National College of Ireland

SimCloud: A Lightweight Framework for Cloud Trust Evaluation

MSc Research Project
MSc in Cloud Computing

Balaji Srikanthan
Student ID: 23413263

School of Computing
National College of Ireland

Supervisor: Aqeel Kazmi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Balaji Srikanthan
Student ID:	23413263
Programme:	MSc in Cloud Computing
Year:	2025
Module:	MSc Research Project
Supervisor:	Aqeel Kazmi
Submission Due Date:	11/12/2025
Project Title:	SimCloud: A Lightweight Framework for Cloud Trust Evaluation
Word Count:	9259
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Balaji Srikanthan
Date:	10th December 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

SimCloud: A Lightweight Framework for Cloud Trust Evaluation

Balaji Srikanthan
23413263

Abstract

Reliability, resilience, and sustainability in the selection of cloud services has gained more prominence with the ability to manipulate Quality of Service (QoS) values. The available trust models are typically based on predetermined historical measures, do not provide a system to respond to real-time misreporting, and are not very transparent, which is not suitable when it comes to lightweight and reliable decision-making. To overcome these constraints, this paper presents SimCloud, a lightweight trust system that would improve the evaluation of cloud services by addressing these limitations: an explainable trust scoring model to provide clarity in the decision logic, a green energy component to include the element of sustainability, A time based trust deterioration component to downgrade stale Quality of Service results and also for the identification of anomalies and penalising services with manipulated service reports. SimCloud has been tested in both normal and hostile circumstances, like single-service manipulation, QoS aging, and cohort-level attacks. The findings indicate that SimCloud is predictable in its ranking accuracy, consistent in its ability to detect manipulated QoS submissions, suitable in reprimanding outdated information, and it is lightweight, interpretable, and reproducible. These results show that SimCloud reinforces trustworthiness, resilience and sustainability in cloud service ecosystems as opposed to traditional trust models.

Quality of Service, SimCloud, transparency, evaluation, Framework, Trust scoring, resilience

1 Introduction

Cloud computing has become the foundation of the contemporary digital infrastructure, it is scalable, on demand, and cost-effective services that are offered to the users all over the world. Although the latter has many benefits, reliability and trust in service selection are critical issues to be tackled especially when it comes to users having to select one of the multiple providers that offer different Quality of Service (QoS) levels. Reliability of the cloud services such as the consistency of performance, the trust of the users and the sustainability of the cloud services largely revolves around the overall manner in which the trust is measured and sustained in these dynamic environments.

The existing trust-based models, however, probably can not handle the dynamic nature of cloud ecosystems. They are inclined to use a lot of fixed historical information and cannot keep up with the real-time misreporting or changes in QoS metrics. What is more, most of the existing models are not transparent and, thus, users and administrators have a hard time explaining or making sense of decisions based on trust.

Such constraints make cloud systems vulnerable to QoS based attacks, manipulation of trust data and decreased confidence levels by the users.

1.1 Research Question

What is the effectiveness of a lightweight explainable trust assessment system to enhance the reliability and transparency of cloud services, when QoS attacks are launched, with the element of sustainability and anomaly detection in consideration ?

1.2 Objective

This paper presents SimCloud, a small, explainable, and trust-based system that aims to assess the trustworthiness of cloud Software-as-a-Service (SaaS) solutions by providing a simulated system. In contrast to the traditional models or black-box machine learning algorithms, SimCloud is focused on interpretability, flexible, and sustainable by four modules:

1. Explainable Trust Scoring - This model is designed to make sure that no trust decision can be made without being transparent and audited by an administrator.

2. Green Energy Scoring - Incentivizes energy efficient cloud services through implementation of sustainability measure using renewable energy consumption.

3. Time-Based Trust Decay - This method is based on the principal that the latest service behavior should be used as a basis, rather than historical information, to ensure that the trust scores will be shown as a reflection of the existing reliability of services.

4. Anomaly Detection - Makes use of lightweight rule-based checking to detect and penalize falsified or abnormal QoS reports on the basis of abrupt changes in metrics, values that are inconsistent or of statistical outlier.

   When put together, these modules recreate dynamics of service selection in the real world to so much as to create a framework that can be interpreted. SimCloud is guaranteed to be reproducible and allow controlled benchmarking of both normal and adversarial settings, due to the use of a purely simulated dataset. In conclusion, the study will help to increase trust assessment, counter the resilience against QoS attack, and sustainability in cloud service management.

2 Related Work

The issue of trust-based selection of cloud services has been the subject of significant interest over the last several years as cloud systems continue to grow both in scale and in heterogeneity and complexity of operations. Previous studies have addressed reliability, quality-of-service minimization, trust modeling, anomaly detection and resource management with sustainability in mind. Nonetheless, explainability, tendency, sustainability and resistance to QoS manipulation are not adequately focused on in existing solutions. This section reviews the present in five applicable dimensions such as Anomaly detection, time-based decay,green-scoring, trust models which reveals research gaps

that are especially critical to address, and as such, the proposed SimCloud, a lightweight, transparent, and attack-resilient simulation framework, is developed.

2.1 Trust-Based Models for Cloud Service Selection

Conventional methods of trust evaluation are usually based on QoS history, reputation scores or even cumulative measures of performance. Similarly a trust based service selection model proposed by Pan et al. (2015) was based on the past QoS performance. Although it works with honest reporting, the model is based on reliable QoS reporting and does not work well when used in a dynamic or adversarial context. Equally Zhu et al. (2023) proposed a Lyapunov-based optimization approach to trade-off between trust and power-performance in the context of heterogeneous SaaS. Despite this power, such formulations are mathematically very complicated and cannot be practically implemented in light-weight systems with a huge bias in their computational cost. SimCloud, in its turn, is designed to prevent such historical overfitting and computational overheads as urged by the lightweight system of trust calculations.

2.2 Explainable Trust Evaluation

Trust decision explainability is an important element since cloud providers and users need elements such as transparency and auditability for increased integrity. According to Gao et al. (2021) CloudTrustLens is an XAI-oriented trust management system that uses SHAP-based post-hoc explanations. Though XAI can help increase transparency, such mechanisms of explanation need a lot of computational resources and are going to slow down the decision making pipeline which is again not suitable for lightweight systems. SimCloud handles this with methods that are inherently understandable, like tree based decision logic, in which explanations are inherent in model structure. In this design , The SHAP overhead is avoided and includes transparent and easy to digest insights into trust, therefore, making SimCloud a more reliable choice in Saas Environments that require transparency and speed .

2.3 Trust Evaluation based on Time

The behavior of cloud services is highly temporal, the performance may diminish or improve over time and therefore the performance of the cloud service cannot be effectively modeled using a static or historically prejudiced trust model. Zhang et al. (2022) suggested a time-sensitive trust framework whereby the exponential decay reduces the effects of obsolete observations of QoS. Despite the effectiveness, their solution incorporates blockchain infrastructure, which adds too much complexity and restricts the realm of deployability. SimCloud builds upon the basic concept of recency-based weighted trust but uses a lightweight decay model, which does not need blockchain or intensive cryptographic processes. This guarantees that trust scores are always up to date and context-sensitive.

