A Coalition Perspective on Federated and Adaptive Clouds for Disadvantaged Tactical Networks

Mattia Fogli*, Thomas Kudla†, Geert Pingen‡, Susan Watson§, Harrie Bastiaansen‡, Pablo Sanchez¶, Niranjan Suri‖** *

University of Ferrara, Ferrara, Italy
mattia.fogli@unife.it

† *The Netherlands Organisation for Applied Scientific Research (TNO)*, The Hague, The Netherlands
{harrie.bastiaansen, geert.pingen}@tno.nl

‡ *Fraunhofer Institute for Communication, Information Processing and Ergonomics*, Wachtberg, Germany
thomas.kudla@fkie.fraunhofer.de

§ *Defence Research and Development Canada*, Ottawa, Canada
susan.watson@forces.gc.ca

¶ *University of Cantabria*, Santander, Spain
sanchez@teisa.unican.es

‖ *Florida Institute for Human and Machine Cognition (IHMC)*, Pensacola, FL, USA
nsuri@ihmc.org

** *US Army Research Laboratory (ARL)*, Adelphi, MD, USA
niranjan.suri.civ@army.mil

***Abstract*—Military missions typically involve joint coalition operations. Disadvantaged tactical networks in which they operate often suffer from limited bandwidth, intermittent connectivity, variable latency, and dynamic topology. Performance assessments have shown that civilian cloud technologies may be deployed in such networks to provide federated and adaptive cloud capabilities, enabling improved data sharing and processing capabilities between mission coalition partners. The NATO IST-193 RTG on *Edge Computing at the Tactical Edge* extends upon these previous performance results by addressing the system architecture challenges of distributing data and processing tasks amongst mission partners by means of federated and adaptive clouds in disadvantaged tactical networks. This paper describes the work on deployment orchestration and associated security challenges within the IST-193 RTG: its ambition, approach, status, and future work. This paper was originally presented in Skopje, North Macedonia, 16-17 May 2023.**

***Index Terms*—Deployment Orchestration, Disadvantaged Tactical Networks, Federated and Adaptive Cloud Architectures, Security, Tactical Clouds**

I. Introduction

Ever more sensing, processing, storage, and communication capabilities are available in military mission contexts for acquiring and processing data [1]. However, these capabilities are often connected over disadvantaged tactical networks, which are (still) characterized by, among others, limited bandwidth, intermittent connectivity, and variable latency. Therefore, federative and adaptive cloud architectures [2] become increasingly more relevant in joint coalition missions, providing major military and IT-operations benefits on data processing efficiency, survivability [3], and (as such) improvement of the overall military missions effectiveness [4].

The North Atlantic Treaty Organization (NATO) Information Systems Technology (IST)-168 Research Task Group (RTG) on *Adaptive Information Processing and Distribution to Support Command and Control* has quantified the performance of various cloud-oriented container orchestrators on emulated tactical networks [5]. The results indicate that state-of-the-art, commercial off-the-shelf, Kubernetes-based orchestrators could be deployed in a federated and adaptive cloud architecture for enabling data sharing and processing capabilities between mission partners over disadvantaged tactical networks. On these outcomes, the architecture challenges and potential solutions for deploying such a federated and adaptive cloud architecture in disadvantaged tactical networks can be further addressed. In this work, we take special interest in the deployment orchestration of data sharing and processing tasks and its associated security aspects. The deployment orchestration and security challenges are both related to the various topology options for the federative and adaptive cloud, for which increasing levels of complexity are foreseen, as previously identified in [6].

The NATO IST-193 RTG on *Edge Computing at the Tactical Edge* therefore addresses these deployment orchestration and security challenges and solutions for federative and adaptive clouds in disadvantaged tactical networks. The IST-193 RTG runs from April 2022 until March 2025. This paper describes the work being done on these aspects within the IST-193 RTG: its ambition, approach, status, and future work.

The reminder of the paper has the following structure. Section II provides an illustrative use case for the proposal. Next, Section III describes an adaptive and federated cloud topology for the tactical domain. This forms the basis for addressing its system architecture challenges: deployment orchestration (Section IV) and security (Section V). Then, Section VI highlights core concepts in an architecture suitable

for federated clouds and Section VII lays out related work. Finally, Section VIII provides conclusions and future work.

II. Illustrative Use Case

This section outlines an illustrative use case highlighting the potential benefits of a federated and adaptive cloud infrastructure in joint coalition operations. Fundamentally, such an infrastructure can provide the mechanisms to discover the mission partners' cloud resources, orchestrate services across the federation, and enforce security policies. This use case presents an urban operation in Wellport, a town in the fictitious country of Anglova, where multiple NATO nations (say A, B, C, and D) jointly participate.

