IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 2 May 2023, accepted 8 June 2023, date of publication 22 June 2023, date of current version 14 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3288428

RESEARCH ARTICLE

Assessment of the Distributed Ledger Technology for Energy Sector Industrial and Operational Applications Using the MITRE ATT&CK® ICS Matrix

ANNABELLE LEE[1], (Member, IEEE), SRI NIKHIL GUPTA GOURISETTI[2], (Member, IEEE), DAVID JONATHAN SEBASTIAN-CARDENAS[3], (Member, IEEE), KENT LAMBERT[4], (Member, IEEE), VICENTE NAVARRO[5], (Member, IEEE), MARCO PASETTI[6], (Member, IEEE), ÜMIT CALI[7], (Senior Member, IEEE), KATERYNA ISIROVA[8], (Member, IEEE), RAMESH REDDI[9], (Member, IEEE), PUICA NITU[10], (Member, IEEE), MD. TOUHIDUZZAMAN[11], (Member, IEEE), MICHAEL MYLREA[12], (Member, IEEE), PHILIP HUFF[13], (Member, IEEE), FARROKH RAHIMI[14], (Life Senior Member, IEEE), AND SHAMMYA SHANANDA SAHA[15], (Member, IEEE)

[1]Nevermore Security, Evergreen, CO 80439, USA
[2]National Resilience Inc., San Diego, CA 92121, USA
[3]Pacific Northwest National Laboratory, Richland, WA 99354, USA
[4]BlockFrame Inc., Colorado Springs, CO 80919, USA
[5]Faculty of Electrical Engineering, Technological University of Panama, Panama City 0819-07289, Panama
[6]Department of Information Engineering, University of Brescia, 25123 Brescia, Italy
[7]Department of Electric Energy, Norwegian University of Science and Technology, 7491 Trondheim, Norway
[8]KPMG, 8036 Zürich, Switzerland
[9]CybSecBCML Inc., Raleigh, NC 27615, USA
[10]Canadian Institute World Energy System, Toronto, ON M5X 1C9, Canada
[11]Pacific Northwest National Laboratory, Portland, OR 97204, USA
[12]Institute for Data Science and Computing, University of Miami, Miami, FL 33146, USA
[13]Department of Computer Science, University of Arkansas at Little Rock, Little Rock, AR 72204, USA
[14]Open Access Technology International Inc., Minneapolis, MN 55418, USA
[15]Electric Power Research Institute, Knoxville, TN 37932, USA

Corresponding author: Ümit Cali (umit.cali@ntnu.no)

ABSTRACT In recent times, Distributed Ledger Technology (DLT) has gained significant attention for its potential application in the energy sector. Utilizing blockchain and DLT has demonstrated the ability to enhance the resilience of the electric infrastructure, which will support a more flexible infrastructure and advance grid modernization. However, the deployment of these technologies increases the overall attack surface. The MITRE ATT&CK® matrices have been developed to document an adversary's tactics and techniques based on real-world observations. The MITRE ATT&CK® matrices provide a common taxonomy for offense and defense and have become a valuable conceptual tool across multiple cybersecurity disciplines for conveying threat intelligence, performing testing through red teaming or adversary emulation, and enhancing network and system defenses against intrusions. The MITRE ATT&CK® for Industrial Control Systems (ICS) matrix was created to provide knowledge about adversary behavior in the ICS technology domain. This study analyzes the relevance of various tactics and techniques across a seven-layer DLT engineering and cybersecurity stack, known as the DLT stack, designed by the Cybersecurity Taskforce under IEEE P2418.5 - Standard for Blockchain in Energy working group sponsored by Power and Energy Systems - Smart Buildings, Loads and Customer Systems (PES/SBLC) Technical Committee. Additionally, this paper identifies specific mitigation strategies tailored to the energy ICS environment.

The associate editor coordinating the review of this manuscript and approving it for publication was Peng-Yong Kong.

INDEX TERMS Cybersecurity, distributed ledger technology, MITRE ICS ATT&CK®, power systems, resiliency.

I. INTRODUCTION

Distributed Ledger Technology (DLT) has shown great potential in enhancing energy infrastructure's security, control, and resilience [1], [2]. The ability to track and trace end-to-end energy processes and data transactions is one of the security benefits that can aid in responding to cyber-attacks and natural environmental hazards. Furthermore, transitioning from centralized to distributed architectures with redundancy can increase the fault tolerance and resiliency of the network while also increasing overall cyber complexity. While the opportunities of applying DLT across various sections have been examined, the literature lacks an examination of how adversaries can exploit vulnerabilities in DLT applications for energy delivery using various Tactics, Techniques, and Procedures (TTPs). Rapid DLT adoption without a comprehensive evaluation of its security and resiliency in the context of energy applications can harm the Operational Technology (OT) infrastructure. Although the inherent components of DLT, such as the database, consensus mechanism, data transmission, and smart contracts, are not novel and have existed as standalone software components for several decades, adversaries can exploit common vulnerabilities of these individual software components to compromise the DLT and impact the energy/OT infrastructure interacting with it. For example, Apache CouchDB is sometimes used as an external state database for Hyperledger Fabric DLT [3]. Still, according to the Common Vulnerabilities and Exposures (CVE) database, Apache CouchDB has 16 CVEs, with 5 having a Common Vulnerability Scoring System (CVSS) score of 9 or 10 (critical) [4]. A similar analysis can be applied to Fabric's chaincode (smart contract) languages, such as Go, Node.js, and Java. Vulnerability analyses can be performed on any DLT protocols, components, and applications, as well as the deployment and management of the DLT since adversaries can exploit existing common vulnerabilities within the DLT components using known TTPs. Therefore, an adversarial TTP analysis is necessary for any DLT.

The present study builds upon the preliminary findings discussed in [5] to address the aforementioned research gaps by utilizing the MITRE ATT&CK® Industrial Control Systems (ICS) matrix [6], [7], [8] to evaluate each layer of the DLT engineering and cybersecurity stack (DLT stack), defined in [9]. The MITRE ATT&CK® ICS matrix is a tool that analyzes TTPs from real-world cyber events, with tactics representing the objectives of an adversary and techniques illustrating how an adversary achieves those objectives [10]. This matrix [6] was developed based on publicly available threats and incidents, and the techniques under the tactics were scoped according to past cyber events. One of the general uses of the MITRE ATT&CK® ICS matrix is to identify if an ICS/OT network is susceptible to threat actors that use the techniques identified in the matrix to exploit vulnerabilities. The study considers adversarial TTPs in emerging DLT architectures, where adversaries can use various techniques to exploit potential vulnerabilities in energy delivery systems. To the authors' knowledge, no relevant events or incidents related to DLT-based ICS/OT use cases have been made publicly available. Using the MITRE ATT&CK® ICS matrix to evaluate the relevance of the techniques across the DLT stack is consistent with the proposed use defined in the MITRE reference document [10]: "ATT&CK® for ICS supports many use cases, including failure scenario development, education, and the existing ATT&CK® use cases". This paper aims to identify potential attack tactics and techniques and mitigation strategies.

This ATT&CK® ICS matrix examines TTP from real-world cyber events. Cyber adversaries are complex, nonlinear, and evolving. These groups will continue using various TTP to exploit energy delivery systems and infrastructure vulnerabilities. This paper examines how the MITRE ATT&CK® ICS tactics and techniques may be applied to each layer of the DLT stack, from the application layer to the physical layer. This work provides information to assist in understanding and mitigating cyber threats to a deployed DLT while addressing the following questions:

- What are the potential TTPs that can be used to exploit potential vulnerabilities in the DLT stack?
- How can these potential vulnerabilities be mitigated to reduce/remove the potential impact on the other layers in the DLT stack?
- How can the proposed framework be applied in real life applications to increase the cyber-physical resilience of critical infrastructure like power systems?

This paper helps answer these questions and provides insight on leveraging the MITRE ATT&CK® ICS matrix in assessing a planned or deployed DLT and remediating/mitigating those threats in applying DLT to advance the grid modernization goals in the electricity infrastructure [11]. This research is timely as the digital transformation of the energy infrastructure is expanding the attack surface and the number of vulnerabilities that adversaries can exploit [12].

The remainder of the paper is organized as follows: Section II provides an overview of distributed ledger technology, while Section III discusses current exploratory applications of DLT in the energy sector. Section IV-A discusses the cybersecurity risk management process, while Section IV-B introduces cybersecurity resilience. After introducing the DLT engineering and cybersecurity stack in Section V, Section VI presents a detailed relationship analysis between DLT and the MITRE ATT&CK® ICS matrix. The technical details about the mapping process between the DLT stack and the MITRE ATT&CK® ICS matrix and all the mappings are presented in Sections VII and VIII. Finally, Section IX summarizes the paper's conclusions and provides recommendations for future work.

II. DLT OVERVIEW

Distributed Ledger Technology is based on information and communication infrastructures and protocols that allow simultaneously distributed access, validation, and record updating of a distributed ledger in an immutable manner across a network of multiple stakeholders, entities, and locations. Storage of information is done securely and accurately using keys and cryptographic signatures. The information stored in the ledger(s) becomes an immutable database, and the rules of the underlying DLT network govern any further processing. The decentralized distributed nature of the ledger makes it resilient to various cyber-attacks requiring all copies stored across the network to be compromised simultaneously for the adversary to succeed.

Distributed ledgers can be classified as public or private, permissionless or permissioned, or any combination. A variety of DLTs exist. Blockchain is the most popular, but others, such as Hashgraph, Directed Acyclic Graph (DAG), Holochin, and Tempo, are also utilized.

The *Consensus* mechanism plays a central role in the DLT architecture. It is used to have transactions verified and validated. Several consensus protocols depend on the primary objective of the specific DLT network. The widely used consensus protocols include Proof of Work (PoW), Proof of Stake (PoS), and Proof of Authority (PoA). In the PoW model, a user is rewarded [13] by being the first to solve a difficult puzzle which is computationally-intensive to solve but easy to verify by others when solved. This consensus model is used by Bitcoin but is seldom used in energy DLT applications. The PoS model is based on the premise that the more stake a user has invested into the system, the less likely they will want to subvert it. With this consensus model, there is no need to perform resource-intensive computations (involving time, processing power, and electricity). The PoA consensus model relies on the partial trust of publishing nodes. Publishing nodes must have their identities proven and verifiable within the DLT network. This algorithm only applies to permissioned DLT networks with high levels of trust. Energy DLT applications often adopt the PoS, PoA, or similar consensus models in permissioned environments.

III. DLT IN THE ELECTRIC SECTOR

In recent years, significant attention has been paid to the potential use of DLT technology in energy applications [14], [15], [16]. Examples include the use of DLT technology in peer-to-peer [17], [18] and peer-to-market energy trading [19], tracking of renewable energy credits [20], electric vehicle charge management [21], trading and record-keeping in energy trading [22], Optimal Power Flow (OPF) solutions [23], and more [24]. It can support the creation of new asset classes like crypto-tokens backed by renewable resources [25]. DLTs have also shown the potential to improve the resilience of the electric infrastructure while allowing significant increases in speed, size, and frequency of data exchange [26]. This digital transformation of energy and other critical infrastructures is increasing external connections to ICS.

However, these gains in advancing grid modernization give impetus to a more agile infrastructure incorporating distributed energy resources in ways that can better respond to all hazards, from naturally occurring environmental hazards to cyber attacks. Therefore, the advantages of the DLT need to be assessed against potential cyber threats as organizations often lack monitoring, visibility, and defenses for their ICS environments [12]. In many energy-related DLT applications, external triggers (known as ''oracles'') play an essential role where smart contracts are triggered by external metering or telemetry signals such as the delivery of a certain quantity of energy or generation from a clean source of energy. The cyber threat seeking to exploit this expanded attack surface can exploit people, processes, and technology aspects of DLT using various TTPs. Cyber threats targeting ICS that underpin the security and operations of critical infrastructure continue to increase, requiring a detailed review of the TTPs.

IV. CYBERSECURITY RISK MANAGEMENT AND RESILIENCE

A. CYBERSECURITY RISK MANAGEMENT

Comprehensive Cyber Risk Management (CRM) requires proactive cyber risk governance with an end-to-end operational and cyber risk assessment to assess the confidentiality, availability, and integrity of information and technology assets. A cyber risk management framework must include a multi-layer, preventive cyber security controls, and measures to safeguard the entire digital chain. A proper risk policy will allow all market agents with different cyber risk appetites and tolerance thresholds that differ according to the asset size and technology to assess their risk exposure according to the same set of rules.

This holistic approach can address specific security needs for the DLT by addressing the upstream and downstream interconnectors with consistent governing principles to cyber security risk management that include:

- A cybersecurity framework for the users/systems located in any downstream or upstream connection to a DLT,
- Consistent reporting mechanism for risk-based incidents and threats,
- Governance and risk oversight of internal controls with risk mitigation techniques, and
- A disclosure policy for cyber security assessments and identified threats.

Comprehensive risk identification, assessment, mitigation, and monitoring apply to all downstream entities, including those not directly connected to a DLT. Some of these assets may be more vulnerable to threats due to their simpler architecture and fewer software-based defenses. A holistic cyber risk management framework should include a reporting mechanism to the DLT and upstream entities of all incidents throughout the ecosystem. This cyber risk framework will provide for the assessment and impact of potential risk

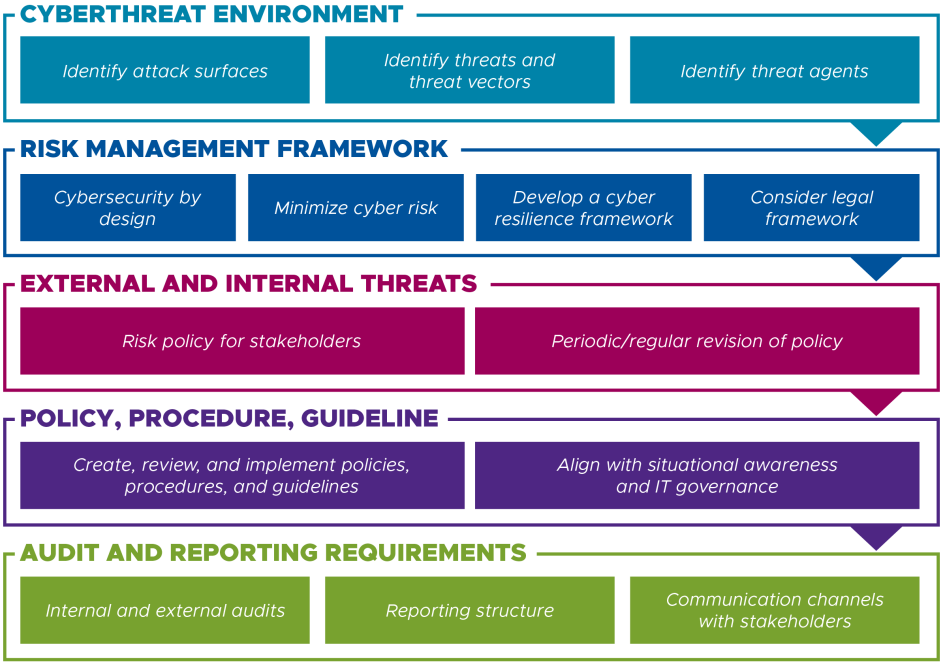

FIGURE 1. Integrated Risk Management Framework Components.

throughout the entire ecosystem. Figure 1 exemplifies the building blocks of an integrated cyber risk framework.

B. CYBERSECURITY RESILIENCE

As defined by [8], ''cybersecurity resilience is the ability of a system to continue operating under adverse conditions, stress, or an attack. The system may operate in a degraded mode while maintaining essential operational capabilities''. Cyber resiliency is critical to energy systems since an ICS/OT network generally cannot be disconnected if an adversary or attack is detected. Similarly, cyber resilience is critical for DLTs since the DLT cannot be taken off the network if adversaries or attacks are detected. For DLTs, cybersecurity resilience may be achieved by implementing fault-tolerant principles in the design of the DLT, such as Crash Fault Tolerance (CFT) or Byzantine Fault Tolerance (BFT). Although private and permissioned DLT deployment for a use case could minimize the risk of adversarial presence, a controlled environment does not eliminate cybersecurity risk. One objective of assessing DLT-based use cases and applications in the energy sector and specifically in the ICS/OT environment is to ensure that the cybersecurity and cybersecurity resilience of the energy ICS/OT system/network is not deprecated but enhanced. For example, suppose DLT is used to assist with energy operations data aggregation and analytics. In that case, the cybersecurity resilience analysis is as follows: 1) resilient operations of the electro-mechanical systems contributing to operations (hardware perspective), 2) resilience of the DLT assisting in the operations, including executing distributed applications (software perspective), and 3) resilience analysis of the overall system (both hardware and software - combined perspective).

Ensuring cybersecurity resilience may enable the ICS/OT systems/networks to continue functioning even during a cyber attack. In all operational states, detecting anomalies, including vulnerabilities, that may be used to exploit operations is crucial. The overall risk must be assessed to identify, select, and implement defensive and reactive response actions.