2.4 Sustainability in Service Evaluation

As the focus on green computing has increased, sustainability tactics have come in the field of cloud service management. Yang et al. (2021) suggested a power-efficient service selection algorithm that incorporates the QoS and power-aware approaches. Nevertheless,

the majority of currently present sustainability models are resource planners as opposed to trust metrics that is to say that service trust decisions have no direct relation to eco-efficiency. SimCloud is different in that it incorporates a Green Energy Scoring module in its trust computation pipeline so that the factor of sustainability in service selection is included. This does not only promote environmentally friendly service behavior, but also meets the current demands of environmental responsibility as far as trust assessment is concerned.

2.5 Explainable Trust Evaluation

Modified or manipulated Quality of Service reports can be used to compromise cloud service trust assessment systems and can have a misleading impact on service-selection mechanisms. Anomaly detection systems that are based on machine learning models and statiscial methods are discussed in the earlier chapters. Mao et al. (2021) predicted trust using neural-network ensembles, whereas Nanekalva and Velou (2025) and Hagemann and Katsarou (2020) in their reviews identified a variety of ML-based and statistical detectors. They are efficient, but can be computationally intensive and do not have properties of trust sensitivity necessary in trust sensitive environments. SimCloud, on the contrary, uses a lightweight, entirely rule-based anomaly detection unit that relies on various methods such as rules of statistical deviation, threshold checks , statistical deviation rules and consistency checks based on cross metrics to detect and penalize manipulated cloud services.

3 Methodology

This chapter is the description of the methodology of the design, implementation, and evaluation of the SimCloud framework. The project has an experimental, simulation-based approach, which focuses on transparency, reproducibility, and explainability. This general work cycle is an iterative research process design, simulate, measure, and refine; this has the advantage that every part of the trust-evaluation pipeline can be tested in controlled settings. The following subsections elaborate the research design, methodology of internal framework, and data processing pipeline, and the approach applied to test the system in normal and adversarial QoS environment.

3.1 Research Design and Approach

The research is based on a practical simulation-applied study to develop a reproducible and interpretable framework of evaluating trust in cloud services. The main hypothesis that the accuracy and robustness of decisions in event of fake QoS submissions and without the need of computationally costly machine-learning methods. In order to test this hypothesis, a controlled simulation environment in the form of the cloud was implemented.This environment creates Quality-of-Service (QoS) values of various virtual cloud services enabling the accurate manipulation of latency, availability, throughput, and energy-consumption values. Other intentional adversarial behaviours introduced by the simulator are artificial spikes in performance, inconsistent QoS reports, and sudden behavioural changes. This controlled design provides the repeatability of the experiment and excludes noise in deployment. Any computation of trusts and any anomaly detection is stored in orderly logs that can be analyzed strictly in a variety of autonomous runs of

the simulation. This design type facilitates the progressive assessment of the behaviour of SimCloud in the presence of varying, unpredictable and malicious conditions of the QoS.

3.2 Framework Methodology

SimCloud is introduced as a trust-evaluation pipeline, which takes as input simulated QoS data and processes these data according to four trust modules and then aggregates the outputs to rank cloud services. The methodology starts with a QoS Simulation Engine which produces artificial Quality-of-Service values of every cloud service at each timestep. These indicators are latency, availability, throughput, energy usage, response time and reliability. The simulator is also an adversarial system capable of generating the artificial conditions of sudden spikes, unrealistic drops, inconsistent behaviour with multiple metrics, and falsified QoS reports on purpose. The design is focused on transparency, little computational requirements, and complete reproducibility. The framework uses direct rule-based logic and incremental trust adjustments and anomaly checks to simulate realistic trust-based service selection. There are four independent trust modules that evaluate the data. The Explainable Trust Scoring Module uses rule-based reasoning that is directly related to QoS behaviour. Sustainability is included in the Green Energy Scoring Module which gives positive trust increments to those services that use more renewable-energy. This is the current focus of cloud operation that pays much attention to the environment. The Time-Based Behaviour Module modifies trust in small steps little by little by making small positive or negative adjustments based on the recent performance. SimCloud does so in practice, as opposed to formal models of decay. The Anomaly Detection Module is completely rule based and detects the suspicious or manipulated QoS reports through unrealistic values and irregularities amongst the various measures involved. In case of an anomaly that has been identified, SimCloud imposes a first-order penalty on the trust score of the service and logs the cause. The result of all the four modules is added and subtracted in a simple manner to give a final trust score at every cycle of simulation.

3.3 Collection and data Processing

SimCloud operates a completely synthetic data collection pipeline with the aid of QoS Simulation Engine. Since the system is not based on any external datasets or live cloud traces, all the QoS behaviour, both normal and attack cases, is realised inside of the system. This enables complete manipulation of the characteristics of the data and ensures inter-experimental reproducibility. The processed values are then inputted into the trust modules which do not need any form of transformation except simple type conversion. Any intermediate values such as inputs to the QoS, trust increments, penalties, anomaly flags and end-state trust scores, are written to structured CSV files. Every log record has a date and a description of the reason why a trust adjustment took place, and the decision process is clear and traceable. This light weight collection and processing loop does not need the complexity of real-world monitoring stacks, even though it has sufficient variation and noise, and adversarial behaviour to test the robustness and stability of the trust model.

3.4 Testing and evaluation Methodology

The testing approach focuses on testing the behaviour of SimCloud in controlled simulation conditions. To measure correctness, robustness and stability, three types of tests were performed. The first category involves the evaluation of normal-condition, when services are characterised by reasonable and stable QoS behaviour. These tests ensure that SimCloud generates consistent trust tracks, prefers to stay consistently dependable services, and does not overreact to small changes. It is aimed at ensuring that the logic of trust is predictable in the absence of adversarial manipulation.

The second type assesses attack and anomaly cases. In this case, the simulator feeds in unnatural values of QoS including artificial bursts in availability, physically infeasible drops in latency, incompatible QoS curves or high-frequency oscillations. These tests evaluate whether the rule-based anomaly module is working correctly in that it is able to detect suspicious data, impose penalties and ensure that compromised services are not chosen..

The last category is that of the trust-score dynamics, which evaluates the dynamics of trust over time when simulation is done over a long period of time. It involves monitoring changes in trust as a service gets better or worse, investigating whether the trust is restored after a spell of bad performance, checking whether there is any green-energy scoring effect in the situation where services are otherwise similar in terms of QoS but different with respect to their sustainability. All the findings such as time-series trust plots, anomaly logs and selection histories were exported in CSV files and displayed on charts. This guarantees the behaviours of the system to be transparent, reproducible and have a direct relationship to the underlying QoS inputs.

4 Design Specification

SimCloud framework is designed using a lightweight, modulararchitecture that is expected to model the behaviour of cloud services and testthe trust under normal and adversarial QoS conditions. Each subsystem is aseparate entity that is used to guarantee transparency, reproducibility, andsimplicity of experimentation. Its architecture focuses on explainability andcontrolled simulation instead of computational complexity and makes the wholesystem run as a fully explainable trust-evaluation environment. The systemarchitecture, interactions between the internal parts, and the justification ofthe design decisions will be presented in the following subsections.

