A novel framework for mitigating insider attacks in big data systems. In Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA, 29 October–1 November 2015; pp. 1876–1885. [CrossRef]
36. Kaushik, A.; Srivastava, V.K. Threat to big data: Common weakness enumerations and vulnerabilities for Hadoop framework. *Int. J. Res. Anal. Rev.* **2020**, *7*, 280–286.
37. Khorshed, M.T.; Wasimi, S. Monitoring insiders' activities in cloud computing using rule-based learning. In Proceedings of the IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Changsha, China, 16–18 November 2011; pp. 757–764.
38. Deshmukh, R.V.; Devadkar, K.K. Understanding DDoS attack and its effect in Cloud environment. *Procedia Comput. Sci.* **2015**, *49*, 202–210. [CrossRef]
39. Enterprise Bigdata Framework. The 4 Characteristics of Big Data. Available online: https://www.bigdataframework.org/the-four-vs-of-big-data/ (accessed on 23 August 2024).
40. Khorshed, M.T.; Ali, A.; Wasimi, S. Trust issues that create threats for cyber-attacks in cloud computing. In Proceedings of the IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS), Tainan, Taiwan, 7–9 December 2011; pp. 900–905.

41. Khorshed, M.T.; Ali, A.S.; Wasimi, S.A. Combating cyber-attacks in cloud systems using machine learning. In *Security, Privacy and Trust in Cloud Systems*; Nepal, S., Pathan, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 407–431.
42. Ali, S.; Azad, S.; Khorshed, T. *Securing the Smart Grid: A Machine Learning Approach*; Smart Grids, Part of the Green Energy and Technology book series (GREEN); Springer: Berlin/Heidelberg, Germany, 2013; pp. 169–198.
43. Cardenas, A.; Manadhata, P.K.; Rajan, S.P. Big data analytics for security. *IEEE Secur. Priv.* **2013**, *11*, 74–76. [CrossRef]
44. Fernando, Y.; Chidambaram, R.M.; Wahyuni-TD, I.S. The impact of Big Data analytics and data security practices on service supply chain performance. *Benchmarking Int. J.* **2018**, *25*, 4009–4034. [CrossRef]
45. What Is Apache Ambari? Mar. 2023. Available online: https://intellipaat.com/blog/what-is-apache-ambari/?US (accessed on 23 August 2024).
46. IBM Analytics. Hortonworks Data Platform: An Open-Architecture Platform to Manage Data in Motion and at Rest. Available online: https://www.ibm.com/downloads/cas/DKWR4KZB (accessed on 23 August 2024).
47. Jain, S. Exploring Ambari Alerts in Hortonworks. 2020. Available online: https://blog.clairvoyantsoft.com/exploring-ambari-alerts-in-hortonworks-936c668df02b (accessed on 23 August 2024).
48. Intel. Intel® Pentium® Processor G3220—3M Cache, 3.00 GHz. Available online: https://www.intel.com/content/www/us/en/products/sku/77773/intel-pentium-processor-g3220-3m-cache-3-00-ghz/specifications.html (accessed on 23 August 2024).
49. VMware. VMware ESXi 5.5.0 (ESXi 5.5.0 ed.). Available online: https://my.vmware.com/web/vmware/details?productId=352&downloadGroup=ESXI550#product_downloads (accessed on 23 August 2024).
50. VMware. The vSphere Client. Apr. 2022. Available online: https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.vm_admin.doc/GUID-588861BB-3A62-4A01-82FD-F9FB42763242.html (accessed on 23 August 2024).
51. Windows Server 2008 Editions and System Requirements. Available online: https://www.techotopia.com/index.php/Windows_Server_2008_Editions_and_System_Requirements (accessed on 23 August 2024).
52. Microsoft. Windows. Available online: https://windows.microsoft.com/en-us/windows/windows-help#windows=windows-7 (accessed on 23 August 2024).
53. RedHat. Chapter 4: New Features Redhat Enterprise Linux 7. Available online: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/7.7_release_notes/new_features (accessed on 23 August 2024).
54. CentOS. Download centOS. Available online: http://www.centos.org/download/ (accessed on 23 August 2024).
55. Hudaib, A.Z. The principles of modern attacks analysis for penetration tester. *Int. J. Comput. Sci. Secur.* **2015**, *9*, 22–84.
56. Batishchev, M. LOIC. 2012. Available online: http://sourceforge.net/projects/loic/ (accessed on 23 August 2024).
57. Security Tube. Attack with RDoS and T3c3i3. 2012. Available online: http://www.securitytube.net/video/4719 (accessed on 23 August 2024).
58. InfoSec. DOS Attacks and Free DOS Attacking Tools. 2015. Available online: http://resources.infosecinstitute.com/dos-attacks-free-dos-attacking-tools/ (accessed on 23 August 2024).
