网络空间威胁对抗与防御技术研讨会
暨 第七届安天网络安全冬训营

下一代威胁检测引擎承载威胁情报

支撑高价值威胁情报落地

安天基础引擎研发部

威胁框架：认知与实践

寒夜远征

寒夜远征

CONTENTS

目 录

寒夜速记

威胁框架：认知与实践

01 威胁情报及其应用

- 变化中的攻击方与防御方
- 满足防御方需求的解决方案
- 威胁情报及其现阶段的应用效果
- 效果不佳的主要原因

变化中的攻击方与防御方

对手在变化

从个体攻击者发展到黑产组织到APT团队

目标在变化

从以经济利益为攻击目标拓展到以基础设施为攻击目标

防御方的需求在变化

- **用户的单点需求：**在最初的威胁对抗中，用户的需求非常纯朴，是要解决黑、白的问题，这样在就可以解决如何判定，如何防御的问题。
- **用户的面的需求：**用户的需求在进一步提升，用户需要解决的问题是：我在面临什么样的威胁？对我的危害是什么？
- **用户的立体需求：**随着APT的到来，用户的需求是要了解对手，谁对我发起了攻击？用了什么手段？我面临什么样的威胁？

满足防御方需求的解决办法

以捕获到一些攻击载荷为例

步骤1 分析攻击的危害

输出**攻击手段**和危害信息

步骤2 判断是否从属已知攻击组织或事件

是

确认**攻击者**身份

否

步骤3 判断攻击载荷之间的关联关系，是否具有同源性

是

作为**重点可疑**观察对象跟踪分析，进一步确认身份信息

否

暂不重点追踪

业界实现的方式

利用ATT&CK分类威胁，输出攻击者使用了哪些手段、可能造成的危害等信息

利用已知**IOC情报信息**对攻击载荷进行身份判定,弥补威胁对抗的时效性和精准指向性

通过关联和聚类，发现可能存在的相关APT组织

威胁情报是什么

- 威胁情报是某种基于证据的知识，包括上下文、机制、标示、含义和能够执行的建议，这些知识与资产所面临已有的或酝酿中的威胁或危害相关，可用于资产相关主体对威胁或危害的响应或处理决策提供信息支持。
- 威胁情报是情报的一种，从情报的本身来看，情报具有三个属性：——《情报学概论》，严怡民

知识性

知识是人的主观世界对于客观世界的概括和反映，这些经过传递的有用知识，按广义的说法， 就是人们所需要的情报。因此，情报的本质是知识，没有一定的知识内容，就不能成为情报。知识性是情报最主要的属性。

传递性

知识之所以成为情报，还必须经过传递，知识若不进行传递交流、供人们利用，就不能构成情报。情报的传递性是情报的第二基本属性。

效用性

人们创造情报、交流传递情报的目的在于充分利用，不断提高效用性。情报的效用性表现为启迪思想、开阔眼界、增进知识、改变人们的知识结构、提高人们的认识能力、帮助人们去认识和改造世界。情报为用户服务，用户需要情报，效用性是衡量情报服务工作好坏的重要标志。

现有威胁情报应用方式的真实效果

业内常见的威胁情报应用方式

IOC情报
（对攻击方所使用的装备或基础设施的**指向能力**）

结合

传统反病毒引擎
（海量的恶意代码的检测和辨识能力）

期望达成

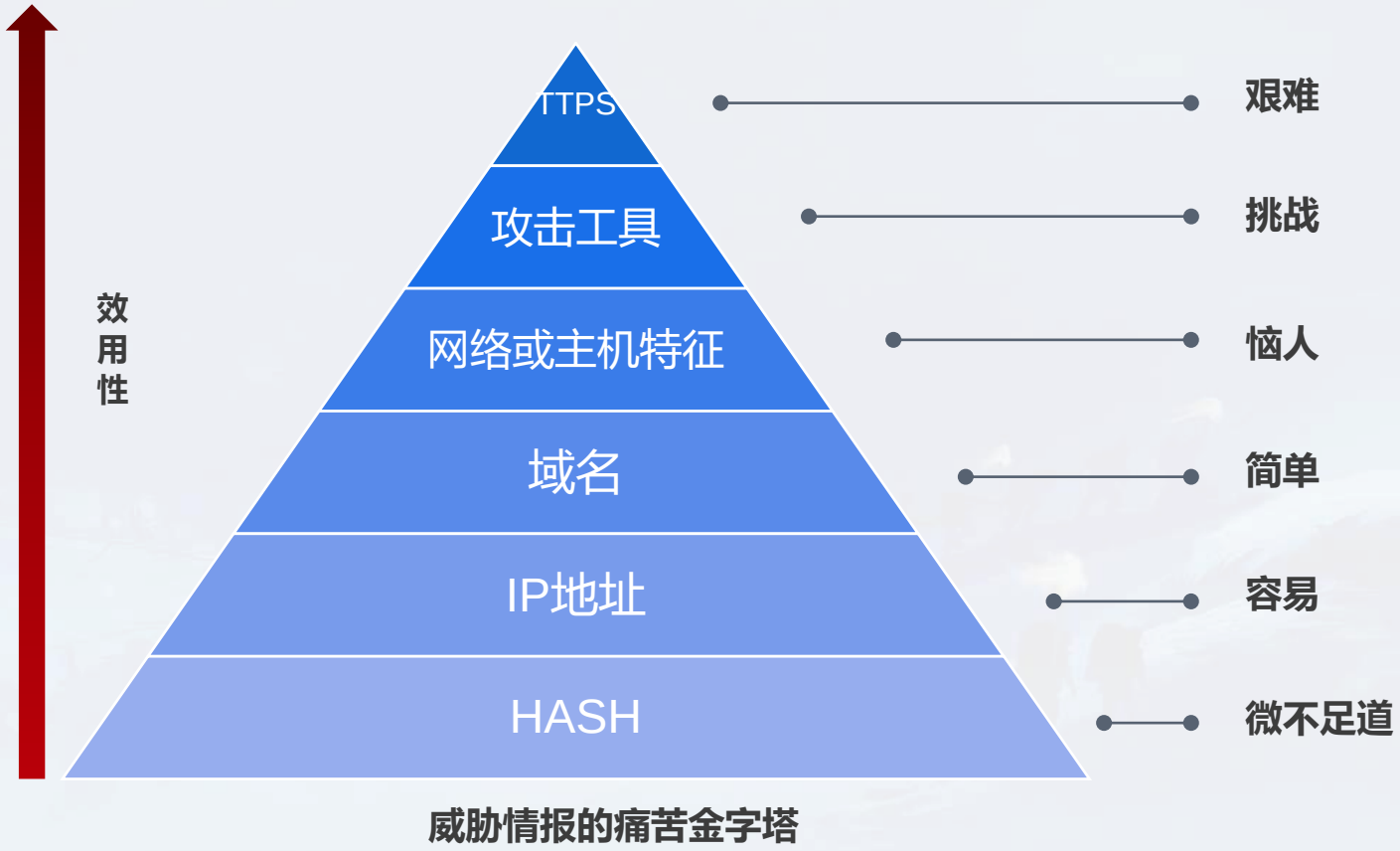
发现、阻断和猎杀高级威胁行动

这种检测能力组合虽然对类似海莲花、白象、绿斑一类的APT攻击具有一定的价值，但在过去十年内对于对抗来自更高水平的威胁行为体的活动，其实收效甚微。针对类似毒曲II、方程式等高级威胁行动或组织的发现、阻断和猎杀活动中，这些情报几乎无法发挥任何作用。

效果不佳的主要原因1—效用性

- **IOC情报检测的鲁棒性较差**

威胁情报试图通过IOC规则来为高级威胁对抗提供更好的指向性，这是目前阶段威胁情报应用的一种**常态**，但对于超高能力的网空威胁活动，实际**IOC几乎是无效**的。



威胁情报的痛苦金字塔

效果不佳的主要原因1 —效用性

木马名称： Stuxnet

出现时间： 2010年6月

主要功能：攻击用于数据采集与监控的工业控制系统。

描述：震网使用模块种类繁多，自身逻辑复杂，利用了多个零日漏洞，通过一套完整的入侵和传播流程，突破工业专用局域网的物理限制，攻击用于数据采集与监控的工业控制系统。

分类	数量	说明
DROPPER	1200+	~WTR4132.tmp,其STUB节的内容变换、样本自身代码的升级与发布、人工二进制更改，组合操作生成多个样本
DROPPER LOADER	460+	~WTR4141.tmp，通过少量原始样本，经过二进制修改、签名、追加损坏签名、签名后继续追加文件等操作，造成样本量增加
LNK	20+	漏洞利用载荷，用于加载恶意DLL文件
其他文件	100+	CAB文件、驱动文件、Step 7使用的DLL等
编译器版本	10+	多版本编译器表明工程代码经过多人编译，生成母体样本基数变大

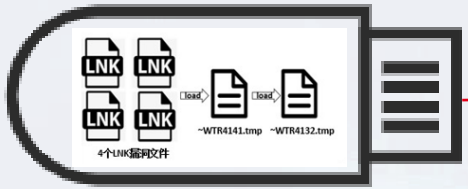
样本1	样本2	异同	原因
BF6E9CBCDA5EF3	F6E836E63974	时间戳相同，代码段，导入表Hash均不同，但代码对比一致	输出文件类型不同，EXE和DLL
F48F28C1539DFC	F6E836E63974	时间戳、导入表HASH、代码段HASH相同，文件HASH不同	Stub包裹内容不同
02BC5EDC93859E	3B8EA5381D81	时间戳、导入表HASH、代码段HASH相同，文件大小不同	有签名和无签名
31E2FD7A131B71	C77CB014E669	时间戳、导入表HASH、stub段hash相同、文件大小相同，代码段HASH不同	链接器版本不同
0C8AB2873E1399	98FBEBD88830	时间戳、导入表HASH相同，代码段HASH不同	无关代码0xFF填充
C77CB014E6694D	5A379BB1480D	时间戳、导入表HASH、代码段HASH、STUB段HASH、文件大小相同，文件HASH不同	PE头部无关数据被0xFF填充
0B5FD57A4F7008	3867ABE7E8F9	除时间戳外其余均相同	

震网样本集差异分析

——《对Stuxnet蠕虫攻击工业控制系统事件的综合分析报告》

效果不佳的主要原因1 —效用性

Dropper样本落地时写入目标配置导致文件变化，HASH也就随之改变



被感染的U盘

主机

内存解密

主DLL

释放

配置文件mdmcpq3.PNF

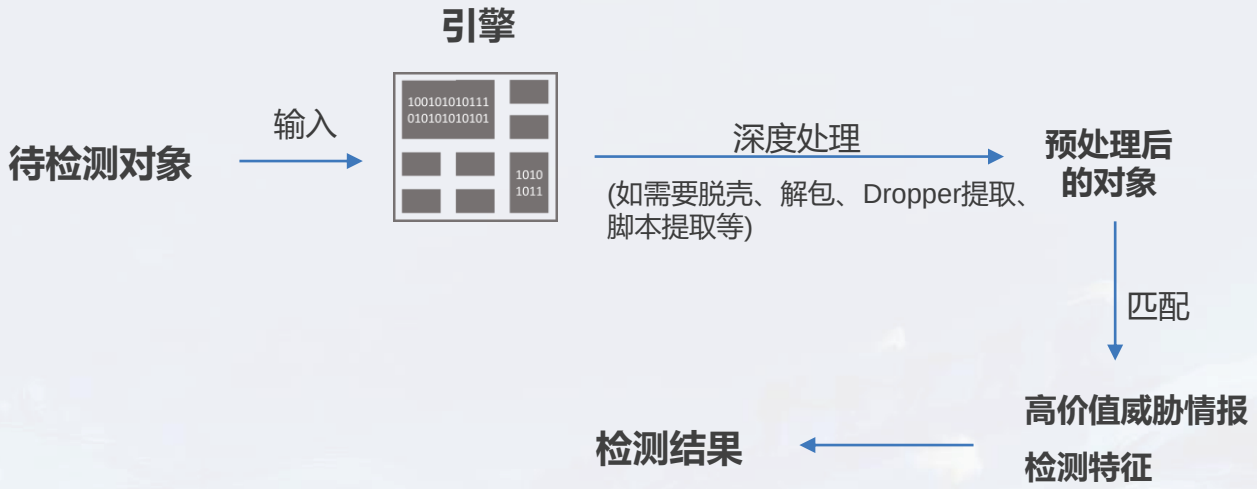
偏移	长度	解释
+00	Dword	配置文件起始标志
+04	Dword	配置文件头长度
+08	Dword	配置文件校验和
+0C	Dword	配置文件长度
+68	Dword	标志位，若为 0 则检查+70 处标志（为 1 则直接感染 USB）
+70	Dword	标志位 若为 0 则检查+78 处的时间戳
+78	Qword	终止感染 USB 的时间
+80	Dword	在 USB 上需存在的文件数
+8C	Qword	终止时间
+A4	Qword	开始感染时间（超过此时间 21 天，则停止对 USB 进行感染）
…	…	…