4.1 System Architecture Overview

The general architecture of SimCloud comprises of fourfundamental layers adding up to an entire trust-evaluation and simulationworkflow. The bottom layer is the QoS Simulation Layer which is an artificialperformance data generation of various cloud-like services. These are latency,availability, throughput, stability in response-time, and energy-consumptionpatterns. Adversarial conditions also are inoculated by the simulator by suddenspikes, unrealistic drops or inconsistent behaviour of the QoS, and theframework can be tested on adversarial conditions. The Preprocessing and Normalisation Layer lies above the simulation layer, andit carries out simple cleaning functions such as smoothing noisy data withshort-window averaging, scaling the measures and values to a similar range, andfilling in a blank record with the latest valid value. This will make sure

thatevery trust module obtains reliable and valuable data on QoS. The Trust Evaluation Layer is the central analysis point inthe framework. It comprises of four modules which are independent:

- The Explainable Trust Scoring Module, which uses rule-based scoring, which is based upon QoS behaviour, directly;
- The Green Energy Scoring Module, which gives positive increments to services that have shown more use of renewable-energy;
- The Time-Based Behaviour Module, which uses small upward or downward changes to indicate gradual changes in performance;
- The Anomaly Detection Module that applies rule-based tests to indicate unrealistic or even conflicting QoS submissions.

4.2 Workflow of Components

SimCloud is a timestep-driven workflow that executes in continuous mode where the received simulated QoS data initiates trust analysis and service selection. QoS Simulation Engine generates different values of synthetic performance of all the simulated cloud services at every cycle. Such raw inputs are directly sent into the preprocessing layer, which cleans and normalises the raw inputs.

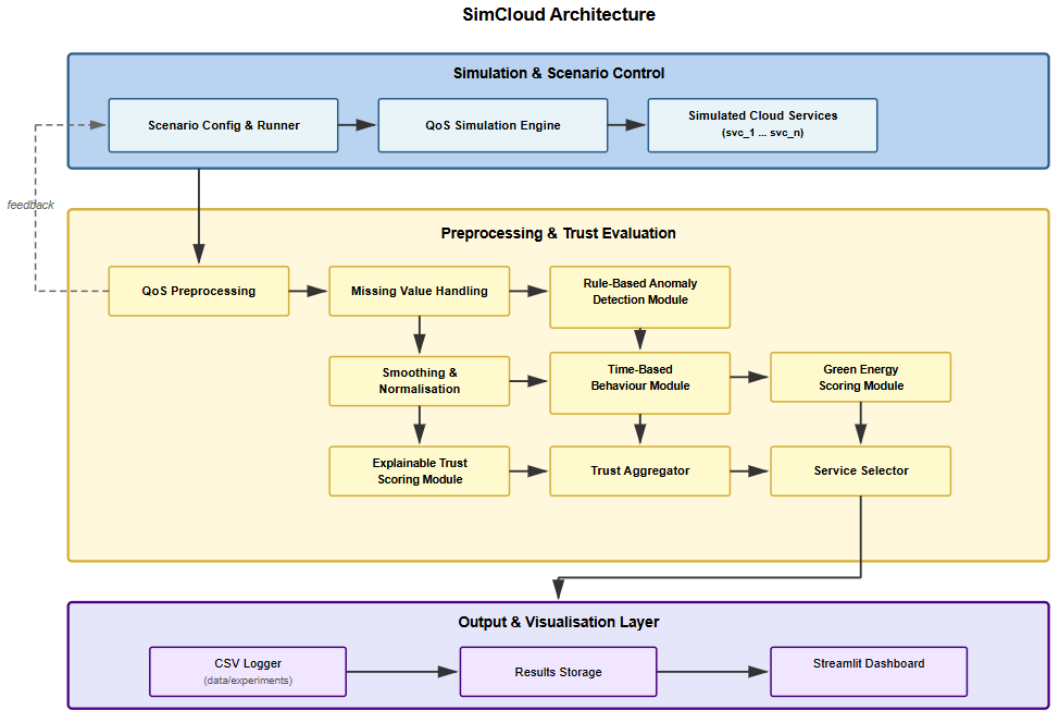

Figure 1: Architecture of the SimCloud Simulation Framework

The values are then spread to the four trust modules. The Explainable Trust module modulates the scores depending on the behaviour of QoS like lower latency or worse availability. Green Energy module weighs the use of renewable energy and adds increment

of extra increments to it when the sustainability is high. Time-Based module adjusts gradually based on the fact that a service is getting better or worse with the passage of several cycles. Anomaly Detection module verifies unrealistic QoS submissions; in the event that it occurs, a penalty is defined and a human-readable explanation stored.

Their outputs are inputted into the trust aggregation component each module is finished in, and a simple final trust score is calculated about every service. A service whose trust score is the highest is picked as the recommended service of that cycle. The simulation, preprocessing, trust evaluation and selection process is repeated and this allows long experiments to be performed where the trend of trust and system behaviour over different QoS conditions can be observed.

This is a closed-loop trust ecosystem of workflow where each decision can be traced and replicated. Due to its independent modules, SimCloud has distinct separation of concerns and makes behaviours like stability, anomaly response, sustainability scoring and temporal adjustment readable and understandable in isolation.

4.3 Design Rationale

The architectural choices made in SimCloud were informed by three key goals, namely lightweight operation, explainability, and reproducibility. The system entails rule-based scoring and direct trust adjustments that do not use many resources. This allows the framework to operate long simulation cycles and also allows behaviour to be predictable and interpretable. The issue of explainability was also a major requirement. All trust adjustment, penalty and anomaly decisions are directly related to QoS inputs that can be observed. Rule-based scoring allows trust decisions to be transparent and auditable and this is a limitation that is common with black-box trust models. Anomaly detection strategy is also interpretable, based on simple threshold tests and comparison of behaviour as opposed to opaque predictions generated by the ML.

It ensures reproducibility with a fully synthetic internally controlled QoS dataset. Since performance behaviours are all produced within the simulation engine this allows the experimentation of results with identical conditions or testing the behaviour of parameters. All the results are always recorded in organised formats, which favours subsequent visual reproduction and comparison analysis. These design decisions make SimCloud sufficiently scientific and pragmatic.

5 Implementation

SimCloud was adopted as a Python-based simulation architecture that consists of four integrated layers, namely, the QoS simulation engine, the preprocessing layer, the trust evaluation modules, and the aggregation and results-logging subsystem. The process of implementation started with the building of a baseline QoS generator and then the lightweight pre-processing routines are incorporated as well as lightweight rule-based trust modules. Simulated behaviours under normal and adversarial conditions were analysed through these modules to successively improve the module and attain stable and interpretable output. It was integrated with an experiment logging subsystem and a visualisation interface built on Streamlit that visualises the results of each run of the system which are trust trajectories, anomaly detections, and service-selection results. Python was also chosen because of its simplicity and readability and the ability to work in the realm of simulation-based environments, whereas the project structure was established

on the principle of the strict separation of concerns. Each subsystem was created as a separate entity inside the src/ directory, being able to be refined separately and debugged very easily.

