Computational Sciences and Engineering 2(1) (2022) 53-68

University of Guilan

Computational Sciences and Engineering

journal homepage: https://cse.guilan.ac.ir/

Toward Cyber Command and Control System Architecture Using Data-Driven Analysis Solutions

Ali Pourghaffari [a,*], Rahim Asghari [b], Ali Jabbar Rashidi [a]

[a] Malek Ashtar University of Technology, Tehran, Iran.
[b] Department of Electrical and Computer engineering, Malek- Ashtar University, Tehran, Iran.

ARTICLEINFO

Article history:
Received 19 June 2021
Received in revised form 4 December 2021
Accepted 20 January 2022
Available online 1 April 2022

Keywords:
Command and control
Information system architecture
Data analysis
Information security

ABSTRACT

Today, countries' sovereignty and national security strongly rely on the reliable operation and continuous monitoring of information technology infrastructure against security threats. As a result, the importance of comprehensive command and control and consistent oversight of IT security has become increasingly apparent in recent years. Modern command and control systems are dynamically and continuously monitoring and analyzing their mission space. This scope of operations increases the need to create a coherent and integrated structure in developing a system based on a well-defined architecture. Best of over knowledge has been little discussion on how to design command and control systems better. In this paper, we proposed architecture using data analysis solutions in cyber command and control missions. The proposed architecture is based on service-oriented and layered architecture to activate the quality features of interoperability, distributability, heterogeneous development, and scalability. Also, a prototype has been implemented to demonstrate its applicability through solution architecture. The online survey questionnaire validates the proposed architecture and its implementation.

1. Introduction

Due to the vulnerability of countries' sovereignty and national security from reliable performance and continuous monitoring of information technology infrastructure, comprehensive C2 and information security monitoring have become an integral part of IT Management and operational technologies [1]. According to the 2018 GCI report, Iran is among the countries that have developed a complex and specific program for their cybersecurity. The advent of new technologies has gradually led to further changes in command, wars, organization of forces, and how to command

[*] Corresponding author.
E-mail address: alipourghaffari@chmail.ir (A. Pourghaffari)

http://dx.doi.org/10.22124/CSE.2022.19924.1017
© 2022 Published by University of Guilan

and control and monitor security. Troops and missions will be increasingly integrated with devices and information technology in the future, increasing the complexity of security oversight operations in command and control of subordinate forces [2]. In the past, C2 systems were based more on providing an overview of mission space using large screens and human-centered analytics, with no attempt to connect information. Today, IT-based C2 focuses on data and analyzes it automatically for stakeholders, and establishes meaningful relationships between collected data [3]. Modern C2 systems are dynamically and continuously monitoring and analyzing their mission space. This scope of operations increases the need to create a coherent and integrated structure in developing a system based on a well-defined architecture. This architecture should be designed based on loosely coupled and iterative incremental development principles in ultra-large systems because a wide range of technologies, tools, programming languages, and infrastructures will contribute to the mission of C2 systems. As a result, architecture must be developed independently of tools and components to integrate all components despite the heterogeneous development of different aspects of time, space, and technology. However, research in this area has frequently focused on describing effective methods for defining the technical and social relationships necessary for the success of these systems, while less attention has been devoted to how to design better C2 systems and describe critical factors for their success [4]. Accordingly, developing a data-driven C2 architecture model was considered because of strategic, structural, and even technical differences. This architecture should cover various C2 missions and should not be dependent on technology and tools.

The questions of this research are:

1. What are the components and layers of data-driven C2 systems architecture to monitor mission space?
2. How can appropriate interaction be established to collaborate on partner components and technologies in C2 architecture?
3. What are the characteristics of C2 architecture for gradual and incremental development based on temporal and spatial distribution?

Despite the vital importance of this field and the significant potential of its research, the literature in this field is very scattered, and only a limited amount of study provides an accurate idea of such systems [4-6]. Most research has focused on novel analysis methods rather than developing common frameworks or improving the overall C2 process. It has not provided an integrated and comprehensive approach that covers the complete system [4].

The rest of the paper is organized as follows: The subject is explained in the research literature in the second section. Next, in the third section, the research background is reviewed and described. In the fourth section, the proposed architecture, its layers, and its aspects are described. In the fifth section, the implementation of the experimental version of the proposed architecture is presented. In the sixth section, the architectural evaluation is given, and finally, the conclusion is shown in the seventh section, and then the references.

2. Research literature

2.1. C2 systems

C2 systems are based on a conventional process complemented by various teams and organizations and include a set of human socio-technical interactions, technical and technological operations, and ultimately activate the C2 process [7-8]. The historical background of C2 refers to the administrative and hierarchical structures that existed in military organizations. In these organizations, workforce positions were defined, and the mission of each position was non-negotiable [5,9]. Over time, C2 refers to any structure formed to work together and achieve common goals, leading to improved decision making and information sharing [10]. Accordingly, C2 has a broader application and meaning than in the past. In addition to the military hierarchy, it additionally includes more general applications such as urban infrastructure management [11]. It extends to partner NGOs in emergencies and virtual organizations without unified leadership or oversight rules [6,12]. These systems combine and integrate various products and convenient solutions to provide interoperability between multiple services and functions in specific areas [13]. Figure 1 shows the general structure of C2 systems, which are described in Section 3.

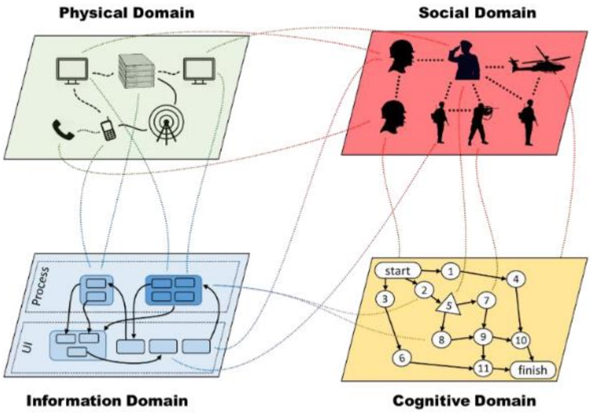


Figure 1. Multilayer network superstructure of NCW1 domains and C2 literature [4].

NCW domains are comprised of distinct subsystems that interact to form a four-layer network superstructure [4].