During the operation, a dismounted soldier from one of the participating NATO nations (Nation A) detects a vehicle speeding away and captures video footage using a head-mounted camera. As the event seems suspicious, the soldier produces a formal intelligence report (i.e., a SPOT report) and sends it to the mission Headquarters (HQ). However, the vehicle's license plate was only vaguely recognizable and thus not included in the report.

Upon receiving the report, the HQ triggers a service workflow to find out the license plate of the suspicious car, which might carry insurgent leaders. Downloading the video feeds that might have recorded the target vehicle to a central location would easily overwhelm the tactical network. Therefore, the HQ uses the federated and adaptive cloud infrastructure to discover which countries have sensing capabilities (e.g., cameras) close to the soldier's position, thus deploying image super-resolution services where those feeds are stored—say, Nation B and C's clouds. Thanks to the deployment of image super-resolution services, the HQ obtains the license plate of the suspicious vehicle. The HQ then disseminates acquired intelligence to the tactical units deployed in the area to elevate situational awareness.

Subsequently, the vehicle is spotted heading beyond the city gates, where no nation has sensing capabilities. Upon receiving this information, the HQ decides to use the Unmanned Aerial Vehicle (UAV) made available by Nation D to follow the target as it speeds towards the countryside. Accordingly, Nation D deploys its UAV, which begins tracking the vehicle's movement using its onboard cameras. The vehicle eventually stops at a remote farmhouse—a potential hideout for insurgent leaders. With this information, the coalition forces can plan and execute operations to neutralize the threat.

As this use case illustrates, a federated and adaptive cloud infrastructure can facilitate efficient sharing and orchestration of resources in joint coalition operations, enabling better situational awareness and decision-making. Such infrastructure can also help mitigate the impact of disadvantaged network conditions, ensure secure access to mutually exposed cloud resources, and leverage partners' cloud capabilities, ultimately leading to improved operational outcomes.

III. Adaptive and Federated Cloud Topology

From an industry point of view, cloud computing utilizes a pool of general-purpose resources available on-demand to run services. Ideally, this pool is a location-agnostic environment where services can transparently migrate across available resources. Such a pool of resources is called a cloud.

The cloud topology can be described as an arbitrary amount of nodes, called cloud nodes, that are responsible for running services and managing the cloud itself. In addition, it is assumed that all cloud nodes interconnect through a reliable and fast network, and the cloud itself connects to other networks, e.g., the Internet, by a reliable and fast connection. For industry clouds, it is also often assumed to have extensive computation and available power.

For example, an enterprise-grade datacenter may consist of thousands of cloud nodes, interconnected by high-speed links and attached to a reliable power grid. Such a cloud is often in control of a single company. While cloud computing fits most of today's use cases, it can be inefficient in cases where data needs to be gathered at specific locations and then transferred back to the cloud for further processing. This inefficiency originates from a combination of a huge amount of data (e.g., high-quality video data) or high latency (e.g., real-time data) between the data sources (e.g., Internet of Things (IoT) or Internet of Battlefield Things (IoBT) devices) and the cloud, as shown in Fig. 1.

Edge computing [7] mitigates these problems by bringing data-processing services closer to data sources. Those data-processing services may run on devices either independent of the cloud or part of the cloud topology. A device being part of the cloud topology is a *cloud* node, which we further distinguish between *edge* and *core* nodes: a cloud node located closer to the data is an edge node, whereas a node located in a data center is a core node.

The proposed cloud topology for the military domain builds upon the general topology mentioned above. There is still a pool of general-purpose resources available on-demand to run services. This pool is also a location-agnostic environment where services can transparently migrate across available resources. However, either connectivity to the cloud or connectivity inside the cloud (or both) relies on disadvantaged tactical networks. In contrast to a general cloud, this kind of cloud is called a *tactical cloud*. For a tactical cloud, two scenarios need to be discussed. The first scenario deals with a tactical cloud at mission infrastructure. Such a cloud consists of a small data center at the compound level or on a ship. As a result, the connectivity between cloud nodes, in this context called Tactical Cloud Nodes (TCNs), is fast and reliable. Connectivity outside the tactical cloud is limited by the disadvantaged tactical network. The second scenario regards a tactical cloud spanning different platforms on the move, e.g., vehicles, ships, or aircraft moving in the theatre. Each platform is at least a TCN. In this case, connectivity outside (cloud-to-cloud/other network) and inside (TCN-to-TCN) the tactical cloud relies on disadvantaged tactical networks. The connectivity outside a cloud is called inter-cloud connectivity, whereas the connectivity inside a cloud is called intra-cloud connectivity, as depicted in Fig. 2.

