Future Generation Computer Systems 102 (2020) 710–722

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

A cloud-edge based data security architecture for sharing and analysing cyber threat information

David W Chadwick[a,*], Wenjun Fan[a], Gianpiero Constantino[b], Rogerio de Lemos[a], Francesco Di Cerbo[c], Ian Herwono[d], Mirko Manea[e], Paolo Mori[b], Ali Sajjad[d], Xiao-Si Wang[d]

[a] University of Kent, Canterbury, CT2 7NF, UK
[b] CNR, Pisa, Italy
[c] SAP Labs, Mougins, France
[d] BT, Ipswich, IP5 3RE, UK
[e] HPE, Cernusco S/N, Italy

ARTICLE INFO

Article history:
Received 14 January 2019
Received in revised form 5 May 2019
Accepted 19 June 2019
Available online 23 August 2019

Keywords:
Data security architecture
Data outsourcing
Cyber threat information
Edge computing
Cloud-edge trust
Cloud security

ABSTRACT

Cyber-attacks affect every aspect of our lives. These attacks have serious consequences, not only for cyber-security, but also for safety, as the cyber and physical worlds are increasingly linked. Providing effective cyber-security requires cooperation and collaboration among all the entities involved. Increasing the amount of cyber threat information (CTI) available for analysis allows better prediction, prevention and mitigation of cyber-attacks. However, organizations are deterred from sharing their CTI over concerns that sensitive and confidential information may be revealed to others. We address this concern by providing a flexible framework that allows the confidential sharing of CTI for analysis between collaborators. We propose a five-level trust model for a cloud-edge based data sharing infrastructure. The data owner can choose an appropriate trust level and CTI data sanitization approach, ranging from plain text, through anonymization/pseudonymization to homomorphic encryption, in order to manipulate the CTI data prior to sharing it for analysis. Furthermore, this sanitization can be performed by either an edge device or by the cloud service provider, depending upon the level of trust the organization has in the latter. We describe our trust model, our cloud-edge infrastructure, and its deployment model, which are designed to satisfy the broadest range of requirements for confidential CTI data sharing. Finally we briefly describe our implementation and the testing that has been carried out so far by four pilot projects that are validating our infrastructure.

1. Introduction

Most organizations today operate one or more security applications, such as firewalls, antivirus software or intrusion detection systems. Each of these systems creates its own log data, containing raw cyber threat information (CTI). CTI is defined quite broadly by the National Institute of Standard and Technology (NIST) as any valuable information that can be used to identify, assess, monitor and respond to cyber threats [1]. Analysis of the CTI produces valuable cyber threat intelligence that informs the user about threats to their systems. Whilst off the shelf security applications contain their own built-in analysis tools, and inform the user about the majority of the threats affecting their systems, they rarely capture all the active threats due to the rapidly evolving threat landscape, the amount of CTI that has to be processed and the sheer complexity of processing this data. Most organizations, particularly small and medium sized enterprises (SMEs), do not have the knowledge, time or resources to analyse the CTI themselves, and either rely on the built-in analysis of the security tools they purchase, or outsource to third party providers that specialize in securing systems and identifying threats. Due to the sensitivity of the CTI, third party providers are invariably constrained to analysing individual organizations' CTIs, and to operating Chinese Walls so that conflicts of interest do not arise, and sensitive information does not leak between its customers. However, the inability to pool the CTI from multiple organizations and to analyse the combined data, means that some threats are bound to be missed.

If the organizations are willing to collaborate and share CTI, the sharing of CTI between collaborating organizations is necessarily complex. On the one hand, an organization may be willing to share its CTI logs with other organizations if this means that remote attacking systems can be more easily identified, but not if

[*] Corresponding author.
E-mail address: d.w.chadwick@kent.ac.uk (D.W. Chadwick).

https://doi.org/10.1016/j.future.2019.06.026
0167-739X/© 2019 Published by Elsevier B.V.

this means that its own vulnerable systems are also identified to its collaborating organizations. It is the CTI sharing problem that our research addresses.

The contributions of this paper are as follows:

– We define a 5-level trust model that allows organizations to determine their level of trust in (a) the cloud infrastructure provider and (b) their peer collaborating organizations,
– We define a data sharing and analysis framework that allows organizations to confidentially share their CTI data with collaborating organizations, dependent upon the level of trust they have in them,
– We specify the various deployment models that are available to cloud instantiations of the framework that allow organizations to regulate the amount of trust they have in the cloud infrastructure provider.

The rest of this paper is structured as follows. Section 2 describes the related research in cloud-edge computing, CTI data sharing and the data security issues in cloud-edge computing. Section 3 describes our trust model, whilst Section 4 provides a detailed description of the data sharing infrastructure and the different deployment models that are supported. Section 5 outlines the four pilot projects that are being used to validate the data sharing infrastructure. It says what levels of trust they have and the different deployment models they have chosen to satisfy their trust requirements. Section 6 provides a brief description of the implementation to date, the acceptance testing methodology that we have adopted and the validation work that we have already performed. Section 7 concludes with the limitations of the work to date and where future work is still required.

2. Related work

This section provides a brief overview of cloud-edge computing followed by a literature review of security research in cloud-edge computing.

2.1. Cloud-edge oriented computing

Edge computing aims to deliver compute, storage, and bandwidth much closer to the data sources and/or end users. Though research on edge-oriented computing is still in its infancy and we lack a universally accepted open standard [2], there are a number of definitions of edge computing available, e.g., Shi et al. [3] say that edge computing refers to the enabling technologies allowing computation to be performed at the edge of the network. Zhang et al. [4] say edge computing is a novel computing model that allows the storing and processing of data at the edge of the network, and provides intelligent services near to the source of the data by collaborating with cloud computing. A similar concept to edge computing is fog computing, which was first proposed by Cisco, and aimed at extending cloud computing to the edge of network [5]. Vaquero and Rodero-Merino [6] defined "fog computing is a scenario where a huge number of heterogeneous (wireless and sometimes autonomous) ubiquitous and decentralized devices communicate and potentially cooperate among them and with the network to perform storage and processing tasks without the intervention of third parties. These tasks can be for supporting basic network functions or new services and applications that run in a sandboxed environment. Users leasing part of their devices to host these services get incentives for doing so".

There is no clear distinguishing feature between fog computing and edge computing, since both push the intelligence and processing capabilities out of a centralized infrastructure into the logical extremes of the network close to the data sources and end users. But from the resource management point of view, the Fog, compared with the Edge, is a highly virtualized platform that provides computation, storage, and networking services between end devices and cloud computing data centres [5]. In most identifiable scenarios, fog computing is often used when the task is service oriented, while edge computing occurs more if it is as an analytical task. From a hierarchical design view, the Fog is located between the Cloud and the Edge [7], such that a cloud-fog-edge three-tiered architecture has been recognized in many prior works [8–10].

In general, edge-oriented computing can bring three prominent benefits to end users. First, reduce latency: the latency to the end user can be lower than it would be if the compute was farther away. Second, mitigate bandwidth limits: the ability to move workloads closer to the end users or data collection points reduces the effect of limited bandwidth at a site. This is especially useful if the service on the edge node reduces the need to transmit large amounts of data to the core for processing. Third, increase security: data can be pre-processed and protected before it is transferred to the cloud. It is the last benefit that we leverage in our project.

2.2. Cloud-edge security

Even though cloud-edge computing can bring a number of benefits compared with pure cloud computing, nevertheless, as edge devices proliferate, new attack vectors are emerging that take advantage of the proliferation of endpoints. Zhang et al. [4] surveyed the recent research of data security in the field of edge computing, which pointed out that the security of outsourcing data is still a fundamental issue in edge computing data security. Their review work comprehensively covers the research focusing on data security, i.e. confidentiality, integrity, availability, authentication, authorization, and privacy preservation. Controlling access to data is well researched, and a standard has been defined for this: XACML [11]. Our research makes use of this standard by employing an enhanced XACML policy decision point (PDP) to enforce our Data Sharing Agreement (DSA) policies. Our proposal for sharing CTI data covers all the issues mentioned by Zhang and is based on research by Carniani et al. [12] that proposed the design and implementation of a Usage Control Service to regulate the usage of resources in a Cloud IaaS service. They enhanced an XACML PDP to achieve this and integrated their solution into the OpenNebula Cloud platform.

Henze et al. [13] proposed a trust point, which is a local security-enhanced gateway at the border of a sensor network, that processes the sensor data before outsourcing it. In this trust point-based security architecture, the authors specified three trust domains: the fully trusted producer domain, containing the sensor nodes, the gateway devices and the data owner; the semi-trusted storage domain, including the cloud and cloud providers, who are assumed to be an honest-but-curious adversary; the untrusted consumer domain, consisting of entities such as services and service providers. Furthermore, the authors presented security solutions to address: the communication channel between the sensor network and the cloud, data confidentiality and privacy preservation for outsourcing the sensor data, and controlling access to the outsourced sensor data. However, their work had a static trust model and did not take into account different data protection mechanisms for different trust domains. In comparison, we assume a dynamic trust model, with user specified data protection mechanisms.

The existing approaches for provably secure outsourcing of data and arbitrary computations are either not scalable (e.g. tamper-proof hardware based) or not efficient (e.g. fully homomorphic encryption). Consequently, the Twin Clouds architecture [14] was proposed, consisting of a trusted private Cloud and an untrusted public Cloud. They apply the concept of garbled circuits to protect data and computation instructions in the public Cloud. The trusted private Cloud is used to encrypt data and computation instructions. Then the protected computation instructions can be securely processed in the public Cloud. The drawbacks of this approach are that the computation instructions can only carry out simple operations and they have to be re-encrypted by the private Cloud after each execution.