2.2. Software architecture

The software architecture is a set of structures needed to reason about that system, including the software elements, the relationships between them, and properties [14]. The importance of software architecture and architectural viewpoint in software development is more significant in large software projects such as C2 systems that are very complex. Software architecture considers the system's quality attributes and defines the decisions made regarding how to manage and compromise between the quality characteristics [14,15]. It also improves the capabilities of gradual development, reuse, and integration [15]. Architectural frameworks such as Open Group and DoDAF[2] architecture represent reasoning methods and create coherent governance in software systems [16].

[1] Network Centric Warfare
[2] Department of Defense Architecture Framework

2.2.1 Ultra-Largescale Systems

The DoD and the SEI first used the term ULS[3]. Achieving the goals and missions of large systems, such as C2 systems, is always immensely complex. This complexity is directly related to the number of decision nodes, the variety of infrastructure and technology, and the variety of methods of connecting nodes. The size of these systems is usually much larger than the size of the regular commercial and even SoS[4] systems. The number of program code lines, system users, connections, accesses, hardware components, and data stored is considerably larger than a typically software system, and the rate of change over time is significantly high [13]. ULS System Development is instead of developing and debugging a massive set of code, enhancing a process of assembling and integrating components, known as enterprise application integration [17].

2.3. Service-oriented architecture and Enterprise Service Bus

Service-oriented architecture is an integrated approach that uses integration solutions instead of communication through software communication interfaces [18]. This approach focuses on services and communication in a standard way and aims to reduce complexity and prevent overwork in integration. ESB[5] is an example of a Middleware [17] and facilitates communication between applications by enclosing services. So, communicating programs do not need to know the location of different services or protocols. In addition, it improves the ability to adapt and convert messages, message mediation and provides security and scalability in the system [20].

2.4. Network Centric Warfare

NCW is one of the most advanced approaches in C2 research and has been developed to exploit the technologies developed in the age of information technology to achieve agility and other benefits [5,21,22]. Agility means the ability to influence, cope with, or exploit conditions and changes [23,24]. In particular, more mature C2 includes recognizing situational change and adopting the C2 approach required to meet that change. Agility is characterized by three dimensions of C2 approach space: Allocation of Decision Rights, Patterns of Interaction, and Distribution of Information [6]. As part of the C2 maturity model, five representative C2 approaches were associated with five specific regions of the C2 approach space. These five C2 approaches are Conflicted C2, De-Conflicted C2, Coordinated C2, Collaborative C2, and Edge C2 [6]. Conflicted C2 has minor agility due to decision-making structure, interaction patterns, and undistributed information. Edge C2 has the most agility due to unstructured decision-making rights, interaction patterns, and distributed information [4].

3. Research Background

According to the NATO[6] subdivision, the C2 systems include at least four basic subsystems [4]: Physical systems, including physical equipment and technology such as sensors and infrastructure networks, Information systems, including activities, methods of creating, manipulating, analyzing, storing, and retrieving information, social systems that deal with human organization and communication and Cognitive systems which includes mental models, perceptions, orientations, and values. The design of C2 systems is also shaped by four constraints, which have component constraints for expressing physical requirements, constraint constraints for system implementation

[3] Software Engineering Institute
[4] Systems of Systems
[5] Enterprise Service Bus
[6] North Atlantic Treaty Organization

interaction, system-level constraints, and emergency constraints representing the physical constraints of real systems [4].

Table 1 shows a comparison of previous works. The first column shows the sources, the second column, and the third, fourth, and fifth columns each represent one of the domains of the C2 subsystems.

Table 1. comparison of past works

COGNITIVE	SOCIAL	INFORMATION	PHYSICAL	REFERENCES
			✓	[25]
		✓		[26]
	✓			[28] , [27]
✓				[33] , [32] ,[31] ,[30] ,[29]
		✓	✓	[38] , [37] ,[36] ,[35] ,[34]
✓			✓	[42] , [41] ,[40] ,[39]
	✓	✓		[46] , [45] ,[44] ,[43]
✓		✓		[48] , [47]
✓	✓			[52] , [51] ,[50] ,[49] ,[7]
	✓	✓	✓	[55] , [54] ,[53]
✓		✓	✓	[59] , [58] ,[57] ,[56]
✓	✓		✓	[61] , [60]
✓	✓	✓		[64] , [63] ,[62]
✓	✓	✓	✓	[68] , [67] ,[66] ,[65]

The one domain-focused works typically involve component-level development, focusing on technology development, network systems design, service-oriented architectures, and human factors. In [26] system architecture, access and correlation of services in knowledge management in service-oriented architecture are discussed. In [25], the throughput and the amount of over-the-network transferable data are used as aspects of physical systems, and [27] examines IT policies in C2 systems. Upstream policies to respond to cyber-attacks have been reviewed in [28] and [32] studies the structure of agile teams in C2 systems. In [30], the effect of technology on the command structure and gaining the advantage in network-based battle has been studied, and in [61] and [60], criteria for improving agility and knowledge distribution among teams have been defined. Cognitive analysis for mapping information flow with workflow and retrieving them while performing tasks is presented in [29]. This method has been employed in command and control networks in various fields, including military missions [33] and hospitals [69]. In [31], an overview of how agile actions are developed and the relationship between C2 activities are provided. Most of the work focused on two domains are in NCW and are divided into two categories of technical-oriented and human-centered studies. An overview of the components, contracts, and system-level constraints of cyber-physical networks is presented in [35]. Other works study the specific types of network heterogeneity by developing cyber-physical technologies [36-38] and [34]. In [7], the taxonomy of the process of social cognition is presented. The limitations of social and cognitive networks integrated with tasks for one [45] and several teams [49] and [52] have been studied. In [50] and [58], the effect of choosing physical communication infrastructure on social hierarchy has been investigated. In [41], Decision-making options and tasks in supply chain construction and management [40], and power delivery [39] and [42] based on access to information technology are examined. Human and information communication factors that link team structure and values to software and digital services are also discussed in [43, 44, 47and 48]. The three-domain category support multi-layer network modeling and analysis. In [62] and [63], modeling of socio-technical relationships in teamwork in various areas has been investigated. In [44, 46, 52, 53, 54, 55, 65, and 66], collaboration, information sharing, and trust for developing dynamic networks relating infrastructure with data flow and accuracy are presented. In [53], [54], and [55], taxonomy is proposed to compare the dynamics of communication systems and human teams. [56] Offers decision models in service-oriented architecture and a modeling framework for cyber-attacks. [57]

Provides a mathematical framework for optimizing the structure of command-and-control cooperation and a basis for measuring the impact of misinformation. Finally, of the domain-wide tasks, an approach to event analysis for a systematic workgroup framework has been developed in [68]. In [64], social network theory combines events analysis for a systematic workgroup framework and collaboration, information sharing. In [65], simple network metrics are used for C2 communication systems in NCW. In [67], a framework for four domains for measuring the awareness of command units in socio-physical social networks consisting of processes, people, programs, systems, and physical network layers is provided.