The performance assessments described in [5] show that the

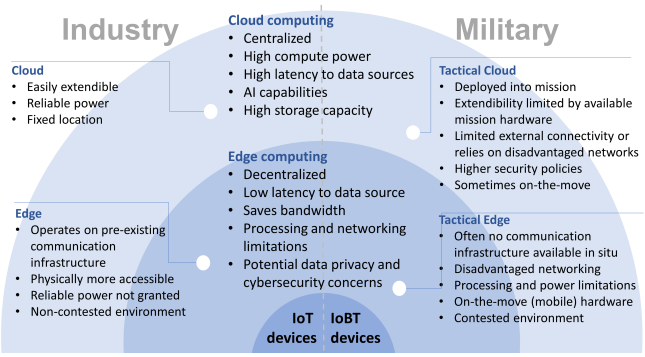

Fig. 1: Cloud computing: Industry vs. military.

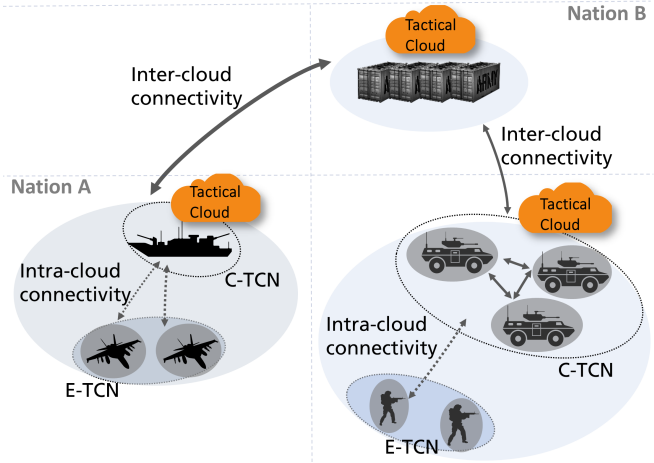

Fig. 2: Tactical clouds in the military domain.

number of TCNs within a tactical cloud is heavily limited by the disadvantaged network conditions. Therefore, in military missions there will be many, small clouds. This leads to a federation of clouds to use the available computing capabilities that are distributed among different tactical clouds.

Tactical clouds, and therefore its TCNs, can move in the theatre. Some nodes can be moved out of communication range of other TCNs, or can be moved in communication range of other tactical clouds. The necessity of moving such nodes depends on their capabilities, e.g., an UAV on reconnaissance or a vehicle with high computing power. Hence, these TCNs will be moved towards the target data sources, such as a geographic location for reconnaissance or another tactical cloud for data processing. While on the move towards a target data source, TCNs may also loose connectivity with the tactical cloud they belong to. Note that they act similarly to the edge nodes described above.

Given the dynamic nature of the tactical domain, a question arises: how to decide which nodes should form a tactical cloud? First, if a given tactical node shares a policy with other nodes (**shared-policy guideline**), it may make sense to cluster them. This means a tactical cloud should contain resources either owned by the same actor (e.g., the partner those resources belong to) or owned by different actors but carrying out a joint mission (e.g., NATO forces). Second, if managing a tactical node incurs an acceptable resource cost (**cost-effectiveness guideline**), it may make sense to cluster it. Typically, cloud management is mostly done by orchestrators on an intra-cloud connectivity level. The orchestration consumes network resources to accomplish orchestration-related tasks. This amount may vary from orchestrator to orchestrator. The orchestration overhead must be quantified in advance [5]. Tactical nodes that satisfy the shared-policy and cost-effectiveness guidelines are eligible for being TCNs. Third, if a given tactical node shares adequate connectivity over time with other tactical nodes (**connectedness guideline**), it may make sense to cluster them as Core TCNs (C-TCNs).

It is worth pointing out that some tactical nodes do not fulfill all previous guidelines (as the so-called C-TCNs do) but still may make sense to integrate their capabilities in a tactical cloud. Specifically, a tactical node may join a cloud as an Edge TCN (E-TCN) if it does not fulfill connectedness but provide capabilities that would not be available otherwise (**special-capability guideline**). A characteristic example is an UAV, which would likely not share good connectivity over time with any other node on the battlefield but offers disruptive capabilities for modern warfare. Typically, E-TCNs provide specialized resources in a location-dependent fashion.

IV. Deployment Orchestration

A federation of clouds implies multiple clouds, each running an individual control plane for orchestrating intra-cloud resources, which are connected to allow inter-cloud resource sharing. This leads to federated decision-making, where several control planes jointly orchestrate resources over partner clouds. Centralized or fully distributed approaches would have been alternative architectural design patterns. However, both of them do not fit coalition tactical operations.

