applied sciences

MDPI

Article

Improving Incident Response in Big Data Ecosystems by Using Blockchain Technologies

Julio Moreno [1,*], Manuel A. Serrano [2], Eduardo B. Fernandez [3] and Eduardo Fernández-Medina [1]

1 GSyA Research Group, University of Castilla-La Mancha, 13071 Ciudad Real, Spain; Eduardo.FdezMedina@uclm.es
2 Alarcos Research Group, University of Castilla-La Mancha, 13071 Ciudad Real, Spain; Manuel.Serrano@uclm.es
3 Computer and Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431, USA; Fernande@fau.edu
* Correspondence: Julio.Moreno@uclm.es

Received: 13 December 2019; Accepted: 17 January 2020; Published: 20 January 2020

Abstract: Big data ecosystems are increasingly important for the daily activities of any type of company. They are decisive elements in the organization, so any malfunction of this environment can have a great impact on the normal functioning of the company; security is therefore a crucial aspect of this type of ecosystem. When approaching security in big data as an issue, it must be considered not only during the creation and implementation of the big data ecosystem, but also throughout its entire lifecycle, including operation, and especially when managing and responding to incidents that occur. To this end, this paper proposes an incident response process supported by a private blockchain network that allows the recording of the different events and incidents that occur in the big data ecosystem. The use of blockchain enables the security of the stored data to be improved, increasing its immutability and traceability. In addition, the stored records can help manage incidents and anticipate them, thereby minimizing the costs of investigating their causes; that facilitates forensic readiness. This proposal integrates with previous research work, seeking to improve the security of big data by creating a process of secure analysis, design, and implementation, supported by a security reference architecture that serves as a guide in defining the different elements of this type of ecosystem. Moreover, this paper presents a case study in which the proposal is being implemented by using big data and blockchain technologies, such as Apache Spark or Hyperledger Fabric.

Keywords: big data security; blockchain; incident response; forensic readiness

1. Introduction

Big data is here. It has ceased to be hype and has become a reality in medium and large companies of any sector. These organizations use the analytical capabilities of big data ecosystems to obtain valuable information to carry out their daily activities and to help in the decision making of top management [1]. A Big data ecosystem is usually defined as a set of different components that allow the storage, processing, preparation, visualization and delivery of useful information to target applications or end-users. Normally, these components are very complex, and need to work together if valuable insights are to be obtained [2]. These components can range from relational and non-relational databases to machine learning algorithms, virtual machines or containers, or even the underlying hardware that supports their services [3].

The use of new technologies brings fresh opportunities and perspectives; however, these new technologies can create new problems; big data ecosystems are no exception. These issues are related

not only to the three typical V's of big data (volume of data, velocity of the rate of data processing and variety of data types and sources), but also to security. Big data not only increases the size of the problems related to security and privacy, as faced in the traditional management of security; it also has new vulnerabilities and threats to face. This problematic situation should be addressed by using different techniques and mechanisms [4]; for example, considering how to increase the privacy of the data used by big data through the use of anonymization techniques [5]. Moreover, these security issues are intensified by the fact that big data was not initially conceived of as a secure environment [6]; the main security problems are thus related to the specific architecture of big data, which makes it more difficult to ensure the privacy of the data that is being used [7].

When carrying out a big data project, it is important to consider the inherent characteristics of this kind of ecosystem, and how it is different than a traditional software development. As stated before, big data ecosystems are usually very complex systems where different technologies work together to reach a goal. Moreover, this kind of systems is usually deployed in companies where a short time-to-market and the need to adapt to change is essential. In addition, many of those companies are immersed in an internal cultural evolution, endeavoring to be more agile and innovative; this is the case in the DevOps movement [8]. As a consequence of this pressure, along with the misunderstanding and misuse of agile methodologies, the development of big data ecosystems is usually chaotic, not giving enough consideration to the importance of the requirements and risks of the system. For that reason, we have proposed a process whose purpose is to implement secure big data ecosystems [9] based on the concept of security-by-design. This process focuses mainly on the phases of analysis, design and implementation of the big data; but what happens once the ecosystem is already deployed and working?

In any kind of IT system, when a security incident occurs, it is crucial to have an early response mechanism that guarantees the continuity of the system. To achieve this objective, it is important to have accurate and reliable information on those operations carried out that may have influenced the incident, so that the response to these incidents is adequate [10]. In a complex environment with a large number of nodes interacting with each other, where multiple users are accessing and generating new data, it is important to have a system that can store forensic evidence on the operations performed in the big data. A monitoring mechanism that captures some subsets of these interactions and uses them to generate interesting insights can often lead to a greater understanding of a system's behavior [11]. This is related to the concept of forensic readiness, which seeks not only to handle the incidents, but also to anticipate them and minimize the costs of investigating their causes. Hence, forensic readiness is usually defined as the ability of an organization to collect, preserve, protect and analyze digital evidence which can be used effectively in any legal matter [12].

In this paper, we propose a process to cover the secure operation of the big data ecosystem. In a nutshell, we deal with how to implement an incident response strategy for this kind of environments. This process will be defined by means of different phases and activities. Moreover, we include a guide for the implementation of this process based on blockchain technologies. Usually, blockchain technologies are defined depending on the purpose they perform. The most common functions include cryptocurrencies, smart contracts, and distributed ledger systems between businesses [13]. To tackle the security challenges stated above, we propose to use blockchain as a distributed ledger to maintain the data needed to address security issues, including all operations performed on data stored in big data, together with the different alerts and warnings produced by the big data ecosystem. The use of blockchain brings some advantages, including immutability and increased data traceability, although it comes at a cost of greater technical complexity. The use of the blockchain technology supposes an evolution with respect to the relational databases in certain aspects; such as that with blockchain the system does not have a single point of failure, which implies, that in case a participant node makes an incorrect change it will be immediately corrected by the other nodes. This function increases the trust of the data stored in the blockchain system so that it can be used as audit-proof evidence in a legal process [14].

It should also be highlighted that the use of machine learning techniques has proved to be a very useful tool when it comes to identifying security problems [14]. The storage of the different security events that happen in the system can thus be leveraged by implementing an intelligent system that studies both the causes of the event and possible solutions to it. Moreover, the analysis of data from previous events can help predict security incidents before they occur, thereby reducing the impact on the big data ecosystem.

In an effort to facilitate understanding of our proposal, the paper also presents an example of the application of our secure operation process for big data. This case study is based on the design and implementation of a hydroponic crop that has different automated sensors and actuators, thanks to a big data ecosystem (cyber-physical system). This big data environment is connected to a private blockchain network, implemented by Hyperledger Fabric, which aims to store all the events produced by the actuators for later analysis through machine learning techniques, such as PCA (Principal Component Analysis) or neural networks. This analysis enables security incidents to be studied, the aim being to make them easier to predict or detect at an earlier stage. This case study can be considered as the first step in the validation of our proposal.

In this paper we present our proposal for a process to manage the response to incidents in big data ecosystems. This process is an evolution of our proposals for a security reference architecture [15] for big data and a process for the analysis, design and implementation of this type of ecosystem [9]. The main contributions of our paper are therefore: (I) the creation of an abstract incident response process for big data ecosystems, (II) its integration with a general process of creation and implementation of secure big data, (III) a technological proposal to support forensic readiness through the use of blockchain and machine learning technologies, and (IV) a case study, which shows the application of our proposal in a cyber-physical environment.

We organize the content of the paper as follows: in the first section, we explain the related work that covers similar issues as our proposal. Section 3 explains the background needed to understand and build our proposal; this includes an explanation of the evolution of the previous research we have conducted. Next, the results obtained from our research are presented in Section 4, which focused on explaining our proposal for approaching the secure operation of big data ecosystems, including all its phases and how to carry these out. As a way to better understand our proposal, Section 5 describes a case study and presents a possible implementation of our proposal. Finally, we present conclusions and the future work proposed.

2. Related Work

Before creating our proposal, we carried out a search for similar studies which aimed to ensure the operation or the incident response process of a big data ecosystem, and more specifically, by using blockchain technologies. However, this search did not produce any closely related results that had followed a methodological perspective similar to ours.

There are numerous proposals that use blockchain to store big data and in doing so, improve security aspects. In [16], the authors propose a mechanism that approaches the problem of data redundancy in a personnel information management system. The authors of [17] expose a method to perform the access control of big data ecosystems by means of blockchain.