4. Proposed Architecture for cyber C2

This section presents the CyC2[7] architecture. The CyC2 is a cyber C2 architecture using solutions based on data analysis in NCW. This layered architecture, aims to provide a coherent way to use mission space data to detect anomalies, identify and manage security risks, and based on the principles of software architecture design [14], [15] and [16] the development of large-scale systems [13], service-oriented architecture [18] and [19], DoDAF architecture [70]. The idea is taken from the FinSec security reference architecture [71]. According to Figure 2, the CyC2 consists of five layers, and each layer encloses a functional requirement.

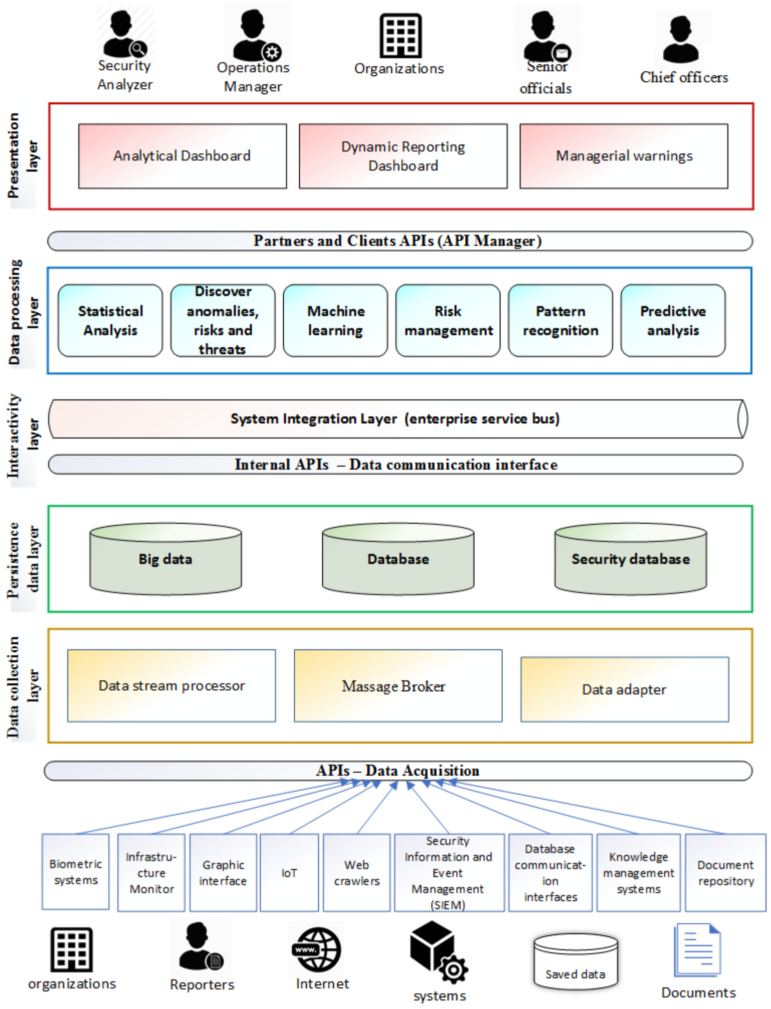


Figure 2. Proposed cyber C2 architecture.

[7] Cyber Command and Control

Designed with the principles of service-oriented architecture, each layer dominates its complexities. Relevant data sources are shown at the lowest level of the architecture and some stakeholders and users at the highest level.

4.1. Data collection layer

This layer contains a set of components and solutions necessary for data acquisition, and its purpose is to properly connect to the various data sources participating in the cyber C2 space. Data are collected through event management and security information systems, web crawlers and network monitoring systems, etc., and includes all data that can be used to extract insights to maintain security and reduce risks.

This layer consists of three core components:

1- Stream processor for real-time processing data streams with minimal latency.
2- Message broker, for conversion and initial adaptations of data received from APIs, which are one of the core elements of message-centric middlewares such as command and control systems.
3- Adapter to Implement advanced adaptations and filtering requirements that message intermediaries and stream processors do not provide.

4.2. Persistence data layer

This layer stores all data types in historical data and includes a set of databases and storage technologies. The security database contains all security datasets, including logs and security reports sent from data sources. The knowledge base continuously stores a set of knowledge and information, including the results of analyses, knowledge, and experiences. Big data also include a vast collection of collected data, including social media data, government systems, and even news, continuously collected and stored in various structured and semi-structured data or unstructured data in databases.

4.3. Interface layers and interactivity

The enterprise service bus is the core component of the interactivity layer. This component improves interoperability and enables components to communicate with various data formats, messages, and standards. Integration using service-oriented architecture enables the heterogeneous development of various parts of the architecture, integrates different solutions and layers at other times and places, and hides its integration operations and complexities from developers and implementation teams. In addition, the intermediary components and API management hide the complexity of interlayer communication and facilitate the use of standard communication interfaces and contracts. API Management Solutions Provide complete lifecycle, utilization, and exploitation of security management and policies on APIs and enable the deployment and management of API-based ecosystems that provide a minor dependency and connectivity.

4.4. Data processing layer

The data processing layer consists of components, tools, technology solutions, and even Ad hoc systems processing and analyzing data based on the requirements of command and control systems. Primary processing components include statistical analysis, anomaly and hazard detection, machine learning for data classification and clustering, risk management, pattern discovery and predictive analysis, and foresight generator components. Layered architecture and interaction management have hidden the complexities of interactivity, data collection, and access through communication interfaces from the data processing layer. Also, data analysis and processing complexities are hidden from the upper layer, responsible for providing results and outputs to the main stakeholders. All processes and analytics zones are encapsulated in the data processing layer. As a result, the development and implementation of the components and functions of the data processing layer are feasible independently of each other and with the least dependence on other tasks.

