Enabling Privacy-preserving Sharing of Cyber Threat Information in the Cloud

Wenjun Fan[1+], Joanna Ziembicka[1], Rogério de Lemos[1], David Chadwick[1], Francesco Di Cerbo[2], Ali Sajjad[3], Xiao-Si Wang[3] and Ian Herwono[3]
[1+]*School of Computing, University of Kent, Canterbury, United Kingdom, CT2 7NZ*
Email: {w.fan, j.i.ziembicka, r.delemos, d.w.chadwick}@kent.ac.uk
[2]*SAP Security Research, 805 av Dr Maurice Donat, 06254 Mougins, France*
Email: francesco.di.cerbo@sap.com
[3]*Security Futures Practice, Applied Research, British Telecommunications plc, Ipswich, United Kingdom, IP5 3RE*
Email: {ali.sajjad, selina.wang, ian.herwono}@bt.com

***Abstract*—Network threats often come from multiple sources and affect a variety of domains. Collaborative sharing and analysis of Cyber Threat Information (CTI) can greatly improve the prediction and prevention of cyber-attacks. However, CTI data containing sensitive and confidential information can cause privacy exposure and disclose security risks, which will deter organisations from sharing their CTI data. To address these concerns, the consortium of the EU H2020 project entitled Collaborative and Confidential Information Sharing and Analysis for Cyber Protection (C3ISP) has designed and implemented a framework (i.e. C3ISP Framework) as a service for cyber threat management. This paper focuses on the design and development of an API Gateway, which provides a bridge between end-users and their data sources, and the C3ISP Framework. It facilitates end-users to retrieve their CTI data, regulate data sharing agreements in order to sanitise the data, share the data with privacy-preserving means, and invoke collaborative analysis for attack prediction and prevention. In this paper, we report on the implementation of the API Gateway and experiments performed. The results of these experiments show the efficiency of our gateway design, and the benefits for the end-users who use it to access the C3ISP Framework.**

Keywords—Cyber Threat Information; Privacy Preserving; Data Sharing; Collaborative Analysis; API Gateway

I. INTRODUCTION

Computer networks confront increasingly crafty and complex security threats that are hard to detect and prevent. In particular, complex large-scale cyber-security threats often affect multiple network domains. For example, the most recent large-scale security incident, WannaCry ransomware attack [1], has affected more than 150 countries and caused financial losses at a tremendous level. To mitigate these threats, cyber organisations like Internet Service Providers (ISP) and Cloud Service Providers (CSP), make use of a number of typical security surveillance and protection tools, like firewalls, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). The logs and reports from these security tools are used to generate Cyber Threat Information (CTI). CTI is any information that can help an organisation identify, assess, monitor, and respond to cyber threats [2]. It includes indicators of compromise; tactics, techniques, and procedures used by threat actors; suggested actions to detect, contain, or prevent attacks; and the findings from the analyses of incidents. Organisations that share CTI can improve their own security profile as well as those of other organisations.

However, the foremost challenge of CTI sharing is establishing trust and safeguarding sensitive information among the sharing parties [2]. CTI data often contains sensitive and confidential information, therefore, sharing of plain text data might cause privacy exposure, and disclose unknown security holes if the CTI is shared with untrusted or unscrupulous partners. These risks deter and discourage organisations from sharing their CTI data. For example, if an organisation is willing to share its CTI data with other organisations in order to better identify remote attacking systems, this also introduces the unintended side-effect of its own vulnerable systems being identified to its collaborating partner organisations.

Hence, before sharing the CTI, it must be sanitised by some privacy-preserving means, such as pseudonymisation, anonymisation, homomorphic encryption, etc. Although these approaches will increase the privacy level, they will also accordingly decrease the data utility, i.e., adversely affect the accuracy of the data analysis. Therefore, a vitally important point for privacy-preserving CTI data sharing is to reach a trade-off between the data utility and data privacy. The C3ISP Project [3] was established to address this very issue, by defining a collaborative and confidential information sharing, analysis and protection framework as a service for cyber threat management. The strategy of the C3ISP Framework uses the trust level of the CTI-sharing infrastructure to determine the appropriate privacy method. For example, if the infrastructure is fully trusted, CTI may be shared in plaintext; if the infrastructure is semi-trusted, the data may be pseudonymised or anonymised; and if the infrastructure is not trusted, homomorphic encryption may be selected, despite its high processing overhead.

The API Gateway, presented in this paper, provides a bridge between end-users and their data sources, and the C3ISP Framework. The target of the API Gateway is the end-user, and the objective is to facilitate how the user retrieves, shares and analyses CTI. In this work, we call this kind of end-user Prosumer (i.e. producer and consumer). So, the API Gateway provides an easy way to allow the user to retrieve CTI data,

regulate data sharing agreements (DSAs) for data protection, share the data using privacy-preserving methods, and invoke collaborative analysis for attack prediction and prevention.