配置文件详细信息

效果不佳的主要原因2 —传递性

- **高价值的指向性条件没有转化为情报**

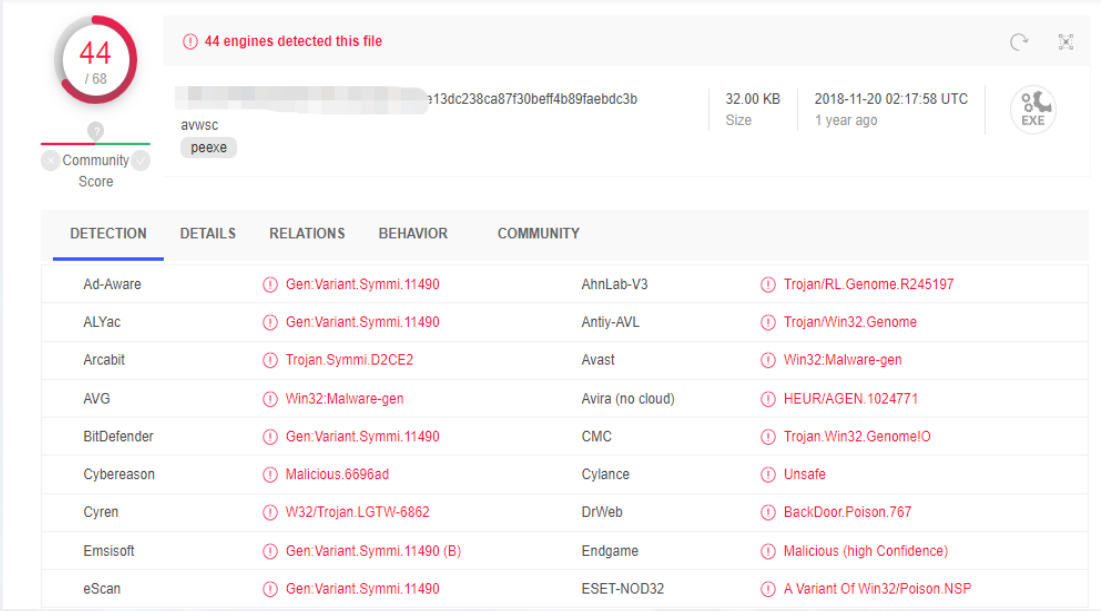
实际上，还存在着比IOC情报更具指向性的一些条件，例如与环境相关的路径、文件名、与身份相关的PDB等等信息，但是这些条件转化为情报需要依赖引擎对检测对象进行深度处理，而安全厂商大多选择OEM其它厂商的引擎，这些OEM来的引擎不对外提供深度拆解能力，因此这些条件无法成为情报。



效果不佳的主要原因3 —知识性

- **传统引擎本身具有局限性**

1. 原有的知识工程体系，不能满足直接定位到高级网空威胁行为体的精准要求
2. 单一病毒名输出的方式缺乏**知识性**，无法满足用户对于威胁想要深入了解的需求。在现有防御体系中，没有把传统引擎当作一个关键环节来看待，原因主要是普遍把引擎作为一个基础支撑能力看待，作为**黑箱**使用，没有把引擎的输出作为**知识供给**。



VirusTotal网站对某高级威胁样本的检测结果

寒夜速记

威胁框架：认知与实践

02 可承载情报的威胁检测引擎

- 引擎承载威胁情报
- 情报生产流程
- 引擎承载情报的应用价值

安天引擎承载威胁情报

- 开放情报承载能力、可输入外部情报，使高价值情报可以落地
- 支持多种输入与输出，可输出ATT&CK框架、TCTF框架、向量、组织描述等知识类信息

下一代威胁检测引擎

- 全面格式识别
- 深度预处理
- 虚拟化执行等机制

承载

高价值威胁情报

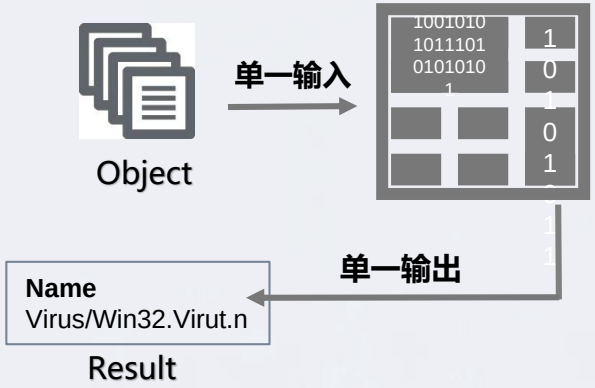
- 基础信息（字符串、编码过的二进制）
- 属性信息（格式、编译器、壳、包、版本信息）
- 结构信息（PE结构、复合文档结构、自定义结构）
- 身份信息（开发者、登录ID、密码、数字签名）
- 环境信息（注册表、路径、GUID）
- 攻击技术（执行、持久化、提权、防御规避、凭证访问、发现、横向移动、收集）

达成

客户防御场景下的可消费情报

- 对攻击手段的揭示
- 对高级威胁攻击方身份的揭示
- 对高级威胁的发现、阻断

安天引擎能力的维度--多种输入输出对象

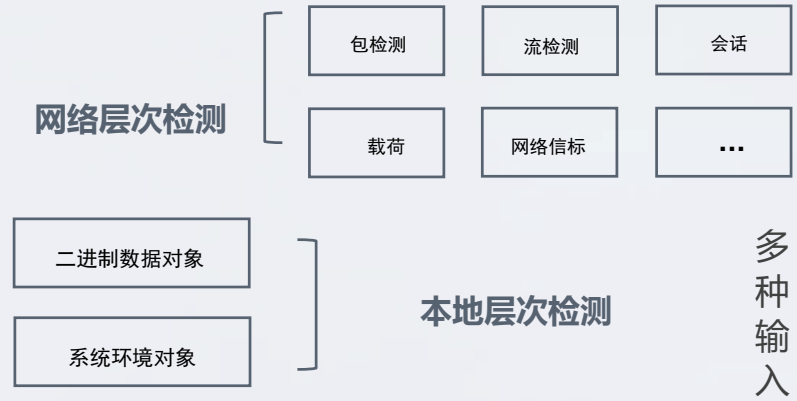


✓ **传统引擎**

主要是以单一对象为输入，以单一结果为输出。而随着威胁的进一步演进和泛化，威胁检测已不能仅仅停留在对单一对象进行鉴定上。

✓ **AVLSDK威胁检测引擎**

多种输入对象，多种输出结果。威胁检测多样化。



输出 1

- 黑白
- 多向量
  - 识别信息
  - 基础信息
  - 附加信息
  - 行为信息
- 核心行为
  - 远控 广告
  - DDoS 下载
  - 窃取
- 威胁行为
  - 传播 伪装
  - 隐蔽 对抗
  - 信息获取 攻击

输出 2

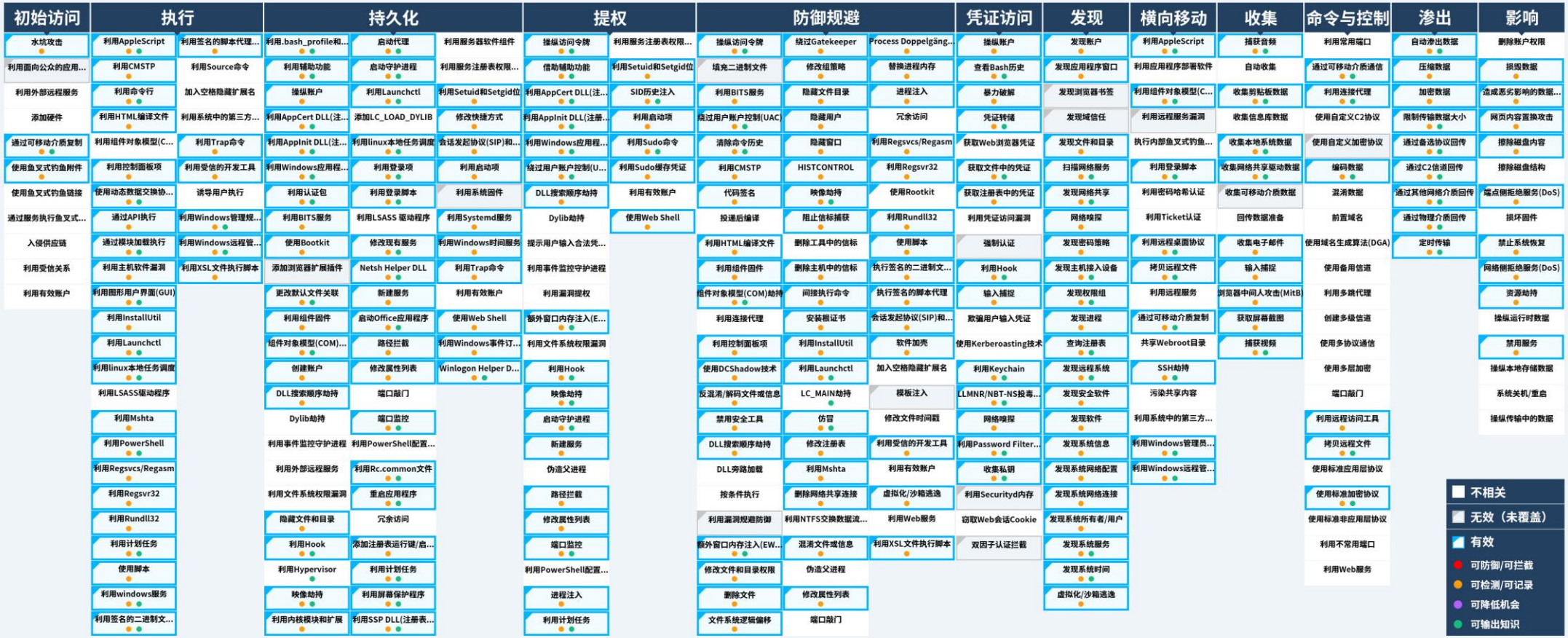
- 组织名称
- 组织简介
- 攻击领域
- 攻击方式
- 活跃时间
- 利用漏洞

