Smart Contract vulnerabilities affecting Federated Learning's privacy guarantees

PAVAN AWADHPERSAD, University of Twente, The Netherlands

Blockchain-based Federated Learning (BCT-FL) integrates blockchain technology with Federated Learning (FL) to enhance privacy, trust, and auditability. The use of smart contracts in this integration raises the discussion of their documented vulnerabilities and their relevance to FL. This paper examines the impact of smart contract vulnerabilities on the privacy guarantees of BCT-FL systems. It provides a structured taxonomy of vulnerabilities based on trusted sources, maps their effects on key privacy-preserving mechanisms (e.g., secure aggregation, differential privacy, certificate-less authentication), and analyzes their severity through a custom evaluation framework. The findings highlight that certain vulnerability classes, such as access control flaws, improper exception handling, and storage design issues, pose a critical threat to confidentiality and trust. It concludes with a discussion on the framework used and recommends directions for privacy-aware smart contract design, specifically tailored to the needs of Federated Learning systems.

Additional Key Words and Phrases: Federated Learning, Blockchain, Smart Contracts, Privacy-Preserving Mechanisms, Privacy Vulnerabilities, Ranking Framework

1 INTRODUCTION

Federated Learning (FL) has emerged as a promising framework for addressing concerns over data privacy in traditional machine learning [33]. By allowing individual clients to collaboratively train a shared model without exchanging raw data, FL significantly reduces the risks associated with centralized data collection. Its applications have quickly expanded into domains like healthcare, finance, and edge computing, where data privacy and local computation are both essential [47]. Yet, the decentralized nature of FL introduces new challenges in coordination, trust, and incentive design. To manage these issues, research has proposed integrating Blockchain Technology (BCT) into FL systems. Blockchain's immutable ledger [38] and smart contracts provide the tools to automate coordination, track contributions, and distribute rewards fairly in a decentralized manner [9]. This integration, often referred to as BCT-FL, is gaining traction not only in academic proposals but also in early-stage implementations [48]. However, this architectural shift is not without its risks. Smart contracts, while central to BCT-FL systems, are known to be vulnerable to various logic and security flaws [44]. From infamous exploits such as the DAO reentrancy attack [16] to more subtle bugs in access control and randomness, the vulnerabilities of smart contracts are well-documented. What remains to be adequately explored is how these vulnerabilities affect the privacy and security purposes of Federated Learning systems. In particular, smart contract vulnerabilities may compromise critical aspects of privacy-preserving mechanisms, model integrity, or the fairness of reward distribution, therefore undermining the very grounds of adopting FL in the first place. This thesis examines this intersection by conducting an assessment of smart contract vulnerabilities within the context of privacy-preserving mechanics of Federated Learning. It identifies and explains which classes of smart contract vulnerabilities are most relevant to FL, and how they affect different mechanisms (such as Differential Privacy, Secure Aggregation, Secret Sharing, and incentive mechanisms governed by smart contracts). Furthermore, to give structure to this research, a Design Science Research methodology (DSRM) will be utilized. It defines a real-world problem (vulnerabilities undermining privacy in BCT-FL systems), develops an artifact (a privacy-focused risk-ranking framework), and evaluates it through conceptual justification and literature alignment.

Research Questions. These questions act as guidelines through the process:

- **RQ1:** What types of smart contract vulnerabilities are relevant to BCT-FL systems?
- **RQ2:** How do these vulnerabilities affect privacy-preserving mechanisms in FL?

Hypotheses. Accompanied with these hypotheses aiming to support the questions:

- **H1** Only vulnerabilities that involve external calls, data exposure, or access control are relevant to BCT-FL systems.
- **H2** Vulnerabilities related to Unsafe External Calls and Improper Access Control, as defined in the OpenSCV taxonomy [61], have a greater impact on privacy-preserving mechanisms in Federated Learning than other vulnerability categories.

The rest of this paper is organized as follows. Section 2 presents the technical background on Federated Learning, its privacy mechanisms, blockchain technology, smart contracts, and how blockchain is usually integrated into the FL framework. Section 3 goes over the research methodology used in this paper. Section 4 presents Smart contract vulnerabilities relevant for the FL domain and the relevant FL mechanism, and their interaction. Section 5 introduces the artifact, a privacy-centered vulnerability ranking framework, and ranks the relevant vulnerabilities identified. Section 6 discusses and evaluates the validity of the ranking, the research contributions, and finally addresses the research questions and hypotheses. Section 7 is the conclusion, discussing the whole paper, its limitations, and suggestions for future work.

2 BACKGROUND

2.1 Federated Learning

Machine learning(ML) has proven valuable across various domains, including cybersecurity, healthcare, smart cities, e-commerce, and

TScIT 43, July 4, 2025, Enschede, The Netherlands

agriculture, enabling threat detection, diagnostics, traffic optimization, and personalized recommendations. As shown in the paper by Kapoor [26], these advances are driven by access to large volumes of high-dimensional data. However, the paper also highlights a drawback: the difficulty of data collection. To combat this drawback, a different framework called **Federated Learning (FL)** has been developed. FL enables collaborative model training without transferring raw data, addressing many of the limitations inherent in traditional ML systems [33]. Similarly, challenges stem in ML from having to aggregate large sets of data from multiple sources into a central server, such as concerns regarding user privacy, regulatory compliance, and resource usage through communication [50]. These are addressed in FL by utilizing a distributed model training method across decentralized devices or institutions. This allows the training of models without sharing raw data. Instead, only model updates are exchanged between the server and clients, allowing participants to collaboratively develop a global model while keeping their data local and private. This process repeats over several rounds [33, 60]. FL shifts the learning process of the model closer to where the data is generated. This is useful for privacy-sensitive applications such as for use with mobile devices (locally generated data, often containing personal information), for hospitals and medical settings (patient data is highly sensitive and institutions prefer keeping it on-premise due to strict HIPAA regulations [31]), and financial institutions (similar to patient data, institutions often prefer keeping financial data on-premise due to strict regulatory frameworks such as the Gramm-Leach-Bliley Act (GLBA), the General Data Protection Regulation (GDPR), the California Consumer Privacy Act (CCPA), and regional data localization laws [42]). This architecture choice introduces three fundamental characteristics: **Privacy by Design**: Since raw data never leaves the client's device or environment, FL inherently reduces the risks associated with data breaches or misuse [27]. **Communication Efficiency**: Instead of sending entire datasets for training, only model updates are shared, making FL suitable for bandwidth-constrained or mobile environments [36]. **Scalability Across Domains**: FL supports interoperability between devices (cross-device, e.g., smartphones) and environments with different institutions and siloed data (cross-silo settings, e.g., hospitals or banks), each with distinct requirements in terms of trust, data size, and system reliability. [12] However, implementing FL is challenging due to issues such as non-IID(independent and identically distributed) data, varying resources, and unreliable connectivity. Zhao et al. report over a 50% drop in accuracy under highly skewed non-IID data distributions [72]. Furthermore, resource disparities introduce the straggler effect, forcing the server to wait for slower clients and risking dropouts, which in turn slows convergence and degrades performance [36]. Lastly, unstable connectivity causes missed communication rounds and incomplete updates, further hindering convergence and model quality [64]. Moreover, FL does not inherently defend against inference attacks or malicious participants aiming to manipulate model updates [36]. To address these concerns, FL research incorporates various privacy-preserving mechanisms, such as differential privacy [40] [1], secure aggregation [7], homomorphic encryption [28, 54], cryptographic approaches [71], as well as architectural innovations like hierarchical and decentralized FL models [12, 49]. Together, these efforts aim to make FL robust, privacy-preserving, and practical for real-world deployment.