This paper proposes a novel API Gateway architecture based on our prior work. In [4], the authors presented an architecture for privacy-preserving sharing of CTI with third party analysis services. Indeed, that is the prior version of our gateway architecture, and it was only used for the Small-Medium Enterprise (SME) pilot[1] to interact with the C3ISP Framework. Later, another paper [5] presents a design of two functional components, i.e. Data Manager (DM) and Collaborative Task Manager (CTM) for the large-scale Enterprise (ENT) pilot of C3ISP Framework. The two original designs of the gateway addressed the specific needs of two pilots. This paper presents a novel API Gateway architecture, which is more modular, domain-agnostic and flexible while addressing the needs of both application domains. Hence, though it is implemented against the C3ISP Framework, but it is not dependent on it. The contribution of this paper can be summarized as follows:

- An effective means of privacy-preserving sharing and analysing CTI in the cloud.
- A flexible architecture of an API Gateway, which can be easily tailored to various application domains.

The remainder of this paper is organized as follows: Section II will briefly describe the C3ISP Framework and in particular, propose the C3ISP Gateway architecture; Section III will show a prototype implementation and deployment for validating the design; Section IV will conduct some tests and show the experimental results; Section V will present the literature review; finally, Section VI will derive conclusions and suggest some future work.

II. Design of the API Gateway

In this section, we first describe the high level design of the SME pilot of the C3ISP Project [3]. Thereafter, we propose a flexible architecture of API Gateway (termed C3ISP Gateway in this paper), which has been subsequently implemented for both the SME and Enterprise pilots. Finally, we present the hybrid deployment mode used in the SME pilot to clearly delineate trust relationships related to CTI data sharing.

A. SME Pilot Overview

C3ISP aims to implement privacy-preserving CTI data sharing by providing a set of flexible mechanisms, regulated by data sharing agreements, which allow owners to retain control of what is shared and to protect the information in the most appropriate way depending on the usage scenario. The main innovation of C3ISP is the possibility to share information in a flexible and controllable manner inside a collaborative multi-domain environment to improve detection of cyber threats and response capabilities, while still preserving the confidentiality of the shared information. The C3ISP Project consists of four pilots, CERT pilot, ISP pilot, ENT pilot and SME pilot. Each pilot makes use of a set of pilot-specific components that interact with the components of the C3ISP Framework. This work concentrates on the SME pilot. Fig. 1 shows the high level design of the SME pilot. We make use of the Fundamental Modeling Concepts (FMC) [6] notation to construct the block diagrams.

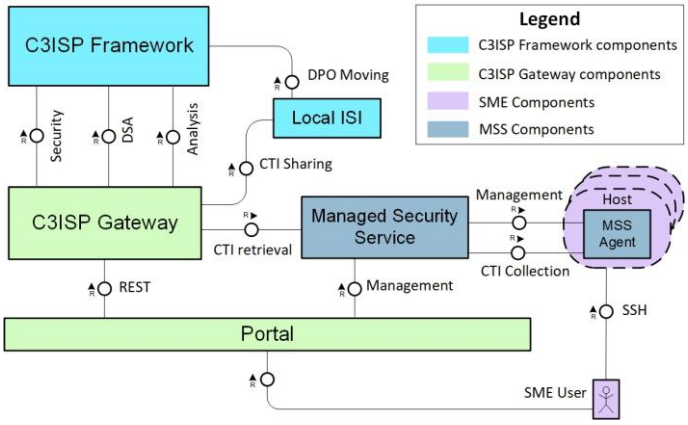

Fig. 1. High Level Design of the SME Pilot: the symbol R▶ indicates the direction of request, which results in a response in the opposite direction

C3ISP Framework enables SMEs to share CTI, collected by different C3ISP partners and stakeholders, for running collaborative analytics. The C3ISP Gateway retrieves CTI data from the SME's Managed Security Service (MSS) and uploads it to the C3ISP Framework for sharing and analysis. Through an easy to use web interface (Portal), the SME user is able to manage all of their C3ISP related tasks, i.e., choosing which CTI data to share, creating and selecting Data Sharing Agreements (DSA), and running collaborative analytics.

The C3ISP Framework comprises several subsystems (that are not shown in Fig. 1, because from the pilot perspective, users only invoke the API and web interface of these subsystems, whose inner working and design details are transparent to the user). These subsystems are:

1) *Data Sharing Agreement (DSA) Manager*

It allows users to define the policies that regulate the CTI data sharing, including defining rules of access and usage control on shared data, on analytic services and results and rules to handle data manipulation operations.

2) *Information Sharing Infrastructure (ISI)*

It allows users to exchange CTI data under the constraints specified in the DSA policies and acts as a storage of CTI data for controlled access by analytics services. The ISI API is used to manage the external communication with the others C3ISP subsystems. Note: a local ISI can be used to offload the data processing from the centralized infrastructure.

