Standardization of Smart Contracts for Energy Markets and Operation

Umit Cali [1], Jonathan Sebastian-Cardenas [1], Shammya Saha [1], Shawn Chandler [1], Sri Nikhil Gupta Gourisetti [1], Tamara Hughes [1], Komal Khan [1], Claudio Lima [1], Farrokh Rahimi [1], and Leonard C. Tillman [1]

[1]Affiliation not available

October 30, 2023

Abstract

This work presents a formal review of smart contracts, including definitions, technical requirements, and potential power and energy-related use cases. This includes in-depth discussions covering cybersecurity, legality and interoperability goals that must be taken into consideration by potential end-users. The paper presents a first attempt towards the standardization of smart contracts (SCs) within the field of power and energy as a work in progress activity under the IEEE Standards Association (IEEE SA) P2418.5 Working Group. This work also proposes a holistic, language-agnostic reference model that is intended to accelerate the adoption of Distributed Ledger Technology (DLT) by industry stakeholders by providing standardized processes. Finally, the paper discusses key takeaways that must continue to be developed to increase SC usage within the energy industry.

Standardization of Smart Contracts for Energy Markets and Operation

Umit Cali[x], D. Jonathan Sebastian-Cardenas[§] *, Shammya Saha[†], Shawn Chandler[**], Sri Nikhil Gupta Gourisetti[§], Tamara Hughes[‡], Komal Khan[‡‡] , Claudio Lima[††], Farrokh Rahimi[‖], Leonard C. Tillman[¶]
[x]Norwegian University of Science and Technology, [§]Pacific Northwest National Laboratory,
[†]Electric Power Research Institute, **Guidehouse, Inc., [‡]TMH Ventures, LLC, [‡‡]University of Oviedo,
[††]Blockchain Engineering Council, [‖]Open Access Technology International, Inc, [¶]Balch & Bingham LLP
Corresponding author: *d.sebastiancardenas@pnnl.gov

***Abstract*—This work presents a formal review of smart contracts, including definitions, technical requirements, and potential power and energy-related use cases. This includes in-depth discussions covering cybersecurity, legality and interoperability goals that must be taken into consideration by potential end-users. The paper presents a first attempt towards the standardization of smart contracts (SCs) within the field of power and energy as a work in progress activity under the IEEE Standards Association (IEEE SA) P2418.5 Working Group. This work also proposes a holistic, language-agnostic reference model that is intended to accelerate the adoption of Distributed Ledger Technology (DLT) by industry stakeholders by providing standardized processes. Finally, the paper discusses key takeaways that must continue to be developed to increase SC usage within the energy industry.**

***Index Terms*—Blockchain, Cybersecurity Distributed Ledger, Smart Contracts, Standardization**

I. Introduction

The global energy landscape is rapidly evolving, leveraging "Industry 4.0" technologies such as artificial intelligence (AI) and Distributed Ledger Technology (DLT) while being driven by the five "D"s: Deregulation, Decarbonization, Decentralization, Digitization, and Democratization. The rapid adoption of such emerging technologies, especially DLT and derivative technologies such as DLT-based Smart Contracts (SCs), requires developing a relevant standardization framework that can help develop a common industrial language and unlock broader potential in terms of more comprehensive industrial implementations. Therefore, standardization of SCs can lead to an effective digitization, decentralization, and democratization of a participatory grid. Furthermore, standardized SCs will manifest the landscape of use cases for energy markets and operations.

A. Brief History of Smart Contracts

The idea of SC was introduced through the work [1] and then advanced through the work done by [2]. The latter intended to digitally facilitate self-executing contractual obligations between two parties without the intervention of any third party. The introduction of SCs transformed blockchain from a distributed database into a platform for decentralized applications. The resulting technology presents a significant opportunity for business and industrial platforms to automate many services, including digital rights, financial transactions, decentralized energy trading, and provenance. In addition, prominent open-source platforms support SCs encouraging developers around the globe to design, test, and deploy innovative applications.

