VMProtect发展史以及特点

**虚拟化**

可执行代码被转化为由虚拟机执行的代码(虚拟代码)。

该编译类型应该应用到所有执行率非常**重要**的代码的**关键部分**以防止破解与解析（对破解与解析的保护程度中等，代码的执行率中等）。

**mutation+虚拟化**

指定是增强虚拟机的混淆程度，改变虚拟机执行流程，这就是所说的变异+虚拟机的模式，其实虚拟化之后本身自带变异

**X86代码动态执行**

因为vmp模拟指令的个数是有一定的局限性，VM[解释程序](https://baike.baidu.com/item/%E8%A7%A3%E9%87%8A%E7%A8%8B%E5%BA%8F/861575)不能执行所有的Intel 8086命令，所以可能有部分远x86指令可能会提前经过保护，当退出虚拟机之前，会动态的创建原x86指令，虚拟机退出之后再执行

**X86虚拟机代码漂移**

这属于区段区域中的混淆手段，增强动态定位难度

**具有较强的数据校验**

当执行程序的时候，VM[解释程序](https://baike.baidu.com/item/%E8%A7%A3%E9%87%8A%E7%A8%8B%E5%BA%8F/861575)会自动读取当执行命令时被运用的程序、opcode以及水印中任意部分的检查结果。VM对象的完整性检查可保护解释程序、opcode以及水印免遭修改。

**逻辑运算指令**

Vmp中的逻辑运算只有一条指令:nor。这个指令在电路门中叫NOR门，它由三条指令组成，即not not and，与NAND门一样，用它可以模拟not and xor or这四条逻辑运算指令。