FrostWatch Whitepaper

jji@frost.watch, cli@frost.watch
frost.watch

Part I. Overview - Finance Model Risk Management: A Necessity For DeFi Protocols

Decentralized Finance (DeFi) has witnessed a massive explosion since the DeFi summer. As the DeFi ecosystem is building a more sophisticated financial system on top of innovative blockchain technology, it must also have effective risk control mechanisms and tools that we have been using on a daily basis in the matured traditional finance world to mitigate its risk exposures. A lot of innovations here - Addresses on blockchain have replaced bank accounts. All the trading functions are fully functional, including Order Book or a more innovative AMM method used by various protocols; lending protocols have been thriving in the turbulent market with an impressive level of automation; derivatives have been growing in size, and users can trade futures easily on blockchain. All the front-end functions that users are accustomed to interacting with on a daily basis have been firmly set up in the blockchain universe.

Most of the DeFi protocols have inherent financial functions built within them. Thus, there is monetary value associated with them, and such value can be huge. Aside from being decentralized or managed by Decentralized Autonomous Organizations (DAO), the protocols serve very much the same function as traditional online financial services.

A traditional financial institution, take a bank for example, typically is split into three groups: the front office, the middle office, and the back office. The front office conducts all client-facing business, including working directly with clients or creating products, research, or analysis for them; the middle office is comprised of all the people in the business divisions that directly support the front office, including the risk department and compliance department, finance, and accounting; the back office works in settlement, trade support, and human resources. They all provide key functions to support healthy business development. Missing either the middle office or back office will bring havoc to the front-office operations.

When we compare the DeFi ecosystem and traditional finance, it's obvious something really meaningful and key is missing - risk management. DeFi is finance, and finance brings risk. Risk can be further categorized into more specific risks: Business Risk, Credit or Default Risk, Country Risk, Exchange Risk, Interest Rate Risk, Liquidity Risk, Finance Model Risk, Code Risk, etc. Those risks bring threat and thus require careful solutions through mitigation, monitoring, and avoidance. A famous example of the impact that Finance Model Risk can bring is the modeling of CDO before the 2008 financial crisis. The CDO market skyrocketed with the invention of a formula called the Gaussian Copula[1], which made it easier to price CDOs quickly. The formula of the CDO, however, turned out to be flawed, as it significantly underestimated the tail risk associated with correlation among defaults. CDO markets finally collapsed when the correlation of all asset classes came to 1, and all those investments, which were supposed to be the safest and rated AAA, became bankrupt. The impact of this event was tragic to the global financial system.

FrostWatch focuses on Finance Model Risk - the risk associated with the design of the financial system in the protocol. The DeFi world has high exposure to Finance Model Risk, and such problems become exacerbated due to the code-driven nature of the onchain world. Among the "13 Biggest DeFi Hacks and Heists" ranked by Decrypt[2], four of them were related to the exploitation of finance models and business logic; that's over 30%. A recent victim was Moola Market. As Rekt noted, "This attack was a simple price manipulation which didn't require any coding." The mechanism of this attack was quite simple in nature - hackers took advantage of the finance model of Moola Market and made millions in profit through pumping one token and borrowing on the other.[3] The issue is quite serious, as such risk might even exist with the most successful DeFi protocols; Avraham Eisenberg noted on Twitter[4] that AAVE may bear a similar type of risk. Such risk cannot be eliminated by simple code review, as it is not an issue with the code but a flaw in system design. Decrypt has an impressive summary: Such events are "a reminder that audits are no guarantee that exploits won't happen."

[1] http://www.columbia.edu/~mh2078/QRM/Copulas.pdf
[2] https://decrypt.co/93874/biggest-defi-hacks-heists
[3] https://rekt.news/moola-markets-rekt/
[4] https://twitter.com/avi_eisen/status/1582763707742183424

We, FrostWatch, understand the business and model logic for DeFi products and we believe in decentralization. We are working hard to helping protocols to better understand their systems and continuously improve their risk profile.

Part II. The FrostWatch Risk Methodology

FrostWatch takes a holistic view of financial risk, and we carry out the risk analytics for protocols using both numerical and simulation methods. Based on the test reports, we carry out our careful review and offer recommendations to optimize the key factors of protocols and improve the vulnerabilities of the current system. We believe such a service is essential to building the protocols' long-term success and ensuring investors' safety.

The FrostWatch Risk Simulation Model

The model carries out tests that include both observation of the onchain behavior of the protocol and our internal standard data sets. The onchain footprint of the protocol provides a tailored vision of the protocol's specific user behaviors. The observed onchain behavior is also used in the calibration of our internal agent (behavior) models and price oracle models. FrostWatch also develops internal data sets that target the long-term equilibrium of DeFi protocols. We combine the data models and behavior models to create different scenarios.

We have chosen the jump-diffusion model[5] as the price oracle for the underlying asset. Such a model is an extension of the Black-Scholes model[6], and it models sudden asset price movements by adding the jump-diffusion parameters. It provides a better depiction of market movement, which can be nonlinear during extremely volatile periods. In other words, it indicates the tail risk is higher than normal distribution or fatter. The fat tails indicate that there is a probability that the market will move in a negative way, far beyond what is expected by normal distribution. Thus, such a model gives more realistic extreme risk projections than standard Brownian motion models.

$$\frac{dS_t}{S_t} = \mu dt + \sigma dZ_t + dP_t$$

$$dP_t : \text{Poisson process}$$

$$dZ_t : \text{Wiener process}$$

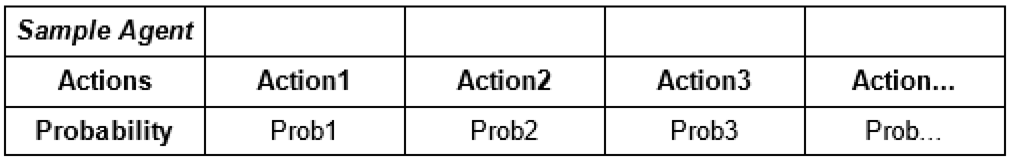
Figure 1.