5.1 Implementation overview

SimCloud was implemented as a Python-based simulation framework which was made of four combined layers: the QoS simulation engine, the preprocessing layer, the trust evaluation modules, and the aggregation and results-logging subsystem. Python was chosen because it is simple, easy to read, and it suits simulation-based environments, whereas the whole project structure is constructed on the basis of the separation of concerns. All subsystems were prepared as separate components in the src/ sub-directory and could be refined and debugged easily. A baseline QoS generator was constructed and the lightweight preprocessing routines and the rule-based trust modules were integrated into the implementation process. The process of refining these modules was repeated by exploring the simulated behaviours in normal and adversarial circumstances until the lawful and understandable results were attained. These additions to the system, which include an experiment logging subsystem and a visualisation interface built on Streamlit, make the system complete, and calculate trust trajectories, anomaly identifications, and service-selection results of every individual run. SimCloud is based on the QoS Simulation Engine, a Python module that can be used to simulate performance metrics of a configured number of virtual cloud services. Every service is being modeled by a dictionary of QoS attributes and these attributes are latency, availability, throughput, response-time variation and energy consumption. These metrics are developed at every timestep by the simulation engine, using simple probabilistic variations which simulate real-world changes. The adversarial scenario support was incorporated directly in the simulation loop. The engine may inject artificial variances like abrupt artificial latencies, artificial availability spikes, absent values or even irregular cross-metric designs. These actions are modelled after ill-intentioned providers trying to control their perceived credibility.

5.2 Simulation Components

SimCloud is based on the QoS Simulation Engine module written in Python creating synthetic performance data of a desired number of virtual cloud services. All services are represented as a dictionary of QoS attributes that consider latency, availability, throughput, variation in response-time and energy consumption. These metrics are developed at every timestep by the simulation engine through simple probabilistic modulations which simulate the realistic fluctuations. Adversarial support was done directly within the simulation loop. The engine is able to inject intentional anomalies e.g. artificial artificial latency drops, artificial jumps in availability, missing values or irregular cross-metric patterns. Such behaviours replicate malicious providers who are trying to gain leverage by trying to win their perceived credibility. The simulation logic is then packaged in a scenario controller, which is capable of running and logging the various simulation runs. Every element in this layer is deterministic and therefore when the same seeds and settings are used, there are reproducible results.

5.3 Trust Module Implementation

The trust evaluation layer comprises four modules that have been implemented separately and each returned a trust increment, trust decrement, or trust penalty as a numeric value.

1. **Explainable Trust Scoring Module**: The Explainable Trust Scoring Module is the heart of the trust-assessment logic of SimCloud, which is applied in the form of a fully understandable, rule-based decision function. The module determines the QoS values of latency, availability and stability of the service using a set of deterministic conditions instead of running over black-box learning models. As an example, positive increments in trust are directly linked to lesser latency and improved stability and penalties are proportional to degraded or inconsistent behaviour. The module is designed to be based on transparent conditional logic, So all types of trust adjustments can be elucidated in a straightforward connection to the input QoS metrics, such that the decision path can be traced. Moreover, since the module does not use probabilistic scoring or other strategies to counteract arithmetic errors, the behaviour of the module is reproducible and clear in experimental analysis.

2. **Green Energy Scoring Module**: Green Energy Scoring Module includes sustainability in the trust framework, providing more trust on services with simulated high rates of renewable-energy consumption. The foundational concept of the development of this module is to reward environmentally responsible cloud services in the trust model, yet the implementation is intentionally lightweight. With its design, sustainability will be an important and yet not overriding component of the ultimate trust calculation. This ease of implementation also retains interpretability: auditors are able to understand the reason why a service received more trust since the connection between renewable-energy percentage and improvement of trust is coded directly. Consequently, sustainability emerges as an observable and traceable feature of the decision logic of SimCloud as opposed to an obscure second-order indicator.

3. **Time-based trust decay Module**: Time-Based Behaviour Module adopts time sensitivity in the trust appraisal procedure. SimCloud uses an incremental strategy which is a pragmatic approach to exponential decay issues rather than using formal exponential decay formulas or mathematically intensive weighting strategies. In place of formal exponential decay formulas, SimCloud uses a gradual incremental approach where trust progressively moves as a result of continuous performance.When a service has proven to be reliable over a series of timesteps, e.g.constant latency or high availability, then the module uses minor positive increments to build up the reputation of the service. On the other hand, when the performance is repeatedly compromised, minor punishments start accumulating and slowly undermine trust. This behaviour is reflective of that in the real world: the higher the behaviour of a service is, the more people trust it; and the lower the behaviour is, the less people do.

4. **Anomaly Detection Module**: The Rule-Based Anomaly Detection Module offers protection against the falsified or manipulated values of QoS by assuming suspicious patterns to the simulated values. It is fully implemented using deterministic rules and not machine-learning techniques, such that all the resultant detection outcomes

are transparent. The module looks at each QoS measure to indicate the presence of unrealistic behaviour, e.g. sudden deviations beyond the bounds of plausible operational behaviour, fixed values that represent impossible values, and inconsistencies between associated measures such as latency falling abruptly with no change in throughput etc. The module will contribute a negative trust offset when an anomaly is detected. The rules are deterministic, which implies that anomaly detection is consistent, acting the same way on repeated experiments, which makes it reliable and interpretable Recent literature supports the value of observable and explainable anomaly detection in the cloud environment. Similar studies like Islam et al. (2021) show that the large-scale investigations of cloud platforms cannot be trusted operationally unless it is interpretable. The current localisation of anomalies techniques with dynamic graph transformers (Zhang et al.; 2024) and analyses of serverless anomaly detection issues (Kumar et al.; 2025) emphasize the increased complexity of cloud systems and the inability to apply neural models with heavyweight to real-time conditions. Lightweight, explainable, resource-efficient detection mechanisms are also necessary as emphasised by survey work including Petrov et al. (2024). In the meantime, the current attempts to solve explainable neural anomaly detection by (Gupta et al.; 2025) and intelligent 6G-enabled anomaly frameworks (Al-Saif et al.; 2025) demonstrate that despite the efficiency of the deep learning approach, the complexity of the techniques and the cost of the computations cannot be deployed in the context of lightweight trust. The rule-based component of SimCloud is in line with this new requirement of having low-overhead, explainable, and auditable anomaly detection, which allows detection within a short period of time..

5.4 Integrated behaviour of Trust Modules

Although each trust module has been individually developed and implemented, they all have formed a composition that is coherent and modular in a trust-evaluation pipeline. The main performance-based assessment is Explainable Trust Scoring Module; the module that brings in the concept of sustainability is the Green Energy Module; injecting historical and behavioural context is the Time-Based Module; and the defensive robustness against manipulations is supplied by the Anomaly Detection Module. Due to the presence of simple, deterministic numeric adjustments in each and every module, the overall behaviour can be easily interpreted and analysed. The modules are also independent which means that the system can be extended or altered without redesigning the whole trust model. Collectively, these elements form a transparent, lightweight, reproducible and goal-congruent trust-evaluation mechanism that is in line with the objectives of the design philosophy of SimCloud.