输出 3

ATT&CK框架信息

初始访问、执行、持久化、提权、防御规避、凭证访问、发现、横向移动、收集、命令控制、渗透

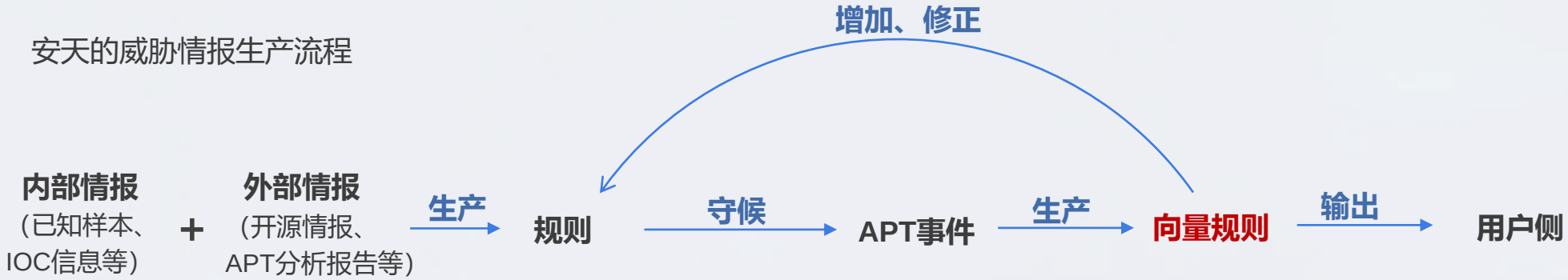
安天引擎能力的维度—对ATT&CK框架的覆盖



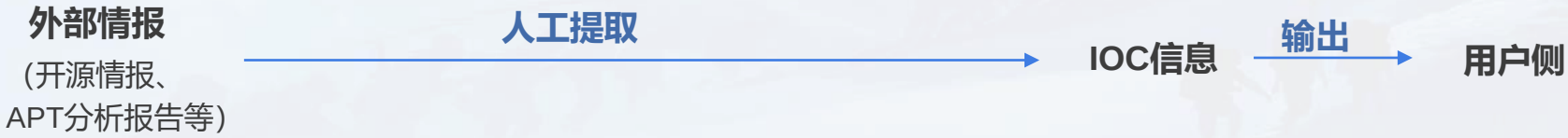
三种强预处理能力引擎的对比

情报的维度

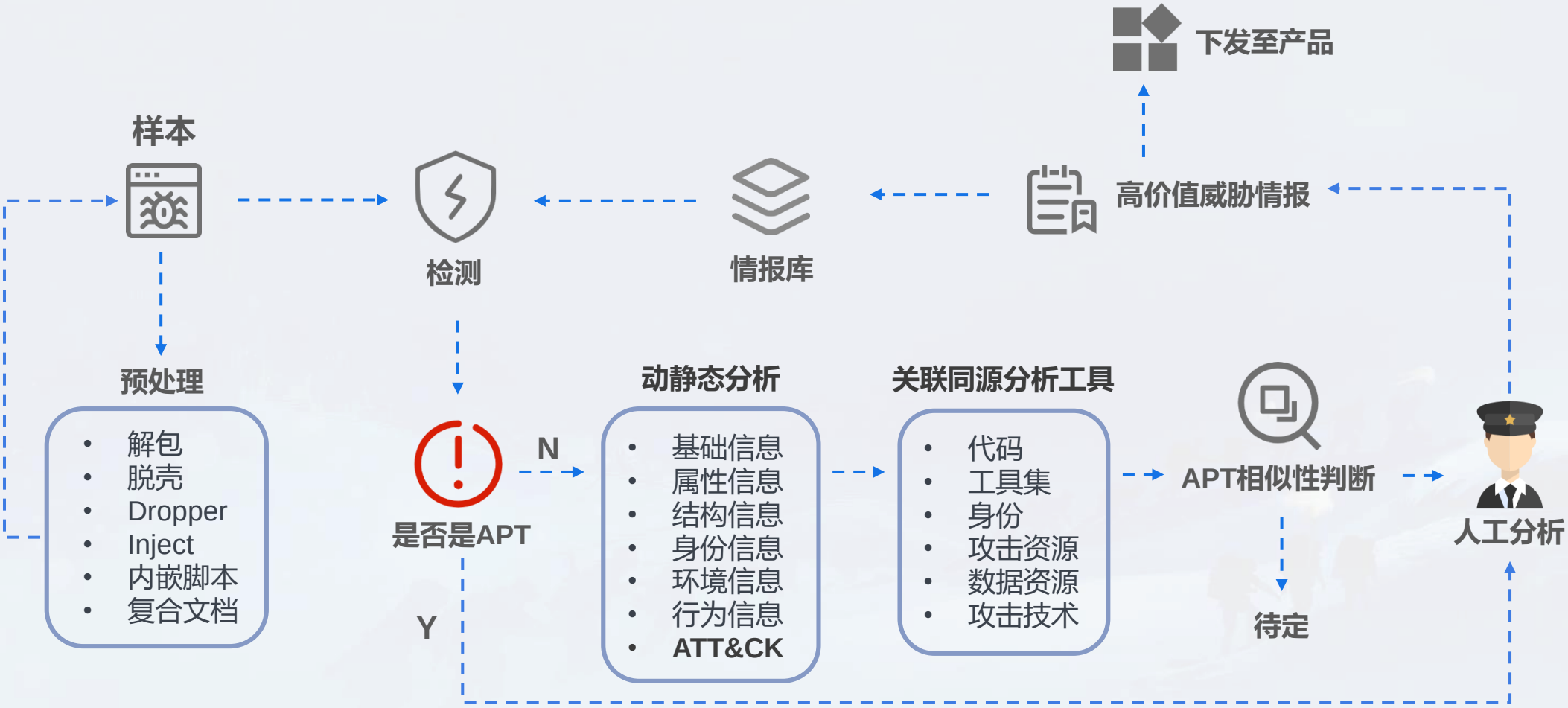
- 安天的威胁情报生产流程



- 以开源收集为主、缺乏分析生产能力的厂商的威胁情报生产流程



情报生产流程


下发至产品
样本
检测
情报库
高价值威胁情报
预处理
解包
脱壳
Dropper
Inject
内嵌脚本
复合文档
是否是APT
N
Y
动静态分析
基础信息
属性信息
结构信息
身份信息
环境信息
行为信息
ATT&CK
关联同源分析工具
代码
工具集
身份
攻击资源
数据资源
攻击技术
APT相似性判断
待定
人工分析

引擎承载情报的应用价值—更好的知识性

078d12eb9fc2b1665c0cc3001448b69b 1487d1dc13314bf04
0796ff1096f7456ef37d81a5b846b61b 1489d2adf0328b6d7
0defd4bda646b1fb058c3abd2e1128e 153ac7591b9326ee6
0837671230288d68b99866197d79646b 15552ebdc4ebe5b4d
08a3776a2c40e569f645a62fdd2fcac3 1579467859b48085b
08f7ead1513bb921c9cdee334a370866 158ff697f8e609316
09947ba52932d10d3c859511a6d31e8f 15e45c24dbe603402
09cdbd5273640ab23112b719c65e4902 166044bf473fc262e
0a0bcd8beb77e67a28a325d8d2a00254 1676ded041404671b
0acdfd9ef4ed3e3f3d9d011aa5e7cd03 168f2c46e15c9ce0b
0ad9583aefede1f355759e0b674930cb 16c11b381cff35283
0b29cd6fc38c0459507e670e9c4547e0 16c140fb61b6d22e0
0b38f87841ed347cc2a5ffa510a1c8f6 16ff5f646196cf297
0b88f197b4266e6b78ea0dcb9b3496e9 176e2277be875e55a
0ba19063dea4ccae0afcd4208781f16b 1785f20ad4883fee5
0bbe6cab66d76bab4b44874dc3995d8f 17a31d1075ebce41b
0c0eb91f318da38e6684bd5250f68378 187dc6afa65cbdd8e
0c2cbfbe3c93b3502f9a60f5fa1188ad 18b9e5fad0f015a0c
0cace87b377a00df82839c659fc3adea 18bc477fa12048fab
0d466e84b10d61031a62affcfff6e31a 1972ae990751fa1b1
0d5956dac2ac56f292ee8fa121450973 1981cc08cdadc971e

Operation Hangover: Unveiling an Indian Cyberattack Infrastructure
Appendixes

	MD5	Kaspersky	Microsoft
1			
2	08a3776a2c40e569f645a62fdd2fcac3	Trojan.Win32.Agent.hwjb	Worm:Win32/Foler.C
3	0acdfd9ef4ed3e3f3d9d011aa5e7cd03	Trojan-Downloader.Win32.Agent.hhgu	NotFound
4	15e45c24dbe603424fcffe4c358556b	Worm.Win32.Agent.ali	Worm:Win32/Foler.C
5	0cace87b377a00df82839c659fc3adea	Trojan.Win32.VB.cefz	NotFound
6	17a31d1075ebce41ba48a9efacb79d28	Exploit.Win32.CVE-2012-0158.j	NotFound
7	18bc477fa12048fab8ec93d5ff942cf5	Trojan.Win32.Bahamut.o	Trojan:Win32/Rimod!gmb
8	1972ae990751fa1b1532aa792bd5c160	Trojan.Win32.AFosT.ity	TrojanSpy:Win32/Dayek.A
9	1487d1dc13314bf0431792b37ec67e2d	HEUR:Trojan.Win32.HangOver.gen	NotFound
10	16c11b381cff35283b879ec1a84f72e4	Virus.Win32.Virut.ce	Worm:Win32/Foler.C
11	0ad9583aefede1f355759e0b674930cb	Trojan.Win32.Agent.hnll	NotFound
12	09947ba52932d10d3c859511a6d31e8f	Trojan-Spy.Win32.Agent.chfl	NotFound
13	0ba19063dea4ccae0afcd4208781f16b	Trojan.Win32.AFosT.flo	TrojanSpy:Win32/Dayek.A
14	0c0eb91f318da38e6684bd5250f68378	Trojan-Spy.Win32.KeyLogger.acqh	TrojanSpy:Win32/Hanove!rfn
15	166044bf473fc262ed97283c6e157eb5	Trojan-Spy.Win32.KeyLogger.acqh	NotFound
16	1489d2adf0328b6d7b42170095f966c9	Trojan-Spy.Win32.KeyLogger.acqh	TrojanSpy:Win32/Hanove.C
17	168f2c46e15c9ce0ba6e698a34a6769e	Trojan-Downloader.Win32.AutoIt.xd	NotFound
18	0796ff1096f7456ef37d81a5b846b61b	Trojan-Spy.Win32.KeyLogger.adar	TrojanSpy:Win32/Dayek.A
19	1981cc08cdadc971e28768dc04d98637	Trojan-Dropper.Win32.Dycler.vrp	NotFound
20	078d12eb9fc2b1665c0cc3001448b69b	HEUR:Trojan.Win32.HangOver.gen	TrojanSpy:Win32/Hanove.F
21	0bbe6cab66d76bab4b44874dc3995d8f	HEUR:Trojan.Win32.HangOver.gen	TrojanSpy:Win32/Hanove.F
22	0c2cbfbe3c93b3502f9a60f5fa1188ad	Worm.Win32.Agent.ali	Worm:Win32/Foler.C
23	0d466e84b10d61031a62affcfff6e31a	Trojan-Downloader.Win32.VB.bkrb	TrojanDownloader:Win32/Adodb.A
24	16ff5f646196cf29792f5b159d1288b8	Trojan-Spy.Win32.KeyLogger.acqh	NotFound
25	0b88f197b4266e6b78ea0dcb9b3496e9	Trojan-Spy.Win32.Agent.chju	TrojanSpy:Win32/Hanove.A
26	0d5956dac2ac56f292ee8fa121450973	HEUR:Trojan.VBS.SAgent.gen	NotFound
27	187dc6afa65cbdd8ee87a58271b56864	Trojan.VBS.Starter.dy	NotFound
28	0a0bcd8beb77e67a28a325d8d2a00254	Worm.Win32.Agent.ali	Worm:Win32/Foler.C
29	176e2277be875e55ad7211ff5e8df7a5	Trojan-Spy.Win32.KeyLogger.acqh	NotFound
30	153ac7591b9326ee63cd36180d39665e	HEUR:Trojan.Win32.HangOver.gen	Backdoor:Win32/Hanove.A
31	158ff697f8e609316e2a9fbe8111e12a	NotFound	NotFound
32	0b29cd6fc38c0459507e670e9c4547e0	Trojan-Downloader.Win32.Agent.gzkf	NotFound
33	1785f20ad4883fee549f0aec5d20aaca	Trojan-Spy.Win32.KeyLogger.acqh	TrojanSpy:Win32/Hanove!rfn
34	0837671230288d68b99866197d79646b	Trojan-Spy.Win32.KeyLogger.adar	TrojanSpy:Win32/Dayek.A
35	16c140fb61b6d22e02aa2b04748b5a34	HEUR:Trojan.Script.Generic	TrojanSpy:Win32/Linog.A
36	09cdbd5273640ab23112b719c65e4902	Backdoor.Win32.RShot.vsl	NotFound
37	15552ebdc4ebe5b4d2f71ab2d2e574cb	Trojan-Spy.Win32.Agent.cpfz	NotFound
38	08f7ead1513bb921c9cdee334a370866	Trojan-Spy.Win32.KeyLogger.acqh	TrojanSpy:Win32/Hanove!rfn
39	07defd4bda646b1fb058c3abd2e1128e	Trojan.Win32.Agent.hwjb	Worm:Win32/Foler.C
40	0b38f87841ed347cc2a5ffa510a1c8f6	Trojan-Spy.Win32.Agent.chfl	NotFound

