A distributed cyber-security framework for heterogeneous environments

Raffaele Bolla[1], Paolo M. Comi[2], and Matteo Repetto[3]

[1] University of Genoa Genoa, Italy
raffaele.bolla@unige.it
[2] Italtel SpA, Settimo Milanese (MI), Italy
paolomaria.comi@italtel.com
[3] CNIT - Research Unit of Genoa, Genoa, Italy
matteo.repetto@cnit.it

Abstract

Evolving business models, computing paradigms, and management practices are rapidly re-shaping the usage models of ICT infrastructures, and demanding for more flexibility and dynamicity in enterprise security, beyond the traditional "security perimeter" approach. Since valuable ICT assets cannot be easily enclosed within a trusted physical sandbox any more, there is an increasing need for a new generation of pervasive and capillary cyber-security paradigms over distributed and geographically-scattered systems.

Following the generalized trend towards virtualization, automation, software-definition, and hardware/software disaggregation, in this paper we elaborate on a multi-tier architecture made of a common, programmable, and pervasive data-plane and a powerful set of multi-vendor detection and analysis algorithms. Our approach leverages the growing level of programmability of ICT infrastructures to create a common and unified framework that could be used to monitor and protect distributed heterogeneous environments, including legacy enterprise networks, IoT installations, and virtual resources deployed in the cloud.

1 Introduction

Current practice in enterprise cyber-security is largely characterized by the dominance of the "security perimeter" model, which assumes safe isolation of enterprise ICT assets by physical or virtual network segmentation, hence concentrating protection at the perimeter only. However, evolving business models and new value chains are increasingly requiring to combine ICT resources strewn across several geographic locations and demanding for more flexibility and dynamicity in enterprise security. ICT systems are undergoing a continuous transformation towards multi-domain architectures (including IoT installations) and virtualization (including externalization and usage of cloud services), which progressively blurs the boundaries between public zones and private domains. Specific factors that are contributing to opening new breaches in the security perimeter, making it an ineffective and obsolete concept, include [25]:

- *externalization and offloading*: the cloud and similar facilities host sensitive processes and data in third parties infrastructures, even shared with other customers;
- *multiplicity and heterogeneity of domains*: sensors, actuators, and other things in harsh environments with limited processing capabilities are more exposed to compromise than other IT assets in enterprise networks; for instance, recent botnets like Mirai, Brickerbot, and Hajime have demonstrated the vulnerability of IoT as well as the possibility to exploit compromised devices to carry out large DDoS attacks (1 Tb/s and above) [19];

- *personal and mobile devices*: bringing personal devices as smartphones, tablets, and removable media at work (*Bring Your Own Device*, BYOD) is already a major trend in organizations [17];
- *Inflexible defense*: DDoS and high rates of traffic often turn into either failing open (allowing traffic to pass without inspection to maintain availability), or failing closed (blocking all traffic to maintain security but causing business disruption).

Since valuable assets cannot anymore be kept inside a trusted physical sandbox, there is an increasing need for new forms of pervasive and capillary control techniques to tackle network threats, which are able to correlate events in both time and space dimensions, provide timely operational information to feed novel disruptive approaches capable of estimating the risk in real-time, and carry out focused and effective defensive and mitigation actions. Current cyber-security technologies suffer from important limitations, which make them less effective in the evolving scenario, especially against recent complex multi-vector attacks:

- intrinsic rigidity, due to the difficult to change architecture and system configuration: network partitioning, deployment of hardware or software security appliances (including the necessary agents), routing and switching policies;
- substantial inefficiency, because a) network traffic is often bounced across different appliances for analysis, inspection, mitigation, and processing, hence wasting bandwidth and increasing latency; b) similar or the same operations are carried out by different appliances (e.g., packet and software inspection, log analysis), which often are not interoperable;
- narrow scope, since most appliances usually deal with specific aspects only (e.g., firewalling, antivirus, event and log management, intrusion, deny of service) and consider a restricted set of events and devices;
- processing overhead to analyze packets, software, behavior, events, and logs, which eventually slows down systems and may be unsustainable by simplest devices (smartphones, IoT);
- outdated models: the presence of personal and mobile devices, the broad availability of removable media, the pervasive coverage of public wireless networks, interconnection to IoT devices, and externalization and offloading of processing/storage are the main factors that, especially when combined together, are increasingly opening new breaches in the security perimeter, making it an ineffective and obsolete concept.