B. Core Definitions used in this work

- **Blockchain:** Named for its data structure, a dataset of transactions or information confined in encrypted blocks linked (e.g., chained) together through cryptographic hashes creating an immutable and traceable record.
- **Agreement:** A concord of understanding and intention, between two or more parties, with respect to the effect upon their relative rights and duties, of certain past or future facts or performances [3].
- **Contract:** A legally binding agreement involving two or more parties that sets forth an exchange of promises of what each party will or will not do [3].
- **Distributed Ledger Technology:** A framework that encompasses shared distributed databases where states, data, transactions, or information is stored/maintained/replicated in digital ledgers and secured through cryptographic functions[1].
- **Smart Contract (SC):** The codified script of tasks, actions, or instructions which are executed automatically and independently according to the terms of contract or an agreement between participating parties. Also interpreted as digital contracts that are stored and executed on the blockchain akin to traditional/physical contracts [5].
- **Decentralized Application (DApp):** A decentralised, deterministic, isolated, and Turing-complete back-end application that may include a user interface (i.e. a SC) running within a virtual environment.
- **Transaction:** The interaction between the participants and the network to read, or update the state in a distributed ledger.

[§]This research is supported by PNNL with funding from the U.S. Department of Energy under contract No. DE-AC05-76RL01830.

[‡‡]Supported by government of Spain - Economy and Industry Minister under grant MCI-20-PID2019-111051RB-I00, by Principality of Asturias - (FICYT) under grant BP19-069.

[1]Even though blockchain is a subset of DLTs; DLT and Blockchain technology are used interchangeably in this work [4].

The subsequent sections describe the energy DLT landscape and an use case segmentation, introduce the IEEE Standards Association (SA) 2418.5 Working Group (WG) [6], and summarize technical, legal, cybersecurity, and inter-operability aspects of SCs.

II. Energy DLT Use Case Segmentation

DLT can be implemented to address several use cases in the energy industry on the entire value-chain of power systems actors and segments, including grid edge devices, bulk generators, electric vehicles (EVs), distributed energy resources (DERs), system operators, energy markets, and more. Work done in [7], [8] proposed a systematic methodology to demonstrate the value of blockchain DLT in various power system use cases. The use cases where DLT can be potentially most effective in the energy industry can broadly be classified into four categories: data recording, financial transactions, energy transactions, and SCs. These areas are briefly explained below:

1) **Data recording**: In energy use cases, DLTs can store all types of operational data through immutable digital data recording while allowing transparent access to all participants. For example, authors in [9] leveraged the DLTs to develop a distributed pricing and verification algorithm for a transactive energy market. Adequate mechanisms for ensuring the privacy and cybersecurity of the data must be developed, ensuring competitive advantages and compliance requirements are met (to be addressed by the cybersecurity TF).
2) **Financial transactions**: For the energy market, energy transactions represent electrical energy as an energy token in DLT records to track its transfer and utilization. Generally, the value record is stored in the digital equivalence of the fiat currency or a virtual currency.
3) **Energy transactions**: The application of DLT currently spreads across transactive P2P retailer electricity and energy trading platforms in the residential, enterprise-campus microgrid, and municipal facilities. While deployments are still in the early proof-of-concept stage, several emerging start-up companies are developing technical solutions, services, and products for these applications.
4) **Smart contracts**: SCs can encode the conditions for fulfilling an agreement among the market participants to govern the exchange of electricity. Applications of SCs in transactive energy systems include calculating the optimum pricing for a double-auction electricity market, EV charging management, renewable energy certification, handling prosumer's market bids, handling complex interaction between utility and prosumers through aggregators, smart metering, prosumer balance management, billing, and more.

III. IEEE SA P2418.5 Working Group

Standardization processes aim to propose, implement, and develop a set of consensus-based rules and procedures generated by various parties such as seasoned industrial and academic professions from interest groups, companies, universities, and standardization entities such as IEEE SA, CIGRE, IEC, and more. The IEEE SA P2418.5 WG carries out standardization efforts of blockchain in energy with the following primary objectives :

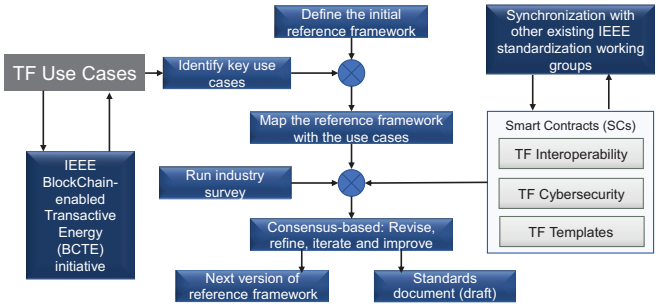

Fig. 1: Overall structure of the IEEE SA P2418.5 Working Group

- Propose and develop a holistic guideline supported with a "reference architecture model" for the energy domain (power, oil, gas and derivatives)
- Mapping the developed guidelines to the most used and / or promising energy blockchain use cases
- Provide open, common, applicable, secure, scalable and inter-operable standards on reference model architecture to the industry, technology provider and end-user
- Disseminate the results to the targeted audience via position papers, articles, newsletter, panels and similar activities