	MD5	所属组织	所属组件	功能
1				
2	09cdbd5273640ab23112b719c65e4902	白象	Backdoor	白象所用的后门软件，可执行shell命令
3	0796ff1096f7456ef37d81a5b846b61b	白象	KeyLogger	白象组织使用的KeyLogger工具，具有开机自启功能
4	0837671230288d68b99866197d79646b	白象	KeyLogger	白象组织使用的KeyLogger工具，具有开机自启功能
5	0d466e84b10d61031a62affcfff6e31a	白象	Dropper	白象组织使用的Dropper文件，释放VBS脚本并执行
6	1489d2adf0328b6d7b42170095f966c9	白象	Downloader	白象组织使用的Downloader文件，下载恶意载荷并执行，窃
7	078d12eb9fc2b1665c0cc3001448b69b	白象	Ron Tools	白象组织使用的后门工具，因其pdb信息而命名，下载恶意载
8	153ac7591b9326ee63cd36180d39665e	白象	Http	白象组织使用的联网组件


```
"result": {
  "desc": "0796ff1096f7456ef37d81a5b846b61b",
  "malware_info": {...},
  "format": {
    "id": 22,
    "name": "PE"
  },
  "packer": {...},
  "hformat": {
    "id": 1,
    "name": " BinExecute/Microsoft.EXE[:X86] "
  },
  "sfx": {...}
},
"knowledge": {
  "sid": 46,
  "malware_name": "Trojan[Spy]/Win32.KeyLogger",
  "type": "APT",
  "description": "白象组织使用的KeyLogger工具，具有开机自启功能",
  "tags": [
    "APT",
    "KeyLogger",
    "开机自启"
  ],
  "group": [
    {
      "name": "白象",
      "info": {
        "id": "G003",
        "publish_name": "白象",
        "alias_name": [
          "Chinastrats",
          "Patchwork",
          "Sarit",
          "Quilted Tiger",
          "Dropping Elephant",
          "APT-C-09",
          "Monsoon"
        ],
        "target_industry": [
          "政府",
```


某白象分析报告

各反病毒引擎厂商检测结果

安天下一代威胁检测引擎输出结果

引擎承载情报的应用价值—更好的传递性

样本 MD5：	2C3B9984BE2D8609F83D10171A4F1059
样本类型：	PE32

解密字符串

样本中关键字符串经过加密处理，解密的Key为:dc67f@#$%hlsdfg

关键解密算法如下：

```
004756F1  .  BB 01000000  mov ebx,0x1
004756F6  >  8B45 FC      mov eax,[local.1]
004756F9  .  8A4418 FF    mov al,byte ptr ds:[eax+ebx-0x1]
004756FD  .  8845 EB      mov byte ptr ss:[ebp-0x15],al
00475700  .  F645 EB E0   test byte ptr ss:[ebp-0x15],0xE0
00475704  .  74 15        jz short ispecsr.0047571B
00475706  .  8D45 FC      lea eax,[local.1]
00475709  .  E8 02F8F8FF  call ispecsr.00404F10
0047570E  .  8B55 E4      mov edx,[local.7]
00475711  .  8A1438       mov dl,byte ptr ds:[edx+edi]
00475714  .  3255 EB      xor dl,byte ptr ss:[ebp-0x15]
00475717  .  885418 FF    mov byte ptr ds:[eax+ebx-0x1],dl
0047571B  >  47           inc edi
0047571C  .  3B7D EC      cmp edi,[local.5]
0047571F  .  75 02        jnz short ispecsr.00475723
00475721  .  33FF         xor edi,edi
00475723  >  43           inc ebx
00475724  .  4E           dec esi
00475725  .  75 CF        jnz short ispecsr.004756F6
```

白象报告中的关键信息

```
rule APT_WhiteElephant_Dropper
{
    meta:
        data = "Sept. 26, 2018"
        description = "Delphi Dropper, According to the key and a piece of byte-code to match the malware."
        md5 = "2c3b9984be2d8609f83d10171a4f1059"

    strings:
        $header = "MZ"
        $string_key = "dc67f@#$%hlsdfg" fullword
        $hex_uncryptcode = {8B F0 85 F6 7E 36 BB 01 00 00 00 8B ?? ?? 8A 44 18 FF 88 45 ?? F6 45 ?? E0 74 15 8D 45

    condition:
        @header == 0 and uint32(uint32(0x3C)) == 0x00004550 and (1 of ($string*) and 1 of ($hex*))
}
```

提取的yara规则

引擎承载情报的应用价值--更好的效用性

文件HASH：7c498b7ad4c12c38b1f4eb12044a9def

- **卡巴斯基输出** Backdoor.Win32.Agent.mytihl
- **ESET输出** Win32/Poison.NOL
- **安天下一代威胁检测引擎配合情报平台的输出结果**

组织名称：绿斑
别名：APT-C-01，毒云藤
攻击目标：中国
攻击领域：政府,军事,科研
攻击方式：钓鱼邮件，水坑攻击
活跃时间：2011年,2012年,2013年,2014年,2017年
利用漏洞
CVE-2012-0158,CVE-2014-4114,CVE-2017-8759,CVE-2017-0199
组织简介
2018年9月安天实验室曝光了绿斑组织，该组织至少从2007年开始活跃，擅长对目标实施鱼叉攻击和水坑攻击、植入修改后的ZXShell、Poison Ivy、XRAT商业木马，并使用动态域名作为其控制基础设施。

普通引擎仅能将其认定为商马，安天的引擎可以依靠情报判断出该样本从属绿斑组织

文件属性信息
文件格式：PE文件
文件版本信息
文件名：avwsc.exe
产品名：AntiVir Desktop
公司名：Avira GmbH
文件结构信息
导入表哈希：
F93AFE4A0FB30B1293FCAA32DDAF59F1
时间戳：1325650785
身份信息
上线ID：motices
上线密码：ps135790
互斥体：)!qacA.I1
解密信息
解密偏移=0x628B 解密方式=异或 密钥=0x22

引擎承载情报的应用价值--更好的效用性

从普通商马样本中分辨出绿斑组织样本

md5	时间戳	域名	上线密码	mutex
425336C6696AD1E404622A6EDA99F2BF	2012-01-04 12:19:45(+0800)	netlink.VizVaz.com	hook32wins	kdraqa.2a
785b24a55dd41c94060efe8b39dc6d4c	2012-01-04 12:19:45(+0800)	netlink.VizVaz.com	hook32wins	)!VoqA.I4

MD5	上线密码
5ee2958b130f9cda8f5f3fc1dc5249cf	#My43@92
7639ed0f0c0f5ac48ec9a548a82e2f50	@1234@
250c9ec3e77d1c6d999ce782c69fc21b	admin
f3ed0632cadd2d6beffb9d33db4188ed	admin
9b925250786571058dae5a7cbea71d28	ftp1234
ae004a5d4f1829594d830956c55d6ae4	ftp1234
785b24a55dd41c94060efe8b39dc6d4c	hook32wins
a73d3f749e42e2b614f89c4b3ce97fe1	ftp443
fccb13c00df25d074a78f1eeeb04a0e7	ftp1234
36c23c569205d6586984a2f6f8c3a39e	kkbox55
81e1332d15b29e8a19d0e97459d0a1de	kkbox55
7c498b7ad4c12c38b1f4eb12044a9def	ps135790
76782ecf9684595dbf86e5e37ba95cc8	updatewin
c31549489bf0478ab4c367c563916ada	updatewin

上线密码命中绿斑集合

安天收集的绿斑上线密码集合

寒夜速记

威胁框架：认知与实践

03 应用案例

- 适用场景
- 应用案例

适用场景

- 安天的高价值威胁情报可以在所有嵌入安天AVLSDK威胁检测引擎的场景下工作，主要面向对高级威胁检测有需求的用户、面向监管型用户、面向合作伙伴。

- **检测产品**

形成针对于攻击者的控制通道、传输通道的检测及拦截能力，在流量检测监测设备上提升威胁检测的深度。

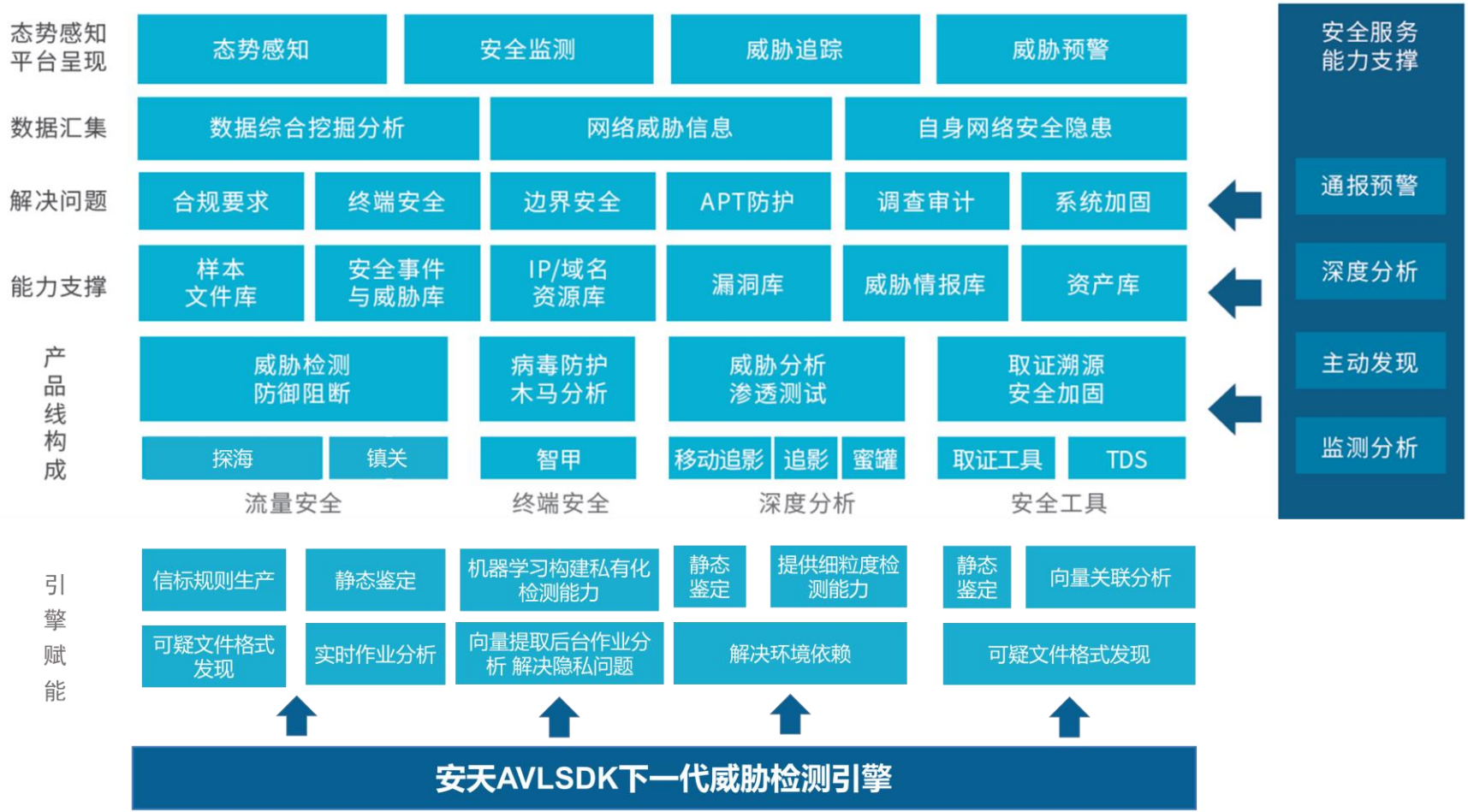
- **分析产品**

形成有效判定能力，其知识化的输出可以让分析人员快速了解威胁，使分析人员聚焦于高等级威胁攻击载荷深入分析层面。

- **人工分析作业**

精准的检测结果也可以让安全运维人员从海量威胁事件中快速定位高等级威胁，知识化的输出能力可以让其理解威胁并快速进行进一步的响应。

安天下一代威胁检测引擎对安天全线产品的支撑



震网样本实例1—生产具有指向性的威胁情报


```
"MD5":"2BDA3159666B29BF6F912A69B9325435"
"malware_name": "Worm/Win32.Stuxnet"
"tag":"Dropper"
```