[5] http://faculty.baruch.cuny.edu/jgatheral/jumpdiffusionmodels.pdf
[6] https://www.sciencedirect.com/topics/economics-econometrics-and-finance/black-scholes-model

We calculate the VaR (Value at Risk) and ES (Expected Shortfall) together to provide a better overall depiction of the risk profile. We witnessed the weakness of the VaR method during the 2008 financial crisis; thus, we propose a more holistic view of risk and incorporate ES metrics in addition to VaR. Analyzing the interaction between models and protocols leads to insights into the optimal balance between safety and capital efficiency.

The FrostWatch Scenario Based Test Suite

We utilize Scenario Based Agent Models to simulate each wallet address as an agent that operates under a matrix of possible actions, with a certain probability distribution calibrated by onchain data. Each agent behaves independently and acts in its best interest. The agents collectively test the stability and reliability of the system. All inputs and assumptions are based on realistic business logic and a market environment, and thus such test results can uncover real issues hidden under the code and the financial models.



Sample Agent				
Actions	**Action1**	**Action2**	**Action3**	**Action...**
Probability	Prob1	Prob2	Prob3	Prob...

Figure 2.

The agents create the input for the test, and the project code on the test chain creates a testing body. The test works to validate the potential outcomes flowing through the project code. The Black Box test suite has sizable action ranges that aim to cover as many potential scenarios as possible. Upon finishing the test, the test suite finds potential flaws in the system that are hidden in the project code. We then generate an overall vulnerability report.

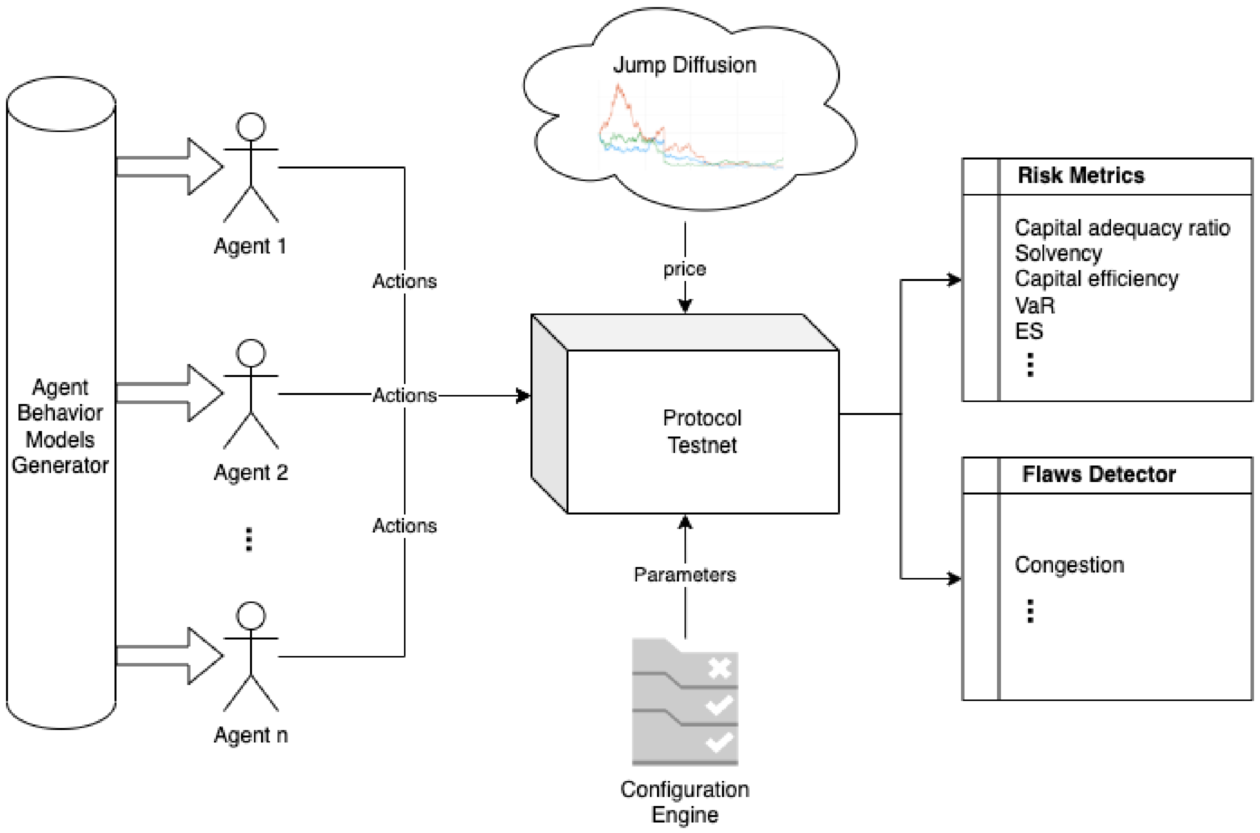


Figure 3.

The FrostWatch Fully Random Test Suite

A more generic and thorough way to test for risk is to utilize minimal information and generate randomized transactions to test the system. This methodology assumes no prior knowledge about the system and thus can cover more cases, both those intended for the product we are trying to test and those unintended for it, creating a more universal testing environment. The benefit of this methodology is that protocols share the minimal amount of information with us and share the highest level of privacy. However, on an individual basis, it's not enough to cover all areas, and thus post-test confidence will be relatively low.

When used together with other testing suites, it helps to reveal problems at the most fundamental level. After the initial work and report, we continuously monitor the risks for our customers and provide recommendations to further improve the risk profile.

Part III. The FrostWatch Tech Platform

FrostWatch's Technology for DeFi risk management is built upon the observation that blockchains are open and append-only databases in which history is immutable and anyone can:

1. fork them to spin up a parallel universe with the full context to date,
2. mock anyone on the mainnet,
3. experiment with all possibilities,
4. know that all learnings will also apply to the actual mainnet.

The challenge is then how to test as many scenarios as possible in a cost-effective way and support as many ecosystems as possible.

Knowing this would be a never-ending endeavor, FrostWatch technology has made several strides towards a comprehensive, cross-chain risk assessment platform that could adapt to various financial product models. The details are explained below.

Our goal is to eventually develop the FrostWatch platform into a system of intelligence that could:

1. automatically verify the protocol under test (target protocol) thoroughly with all the domain knowledge learned along the way and
2. continuously monitor key health metrics to enable fast risk mitigations.