2.2 Blockchain

Blockchain is a decentralized, tamper-proof, distributed digital ledger that records transactions securely, utilizing cryptographic hashes and consensus across a network of computers, eliminating the need for a central authority. [13]. Initially introduced by Satoshi Nakamoto to support Bitcoin, which focused on solving the double-spending problem without relying on trusted third parties to act as central figures [38]. Transactions are grouped into blocks, each cryptographically linked to the previous one using hash functions, forming an immutable and transparent chain. To achieve consensus across the network, blockchain systems employ mechanisms such as Proof-of-Work (PoW)[38], Proof-of-Stake (POS)[51], Delegated Proof of Stake (DPoS)[22], and Proof of Authority (PoA)[24]. These are all designed to maintain data integrity and security, ensuring consistency throughout the blockchain network.

The blockchain network operates by utilizing miners or validators, depending on the consensus mechanism employed, to verify and record transactions in the ledger. In order to motivate participants to perform these tasks, incentive mechanisms are used. These incentives are typically governed by tokenomics, the economic design of token-based ecosystems, which focuses on aligning individual behavior with the overarching system goals [51]. Blockchain participants, such as miners in PoW or validators in PoS, are rewarded with native tokens for securing the network, validating transactions, and reaching consensus [23, 38]. These tokens have monetary value, creating economic incentives for the expected behavior and system maintenance. This structure is based on mechanism design theory, aiming to develop rules and reward structures that lead to the desired behavior (e.g., honest validation, high availability) [30]. Furthermore, blockchain has evolved into a broader technological framework that enhances transparency, trust, and security in various sectors, including finance, healthcare, logistics, and governance [14]. However, despite its broad usability, blockchain faces challenges related to scalability, energy consumption, and regulatory uncertainty [56].

2.2.1 ***Smart contracts.*** Smart contracts are self-executing, immutable contracts stored on a blockchain that automatically enforce the terms of an agreement once the predefined conditions are met. Proposed by Nick Szabo and popularized by platforms like Ethereum, these contracts eliminate the need for intermediaries, reduce transaction costs, and improve reliability by enforcing contractual obligations without the risk of tampering due to their immutability and self-executing nature[56]. Smart contracts operate within the blockchain environment, supporting various processes such as decentralized finance (DeFi), supply chain traceability, digital identity verification, secure healthcare data management, and automated legal agreements by automating and securing interactions across these domains [35].

Despite their potential, smart contracts have their limitations, including coding vulnerabilities, a lack of legal recognition in many jurisdictions, creating uncertainty around their enforceability and limiting their utilization in legally regulated industries or cross-border

applications [4], and difficulties in enforcing and representing real-world events in digital logic, continue to pose significant challenges [56]. To address these, current research explores formal verification methods, better programming practices, and dynamic compliance frameworks that can adapt to different regulatory requirements[25]

2.3 Blockchain integrated Federated Learning (BCT-FL)

FL faces several critical challenges, including reliance on a central aggregator, risks of data or model manipulation, difficulty in tracking and verifying contributions, and a lack of auditability [64]. The integration of blockchain technology addresses these issues by utilizing the decentralized consensus mechanisms that remove the reliance on a trusted central party. Furthermore, blockchain ensures a tamper-proof model update and transparency throughout the entire process. At the same time, smart contracts play a crucial role in enforcing rules and processes, distributing incentives, and automating logic securely and transparently.[39] One of the key synergies between blockchain and FL is the implementation of incentive mechanisms, which encourage client participation in training tasks. Due to the resource-intensive nature of FL, requiring significant computational effort, communication bandwidth, and battery life, clients may be reluctant to contribute without compensation [70]. Blockchain enables a transparent environment for tracking contributions and distributing rewards fairly, often via the use of smart contracts [9, 40].

Smart contracts would automate the process of coordinating client participation, reward allocation, and process enforcement without requiring a central authority [9]. They are crucial in maintaining accountability and decentralized trust in FL systems. However, smart contracts are not immune to vulnerabilities such as reentrancy attacks, arithmetic bugs, and access control issues [1][44]. In the context of FL, such vulnerabilities could be exploited to manipulate contributions, bypass participation policies, or leak data, leading to privacy issues.

2.4 Literature Gap

An increasing number of research efforts and practical applications have demonstrated the potential of integrating blockchain technology (BCT) with federated learning (FL), particularly to enhance data privacy, decentralization, and auditability [9, 29, 46, 48, 53]. These Blockchain-based Federated Learning (BCT-FL) systems have been applied in various domains, including healthcare, IoT, and industrial automation, often relying on smart contracts and consensus mechanisms to coordinate training and establish trust in decentralized environments. However, while the benefits of this integration are well-documented, limited attention has been paid to the security risks and vulnerabilities introduced by smart contracts in FL systems. In particular, there is a lack of systematic analysis on how these vulnerabilities might impact privacy-preserving mechanisms within FL. Furthermore, a domain-specific framework that ranks these vulnerabilities based on their potential to impact FL system privacy or integrity does not exist. This highlights the need for deeper investigation into the intersection of smart contract security and federated learning, especially regarding threat models, attack surfaces, and mitigation strategies specific to BCT-FL deployments.

This thesis addresses that gap by proposing a **structured risk-ranking framework**. The aim is to identify which vulnerabilities pose the greatest threat to privacy in BCT-FL systems and to provide a way to prioritize mitigation efforts. By focusing specifically on privacy impact, instead of general exploit possibility, the framework introduces a unique perspective focused on the data protection requirements of BCT-FL.

3 RESEARCH METHODOLOGY

3.1 Design Science Research Methodology (DSRM) Overview

A Design Science Research Methodology(DSRM) is utilized throughout this paper. DSRM is particularly useful for research that aims to solve practical problems through the design, construction, and evaluation of artifacts, such as models, or in our case, frameworks [43]. Table 1 summarizes the six core activities of DSRM and their specific relevance to this paper.

Table 1. Design-Science Research Guidelines [21]

Guideline	Description
Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.
Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.
Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Design as a Search Process	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

Design as an Artifact: The proposed artifact, a privacy-oriented risk-ranking framework, is constructed through literature synthesis and analytical mapping of vulnerability classes to privacy-preserving mechanisms.

Problem Relevance: The absence of a structured approach to the evaluation of smart contract vulnerabilities in the context of FL privacy is the problem identified. This is grounded in the increasing amount of BCT-FL systems [9, 29, 46, 48, 53] and the security risks that come with those.

Design Evaluation: Its evaluation follows a descriptive approach, as outlined by Hevner et al. [21]. Specifically, it adopts an ex-ante, artificial evaluation strategy, relying on literature alignment, logical reasoning, and relevance to BCT-FL workflows rather than empirical testing.

Research Contributions: This work introduces a novel framework that links smart contract vulnerabilities to FL privacy mechanisms, an underexplored intersection. It offers a structured framework for assessing privacy risks in BCT-FL, contributing to both design theory and practical threat modeling.

Research Rigor: The framework is grounded in established taxonomies and privacy models, constructed by systematically integrating findings from established sources. Its evaluation follows DSRM

[1] explained in more detail in their respective sections

standards for conceptual artifacts, ensuring internal consistency and theoretical soundness.

Design as a Search Process: The framework reflects a design-as-search approach, as outlined by Hevner et al. [21], by comparing multiple taxonomies and mechanisms to identify the most relevant components for BCT-FL. Its structure and scoring criteria were guided by recurring patterns and practical relevance.

Communication of Research: The findings and design process are communicated in this paper in an attempt to support both researchers and practitioners working on privacy, blockchain security, and federated learning systems.