特异性规则

- 基础信息
- 基础信息
- 结构信息

解密算法

```
do
{
  v2 = 0;
  if ( a2 )
  {
    do
    {
      *(_BYTE *)(v2 + a1) ^= -106 * (_BYTE)v2;
      ++v2;
    }
    while ( v2 < a2 );
  }
  v3 = 0;
  if ( v6 )
  {
    do
    {
      *(_BYTE *)(v3 + a1) ^= *(_BYTE *)(((a2 + 1) >> 1) + a1 + v3);
      ++v3;
    }
    while ( v3 < v6 );
  }
  for ( result = a2 - 1; result >= 1; --result )
    *(_BYTE *)(result + a1) -= *(_BYTE *)(result + a1 - 1);
  --v5;
}
while ( v5 >= 0 );
```

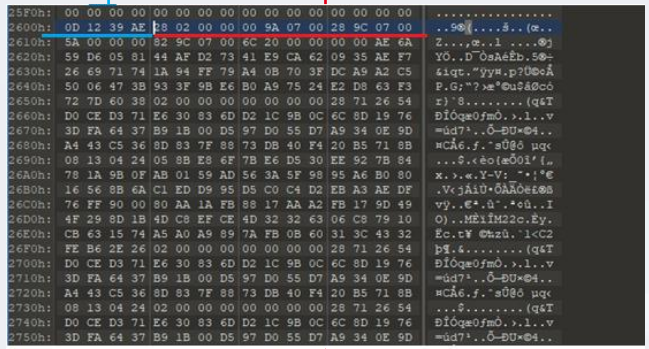
解密密钥

```
*(_DWORD *)a1 = a1 ^ 0xAE1979DD;
*(_DWORD *)(a1 + 4) = 0;
*(_DWORD *)(a1 + 12) = sub_10002334;
return 0;
```

在进行.stub节解密时，会读取解密配置信息
解密数据的Magic: AE39120D，后面是释放文件的相对偏移，及释放文件的长度

Magic:
0xAE39120D

释放文件的相对偏移及长度



震网样本实例1—引擎输出结果


```
"result": {
  "desc": "2BDA3159666B29BF6F912A69B9325435",
  "malware_info": {...},
  "format": {...},
  "packer": {...},
  "hformat": {
    "id": 1,
    "name": " BinExecute/Microsoft.EXE[:X86] "
  },
  "sfx": {...}
},
"knowledge": {
  "sid": 40,
  "malware_name": "Worm/Win32.Stuxnet",
  "type": "APT",
  "description": "震网Dropper样本，搜索.stub节，解密并执行,该节包含Stuxnet
    DLL文件，该节包含Stuxnet
    DLL文件，这个DLL包含了stuxnet的所有功能。同时，该文件还被用作用户模式rookit，用于
    隐藏stuxnet文件。具有挂钩API行为。",
  "tags": [
    "APT",
    "Dropper"
  ],
  "group": [
    {
      "name": "Equation Group",
      "info": {
        "id": "G001",
        "publish_name": "Equation Group",
        "alias_name": [
          "Tilded Team",
          "Lamberts",
          "APT-EQGRP",
          "方程式组织"
        ],
```

```
"target_location": [
  "巴基斯坦",
  "阿富汗",
  "印度",
  "伊朗",
  "伊拉克"
],
"member": [],
"org_nature": "超高级能力国家/地区行为体",
"ttp": {
  "attack_method": [
    "U盘摆渡",
    "漏洞利用"
  ],
  "persistence_method": "防火墙固件植入",
  "algorithm": [
    "AES"
  ]
},
"purpose": [
  "窃密",
  "破坏"
],
"location": [
  "美国"
```


安天下一代引擎对震网样本输出信息

震网样本实例1--产品界面

APT攻击组织"方程式"情报分析报告

组织信息 | 意图及目标 | 攻击活动 | 技术战术过程

①组织信息

方程式(Equation Group)

性质： 超高能力国家/地区行为体　　别名： Equation/方程式/方程式集团/EquationGroup

归属地： 美国　　成员： 未知

首次公开： 2015-02-16 00:00:00　　最后活跃： 2019-12-20 14:48:20

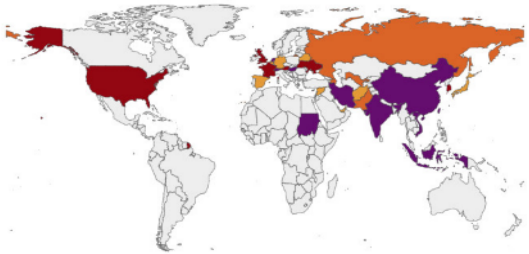
组织描述： 方程式组织是具有美国背景的超高能力国家/地区行为体，又名Tilded Team、Equation Group等，由卡巴斯基于2015年2月16日首次披露，是一个活跃了近20年的攻击组织。该组织主要针对伊朗、中东、中国、印度、俄罗斯、德国、西班牙、韩国等国家/地区，针对工业控制系统、SWIFT服务提供商、核工业、教育、政府、金融、科研、运营商、网络安全等行业进行破坏、修改、窃密、监视等攻击行动。该组织主要采用零日漏洞利用、U盘摆渡攻击、数十种常见品牌硬盘固件修改、攻击载荷原子化、多种加密算法、安全软件规避、持久化等攻击手法，利用的漏洞涉及CVE-2010-2568、CVE-2011-3402、CVE-2015-2360、打印机后台程序服务漏洞（MS10-061）、快捷方式文件解析漏洞（MS10-046）、RPC远程执行漏洞（MS08-067）等，使用EquationLaser、EquationDrug、DoubleFantasy、TripleFantasy、Fanny、GrayFish等攻击装备。

标签： 方程式　APT

研究报告：
2019-06-01 "方程式组织"攻击SWIFT服务提供商EastNets事件复盘分析报告 https://www.antiy.cn/research/notice&report/research_report/20190601.html
2017-01-26 安天图解方程式组织标志式主机作业 https://www.antiy.cn/research/notice&report/research_report/663.html

②意图及目标

"方程式"组织主要针对伊朗、中东、中国、印度、俄罗斯、德国、西班牙、韩国等国家/地区，针对工业控制系统、SWIFT服务提供商、核工业、教育、政府、金融、科研、运营商、网络安全等行业进行破坏、修改、窃密、监视等攻击行动。其攻击意图包括窃密、破坏、获取系统信息、系统破坏、修改可编程逻辑控制器（PLC）的代码、窃取机密信息、修改数据、物理影响、收集信息、修改硬盘固件、隐藏、窃取信息、修改PLC、监视、控制、间谍活动、收集受害主机信息、获取基础设施等。



③攻击活动

研究人员发现了2014年出现的恶意软件Stuxnet新组件，这揭示了参与Stuxnet恶意软件的早期开发的第四个组织。此次发现的Stuxshop新模块与另一种名为Flowershop的恶意软件家族与存在关联，Stuxshop是一个旨在提供基本C&C和登记计划功能的模块，此外所发现的新版本Stuxnet的Duqu 1.5、Flame 2.0也与Flowershop存在关联。Flowershop于2015年被发现，它与潜在方程式工具Territorial Dispute（TaDi）有关。

④威胁分析

载荷分析：

MD5:	2BDA3159666B29BF6F912A69B9325435	威胁名称：	Worm/Win32.Stuxnet
组织信息：	方程式 方程式组织是一个美国的超高能力国家行为体，又名Tilded Team、Equation Group等，由卡巴斯基于2015年2月16日首次披露，是一个活跃了近20年的攻击组织。该组织主要针对伊朗、中东、中国、印度、俄罗斯、德国、西班牙、韩国等国家/地区。		
框架信息：	Tided	工具组件：	Stuxnet Dropper
威胁描述：	震网Dropper样本，搜索.stub节，解密并执行,该节包含Stuxnet DLL文件，该节包含Stuxnet DLL文件，这个DLL包含了stuxnet的所有功能。同时，该文件还被用作用户模式rookit，用于隐藏stuxnet文件。具有挂钩API行为。		
标签：	Stuxnet　APT　Dropper　Equation　释放文件　Rootkit　Hook　Shellcode		
情报向量拓展：	关键字：Dropper 其他：.stub		

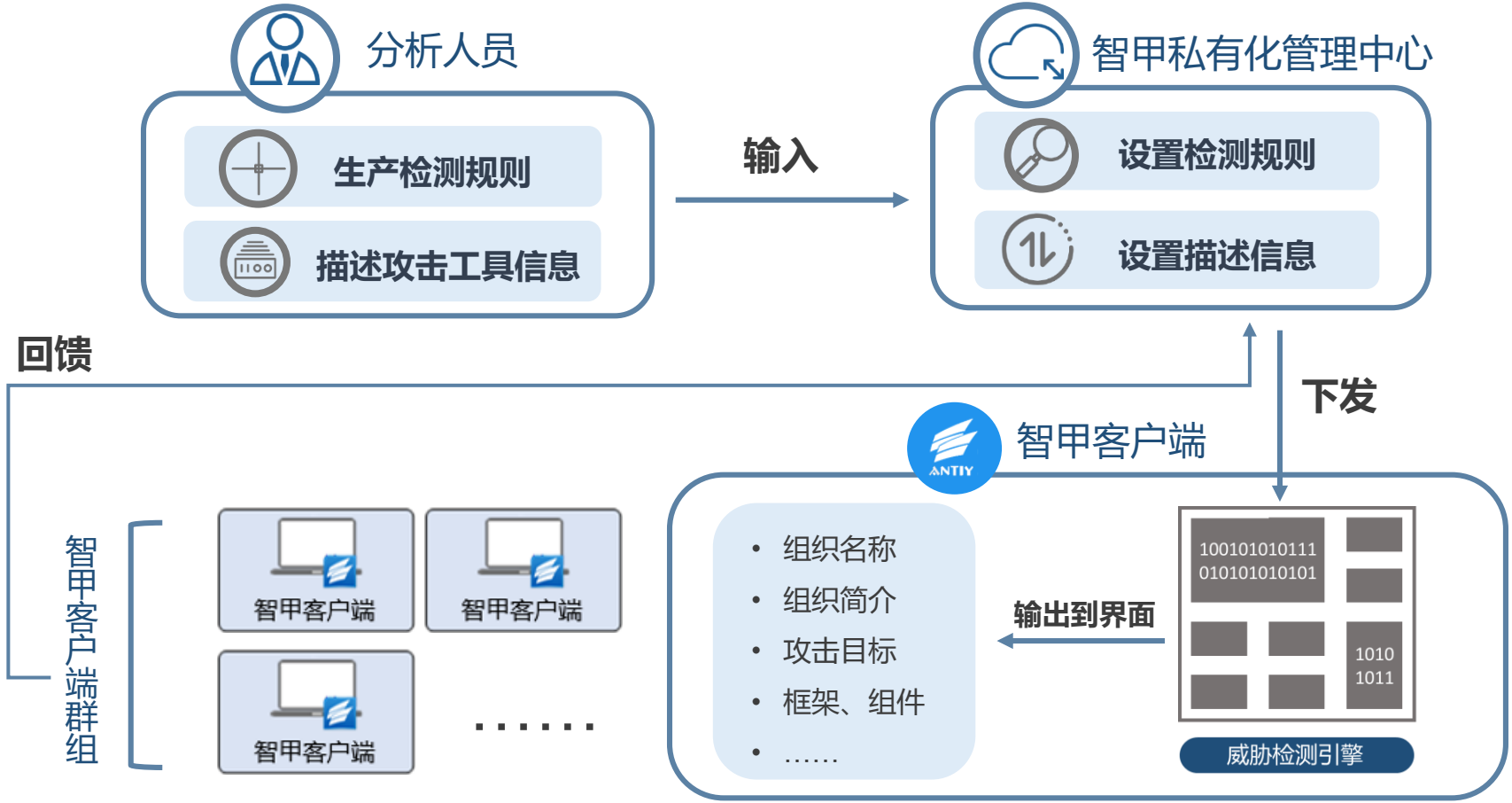
⑤战术技术过程

战术技术过程

MITRE ATT&CK

初始访问	执行	持久化	提权	防御规避	凭证访问	发现	横向运动	收集	命令控制	渗漏	对抗
		T1179 挂钩	T1179 挂钩	T1140 反混淆/解码文件或信息 T1014 Rootkit	T1179 挂钩					T1022 数据加密	