Pearson et al. [15] proposed a data management solution for protecting data in the Cloud that focuses on fine-grained access control of the outsourced data by attaching a sticky policy to the data, which states how and under which circumstances the data can be accessed. This is similar to our work. However, their trusted policy enforcement point requires the establishment of a trust metric for all external entities, rather than this being controlled by the user, as in our work.

Martinelli et al. [16] proposed a general model for a privacy aware collaborative information sharing and analysis system. This approach can calculate a trade-off score on privacy gain and data utility loss over the privacy preserving mechanism. The trade-off score leads to optimizing the analysis result with regard to the balance between privacy and accuracy. However, this paper only considered either a fully centralized (i.e. cloud) or fully distributed/P2P (i.e. edge) architecture, rather than the more practical hybrid (cloud-edge) architecture, with different trust domains, which is a feature of our work.

Several authors propose to process the data locally [17–19] and only utilize the cloud for storage of sanitized data. They cannot benefit from the computation resources provided by the cloud but they can guarantee that neither the cloud provider nor any other unauthorized third-parties can access their sensitive information. In comparison, our data security model is tailored towards not only storage but also processing in the cloud.

2.3. CTI data sharing

Sharing of CTI within a consortium or collection of similar organizations can be extremely beneficial because the member organizations often face common threats that are targeted towards similar type of systems, services and data. Cyber-security will be more effective if these organizations could work together to detect or prevent cyber threats facing them. Such collaboration helps in reducing risks faced by both the individual organization as well as the whole collective. Some basic methods of sharing CTI include public publishing of security alerts (like US-CERT alerts) [20], NVD vulnerability advisories [21], and security vendors' security bulletins. A more extensive list of potential CTI is given in the CWE [22], CVE [23] and CVSS [24] listings. Although all of these solutions and services share valuable information with the consumers, it is a one-way approach and most organizations do not or cannot easily reciprocate by sharing their CTI with these services.

Some cyber-security solutions do exist that are more closely related to our work, both in the proprietary and public domains. Proprietary solutions, like BT Security Threat Monitoring [25], monitor and collect security events from their customers. However almost without exception, the collated data is not shared with anyone. In the public domain, there were EU projects like Coco-Cloud [26] that enabled cloud users to securely and privately share their information. Similarly, CIF [27] is a CTI management system that supports aggregation, processing and sharing of CTI, but does not have any capabilities for addressing the sensitive nature of some CTI data by using anonymization or encryption techniques.

Zhou et al. [28] surveyed collaborative intrusion detection systems (CIDS) that address coordinated attacks. Such attacks (e.g. large-scale scans, worm outbreaks and DDoS attacks) often occur simultaneously in multiple networks. Consequently, sharing alert data in CIDS can bring a global view and collaborative analysis results to the users. The main research challenges are alert correlation algorithms and appropriate CIDS architectures, which were categorized as centralized CIDS, hybrid CIDS and fully distributed CIDS, which are similar to our deployment models described later. Both Lo et al. [29] and Shu et al. [30] proposed using CIDS to detect DDoS attacks to Cloud Computing, whereby one regional IDS shares its alert data with the other IDS systems. This can help to reduce the overall computational costs of detecting the same attacks in multiple IDS systems and therefore improves overall detection rates. The difference between them is that Lo uses the fully distributed architecture while Shu uses the hybrid architecture. Furthermore, Shu proposed using a Back-Propagation Neural (BPN) network to detect unknown attacks.

The utility and analysis accuracy of shared CTI data are obviously based on the utility of the CTI data obtained from the different sources. Clear text data has most utility, but sharing this often results in privacy leaks, since the CTI may include sensitive information that should not be shared with the other untrusted or unauthenticated partners. There has been plenty of research investigating privacy-preserving data sharing [9]. Fung et al. [8] proposed the privacy-preserving data publishing approach in order to optimize the trade-off between data utility and privacy. Different security requirements and metrics often lead to the use of distinct privacy-preserving data mining techniques [10]. Thus, it is necessary to have a flexible privacy-preserving data sharing model that can address the trade-off between data privacy and data utility, whilst taking into account the different levels of trust that collaborators have in each other. This is a subject of our research.

The EU H2020 project, Proactive Risk Management through Improved Cyber Situational Awareness (PROTECTIVE) [31] is investigating how to improve cyber security incident and risk management for public domain CSIRTs and SMEs. Coco-Cloud was an earlier EU project that enabled cloud users to share data securely and privately. It first proposed the use of Data Sharing Agreement (DSA) between collaborating users [26], but was not tailored to CTI. We make use of the DSA developed by the Coco Cloud project, and have enhanced it for use with CTI.

2.4. Use of distributed ledger technology

Distributed Ledgers Technology (DLT) [32] is proposed as the evolution of data control from a single entity to multiple parties. Rather than having the central administrator of a traditional database, a distributed ledger is a synchronized database across several locations and among multiple participants. This provides an auditable history of data transactions which is visible to every participant. One might consider blockchain to be a possible solution for the distributed deployment of C3ISP, e.g. Liang et al. [33] proposed a mobile healthcare system Integrating blockchain for personal health data sharing and collaboration. Although this addressed the sharing of data using blockchain, it did not consider the analysis of the data or the sharing of the results. Since DLT is more about the shared control of data rather than the confidential sharing of data, which is the subject of our research, one might consider DLTs to make the confidential sharing of data more difficult since there are now more nodes in control of copies. To address this, Es-Samaali et al. [34] presented a new distributed access control framework for big data based on blockchain technology. The authors applied Smart contracts [35] to express fine-grained and contextual access control policies for authorization decisions. Likewise, Wang et al. [36] proposed a blockchain-based framework for data sharing, which uses attribute-based encryption (ABE) technology for facilitating data privacy and fine-grained access control. In summary, there is no blockchain solution that would allow a straightforward deployment of the C3ISP Framework, although some solutions for specific functions do exist.

3. The trust model

Our trust model recognizes three basic levels of trust in the various parties: fully trusted, partially trusted and untrusted. Partially trusted means a party who is honest but curious i.e. it will not deviate from the protocol, but it may attempt to learn information from legitimately received messages e.g. an administrator may read the contents of files that has been stored on his machine, but would not modify them nor transfer them elsewhere. These three levels of trust can be applied to the following parties: the cloud infrastructure provider that runs the data sharing and analysis infrastructure,[1] and the collaborating organizations with whom the CTI data is to be shared for subsequent analysis. Data owners have to decide what trust they place in each of these parties and then act accordingly.

This leads to a five-level trust model as follows:

– Level 1, fully trusted: the organization sharing its CTI fully trusts all the other parties (both the collaborating organizations and the cloud infrastructure provider), and is willing to share its CTI with them "as is" for analysis i.e. as un-sanitized plain text;
– Level 2, fully trusted cloud infrastructure provider, partially trusted collaborators: the organization fully trusts the provider and is therefore willing to share its CTI "as is" with the cloud, providing that the CTI can be sanitized by the provider by either anonymization or pseudonymization before it is stored and shared with the other collaborators for analysis;
– Level 3, fully trusted cloud infrastructure provider, untrusted collaborators: the organization sharing its CTI fully trusts the provider and is therefore willing to share its CTI "as is" with the cloud, providing that the CTI can be protected by the provider encrypting the CTI before it is stored and shared with the other untrusted collaborators. Since the shared CTI needs to be analysed, this means that homomorphic encryption must be employed
– Level 4, partially trusted: the organization sharing its CTI partially trusts all the other parties, and is willing to share its CTI providing that the sensitive fields can either be anonymized or pseudonymized prior to sharing;
– Level 5, not trusted: the organization sharing its CTI does not trust the other collaborating organizations or the cloud infrastructure provider performing the analysis, and is only willing to share its CTI if it is fully encrypted prior to sharing. Since the shared CTI needs to be analysed, this means that homomorphic encryption must be employed.

Note that fully or partially trusted collaborators with a less trusted cloud infrastructure provider are not viable trust levels as the CTI would necessarily need to be sanitized to the trust level of the cloud infrastructure provider before it is transferred to it for sharing and analysis with the collaborators. Furthermore, a partially trusted cloud infrastructure provider and untrusted collaborating organizations (aka level 4.5) is also not a viable option from a performance perspective as this would require data sanitization to be undertaken by both the owning organization and then the cloud infrastructure provider. Although we have not separately enumerated it, it is theoretically possible.

In order to support our trust model, the protection of the CTI prior to sharing is conceptually performed by one or more Data Manipulation Operations (DMOs) on the CTI. A DMO can either remove, anonymize, pseudonymize or homomorphically encrypt a CTI field prior to sharing it. For level 1 trust, a null DMO is needed i.e. the original CTI data can be shared. For level 2 trust, the DMO operation will pseudonymize or anonymize a CTI field prior to sharing it. This is carried out by the trusted cloud provider. The ultimate in anonymization is to remove the field altogether. For level 3 trust the DMO operation homomorphically encrypts the CTI field. Again, this is carried out by the trusted cloud infrastructure provider. For level 4 trust, the DMO operations of level 2 are carried out by the sharing organization in its edge device. For level 5 trust the level 3 DMO operations are carried out by the sharing organization in its edge device. (For level 4.5 the DMO operations of level 4 would be carried out, followed by the DMO operations of level 3).