High-Level Architecture: A FrostWatch Parallel Universe

As shown in Figure 4, A FrostWatch Parallel Universe for Risk Assessment consists of three major components:

1. the **Agent Platform**, which can generate actions based on model design with various level of randomness;
2. the **System Under Test (SUT)**, which is a forked blockchain with FrostWatch system instrumentations to allow low-cost simulation runs of system assessment; and
3. the **Event Data Plane**, which includes peripheral components that conduct data collection (Tracer) and data observation (Dashboard/Alert) for:
    a. Runtime insight
    b. Analytics and
    c. Realtime monitoring (which also applies to mainnet products).

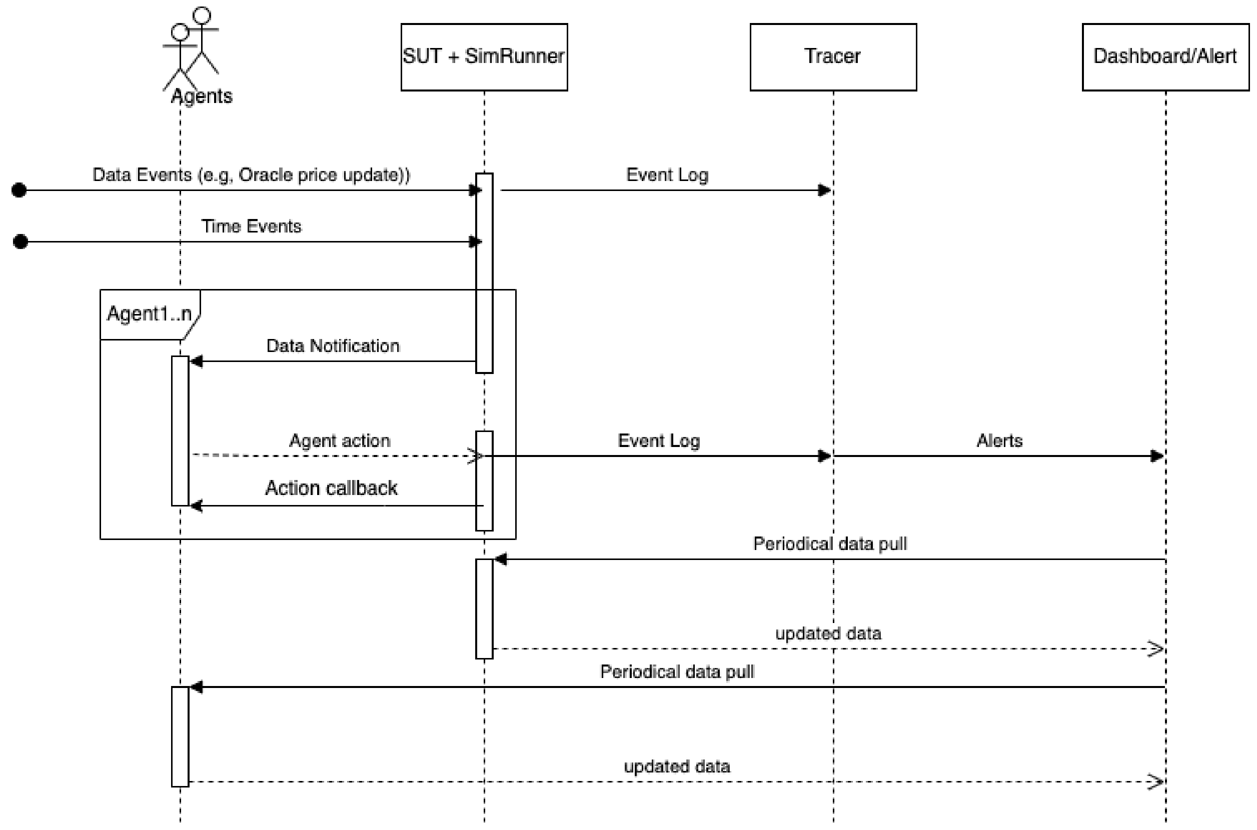


Figure 4. FrostWatch Parallel Universe

In a simulation system like the FrostWatch Parallel Universe, the test bed is formed by a forked chain with the target DeFi protocol with all state to date and all other products and protocols that interact with it on the mainnet, and the simulation is driven by either market data feeds or simply time epochs.

Also, to make sure we capture the events and save them for later analysis with different methods or even replays, the system under test is instrumented with various tracing and logging facilities to log both events from agents and the protocol itself. A sequence of events and their logging is shown in Figure 5

