『功守道』软件供应链安全大赛·PE二进制赛季启示录：上篇

chybeta / 2018-08-31 11:38:26 / 浏览数 2510 技术文章 技术文章 顶(0) 踩(0)

Author: 弗为@ASRC

综述：办公软件系统供应链安全

软件供应链安全态势，以及『功守道』比赛整体的目标和愿景，请参考系列前篇。一言以蔽之，就是针对构成互联网全链路的软件系统中，以往被无条件信任、但实则仅仅因

每个赛季，我们关注软件供应链中的一个关键环节。在之前结束的赛季中，我们探讨了针对互联网最基础软件设施——Linux操作系统、基础开源软件组件的可能风险。而在

个人电脑在企业的环境之下，有两种角色。作为个人使用的办公、计算、处理设备，对其进行控制权窃取、个人信息窃取，可完成浅层次的攻击；作为企业办公文档编辑的终

威胁定义

Windows上软件的分发一般采用PE二进制程序文件的形式，这决定了从本质上讲，相比于海量开源软件代码，这一赛季中载体内攻击载荷的隐蔽性、开放性更强，威胁场景

IT[redacted]

PE二进制赛季共两轮分站赛。相比于C源代码赛季按照难度递进方式划分的三轮分站赛，PE赛季则针对不同的场景进行设计，两轮比赛分别针对“软件供应链上游（生产者）

软件供应链上游场景

企业作为软件和服务供给侧，软件开发者是上游源头角色，而使用工作电脑进行开发的的工具链（如各类IDE软件、开发库），就是处于软件供应链最上游的存在。

针对个人开发环境，如果能够污染到开发集成工具链，那么对于企业最核心的资产——代码，就能够最直接地获取甚至篡改。因此在这个场景下，我们将聚焦在一个特定的开

软件供应链下游场景

企业员工办公电脑上当然有很多的高价值资产，如各种文档文件；但最重要的资产永远是企业线上环境的稳定和安全，从这角度说，员工直接无条件信赖使用、但潜在风险最

此外，考虑从一个看似无害的客户端软件，从其大量正常操作中区分辨识出可疑的行为，那么最高级的隐藏性就体现为对载体程序原有功能、逻辑的复用。沿此思路，在一个

比赛具体设置

赛题环境

每一轮分站赛，我们都将把目标设定到一到两个代表性软件，以及围绕该软件的若干支撑或扩展程序、组件上。比赛目标设置为Windows 32位。

软件本身可能由一个或多个可执行文件、动态链接库文件组成，相关程序、组件或扩展也为PE文件。所有相关PE文件都可能搭载有恶意代码。作为题目的软件不保证在直接

软件供应链上恶意行为范畴

Windows上二进制程序向来是恶意程序的主要战场，各式各样的恶意行为不胜枚举；本次比赛不排斥使用典型已知恶意代码，但是均紧扣“软件供应链安全”主题以及上述场

类似的，业界与学术界针对PE程序进行分析的方法和研究，也非常丰富，涵盖了静态、动态，人工、自动。各种方法往往有其专长的适用问题域，也可以进行整合，这些都

因此，我们针对本赛季，从题目设计和解题预期两方面，都将不设严格的限制，不给出条例清晰的“考题范围”；但是为了有效引导出题脑洞方向，以及解题队知识积累、方法

- 目标、行为与软件供应链的强关联性。通过污染程序文件，使得执行该程序的个人电脑不可用、文件丢失、被控制等等，均可以达到破坏效果，但是本身仅是病毒木马的
- 非单纯、单点破坏性。典型的破坏行为和特征，仅以一个工作站为攻击目标，造成的危害有限，目标可以恢复状态，单纯破坏行为也容易被早期、中期检出。因此，如果
- 趋利性与隐藏性。除了破坏行为之外，软件供应链污染最符合常理的目标，就是针对目标核心资产的侵害，亦即攻击者己方的获利。但为了保证这样的不当获利最大化，