There is always a trade-off between data anonymization and data utility. As a generalization, the more anonymous the data is made, the less useful it becomes. The most extreme example of data anonymization is to remove certain fields from the data as stated above, but less extreme examples of anonymization can also significantly impact the utility. For example, by anonymizing IP addresses before the analysis, if the analysis subsequently reveals that a certain (anonymized) IP address is attacking the organization, this information is much less useful since it is impossible to learn what this IP address is and therefore configure a firewall to block it. Pseudonymisation suffers far less in this respect, since it is always possible to retrieve the original data again after pseudonymisation.

The DMOs are specified in a Data Sharing Agreement (DSA), which is the mechanism used by a sharing organization to specify what its trust policy is for sharing data with the cloud infrastructure provider and its collaborating partners. An edge device of the sharing organization will bind its DSA to the CTI before it is transferred to the cloud-edge infrastructure for sharing. The DMOs will be executed either in an edge device, or in the cloud by the cloud provider as described above, depending upon the level of trust the organization has. The DSA policy (enforced by the DMO) regulates the means to sanitize the data according to the user's requirements. Such requirements will ensure that the data does not lose its utility after the application of the DMOs (otherwise there would be little point in sharing it for analysis).

4. The C3ISP architecture

This section describes the C3ISP[2] architecture and its subsystems, as well as their interactions. It then describes the various deployment models for the infrastructure, for distributing it between the cloud and edge devices.

The C3ISP Framework is the main component designed to run in the cloud. The Local ISI is a component of the C3ISP Framework, and is designed to run in an edge device for trust levels 4 and 5 (i.e. where the cloud provider is not fully trusted). The C3ISP Gateway and Portal are designed to run in either the user's edge devices or in the cloud services, depending on the different use cases as presented in the later sections on the pilot studies. Fig. 1 makes no assumptions about where the CTI data comes from. For example, the user's organization could be using an outsourced

[1] In all cases, we assume that the software implementation of the infrastructure is fully trusted. By this, we mean that the software implementation does not have any backdoors or trojans embedded in it., and that all known bugs are fixed immediately.

[2] The name is derived from the name of the EU H2020 project in which it was developed: Collaborative and Confidential Information Sharing and Analysis for Cyber Protection (C3ISP) [37].

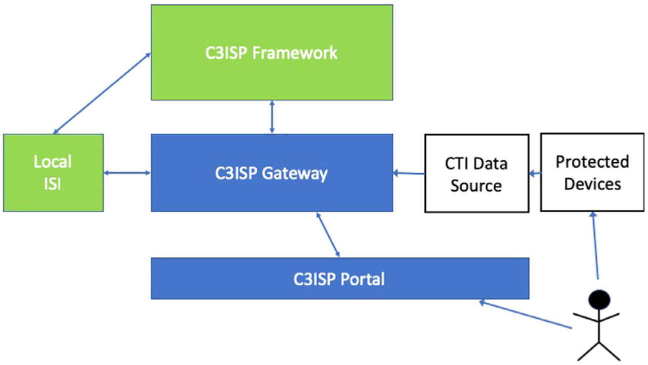


Fig. 1. An overview of C3ISP architecture.

Managed Security Service provider to operate its firewalls, AVS, IDS etc. in its devices, and this third party would collect the CTI that is subsequently going to be analysed by the C3ISP Framework. Alternatively, the user's organization could be operating its own protection software and producing the CTI itself. Either way, the CTI is provided to the C3ISP Gateway via a pluggable component (the MSS client in Fig. 4). Users may access the C3ISP infrastructure through a web browser, whose access to all the C3ISP components is mediated by the C3ISP Portal and Gateway.

4.1. The C3ISP cloud framework

The C3ISP Framework comprises 4 main subsystems: the Data Sharing Agreement (DSA) Manager, the Information Sharing Infrastructure (ISI), the Information Analytics Infrastructure (IAI), and the Common Security Services (CSS) as shown in Fig. 2.

A **Prosumer** is an actor who may play the role of Producer and/or Consumer. A Producer is an actor who supplies its own CTI data to the C3ISP infrastructure for sharing with other actors prior to analysis. A Consumer is an actor who performs analysis on the shared CTI data and consumes the results. CTI data sharing is regulated by policies (i.e. a set of rules) called a Data Sharing Agreement (DSA).

The **Data Sharing Agreement (DSA) Manager** is in charge of handling the DSA lifecycle, from template creation, policy instantiation, DSA usage and termination. A security expert defines a set of policy templates, each with a specific purpose for a particular trust model e.g. anonymize particular CTI fields, omit certain CTI fields, or homomorphically encrypt the CTI. A large set of policy templates have already been defined by the security experts in the C3ISP consortium, and these will be released along with the open source code. A set of prosumers collaboratively instantiate a policy from one of the templates, in order to tailor it to their specific requirements e.g. say who the collaborating prosumers are, who can access the analysis results etc. Templates and instantiated policies are held in the DSA store for subsequent retrieval by authorized users. DSAs have a lifetime, so that when a DSA expires, prosumers may no longer access any CTI that is protected by this policy. This ensures that CTI is not inadvertently left stored for long periods of time, giving unauthorized users more chance of accessing it. Templates and policies are initially written in a human readable controlled natural language (CNL) via the Policy Editor (see Figs. 3 and 6), and the DSA Mapper converts this into an enhanced XACML policy for enforcement at run time.

A DSA comprises the following components:

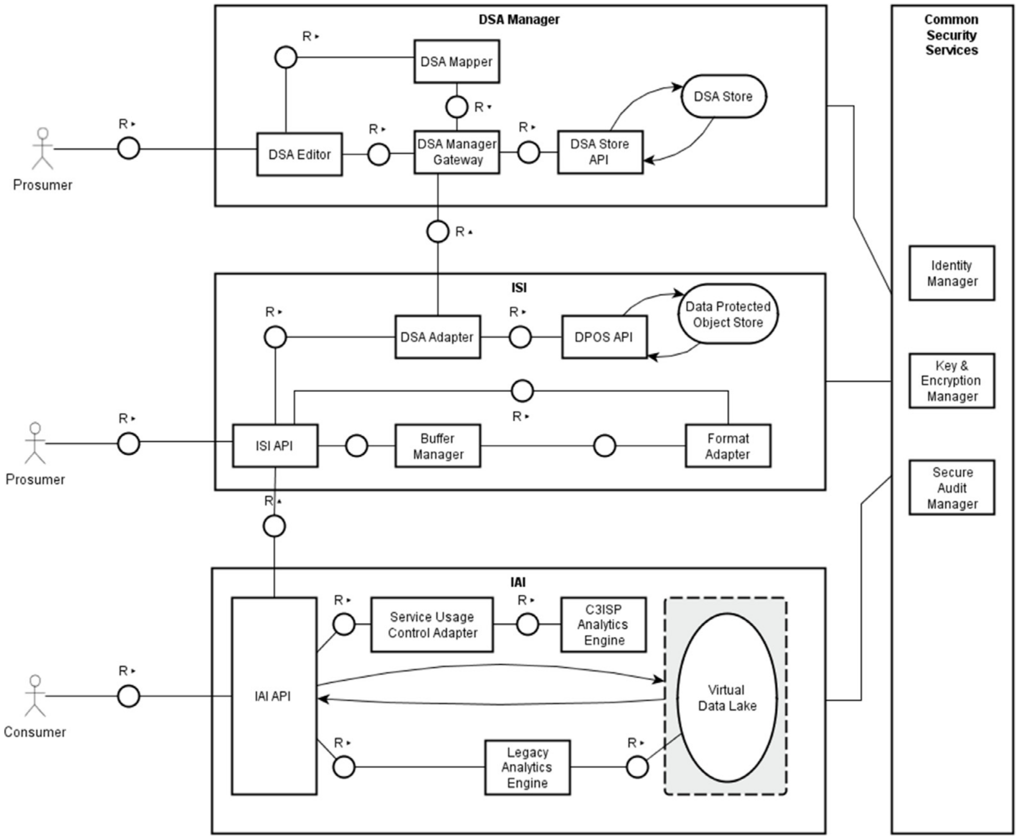
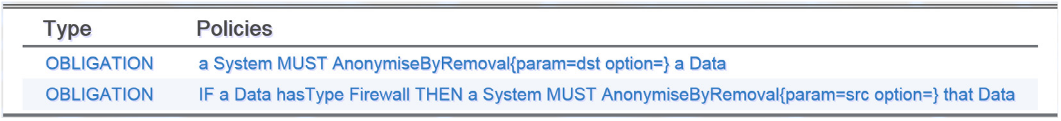


Fig. 2. An overview of C3ISP cloud Framework.



Type	Policies
OBLIGATION	a System MUST AnonymiseByRemoval{param=dst option=} a Data
OBLIGATION	IF a Data hasType Firewall THEN a System MUST AnonymiseByRemoval{param=src option=} that Data

Fig. 3. Example DMOs in CNL for anonymization by suppression.

– a set of Data Manipulation Operations (DMOs), which say how the attached CTI should be protected prior to storage;
– a usage control access control policy, that defines who is allowed to access the attached CTI for what purposes, and under what constraints;
– a set of obligations that are placed on the use of the attached CTI data, and in particular, rules for how the DSA should be inherited by any analysis results that are derived from the attached CTI.

