Assessing System of Systems Security Risk and Requirements with OASoSIS

Duncan Ki-Aries, Shamal Faily, Huseyin Dogan
Bournemouth University
Fern Barrow, Poole, UK
{dkiaries,sfaily,hdogan}@bournemouth.ac.uk

Christopher Williams
Defence Science and Technology Laboratory
Porton Down, UK
cwilliams@mail.dstl.gov.uk

***Abstract*—When independent systems come together as a *System of Systems* (SoS) to achieve a new purpose, dealing with requirements conflicts across systems becomes a challenge. Moreover, assessing and modelling security risk for independent systems and the SoS as a whole is challenged by a gap in related research and approaches within the SoSs domain. In this paper, we present an approach for bridging SoS and Requirements Engineering by identifying aligning SoSs concepts to assess and model security risk and requirements. We introduce our OASoSIS approach modifying OCTAVE Allegro for SoSs using CAIRIS (Computer Aided Integration of Requirements and Information Security) with a medical evacuation (MEDEVAC) SoS exemplar for Security Requirements Engineering tool-support.**

***Index Terms*—System of Systems, Security, Risk, Human Factors, Requirements Engineering, CAIRIS.**

I. Introduction

For independent systems coming together as a *System of Systems* (SoS) to achieve a greater collaborative purpose whilst also maintaining their own 'day job', assessing security risk between these inter-dependent systems of the SoS presents greater challenges than that of a single system's focus on its own people, process, software and hardware, integrated to achieve a purpose. The focus and challenges in SoSs are increased by multi-stakeholder collaborations, creating new risks from the evolution, interoperability needs, and emergent behaviours by the SoS coming together. This convergence provides a set of systems for a task that none can accomplish on their own. Each independent system continues to retain their own management and operations, whilst co-ordinating with the SoS, adapting to meet additional SoS goals [1].

SoS dynamics often depend upon the type and level of management and collaboration from independent systems, their sub-systems, and varying trust boundaries. Given the differences in managerial and operational control of SoS examples discussed in previous work [2][3][4], challenges arise where there is a weak collaboration or trust relationships providing limited information. Further complexities form when integrating multiple independently managed systems and requirements that need to be co-ordinated in order to achieve the SoS objectives [5]. Having detailed information of the SoS interactions as a whole may therefore not be available or achievable in some SoS scenarios, yet we need to understand the given SoS scenario if we are to identify security risks and mitigating requirements. Therefore, identifying the minimum level of detail to adequately assess SoS security risk is a challenge, certainly towards bridging operational needs of independent systems to Requirements Engineering (RE), meeting the criticality of the independent requirements accurately reflecting interdependent users' needs crucial to the success of the RE in the SoS [6][7].

Although some engineering methods exist for SoS engineering, e.g. [1][8][9], further work is required towards how we may assess and model security risk in SoSs. There appears to be no SoS focused security risk approach or tool-support to model and visualise SoS security risk, helping to bridge the communication gap between operational needs and RE. There are a range of tools or approaches designed for a single system context, but no clear guidance or limited tool-support integrating different modelling elements to visualise and assess the SoS security consequences in greater detail. There is a need for better models visualising how various people approach a security task, their mental models or security-related skills and knowledge. Current informal and implicit models of people are not always robust enough or rarely focus on how people make security decisions [10]. Identifying and integrating combinations of tool elements to suitably visualise these elements in a SoS context is required to account for independent and interdependent system interactions of a SoS.

To address this need, we build on previous work by continuing to identify the alignment of SoS factors and concepts suitable for eliciting, analysing, and validating SoS security risks using tool-support. We implement a version of the Operationally Critical Threat, Asset, and Vulnerability Evaluation process of OCTAVE Allegro (OA) for Information Security risk assessment [11]. Outputs from OA are then aligned with tool-support from the open-source Computer Aided Integration of Requirements and Information Security (CAIRIS) platform [12]. Because it automatically analyses and visualises design data as it is added, it is potentially useful for modelling different perspectives during SoS security risk and RE activities.

By combining the use of OA for SoS with CAIRIS, we refer to this combination as OASoSIS. We describe related work in Section II before introducing the OASoSIS approach and its related medical evacuation SoS case-study, the *MEDEVAC Mission Network* in Section III. A discussion of findings and lessons learnt are discussed in Section IV, with conclusions towards future work in Section V.

