

Chaining Digital Services: Challenges to Investigate Cyber-Attacks at Run-Time

Matteo Repetto

***Abstract*—Today, Digital Service Providers (DSPs) increasingly use external services provided by third parties to create complex business chains, leveraging service-oriented architectures and everything-as-a-Service (XaaS) management paradigms. However, the heterogeneity and dynamicity of such environments represents a challenge for DSPs. In particular, Digital Service Chaining (DSC) brings the risk to amplify the scope of cyber-threats and to elude local security controls.**

In this position paper, building on our past experience in distributed monitoring and detection frameworks, we elaborate on the main challenges to investigate cyber-attacks in complex, heterogeneous, multi-ownership, and interconnected systems. Our main contributions are the operational workflow and reference architecture for run-time protection of digital service chains, as well as the identification of the main issues and research directions that the networking and cybersecurity communities should jointly address.

***Index Terms*—Cyber-threats propagation, Digital Service Chaining, Monitoring, detection, and response, Multi-stage and multi-vector attacks**

I. Introduction

Today, data is increasingly becoming the main valuable asset in the digital economy, shifting business models from infrastructure ownership to data transformation and trading. Accordingly, many Digital Service Providers (DSPs) are now looking for open, transparent and secure digital ecosystems where data and services can be made available, collated, shared, and *chained*. The scope extends from Internet of Things (IoT) and fog devices, to edge, cloud, and network infrastructures, up to applications and functions.

The main challenge behind this new business paradigm is represented by multi-ownership, which hinders the implementation of homogeneous and coherent end-to-end management and operational policies, just because digital services are owned and operated by different providers. In this respect, the concept of *federation* has been recently proposed for both network [1] and cloud infrastructures [2] to address the heterogeneity of network policies and configurations in multiple domains. The scope is usually focused on management aspects to abstract, discover, chain, and orchestrate services, including Cyber-Security Functions (CSFs) [1].

Unfortunately, the common assumption that standalone cyber-security appliances are enough to defend against cyber-attacks has largely demonstrated to be wrong in the real world, because modern multi-stage and multi-vector attacks are explicitly conceived to elude traditional signature- and rule-based detectors, and even to deceive most recent Artificial Intelligence (AI)-based techniques [3]. The analysis of security-related events, alerts, and anomalies from CSFs remains the fundamental process to investigate attacks at run-time, but it now faces unprecedented challenges for DSC, due to heterogeneous, agile and multi-ownership environments.

In this paper, we elaborate on the existing scientific and technical gap to investigate cyber-attacks in interconnected systems. The main contributions of our work are the operational workflow and reference architecture for providing an holistic view of digital service chains and making the investigation process more adaptive to the evolving context (i.e., the composition and topology of the system, attack strategies and techniques). The discussion builds on and extends our previous work in this field [4]. The novelty is represented by the focus on *investigation* of complex attacks, which extends plain detection of and response to individual stages, and the description of the corresponding enhancements that are necessary in this respect.

The rest of the paper is organized as follows. We first briefly review the concepts of DSC and multi-stage attacks with a concrete example, and discuss the main challenges to be addressed. Then we elaborate on the concept of Security Orchestration and Automated Response (SOAR) and how it can be implemented in complex, heterogeneous, and multi-ownership environments. Finally, we discuss open issues and the most urgent research topics to be addressed for investigation of cyber-attacks in DSC.

II. Challenges to protect digital service chains

A. Digital Service Chaining concept and use cases

DSC can be viewed as the generalization of Service Function Chaining (SFC) to a broader set of functions (devices, software, cloud/edge infrastructures), exposed by multiple DSPs with diverse types of relationships (network and software interfaces, hosting services). It basically assumes the presence of a large digital continuum made of software applications and functions, cloud and network infrastructures, edge and fog installations, and user/IoT devices. The interactions between services is often based on common protocols for web services (HTTP, SOAP), message queuing (AMPQ, Apache Kafka), and query languages (SQL, GraphQL, LDAP, Cypher, and many other for specific database technologies). Moreover, open API and data models have been defined by the FIWARE and Smart Data Models initiatives, respectively.

This work was supported in part by the European Commission under Grant Number 833456 (GUARD).

Corresponding author: Matteo Repetto.

Matteo Repetto is with IMATI-CNR, Genoa, Italy (email: matteo.repetto@ge.imati.cnr.it).