In this paper, we envision a new paradigm for managing cyber-security threats in heterogeneous environments. Our approach aims at fostering the transition from multiple independent security appliances to a common framework, leveraging the increasing level of programmability of ICT infrastructures. We describe a multi-tier architecture (Fig. 1) that decouples a pervasive and shared context fabric from centralized business logic; the former is responsible to monitor the environment and to enforce security actions in a capillary way, whereas the latter collects detection and mitigation algorithms that are usually provided by different security appliances. A comprehensive presentation layer will facilitate the interaction with users and other security systems. Our architecture also includes specific elements for collection and secure conservation of forensic information for future investigation and possible use as evidence in court, since compliance to relevant normative will be ever more part of the design of cyber-security systems in the next future.

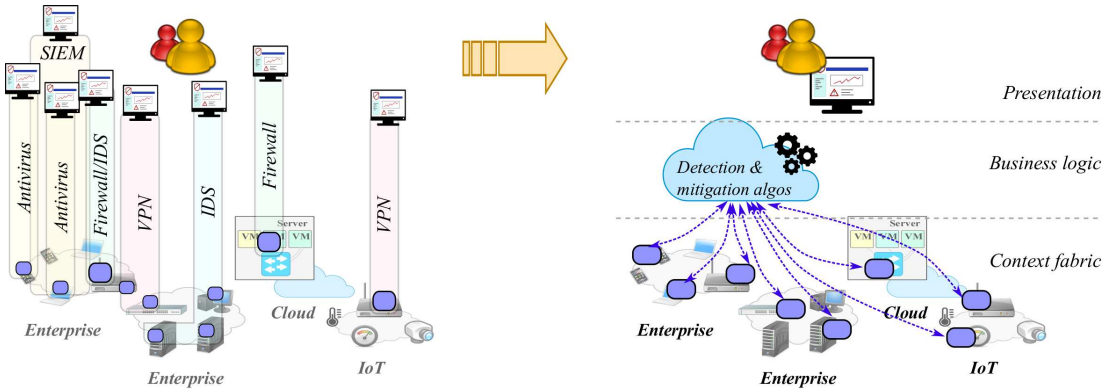

Figure 1: The complexity and multi-vector nature of recent cyber-security threats require a transition from current narrow-scope silos to a more integrated multi-vendor layered and open framework.

The paper is organized as follows. Section 2 briefly reviews current approaches and trends in distributed security monitoring. In Section 3 we outline the main concept behind our approach and describe the overall framework. In Section 4 we discuss the architecture design and the main components we are going to use for realizing our framework. Finally, we give our conclusions and describe future work in Section 5.

2 Related work

With the increasing uptake of cloud and IoT technologies, micro-firewalls and distributed firewalls are emerging to protect isolated or virtual resources outside of the enterprise perimeter [20].

Threat identification by exploiting network programmability has already been investigated by several research papers (e.g., [27, 21]. Several tools are already available that collects flow statistic from network devices through protocols as SMTP, NetFlow, sFlow, IPFIX, and, more recently, OpenFlow [23].

We observe a generalized trend from centralized standalone security boxes to more distributed frameworks, though programmability of networks is still partially exploited and unified control and management is missing.

