

威胁情报的层次分析

汪列军 / 奇虎360科技有限公司

威胁情报，作为近几年来兴起的安全热点，已经从理念到技术再到平台逐步开始落地。不管是老牌的安全公司还是新兴的厂商都正在这方面探索，包括可行的技术方案、交换标准及商业模式。本文在探讨威胁情报的作用的基础上，着重为读者介绍一下威胁情报的层次，并结合自身的实践，提出一个金字塔模型，希望有助于提升业界对威胁情报的认知。

一、威胁情报的作用

目前，Gartner对于威胁情报的定义比较广泛地被引用，这是一个比较理想化的定义，对情报应该包含的信息量提出了明确的要求，面向高端用户提供决策依据的完整情报样式，可以认为是狭义的威胁情报。

事实上，大多数的组织机构得不到那样准确和全面的情报服务，即便得到也无法采取应对措施。想象一下，即使有安全厂商能够告诉某个大公司一个安全威胁的背后组织、国家背景甚至人员信息（这些都是高端威胁情报的必要组成部分），那又能如何？通常的组织机构不是执

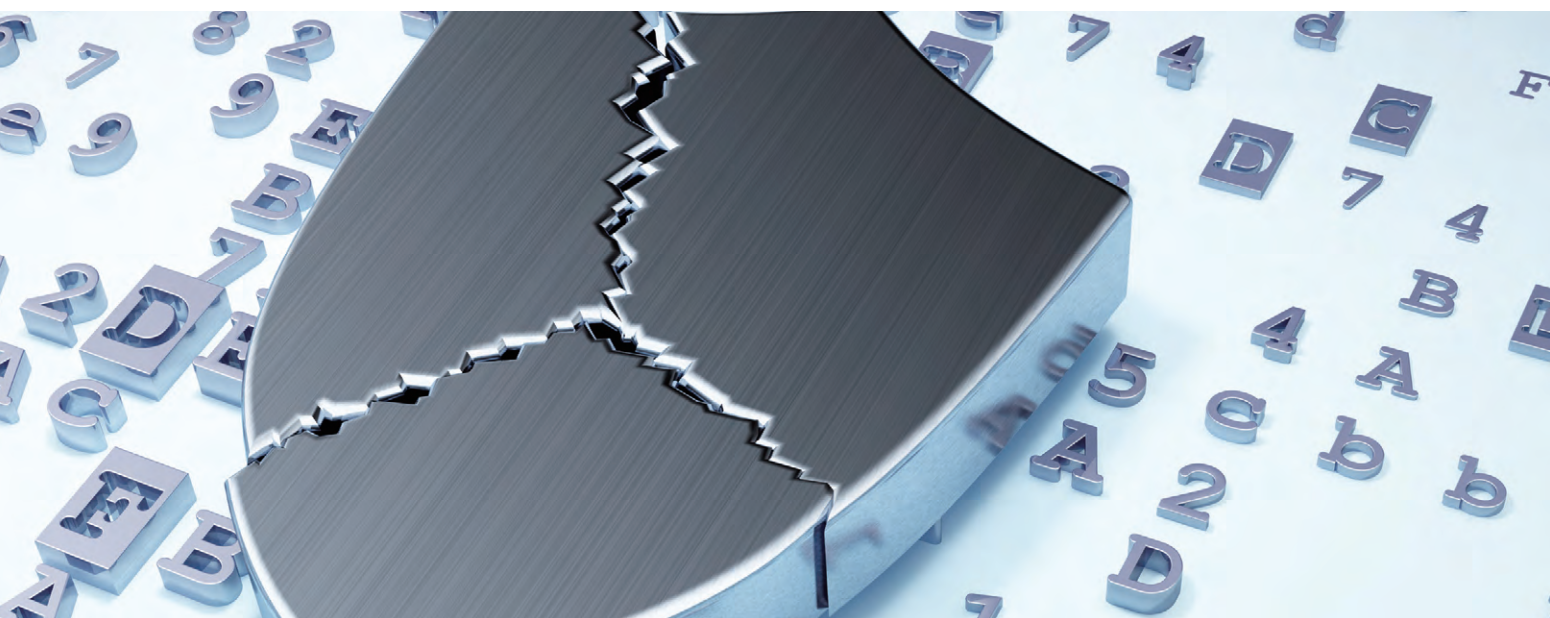
法机构，无法对这些情报采取什么缓解威胁的措施。

对一般的公司，部署相对低端的入侵指示器（Indicator Of Compromise, IOC）可能更为实际些，它由可被边界安全设备和主机安全防护软件所使用的数据所构成，典型的入侵指示器有文件HASH、IP、域名、程序运行路径、注册表项等，这类威胁情报将在下文进行详细分析。

通过对威胁情报的交换和共享，联合安全业界各方的力量，整合信息资源实现更大范围内的快速响应，以对抗也在不停进化的安全威胁，笔者认为这就是威胁情报的作用。图1Webroot小纪



图1 Webroot小纪念品



念品上的图非常形象地表现了这点。

当用户试图分析一个可疑事件时，威胁情报可以为用户判定可疑事件的恶意性提供有用的参考资料，比如事件所涉及的IP是否在某些已知的黑名单之中，相关的域名是否被已知的APT活动使用等。准确及时的入侵指示数据可以帮助用户快速处理已经或正在发生的威胁，比如黑样本的HASH、对外连接的C&C及Downloader服务器的IP或域名，网络边界设备或运行于主机上的Agent可以通过简单的匹配就能发现并采用自动化的应对措施。

二、威胁情报的层次

图2是我们构建的一个展示威胁情报层次的金字塔图，下面从下到上逐层解释每个层次的信息构成，所能发挥的作用及分析获取的方法。

(一) 文件HASH

最底层威胁情报由文件构成，主要涉及恶意网络活动相关的各种恶意代码：Trojan、Backdoor、Downloader、Dropper等。一般来

说，文件样本是整个事件分析的起点和基础性数据，其重要性相当于刑事案件调查中最主要的物证，比如凶器。用于标记文件的各种HASH是最基本的威胁情报信息，可以方便地用于在目标系统上进行搜索，如果一个木马文件在系统上被发现则对象被感染的可能性就非常大。表1是Symantec发布的Butterfly攻击活动相关的部分文件HASH列表。