The **Information Sharing Infrastructure (ISI)** supports the management of Data Protected Objects (DPOs). A DPO is CTI data with a sticky DSA policy attached to it. This bundle is first encrypted by the DSA Adapter, which includes a specific component, the Bundle Manager, for managing DPOs, before it is stored in the DPO store. The Bundle Manager employs the Encryption Manager of the CSS subsystem for performing encryption/decryption operations. This ensures that if the DPO store is compromised, the attacker will not retrieve any useful information from it. DPOs can be created, retrieved (for analysis), deleted, and moved (from edge to cloud) by the ISI API.

CTI comes in many different formats — nearly every security application has its own unique format. We have chosen to use the standard STIX format defined by NIST [38] as the common DPO storage format. Consequently, the Format Adapter, which is part of the ISI Subsysetm, converts the CTI from its local format into the STIX format before it is inserted in the DPO.

The DSA Adapter also enforces the prosumer's DSA on the CTI before it is stored in the DPO store, and every time it is retrieved from the DPO store for performing an analytics service. As stated, the DSA contains a set of authorization rules with a set of Data Manipulation Operations (DMOs). The authorization rules are evaluated by the Authorization Engine embedded in the DSA Adapter in order to determine whether the CTI can be exploited to perform the requested operation. If the authorization rules allow the usage of the CTI, the related DMO must be executed before the CTI can be actually used. Each DMO specifies one operation to be performed on the CTI e.g. anonymize the source IP address via substring removal, or remove the number of bytes transferred. Another form of anonymization adopts Differential Privacy techniques [39]: intuitively, they allow us to preserve specific properties of a dataset while altering each of the dataset's values. As an example, Geo-indistinguishability [40] allows us to scramble each geographic location in a dataset (thus making re-identification a very difficult exercise) but preserving an arbitrary consistency with the original distribution DMOs provide users with full control over the confidentiality of their CTI, and allow them to remove, pseudonymize, anonymize or homomorphically encrypt the fields of their CTI before it is shared with other prosumers. DMOs are associated with authorization rules by means of obligations similar to those stated in the UCON_{ABC} model [41]. Such obligations allow additional actions to be performed, for example, a forced deletion of CTI data when the DSA-prescribed retention period expires. Such mechanisms reinforce the control that prosumers have on their CTIs.

The **Information Analytics Infrastructure (IAI)** provides the interface for invoking analytics services on the DPOs that have been shared and (centrally) stored through the ISI. A consumer selects the set of DPOs to be combined for analysis, and says which analytics service should be invoked. The analytics execution result is computed under the control of the DPOs' common DSA, which provides the usage control policy for the Service Usage Control Adaptor (an enhanced XACML PDP).

The DSA also contains inheritance rules for the handling of the analytics result. The analytics result, coupled with the DSA inherited from the common DSA, is submitted as a new DPO to the ISI, from where it can be retrieved by authorized consumers, and also used as an input to subsequent analytics services.

Finally, the **Common Security Services (CSS)** support several of the functions of the C3ISP architecture. The Identity Manager authenticates the prosumers, and provides their identity attributes to the enhanced XACML PDP. The Secure Audit Manager is necessary to trace the operations performed within the C3ISP infrastructure, in particular those related to access and usage decisions, and to guarantee system accountability to show it operates as planned and as specified in the DSA rules. The Key and Encryption Manager provides for the confidentiality of the computations by homomorphically encrypting the CTI data, and the secrecy of the stored CTI data by encrypting the DPOs.

4.2. The C3ISP cloud-edge framework

Edge computers normally run the C3ISP Gateway and the Portal of the C3ISP Architecture. All the dashed components in Fig. 4 are optional edge components, and depend upon the resources and trust level used by the prosumer's organization.

The **C3ISP Gateway** is the interface/middleware between the end user's environment and the C3ISP Framework. The C3ISP Gateway retrieves/collects CTI data from different data sources. The Managed Security Service (MSS) Client in Fig. 4 is a component that has to be tailored to the end user's environment. It is responsible for collecting the CTI and uploading it to the C3ISP infrastructure (in the cloud or the edge) for sharing and analysis. Through an easy to use web interface (the Portal) to the C3ISP Gateway, the user is able to manage all of their C3ISP related tasks with only a standard web browser, i.e. choosing which CTI data to share and on what schedule, creating and selecting DSAs, running collaborative analytics etc. The Orchestrator will accept user scheduled tasks and periodically collect CTI and run analytics tasks automatically on behalf of the user.

The **local ISI** has identical functionality to the cloud-based ISI, but has a different placement and deployment configuration (see below). It is needed for trust levels 4 and 5 where the operator of the cloud-based ISI is not fully trusted to sanitize the sensitive CTI of the prosumer. In particular, the local ISI supports the Move operation, to move protected DPOs from the edge to the cloud ISI. The local ISI configures a local DPO store for storing the protected CTI data in the edge prior to invoking the Move operation. The DSA Adaptor in the local ISI is used to sanitize the CTI data before sharing it with the central C3ISP Framework, through evaluating the DMOs contained within the DSA. In this way any sensitive CTI fields are protected before moving them to the (partially) untrusted cloud provider.

4.3. Deployment models

We foresee different ways to instantiate the C3ISP Framework and we refer to them as deployment models. A deployment model

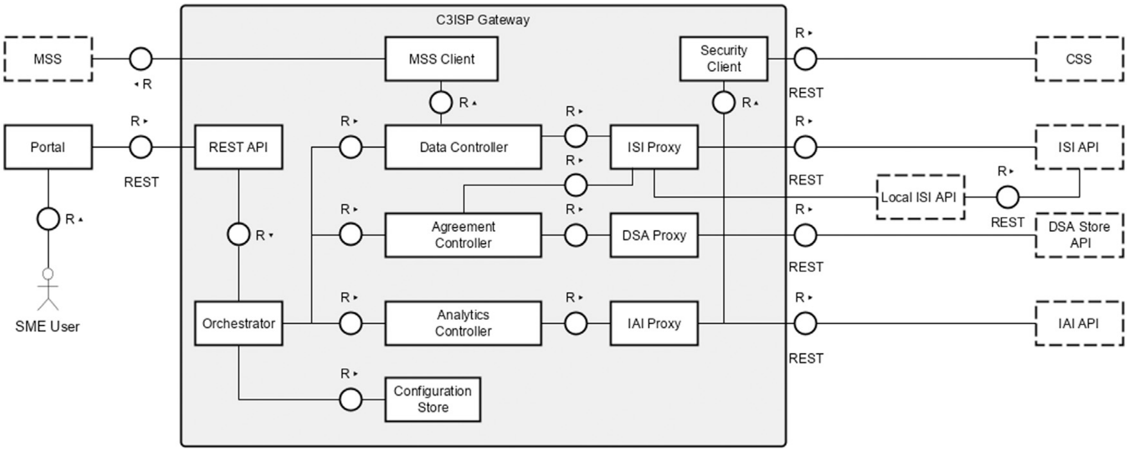


Fig. 4. The C3ISP Gateway Architecture.

Table 1
Relative effects of data manipulation on privacy, accuracy and performance.

Data manipulation operation	Privacy	Accuracy	Performance
None — Plain text	Low	High	High
Pseudonymization	Medium	High	Medium
Anonymization	High	Low	Medium
Homomorphic encryption	High	High	Very low

is a specific C3ISP configuration in which the C3ISP Framework subsystems are deployed to match specific use case scenarios and trust requirements. Prosumer organizations choose the deployment model according to their specific business requirements, in particular taking into account the level of trust they have or need for the use cases to be supported.

The deployment models describe where the main C3ISP Framework subsystems can be deployed, either in edge devices in the prosumer's environment, or in a cloud environment, or in a combination of the two. We have identified four deployment models for supporting a broad range of possible scenarios. In most cases, the Gateway and Portal run on edge devices.

- **Fully centralized**: all C3ISP Framework subsystems operate in the cloud only. This model supports the three fully trusted levels of trust.
- **Hybrid**: the ISI operates on both an edge device and the cloud, with all the other framework subsystems being cloud based. This model is designed primarily to supports trust levels 4 and 5 by performing all the data manipulation operations in the Local ISI, although it can support trust 1, 2 and 3 by not performing any data manipulation operations in the Local ISI.
- **Distributed ISI**: the ISI is on an edge device only and the IAI and DSA Manager are in the cloud. This model supports the partially trusted and untrusted levels of trust.
- **Fully distributed**: all C3ISP Framework subsystems operate on edge devices. This model supports the untrusted level only. The storage, sharing and analysis is done in a completely distributed manner, exploiting a distributed hash table (DHT) based model for communication, distribution of information and computation. Some analytics such as those based on Secure Multi-Party Computation are particularly suitable for this deployment model.

Fig. 5 shows the trust versus data control trade-offs in the deployment models:

A special consideration has to be made for the CSS subsystem: this subsystem has a critical role in the trustworthiness of the C3ISP Framework, in particular when considering distributed scenarios. For example, a distributed CSS for identity management could leverage identity federation technologies. Key and encryption services could leverage a PKI to address key distribution issues. Auditing, however, should preferably be centralized, maybe at a trusted third party, to address segregation of duties and non-repudiation. The simplest scenario is a centralized cloud deployment for the CSS with the assumption that it is fully trusted by all the Prosumers. However, the architecture can cope with existing consolidated solutions in the domain of identity management, key and encryption manager, and auditing, provided that they use standard interfaces and are aligned with the expectations expressed here.

