Working prototype:

- Hold off on virustotal (its only like two lines of code)
- Use powershell to analyze DLL calls (not the best way)
- Investigate ProcMon logs for DLL analysis
    - Do testing with ProcMon
- Look into using Windows API Override functions

Do for next week:

- Identify if procmon or process explorer can identify DLL invocation.
- Write down list of imported and already known DLLs (to get a baseline) to get a list of the trusted ones
    - If anything else is being invoked, its probably not good
- Get a list of known malicious DLLs and how they can be detected
- Are there hashes of DLLs, names/location in directory, how to identify them
- DLL injection, can we log if one of the trusted programs is loading a DLL that is not from our trusted list of DLLs
- Write powershell script that parses through the logs (using process IDs)
    - Use ProcMon or Process Explorer
- Static and **dynamic** DLLs, see which one is more important to go for first
- Use sandboxes or any.run so you never have to download malware lol
- Use powershell script that can run for DLLs
    - Whole program can be calling a DLL
    - Make sure its logged
- Find and parse and generate logs. Do some testing.
    - Make sure to find known list of good and bad DLLs
    - Find out what a bad guy will try to do
    - See if defender or anything picks up on it

- Do research on DLL detection tools and use their techniques so we can try to emulate it