3.2 Research Questions

This paper attempts to answer two research questions. Research question 1 (RQ1) asks: "What types of smart contract vulnerabilities are relevant to BCT-FL systems?" The second research question (RQ2) explores: "How do these vulnerabilities affect privacy-preserving mechanisms in Federated Learning?" To support these questions, the following hypotheses are proposed. H1 claims that only vulnerabilities that involve external calls, data exposure, or access control are relevant to BCT-FL systems, due to the specific roles smart contracts play in such architectures. H2 hypothesises that vulnerabilities related to Unsafe External Calls and Improper Access Control, as defined in the OpenSCV taxonomy [61], have a greater impact on privacy-preserving mechanisms in Federated Learning than other vulnerability categories.

RQ1: To address RQ1, a multi-step analysis was conducted, combining literature review and vulnerability classification. First, a targeted literature review was conducted to identify the privacy-preserving mechanisms used in Federated Learning (FL), including differential privacy[40] [1], secure aggregation [7], homomorphic encryption [28, 54], secret sharing, and related cryptographic techniques [71]. These mechanisms were used as a guide to analyze whether each vulnerability could pose a threat to privacy or data integrity in FL. In the second step, established smart contract vulnerability taxonomies were consulted, specifically OpenSCV [61], while Pishdar et al. [44] was used as an empirical reference to validate relevance and frequency of vulnerabilities in practice. Other sources were either outdated or conceptually subsumed by these. Each listed vulnerability was examined for its potential to compromise any of the identified FL mechanisms or to affect the confidentiality, integrity, or authenticity [10] of client data and model updates. Vulnerabilities were kept if they posed a reasonable threat to FL operations, such as model update submission, reward distribution, participant identification, or training data protection. This included, for example, vulnerabilities like reentrancy, improper input validation, and delegatecall to untrusted contracts. Vulnerabilities irrelevant to FL-specific architectures, such as those addressing only gas efficiency or generic ERC-20 token logic, were excluded. In the final step, each selected vulnerability was documented and analyzed concerning its possible impact on FL system security, with results structured to support the following privacy risk mapping. **RQ2**: To address Research Question 2 (RQ2), the subset of smart contract vulnerabilities identified as relevant to BCT-FL systems will be analyzed to determine their impact on the privacy-preserving mechanisms within FL. The analysis is set up as a three-phase approach. First, each vulnerability was mapped to potential attack surfaces within a blockchain-based FL architecture, focusing on elements such as model update submission, reward distribution, and participant authentication. Second, for each privacy mechanism, the potential for data leakage, unauthorized access, manipulation of logic, or violation of integrity was assessed. Supporting evidence, including technical descriptions and examples from the literature and vulnerability taxonomies, was referenced to affirm the mappings. Finally, each vulnerability's impact was evaluated using the developed structured classification framework and categorized as high-, medium-, low-impact, or in between, based on the severity of its effect on privacy-preserving mechanisms, its exploitability, and detectability. This impact classification enables a detailed understanding of how different types of vulnerabilities compromise privacy in Federated Learning and forms the basis for future risk prioritization.

3.3 Artifact development

The artifact developed in this study is a privacy-focused risk-ranking framework aimed at evaluating how smart contract vulnerabilities affect BCT-FL. Its creation followed a structured, literature-driven process. Vulnerability categories were selected based on existing taxonomies, primarily OpenSCV [61], and further filtered using empirical insights from Pishdar et al. [44] to ensure relevance to smart contracts typically deployed in FL coordination, aggregation, and incentive mechanisms. Each category was conceptually mapped to privacy-preserving techniques used in FL, such as Differential Privacy, Secure Aggregation, and secret sharing, with attention to how specific exploits could compromise their guarantees.

The framework uses three scoring dimensions: Privacy Impact (I), Exploitability (E), and Detectability (D), inspired by the OWASP risk rating methodology[41] as adapted by Ula et al. [59] and adjusted for the privacy priorities of FL. To reflect the importance of privacy, the final Composite Risk Score (CRS) assigns a weight of 0.5 to privacy impact, 0.3 to exploitability, and 0.2 to detectability.

Each vulnerability category was assessed across the three scoring dimensions using a qualitative scale from 0 (negligible) to 3 (severe). These scores were based on documented exploit cases, existing taxonomies, and conceptual analysis of how each vulnerability interacts with privacy-preserving mechanisms and functional roles in FL systems. Privacy Impact (I) was given the highest weight (0.5) to reflect the central aim of the framework: assessing vulnerabilities through a privacy lens. In BCT-FL, preserving privacy is not just a design preference but a foundational requirement, especially in sensitive domains like healthcare or finance, where FL is commonly applied. Exploitability (E) was weighted at 0.3, recognizing that some vulnerabilities, while severe in theory, may be difficult to exploit in real-world deployments due to distinct system configurations or user behavior. Detectability (D) received the lowest weight (0.2), as the presence of a vulnerability, even if easy to detect, can still pose a significant threat if it directly compromises privacy or enables adversarial inference.

This weighting scheme was chosen to balance theoretical risk with practical feasibility. It allows the framework to highlight vulnerabilities that may be easy to overlook during development (e.g.,

metadata exposure or weak access control) but carry substantial privacy consequences if exploited. This structured design allows for meaningful prioritization of vulnerabilities within the BCT-FL threat landscape.

4 LITERATURE REVIEW

4.1 Federated Learning

Federated Learning (FL) was introduced as a decentralized alternative to traditional machine learning, aiming to protect user privacy by keeping data local to each client [33]. However, FL does not inherently guarantee privacy [12, 19, 36, 67], as model updates exchanged during training can leak sensitive information through attacks like model inversion, membership inference, or gradient leakage.

This section outlines the core privacy mechanisms in FL, followed by cryptographic enhancements and architectural considerations, ending with a discussion on smart contracts and their associated vulnerabilities.

4.2 Privacy Mechanisms in Federated Learning

Federated Averaging. FedAvg is the foundational algorithm in most FL implementations, where clients train locally and send updates to a central server for aggregation [32]. While it retains raw data locally, it lacks cryptographic protections, making updates susceptible to inference over time. Moreover, its assumption of IID data doesn't hold in practice, leading to model drift on non-IID datasets [57].

Differential Privacy. Differential Privacy (DP) protects against inference attacks by adding noise to client updates before sharing [40]. Local DP is commonly used in FL, but it introduces a privacy-utility trade-off, especially under non-IID data. DP also requires careful management of the privacy budget (epsilon) [1].

Secure Aggregation. Secure aggregation ensures that the server sees only the sum of client updates, not individual ones. Protocols like Bonawitz et al. use random masking to achieve this even under client dropout [7]. SAFELearn further improves performance by reducing communication rounds and supporting flexible cryptographic backends [15], although this approach increases system overhead.

4.3 Cryptographic Enhancements

Secret Sharing. In FL, updates can be split into shares using Shamir's Secret Sharing, so that only a threshold number is needed to reconstruct the original [54]. This enhances fault tolerance but requires careful tuning of the threshold to strike a balance between security and availability.

Homomorphic Encryption. HE enables computation on encrypted updates, allowing aggregation without decryption. While it ensures confidentiality, Fully Homomorphic Encryption (FHE) is still impractical for large-scale FL, making Partial HE or hybrid schemes more viable [28, 54].

Secure Multi-Party Computation and Zero-Knowledge Proofs. SMPC and ZKPs enable secure aggregation and verifiable updates without revealing inputs [37, 58, 68, 71]. Although powerful, they are limited by significant computational and coordination overhead, especially at scale.