59. Sourceforge. Low Orbit Ion Cannon—A Java-Based Network Stress Testing Application. 2013. Available online: http://sourceforge.net/projects/javaloic/ (accessed on 23 August 2024).
60. Witten, H.; Frank, E.; Hall, M.A. *Data Mining: Practical Machine Learning Tools and Techniques (The Morgan Kaufmann Series in Data Management Systems)*, 3rd ed.; Morgan Kaufmann: San Francisco, CA, USA, 2011; pp. 1–664.
61. Kohavi, R.; Provost, F. Glossary of terms. *Mach. Learn.* **1998**, *30*, 271–274.
62. Orr, K. Data quality and systems theory. *Commun. ACM* **1998**, *41*, 66–71. [CrossRef]
63. Quinlan, J.R. *Book Review—C4.5: Programs for Machine Learning*; Machine Leaning; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1994; Volume 16, pp. 235–240.
64. John, G.H.; Langley, P. Estimating continuous distributions in Bayesian classifiers. In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, CA, USA, 18–20 August 1995; pp. 338–345.
65. Platt, J.C. *Fast Training of Support Vector Machines Using Sequential Minimal Optimization*; CogNet: Chennai, India, 1999; pp. 185–208.
66. Frank, E.; Witten, I.H. Generating accurate rule sets without global optimization. In Proceedings of the Fifteenth Inter-national Conference on Machine Learning, San Francisco, CA, USA, 24–27 July 1998; pp. 144–151.
67. Cao, Y.; Wu, J. Projective ART for clustering data sets in high dimensional spaces. *Neural Netw.* **2002**, *15*, 105–120. [CrossRef] [PubMed]
68. Gavin, A.-C.; Bösche, M.; Krause, R.; Grandi, P.; Marzioch, M.; Bauer, A.; Schultz, J.; Rick, J.M.; Michon, A.-M.; Cruciat, C.-M.; et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* **2002**, *415*, 141–147. [CrossRef] [PubMed]
69. Xiong, H.; Pandey, G.; Steinbach, M.; Kumar, V. Enhancing data analysis with noise removal. *IEEE Trans. Knowl. Data Eng.* **2006**, *18*, 304–319. [CrossRef]
70. Liu, H.; Hussain, F.; Tan, C.L.; Dash, M. Discretization: An enabling technique. *Data Min. Knowl. Discov.* **2002**, *6*, 393–423. [CrossRef]
71. Xiao, N. Maximum Homogeneity Clustering for One-Dimensional Data. Available online: https://cran.r-project.org/web/packages/oneclust/vignettes/oneclust.html (accessed on 23 August 2024).
72. Dougherty, J.; Kohavi, R.; Sahami, M. Supervised and unsupervised discretization of continuous features. In *Machine Learning, Proceedings of the Twelfth International Conference, Tahoe City, CA, USA, 9–12 July 1995*; Morgan Kaufmann: Burlington, MA, USA; pp. 194–202.

73. Frank, E.; Witten, I.H. *Making Better Use of Global Discretization*; Technical Report; Morgan Kaufmann Publishers: San Francisco, CA, USA, 1999; pp. 1–12.
74. Friedman, N.; Goldszmidt, M. *Discretizing Continuous Attributes While Learning Bayesian Networks*; ICML: Vienna, Austria, 1996; pp. 157–165.
75. Lud, M.-C.; Widmer, G. Relative unsupervised discretization for association rule mining. In *Principles of Data Mining and Knowledge Discovery*; Zighed, D.A., Komorowski, J., Żytkow, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2000; pp. 148–158.
76. Patel, A.; Taghavi, M.; Bakhtiyari, K.; Júnior, J.C. An intrusion detection and prevention system in cloud computing: A systematic review. *J. Netw. Comput. Appl.* **2013**, *36*, 25–41. [CrossRef]
77. Pietraszek, T.; Tanner, A. Data mining and machine learning—Towards reducing false positives in intrusion detection. *Inf. Secur. Tech. Rep.* **2005**, *10*, 169–183. [CrossRef]
78. Hoang, D.; Hu, J.; Bertok, P. A program-based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference. *J. Netw. Comput. Appl.* **2009**, *32*, 1219–1228. [CrossRef]
79. Tjhai, G.C.; Furnell, S.M.; Papadaki, M.; Clarke, N.L. A preliminary two-stage alarm correlation and filtering system using SOM neural network and K-means algorithm. *Comput. Secur.* **2010**, *29*, 712–723. [CrossRef]
80. Spathoulas, G.P.; Katsikas, S.K. Reducing false positives in intrusion detection systems. *Comput. Secur.* **2010**, *29*, 35–44. [CrossRef]
81. Al-Mamory, S.O.; Zhang, H. New data mining technique to enhance IDS alarms quality. *J. Comput. Virol.* **2010**, *6*, 43–55. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.