4.5. Presentation layer

This layer communicates with key stakeholders, including senior management, executives, relevant organizations, operations managers, and other security analysts and decision-makers. It provides graphical user interfaces, dynamic reporting dashboards, and analytical and configurable tools as needed. The analytics dashboard is the interface for creating and producing detailed and specialized reports, developing divergent reports for managers and decision-makers to develop a long-term security strategy. The Management Alerts interface provides a set of notifications to security managers and system administrators to respond promptly to security risks and issues.

4.6. Processing levels and aspects in the proposed architecture

To describe the aspects of processing and analysis and the data sources in the proposed architecture, the architectural cube is presented in Figure 3 as a support artifact of the proposed architecture.

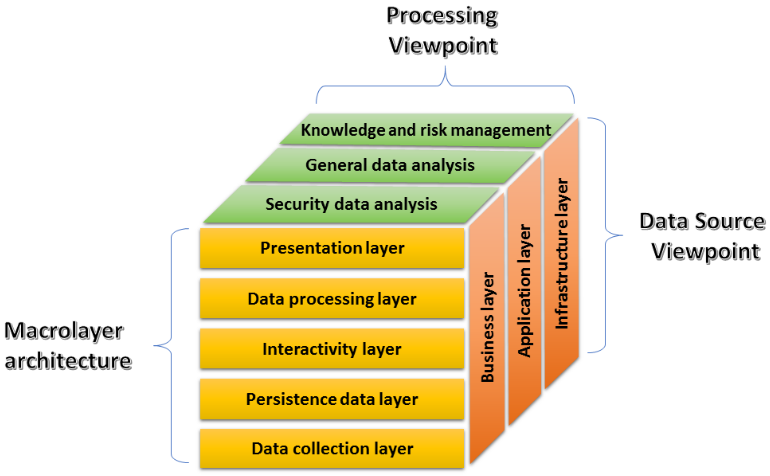


Figure 3. Proposed C2 architecture support cube.

One cube dimension consists of the proposed architecture, one dimension of analytical data sources (according to enterprise architecture layers), and the other dimension consists of various aspects of processing and analysis.

Data sources can be categorized and divided into three levels based on the layers of the enterprise. The business layer includes a set of tools such as BPMS[8] and office automation to serve the organization's business. The data obtained from these tools are used to discover security patterns (for example, using process events log in situational awareness through process mining). The application layer includes a variety of applications developed to implement the mission and functions of the organization. Data from this layer, such as operating system logs or service call logs, can also be used in security analysis. The network and infrastructure layer includes all hardware and network technologies and infrastructure. Data from devices in this layer, like router

[8] Business process management systems

logs or firewall data, are essential in NCW analysis. To generate a comprehensive and proper view through the proposed architecture, it is significant to consider data processing at all levels. The collected data must be processed and analyzed from various aspects to develop a security strategy and C2. Security insight extraction from security data of varying levels like firewalls, access control, user information, and activities collected is an aspect of data processing. In addition, in general, aspects that may affect security, data processing is critically important in developing the strategy and establishing a coherent vision of the C2 ecosystem; For example, data collected from the web or social networks. Another aspect of data analysis in control command systems is the analysis of knowledge and experiences utilized to decision support based on experience.

5. The Implementation of the Proposed Solution Architecture

The solution architecture has been used to implement the experimental prototype of the proposed architecture. Solution architecture focuses on translating and transforming requirements into IT solutions, tools, and systems [72]. Instead of developing from scratch in solution architecture, the architecture is implemented by integrating tools and solutions. The solution architecture is based on component-based development in the principles of ULS systems development, reduces development time and cost, and facilitates the integration process due to the use of standard tools. According to Table 2, for each layer of the proposed architecture, a solution has been selected for implementation. In our selection, the open-source tools are prioritized.

Table 2. Mapping implementation solutions to the proposed architectural layers.

Architectural layers	Implementation solution	Key features
1. Data collection layer	Apache Kafka	Open source, real-time stream processing, distributed, publisher, and applicant template support.
2. Persistence data layer	MySQL	Open source, robust user interface, advanced query support, replication, and distribution support.
3. Interoperability layer	ODBC/JDBC	Connectors and converters connect to data sources and store and retrieve data.
4. Data processing layer	Pentaho data integration	Open source, data analytics, and processing, business intelligence capabilities.
5. Presentation layer	Microsoft Power Pivot	Microsoft office, data cube, and multiple dimensions, interactive and graphic dashboard.

Apache Kafka is used to implementing the data collection layer, a real-time event processing and support engine using the publisher and applicant template, and has high distribution capability and scalability improvements. In the Persistence data layer, the MySQL database is used to store and retrieve data. Due to the limitations of the solutions used in the experimental implementation, only connectors and database connection interfaces have been used in the interoperability layer. By increasing the number and complexity of communications, WSO2 Enterprise Service Bus and API

Management products can be used. The data processing and ETL process is implemented through Pentaho Data Integration (PDI), an open-source tool for data analysis, processing, and business intelligence to enrich, integrate, aggregate, and clean data. The presentation layer is implemented through Microsoft Power Pivot to provide an analytics dashboard that can develop a semantic connection between data and constitute multiple dimensions (data cubes) between them and provide data analysis dashboards. Due to the various applications of the system, two different data sets have been used in the implementation. A network security dataset and an urban transportation management dataset were used. The security data set (Table 3) logs various network attacks such as DDOS and Break-DNS attacks with information like port number, IP number, date, and time of the attack. The urban management data set (Table 4) is information on the use of public transport ticket cards and includes items such as route, type of card, age of public transport drivers, and the like.

Table 3. An example of a network security data log.

	C2S ID	Source IP	Source Port(s)	Destination IP	Destination Port(s)	Start Date	Stop Time
C2 + TCP control channel exfil - no precursor NC	43557	138.106.196.178	0	172.28.219.190	10000	11/3/2009	11/3/2009
C2 + control channel exfil - no precursor NC	43560	198.123.37.66	0	172.28.14.52	10000	11/3/2009	11/3/2009
failed attack or scan exploit/bin/iis_nsiislog.pl	43561	161.154.58.214	0	255.255.255.255	21	11/3/2009	11/3/2009
scan /usr/bin/nmap	43562	151.243.222.89	54527	172.28.52.6	22321	11/3/2009	11/3/2009
DDoS	45500	1.23.177.16	0	172.28.4.7	80	11/3/2009	11/3/2009
compromised_server	45496	172.28.119.228		255.255.255.255	80	11/3/2009	11/3/2009
compromised_server	45497	172.28.126.109		255.255.255.255	80	11/3/2009	11/3/2009
DDoS	45498	155.108.237.71	0	172.28.4.7	80	11/3/2009	11/3/2009

Table 4. An example of a public transport data log.