Whilst there are several valid data processing approaches for privacy-preserving the outsourcing of CTI data analysis, nevertheless there are some other concerns that also need to be taken into account, because the distinct privacy-preserving outsourcing approaches have different impacts on data analysis accuracy and data processing performance. Table 1 shows the relative effects of various data manipulation approaches on the privacy, accuracy and performance of the data analytics service.

We can see that analysing plain text has the highest accuracy and performance but no privacy preserving. Pseudonymization (e.g. encryption based), homomorphic encryption techniques and anonymization all provide privacy but at some cost to performance, with homomorphic encryption being the worst. In addition, anonymization techniques suffer from loss of accuracy. Hence, only in the extreme case of no trust in the analytics service or in the other data sharing prosumers should homomorphic encryption be used.

5. The C3ISP pilots

Four pilot projects are validating the C3ISP infrastructure, to see how well it fits their various security and trust requirements for confidentially sharing CTI data for analysis. These are the Enterprise Pilot, the SME Pilot, the ISP Pilot and the CERT Pilot. The pilots represent a wide set of use cases, to see if the infrastructure is widely applicable.

5.1. The enterprise pilot

The Enterprise pilot is concerned with providing a Security and Threat Intelligence Monitoring service to relatively large public and private sector organizations. These organizations typically outsource to Managed Security Service Providers (MSSPs) [42].

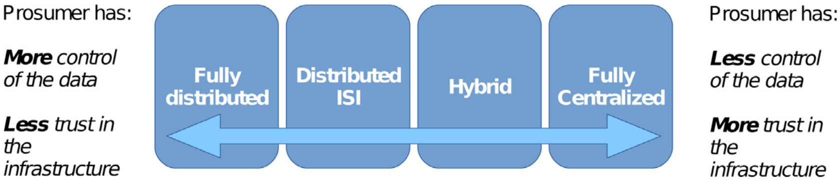

Prosumer has:
More control of the data
Less trust in the infrastructure
Fully distributed
Distributed ISI
Hybrid
Fully Centralized
Prosumer has:
Less control of the data
More trust in the infrastructure

Fig. 5. Trust vs. data control trade-off in the deployment models.

MSSPs quite often offer a platform solution as the MSS for cyber threat monitoring and analytics for these organizations. Examples of such platform solutions include BT Cyber Security Platform [43], BT Security Threat Monitoring [25], SAP Enterprise Threat Detection [44], McAfee Enterprise Security Manager [45] and Alien Vault Unified Security [46]. The MSS platform is a centralized solution at the MSSP side. Such a platform typically but not necessarily consists of a Data Lake, existing portals and tools. The collected CTI data is processed, analysed and in case threats are detected, reactions are triggered in collaboration with the customer. At present, the CTI data of each customer are stored in distinguished data lakes and analysed in isolation. In principle, the greater the volume and variety of data available for analysis and correlation, the better or higher the quality of threat information that can be provided. Thus combining the analysis of data from multiple customers has advantages both to the MSSP and its customers. However, concerns about exposing sensitive information to competitors and threat agents may make security conscious enterprises reluctant to allow this without safeguards and assurances. At a higher level, there are also benefits to be obtained by sharing threat intelligence among service providers and CERTs, but the initial focus of this pilot is on the intra-service-provider application of the C3ISP infrastructure.

Since all the CTI data, which has previously been collected from the enterprises premises, is stored centrally at the MSSP's premises (i.e. on a multi-tenanted data lake), and the MSSP will be the operator of the C3ISP cloud infrastructure, then the *fully centralized* deployment model is chosen. Each enterprise customer (or MSSP analyst working on behalf of the customer) is able to define its own DSA policies, depending upon its level of trust in the other customers with whom it will share its CTI data.

In this Pilot, the Gateway is run by the MSSP rather than the enterprise. The enterprise accesses the Gateway to request the MSSP to share its CTI with the C3ISP infrastructure. The MSSP processes the relevant security events and logs for that enterprise in order to generate the CTI. The enterprise selects its DSA, again via the Gateway, and thereafter, the MSSP will be in charge of sending the CTI and DSA to the C3ISP Framework, which will analyse the CTI in terms of the DSA and send the C3ISP analysis results back to the enterprise. Note that in this Pilot, the MSSP also has its own data analysis capability. However, this capability is of limited scope as it only focuses on the data from a single enterprise domain. The C3ISP Framework, on the other hand, is able to collect and aggregate CTI from multiple enterprises, so that it can have a broader scope to carry out the security analysis.

5.2. The SME pilot

The SME Pilot is concerned with the collection and sharing of SME cyber security data with the C3ISP service without disclosing privacy sensitive information. SMEs typically have limited resources and expertise. Consequently, SME participation in the C3ISP eco-system needs to be done seamlessly with as little effort as possible, which means that most of the required management and operational processes should be offloaded in order to minimize the utilization of SME resources (i.e. software and hardware). The SMEs need to be able to choose the type of confidentiality controls that are appropriate for safeguarding their CTI data by the C3ISP Service, e.g., to go for either open access, or data anonymization/pseudonymization techniques, or even use homomorphic encryption based techniques. Specific SME requirements are:

- Due to the availability of different data confidentiality and access options, the SMEs need to be able to confidently share their specific types of CTI data via the C3ISP Framework, with fully trusted or even non-trusted collaborators.
- The C3ISP Framework should incorporate diverse techniques for supporting the protection of CTI data, and the SMEs do not have to be aware of the inner workings of these techniques. Thus, the SMEs should be able to choose from the alternative techniques most suitable to them from their own perspective without worrying about their design and implementation.
- The C3ISP Framework should incorporate diverse techniques for analysing the shared CTI without the SMEs worrying about issues like information leakage, as this process should be transparent to the SMEs.

The above are achieved by employing the hybrid deployment model, and delegating the tasks for collecting, processing and sharing the CTI to the C3ISP Gateway and local ISI. Each SME may define its own DSA policies, and delegate the task of 'own CTI data' analysis to the C3ISP cloud infrastructure. Alternatively, the SME may collaborate with other SMEs, and define a common DSA policy so that analysis of shared CTI data can be performed by the C3ISP cloud infrastructure. DSA policy enforcement is performed by the C3ISP infrastructure, and this software is always assumed to be fully trusted. The C3ISP infrastructure provider however need not be fully trusted, nor need the other SME collaborators.

In the current SME Pilot, all the SMEs have subscribed to the same Managed Security Service (MSS), which collects all the security events and logs from the assets that the SMEs have configured to be protected by the MSS (which are usually virtual machines). This CTI is collected by the C3ISP Gateway, which uses the local ISI component to process and format them into the standardized STIX format, and to then perform protection operations (DMOs) on the formatted CTI according to the rules in the DSA. Finally, the protected DPO is moved from the local DPO store to the C3ISP Framework DPO store, from where it can be subsequently analysed, either on its own, or in combination with other DPOs that share the same DSA policy. Each analysis generates a security report containing the analysis results, which are stored back in the Framework DPO store. Subsequently, the security report may be transferred, via the C3ISP Gateway, back to the SMEs for review. Note that the MSS used in the SME Pilot is different from the MSSP used in the Enterprise Pilot. The former is the common and singular source of security services and CTI for all the SMEs, whereas the latter works at a higher level of abstraction and is the provider of possibly unique MSS to each enterprise that is collaborating in the C3ISP eco-system.

5.3. The ISP pilot

The ISP Pilot is concerned with the sharing of CTI that comes from the Italian ISPs and *Registro.it* (the entity responsible for managing Italy's top-level domain names), in order to discover and mitigate possible attacks. The C3ISP Framework provides analytics to ISPs, which can benefit from a federation of data analysis that is performed in a secure and private way. Thus, ISPs will benefit from data-manipulation operations such as data-anonymization, controlled by Data Sharing Agreements (DSAs) to protect, regulate and guarantee an expected privacy level of CTI shared with the C3ISP Framework. The CTI is created as a set of security reports that are produced by Security Scan Software, which Registro.it provides to the ISPs as a remote tool.

An ISP can be seen as an isolated entity with enough resources to deploy some of the C3ISP components locally. Therefore, the *hybrid deployment model* is chosen for the ISP Pilot, where each ISP hosts a local ISI and remotely communicates with the centralized ISI and IAI subsystems hosted by the C3ISP infrastructure provider.

The Gateways, running in edge devices at each ISP, collect the CTI from dedicated servers, e.g. DNS, SSH, Netflow and the Security Scan Software and pass it to their local ISI for formatting into the STIX format and applying DMOs, prior to moving it to the C3ISP infrastructure for storage, merging and subsequent analysis. Finally, the results of the analytics are provided back to the ISPs via their Gateways, so that they can react depending of the kind of results received.

5.4. The CERT pilot

The CERT Pilot is concerned with fostering cyber threat information sharing between the Italian CERT and other C3ISP stakeholders, in particular ISPs and Enterprises, with the aim of preventing or timely reacting to security attacks. The CERT pilot is the most general of all the pilots, and imposes noticeable challenges, since, differently from other pilots, it has to be ready to receive and handle any possible type of CTI information, managing data with different formats and semantics. Moreover, this pilot supports a plurality of possible prosumers, hence the interface must be general enough to match the requests of both private users, public and private organizations of different sizes. Consequently, the CERT Pilot has adopted the *hybrid deployment model* in which it is the central C3ISP cloud infrastructure provider and its prosumers, e.g. ISPs or large enterprises, are edge organizations that run the local ISI and Gateway on their premises in edge devices in order to sanitize and cleanse their CTI data prior to sharing it with the CERT.