II. Related Work

A. Systems of Systems

Systems are a group of regularly interacting interdependent elements forming a unified whole [1], although this unification needs to be considered to understand and appreciate its complexities, and how the system forms as a whole [13]. We can summarise Systems as *'a coming together of people, process, software and hardware, integrated to achieve a purpose'*. Organisational and technological systems are decomposed of various sub-systems and component systems interconnecting to fulfil system needs. The term "System of Systems" is often applied in different scenarios with varying scale or complexity of interconnected systems as detailed further in previous work [2][3][4]. There are many examples of systems built and used for one purpose, and interconnected with a SoS for another. These range from small-scale Internet of Things (IoT) devices, software dependent systems, emergency response units, larger-scale military operations, Smart cities, and the over-arching role of critical infrastructure, e.g. [2][14][15]. Despite the limited range of SoS-based engineering guides or literature, many of them reproduce SoS descriptions and definitions largely founded and supported by Maier [16] along with Dahmann and Baldwin [17], bringing together the main four categories of SoSs. These are summarised as:

- *Directed SoSs* possess central management, operation and control over the SoS as a whole;
- *Acknowledged SoSs* have designated management, but limited control over the SoS as a whole;
- *Collaborative SoSs* have no central management, so operation and control is formed and agreed as a mutual independent collaboration;
- *Virtual SoSs* have individual independent collaboration with no central management, operation or control of the SoS as a whole.

Given the inherent socio-technical nature of SoSs, we also need to account for the people in SoSs, and the effect uncertainty might have towards risk. Trust and assurance are also important factors, particularly as SoSs evolve [18].

B. SoS Risk Assessment and OCTAVE

Attending to risk in a SoS depends on its type and complexity, and should consider a range of risk-based contexts. However, many systems of a SoS may not have gone through the same risk or security processes, presenting the potential for new risks across the SoS [5]. It is, however, likely that systems may have applied one of a number of methods for security risk management covering a wide range of security techniques, controls and considerations towards security protection, e.g. [19][20]. Used in parallel, risk assessment approaches such as the Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) methodology provide a thorough approach for risk assessment in large systems, and offers three differing levels of skill and application. For example, OCTAVE Allegro (OA) is suitable for assessing Information Security risk without the need for extensive risk assessment knowledge, while reducing the need for participatory workshops and interaction from all organisational system levels [11], thus benefiting the SoS context of differing levels of collaboration across stakeholders. It should, however, aim to identify where independent system changes alter risk equations that might go unidentified [21]. Dealing with security risk within or from the supply chain requires further consideration as the SoS attack surface grows, requiring assurance of security throughout acquisition and the development life-cycle [22].

In SoSs, risks and mitigations need to focus on desired outcomes against undesirable events and emergent behaviours of the SoS. Emergent behaviours can form due to development or evolutionary processes coming together in the SoS, evolving through the on-going interactions and collaborations [16][5]. Moreover, achieving interoperability depends on the ability of two or more systems or elements to use and exchange information, thus creating further challenges for other security related aspects along the communication channels between systems and the external world [23][24]. The communication between independent system stakeholders and RE is essential for achieving end-to-end risk reduction in SoS security.

To address the risk, security requirements should begin with asset analysis and the context in which they are in [25]. Security risk assessment considers these asset interactions in the operational and developmental life-cycles, and should continue to focus on the human factors and interoperability constraints critical for the SoS operation, leading to applicable risk reducing requirements. This may be addressed using Security RE approaches, e.g. Security Quality Requirements Engineering (SQUARE) [26], or other methods to elicit and prioritise security requirements. There are a number of security requirements approaches, one of which specifically considers vulnerability assessment as a vehicle to requirements, although we argue this should apply regardless given it forms the risk equation [27]. Nonetheless, few approaches exist towards translating SoS security risk to requirements [28].

C. Models and Tool-support

To further support the output of a SoS security risk assessment and enhance the reasoning behind security concerns during development, good tool-support is required that can integrate with other current development tools or be used by other stakeholders [29]. Sharing models with others contributes to greater awareness of security issues, although better models are required to visualise how various people approach security tasks across the SoS. Models of SoSs need to capture the role of each independent system, its SoS purpose, mission and requirements, and the implications of interactions of different security decisions. The modelling may use a combination of top-down and bottom-up processes, but would require modelling of system goals in the SoS context [7]. A range of modelling tools or approaches can potentially be used to assist a model-based SoS risk assessment. For example, Secure Tropos [30] can be used to model stakeholders, system and social goals, and the impact of risk-related concepts on these goals. The CORAS method to risk analysis uses a tool

designed to support documenting, maintaining and reporting security analysis, using UML based threat and risk modelling to capture and model relevant information [31].

These type of RE modelling approaches are often designed or used in a single system context, therefore, identifying and integrating combinations of tool elements to suitably visualise these elements in a SoS context is required to account for independent and interdependent system interactions of a SoS. Moreover, to provide assurance that countermeasures address risks, the behaviours of attackers, vulnerabilities, and threats need to be understood [32]. Models help reason about these concerns, but are time-consuming to build, and maintaining model consistency when changes are made is expensive [33].