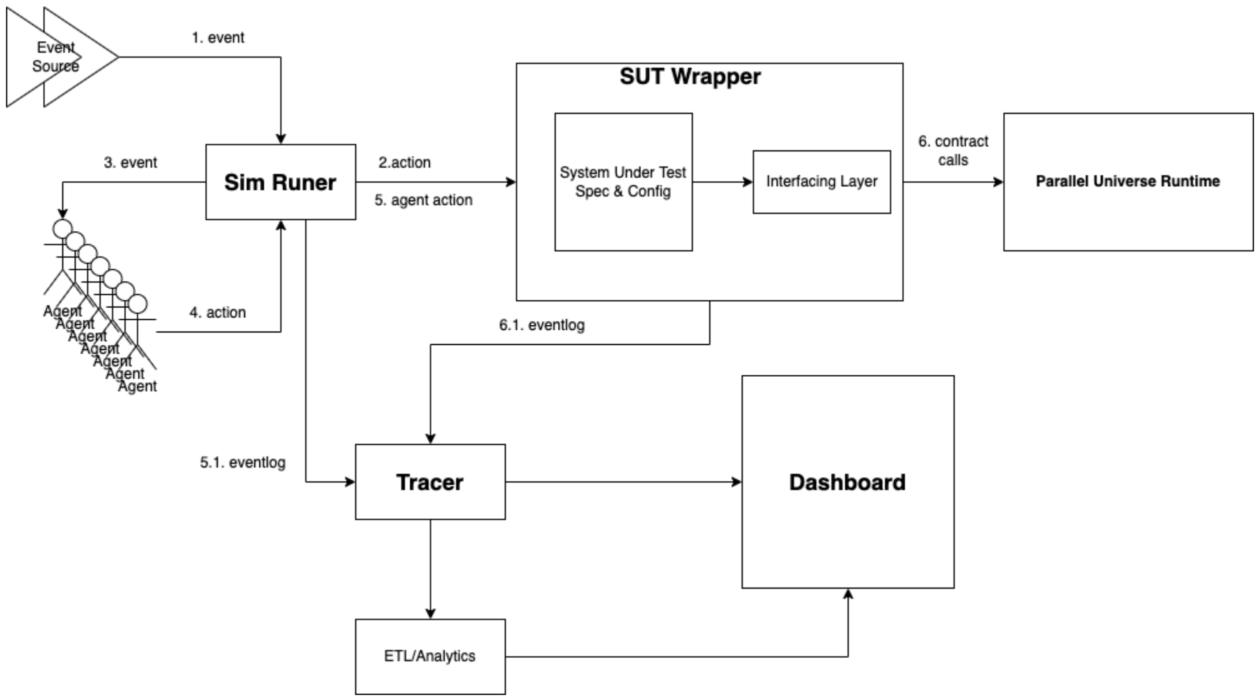


Figure 5. FrostWatch Tech Platform

Balancing Between Coverage And Efficiency

With a system like this, one immediate consideration is what assumptions you want to make and how much fuzziness you want to have.

On one extreme side, one can just randomly generate actions with no limitation or guidance, solely based on the machine-readable interfaces provided by the spec (e.g, an ABI). These are then converted to protocol function calls by the SUT wrapper and every combination of functions and parameters is explored. This will certainly cover the widest variety of cases, both by design and unintended ways of interacting with the target protocol. However, like all other brutal force testing methods or fuzz test methods, this is quite time-consuming and less effective.

On the other side, we can leverage our teams' in-depth understanding of technology and financial basics to curate the test sequence. We can then perform specific tests that can help expose potential issues with the target protocol. This is highly efficient but also very customized. It will not only incur higher human costs but also will require a non-trivial amount of customization for each target protocol, which means it is not a very scalable solution. On the downside, a high level of human intervention might make this method subject to human errors, such as false assumptions and the limitations of individual knowledge.

We are combining both methods in our full analysis of the target protocol so they can complement each other and achieve a high level of efficiency. We are also exploring ways to improve the efficiency of higher precision probing via historical data-backed heuristic and reinforced artificial intelligence.

Data Logging And Analysis

The FrostWatch Parallel Universes are just the logistic part of our platform; the core value of the FrostWatch platform is delivered through data, and data is the centerpiece of our system.

Our data plane handles real-time data collection and visualization. The data collected during simulation runs is later used for processing, event sequence reconstruction, and analytics.

The combination of real-time and batch data processing will allow us to learn more from fewer runs and to make iterative improvements to real-time data monitoring, which will involve monitoring risks for the target protocol on a continuous basis.

A Use Case Demo: Perpetual Futures

To demonstrate how the FrostWatch platform really works, let's look at a demo and conduct some simple experiments.

In this experiment, we are conducting a simulation against a perpetual futures product. Perpetual futures products like Perpetual Protocol (https://perp.com/) are one of the true innovations in the crypto space and a great addition to the DeFi ecosystem, enabling more sophisticated trading strategies. Perpetual Protocol is built on top of Uniswap and Chainlink oracles, which demonstrates the value of DeFi building blocks working in coordination. In the meantime, this actually makes testing/assessing more difficult because they are intertwined systems. Thus, we prefer the method using a forked chain as it provides us the perfect isolated testing environment.

System Under Test: BabyPerp

For the sake of demonstration, we deployed a simple product (let's call it BabyPerp) to be used as our System Under Test. We tweaked it with our own configurations, with notable parameters listed below:

```
// Maximum leverage is 5X the user's collateral.
MAX_LEVERAGE = 5

// When the loss is 1.1x the user's collateral, their positions get liquidated.
LIQUIDATE_THRESHOLD = 1.1

// The fee charged in liquidation actions is 20% of the liquidated value remaining and is
contributed to the product's insurance pool.
LIQUIDATE_FEE_RATIO = 0.2
```

Table 1.

BabyPerp supports only one trading pair, ETH/USDC, and will allow traders to deposit USDC as collateral to start trading.

BabyPerp also maintains an insurance pool of USDC to make sure the product is solvent in the case of mass liquidation, when some of the liquidations would not be able recoup enough value to pay for the loss in a user's position even with all the collateral the user has deposited. Therefore, each solvent liquidation will contribute some of the remaining value (percentage configured by LIQUIDATE_FEE_RATIO) to the insurance pool to make sure it is sustainable.

Similar to the Perpetual Protocol, the group of agents is able to initiate actions like deposit USDC, open long/short positions with leverage, and close their positions, as well as withdraw their funds. To better demonstrate our scenarios, we configured the chances of each action performed by each agent in any triggering epoch in the simulation. We reuse most of the terms and concepts defined by Perpetual Protocol.

To run the simulation, we deployed an automation mechanism similar to Chainlink Automation to feed oracle prices to BabyPerp and, in the meantime, trigger agent actions and clearing house tasks.

In this demo simulation, we wanted to simulate the impact of a sudden price jump, which is commonly seen in a lot of cross-protocol DeFi attacks. In most of these scenarios, the attacker will try to manipulate the oracle price or the computed price of certain trading pairs in certain DeFi markets via methods like flash loans, hoping to exploit potential technical or financial vulnerability for arbitrage opportunities.

Using Real Data To Drive The Simulation

In our demo, we used some historical ETH/USD price data (e.g, https://www.cryptodatadownload.com/cdd/Bitstamp_ETHUSD_1h.csv) to serve as the oracle price event trigger to drive the simulation.

Let’s check the data first. The chart below shows a comparison of the original historical ETH/USD price data and the test data with a price jump of 50% downward added to simulate an oracle attack (e.g, via flash loans).