Card serial number	Product code	Agent ID number	Route ID	SV decrease amount	Validation time (hh:mm:ss)	Service start Date	Equipment code
1.97E+09	106	9998	2501	0	00:20:00	1/10/2011 00:00	1027
3.33E+08	132	9998	2501	1000	00:20:00	1/10/2011 00:00	1006
2.29E+09	132	54664	2602	1000	00:20:00	1/10/2011 00:00	6121741
4.31E+08	132	50082	640	500	00:20:00	1/10/2011 00:00	3321297
1.58E+09	102	38196	2805	0	00:20:00	1/10/2011 00:00	3921518
1.48E+09	102	38196	2805	0	00:20:00	1/10/2011 00:00	3921518
3E+09	106	51480	2602	0	00:20:00	1/10/2011 00:00	3621132
3.72E+09	132	51480	2602	1000	00:20:00	1/10/2011 00:00	3621132

6. Architectural Evaluation

To evaluate the proposed architecture, ATAM[9] methods and expert evaluation have been used. The architectural compromise analysis method represents a complete and comprehensive method for assessing architecture and how architectural goals and quality features are met through architectural decisions [14]. Beforehand, the architectural impulses are extracted according to Table 5.

Table 5. Architectural impulses of the proposed architectural design.

ARCHITECTURAL IMPULSES		
A) Important functions of the system	▪ C2 in crucial security areas such as NCW and intrusion detection	▪ C2 of public areas such as urban management
B) Any kind of technical, managerial compulsion	▪ Heterogeneous development in terms of time and place ▪ Distributability	▪ Scalability
C) Objectives and contexts related to the project	▪ Data-driven analysis to support command and control	
D) The main stakeholders	▪ Security decision-makers and institutions ▪ Security analysts	▪ Decision-makers and government institutions ▪ High-level management analysts
E) Architectural Motivation includes the significant goals of the qualitative features that influenced the architecture	▪ Allocation of decision-making rights ▪ Information distribution	▪ Interaction patterns

Then, according to the three criteria of agility in C2 introduced earlier and have been the most significant drivers of architectural design and three qualitative characteristics of heterogeneous development in terms of time and place, scalability and distributability are the primary technical and managerial obligations of architectural design. It is suggested that four properties have been selected to form a utility tree in the evaluation, as shown in Figure 4.

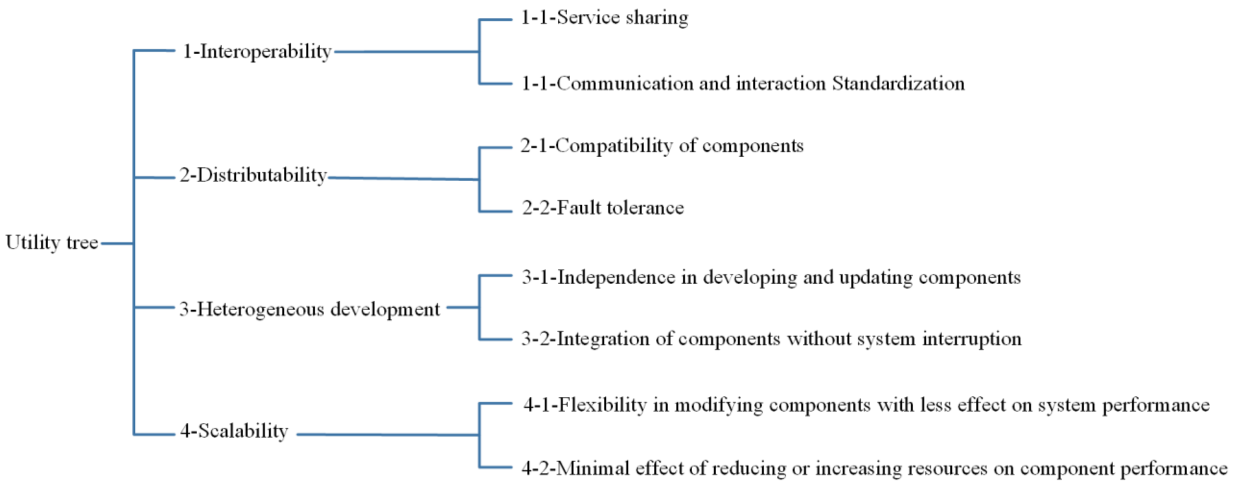

Utility tree
1-Interoperability
1-1-Service sharing
1-1-Communication and interaction Standardization
2-Distributability
2-1-Compatibility of components
2-2-Fault tolerance
3-Heterogeneous development
3-1-Independence in developing and updating components
3-2-Integration of components without system interruption
4-Scalability
4-1-Flexibility in modifying components with less effect on system performance
4-2-Minimal effect of reducing or increasing resources on component performance

Figure 4. Utility tree evaluating the proposed architecture.

These four features are interoperability (support for interaction patterns), distributability, heterogeneous development, and scalability. Using the service sharing feature, each architectural component can use the services provided by the other components, without having to know the source of them. Communication and interaction Standardization eases the interoperability of

[9] Architecture tradeoff analysis method

components. Compatibility of components enables activities and tasks to distribute and perform simultaneously on different machines. Fault tolerance prevents the entire system from becoming inaccessible if a component, hardware, or network fails. The independence in developing and updating components minimizes the need for coordination and exchange of information between component development teams. Adding and integration of components without system interruption is an important requirement. Table 6 shows the proposed architectural decisions for each of the quality features and quality criteria for evaluating the quality features of the architecture.

Table 6. Qualitative features and architectural decisions.