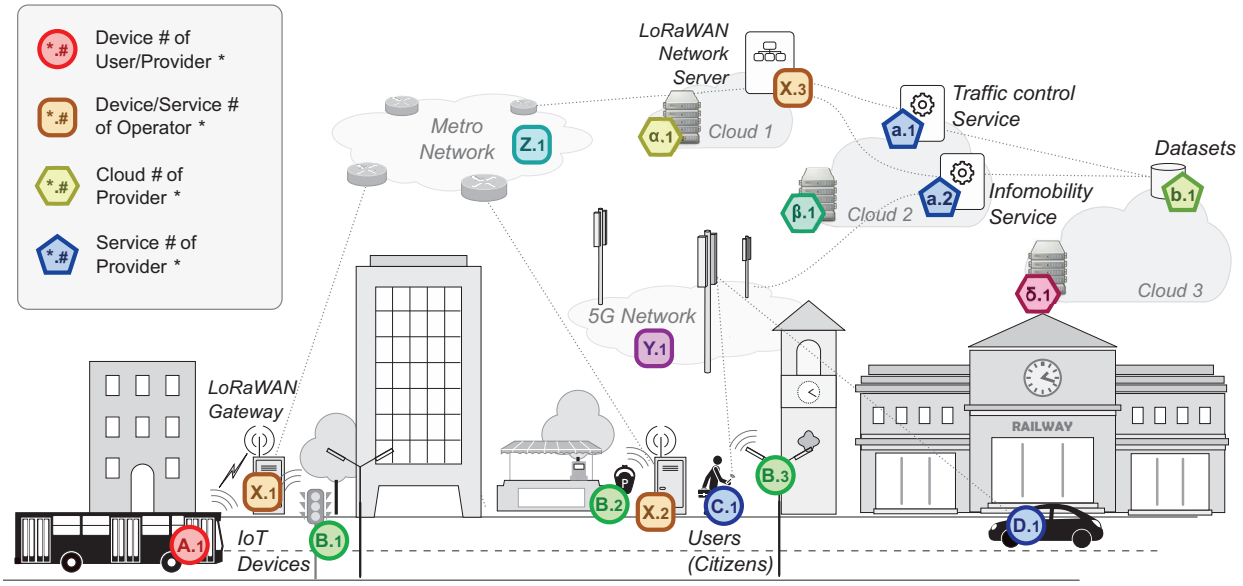

Fig. 1. A realistic example of devices, infrastructures, and services chained to create Smart City applications.

The DSC paradigm is already used in the implementation of Smart City services. Fig. 1 shows a typical example where multiple devices around the city feed a set of smart applications in the cloud (e.g., traffic control, infomobility). The picture highlights how complementary digital services (devices, infrastructures, applications) owned by different providers are linked together to create one or more value-added chains. As an illustrative example, each chain includes a LoRaWAN and a metropolitan fiber network, datasets (timetables, news, events, etc.), cloud infrastructures, 5G access, and so on. Different shapes and provider/name encoding are used to represent the heterogeneity of services and multi-ownership, respectively, as explained in the legend. In a real scenario, the number and type of devices, infrastructures, and applications is expected to be far larger, which complicates the identification of interactions and cross-dependencies.

A schematic representation of the previous service chain is given in Fig. 2. Modeling DSC is a complex task, especially for security purposes, because of the different kinds of interactions between applications and infrastructures. Applications usually interact each other by exchanging network flows or by invoking remote functions – Application Programming Interfaces (APIs); the former is the typical case for network functions, and the latter for cloud services. Infrastructures host applications and services by providing computing, networking, and storage resources according to different virtualization models.

The singular points for DSC are the large heterogeneity of services and environments, agile composition, and multi-ownership, which often lead to complex, unpredictable, and partially inscrutable topologies. Indeed, software interfaces allow more flexibility in adding, removing, or replacing services to the chain than legacy monolithic applications, hence the chain is expected to change at run-time according to the evolving context (business needs, workload, service availability). For instance, DSPs may move applications to different cloud

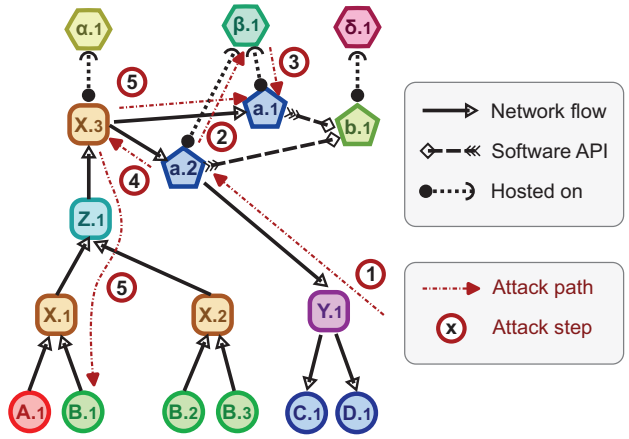

Fig. 2. Model of a digital service chain and multi-step attack path.

infrastructures, add/remove IoT devices, change or update their own application software.