V. DLT STACK LAYERS

The DLT engineering and cybersecurity stack defined in [9] is a seven-layer DLT cybersecurity and engineering stack that comprises several relevant components and attributes and is designed for researchers, DLT technology developers, and end users (such as utilities). The DLT stack can be used as an architectural framework that is synergistic with the function of power grid applications. Furthermore, the DLT engineering and cybersecurity stack is designed to be used with existing cybersecurity and DLT applicability models [27], [28]. Included below are descriptions of the DLT stack layers extracted from the referenced paper:

A. APPLICATION LAYER

This layer contains applications, software, scripts, and programs that the users can use (e.g., human users and nodes) to interact with the DLT. These software applications are above the DLT core and therefore do not fully belong to the DLT. Elements of the application layer can trigger rule bases and program code (such as smart contracts, chaincode, atomic swaps [29], etc.) that reside in the execution layer (below).

Other elements of this layer include User Interface/Graphical User Interface (UI/GUI), performance analysis applications such as Hyperledger CaliperTM, etc. [1].

B. EXECUTION LAYER

This layer contains the DLT rules and program logic, such as smart contracts, chaincode, etc. The software applications from the application layer trigger the code and rules in the execution layer and instruct the code in the execution layer, which results in the execution of a transaction. In cases where the execution layer code requires data from off-chain databases, the code can trigger oracles that reside in the application layer (or between the application layer and the outside world) to fetch data/information from off-chain sources to the execution layer code.

C. CONSENSUS LAYER

The consensus layer is a critical component of DLT technologies facilitating distributed trust, ownership, and control. In this, widespread consensus-forming nodes across different geographical and network locations work independently toward the consensus of transactions. There are two common consensus types: (1) lottery-based and (2) voting-based consensus. Consensus has two main properties: (1) indicates an agreement among the distributed nodes and synchronizes them, and (2) validates transactions and ensures reliable and fault-tolerant operations.

D. DATA MODEL LAYER

This layer handles functions and operations related to block creation and ledger maintenance tasks. This layer does not define the final ledger state, and a global consensus is required to approve the final transactions and block creations. However, the process of grouping the transactions into the block, creating a block (or appending to the ledger), maintaining a common state of the ledger, etc., are handled in this layer. Functions in this layer are primarily related to data orchestration processes but in the context of distributed databases, ledgers, etc. Examples of such processes are grouping or arranging the transactions into blocks, appending the block to the distributed ledger, and updating data-structure/ledger across the network (e,g,, via replication). The block content or structure of a block depends on the blockchain/DLT technology. When transactions are submitted to a blockchain network, the transactions are ordered in a block.

E. NETWORK LAYER

The network layer corresponds to the communication infrastructure facilitating transaction, information, and data sharing between the nodes. Protocols and methods to facilitate discovery and communication between peer nodes belong in this layer. If nodes are expected to transact by digitally signing data-in-transit or engage in verification and validation of the transactions, such processes should be defined in this layer. Transport Layer Security (TLS) and other secure node-to-node handshaking mechanisms should be identified under this layer. Standard protocols are recommended instead of custom-defining new/proprietary protocols.

F. INFRASTRUCTURE LAYER

This layer corresponds to the virtual and physical computers or software agents participating as authorized blockchain nodes. The nodes should be capable of performing cryptographic operations (such as digital signature and hashing), maintaining and varying the identity of other nodes, and providing identity information for authentication and authorization by the network/other nodes. Depending on the DLT, security aspects related to Membership Service Provider (MSP) and Active Directory (AD) fall within this layer. Hence, tools and processes facilitate access controls, define the identity of the nodes, and ensure permissions belong in this layer as part of the nodes. Furthermore, aspects related to on-chain and off-chain storage infrastructures are included in this layer.

G. PHYSICAL LAYER

This layer may not be relevant in several use cases. The use of DLT-based industrial use cases where Internet of Things (IoT) devices and sensors play a central role is an emerging trend [30]. However, in use cases where sensors and IoT devices are expected to participate in the blockchain, these systems are expected to be part of this layer. The sensor systems may not have the capacity or capability to join as nodes in the DLT directly. In such cases, the sensors would need to interact with the middleware agents as part of the infrastructure layer to participate in the DLT network.

VI. THE MITRE ICS ATT&CK® MATRIX

The MITRE ATT&CK® ICS matrix uses principles from the cyber kill chain [31] and information acquired from past cyber incidents to evaluate what an adversary can do to a system and how the adversary can accomplish their goals. A simplified overview of the MITRE ATT&CK® ICS Matrix tactics shown below illustrates the 12 tactics and what the adversary is trying to achieve [1]:

1) *Initial Access:* The adversary is trying to use entry vectors to gain an initial foothold within an ICS environment. These techniques include compromising OT assets, Information Technology (IT) resources in the OT network, and external remote services and websites. They may also target third-party entities and users with privileged access. IT resources in the OT environment are also potentially vulnerable to the same attacks as enterprise IT systems. Trusted third parties of concern may include vendors, maintenance personnel, engineers, external integrators, and other outside entities involved in expected ICS operations. Vendor-maintained assets may include

[1]The text included with each tactic is a summary from the MITRE ICS ATT&CK® website. Additional content is provided at [6].

physical devices, software, and operational equipment. Initial access techniques may also leverage outside devices, such as radios, controllers, or removable media, to remotely interfere with and possibly infect OT operations.

2) *Execution:* The adversary is trying to run code or manipulate system functions, parameters, and data unauthorizedly. Execution consists of techniques that result in adversary-controlled code running on a local or remote system, device, or other assets. This execution may also rely on unknowing end users or manipulating device operating modes. Adversaries may infect remote targets with programmed executables or malicious project files that operate according to specified behavior and alter expected device behavior subtly. Commands for execution may also be issued from command-line interfaces, Application Programming Interfaces (APIs), GUIs, or other available interfaces. Techniques that run malicious code may also be paired with techniques from other tactics.
3) *Persistence:* The adversary is trying to maintain their foothold in the ICS environment. Persistence consists of techniques that adversaries use to maintain access to ICS systems and devices across restarts, changed credentials, and other interruptions that could cut off their access. Techniques used for persistence include any access, action, or configuration changes that allow them to secure their ongoing activity and keep their foothold on systems. This may include replacing or hijacking legitimate code, firmware, and other project files, adding startup code, and downloading programs onto devices.
4) *Privilege Escalation:* The adversary is trying to gain higher-level permissions. Privilege Escalation consists of techniques adversaries use to gain higher-level permissions on a system or network. Adversaries can often enter and explore a network with unprivileged access but require elevated permissions to follow through on their objectives. Common approaches are to take advantage of system weaknesses, misconfigurations, and vulnerabilities.
5) *Evasion:* The adversary is trying to avoid security defenses. Evasion consists of techniques adversaries use to avoid technical defenses throughout their campaign. Techniques used for evasion include the removal of indicators of compromise, spoofing communications, and exploiting software vulnerabilities. Adversaries may also leverage and abuse trusted devices and processes to hide their activity, possibly by masquerading as master devices or native software. Methods of defense evasion for this purpose are often more passive.
6) *Discovery:* The adversary is locating information to assess and identify their targets in the environment. Discovery consists of techniques adversaries use to survey the ICS environment and gain knowledge about the internal network, control system devices, and how their processes interact. These techniques help adversaries observe the environment and determine the next steps for target selection and Lateral Movement. They also allow adversaries to explore what they can control and gain insight into interactions between various control system processes. Adversaries may use Discovery techniques that result in Collection to determine how available resources benefit their current objective.
7) *Lateral Movement:* The adversary is trying to move through the ICS environment. Lateral Movement consists of techniques adversaries use to enter and control remote systems on a network. These techniques abuse default credentials, known accounts, and vulnerable services and may also leverage dual-homed devices and systems that reside on both the IT and OT networks. The adversary uses these techniques to pivot to their next point in the environment, positioning themselves to where they want to be or think they should be. Reaching this objective often involves pivoting through multiple systems, devices, and accounts. Adversaries may install their remote tools to accomplish Lateral Movement or leverage default tools, programs, manufacturer set, or other legitimate credentials native to the network, which may be stealthier.
8) *Collection:* The adversary is trying to gather data of interest and domain knowledge on the ICS environment to inform their goal. The collection consists of adversaries' techniques to gather domain knowledge and obtain contextual feedback in an ICS environment. This tactic is often performed as part of Discovery to compile data on control systems and targets of interest that may be used to follow through on the adversary's objective. Examples of these techniques include observing operation states, capturing screenshots, identifying unique device roles, and gathering system and diagram schematics. Collection of this data can play a key role in planning, executing, and even revising an ICS-targeted attack. Collection methods depend on the targeted data categories, including protocol-specific, device-specific, and process-specific configurations and functionality. Information collected may pertain to system, supervisory, device, and network-related data, which conceptually fall under high, medium, and low levels of plan operations.
9) *Command and Control:* The adversary is trying to communicate with, and control compromised systems, controllers, and platforms with access to the ICS environment. Command and Control consist of techniques adversaries use to communicate with and send commands to compromised systems, devices, controllers, and platforms with specialized applications in ICS environments. Adversaries often seek to use commonly available resources and mimic expected network traffic to avoid detection and suspicion. Command and Control may be established to varying degrees of stealth,

often depending on the victim's network structure and defenses.

10) *Inhibit Response Function:* The adversary tries to prevent safety, protection, quality assurance, and operator intervention functions from responding to a failure, hazard, or unsafe state. Inhibit Response Function consists of adversaries' techniques to hinder the safeguards for processes and products. These techniques aim to actively deter and prevent expected alarms and responses due to statuses in the ICS environment. Adversaries may modify or update system logic or even outright prevent responses with a Denial-of-Service (DoS). They may result in the prevention, destruction, manipulation, or modification of programs, logic, devices, and communications. As prevention functions are generally dormant, reporting and processing functions can appear fine but may have been altered to prevent failure responses in dangerous scenarios.
11) *Impair Process Control:* The adversary tries to manipulate, disable, or damage physical control processes. Impair Process Control consists of techniques that adversaries use to disrupt control logic and cause detrimental effects to processes being controlled in the target environment. Targets of interest may include active procedures or parameters that manipulate the physical environment. These techniques can also include preventing or manipulating reporting elements and control logic. If an adversary has modified process functionality, they may also obfuscate the results, which are often self-revealing in their impact on the outcome of a product or the environment. The direct physical control these techniques exert may also threaten the safety of operators and downstream users, which can prompt response mechanisms.
12) *Impact:* The adversary is trying to manipulate, interrupt, or destroy the ICS systems, data, and their surrounding environment. Impact consists of techniques that adversaries use to disrupt, compromise, destroy, and manipulate the integrity and availability of control system operations, processes, devices, and data. These techniques encompass the influence and effects of adversarial efforts to attack the ICS environment or that tangentially impact it. Impact techniques can result in more instantaneous disruption to control processes and the operator or long-term damage or loss to the ICS environment and related operations.

VII. DLT STACK AND MAPPING PROCESS FLOW

This paper assesses the DLT stack against the ATT&CK® ICS matrix to understand the adversarial pivot points better. This includes identifying an adversary's tactics and techniques to compromise one or more DLT stack layers. Tactics represent the "why" of an attack technique. It is the adversary's tactical goal. The techniques represent "how" an adversary achieves a tactical goal by acting. The following analysis is based on the specific language and descriptions included in the MITRE ATT&CK® ICS techniques.[2] The IEEE SA P2418.5 working group's cybersecurity task force investigates cyber resilience of DLT-based energy applications in the ICS/OT environment against the MITRE ATT&CK® ICS matrix through the following activities:

1) The task force has been assessing the mapping between the DLT stack against the MITRE ATT&CK® ICS matrix. Mapping the stack and the matrix can help identify the implications of various tactics and techniques that an adversary may use to compromise one or more of the seven DLT stack layers. This can provide valuable insights into possible exploits in DLTs and assist in defining security and resilience measures;
2) Identifying these tactics/techniques can also be used in a risk assessment for an ICS/OT system or network. The objective is to identify mitigation strategies to address cybersecurity resilience, including proactive and reactive cybersecurity controls;
3) The task force aims to use the mapping to provide guidance that can assist DLT designers and DLT users (application owners) in increasing the overall cyber resilience and cybersecurity for energy sector ICS/OT systems and networks — leading to a cyber resilient model;
4) A cyber resilience model should support a comprehensive cyber risk management framework with a well-defined multi-layer and preventive cyber security controls and measures to safeguard the entire digital chain. Furthermore, an acceptable risk policy should allow all market agents with different cyber risk appetites and tolerance thresholds that differ according to the asset size and technology to self-assess their risk exposure according to the same set of rules.

Figures 2 and 3 illustrate the process used to develop the mapping of the MITRE ATT&CK® ICS tactics and techniques against the layers in the DLT stack. Figure 2 illustrates the process used to identify the various tactics and techniques applicable at each layer in the DLT stack. Figure 3 expands upon Figure 2 and shows the overall process flow of the analysis as a component of a cyber resilience model process.

The first three steps involve the development builds of the DLT stack, which was completed by the task force and presented in [9]. Steps 4-8 of Figure 3 encompass the pre- CRM analysis that evaluates the relevance of various tactics and techniques from the MITRE ATT&CK® ICS matrix against the DLT stack. Using the mapping information between the DLT stack and the MITRE ATT&CK® ICS Matrix along with the DLT stack descriptions, the final steps include the completion of the CRM phase (steps 9-12 of Figure 3).

Since the sequence on which the tactics listed by MITRE (left-to-right) is based on adversarial propagation on a kill chain or similar construct, the same sequence has been

[2]Some of the techniques, for example, *Project File Injection (T0873)*, *Detect Operating Mode (T0868)* and *I/O Image (T0877)* that are tailored to a specific device, such as a Programmable Logic Controller (PLC) was not included.

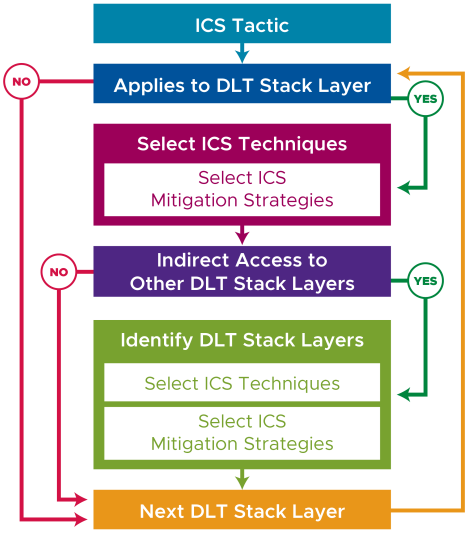

FIGURE 2. Analysis Process.

followed for this mapping. Starting with the first tactic (Initial Access), each technique has been revised to evaluate its applicability to each DLT stack layer. If the tactic applies, the relationship is further analyzed to identify the specific techniques and mitigation strategies. This process is repeated for all of the 12 tactics. The result is a detailed map between the DLT stack and the MITRE ATT&CK® ICS tactics and techniques. The mitigation recommendations (step 7 of Figure 3) are selected from MITRE's existing mitigations for the tactics and techniques. After an initial analysis, indirect access is analyzed to determine if an adversary can compromise DLT components at other stack layers.

The determination of indirect implications was made based on the tactic that was being analyzed, the objective of that tactic, and how this would apply at each DLT stack layer. Direct vs. indirect implications was based on the following approach: 1) first, the objective of a tactic in the context of ICS environment was analyzed; then 2) a tactic's direct application/relevance to a DLT layer was evaluated. This was done by studying the techniques that are listed under a particular tactic in scope. In addition, ''indirect implications'' can be synonymously thought of as ''indirect impact''. Specifically, if an adversary's action could impact a different stack layer's components, that can be defined as ''indirect access''.

The analysis presented in this work is based on several assumptions, as cyber resilience and cybersecurity are broad areas for any application. The assumptions of the present work are:

Assumption 1: Every DLT architecture includes the following three components:

1) Smart contracts or similar artifacts that allow defining rules, programs, etc.;
2) A distributed ledger or a database-style artifact that is immutable and captures transactional history;
3) A consensus mechanism or similar artifact that allows for fault-tolerant (crash or byzantine) group agreement and is subject to a governance model.

Assumption 2: Immutability of the ledger is achieved through cryptography. If the cryptographic algorithm is compromised, the immutability of the ledger should be assumed to be compromised.

Assumption 3: The DLT includes the seven layers defined in the DLT stack.

Assumption 4: Only permissioned/private DLT is encouraged to use in ICS/OT applications/use cases in the energy sector. This is important because the participating entities (e.g., utilities, regulators, auditors, etc.) require mechanisms to ensure non-repudiation in case of anomalous behavior.

Assumption 5: The following common security practices are implemented for the ICS/OT systems and networks: configuration management, auditing, system monitoring, a security software design and development, secure product life-cycle management, implementation of a security framework, and security patch management.