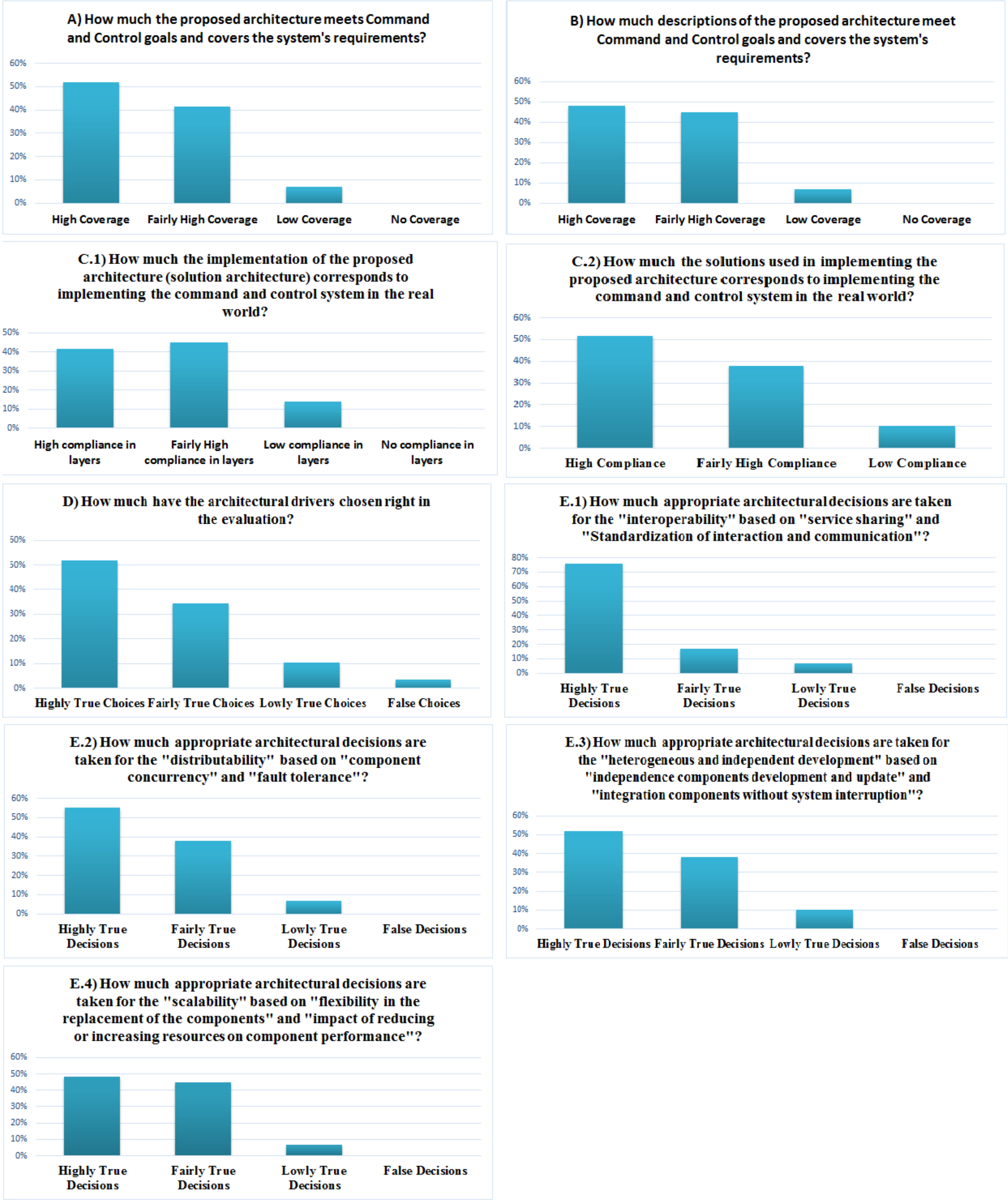
Qualitative Features	Evaluation criteria	Proposed architectural decisions
1. Interoperability	1-1. Sharing services	Service-Oriented Architecture
		Enterprise service bus
	1-2. Communication and interaction Standardization	API management
		System connection interface (REST / SOAP API)
2. Distributability	2-1. Compatibility of components	Component dependency Reduction
		Enclosing each functionality in one component
		Layered architecture
	2-2. Fault tolerance	Component dependency Reduction
		Encapsulate each requirement in one component
		Layered architecture
3.Heterogeneous development capability	3-1. Independence in developing and updating components	Service-Oriented Architecture
		Standard interaction interfaces
	3-2. Integration of components without system interruption	Enterprise service bus
		Component dependency Reduction
4. Scalability	4-1. Flexibility in changing components	Separation of function groups into independent layers
	4-2. Minimal effect of reducing or increasing resources on component performance	Enterprise service bus
		Independence of components in deployment on machines

We exposed the proposed architecture to experts using an online questionnaire. The assessment was performed by more than 100 academic and technical experts active in computer science, computer security, and C2. The response rate is 54%. Finally, the evaluation results with 29 complete answers are shown in Figure 5.

Of the participants in the evaluation, 62% had a bachelor's and master's degree, and about 38% had a doctorate or higher. About 48% were utterly familiar with command and control systems, 27% were moderately familiar, about 45% were fully familiar, and 35% were moderately familiar with software architecture. About 35% were quite familiar with security topics, and 42% were moderately familiar. Based on the evaluation results (Figure 1), the proposed architecture and its descriptions have thorough and appropriate coverage. The implementation of the architecture is highly consistent with the requirements and the actual space of C2 systems. Architectural interoperability in more than 70% of cases has been assessed as entirely appropriate, distribution and heterogeneous development capability in more than 50% of cases have been evaluated as altogether appropriate, and scalability in about 50% of cases has been assessed as perfectly appropriate based on decisions made in architecture.

7. Conclusion

Modern command and control systems, using new technologies and data-based solutions, offer a significant ability to extract forward-looking insights for the decision-making and guidance of subordinate forces. The vastness and scope of operations of these systems cause their development based on a coherent and integrated architecture important. In this paper, while describing the command and control literature and reviewing and categorizing the research background, a proposed architecture for using data-based solutions in command and control missions was presented. The proposed architecture at various layers supports the common quality features of C2 systems using software architecture and service-oriented architecture principles. The evaluation based on the method of interview analysis and expert evaluation confirms the quality of the presented architecture. In future work, detailed technology layer and architecture development based on cloud computing solutions will be researched and developed.



A) How much the proposed architecture meets Command and Control goals and covers the system's requirements?

B) How much descriptions of the proposed architecture meet Command and Control goals and covers the system's requirements?

C.1) How much the implementation of the proposed architecture (solution architecture) corresponds to implementing the command and control system in the real world?

C.2) How much the solutions used in implementing the proposed architecture corresponds to implementing the command and control system in the real world?

D) How much have the architectural drivers chosen right in the evaluation?

E.1) How much appropriate architectural decisions are taken for the "interoperability" based on "service sharing" and "Standardization of interaction and communication"?