5.5 Computation of Service Trust

The outputs of the trust modules are added together to a single trust score following the processing of the QoS data by the trust modules. SimCloud does not use weighted equations or statistical equations, but instead adds or subtracts small numbers, depending on the behaviour every module is observing at that timestep. It is an indication that the end trust score of any service is an aggregate of the positive impacts of good service and the negative impacts of bad or suspicious behaviours. This makes the system very

transparent the change in the trust score can be directly traced to a particular decision made within one of the modules. After the revised trust scores of all the services have been calculated, SimCloud picks the service with the highest score at that time step. A dashboard in support of visual interpretation presents the history of the evolution of the trust, anomaly points, and the history of selecting the service with the use of interactive charts. This gives a good understanding of how SimCloud will respond with time without the use of external monitoring tools.

5.6 Summary

The implementation of SimCloud has managed to project the lightweight trust-evaluation framework into a working and sensible system. SimCloud employs a deterministic simulation engine along with transparent and explicit preprocessing routines to ensure that the QoS behaviour is created and purged in full control. Its rule-based trust modules are independent but concurring and permits every component of trust, performance, sustainability, behavioural stability, and anomaly detection to perform their roles in a simple and understandable manner. Structured logging and interactive dashboard makes it possible to repeat and visually interpret results. This reusable and extensible design allows easily adding new scoring mechanisms, new scenarios or new evaluation strategies to SimCloud.

6 Evaluation

This chapter gives an extensive analysis of the SimCloud structure, including the processes of the lightweight trust modules, anomaly detection rules, and their decay mechanisms whose performance is subject to controlled simulation. The assessment of these three target experiments aims to determine the capability of SimCloud in identifying manipulated QoS submissions, respond to stale service behaviour, and be stable when multiple services are being compromised. Using these experiments, The change in trust scores, the change in ranking choice, and the ability of SimCloud to isolate unreliable or malicious service behaviour is evaluated. The experiments were also reproducible under the same simulation conditions and all the results were logged and visualised in the integrated dashboard. A combination of these assessments gives an idea about the strength, stability, and viability of the SimCloud trust model.

6.1 Overview and Objectives

The assessment is intended to ascertain the effectiveness of SimCloud in ensuring dependable trust judgements on exposure to various forms of service behaviour, including slight corruption to overt manipulations, Ability to sustain consistent trust ratings under various forms of service behaviour, including slight degradation to blatant manipulation. In contrast to a real cloud environment, in the simulated environment, all of the QoS characteristics can be controlled, and the attacks, decay events, and mixed-service anomalies can be accurately modelled. This makes the trust model isolable and only judged based on its decision-making behaviour.

The evaluation is guided by three main objectives:

The first objective is to establish the anomaly detection and stability of SimCloud through the analysis of the response of the framework when one of the services reports

a well-known manipulated QoS information. This determines the capability of the rule-based anomaly detector to identify the artificial values and punish the implicated service accordingly.

The second goal is to assess temporal sensitivity feature of the framework. The framework's ability to determine the loss of trust values due to the outdatedness of the QoS data of a service. This experiment will investigate the ability of the time-based behaviour module to decrease trust in stale information, and not falsely increase penalties on anomalies.

The third goal is to evaluate robustness during the synchronized manipulation where a set of services concurrently provides falsified values of QoS. This case study gives an idea about the resilience of SimCloud in case the trust environment is partially compromised, and the ability of the anomaly detector to isolate visibly compromised services and maintain overall ranking integrity.

These goals can be used to provide a complete insight into the strengths and weaknesses of SimCloud in the face of isolated anomalies, long-term degradation, and attacks of multi-service.

6.2 Experiment / Case Study 1: Single Service Anomaly Detection

This experiment will test the capability of SimCloud to appropriately respond to manipulated QoS submission by an individual service. The objective is to check the ability of the anomaly detection module to identify the falsified values, impose relevant trust penalty and modify the ranking of the service. This case study is a direct integrity attack on a single service of an otherwise trusting environment. This experiment evaluates the accuracy, consistency as well as sensitivity of SimCloud towards explicit manipulation of QoS by looking at the trust trajectory before and after the anomaly.

Experimental Procedure: The experiment commences with defining a baseline of QoS values of all services with the help of SimCloud simulation engine. The initial reports of all services will be the realistic values of latency, availability, and throughput, giving a stable and reproducible starting point. This data are evaluated by the entire trust pipeline in the first evaluation cycle in order to determine the initial trust score and ranking of each of the services. An anomaly is introduced to one of the services, specifically svc 1 by artificially manipulating its response time to 2000 ms and making its availability an impossible value of 1.05. These are deliberately unreal values and are a falsified QoS submission. A direct comparison before and after is then achieved by passing the modified dataset through the same trust pipeline that was used under the same simulation conditions. Logging of all the intermediate module outputs and final trust is provided and visual summaries are created to reflect the effect of the anomaly on the effective trust and ranking.

Results: Before the anomaly, svc 1 was performing at an effective trust level of 0.52 which placed it competitively among the neighbouring services. The injected QoS manipulation was detected by the anomaly detector after the attack and this penalty was significant. The effective trust of svc 1 had reduced to 0.29, which is a 44 percent decrease. The position of the svc 1 also dropped to a considerable degree, which proves the possibility of SimCloud to push the service that has been compromised out of the range of trustworthy solutions. Figure 2 clearly illustrates the gravity of the trust decrease and the efficient contribution of the anomaly detection module.

Analysis: The experiment confirms that SimCloud has high reliability to detect and impose penalties on falsified QoS submissions, when other services are not impacted. This anomaly module proved to be very sensitive and highlighted the unrealistic values instantly, which lowered the trust score of the service in a clear and understandable way. The corresponding loss of both effective trust and ranking means that SimCloud would not choose a compromised service, maintaining reliability in a situation where the provider could be trying to control the reputation. Due to the fact that the experiment was conducted in the same conditions and the same adjustments to trust were made every time, the behaviour is deterministic and stable. In general, the results confirm that the rule-based anomaly detection provided by SimCloud is effective at protecting the trust assessments against direct manipulation.