Certificateless Authentication. Certificateless schemes allow identity verification without central certificate authorities, using partial keys and user identities [67]. This reduces key escrow risks and enhances trust in decentralized FL settings without central bottlenecks.

4.3.1 *Architectural Considerations.*

System Architecture. FL systems may be centralized (with a single aggregator), decentralized (fully peer-to-peer), or hierarchical (with edge aggregators) [12, 49]. These choices affect both the privacy risk and the deployability of cryptographic tools, such as SMPC or secure aggregation.

Smart Contract-Based Incentives. Smart contracts automate rewards, enforce protocol rules, and track contributions in decentralized FL [66]. However, bugs in contract logic or role management may leak metadata or be exploited, undermining privacy. Their immutable nature makes proper design and auditing essential.

4.4 Smart Contract Vulnerabilities Relevant to BCT-FL

Smart contracts integrated into FL expose the system to various logic and design-level vulnerabilities. Based on OpenSCV [61] and Pishdar et al. [44], we identify six categories relevant to privacy in BCT-FL:

- **Control Flow Vulnerabilities:** These include issues such as unsafe external calls, reentrancy, and misuse of delegatecall, which allow malicious contracts to hijack or recursively trigger logic within an FL pipeline. In BCT-FL, such vulnerabilities can be exploited to manipulate the execution of reward distribution, skip validation checks, or inject malicious updates. For example, an attacker could exploit reentrancy to repeatedly claim rewards before internal state changes occur [16].
- **Exception Handling & Gas Logic:** Smart contracts that lack robust exception handling may silently fail when confronted with unexpected behavior, such as an out-of-gas error or an unhandled return value. In the context of FL, this could mean that a participant's contribution is skipped without notification or logged feedback, reducing the quality of aggregation and potentially revealing who did or did not contribute based on observed output discrepancies [62].
- **Reward & Incentive Issues:** Vulnerabilities in incentive mechanisms allow adversaries to perform actions like free-riding, where they receive rewards without contributing valid updates, or spoof participation to inflate earnings. These attacks degrade trust in the system and often rely on on-chain logs that inadvertently leak behavioral metadata such as participation frequency or timing information that can later be linked to user identity or data value [29].
- **Storage & Memory Exploits:** Improper handling of storage, such as leaving state variables public or failing to restrict write access, can expose or allow tampering with sensitive FL metadata. For example, attackers may observe update

hashes, learning rates, or training timestamps to infer data distribution or model sensitivity. In more severe cases, they could overwrite stored model parameters or participation logs to bias outcomes [6].

- **Access Control & Identity:** In FL, participants often have distinct roles—clients, validators, aggregators, that must remain isolated for system integrity. Weak access control, such as missing ownership checks or role verification, enables impersonation and privilege escalation. An attacker could pose as both a contributor and a validator to approve their own malicious updates, undermining model trustworthiness and user anonymity [17].
- **Bad Randomness:** Many FL operations rely on randomness for client selection, task allocation, or timing. When contracts use weak sources like block timestamps or block hashes, attackers can predict or bias the randomness to their advantage—e.g., by manipulating when to send transactions to increase their selection odds. This opens the door to Sybil attacks and biases the training dataset, reducing fairness and potentially revealing patterns in participation [11].

A summary of the key smart contract vulnerabilities relevant to federated learning, along with their implications for privacy and system integrity, is presented in Table 2.

Vulnerability Type	Examples / Subtypes	Potential Impact on FL Privacy Mechanisms
Control Flow Vulnerabilities	Reentrancy, Improper check of external call return values, Malicious fallback function, delegatecall misuse, Improper external locking	Enables recursive calls or logic hijacking, possibly leaking rewards, states, or disrupting coordination and aggregation workflows
Exception Handling & Gas Logic	improper exception handling, gas mismanagement	Silent failure of model updates, skipped logic branches, skewed aggregation or broken training pipelines
Reward & Incentive Issues	Unsafe credit transfer, missing token verification, spoofed participation, free-riding	Tampered or unfair reward distribution, exposure of participant contribution patterns, broken trust
Storage & Memory Exploits	Public state variables, overwritable mappings, misuse of arrays or storage slots	Tampering with model parameters, leaking metadata, compromising integrity of training state or behavior
Access Control & Identity	Weak ownership checks, role confusion, impersonation, ID leakage	Unauthorized updates, Sybil attacks, role abuse, loss of anonymity or data confidentiality
Bad Randomness	Predictable randomness sources (e.g., blockhash, timestamp)	Manipulation of participant selection, contribution timing or reward bias; enabling fairness and participation attacks

Table 2. Summary of Smart Contract Vulnerabilities and Their Impact on FL Privacy Mechanisms

5 VULNERABILITY RANKING FRAMEWORK AND DISCUSSION

To evaluate the severity of each smart contract vulnerability in blockchain-based federated learning (BCT-FL), this section introduces a structured risk scoring model inspired by the OWASP methodology as adapted by Ula et al. [59]. Each vulnerability is assessed across three dimensions:

- **Privacy Impact (I)**: The degree to which the vulnerability compromises privacy-preserving mechanisms in FL.
- **Exploitability (E)**: The likelihood of the vulnerability being exploited in practice.
- **Detectability (D)**: The difficulty of detecting the vulnerability using current tools and auditing practices.

Each dimension is scored from 0 (negligible) to 3 (severe), and the final *Composite Risk Score (CRS)* is calculated using a weighted formula:

$$\text{CRS} = 0.5 \cdot I + 0.3 \cdot E + 0.2 \cdot D$$

This yields a normalized score on a 0–3 scale that prioritizes privacy impact while also considering exploitability and detectability. Table 3 shows the scores and final risk levels.

Table 3. Composite Risk Ranking of Smart Contract Vulnerabilities

Vulnerability Type	I	E	D	CRS	Risk Level
Access Control & Identity	3	3	2	2.7	High
Reward & Incentive Logic	2	3	2	2.3	Med–High
Bad Randomness	2	3	1	2.1	Med-High
Control Flow	2	2	2	2.0	Medium
Storage & Memory	2	2	2	2.0	Medium
Exception & Gas Handling	1	2	2	1.5	Med–Low

5.1 Analysis and Discussion

Access Control & Identity (CRS: 2.7 ***High Risk****).*

- **Impact (3):** Access control flaws enable impersonation, Sybil attacks, and unauthorized data access, violating fundamental privacy guarantees in FL systems [**? ?**]. They directly compromise identity protection, allowing attackers to validate their malicious updates and increase the chance of inference attacks.
- **Exploitability (3):** These issues are commonly exploited in deployed smart contracts due to the frequent omission of strict role checks and reliance on address-based logic [20].
- **Detectability (2):** While some access control issues can be statically detected (e.g., using Slither), deeper privilege escalation or identity leakage is often missed without formal role verification tools[34].

Reward & Incentive Logic (CRS: 2.3 ***Medium-High Risk****).*

- **Impact (2):** These flaws primarily expose metadata (e.g., participation frequency, reward logs) rather than raw data, but enable behavioral inference and undermine fairness in FL [67, 70].
- **Exploitability (3):** Attacks such as free-riding or reward inflation can be easily executed by simulating or duplicating participation [29].
- **Detectability (2):** Although anomalous token flows may eventually be flagged, subtle metadata leakage or participation spoofing is not easily identified without private logging mechanisms[5].

Bad Randomness (CRS: 2.1 ***Medium-High Risk****).*

- **Impact (2):** Weak randomness (e.g., `blockhash`, `timestamp`) allows adversaries to manipulate selection, timing, or rewards, leading to fairness degradation and enabling Sybil attacks [11].