Assumption 6: Many general cybersecurity controls address data security and privacy across different vendors and manufacturers. Many of these cybersecurity controls are outside the scope of a DLT. Routine auditing of the DLT ledgers should be an ongoing process to detect malicious activity.

Assumption 7: Use case-specific threat models and attack trees are outside the scope of this analysis. The paper focuses on assessing potential compromises to a DLT and not on specific OT/ICS use cases. Specific use cases can be developed in future papers. Given this focus, false data injection from distributed IoT devices and distributed bots, etc., are outside the scope of this paper, even though they may adversely impact the DLT.

Assumption 8: No backdoors have been implemented to infiltrate an organization that develops/maintains a DLT. If backdoors have been installed, exploitation is outside the scope of the DLT.

Assumption 9: Attacks described below are limited to the DLT environment and the DLT stack. Attacks at the system level are outside the scope of this paper.

Assumption 10: The effects of the DLT making or influencing decisions in the OT environment are outside the scope.

Assumption 11: The effects of incorrect/invalid or missing data incoming from field devices to a DLT is outside the scope.

Assumption 12: Use of a DLT for command and control operations is outside the scope of this paper.

Assumption 13: This analysis focuses solely on the MITRE ATT&CK® ICS Matrix tactics and techniques and the specific language in the technique. Some techniques reference devices and processes in the OT environment, such as PLCs, controllers, process control, safety systems, and OT networks. This assessment did not expand upon these references. The MITRE ATT&CK® Enterprise Matrix has additional tactics and techniques that may be exploited to compromise a DLT. These additional tactics and techniques should be considered for future efforts.

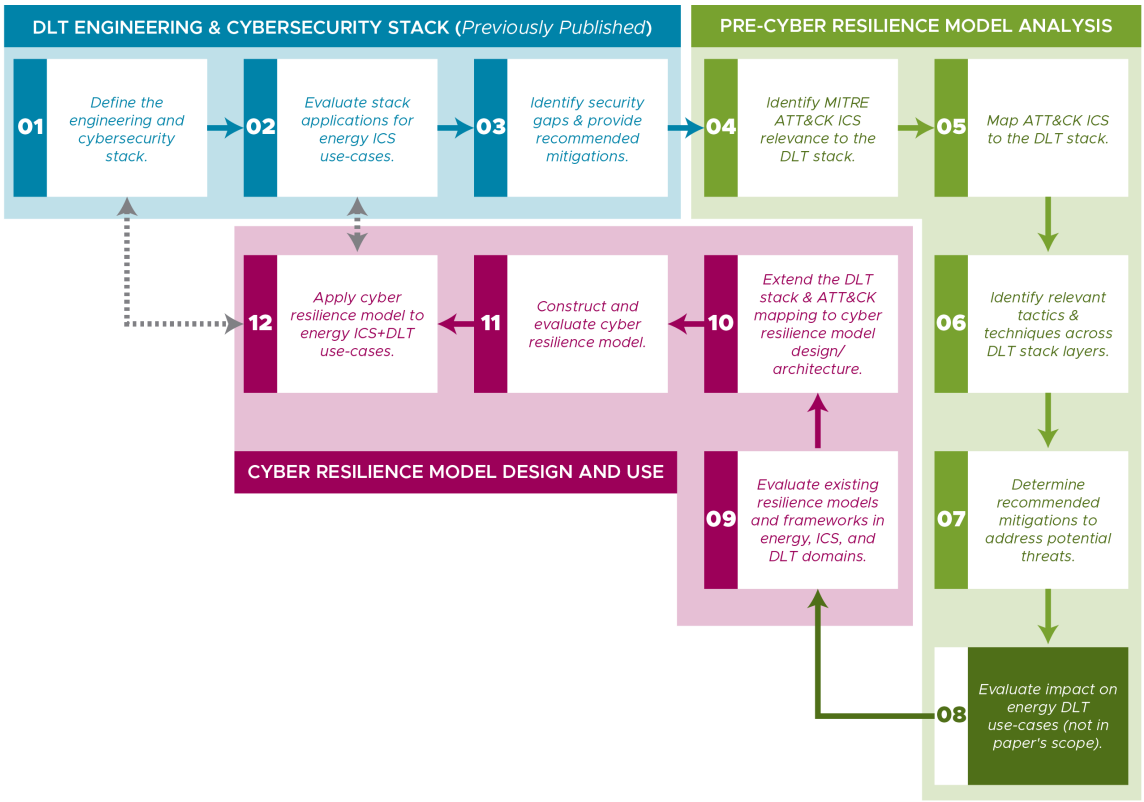

FIGURE 3. The Cyber Risk Management Process Flow, it illustrates how the team has structured the CRM flow across three staggered phases.

VIII. MAPPING METHODOLOGY

Each diagram below includes an ICS tactic and identifies whether an adversary may potentially use the tactic to compromise one (or more) layers in the DLT stack. If a tactic is selected, the applicable techniques are identified. Layers that cannot be directly compromised using the tactic are marked with an X. In addition, if a DLT stack layer is directly compromised, an adversary may access other DLT stack layers using the same techniques or other techniques within the tactic. This is defined as indirect access, as illustrated in the diagrams with a dotted line from the initial point of compromise. In addition, some of the techniques are applicable in specific use cases. These techniques in each figure are in black letters. In each of the below sub-sections, a table of mitigations is presented after the techniques of a tactic are mapped to the DLT stack layers. The mitigations are directly exported from MITRE's guidance. DLT-tailored guidance on operationalizing the mitigations is shown in the Appendix.

The tactics cannot be assessed independently of each other. Initially, an adversary must be successful at the Initial Access and Execution tactics. If an adversary cannot access a DLT stack layer and execute techniques, the other tactics and techniques are not applicable. Compromises using Persistence, Privilege Escalation, and Lateral Movement tactics may result in a more significant effect at the different DLT stack layers.

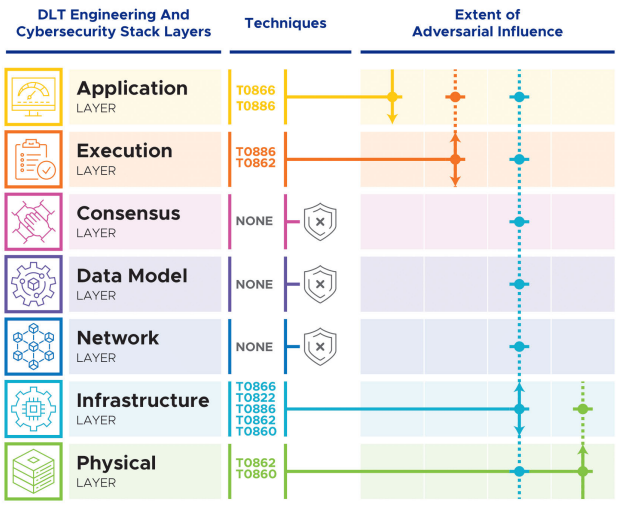

FIGURE 4. Initial access tactic analysis and mapping to the DLT stack.

A. INITIAL ACCESS

Initial Access consists of techniques that adversaries may use as entry vectors to gain an initial foothold within an ICS environment. Following is an analysis of the applicability of the Initial Access tactic to the DLT stack layers and the rationale. Figure 4 illustrates the analysis.

1) APPLICATION LAYER

This layer includes applications that trigger rule-bases and program code, APIs, UI/GUIs, Oracles, distributed applications, marketplace, monetization, etc. Because the application layer includes software and applications that interact with the DLT, this tactic applies. The specific techniques that may be used are: *Exploitation of Remote Services (T0866)* and *Remote Services (T0886)*. For most DLTs, these external applications only have limited connectivity to a DLT's components listed in *Assumption 1* above. Therefore, if any of the techniques listed under the Initial Access tactic are used to gain an initial foothold, other tactics or techniques may be used to compromise the DLT applications (e.g., corrupting the source code of the DLT application or DLT application patch through supply chain attack on libraries, etc.). Also, an adversary should be unable to modify the smart contract interacting with the application.

2) EXECUTION LAYER

This layer deals with rule-bases and program code, for example, smart contracts, chaincode, atomic swaps (exchanging cryptocurrencies between the cryptographic networks), tokens, etc. Because this layer contains the DLT rules and program logic, the Initial Access tactic is relevant at this layer; the adversary may use the following techniques to gain initial access:

- *Remote Services (T0886)* for accessing the code;
- An uncommon technique would be *Supply Chain Compromise (T0862)* for unauthorized access to code developed by vendors/third parties. Because the execution layer deals with the governing program logic/code that is integral to the DLT, gaining the initial foothold at this layer's components could allow the adversary to have an impact at this layer using other tactics or techniques and traverse to the application layer (indirect access);
- If a smart contract is compromised and a transaction is approved based on the compromised smart contract, the transaction would be treated as valid, and the respective block would also be considered valid. Therefore, this would potentially have an impact on the data model layer.

3) CONSENSUS, DATA MODEL, AND NETWORK LAYERS

Unlike the components at the Application and Execution layers, these three layers have process components such as the following:

1) Process or ability to form consensus;
2) Process or ability to create blocks, and
3) Process or ability to transact over the network.

In a DLT implementation, the network layer includes the underlying blockchain platform protocols, such as Hyperledger, Ethereum, etc. The network layer also includes the communication infrastructure that supports the DLT and facilitates transactions, information, and data sharing between the nodes. Although the organization-level network infrastructure, such as routing/switching, etc., is required to host DLT nodes, compromising the network infrastructure by exploiting network system vulnerabilities can be done with or without DLT. Therefore, direct compromise of network infrastructure to pivot into the DLT environment is out of the scope of this assessment. However, if an adversary compromises a DLT node and pivots into the network infrastructure, that will be in scope (covered under the Infrastructure layer). The Initial Access tactic does not apply to these three layers. If applicable, the primary concern would be a *Supply Chain Compromise (T0862)* technique for unauthorized modification of the protocols.

4) INFRASTRUCTURE LAYER

This layer deals with data storage entities, logical DLT nodes, virtual machines, clusters, and containers. An adversary may be able to gain initial access to the physical and/or virtual components (e.g., gain initial access to a virtual machine running the DLT node - transacting node, consensus node, or ordering node). The nodes should be capable of performing cryptographic operations, providing information for authentication and authorization by the network/other nodes, and configuring on-chain and off-chain storage. Depending on the compromised node(s) to which the threat actor gained initial access, the adversary may gain access to the other six layers of the DLT stack. The applicable techniques are: *Exploitation of Remote Services (T0866)*, *External Remote Services (T0822)*, *Remote Services (T0886)*, *Supply Chain Compromise (T0862)*, and *Wireless Compromise (T0860)*. Network-level hardening and host-level hardening are out of scope. Therefore, aspects of cybersecurity best practices such as network segmentation, segregation through firewall rules, Virtual Local Area Networks (VLANs), etc., are out of scope.

5) PHYSICAL LAYER

This layer pertains to the systems participating on behalf of the users and lower level ICS/OT systems such as sensors, IoT devices with Unique Identifiers (UIDs), Operating System (OS)/firmware, etc. Typically, sensors and IoT devices are external to the DLT and provide data to the DLT. If the sensors and IoT devices are external to the DLT, the Initial Access Tactic will not apply. However, if these devices are included in the DLT, techniques such as *Supply Chain Compromise (T0862)* and *Wireless Compromise (T0860)* may be used. The two techniques will allow an attacker to access the device. If a threat actor gains initial access to the physical layer, this may allow indirect access to the infrastructure layer. As described above, the other DLT stack layers may be accessed once the infrastructure layer is compromised.

6) MITIGATION

Table 1 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Initial Access tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 1. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Initial Access tactic.

Technique	Mitigation
Exploitation of Remote Services (T0866)	M0948 Application Isolation and Sandboxing M0926 Privileged Account Management
Remote Services (T0886)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication M0918 User Account Management
Supply Chain Compromise (T0862)	M0945 Code Signing M0817 Supply Chain Management
External Remote Services (T0822)	M0918 User Account
Management Wireless Compromise (T0860)	M0813 Software Process and Device Authentication

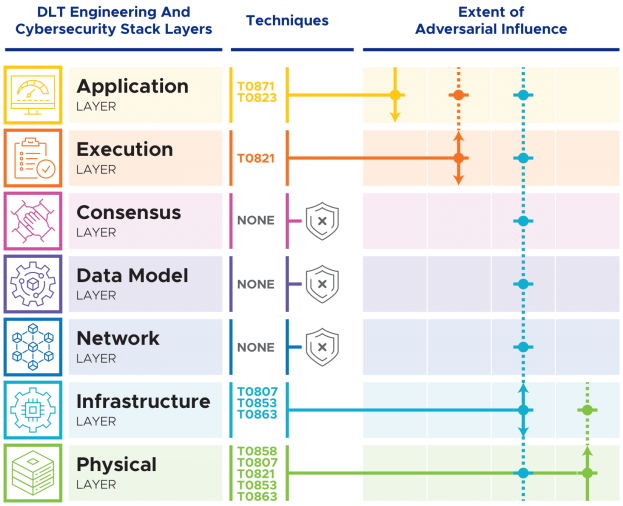

FIGURE 5. Execution tactic analysis and mapping to the DLT stack.

B. EXECUTION

The adversary is trying to run code or manipulate system functions, parameters, and data unauthorizedly. Following is an analysis of the applicability of the Execution tactic to the DLT stack layers and the rationale. Figure 5 illustrates this analysis.

1) APPLICATION LAYER

As the application layer contains APIs, the Execution tactic applies, particularly the *Execution through API (T0871)* and *Graphical User Interface (T0823)* techniques that would allow an adversary to alter the DLT API and UI/GUIs.

2) EXECUTION LAYER

One focus of the Execution tactic is to run adversary-controlled code. This malicious code may alter the way the DLT rule-bases and program codes operate. Therefore, the Execution tactic applies, specifically the *Modify Controller Tasking (T0821)* technique that will allow an adversary to modify the rules and code. The result could be the execution of malicious programs or the manipulation of rule-based execution flow.

3) CONSENSUS AND DATA MODEL LAYERS

The Execution tactic will not apply to these two layers as it focuses on running adversary-controlled code and maliciously manipulating parameters and data. Note that, the data model layer focuses on the model structure and associated parameters. However, any alteration of the consensus protocol should be quickly identified.

4) NETWORK LAYER

The Network layer focuses on communications and connectivity. The Execution tactic will not apply as it focuses on running adversary-controlled code and maliciously manipulating parameters and data. As mentioned, any attack that targets the network infrastructure directly instead of propagating through a DLT is out of scope, including attacks such as Distributed Denial-of-Service (DDoS), etc., that directly target the network infrastructure.

5) INFRASTRUCTURE LAYER

This layer comprises virtual and physical computers or software agents participating as authorized blockchain nodes. Because this layer also includes middleware and applicable tools and processes for the DLT, the Execution tactic applies. The specific techniques are: *Command-Line Interface (T0807)*, *Scripting (T0853)*, and *User Execution (T0863)*. The adversary's objective would be to maliciously alter the middleware/processes to ensure the DLT did not function correctly. An adversary may have indirect access to the other six DLT stack layers if the infrastructure layer is compromised. If an adversary moves/pivots into the network layer, the *Scripting (T0853)* and *User Execution (T0863)* techniques may be used in the network layer.

6) PHYSICAL LAYER

Typically, sensors and IoT devices are external to the DLT and provide data to the DLT. The Execution tactic focuses on modifying: controller operating mode, command line interfaces, APIs/GUIs, and controller tasking. Because these are outside the scope of the DLT physical layer, the Execution tactic does not apply. However, if these devices are included in the DLT, techniques such as *Change Operating Mode (T0858)*, *Command Line Interface (T0807)*, *Modify Controller Tasking (T0821)*, *Scripting (T0853)*, and *User Execution (T0863)* may be used. All five techniques will allow an attacker to access the device. If the physical layer is compromised, an adversary may have indirect access to the infrastructure layer.

7) MITIGATION

Table 2 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Execution

TABLE 2. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Execution tactic.