绝大多数文件HASH的主要问题在于特异性太强，无论是MD5、SHA1、SHA256，只要文件出现一个比特位的变化就会导致完全不同的

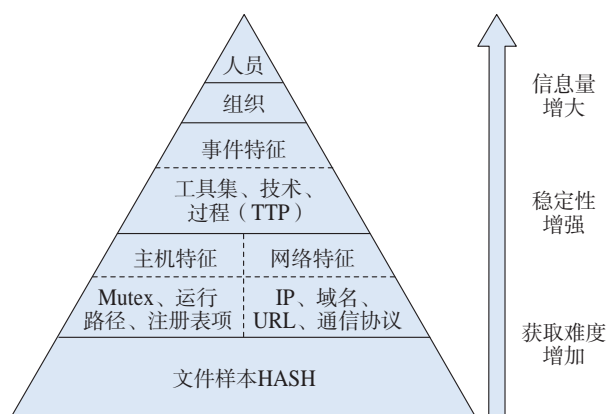


图2 威胁情报层次金字塔

表1 Butterfly攻击活动相关的部分文件HASH列表

Table 2. File hashes of tools used by the Butterfly attackers, including filenames. (List includes clean files)		
SHA-256	File name	Description
2a8cb295f85f8d1d5aae7744899875ebb4e6c3ef74fbc5bfad6e7723c192c5cf	winsession.dll	Hacktool
da41d27070488316cbf9776e9468fae34f2e14651280e3ec1fb8524fda0873de	bj.dat	Hacktool.Bannerjack
796b1523573c889833f154aeb59532d2a9784e4747b25681a97ec00b9bb4fb19	bj.dat	Hacktool.Bannerjack
c54f31f190b06649df91f6b915273b88ee27a2f8e766d54ee4213671fc09f90	pc.dat	Hacktool.Multipurpose
54a8afb10a0569785d4a530ff25b07320881c139e813e58cb5a621da85f8a9f5	pc.dat	Hacktool.Multipurpose
2bd5f7e0382956a7c135cdeb96edfdbcfcfc1955d26e317e2328ea83ace7cee	pc.dat	Hacktool.Multipurpose
c83bb0330d69f6ad4c79d4a0ce1891e6f34091aecfeaf72cf80b2532268a0abc	pc.dat	Hacktool.Multipurpose
178b25ddca2bd5ea1b8c3432291d4d0b5b725e16961f5e4596fb9267a700fa2f	PC.DAT	Hacktool.Multipurpose
9bfff19ca48b43b148ff95e054efc39882d868527cdd4f036389a6f11750adddc	PC.DAT	Hacktool.Multipurpose
e8591c1caa53dee10e1ef748386516c16ab2ae37d9555308284690ea38df0c5	clapi32.dll	Clean Cygwin DLL
d15b8071994bad01226a06f2802cbfe86a5483803244de4e99b91f130535d972	Bda9.tmp.	Backdoor.Jiripbot
0ac7b594aaae21b61af2f3aabdc5eda9b6811eca52dcfb4691c4ec6dfd2d5cd8	wlc.dat	Hacktool.EventLog

HASH值，这个特性在避免误报的同时也使攻击者可以通过最简单的内容修改来躲过检测，所以一旦被公开揭露，几乎立即过期。因此，文件HASH作为入侵指示器基本只能用来发现已发生的事件，对防御方来说需要用自动化的搜索匹配机制，尽可能用其来缩短从事件发生到发现的时间窗口，以最大程度减少损失。

（二）主机和网络特征

在文件HASH之上的是通过分析文件样本得到的直接关联的各类基于主机和网络的特征，这些数据可以被用来作为入侵指示器。简单来说，主机特征可能包含恶意代码在机器上运行时产生的有区分能力的特征，比如程序运行时的Mutex、写入的注册表项、文件路径等，网络特征可能包含对外连接的C&C或组件下载的IP/域名、访问的URL、通信协议等信息。图3是一个木马内置的主机与网络特征数据的例子。

这些数据大多可以通过使样本运行于受控环境（沙箱和虚拟机）来获取，对于对抗强度较高的或无法运行的样本手工的逆向调试分析则有可能是必需的，这时就需要很大的时间与精力投入。相较于文件HASH，这些从文件中通过静态或动态分析得到的特征相对稳定，但改变的代价依然很小，特别是在被公开揭露出来以后，作为入侵指示器的价值也会很快消失。

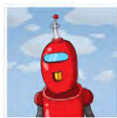
Details	
	Malware Family nRut Date Added April 15, 2016, 3 a.m. MD5 860090644940c8973aa10a96f104076 Sha256 6493a1d2e0101b1bf79e6603fb55acd28801044854baf266781d595c1b6001 Robot Robots lovingly delivered by roboshash.org
Config Sections	
Registry Value	b4c12bec3027d94da0c81d02d1960f9
Domain	factor2016-no-ip.org
Install Dir	TEMP
Campaign ID	HackEd
Port	1177
Version	0.6.4
Install Name	svchost.exe

图3 木马内置的主机与网络特征数据

（三）事件层次情报

在单个样本相关的信息以上为事件层次的威胁情报。当我们得到了大量文件样本相关的细节以后，通过分析其各个维度上的相似度就可以实现样本家族的分类。图4是三个疑似欧洲来源的秘密监控软件基于样本特征的同源性分析，可以看到一些关键特征保持一致，暗示它们有共同的源头。

通过分析样本之间的上下游关系，可以推断攻击发生时恶意代码的进入渠道，从而了解对手的攻击手法，如，是通过鱼叉邮件、水坑攻击、U盘植入还是其他主动性的攻击，是否利用了安全漏洞，使用了什么样的社工技巧，等等。了解攻击的方法对于防御方调整防护方案，填补漏洞和盲点，