B. Multi-step attacks

Direct attacks to valuable targets are seldom possible today, because of the massive deployment of cyber-security appliances, especially at the network perimeter. Unfortunately, distributed systems like DSC do not have a sharp perimeter [5]. Additionally, attackers have been increasingly adopting sophisticated intrusion techniques, consisting of multiple phases (or *steps*) that exploit apparently uncorrelated vulnerabilities, even on different systems [6]. Consequently, modern attacks are typically modeled as "cyber kill chains," which include several phases, from reconnaissance to delivery of malware and finally exploitation [7]. The strategy usually leverages the weakest or more vulnerable links, and then propagates to more valuable targets. A typical example is represented by IoT devices, which often represent the first step to tear down entire services or infrastructures [8].

Several security incidents have been reported in the last years against digital supply chains (e.g., SolarWinds, Atlassian, British Airways) [9], but the situation is expected to get worse with the massive adoption of DSC. Even if no incidents have been publicly reported so far (also due to the prevailing experimental nature of existing installations), we can describe a realistic example of multi-step attack against a Smart City environment, where an attacker progressively move laterally along the chain until he reaches his primary target.

With reference to Fig. 2, let us suppose a criminal wants to disrupt traffic control operations for the entire city, hence targeting service `a.1`. Of course, such service is protected by access control and firewalling, but the presence of shared infrastructure and connected services might open indirect paths to adversaries. For instance, the attacker could exploit the public facing web application of the infomobility service to open a reverse shell within the system (step 1, CVE-2020-17530). Even if he is inside a container/Virtual Machine, the attacker can now look for instance metadata, which gives him direct access to the cloud management interface (step 2). If such credentials are not the same used for the target service, privilege escalation might still be possible due to policy misconfiguration (step 3) [10]. Alternatively, the attacker might try a cross-site scripting attack towards the LoRaWAN server (step 4, CVE-2020-7656) with the purpose of generating fake messages addressed to the traffic control service or the field devices, e.g., traffic lights (step 5).

C. Main issues for run-time detection

Common practice for run-time detection of threats and attacks is largely based on Security Information and Event Management (SIEM) platforms that collect logs and events from multiple sources, including security appliances and digital services. They help experts to filter out false positives, to correlate anomalies in different subsystems, and to investigate incidents. Unfortunately, the usage of SIEM for DSC is not straightforward for a number of reasons.

1) Fragmented operations: Multi-ownership of federated environments leads to a substantial fragmentation of cybersecurity operations, with two main consequences. First, visibility is limited to individual subsystems. A successful attack to a public cloud infrastructure is likely to threaten its tenants in either a direct or indirect way, but the latter have no visibility on it. Second, multiple stages of the same attack carried out in different subsystems cannot be correlated, hence hindering the identification of the whole cyber kill chain. Third, each standalone provider can detect and resist to cyberattacks originated from other domains, but without fighting back; therefore attacks persist and might find other vulnerable points to move laterally along the chain.

2) Dynamic service topologies: The detection logic is often based on static configurations and rigid assumptions about the chain composition and topology. However, DSC is highly dynamic in nature: management operations like scaling, replication, migration, and replacement alter the system composition and topology at run-time, hence jeopardizing the coverage and quality of data used by the detection algorithms. For instance, replacing or updating the software may change the format of its logs and events. Changing the number of running replicas may mislead the detector with different amount or frequency of data. Migration to different infrastructures may result in different traffic patterns, hence invalidating previously trained models.

3) Multi-ownership and externalization: Beyond fragmented operation, multi-ownership raises privacy and confidentiality concerns related to the visibility of internal data and resources. In a DSC scenario there might be more than one Security Operator that operate on behalf of different DSPs, hence demanding for dynamic and agile trust mechanisms.

4) Lightweight environments: Containers and serverless computing greatly enhance the efficiency of virtualized environments but also hinder the deployment of legacy security appliances. Infrastructure-level tools provided by Cloud Service Providers are often extremely heterogeneous in terms of capabilities and interfaces, which makes the implementation of uniform monitoring and response policies really challenging without the adoption of an additional layer implemented by Cloud Access Security Brokers.