Technique	Mitigation
Execution through API (T0871)	M0801 Access Management M0800 Authorization Enforcement M0938 Execution Prevention M0804 Human User Authentication
Graphical User Interface (T0823)	M0816 Mitigation Limited or Not Effective
Modify Controller Tasking (T0821)	M0945 Code Signing
Command-Line Interface (T0807)	M0942 Disable or Remove Feature or Program
Scripting (T0853)	M0948 Application Isolation and Sandboxing
User Execution (T0863)	M0945 Code Signing M0938 Execution Prevention M0921 Restrict Web-Based Content
Change Operating Mode (T0858)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication

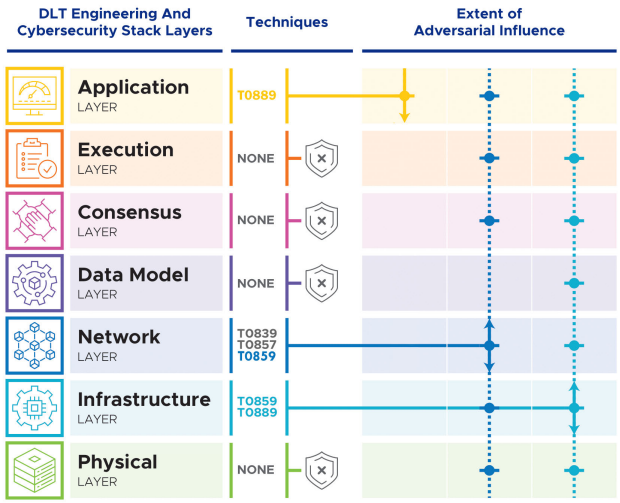

FIGURE 6. Persistence tactic analysis and mapping to the DLT stack.

tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

C. PERSISTENCE

The adversary is trying to maintain a foothold in the ICS environment. Following is an analysis of the applicability of the Persistence tactic to the DLT stack layers and the rationale. Figure 6 illustrates this analysis.

1) APPLICATION LAYER

The Persistence tactic may be applicable if the API under the application layer enables users to modify either 1) application-specific settings or 2) DLT platform-specific settings. The applicable technique is *Modify Program (T0889)*. For example, if a grid-based application allows users to store grid topology data, deliberate errors could cause dependent applications to fail, yield incorrect results, or help the attacker masquerade future attacks. A second example may be applicable if the API allows remote configuration of the underlying physical/network infrastructure (such as DLT node properties). Mitigation mechanisms include performing sanity checks, validations to ensure bad data is not transmitted, and ongoing checks to validate/flag previously stored data.

2) EXECUTION LAYER

This layer deals with Rule-bases and program code. Examples include smart contracts, chaincode, atomic swaps, tokens, etc. Therefore, the Persistence tactic does not apply as the associated techniques target and focus on controllers and PLCs.

3) CONSENSUS LAYER

The consensus layer includes the decision rules in the DLT system. Within permissioned networks, the credential management system plays a key role in enabling consensus. Typically, the credential management system is outside the scope of the DLT system. Therefore, the Persistence tactic does not apply.

4) DATA MODEL LAYER

Direct influences to synchronization, services, block creation, chain structure, or hashing are unrealistic using privilege escalation. Based on the current knowledge and known architectures, this layer cannot be reached directly using Persistence techniques.

5) NETWORK LAYER

Based on current knowledge and known architectures, this layer can be compromised directly using the following Persistence techniques. An adversary may indirectly access the other DLT stack layers if the network layer is compromised.

- *Valid Accounts (T0859)*: Within a permissioned DLT environment, peer participation, and thus access permissions, are tied to a credential management system. Since such a system is a key component, it must be engineered to:
  1) Prevent unauthorized systems from obtaining valid credentials;
  2) Identify and handle credential theft (from the server side);
  3) Limit the risks and scenarios in which the credential management system can restrict or limit valid transactions.

  In addition, peers must be diligent in preventing and reporting credential theft. This could be done proactively (e.g., periodic credential renewals) or actively (e.g., analyzing ledger operations vs. submitted transactions).

Although the following techniques cannot be directly exploited/used through the DLT, exploits can be used to disrupt DLT operations

- *Module Firmware (T0839)*: Adversaries may install malicious or vulnerable firmware onto network hardware devices;
- *System Firmware (T0857)*: An attacker may install malicious or out-of-date firmware that could be used to disrupt the DLT.

6) INFRASTRUCTURE LAYER

Data storage entities and logical blockchain nodes may include traditional virtual machines, virtual containers (e.g., Kubernetes), and cloud-managed infrastructures. The Persistence tactic is applicable. If the infrastructure Layer is compromised, an adversary may have indirect access to the other DLT stack Layers. Similar to the Consensus layer, the following technique should be considered in the system's design.

- *Valid Accounts (T0859)*: The infrastructure Layer can be compromised by abusing or exploiting the management accounts. This could lead to the exploitation of additional tactics, such as a system stop, data extraction, or rootkit installation, which may lead to a full system compromise. Therefore, infrastructure-related accounts must be protected using industry best practices, such as multi-factor authentication, periodic password changes, and least privilege approaches;
- *Modify Program (T0889)*: The infrastructure Layer can be compromised by modifying a program that manages the DLT infrastructure.

7) PHYSICAL LAYER

This Layer pertains to the systems participating on behalf of the users and lower level ICS/OT systems such as sensors, IoT devices with UIDs, OS/firmware, etc. Although a wide variety of persistence techniques can be applied, these are generic and do not necessarily reflect the vulnerabilities introduced by DLT platforms.

8) MITIGATION

Table 3 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Persistence tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

D. PRIVILEGE ESCALATION

The adversary is trying to gain higher-level permissions. Regarding DLT systems, privilege escalation [32] may not significantly affect unless the adversary can gain higher-level permissions to most DLT nodes. This will also depend on the specific DLT architecture. Following is an analysis of the applicability of the Privilege Escalation tactic to the DLT stack layers and the rationale. Figure 7 illustrates the analysis.

1) APPLICATION LAYER

An adversary successfully exploiting the privilege escalation tactic may use other tactics and techniques to compromise

TABLE 3. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Persistence tactic.

Technique	Mitigation
Execution through API (T0871)	M0801 Access Management M0800 Authorization Enforcement M0938 Execution Prevention M0804 Human User Authentication
Graphical User Interface (T0823)	M0816 Mitigation Limited or Not Effective
Modify Controller Tasking (T0821)	M0945 Code Signing
Command-Line Interface (T0807)	M0942 Disable or Remove Feature or Program
Scripting (T0853)	M0948 Application Isolation and Sandboxing
User Execution (T0863)	M0945 Code Signing M0938 Execution Prevention M0921 Restrict Web-Based Content
Change Operating Mode (T0858)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication

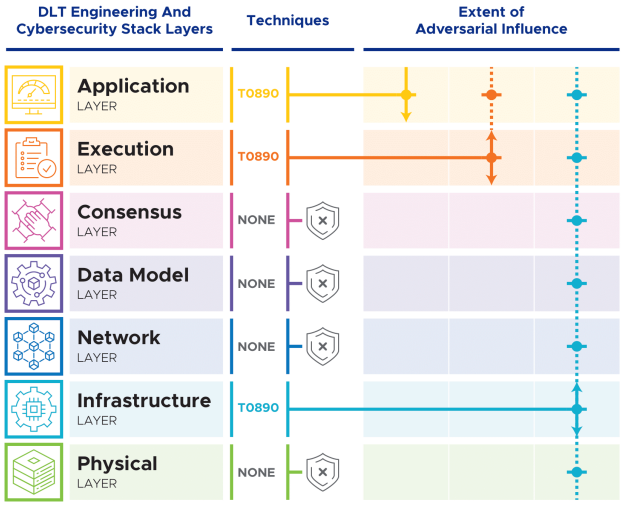

FIGURE 7. Privilege Escalation tactic analysis and mapping to the DLT stack.

the DLT application layer nodes. The applicable technique is *Exploitation for Privilege Escalation (T0890)*.

Although *Hooking (T0874)* may not be applicable in this layer, the risk of an attacker exploiting a low-level API management call to gain access to underlying levels is a possibility. For example, DLT Software Development Kits (SDKs) may expose configurable endpoints to define execution or consensus options that may be abused to inject malicious hooks. This vulnerability should be identified and addressed in the secure system design.

2) EXECUTION LAYER

This layer deals with Rule-bases and program code. Examples include smart contracts, chaincode, atomic swaps, tokens, etc. Smart contracts' vulnerabilities and errors

(intentional or accidental) can lead to privilege escalation. Compromising or usurpation of tokens can cause malfunctions in the DLT system. The applicable technique is Exploitation for *Privilege Escalation (T0890)* within an application user space context.

3) CONSENSUS LAYER

The consensus layer includes the decision rules in the DLT system. Since there are consensus algorithms (e.g., PoS and PoA) that are based on a node's reputation, this tactic is applicable. An attacker who has gained access to a node using the Initial Access tactic can behave like an honest network member, waiting for the opportune moment after having accumulated enough reputation to try to influence the result of the decision in a particular situation. However, the current version of the ICS matrix does not have a directly applicable technique but could be considered an extension of *Exploitation of Privilege Escalation (T0890)*.

4) DATA MODEL LAYER

Direct influences on synchronization, services, block creation, chain structure, or hashing are unrealistic using Privilege Escalation techniques.

5) NETWORK LAYER

The Network Layer cannot be reached directly using Privilege Escalation techniques.

6) INFRASTRUCTURE LAYER

This is the most critical layer. Once an adversary successfully implements the Initial Access tactic and obtains access to a DLT node, the Privilege Escalation tactic may be used to gain control over the data storage entity or local blockchain node. After the other DLT stack layers are compromised using the Initial Access tactic, an adversary may exploit software vulnerabilities to elevate privileges. The level of compromise will depend on the security architecture. The applicable technique is Exploitation for *Privilege Escalation (T0890)*. The applicability is by compromising software systems in the Application and Execution layers through the compromised infrastructure. Once the infrastructure layer is compromised, an adversary may have indirect access to the other six DLT stack layers.

7) PHYSICAL LAYER

Privilege escalation on this layer in DLT systems is not applicable because of the distributed structure. Alternatively, the physical layer could be indirectly accessed through the infrastructure layer.

8) MITIGATION

Table 4 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Privilege Escalation tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 4. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the privilege escalation tactic.

Technique	Mitigation
Exploitation for Privilege Escalation (T0890)	M0948 Application Isolation and Sandboxing

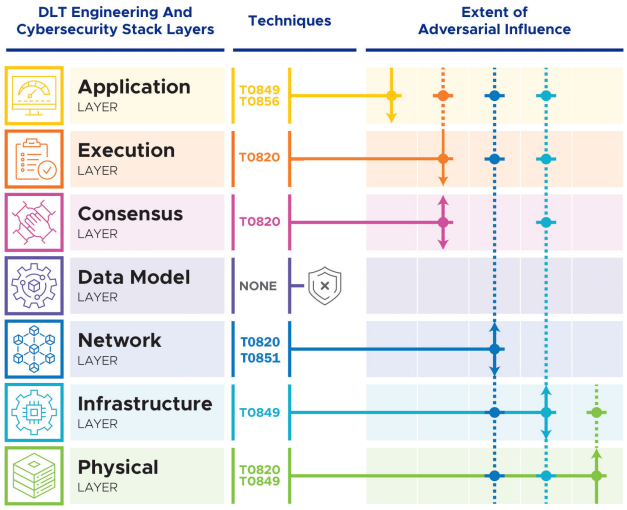

FIGURE 8. Evasion tactic analysis and mapping to the DLT stack.

E. EVASION

An adversary uses evasion techniques to avoid security controls. Evasion's techniques can only be implemented after an adversary has reached a DLT node and its specific functionality. This tactic is applicable only at three layers: application, infrastructure, and physical. Figure 8 illustrates the analysis.

1) APPLICATION LAYER

An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. The applicable techniques are:

- *Masquerading (T0849)*: An adversary may use masquerading to disguise a malicious application or executable as another file, which can be achieved by duplicating an API front end, leading to deceit or impersonation of a given state;
- *Spoof Reporting Message (T0856)*: An adversary could spoof the interactions between the Application layer components and the database that the application may be interacting with. Furthermore, the same technique can be used to fake a DLT system's health system/messages.

2) EXECUTION LAYER

The only applicable technique is *Exploitation for Evasion (T0820)*, which may be used to exploit potential program bugs/errors in the smart contract code. Such exploitation may require the adversary to gain access to the infrastructure layer to access the smart contract program.

3) CONSENSUS LAYER

- The only applicable technique is *Exploitation for Evasion (T0820)*, which may be applied to this layer if the adversary could compromise inherent vulnerabilities of the consensus mechanism itself;
- Evasion occurs when the consensus-forming nodes are compromised and form a consensus that is against the rules defined in the smart contract. If this happens, the applications and smart contracts will not have any context around what happened in the consensus layer. The adversary could execute an attack by targeting the consensus nodes.

4) DATA MODEL LAYER

The Data Model Layer cannot be compromised using the Evasion tactic. The nature of DLT systems explains this. Every adversary's attempt to use evasion techniques on this layer should be identified. However, these layers may be compromised through the infrastructure layer as indirect access.

5) NETWORK LAYER

The following technique can be applied:

- *Exploitation for Evasion (T0820)*: By taking advantage of programming errors and erroneously deployed configurations, an attacker may perform exploitations that affect or compromise a DLT's ability to operate;
- *Rootkit (T0851)*: Attackers may inject malicious code to disrupt a DLT's ability to operate, potentially leading to service outages and/or the application of other tactics.

Note that impacts can be observed at higher levels once the network infrastructure is compromised due to disrupted network traffic patterns. For example, smart contracts deployed within the execution layer could experience communication timeouts, while API end-points may experience demand surges due to replay attacks or DoS attacks. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

6) INFRASTRUCTURE LAYER

An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

An adversary may use masquerading to disguise a malicious application or executable as another file. The applicable technique is *Masquerading (T0849)*. A supervisory agent may be deceived by impersonating services, and/or devices, leading to an incorrect system evaluation.

7) PHYSICAL LAYER

An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. The following techniques may be applied:

- *Masquerading (T0849)*: an adversary may disguise a malicious application or executable as another file;

TABLE 5. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the evasion tactic.

Technique	Mitigation
Exploitation for Evasion (T0820)	M0948 Application Isolation and Sandboxing
Masquerading (T0849)	M0945 Code Signing M0938 Execution Prevention
Spoof Reporting Message (T0856)	M0802 Communication Authenticity
Rootkit (T0851)	M0945 Code Signing

- *Exploitation for Evasion (T0820)*: this is applicable if there are programming errors and erroneously deployed configurations. An attacker could perform exploitations that affect or compromise DLT's operating ability.

Change Operating Mode (T0858) and *Spoof Reporting Message (T0856)* are valid techniques that could be implemented at the controller level. However, they do not have a direct relationship to the DLT or its use in the OT/ICS network. The adversarial compromise and use of these techniques are not specific to the DLT and can occur independently of the deployment of a DLT.

8) MITIGATION

Table 5 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Evasion tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

F. DISCOVERY

The Discovery tactic includes techniques that adversaries may use to survey (recognize) the ICS environment and gain knowledge about the internal network, control system devices, and how the processes interact. The specific techniques help adversaries observe the environment and determine the next steps for subsequent target selection and lateral movement. The techniques allow adversaries to explore what they can control and gain insight into interactions between various control system processes [11]. Figure 9 illustrates the analysis.

1) APPLICATION LAYER

The tactic is not applicable at the application layer.

2) EXECUTION LAYER

Rule-bases and program code. Examples: smart contracts, chaincode, atomic swaps, tokens, etc. The tactic is not applicable at this layer.

3) CONSENSUS LAYER

Consensus protocols include lottery-based, voting-based, etc. The tactic is not applicable at this layer.

4) DATA MODEL LAYER

Data (and time) synchronization. Ordering services, block creation, chain structure, hashing, etc. The tactic is not applicable at this layer.

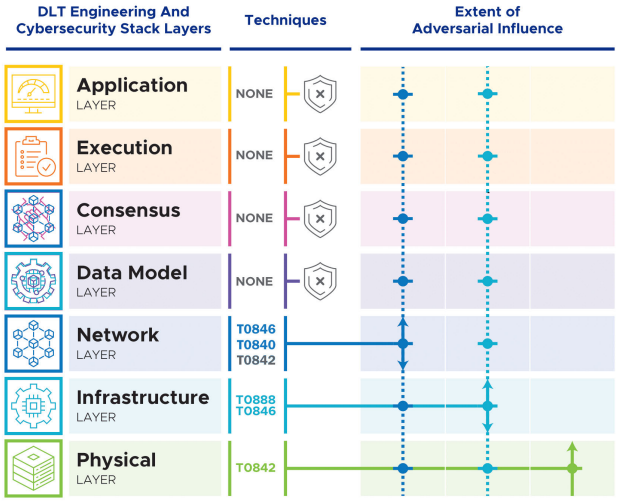

FIGURE 9. Discovery tactic analysis and mapping to the DLT stack.

5) NETWORK LAYER

Peer-to-peer transaction broadcast/discovery, including connectivity, runtime, telecommunications, and network parameters. The following techniques apply in the network layer:

- *Network Connection Enumeration (T0840)*: This technique may be used to acquire connection profiles used by clients to connect to the DLT and the neighbor peers' connection addresses to perform targeted network disruptions;
- *Remote System Discovery (T0846)*: This technique may be used to acquire the neighbor peers' connection addresses that may be used with other techniques;
- *Network Sniffing (T0842)*: From a local node, local network visibility can be attained to launch eclipse-like attacks. Network sniffing may be possible depending on the network topology, particularly when traffic crosses public/non-trusted networks or unprotected devices. Some of the network sniffings would be outside the scope of the DLT.

Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

6) INFRASTRUCTURE LAYER

Data storage entities. Logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The following techniques are applicable.

- *Remote System Information Discovery (T0888)*: This technique may be used to acquire connection profiles used to connect to the DLT and to discover the oracle or off-chain database used by the smart contract to make decisions;
- *Remote System Discovery (T0846)*: This technique may acquire the neighbor peers' connection addresses to perform targeted network disruptions.

Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

TABLE 6. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the discovery tactic.

Technique	Mitigation
Network Connection Enumeration (T0840)	M0816 Mitigation Limited of Not Effective
Remote System Discovery (T0846)	M0814 Static Network Configuration
Remote System Information Discovery (T0888)	M0814 Static Network Configuration
Network Sniffing (T0842)	M0808 Encrypt Network Traffic M0930 Network Segmentation M0926 Privileged Account Management M0814 Static Network Configuration

7) PHYSICAL LAYER

Systems participating on behalf of the users. Examples: sensors, IoT devices with UID, OS, etc. The tactic is not directly applicable to this DLT stack layer. However, depending on the OS and the available system libraries, then network sniffing may be possible at a local level (via *T0842*). If running a DLT node, they may learn more about the application.

8) MITIGATION

Table 6 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Discovery tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

G. LATERAL MOVEMENT

In the Lateral Movement tactic, the adversary tries to move through the ICS environment. In this assessment, the focus is on movement through the various DLT stack layers. Lateral movement can only be successful after the Initial Access and Execution tactics are effective against a specific DLT stack layer, as described above. If the Privilege Escalation tactic is successful, the effect of the Lateral Movement tactic may be more significant. Figure 10 illustrates the analysis.

1) APPLICATION LAYER

An attacker may exploit vulnerabilities in software installed on DLT nodes or virtual machines, clusters, or Kubernetes. The only applicable technique is the *Valid Accounts (T0859)*. In this case, compromised credentials may be used to bypass access controls placed on the API layer and may be used to maintain persistent access to remote systems. Compromised and default credentials may also grant an adversary increased privilege to application-specific functionalities.

2) EXECUTION LAYER

Rule-bases and program code. The Lateral Movement tactic is applicable, including the *Remote Services (T0886)* and *Exploitation of Remote Services (T0866)* techniques that are applicable at the execution layer, which could result in application-level compromises if a vulnerable piece of code

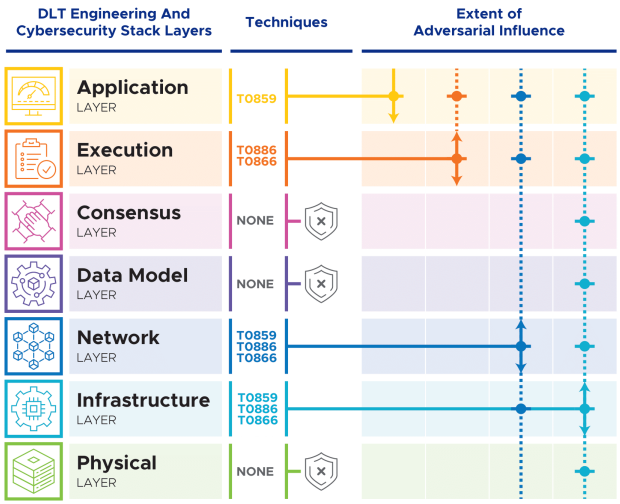

FIGURE 10. Lateral Movement tactic analysis and mapping to the DLT stack.

within a smart contract or a library *node.js* library was used to write a smart contract. These techniques may allow an adversary to move between DLT components.

3) CONSENSUS LAYER AND DATA MODEL LAYER

Lateral Movement techniques will not be effective at these layers because any attacker's attempt can be easily traced. This is achieved through the distributed nature of the network and the use of cryptography.

4) NETWORK LAYER

The Lateral Movement tactic is applicable, including the following techniques. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Valid Accounts (T0859)*: An adversary may steal the credentials of a specific user or account. Compromised credentials may be used to bypass access controls;
- *Remote Services (T0886)*: This technique may be used to move between DLT assets and network components;
- *Exploitation of Remote Services (T0866)*: This may allow an adversary lateral movement between the DLT components.

5) INFRASTRUCTURE LAYER

If attacker software was installed on the validator node (and this node did not establish the fourth level of authentication, i.e., storing private keys on a secure hardware token), an attacker could gain access to the validator node's private key by reading files/while processing the private key in the device memory. Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Valid Accounts (T0859)*: An adversary may steal the credentials of a specific user or account. Compromised credentials may be used to bypass access controls;

TABLE 7. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the lateral movement tactic.

Technique	Mitigation
Remote Services (T0886)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0813 Software Process and Device Authentication M0918 User Account Management
Exploitation of Remote Services (T0866)	M0942 Disable or Remove Feature or Program M0926 Privileged Account Management
Valid Accounts (T0859)	M0801 Access Management M0926 Privileged Account Management M0918 User Account Management

- *Remote Services (T0886)*: This technique may be used to move between DLT assets;
- *Exploitation of Remote Services (T0866)*: This may allow an adversary lateral movement between the DLT components.

6) PHYSICAL LAYER

Purdue Level 0 systems (controllers, PLC, safety systems) should not be directly connected or part of a DLT; therefore, techniques such as *Program Download (T0843)* do not apply. Lateral Movement techniques should not be effective at these layers because any attacker's attempt can be easily traced. This is achieved through the distributed nature of the network and the use of cryptography.

7) MITIGATION

Table 7 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Lateral Movement tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

H. COLLECTION

The Collection tactic consists of techniques that allow attackers to collect information that would later be used for illegal purposes. The adversary tries to collect data of interest and information about the domain, thus obtaining feedback on the ICS environment. Figure 11 illustrates the analysis.

1) APPLICATION LAYER

- Applications that trigger rule-bases and program code, such as APIs, UI/GUIs, Oracles, distributed applications, marketplace and monetization, exchange, etc. The Collection tactic does not apply at this DLT stack layer;
- An adversary could compromise APIs to collect information on the DLT and about the DLT network of peers. Currently, the techniques included in the Collection tactic do not apply. There may be other techniques not included in the MITRE ATT&CK® ICS Matrix that could be used.

2) EXECUTION LAYER

This layer deals with Rule-bases and program code. Examples include Smart Contracts, Chaincode, atomic swaps,

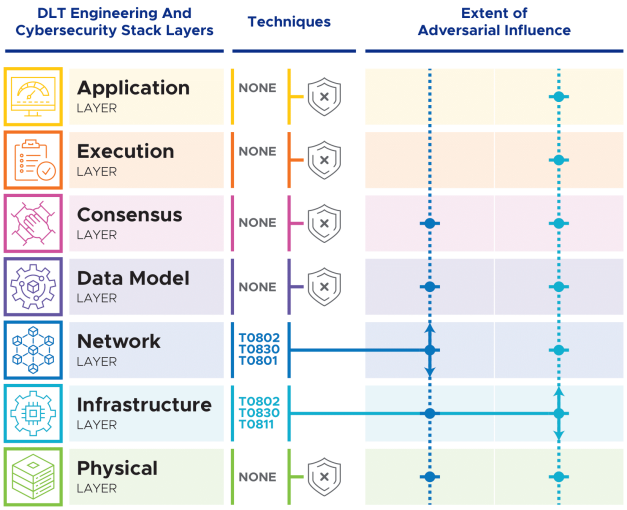

FIGURE 11. Collection tactic analysis and mapping to the DLT stack.

Tokens, etc. The Collection tactic does not apply at this DLT stack layer. An adversary could gain access to the smart contract and reverse engineer the functions, etc., to enumerate inner workings. Currently, the techniques included in the Collection tactic do not apply. Other techniques not included in the MITRE ATT&CK® ICS Matrix could be used.

3) CONSENSUS LAYER

Consensus protocols include proof-based, voting-based, etc. The Collection tactic does not apply at this DLT stack layer.

4) DATA MODEL LAYER

Data (and time) synchronization, ordering services, block creation, chain structure, hashing, etc. The Collection tactic applies and the following is the *Automated Collection (T0802)* technique that may be used. The adversary could use data collection strategies in network traffic or script execution automatically in the DLT environment. This may allow an adversary to more accurately determine the structure of the data model layer.

5) NETWORK LAYER

Peer-to-peer transaction broadcast/discovery. Connectivity, runtime, telecommunications, and network parameters. The Collection tactic applies, and the following techniques may be used. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Automated Collection (T0802)*: The adversary could use data collection strategies in network traffic or script execution automatically in the control environment. This may allow an adversary to identify the DLT network architecture;
- *Adversary-in-the-Middle (T0830)*: The adversary with certain privileges could alter the network traffic in search of benefiting their illegal activities. Specifically, an adversary could intercept traffic associated with the various DLT nodes and potentially block or modify the traffic;
- *Monitor Process State (T0801)*: The adversary may use the information of the processes to cancel or allow actions that help him in his fraud purposes. Once this layer is compromised, this technique will allow an adversary to map the DLT more accurately.

Compromising the network infrastructure will allow an adversary to collect data-in-transit, such as the communication between consensus-forming nodes and the block-creation nodes. Similarly, the adversary would be able to intercept the traffic between nodes. The knowledge obtained from this collection process could then be used to launch targeted attacks.

6) INFRASTRUCTURE LAYER

Data storage entities. Logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The Collection tactic applies, and the following techniques may be used. Once the infrastructure layer is compromised, an adversary may have indirect access to the other DLT stack layers.

- *Automated Collection (T0802)*: The adversary could use data collection strategies in network traffic or script execution automatically in the control environment. This may allow an adversary to identify the DLT network architecture;
- *Adversary-in-the-Middle (T0830)*: The adversary with certain privileges could alter the network traffic in search of benefiting their illegal activities. Specifically, an adversary could intercept traffic associated with the various DLT nodes and potentially block or modify the traffic;
- *Data from Information Repositories (T0811)*: The adversary seeks to obtain available information about the system parts, whether in parts of used codes, version history, or any information about system or environment components. An attacker may gather information about the DLT off-chain and on-chain storage devices, including the content and structure.

7) PHYSICAL LAYER

System participating on behalf of the users, for example, Sensors, IoT devices with UID, OS, etc. Traditional DLT systems do not have a physical layer in themselves.

8) MITIGATION

Table 8 provides a list of suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Collection tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

I. COMMAND AND CONTROL

Command and Control techniques are typically used to communicate between affected parts of the system. In the case

TABLE 8. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the collection tactic.

Technique	Mitigation
Automated Collection (T0802)	M0807 Network Allowlists
Adversary-in-the-Middle (T0830)	M0802 Communication Authenticity M0813 Software Process and Device Authentication
Monitor Process State (T0801)	M0816 Mitigation Limited or Not Effective
Data from Information Repositories (T0811)	M0926 Privileged Account Management M0922 Restrict File and Directory Permissions M0918 User Account Management

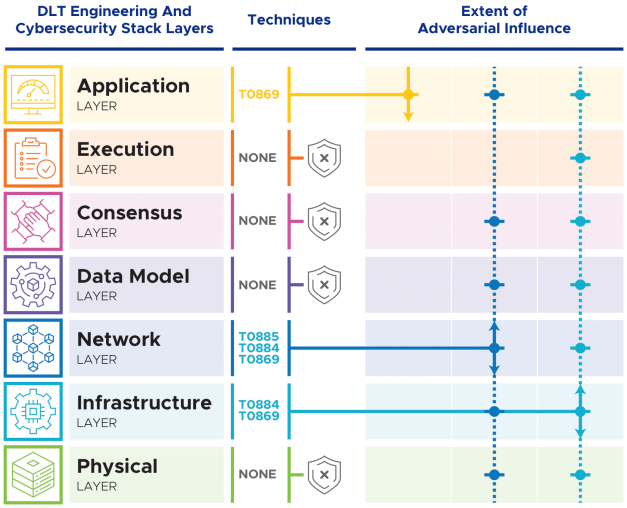

FIGURE 12. Command and control tactic analysis and mapping to the DLT stack.

of DLT systems, this might have a place in Network and Infrastructure layers. Figure 12 illustrates the analysis.

1) APPLICATION LAYER

The Command and Control tactic does not directly apply at this layer. However, The *Standard Application Layer Protocol (T0869)* technique may indirectly apply if the Infrastructure and/or Network layers are compromised.

- The application layer components, such as the APIs, software, etc., may be programmed to initiate network connections over protocols, such as Hypertext Transfer Protocol Secure (HTTPS), etc., to reach outside the network. The equipment in the network and infrastructure layers facilitate such connections. Therefore, if an adversary aims to compromise communications over protocols such as HTTPS, etc., the adversary would have to go through the network and infrastructure layers. This can then affect the application itself because the application may accept an adversarial connection as a legitimate connection;
- This technique may be indirectly applicable by disguising non-DLT traffic as DLT legitimate traffic. This would enable an attacker to control/disrupt services without supervisory agents. For example, protocols such as gRPC may be used to encode various API calls ranging from transaction submission to peer-level management. This would allow supervisory agents to differentiate using simple network protection rules.

Special attention should be devoted to systems that rely on common or shared endpoints to manage or operate the system. For example, suppose a given port is used to configure, monitor, and propose to commit transactions. In that case, then deep-packet inspection tools and firewalls may need to be deployed to ensure these endpoints do not process low-level API calls from untrusted systems.

2) EXECUTION, CONSENSUS, AND DATA MODEL LAYER

The adversary cannot initiate a Command and Control tactic attack in these layers. Therefore, the Command and Control tactic does not apply.

3) NETWORK LAYER

Peer-to-peer transaction broadcast or discovery, including connectivity, runtime, telecommunications, and network parameters. Using a secure local network environment, an adversary can try to reach other points or devices in the network. Once the network layer is compromised, an adversary may have indirect access to the other DLT stack layers. The Command and Control tactic applies, and the following techniques may be used.

- *Commonly Used Port (T0885)*: an adversary may communicate over a commonly used port to blend in with normal network activity associated with the DLT to execute attacks such as brute force, DDoS, port denial, etc;
- *Connection Proxy (T0884)*: an adversary may use a connection proxy to direct the DLT traffic among the various devices and alter the DLT communications;
- *Standard Application Layer Protocol (T0869)*: this technique directly applies. An adversary could gain logical or physical port-level access to the network infrastructure through a compromised DLT node and/or DLT application.

4) INFRASTRUCTURE LAYER

Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. DLT nodes running on local or virtual machines or clusters might be subject to standard attacks. The attacks at the infrastructure layer can enable indirect access to other layers in the DLT stack. The Command and Control tactic applies, and the following techniques may be used.

- *Connection Proxy (T0884)*: the adversaries may use a connection proxy to redirect the DLT traffic among the various devices and alter the DLT communications. The impact may be more significant because an attack at this DLT stack layer will allow indirect access to the other

TABLE 9. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the command and control tactic.

Technique	Mitigation
Standard App Layer Protocol (T0869)	M0807 Network Allowlists
Commonly Used Port (T0885)	M0942 Disable or Remove Feature or Program M0804 Human User Authentication
Connection Proxy (T0884)	M0937 Filter Network Traffic M0807 Network Allowlists

DLT stack layers. For example, by modifying a Virtual Machine (VM) or Docker image launch parameters, an attacker can set up additional ports that may later be used to take control of the infrastructure layer;

- *Standard Application Layer Protocol (T0869)*: an adversary could gain logical or physical port level access to the network infrastructure through a compromised DLT node and/or DLT application.

5) PHYSICAL LAYER

The adversary cannot initiate a Command and Control tactic attack in this layer. Therefore, the Command and Control tactic does not apply because of the limited effects of compromising a single unit or field device.

6) MITIGATION

Table 9 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Command and Control tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

J. INHIBIT RESPONSE FUNCTION

Adversaries use this tactic to circumvent a system's health monitoring or inherent self-protection mechanisms. This may result in a system losing its safety guarantees or preventing human (or automated) monitors from performing service and quality evaluations. Depending on the systems involvement, these attacks could result in human loss of life, equipment destruction, and production interruption. Figure 13 illustrates the analysis.

1) APPLICATION LAYER

The Inhibit Response tactic applies at this DLT stack layer, and the following techniques may be used.