We also tried to look for proposals as regards the monitoring of the operation of Big Data ecosystems, or studies related to managing the response to incidents in this type of environment; however, we found only one paper in this sense, on a specific tool that focuses on performance and not on security [18], and another study carried out ten years ago on how to monitor distributed environments [19].

On the other hand, there are also proposals that use big data as a way to implement and improve SIEM (Security Information and Event Management) systems for early detection and real time analysis of patterns and threats within another system; such as [20], where the authors propose a big data architecture to analyze relevant events to improve security, or [21], which exposes a method to perform

big data analytics using machine learning techniques. In addition, there are several works that address the security of distributed ecosystems through the use of machine learning algorithms. These works obtain the data from different data sources such as logs or system sensors, which allows them to perform analysis for intrusion detection [22] or vulnerability analysis [23]. There are, nonetheless, some proposals for using machine learning to support an incident response process; for example, in [24] the authors propose a system that uses a supervised learning model to analyze different types of attacks, or in [25], where the authors use machine learning models to categorize whether a reported event is an incident or not. None of these pieces of work follows a methodological approach, however, nor do they focus on ensuring the operation of the big data environment itself. Furthermore, these studies do not use blockchain as a way of storing events and incidents from the systems they monitor. There are established processes that can help to conduct that task, such as Veeramachaneni's et al. [26] work, which proposes a series of steps to teach a big data system to detect attacks: first, an unsupervised learning algorithm is performed to detect outliers that are then analyzed by cybersecurity experts; the result of this analysis will be used as a set of data to train the first unsupervised learning algorithm, so that different iterations are executed in the effort to improve prediction.

To our knowledge, therefore, there is no proposal from the scientific community that is similar to ours. Table 1 highlights the main differences between the papers showed in this section. Our proposal brings together all these different concepts, which when used together, allow us to improve security aspects of big data ecosystems during their operation phase. Therefore, as can be observed, there is no proposal that covers the same topics as ours.

Table 1. Comparison of the related work.

Proposals	Incident Response	Methodological Approach	Security Aspects	Blockchain Technologies	Machine Learning
Big data with blockchain [16,17]	✗	✗	✓	✓	✗
Big data operation monitor [18,19]	✗	✗	✗	✗	✗
Machine learning for security [20–23]	✗	✗	✓	✗	✓
Machine learning for incident response [24,25]	✓	✗	✗	✗	✓

3. Background

In this section, we explain the main topics that are used to approach our proposal for secure operation in big data environments. Blockchain technology serves as a tool to store the different data that will be used in managing the state of the ecosystem, as well as to analyze and respond to incidents that occur. Hence, the inherent characteristics of this type of technology will be used to guarantee the immutability and traceability of the data, thus increasing the security of the big data ecosystem. In addition, our proposal does not attempt to manage the entire operation of the big data ecosystem (including aspects such as system performance), but rather seeks to secure the environment during its operation phase. Normally, this type of action is carried out through an incident response process. In this section, the main incident response standards are explained. It is also important to recognize that big data ecosystems are often very complex, with a large number of technologies available for the implementation of different requirements. In order to make our proposal useful for any type of organization in any context, it is necessary to have a global perspective of what defines a big data ecosystem. In that sense, the reference architectures (RAs), and more specifically the security reference architectures (SRAs), have proven to be valuable guides in defining different environments [27,28].

The components specified by the SRA will be used in our proposal to guide us in covering all the characteristics of a big data ecosystem.

Blockchain is a fault-tolerant, peer-to-peer node network that operates on the principle of transactional agreement and with no central entity. For this operation to work, all parties validate each other's transactions using an agreed consensus algorithm and aggregate them into blocks, building the structure of the block chain [29]. The uses of blockchain technology are varied, and range from the creation of smart contracts to supply chain auditing. In addition, the integration of blockchain technology in big data ecosystems allows there to be an increase in the security and value of the data in the system. This is because the data it handles are more reliable and valuable, as they cannot be accessed, modified, or eliminated without the operation being stored in blockchain; traceability of the operations performed is therefore possible [30]. In comparison with traditional technologies, such as logs (even if implemented as append-only) or relational databases, the main difference between their use and blockchain lies in the addition of new capabilities beyond data storage. It enables at the protocol level the use of cryptography or smart self-executable contracts that help ensure the immutability of digital assets and traceability of data. In addition, the use of consensus algorithms can secure data integrity for all nodes participating in the distributed network [31].

There are several proposals from the scientific community that use blockchain technologies to improve the security of distributed systems in various ways. These include maintaining digital evidence of operations carried out in a system [32] to improve the security of their communications [33], in the effort to preserve personal data from unauthorized access [34]. It is not only in big data ecosystems that blockchain technology has been seen as beneficial; the use of blockchain systems has also proven itself to be effective in improving the security of other kinds of ecosystems such as IoT (Internet of Things) [35] or fog computing environments [36].

Incident response is usually defined as a plan to respond to a cybersecurity incident in a systematic way. Should the incident be malicious, there is a set of steps which should be taken with the objective of containing, minimizing and learning from the damage [37]. There exist several frameworks that focus on incident response; however, the industry has recognized two as de facto standards: the NIST (National Institute of Standards and Technology) and SANS (incident response processes (https://www.alienvault.com/blogs/security-essentials/incident-response-steps-comparison-guide) The NIST proposal contains guidelines for establishing an effective incident response program; its main purpose, however, is incident management. Moreover, it provides a set of examples (including different scenarios and questions to use in incident response) [38]. The SANS proposal, for its part, seeks to be a baseline for each organization when implementing its own incident response process. To this end, its guide explains each of its phases in depth [39]. There are no major differences between the two proposals beyond nomenclature changes or their phases. However, the NIST proposal appears to be more thorough than that of SANS. We will use these proposals as a base on which to build our own incident response process for big data ecosystems.

The proposal presented in this paper represents a new iteration within our line of research. The following subsections explain our SRA proposal which serves as a framework to abstract the main components that form a big data ecosystem and is used as a basis to support the incident response process. In addition, the process presented in this paper is based on ensuring the operation phase of a big data environment, but previously we defined the processes needed for its analysis, design and implementation. It is important to know these processes, since some of the artifacts generated are used in this proposal.

3.1. Security Reference Architecture for Big Data Ecosystems

The term big data refers to an environment that allows the analysis and management of large amounts of data, which can be generated in almost real time and can come from different data sources with different formats; these are the main differences with traditional data processing techniques [40]. In this type of environment, these traditional techniques are not capable to properly handle unstructured

data or high volumes of real-time datasets, but they could be part of the big data ecosystem; for example, both relational and non-relational databases do not provide analytical services they can be used as storage for different kinds of data. However, not all scenarios are suitable for using big data technologies, a deep understanding of the project requirements and the business context is needed to determine if big data is the most appropriate solution. This is crucial since a big data ecosystem is usually a very complex system and its implementation it is not an easy task.

In order to consider all the components of a big data ecosystem it is necessary to use abstraction. This being so, we have developed an SRA for big data ecosystems [15]. This SRA defines the main components of a big data environment, and highlights the importance of using security patterns as a way to handle threats and vulnerabilities. Our SRA is based on NIST's Reference Architecture for Big Data, which has received the general consensus of the industry and scientific community [41]. However, NIST's proposal is too abstract, not placing enough emphasis in how the components are connected to each other. Figure 1 summarizes the main components of our SRA and how they are connected to each other; it also highlights the importance of defining security requirements in the early stages of the development. The monitor system is the novelty we want to add to the reference architecture as a way to improve its security during the operation phase, however, it does not belong to the SRA as such; this is the main objective of this paper. The definition of the incident response process that covers the operation phase, as well as its possible implementation will be shown in the following sections.

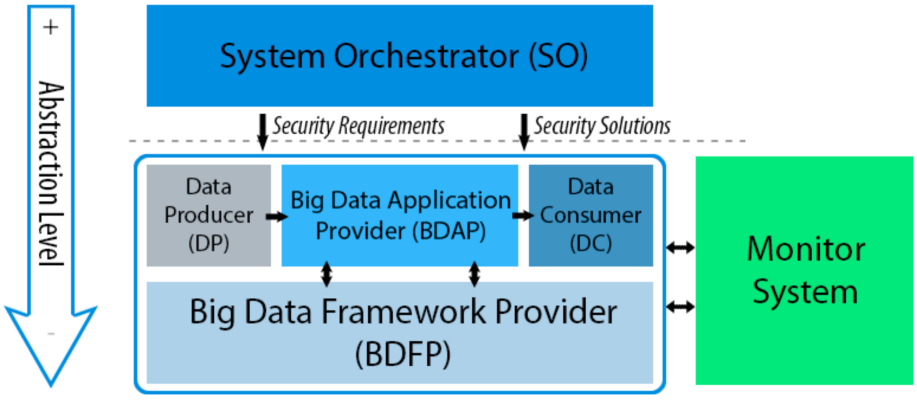

Figure 1. Main components of the security reference architectures (SRAs) for big data ecosystems.