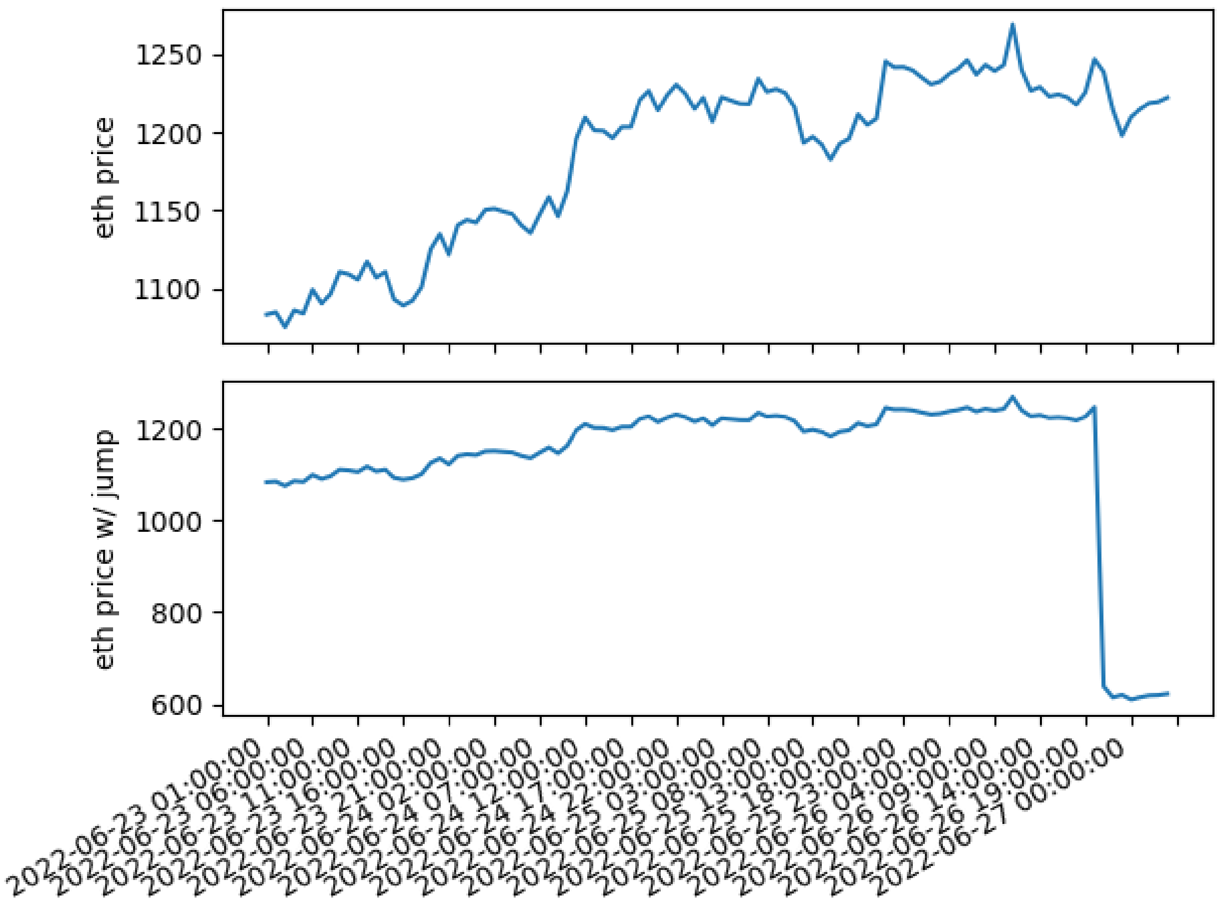


Figure 6.

Simulating A 50% Price Drop

With the FrostWatch platform, we can easily spin up a test environment (a local blockchain forked from its mainnet) and launch a non-trivial number of intelligent agents to interact with the deployed BabyPerp product. This will give us real-time insight and event logs as output.

A sample run result is shown below:

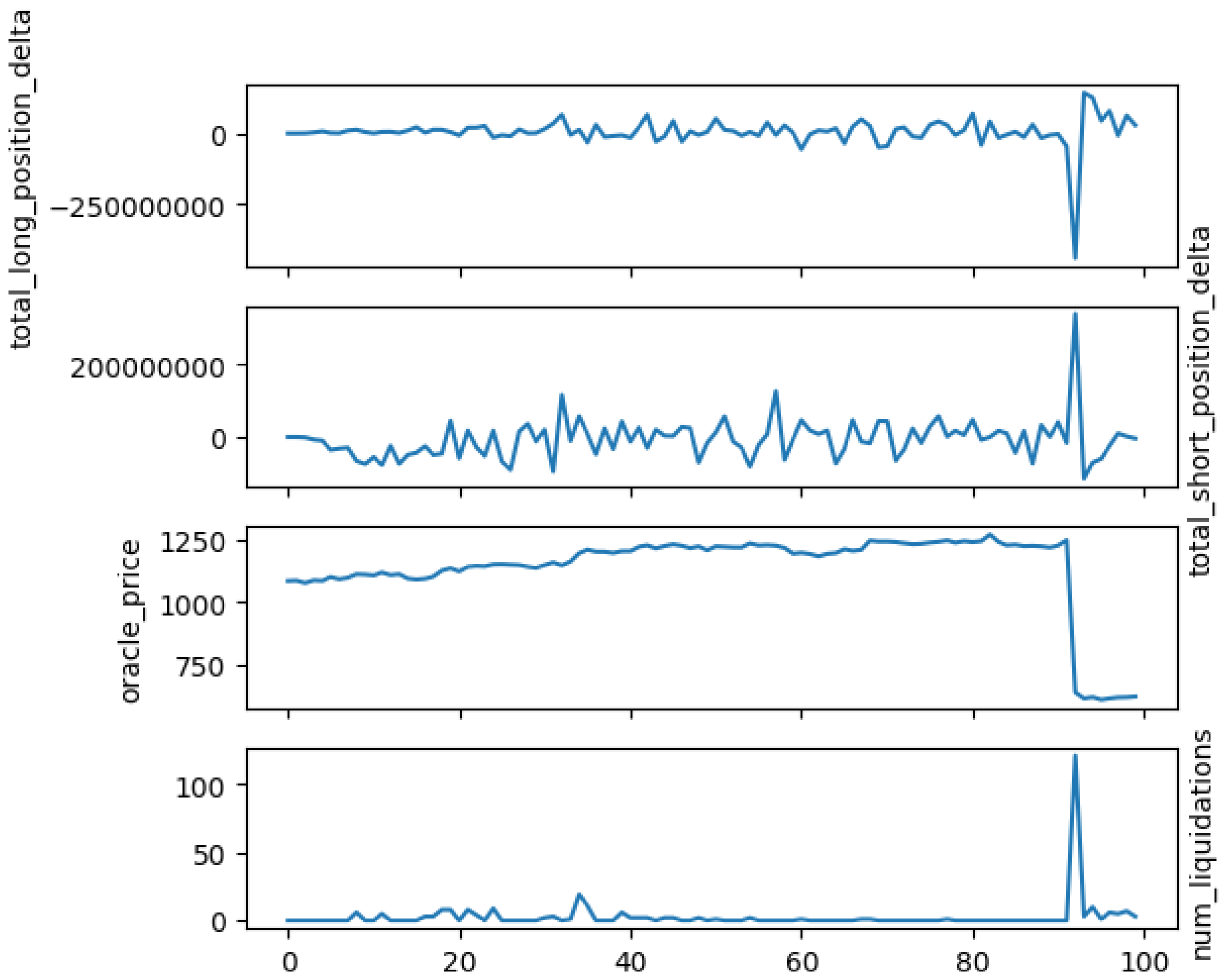


Figure 7.

As we can see from the chart above, There were liquidations from time to time, showing that some of the agents just went rogue and used up all their leverage, and hence could not withstand even a mild price change.

The more dramatic scene came at the end of our simulation, with a ~50% drop of the oracle price: The short-position holders had their position value skyrocket while most of the long-position holders got liquidated. This is shown in the last trend, labeled “num_liquidations.”

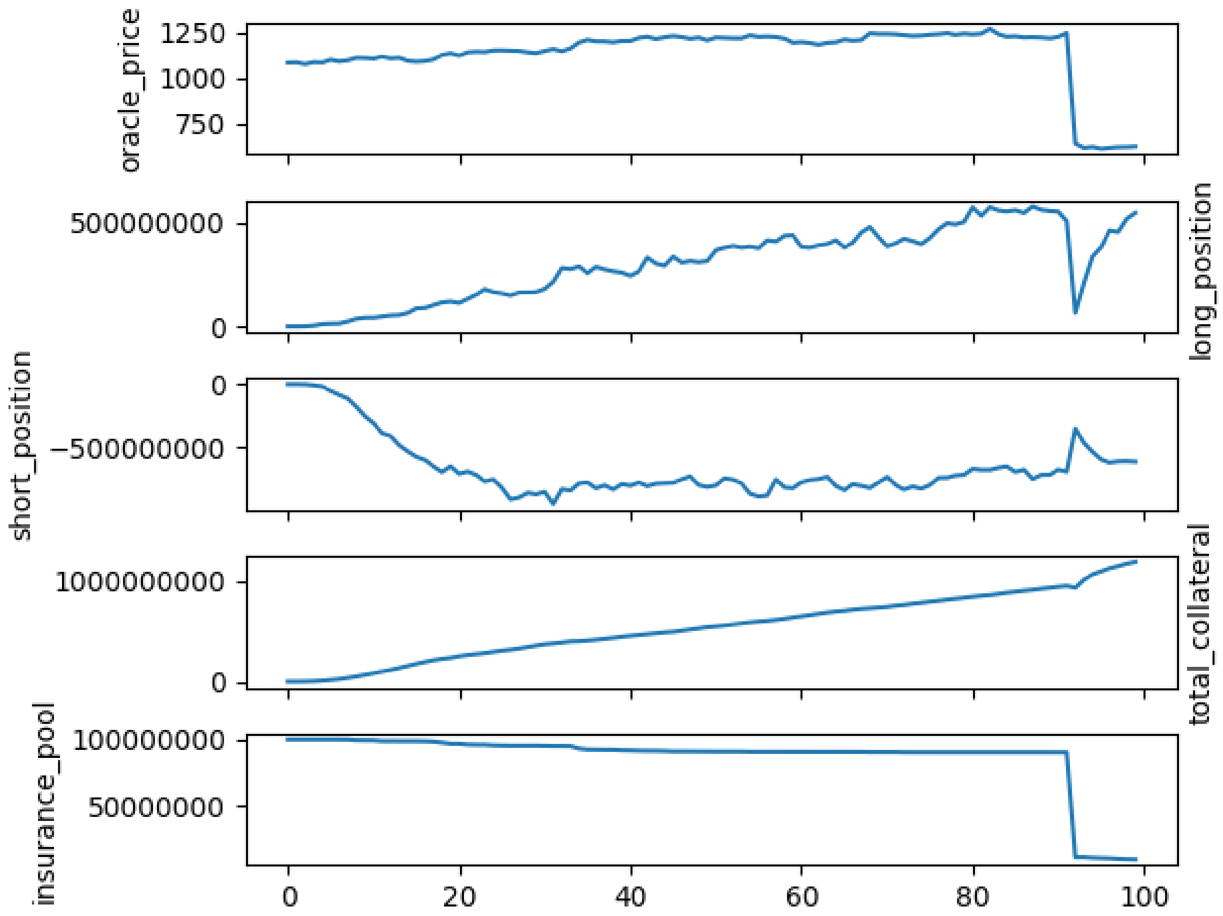
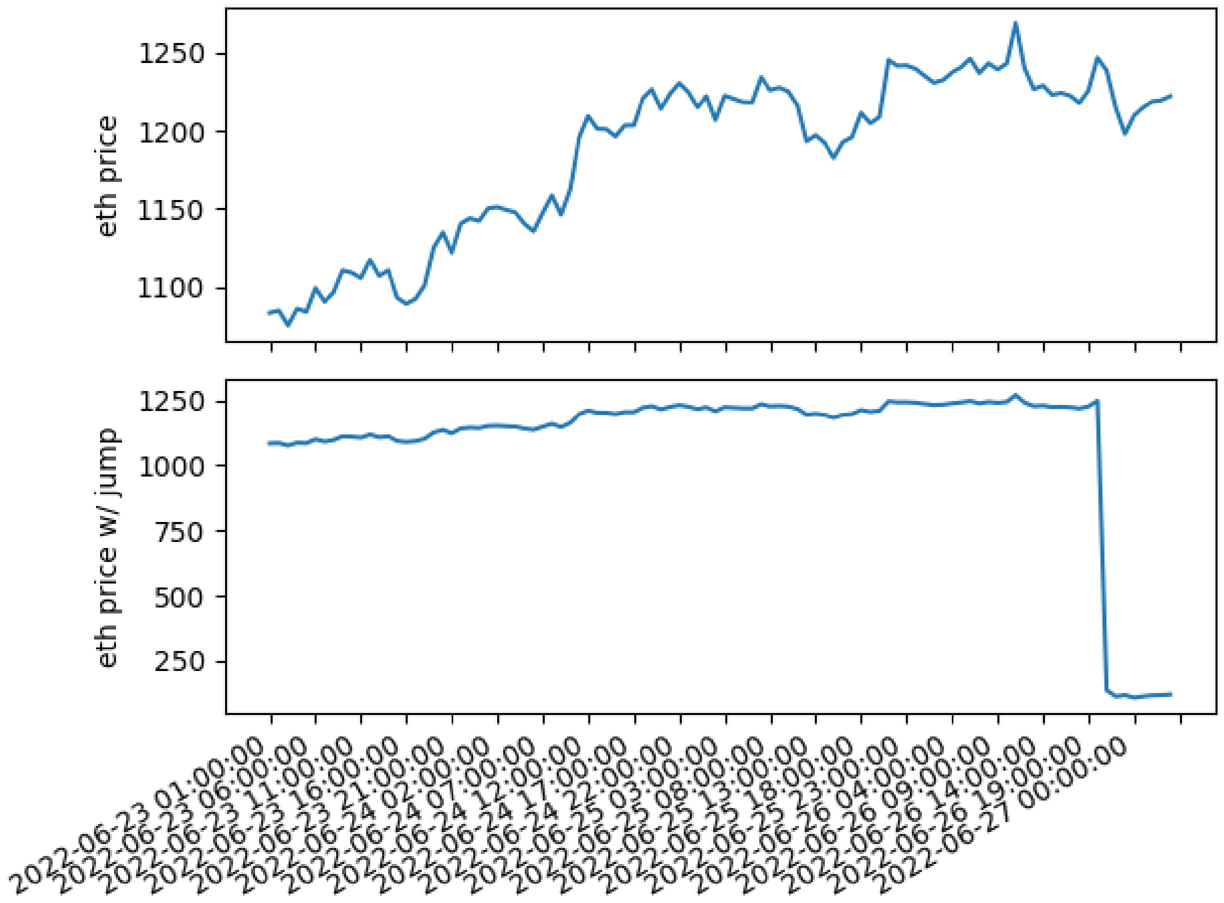