The former assumes that a centralized location can get information about all nodes and links, prioritize requests based on coalition-wide policy, and make decisions about coalition resources accordingly. These assumptions, however, would not be realistic. Disadvantaged tactical networks typically consist of nodes on the move interconnected through wireless links. Such networks experience variable bandwidth and latency, intermittent connectivity, unreliability, frequent partitions, nodes running out of battery or going out of range, and adversarial attacks. Additionally, coalition tactical operations bring together several administrative domains, each potentially wanting to retain sovereignty (at least to some extent) over its resources. This excludes a centralized location to run operations.

The latter does not assume a centralized location but introduces inefficiency and complexity, especially as the number of nodes increases. A fully distributed approach implies that each node makes decisions independently. Such a node would make decisions by taking into account a partial picture of what is going on, leading to suboptimality. Also, the coordination algorithm to serve concurrent resource requests among nodes would be highly complex. For example, the same resources

may be requested by multiple nodes concurrently. The node serving those resources must decide to whom to lock those resources and reply back to the requesting nodes. The underlying disadvantaged tactical network(s) is supposed to be neither reliable nor timely. Therefore, replies may reach requesting nodes once they have already gotten the target resources from someone else. This process would occur for each request, making the resource coordination intractable at a large scale.

The federated approach is a hybrid between the centralized and fully distributed ones. A federation of tactical clouds consists of loosely coupled clouds, where each federated cloud retains sovereignty over its resources. This means partners retain granular control over the policies that define what others can see and do. As a result, the federated approach naturally fits scenarios involving several administrative domains, such as coalition tactical operations. Specifically, each tactical cloud runs an independent control plane to orchestrate resources internally (intra-cloud orchestrator). Note that an intra-cloud orchestrator is expected to allocate cloud-level resources optimally if implemented in a centralized fashion. The scope of an intra-cloud orchestrator is the set of resources jointly constituting an individual cloud. The inter-cloud orchestrator, which sits logically on top of the intra-cloud one, is responsible for federating resources across tactical clouds. It is worth mentioning that tactical clouds individually may opt not to disclose the complete internal picture of locally available resources to partner clouds. Therefore, the inter-cloud orchestrator may make suboptimal decisions based on the information quality available at a given time.

The existence of E-TCNs as part of a tactical cloud requires ad-hoc intra-cloud orchestration mechanisms because they may frequently lose connection with the intra-cloud orchestrator even for long periods. This means the intra-cloud orchestrator must make decisions about service deployment accordingly. For example, a service that does not require any specialized resource should be deployed on a C-TCN to increase its chance of being consistently available. Note that such a decision may not even be the fairest one. Let us assume that an E-TCN with far more resources than the C-TCN would also be available when the intra-cloud orchestrator was running the scheduling algorithm. Deploying the service to this E-TCN would be the fairest decision with respect to current compute resource allocation, but not with respect to long-term connectivity demands. Indeed, the E-TCN might later move away. This would force the intra-cloud orchestrator to re-schedule the service somewhere else with all the resulting drawbacks. It is worth mentioning that the physical movement of E-TCNs may be under control, such as an UAV following pre-defined flight plans, but also out of control, such as a sensor floating on the ocean. Additionally, even those potentially under control might change position unexpectedly as a result of adversarial attacks, environmental conditions, or changing mission requirements.

Note that inter- and intra-cloud orchestrators must continuously work to keep services running as demanded. The ever-changing circumstances might undermine links, nodes, and partner clouds unpredictably. For example, if an inter-cloud orchestrator had deployed a service on a partner cloud that went out of range, that service needs to be re-scheduled elsewhere. Additionally, a service with a higher priority might preempt resources previously allocated for other services. This, in turn, might cause the de/re-scheduling of one or more services with lower priority to make room. Therefore, a one-shot ("fire-and-forget") orchestration mechanism would not represent a feasible solution for the tactical domain.

V. Security Aspects

Deployment and orchestration of services across federated, multi-national tactical clouds in the military environment raises unique security requirements. Fundamentally, the notion of a federation of clouds implies that one nation that has developed a service would have that service instantiated and executed on a different nation's cloud infrastructure. From a security perspective, this requires careful consideration of the authentication and authorization processes to enable such deployment and activation, the container management environment and the isolation it provides, monitoring resource access and resource utilization, as well as access to locally residing data and to remote network endpoints.

As an example, consider that nation A might be willing to allow partner nation B to execute a service on nation A's cloud infrastructure. But, nation A might want to limit the execution to specific services, to limit CPU and memory utilization of those services, as well as to limit the network operations that the service might be allowed (for example, to ensure QoS for other services, or to prevent denial-of-service attacks). The opposite problem must also be considered – for nation B to trust that it can execute a service on another nation A's infrastructure. For example, nation B might be concerned that nation A might do data extraction from its service, nation A could have access to proprietary software and algorithms of nation B, or that false data would be provided to the service by the hosting nation.

