Smart Contract Monitoring using Cloud Driven Technology

# Introduction

In this paper, we will present a review of how cloud computing can be interacted with on the blockchain. More specifically we explored the area of Product as a service. This brought several limitations such as security etc into play. We think this is a valuable field of research since it could offer more security and cheaper ways of storing data. Some other interesting parts include decentralized access to data and some real word use case research. With this paper we contribute to the field by researching hybrid systems and developing a prototype for a cloud to blockchain application.

# Literature Review

In this section I give an overview of the research I done for this project, what implementation of integrated blockchain and cloud services I found. I also discuss other peoples research and how it aided me over the course of this project.

Cloud computing interaction with blockchain is not a common interaction of services. I think it could be useful since it offers, an easier storage of data and way for people to interact with the blockchain if they don’t understand it. It offers several benefits like offering storage of data, databases, easier monitoring of the blockchain and so on. You can also host nodes themselves on cloud services

An interesting case study for a cloud system that integrates cloud computing is the IBM food trust. Under this system blockchain was used to enhance transparency and efficiency. It allowed for a secure way of tracking transactions and food products to avoid waste, fraud and to track origin of food products that spread diseases. This system also allows for the creation for a standard ledger that could be used by all members of a particular supply chain. How it works is by only giving permission to trusted participants to enter data, like farmers or shopkeepers. Then those shopkeepers instead of entering their data into a paper or excel ledger would enter it into the food trusts distributed ledger on the blockchain, allowing everyone easy access to view. [1]

In a paper called about blockchain based cloud computing by Bharathi Murthy et al. analyses how integrating cloud computing with blockchain technologies can be used to mitigate certain key limitations in the field of cloud computing by surveying previous papers that have been written on this subject. It looks at how things like privacy, data security and management could be improved through the blockchain. It explains each of the technologies in the great detail and proposes briefly a hybrid system using both technologies. This paper proved useful towards my proposed system cause it showed how I can combine the 2 systems. [2]

In a survey done by Jinglin et al. called “Integrated Blockchain and Cloud Computing Systems: A Systematic Survey, Solutions, and Challenges” they explore the combination of blockchain and cloud computing systems. It is focused on how hybrid system works the different combinations and improvements to security. It also explores other subjects such as consensus protocols, recent industry trends and different models. It also explores different gaps in research and some challenges such as performance issues, the energy costs and compliance with different regulations. This survey helped me explore different models of blockchain and cloud computing models that I could implement into my own research, as well as some interesting implications of it. [3]

An interesting article by the Swiss cyber institute talks about how blockchain technologies can enhance security in cloud environments. This article talks about how blockchains security systems such as immutable blocks and consensus protocols could be used to provide additional security. It also talks about how its decentralized nature allows for greater resistance against failure or attacks. What I gained from this article was the benefits of security blockchain systems provide [5].

I also looked into other poterntial applications of blockchains to give me an idea of how to best apply my project. I looked into an article to better understand the benefits of an integration of the technologies could provide for scalability and security. [6]

How I plan to imply the research I done into the project I’m creating is by looking through this and choosing what particular aspects from each paper I want to apply. How I am going to integrate the traditional cloud system and see how I can apply it to my project.

## Similar Projects

Filecoin is a decentralized storage network designed to store information. How it works is a user can rent out space on their computer for other users. This effectively transforms the users device into a cloud storage center. It is a cloud alternative to centralized solutions like AWS. The big issue with this type of solution is that after a user stops mining/storing data the data is lost. To solve that issue there is a degree of redundancy with multiple copies, but this just causes additional overhead.

Arweave is another decentralized storage network designed to store information.

Stori

Amazon Web Services (AWS) Blockchain Solutions

# System Architecture

In this section I give an overview of the technologies I interacted with for this project. I give a high level overview of how the project was set up. I explain each of the individual components that were used to implement the project and its technologies, I also include a diagram to aid in the explanation of this system.

## Technologies

### Blockchains

A blockchain is a type of a distributed digital ledger, it is comprised of “blocks” these blocks store information such as transactions, people involved, rewards for creating blocks and other information for security or building programs. The key concept of blockchains is that they are decentralized unlike traditional cloud systems they are not stored in a specific location instead they could be hosted on thousands of machines across the world connected through the internet.

The blockchain technology I will be using for this project will be Ethereum.

### Smart Contracts

Smart Contracts is a concept in distributed ledger technology where 2 parties can agree on a certain action that will be executed. It is implemented as a piece of code that self executes once a certain criteria is met. This is how you create functions and implement applications on the blockchain. To implement this concept I will write code on the Ethereum virtual machine using solidity.