E.2) How much appropriate architectural decisions are taken for the "distributability" based on "component concurrency" and "fault tolerance"?

E.3) How much appropriate architectural decisions are taken for the "heterogeneous and independent development" based on "independence components development and update" and "integration components without system interruption"?

E.4) How much appropriate architectural decisions are taken for the "scalability" based on "flexibility in the replacement of the components" and "impact of reducing or increasing resources on component performance"?

Figure 5. Results of architectural evaluation based on expert assessment.

References

[1] P. S. John McIlvain, Jason D. Christopher, Cliff Glantz, Fowad Muneer, John Fry, Laura Ritter, "OIL AND NATURAL GAS SUBSECTOR CYBERSECURITY CAPABILITY MATURITY MODEL (ONG-C2M2) 1.1," 2014.

[2] B. Su, H. Zhao, T. Qi, X. Liu, and R. Yu, "Research on Architecture of Intelligent Command and Control System," in 2019 International Conference on Virtual Reality and Intelligent Systems (ICVRIS), pp. 362–364, 2019.

[3] P. Kalarani and S. S. Brunda, "A survey on efficient data mining techniques for network intrusion detection system (IDS)," Int. J. Adv. Res. Comput. Commun. Eng., vol. 3, no. 9, pp. 8028–8031, 2014.

[4] D. A. Eisenberg, D. L. Alderson, M. Kitsak, A. Ganin, and I. Linkov, "Network foundation for command and control (C2) systems: literature review," IEEE Access, vol. 6, pp. 68782–68794, 2018.

[5] D. S. Alberts and R. E. Hayes, "Power to the edge: Command... control... in the information age," 2003.

[6] D. S. Alberts, R. K. Huber, and J. Moffat, "NATO NEC C2 maturity model," 2010.

[7] N. A. Stanton et al., "Development of a generic activities model of command and control," Cogn. Technol. Work, vol. 10, no. 3, pp. 209–220, 2008.

[8] D. P. Jenkins, G. H. Walker, N. A. Stanton, and P. M. Salmon, "Command and Control: The Sociotechnical Perspective," Ashgate Publishing, Ltd., 2012.

[9] D. S. Alberts and R. E. Hayes, "Understanding command and control," 2006.

[10] I. Linkov et al., "Measurable resilience for actionable policy." ACS Publications, 2013.

[11] B. Petrenj, E. Lettieri, and P. Trucco, "Information sharing and collaboration for critical infrastructure resilience--a comprehensive review on barriers and emerging capabilities," Int. J. Crit. infrastructures, vol. 9, no. 4, pp. 304–329, 2013.

[12] M. Grabowski and K. H. Roberts, "Reliability seeking virtual organizations: Challenges for high reliability organizations and resilience engineering," Saf. Sci., vol. 117, pp. 512–522, 2019.

[13] L. Northrop et al., "Ultra-large-scale systems: The software challenge of the future," 2006.

[14] L. Bass, P. Clements, and R. Kazman, "Software architecture in practice, third edition.", Addison-Wesley Professional, 2003.

[15] H. Cervantes and R. Kazman, "Designing software architectures: a practical approach." Addison-Wesley Professional, 2016.

[16] P. Clements, D. Garlan, R. Little, R. Nord, and J. Stafford, "Documenting software architectures: Views and beyond," Proceedings - International Conference on Software Engineering. pp. 740–741, 2003, doi: 10.1109/icse.2003.1201264.

[17] A. W. Brown, "Large-scale, component-based development,", Prentice Hall PTR Englewood Cliffs, vol. 1,2000.

[18] G. Schmutz, D. Liebhart, and P. Welkenbach, "Service-oriented architecture: an integration blueprint: a real-world SOA strategy for the integration of heterogeneous enterprise systems: successfully implement your own enterprise integration architecture using the trivadis integration architecture blu." Packt Publishing Ltd, 2010.

[19] D. A. Chappell, "Enterprise service bus." O'Reilly Media, Inc.," 2004.

[20] J. Lee, K. Siau, and S. Hong, "Enterprise Integration with ERP and EAI," Commun. ACM, vol. 46, no. 2, pp. 54–60, 2003.

[21] A. Dekker, "A taxonomy of network centric warfare architectures," 2008.

[22] A. K. Cebrowski and J. J. Garstka, "Network-centric warfare: Its origin and future," in US Naval Institute Proceedings, 1998, vol. 124, no. 1, pp. 28–35.

[23] D. Laney and others, "3D data management: Controlling data volume, velocity and variety," META Gr. Res. note, vol. 6, no. 70, p. 1, 2001.

[24] D. S. Alberts, "The agility advantage: a survival guide for complex enterprises and endeavors," 2011.

[25] H. Huang, N. Ahmed, and P. Karthik, "On a new type of denial of service attack in wireless networks: The distributed jammer network," IEEE Trans. Wirel. Commun., vol. 10, no. 7, pp. 2316–2324, 2011.

[26] M. A. Mohamed and S. Pillutla, "Cloud computing: a collaborative green platform for the knowledge society," Vine, 2014.

[27] J. Kadtke, I. I. Wells, and others, "Policy challenges of accelerating technological change: Security policy and strategy implications of parallel scientific revolutions," 2014.

[28] B. Krekel, P. Adams, and G. Bakos, "Occupying the information high ground: Chinese capabilities for computer network operations and cyber espionage," Int. J. Comput. Res., vol. 21, no. 4, p. 333, 2014.

[29] N. Buchler, L. Marusich, J. Z. Bakdash, S. Sokoloff, and R. Hamm, "The warfighter associate: objective and automated metrics for mission command," 2013.

[30] E. I. Neaga and M. Henshaw, "A stakeholder-based analysis of the benefits of network enabled capability," Def. Secur. Anal., vol. 27, no. 2, pp. 119–134, 2011.

[31] R. Oosthuizen and L. Pretorius, “Modelling of command and control agility,” 2014.

[32] L. Dodd, M. Lloyd, and G. Markham, "Functional impacts of network-centric operations on future C2," 2005.

[33] R. Oosthuizen and L. Pretorius, "Assessing command and control system vulnerabilities in underdeveloped, degraded and denied operational environments," 2013.

[34] [34] M. Mihailescu, H. Nguyen, and M. R. Webb, "Enhancing wireless communications with software defined networking," in 2015 Military Communications and Information Systems Conference (MilCIS), pp. 1–6, 2015.