Cloud-native security concepts can be organized according to the four layers of the cloud stack, namely: the application code; the containers; the cloud nodes; and the underlying infrastructure [8]. We examine security implications of tactical cloud federation at each layer, with a focus on the latter two layers.

A. Application and container

A service deployed across national cloud boundaries is essentially a cloud application running within a container, isolated from the underlying operating system. The attack surface of (cloud-native) application code itself can be reduced through code hardening practices and vulnerability scanning tools, ideally in a consistent manner across the federations of nations. When building containers, runtime isolation mechanisms can be leveraged, such as Linux namespaces to partition kernel resources, and cgroups to limit and control process resource usage (i.e., CPU, memory, disk I/O, network). Security policies (e.g., AppArmour or SELinux) configured to restrict

privileged operations. Known trusted container images to be deployed within the federation of clouds may be signed and managed by a federated certificate authority.

B. Cloud

The standard Kubernetes Application Programming Interfaces (API) employs a number of additional security concepts to provide protection at the individual cloud level.

To protect the cloud's control plane, Kubernetes uses role-based access control to limit access to the central API server component for both users and system components and workloads, together with an extendable authentication system. The API server allows for logging all incoming requests and authorization interactions for later auditing. A federated API server, enabling users to interact with multiple clouds, allows a central point for monitoring and enforcing authenticated, well-formed API accesses, and rate limiting for mitigation of denial-of-service attacks.

Workload protection and monitoring will be critical in the federated cloud concept. To protect workloads, the above-mentioned constraints on kernel resources for containers can be defined, together with restrictions on privileges for loading kernel modules or, e.g., mounting the filesystem as read-only. In addition, a fine-grained policy-based approach is required that can regulate the types of operations that might be allowed when a foreign nation is deploying, activating, and/or invoking a service on a local nation's cloud. Such a policy-based approach should enable specific operations by authorization only, so as to restrict in- and outgoing communication to other services, over specific ports, or to a set of IP blocks. The policy should also be able to limit the number of service instances that are allowed (e.g., no more than three service invocations) as well as the rate at which these invocations may be performed (e.g., no more than 10 service activations in 24 hours); this is important to prevent resource depletion attacks or faults. However, a related problem is that malicious nations may conduct attacks by advertising that they have cloud computing resources available, but then denying the advertised services to the other nations (e.g., by falsely advertising the resources available or throttling the services on purpose to provide a poor QoS).

Many external tools exist to facilitate identity & policy management. In particular, service meshes inject small sidecar containers that act as a proxy for service-to-service communication. This newly constructed data plane can then be controlled by service-mesh control plane components to enforce cloud-wide mTLS, apply traffic control, and provide observability.

C. Infrastructure

At the infrastructure level, network access to the cloud's control plane is typically limited to administrators through above-mentioned access control mechanisms; and communication to individual nodes and the cloud's storage backend restricted to only allow interaction with control plane components. The container orchestrator's components itself can also be hardened by running them as non-root user, and by opting for a secure container runtime (e.g. Kata Containers) and using microVMs (for example through Firecracker), or security sandbox (e.g., gVisor). Further, immutable operating systems can improve security through a read-only filesystem and disabling the use of SSH, and shell or console interaction. Regarding observability, many tools exist that provide telemetry and alerting for infrastructure- and cloud-level resource usage (CPU, memory), as well as lower-level system calls through eBPF-based monitoring, or give security-specific insights by probing for possible attack vectors. While each nation will be responsible for monitoring its own cloud, the question arises of how to share telemetry and event data for the purposes of implementing a monitoring function at the federated multi-cloud level.

D. Security Challenges for Federated Tactical Clouds

While many existing cloud-native security approaches can be applied, we find there are certain challenges that are unique to the federated cloud concept. Of these, many are trust challenges; such as, how to monitor and track services' (that is, nations) adherence to best practices and good behavior when these services are deployed and running. Reputation services and monitoring could play a central role in addressing some of these trust challenges in multi-national cloud computing. Subsequently, it is obvious that adding extensive security, isolation, and monitoring mechanisms incur a performance penalty. However, further resources will be required to support careful auditing and resource monitoring of services and their performance on partner nation cloud environments. In disadvantaged tactical networks, where power, compute, and network resources are scarce to begin with, it becomes especially relevant to balance security measures against resource consumption.

VI. Architecture Overview

An architecture for a federated and adaptive cloud infrastructure to function effectively in a military context has to take the previously stated guidelines (see Section III) as well as the inter- and intra-cloud orchestration capability (see Section IV) into account. Additionally, each cloud should have sovereignty over its infrastructure, data, and services due to individual security and classification policies. Also, it should adopt a vendor-agnostic approach, while ensuring interoperability and compatibility between partner clouds. This can be achieved through standards-based cloud container technology, such as the Open Container Initiative (OCI) and Kubernetes as the intra-cloud orchestration provider, while information interoperability is facilitated by using standardized NATO services. In Fig. 3, a high-level architecutre overview is depicted.