Figure 8.

Correlated with the charts above, we can see that most of the metrics followed a stable trend until the price jump, during which most of the long-position value went down and a lot of long-position holders became insolvent. As a result, we can see a significant drop in the insurance fund value in BabyPerp's insurance pool; fortunately, it didn't go down to zero, so our BebyPerp survived this attack, thanks to its moderate leverage (5X) and deep insurance pool (100M USDC).

Simulating A 90% Price Drop

Now let's move to a more extreme case the price goes down 90% over a short period of time, which could happen in some markets with lower liquidity and/or TVL. In this case, we simulated a scenario in which the price of ETH suddenly drops by as much as 90%, as shown in the chart below:



eth price
1250
1200
1150
1100
eth price w/ jump
1250
1000
750
500
250
2022-06-23 01:00:00
2022-06-23 06:00:00
2022-06-23 11:00:00
2022-06-23 16:00:00
2022-06-23 21:00:00
2022-06-24 02:00:00
2022-06-24 07:00:00
2022-06-24 12:00:00
2022-06-24 17:00:00
2022-06-24 22:00:00
2022-06-25 03:00:00
2022-06-25 08:00:00
2022-06-25 13:00:00
2022-06-25 18:00:00
2022-06-25 23:00:00
2022-06-26 04:00:00
2022-06-26 09:00:00
2022-06-26 14:00:00
2022-06-26 19:00:00
2022-06-27 00:00:00

Figure 9.

With all other parameters staying the same, we see similar situations in the product's behavior.