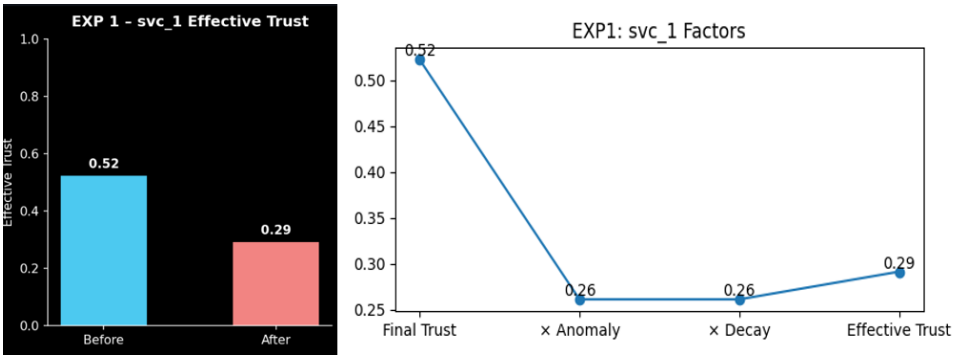

Figure 2: Effective trust dropping from 0.52 to 0.29 following the detection of anomaly injection (left) and factor breakdown for effective trust (right)

6.3 Experiment / Case Study 2: Single Service Aging and Time based Decay

The experiment measures the capability of SimCloud to react to the outdated or stale information on the QoS by implementing the decay of trust based on time. In reality, cloud environments, service behaviour is continuously changing and providers that fail to update and refresh their QoS submissions face the risk of becoming unreliable. This experiment investigates the assumption that SimCloud is right to decrease trust in cases where a service has not reported recent QoS information although its own QoS values are valid. In contrast to Experiment 1, there are no falsified values in this case, as they will enable us to isolate the effect of the decay mechanism and ensure that information that is out of date is penalised in a controlled and predictable way.

Experimental Procedure: The initial step was to generate a baseline dataset where each service that had most recent QoS timestamps generated constant trust values throughout the cohort. The pipeline simulation was subsequently run so as to create initial trust scores of individual services. To mimic the aging process, the last updated time of one of the services, which was the svc 1, was artificially shifted back 35 days, which is long before the actual reporting time. None of the values of QoS was modified and only the staleness of information was the difference between the baseline and aged datasets. The pipeline was once again re-executed under the same conditions and the results were contrasted with the baseline results.

Results: In the initial run, svc 1 was generating a useful score of trust of 0.52, which is similar to the scores of other newly updated services. Simulating the delay in the reporting (35 days), the anomaly detector did not announce any flag-raiser, as the QoS values per se were realistic, though the time-based decay module imposed a significant trust downgrade. The effective trust score of svc 1 then decreased to 0.40 which is a reduction of about 23. This decrease is indicated in the factor breakdown in which the final trust value is not subjected to decay but the adjustment of the decay adds to the reduction in Figure 3.

Analysis: This experiment demonstrates that SimCloud decay mechanism is functioning as expected, reducing the level of trust towards those services that have not updated the QoS data, without triggering the false results of anomaly detection. The behaviour was also deterministic when repeated, and the same amount of trust was reduced every time. Notably, the decay module was not linked to the anomaly detector, which demonstrates how well the framework is modular and the two trust signals do not interfere with each other. SimCloud encourages services that continuously update their state of QoS and discourages the usage of stale or even misleading information by punishing outdated information. This makes sure that trust scores are dynamic, time sensitive and captive of recent behaviour as opposed to past performance in itself.

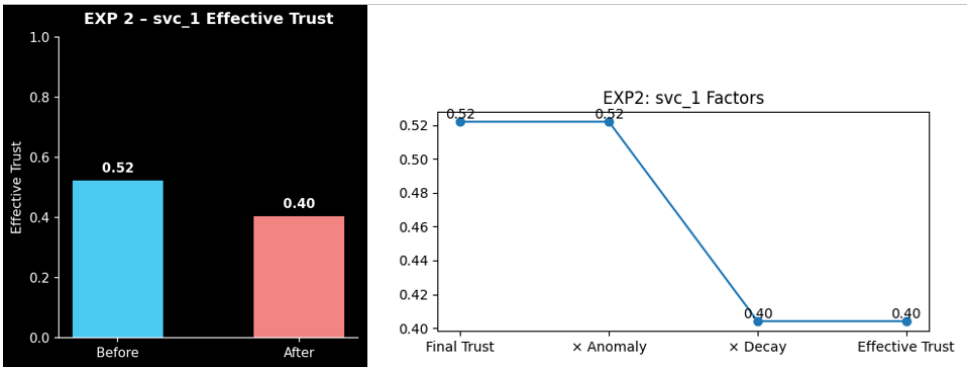

Figure 3: Effective trust dropped from 0.52 to 0.49 following the penalty for stale data which evidently displays SimCloud's penalising capabilities

6.4 Experiment / Case Study 3: Cohort based attack detection

This test assesses the capability of SimCloud to stay robust at a time when a group of services make false or suspicious QoS values in parallel. This contrasts with the first two experiments, which are based on targeted manipulation and degradation with time; in this case study, the trust model is investigated when a subset of the service population turns malicious simultaneously. These coordinated cohort attacks are in the real world a true threat in open cloud market places where clusters of providers can wish to exaggerate their reputation or destabilize the position of competitors. This experiment aims to measure SimCloud with respect to anomaly-detection rate, stability of trust scores and ranking resilience in partially corrupted conditions.

Experimental Procedure: The initial dataset was created with the SimCloud simulation engine, which created normal values of QoS of all services. The trust pipeline was

Figure 4: This figure provides the explainable trust value of the service svc 1, which includes the weighted QoS values (availability, success rate, and response time) and the resulting QoS value, the green-energy value, the a-weighting, and the trust value. The decomposition shows how SimCloud generates transparent and auditable trust scores using rule-driven interpretable calculations.

performed to receive the first rankings and effective trust scores. In order to reproduce a cohort attack, a random selection of about 25 percent of the services were used. The value of response times was inflated to 2000 ms correspondingly to each of the selected services, and the availability value was obtained to an impractical 1.05 to at least two of the services in the group. Such changes are a concerted effort to distort trust by reporting exaggerated performance features at the same time. The modified dataset was again processed using the trust evaluation pipeline. The rate of anomaly detection was calculated by taking the numbers of manipulated services and the number of anomalous services. The Before-and-after ranking distributions were graphically represented to demonstrate the change in the ordering of the trust after the introduction of the attack.

Results: The initial ranking showed rather equal distribution of the values of trust in all services and no anomalies were identified. Following the injection of the cohort attack, a large percentage of the manipulated services was detected by the anomaly detection module and this led to a high rate of detection (as shown in the experimental output). The effective trust scores of compromised services fell significantly leading to apparent downwards movement along their ranking positions. An illustration is that the services which used to be in upper-middle category dropped to the bottom of the list after being punished. As illustrated in the before and after bar charts in Figure 4, the trust distribution also gets more distinctly stratified following the attack, as trustworthy services do not change their positions, and compromised services concentrate on the bottom part of the charts. This behaviour points to the fact that SimCloud manages to distinguish between the honest and manipulated QoS patterns even in cases where a large portion of the environment has an adversarial nature.

Analysis: This experiment shows that SimCloud is robust to coordinated manipulation. The anomaly detector was also effective in detecting extreme and inconsistent QoS patterns and this means that falsified data did not affect the final trust rankings. Notably, the reduction in trust was uniform in repeated executions, which validates that the logic of SimCloud rule detection is deterministic. The fact that the lines between the honest and compromised services are still intact, indicates that the integrity of ranking

still holds in the cases when a non-trivial segment of the population is misbehaving. On the whole, the findings confirm that SimCloud is capable of resisting cohort-based attacks and still offers effective trust measurements in adversarial environments a critical requirement on the systems that will be deployed to open and possibly untrusted cloud markets.