The architecture utilizes the already deployed kubernetes infrastructure, by adding services that are responsible for federation by exposing specific capabilities through APIs. The four services that are deployed on each cloud are:

- Federated Resource Discovery (FRD) - the FRD is responsible for gathering information about available re-

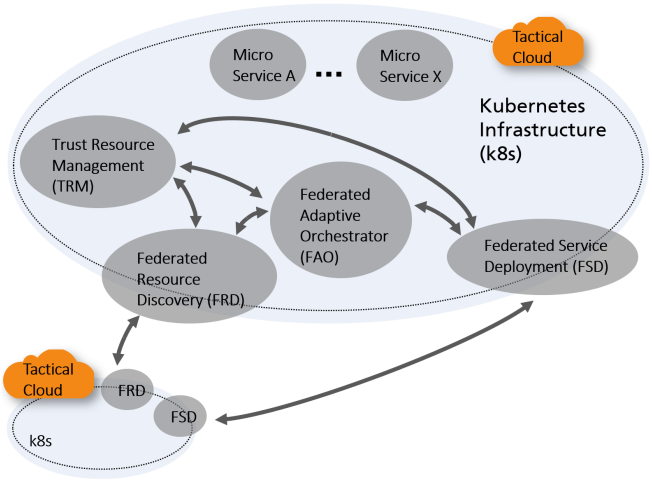

Fig. 3: Federated cloud architecture overview.

sources, such as CPUs, RAM, storage, etc., and what kind of services are available for deployment and already deployed across all clouds in the federation. Therefore, it queries its own cloud and available other clouds for this information. Additionally, it exposes resource and service information about its own cloud through an external API in regards to any policy constraints.

- Federated Service Deployment (FSD) - the FSD is responsible for deploying services or parts of services, e.g. a database, either on its own cloud or on other available clouds. A request for deployment can be initiated by either its own orchestrator or by another FSD. It also exposes an external API.
- Federated Adaptive Orchestrator (FAO) - the FAO is responsible for orchestrating service deployments across available clouds, as described in IV. Its uses information from the FRD and initiates service deployments or updates through the FSD. It does not expose an external API.
- Trust Resource Management (TRM) - the TRM is responsible for monitoring activities in a single federated cloud and updates its policies that other services, FSD, FRD, FOA, must enforce.

Each of these capabilities should be available on each cloud. Although their implementations can be different among clouds, the two exposed APIs from the FRD and the FSD must be the same. Only then a cross-cloud service deployment in a federated and adaptive cloud infrastructure could be possible.

VII. Related Work

This section gives an overview of the related work that addresses challenges that arise when dealing with federated and adaptive cloud infrastructures in tactical networks. The outlined solutions on the one hand often deal with specific problems highlighted in this paper and on the other hand miss another specific problem, like in Kubefed [9], by taking a centralized approach in managing the federation, whereas this paper uses a decentralized approach. Hence, the related work presented in the following collectively contribute to the development of a comprehensive approach for federated and adaptive cloud infrastructure in tactical networks.

A. Federated Cloud Orchestration

There is a need to synthesize and process information that has been generated at the edge. [10] gives an extensive overview of existing edge computing architectures, and aggregate insights into a proposed Global Edge Computing Architecture, showing results in an Industry 4.0-based use-case with specific attention to security. [11] and [12] give general insight into (Kubernetes-based) edge, and federated orchestration respectively, and assert that current shortcomings in cloud-edge scheduling are related to a lack of real-time resource-aware and network(-topology)-aware scheduling and a lack of suitable provisioning models. They specifically note a need in federated orchestration for exposing inter-cloud telemetry (especially metrics on network characteristics); application performance profiling; as well as improved policies for application placement that also take into account application's resource usage, underlying communication model, and task interdependence. Finally, a need for improved straggler detection and management is identified.

Many solutions have been proposed to address the various challenges in creating federated clouds and orchestrating workloads on top of those multi-cloud environments. Specifically, we can distinguish solutions that are aimed at inter-cloud networking; solutions that focus on inter-cloud workload orchestration; or hybrid solutions that try to tackle both. Most of these solutions, however, assume a static high-bandwidth scenario, and are not tailored to constrained and uncertain edge environments where clouds may join in an ad-hoc fashion. Contrarily, there are solutions that focus on resource-constrained edge orchestration, but only from a single cloud point of view. In multi-cloud solutions, a perspective that is often applied is that these multiple clouds are administered by a single entity, and can be managed from a central point. This assumption relaxes both security and architectural constraints, but does not hold in our multi-nation scenario. Finally, there exist solutions to enable multi-tenancy and isolate workloads belonging to other tenants on a cloud, but these, again, often assume an abundance of resources (energy, bandwidth, etc.) to be available.