安天引擎承载威胁情报的应用流程



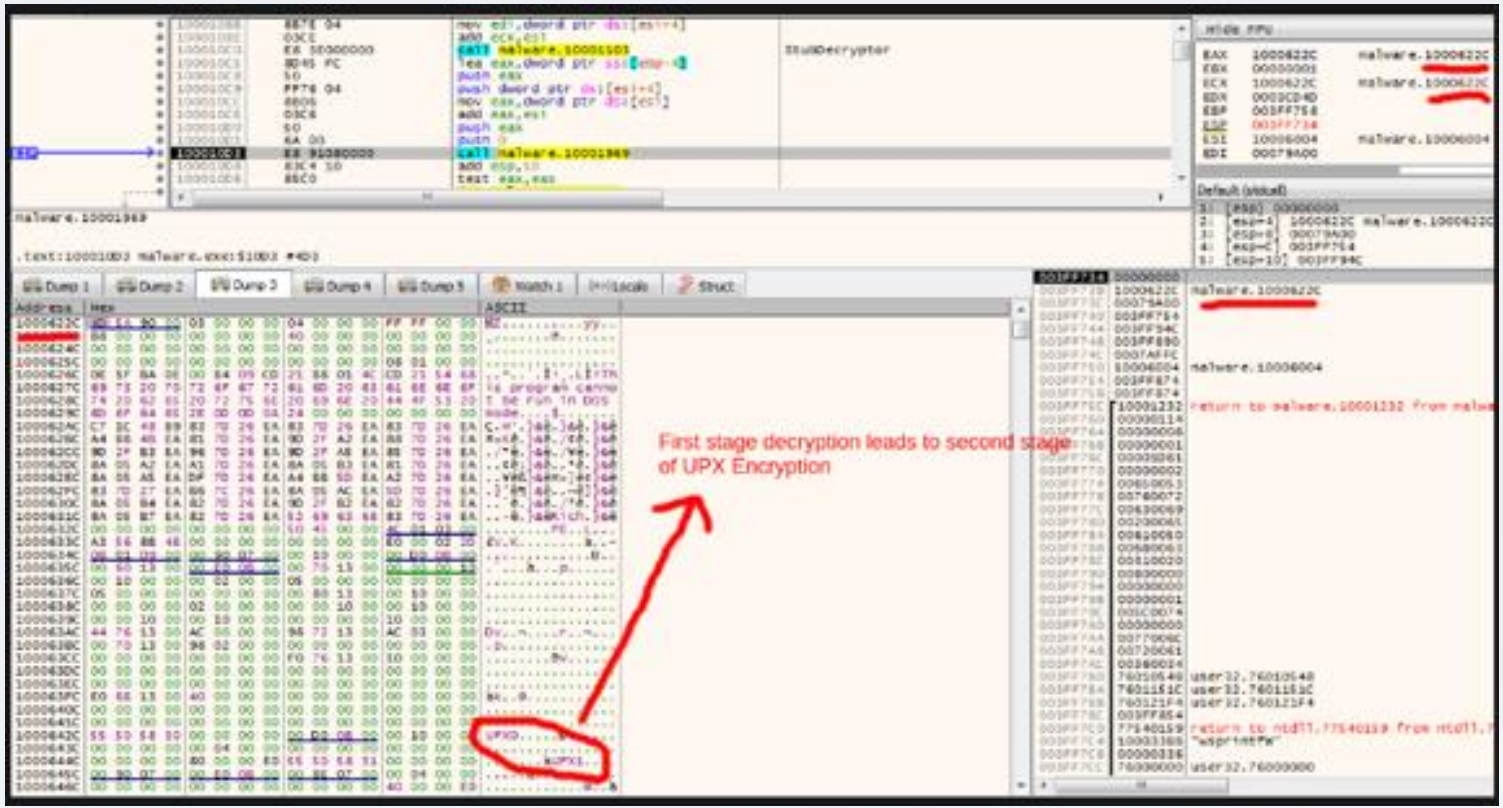
震网本实例2

The History and Analysis of Stuxnet

Lilly Chalupowski
September 13, 2018

这是一篇来源于网络的第三方分析报告，我们将以本报告中的信息为例，展示情报的提取过程。

震网本实例2—了解报告中的信息



某APT报告对震网样本的分析

震网的主DLL文件，在被释放后采用UPX压缩，原因在于防止样本过大。某报告中对该DLL进行了分析。该报告描述了脱壳的过程，以及脱壳后的DLL某些行为。由于引擎开放脱壳接口，所以用户只需要提取脱壳后的行为即可对该DLL文件进行检测。

震网样本实例2--根据报告对样本提取规则

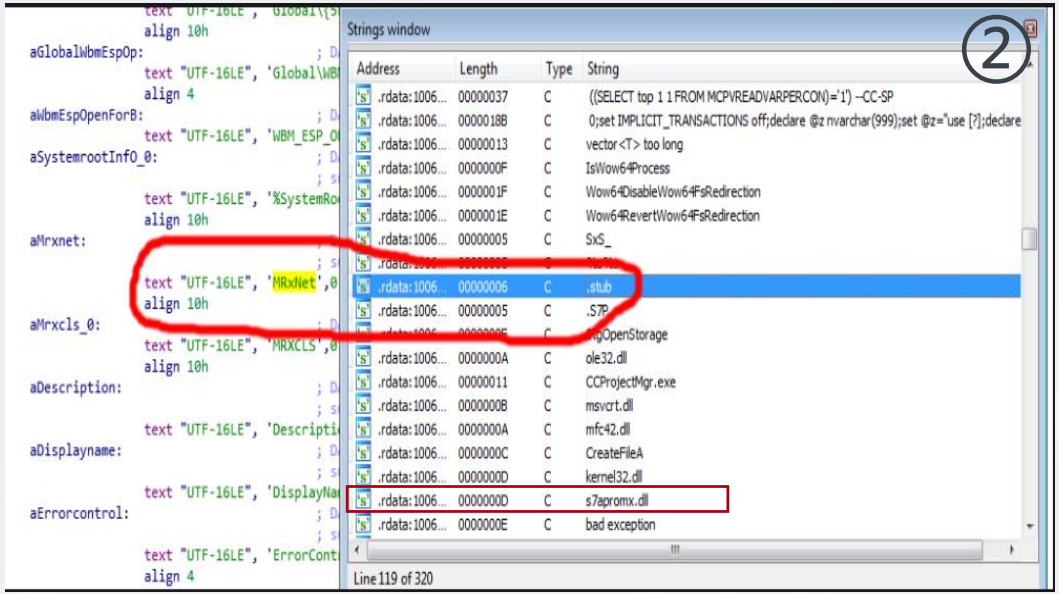
1、脱壳后，报告中介绍了以下可被提取的特征点：

```
034 lea     esi, [ebp-28h]
034 call    sub_100015AD
02C lea     eax, [ebp-28h]
02C push    offset aSystemrootSyst_1 ; %SystemRoot%\\system32\\Drivers\\mrxnet.sys
030 push    eax
034 call    sub_10026709
034 pop     ecx
030 pop     ecx
02C mov     byte ptr [ebp-4], 0Ah
02C push    0FFFFFFFFh
030 push    ebx
034 push    eax
038 lea     eax, [edi+0E4h]
038 call    sub_10001520
02C mov     byte ptr [ebp-4], 1
02C push    ebx
030 push    1
034 lea     esi, [ebp-28h]
034 call    sub_100015AD
02C lea     eax, [ebp-28h]
02C push    offset aSystemrootSyst_0 ; %SystemRoot%\\system32\\Drivers\\mrxsmb.sys
030 push    eax
034 call    sub_10026709
```

①

释放的驱动文件名
(文件名：WTR4132.TMP)

2、结构信息：.stub节，加载的dll文件名s7apromx.dll



②

数据库查询语句
((SELECT top 1 1 FROM MCPVREADVARPERCON)='1') --CC-SP

震网本实例2--根据报告对样本提取规则

1、根据报告对该样本提取规则：

①

```
rule Stuxnet_maindll : Stuxnet
{

    meta:
        description = "Stuxnet Sample - file maindll.decrypted.unpacked.dll_"
    strings:
        $s1 = "%SystemRoot%\\system32\\Drivers\\mrxcls.sys" ascii wide
        $s2 = "%SystemRoot%\\system32\\Drivers\\mrxnet.sys" ascii wide
        $s3 = "~WTR4141.tmp" ascii wide
        $s4 = "s7apromx.dll" ascii wide
        $s5 = ".stub" ascii wide
        $s6 = "((SELECT top 1 1 FROM MCPVREADVARPERCON)='1') --CC-SP" ascii wide

    condition:
        all of them
}
```

2、右面是对该规则的输出情况，可以看到，引擎对1EC9EA8F6F82140443F8BB250427134B（UPX）样本进行脱壳，对释放出的文件进行匹配。

②

```
Scan File: 1EC9EA8F6F82140443F8BB250427134B .......
Scan 1EC9EA8F6F82140443F8BB250427134B start:
Scan 1EC9EA8F6F82140443F8BB250427134B=>upx start:
{
    "ATYaraVector": {
        "YaraRules": {
            "APT": {
            "Abnormal": {
            "Stuxnet": {
                "TypeDes": "The name of the yara rule that was hit",
                "TypeLabel": "",
                "VectorInfo": [{
                    "VecDict": {
                        "VecContent": "Stuxnet_maindll",
                        "YaraTags": "Stuxnet"
                    },
                    "VectorLabel": "",
                    "ThreatLevel": 1,
                    "Vector_des": ""
                }]
            }
        }
    }
}
Scan 1EC9EA8F6F82140443F8BB250427134B=>upx end.
{
Scan 1EC9EA8F6F82140443F8BB250427134B end.
Scan File: 1EC9EA8F6F82140443F8BB250427134B Finish !
```