### Cloud Computing

Cloud computing is the process of delivering certain useful services over the “cloud”, which in this case the cloud refers to the internet. There are 3 main categories for cloud services. Infrastructure as a service(IaaS) which means it provides resources/infrastructure over the internet. Platform as a service(PaaS), it provides vital services for development like development environments over the cloud and software as a service(SaaS), this means they Provide user applications over the internet like office 365 or games. For this project I’m going to focus on the concept of IaaS.

IaaS essentially allows you to outsource resources, such as RAM, Processing power or secondary storage. This allows you to safe your time and resources on setting up your equipment and in essence just allows you to rent out others people hardware and receive the output or input through the internet. For this purpose, I will be using AWS servers.

## Overview of my system

The system I propose is simple system used to specific transactions and events done by players on the blockchain and then those events get stored on a cloud database. How this works is for example a player buys a sword in an online game and then that transaction is done through the blockchain and the data saved. This is especially useful in case of data being lost due to several reasons in the real world or a hacker stealing user data. The advantage of this system is that no matter what happens to the database stored by the company, also since this system allows purchases of items the creation of blocks could be funded by players spending money. Also this could potentially give another way to bring in revenue since you could tax transactions between these items.

The purpose of this system is to investigate how you can use the cloud to aid in the monitoring of events that take place on the blockchain. This could be useful since the blockchain itself is more difficult to interact with the data than a simple database.

How this system works is by first creating a smart contract using solidity that would be deployed onto the Ethereum testing blockchain called Seploia. Solidity is a high level language used to interact with Ethereum code. This smart contract has the ability to track what players own which items, which allows a secure and reliant way of storing information about the game state. This smart contract was deployed using infura, this basically gives you a node where you can connect to the blockchain. If a user made a transaction using this smart contract, a listener function written in JavaScript would pick up on it and send this update to the table stored on AWS. Then the game code could pick up on changes from this table and updates the game state.

Each component used in this project:

* Cloud Server: This is where the data can be stored or retrieved from. Examples of these include AWS or Azure
* Blockchain: Here is the system where the underlying Ethereum is.
* Smart Contracts: They are hosted on the blockchain, these are programs where the users can interact with the blockchain or different data.
* Web3 programs: Here the user can use as a point to interact with blockchain from.
* UI: Here is the actual interface the user interacts with
* Lamba function: This listens out for the changes on the blockchain.

## Architecture diagram

A diagram of a computer game

AI-generated content may be incorrect.FIGURE 1. The proposed system architecture

A – User purchases and item using Ethereum blockchain.

B – The lambda listener function catches that a new event fired

C – The lambda function updates the AWS database table to reflect the new purchase.

D – The game checks and sees a new entry meaning the game state should be updated.

E – The player can now use their new item

## Advantages of this system

Blockchains themselves offer enormous security benefits but have many drawbacks such as performance issues. This system bridges that gap. You can easily query, search and display what items belong to who which are listed for sale etc. using the cloud database, while still having the security provided by the blockchain. This ensures that each item can’t be duplicated, your system will by more fraud proof as well, as if your database will get wiped due to an error you can easily work your way through the blockchain and rebuild your system.

# Experiment setup and performance evaluation

In this section I will walk you through how I set up a partial implementation of the system proposed. I will go step by step through how I set up the experiment and then I will evaluate the results, whether it was successful and the fees paid for each transaction.

So for this experiment I created a simple program that checks whether it would be possible to create a smart contract tracking game items and then having them updated on an AWS server. I didn’t focus on the AWS to game aspect of it since it would be much simpler to implement.

Also in this experiment I didn’t place any owner address since I assumed the players account would be linked to their eth address.

## Experiment setup

How this project was set up was through several steps:

1. Created a smart contract where a developer can create Items, players can then purchase each Item from the developer or from each other. They could also set their own prices so the players can make profit, also the developers would take a portion of these earnings as a fee. Code Listing A shows this function
2. I tested out the smart contract using foundry to see if it’s functioning properly.
3. I created a project-id using infura so I could connect to an Sepolia node.
4. Deployed the contract using my MetaMask account, infura and foundry.
5. Created a table on DynamoDB on AWS services called itemEvents that tracks each item on the smart contract.
6. Created a lambda function in JavaScript to listen out for events omitted by my smart contract and update the cloud table. The function was deployed using ether library. Code Listing B shows this function.
7. The system was tested to see if it works properly.

## Evaluation

I managed to achieve several functionality goals:

* The smart contract was successfully deployed onto the blockchain.
* The event listener successfully detects itemCreated() events.
* The table was successfully updated after an item was created.
* The system doesn’t miss any events when it is actively listening.
* The system takes under 6 seconds to go from emitted event to updated table.
* Overall, the experiment proof to be a success

# Discussion

## Limitations

## Cost Constraints

Implementing an additional system on the blockchain for security could add additional costs related to the users spending extra money on gas fees instead of on the game store.