Figure 1 demonstrates the overall structure of the entire WG that consists of various Task Forces (TFs), like TF Use Cases, TF Interoperability, TF Cybersecurity, and TF Smart Contracts. The IEEE Blockchain enabled Transactive Energy (BCTE) Initiative, comprehensive literature review, and the responses provided by the WG members as subject matter experts yielded the identified use cases leading to define the initial reference framework. The outcomes of the TFs are processed interactively via periodical updates to refine and improve the existing content to form the next version of the reference framework and standards documentations.

IV. Technical Aspects of Smart Contracts

SCs offer participants the ability to automate operations while leveraging the benefits (and drawbacks) of DLTs. These inherited capabilities enable SCs to operate over an immutable world-state while providing interfaces to invoke and resolve requests within a fully decentralized environment. Within the energy space, SCs thus offer the needed versatility to integrate (or decouple) different operations/attributes to satisfy the application's needs, thus fulfilling a comprehensive set of technological needs. A suitable SC implementation must be chosen in such a manner that the end-user needs are satisfied and key design elements (or traits) are considered into the final design. A highly simplified taxonomy of these traits is introduced in Figure 2.

- **Determining execution and complexity needs:** Based on the application's needs, users must define if a Turing

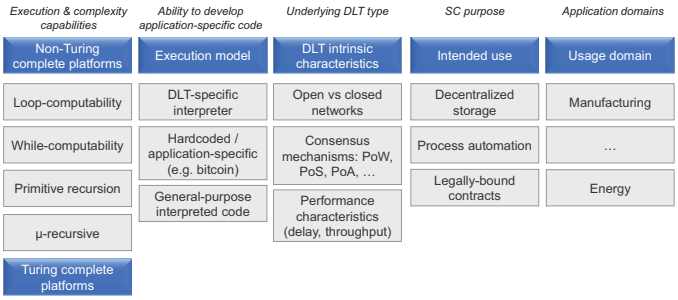

Fig. 2: A taxonomy of SC types

complete language is required. For example, for some instances, such as decentralized storage, non-Turing complete platforms may be sufficient.

- **Execution model and auxiliary tools:** The language characteristics must be capable of supporting the software requirements specification (SRS). In addition, associated development tools should provide core features such as agent enrollment, identity verification, deployment capabilities (including debugging capabilities).
- **DLT characteristics/performance needs:** The selected platform must be capable of satisfying the security, business model and performance of the end-user needs. Examples in this category include selecting open vs permissioned networks, underlying consensus mechanism, and ledger capabilities (delay, latency, storage scalability).
- **SC capabilities:** SCs can satisfy a wide assortment of use cases, ranging from simple decentralized storage to digitally enforcing legal contracts. In all cases SCs should remain deterministic and pass reachability tests(i.e., the code will eventually reach a desired state).

Within the energy sector, SCs can fall into two categories. The first category, namely Smart Energy Contracts (SECs), can automate logical conditionals in the form `if x then y`. However, they are not intended to dictate the behavior of subscribing agents. The second variety, referred to as Smart Legal Energy Contracts (SLECs), integrates legal clauses enforced at the logical level and therefore offer much stronger guarantees [10]. Due to the complexity of designing adequate SLECs, it may be helpful to adapt designs from more generic templates that can later be adapted to meet the needs of a specific application. These template-based SCs could potentially reduce the initial complexity of developing SC, thereby fostering rapid technology deployment. Such templates should be generic enough to enable systematic case deployments while exposing extendable interfaces that can be mapped to the unique grid application.

Works such as [11], build on this concept and propose a modular market architecture that can enable distributed agents (i.e., customers) to participate in a transactive energy market using an extensible platform. Although the reported test case only shows the use of a double auction market in a distribution grid, the underlying concept can be expanded into other market mechanisms due to its: 1) modular architecture, 2) reliance on an Energy Exchange Interface (ESI), which abstracts the load/generation models and 3) its reliance of Unified Model Language that enables language-agnostic implementations.