	A	B	C	D	E
1			NBOT/TFC	Bunny	Babar
2	String constants				
3	Error / status messages	No	Many	Many	Many
4	String formatting style	All plain, commands/config all caps, no special charact	Partially plain, config encrypted, config all caps in XML	All plain, config all caps, enclosed in %N characters	All plain, config all caps, enclosed in %N characters
5	English grammar mistakes	No	Many	Many	Many
6	C&C commands	PING, EXEC, HTTPF, ASPFLOOD, TCPFLOOD, WEBFLOOD, POSTFL	mainfrequency, getconfig, fipput, fipget, sendfile, getfile, u	N/A	N/A
7	Timestamp formatting	Time APIs _time64, _mktime64, %02d:%02d:%02d	Time/Time API GetSystemTime(), 'timestamp %04d-%02d-%02c	N/A	N/A
8	Implementation traits				
9	Memory allocation habits	direct calls to _malloc/_free, no wrappers	GetProcessHeap()/HeapAlloc()/HeapFree() in large num	direct calls to _malloc/_free, no wrappers	direct calls to _malloc/_free, no wrappers
10	Use of global variables	Few	Few, storing of event handles, strings, global flags use	Few, storing of event handles, strings	Few, storing of event handles, strings
11	Multi-threading model	Simple, main thread with several worker threads	Simple, main thread with several worker threads	Complex, multi-threading in various instances coordina	Complex, multi-threading in various instances coordina
12	Software architecture and design	Standalone executable, classical bot structure	Standalone executable, classical bot structure, integrat	DLL, designed to run in context of arbitrary process, mai	DLL, designed to run in context of arbitrary process, mai
13	Constructor design	MSVC++ default	MSVC++ default	MSVC++ default with complex object dependencies	MSVC++ default with complex object dependencies
14	Dynamic API loading technique	Present, subset of APIs only, per API, API name identifi	Present, subset of APIs only, per API, API name identifi	Present, subset of APIs only, per API, API name identifi	Present, subset of APIs only, per API, API name identifi
15	Exception handling	C++ EH and unhandled exception filter: ExitThread()	C++ EH and unhandled exception filter: ExitThread() (dyr	C++ EH default	C++ EH default
16	Usage of public source code	None (known)	Lua engine, C/invoke bindings	Keylogger from codeproject.com, OpencoreAMR library, f	Keylogger from codeproject.com, OpencoreAMR library, f
17	Programming language and compiler	C++ / MSVC++ 8.0	C++ / MSVC++ 8.0	C++ / MSVC++ 8.0	C++ / MSVC++ 8.0
18	Compilation time stamps and time zones	2010-03-11 17:55:03+01:00 2010-02-18 05:54+01:00 2010-05-06 15:47:37+02:00	2011-10-25 21:28:39+02:00 2011-10-25 21:28:00+02:00	2011-08-29 15:02:29+02:00 2011-08-29 15:48:43+02:00 2011-07-06 15:50:11+02:00	2011-08-29 15:02:29+02:00 2011-08-29 15:48:43+02:00 2011-07-06 15:50:11+02:00
19	Custom features				
20	Obfuscation techniques	Obfuscation of subset of API names that are to be load	Obfuscation of subset of API names that are to be load	Obfuscation of subset of API names that are to be load	Obfuscation of subset of API names that are to be load