In recent years, the open-source Computer Aided Integration of Requirements and Information Security (CAIRIS) platform [12] has evolved, providing the potential for eliciting, specifying, and validating secure and usable systems. Several types of system models can be automatically generated based on requirements, security, and usability model elements added to a CAIRIS model, e.g. goal, task, asset, and risk views. These are situated in model *environments* that can be used for each independent system of the SoS to capture the contexts of use within which a system specification needs to be situated for. CAIRIS can be used to facilitate collaboration between different types of systems stakeholder, and its API can be used to facilitate integration with complementary tools [34]. Most CAIRIS model concepts, such as goals, assets, and threats, can be predicated by environment. This makes it possible to explore the impact of a threat on different systems, and which can be shared and discussed with stakeholders.

III. OASoSIS Approach and Application

To aid the information gathering of the SoS when using OA, we integrated a process based on recent work characterising a SoS, identifying system stakeholders, levels of operational and managerial control, and main system interactions of the SoS [4]. This *Step 0* of the OA SoS process is required to guide the minimum amount of information to determine the scope of the independent system collaboration and its interdependencies, specifically where SoS managerial and operational control is in place, if at all. This may, however, present challenges for some systems or types of SoS where there is a weak collaboration or trust relationships providing limited systems and risk-based information. Continuing with Step 1 of OA, we extend the standard or suggested Risk Criteria impact area with elements of a combined Human System Integration (HSI) and Human Factors Integration (HFI) approach [35][36] – HFSI – to acknowledge human related impacts of the SoS. We then considered how using CAIRIS as tool-support could align with OA to increase the efficiency of using data collected from OA to model the SoS interactions, threats, vulnerabilities, and risk, and aligned the CAIRIS risk ratings into OA. CAIRIS would primarily be used to elicit, model, and visualise security risks, producing security requirements as an output.

A. System of Systems Exemplar Scenario

To apply and test the tool-supported security risk assessment approach for SoSs, we first implemented a reduced-scale exemplar of a Military MEDEVAC SoS case-study as described in [4]. The *MEDEVAC Mission Netwok* (MMN) consists of a typical patient data-flow and interconnections of three collaborating independent systems – *Alpha*, *Bravo*, and *Charlie*. These are representative of a relationship such as a NATO operation with two Troop Contributing Nations (TCNs), coming together as independent systems collaborating to achieve a higher purpose; to perform a continuum of care through medical evacuation. *Alpha* provides designated management with Command and Control, whereas *Bravo* triggers the MEDEVAC process, and *Charlie* provides the systems for forward transportation and medical facilities. Each system is also reliant on other sub-system interactions to fulfil the continuum of care.

Although the selection of a military system may seem a surprising choice for a SoS exemplar, it was motivated by several reasons. First, armed forces around the world rely on a symbiotic relationship between people, processes, and technologies, and their systems have been designed with emergence in mind. Second, many goals that armed forces are called in to achieve rely on *coalition* forces. Each TCN to this coalition relies on its own people, processes, and technologies, and while each contribute to achieving an overall SoS mission goal, each nation may have other goals that conflict with the goals of other nations. The Afghan Mission Network (AMN) used by NATO forces in Afghanistan is an example of such a SoS [2]. Third, as a corollary of the second, there is much publicly available data on military SoSs, e.g. doctrine documents that summarise SoS goals, assisting with the identification of related requirements for the scenario.

The full MEDEVAC continuum of care provides additional patient evacuation co-ordination to other stage hospitals outside the area of operation, often leading to repatriation. In this scenario, we focus on the initial MEDEVAC mission goal for *Bravo* to initiate the process with *Alpha*, then for *Charlie* Forward Air MEDEVAC to transport a patient from the Point of Injury (PoI) to a Forward Surgical Team (FST) within one hour – *The Golden Hour*. This scenario includes certain stakeholders within the chain of care responsible for retaining and communicating patient information at each stage. Tracking casualty movement from PoI through to repatriation is required to regulate the treatment and flow of casualties, providing effective correctly documented treatment meeting patient, organisational and regulatory needs [37]. Patient data is at the centre of the continuum of care and provides a focus for testing our approach when considering examples of critical information assets within the SoS security risk assessment.

B. Applying the OASoSIS Approach

1) Modifying OCTAVE Allegro: One aspect of choosing OA related to the benefits of reducing stakeholder interaction, as this would otherwise be a challenge across all systems of the SoS. Nevertheless, within a risk management process, the risk

assessment requires an amount of information gathering to identify data assets and associated system asset interactions where data may be processed, stored, and transmitted. To tailor the information gathering, our approach used in [4] was used to frame the SoS context, identifying the type of SoS by its characteristics from the given scenario. For example, understanding where various management and control is in place for systems and the SoS, indicating where accountability or conflicts may exist. This formed a new Step 0 for OA.

