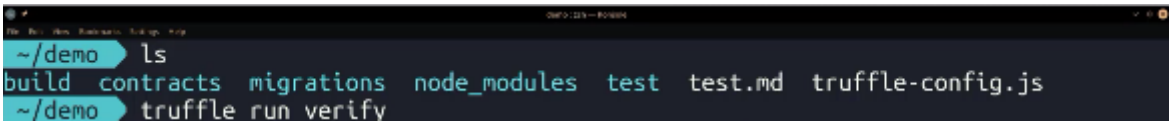
MythX is the premier security analysis service for Ethereum smart contracts. Their mission is to ensure development teams avoid costly errors and make Ethereum a more secure and trustworthy platform. In this article I will talk about how I can use Mythx during the life cycle of development, then how a client can use this plugin to automate smart contract security analysis of the Truffle Framework, after that I am going to talk about the vulnerability meaningful and test our Smart contract with Mythx.

MythX Security Analysis Plugin for Truffle Framework

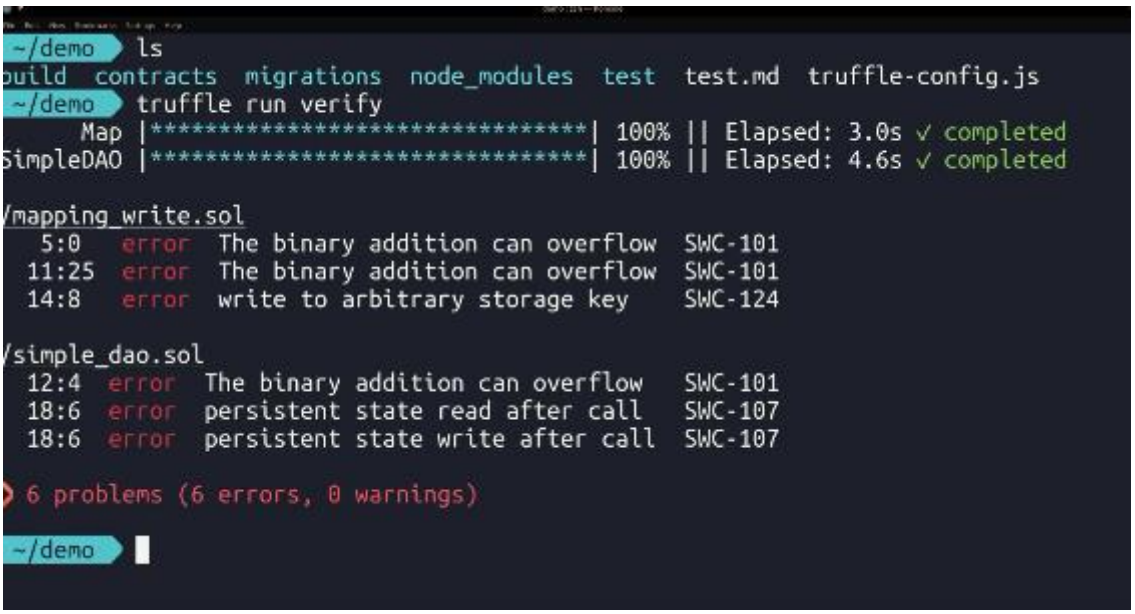
This plugin is compatible with Truffle 5.0 or higher in this section I am going to demonstrate how to use the Mythx Security Analysis plugin for Truffle

to analyze these contracts to see what wrong run the following command truffle run verify inside your truffle project.

In our case we're going to analyze all of our contracts though three is a contract built into all truffle projects called migrations and that one will be skipped.



the results take a few seconds to come in the length of time is determined by a number of factors, including the type of analysis desired the progress bar shows the status of the analysis once completed here we see the issues detected by mythX for each contract we see violation and violation severity as well as the ID of the violation the output also shows the specific line number and location of where the violation occurred in the code the swc Id refers to the smart contract weakness registry a standard list of vulnerabilities.



We originally ran the command from the terminal, but like most truffle commands you can also run the mythx plug-in from within the truffle develop console so let's do that while we're

doing this let’s add a few new options first the style flag, let’s use configure how the output result should be formatted and now we are only running the analysis on the simple smart contract (Many option to add).

There are a number of option for running the mythx plugin we can bring up the full list of options by using the help flag, let’s talk about some of the most commonly used options mode determines how in-depth you want your analysis to be the default is quick full which uses a more in-depth range of analyses can take on the order of minutes to return style as we saw previously allows the user to format the output results timeout will let you decide how long you want the command run on the terminal before giving up and returning control back to you note that processing may still continue on the Mythix server so you’ll get a unique identifier that you can later use to retrieve results the debug flag provides additional debugging output there is also debug too for even more verbose output there is lots more to talk about with mythX and its integration with truffle but we suggest you giving it a try yourself mythX for Truffle is available as standard NPM module and you can use the mythx service on a trial basis for free right now if you’d like to learn more about the mythX platform go to MythX .io