The system orchestrator (SO) focuses primarily on the definition of the requirements and security solutions that will be implemented in the other components of the SRA. The next component is the big data application provider (BDAP), which is composed of the different services offered by big data (collection, preparation, analysis, visualization and access control). These services are implemented in the next component: the big data framework provider (BDFP), which can be considered as the hardware layer of the big data ecosystem. The BDFP supports the functionalities of the BDAP. To do that, it is usually composed of one or more clusters which are in turn composed of nodes. In addition to the hardware infrastructure, this component provides storage, processing, and other services such as communications or resource management. Finally, the last two components of the SRA are the data producer (DP) and the data consumer (DC), which have a similar function, but at opposite edges of the architecture. On the one hand, the DP is responsible for feeding data to the big data ecosystem, serving as a connection point with data sources, which can be both structured and unstructured. The DC, for its part, is the component that consumes the information generated by the big data ecosystem, serving as a connection point with the end user of the data. In this kind of ecosystem, the data processed usually contains personal data, so all actions performed in these components should be logged in order to improve the control over the ecosystem.

3.2. Process for the Development of a Secure Big Data Ecosystem

The creation of a secure big data ecosystem is usually a very complex task that should be supported by guidance and methodologies if its success is to be guaranteed. For that reason, we have defined a process that integrates security aspects into the development of a big data ecosystem; at the same time, it considers the inherent characteristics of that kind of system [9]. Our proposal is composed of eleven different phases covering the main stages of development, including analysis and design, which are not normally considered sufficiently in this kind of scenarios. Moreover, it is important to highlight that a process of this kind should be more than a mere a description of a set of activities; it should in fact be supported by a conceptual framework that defines the main components of the system under development [42]. In our case, we needed a metamodel that covers the main components of a big data ecosystem and which at the same time incorporates security aspects into those components: the security reference architecture (SRA) for big data, defined in the previous section. The process follows the recommendations from the security-by-design approach [43,44] by taking security into consideration from the early stages of the process and by including security aspects during the whole process.

Our process was originally composed of two different set of phases: on the one hand, analysis and design, and on the other hand, implementation. The initial phases focus primarily on the definition of requirements, security solutions and risk analysis, all of which will guide the implementation of the big data ecosystem in the second set of phases. Our process is intended to be performed iteratively, so the two sets of phases are closely related to each other. The artifacts obtained from the first set of phases are thus not definitive, but will be refined when needs are discovered with an adequate level of detail in the implementation.

In addition to this first overview of the process, in this paper we include a new process that covers the operation of the big data ecosystem once it is implemented in the organization. Figure 2 depicts our proposal to cover a secure lifecycle of a big data ecosystem, including the different sets of phases and how they relate to each other. Moreover, if new security requirements are discovered during the operation phase, these requirements must be analyzed in a new iteration of the cycle (Analysis and Design and Implementation). The concepts and components used in the previous stages will therefore be crucial in gaining a better understanding of the needs of the big data ecosystem that has been implemented and in checking the fulfillment of the requirements defined previously.

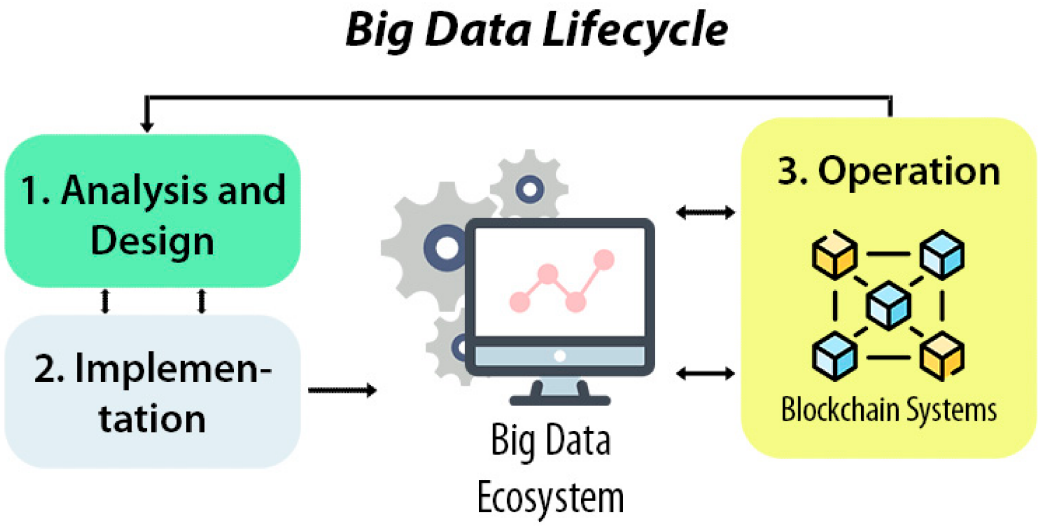

Figure 2. A lifecycle for a big data ecosystem.

4. Incident Response for Big Data Ecosystems

The objective of this new set of phases is to cover the lifecycle of a big data ecosystem, from its conception to its operation. The events collected and detected will be used to carry out the maintenance

and continuous improvement of the system. This operation process is composed of five phases, which in turn are made up of different activities. Figure 3 displays the complete process from the analysis and design to the operation process that we present in this paper. The inputs of the activities may come from the analysis, design and implementation phases defined in [9], or they may be the result of another activity defined in this process. Table 2 serves as a kind of guide to the reader, summarizing the different inputs that will be used by the phases of this process, as well as their origin, such as information about the company, or an output during the previous phases of analysis, design or implementation. In the following subsections, each phase will be described. Our process for the operation phase is based mainly on the proposals made by NIST and SANS, as described in Section 2, but adapted to a big data ecosystem.

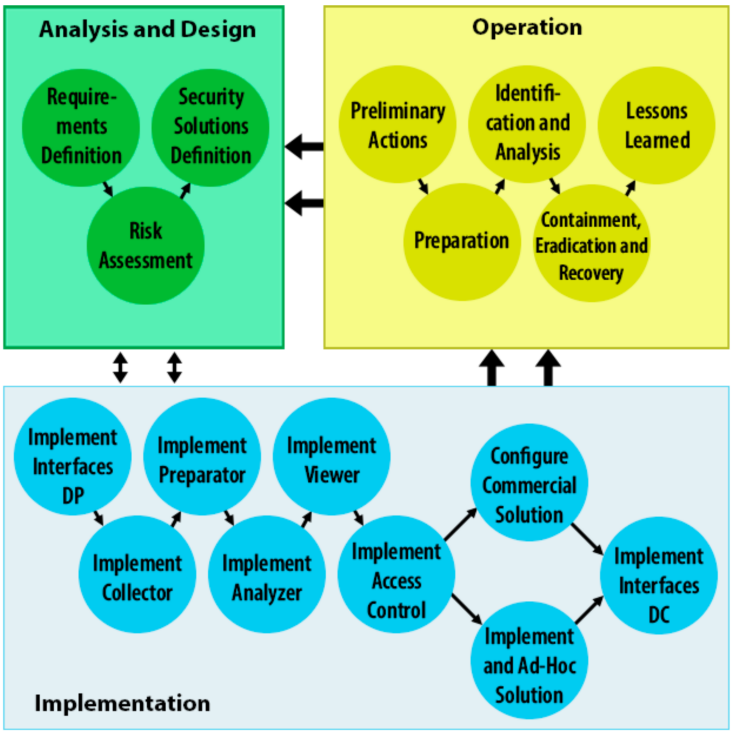

Figure 3. Processes and phases to ensure the security of big data.

Table 2. Inputs obtained previously and needed for the process.