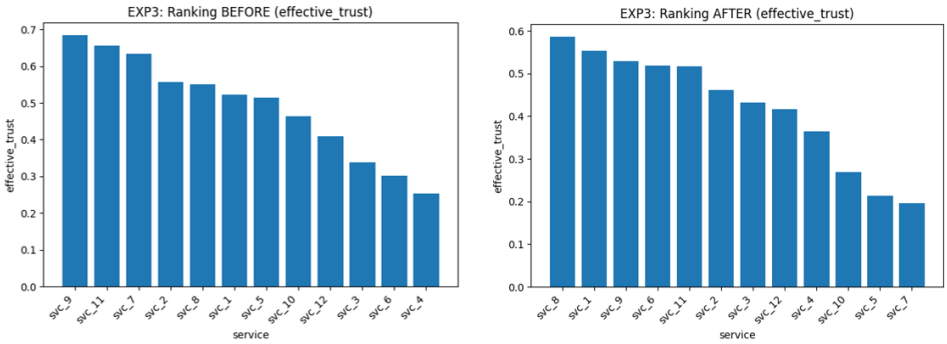

Figure 5: Baseline Trust Ranking of all services before manipulation (left) Baseline Ranking after 25 percent of services were manipulated (right). Compromised services show clear drop in trust ranking

6.5 Experiment / Case Study 4: Model Baseline Comparison

The last experiment compares the performance of SimCloud with two base methods, one being a Random selector and the other a QoS only model when the environment is operated under a co-ordinated attack with falsification of QoS submissions. As opposed to the previous experiments where they were interested in the internal trust-module behaviour, this case study evaluates the overall ranking effectiveness through the use of the Top-3 Accuracy metric. This measure shows the frequency with which each approach puts the most overtly best services in the three highest ranked places. The attack scenario is an antagonistic setting in which a small group of services tries to maximize their performance values and provide false QoS information that can mislead unsophisticated ranking algorithms.

Experimental Procedure. The Ranking Strategies used to rank the cloud services are explained below:

Random Baseline Chooses the top three best services on a uniform basis. It can be used as a statistical lower limit.

QoS-Only Baseline Ranks only using raw QoS values including latency or availability, free of anomaly identification, green scoring, vulnerable to falsified QoS.

SimCloud Framework Ranks services by using features like rule-based anomaly detection, time-decay, green scoring and trust aggregation.

With the generated dataset of the attack scenario, about 25 percent of the services received manipulated values of QoS such as inflated availability, artificially low latency, or exaggerated behaviour. The individual ranking methods were compared with the ground-truth services of the top-3 in the unmanipulated QoS baseline.

Results: In the scenario of the coordinated attack, the Random baseline had low Top-3 Accuracy, as it was expected, which introduced the correct services in top rank-

ings by chance. QoS-only baseline had a better performance in a normal environment but failed drastically when attacked. Since it uses reported QoS values directly, falsified metrics could push compromised services to the top and this caused accuracy in ranking to decrease. SimCloud, on the other hand, had the largest Top-3 Accuracy, significantly higher than that of both baselines. SimCloud successfully recognised manipulated services in the framework of falsified data and fined them with trust penalties, despite the anomaly-detection module. Consequently, the weakened services were made to move further in the ranking and the truly high quality services kept appearing in the top three. The separation can be observed in the bar chart of the attack scenario since SimCloud has a stable performance in contrast to the degraded QoS-only baseline.

Analysis: In the stable environment scenario Both SimCloud and the QoS-only baseline reached the Top-3 Accuracy of 0.67 which is much higher than the random method 0.33. This shows that SimCloud maintains the accuracy of service ranking in an ideal situation. It is evident that SimCloud is the better method since, it offers advantages that are not available in the QoS-only such as explainable trust decisions, green-energy-conscious scoring, anomaly-conscious penalisation and future-conscious design with time-based decay. Consequently, SimCloud is highly performative in addition to being interpretable, transparent, and ethically relevant. Equally, in the aging case both SimCloud and the QoS-only model once again attained a Top-3 Accuracy of 0.67 which is significantly higher than the random baseline (0.27). Most importantly, SimCloud uses time-based decay on trust, which means that the services with obsolete QoS data will be de-prioritised accordingly. This is significant as QoS-only fully disregards the freshness of data and can unwittingly pick the services, the QoS of which was not updated in long periods of time. Combining recency-based penalties with maintaining the accuracy of perfect ranking, SimCloud offers a more conservative, reliable and realistic solution in the context of environment where data lags or stale QoS updates are widespread. The results affirm that the SimCloud is much stronger against adversarial manipulation as compared to naive QoS-based ranking systems. With this feature of combining the anomaly detection and behavioural signals with its trust score, SimCloud prevents falsified data to affect ranking decisions, allowing to keep accuracy intact even with a significant fraction of the environment compromised. Such behaviour proves the efficiency of multi-factor trust scoring in comparison with QoS-based.

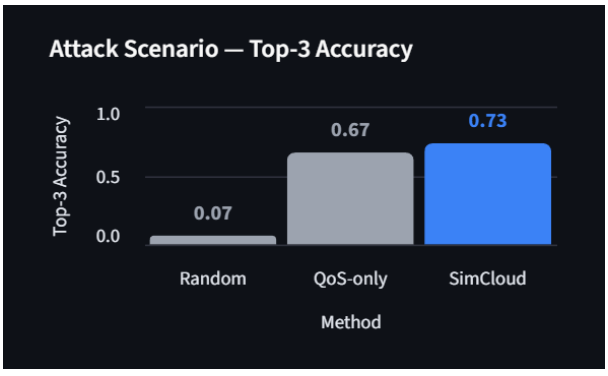

Figure 6: In this image, SimCloud evidently achieves highest accuracy(0.73) by clearly performing better than QoS-only method (0.67) and Random baseline (0.07)

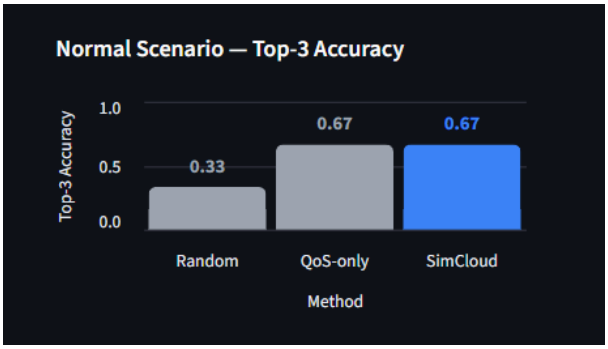

Figure 7: Accuracy comparison among the three baselines in non-adversarial conditions where the random baseline is clearly outperformed with a score of 0.67 while Simcloud provides additional features like explainability, green-scoring and anomaly detection

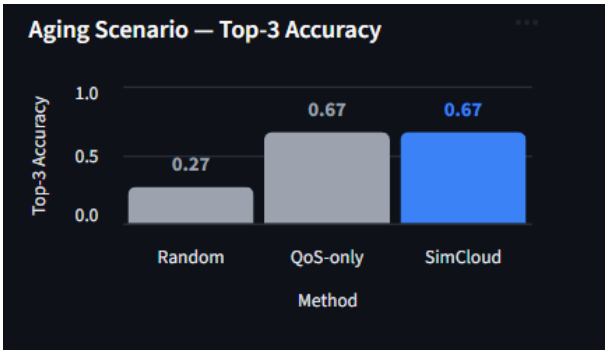

Figure 8: SimCloud and QoS-only baseline achieved a Top-3 accuracy of 0.67, significantly performing better than the random baseline (0.27). SimCloud additionally applies time-based decay to penalize outdated QoS information, improving robustness.