震网样本实例2—引擎检测结果

1、规则通过产品加入引擎后即可使用，这极大提高了威胁情报信息的可消费性及情报的利用效率。

2、通过该规则对库中样本进行扫描，扫描到其他样本，经核实，这些样本确实属于Stuxnet样本，该规则对于震网组织的检测具有**通用性**，可作为普查规则。

```
Stuxnet_maindll stuxnet_vname//0036D8F64CA4B1DF57ABA22D53112CC3
Stuxnet_maindll stuxnet_vname//ED2FD97DE0A109D88FAA8584A1E6D47A
Stuxnet_maindll stuxnet_vname//C256DE6F0B046CC99B74401FC99867BC
Stuxnet_maindll stuxnet_vname//3C3CA801E767E179BAB304E67CDF6327
Stuxnet_maindll stuxnet_vname//36CD5FFAEDB3E3F0B81D2026D1157CAD
Stuxnet_maindll stuxnet_vname//D740B143D54BE71893C5E7FC54F88085
Stuxnet_maindll stuxnet_vname//32BA631DAD54FAEA08D7627DE69C2D34
Stuxnet_maindll stuxnet_vname//6D561143FDD30EA647B42ED1776EDF2E
Stuxnet_maindll stuxnet_vname//427EC50EFA30B23DD21E839B042A9EF1
Stuxnet_maindll stuxnet_vname//65FF142A9184AFBAFB92BFA5CC8A7B51
Stuxnet_maindll stuxnet_vname//7AB16451D76A21C3EC21FB824FED0BCB
Stuxnet_maindll stuxnet_vname//D40F6E268A258A99341E769C721FF549
Stuxnet_maindll stuxnet_vname//C654AC16BEE0407F5516449F7079E7DE
Stuxnet_maindll stuxnet_vname//59F78BFE7DC168A3ED72FD0E91BCFE3B
Stuxnet_maindll stuxnet_vname//ED912C2B562683A5D2775C72AE2D20C0
Stuxnet_maindll stuxnet_vname//63DC81F7A614FDB3AF5AFC39559C7AF9
Stuxnet_maindll stuxnet_vname//AE597B72599C611C72CCC39A1A1D6FA2
Stuxnet_maindll stuxnet_vname//CB45ABA22B8480A276D9FD42B19DA760
Stuxnet_maindll stuxnet_vname//D216F60B947DC6D4A4014EC2CEC9B265
Stuxnet_maindll stuxnet_vname//503C9A882D1B9312BDD2F82D459FBE47
```

Donot样本实例

- 背景：Donot主要针对巴基斯坦等南亚地区国家进行网络间谍活动，该组织主要针对政府机构等领域进行攻击，其中以窃取敏感信息为主。
- 以下两篇报告均为第三方网站于2018年发布的有关Donot组织的分析报告，我们尝试从这些报告中获取一些情报。

2018年7月发布

2018年12月发布

Donot样本实例—了解报告中的信息

- 仅能从报告中获取IOC信息，得到一些HASH情报。

IOC

MD5
82a5b24fddc40006396f5e1e453dc256
f67595d5176de241538c03be83d8d9a1
e0c0148ca11f988f292f527733e54fca
2320ca79f627232979314c974e602d3a
68e8c2314c2b1c43709269acd7c8726c
35ec92dbd07f1ca38ec2ed4c4893f7ed
88f244356fdaddd5087475968d9ac9bf
14eda0837105510da8beba4430615bce
2565215d2bd8b76b4bff00cd52ca81be
23386af8fd04c25dcc4fdbbeed68f8d4
b47386657563c4be9cec0c2f2c5f2f55

2018年7月份某报告

IOC

C&C
qwe.drivethrough.top
qwe.sessions4life.pw
aoc.sessions4life.pw
mon.sesions4life.pw
tes.sessions4life.pw
5.135.199.0
yty框架的恶意文件MD5
f422bc9c0d0b9d80d09ee1fc7aed3682
3fca54599f30f248246f69290c07696e
e534cf9606a1b9f9a05c6c5514603f77
ff630e55e7278aab1683c7fdc23e9aa9

2018年12月份某报告

Donot样本实例—了解报告中的信息

- 试图从报告中寻找其他信息，获得更具有指向性的情报。

mboard.exe	系统信息插件

mboard.exe

mboard.exe使用UPX加壳，脱壳后根据字符串相关
取系统相关信息，并将获取的信息保存到" %user%
pdf、msg等文件保存到" %user%/LanConfig/ mbo

18年7月份某报告中体现了Go语言配置信息

相关CMD命令如下表

命令	功能
dir /a /s 磁盘名:\	获取磁盘相关文件
systeminfo	获取系统信息
Ipconfig /all	IP相关信息
net view	当前域的计算机列表
tasklist	进程列表

1.总体功能相同：信息收集插件

2.壳编译器相同：UPX壳，Go语言配置信息

3.执行命令相同

相同点

1.文件名和Hash不同

2.信息保存目录不同：

%user%/LanConfig/mboard/mboard（左侧）

%USERPROFILE%\Printers\Neighbourhood\Spools（右侧）

不同点

Systeminfo – spsvc.exe

文件名	spsvc.exe
MD5	2565215d2bd8b76b4bff00cd52ca81be

系统信息搜集插件使用UPX加壳，脱壳后根据字符串相关信息可以知道是go语言编写的程序。该插件会创建多个CMD进程执行命令，获取系统相关信息，并将获取的信息保存到目录%USERPROFILE%\Printers\Neighbourhood\Spools:

创建新进程	cmd /C dir /a /s c:\
创建新进程	cmd /C dir /a /s d:\
创建新进程	cmd /C dir /a /s e:\
创建新进程	cmd /C dir /a /s f:\
创建新进程	cmd /C dir /a /s g:\
创建新进程	cmd /C dir /a /s h:\
创建新进程	cmd /C dir /a /s i:\
创建新进程	cmd /C systeminfo
创建新进程	systeminfo
创建新进程	cmd /C "ipconfig /all"
创建新进程	ipconfig /all
创建新进程	cmd /C "net view"
创建新进程	net view
创建新进程	cmd /C tasklist
创建新进程	tasklist

18年12月份某报告中显示了样本执行的命令

Donot样本实例—根据报告对样本提取规则

- 提取规则

```
rule Donot_System_info : Donot
{
  meta:
    description = "Donot Component,Get Information"
  strings:
    $b0 = "C:/Users/USESR/go/src/github.com/go-ole/go-ole/error.go"
    $b1 = "C:/Users/USESR/go/src/github.com/go-ole/go-ole/com.go"
    $b2 = "C:/Go/src/encoding/binary/varint.go"
    $b3 = "C:/Go/src/os/user/lookup_windows.go"
    $b4 = "C:/Go/src/os/user/lookup.go"
    $b5 = "C:/Go/src/os/exec/lp_windows.go"
    $b6 = "C:/Go/src/os/exec/exec_windows.go"
    $b7 = "C:/Go/src/os/exec/exec.go"
    $b8 = "C:/Go/src/context/context.go"
    $b9 = "C:/Go/src/math/rand/rng.go"
    $b10 = "C:/Go/src/path/filepath/path_windows.go"
    $b11 = "C:/Go/src/path/filepath/match.go"
    $b12 = "C:/Go/src/strings/strings_generic.go"
    $b13 = "C:/Go/src/strings/strings.go"
    $b14 = "C:/Go/src/sort/sort.go"
    $b15 = "C:/Go/src/fmt/scan.go"
    $b16 = "C:/Go/src/fmt/print.go"
    $b17 = "C:/Go/src/reflect/asm_386.s"
    $b18 = "C:/Go/src/reflect/value.go"
    $b19 = "C:/Go/src/reflect/makefunc.go"
    $b20 = "C:/Go/src/os/str.go"
    $b21 = "C:/Go/src/os/stat_windows.go"
    $b22 = "C:/Go/src/os/proc.go"

    $c1 = "ipconfig /all"
    $c2 = "net view"
    $c3 = "systeminfo"
    $c4 = "tasklist"
    $c5 = "LanConfig\\mboard"
    $c6 = "Printers\\Neighbourhood\\Spools"

  condition:
    15 of ($b*) and 5 of ($c*)
}
```

①

Go语言相关字符串

运行命令及保存目录

- 加壳后的Donot样本

IDA View-A | Strings window | Hex View-1

②

Address	Length	Type	String
HEADER:00…	00000005	C	3.91
HEADER:00…	00000008	C	UPX!\r\t\b\n
HEADER:00…	0000001E	C	Go build ID: \"f38fcbeadd3d76b
HEADER:00…	00000013	C	38510a20d6267ab56\"\n
HEADER:00…	00000005	C	;a\bvk
HEADER:00…	00000007	C	\\Sp?D]\b
HEADER:00…	00000006	C	\b\b8 !t
HEADER:00…	00000007	C	\\2P_+DD
HEADER:00…	00000005	C	<_\"\"%
HEADER:00…	00000009	C	[\b4-\b!Q[I
HEADER:00…	00000005	C	\t<\b\t<
HEADER:00…	00000005	C	$@QDO
HEADER:00…	00000006	C	\ty\v,S\x1B
HEADER:00…	00000006	C	!X(\\~k
HEADER:00…	00000005	C	[@9@
HEADER:00…	00000005	C	@\t\bM\v
HEADER:00…	00000006	C	rL&u G
HEADER:00…	00000007	C	juF h\vO
HEADER:00…	00000007	C	|U5]#$\b
HEADER:00…	00000005	C	X9vu6
HEADER:00…	00000005	C	wZ\\\bS
HEADER:00…	00000005	C	8\v08E
HEADER:00…	00000005	C	;w O\r
HEADER:00…	00000006	C	@dn5(\b
HEADER:00…	00000006	C	VcL[q\n
HEADER:00…	00000005	C	fD(U(
HEADER:00…	00000005	C	LlxyM
HEADER:00…	00000005	C	G1IP9
HEADER:00…	00000005	Pas…	)uVu
HEADER:00…	00000005	C	b\a?G;
HEADER:00…	00000006	C	O:4\\<P
HEADER:00…	00000005	C	:\rpZ[
HEADER:00…	00000007	C	<O$<(@
HEADER:00…	00000006	C	3V8\aR3

Donot样本实例—利用引擎使情报落地

- **IOC情报检测结果**

至多可检测n个不同的样本

Hash 1

Hash 2

…

Hash n

- **具有指向性的规则检测结果**

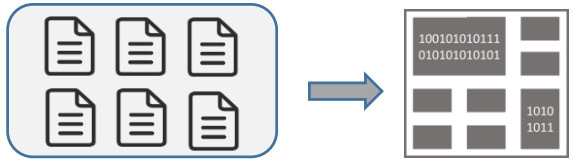
不能检出

规则与未脱壳的样本进行匹配

Donot其它相似带壳样本集合

- **下一代威胁检测引擎+具有指向性的威胁情报检测结果**

Donot其它相似带壳样本集合均可检出

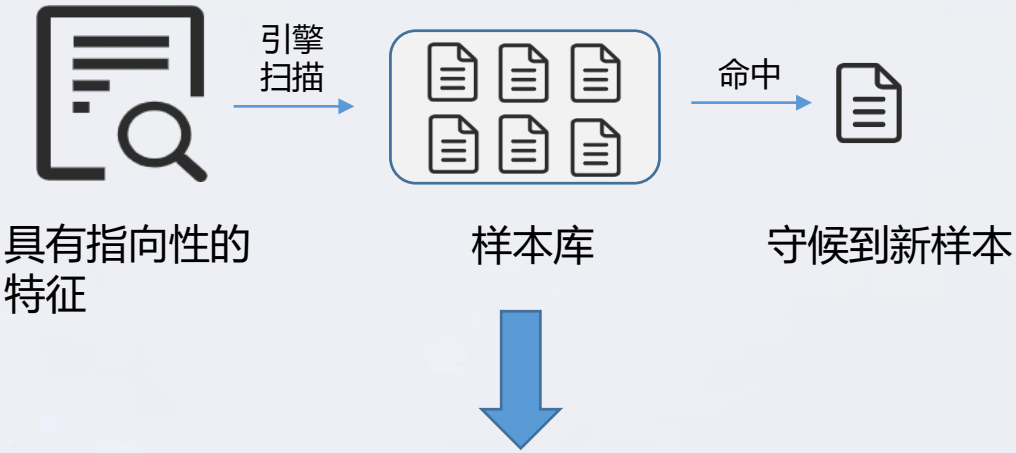


Donot其它相似带壳样本集合

规则与脱壳后的样本进行匹配

Donot样本实例—引擎检测结果

- 守候到的新样本



文件名	md5	发现日期	来源
mboard.exe	1d5e98fc11a1fc4e166010ba78ef907d	2018.07	开源报告
spsvc.exe	2565215d2bd8b76b4bff00cd52ca81be	2018.12	开源报告
3.exe	fee4bd924bd3e55186643839b001bb1a	2019.02	样本库