21	Stealth and evasion techniques	Obfuscation of subset of APIs	Emulator check, Containing directory name check, Payloa	Obfuscation of subset of APIs, 'infection_strategy' based	Obfuscation of subset of APIs, 'infection_strategy' based
22	Use of encryption and compression algorithms	API name obfuscation custom algorithm	API name obfuscation custom algorithm	API name obfuscation custom algorithm, adaption of Sh	API name obfuscation custom algorithm, adaption of Sh
23	(Shared) encryption keys	XOR key AB34CD77h	XOR key AB34CD77h, keys for command/data en-/decrypt 128bit	AES, 24 FE CS AD 34 56 F7 F8 12 01 00 AE B6 7C DE A	AES, 24 FE CS AD 34 56 F7 F8 12 01 00 AE B6 7C DE A
24	Re-used source code in general	Timestamp generation, API name hashing and loading	API name hashing and loading, infection strategy and A	infection strategy and A	infection strategy and A
25	DDoS bot for flooding of network packets	DDoS bot for flooding of network packets	Lua scripted bot for automation of tasks	Espionage malware and userland rootkit	Espionage malware and userland rootkit
26	System infiltration	Designed to be loaded as service, running in context of	Loaded by a registry key on startup, running in context o	Loaded through registry key which invokes regsvr32.exe	Loaded through registry key which invokes regsvr32.exe
27	Propagation mechanisms	N/A	N/A	N/A	N/A
28	Artifact naming schemes / algorithms	Config value naming prefix NBOT_	Internal name bunny 2.3.2, payload name netmgr.exe, re	internal name Babar64, payload dump21cb.dll, director	internal name Babar64, payload dump21cb.dll, director
29	Data exfiltration techniques	Statistics file sent to C&C on demand	Log/dumpfile regularly pushed to C&C	Dumpfiles regularly pushed to C&C (assumption)	Dumpfiles regularly pushed to C&C (assumption)
30	System / OS version determination technique	N/A	N/A	N/A	N/A
31	Malware specific features	Encrypted command received from C&C, handed over as	Encrypted command received from C&C, command parsir	N/A	N/A
32	C&C command parsing implementation	hardcoded / plaintext	hardcoded / encrypted	hardcoded / encrypted	hardcoded / encrypted
33	Infrastructure				
34	C&C servers	http://callienteferver.info/, http://fullapple.net/	http://le-progres.net/, http://ghatreh.com/, http://ustb	http://www.horizons-tourisme.com/, http://www.gezel	http://www.horizons-tourisme.com/, http://www.gezel
35	Countries / languages used for domain hosting	US/French, US/Iranian, US/Algerian	US/French, US/Iranian, US/Algerian	US/Algerian, US/Turkish	US/Algerian, US/Turkish
36	User agent / beaconing style	User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows	User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Win32)	User-Agent: Mozilla/4.0 (compatible; MSi 6.0; Windows	User-Agent: Mozilla/4.0 (compatible; MSi 6.0; Windows
37	Communication protocol / port	HTTP/80	HTTP/80	HTTP/80	HTTP/80
38	Communication intervals	On demand	Regular, interval configurable	Regular (assumption)	Regular (assumption)