5) Evolving attack patterns: Attacks continuously evolve to elude static rules and definitions of Indicators of Compromise (IoCs). This issue is even more tricky for DSC, because many configuration parameters often change at run-time (e.g., network addresses).

III. Improving SOAR architectures

The concept of SOAR is now emerging for investigating new techniques and their variants, predicting next phases and the final target, and elaborating the most effective response contextualized to the current environment. To this purpose, it integrates the complementary workflows of different teams:

- the Security Operation Center (SOC) processes large amount of events and alerts, filters out false positives, looks for known IoCs, and starts pre-defined response and mitigation actions;
- Incident Response (IR) investigates incidents once occurred, using cyber evidence to reconstruct the attack strategy and derive new IoCs for SOCs;
- Threat Intelligence Management (TIM) documents incidents, including the attack technique, vulnerabilities exploited, and response strategy, to create the necessary knowledge to block the same attack in the future.

The main objective for a SOAR architecture is to automate as much as possible the overall process, from investigation of incidents (IR), to the derivation of new signatures and response strategies (SOC), up to description and sharing of threat intelligence (TIM).

A. Enhancing workflows with more context

Typical analysis, investigation, and response workflows assume rather static environments, where changes are infrequent and can be reported by manual processes. However, the concept of DSC implicitly leads to more agility and heterogeneity, hence an additional abstraction layer is necessary to mediate between such workflows and the underlying monitoring and

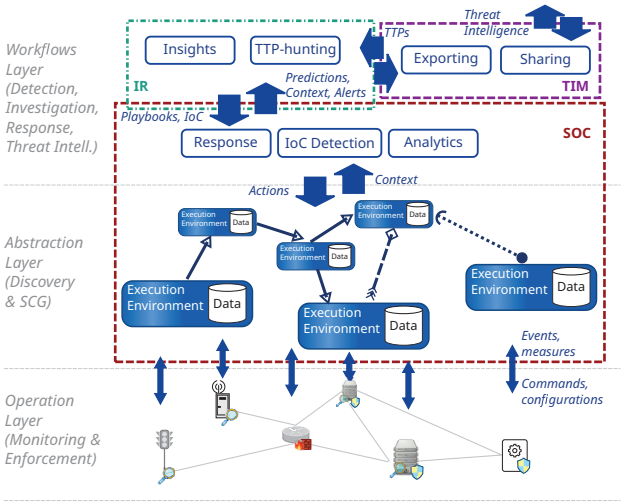

Fig. 3. Integration of SOC/IR/TIM workflows.

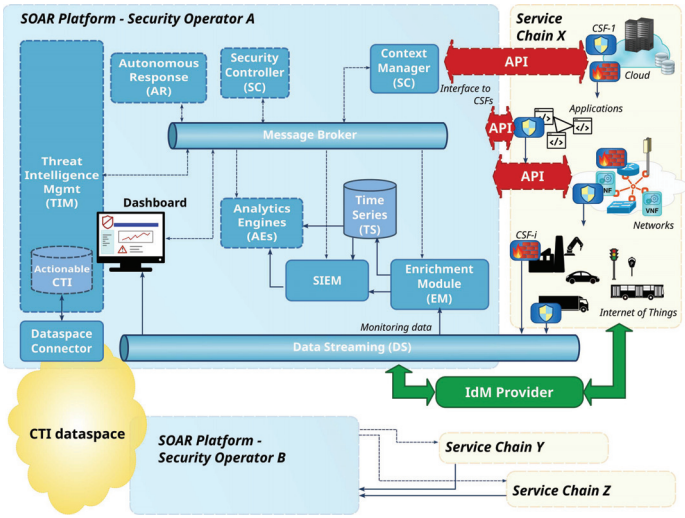

Fig. 4. Enhanced SOAR architecture.

enforcement tasks. This layer becomes responsible for automatically discovering the whole service topology and CSFs available therein. This is abstracted as Service Context Graphs (SCGs) that describe:

1) identity and ownership of digital services,
2) the execution environment of each service, namely relevant software/hardware and configurations,
3) operational relationships and communication patterns between services,
4) identity, monitoring and enforcement capabilities for each CSF, and
5) security-related data from CSFs.

Fig. 3 depicts our vision for integrating and enhancing SOC/IR/TIM workflows with more context.