[35] V. Chan et al., "Future heterogeneous networks". National Science Foundation, 2011.

[36] M. Fidjeland and B. K. Reitan, "Web-oriented architecture: network-based defence development made easier," 2009.

[37] T. Zhang, "Optimization of spectrum allocation in cognitive radio and dynamic spectrum access networks," 2012.

[38] F. Junyent, V. Chandrasekar, D. McLaughlin, E. Insanic, and N. Bharadwaj, "The CASA Integrated Project 1 networked radar system," J. Atmos. Ocean. Technol., vol. 27, no. 1, pp. 61–78, 2010.

[39] M. Haghnevis, R. G. Askin, and D. Armbruster, "An agent-based modeling optimization approach for understanding behavior of engineered complex adaptive systems," Socioecon. Plann. Sci., vol. 56, pp. 67–87, 2016.

[40] J. Wang et al., "Toward a resilient holistic supply chain network system: Concept, review and future direction," IEEE Syst. J., vol. 10, no. 2, pp. 410–421, 2014.

[41] E. Alfnes and J. O. Strandhagen, "Enterprise design for mass customisation: The control model methodology," Int. J. Logist., vol. 3, no. 2, pp. 111–125, 2000.

[42] M. Haghnevis, "An agent-based optimization framework for engineered complex adaptive systems with application to demand response in electricity markets," Arizona State University, 2013.

[43] J. Crebolder, S. Pronovost, and G. Lai, "Investigating virtual social networking in the military domain," in Proceedings of the 14th International Command and Control Research and Technology Symposium, Washington, DC, June, pp. 15–17, 2009.

[44] H. Joglar-Espinosa, I. Seccatore-Gomez, and J. Lamas-Barrientos, "Testing edge versus hierarchical c2 organizations using the elicit platform and common identification picture tool," 2011.

[45] D. K. Brown, "More than a capable mariner: Meeting the challenges of command at sea—Views from the bridge," Capella University, 2012.

[46] K. Chan, J.-H. Cho, and S. Adali, "A trust based framework for information sharing behavior in command and control environments," 2013.

[47] M. Persson and A. Worm, "Information experimentation in command and control," 2002.

[48] B. Solaiman, E. Bosse, L. Pigeon, D. Gueriot, and M. C. Florea, "A conceptual definition of a holonic processing framework to support the design of information fusion systems," Inf. Fusion, vol. 21, pp. 85–99, 2015.

[49] M. Joblin, S. Apel, and W. Mauerer, "Evolutionary trends of developer coordination: A network approach," Empir. Softw. Eng., vol. 22, no. 4, pp. 2050–2094, 2017.

[50] N. A. Stanton et al., "Experimental studies in a reconfigurable C4 test-bed for network enabled capability," 2006.

[51] T. Gregory, "Traveling of requirements in the development of packaged software: An investigation of work design and uncertainty," 2014.

[52] J. M. Schraagen, M. H. in 't Veld, and L. De Koning, "Information sharing during crisis management in hierarchical vs. network teams," J. contingencies Cris. Manag., vol. 18, no. 2, pp. 117–127, 2010.

[53] K. Chan, J.-H. Cho, and A. Swami, "Impact of trust on security and performance in tactical networks," 2013.

[54] K. Chan, R. Pressley, B. Rivera, and M. Ruddy, "Integration of communication and social network modeling platforms using elicit and the wireless emulation laboratory," 2011.

[55] K. S. Chan and N. Ivanic, "Connections between communications and social networks using ELICIT," 2010.

[56] S. Noel et al., "Analyzing mission impacts of cyber actions (AMICA)," 2015.

[57] Y. Feng, B. Xiu, and Z. Liu, "A dynamic optimization model on decision-makers and decision-layers structure (DODDS) in C2-organization," Comput. Model. New Technol, vol. 18, no. 2, pp. 192–198, 2014.

[58] N. A. Stanton et al., "A reconfigurable C4 testbed for experimental studies into network enabled capability," 2005.

[59] É. Bossé and B. Solaiman, Information fusion and analytics for big data and IoT. Artech House, 2016.

[60] H. T. Tran and D. N. Mavris, "A system-of-systems approach for assessing the resilience of reconfigurable command and control networks," in AIAA Infotech@ Aerospace, p. 640, 2015.

[61] H. T. Tran, J. C. Domercant, and D. Mavris, "Trade-offs between command and control architectures and force capabilities using battlespace awareness," 2014.

[62] G. H. Walker et al., "From ethnography to the EAST method: A tractable approach for representing distributed cognition in Air Traffic Control," Ergonomics, vol. 53, no. 2, pp. 184–197, 2010.

[63] G. H. Walker et al., "Using an integrated methods approach to analyse the emergent properties of military command and control," Appl. Ergon., vol. 40, no. 4, pp. 636–647, 2009.

[64] N. A. Stanton, L. Rothrock, C. Harvey, and L. Sorensen, "Investigating information-processing performance of different command team structures in the NATO Problem Space," Ergonomics, vol. 58, no. 12, pp. 2078–2100, 2015.

[65] N. A. Stanton, G. H. Walker, and L. J. Sorensen, "It’s a small world after all: contrasting hierarchical and edge networks in a simulated intelligence analysis task,” Ergonomics, vol. 55, no. 3, pp. 265–281, 2012.

[66] D. M. Wynn, M. Ruddy, and M. E. Nissen, “Command & control in virtual environments: Tailoring software agents to emulate specific people,” 2010.

[67] A. Wong-Jiru, “Graph theoretical analysis of network centric operations using multi-layer models,” 2006.

[68] G. H. Walker et al., “Analysing network enabled capability in civilian work domains: a case study from air traffic control,” 2005.

[69] J. de Visser, P. A. Wieringa, J. Moss, and Y. Xiao, “Supporting distributed planning in a dynamic environment: An observational study in operating room management,” Hum. Decis. Mak. Control, 2002.

[70] U.S. Department of Defense, “The DoDAF Architecture Framework Version 2.02 - [Online], Access year: 2021.” https://dodcio.defense.gov/library/dod-architecture-framework/.

[71] “The FINSEC Reference Architecture (RA) - [Online], Access year: 2021.” https://finsecurity.eu/.

[72] The Open Group, “Architecture Framework TOGAFTM Version 9.2 - [Online Version], , Access year: 2021.” https://pubs.opengroup.org/architecture/togaf91-doc/arch/chap03.html.