The CERT is a public entity which collects CTI from multiple prosumers, stores and categorize the collected information and uses it to run analysis. In particular, an analysis can be requested by a specific prosumer, or by the CERT itself. All analysis results are stored as DPOs, and are dispatched to interested prosumers providing the DSA policy allows it. Finally, cyber intelligence, collected or inferred from the analysis results, will be made publicly available through the CERT website as a newsfeed related to cyber threats of public interest.

In the next phase of development, the CTI collected from all the different Pilots should also be sharable with the CERT C3ISP infrastructure, in order for the CERT to be able to create a common knowledge base to prevent or react against security threats targeting all the Pilots' participants. Hence, the C3ISP platforms hosted by the CERT, the Enterprise pilot (and ultimately any other organizations) should also be able to share their CTI and or DPOs with each other for subsequent analysis.

Table 2
Deployment models of the Pilots.

	Hybrid	Fully centralized	C3ISP infrastructure provider
ISP Pilot	✓		Cloud provider
CERT Pilot	✓		CERT
Enterprise Pilot		✓	MSSP
SME Pilot	✓		Cloud provider

5.5. Summary of deployment models used by each C3ISP pilot

Table 2 summarizes the deployment models chosen by each of the four Pilots.

It can be seen that none of the Pilots have chosen either of the distributed deployment models. This is because:

(a) The Distributed ISI model is subsumed by the Hybrid model and
(b) The Fully Distributed model is only applicable for organizations that have zero trust in the other stakeholders.

The four Pilots require the collection of CTI data from different internal and external sources. Table 3 summarizes how this is handled by each Pilot. Note that CTI collection from the wide range of prosumers in the CERT pilot is not yet complete.

All the Pilots need to have some form of data sanitization available to them before they may be willing to share their CTI with the C3ISP Framework. Different prosumers will choose different DMOs to perform this sanitization. Table 4 summarizes the DMOs (Data Manipulation Operations) that need to be available to each Pilot.

6. Implementation, acceptance tests and validation

In this section, we briefly describe the implementation, the acceptance and validation methodology, and the results of the validation performed so far.

6.1. Implementation

The development was carried out using Jenkins to manage the continuous integration and deployment automation processes. Jenkins was integrated with various quality and assurance tools, in particular: CheckStyle for Java code syntactical checking and standard adherence; FindBugs and FindSecurityBugs for Java code bugs discovering and security static analysis; OWASP Dependency Check for checking security vulnerabilities in external code dependencies (e.g. used Java libraries); the Junit framework for creating unit tests; Cobertura for measuring the percentage of code covered by the unit test and for identifying code not involved in the tests.

The developed C3ISP infrastructure runs in a series of VMs (with a VMware hypervisor) on Ubuntu 16.04.3 LTS as shown in Table 5 below. The base configuration includes the following software components: Oracle Java Development Kit, version 1.8; GCC (GNU project C and C++ compiler), version 5.4; and Python, versions 2.7 and 3.5. OpenLDAP was used as the initial user authentication and identification repository, but it is planned to enhance this with Open ID Connect. All the infrastructure components support REST interfaces and SpringBoot was used in their development.

Table 3
Mapping of CTI collection components.

Pilot	CTI source	Data owners	Component responsible for collection
ISP	Dedicated services and Registro.it	ISPs	C3ISP Gateway (Security Scan Software client)
CERT	Prosumers' systems	Prosumers and CERT	C3ISP Gateway (Prosumer specific clients TBD)
ENT	MSSP's Data Lake	Enterprise customers	C3ISP Gateway (Data Lake client)
SME	MSS	SMEs	C3ISP Gateway (MSS client)

Table 4
Availability of DMO components.

Pilot	Component responsible for DMO processing	Type of DMO processing			
		Plain text formatting	Pseudonymisation	Anonymization	Homomorphic encryption
ISP	Local ISI	Yes	No	Yes	Yes
CERT	Local and remote ISI	Yes	No	Yes	Yes
ENT	Central ISI	Yes	No	Yes	No
SME	Local ISI	Yes	Yes	No	No

Table 5
C3ISP virtual machine.

VM	CPU	RAM GB	DISK GB
ISI	4	12	100
IAI	8	16	400
DSA Manager	2	4	40
Audit Manager	1	2	22
Key Enc Manager	8	16	100

6.2. *Performance evaluation*

Fig. 6 shows the time required by our framework to perform the DetectDGA[3] analytic, reporting the time of the internal components depicted in Fig. 2. From Fig. 6 we can clearly see that the execution time is mainly due to the time required to execute the DetectDGA analytic and to execute the decryption operation by the Bundle Manager (which is internal to the DSA Adapter component). Remember that all the CTI is stored encrypted in DPOs so has to be decrypted before any data analysis can be performed. These times grow with the size of the DPO. The time required by the Usage Control component (which is the authorization engine embedded in the DSA Adapter component) to perform the evaluation of the usage control policy is about 1 s, and it does not depend on the size of the DPO. In fact, it depends on the complexity of the policy. Both the times required by the DPOS API to read the DPO from the DPOS and the time required by the Buffer Manager and the Format Adapter components to create a data lake in the right format for the execution of the analytics are very low with respect to the other components. For instance, to read a DPO of 10 kb, 1 Mb, and 3 Mb takes respectively about 70, 270, and 330 ms.

6.3. *Validation methodology*

The pilots are validating the operation of the infrastructure in two stages. The first stage was to define a comprehensive set of functional and non-functional acceptance tests for each of the pilots, based on their requirements for the confidential sharing and analysis of CTI data. 30 acceptance tests were defined in total for the ISP pilot, of which 8 appertained to the collection and confidential sharing of CTI — the subject of this paper. 15 tests were defined for the CERT pilot, of which 8 related to this paper; 22 were defined for the SME pilot, of which 10 related to this paper, and 27 for the enterprise pilot, of which 7 related to this paper. The second stage was the application of the Goal, Question, Metric (GQM) [47] method to the user stories associated with each of the C3ISP pilots. The GQM method incorporates the acceptance tests as a set of questions, as a way of obtaining key evidence from the users regarding the acceptance tests and the validation of the C3ISP Framework.

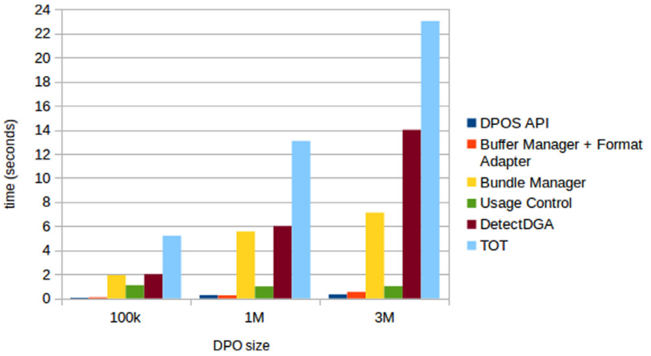


Fig. 6. Example ISP Pilot in policies in CNL for the validation.

6.4. *Validation results*

At the time of writing, the C3ISP project was 30 months through its 36-month timeframe, so some of the components were not fully implemented, for example, not all the CTI sanitization or analytical functions were available. In particular, the homomorphic encryption function was only available in the cloud/central ISI, and not in the local ISI, meaning that Trust Level 5 cannot currently be supported. Consequently, whilst we were able to test the various deployment models and levels of trust, we could not validate that the implementation had reached sufficient maturity to be able to cater for all the identified sanitization requirements and levels of trust. However, sufficient of them were available to show that both data sanitization and data analytics can be performed by the C3ISP infrastructure. When other functions become available they should be able to be plugged into the infrastructure via the REST interfaces that have been defined and validated. Finally, the user-friendly GUIs were not all complete. The testing to date has allowed us to gather direct feedback from the project's stakeholders in order to steer the remaining activities during the final months of the project.

[3] The DetectDGA analytic works on DNS request logs and identifies whether domain names have been resolved within domains that refer to a Domain Generating Algorithm (DGA). These domains are often used by malware to register new domains on the fly to avoid the malware depending on a fixed domain or an IP address that could quickly be blocked. Thus, the malware switches to a new domain at regular intervals and thus prevents a new version of the malware needing to be released.

6.4.1. The ISP pilot validation results

The ISP Pilot used both synthetic and test CTI data for its validation. The test data was extracted from the BIND DNS server hosted in the testbed virtual machine, and generated for the purposes of internal validation. The synthetic data was from a BIND DNS server of an ISP that participated during the requirements collection phase, and made its data available for the validation phase. The size of the test log was 10 kb with 79 requests. In comparison the synthetic log was ∼30 Mb with 250 k requests.

In general, the functionality related to CTI data collection and sharing performed well for small files: the pilot was able to upload and retrieve CTI data to and from the C3ISP Framework. With larger files e.g. the synthetic CTI, the testers reported that the Format Adapter, DPOS and DetectDGA analytics function worked well but required a number of seconds to process large CTI files (see Fig. 6).

During the validation, the testers were able to set up policies to manage the DPO access. Fig. 7 shows four policies (three authorizations and one prohibition) defined for the validation.