- **Exploitability (3):** These sources are easily accessible and predictable by miners or attackers; randomness manipulation has been repeatedly demonstrated in DeFi and gaming applications [45, 55].
- **Detectability (1):** The use of insecure entropy sources is relatively easy to detect through static analysis or code review[45].

*Control Flow (CRS: 2 **Medium Risk**).*

- **Impact (2):** Reentrancy and delegatecall misuse affect control flow integrity and can corrupt training, reward logic, or model state in BCT-FL systems [25]. They directly compromise role isolation, enabling attackers to impersonate multiple roles and manipulate validation outcomes, thereby amplifying the risk of biased aggregation and metadata exposure.
- **Exploitability (2):** These are well-documented vulnerabilities, but exploiting them requires transactional timing or interface flaws, which are not always present [**?**].
- **Detectability (2):** Detectable using symbolic execution tools like Oyente or Mythril, but complex inter-contract logic may bypass such static checks [8, 20].

*Storage & Memory Exploits (CRS: 2.0 **Medium Risk**).*

- **Impact (2):** Improper storage design (e.g., public variables, indexable arrays) may leak update metadata or allow tampering with model parameters, undermining privacy indirectly [6].
- **Exploitability (2):** Public blockchains expose contract state by design, making information leakage likely without deliberate obfuscation [61].
- **Detectability (2):** Memory-related bugs and leakage are often missed during audit unless explicitly modeled; access control to storage slots is rarely formally verified[52].

*Exception Handling & Gas Logic (CRS: 1.7 **Medium-Low Risk**).*

- **Impact (1):** While these issues don't typically leak raw data, they cause silent failures, skipped updates, dropped participants, or stalled aggregation rounds, leading to bias or incomplete learning [62].
- **Exploitability (2):** Adversaries can intentionally exploit gas limits or exception blindness in contracts to create denial-of-service or dropout conditions [20].
- **Detectability (2):** Whilst detection tools specifically for exception handling and Gas-related exceptions exist, these types of failures often go unnoticed unless comprehensive runtime monitoring and fallback mechanisms are implemented [2, 18].

From the table 3, it seems that Access control & Identity, Reward & Incentive Logic, and Bad Randomness vulnerability have the most significant impact regarding Privacy-Preserving Mechanisms in FL. These vulnerabilities not only facilitate direct attacks, such as impersonation, free-riding, or Sybil manipulation, but also expose behavioral metadata that can compromise participant anonymity and system fairness. While control flow, storage, and gas-related issues remain relevant, their impact on privacy is often indirect or context-dependent. The overall risk analysis confirms that safeguarding FL privacy can not be done by cryptographic techniques alone; it requires secure, formally verified smart contract design, careful management of on-chain metadata exposure, and runtime protections against silent failures. These findings provide a ranked view of which vulnerabilities demand the most urgent attention in FL deployments that rely on blockchain infrastructure.

6 EVALUATION

6.1 Validity & Alignment with current Frameworks

The proposed ranking framework is conceptually grounded in and thematically aligned with established classification schemes such as OWASP[41], Pishdar et al.[44], and OpenSCV[61]. OWASP's risk assessment methodology was utilized for the structure of the scoring system, particularly the use of Impact, Exploitability, and Detectability as core dimensions. However, the weights were adjusted to emphasize privacy, reflecting the unique concerns of FL, by assigning a higher value to impact over detectability. Pishdar et al.'s [44] taxonomy contributed empirically observed smart contract vulnerability categories, while OpenSCV [61] provided detailed, systematic coverage of lesser-known but relevant exploit types. Together, these sources shaped a taxonomy that is both rigorous and applicable to the privacy context of BCT-FL.

6.2 Research contribution

This paper makes two key contributions. First, it introduces a privacy-focused vulnerability ranking framework specifically for BCT-FL, one of the first structured efforts to assess smart contract risks through a privacy lens rather than general security. Second, it conceptually integrates vulnerability classifications with FL-specific mechanisms like secure aggregation, secret sharing, and differential privacy, revealing how incorrectly secured smart contracts can (in)directly compromise these privacy protections.

6.3 Implications for Developers and Researchers

The results highlight the importance of secure-by-design practices [63] in BCT-FL development, suggesting that developers should adopt formal verification methods for role enforcement and randomness generation, consider using Verifiable Random Functions (VRFs) [69] and threshold cryptography, and design smart contracts to minimize on-chain metadata exposure. For researchers, the framework provides a basis for extending analysis into empirical validation, testing, or simulation, encouraging more privacy-aligned smart contract architectures within federated learning ecosystems.

6.4 Revisiting Research Questions and Hypotheses

RQ1: The literature-based taxonomy and ranking framework developed in this study identify the most relevant smart contract vulnerabilities for BCT-FL as those that intersect with privacy-critical roles and data flows, particularly access control, incentive logic, randomness, and storage mechanisms. While access control, external calls, and data exposure strongly align with privacy risks in FL, the inclusion of randomness and exception-handling vulnerabilities, despite not involving direct data exposure, indicates that Hypothesis 1 is only partially supported. These findings suggest that the scope of relevant vulnerabilities extends beyond the original hypothesis, encompassing indirect threats that compromise fairness, anonymity, or aggregation integrity.

RQ2: These selected vulnerabilities were shown to compromise privacy by enabling identity misuse and role escalation (via access control flaws), leaking behavioral metadata (via reward and storage designs), and degrading the guarantees of secure aggregation and fairness (via randomness manipulation or exception handling). Hypothesis 2 is thus only partially supported. While improper access control emerged as the most critical category, unsafe external calls, though impactful, did not show the highest privacy consequences. Instead, vulnerabilities in randomness and incentive logic were more significant, indicating that privacy threats in BCT-FL arise from a broader set of mechanisms than initially hypothesized.

7 CONCLUSION

7.1 Contributions

This thesis examined the intersection of Federated Learning (FL) and Blockchain Technology (BCT), focusing specifically on how smart contract vulnerabilities affect privacy-preserving mechanisms. Through the Design Science Research Methodology (DSRM), a novel artifact was proposed: a privacy-oriented vulnerability ranking framework. The framework maps known smart contract vulnerabilities to FL mechanisms such as secure aggregation, differential privacy, and secret sharing. Informed by OpenSCV [61] and supported by empirical patterns from Pishdar et al. [44], the taxonomy emphasizes categories most likely to disrupt confidentiality, fairness, or anonymity in BCT-FL.

By prioritizing privacy impact over general exploitability, the framework offers developers and researchers a focused lens for evaluating risk in BCT-FL environments. The structured scoring and conceptual alignment with existing threat models contribute to bridging the gap between smart contract security and privacy engineering in federated learning systems.

7.2 Discussion

This paper initially focused on examining which smart contract vulnerabilities are most relevant to BCT-FL and how they affect privacy-preserving mechanisms. The findings show that while access control, external calls, and data exposure pose significant privacy risks, other categories, such as randomness manipulation and incentive logic, also play a critical role. These affect fairness, anonymity, and aggregation integrity, even without directly leaking data. As a result, both research hypotheses are only partially supported: access control is confirmed as a key concern, but the impact of external calls is less dominant than expected, and the broader set of vulnerabilities has greater implications for privacy than originally assumed.

