Shellcode & Process Injection

8-12 FEB 21

Tasking

Getting familiar with native encryption and in-memory execution techniques.

Detourer

Detourer is finished, though future work could be done towards optimizations and extending via new modules. This week's changes:

1. Completed dynamic module inclusion
    - Enabled single-file payload modules
    - A minimal, but complete module needs only be ~5 lines
2. Fixed bugs
    - The sample VirtualAlloc module no longer crashes the target
        1. Apparently not everyone likes free RWX
    - Modules are much more interoperable, less hooking each other
3. QoL & Honorable mentions
    - Combined the remaining core files (16 KB of code across 10 files)
    - IO no longer stalls the target (much)
    - Optimized code size, PE size, PE speed, and build time significantly
        1. x64 Release DLL (bundling MS Detours + 3 complete payload modules):
            - 56KB (76KB previously)
            - Rebuilds in 3s (7s previously)

Future Plans

1. Perhaps inject into MSBuild to pack arbitrary malcode into legitimate binaries
2. Remaining objectives for Detourer (to be archived)
    - Explore potential for global hooks
    - Complete and further modularize payloads
        1. Add more generic payload wrappers to exfil to file/IPC, modify args, etc.
    - Slim down project and PE even more
        1. Improve backwards compatibility
            - Ideally, pure C eventually
            - Conform to standard, don't rely on MS stuff
            - Minimize needed Windows SDK version and inclusion/usage
        2. Compare runtimes and avoid building from multiple stacks
        3. Remove unnecessary items from MS Detours, e.g. sample code, other archs