- *Alarm Suppression (T0878)*: tools that are used to monitor the DLT health, such as Hyperledger Caliper and Eth-netstats, rely on DLT-provided logs and endpoints to enable end users to interpret DLT operations. These logs do not always inherit the same protection mechanisms in a DLT; therefore, mechanisms that ensure valid data are presented to the supervisor agent must be devised;
- *Modify Alarm Settings (T0838)*: similarly to the alarm suppression attack, mechanisms to ensure that correct logs are being used and tools to interpret them can be

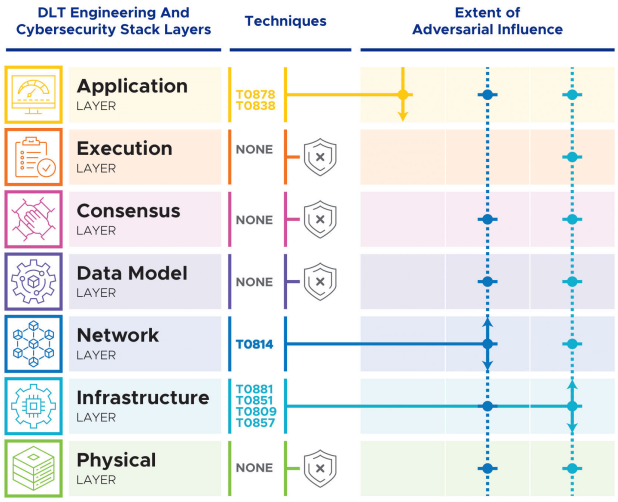

FIGURE 13. Inhibit response function tactic analysis and mapping to the DLT stack.

trusted must be in place to prevent the end user from receiving erroneous alarms.

2) EXECUTION LAYER

Rule-bases and program code. Examples: smart contracts, chaincode, atomic swaps, tokens, etc. The tactic is not applicable at this DLT stack layer.

3) CONSENSUS LAYER

Consensus protocols: proof-based, voting-based, etc. This layer would play a role, but other layers would reflect the impact. This layer should be considered a ''transient'' layer where attacks will go through. For example, a DoS or data destruction attack will require some knowledge of this layer to be successful, but it is not the primary target or entry point.

4) DATA MODEL LAYER

Data (and time) synchronization. Ordering services, block creation, chain structure, hashing, etc. The tactic is not applicable at this DLT stack layer.

5) NETWORK LAYER

Peer-to-peer transaction broadcast/discovery. Connectivity, runtime, telecommunications, and network parameters. The attacks at the network layer can enable indirect access to other layers in the DLT stack. The only applicable technique is the *Denial of Service (T0814)*. An adversary may perform a DoS attack to disrupt the operation of the DLT (e.g., at the network switch to affect DLT nodes).

6) INFRASTRUCTURE LAYER

Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The Inhibit Response tactic applies at this DLT stack layer, and the following techniques may be used. The attacks that happen at the

TABLE 10. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the inhibit response function tactic.

Technique	Mitigation
Alarm Suppression (T0878)	M0807 Network Allowlists M0814 Static Network Configuration
Modify Alarm Settings (T0838)	M0801 Access Management M0800 Authorization Enforcement M0804 Human User Authentication M0807 Network Allowlists M0813 Software Process and Device Authentication M0918 User Account Management
Denial of Service (T0814)	M0815 Watchdog Timers
Service Stop (T0881)	M0922 Restrict File and Directory Permissions M0918 User Account Management
Data Destruction (T0809)	M0926 Privileged Account Management M0922 Restrict File and Directory Permissions
Rootkit (T0851)	M0945 Code Signing
System Firmware (T0857)	M0801 Access Management M0945 Code Signing M0937 Filter Network Traffic M0804 Human User Authentication M0813 Software Process and Device Authentication

infrastructure layer can enable indirect access to other layers in the DLT stack.

- *Service Stop (T0881)*: If attackers gain access to the underlying infrastructure hosting the DLT nodes, there exists a risk of having a service/stop the attack. Host-level diversity and distinct host networks may limit the ability of an attacker to succeed (e.g., use a mixture of cloud vendors, operating systems, etc.);
- *Data Destruction (T0809)*: An adversary may perform data destruction to disrupt the DLT from reporting potential adversarial actions, such as data modification. This technique may then be implemented through the infrastructure layer at the other DLT stack layers;
- *Rootkit (T0851)*: Attackers may inject malicious code to disrupt a DLT's ability to prevent expected alarms and response mechanisms;
- *System Firmware (T0857)*: An attacker may install malicious or out-of-date firmware that could be used to disrupt the DLT.

7) PHYSICAL LAYER

System participating on behalf of the users. Examples: sensors, IoT devices with UID, OS, etc. The tactic is not applicable at this DLT stack layer because traditional DLT systems do not have a physical layer.

8) MITIGATION

Table 10 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Inhibit Response Function tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

K. IMPAIR PROCESS CONTROL

Field devices rely on API calls to reach the DLT infrastructure (e.g., via a web service). The execution layer (e.g., the smart contract) may rely on the application layer to access external oracles or off-chain storage databases. Figure 14 illustrates the analysis.

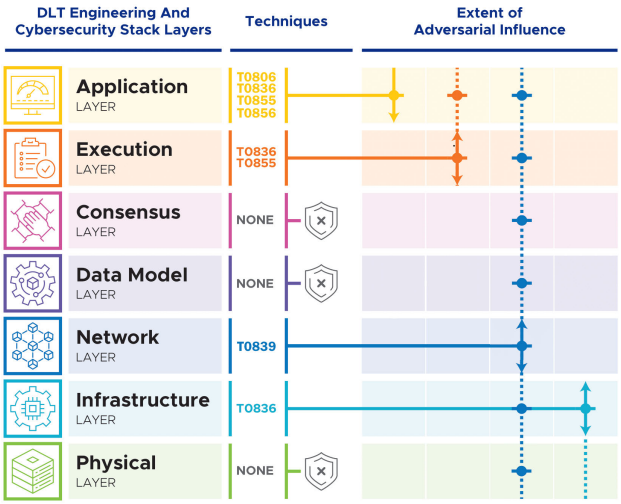

FIGURE 14. Impair process control tactic analysis and mapping to the DLT stack.

1) APPLICATION LAYER

The Impair Process Control tactic applies at this DLT stack layer, and the following are the techniques that may be used:

- *Brute Force I/O (T0806)*: a brute force attack may base its functionality on issuing malformed or random commands. Such attacks will likely affect API entry points, reducing their ability to serve legitimate traffic. To reduce the risk, APIs should be protected by brute force detection tools and gracefully handling malformed requests;
- *Modify Parameter (T0836)*: similar to the Brute Force I/O attack, service APIs may be vulnerable to maliciously crafted API calls (e.g., via a web service). Therefore, mechanisms to authenticate and secure a request's validity should be deployed (e.g., to avoid replay or modification);
- *Unauthorized Command Message (T0855)*: this technique should be considered in the design of the system. API entry points must also check the identity or role of an agent submitting an API request to reduce the risk of bypassing internal control mechanisms, for example, in a system with 1) a measurement device, 2) a control algorithm, and 3) an actuator. The measurement device should not be allowed to issue control commands to the actuator;
- *Spoof Reporting Message (T0856)*: a DLT can secure the integrity of a message once the transaction hits the consensus layer. This implies that DLT's API entry points must be capable of serving as gatekeepers and detecting message spoofing (e.g., via nonces and signatures). Failure to do so may result in incorrect messages reaching

the consensus layer, thus leading to incorrect states at the receiver end. APIs used to access external oracles or storage services must be equally secure, specifically when the overall solution assumes a high-level of trust towards these external systems.

2) EXECUTION LAYER

Rule-bases and program code. Examples: smart contracts, chaincode, atomic swaps, tokens, etc. The Impair Process Control tactic applies at this DLT stack layer, and the following are the techniques that may be used:

- *Unauthorized Command Message (T0855)*: similar to the API entry points, smart contracts must verify the identity/role and effective permissions over the target request to reduce the risk of bypassing internal control mechanisms. For example, in a system with 1) a measurement device; 2) a control algorithm; and 3) an actuator, the measurement device should not be allowed to issue direct control commands to the actuator;
- *Modify Parameter (T0836)*: similar to service APIs, smart contracts may be vulnerable to maliciously crafted invocations. Therefore, mechanisms to handle incorrect parameters or data formats should be developed.

3) CONSENSUS LAYER

Consensus protocols include proof-based, voting-based, etc. This layer would play a role, but other layers would reflect the impact. This should be considered a ''transient'' layer used by an attacker. For example, a DoS or data destruction attack will require some knowledge of this layer to be successful, but it is not the primary entry point. Therefore, this tactic does not apply at this DLT stack layer.

4) DATA MODEL LAYER

Data and time synchronization include ordering services, block creation, chain structure, and hashing. The tactic does not apply at this DLT stack layer.

In specific application use cases, incorrectly-handled application and execution layer behaviors could lead to inconsistent or incorrect data states. Such inconsistencies could lead to incorrect decisions (e.g., clearing a market that appears to have not received bids). This issue can be compounded by the ledger's immutability, making it difficult for DLT applications to correct the error.

5) NETWORK LAYER

Peer-to-peer transaction broadcast/discovery, including connectivity, runtime, telecommunications, and network parameters. The Impair Process Control tactic applies at this DLT stack layer, and the following technique may be used. Once the network layer is compromised, an adversary may access the other six DLT stack layers indirectly. The only applicable technique is the *Module Firmware (T0839)*. Adversaries may install malicious or vulnerable firmware onto network hardware devices.

TABLE 11. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the impair process control tactic.

Technique	Mitigation
Brute Force I/O (T0806)	M0937 Filter Network Traffic M0807 Network Allowlists
Modify Parameter (T0836)	M0800 Authorization Enforcement
Module Firmware (T0839)	M0801 Access Management M0945 Code Signing M0804 Human User Authentication M0813 Software Process and Device Authentication
Spoof Reporting Message (T0856)	M0802 Communication Authenticity M0937 Filter Network Traffic
Unauthorized Command Message (T0855)	M0802 Communication Authenticity M0937 Filter Network Traffic M0813 Software Process and Device Authentication

6) INFRASTRUCTURE LAYER

Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. The Impair Process Control tactic applies at this DLT stack layer, and the following technique may be used. The only applicable technique is the *Modify Parameter (T0836)*. An adversary may maliciously modify DLT values to ensure integrity in the OT/ICS environment. If these DLT values are modified, the resulting OT/ICS parameters may be invalid.

7) PHYSICAL LAYER

Systems participating on behalf of the users, for example, sensors, IoT devices with UID, OS, etc. This tactic does not apply at this DLT stack layer.

Unauthorized or invalid command messages may reach the physical layer. It is difficult for end/edge devices to differentiate between genuine/fake commands if they appear to originate from the DLT because the end/edge devices do not often have the built-in capability to contextualize commands and decide if a command is legitimate or adversarial. Therefore, protection mechanisms will be limited to ensuring the validity of the commands (e.g., is the command within the device capabilities? Is the action consistent with the local observation?). The protection mechanisms should be implemented in the other DLT stack layers. One mitigation strategy is implementing device-level whitelisting to ensure participants are positively identified before access is granted.

8) MITIGATION

Table 11 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Impair Process Control tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

L. IMPACT

For this analysis, an assumption is that the DLT is not an integral part of the OT/ICS environment. This means that the DLT cannot monitor and control physical devices. Any interaction with the OT/ICS environment is secondary to the

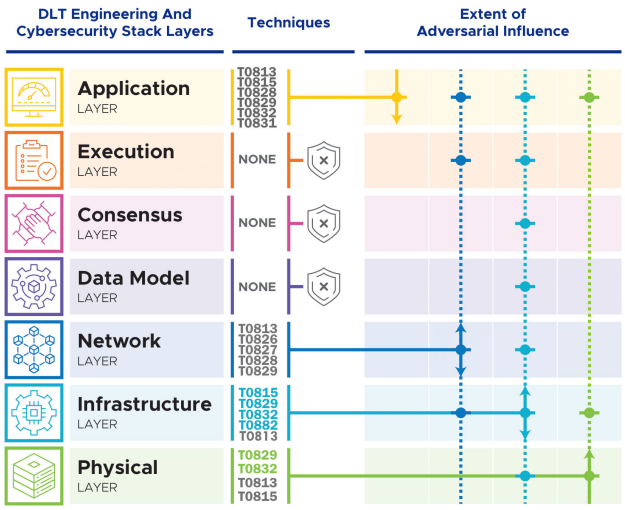

FIGURE 15. Impact tactic analysis and mapping to the DLT stack.

impact on the DLT stack layers. OT/ICS devices should have autonomous, fail-safe operational schemes that can guarantee end-devices will continue to operate securely and functionally. Loss-of-control and loss-of-visibility due to DLT-related failures are valid concerns. Therefore, mechanisms to avoid, detect, and mitigate these scenarios should be engineered from the beginning. Figure 15 illustrates the analysis.

1) APPLICATION LAYER

Applications that trigger rule-bases and program code, including APIs, UI/GUIs, Oracles, distributed applications, marketplace, monetization, etc. The Impact tactic does not apply at the application layer.

The following techniques may be applicable in application-specific use cases in the OT/ICS environment. However, these are outside the scope of the DLT.

- *Denial of Control (T0813)*: Attackers could compromise applications and deny user control of DLT applications. This technique may be used in applications that enable control actions via a DLT system;
- *Denial of View (T0815)*: Attackers could disrupt operator oversight on the status of the ICS environment;
- *Loss of Productivity and Revenue (T0828)*: Attackers could disrupt or damage the availability and integrity of control system operations, devices, and related processes;
- *Loss of View (T0829)* and Manipulation of View (T0832): Incorrectly protected telemetry APIs could be exploited, leading to incorrect health or operational state records. This is particularly applicable when a "translator or gateway" service is used to interact with the external world, and this interface can be abused. Mitigation strategies include providing operators with redundant, out-of-band communications to support monitoring and control of the operational processes;
- *Manipulation of Control (T0831)*: If devices are dependent on control signals transmitted through a DLT, there is a risk of causing Stuxnet/Industroyer type attacks if the device communication channel is not adequately protected by digital signatures or authentication.

2) EXECUTION LAYER

Rule-bases and program code, examples include smart contracts, chaincode, atomic swaps, tokens, etc. This layer would play a role, but other layers would reflect the impact. For other tactics and techniques, this should be considered a "transient" layer. For example, although manipulating the data input will lead to an incorrect decision, the impact will appear on other DLT stack layers. Therefore, the Impact tactic does not apply at this layer.

3) CONSENSUS LAYER

Consensus protocols include proof-based, voting-based, etc. This layer would play a role, but other layers would reflect the impact. For other tactics and techniques, this should be considered a "transient" layer. For example, although manipulating the data input will lead to an incorrect decision, the impact will appear on other DLT stack layers. Therefore, the Impact tactic does not apply at this layer.

4) DATA MODEL LAYER

Data and time synchronization include ordering services, block creation, chain structure, and hashing. This tactic does not apply at the data model layer.

5) NETWORK LAYER

Peer-to-peer transaction broadcast/discovery, including connectivity, runtime, telecommunications, and network parameters. The Impact tactic does not apply at this layer.

For the OT/ICS environment, the following techniques are applicable. However, these are outside the scope of the DLT.

- *Denial of Control (T0813)*: If devices are dependent on control signals transmitted through a DLT, there is a risk of causing communication disruptions that may lead to a loss of control if the remote device does not include an autonomous, fail-safe controller;
- *Loss of Availability (T0826)*: An adversary may disrupt components and services;
- *Loss of Control (T0827)*: This includes sustained loss of control and/or the inability to recover from a malicious event;
- *Loss of Productivity and Revenue (T0828)*: Attackers could disrupt or damage the availability and integrity of control system operations, devices, and related processes;
- *Loss of View (T0829)*: An adversary may cause a temporary or permanent loss of view of the DLT.

6) INFRASTRUCTURE LAYER

Data storage entities include logical blockchain nodes: virtual machines, clusters, Kubernetes, etc. This tactic applies to the infrastructure layer using the following techniques. Once the

infrastructure layer is compromised, an adversary may have indirect access to the other six DLT stack layers.

- *Denial of View (T0815)*: An adversary may modify the DLT information to disrupt operations;
- *Loss of View (T0829)*: An adversary may cause a temporary or permanent loss of view of the DLT;
- *Manipulation of View (T0832)*: An adversary may manipulate DLT status and information sent to the DLT nodes.
- *Theft of Operational Information (T0882)*: Public, encrypted ledgers may be left exposed during an attack, leading to unwanted data exfiltration. Similarly, compromised peers could be used to extract all the data that is accessible to the peer.

For the OT/ICS environment, the *Denial of Control (T0813)* technique is applicable at the Infrastructure layer. Suppose devices depend on control signals transmitted through a DLT. In that case, there is a risk of causing communication disruptions that lead to losing control if the remote device does not include an autonomous, fail-safe controller. However, this is outside the scope of the DLT.

7) PHYSICAL LAYER

Systems participating on behalf of the users, examples include sensors, IoT devices with UID, OS, etc. This tactic applies to the physical layer using the following techniques.

- *Loss of View (T0829)*: An adversary may cause a temporary or permanent loss of view of the DLT;
- *Manipulation of View (T0832)*: An adversary may manipulate DLT status and information sent to the DLT nodes.

For the OT/ICS environment, the following techniques are applicable. However, these are outside the scope of the DLT.

