* **Anti-debugging:**

*Common anti-debugging techniques:*

1. **Straight checks for breakpoints:**

**software breakpoints** - replacing original instruction with 0xCC and raising interrupt routine for debugger to handle it

Detection of 0xCC bytes. Checks may include comparison to xor’ed value too, e.g. to 0x99 (0xCC ^ 0x55).

Detection of guard pages is **somewhat rare and based on imitation of debugger behavior** - i.e. creation of PAGE\_GUARD memory page and accessing it, previously put return address onto the stack. If **STATUS\_GUARD\_PAGE\_VIOLATION** occurs, it’s **assumed no debugging is in place**.

1. **API calls:**

There are many API calls can be used to detect the debuggers such as

IsDebuggerPresent, CheckRemoteDebuggerPresent, FindWindow, NtQueryObject, NtQuerySystemInformation, NtSetInformationThread, NtContinue, CloseHandle and NtClose, GenerateConsoleCtrlEvent - event-based detection and OutputDebugString with a valid ASCII strings, … etc.

1. **Timing:**

There are many Typical timing functions which are used to measure time needed to execute some function / instruction set. If difference is more than fixed threshold, the process exits such as etTickCount, GetLocalTime, GetSystemTime, timeGetTime and NtQueryPerformanceCounter .

1. **Checksums:**

This method is based on **calculation of CRC32** for certain blocks or whole binary and comparing to hardcoded value. If values differ, it indicates dynamic code changes were made (breakpoints/patches), and the process usually exits.

**The polynomial for CRC32** is:

0x04C11DB7

x26 + x23 + x22 + x16 + x12 + x11 + x10 + x8 + x7 + x5 + x4 + x2 + x + 1

1. **Self-debug:**

There are different approaches for this, probably the most recongnized one is to create a new process and call DebugActiveProcess (pid) on the parent process. If the process is already being debugged, associated sys call ZwDebugActiveProcess() will fail, making it clear something is wrong .

1. **Bonus:**

The best protection against debugging so far seems to be own virtual machine. Effectively, part of object code is converted to self-bytecode format, which is run on a self-written VM. The only way to properly debug such code will be emulator/disassembler for custom VM instruction format.

**References:**

[1] <http://antukh.com/blog/2015/01/19/malware-techniques-cheat-sheet/>

[2] <https://stackoverflow.com/questions/2587766/how-is-a-crc32-checksum-calculated>