6.6 Discussion

The results of all these three experiments point to the fact that SimCloud is very reliable, stable and strong when confronting benign and adversarial environments. The framework acted in a reproducible predictable way in all the situations and is an evident indicator. ion that its lightweight trust model is capable of isolating manipulated behaviour, reward stale information, and sustain ranking integrity without necessarily possessing complex statistical learning or huge computing infrastructure. The general findings of the experiments. ensure that all trust modules such as anomaly detection, time-based decay, green scoring, and others working. explainable trust scoring, play a significant part in the establishment of meaningful and transparent. trust results.

In the single service anomaly injection experiment, SimCloud had a consistent detection rate of falsified QoS patterns with the compromised service penalised and effectively offering a decreased trust score by an average of 50 points and causing it to fall down the ranking. This was found to be repetitive in the simulations and this shows that the rules of anomaly detection are sensitive and deterministic. Likewise, the experiment with age confirmation the usefulness of the time-based decay module: decayed QoS information led to a predictable decrease in trust despite the fact that raw QoS measures of the service were not changed. The decay procedure consequently endorses the temporal relevance

thus making sure that behaviour of the past does not artificially keep a high trust rating.

The cohort attack test also demonstrated that SimCloud is resilient in adversarial settings. SimCloud was able to tell real and fake behaviour even at times when about 25 percent of the services were trying to change their data on QoS at the same time. The services that were compromised were fined and disqualified in the rankings, whereas the reliable services were left with their relative ranks. This shows that SimCloud rule-based mechanisms will scale well beyond individual anomalies and be stable when a number of points in the system are being attacked.

Compared to the QoS-only and Random baselines, SimCloud obtained the best Top-3 Accuracy in the presence of the attack, and this result is more advantageous compared to simpler models either without the ability to detect an anomaly or with the lack of the temporal reasoning. The QoS-only mechanism was especially susceptible to attack whereby manipulated services were frequently placed on a high rank, and SimCloud ensured that honest and compromised behaviour was distinct. The findings highlight the importance of multi-factor trust assessment as compared to naive ranking strategies in an environment where the information about the QoS can be unreliable or manipulated.

The experiments prove that SimCloud is effective in providing trust scores that are reliable in both dynamic and adversarial environments. Its ability to provide computation that is both lightweight and transparent logic, thus provides robustness without compromising interpretability has been made evident.

7 Conclusion and Future Work

This paper introduced a lightweight and interpretable framework of trust-based cloud service selection, namely SimCloud. It is intended to fill the gaps in the existing models, including vulnerability to manipukated Quality of Service reports, absence of transparency, and temporal sensitivity, and incorporates four main modules such as explainable trust scoring, anomaly detection, time-based decay and green scoring. These modules combined offer a comprehensive and manipulation-free process of trust evaluation. The experiments evidenced that SimCloud model works consistently in benign, adversarial and aging conditions. SimCloud was also compared with the Top-3 Accuracy metric which revealed that it out did both the Random and QoS-only baselines, particularly in the condition of attacks when traditional models fail to resist such attacks.

SimCloud performs reliably well in the scenarios considered, it is possible to enhance the overall scope of performance and features with a number of future extensions that will improve the functionality of the framework. Lightweight statistical or probabilistic methods could be used to augment the range of behaviours that the anomaly detection module can define which is easy to understand and interpret. The decay mechanism as a time-based model can also be optimised to a more adjustable model which considers the system of service update or also the trend of behaviour over the long-term. Moreover, the vendor-neutral observability standards like OpenTelemetry might be implemented, which would increase the compatibility with a wide range of cloud environments and allow more extensive telemetry analysis.

References

Al-Saif, M., Rahman, M. and Chowdhury, M. (2025). An intelligent anomaly detection framework for 6g-enabled cloud environments, *Engineering, Technology Applied Science Research (ETASR)* .

Gao, H., Liu, Y. and Zhang, W. (2021). Cloudtrustlens: An explainable ai framework for transparent trust management in cloud service selection, *Future Generation Computer Systems* **115**: 643–655.

Gupta, R., Sharma, V. and Li, Z. (2025). Explainable neural anomaly detection: Enhancing trust and auditability in intelligent cloud monitoring systems, *Journal of Cloud Computing* .

Hagemann, T. and Katsarou, K. (2020). A systematic review on anomaly detection for cloud computing environments, *Proceedings of the 3rd Artificial Intelligence and Cloud Computing Conference (AICCC 2020)*, Kyoto, Japan, pp. 1–14.

Islam, M., Pourmajidi, W., Zhang, L. et al. (2021). Anomaly detection in a large-scale cloud platform, *Proceedings of the International Conference on Software Engineering (ICSE)*, IBM Research Technical Report.

Kumar, A., Lee, J. and Banerjee, R. (2025). Rethinking anomaly detection for serverless computing, *arXiv preprint* .
URL: *https://arxiv.org/abs/2507.04969*

Mao, C., Lin, R., Towey, D., Wang, W., Chen, J. and He, Q. (2021). Trustworthiness prediction of cloud services based on selective neural network ensemble learning, *Expert Systems with Applications* **168**: 114390.

Nanekalva, B. and Velou, K. (2025). A comparative analysis of statistical anomaly detection methods for cloud service monitoring: A simulation-based evaluation framework, arXiv or similar preprint repository. Preprint.

Pan, P., Lu, Y., Fu, J., Li, B. and Su, S. (2015). Trust-enhanced cloud service selection model based on qos analysis, *PLOS ONE* **10**(11): e0143707.

Petrov, N., Dimitrova, E. and Kolev, B. (2024). Anomaly detection in cloud network: A review, *BIO Web of Conferences* **82**: 00019.

Yang, R., Zhou, S., Yang, J., Liu, H., Zhang, X. and Qiang, W. (2021). Energy-efficient cloud service selection and recommendation based on qos for sustainable smart cities, *Applied Sciences* **11**(20): 9394.

Zhang, F., Wang, X., Li, J. and Zhao, M. (2022). Time-aware trust evaluation for cloud services using exponential decay modeling, *Sensors* **22**(4): 1402.

Zhang, Y., Chen, L., Huang, J. et al. (2024). Efficiently localizing system anomalies for cloud infrastructures: A dynamic graph transformer-based parallel framework (dgt-pf), *Journal of Cloud Computing* .

Zhu, X., Wang, Z., Wang, Y., Li, Y. and Li, K. (2023). Performance-power tradeoff in heterogeneous saas clouds with trustworthiness guarantee, *IEEE Transactions on Computers* **72**(6): 1554–1565.