Needed Input	Description	Origin
Security requirements	List of security requirements that were implemented in the big data ecosystem.	Process: Analysis and design Phase: Requirements definition Activity: Define requirements
Big data assets	List of big data ecosystem assets. They were selected according to the needs of the environment.	Process: Analysis and design Phase: Requirements definition Activity: Acquire assets
Repository with the prioritized risks	Prioritized risk list covering all identified threats to the big data ecosystem.	Process: Analysis and design Phase: Risk assessment Activity: Risk assessment
Repository with the security policies of the company	List with the company's security policies. They usually have a high level of abstraction that must be developed into more concrete artifacts.	Knowledge from the organization
Regulations	Set of regulations that may affect the operation of the big data ecosystem.	Knowledge from the organization
Repository with the business policies	List with the company's business policies. They express a set of restrictions and guidelines for the company to achieve its objectives.	Knowledge from the organization

4.1. Preliminary Actions

The main goal of this preliminary phase is to establish the objectives of the incident response plan, as well as its scope. To that end, there are two activities, which are summarized in Table 3.

Table 3. Activities of Phase 1: preliminary actions.

Activities	Description	Inputs	Outputs
Definition of basic concepts	Define what an event is, and what an incident is, in the target environment	Repository with the security policies of the company Security requirements Big data assets Regulations	Event description Incident description
Definition of roles	Define the roles and functions that each role will perform during the incident response process.	Event and incident description Big data assets Repository with the business policies	List of stakeholders and their roles

The first of these activities aims to define what each organization considers to be an event and what it considers an incident, i.e., to clarify the differences between them. In any IT system an event is normally any occurrence within normal behavior and anticipated in the requirements; for example, in a big data ecosystem an event would be any action carried out on the data, including its analysis, access or visualization. But it also includes the system errors that do not result in the operation's being halted. On the other hand, an incident can be defined as any malicious attack that has a nefarious purpose and which is against the security policies of the organization. For example, an illegal reading would be an event at first, but it is actually an incident. Obviously, incidents must be stored in the blockchain environment, so that they can be used as audit evidence in a legal process, or to carry out an investigation into the causes of the incidents. Moreover, events should be stored too, since they can be useful later gain a better understanding of what led to the incident; it may even occur that a "normal" event was not correctly tagged in the first place. Events are important in recovery from errors.

The purpose of the second activity is to define the different roles that will act in the incident response process. There are some typical roles in this type of activity, such as the incident response team, which aims to perform the tasks needed when carrying out this activity, including collecting evidence, studying incidents and reporting when certain types of incidents occur. Depending on the size of the organization, there will be one or more incident response teams, with their own respective focus on different parts of the ecosystem. Moreover, there are other types of common security roles that should play an important part in this process, such as the CISO (chief information security officer), the CTO (chief technology officer) and the CSO (chief security officer) who should be involved in the high-level decisions that are made, especially when the incident is severe. In addition, the rest of the stakeholders that interact with the big data ecosystem need to be involved in the incident response process, since they can be the first to detect that something is going wrong with the system. Typical stakeholders in a big data ecosystem are data scientists, big data architects, and data engineers.

4.2. Preparation

This second phase focuses on defining the concepts that are necessary for the preparation and prevention of incidents. To this end, one activity is carried out, which is summarized in Table 4. This activity should be conducted mainly by the incident response team, but senior management (and more specifically the CISO, CTO and CSO) should be aware of the decisions made at this stage. This phase will determine how prepared the team is to respond to an incident. The main standards for incident response normally define two activities in this phase, one of which is related to ensuring ecosystem security controls. However, in our case, the security mechanisms are implemented at the

same time as the big data ecosystem. In addition, we added a new activity that focuses on defining the different indicators that should be considered when storing the events.

Table 4. Activities of Phase 2: preparation.

Activities	Description	Inputs	Outputs
Preparing to handle incidents	Plan the different actions to be taken when an incident occurs	Event description Incident description Repository with the prioritized risks List of big data assets List of stakeholders and their roles	Communication plan RACI (responsible, accountable, consulted and informed) matrix for incidents List of incident analysis resources
Definition of indicators	Identify all big data resources that may indicate a security incident is occurring	List of big data assets Repository with the prioritized risks	List of sources for indicators

The first activity of this phase focuses on defining a set of actions and guidelines that must be taken when an incident happens. This activity is crucial in ensuring the effective implementation of the incident response plan, as it includes the necessary steps for responding to each type of incident and asset type. When we talk about a big data ecosystem, we have to take into account the types of assets specific to these environments. According to the NIST organization, the main types of big data assets are: the hardware infrastructure that supports big data functions, the services and applications provided by the ecosystem, the analytical resources, such as MapReduce algorithms, the data used to perform the analyses, the insights obtained, the security mechanisms to protect the privacy of the data, as well as the individuals and their roles that interact with the system [41]. As a result of this activity, there are at least three artifacts that should be generated: the communication plan, which must include the contact information for all the team members, along with the protocol for establishing the connection in case of an incident, a RACI matrix (responsible, accountable, consulted and informed) for each of the kinds of incident, in order to understand the functions of each of the roles better, and finally a list of the different analytic resources (hardware and software) that can be used in this kind of situation.

The second activity, for its part, tries to define all the indicators that can give information on the causes of the incident. For it do so, all the assets of the big data ecosystem must be studied, including their communications and the data they use and generate. The resulting list of indicators will be used in the next phase to study and detect unusual incidents or situations. A good selection of indicators can facilitate significantly the task of responding to incidents.

4.3. Identification and Analysis

This is the longest phase, since it is developed throughout the entire operation phase of the big data environment. This phase pursues the objective of the detection and later analysis of the different incidents that occur in the ecosystem. All these activities are carried out sequentially, except for the fourth activity on the storage of events and incidents in blockchain, which is carried performed in parallel throughout this phase, the activities of which are summarized in Table 5.

Table 5. Activities of Phase 3: identification and analysis.

Activities	Description	Inputs	Outputs
Incident analysis	Detection and analysis of possible incidents that may occur in the environment	List of sources for indicators Documentation from previous casesBlockchain systems	Incident information
Incident prioritization	Prioritize incidents to determine necessary recovery actions	Incident Documentation from previous cases	Incident information updated
Incident communication	Communication of the incident to the necessary stakeholders	Incident information Communication plan	Stakeholder reported about the problem
Blockchain storage	Store all evidence related to the incident in the incident blockchain system. Prior to this, events that occurred in the big data ecosystem must be stored in the event blockchain	List of sources for indicators Incident information updated	New block in the incident blockchain New block in the event blockchain

The first activity is fundamental throughout the incident response process, since it is a matter of analyzing incidents to determine whether they have actually occurred, i.e., this is about detecting them. In order to carry out this activity, an analysis must be made of the indicators defined in the previous phase. This study must be conducted by the incident response team, since there may be false positives, in which an event is labelled as an incident when it is really a simple system error. To avoid these false positives, previous cases can be used to understand the behavior of the big data ecosystem better. In this type of environment, it is especially critical to emphasize the analysis of the typical functions that are performed during the operation, i.e., the preparation, analysis and visualization of the data. An intelligence system can be implemented with a view to carrying out that task. This system can use the big data ecosystem itself as an engine to perform the machine learning techniques, or it can employ another isolated system. The use of big data analytics can help the incident response team to identify patterns and threats in a more effective way. As an output of this activity, the incidents that have definitely been identified as such will be obtained.

The second activity consists of categorizing the incident according to its severity. It is therefore not a good policy to treat incidents on a first-come, first-served basis. In addition to the impact made by the incident, another factor to take into account is the recoverability of the components affected. This activity is closely connected to the phase of risk assessment in the analysis and design process (see Figure 3), since this incident should be the materialization of a threat. However, it is possible that the threat was not anticipated previously; this implies the need to update the list of risks. Once the incident has been analyzed and prioritized, and based on the communications plan defined during the preparation phase, the third activity informs the responsible stakeholders so that they can carry out the necessary actions.

The last activity of this phase deals with the storing of the event and incident data in the blockchain system. Our proposal has two blockchain systems running in parallel. One of them is in charge of storing the data generated by the defined indicators (event blockchain). Hence, this blockchain is connected to the main components of the big data ecosystem: the big data application provider (BDAP) and the big data framework provider (BDFP). This first blockchain system should store all the data related to the actions performed on the data, i.e., the typical big data services: collection, preparation, analysis, visualization and access control to the data. Each block of this system should have a similar structure: ID, user ID (the user that performed the operation), role of the user, timestamp, type of

operation performed, and the data affected by the operation. Each operation has its own characteristics that must be taken into account, however; for example, the collection can also store the information about the data source from which the data is being stored in the big data, or the visualization should consider that it is possible to infer sensitive information from anonymized data by performing a large number of queries on the same dataset. Figure 4 shows a UML (Unified Modeling Language) diagram with a possible implementation of the event blockchain, displaying how it is connected to the big data ecosystem. As it can be seen in the figure, each blockchain system is composed of a set of blocks and operations. In turn, each block contains the information of an operation performed on the data.