3) *Information Analysis Infrastructure (IAI)*

It allows users to request the execution of analytics services on the data protected and shared by the ISI. It supports both *C3ISP-aware* analytics services, jobs that can exploit the full capabilities of the C3ISP Framework, and *legacy analytics* service (i.e. pre-existing analytics incorporated into the framework), that can run on the shared data but have limitations with respect to data protection. The IAI API

[1] Note that in the context of this paper, the term "pilot" refers to the specific application domain, e.g., SMEs or Enterprises.

provides an interface for external interaction with the users (or their applications) and other C3ISP subsystems.

4) *Common Security Services (CSS)*

The CSS consists of Identity Manager (IM), Key and Encryption Manager (KEM), and Secure Audit Manager (SAM), which are available to all the subsystems and their components in order to satisfy security requirements.

B. C3ISP Gateway Architecture

As aforementioned, C3ISP Gateway is an interface between the SME environment and the centralised C3ISP Framework. Essentially, it complies with the API Gateway pattern of the Micro-services Architecture paradigm[2]. The C3ISP Gateway can be attached to various C3ISP Framework-like micro-services for data retrieving, sharing and analysis, which have very clear API identified. The C3ISP Gateway is the entry point for all Prosumers. It handles requests in one of two ways: some requests are simply proxied/routed to the appropriate service; other requests are fanned out to multiple services.

To allow for adaptable as well as domain-agnostic design, the C3ISP Gateway has a flexible architecture that can accommodate a variety of CTI data sources and various APIs that are fundamental for sharing and analysis of CTI data. As a concrete example, this design has allowed shared development of core components between the Enterprise and SME pilots, and also allows easy adaptation for use with other pilots/CTI data sources in the future. Fig. 2 using FMC notation [6] shows the detailed component architecture of the C3ISP Gateway. Note that it is not limited to this application domain, but can be tailored and configured to adapt to other ones. The following discussion will present the details of the components.

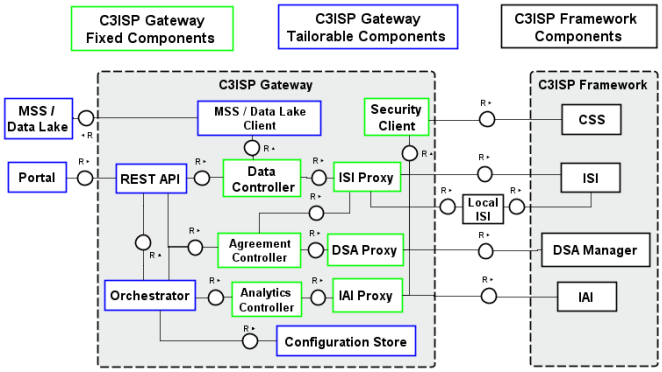

Fig. 2. C3ISP Gateway Architecture: the green components are fixed across all pilots, while and the blue need to be modified according to specific application domain.

1) *SME MSS*

The Managed Security Service (MSS) is a third-party security service, which enables its customers (the SMEs) to assess the security threats and vulnerabilities of data and applications they run in physical or virtual machines hosted on many different kinds of computing infrastructures or platforms. It stores CTI data in the form of event logs, which can be viewed in a web-based user interface, or retrieved by an API for further processing. In the case of the SME Pilot, the MSS plays the role of data source. The ENT Pilot, in turn, uses a Data Lake as its data source.

2) *MSS Client*

The MSS Client provides access to CTI data collected by the MSS. It handles authentication with the MSS, translates client queries from those provided through the REST API into MSS-specific input, and uses them to retrieve CTI data from the MSS. It then packages the CTI data into a CSV-based format understood by the C3ISP Gateway, and generates a subset of CTI metadata based on the retrieval criteria and data characteristics. The self-contained nature of this module allows it to be exchanged for another Data Source client (for example, one accessing a Data Lake or database instead of the MSS), without any other changes to the C3ISP Gateway. In the case of ENT Pilot, the Data Lake client uses a different query language and data retrieval protocol to import CTI data. The current implementation of the SME MSS client supports collection of two CTI event types: Firewall and Anti-malware events.

3) *REST API*

The REST API provides a programmable remote interface. It is used by the Portal to interact with the C3ISP Gateway. The REST API methods and their signatures are consistent across the pilots sharing the C3ISP Gateway codebase (i.e., the SME and ENT Pilots at the present time).

4) *Portal*

The Portal is a graphical interface that allows the SME user to interact with the C3ISP Framework via the C3ISP Gateway. It communicates with the C3ISP Gateway using the Gateway's REST API. Through the Portal, the SME user is able to import MSS data according to selection criteria, select DSAs to associate with CTI data, trigger analytics, and access analytics results. Additionally, the Portal provides the SME users with links to external tools such as the MSS Manager Web portal and the C3ISP DSA Editor.

Arguably, a unified API Gateway that provides a common user-friendly interface to all pilots will be more efficient than one consisting of two applications (the Portal and the Gateway) connected by a REST API. However, the current C3ISP Gateway is required by not only SME pilot but also ENT pilot, with other pilots possibly requiring its use in the future. Each pilot has its own custom Portal and only needs the Gateway's inner working components for accomplishing its tasks. Therefore, the current design provides flexibility and modularity to the Gateway, which by exposing the REST API makes it compatible for different Portal applications belonging to different pilots.