3 A Distributed Framework for Identification of Network Threats

The underpinning concept of our approach is a transition from multiple independent security appliances to a common framework, where cyber-security is managed in a coordinated way, as shown in Fig. 1. Our purpose is to exploit and improve available interfaces and protocols for programmable communication infrastructures, hence our work will focus on network threats. In this context, the main challenge is to build the necessary knowledge and awareness both in physical and virtual environments, by real-time collection of massive events from a multiplicity of capillary sources, their inter- and intra-domain correlation in space and time, and the appli-

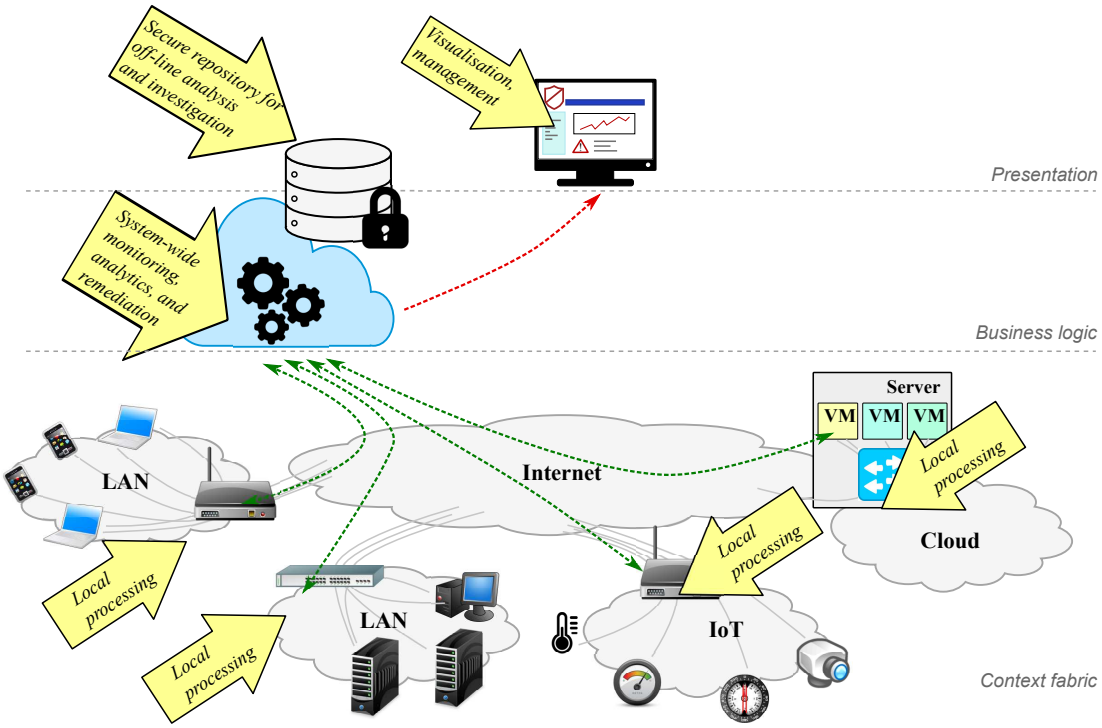

Figure 2: Overall concept and structure of the proposed framework.

cation of remediation actions, while maintaining essential properties such as forwarding speed, scalability, autonomy, usability, fault tolerance, resistance to compromises, and responsiveness.

Fig. 2 pictorially depicts the reference scenario and the structure for a novel multi-layer lawful-compliant cyber-security framework over large, distributed, and even virtualized environments. It drives beyond the legacy "security perimeter" concept by building on pervasive and capillarity programmability of the communication infrastructure. The framework is organized in three logical layers: context fabric, business logic, and presentation. The picture also shows the logical components, together with their main implications and tasks at each layer.