图4 秘密监控软件基于样本特征的同源性分析

使防护更有针对性同时降低成本有重要的意义。

图5是2015年360公司揭露的“海莲花”APT事件中攻击者所使用的恶意代码感染方式。

如果收集到的数据包包含受害者内部网络和主机的行为，我们还能了解攻击者在受害者内部系统中尝试进一步获取控制的方式和方法。因此，要得到准确的事件层次的威胁情报，需要更多的数据输入，更多的分析资源投入，所以其得到的结果也包含了更大信息量。

表2是我们整理的基于Lockheed Martin Kill Chain模型各环节中可用于做关联性分析的

维度。

(四) 组织情报

基于事件层次的事实收集与分析，我们可以分辨出多个攻击事件背后的同一个组织，并判定组织的来源、分工、资源状况、人员构成、行动目标等要素。

通过分析对手所使用工具开发维护状况、突破技术及通信基础设施，可以推断对手的资源状况。一般来说，使用自己开发的成系列的漏洞利用及控制工具，通信网络使用了较多的IP、域名



图5 “海莲花”事件恶意代码感染方式

表2 基于Lockheed Martin Kill Chain模型各环节中可用于做关联性分析的维度

侦察跟踪	武器构建	载荷投递	突防利用	安装植入	通信控制	达成目标
目标国家	特定互斥量	恶意代码进入方式	漏洞利用	初始启动路径	域名注册信息	目标数据
目标个体	执行流程	鱼叉邮件	社会工程学	持续启动方式	域名使用偏好	打包方法
涉及行业	加解密方式	水坑攻击		伪装正常模式	域名命名偏好	传输方法
	特定功能模块	U盘			IP所在ASN	破坏功能
	对抗分析措施	主动渗透			后门工具	
	源码工程路径				工具类型	
	特定数据字符串				工具配置	
	语言编译环境				通信协议	
	特定数字签名				认证凭据	
	组件组织架构				SSL证书	
	特别的错误					