Typical SOC tasks are extended with remote discovery and control of CSFs, as well as abstraction of the composition and topology of digital chains in terms of SCGs. Security-related events and data are used for *IoCs Detection* and advanced *Analytics*; the latter looks for anomalies and makes predictions, also leveraging Machine Learning (ML) or other forms of AI. *Response* is largely automated by executing playbooks tailored to the current environment.

IR makes extensive usage of alerts, predictions, and context to recognize patterns that can be traced back to known techniques, or to identify new techniques (i.e., zero-day attacks). In this context, looking for known Tactics, Techniques, and Procedures (TTP) from threat intelligence (a process known as *threat-hunting*) provides a more robust and powerful approach than plain detection of IoCs. It works with many attack variants (source IP address, Command and Control address, malicious URLs) and allows to infer the potential targets. The same approach can be used to extract new *Insights* about attack techniques, that enrich threat intelligence, and to translate threat intelligence into low-code playbooks tailored to the set and position of CSFs available. For instance, Denial of Service (DoS) traffic could be dropped at the edge, cascading firewalls could be applied for load balancing and selective protection of individual vulnerable resources. Playbooks may also include management actions, if such controls are available from the cloud infrastructure. The scope should be limited to evidence collection and attack mitigation (e.g., to create snapshots of cloud services, to deploy clean software images or virtual security functions, to divert network packets), to avoid interfering with external orchestrators.

Finally, *Exporting* will encode novel findings in the form of *actionable* Cyber-Threat Intelligence (CTI) [11], suitable to be consumed by machines and automatically shared with other stakeholders, including other providers involved in the same or similar business chains. *Sharing* threat intelligence is of paramount important for DSC, since the usage of the same orchestration templates results into similar threats for different DSPs.

B. Functional elements

Fig. 4 shows the main functional elements that are necessary to implement the workflows previously described, framed in a reference architecture derived from our past experience on this topic [4]. This design allows a common Security Operator to run detection, investigation, and response processes on heterogeneous service chains operated by multiple providers. Dashed boxes denote open research challenges that will be discussed in the next Section.

Multi-ownership and agile composition require one or more independent Identity Management (IdM) providers that ensure mutual identification between the Security Operator and DSPs. The identity also includes a set of attributes (e.g., nationality, role, certifications) that can be used to create trust and access policies, e.g. with Attribute-Based Access Control (ABAC). For instance, a DSP may only send data to Security Operators from a given country, political region, or reputation. Similarly, Security Operators may use the same properties to assess the quality and reliability of data retrieved from each DSP.

At the bottom, Data Streaming provides continuous delivery of events, measures, and metrics through message brokers (e.g., AMPQ or Kafka), which are then enriched, aggregated,

transformed, and indexed on the fly by a combination of transformation filters in the Enrichment Module before being stored in the Time Series database and/or processed by a legacy SIEM. A battery of Analytics Engines extends the SIEM with prediction, forensics, and threat hunting capabilities.

Adaptivity is implemented by the functional elements at the top. The Context Manager is responsible for maintaining the current context in the form of SCGs. It needs a suitable API for i) discovering services, relationships, CSFs and their capabilities, and ii) configuring CSFs.

Autonomous Response refines high-level strategies, supplied by either actionable CTI or human operators, into concrete configuration rules, tailored to the current context (e.g., CSF capabilities and service topology). Threat Intelligence Management creates, shares, and consumes actionable CTI. The latter can be shared in a common, open, and trustworthy dataspace with fine-grained access rules and anonymization, to reduce the impact of zero-day attacks.

Finally, the Security Controller is the smart engine that automates monitoring, detection, investigation, and response processes. It starts from high-level definitions supplied by humans and translates them into concrete configurations for all the underlying components, including CSFs, the Enrichment Module, the SIEM, and the Analytics Engines. Its scope also includes i) loading of playbooks from Autonomous Response and their execution when triggered by specific events; ii) synchronization of the different processes for prediction, forensics, threat management.

C. Existing tools

The technical feasibility of the proposed vision is proven by a preliminary framework [4], which mostly cover the functional blocks depicted with a solid line in Fig. 4. Such framework demonstrates the fundamental aspects for discovering services and automatically configuring elementary monitoring and detection pipelines, but lacks fully adaptive processes for dynamic and heterogeneous environments.

Existing tools in this framework include the Context Manager, which creates the internal abstraction of SCGs. A visual representation is also provided, which looks as shown in Fig. 5. The control interface to CSFs adopts a custom data model derived from on-going initiatives in the service management domains (i.e., Smart Data Models), which could be updated based on emerging standards like OASIS OpenC2 [12] and IETF Interface to Network Security Functions (I2NSF).