赛题选用

在先需要特别强调的概念是，类似于C源代码赛季我们看到的，在一个软件系统中，针对特定软件目标发起攻击，不一定来源于相关软件或组件的污染；一旦攻击者可以污染

软件供应链上游攻防战场

在本轮比赛中，我们唯一选定一款开源IDE软件：Code::Blocks，作为目标载体。该软件有扩展机制，并有一定数量的官方、第三方插件，软件本体、插件均为C++语言开发

同时，围绕该软件展开，比赛指定若干开发工具链通用工具，也是该软件中污染代码可能针对的攻击目标、数据窃取渠道，包括开发库，代码版本控制终端软件等；除此以外

软件供应链下游攻防战场

在本轮比赛中，我们选定了两款开源客户端软件为目标载体。

其中一款是一个本地客户端软件ConEmu，它是很多第三方Windows系统上cmd替代软件的底层实现。作为一款终端，其中本身出现文件操作、网络行为都是貌似合理的。

另外一款是一个网络客户端软件eMule，它是一个老牌开源p2p下载管理软件。作为p2p软件，其自然地以磁盘文件访问甚至遍历、下载与本地资源上传、网络端口探测等行

赛题特殊处理与披露

考虑到本次比赛赛题不是传统恶意软件，由主流杀毒软件引擎无法检出，为防止流失到社会上造成真实风险，组织方尽量保证题目不传播、不外泄、可判别。

软件在启动过程中增加了弹框进行"软件特殊场景拷贝、存在恶意代码"的警示，并对启动过程相关函数进行加壳混淆，初步保证无法直接将二进制程序文件被直接流失到外部

```
# Code::Blocks
f8f59f4a417a1cc6006a0826ae8863e64f565df8478657a8635a9d321c908539 *autorevision.exe
22808cb74a5e038958813d999ae214b06d08321f597ee436a1e8c53716c2dccc *cb_console_runner.exe
6eeca307be321870c6a0a0c91f0665649e2470334e7372dd9701d91545fd6660 *codeblocks.dll
8bceffd76ddfab3861c657695765bc1bdc092617cd677cd63a1cf5f7e7fec146 *codeblocks.exe
1339b2de06cf83b2f7c6f83b48f1bc8a3d64f696788293e86c4a0dea4a62274c *abbreviations.dll
f9ab0deb5c87aefec6c32589c640de1a06838d5cdcc5cac7ca251623053bf2d4 *autosave.dll
4517a8bdd5993d2b438cefafe2c06e2dc3baba6b6194bb3a9dd2bf36f764d636 *classwizard.dll
9f1a28d998343a6395c3bb07cc92d893e076f0ba89f50e4f7fc7b622910a7b7f *defaultmimehandler.dll
5cfcb7f143f11e7b152d546da36031a7464e112e0fa056e99a8af9fb563bd01e *debugger.dll
1bc7cc00b2325c358c018e64e5aa8de4cb3db962ccd529929544a33ca262977e *compiler.dll
06e0a16353d50e03e9f7026b58f5317e27c1925aa0465551a4b091dfe87583d3 *EditorConfig.dll
b92b5be5da0fa1b01d823a8dab1b73f45e984c274a0b17b3938495dcd485f976 *FileManager.dll
249ed7e1c91f7289487f28b2dcc759e6eea2615a63a255b906a9f543f0690c23 *openfileslist.dll
f90c8fd13af6347a408ddb968ebc7560240ef053bb9f52f461a25ee33f1b9153 *xpmanifest.dll
# ConEmu
887f4627e0da26a50851634ecef7106bbcc5b1db3a510450cad2757edf4d333d *ConEmu.exe
4f71dbf6f95a3702d81de7eb7f1f7470821bad58d944f922033900b3a58e63ea *ConEmuC.exe
37e66ee0b610e2070bb3eca935805e96a9dc12a49ddf03eca550777d184d05e3 *ConEmuCD.dll
# eMule
09e6a93b6cc560f3d50482415cd95d0dabe993a3a8c8f2912ffb6d73b95d4f66 *emule.exe
```

干货分享

以上进行了本赛季的考量与详细设计陈述。在已经发布的下篇中，对我们本赛季中最能代表场景意图、代表出题脑洞、代表迫切风险的精选题目进行讲解，同时还对头部解是