In Step 1, system stakeholders are likely to be relied upon to collaboratively agree the criteria in which risk may impact upon the system's SoS interaction, and within which financial parameters. For example, the impact of a risk may come with financial penalties, and the criteria is used as a scale representing a low to high impact of risk. Much of the standard criteria gives focus towards business impacts, but accounts less for the impact on human factors. Given the socio-technical nature of SoSs, aligning the concept of HFSI in Step 1 aimed to address this gap. As the criteria is prioritised, e.g. 10 to 1, with 10 holding the highest importance, balancing business and human needs or impact will likely require further stakeholder discussion, particularly in SoSs where safety is paramount. We then multiply these criteria levels against each asset impact level, then multiply again against the probability to account for the likelihood of the impact and severity within the overall risk score for the system interaction with the SoS.

However, a further question that needs to be considered is whether the risk criteria is related to the impact on the individual system, or the SoS as a whole. In some scenarios, a unified criteria may be agreed upon; in other scenarios, systems may only be able to assess their own interaction with the SoS or elements of it. For this reason, in this iteration of testing our approach, each system criteria would be related to the impact on itself integrating with the SoS. This allowed for the example where a *Bravo* impact of £50,000 could be catastrophic, whereas the same upper limit of *Charlie* could be $500,000. Once the criteria is agreed, each system may continue with other steps detailed in OA [11].

2) Applying OCTAVE Allegro: Using the modified OA as the first element of OASoSIS, Steps 0-7 were used to produce an example security risk assessment using the MMN from the view of one independent system, *Bravo* and their interaction with the SoS, then later repeating the process for other system assessments. Having characterised the MMN scenario as an Acknowledged SoS, this process identified relevant stakeholders, boundaries, and where managerial and operational independence and control are in place for MMN, pointing to areas of dependency, complexity, and potential risk. Much of this information helped to identify critical assets that supported steps within OA, and which could be later translated into CAIRIS asset models with related roles, personas, tasks, and other associations.

By the nature of OA, documenting threats and concerns of critical patient information assets could be spread out over many sheets of paper for a single asset. For flexibility, this was instead entered into spreadsheets, but later converted to a single line all-in-one spreadsheet, considering areas of concern for the process, storage and transmission of data, by people, physical, and technical means, then assessed the impact and probability of the occurrence. Information assets with areas of concern that indicated higher probability and severity risk scores were selected for further modelling using CAIRIS, although the challenge was to identify how and where this information can be extracted from OA into CAIRIS.

3) Integrating CAIRIS: To begin modelling a SoS in CAIRIS, a separate environment was created to represent the view of each independent system and an additional overview environment to capture all interactions. In the initial *Bravo* view, we first populated an asset model where an asset is used to represent the SoS as a single entity. This SoS could then be decomposed using a top-down approach associating each of the main independent systems, their sub-systems and information assets, and the known interactions between the constituent systems where *Bravo* has direct interaction. Associations may also be an aggregation or composition to the operation of its parent system. This was later repeated for the other systems providing a bigger picture.

Where we describe Systems as *'a coming together of people, process, software and hardware, integrated to achieve a purpose'*, these are represented at higher level as an organisational level system asset who may in turn have lower level organisational systems, each of which have technological systems where human actors interact with software/hardware combinations. Given the theme is Information Security, information or data assets may also be physical and paper-based, a person and the knowledge they hold that may be communicated verbally, and which may then be entered into a software interface and database, creating an electronic version of the data.

All main assets within the continuum of care were modelled and associated with roles of key stakeholders and actors performing the continuum of care, reflecting areas of responsibility for systems. This included certain activities and tasks carried out by specific roles undertaken by a person. Specific risks highlighted in OA also helped link these activities where a data asset may be at risk by a human, accidentally or maliciously. Roles were then associated with personas, representative of archetypical descriptions embodying the goals of business users offering insights into threats, vulnerabilities and likely areas of risk that may otherwise be overlooked [38]. Attackers were modelled and assessed in a similar way, reasoning about the intent, skill, or means of an attack by an actor internal or external to the SoS.