Following the above research, the SC template architecture can be constructed based on the work in [12], that illustrated an ESI-based five-step market architecture that decouples the market operations across time (specific events/assets within a period are decoupled via object instances). The market architecture can be summarized as follows (these steps can occur using a pipeline-like approach or as part of a single-cycle operation):

- **Registration and qualification:** In this step, the grid objects are added to the DLT. The objects, for example, could represent a load or generation unit operated by a peer participant. It is expected that an intermediary agent validates the information provided by the claimant (i.e., certifies the generator capabilities).
- **Negotiation process:** In this stage, the agents can negotiate the prices they are willing to pay/receive for an asset (such as power). Negotiations can occur in a decentralized P2P environment, via a centralized clearance house, or a combination of both. Agreed negotiations are stored in the ledger for future use.
- **Operation process:** The actual assets are transferred in this stage which could be energy or another derivative, including intangible assets such as carbon credits.
- **Measurement and Verification:** During this phase, a grid-owned system (or agreed third-party) keeps track of the exchanged assets. This could be tied back to a physical quantity (e.g., measured energy) or by querying agents (for intangible/non-measurable assets). Asset's life cycles/states are maintained in the ledger for future use.
- **Settlement:** During the last phase of the process, transferred assets are compared against the negotiated contracts, and penalties/fees are applied.

A. *Legal Aspects*

SCs continue to develop as an area of the law and will need to continue to do so. While the automated execution of the terms of an `if/then` clause can remove much of the uncertainty from the performance and enforcement of a contract term, contracts are typically much broader and often more nuanced. Efforts are underway to advance the functionality of SCs. Among the other issues that will have to be addressed is the interface of the typically precise nature of computer code with the often more flexible nature of the law. Often by design, contracts are an imprecise rendering of the terms of an arrangement/transaction. Parties have become used to 'gaps' in their written contracts being completed by some combination of codified law, relevant court decisions, industry practice, and the course of dealing between the parties. Incorporating such an understood background into SC may prove very difficult, but the parties might be pretty surprised to find their SC having executed without giving effect to such considerations.

Similarly, contracting parties typically expect some degree of flexibility when performing or enforcing their contracts. Such desire for flexibility is reflected in such familiar legal

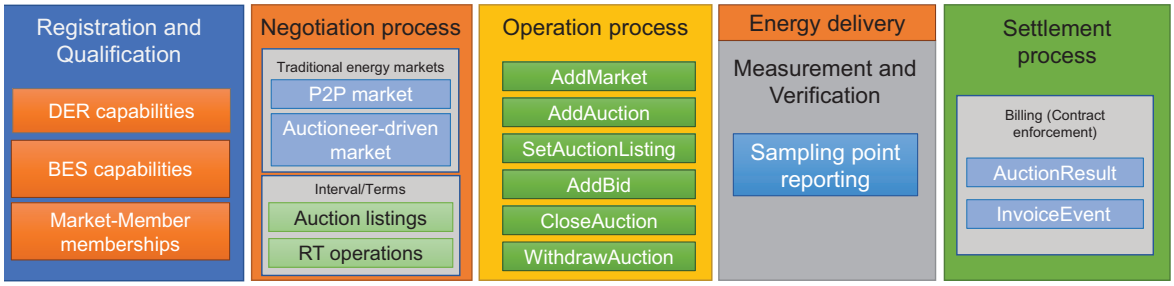

Fig. 3: Functions/objects present within a SC template

terms as "reasonable," "good faith," and "satisfactory." Addressing these and similar issues may lie in a hybrid approach, at least preliminarily. Under such a construct, only specific contractual provisions would be included in the SC, while many more standard contractual provisions could be off-chain and traditionally handled. Of course, such a solution is not ideal and creates its own set of issues. Nevertheless, it may help accelerate the SC era much faster than waiting on the arrival of an all-encompassing SC solution to pressing forward.

B. Cybersecurity Aspects

By levering the capabilities of DLT, SCs can provide transparency and self-governing capabilities that can be used to bring openness and equity into energy applications, thereby potentially increasing the cyber security properties of the target application. At its core, an SC is expected to follow a procedure dictated by an algorithm defined by the application needs and backed up by stakeholders willing to follow its logical constructs. The level of complexity that can be ultimately deployed into a DLT-based solution will be determined by the platform's computational capabilities (e.g., its level of Turing completeness), the required response times, and the developers' ability to implement the software requirements. Under this context, it would be expected that stakeholders (e.g., participating agents) will be subject to the logical outcomes of the deployed SC (i.e., become a legally binding contract).