5) *Proxies*

The C3ISP Gateway comprises three separate proxies for interacting with C3ISP Framework: the DSA Proxy manages RESTful interactions of the C3ISP Gateway with the DSA Store API on the C3ISP Framework. It retrieves the URI for the DSA Store from a configuration file; the ISI Proxy manages RESTful interactions of the C3ISP Gateway with the ISI API on both the Local ISI and the central ISI. It retrieves the URIs for both ISIs from a configuration file; the IAI Proxy manages RESTful interactions of the C3ISP Gateway with the

[2] https://microservices.io/patterns/apigateway.html

IAI API on the C3ISP Framework. It retrieves the URI for the IAI from a configuration file.

6) *Controllers*

Correspondingly, the C3ISP Gateway includes three separate controllers: the Agreement Controller manages functionality related to the Data Sharing Agreements via the ISI Proxy (for DSA search) and DSA Proxy (for CRUD operations); the Data Controller manages the retrieval, packaging and sharing of CTI data. It retrieves CTI data from the MSS, sanitises it via the Local ISI, and imports it into the C3ISP Framework using the ISI Proxy; the Analytics Controller manages requests for analytics on the central C3ISP Framework via the IAI Proxy.

7) *Orchestrator*

The Orchestrator stores and executes workflows performed by the C3ISP Gateway. Those workflows are then exposed by the C3ISP Gateway REST API. The design also allows for the Orchestrator to handle scheduling of workflows.

8) *Configuration Store*

The Configuration Store stores stateful information about the C3ISP Gateway. Within the SME pilot's scope, the Configuration Store keeps track of default DSAs assigned to each type of CTI event. Configuration Store will additionally store Orchestrator's scheduling information.

9) *Security Client*

The Security Client handles user authentication and identity management by interacting with the C3ISP Framework CSS component.

III. Implementation and Deployment

The C3ISP Gateway has been implemented using the micro-services paradigm based on the Spring Boot framework [7]. It exposes a REST API [8] as its entry point, and also communicates with C3ISP Framework components using REST. It communicates with the MSS using a SOAP API [9]. Further, a graphical web interface – the Portal – has been developed that communicates with the C3ISP Gateway through REST as well.

The SME Pilot applies the hybrid deployment model of the C3ISP Framework, which means that the ISI is deployed both locally and centrally. The Local ISI is deployed within the SME's and C3ISP Gateway's trust domain, and applies the DSA to CTI data before uploading it to the centralised ISI. This deployment model ensures that unprotected, sensitive or unauthorised CTI data never leaves the SME's trust domain.

For the SME Pilot, the gateway fulfils the role of a Prosumer. Hence, the local trust domain hosts a Local ISI, which sanitises the CTI data according to the DSA before sharing it with the central C3ISP Framework. The Local ISI is used solely for processing data rather than as local CTI storage. The central C3ISP Framework collects and aggregates CTI from different sources and performs different sorts of threat and vulnerability analytics on the combined data to produce useful results and reports. The centralised ISI and IAI subsystems are deployed in the central C3ISP infrastructure, which lie outside the trust domain of the SMEs.

The prototype of SME pilot has been deployed in a distributed cloud environment, with some components deployed in the shared and central C3ISP infrastructure and others in the individual SME's premises. TABLE I. shows the shared C3ISP Framework components that have been deployed. They were deployed in the testbed that comprised 22 micro-services distributed among them.

TABLE I. Shared components

Component	CPU	RAM	DISK	Hosted by
MSS	4 Core AMD Opteron @ 2.30 GHz	16 GB	500 GB	BT
ISI (Central)	4 Core Intel Xeon @ 2.30GHz	12 GB	100 GB	CNR
IAI	8 Core Intel Xeon @ 2.30GHz	16 GB	400 GB	CNR
DSA Manager	2 Core AMD Opteron @ 2,493GHz	4 GB	40 GB	CNR
SAM	1 Core AMD Opteron @ 2,493GHz	2 GB	22 GB	CNR
KEM	8 Core Intel Xeon @ 2.30GHz	16 GB	100 GB	CNR
IM	1 Core AMD Opteron @ 2,493GHz	2 GB	22 GB	CNR

The MSS Clients are deployed in the individual SME's virtual machines (VMs). TABLE II. lists the testbeds used by different SMEs for hosting the clients in order to capture data. Due to the attack-accessible network environment provided by the SMEs, we were able to gather real attacking data.

