Vision Statement

Team Name: FortaKnight

Team Lead: Nicholas Brown (nicholasbrown@ucsb.edu)
Members: Khalid Mihlar (mkmihlar@ucsb.edu)
Alejandro Rojas Rodriguez (arojasrodriguez@ucsb.edu)
Andy Wu (andywu@ucsb.edu)
John Lin (yongchen@ucsb.edu)

Project Description:

Web 3.0 presents itself as a new, cutting-edge blockchain technology for designing and structuring the internet. However, hackers are currently attacking and exploiting Web 3.0, allowing hackers to siphon money from the blockchains. When attacks occur, hackers must deploy attacker smart contracts several blocks prior to the actual exploitation. We can take advantage of these smart contracts to detect oncoming attack blocks before any damage is done. To aid in the security of Web 3.0, this project aims to develop a bot that can detect threatening smart contracts through both dynamic and static analysis.

This problem is important because hackers are stealing a large amount of money from Web 3.0, leading to significant financial losses. This issue undermines the development of internet technology while allowing hackers to gain more funds needed to continue attacking the blockchain. Therefore, developing a detection bot would hinder hacking attempts, enhancing the security and safety of Web 3.0 as this technology develops. Addressing this issue can protect the financial stability of users, giving them a safe environment to develop and use blockchain technology.

Currently, companies are searching for methods to deal with this type of hacking. One method incorporates smart contract audit checks to identify potentially hacked smart contracts that can attack a blockchain system. However, this approach is ineffective because the engineers have to manually run the audit checks, allowing the exploitations to occur before the algorithm can detect an attacker smart contract, leading to financial loss within Web 3.0. This solution lacks sustainability because audit checks are more of a system that aids in dealing with the symptoms of an attack, but does not proactively solve the problem before the blockchain is exploited. Therefore, our proposed detection bot can lead to an increase in identification of attacker smart contracts before financial damage occurs within Web 3.0 blockchain.

Outcome of Project:

The goal of this project is to develop a detection bot that can detect attacker smart contracts several blocks before the exploitation and signal relevant hosts. The bot should be able to both dynamically and statically decompile attacker smart contracts. Ultimately, we hope to develop an effective and easily deployable detection bot to protect Web 3.0 blockchains.

Initial Project Milestones:

Our first goal would be to document on paper an understanding of how our bot would function given different scenarios that Forta has had to experience in the past with attacker smart contracts. We will look to identify patterns to understand how to build an initial prototype for our bot. Then we will start to create a bot focused on identifying specific attributes of smart contracts to see if it can identify attacker smart contracts given training data. If we are able to successfully create a bot to notice certain aspects, we can organize our code to focus on scalability, increasing the ways the bot can identify aspects of attacker smart contracts using Ganache for dynamic and decompiling for static. We would finally create an autonomous design for the bot to work seamlessly for both static and dynamic analysis, making sure that it is able to identify the most amount of attacker smart contracts and improving it as time moves forward.

Methods & Design:

Collectively, we have a handful of tools that we have significant experience in and could use for the project. However, we will also be willing to learn any new languages and tools needed for this project as we converse with the mentors at Forta. For starters, we will be using GitHub to manage push/pull requests along with a place to store our code. We prefer to use Python, but we are willing to change and learn new languages based on what the company wants. If we need to host a website, we can use Heroku, and if we need authentication services we will use Google OAuth as we all have experience with it. If there is a need for backend storage services, we will use RDS, but based on the project description they most likely have backend services for us to use for this project. We are open to learning new softwares based on what Forta already uses and are adaptable to new services that the company already uses. The exact specification as to what will be used will be updated after we touch base with Forta about what the project entails and how we will go about accomplishing our goals.