7.3 Limitations

Several limitations should be acknowledged. First, the evaluation in this study is conceptual rather than empirical; the framework has not been tested in a live or simulated BCT-FL environment. This was a deliberate choice, given that BCT-FL systems are still in their early stages and lack widely adopted testing infrastructures. The aim was to build a clear and structured foundation that can guide future implementations. As outlined by Hevner et al.[21], conceptual evaluation through logical reasoning and literature alignment is a valid and rigorous form of assessment in design science, especially in early phases where real-world experimentation may not yet be feasible. Second, the analysis is centered on Ethereum-style smart contracts and may not fully generalize to other platforms like Hyperledger Fabric [3] or Polkadot[65]. Third, the vulnerability scoring is heuristic, relying on theoretically informed judgments rather than statistically calibrated risk metrics. Lastly, the cybersecurity landscape is rapidly evolving. New attack vectors, tooling, and architectural shifts may render some assumptions outdated. As such, the framework should be periodically revisited and adapted to reflect emerging threats and mitigation strategies.

7.4 Future Work

Future research can expand on this foundation in multiple directions. First, empirical validation of the framework, via case studies, testbeds, or simulations, would help assess its practical effectiveness and refine the ranking logic. Second, there is significant potential to develop tooling support for static or dynamic analysis of privacy-relevant vulnerabilities in smart contracts used within FL workflows. Third, adapting the framework across different blockchain ecosystems could improve its generalizability and expose platform-specific risks or advantages. Finally, as smart contracts increasingly automate sensitive coordination and reward processes, future work should explore regulatory and ethical implications, particularly the privacy risks of on-chain metadata exposure and their interaction with compliance frameworks such as GDPR [42] or HIPAA [31].

REFERENCES

[1] Gorka Abad, Stjepan Picek, and Aitor Urbieta. 2021. SoK: On the Security & Privacy in Federated Learning. *CoRR* abs/2112.05423 (2021). arXiv:2112.05423 https://arxiv.org/abs/2112.05423

[2] Elvira Albert, Jesús Correas, Pablo Gordillo, Guillermo Román-Díez, and Albert Rubio. 2020. GASOL: Gas Analysis and Optimization for Ethereum Smart Contracts. In *Tools and Algorithms for the Construction and Analysis of Systems*, Armin Biere and David Parker (Eds.). Springer International Publishing, Cham, 118–125.

[3] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolić, Sharon Weed Cocco, and Jason Yellick. 2018. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference (EuroSys '18)*. ACM, 1–15. https://doi.org/10.1145/3190508.3190538

[4] International Business Law Blog. 2023. Legal Challenges in Defining and Regulating Smart Contracts. https://ibl.law/legal-challenges-in-defining-and-regulating-smart-contracts/?utm_source=chatgpt.com Accessed 2025-06-25.

[5] Franziska Boenisch, Adam Dziedzic, Roei Schuster, Ali Shahin Shamsabadi, Ilia Shumailov, and Nicolas Papernot. 2023. When the Curious Abandon Honesty: Federated Learning Is Not Private. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroSP)*. 175–199. https://doi.org/10.1109/EuroSP57164.2023.00020

[6] William Boitier, Antonella Del Pozzo, Álvaro García-Pérez, Stephane Gazut, Pierre Jobic, Alexis Lemaire, Erwan Mahe, Aurelien Mayoue, Maxence Perion, Tuanir Franca Rezende, Deepika Singh, and Sara Tucci-Piergiovanni. 2024. Fantastyc: Blockchain-based Federated Learning Made Secure and Practical. arXiv:2406.03608 [cs.CR] https://arxiv.org/abs/2406.03608

[7] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2016. Practical Secure Aggregation for Federated Learning on User-Held Data. arXiv:1611.04482 [cs.CR] https://arxiv.org/abs/1611.04482

[8] Jie Cai, Jiachi Chen, Tao Zhang, Xiapu Luo, Xiaobing Sun, and Bin Li. 2025. Detecting Reentrancy Vulnerabilities for Solidity Smart Contracts With Contract Standards-Based Rules. *IEEE Transactions on Information Forensics and Security* 20 (2025), 3662–3676. https://doi.org/10.1109/TIFS.2025.3551535

[9] Zeju Cai, Jianguo Chen, Yuting Fan, Zibin Zheng, and Keqin Li. 2025. Blockchain-empowered Federated Learning: Benefits, Challenges, and Solutions. *IEEE Transactions on Big Data* (2025). https://doi.org/10.1109/TBDATA.2025.3541560 Cited by: 0; All Open Access, Green Open Access.

[10] Kar Yee Chai and Mohamad Fadli Zolkipli. 2021. Review on Confidentiality, Integrity and Availability in Information Security. *Journal of ICT in Education* 8, 2 (2021), 34–42. https://doi.org/10.37134/jictie.vol8.2.4.2021

[11] Krishnendu Chatterjee, Amir Kafshdar Goharshady, and Arash Pourdamghani. 2019. Probabilistic Smart Contracts: Secure Randomness on the Blockchain. arXiv:1902.07986 [cs.GT] https://arxiv.org/abs/1902.07986

[12] Edward Collins and Michel Wang. 2025. Federated Learning: A Survey on Privacy-Preserving Collaborative Intelligence. arXiv:2504.17703 [cs.LG] https://arxiv.org/abs/2504.17703

[13] Massimo Di Pierro. 2017. What Is the Blockchain? *Computing in Science Engineering* 19, 5 (2017), 92–95. https://doi.org/10.1109/MCSE.2017.3421554

[14] Yaga Dylan, Peter Mell, Nik Roby, and Karen Scarfone. 2019. Blockchain Technology Overview. *arXiv preprint arXiv:1906.11078* (2019).

[15] Hossein Fereidooni, Samuel Marchal, Markus Miettinen, Azalia Mirhoseini, Helen Möllering, Thien Duc Nguyen, Phillip Rieger, Ahmad-Reza Sadeghi, Thomas Schneider, Hossein Yalame, and Shaza Zeitouni. 2021. SAFELearn: Secure Aggregation for private FEderated Learning. In *2021 IEEE Security and Privacy Workshops (SPW)*. 56–62. https://doi.org/10.1109/SPW53761.2021.00017

[16] Gemini Staff. 2022. What Was The DAO? https://www.gemini.com/en-SG/cryptopedia/the-dao-hack-makerdao. Accessed: 2025-06-22.

[17] Eunsu Goh, Dae-Yeol Kim, Kwangkee Lee, Suyeong Oh, Jong-Eui Chae, and Do-Yup Kim. 2023. Blockchain-Enabled Federated Learning: A Reference Architecture Design, Implementation, and Verification. arXiv:2306.10841 [cs.LG] https://arxiv.org/abs/2306.10841

[18] Neville Grech, Michael Kong, Anton Jurisevic, Lexi Brent, Bernhard Scholz, and Yannis Smaragdakis. 2018. MadMax: surviving out-of-gas conditions in Ethereum smart contracts. *Proc. ACM Program. Lang.* 2, OOPSLA, Article 116 (Oct. 2018), 27 pages. https://doi.org/10.1145/3276486

[19] Blessing Guembe, Sanjay Misra, and Ambrose Azeta. 2024. Privacy Issues, Attacks, Countermeasures and Open Problems in Federated Learning: A Survey. *Applied Artificial Intelligence* 38, 1 (2024). https://doi.org/10.1080/08839514.2024.2410504 Cited by: 0; All Open Access, Gold Open Access.

[20] Niosha Hejazi and Arash Habibi Lashkari. 2025. A Comprehensive Survey of Smart Contracts Vulnerability Detection Tools: Techniques and Methodologies. *Journal of Network and Computer Applications* 237 (2025), 104142. https://doi.org/10.1016/j.jnca.2025.104142

[21] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. 2004. Design Science in Information Systems Research. *MIS Quarterly* 28, 1 (2004), 75–105. http://www.jstor.org/stable/25148625

[22] Qian Hu, Biwei Yan, Yubing Han, and Jiguo Yu. 2021. An Improved Delegated Proof of Stake Consensus Algorithm. *Procedia Computer Science* 187 (2021), 341–346. https://doi.org/10.1016/j.procs.2021.04.109 2020 International Conference on Identification, Information and Knowledge in the Internet of Things, IIKI2020.