Of the 8 relevant acceptance tests, 4 Passed, whilst 3 Partially Passed and 1 was N/A (not available yet) due to the lack of full feature availability.

6.4.2. The CERT pilot validation results

The CERT Pilot used a combination of real and honeypot data. The majority of data used for validation was real data either coming from public sources or provided by CERT stakeholders. One ISP provided a set of real emails to be analysed for spam analysis, together with a set of malware collected through internal honeypots.

Of the 8 relevant CERT acceptance tests, 5 were reported as Passed, 2 as Partial, and 1 as N/A. The tests marked as Partial or N/A were due to a lack of full analysis functionality.

The CERT Pilot reported that the CTI sharing aspects of the C3ISP Framework worked well. Since it used only small email data files, no performance problems were encountered when importing or retrieving data from the framework ISI. The CERT partner was quite happy with the DSA Editor, but reported being able to create only a subset of the desired policies.

6.4.3. The enterprise pilot validation results

The Enterprise Pilot used two public datasets of cyber security information for the validation:

1. Intrusion Detection System dataset from "1999 DARPA Intrusion Detection Evaluation Dataset",[4] week 2 training data.
2. Honeypot dataset from "DDS Dataset Collection".

An additional synthetic Malware alerts dataset was derived from a subset of the public honeypot data (i.e. using the Source IPs and their geolocation, and assigning them with malware names). The reason for this was motivated by the need to ensure compliance with the legal obligations in processing real data, considering the limited maturity of the implementation of C3ISP Project (e.g. full privacy could not be assured).

Of the 10 relevant acceptance tests specified for the Enterprise Pilot, 5 were reported as Passed, 4 as Partially Passed and 1 as Not Assessed. The validation method foresaw the involvement of these pilot stakeholders: domain experts and cyber security analysts. They were presented with a demo and asked to fill in the questionnaire. These surveys received generally positive feedback for the Y/N questions, other than that related to the DSA editing functionalities.

The Pilot reported that the C3ISP Framework functionality for CTI sharing, search and retrieval worked very well, as did the user-end Pilot-specific software. One of the pilot strategic objectives was to combine the new data sharing and analysis capabilities brought by C3ISP with the existing MSSP legacy solutions. The surveys reported that the legacy Saturn analytics and their C3ISP support framework (creation and population of the data lake) were quite successful. The end-users were generally satisfied with DSA policy editing (which was performed by security experts), but would prefer to be able to create policies based on data classifications (such as less sensitive, sensitive, highly sensitive). This can be achieved by creating pre-defined DSA templates that are ready to be instantiated as DSAs.

6.4.4. SME pilot

This Pilot used primarily a combination of simulated attack data and passive test environment data. The simulated attack data included Firewall and Anti-Malware events triggered by the tester through simulation scripts. Passive test environment data included CTI events encountered during normal operations by an SME host.

Of the 7 relevant acceptance tests, 5 Passed, 2 Partially Passed, and 0 were N/A.

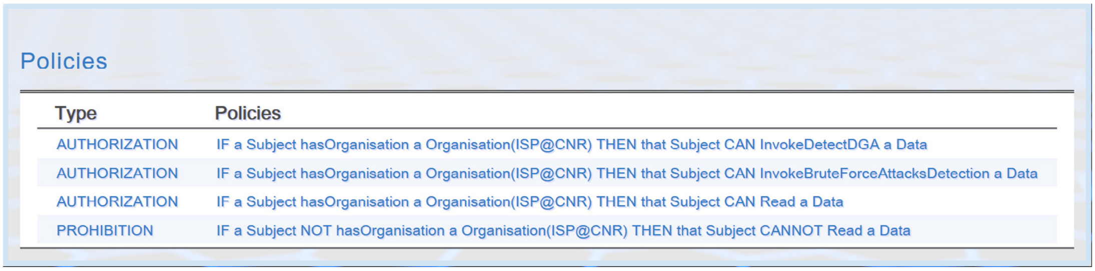
Interestingly, some Y/N questions were answered Y by the developers and N by the SME users. The SME users reported that this was due to the MSS Server and DSA Editor user interfaces being very difficult to use. However, DSA search/selection and importing CTI to the C3ISP Framework worked well using the end-user Portal software.

Regarding the analysis, we developed a specific analytic service for the SME pilot, termed findAttackingHosts, which analyses firewall CTI data for attacks, and lists the IP addresses of attacking hosts. We used four sets of CTI data for testing the analysis, which were: GridPocket, UniKent, BT, and a combined set (that contains all the former three individual datasets). We ran the analysis over each organization's own data separately, and then ran the analysis over the collaborative data as well, in order to determine if combining data reveals new IP addresses for each organization. We set 50 as the threshold of the attack count, which means that we consider an IP address malicious if it tries to establish a connection more than 50 times. Table 6 shows the incoming IP addresses that are detected 50 or more times. We can observe that UniKent and BT detected one malicious host each, i.e. 10.255.92.123 and 129.12.21.67 respectively, and GridPocket detected 17. However, by using the collaborative data, a new host (i.e. 91.189.95.83) was revealed, which was below the threshold of each organization's individual dataset. We found that this host was detected by BT 48 times and by GridPocket 2 times. So, by performing the C3ISP analysis service over collaborative data, new malicious hosts can be revealed and the result is mutually beneficial for all the SMEs. Another benefit of collaboration is that one SME that has already been attacked (e.g. GridPocket) can reveal its attackers to the other SMEs through the collaborative results, perhaps before they themselves are aware of it.

7. Limitations, conclusion and future work

The C3ISP project's objective is to define a collaborative and confidential information sharing, analysis and protection infrastructure as a service for cyber security management. It caters for a wide range of trust levels, ranging from full trust in both the cloud infrastructure provider and the collaborating parties to no trust in any of them. Four deployment models are supported in order to cater for the various levels of trust, and range from all CTI data processing being performed in the cloud to all CTI processing being performed in edge devices. Even though the project still had

[4] https://www.ll.mit.edu/r-d/datasets/1999-darpa-intrusion-detection-evaluation-dataset.



Policies

Type	Policies
AUTHORIZATION	IF a Subject hasOrganisation a Organisation(ISP@CNR) THEN that Subject CAN InvokeDetectDGA a Data
AUTHORIZATION	IF a Subject hasOrganisation a Organisation(ISP@CNR) THEN that Subject CAN InvokeBruteForceAttacksDetection a Data
AUTHORIZATION	IF a Subject hasOrganisation a Organisation(ISP@CNR) THEN that Subject CAN Read a Data
PROHIBITION	IF a Subject NOT hasOrganisation a Organisation(ISP@CNR) THEN that Subject CANNOT Read a Data

Fig. 7. Example ISP pilot in policies in CNL for the validation.

Table 6
Table of attacking IP addresses.

Attacking hosts	Hits by BT	Hits by UniKent	Hits by GridPocket	Hits by collaboration
10.255.92.123	618			618
149.202.34.1			165	165
46.161.27.243			150	150
46.161.27.224			67	67
46.101.243.223			91	91
77.72.85.20			247	247
194.55.142.5			50	50
151.80.118.108			100	100
78.128.112.54			196	196
78.128.112.46			59	59
5.101.40.53			52	52
5.101.65.187			51	51
5.188.10.242			140	140
128.1.49.34			1513	1513
146.185.222.28			51	51
146.185.222.40			58	58
146.185.222.15			64	64
51.255.45.115			125	125
129.12.21.67		4482		4482
91.189.95.83	(48)		(2)	50

several months to run at the time of writing, and several features were either not integrated or only partially implemented, nevertheless the validation results show that the C3ISP architecture performs according to its design. Four different pilot were used in the validation, and each had its own requirements for trust and CTI data protection, resulting in different deployments for processing CTI data in either the edge and/or the cloud.

This architecture is successful because Data Sharing Agreement (DSA) policies are stuck to the CTI data, in Data Protected Objects (DPOs) and the DSA is enforced at either the edge or the cloud, or both, giving users confidence that their sensitive data will not leak to untrusted or partially trusted entities before it is appropriately sanitized. A limitation of the current design is that all DPOs that are shared for analysis have to have the same common DSA policy encapsulated in them. This means that all the cooperating organizations must agree on the common DSA to use with their CTI before the data sharing starts. We recognize the inconvenience of this, but must weight this against the complexity of trying to resolve conflicts between different DSAs that want to merge their CTI for analysis. Due to the complexity of the DSAs, comprising as they do of: Data Manipulation Operations, enhanced XACML policies to support usage control, and obligations to support policy inheritance, we believe that defining algorithms to support policy merging and conflict resolution is a complete research project of its own right.

Another current implementation limitation (though not a design limitation) is that local ISIs can only move DPOs to a single cloud based ISI, and a cloud based ISI cannot share its DPOs with other cloud based ISIs. Once this limitation is removed, the C3ISP platforms hosted by both the CERT and the Enterprise pilots (and ultimately by any other organization) will be able to share their CTIs and/or DPOs with each other for subsequent analysis.

Other limitations are that the C3ISP infrastructure is complex and large, and requires security experts to initiate the policy templates. Consequently, we envisage that specialist IT security organizations that process CTI on behalf of others are the only ones likely to have the specialist skills and knowledge necessary to be the operators of the infrastructure, and to run it as a service for others.

