How Revest Finance Used Tenderly to Implement Improved DevSecOps

Company: Revest Finance
Website: revest.finance
Industry: Decentralized Finance (DeFI)

Key Challenges

- Lack of forensics tools for SC and transaction debugging
- Lengthy development and testing processes
- Too much time spent on implementing governance mechanisms

Key Results

- Reduced debugging time from 3 days to hours
- Streamlined oracle testing led to more onboarded users
- Improved code quality and security

Building a new class of NFTs

Revest Finance operates in the DeFi space as the "world's first platform to offer instant liquidity for locked assets." With almost $2 million in value locked, Revest is a protocol for locking ERC-20 tokens into ERC-1155 NFTs, creating a new asset class within the crypto ecosystem – the Financial Non-Fungible Tokens (FNFT). Shortly after its launch in 2021, the company released Resonate, a flagship system that uses the Revest Protocol to construct fixed interest rates and offer users boosted yields.

Investigating & mitigating a $2M exploit

In early 2022, the Revest Protocol experienced an exploit that resulted in $2 million being stolen from their Ethereum smart contracts. To prevent further damage and secure user funds, Revest's incident response team needed a forensics tool that would allow them to dig through the execution trace line by line to uncover what went wrong. Moreover, the team experienced challenges while testing fixes and simulating outcomes that would confirm that an issue had been resolved successfully.

While developing one of their products - Resonate, Revest encountered challenges related to testing oracles on network forks. Rob Montgomery, CEO of Revest Finance and Resonate, says that testing Time-Weighted Average Price (TWAP) oracles on network forks is impossible, pushing the team to turn to testnets. Testing on testnets also proved to be unreliable and time-consuming.

"It took me three days to deploy the oracle to the testnet, testing if it worked, finding out that it didn't, and redeploying it after making a small change."

Revest implements a complex security-centric governance system comprised of a governance controller contract, other connected contracts, a time-locking mechanism, and other components. In the process of implementing governance proposals, Revest developers wanted to avoid wasting time writing JavaScript to dry-run transactions.

Instead, Revest needed a UI-based solution that would allow them to quickly simulate transactions to obtain the raw calldata in hexadecimal form. Doing this on a testnet would be time-consuming since there is no way to bypass the smart contract time-lock restrictions.

Cutting debugging time from days to hours

The Tenderly development platform equips Revest with vital tools for smart contract and transaction debugging. Using Tenderly's Simulator and Debugger tools, Revest is able to simulate the outcomes of governance proposals with zero risk and without spending real money. These tools also enable Revest to reduce development time and ensure the security and safety of their platform and user funds.

According to Rob Montgomery, CEO of Revest and Resonate, Tenderly Debugger is a standard tool in their debugging and development arsenal. Before a governance proposal is implemented, Debugger is the team's go-to forensics tool for uncovering bugs and detecting vulnerabilities.

> "The safest code we've put out was because of Tenderly. We can build confidence in our code and make it more secure because we know exactly what went wrong."
>
> **Rob Montgomery,**
> CEO of Revest and Resonate

"When we were hit with the exploit, Tenderly was used after the fact to figure out how the exploit happened, to find the origin of the exploit, understand how the hacker had stolen the money, and how we could fix the problem."

Tenderly Debugger gives the Revest team an in-depth view into the execution trace, call and function trace, and state changes. The ability to step through the execution trace function by function makes it easier for Revest to pinpoint problematic lines of code, make changes, and simulate the outcome of those changes before deployment. Debugger also enables Revest to get a deeper understanding of what is causing transactions to revert.

"Every time something goes wrong, we check Tenderly to tell us exactly what happened. Tenderly tells us where something went wrong. Knowing this and having the ability to step through everything saves us hours–dozens of hours."

With Tenderly, Revest is no longer confined to using unreliable testnets. Tenderly Simulator allows Revest to modify smart contract code or transaction parameters directly from the Dashboard before launching a simulation. Before Tenderly, Revest typically spent three days deploying, making changes, and redeploying oracles to testnets. With Tenderly Simulator, Revest engineering team was able to optimize their process by cutting this time from days to hours or less.

"With Tenderly, we didn't have to redeploy the entire beta ten times. We virtually redeploy that one contract via Tenderly and test whether it works. What took us three days to do, with Tenderly, it takes us three hours. Being able to modify smart contract code is one of the most useful tools in our arsenal for debugging. It's incredible!"

In testing, Tenderly Simulator makes it easy for Revest to simulate governance-related functions to obtain raw calldata in hex form without hitting time-lock restrictions imposed by smart contracts. This unique feature facilitates faster development that would be impossible to achieve on a testnet. Once Revest is confident in their code, the team uses the raw calldata generated by Simulator to run it from the controller contract directly.

By speeding up development and debugging time, Revest is able to implement governance proposals faster and onboard more teams to the platform.

Faster implementation of governance proposals

Revest relies on Tenderly throughout the development lifecycle. From development and testing to debugging and deployment, Tenderly helps the Revest team ensure the highest levels of security and integrity of their platform.

For Revest's CEO Rob Montgomery, security and transparency are at the heart of the process of implementing governance proposals. Tenderly Simulator and Debugger provide the Revest team with accurate and reliable data needed to guide their critical decisions when implementing governance proposals. With Tenderly, Revest was able to eliminate manual log checking which resulted in significantly faster governance proposal implementation.

"Simulator is an incredibly powerful tool. It saves us hours every single time we implement a governance action. Half an hour of work for every single governance action is saved because of Tenderly Simulator."