Under ideal-world conditions, these outcomes should obey the agreed SRS, a concept that is analogous to a legally binding contract. However, deviations from this ideal behavior could arise from: 1) implementation mistakes (e.g., by the software developer); 2) unconsidered, non-deterministic behaviors (such as out-of-sequence events); 3) incorrect or vague SRS that does not reflect the intended outcome; 4) hidden/unknown DLT platform bugs (as well as zero-day vulnerabilities); and 5) malicious or vulnerable application segments (e.g., hidden backdoors, stack vulnerabilities).

Given these risks, contracts must be vetted for adequacy by subscribing parties (a concept equivalent to the legal review process performed by legal teams in enterprise environments). Although this vetting process may have practical limitations due to the complex nature of computer code(and the parties overall resources), there exists a minimum set of best practices that can be performed to minimize potential risk, summarized below:

- **Independent code assurance reviews:** Subscribing entities should seek to employ third-party code reviewers that can compare the implemented code vs. the SRS and perform static and dynamic execution tests to guarantee that the application behaves as intended.
- **Active monitoring for potential vulnerabilities:** During the lifetime of the SC, participating entities must remain informed about potential vulnerabilities that may impact the system. Therefore, corrective actions (such as patching) should be categorized as an essential, operational activity.
- **Follow basic cyber security practices:** Techniques such as fail-safe defaults (e.g., avoid undefined states), principles of least privilege (limit access to activities strictly required for fulfilling a job), and access control mechanisms must be part of the SRS (and enforced in code) [13]
- **Enable mechanisms that can be used to roll-back operations:** Back-up mechanisms that can be used to perform a rollback to a "good state" in case an "unintended state" is reached must be designed from the beginning. However, the mechanisms should be designed to prevent unilateral rollbacks while ensuring that powerful majorities cannot abuse the system (i.e., override small players).
- **Understand the limitations of the technology:** Users must understand the technical limitations of DLTs; for example, relaying on off-chain storage may prevent data corruption but does not eliminate the risk of data loss. Another risk vectors could arise from relying on off-DLT operations to fulfill tasks without analyzing consequences.
- **Promote openness & standardization:** By using standard, open-access components, organizations can deploy secure solutions irrespective of their resources.

Depending on the DLT, SCs may have the ability to access the on-chain data (immutable ledger) and access the off-chain data that resides outside the operating boundaries of the DLT. In such cases, the SC may need to perform out-of-bounds verification and validation tasks while establishing trust anchors and maintain secure sessions with the off-chain databases. The implications of the versioning of SCs must also be addressed at the application level. Such implications include backward compatibility with the existing records of the ledger, which may lead to revisions of previously generated

responses (e.g., to patch a bug). In that context, any implications of identified discrepancies on the current application processes that inherently used previously generated historical outcomes need to be carefully addressed with their potential repercussions clearly understood.

Regarding the security of the SC itself, the programming language of choice and the libraries used to develop the SCs need a thorough evaluation. Hidden vulnerabilities in the third-party dependency libraries often cause significant security issues to the application and the participating entities. On a similar note, misuse of SC should be handled by the authorized entities or the DLT itself. Many of the known SC abuse cases were a result of SC ambiguities. Such cases were seen in publicly run lotteries and financial bidding applications executed in public blockchains. Depending on the complexity of the SC, the program logic needs to go through vulnerability analysis, verification, validation, testing, and external auditing processes.

Depending on the DLT implementation, certain platforms can support native access controls mechanisms to provide identity management/verification, while others will require peer authentication and authorization to be handled (at least partially) through SCs. Therefore, secure software design principles [14], SDLC framework [15], CI/CD pipelining (if and when possible in this context) [16], etc. may be necessary to ensure that the SCs are capable of handling security aspects of the applications.

C. Interoperability Aspects

Interoperability is defined as communication between two or more systems with the ability to exchange actionable information. The WG considered a traditional architectural data approach and defined three layers of the technology-agnostic interoperability framework with device, syntactic and semantic reference characteristics.