Finally, we have some implementation activities already scheduled on the short-term roadmap that are still to be completed. These include: completing the CTI sanitization functionality, adding an OpenID Connect federated identity management infrastructure for single sign on; implementing a Secure Audit Manager, most probably using the Elastic Stack open source solution; and adding additional analytics services to improve threat detection. Non-functional improvements include improved GUIs and increased performance.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 700294. We would like to thank all project partners for contributing to the project, including Fabio Martinelli, Stefano Tranquillini, and Thanh Hai Nguyen.

References

[1] S. Barnum, Standardizing cyber threat intelligence information with the structured threat information expression (stix^{TM}), MITRE Corp. 11 (2012) 1–22.

[2] C. Li, Y. Xue, J. Wang, W. Zhang, T. Li, Edge-oriented computing paradigms: A survey on architecture design and system management, ACM Comput. Surv. 51 (2) (2018) 39:1–39:34.

[3] W. Shi, J. Cao, Q. Zhang, Y. Li, L. Xu, Edge computing: Vision and challenges, IEEE Internet Things J. 3 (5) (2016) 637–646.

[4] J. Zhang, B. Chen, Y. Zhao, X. Cheng, F. Hu, Data security and privacy preserving in edge computing paradigm: Survey and open issues, IEEE Access 6 (2018) 18209–18237.

[5] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the internet of things, in: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, MCC'12, Vol. 1, ACM, New York, USA, 2012, pp. 3–16.

[6] L.M. Vaquero, L. Rodero-Merino, Finding your way in the fog: Towards a comprehensive definition of fog computing, SIGCOMM Comput. Commun. Rev. 44 (5) (2014) 27–32.

[7] T. Wang, G. Zhang, M.Z.A. Bhuiyan, A. Liu, W. Jia, M. Xie, A novel trust mechanism based on fog computing in sensor-cloud system, Future Gener. Comput. Syst. (2018).

[8] Benjamin C.M. Fung, Ke Wang, Rui Chen, Philip S. Yu, Privacy-preserving data publishing: A survey of recent developments, ACM Comput. Surv. 42 (4) (2010) 14, 53 pages.
[9] Elisa Bertino, Dan Lin, Wei Jiang, A survey of quantification of privacy preserving data mining algorithms, in: Privacy-Preserving Data Mining: Models and Algorithms, Springer US, 2008, pp. 183–205.
[10] L. Xu, C. Jiang, J. Wang, J. Yuan, Y. Ren, Information security in big data: privacy and data mining, in: IEEE Access, Vol. 2, 2014, pp. 1149–1176.
[11] OASIS eXtensible Access Control Markup Language (XACML) v3.0, 8 2012, available from http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cs02-en.doc.
[12] E. Carniani, D. D'Arenzo, A. Lazouski, F. Martinelli, P. Mori, Usage control on cloud systems, Future Gener. Comput. Syst. 63 (C) (2016) 37–55, http://dx.doi.org/10.1016/j.future.2016.04.010, (October 2016).
[13] M. Henze, R. Hummen, R. Matzutt, K. Wehrle, A trust point-based security architecture for sensor data in the cloud, in: Trusted Cloud Computing, Springer, 2014, pp. 77–106.
[14] S. Bugiel, S. Nurnberger, A.-R. Sadeghi, T. Schneider, Twin clouds: Secure cloud computing with low latency, in: Communications and Multimedia Security, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 32–44.
[15] S. Pearson, M.C. Mont, L. Chen, A. Reed, End-to-end policy-based encryption and management of data in the cloud, in: 2011 IEEE Third International Conference on Cloud Computing Technology and Science, 2011, pp. 764–771.
[16] F. Martinelli, A. Saracino, M. Sheikhalishahi, Modeling privacy aware information sharing systems: A formal and general approach, in: 2016 IEEE Trustcom/BigDataSE/ISPA, 2016, pp. 767–774.
[17] K.D. Bowers, A. Juels, A. Oprea, Hail: A high-availability and integrity layer for cloud storage, in: Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS '09, ACM, New York, USA, 2009, pp. 187–198.
[18] S. Kamara, K. Lauter, Cryptographic cloud storage, in: Financial Cryptography and Data Security, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 136–149.
[19] G. Danezis, B. Livshits, Towards ensuring client-side computational integrity, in: Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop, CCSW '11, ACM, New York, NY, USA, 2011, pp. 125–130.
[20] US-CERT: (https://www.us-cert.gov/).
[21] P. Mel, T. Grance, NVD national vulnerability database. National Institute of Standards and Technology .http://nvd.nist.gov (2007).
[22] MITRE Common Weakness Enumeration (http://cwe.mitre.org/).
[23] Common Vulnerabilities and Exposures (http://cve.mitre.org/).
[24] NIST IR 7435, The Common Vulnerability Scoring System (CVSS) and Its Applicability to Federal Agency Systems, 2007.
[25] BT Security Monitoring. https://www.globalservices.bt.com/btfederal/en/products/security-threat-monitoring.
[26] C. Caimi, C. Gambardella, M. Manea, M. Petrocchi, D. Stella, Legal and technical perspectives in data sharing agreements definition, in: Annual Privacy Forum, Springer, Cham, 2015, pp. 178–192.
[27] CIF. The Collective Intelligence Framework. https://code.google.com/p/collective-intelligence-framework/.
[28] Chenfeng Vincent Zhou, Christopher Leckie, Shanika Karunasekera, A survey of coordinated attacks and collaborative intrusion detection, Comput. Secur. 29 (1) (2010) 124–140.
[29] C. Lo, C. Huang, J. Ku, A cooperative intrusion detection system framework for cloud computing networks, in: 2010 39th International Conference on Parallel Processing Workshops, San Diego, CA, 2010, pp. 280–284.
[30] X. Shu, D. Yao, E. Bertino, Privacy-preserving detection of sensitive data exposure, IEEE Trans. Inf. Forensics Secur. 10 (5) (2015) 1092–1103.
[31] EU PROTECTIVE project. See https://cordis.europa.eu/project/rcn/202674_en.html.
[32] David C. Mills, Kathy Wang, Brendan Malone, Anjana Ravi, Jeffrey Marquardt, Anton I. Badev, Timothy Brezinski, Lind a Fahy, Kimberley Liao, Vanessa Kargenian, Max Ellithorpe, Wendy Ng, Maria Baird, Distributed Ledger Technology in Payments, Clearing, and Settlement (2016-12). FEDS Working Paper No. 2016-095.
[33] X. Liang, J. Zhao, S. Shetty, J. Liu, D. Li, Integrating blockchain for data sharing and collaboration in mobile healthcare applications, in: 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, 2017, pp. 1–5.
[34] H. Es-Samaali, A. Outchakoucht, J.P. Leroy, A blockchain-based access control for big data, Int. J. Comput. Netw. Commun. Secur. 5 (7) (2017) 137147.
[35] V. Buterin, Ethereum : A Next-Generation Smart Contract and Decentralized Application Platform, 2014.
[36] S. Wang, Y. Zhang, Y. Zhang, A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems, IEEE Access 6 (2018) 38437–38450.
[37] EC C3ISP Project, see https://c3isp.eu/.
[38] OASIS STIX™ Version 2.0. Part 1: STIX Core Concepts. Committee Specification 01, 19 2017. Available from http://docs.oasis-open.org/cti/stix/v2.0/cs01/part1-stix-core/stix-v2.0-cs01-part1-stix-core.docx.
[39] C. Dwork, Differential privacy, in: Proc ICALP 2006, Part II, Springer, 2006.
[40] M.E. Andrés, N.E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, Geo-indistinguishability: Differential privacy for location-based systems. arXiv preprint arXiv:1212.1984 (2012).
[41] J. Park, R. Sandhu, The UCON ABC usage control model, ACM Trans. Inf. Syst. Secur. 7 (1) (2004) 128–174.
[42] X. Wang, I. Herwono, F.D. Cerbo, P. Kearney, M. Shackleton, Enabling cyber security data sharing for large-scale enterprises using managed security services, in: 2018 IEEE Conference on Communications and Network Security (CNS), Beijing, 2018, pp. 1–7, http://dx.doi.org/10.1109/CNS.2018.8433212.
[43] BT Cyber Security Platform. https://www.globalservices.bt.com/en/solutions/products/cyber-security-platform.
[44] SAP Enterprise Threat Detection. https://www.sap.com/uk/products/enterprise-threat-detection.html.
[45] McAfee Enterprise Security Manager. https://www.mcafee.com/us/products/enterprise-security-manager.aspx.
[46] Alien Vault Unified Security. https://www.alienvault.com.
[47] V. Basili, D. Weiss, A methodology for collecting valid software engineering data, IEEE Trans. Softw. Eng. (1984).

David W Chadwick, BSc, PhD is Professor of Information Systems Security at the University of Kent. He has published widely, with over 150 publications in international journals, conferences and workshops. He is actively involved in standards' meetings, being the BSI lead representative to ISO/ITU-T X.500, the author of 2 Internet RFCs and an invited expert to the W3C Verifiable Credentials Working Group.

He specializes in identity management, policy-based authorization, privacy protection, the management of trust, cloud security and Internet security in general. Current research topics include: autonomic authorization, verifiable (self-sovereign) credentials, natural language specification of authorization policies, sticky policies and federated authorization.

Wenjun Fan is a postdoctoral researcher of cyber security at the University of Kent, Canterbury, United Kingdom. He received a Ph.D. degree in telematics engineering from the Universidad Politécnica de Madrid, Madrid, Spain, in 2017. His research interests include: cyber security, network softwarization, cloud computing, and machine learning.