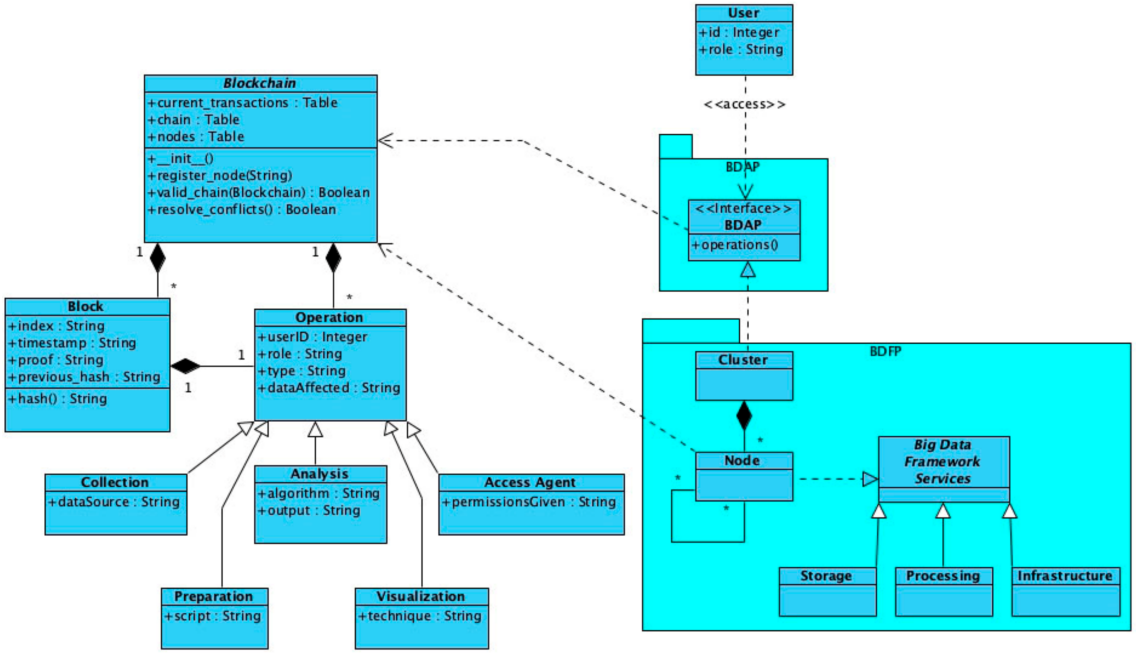

Figure 4. Interaction between big data ecosystems and blockchain systems.

In parallel, a second blockchain system stores all the data related to the incidents already identified (incident blockchain), and order to do that, this blockchain is connected to a component that monitors the compliance of the requirements of the big data ecosystem. This means that when an incident occurs it is detected and stored in the incident blockchain. It is, however, necessary to store more incident data that could help in the recovery of the system. To that end, when an incident is identified, all event data that may be related to it will be copied to this second system by using a proxy between both blockchain systems. Once all the data are collected in the blockchain systems, it is time to carry out the analysis of them. This implies the need to implement an intelligent system that employs machine learning techniques to obtain value from these data, to identify the reasons why the incident happened, for example, or to try to predict the occurrence of a new incident by analyzing the events that happen in real time. Moreover, all this data should be visualized by means of a dashboard that is checked by the incident response team. Figure 5 depicts the interaction between the big data ecosystem and the blockchain systems. This figure shows a summarized version of the different components of the SRA for big data ecosystems defined in [15]. As explained previously, this activity is not carried out sequentially, but rather in parallel to the rest of the activities of this phase.

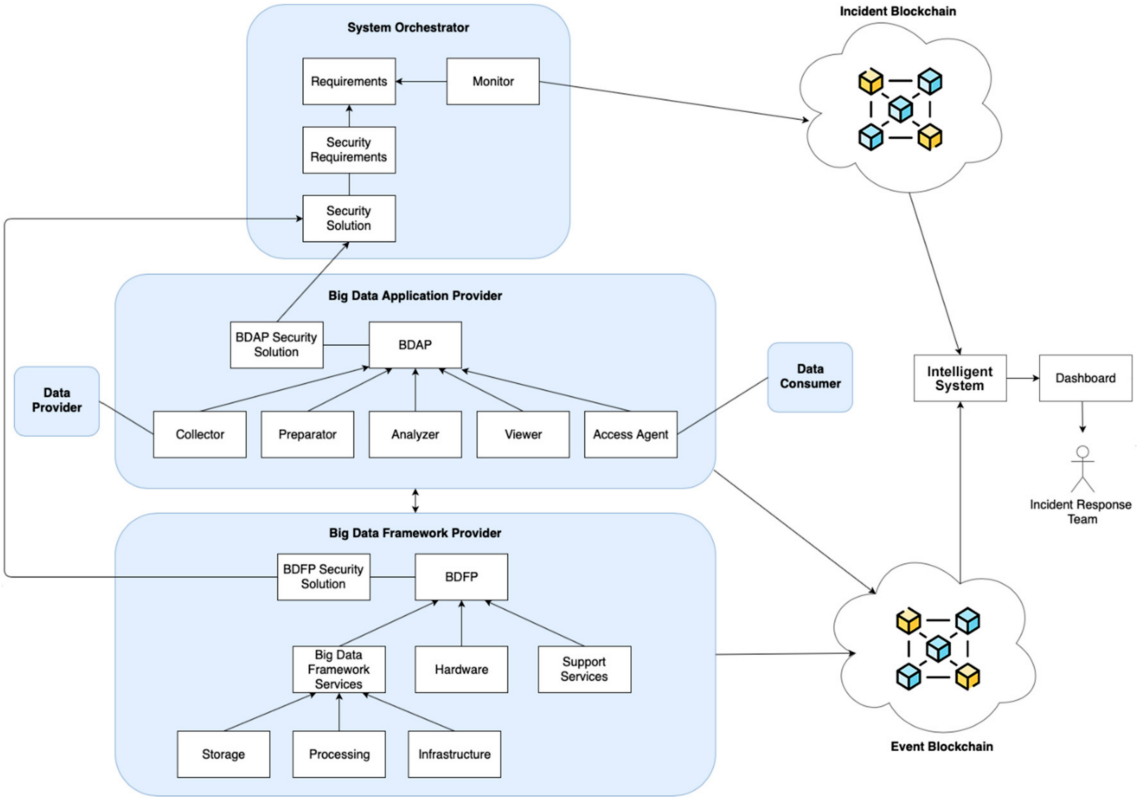

Figure 5. Interaction between big data ecosystems and blockchain systems.

Since our process aims to have a high level of abstraction so that it can be applicable to any scenario, in this section we do not provide details on specific technologies for its implementation. However, wishing to help improve understanding of the implementation of these systems, and to show how the components relate to each other, we explain a possible implementation of them in Section 5. But it is important to note that this is only a context-specific proposal; each scenario must consider its own requirements to find the solution that fits it best.

4.4. Containment, Eradication and Recovery

This phase covers activities aimed at returning the big data ecosystem to a normal state. These activities range from the first actions taken to try to limit the damage caused by the incident right through until the system is considered to have recovered fully. Table 6 contains all the activities, inputs, and outputs of this phase.

The first activity is short-term containment, which aims to limit the impact caused by the incident and prevent further damage to the system. It tries to apply simple but effective actions that can be applied at that moment in time. Examples of this type of action are those involving disconnecting servers or totally restricting access to sensitive data that may be affected. The next activity then consists of carrying out a check on the environmental assets that may have been affected by the incident. In the case of a big data ecosystem, this includes checking the system nodes, the analysis and preparation algorithms, the security mechanisms already implemented and the data.

The activity that takes place next focuses on implementing actions to prevent the incident from happening again. This requires an in-depth investigation of the causes of the incident: What controls have failed? How long did it take to respond to the incident? Could it have been avoided? This research will be conducted using the evidence stored in the blockchain systems. As in the incident analysis phase, big data's analytical capabilities can be used to assist in the study of the data. In the quest to do just that, big data can perform diagnostic analytics based on the event and incident data stored in the

blockchain systems. The types of actions that can be applied range from installing security patches on software that had some vulnerability to implementing new security mechanisms, or even expelling malicious users. While this activity is being performed, the system should be operating normally.

Table 6. Activities of Phase 4: Containment, eradication and recovery.