[23] Jiyue Huang, Kai Lei, Maoyu Du, Hongting Zhao, Huafang Liu, Jin Liu, and Zhuyun Qi. 2019. Survey on Blockchain Incentive Mechanism. In *Data Science*, Xiaohui Cheng, Weipeng Jing, Xianhua Song, and Zeguang Lu (Eds.). Springer Singapore, Singapore, 386–395.

[24] Shashank Joshi. 2021. Feasibility of Proof of Authority as a Consensus Protocol Model. arXiv:2109.02480 [cs.DC] https://arxiv.org/abs/2109.02480

[25] Niclas Kannengießer, Sebastian Lins, Christian Sander, Klaus Winter, Hellmuth Frey, and Ali Sunyaev. 2022. Challenges and Common Solutions in Smart Contract Development. *IEEE Transactions on Software Engineering* 48, 11 (2022), 4291–4318. https://doi.org/10.1109/TSE.2021.3116808

[26] Aditya Kapoor. 2024. ML Approach: Algorithms, Real-World Applications and Research Directions. SSRN. https://doi.org/10.2139/ssrn.5021508

[27] Sameera K.M., Serena Nicolazzo, Marco Arazzi, Antonino Nocera, Rafidha Rehiman K.A., Vinod P., and Mauro Conti. 2024. Privacy-preserving in Blockchain-based Federated Learning systems. *Computer Communications* 222 (2024), 38–67. https://doi.org/10.1016/j.comcom.2024.04.024

[28] Hendra Kurniawan and Masahiro Mambo. 2022. Homomorphic Encryption-Based Federated Privacy Preservation for Deep Active Learning. *Entropy* 24, 11 (2022). https://doi.org/10.3390/e24111545

[29] Yunlong Lu, Xiaohong Huang, Yueyue Dai, Sabita Maharjan, and Yan Zhang. 2020. Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial IoT. *IEEE Transactions on Industrial Informatics* 16, 6 (2020), 4177–4186. https://doi.org/10.1109/TII.2019.2942190

[30] Eric S. Maskin. 2008. Mechanism Design: How to Implement Social Goals. *American Economic Review* 98, 3 (June 2008), 567–76. https://doi.org/10.1257/aer.98.3.567

[31] Scholas Mbonihankuye, Athanase Nkunzimana, and Ange Ndagijimana. 2019. Healthcare Data Security Technology: HIPAA Compliance. *Wireless Communications and Mobile Computing* 2019, 1 (2019), 1927495. https://doi.org/10.1155/2019/1927495 arXiv:https://onlinelibrary-wiley-com.ezproxy2.utwente.nl/doi/pdf/10.1155/2019/1927495

[32] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 54)*, Aarti Singh and Jerry Zhu (Eds.). PMLR, 1273–1282. https://proceedings.mlr.press/v54/mcmahan17a.html

[33] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2023. Communication-Efficient Learning of Deep Networks from Decentralized Data. arXiv:1602.05629 [cs.LG] https://arxiv.org/abs/1602.05629

[34] Marwa Mnasri, Afef Jmal Maâlej, and Mohamed Jmaiel. 2025. A systematic literature review on security testing of Ethereum smart contracts. *Blockchain: Research and Applications* (2025), 100314. https://doi.org/10.1016/j.bcra.2025.100314

[35] Bhabendu Kumar Mohanta, Soumyashree S Panda, and Debasish Jena. 2018. An Overview of Smart Contract and Use Cases in Blockchain Technology. In *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. 1–4. https://doi.org/10.1109/ICCCNT.2018.8494045

[36] Ervin Moore, Ahmed Imteaj, Shabnam Rezapour, and M. Hadi Amini. 2023. A Survey on Secure and Private Federated Learning Using Blockchain: Theory and Application in Resource-Constrained Computing. *IEEE Internet of Things Journal* 10, 24 (2023), 21942 – 21958. https://doi.org/10.1109/JIOT.2023.3313055 Cited by: 12; All Open Access, Green Open Access.

[37] Vaikkunth Mugunthan, Antigoni Polychroniadou, David Byrd, and Tucker Hybinette Balch. 2019. Smpai: Secure multi-party computation for federated learning. In *Proceedings of the NeurIPS 2019 Workshop on Robust AI in Financial Services*, Vol. 21. MIT Press Cambridge, MA, USA.

[38] Satoshi Nakamoto. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf.

[39] Ammar Odeh and Anas Abu Taleb. 2025. Federated Learning and Blockchain Framework for Scalable and Secure IoT Access Control. (2025).

[40] Menna Mamdouh Orabi, Osama Emam, and Hanan Fahmy. 2025. Adapting security and decentralized knowledge enhancement in federated learning using blockchain technology: literature review. *Journal of Big Data* 12, 1 (2025). https://doi.org/10.1186/s40537-025-01099-5 Cited by: 0; All Open Access, Gold Open Access.

[41] OWASP Foundation. 2021. OWASP Risk Rating Methodology. https://owasp.org/www-community/OWASP_Risk_Rating_Methodology Accessed: 2025-06-29.

[42] Adedoyin Tolulope Oyewole, Bisola Beatrice Oguejiofor, Nkechi Emmanuella Eneh, Chidiogo Uzoamaka Akpuokwe, and Seun Solomon Bakare. 2024. Data privacy laws and their impact on financial technology companies: a review. *Computer Science & IT Research Journal* 5, 3 (2024), 628–650.

[43] Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee and. 2007. A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems* 24, 3 (2007), 45–77. https://doi.org/10.2753/MIS0742-1222240302 arXiv:https://doi.org/10.2753/MIS0742-1222240302

[44] Mohammad Pishdar, Mahdi Bahaghighat, Rajeev Kumar, and Qin Xin. 2025. Major vulnerabilities in Ethereum smart contracts: Investigation and statistical analysis. *EAI Endorsed Transactions on Internet of Things* 11 (2025). https://doi.org/10.4108/eetiot.5120 Cited by: 0; All Open Access, Gold Open Access.

[45] Peng Qian, Jianting He, Lingling Lu, Siwei Wu, Zhipeng Lu, Lei Wu, Yajin Zhou, and Qinming He. 2023. Demystifying Random Number in Ethereum Smart Contract: Taxonomy, Vulnerability Identification, and Attack Detection. *IEEE Transactions on Software Engineering* 49, 7 (2023), 3793–3810. https://doi.org/10.1109/TSE.2023.3271417

[46] Youyang Qu, Md Palash Uddin, Chenquan Gan, Yong Xiang, Longxiang Gao, and John Yearwood. 2022. Blockchain-enabled Federated Learning: A Survey. *ACM Comput. Surv.* 55, 4, Article 70 (Nov. 2022), 35 pages. https://doi.org/10.1145/3524104

[47] K. M. Jawadur Rahman, Faisal Ahmed, Nazma Akhter, Mohammad Hasan, Ruhul Amin, Kazi Ehsan Aziz, A. K. M. Muzahidul Islam, Md. Saddam Hossain Mukta, and A. K. M. Najmul Islam. 2021. Challenges, Applications and Design Aspects of Federated Learning: A Survey. *IEEE Access* 9 (2021), 124682–124700. https://doi.org/10.1109/ACCESS.2021.3111118