Although descriptive personas may not have been fully implemented initially, they were later populated to further reason with human factor considerations and the consequence of actions when assessing security risk and related requirements. To do this, CAIRIS supports the alignments of Toulmin argumentation models to justify persona characteristics [39]. The Persona Helper Chrome plugin [40] was used to capture factoids from online and offline data, such as a webpage and clips of text within it. These factoids were stored within CAIRIS, and exported to a Trello board [41] that itself can be

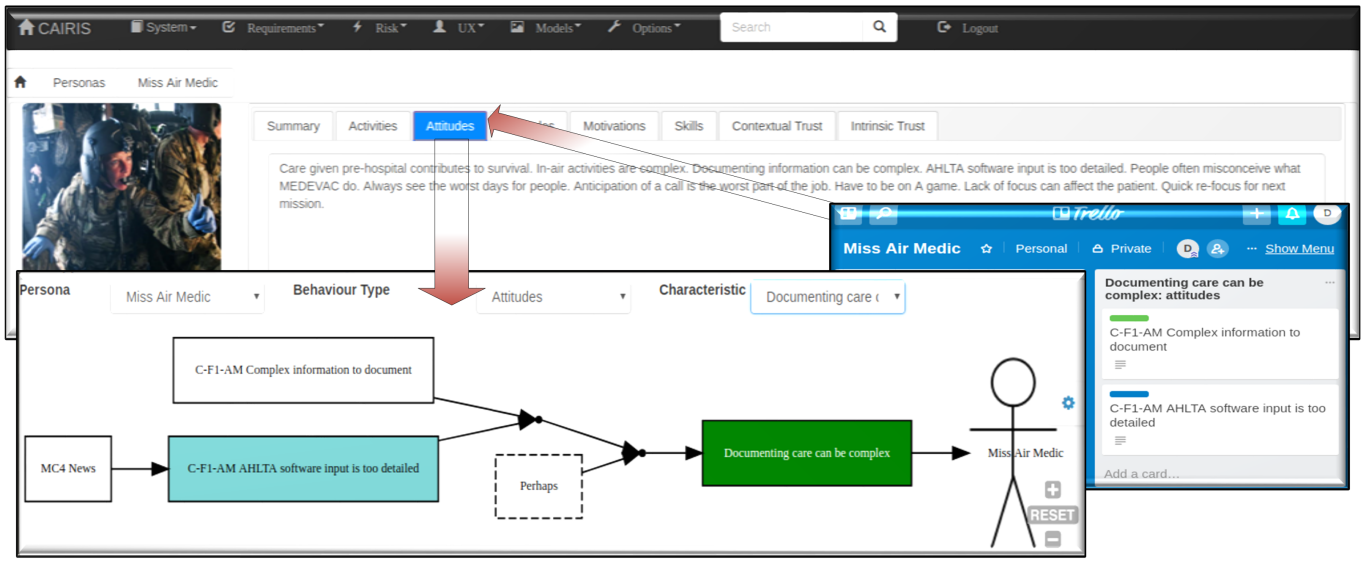

Fig. 1. CAIRIS Persona Characteristics and Model with Trello

used as part of the affinity diagramming process. Elements of this relating to an Air Medic persona characteristic is shown in Figure 1. Once the factoids were grouped into characteristics, these were marked as a grounds, warrant or rebuttal supporting the argumentation of the characteristic, and imported directly back into CAIRIS to create a persona and related model.

Personas were associated with tasks, and use cases were created and linked to represent steps of the task. The use case and its sub-steps represented the process for completing a step carried out by an actor of a task. In this scenario, we have considered steps where no software-hardware interaction may occur with physical patient data, but led to steps where this does occur. For example, where the Field Medical Card is completed based on patient injuries and care given, and travels along the patient journey across organisational systems, but is also copied into electronic formats by two personas.

Once created, use cases were linked to its related tasks, and enabled goals. In parallel, data flows and trust boundaries were then mapped. To create data flows, assets were used to represent external entities as people, systems or hardware, information assets were used as data stores, and use cases represented the processes between which data flows. As some data flowed from assets of one environment to another, we can represent these interactions from one trust boundary to another, viewed in the Data Flow model. Boundaries were further represented using CAIRIS' Location model, where a location can represent sub-locations in which an instance of an asset occurs, e.g. a house has rooms. We can also link these sub-locations, e.g. if we have a hall, these can be linked to the rooms. All related assets for that location were populated along with personas carrying a task in that environment. When risks were created, these were also seen in the Location model.

There were a number of other options for modelling and visualising elements of risk in CAIRIS. The primary risk-focused option entailed modelling where threats and vulnerabilities were associated, which equate to a risk for systems and the SoS. Once assets, tasks, roles and attackers were created, threats and vulnerabilities could be added with an associated misuse case equating to a risk, viewed in the CAIRIS Risk Analysis and Task models. This indicated where some risks may occur in one environment which may affect a system in another environment, or some risks may occur across all environments, or be specific to a sub-system in one environment. This representation originally created a strange effect in CAIRIS, where a risk could be situated in one environment, but is applicable and visible to another where no misuse case is present. To remedy this, in addition to other built-in validation, CAIRIS now has the means to identify and alert to where an instance of this risk scenario occurs. The representation of Responsibility models also added value by demonstrating where a role is responsible or accountable towards an asset, task, goal, requirement, and elements of risk. This is also one example of a self-populating model as other elements are added and interlinked within CAIRIS.