## Privacy

This system does come at the loss of the users’ transactions being visible. So other people could view how much they are buying, what they’re buying and when. To some users this could prove like a too big of an invasion of privacy stopping them from interacting with the system.

### Variance in Cost

Due to a lack in a centralized entity controlling the price of coins, the value of your items could vary drastically day to day therefore your income would vary a lot over time even if your sales remain the same.

## Possible Additional Features

Cloud Gaming

As the internets speeds continue to improve, it has become increasingly popular to host games on the cloud. You could apply this system to pay for resources on the cloud instead of a subscription model using local currency like is common place now. This would involve a similar technology like in the proposed system.

NFTs

A user could get NFTs as well with each sold item or for each participated event or for getting achievements. This NFTs could be traded as well and the platform could take fees for each transaction.

# Conclusion

In this I presented a model for gaming system interacting and taking advantage of both the blockchain and cloud services. It solves issues of things like security and fault resistance in products that could be wiped out by a faulty server or stolen if not secure. The experiment performed showed that the proposed system is feasible. This research also opens the door to other potential areas that this system could be applied to such as aiding supply chain systems and other areas. Overall the findings have the potential to aid future cloud services to be more secure.

# References

const { ethers } = require("ethers");

const AWS = require("aws-sdk");

// AWS Regon setup

AWS.config.update({ region: "eu-north-1" });

// DynamoDB client

const dynamoDb = new AWS.DynamoDB.DocumentClient();

// Ethereum provider (Sepolia via Infura)

const provider = new ethers.JsonRpcProvider("XXXXXXXX");

// Contract address and ABI

const contractAddress = "0x9d58134Dd3fba0B3dB67264E73195186E28BafA7";

const abi = [

    "event VaultAdded(uint256 vaultId, address owner)"

];

// This section listens for sepolia vaultAdded events

const contract = new ethers.Contract(contractAddress, abi, provider);

console.log("Searching for events");

contract.on("VaultAdded", async (vaultId, owner) => {

    console.log(`VaultAdded event: Vault ID ${vaultId.toString()}, Owner ${owner}`);

    const params = {

        TableName: "UserVaults",

        Item: {

            vaultId: vaultId.toString(),

            ownerAddress: owner

        }

    };

    // Store the event in DynamoDB or error handling

    try {

        await dynamoDb.put(params).promise();

        console.log("Successfully stored event in DynamoDB:", params.Item);

    } catch (err) {

        console.error("Error with storing data", err);

    }

});

[1] High, M. (2020). *Supply chain insight: Inside IBM’s Food Trust Blockchain system*. [online] supplychaindigital.com. Available at: <https://supplychaindigital.com/technology/supply-chain-insight-inside-ibms-food-trust-blockchain-system>

[2] Bharathi Murthy, Ch.V.N.U., Lawanya Shri, M., Kadry, S. and Lim, S. (2020). BLOCKCHAIN BASED CLOUD COMPUTING: ARCHITECTURE AND RESEARCH CHALLENGES. *IEEE Access*, pp.1–1. doi:https://doi.org/10.1109/access.2020.3036812.

[3] Jinglin Zou, Debiao He, Sherali Zeadally, Neeraj Kumar, Huaqun Wang, and Kkwang Raymond Choo. 2021. Integrated Blockchain and Cloud Computing Systems: A Systematic Survey, Solutions, and Challenges. ACM Comput. Surv. 54, 8, Article 160 (November 2022), 36 pages. <https://doi.org/10.1145/3456628>

[4] Fadele Ayotunde Alaba, Hakeem Adewale Sulaimon, Madu Ifeyinwa Marisa, Owamoyo Najeem. Smart Contracts Security Application and Challenges: A Review. Cloud Computing and Data Science [Internet]. 2023 Sep. 1 [cited 2025 May 16];5(1):15-41. Available from: <https://ojs.wiserpub.com/index.php/CCDS/article/view/3271>

[5] Swiss Cyber Institute. (2021). *What is Blockchain Security in Cloud Computing?* [online] Available at: https://swisscyberinstitute.com/blog/blockchain-security-in-cloud-computing/.

‌[6] WebClues Infotech (2025). *Blockchain Meets Cloud Computing: A Powerful Integration for Scalability & Security ☁️🔗*. [online] Medium. Available at: https://medium.com/coinmonks/blockchain-meets-cloud-computing-a-powerful-integration-for-scalability-security-%EF%B8%8F-331bb33cd88b [Accessed 15 May 2025].

‌

‌

‌