[48] Montaser N.A. Ramadan, Mohammed A.H. Ali, Hadi Jaber, and Mohammad Alkhedher. 2025. Blockchain-secured IoT-federated learning for industrial air pollution monitoring: A mechanistic approach to exposure prediction and environmental safety. *Ecotoxicology and Environmental Safety* 300 (2025), 118442. https://doi.org/10.1016/j.ecoenv.2025.118442

[49] Omer Rana, Theodoros Spyridopoulos, Nathaniel Hudson, Matt Baughman, Kyle Chard, Ian Foster, and Aftab Khan. 2022. Hierarchical and Decentralised Federated Learning. In *2022 Cloud Continuum*. 1–9. https://doi.org/10.1109/CloudContinuum57429.2022.00008

[50] Maria Rigaki and Sebastian Garcia. 2023. A Survey of Privacy Attacks in Machine Learning. *ACM Comput. Surv.* 56, 4, Article 101 (Nov. 2023), 34 pages. https://doi.org/10.1145/3624010

[51] Fahad Saleh. 2020. Blockchain without Waste: Proof-of-Stake. *The Review of Financial Studies* 34, 3 (07 2020), 1156–1190. https://doi.org/10.1093/rfs/hhaa075 arXiv:https://academic.oup.com/rfs/article-pdf/34/3/1156/36264598/hhaa075.pdf

[52] Christoph Sendner, Lukas Petzi, Jasper Stang, and Alexandra Dmitrienko. 2024. Large-Scale Study of Vulnerability Scanners for Ethereum Smart Contracts. In *2024 IEEE Symposium on Security and Privacy (SP)*. 2273–2290. https://doi.org/10.1109/SP54263.2024.00230

[53] Muhammad Shayan, Clement Fung, Chris J. M. Yoon, and Ivan Beschastnikh. 2021. Biscotti: A Blockchain System for Private and Secure Federated Learning. *IEEE Transactions on Parallel and Distributed Systems* 32, 7 (2021), 1513–1525. https://doi.org/10.1109/TPDS.2020.3044223

[54] Zhaosen Shi, Zeyu Yang, Alzubair Hassan, Fagen Li, and Xuyang Ding. 2023. A Privacy Preserving Federated Learning Scheme Using Homomorphic Encryption and Secret Sharing. *Telecommunication Systems* 82 (2023), 419–433. https://doi.org/10.1007/s11235-022-00982-3

[55] Amritraj Singh, Reza M. Parizi, Qi Zhang, Kim-Kwang Raymond Choo, and Ali Dehghantanha. 2020. Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities. *Computers Security* 88 (2020), 101654. https://doi.org/10.1016/j.cose.2019.101654

[56] Jaibir Singh, Suman Rani, and Parveen Kumar. 2024. Blockchain and Smart Contracts: Evolution, Challenges, and Future Directions. In *2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS)*, Vol. 1. 1–5. https://doi.org/10.1109/ICKECS61492.2024.10616652

[57] Tao Sun, Dongsheng Li, and Bao Wang. 2023. Decentralized Federated Averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 4 (2023), 4289–4301. https://doi.org/10.1109/TPAMI.2022.3196503

[58] Xiaoqiang Sun, F. Richard Yu, Peng Zhang, Zhiwei Sun, Weixin Xie, and Xiang Peng. 2021. A Survey on Zero-Knowledge Proof in Blockchain. *IEEE Network* 35, 4 (2021), 198–205. https://doi.org/10.1109/MNET.011.2000473

[59] Munirul Ula, Rizal Tjut Adek, and Bustami. 2023. Vulnerability risk assessment using Open Web Application Security Project (OWASP) methodology for e-marketplace. *AIP Conference Proceedings* 2431, 1 (08 2023), 080011. https://doi.org/10.1063/5.0118272 arXiv:https://pubs.aip.org/aip/acp/article-pdf/doi/10.1063/5.0118272/18077924/080011_1_5.0118272.pdf

[60] M. Victoria Luzon, Nuria Rodriguez-Barroso, Alberto Argente-Garrido, Daniel Jimenez-Lopez, Jose M. Moyano, Javier Del Ser, Weiping Ding, and Francisco Herrera. 2024. A Tutorial on Federated Learning from Theory to Practice: Foundations, Software Frameworks, Exemplary Use Cases, and Selected Trends. *IEEE/CAA Journal of Automatica Sinica* 11, 4 (2024), 824 – 850. https://doi.org/10.1109/JAS.2024.124215 Cited by: 18.

[61] Fernando Richter Vidal, Naghmeh Ivaki, and Nuno Laranjeiro. 2024. OpenSCV: an open hierarchical taxonomy for smart contract vulnerabilities. *Empirical Software Engineering* 29, 4 (2024), 101. https://doi.org/10.1007/s10664-024-10446-8

[62] Heqiang Wang and Jie Xu. 2023. Combating Client Dropout in Federated Learning via Friend Model Substitution. arXiv:2205.13222 [cs.LG] https://arxiv.org/abs/2205.13222

[63] CYNTHIA WEBER and MARK LACY. 2011. Securing by design. *Review of International Studies* 37, 3 (2011), 1021–1043. https://doi.org/10.1017/S0260210510001750

[64] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang. 2023. A survey on federated learning: challenges and applications. *International Journal of Machine Learning and Cybernetics* 14, 2 (2023), 513–535. https://doi.org/10.1007/s13042-022-01647-y

[65] Gavin Wood. 2016. Polkadot: Vision for a heterogeneous multi-chain framework. *White paper* 21, 2327 (2016), 4662.

[66] Bijun Wu and Oshani Seneviratne. 2025. Blockchain-based Framework for Scalable and Incentivized Federated Learning. arXiv:2502.14170 [cs.LG] https://arxiv.org/abs/2502.14170

[67] Caihong Wu and Jihua Liu. 2024. Smart contract assisted secure aggregation scheme for model update in federated learning. *Computer Networks* 250 (2024), 110542. https://doi.org/10.1016/j.comnet.2024.110542

[68] Zhibo Xing, Zijian Zhang, Meng Li, Jiamou Liu, Liehuang Zhu, Giovanni Russello, and Muhammad Rizwan Asghar. 2023. Zero-Knowledge Proof-based Practical Federated Learning on Blockchain. arXiv:2304.05590 [cs.CR] https://arxiv.org/abs/2304.05590

[69] Shuang Yao and Dawei Zhang. 2022. An anonymous verifiable random function with applications in blockchain. *Wireless Communications and Mobile Computing* 2022, 1 (2022), 6467866.

[70] Yufeng Zhan, Jie Zhang, Zicong Hong, Leijie Wu, Peng Li, and Song Guo. 2022. A Survey of Incentive Mechanism Design for Federated Learning. *IEEE Transactions on Emerging Topics in Computing* 10, 2 (2022), 1035–1044. https://doi.org/10.1109/TETC.2021.3063517

[71] Wenxuan Zhao, Wei Mi, and Xiaodan Zhang. 2024. The Security Paradox of Smart Contracts: Blind Spots and Prospects of Current Detection Strategies. In *Proceedings of the 2024 27th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2024*. Institute of Electrical and Electronics Engineers Inc., 1546–1551. https://doi.org/10.1109/CSCWD61410.2024.10580546

[72] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. 2018. Federated Learning with Non-IID Data. *arXiv preprint arXiv:1806.00582* (2018). https://arxiv.org/abs/1806.00582

A APPENDIX A

During the preparation of this work, I used ChatGPT to give me ideas and help me with structuring sections of the thesis. After using ChatGPT, I thoroughly reviewed and edited the content as needed, taking full responsibility for the outcome.