- *Denial of Control (T0813)*: If devices are dependent on control signals transmitted through a DLT, there is a risk of causing communication disruptions that could lead to a loss of control if the remote device does not include an autonomous, fail-safe controller;
- *Denial of View (T0815)*: An adversary may modify the DLT information to disrupt operations.

8) MITIGATION

Table 12 provides a list of the suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the Impact tactic. Each mitigation strategy may need to be tailored for the specific DLT architecture.

TABLE 12. Suggested MITRE ATT&CK® ICS mitigations for the cited techniques in the impact tactic.

Technique	Mitigation
Manipulation of Control (T0831)	M0802 Communication Authenticity
Manipulation of View (T0832)	M0802 Communication Authenticity
Theft of Operational Information (T0882)	M0922 Restrict File and Directory Permissions
Manipulation of View (T0832)	M0802 Communication Authenticity

IX. CONCLUSION AND POTENTIAL FUTURE WORK

This work uses the MITRE ATT&CK® ICS matrix to explore the security aspects of using DLT for the energy sector. Building upon the previously published DLT technology stack, this paper provides an in-depth analysis of potential adversarial techniques and tactics potentially relevant to the layers of the DLT stack. Because DLT is not largely used in real-world energy sector use cases and deployments, particularly in the OT/ICS environment, the adversarial threat analysis was approached by allocating adversarial tactics and techniques to the DLT stack layers. Future work will propose a comprehensive approach to address security needs for the entire DLT for upstream and downstream applications. This may include:

1) Governing principles to cyber security risk management;
2) A cybersecurity framework for the agents in any downstream or upstream device connected to a DLT;
3) Principles for cyber risk assessment by technology type such as remotely controlled Distributed Energy Resources (DERs);
4) Reporting mechanisms for risk-based incidents and threats;
5) Governance and oversight of internal controls and risk mitigation techniques;
6) Disclosure policy concerning cyber security assessment and identified threats;
7) In-depth attack propagation analysis through threat modeling driven by specific scenarios. Examples are as follows:
   a) If an adversary uses a worm or malicious code to target the components of the execution layer, the adversary could laterally move and impact components in other DLT layers. If malware is designed to target components of a DLT layer, what level of propagation and impact can the malware have on components in other layers?
   b) In a large DLT ecosystem, when the execution layer processes are distributed across multiple edges/nodes, the execution layer spans across different nodes executing in parallel. In such scenarios, qualitative estimates of the attack propagation and possible attack trees could be investigated;
   c) T0889 (Modify Program), T0839 (Module Firmware), T0873 (Project File Infection), and T0857 (System Firmware) are all related to the upgrade, update, or maintenance of operating systems or applications running on Intelligent Electronic Devices (IEDs), so they are typically outside the domain of a DLT. However, if a DLT-based package manager is used to track or validate changes or updates to IED software or firmware, incorrect information could cause

TABLE 13. Tailored mitigation strategies. Part 1 of 3.

Mitigation	DLT-specific guidance
M0948 Application Isolation and Sandboxing	This applies to all programmable components, including smart contracts, software in the application layer, and components of ICS/OT systems participating in the DLT.
M0926 Privileged Account Management	The DLT-based use case environment will involve multiple account classes, and the privileged users may change depending on the DLT layer. For instance, applications running on non-utility systems may have the system owner as the privileged user for the application. A consortium of utilities and other administrative entities responsible for DLT operations, such as consensus, block formation, Verification and Validation (V&V) at the DLT level, etc., may involve privileged users across multiple administrative entities. At the infrastructure and physical layer level, the assets may have utility-based and non- utility-based owners. Across these varying user classes, privileged users and their controls should be defined. Specific best practices around this mitigation are beyond the scope of this paper - generally known best practices should be used.
M0801 Access Management	The general guidance discussed for M0926 applies here but to all users, not just the privileged ones. The DLT nodes within the network premises of an organization could take advantage of their AD to define access management to the nodes. Similarly, access controls between the systems and the nodes can be restricted through routing Access Control Lists (ACL) and firewall rules. If an OT/ICS asset directly participates in the DLT network without an intermediate node (infrastructure layer), the device may or may not synchronize with the AD. Instead, the device may have local authentication mechanisms. In such cases, the use case owner should decide if the asset can directly participate in the DLT or through an intermediate DLT node that runs on a virtual/physical machine that can be connected to AD. Evaluating device-level security hardening will be crucial in making the decision. Principles of least privilege and need-to-know should be used in user access management and system access management. Also, principles of least functionality should be implemented.
M0800 Authorization Enforcement	As suggested by MITRE, Role-based Access Controls (RBAC) should be used within the DLT and interactions of the systems/users with the DLT.
M0804 Human User Authentication	As discussed under M0926, various user groups and classes may exist. Some users may be responsible for the nodes forming consensus, and some may be the owners of smart contracts. Also, some users may be OT/ICS asset owners, application owners, or application users, etc. Shared/same credentials and default credentials should not be implemented in the ecosystem. Multi-factor authentication should be implemented across all aspects of user interaction with the DLT components and DLT connected/peripheral components, including the ICS/OT devices that participate in the DLT.
M0813 Software Process and Device Authentication	The authentication between the application and smart contract, DLT node and corresponding ICS/OT asset, etc. may rely on APIs. Secure design, verification, and monitoring of the APIs and their use is critical. Furthermore, DLT's peripheral network monitoring solutions can be used to monitor all traffic associated with the DLT within an organization and between organizations in an agreed-upon consortium that are ultimately responsible for fault-tolerant DLT operations.
M0918 User Account Management	The scope of this may be limited to the users within the organization. A layer of administrative users should enforce the user access management controls for those users. The permissions will vary based on the user access to the DLT nodes and participating ICS/OT or software assets. For the non-organizational users (owners of devices that run applications that interact with the smart contracts and owners of external ICS/OT systems such as DERs), an authorized/whitelisted user list should be maintained. Furthermore, monitoring solutions, such as Intrusion Detection System (IDS) and Intrusion Prevention System (IPS), should be in place for deep packet inspection of traffic in and out of the DLT network.
M0945 Code Signing	Binary and application integrity with digital signature verification should be enforced for all software components, including the application code, smart contracts, code that runs on the DLT node, and the code that runs on the ICS/OT systems that participate in the DLT through the respective DLT nodes.
M0817 Supply Chain Management	The supply chain management policies should be enforced on the DLT ecosystem (network, nodes, connected assets, etc.). Software Bill of Materials (SBOM) of all code should be performed to identify malicious libraries in the applications, smart contracts, etc. If hardware DLT nodes are used, Hardware Bill of Materials (HBOM) of the nodes and associated ICS/OT should be performed, and discovered vulnerabilities should be mitigated. To prioritize mitigations, the scope of vulnerabilities and threats should be evaluated in the context of DLT operations that support the use cases. Suppose vulnerabilities and threats are identified in the DLT (or its components). In that case, the risk should be evaluated in the context of the consequences of continuing/operating the DLT components and the potential impact on the ICS/OT systems. Principles of Safety, Reliability, and Productivity (SRP) should be used in the risk-mitigation decision-making process.
M0938 Execution Prevention	Following MITRE's guidance, execution of all unauthorized code should be prevented in all virtual and physical systems in the DLT ecosystem.

failures or security bugs. Any analysis would be use case specific.

8) Perform an analysis using the MITRE Enterprise ATT&CK® Enterprise matrix. Although there is some commonality across both MITRE matrixes, there are some unique differences across tactics and techniques;

9) Evaluate if the proposed mapping process can be extended to the attacks such as exploitation for Privilege Escalation in proof-based consensus such as PoS consensus. Such analysis will lead to understanding the adversarial ability to manipulate privileges of components across the DLT stack layers

TABLE 14. Tailored mitigation strategies. Part 2 of 3.

Mitigation	DLT-specific guidance
M0816 Mitigation Limited or Not Effective	Unused native features and services of all components/assets in the DLT should be disabled. For instance, all DLT nodes can be evaluated with the Security Technical Implementation Guides (STIG) and necessary configuration settings to harden the OS of the systems that host the DLT nodes, run smart contracts, form consensus, etc. Similarly, the ICS/OT participating in the DLT should be thoroughly evaluated to identify features and services that could be abused. In such cases, those vulnerabilities should be mitigated. If that is not possible, the use case owner should strongly consider the risks of connecting the ICS/OT devices directly or indirectly to the DLT.
M0942 Disable or Remove Feature or Program	The guidance discussed under M0816, M0945 is applicable.
M0921 Restrict Web-Based Content	The guidance discussed under M0816 applies here. In addition, the DLT nodes may converse with other DLT nodes outside the organization and over the internet. As much as possible, the external firewall should be used to granularly define the rules for the nodes' interaction. This includes specifying the ports, protocols over expected communications, etc. web-based browsing access should not be needed for DLT operations. Therefore, these services should be restricted at the host/network level and monitored for abuse or anomalous behavior.
M0802 Communication Authenticity	When possible, all interactions between the systems and their respective DLT nodes and all interactions between the DLT nodes (such as consensus forming or block creation nodes, if any interactions exist) should be performed on a trusted network. In architectures with untrusted network segments, MITRE's guidance to utilize secure protocols to authenticate the message and verify the integrity of the message should be followed. This can be achieved through digital signatures or Message Authentication Codes (MACs). Network and endpoint detection should be positioned to detect spoofed messages, unauthorized connections, and communication from and to the DLT nodes.
	M0814 Static Network Configuration The guidance provided by MITRE applies to all computing sources (e.g., VMs and physical systems with DLT nodes, connected storage, etc.) utilized in the DLT environment.
M0808 Encrypt Network Traffic	All DLT transactions or interactions should be encrypted to the maximum possible extent. This includes any API-driven interactions between the application and execution layers and interactions between the infrastructure, physical, execution, and application layers. All interactions will leverage network layer components, and therefore end-to-end encryption of all traffic should be strongly considered. All interactions about the data-model and consensus layers should be encrypted to ensure that the independent decisions by the nodes in these layers are confidential. An expansion to this mitigation strategy is to ensure that data-at-rest is encrypted, including the ledger, all static or at-rest data records on systems hosting the nodes, etc., any ICS/OT cannot inherently support encryption. The infrastructure hosting the DLT nodes could most likely support encryption, but if the respective ICS/OT systems cannot support encryption, then in-line peripheral encryption solutions may be considered. The primary concern is the impact on latency and availability.
M0930 Network Segmentation	Physical and logical network and perimeter segmentation and segregation should be used. Included is some tailored guidance: depending on the DLT architecture (single organization and single network vs. multi-organizational/multi-network architecture), the DLT nodes will most likely communicate over the internet. Depending on the use case, external non-utility applications may interact with the DLT (e.g., a non-utility node submits a bid to the DLT that runs the market logic). Irrespective of the use case, it is important not to compromise the organization's SRP. Therefore, the ICS/OT that supports the SRP triad should not be disrupted. Evaluate each use case architecture with and without DLT and compare the attack surface analysis. Perform an attack surface analysis and threat modeling of a use case with non- DLT-based traditional methods and compare against the attack surface analysis with DLT. Based on the findings, define network and device level segmentation and segregation. For instance, to limit any lateral movement through compromised DLT node systems, ensure that the infrastructure with DLT nodes is on a separate VLAN compared to the respective ICS/OT systems. Such design will ensure layer-2 segmentation, which is insufficient. Ensure that explicit firewall rules are defined to allow and deny communication/traffic between the DLT nodes and other systems. Such design will ensure layer 3-7 segmentation. Deploy IDS/IPS with deep packet inspection on communications with the DLT nodes. Suppose DLT has been used in ICS/OT use cases with some level of command and control aspects. In that case, the network and endpoint security tools can be deployed in monitor mode not to disrupt the interactions and minimize the risk of false positives. Ensure to do log collection and analysis from the DLT nodes (physical and virtual). When possible, use technologies such as Extended Detection and Response (XDR) / Endpoint Detection and Response (EDR) / Network Detection and Response (NDR) agents and antivirus/antimalware agents on the DLT nodes (physical and virtual). Leverage the guidance from IEC 62443 and implement the zoning and conduit model that includes the DLT infrastructure in the environment. Furthermore, leverage Demilitarized Zone (DMZ) for internet-facing communications from the DLT nodes.

TABLE 15. Tailored mitigation strategies. Part 3 of 3.

Mitigation	DLT-specific guidance
M0807 Network Allowlists	Most of the guidance from M0930 applies here. The use case owners should leverage firewalls and granularly define allow lists. However, this can be challenging for some DLTs if the DLT network is configured with dynamic Internet Protocol addresses (IPs) assigned to the nodes. In such a case, whitelisting/ allows listing the IPs is impossible. If dynamic IP addressing is used for the DLT nodes across participating organizations, note that the Security Information and Event Management (SIEM) in the respective organizations may falsely categorize the inter-organization DLT node traffic as malicious. Ensure that the organizations define firewall rules to allow specific IPs of the DLT nodes to communicate, which may even involve static IP addressing. Some challenges can be addressed at the governance stage of using a permissioned DLT by agreeing upon ports, IPs, and other network details to build Allowlists. For some architectures, technologies such as VPN, Internet Protocol Security (IPSec), etc., can establish predetermined connections between the DLT nodes.
M0922 Restrict File and Directory Permissions	Guidance discussed in M0801, M0816, and M0800 apply here. Since the DLT nodes might communicate between zones and to outside nodes through the internet, evaluate the adversarial impact if the infrastructure hosting the nodes and supporting network components is compromised. Use the inferences to define file and directory-level access to assets (human, hardware, and software assets).
M0937 Filter Network Traffic	Most of the network and end-point security and network traffic analysis guidance discussed in the above mitigations is applicable here. DLT-tailored guidance under M0807 and M0930 is relevant here. For more details, MITRE's baseline guidance applies to the DLT supporting and hosting infrastructure and pertinent network traffic.
M0815 Watchdog Timers	Use of watchdog timers will help address denial of service attacks. If the DLT-based use case architecture is not configured/built based on the guidance discussed in the above mitigations, a compromised node can be used by the adversary to start a DoS attack on the ICS/OT core operations. Watchdog timers will add another layer of detection measures to check for system responsiveness. This mitigation may have little to do with the core DLT infrastructure itself but provides advantages to protect/defend the operations infrastructure that communicates with the DLT nodes.

of the nodes that have more weight in consensus mechanisms.

10) Evaluate and apply to distributed multi-agents systems: A longer-term noteworthy future opportunity of the presented work is in the space of distributed multi-agents systems research:
    a) Use and integration of other global information sources and threat intelligence sources that could potentially inform the severity of the TTPs across DLT layers;
    b) Ability to use the stack to explore the opportunities in the multi-agent systems to reduce resource wastage through event-triggered mechanisms. For instance, exploring the role of DLT in cooperative fault-tolerant output regulation of linear heterogeneous multi-agent systems via an adaptive dynamic event-triggered mechanism and robust adaptive event-triggered fault-tolerant consensus control of multi-agent systems with a positive minimum inter-event time. We believe that the presented stack can be useful in architecting the above solution space.

During the execution of the presented work, the authors faced three noteworthy challenges:

1) Maintain consistency across the mappings presented due to the lack of automated tools to assist with the effort;
2) Lack of extensive scientific literature (conference and journal publications) that is similar to the presented work; and
3) Inadvertent confusion between cybersecurity and cyber resiliency.

The team addressed those challenges by defining the rules of execution/mapping (discussed in Section VII) that were consistently followed across all mapping exercises. The team then used MITRE's documentation and NIST definitions/standards extensively in parallel and used limited but relevant conference/journal publications.

The present work involved a manual process of mapping and analysis. Therefore, the future work opportunity here is to automate the rules of execution/mapping discussed in this paper to accommodate any future changes to the ATT&CK® matrix and the DLT stack. In the OT environment, performing analysis while the system is operational may have a serious impact, such as bricking devices or impacting latency. Digital models could be used, but they would need to be developed and assessed in the IT environment. Also, if a digital model is to be useful, it must match the operational architecture.

Future studies will focus on the cyber risk assessment, cybersecurity, and cyber resiliency analysis of DLTs applied in the power energy sector for the remote monitoring and control of DERs, particularly in the case of scenarios involving the tokenization of energy flows between prosumers and the activation of smart contracts among energy aggregators/prosumers and market operators or power utilities.

Considering the interdependent structure of power systems as previously expounded, it is plausible that susceptibilities within the DLT stack could be exploited by means of malware injection, DoS attacks, or man-in-the-middle attacks. Furthermore, owing to the decentralized structure of DLTs, power systems may be vulnerable to double-spending, 51% and similar attacks, in which an actor acquires control over the majority of the network's computational resources to

influence transactions. These attacks depend on the specific consensus protocol that is implemented.

The utilization of this framework, in conjunction with the inherent benefits of DLT such as transparency and immutability, has the potential to enhance the robustness and resilience of power systems in the face of cyber threats.

Businesses and industry should adhere to cybersecurity regulations and legislative standards, such as the Directive on Security of Network and Information System (NIS and NIS 2.0 Directives) within the European Union and the guidelines set forth by the National Institute of Standards and Technology (NIST) in the United States.