TABLE II. Organisations' testbeds

Oranization	OS information	Host Information
CHINO	Amazon Linux	2x VMs hosted by AWS
GPS	Ubuntu 16.04	2x VMs hosted by OVH
3DRepo	CentOS Linux 7	5 x VMs hosted by Google Cloud

In order to complete the CTI data sharing workflow, we also defined and created an applicable DSA, which includes two authorization policies as follows:

IF a Subject hasOrganisation a Organisation(UNIKENT) AND that Subject isMemberOf a Group(SecurityAnalyst) THEN that Subject CAN Read a Data

IF a Subject hasOrganisation a Organisation(UNIKENT) AND that Subject isMemberOf a Group(SecurityAnalyst) THEN that Subject CAN invokeMaliciousHostsAnalysis a Data

One policy identifies which user can read the data if the user belongs to an authorized group. The other one specifies who can invoke analytics service to analyse the data if the user belongs to an authorized group. The experimental results regarding the CTI data sharing and analysis workflows will be shown in the next section.

IV. Experimental Validation

The SME pilot is used to validate the operation of the C3ISP Framework through the C3ISP Gateway. We used primarily a combination of simulated attack data and passive test environment data for experiments. The simulated attack data included Firewall and Anti-Malware events triggered by

the tester through attack simulation scripts. Passive test environment data included CTI events encountered during normal operations by an SME host. We also defined a comprehensive set of functional and non-functional acceptance tests (ATs), based on their requirements for the confidential sharing and analysis of CTI data. Out of 22 originally defined acceptance tests, 10 Passed, 6 Partially Passed, and 6 were not applicable (N/A). In general, the tests that related to the basic sharing of data and to MSS Server management tended to pass successfully. While the C3ISP Framework is still under development, and its full suite of analytics/data protection tools was not available at the time of testing, we were merely able to validate our fully-functional SME pilot components against the basic end-to-end functionality of the framework prototype.

For evaluation of collaborative analysis capability of the C3ISP Framework, we developed a basic analytics service for the SME pilot, termed findAttackingHosts. It analyses all the shared firewall CTI data, in accordance to the applied DSA, and lists the IP addresses of attacking hosts. In this case, the data was composed of three sets of CTI (collected on March 5th, 2019) from 3DRepo, CHINO and GPS, and a combined set (that contains all the former three individual data sets). We ran the analysis over each organisation's own data separately, and then ran the analysis over the collaboratively shared data as well. Some statistic information of these data sets is shown in TABLE III.

TABLE III. STATISTIC INFORMATION OF THE DATA SETS

Statistic information	3DRepo data set	CHINO data set	GPS data set	Collaborative data set
Amount of connection attempts	6811	2788	2868	12467
Amount of attacking hosts	938	1796	783	3135

In each data set, one entry represents one connection attempt. Hence, 3DRepo's data set contains the most connection attempts among the three data sets. Also, we observed that a large number of hosts only launched one connection attempt, while some have issued multiple attacks. CHINO's data set discloses the largest number of attacking hosts. Further, we found that some attacking hosts had attempted to access more than one SME, so their connection attempts were logged by multiple organisations. That is why the amount of attacking hosts disclosed by the collaborative data set is less than the sum of the individual amount.

Owing to the SMEs running their VMs as honeypots [10], any connection attempt to them is suspicious. Honeypots are often used to investigate new attacks by filtering repeated and uninteresting connections [11]. We did find that the collaborative analysis results reveal more new attacks than the individual analysis results. TABLE IV. shows two examples, both new attacking hosts can be revealed by the collaborative data set, while the first one was missed by CHINO and GPS, and the second one was missed by 3DRepo and CHINO.

TABLE IV. COMPARISON OF DETECTING NEW ATTACKS

Attacking Hosts	Hits by 3DRepo	Hits by CHINO	Hits by GPS	Hits by Collaboration
190.254.122.125	1	0	0	1
105.247.141.227	0	0	1	1

Another case in point is to detect suspicious scanners, which means an attacking host attempts to connect multiple destinations within a time interval [12]. This is hard to conduct by only using an individual data set since it only contains one destination. However, by conducting collaboratively analysis, it is effective to unveil the suspicious scanners. TABLE V. lists several ones. All of the listed hosts are very suspicious since they had tried to access different organizations multiple times within one day.

TABLE V. DETECTING SCANNERS BY USING COLLABORATIVE ANALYSIS

Attacking Hosts	Hits by 3DRepo	Hits by CHINO	Hits by GPS	Hits by Collaboration
185.176.27.106	48	11	38	97
185.176.27.6	36	7	34	77
185.176.27.246	36	5	33	74
185.176.27.2	36	2	34	72

The other benefit is that one SME that has already been attacked can prevent the same attack from occurring on other SMEs, by notifying them in advance. TABLE VI. shows one instance, where an attacking host launched thousands connection attempts against 3DRepo, which might be a DoS flooding like attack. The other SMEs can blacklist this malicious IP in advance in order to prevent the same attack.

TABLE VI. PREVENTING THE SAME ATTACK IN ADVANCE

Attacking Hosts	Hits by 3DRepo	Hits by CHINO	Hits by GPS	Hits by Collaboration
169.254.169.254	4187	0	0	4187

Therefore, through performing the C3ISP analysis service over collaboratively shared data, novel malicious hosts can be revealed and the results are mutually beneficial for all the SMEs. In addition, Fig. 3 shows the performance of running the analysis over different data sets for reference. The collaborative analysis has the greatest time cost since it runs over the largest data set.