The Security Controller automatically configures monitoring and detection pipelines, but this is currently limited to the selection of CSFs that feed specific detectors. However, the current abstraction cannot fully decouple data from its sources, and therefore it is not possible to adapt the pipeline to changes in the underlying service composition and topology. Similarly, the set of detectors encompasses ML models for anomaly detection, innovative signature detection rules based on extended Berkeley Packet Filter (eBPF) programs, an experimental alert aggregation module, and dynamic parsers that automatically analyze new log formats. This represents a meaningful step in the direction of adaptive processes, but more research is needed towards automated investigation and threat hunting.

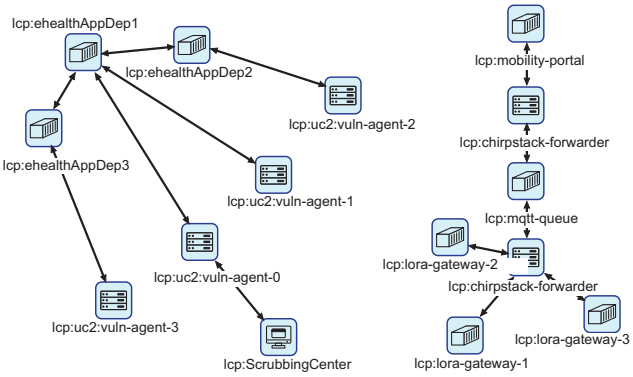

Fig. 5. Examples of service chains automatically discovered.

TABLE I
CYBER-SECURITY ISSUES AND SOLUTIONS FOR DSC.

Issue	Research directions
Fragmented operations	SCG. Threat-hunting. Actionable CTI.
Dynamic topologies	SCG. Control interfaces. Adaptive processes. Automated response.
Multi-ownership and Externalization	Control interfaces. IdM and access control. Privacy and confidentiality.
Lightweight environments	Programmatic monitoring and enforcement.
Evolving attack patterns	Threat-hunting. Actionable CTI.

The mutual authentication of DSPs and Security Operators is based on JSON Web Tokens (JWTs), which include the necessary attributes for ABAC in both the control API and Data Streaming. The generation of encryption keys follows common practice based on a Public Key Infrastructure.

Notwithstanding the limited coverage of the broader vision discussed in the paper, the existing tools provide useful hints on performance aspects [4]. Performance analysis was carried out on the most critical components from the computing perspective, namely the Security Controller and the whole data handling pipeline. It demonstrated that this kind of architecture is feasible and has good performance. However, data transformation represents the main bottleneck with current technology, especially in resource-constrained environments (IoT devices, cloud containers). This suggests to prefer data transformation at a central location instead of local processing; the impact on bandwidth is limited, since current practice already collects the widest set of data.

IV. Challenges and Open Issues

Table I gets back to open issues for DSC previously analyzed and identifies the most promising research directions briefly discussed below.

A. Service Context Graphs

The development of digital services with microservices and Service-Oriented Architectures (SOAs) makes applications, infrastructures, and devices more interconnected than ever

and in ways that are challenging to map or visualize. The main need here is the definition of both information and data models for SCGs, which facilitate the investigation of how attacks can propagate and vulnerabilities could be exploited across different systems. The long-term objective should be the creation of true *digital twins* [13], which allow emulation and investigation of cyber-security processes in a safe sandbox fed by real inputs.

While metadata about applications and the underlying infrastructure are quite trivial to collect, the technical and business relationships are more difficult to retrieve. New techniques are therefore necessary to infer this interactions by monitoring network flows or extracting metadata from service providers.

B. Adaptive processes

The dynamicity of digital service chains requires to make monitoring, detection, and response processes adaptive to the evolving context. The objective is to automatically update such processes in case of changes to service instances, their relationships, and CSFs, according to high-level intents and policies.

For instance, collection, enrichment, transformation, and indexing tasks should be automatically re-configured as data sources change over time. In this case, the challenge is represented by the typical heterogeneity in capabilities and scope of CSFs. Additionally, the automatic generation of parsers at runtime is necessary to address changes in data structure and to make unstructured textual data (e.g., system and application logs) accessible to detection algorithms in a seamless way. The solution is not straightforward, because data coverage and quality must be retained over changes to not deceive the detection algorithms, especially when ML/AI models are used.