The device layer defines the physical characteristics with three universal features. First, a device must have a secure identity and possess communication features that describe its security and level of trust to participate, considering its inclusivity for the system or systems based on control, risk, reliability, membership, or other factors. Second, a device must have a location with resolution sufficient to engage within a system it is expected to interoperate. Finally, a device must record time or operate as a sensor with a system that operates sufficiently based on inclusion requirements. The syntactic layer defines the communication protocols expected to align generally to prevailing industry methods. One distinction is the need to define the network layer, which through the application of a broadcast type protocol could permit peer-to-peer broadcast and discovery in addition to point-to-point and point-to-multipoint communications. In addition, the WG is exploring handshaking methods and serialization of data structures into byte sequences expected to depend upon the interpretation of device layer characteristics. Finally, the semantic layer defines contract meaning, dynamics, behavioral and conceptual characteristics of interoperability.

V. Discussion

This work aimed to provide a brief overview of DLTs and SCs, including definitions, taxonomy, technical requirements, and applicability of potential power and energy-related use cases. Furthermore, the work discusses additional aspects of Smart Contracts regarding standardization, cybersecurity, legality, and inter-operability to help users and policymakers in the decision-making process. This article also demonstrates the first standardization efforts regarding Smart Contracts in power and energy as a work in progress under IEEE SA P2418.5 Working Group. Finally, a generic SC template for energy use cases is proposed to support the development of the DLT agnostic reference model, intended to accelerate the adoption of DLT technology by industry stakeholders in a standardized manner. Future work will further develop the proposed approach and offer it to the industry's service.

References

[1] V. Buterin, "Ethereum white paper ," Tech. Rep., 2013.

[2] "The Idea of Smart Contracts," https://nakamotoinstitute.org/the-idea-of-smart-contracts/, Accessed: 2021-09-15.

[3] "Merriam Webster Dictionary," https://www.merriam-webster.com/dictionary/, Accessed: 2021-09-15.

[4] "Blockchain vs Distributed Ledger Technology," https://101blockchains.com/blockchain-vs-distributed-ledger-technology/, Accessed: 2021-09-15.

[5] S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin, and F.-Y. Wang, "An overview of smart contract: Architecture, applications, and future trends," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 108–113.

[6] "IEEE P2418.5 Working Group," https://sagroups.ieee.org/2418-5/, Accessed: 2021-09-15.

[7] U. Cali, C. Lima, X. Li, and Y. Ogushi, "DLT / Blockchain in Transactive Energy Use Cases Segmentation and Standardization Framework," in *2019 IEEE PES Transactive Energy Systems Conference (TESC)*, 2019, pp. 1–5.

[8] U. Cali and C. Lima, "Energy informatics using the distributed ledger technology and advanced data analytics," in *Cases on Green Energy and Sustainable Development*, 1st ed., P. Yang, Ed. Hershey, PA: IGI Global, 2020, pp. 438–481.

[9] 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," *Applied Energy*, vol. 282, p. 116208, 2021.

[10] C. Lima, "Smart Legal Energy Contract (SLEC): The Need for a Standard-Based Blockchain Contract Framework for the Energy Industry," in *Proceedings of the IEEE PES General Meeting 2020 Conference, August 4th*, 2020.

[11] S. N. G. Gourisetti, D. J. Sebastian-Cardenas, B. Bhattarai, P. Wang, S. Widergren, M. Borkum, and A. Randall, "Blockchain smart contract reference framework and program logic architecture for transactive energy systems," *Applied Energy*, vol. 304, p. 117860, 2021.

[12] S. N. G. Gourisetti, S. E. Widergren, M. E. Mylrea, P. Wang, M. I. Borkum, A. M. Randall, and B. P. Bhattarai, "Blockchain smart contracts for transactive energy systems," 8 2019. [Online]. Available: https://www.osti.gov/biblio/1658380

[13] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, 1975.

[14] S. N. G. Gourisetti, S. Mix, M. Mylrea, C. Bonebrake, and M. Touhiduzzaman, "Secure Design and Development Cybersecurity Capability Maturity Model (SD2-C2M2): Next-Generation Cyber Resilience by Design," in *Proceedings of the Northwest Cybersecurity Symposium*, ser. NCS '19. New York, NY, USA: Association for Computing Machinery, 2019.

[15] "Cloud security engineering: Early stages of sdlc," *Future Generation Computer Systems*, vol. 74, pp. 385–392, 2017.

[16] Red Hat Incorporated, "What is a ci/cd pipeline?" [Online]. Available: https://www.redhat.com/en/topics/devops/what-cicd-pipeline