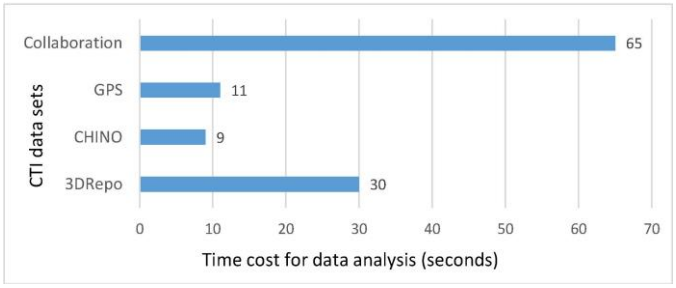

Fig. 3. Performance of CTI data analysis

Furthermore, the main aim of this effort was to provide the SME users with an easy-to-use solution for accessing the services of the C3ISP Framework. During the course of this experiment's workflow, the SME users were able to utilise the various micro-services of the C3ISP Framework in the following order: 1. DSA search and selection; 2. CTI retrieval; 3. CTI processing; 4. CTI sharing; 5. CTI analysis and results.

As all of these micro-services were exposed to the SME users through the C3ISP Gateway, which was deployed inside their respective trusted domains. A non-functional benefit of

this approach is its easier adoption by the SMEs, which might otherwise be reluctant to participate in a collaborative data sharing environment.

V. Related Work

A. Privacy-preserving Data Sharing in Cloud.

Data protection/security is a typical issue in cloud computing. Sharing data in the cloud may involve leakage of private and sensitive data [13], which should avoid being shared with other untrusted partners. If compromised, the data could be used by the adversary to launch attacks and create unexpected risk. Hence, in order to get a trade-off between the data utility and privacy, the privacy-preserving data publishing approach [14] for data sharing is a vitally important point. Plenty of research effort [15] has been undertaken in the area of preserving privacy. Recently, it has also become an important research aspect in the area of Big Data processing [16]. Generally speaking, privacy-preserving aims at extracting relevant knowledge from large amounts of data while at the same time preventing exposure of sensitive information. Research in this field has devoted much effort to determine a balance between the right to privacy and the need for knowledge discovery.

In practice, different security requirements and metrics can often lead to using distinct privacy-preserving data mining techniques [17]. Thus, it is desirable to have a flexible privacy-preserving mechanism that can be customized in terms of different requirements to approach a balance between data privacy and the data utility. The work [18] proposed an authorization service based on the Usage Control (UCON) model as well as U-XACML (an extension of XACML for Usage Control) to regulate the usage of resources in Cloud IaaS services, which was integrated within OpenNebula [3]. The UCON model, was introduced alongside a formal data sharing control mechanism termed Data Sharing Agreement (DSA) [19]. A DSA is a human readable, yet machine-processable contract, regulating how organisations share data. The concept of DSA was first applied in the CoCo Cloud Project [20], while the C3ISP Framework has adapted it for use with CTI data.

B. CTI Data Sharing for Collaborative Analysis

STIX [21], a standard information-exchange language, has been proposed for the purpose of CTI data sharing. STIX provides a common mechanism for addressing structured cyber threat information across and among a full range of use cases improving consistency, efficiency, interoperability, and overall situational awareness. Additionally, STIX provides a unifying architecture that is able to tie together a diverse set of cyber threat information, e.g. US-CERT [22], CVE [23] and CWE [24]. In addition to STIX, there are some other ontologies that attempt to enable sharing of CTI, such as IODEF [25] and OpenIOC [26]. Burger et al. [27] proposed a taxonomy for classifying these CTI sharing technologies by using an agnostic framework, in order to identify gaps in existing technologies and explain their differences from a scientific point view.

[3] https://opennebula.org/

CYBEX (Cybersecurity Information Exchange Framework) [28] describes how cybersecurity information is exchanged between cybersecurity entities on a global scale and how the exchange is assured. CYBEX does not depend on STIX but rather on IODEF for describing information. It aims to provide a service of structured information exchange about measurable security states of systems, together with incidents stemming from cyberattacks. Later, D. Tosh et al. [29] proposed an evolutionary game-theoretic framework for CTI sharing that can guide: (i) the organisation to independently decide whether to "participate in CYBEX and share" or not; (ii) the CYBEX framework to utilize the participation cost dynamically as incentive (to attract firms toward self-enforced sharing) and as a charge (to increase revenue).