APPENDIX. TAILORED MITIGATION STRATEGIES

See Tables 13–15

ACKNOWLEDGMENT

The content presented in this article documents the activities of the IEEE Standards Association (SA) P2418.5 Working Group and IEEE Energy and Blockchain Technical Community members. This includes the IEEE SA, IEEE Power and Energy Society (PES) Smart Building, Loads and Customer System (SBLC) Technical Committee, and the IEEE Blockchain Initiative. We acknowledge these organizations for sponsoring and promoting some of these initiatives.

REFERENCES

[1] S. Saha, N. Ravi, K. Hreinsson, J. Baek, A. Scaglione, and N. G. Johnson, "A secure distributed ledger for transactive energy: The electron volt exchange (EVE) blockchain," *Appl. Energy*, vol. 282, Jan. 2021, Art. no. 116208.

[2] Ü. Cali, M. Kuzlu, M. Pipattanasomporn, J. Kempf, and L. Bai, *Digitalization of Power Markets and Systems Using Energy Informatics*. New York, NY, USA: Springer, 2021.

[3] E. Androulaki et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proc. 13th EuroSys Conf.*, New York, NY, USA, 2018, pp. 1–15.

[4] The MITRE Corporation. (2022). *CVE Security Vulnerability Database: Apache Couchdb Security Vulnerabilities*. [Online]. Available: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-19046/Apache-Couchdb.html

[5] S. N. G. Gourisetti, A. Lee, R. Reddi, K. Isirova, M. Touhiduzzaman, D. J. Sebastian-Cardenas, K. Lambert, Ü. Cali, M. Mylrea, F. Rahimi, P. Nitu, P. Huff, M. Pasetti, and S. S. Saha, "Assessing cybersecurity resilience of distributed ledger technology in energy sector using the MITRE ATT&CK ICS framework," in *Proc. IEEE 1st Global Emerg. Technol. Blockchain Forum, Blockchain Beyond (iGETblockchain)*, Nov. 2022, pp. 1–5.

[6] MITRE Corporation. (2022). *ICS Matrix*. Accessed: Aug. 26, 2022. [Online]. Available: https://attack.mitre.org/matrices/ics/

[7] R. Ross, V. Pillitteri, R. Graubart, D. Bodeau, and R. Mcquaid, "Developing cyber-resilient systems: A systems security engineering approach," Nat. Inst. Standards Technol. (NIST), Gaithersburg, MD, USA, NIST Special Publication 800-160, Dec. 2021. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160v2r1.pdf

[8] *Joint Task Force Transformation Initiative of the National Institute of Standards and Technology (NIST)*, document NIST Special Publication 800-39, U.S. Department of Commerce, Computer Security Division, Information Technology Laboratory, Gaithersburg, MD, USA, Managing Information Security Risk: Organization, Mission, and Information System View, Mar. 2011. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/ nistspecialpublication800-39.pdf

[9] S. N. G. Gourisetti, Ü. Cali, K.-K.-R. Choo, E. Escobar, C. Gorog, A. Lee, C. Lima, M. Mylrea, M. Pasetti, F. Rahimi, R. Reddi, and A. S. Sani, "Standardization of the distributed ledger technology cybersecurity stack for power and energy applications," *Sustain. Energy, Grids Netw.*, vol. 28, Dec. 2021, Art. no. 100553.

[10] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and philosophy," MITRE Corporation, McLean, VA, USA, Tech. Rep. MP180360R1, Jul. 2018.

[11] O. Alexander, M. Belisle, and J. Steele, "MITRE ATT&CK for industrial control systems: Design and philosophy," MITRE Corporation, McLean, VA, USA, Tech. Rep. MP01055863, Mar. 2020.

[12] Dragos Inc. *Dragos ICS/OT Cybersecurity Year in Review (YIR) Report*. Accessed: Aug. 24, 2022. [Online]. Available: shttps://www.dragos.com/resource/dragos-2021-industrial-cybersecurity-year-in-review-announcement/

[13] A. Meneghetti, M. Sala, and D. Taufer, "A survey on pow–based consensus," *Ann. Emerg. Technol. Comput.*, vol. 4, no. 1, pp. 8–18, 2020.

[14] A. Hrga, T. Capuder, and I. P. Žarko, "Demystifying distributed ledger technologies: Limits, challenges, and potentials in the energy sector," *IEEE Access*, vol. 8, pp. 126149–126163, 2020.

[15] T. Roth, M. Utz, F. Baumgarte, A. Rieger, J. Sedlmeir, and J. Strüker, "Electricity powered by blockchain: A review with a European perspective," *Appl. Energy*, vol. 325, Nov. 2022, Art. no. 119799.

[16] K. Y. Yap, H. H. Chin, and J. J. Klemes, "Blockchain technology for distributed generation: A review of current development, challenges and future prospect," *Renew. Sustain. Energy Rev.*, vol. 175, Apr. 2023, Art. no. 113170.

[17] P. R. Padghan, S. A. Daniel, and R. Pitchaimuthu, "Grid-tied energy cooperative trading framework between prosumer to prosumer based on ethereum smart contracts," *Sustain. Energy, Grids Netw.*, vol. 32, Dec. 2022, Art. no. 100860.

[18] A. E. Soto, B. L. Bosman, E. Wollega, and D. W. Leon-Salas, "Peer-to-peer energy trading: A review of the literature," *Appl. Energy*, vol. 283, Feb. 2021, Art. no. 116268.

[19] J. Ping, Z. Yan, and S. Chen, "A privacy-preserving blockchain-based method to optimize energy trading," *IEEE Trans. Smart Grid*, vol. 14, no. 2, pp. 1148–1157, Mar. 2023.

[20] U. Cali, M. Kuzlu, D. J. Sebastian-Cardenas, O. Elma, M. Pipattanasomporn, and R. Reddi, "Cybersecure and scalable, token-based renewable energy certificate framework using blockchain-enabled trading platform," *Electr. Eng.*, doi: 10.1007/s00202-022-01688-0.

[21] T. R. Alsenani, "The participation of electric vehicles in a peer-to-peer energy-backed token market," *Int. J. Electr. Power Energy Syst.*, vol. 148, Jun. 2023, Art. no. 109005.

[22] Y. Wu, X. Zhang, and H. Sun, "A multi-time-scale autonomous energy trading framework within distribution networks based on blockchain," *Appl. Energy*, vol. 287, Apr. 2021, Art. no. 116560.

[23] M. Foti, C. Mavromatis, and M. Vavalis, "Decentralized blockchain-based consensus for optimal power flow solutions," *Appl. Energy*, vol. 283, Feb. 2021, Art. no. 116100.

[24] U. Cali, M. Deveci, S. S. Saha, U. Halden, and F. Smarandache, "Prioritizing energy blockchain use cases using type-2 neutrosophic number-based EDAS," *IEEE Access*, vol. 10, pp. 34260–34276, 2022.

[25] K. Zhou, J. Chong, X. Lu, and S. Yang, "Credit-based peer-to-peer electricity trading in energy blockchain environment," *IEEE Trans. Smart Grid*, vol. 13, no. 1, pp. 678–687, Jan. 2022.

[26] D. Bose, C. K. Chanda, and A. Chakrabarti, "Blockchain insisted resilience enhancement of power electricity markets using distributed energy trading," *Int. J. Emerg. Electr. Power Syst.*, vol. 23, no. 5, pp. 663–671, Oct. 2022.

[27] S. N. G. Gourisetti, M. Mylrea, and H. Patangia, "Evaluation and demonstration of blockchain applicability framework," *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1142–1156, Nov. 2020.

[28] Y. J. Song, "Design of binary sequences with optimal frame synchronization property," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2000, p. 353.

[29] F. Schär, "Decentralized finance: On blockchain- and smart contract-based financial markets," *Review*, vol. 103, no. 2, pp. 1–8, 2021.

[30] J. Wang, W. Chen, L. Wang, Y. Ren, and R. Simon Sherratt, "Blockchain-based data storage mechanism for industrial Internet of Things," *Intell. Automat. Soft Comput.*, vol. 26, no. 5, pp. 1157–1172, 2020.

[31] J. M. Assante and M. R. Lee. (2021). *The Industrial Control System Cyber Kill Chain*. [Online]. Available: https://sansorg.egnyte.com/dl/HHa9fCekmc

[32] Y. A. Min, "A study on privilege elevation attack management for smart transaction security on blockchain Etherium based system," *J. Korea Soc. Comput. Inf.*, vol. 24, no. 4, pp. 65–71, 2019.

ANNABELLE LEE (Member, IEEE) is currently a Chief Cyber Security Specialist with Nevermore Security, Evergreen, CO, USA. Her experience comprises 40 years of technical experience in information technology system design and implementation, 15 years of operational technology cyber security for the electric sector, and 30 years of cyber security design, specification development, and testing. She has authored or coauthored many documents on cyber security, cryptography, and testing and has worked with utilities around the world.

SRI NIKHIL GUPTA GOURISETTI (Member, IEEE) received the Ph.D. degree in engineering sciences and systems on EE/ICS/OT cybersecurity research, coupled with multiple top/tier cybersecurity certifications (CISSP, GICSP, and GRID). He led projects in multiple sectors, such as energy and power, biomanufacturing, and advanced manufacturing. He has more than ten years of experience in the energy and manufacturing sectors with a mix of research and development, engineering, operations, and application development. His professional history ranges from driving strategy and engineering execution as a cybersecurity lead in a multi-billion-dollar startup to collaborating and leading research and development for the U.S. DOE, DARPA, and DOS in interdisciplinary for-profit and non-profit (national laboratory) organizations. He currently leads the cybersecurity task force for the IEEE Blockchain in Energy Standards WG (P2418.5).

DAVID JONATHAN SEBASTIAN-CARDENAS (Member, IEEE) received the B.E. and M.Sc. degrees in electrical engineering from Instituto Politécnico Nacional, Mexico City, Mexico, in 2013 and 2015, respectively, and the Ph.D. degree in computer science from Washington State University, in 2021. He is currently a Research Engineer with the Pacific Northwest National Laboratory (PNNL). His current research interest includes developing tools for an inherently secure grid.

KENT LAMBERT (Member, IEEE) received the B.S. degree in military history from the USAF Academy, in 1974, the M.A. degree in international relations from the University of Southern California, in 1981, the M.S. degree in strategic and tactical sciences from the Air Force Institute of Technology, in 1983, and the Doctor of Engineering degree in systems engineering from Colorado State University. He is currently a Chief Operations Officer with BlockFrame Inc. His research interests include model-based systems engineering, blockchain applications, cybersecurity, supply chain security, and emergence. He is a member of Special Committee on Security & Privacy.

VICENTE NAVARRO (Member, IEEE) received the bachelor's degree in electromechanics engineering and the master's degree in mathematics from the Technological University of Panama, in 2017 and 2022, respectively. His research interests include across multiple disciplines, including electrical and mechanical engineering, artificial intelligence, population demographics, cyber security, quantum computing, nuclear fusion, and mathematics.

MARCO PASETTI (Member, IEEE) received the M.Sc. degree in industrial engineering and the Ph.D. degree in mechanical engineering from the University of Brescia, Brescia, Italy, in 2008 and 2013, respectively. He is currently an Assistant Professor in electrical energy systems with the Department of Information Engineering, University of Brescia. His current research interests include energy systems, distributed generation, renewable energy sources, photovoltaics, energy storage, demand-side management, electric vehicles charging systems, energy management systems, and smart grids. Since 2021, he has been an active member of the IEEE Blockchain in Energy Standards WG (P2418.5).

ÜMIT CALI (Senior Member, IEEE) received the B.E. degree in electrical engineering from Yildiz Technical University, Istanbul, Turkey, in 2000, and the M.Sc. degree in electrical communication engineering and the Ph.D. degree in electrical engineering and computer science from the University of Kassel, Germany, in 2005 and 2010, respectively. With over 20 years of experience in both industry and academia, he is currently an Associate Professor with the Norwegian University of Science and Technology. His research interests include energy informatics, artificial intelligence, blockchain technology, renewable energy systems, and energy economics. He holds the position of the Chair of the Digital Privacy-Energy Industry TC under IEEE Future Directions and serves as the Vice Chair for the IEEE Blockchain in Energy Standards WG (P2418.5).

KATERYNA ISIROVA (Member, IEEE) received the dual Ph.D. degree in information technologies from V. N. Karazin Kharkiv National University and Aston University, U.K. Her research was decentralized PKI and e-voting systems with post-quantum cryptography. In 2019, she was involved in "Redesigning Trust: Blockchain Deployment Toolkit" within World Economic Forum. From 2014 to 2022, she was an Information Security Analyst with the Institute of Information Technologies (JSC), Kharkiv, Ukraine. In 2022, she joined KPMG Switzerland as a Senior Consultant with the KPMG's Cyber & Digital Risk Consulting Department and provides information security services to clients in a range of industries.

RAMESH REDDI (Member, IEEE) received the M.E. degree in electrical engineering from the Indian Institute of Science, the M.S.E. degree in computer science from the University of Pennsylvania, and the M.B.A. degree from Northeastern University. He holds CISSP and automotive cyber security certificates. He worked with national cyber security professionals to develop NIST smart grid cyber security standards. He is currently an experienced cyber security professional in energy, utilities, and critical infrastructure protection fields. He worked on projects at U.S. utilities—Exelon, PG&E and FP&L, IBM, and HP. He is also the President and the CTO of CybSecBCML Inc. He has experience in collaborating with global clients and speaks multiple languages. He is an active Contributor to IEEE Blockchain in Energy Standards WG (P2418.5) and its task forces.

PUICA NITU (Member, IEEE) received the M.Sc. degree in power systems and economics from the Polytechnic University of Bucharest, Romania. She was with Power System Planning and Operations, Hydroelectric, Energy Markets, IT (WINDII Project for NASA Environmental Satellite). She is engaged in smart grid and renewables with an increasing international management consulting role. She is currently the Chair of Blockchain Use Cases TF for the IEEE Blockchain in Energy Standards WG (P2418.5). Her research interests include utilities, capital programs, power systems, power system planning, operations, policy, specialized seminars in financial risk in electricity, smart grids, and decarbonization.

MD. TOUHIDUZZAMAN (Member, IEEE) received the Ph.D. degree in electrical engineering from Washington State University. He is currently a Cyber Security Engineer with the Pacific Northwest National Laboratory, Electric Security Group. His research interests include power system vulnerability assessment, grid communication systems, cybersecurity framework, and risk management framework.

MICHAEL MYLREA (Member, IEEE) received the Ph.D. degree in artificial intelligence from Capitol Technology University and an NSF CyberCorps Doctorate focused on cybersecurity resilience from George Washington University. He is a CISO and Founder of Cyber Team 7, Distinguished Fellow for applied AI/ML with the Institute of Data Science and Computing, University of Miami. He has more than 20 years of experience in technical leadership (Distinguished Engineer, CISO, CTO, CPO, CSA), Board and Advisory positions with Industry, Government and Academia (Harvard, MIT and Tufts). He has more than 70 technical publications, 20 patents/pending and applied work credited for advances in AI/ML, OT/ICS cybersecurity and DLT. He is active on several Advisory Boards, Consortium and Task Force.

PHILIP HUFF (Member, IEEE) received the M.S. degree in computer science from James Madison University and the Ph.D. degree in computer science from the University of Arkansas. He is currently an Assistant Professor and the Director of Cybersecurity Research with the Emerging Analytics Center, University of Arkansas at Little Rock. With 15 years of experience in cybersecurity management in the electric sector, he has developed multiple degree and certificate programs in cybersecurity and co-founded Bastazo. He also directs the Emerging Threat Information Sharing and Analysis Center and a CISSP.

FARROKH RAHIMI (Life Senior Member, IEEE) received the Ph.D. degree in electrical engineering from MIT, along with over 50 years of experience in electric power industry. He is currently the Executive Vice President of Market Design and Consulting with Open Access Technology International Inc., (OATI), he oversees OATI's market design initiatives and related consulting services. He is also a Key Contributor with OATI's smart grid and grid modernization solutions. He is a member of Grid Wise Architecture Council (GWAC), the Governing Board of IEEE Blockchain Community, and a number of smart grid and grid modernization task forces and committees collaborating with IEEE, NERC, NIST, WECC, and NAESB.

SHAMMYA SHANANDA SAHA (Member, IEEE) received the B.Sc. degree in electrical engineering from the Bangladesh University of Engineering and Technology, in 2014, and the Ph.D. degree in electrical engineering from Arizona State University, in 2021. From 2015 to 2021, he was a Research Assistant with the Laboratory for Power and Energy Solutions (LEAPS), Arizona State University. He is currently an Engineer/Scientist II with the Electric Power Research Institute, Knoxville, TN, USA. His research interests include the stability analysis of distribution power networks, synthetic distribution feeder generation, blockchain application for transactive energy management, and enhanced grid modeling.