The *context fabric* entails a rich set of traffic filtering, packet inspection, processing (and, likely, storage) functions that are delegated to specific (physical or virtual) networking devices for performance and privacy matters. In our vision, such functions will no more rely on dedicated hardware appliances or virtual software functions; rather, the ambition is to shape the network behavior by building on the growing availability of flexible and programmable data planes and acceleration features. The *business logic* combines and correlates local information from multiple monitoring points, in the same or different domains, and builds system-wide situational awareness that allows to promptly detect and predict multi-vector and interdisciplinary cyber-attacks. Knowledge created at this level is presented to users for awareness and for identification of remediation and mitigation actions; it can then be shared with other administrative domains to coordinate response to new threats and attacks. In this layer, storage of data with legal validity is essential for off-line analysis and cyber-crime investigation. Finally, *presentation* of the processed information, events, and knowledge to humans provides knowledge of vulnerabilities, threats, anomalies, and attacks, so that countermeasures can be taken manually

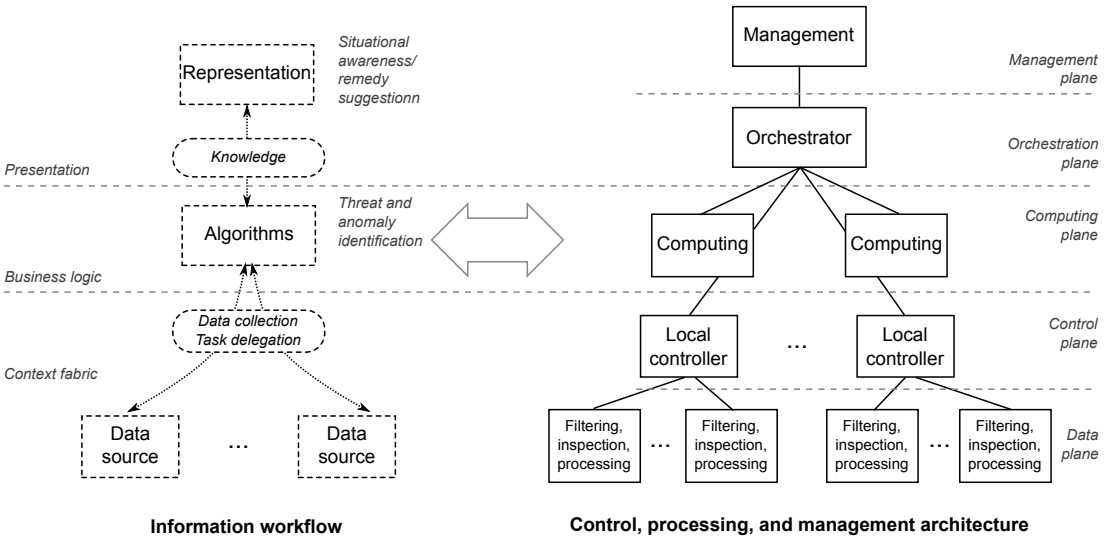

Figure 3: Information workflow and system architecture.

or automatically; in addition, log and proof of suspicious events are to be collected and should allow backtrace of activity in relevant domains.

Our design considers the duality and synergy between the information workflow and the control, processing and management architecture, as shown in Fig. 3. On the one hand, the information workflow better explains the logical processing flow. Local tasks (to be delegated to data sources) collect and aggregate data, and filter and process packets; identification algorithms (deep data analysis and correlation) build system-wide knowledge and situational awareness; data representation stores and visualize information for humans, and allows system administrators to identify possible remediation and mitigation actions. On the other hand, the control, processing, and management architecture shows the architectural elements and the orchestration functions. Due to its complexity, the architecture is further split into several logical planes: data, control, computing, orchestration, and management.

3.1 Context Fabric

The context fabric is responsible for collecting and aggregating information that is relevant for threat identification, as well as for classification, filtering, and processing of network packets. Data, events, and security logs must be collected capillary within the system, from heterogeneous sources in networking and computing devices.

The context fabric runs lightweight tasks in local networking devices. It leverages the growing level of programmability of networking infrastructures, which enable to push much more intelligence to network devices, well beyond the flow-level reporting already available today for anomaly detection (e.g., NetFlow, sFlow, IPFIX). In addition, emerging architectures and paradigms known as fog computing [24, 14] can be used to delegate processing tasks to a broader range of devices.