Furthermore, the notion of collaborative CTI data sharing and analysis is applied in the Collaborative Intrusion Detection Systems (CIDS), which are designed to analyse threat information from multiple networks simultaneously for the purpose of detecting the attacks at an early stage and before they have caused signification impact on the Internet. Taking advantages of the CIDS, a global view of the suspicious events on the monitored targets [30] [31] can be gained and the ratio of false alarms [32] can be reduced. Zhou et al. [33] surveyed the collaborative intrusion detection approaches in cope with the emergence of coordinated attacks. Such attacks (e.g. large-scale scans, worm outbreaks and DDoS attacks) often occur in multiple networks simultaneously. The authors stress that the main research challenges are alert correlation algorithms and CIDS architectures that were categorised into centralised architecture, hybrid architecture and fully distributed architecture, which are similar to the deployment models of C3ISP infrastructure. For instance, two works [34][35] proposed using CIDS to detect DDoS attacks for Cloud Computing, whereby one region's IDS can share its alert data with the other IDS systems. This helps to reduce computational cost for detecting the same attacks at other IDS systems and therefore improves detection rate in overall cloud environment. The difference is the paper [34] rides on the fully distributed architecture while the paper [35] uses the hybrid architecture.

VI. Conclusions and Future Work

This paper presents a means of privacy-preserving sharing of Cyber Threat Information (CTI). In particular, it proposes a flexible architecture of API Gateway that facilitates the retrieving, sharing and analysis of CTI data, which can be easily tailored to specific application domains. In this paper, we show how the C3ISP Gateway can be integrated to the C3ISP Framework, thus allowing end-users to fully take advantage of the C3ISP Framework collaborative analytics. The flexible architecture of the C3ISP Gateway enables different C3ISP pilots to share multiple common components, which reduces pilot development costs, and facilitates the ease of replacing or augmenting the Managed Security Service (MSS) data with other data sources.

For future work, firstly, we would like to extend the use of the C3ISP Gateway to the other pilots, resulting in a unified joint C3ISP Gateway architecture. Secondly, we plan to conduct more performance-focused validation of the C3ISP Framework, especially for CTI data sharing and analysis

workflows. Lastly, we plan to develop more advanced, privacy-aware, collaborative analytics for the SME pilot with the purpose of exploiting the potential of collaborative data sharing within C3ISP Framework.

ACKNOWLEDGMENT

This work was supported by the European Union's Horizon 2020 research and innovation programme through the C3ISP Project under Grant 700294.

REFERENCES

[1] CERT-EU Security Advisory, "WannaCry Ransomware Campaign Exploiting SMB Vulnerability," May 22, 2017, available on: https://cert.europa.eu/static/SecurityAdvisories/2017/CERT-EU-SA2017-012.pdf.

[2] Johnson, Chris, Lee Badger, David Waltermire, Julie Snyder, and Clem Skorupka. "Guide to cyber threat information sharing." NIST special publication 800, 150, 2016.

[3] EC C3ISP Project, see https://c3isp.eu/

[4] F. Giubilo, A. Sajjad, M. Shackleton, D. W. Chadwick, W. Fan and R. de Lemos, "An architecture for privacy-preserving sharing of CTI with 3rdparty analysis services," *2017 12th International Conference for Internet Technology and Secured Transactions (ICITST)*, Cambridge, 2017, pp. 293-297.

[5] X. Wang, I. Herwono, F. D. Cerbo, P. Kearney and M. Shackleton, "Enabling Cyber Security Data Sharing for Large-scale Enterprises Using Managed Security Services," *2018 IEEE Conference on Communications and Network Security (CNS)*, Beijing, 2018, pp. 1-7.

[6] R. Alpfelbacher, "FMC Visualization Guidelines," 2017. [Online]. Available: http://www.fmc-modeling.org/notation_reference/. [Accessed 30 September 2017].

[7] Pivotal Software, "Spring Boot, Simplifying Everything," Pivotal Software, 1 October 2002. [Online]. Available: https://spring.io/projects/spring-boot. [Accessed 1 December 2018].

[8] H. Zhao, "RESTful API Design Specification," ONAP, 18 January 2018. [Online]. Available: https://wiki.onap.org/display/DW/RESTful+API+Design+Specification. [Accessed 1 December 2018].

[9] Y. L. Nilo Mitra, "SOAP Specification version 1.2," W3C, 27 April 2007. [Online]. Available: http://www.w3.org/TR/soap12. [Accessed 1 December 2018].

[10] W. Fan, Z. Du, D. Fernández and V. A. Villagrá, "Enabling an Anatomic View to Investigate Honeypot Systems: A Survey," in *IEEE Systems Journal*, vol. 12, no. 4, pp. 3906-3919, Dec. 2018.

[11] W. Fan, Z. Du, M. Smith-Creasey and D. Fernández, "HoneyDOC: An Efficient Honeypot Architecture Enabling All-Round Design," in *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 3, pp. 683-697, March 2019.

[12] Jaeyeon Jung, V. Paxson, A. W. Berger and H. Balakrishnan, "Fast portscan detection using sequential hypothesis testing," *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004*, Berkeley, CA, USA, 2004, pp. 211-225.