Donot样本实例—结果验证

- 守候到的样本和报告中的样本从基本信息上看，Hash不同，但都加了UPX壳，大小一致

检测对象	1d5e98fc11a1fc4e166010ba78ef907d			对象来源	none
MD5	1D5E98FC11A1FC4E166010BA78EF907D	格式名字(ID)	BinExecute/Microsoft.PE[:X86](22)	文件大小(Bytes)	552928
编译器/壳识别(ID)	Packer_Compression/Markus.UPX(1003)		自解压包识别(ID)	none	

① 报告中的样本

检测对象	FEE4BD924BD3E55186643839B001BB1A			对象来源	none
MD5	FEE4BD924BD3E55186643839B001BB1A	格式名字(ID)	BinExecute/Microsoft.PE[:X86](22)	文件大小(Bytes)	552928
编译器/壳识别(ID)	Packer_Compression/Markus.UPX(1003)		自解压包识别(ID)	none	

② 新守候到的样本

- 由于加了UPX壳，很难进一步判断两个文件之间的关系，于是对这两个文件进行脱壳

检测对象	1d5e98fc11a1fc4e166010ba78ef907d=>upx			对象来源	APUnPack_UPX_Compress
MD5	117E45E4D8E1817BA1AC302668B8F8D5	格式名字(ID)	BinExecute/Microsoft.PE[:X86](22)	文件大小(Bytes)	1492992
编译器/壳识别(ID)	Compiler/Google.GoLang[:ALL](406)		自解压包识别(ID)	none	

报告中的样本脱壳后

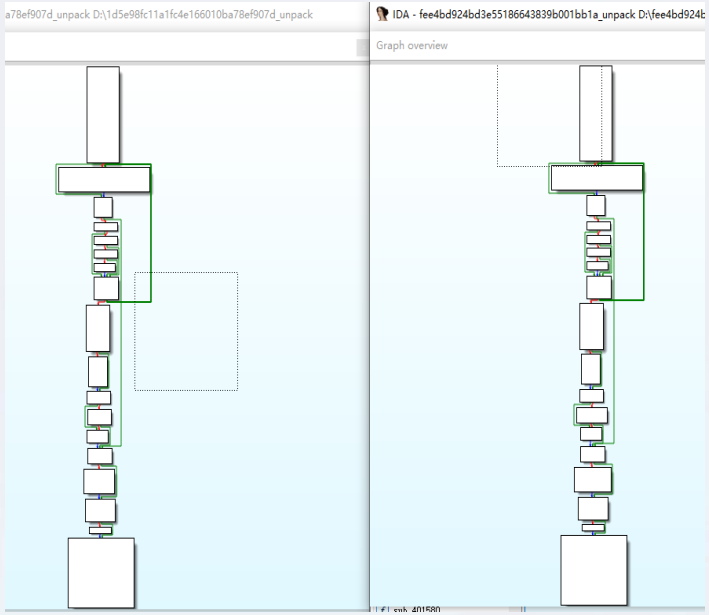
检测对象	FEE4BD924BD3E55186643839B001BB1A=>upx			对象来源	APUnPack_UPX_Compress
MD5	B02AC1F8671D48D399ACA168E721A8C7	格式名字(ID)	BinExecute/Microsoft.PE[:X86](22)	文件大小(Bytes)	1493504
编译器/壳识别(ID)	Compiler/Google.GoLang[:ALL](406)		自解压包识别(ID)	none	

新守候到的样本脱壳后

Donot样本实例—结果验证

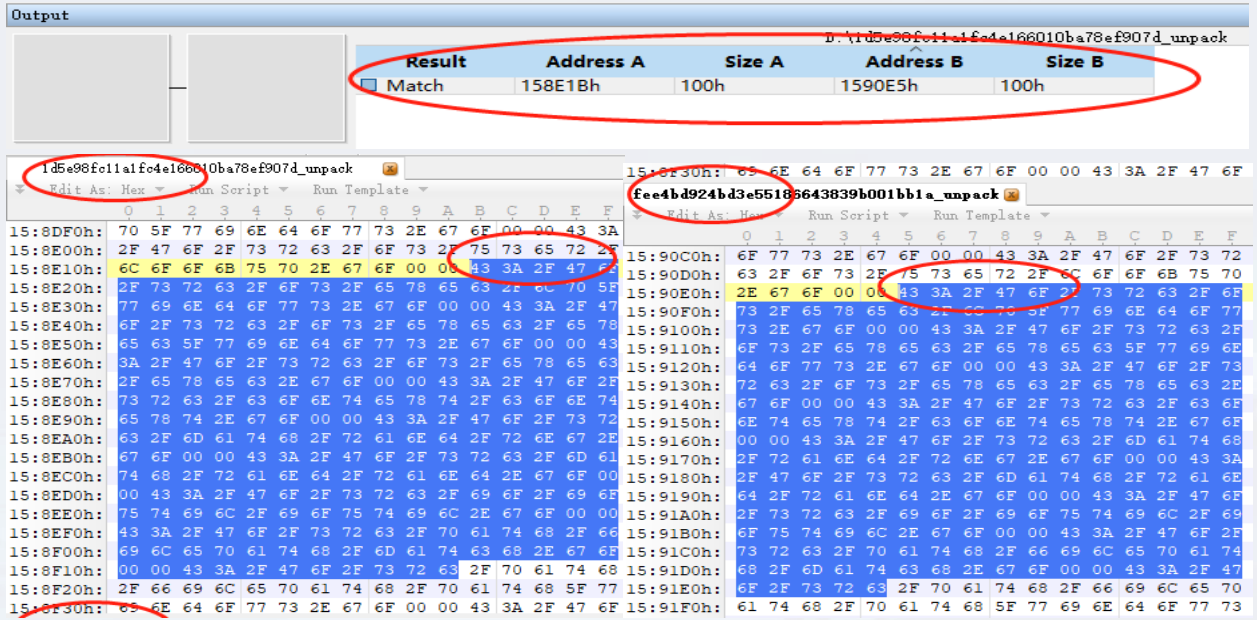
- 对脱壳样本进行继续分析，发现以下几处相似点：

1. 代码流程



脱壳后的样本代码流程相似

2. 两个样本都包含报告中提到的关键字符串



报告中的样本

新守候到的样本

Donot样本实例—结果验证

1. 相同的内嵌脚本

报告中的样本

1d5e98fc11a1fc4e166010ba78ef907d_unpack

Edit As: Hex Run Script Run Template

```
E:7060h: 73 69 67 6E 61 6C 20 09 73 74 61 63 6B 3D 5B 5F  signal .stack=[_
E:7070h: 4E 65 77 45 6E 75 6D 63 67 6F 63 68 65 63 6B 63  NewEnumcgocheckc
E:7080h: 73 20 20 20 20 20 20 65 61 78 20 20 20 20 20 65  s      eax     e
E:7090h: 62 70 20 20 20 20 20 65 62 78 20 20 20 20 20 65  bp     ebx     e
E:70A0h: 63 78 20 20 20 20 20 65 64 69 20 20 20 20 20 65  cx     edi     e
E:70B0h: 64 78 20 20 20 20 20 65 66 6C 61 67 73 20 20 65  dx     eflags  e
E:70C0h: 69 70 20 20 20 20 20 65 73 69 20 20 20 20 20 65  ip     esi     e
E:70D0h: 73 70 20 20 20 20 20 66 73 20 20 20 20 20 20 67  sp     fs      g
E:70E0h: 73 20 20 20 20 20 20 6E 65 74 20 76 69 65 77 6E  s      net viewn
E:70F0h: 6F 20 61 6E 6F 64 65 72 75 6E 6E 61 62 6C 65 72  o anoderunnabler
E:7100h: 75 6E 74 69 6D 65 2E 73 63 61 76 65 6E 67 65 73  untime.scavenges
E:7110h: 68 75 74 64 6F 77 6E 73 74 72 63 6F 6E 76 2E 74  hutdownstrconv.t
E:7120h: 61 73 6B 6C 69 73 74 75 6E 6B 6E 6F 77 6E 28 77  asklistunknown(w
E:7130h: 73 61 69 6F 63 74 6C 20 28 66 6F 72 63 65 64 29  saioctl (forced)
E:7140h: 20 2D 3E 20 6E 6F 64 65 3D 20 62 6C 6F 63 6B 65  -> node= blocke
E:7150h: 64 3D 20 64 65 66 65 72 73 63 3D 20 69 6E 20 75  d= defersc= in u
E:7160h: 73 65 29 0A 20 6C 6F 63 6B 65 64 67 3D 20 6C 6F  se). lockedg= lo
E:7170h: 63 6B 65 64 6D 3D 20 6D 2D 3E 63 75 72 67 3D 20  ckedm= m->curg=
E:7180h: 6D 73 20 63 70 75 2C 20 20 6E 6F 74 20 69 6E 20  ms cpu,  not in
E:7190h: 5B 20 6F 66 20 73 69 7A 65 20 20 72 75 6E 74 69  [ of size  runti
```

新守候到的样本

fee4bd924bd3e55186643839b001bb1a_unpack

Edit As: Hex Run Script Run Template

```
E:71F0h: 6E 61 6C 20 09 73 74 61 63 6B 3D 5B 5F 4E 65 77  nal .stack=[_New
E:7200h: 45 6E 75 6D 63 67 6F 63 68 65 63 6B 63 73 20 20  Enumcgocheckcs
E:7210h: 20 20 20 20 65 61 78 20 20 20 20 20 65 62 70 20      eax     ebp
E:7220h: 20 20 20 20 65 62 78 20 20 20 20 20 65 63 78 20      ebx     ecx
E:7230h: 20 20 20 20 65 64 69 20 20 20 20 20 65 64 78 20      edi     edx
E:7240h: 20 20 20 20 65 66 6C 61 67 73 20 20 65 69 70 20      eflags  eip
E:7250h: 20 20 20 20 65 73 69 20 20 20 20 20 65 73 70 20      esi     esp
E:7260h: 20 20 20 20 66 73 20 20 20 20 20 20 67 73 20 20      fs      gs
E:7270h: 20 20 20 20 6E 65 74 20 76 69 65 77 6E 6F 20 61      net viewno a
E:7280h: 6E 6F 64 65 72 75 6E 6E 61 62 6C 65 72 75 6E 74  noderunnablerunt
E:7290h: 69 6D 65 2E 73 63 61 76 65 6E 67 65 73 68 75 74  ime.scavengeshut
E:72A0h: 64 6F 77 6E 73 74 72 63 6F 6E 76 2E 74 61 73 6B  downstrconv.task
E:72B0h: 6C 69 73 74 75 6E 6B 6E 6F 77 6E 28 77 73 61 69  listunknown(wsai
E:72C0h: 6F 63 74 6C 20 28 66 6F 72 63 65 64 29 20 2D 3E  octl (forced) ->
E:72D0h: 20 6E 6F 64 65 3D 20 62 6C 6F 63 6B 65 64 3D 20   node= blocked=
E:72E0h: 64 65 66 65 72 73 63 3D 20 69 6E 20 75 73 65 29  defersc= in use)
```

报告中的样本 →

Edit As: Hex Run Script Run Template

```
E:8130h: 62 79 6E 61 6D 65 69 6E 76 61 6C 69 64 20 62 61  bynameinvalid ba
E:8140h: 73 65 20 69 70 63 6F 6E 66 69 67 20 2F 61 6C 6C  se ipconfig /all
```

新守候到的样本 →

Edit As: Hex Run Script Run Template

```
E:8130h: 62 79 6E 61 6D 65 69 6E 76 61 6C 69 64 20 62 61  bynameinvalid ba
E:8140h: 73 65 20 69 70 63 6F 6E 66 69 67 20 2F 61 6C 6C  se ipconfig /all
```

2. 具有相同的命令信息

综合性价值

提升威胁情报的泛化性及精准性，面向高级威胁，脱离狭义的威胁情报，基于对重点威胁对象和其高级恶意代码使用特点，形成高度精准的具有泛化性的威胁情报。

构建面向高级威胁的检测能力的闭环，同时利用知识化的输出提升用户对于高级威胁的感知能力及理解能力，使用户全面了解攻击对手。

面向高级威胁的对抗，提高高级网空威胁行为体的攻击成本。以强大的防御实现减少对手的可攻击面、限制对手的行动能力、提升对手的攻击成本的目的。

谢谢大家