Obstacle modelling were used as another tool to represent threats and vulnerabilities towards the completion of goals. Goal and Obstacle models in CAIRIS provided the option to model system-specific requirements, using a top-down or bottom-up approach, where goals and sub-goals were operationalised by tasks, and refined into requirements. However, in our scenario, we knew the required tasks and high-level system goals, but needed to identify areas in which to elicit the system sub-goals. These sub-goals were therefore selected to enable steps of a task carried out by a persona. Obstacles were then used to represent a threat or vulnerability towards an information asset identified in the Risk model potentially obstructing the completion of other goals. For example, threats of unauthorised access, use, disclosure, disruption, modification, or destruction of data or systems affecting the continuum of care. To address the goal obstacles, these were refined into

requirements to satisfy the system interaction with SoS goals. This became more difficult when there were conflicting requirements or where there was no direct relationship between some systems, meaning trade-offs needed to occur between systems and requirements. For example, the originator of the Field Medical Card may hold its *Integrity* and accuracy of patient data as important. Whereas, once used in another environment by another system, *Availability* may be desired, because without the information, treating the patient accurately is difficult. However, in both cases, once in electronic format, *Confidentiality* may be of higher importance. In all cases though, *Accountability* should be present.

IV. Discussion

We applied the OASoSIS approach to further identify the alignment of SoS factors and concepts suitable for eliciting, analysing, validating security risks using tool-support within the SoS context. The application of a reduced-scale exemplar of a Military MEDEVAC SoS case-study was purposely limited to a simplified abstraction of a SoS. However, as is often the case, with any simplicity there is always complexity, perhaps more so in a SoS scenario.

Although OA will continue to be modified to provide a simple repeatable and reusable process for identifying security risk in a SoS, early findings suggest the alignment of its output with a tool such as CAIRIS provides many benefits for translating operational needs into requirements. We found that OA was generally asking the right questions, and could be useful as a means through CAIRIS to convey operational needs to RE, but needs further refinement. For example, Step 0 already begins to capture details of stakeholders, organisations and persons of accountability and their related SoS assets. However, as this feeds into Steps 3 and 4, we need to document this earlier as part of OA. This also means that Steps 1-3 may run in parallel, thus changing the the orignal flow of OA. Moreover, Steps 4 and 5 of OA capture areas of concern for the assets, then thinks of threat scenarios to capture more potential areas of concern. However, this step could be reversed or merged to better guide stakeholder discussion. Furthermore, where it considers concerns, threats and threat scenarios, it does not explicitly document the potential weakness or vulnerability, where it perhaps should. This provides a more clear and complete risk equation, and further enables better data capture into CAIRIS towards mitigating the weakness.

Data output from OA into CAIRIS provided most of the information required to generate these models and requirements, with some additional details from initial data collection for rational. Unlike other versions of OCTAVE, the benefit of OA to operational areas is that it gives a specific focus towards the information asset and its related security properties, e.g. Confidentiality, Integrity, Availability, and Accountability. When translating this into CAIRIS, we find we can identify what security properties must hold for each information asset, but have little indication of security needs for other types of system assets. This appears to be a weakness or limitation of OA, but could be turned into a strength when considering how *Bravo* information assets should be treated by other systems in process, storage and transmission, some of which are outside of their control; requirements conflicts or needs may then be identified and addressed for the SoS.

Combining models first provided a view for *Bravo* and their SoS interactions, with additional views added for *Alpha* and *Charlie*, highlighting where dependent relations and security risk exists towards fulfilling the continuum of care, whilst supplying reasoning towards RE. When modelling multiple systems, naming convention and terms across environments did become a challenge to indicate which element related to each independent system. Understanding in what order to build SoS models is also a process efficiency consideration. However, models may also be used for various purposes across different engineering or design teams, therefore, understanding how these models inter-link plays a further role in understanding the viewpoints and varying needs of SoS engineering. Capturing different stakeholder and user views of the SoS interaction is important towards the modelling process, the output of which would aim to assist subsequent risk-based decision making processes of risk management, providing a means to assist reasoning towards security and risk in RE during the SoS development life-cycle.