[13] X. Shu, D. Yao and E. Bertino, "Privacy-Preserving Detection of Sensitive Data Exposure," in *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 5, pp. 1092-1103, May 2015.

[14] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu., "Privacy-preserving data publishing: A survey of recent developments," *ACM Comput. Surv.* 42, 4, Article 14, 53 pages, June 2010.

[15] Bertino, Elisa, Dan Lin, and Wei Jiang. "A survey of quantification of privacy preserving data mining algorithms," *Privacy-preserving data mining: Models and Algorithms*, pp. 183-205, Springer US, 2008.

[16] Bertino, Elisa, "Data security and privacy: concepts, approaches, and research directions," Proceedings of IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), vol. 1, pp. 400-407. Atlanta, GA, 2016.

[17] L. Xu, C. Jiang, J. Wang, J. Yuan an//d Y. Ren, "Information Security in Big Data: Privacy and Data Mining," in *IEEE Access*, vol. 2, no. , pp. 1149-1176, 2014.

[18] Enrico Carniani, Davide D'Arenzo, Aliaksandr Lazouski, Fabio Martinelli, Paolo Mori, "Usage Control on Cloud systems," Future Generation Computer Systems,Volume 63, 2016, Pages 37-55.

[19] Caimi C., Gambardella C., Manea M., Petrocchi M., Stella D., "Legal and Technical Perspectives in Data Sharing Agreements Definition," In:. In Annual Privacy Forum, vol 9484., pp. 178-192, Springer, Cham, 2016.

[20] A. M. Garcia, R. S. Requena, A. Alberich-Bayarri, G. García-Martí, M. Egea and C. M. Martinez, "Coco-Cloud project: Confidential and compliant clouds," *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, Valencia, 2014, pp. 227-230.

[21] Barnum, S. "Standardizing cyber threat intelligence information with the Structured Threat Information eXpression (STIX™)." MITRE Corporation 11: 1-22, 2012.

[22] US-CERT: (https://www.us-cert.gov/).

[23] MITRE, "Common Vulnerabilities and Exposures (CVE)," URL https://cve.mitre.org

[24] MITRE, "Common Weakness Enumeration (CWE)," URL https://cwe.mitre.org

[25] Danyliw, R., Meijer, J., Demchenko, Y., The Incident Object Description Exchange Format, IETF RFC5070, December 2007.

[26] MANDIANT, "OpenIOC," http://www.openioc.org/. Accessed: July 22, 2014.

[27] Eric W. Burger, Michael D. Goodman, Panos Kampanakis, and Kevin A. Zhu, "Taxonomy Model for Cyber Threat Intelligence Information Exchange Technologies," In Proceedings of the 2014 ACM Workshop on Information Sharing & Collaborative Security (WISCS '14), pp.51-60, New York, USA, 2014.

[28] Anthony Rutkowski, Youki Kadobayashi, Inette Furey, Damir Rajnovic, Robert Martin, Takeshi Takahashi, Craig Schultz, Gavin Reid, Gregg Schudel, Mike Hird and Stephen Adegbite. "CYBEX: the cybersecurity information exchange framework (x.1500)." *ACM SIGCOMM Computer Communication Review* 40, 59-64, 2010.

[29] D. Tosh, S. Sengupta, C. Kamhoua, K. Kwiat and A. Martin, "An evolutionary game-theoretic framework for cyber-threat information sharing," *2015 IEEE International Conference on Communications (ICC)*, London, 2015, pp. 7341-7346.

[30] Man, Nguyen Doan and Huh, Eui-Nam, "A Collaborative Intrusion Detection System Framework for Cloud Computing," Proceedings of the International Conference on IT Convergence and Security, pp. 91-109, Dordrecht, Netherland, 2011.

[31] Xiaofan Chen and Shunzheng Yu, "A Collaborative Intrusion Detection System against DDoS for SDN," IEICE Transactions on Information and Systems, vol. E99-D, No.9, pp.2395-2399, Sept. 1, 2016.

[32] Yu-Sung Wu, B. Foo, Y. Mei and S. Bagchi, "Collaborative intrusion detection system (CIDS): a framework for accurate and efficient IDS," Proceedings of *19th Annual Computer Security Applications Conference*, pp. 234-244, 2003.

[33] Chenfeng Vincent Zhou, Christopher Leckie and Shanika Karunasekera, "A survey of coordinated attacks and collaborative intrusion detection," Computers & Security, Volume 29, Issue 1, 2010, Pages 124-140. February, 2010.

[34] C. Lo, C. Huang and J. Ku, "A Cooperative Intrusion Detection System Framework for Cloud Computing Networks," 2010 39th International Conference on Parallel Processing Workshops, San Diego, CA, pp. 280-284, 2010.

[35] Z. Chiba, N. Abghour, K. Moussaid, A. El omri, M. Rida, "A Cooperative and Hybrid Network Intrusion Detection Framework in Cloud Computing Based on Snort and Optimized Back Propagation Neural Network," Procedia Computer Science, Volume 83, pp.1200-1206, 2016.