B. Multi-cloud networking

Multi-cloud networking solutions that focus on connecting multiple clouds together and support, e.g., multi-cloud service discovery and invocation, and inter-cloud network policies, include approaches that are incorporated into container network interfaces, such as Cilium ClusterMesh [13]; those that are service mesh-based, routing internal and inter-cloud traffic through lightweight proxies, such as Istio [14]; and standalone solutions such as Submariner [15], which uses a broker architecture to facilitate multi-cloud discovery. In the federated workload orchestration space, solutions like Kubefed

[9] take a centralized approach; while others such as Admiralty [16] or Liqo [17] also support decentralized (peer-to-peer) topologies, but require a fair bit of trust between collaborators. Some, such as mck8s [18] or Tensile-kube [19], build on previously mentioned technologies like Cilium ClusterMesh to facilitate inter-cloud networking, and add federated workload management capabilities.

C. Edge computing

In the edge domain specifically, FLEDGE [20] enables small edge devices that are constrained for resources to join a larger cloud and boosts more efficient container orchestration for resource-constrained environments, but does not incorporate multi-cloud functionality. [21] takes a decentralized approach to the scheduling problem, allowing nodes to bid on resource allocation inspired by the auction house model, whilst maintaining guarantees on solvability. KaiS [22] is an alternative, learning-based, scheduler that aims to improve system throughput in cloud-edge environments by dynamically learning scheduling policies with deep neural network-based models. Other approaches, such as the one proposed in [23], aim to relax the latency requirements of Kubernetes by employing a different (conflict-free replicated data type-based) storage backend which is more robust to high latency but still provides guarantees on eventual consistency and therefore more applicable to edge-like environments.

D. Resource sharing and computing

Regarding secure federated resource sharing and computing, there exist different options to isolate workloads of other tenants on a shared or federated cloud infrastructure. Ranging from hosting fully seperated clouds; to employing secure container runtimes to run workloads for other tenants; to custom multi-tenancy solutions like vcluster [24]; to more lightweight options such as using existing Kubernetes constructs to improve namespace isolation [25]. In the military domain, Nexium [26] aims to facilitate Federated Mission Networking (FMN)-compliant cloud-edge interoperability.

VIII. Conclusions and Future Work

As follow-up of our previous work in the IST-168 RTG on the performance assessment and feasibility of common off-the-shelf Kubernetes distributions for deployment in federated and adaptive cloud architecture in disadvantaged tactical networks, the IST-193 RTG addresses the system architecture and potential solutions for their further development. Based on the complexities of the topology of the federation of clouds in disadvantaged tactical networks, this paper focused on the resulting challenges for deployment orchestration of data sharing and processing tasks and the associated security aspects.

The coalition perspective on a federated and adaptive cloud architecture as described in the paper forms the basis for IST-193 RTG's work towards an improved exploitation of available information in disadvantaged tactical networks. Future work items includes continuing to better characterize the problem domain as well as exploring specific topics that include: i) implementation of architecture, design guidelines, and methodologies for deployment orchestration and security, ii) further assessment of their suitability in disadvantaged tactical networks, and iii) input for and alignment with the NATO FMN architecture and development.

Acknowledgment

The work presented in this article is being done as part of the IST-193 RTG on *Edge Computing at the Tactical Edge*. We would like to thank the NATO IST-Panel for providing us the opportunity to do this highly relevant and interesting research and the individual participating partners for providing valuable inputs within a stimulating and cooperative setting.

References

[1] M. Tortonesi, A. Morelli, M. Govoni, J. Michaelis, N. Suri, C. Stefanelli, and S. Russell, "Leveraging internet of things within the military network environment — challenges and solutions," in *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, 2016, pp. 111–116.

[2] H. Bastiaansen, J. v. d. Geest, C. v. d. Broek, T. Kudla, A. Isenor, S. Webb, N. Suri, M. Fogli, B. Canessa, A. Masini, R. Goniacz, and J. Sliwa, "Federated control of distributed multi-partner cloud resources for adaptive c2 in disadvantaged networks," *IEEE Communications Magazine*, vol. 58, no. 8, pp. 21–27, 2020.

[3] K. Zaerens, "Enabling the benefits of cloud computing in a military context," in *2011 IEEE Asia-Pacific Services Computing Conference*, 2011, pp. 166–173.