Activities	Description	Inputs	Outputs
Short term containment	Limit the impact of the incident by carrying out effective and easy actions	Incident information blockchain systems	List of primary actions taken
System check	Check the general state of the different assets affected by the incident	Incident information blockchain systems	Status of the big data ecosystem
Long term containment	Implement definitive solutions that allow the continued use of the ecosystem	Incident information blockchain systems List of primary actions taken	List of actions taken
Recovery	Check the normal execution of the ecosystem, with greater emphasis on the components affected	Incident information blockchain systems List of actions taken	Status of the big data ecosystem
Visualization of the status of the big data ecosystem	Real-time monitoring of the operation of the components affected	Incident information Blockchain systems List of sources for indicators	Status of the big data ecosystem

The fourth activity focuses on checking that, after applying the necessary actions to recover from the incident, the system works as it was designed to. Greater emphasis needs to be placed on those components of the big data ecosystem that were affected by the incident. The final activity can be considered an auxiliary activity to the previous one, since it consists of making a real-time visualization of the behavior of the affected components. To that end, a dashboard can be produced to show the data stored by the indicators defined in the second phase of our process.

4.5. Lessons Learned

Finally, the last phase is about documenting the incident, writing down lessons learned; the objective is to obtain benefits from the incident. That is not the only goal, since at this point it is possible that new security requirements will be discovered that were not considered at first and that must be implemented in the system. If this scenario occurs, the analysis, design, and implementation phases defined previously must be performed again [9]. This phase is composed of two activities, summarized in Table 7.

Table 7. Activities of Phase 5: lessons learned.

Activities	Description	Inputs	Outputs
Write documentation	Complete all the documentation about the incident in order to benefit from this in future incidents	Blockchain systems Incident information List of actions taken	Lessons learned
Continuous improvement	Use the lessons learned to keep improving the security of the ecosystem	Lessons learned	New security requirements

The first activity of this last phase consists of writing all the documentation that has not already been written up during the incident. This phase should not be taken lightly, since a good writing-up of the lessons learned can be very beneficial, helping to avoid the repetition of this type of incidents. Other new incidents could even be discovered, ones which may have taken advantage of a hole in the security of the system. When carrying out this activity it is advisable to hold a meeting between all the stakeholders who have been involved in one way or another during the incident. Furthermore, it is also desirable to use the incident data stored in the Blockchain systems to obtain a greater understanding of the incident itself.

The second activity consists of using the lessons learned to try to improve the security of the system. In many cases, new security requirements may be identified that should be implemented in the system; to do that, it will be necessary to go back to the beginning of the cycle, as is shown in Figure 2.

5. Integration of Blockchain Systems into a Big Data Ecosystem: A Case Study

In an effort to assist the reader to understand our proposal in greater depth, as well as to provide clarity as regards the more technological details of its implementation, this section presents a case study that implements a blockchain solution. This case study should be considered as an example to demonstrate the feasibility of our proposal. This example is applied to an experimental hydroponic crop that we have designed, implemented and automatized. This type of cultivation has become very popular in recent times, due to its reduced demand for water, the use of nutrients instead of soil to ensure its growth, and the smaller surface area needed to produce benefits. The problem that is presented, however, is that these kinds of cultivation require greater attention on the part of the producer (https://www.weforum.org/agenda/2019/02/hydroponics-future-of-farming/). To solve that problem, our research group created a prototype solution based on cyber-physical systems (CPS).

This CPS system has a series of sensors and actuators that in turn are controlled by a big data ecosystem. This big data ecosystem makes the necessary decisions as to when to activate the actuators, depending on the readings of the sensors. For example, it controls the optimal temperature for plant growth or the nutrients needed at any given time. These sensors are very varied, and the analysis of the data they generate must be performed in near real time in order to trigger the actuators; it was thus decided to use big data technologies to control this.

This system is not beyond the danger of cyberattacks; in fact, it is especially sensitive to them, since the modification or masking of the values or readings produced by the sensors could lead to the loss of all plants. This being the case, we decided to incorporate our blockchain solution to control the production phase of this environment. Figure 6 shows the global architecture of the implemented system, including the blockchain-based systems that store the different events and incidents that occur in the CPS system.

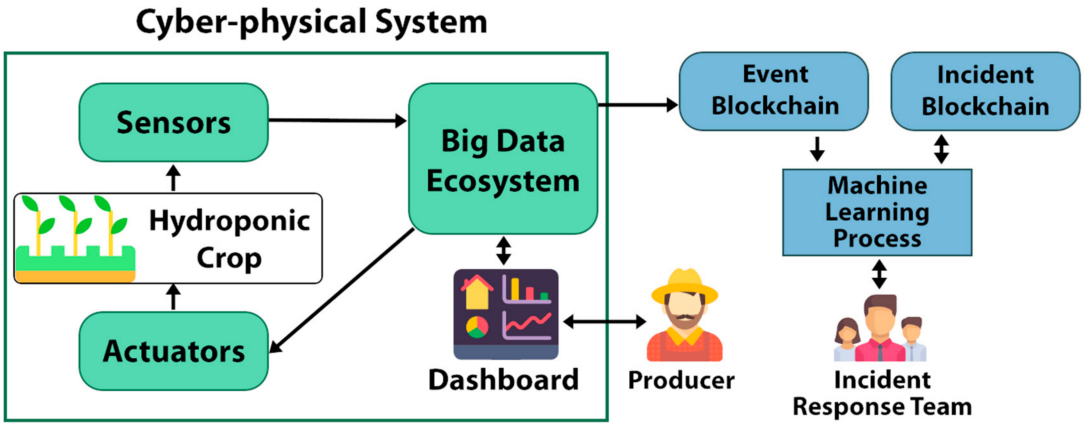

Figure 6. General architecture of the case study.

As can be seen in the figure, the hydroponic crop has a number of sensors that generate a flow of data that feeds the big data ecosystem. Depending on the data analyzed, the different actuators are activated or deactivated; they can also be activated manually by means of a control panel. In addition, any action that occurs in the big data ecosystem is stored as an event in the event blockchain. This includes all actions ranging from the actions performed by the actuators to the operations performed on the data set.

Once you have a collection of events, a machine learning process is applied, consisting of the following phases: first, there is a detection of outliers using the PCA anomaly detection algorithm, then the incident response team tags the results from the previous step, indicating whether it is an incident or not. This outcome is subsequently used as a way to train a neural network, so that after a series of iterations, it is able to identify when an incident has occurred automatically and without human intervention. The choice of a neural network is because of its fault tolerance, since in this type of scenario it is common to label a security incident incorrectly; it is also chosen for its ability to work in real time, which makes it a good tool for incident detection during the operational phase. These algorithms are performed by using the TensorFlow tool. However, this case study is under development and we do not have enough data to obtain relevant results although, as stated on the related work section, there are successful proposals that apply these techniques to discover security problems.

Regarding the technologies used to implement both the big data ecosystem and the blockchain systems, a summary scheme is shown in Figure 7. The big data ecosystem is implemented mainly on an Apache Hadoop environment. Sensor data is thus stored in the Hadoop HDFS (Hadoop Distributed File System), which is used by HBase, to make for easier access and to perform the different queries. As far as data analysis is concerned, this is performed using Apache Spark, since it must be executed in near real-time. The results obtained from the execution of the analysis algorithms are displayed on a dashboard implemented using the Dash by Plotly tool. The Apache Hadoop YARN tool is used to coordinate the jobs and manage the resources of the different nodes of the big data ecosystem.