The architecture of the context fabric is organized in two planes (see Fig. 3). At the *data plane*, lightweight filtering and inspection tasks handle packets without putting significant stress to the computing resources needed. Our interest is not only limited to flow programmability

(e.g., OpenFlow [11], NetConf [13]), but it also extends to hardware and software acceleration frameworks for fast packet processing (e.g., Intel DPDK [4], FD.io [5], Snabb switch [10], IOVisor [6], BESS [2]) that create fast paths inside switching and routing devices, including virtual functions in hypervisors.

At the *control plane*, the need is the ability to discover, configure, and manage local heterogeneous resources. The purpose is to build a common and uniform abstraction of the underlying data planes, by extending existing protocols and interfaces (e.g., OpenFlow [11], Netconf/Yang [13], P4 [9, 15], RestConf [12]), while avoiding to overwhelm the network with excessive overhead. Access control is expected to provide fine-grained policies for data access, usage, and storage. This includes the capability of programming the underlying network layer, which may lead to serious security concerns if carried out in an uncontrolled and untrusted way.

One of the main limitation of existing technologies is that only simple data plane programs are allowed, i.e., without support for complex programs created according to the split data/control plane paradigm as originally proposed with SDN/OpenFlow. The challenges are therefore *i)* to support more powerful programs, which can operate according to the split data/control plane paradigm; *ii)* to support more powerful actions on the data in transit, which enable to implement some proactive security actions (e.g., drop network traffic, modify packet information, craft ad-hoc packets for specific purposes) that go beyond simple monitoring.

3.2 Business Logic

The main task of the business logic layer is to extract knowledge from the multiplicity and heterogeneity of data collected by the context fabric. The challenge is the definition of innovative algorithms that define which metrics are needed for each monitored point and correlate them in both time and space dimensions. Such analysis could be based on Attack Graphs, Attack Surface analysis, Kill Chain definitions and Attack trees models with the support of the deep learning techniques, Petri nets, and strategic models such as Stackelberg leadership model, Artificial Neural Networks.

The definition of algorithms for threat and anomaly detection should consider big data and machine learning capabilities to add predictive and proactive capabilities to existing security tools and systems. Multi-domain analysis and correlation of capillary data allows to promptly detect and predict multi-vector and interdisciplinary cyber-attacks, and to build wide awareness and coordinated response to new threats and attacks. The framework should also account for relevant events and data to be stored as evidence in case of suspicious activity, in order to be used as evidence in case of forensics investigation.

Processing and analysis of network traffic flows can reveal personal data, secrets, intellectual properties, habits, behaviors, etc., hence harming fundamental rights of organizations and humans. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data is going to revolutionize the privacy landscape in all European Countries. It is of paramount importance that all data gathered in any cyber-security monitoring framework will be compliant with said Privacy Regulation. In addition, there are specific requirements for legal validity of logs and events in court.

Data protection should consider anonymization and pseudonymization techniques that can hide the identify and the behavior of systems and users outside criminal investigations. Legal validity of the extracted data to be used in case of forensics investigation should be tackled, by considering technologies and mechanisms to protect the integrity, origin, and trustworthiness of data (e.g., digital signing, timestamping, message integrity codes, one-way encryption,)

according to the requirements and guidelines settled by normative frameworks.

3.3 Presentation

Risks, vulnerabilities, on-going attacks, and threats must be depicted to organizations and their security staff in the appropriate manner to support timely and effective reaction to cyber-security threats, by settling proper remediation and mitigation actions.

At the *management* plane (see Fig. 3), visualization solutions may rely on multi-layer software architectures and REST-based APIs for accessing threats and attacks database by multiple devices, flexible graphical layouts defined by templates and style-sheets to adapt the representation to heterogeneous devices and platforms, event-driven publish/subscription mechanisms for real-time notification of threats, anomalies, and attacks. The interface should also provide the ability to trigger pre-defined remediation actions, as well as to define new ones as new threats are identified. The capability to automate response allows faster mitigation for well-known attacks.