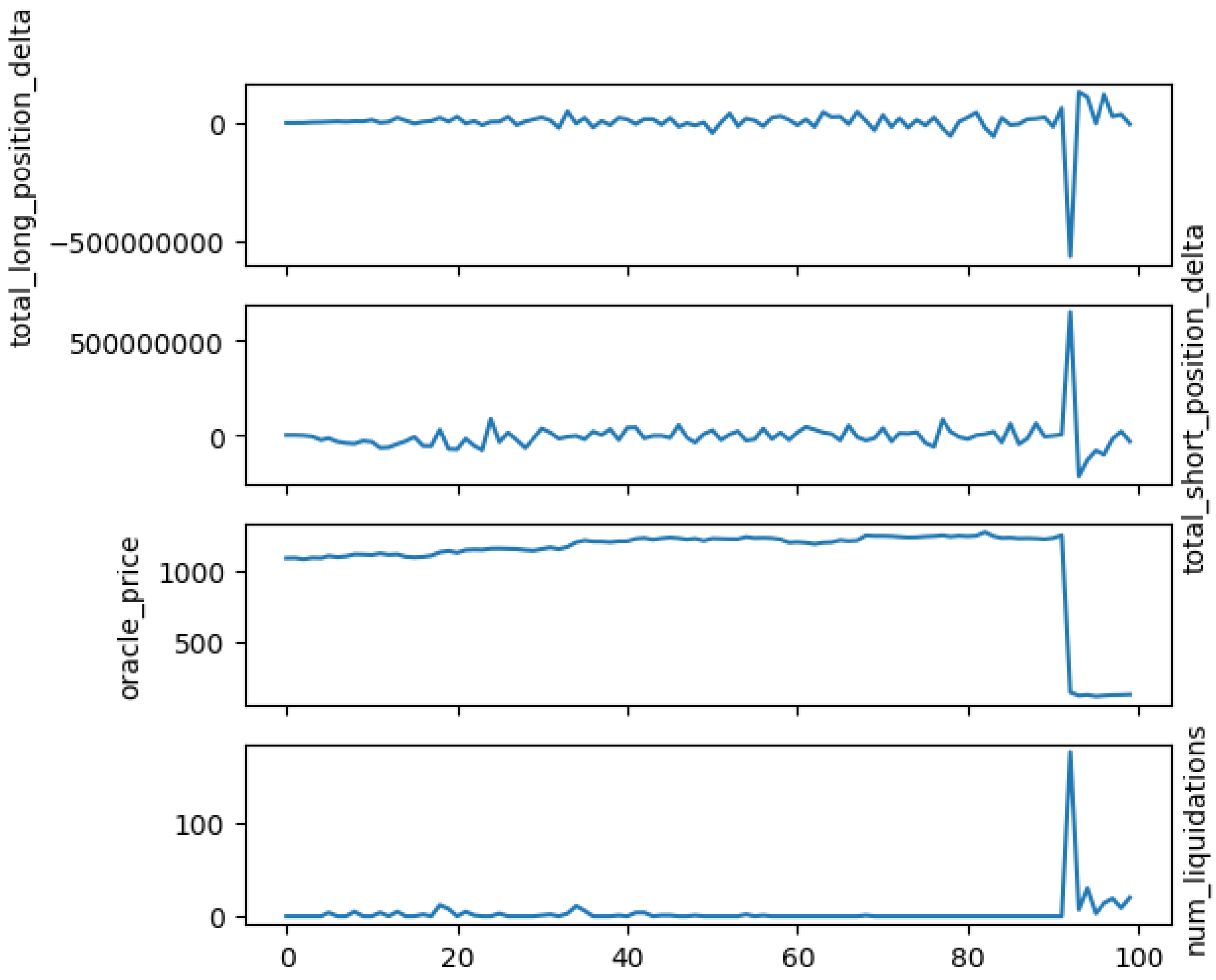


Figure 10.

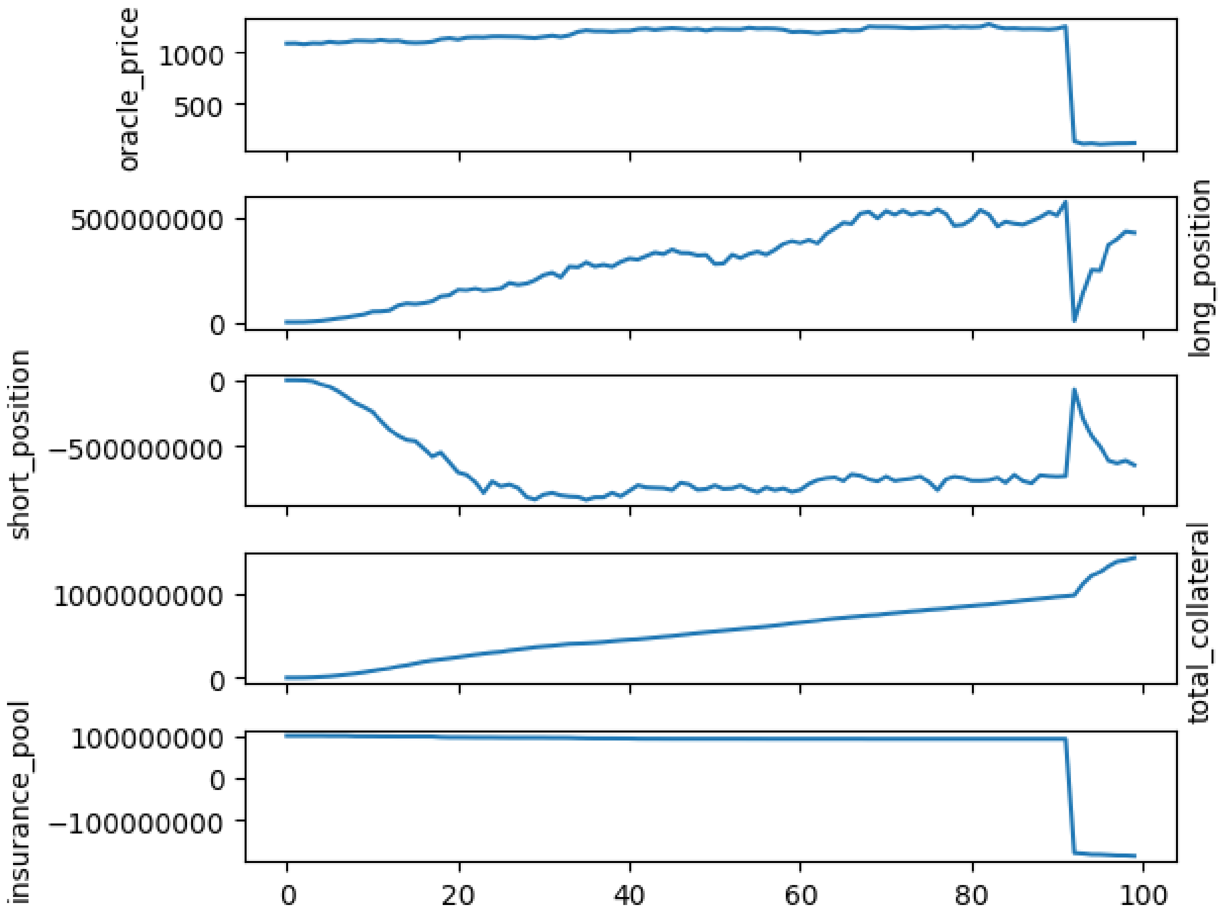


Figure 11.

However, because of the huge gap in value during forced mass liquidation events, the insurance pool value dropped below zero, which means our BabyPerp as a DeFi product became insolvent.

Demo Summary

As you can see, with the power of high-fidelity simulations infused with domain expertise and low lift enabled by technology, it is easy to test and demonstrate different aspects of a DeFi product and verify all its tech and financial assumptions.

With a better understanding of how our product will behave under various circumstances, it would be easy to understand and quantify the risk metrics of the system.

In the DeFi world, risks and vulnerabilities do not only mean code vulnerabilities but also are associated with financial model design flaws or misconfigurations. Some of them could be eliminated with better product design and fine-tuning of the parameters defined in the protocol. The FrostWatch platform could serve as both an identifier and verifier in the process of resolving such issues.

Part IV. List of Images and Tables

List Of Images

List Of Tables

Part V. Revision History and Appendix

Version Date	Release Notes	
Oct 17, 2022	Initial Draft Ready	
Nov 10, 2022	Added Demo case and charts	
Dec 22, 2022	Internal Review Done. Added figure/table captions.	

Table 2.