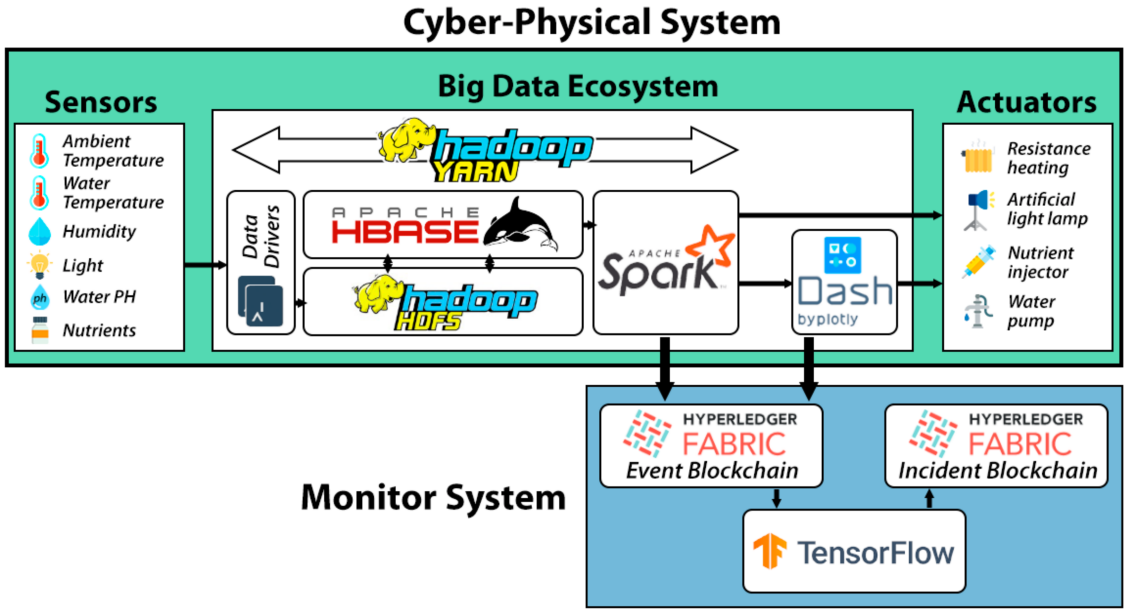

Figure 7. Technological instantiation.

It is important to highlight the fact that the implementation of the Blockchain System should not place an extra computational burden on the CPS system since it is separated from it. On the other hand, when executing the machine learning algorithms there are two possibilities: that they are executed within the same big data ecosystem that supports the CPS system or that they are executed in a new

system. Checking the performance that these two possibilities could have is out of the scope of the main objective of the paper, and therefore, it is considered as a future work task.

It should also be mentioned that to implement blockchain systems it was decided to make use of Hyperledger Fabric technology. This technology allows us to create a private blockchain network composed of a set of nodes authorized for the introduction of events and incidents that happen in the CPS environment. When establishing a consensus mechanism, we decided in our case to make use of proof of authority, since only authorized network nodes can introduce new blocks and transactions in the chain. In Hyperledger, the data transactions that are added to the blocks are entered in JSON format. By way of example, Figure 8 shows a simplified example of the Smart Contract used to create events when there is any action performed by an actuator of the hydroponic crop: (a) automatically, due to a decision of the big data environment, or (b) by a user registered in the system through the dashboard.

```
class Events extends Contract {

    async initLedger(ctx) {
        console.info('============= Ledger Initialized ===========');
    }

    async createEvent(ctx, idEvent, idActuator, actionType, actionDescription, actionTrigger, timestamp) {
        console.info('============= START : Create Event ===========');

        const event = {
            idEvent,
            idActuator,
            actionType,
            actionDescription,
            actionTrigger,
            timestamp,
            docType: 'event',
        };

        await ctx.stub.putState(idEvent, Buffer.from(JSON.stringify(event)));
        console.info('============= END : Create Event ===========');
    }

    async getEvents(ctx, begin = 0, end = 0) { ...
    }

     async getAllEventsByType(ctx, actionType) {
        const startEvent = '0';
        const endEvent = '99999';
        const allResults = [];

        for await (const {key, value} of ctx.stub.getStateByRange(startEvent, endEvent)) {
            const strValue = Buffer.from(value).toString('utf8');
            let event;
            try {
                event = JSON.parse(event);
                if (event.getActionType() == actionType){
                    allResults.push({ Key: key, Event: event });
                }
            } catch (err) {
                console.log(err);
                event = strValue;
            }
        }

        console.info(allResults);
        return JSON.stringify(allResults);
    }

    async getAllEventsByTrigger(ctx, actionTrigger) { ...
    }

}

module.exports = FabEvent;
```

Figure 8. Smart contract to handle events.

Figure 8 shows how a new event is created in the system by including its main characteristics, such as the type of action, or what triggered the event. In addition, methods are included for obtaining a specific event through its ID, the events that happened in a time interval (if no interval is introduced, all the stored events will be obtained), the events that happened of a specific type, and the events stored according to the type of trigger that originated them (big data or user). These functions allow the incident response team to perform analytical studies and apply the machine learning process to detect incidents that are later stored in the incident blockchain following a similar Smart Contract format.

6. Conclusions and Future Work

A big data ecosystem is usually a very complex environment, therefore, to try to manage its operation it is advisable to have a process based on the main best practices in the industry and that is sufficiently abstract to be applicable to any scenario. Hence, the main contribution of this paper is the definition of an incident response process that is specific to big data ecosystems. This process consists of five different phases which in turn are divided into different activities with inputs and outputs. For the implementation of the process, we propose the use of blockchain technology; this allows the improvement of the security of the data stored by increasing its immutability and traceability. Thanks to these characteristics, the records stored can be used as forensic evidence. In addition, these stored records can be analyzed through the use of different machine learning models. This analysis aims to discover different attack patterns, aiming to achieve early detection and prediction of security incidents.

This research is part of a long research project, since it is supported by an SRA that acts as a metamodel of the different components that normally make up a big data ecosystem. In addition, this process integrates and evolves our proposal for the development process of secure big data ecosystems.

In this paper, and as a first step in the validation of our proposal, we include a case study on a CPS system, which is automated by a big data ecosystem based on Apache Spark. This system has been connected to a private blockchain network, implemented by Hyperledger Fabric, that stores all the events that occur in the system for later analysis through a machine learning process.

This case study is still in a stage of development, so as future work, we aim to implement the machine learning process that we present, as well as test the system by performing different attack scenario on the system. These attacks will be stored along with the data generated by the big data ecosystem for later analysis. After several iterations we expect to obtain the optimal parameterization of the algorithms. Moreover, it would be necessary to verify the computational cost increase involved in adding a blockchain system and its scalability in larger projects. In addition, the process will continue to be validated through its application in different big data and attack scenarios. Finally, we are considering creating a process to manage the end-of-life of a big data ecosystem.

Author Contributions: Conceptualization, J.M. and E.B.F.; methodology, M.A.S.; validation, M.A.S. and E.F.-M.; formal analysis, J.M.; investigation, J.M.; writing—original draft preparation, J.M.; supervision, M.A.S., E.F.-M. and E.B.F.; project administration, M.A.S. and E.F.-M.; funding acquisition, E.F.-M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the ECLIPSE project (RTI2018-094283-B-C31 funded by "Ministerio de Economía y Competitividad and the Fondo Europeo de Desarrollo Regional FEDER") and the GENESIS project (SBPLY-17-180501-000202 funded by "Consejería de Educación, Cultura y Deportes de la Dirección General de Universidades, Investigación e Innovación de la JCCM") and the Programa Operativo Regional FEDER 2014/2020.

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

References

1. Tamura, Y.; Yamada, S. Reliability Analysis Based on a Jump Diffusion Model with Two Wiener Processes for Cloud Computing with Big Data. *Entropy* **2015**, *17*, 4533–4546. [CrossRef]
2. Demchenko, Y.; de Laat, C.; Membrey, P. Defining architecture components of the Big Data Ecosystem. In Proceedings of the 2014 International Conference on Collaboration Technologies and Systems (CTS), Minneapolis, MN, USA, 19–23 May 2014; pp. 104–112. [CrossRef]
3. Rao, T.R.; Mitra, P.; Bhatt, R.; Goswami, A. The big data system, components, tools, and technologies: A survey. *Knowl. Inf. Syst.* **2019**, *60*, 1165–1245. [CrossRef]
4. Wang, H.; Jiang, X.; Kambourakis, G. Special issue on Security, Privacy and Trust in network-based Big Data. *Inf. Sci.* **2015**, *318*, 48–50. [CrossRef]
5. Eyupoglu, C.; Aydin, M.A.; Zaim, A.H.; Sertbas, A. An Efficient Big Data Anonymization Algorithm Based on Chaos and Perturbation Techniques. *Entropy* **2018**, *20*, 373. [CrossRef]
6. Sharma, P.P.; Navdeti, C.P. Securing big data hadoop: A review of security issues, threats and solution. *Int. J. Comput. Sci. Inf. Technol* **2014**, *5*, 2126–2131.

