What is the project

The project focuses on the creation of a crypto mining website using javascript/webasm that can evade detection by recent crypto mining defenses as well as analysis on the limitation and robustness of these detection mechanisms. Current detection mechanism we found includes fingerprinting crypto mining APIs[1], fingerprinting hashing algorithms that are used in mining, monitoring CPU cache events[2], identifying repeated patterns in executable stack[3]. Other static/dynamic analysis methods and machine learning techniques we come across will also be part of this project as we will create evasion methods for them.

Why is tackling this project important

This is important because in-browser mining techniques that are undetectable can be used for attackers for cryptojacking. Investigating methods of evasion will allow future developers to implement a defense over it.

How do you plan on tackling the project

April 5

- Identify and implement crypto jacking detection programs.
    - MineSweeper and/or CMTracker, which are open-source and documented in their paper.
- Setup crypto wallet for Monero
- Setup html/javascript that utilizes third party mining API for mining
    - Current choices are CryptoLoot, CryptoImp, Webminepool
    - Third party APIs are fingerprinted by detection programs. They also utilized websockets to communicate with mining pools that theoretically are being monitored by detection programs.

April 12

- Test our mining websites against detection programs to see if these mining javascript can be detected, begin modifying miners to evade detection
    - Identify the area of the miner that was flagged by detection
    - Identify thresholds of detection through modification of our miners
- Present Mid-epoch, discuss current issues with development, demonstrate how crypto jacking detection works, demonstrate our current mining javascript

April 19

- Continue to modify our mining script to evade detection

April 26

- Prepare presentation
- Demo will be our crypto mining website that successfully evade detections

List of 3 sets of deliverables (I'll take these under advisement when grading, but I don't promise to strictly abide by them):

Set of deliverables that will yield a passing grade

- Evasion on detection mechanisms that are focus on fingerprinting hashed algorithms and APIs and CPU monitoring (MineSweeper and CMTracker)

---

[1] https://www.researchgate.net/publication/323654794_A_First_Look_at_Browser-Based_Cryptojacking
[2] https://dl.acm.org/doi/pdf/10.1145/3243734.3243858
[3] http://www.cs.ucr.edu/~zhiyunq/pub/ccs18_cryptojacking.pdf

- Analysis and robustness evaluation of fingerprinting based detection mechanisms

Set of deliverables that will yield an A grade

- Analysis, categorization, robustness evaluation of existing detection mechanisms
- Evasion of several categories of detection mechanisms, and cost analysis of evasion strategies

Set of deliverables that shows work clearly beyond an A

- Find cryptojackers in the wild that are currently employing some or all of the evasion mechanisms (too hard?)

Link to a git repository where you'll keep all the code, documentation, and development through the project.

- https://github.com/apollolo/CryptoJacking