[4] W. Smith, G. Kuperman, M. Chan, E. Morgan, H. Nguyen, N. Schear, B. Vu, A. Weinert, M. Weyant, and D. Whisman, "Cloud computing in tactical environments," in *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)*, 2017, pp. 882–887.

[5] T. Kudla, M. Fogli, S. Webb, G. Pingen, N. Suri, and H. Bastiaansen, "Quantifying the performance of cloud-oriented container orchestrators on emulated tactical networks," *IEEE Communications Magazine*, vol. 60, no. 5, pp. 74–80, 2022.

[6] N. Suri and A. C. Scott, "A perspective on defining the collective adaptive systems problem," in *2014 IEEE Eighth International Conference on Self-Adaptive and Self-Organizing Systems Workshops*, 2014, pp. 26–31.

[7] K. Bilal, O. Khalid, A. Erbad, and S. U. Khan, "Potentials, trends, and prospects in edge technologies: Fog, cloudlet, mobile edge, and micro data centers," *Computer Networks*, vol. 130, pp. 94–120, 2018.

[8] "Overview of cloud native security," https://kubernetes.io/docs/concepts/security/overview/, [Online; accessed 20-June-2022].

[9] "Kubernetes cluster federation," https://github.com/kubernetes-sigs/kubefed, [Online; accessed 20-June-2022].

[10] I. Sittón-Candanedo, R. S. Alonso, J. M. Corchado, S. Rodríguez-González, and R. Casado-Vara, "A review of edge computing reference architectures and a new global edge proposal," *Future Generation Computer Systems*, vol. 99, pp. 278–294, 2019.

[11] S. Böhm and G. Wirtz, "Cloud-edge orchestration for smart cities: A review of kubernetes-based orchestration architectures," *EAI Endorsed Transactions on Smart Cities*, vol. 6, no. 18, 2022.

[12] D. Lindsay, G. Yeung, Y. Elkhatib, and P. Garraghan, "An empirical study of inter-cluster resource orchestration within federated cloud clusters," in *2021 IEEE International Conference on Joint Cloud Computing (JCC)*, 2021, pp. 44–50.

[13] "ebpf-based networking, security, and observability," https://github.com/cilium/cilium, [Online; accessed 20-June-2022].

[14] "Connect, secure, control, and observe services," https://github.com/istio/istio, [Online; accessed 20-June-2022].

[15] "Connect all your kubernetes clusters, no matter where they are in the world," https://github.com/submariner-io/submariner, [Online; accessed 20-June-2022].

[16] "A system of kubernetes controllers that intelligently schedules workloads across clusters," https://github.com/admiraltyio/admiralty, [Online; accessed 20-June-2022].

[17] "Enable dynamic and seamless kubernetes multi-cluster topologies," https://github.com/liqotech/liqo, [Online; accessed 20-June-2022].

[18] M. A. Tamiru, G. Pierre, J. Tordsson, and E. Elmroth, "mck8s: An orchestration platform for geo-distributed multi-cluster environments," in *2021 International Conference on Computer Communications and Networks (ICCCN)*, 2021, pp. 1–10.

[19] "A kubernetes provider," https://github.com/virtual-kubelet/tensile-kube, [Online; accessed 20-June-2022].

[20] T. Goethals, F. De Turck, and B. Volckaert, "Fledge: Kubernetes compatible container orchestration on low-resource edge devices," in *Internet of Vehicles. Technologies and Services Toward Smart Cities*, C.-H. Hsu, S. Kallel, K.-C. Lan, and Z. Zheng, Eds. Cham: Springer International Publishing, 2020, pp. 174–189.

[21] C. Avasalcai, C. Tsigkanos, and S. Dustdar, "Decentralized resource auctioning for latency-sensitive edge computing," in *2019 IEEE International Conference on Edge Computing (EDGE)*, 2019, pp. 72–76.

[22] Y. Han, S. Shen, X. Wang, S. Wang, and V. C. Leung, "Tailored learning-based scheduling for kubernetes-oriented edge-cloud system," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 2021, pp. 1–10.

[23] A. Jeffery, H. Howard, and R. Mortier, "Rearchitecting kubernetes for the edge," in *Proceedings of the 4th International Workshop on Edge Systems, Analytics and Networking*, ser. EdgeSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 7–12. [Online]. Available: https://doi.org/10.1145/3434770.3459730

[24] "Create fully functional virtual kubernetes clusters," https://github.com/loft-sh/vcluster, [Online; accessed 20-June-2022].

[25] "Multi-tenancy and policy-based framework for kubernetes," https://github.com/clastix/capsule, [Online; accessed 20-June-2022].

[26] "Nexium defence cloud edge | thales group," https://www.thalesgroup.com/en/nexium-defence-cloud-edge, [Online; accessed 20-June-2022].