Managing the underlying infrastructure and adapting it to the evolving scenario is perhaps the most challenging issue for the whole framework. The *orchestration* plane is responsible for deploying and coordinating the hardware/software components over a mix of pervasive resources. The purpose is to understand which tasks should be offloaded and which tasks must be performed centrally. To this end, the target is to run detection algorithms and the business logic on high-performance, reliable, and protected infrastructures (e.g., private cloud installations), while offloading monitoring, inspection, and filtering tasks to local resources. In case of offloading, the orchestrator is responsible to select the proper resource (e.g., hardware switch or virtual switch in hypervisor) according to specific requirement and constraints by the algorithms (e.g., inspect incoming traffic, monitor traffic sent by host X, etc.).

4 Platform Design

Starting from the main concept and the conceptual architecture devised in Section 3, we have already derived the preliminary platform design shown in Fig. 4. It entails a number of functional elements that are necessary to implement the whole framework.

The *programmable switch* is responsible for traffic inspection and simple analysis, hence implementing the data plane of the context fabric. It will carry out filtering and processing operations by hardware or software acceleration mechanisms. Data, events, and measurements extracted will undergo a certification process to produce trusted information for forensics and lawful investigation. The switch will offer both a programming interface to configure filtering rules and offload simple tasks, as well as a data publication interface to export collected information to the controller (control plane of the context fabric). The switch will be an enhanced version of existing SDN devices; Open vSwitch [7] and Quake (user-space switch part of the OpenVolcano suite [16]) are two alternatives that will be considered in the implementation. OpenFlow [11] will likely be used as protocol interface between the controller and the switch, but we will also consider the applicability and appropriateness of NetConf [13]. Our target is a modular software implementation, which could be used in hypervisors and easily combined with software/hardware acceleration frameworks.

The *controller* translates (or "compiles") technology agnostic programs and configurations (Northbound interface) into specific SDN protocols (Southbound interface). A similar operation is performed in the opposite direction for data, events, and measurements. The controller is responsible for all switches in a domain or subdomain: it manages network topology, recovers

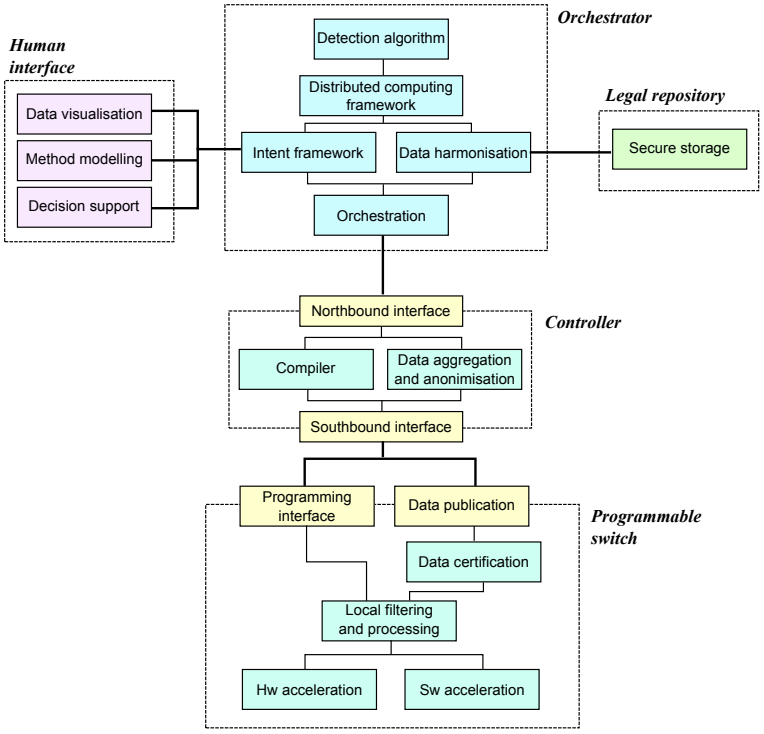

Figure 4: Reference platform design

from errors, collect data and measurements; in our architecture, it will also be responsible for data aggregation and anonymization. Anonymization is required to use data and measurements outside of legal investigation for threat identification without breaking confidentiality and privacy. The RestConf [12] protocols already provides a descriptive semantics for describing the network behavior, but it may be unsuitable for task offloading and complex filtering rules; we will look for existing or new alternatives. OpenDayLight [8], a modular SDN platform, and Magma, SDN controller part of the OpenVolcano suite [16], will be considered as base technologies to implement the controller.

