Project Update 1: Progress in Prioritizing Patch Management Using Quantum-Inspired Vertex Cover Algorithm

Alli Ajagbe, Ayush Sheth

Project Methodology

This project leverages the Minimum Vertex Cover problem and quantum computing frameworks like Classiq to analyze vulnerability data represented as a bipartite graph. A dual graph capturing connections between vulnerabilities is constructed. By solving the Minimum Vertex Cover problem on the dual graph using the QAOA algorithm, we aim to identify a minimal set of vulnerabilities to patch, effectively disrupting potential kill chains.

Updates till Now:

Literature Reviews

This project is a relatively challenging one for both of us as we are new to cybersecurity. We have spent a huge chunk of our time pre-mid-sem reviewing literature published on the domain to reinforce our understanding of the concepts required for the project. To date, we have meticulously reviewed over 20 papers and websites. Our approach and analysis are cataloged in this Excel sheet, comprising two sections. The first sheet contains the methodologies extracted from each literature source, accompanied by three proposed strategies for their integration into our project framework. The second sheet contains a component-wise review of the literature, required for quick revisits and perusal.

Algorithm Development

Prior to the integration of quantum computing/QAOA algorithms, we wanted to make sure we could fully implement the classical algorithms of interest in order to properly appreciate why quantum computing works well in our project domain. We have implemented a vertex cover algorithm and bipartite graph representation. These implementations are coupled with tests to validate functionality and assess performance both now and at later stages of our project. For the bipartite graph, we have also integrated NetworkX for the respective visualization.

Github:

Our GitHub repository contains a folder - ‘LearningMaterials’ with a ‘description.md’ file to explain in paragraphs some of the terms and concepts we are using in our project - they are quite a lot. Additionally, the Python files with the implementation of the algorithms mentioned above can be found here. We have also included our Excel file from our literature review and a ‘progress.txt’ file to help us stay on track.

Plans for the Coming Month(s):

1. We will construct a "connectivity dual graph," where connections between vulnerabilities themselves will be represented as edges, allowing us to analyze how patching specific vulnerabilities disrupt potential attack chains.
2. To find the most optimal set of vulnerabilities to patch, we will incorporate the Pyomo library, which specializes in solving optimization problems, into our analysis of the connectivity dual graph.
3. We will integrate the QAOA (Quantum Approximate Optimization Algorithm) with the Classiq framework, essentially translating our problem of finding the optimal patch strategy into a format suitable for execution on quantum-inspired computing platforms.
4. Once we have solutions from the QAOA integration, we will refine them by potentially considering additional factors or constraints not directly captured in the initial model.
5. Throughout this process, we will meticulously document our findings, including the construction of the connectivity dual graph, the optimization process using Pyomo and QAOA, and the final refined solutions we obtain.

We would also like to point out that the GitHub repo will be updated regularly, so it would be great to keep up with it if possible.
Moreover, we are still learning whatever we will have to implement for the project, so kindly point out if there are any discrepancies, and we will take due note to correct them.

Do reach out to us in case of any questions!