7. Moreno, J.; Serrano, M.A.; Fernández-Medina, E. Main Issues in Big Data Security. *Future Internet* **2016**, *8*, 44. [CrossRef]
8. Carrasco, J.; Durán, F.; Pimentel, E. Trans-cloud: CAMP/TOSCA-based bidimensional cross-cloud. *Comput. Stand. Interfaces* **2018**, *58*, 167–179. [CrossRef]
9. Moreno, J.; Fernandez, E.B.; Serrano, M.A.; Fernández-Medina, E. Secure Development of Big Data Ecosystems. *IEEE Access* **2019**, *7*, 96604–96619. [CrossRef]
10. Sahebjamnia, N.; Torabi, S.A.; Mansouri, S.A. Integrated business continuity and disaster recovery planning: Towards organizational resilience. *Eur. J. Oper. Res.* **2015**, *242*, 261–273. [CrossRef]
11. Massie, M.L.; Chun, B.N.; Culler, D.E. The ganglia distributed monitoring system: Design, implementation, and experience. *Parallel Comput.* **2004**, *30*, 817–840. [CrossRef]
12. Communications-Electronics Security Group. *Digital Continuity to Support Forensic Readiness*; The National Archives: Richmond, UK, 2011.
13. NIST. Blockchain Technology Overview. Available online: https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8202.pdf (accessed on 18 October 2017).
14. Wang, Y.; Kogan, A. Designing confidentiality-preserving Blockchain-based transaction processing systems. *Int. J. Account. Inf. Syst.* **2018**, *30*, 1–18. [CrossRef]
15. Liu, Q.; Li, P.; Zhao, W.; Cai, W.; Yu, S.; Leung, V.C.M. A survey on security threats and defensive techniques of machine learning: A data driven view. *IEEE Access* **2018**, *6*, 12103–12117. [CrossRef]
16. Moreno, J.; Serrano, M.A.; Fernandez-Medina, E.; Fernandez, E.B. Towards a security reference architecture for big data. In Proceedings of the DOLAP Workshop Colocated with EDBT/ICDT Conference, Vienna, Austria, 26–29 March 2018.
17. Chen, J.; Lv, Z.; Song, H. Design of personnel big data management system based on blockchain. *Future Gener. Comput. Syst.* **2019**, *101*, 1122–1129. [CrossRef]
18. Uchibeke, U.U.; Schneider, K.A.; Kassani, S.H.; Deters, R. Blockchain Access Control Ecosystem for Big Data Security. In Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 30 July–3 August 2018; pp. 1373–1378. [CrossRef]
19. Shi, M.; Yuan, R. MAD: A monitor system for big data applications. In *Lecture Notes in Computer Science*; Including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics; Springer: Cham, Switzerland, 2015; Volume 9243, pp. 308–315. [CrossRef]
20. Khanna, G.; Varadharajan, P.; Bagchi, S. Automated online monitoring of distributed applications through external monitors. *IEEE Trans. Dependable Secure Comput.* **2006**, *3*, 115–129. [CrossRef]
21. Fetjah, L.; Benzidane, K.; Alloussi, H.E.; Warrak, O.E.; Jai-Andaloussi, S.; Sekkaki, A. Toward a Big Data Architecture for Security Events Analytic. In Proceedings of the 3rd IEEE International Conference on Cyber Security and Cloud Computing, CSCloud, Beijing, China, 25–27 June 2016; pp. 190–197. [CrossRef]
22. Li, T.; Yan, L. SIEM based on big data analysis. In *Lecture Notes in Computer Science*; Including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics; 10602 LNCS; Springer: Cham, Switzerland; Kracow, Poland, 2017; pp. 167–175. [CrossRef]
23. Hassan, M.M.; Gumaei, A.; Alsanad, A.; Alrubaian, M.; Fortino, G. A hybrid deep learning model for efficient intrusion detection in big data environment. *Inf. Sci.* **2020**, *513*, 386–396. [CrossRef]
24. Zolanvari, M.; Teixeira, M.A.; Gupta, L.; Khan, K.M.; Jain, R. Machine Learning-Based Network Vulnerability Analysis of Industrial Internet of Things. *IEEE Internet Things J.* **2019**, *6*, 6822–6834. [CrossRef]
25. Souissi, S.; Serhrouchni, A.; Sliman, L.; Charroux, B. Security incident response: Towards a novel decision-making system. *Adv. Intell. Syst. Comput.* **2017**, *557*, 667–676. [CrossRef]
26. Ibrishimova, M.D.; Li, K.F. Automating incident classification using sentiment analysis and machine learning. In *Lecture Notes in Computer Science*; Including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics; 11317 LNCS; Springer: Cham, Switzerland; Vancouver, BC, Canada, 2018; pp. 50–62. [CrossRef]

27. Veeramachaneni, K.; Arnaldo, I.; Korrapati, V.; Bassias, C.; Li, K. AI^2: Training a Big Data Machine to Defend. In Proceedings of the 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), New York, NY, USA, 9–10 April 2016; pp. 49–54. [CrossRef]
28. Fernandez, E.B.; Monge, R.; Hashizume, K. Building a security reference architecture for cloud systems. *Requir. Eng.* **2016**, *21*, 225–249. [CrossRef]
29. Krco, S.; Pokric, B.; Carrez, F. Designing IoT architecture (s): A European perspective. In Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT), Seoul, Korea, 6–8 March 2014; pp. 79–84.
30. Moubarak, J.; Filiol, E.; Chamoun, M. On blockchain security and relevant attacks. In Proceedings of the 2018 IEEE Middle East and North Africa Communications Conference (MENACOMM), Jounieh, Lebanon, 18–20 April 2018; pp. 1–6. [CrossRef]
31. Vo, H.T.; Mohania, M.; Verma, D.; Mehedy, L. Blockchain-Powered Big Data Analytics Platform. In *Big Data Analytics*; Mondal, A., Gupta, H., Srivastava, J., Reddy, P.K., Somayajulu, D.V.L.N., Eds.; Springer: Berlin/Heidelberg, Germany, 2018; pp. 15–32.
32. Hughes, L.; Dwivedi, Y.K.; Misra, S.K.; Rana, N.P.; Raghavan, V.; Akella, V. Blockchain research, practice and policy: Applications, benefits, limitations, emerging research themes and research agenda. *Int. J. Inf. Manag.* **2019**, *49*, 114–129. [CrossRef]
33. Sutton, A.; Samavi, R. Blockchain enabled privacy audit logs. In Proceedings of the International Semantic Web Conference, Vienna, Austria, 21–25 October 2017; Springer: Cham, Switzerland, 2017; pp. 645–660.
34. Rathee, G.; Sharma, A.; Iqbal, R.; Aloqaily, M.; Jaglan, N.; Kumar, R. A Blockchain Framework for Securing Connected and Autonomous Vehicles. *Sensors* **2019**, *19*, 3165. [CrossRef]
35. Zyskind, G.; Nathan, O.; Pentland, A. Decentralizing Privacy: Using Blockchain to Protect Personal Data. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015; pp. 180–184. [CrossRef]
36. Reyna, A.; Martín, C.; Chen, J.; Soler, E.; Díaz, M. On blockchain and its integration with IoT. Challenges and opportunities. *Future Gener. Comput. Syst.* **2018**, *88*, 173–190. [CrossRef]
37. Tariq, N.; Asim, M.; Al-Obeidat, F.; Zubair Farooqi, M.; Baker, T.; Hammoudeh, M.; Ghafir, I. The Security of Big Data in Fog-Enabled IoT Applications Including Blockchain: A Survey. *Sensors* **2019**, *19*, 1788. [CrossRef] [PubMed]
38. ISO/IEC. *ISO/IEC 27035:2016, Information Technology—Security Techniques—Information Security Incident Management—Part 1: Principles of Incident Management*; ISO/IEC: Geneva, Switzerland, 2016.
39. Cichonski, P.; Millar, T.; Grance, T.; Scarfone, K. Computer security incident handling guide. *NIST Special Publ.* **2012**, *800*, 1–147.
40. Kral, P. *The Incident Handlers Handbook*; SANS Institute: Rockville, MD, USA, 2011.
41. Chen, C.P.; Zhang, C.-Y. Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Inf. Sci.* **2014**, *275*, 314–347. [CrossRef]
42. NBD-WG.; NIST. NIST Big Data Reference Architecture. Available online: https://bigdatawg.nist.gov/_uploadfiles/M0639_v1_9796711131.docx (accessed on 18 October 2017).
43. Uzunov, A.V.; Fernandez, E.B.; Falkner, K. Assessing and improving the quality of security methodologies for distributed systems. *Journal of Software: Evol. Process* **2018**, *30*, e1980. [CrossRef]
44. Casola, V.; De Benedictis, A.; Rak, M.; Rios, E. Security-by-design in Clouds: A Security-SLA Driven Methodology to Build Secure Cloud Applications. *Procedia Comput. Sci.* **2016**, *97*, 53–62. [CrossRef]