The *orchestrator* is the smarter engine of the whole platform. It includes abstractions and models to split detection algorithms and mitigation policies in centralized and distributed computing tasks; centralized tasks perform data correlations and analysis, whereas distributed tasks are responsible for filtering, deep packet inspection, simple local processing. The distribution of tasks between the cloud and local devices is very similar to fog computing, hence we will consider recent concepts in this field, such as [18, 26]. A specific task for the orchestrator is automation and abstraction of the underlying infrastructure configuration starting from high-level policies; in this respect, an "intent framework" is envisaged to translate high-level description of monitoring and analysis information into specific instructions; this can be viewed as special application of the Network-as-a-Service paradigm to Monitoring/Inspection-as-a-Service [22]. Finally, data harmonization is necessary to provide common formats and syntax for storage of data coming from different domains and (possible) different controllers.

The *legal repository* will be responsible for secure and trusted storage of data, information, and events for successive lawful investigation. Key features in this case is trustworthiness, avail-

ability, integrity, resilience, resistance to attacks, and scalability, in order to prevent alteration or losses of data in case of attack. The repository will be an innovative storage infrastructure specifically designed and implemented to comply with requirements for lawful applications. We target design and implementation of a storage system suitable for preserving data with innovative reliability, security, availability, and scalability features inherited by existing virtual file systems for distributed storage of information (e.g., Ceph [3] or Hadoop [1]), through splitting of sensitive information among different storage facilities.

The *human interface* is the visualization tools to draw the current cyber-security picture and to enable quick and intuitive response to attacks. It will correlate attacks and threats with the actual network topology, suggest remediation and countermeasures, and enable definition of custom reaction strategies in case of new and unknown threats. Our human interface will be developed to explicitly address interaction with management of distributed resources.

5 Conclusions

In this paper, we have described our concept about a new cyber-security paradigm beyond the security perimeter model. Our work leverages programmable communication infrastructures and focus on network threats, but we think it could be easily extended to a broader scope (including monitoring of application events and system calls) once the first protocols for fog computing will be available.

We have already carried out preliminary design and identification of functional components for the whole framework. We have already gathered complementary skills and expertise in a research consortium and we are going to start the envisaged research and implementation work in the next months.

6 Acknowledgments

The authors would like to thanks other people who contributed to the discussion and the preparation of the SAMOA proposal: Joanna Kolodziej (Cracow University of Technology), Flora Strohmeier (Synyo Gmbh), Olaf Gebauer (Ostfalia University), Stefano Mele (Carnelutti Law Firm).

This work was supported in part by the European Commission, under project Matilda (grant no. 761898).

References

[1] Apache hadoop. [online]. Available at https://hadoop.apache.org, last visited in Sep 2017.

[2] Berkeley extensible software switch (BESS). [online]. Available at http://span.cs.berkeley.edu/bess.html, last visited in Aug 2017.

[3] Ceph – the future of storage. [online]. Available at http://ceph.com, last visited in Sep 2017.

[4] DPDK – data plane development kit. [online]. Available at http://dpdk.org, last visited in Aug 2017.

[5] Fd.io – the fast data project. [online]. Available at https://fd.io, last visited in Aug 2017.

[6] IOVisor project – advancing in-kernel io virtualization by enabling programmable data planes with extensibility, flexibility, and high-performance. [online]. Available at https://www.iovisor.org, last visited in Aug 2017.

[7] Open vswitch (OVS). [online]. Available at http://openvswitch.org, last visited on Aug 2017.