C. Control interfaces

Standard interfaces are necessary to describe, discover, and control in a common way CSFs from different vendors, in order to overcome the current heterogeneity in protocols and capabilities. In this respect, common models must be defined to expose homogeneous monitoring/enforcement capabilities and configuration properties, so as to support seamless interaction with alternative implementations of the same function. OpenC2 already proposed a language with flexible syntax and semantics to interplay with remote CSFs, which can be declined to different classes of functions by defining *profiles* [12]. However, the only official profile available is for the trivial case of stateless packet filtering, while a clear answer must still be given about the possibility of modeling a broader set of relevant CSFs in a uniform way. Additionally, discovery of functions and their profiles at run-time has not been covered yet.

D. Programmatic monitoring and enforcement

Instrumenting containers with traditional CSFs may be challenging, especially in case of certified images of Virtual Network Functions that cannot be modified by the users (a typical use-case for the telecommunication industry). Relying on monitoring and enforcement operations implemented at the hypervisor/infrastructure layer is usually possible, but this solution is rather inflexible and hinders portability to different environments. Novel software techniques for code augmentation at run-time promise to become a flexible, tailored, and extensible solution to cover a broad range of monitoring and enforcement scenarios, including Deep Packet Inspection (DPI), flow measurements, software tracing. In this respect, the eBPF framework has already demonstrated to be a suitable technology [14], although many issues must still be investigated about management aspects, slow learning curve, and privacy.

E. Prediction, forensics, and threat hunting

IR faces increasing challenges in reconstructing and analyzing digital evidence scattered among different physical or virtual locations, due to the growing usage of anti-forensics and obfuscation techniques. However, post-mortem investigation is the last resort once an incident has occurred. The best option would be to anticipate incidents, by proactive, adaptable, and CTI-driven forensics that can build a case even before the need arises, hence going beyond static, short-lived, and contextless IoCs [15]. This is possible by correlating digital evidence with TTP threat intelligence.

The correlation of events (alerts, logs, and network measures) from both the edge and the cloud allows to understand the overall strategy, and consequently, to identify the source and target of the attack. Alert aggregation should also address the long-standing problem of false positives, by combining individual alarms into logical groups, which allows to understand new and complex multi-vector or multi-stage attack patterns.

Information sharing should be used for building more accurate and robust AI/ML prediction models, by leveraging Federated Learning and Transfer Learning techniques that exploit the similarities in digital services to create reliable and robust models in acceptable time.

F. Automated response

Existing SOAR implementations support a large number of actions, tools, and static response playbooks, but are quite rigid to changes in the underlying infrastructures and CSFs. More automation is necessary that can automatically translate actionable CTI into concrete configurations (i.e., low-code playbooks) at run-time, tailored to the evolving context (service composition and topology, CSF capabilities), also taking into account functional and performance constraints. This is a well-known challenge also in network management, which still lacks proper mechanisms.

G. Actionable CTI

Threat intelligence is today largely created and consumed directly by humans. The research challenge is a threat intelligence management process that is contextualized, automated, collaborative, and, most of all, actionable. The necessary

pathway goes through complementary aspects: i) the extension of current standards and tools (e.g., STIX, MISP) to include, for instance, the technical description of the execution environment (infrastructure, software, configurations), as well as sharing of AI models in additions to basic Intrusion Detection System/Firewall/Antivirus configurations; ii) automatic clustering and linking of anomalies to common attack tactics, in order to create new operational threat intelligence; iii) dataspace models and policies to facilitate discovery and exchange of CTI in a controlled, privacy-preserving, and trustworthy way.

H. Privacy and confidentiality

The dichotomy between service owners and security operators may appear a big concern for privacy and confidentiality issues. However, delegation of detection and response operations to external SOCs is already a growing trend, especially for small and medium businesses that cannot afford dedicated staff with the high-level of skills required to manage complex attacks. Differently from existing practice, DSC should result in a more dynamic environment, where trust relationships between DSPs and Security Operators are automatically established at run-time.

From a technical perspective, the interface to CSFs plays a crucial role in limiting access to confidential and sensitive data. Indeed, DSPs retain full authority (and responsibility) on what is shared and how, including filtering out, obfuscation, and anonymization of private and confidential data, hence preserving sovereignty, confidentiality, and privacy according to the need-to-know principle. Non-disclosure agreements between providers and informed consent from end users represent an additional layer of data protection beyond IdM and access control. As a matter of fact, security operators must avoid to incidentally reveal private and confidential data that they might infer from their prediction, forensics, and threat hunting processes (e.g., vulnerability to specific threats).