V. Conclusion

In this paper, we present OASoSIS: an approach that aligns SoS factors and concepts suitable for eliciting, analysing, validating security risks using tool-support within the SoS context. Although OA aims to provide a simple repeatable and reusable process for identifying security risk in a SoS, early findings suggest the alignment with a tool such as CAIRIS provides many benefits for translating operational needs into requirements. However, due to the nature of SoSs independent collaborations, there will always be an element of unknown or unavailable risk-based information in which to base risk assessment on. Interoperability across dependent systems will be difficult to achieve in a SoS without understanding the bigger picture. Therefore, understanding what the minimum level of information is required to make a satisfactory security risk assessment is of importance, certainly when translating these to requirements. This process will be further refined using MMN, before moving to test with a healthcare, smart city or stabilisation based SoS, to identify and assess areas of security risk for further validation of the OASoSIS approach. Future work will continue to identify gaps and opportunities for risk assessment of security in SoSs, and how combining elements with the use of tool-support can assist with risk-based visualisation supporting decision making for the SoS and Security Requirements Engineering communities.

Acknowledgement

The research described in this paper was funded by Bournemouth University studentship DSTLX1000104780R_BOURNEMOUTH_PhD_RASOS.
We are also grateful to Dstl for their sponsorship of this work.

REFERENCES

[1] Office of the Deputy Under Secretary of Defense, for Acquisition and Technology, Systems and Software Engineering, *Systems and Software Engineering. Systems Engineering Guide for Systems of Systems*, 1st ed., Washington, DC: ODUSD(A&T)SSE, 2008, 2008.

[2] D. Ki-Aries, S. Faily, H. Dogan, and C. Williams, "Re-framing The AMN: A Case Study Eliciting and Modelling a System of Systems using the Afghan Mission Network," in *11th IEEE International Conference on Research Challenges in Information Science 10-12 May 2017 Brighton, UK*. IEEE, May 2017.

[3] D. Ki-Aries, H. Dogan, S. Faily, P. Whittington, and C. Williams, "From Requirements to Operation: Components for Risk Assessment in a Pervasive System of Systems," in *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)-Proceedings of 4th International Workshop on Evolving Security & Privacy Requirements Engineering*. IEEE, 2017, pp. 83–89.

[4] D. Ki-Aries, S. Faily, H. Dogan, and C. Williams, "System of systems characterisation assisting security risk assessment," in *IEEE 13th System of Systems Engineering Conference*. IEEE, Jun. 2018.

[5] V. Chiprianov, L. Gallon, M. Munier, P. Aniorte, and V. Lalanne, "Challenges in Security Engineering of Systems-of-Systems," in *Troisième Conférence en IngénieriE du Logiciel*, 2014, p. 143.

[6] C. Ncube, S. L. Lim, and H. Dogan, "Identifying top challenges for international research on requirements engineering for systems of systems engineering," in *Requirements Engineering Conference (RE), 2013 21st IEEE International*. IEEE, 2013, pp. 342–344.

[7] S. AlhajHassan, M. Odeh, and S. Green, "Aligning systems of systems engineering with goal-oriented approaches using the i* framework," in *Systems Engineering (ISSE), 2016 IEEE International Symposium on*. IEEE, 2016, pp. 1–7.

[8] International Council of Systems Engineering, *Systems Engineering Handbook*, version 3.1 ed., INCOSE, Aug. 2007.

[9] R. Ross, M. McEvilley, and J. C. Oren, "Systems Security Engineering," *NIST Special Publication*, vol. 800, p. 33, 2016.

[10] A. Shostack, *Threat modeling: Designing for security*. John Wiley & Sons, 2014.

[11] R. A. Caralli, J. F. Stevens, L. R. Young, and W. R. Wilson, "Introducing OCTAVE Allegro: Improving the information security risk assessment process," DTIC Document, Tech. Rep., 2007.

[12] S. Faily, "CAIRIS web site," https://cairis.org, June 2018.

[13] J. Boardman and B. Sauser, "System of Systems-the meaning of of," in *2006 IEEE/SMC International Conference on System of Systems Engineering*. IEEE, 2006, p. 6.

[14] F. Alkhabbas, R. Spalazzese, and P. Davidsson, "IoT-based Systems of Systems," in *Proceedings of the 2nd edition of Swedish Workshop on the Engineering of Systems of Systems (SWESOS 2016)*. Gothenburg University, 2016.

[15] P. Whittington and H. Dogan, "SmartPowerchair: Characterization and Usability of a Pervasive System of Systems," *IEEE Transactions on Human-Machine Systems*, 2016.

[16] M. W. Maier, "Architecting principles for systems-of-systems," in *INCOSE International Symposium*, vol. 6, no. 1. Wiley Online Library, 1996, pp. 565–573.

[17] J. S. Dahmann and K. J. Baldwin, "Understanding the current state of US defense systems of systems and the implications for systems engineering," in *Systems Conference, 2008 2nd Annual IEEE*. IEEE, 2008, pp. 1–7.