[8] OpenDayLight – open source SDN platform. [online]. Available at https://www.opendaylight.org, last visited on Sep 2017.

[9] P4 – programming protocol-independent packet processors. [online]. Available at http://p4.org, last visited in Aug 2017.

[10] Snabb – simple and fast packet networking. [online]. Available at https://github.com/snabbco/snabb, last visited in Aug 2017.

[11] OpenFlow switch specification. ONF TS-025, March 2015. Version 1.5.1 (Protocol version 0x06), Available at https://3vf60mmveq1g8vzn48q2o71a-wpengine.netdna-ssl.com/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf.

[12] A. Bierman, M. Bjorklund, and K. Watsen. RESTCONF protocol. RFC 8040, January 2017. Available at https://tools.ietf.org/html/rfc8040.

[13] M. Bjorklund, J. Schoenwaelder, and A. Bierman. Network configuration protocol (NETCONF). RFC 6241, June 2011. Available at https://tools.ietf.org/html/rfc6241.

[14] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu. Fog computing: A platform for Internet of Things and analytics. In N. Bessis and C. Dobre, editors, *Big Data and Internet of Things: A Roadmap for Smart Environments. Studies in Computational Intelligence*, volume 546, pages 169–186. Springer, Cham, 2014.

[15] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker. P4: Programming protocol-independent packet processors. *Computer Communication Review*, 44(3):88–95, July 2014.

[16] R. Bruschi, P. Lago, G. Lamanna, C. Lombardo, and S. Mangialardi. Openvolcano: An open-source software platform for fog computing. In *28th International Teletraffic Congress (ITC 28)*, pages 22–27, Würzburg, Germany, September, 12th–16th, 2016.

[17] European Cyber-Security Organisation. European cybersecurity Strategic Research and Innovation Agenda (SRIA) for a contractual Public-Private Partnership (cPPP). , June 2016. Available at http://www.ecs-org.eu/documents/ecs-cppp-sria.pdf.

[18] Huber Flores, Pan Hui, Sasu Tarkoma, Yong Li, Satish Srirama, and Rajkumar Buyya. Mobile code offloading: From concept to practice and beyond. *IEEE Communications Magazine*, 53(3):80–88, March 2015.

[19] David Holmes. DDoS attack trends. f5 whitepaper, November 2016.

[20] Wade Holmes. VMware NSX micro-segmentation. Vmware press.

[21] Hiroyuki Kasai, Wolfgang Kellerer, and Martin Kleinsteuber. Network volume anomaly detection and identification in large-scale networks based on online time-structured traffic tensor tracking. *IEEE Transactions on Network and Service Management*, 13(3):636–650, September 2016.

[22] ON.Lab. Introducing onos – a sdn network operating system for service providers. Whitepaper, November 2014. Available at http://onosproject.org/wp-content/uploads/2014/11/Whitepaper-ONOS-final.pdf.

[23] OpenDayLight. Defense4All security SDN application tutorial. Tutorial, Sep 2014.

[24] OpenFog Consortium Architecture Working Group. Openfog reference architecture for fog computing. [online], February 2017. Available at https://www.openfogconsortium.org/ra/.

[25] J. Pescatore. How DDoS detection and mitigation can fight advanced targeted attacks. SANS Whitepaper, September 2013. Available at https://www.sans.org/reading-room/whitepapers/analyst/ddos-detection-mitigation-fight-advanced-targeted-attacks-35000.

[26] Mahadev Satyanarayanan, Rolf Schuster, Maria Ebling, Gerhard Fettweis, Hannu Flinck, Kaustubh Joshi, and Krishan Sabnani. An open ecosystem for mobile-cloud convergence. *IEEE Communications Magazine*, 53(3):63–70, March 2015.

[27] S. Shirali-Shahreza and Y. Ganjali. Efficient implementation of security applications in openflow controller with flexam. In *21st Annual Symposium on High-Performance Interconnects (HOTI)*, pages 49–54, San Jose, CA – USA, August, 21st-23rd, 2013.