V. Conclusion

Investigation of cyber-attacks to digital service chains requires best of breed detection, investigation, and response techniques, tailored to the heterogeneous, agile, and multi-ownership nature of these environments. In this respect, service management and cybersecurity operation are more intertwined than what typically considered today, and scientific and technological advances are expected in both domains. On the one hand, service management should improve visibility by providing uniform security controls for monitoring and enforcement. On the other hand, security operations should become more adaptive to the evolving environment and context, both in the detection and response phase.

Despite the feasibility of the proposed vision is already demonstrated by recent advances in this field, many challenges and open issues still remain to be addressed. Beyond the scientific and technical aspects discussed in this paper, business and administrative factors should be considered as well, especially who should be responsible for the protection of the whole chain, what is its responsibility, and how it relates to national and supranational incident response organizations.

References

[1] L. Cui, F. P. Tso, and W. Jia, "Federated service chaining: Architecture and challenges," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 47–53, March 2020.

[2] R. Buyya et al., "A manifesto for future generation cloud computing: Research directions for the next decade," *ACM Computing Surveys*, vol. 1, no. 5, pp. 105:1–105:38, September 2019.

[3] J. Navarro, A. Deruyver, and P. Parrend, "A systematic survey on multi-step attack detection," *Computers & Security*, vol. 76, pp. 214–249, 2018.

[4] A. Carrega et al., *Cybersecurity of Digital Service Chains: Challenges, Methodologies and Tools*, ser. LNCS. Springer Nature, April 2022, vol. 13300, ch. A Reference Architecture for Management of Security Operations in Digital Service Chains, pp. 1–31.

[5] R. Rapuzzi and M. Repetto, "Building situational awareness for network threats in fog/edge computing: Emerging paradigms beyond the security perimeter model," *Future Generation Computer Systems*, vol. 85, pp. 235–249, August 2018.

[6] T. Shawly, A. Elghariani, J. Kobes, and A. Ghafoor, "Architectures for detecting interleaved multi-stage network attacks using hidden markov models," *IEEE Trans. Depend. Sec. Comput.*, vol. 18, no. 5, pp. 2316–2330, Sep.-Oct. 2021.

[7] E. Hutchins, M. Cloppert, and R. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," in *6th Intl. Conf. on Inf. Warfare and Secur.*, Washington, DC, USA, Mar. 17th–18th, 2011, pp. 113–125.

[8] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[9] "Supply chain attacks: 7 examples and 4 defensive strategies," BlueVoyant Knowledge Center, accessed on April 26th, 2023. [Online]. Available: https://www.bluevoyant.com/knowledge-center/supply-chain-attacks-7-examples-and-4-defensive-strategies

[10] S. Chierici, "Cloud lateral movement: Breaking in through a vulnerable container," Sysdig blog, Jul. 25th, 2022, accessed on April 26th, 2023. [Online]. Available: https://sysdig.com/blog/lateral-movement-cloud-containers/

[11] F. Skopik, A. Bonitz, V. Grantz, and G. Göhler, "From scattered data to actionable knowledge: flexible cyber security reporting in the military domain," *Int. J. Inf. Secur.*, vol. 21, no. 6, pp. 1323–1347, December 2022.

[12] V. Mavroeidis and J. Brule, "A nonproprietary language for the command and control of cyber defenses – OpenC2," *Computers & Security*, vol. 97, no. 101999, October 2020.

[13] S. Mihai et al, "Digital twins: A survey on enabling technologies, challenges, trends and future prospects," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 4, pp. 2255–2291, 2022.

[14] H. Sharaf, I. Ahmad, and T. Dimitriou, "Extended berkeley packet filter: An application perspective," *IEEE Access*, vol. 10, pp. 126 370–126 393, 2022.

[15] C. O'Brien, A. Jarosz, and J. Abraham, "Beyond the IoC," EclecticIQ whitepaper, 2021, last accessed: Dec 20th, 2022. [Online]. Available: www.eclecticiq.com

Biography

Matteo Repetto is research scientist at the IMATI institute, National Research Council of Italy (CNR). His main research interests include software-defined networking, network function virtualization, cloud computing, and cybersecurity. He recently served as scientific and technical coordinator of the EU-funded projects ASTRID and GUARD, which investigated new cybersecurity paradigms that fit the progressive softwarization of telecommunication networks and computing infrastructures.