[18] C. Ncube and S. L. Lim, "On Systems of Systems Engineering: a Requirements Engineering Perspective and Research Agenda," in *Requirements Engineering Conference (RE), 2018 IEEE 26th International*. IEEE, 2018.

[19] British Standards Institution, "BS ISO/IEC 27005, Information technology - Security techniques - Information security risk management." 2011.

[20] NIST, "NIST Special Publications [online]," NIST Computer Security Resource Centre, 2017, Available From: http://csrc.nist.gov/publications/PubsSPs.html [Accessed 22 April 2017].

[21] J. Dahmann, G. Rebovich, M. McEvilley, and G. Turner, "Security Engineering in a System of Systems environment," in *Systems Conference (SysCon), 2013 IEEE International*. IEEE, 2013, pp. 364–369.

[22] S. Boyson, "Cyber supply chain risk management: Revolutionizing the strategic control of critical IT systems," *Technovation*, vol. 34, no. 7, pp. 342–353, 2014.

[23] Institute of Electrical and Electronics Engineers (IEEE), *Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries*, IEEE, New York, NY, 1990.

[24] B. Zhou, O. Drew, A. Arabo, D. Llewellyn-Jones, K. Kifayat, M. Merabti, Q. Shi, R. Craddock, A. Waller, and G. Jones, "System-of-systems boundary check in a public event scenario," in *System of Systems Engineering (SoSE), 2010 5th International Conference on*. IEEE, 2010, pp. 1–8.

[25] D. G. Firesmith, "Analyzing and specifying reusable security requirements," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep., 2003.

[26] N. R. Mead and T. Stehney, *Security quality requirements engineering (SQUARE) methodology*. ACM, 2005, vol. 30, no. 4.

[27] G. Elahi, E. Yu, and N. Zannone, "A vulnerability-centric requirements engineering framework: analyzing security attacks, countermeasures, and requirements based on vulnerabilities," *Requirements engineering*, vol. 15, no. 1, pp. 41–62, 2010.

[28] D. Trivellato, N. Zannone, M. Glaundrup, J. Skowronek, and S. Etalle, "A semantic security framework for systems of systems," *International journal of cooperative information systems*, vol. 22, no. 01, p. 1350004, 2013.

[29] P. H. Meland and J. Jensen, "Secure software design in practice," in *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*. IEEE, 2008, pp. 1164–1171.

[30] H. Mouratidis, "Secure software systems engineering: the Secure Tropos approach," *JSW*, vol. 6, no. 3, pp. 331–339, 2011.

[31] F. Den Braber, G. Brændeland, H. E. Dahl, I. Engan, I. Hogganvik, M. Lund, B. Solhaug, K. Stølen, and F. Vraalsen, "The coras model-based method for security risk analysis," *SINTEF, Oslo*, vol. 12, pp. 15–32, 2006.

[32] S. Ardi, D. Byers, P. H. Meland, I. A. Tondel, and N. Shahmehri, "How can the developer benefit from security modeling?" in *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*. IEEE, 2007, pp. 1017–1025.

[33] I. Sommerville, *Software Engineering*, 10th ed. Pearson, 2015.

[34] S. Faily and C. Iacob, "Design as code: Facilitating collaboration between usability and security engineers using cairis," in *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, 2017.

[35] National Research Council and others, *Human-system integration in the system development process: A new look*. National Academies Press, 2007.

[36] A. Bruseberg, "Human views for MODAF as a bridge between human factors integration and systems engineering," *Journal of Cognitive Engineering and Decision Making*, vol. 2, no. 3, pp. 220–248, 2008.

[37] Col. Dr. I. Hartenstein, "Medical Evacuation Policies in NATO: Allied Joint Doctrine for Medical Evacuation [online]," Tech. Rep., 2008, available From: https://stopthemedevacmadness.files.wordpress.com/2012/02/nato-medical-evacuation-policies-in-nato-mp-hfm-157-01.pdf [Accessed 19 January 2018].

[38] S. Faily and I. Fléchais, "Barry is not the weakest link: Eliciting Secure System Requirements with Personas," in *Proceedings of the 24th BCS Interaction Specialist Group Conference*. British Computer Society, 2010, pp. 124–132.

[39] S. Faily and I. Fléchais, "The secret lives of assumptions: Developing and refining assumption personas for secure system design," in *Proceedings of the 3rd Conference on Human-Centered Software Engineering*, vol. LNCS 6409. Springer, 2010, pp. 111–118.

[40] S. Faily, "Personahelper," Chrome Web Store, 2018, Available From: https://chrome.google.com/webstore/detail/personahelper/mhojpjjecjmdbbooonpglohcedhnjkho [Accessed 2 May 2018].

[41] Trello, "Trello," Trello, 2018, Available From: https://trello.com/ [Accessed 2 May 2018].