及服务器资源，载荷投递时显示出专业的针对性技巧，则可以判断对手拥有较强的能力，有足够的资源支持，组织内部可能存在明确的分工。

组织的来源可以通过分析事件涉及的样本文件中包含的语言相关的特征来推断，比如字符串的语言、开发或打包工具的语言版本，对于非可执行的样本，也可以分析其默认的配置情况。在占有大量样本的情况下，可以通过分析样本的生成时间推断对手的日常工作时间，甚至休假情况，与特定国家的节假日做匹配，也可能成为分析对手来源的有效线索。如果有资源了解受影响目标的国家、行业及个体的分布，以及对手所发动的攻击类型（窃密型还是求财型）所使用的基础设施的地理位置，我们也可以非常有把握地推测出对手的来源。

图6是Cylance一个名为Operation Cleaver的APT活动的来源要点分析。

（五）人员情报

在组织之上的是人员相关的威胁情报，这是威胁分析的最后一环，实现虚拟身份到现实身份

Iranian Actors Are Behind Operation Cleaver

- Persian hacker names are used throughout the campaign including: Salman Ghazikhani, Bahman Mohebbi, Kaj, Parviz, Allreza, and numerous others.
- Numerous domains used in the campaign were registered in Iran.
- Infrastructure leveraged in the attack was registered in Iran to the corporate entity Tarh Andishan, which translates to "invention" or "innovation" in Farsi.
- Source netblocks and ASNs are registered to Iran.
- Hacker tools warn when their external IP address traces back to Iran.
- The infrastructure is hosted through Netafraz.com, an Iranian provider out of Isfahan, Iran.
- The infrastructure utilized in the campaign is too significant to be a lone individual or a small group. We believe this work was sponsored by Iran.

图6 Operation Cleaver来源要点分析

的映射。

人处在威胁情报金字塔的顶端，因为它是整个威胁体系中最稳定的部分，一旦定位到人也就定位到了威胁产生的根源，解决人的问题是真正釜底抽薪的解决方案。弗诺·文奇写过一篇非常精彩的小说，名字就叫《真名实姓》，未来虚拟空间里最大的问题就是如何把其中的角色对应到现实中的人，一旦完成映射就意味着战斗的结束。这是因为如果我们处理的对象是人，就意味着解决问题的手段不会再受限于技术可能性。所有处于人之下的威胁情报类型我们一般只能采取技术手段来处理，比如我们知道一些攻击相关的IOC（样本、IP或域名），对其有效的处置手段主要局限于人工或自动化的识别、隔离及阻断



图7 交互式数据关联系统对XcodeGhost事件的始作俑者的追溯演示

等。而当我们对抗对象是人时，那么可采用的手段就可以丰富得多，我们不仅可以对目标本身，还可以对其所在环境施加压力，利用人性的弱点就能实现类似降维攻击的效果。

要得到这类情报，信息的输入也就不再限于技术分析，可能需要其他方面的数据输入及非常规的取证手段，比如真实注册信息、社交账号的关联数据、交易数据、蜜罐及反制。图7是360公司使用交互式数据关联系统对XcodeGhost事件的始作俑者的追溯演示，从攻击者所使用的一个C&C域名（做了隐私保护）出发通过层层前推最终定位到一个关联域名，该域名是攻击者用其真实名字注册的。END

参考链接：

- [1] Malwareconfig[EB/OL].<https://malwareconfig.com/>.
- [2] Butterfly: Corporate spies out for financial gain[EB/OL].
<https://www.symantec.com/content/en/us/enterprise/>

media/security_response/whitepapers/butterfly-corporate-spies-out-for-financial-gain.pdf.

- [3] BIG GAME HUNTING: THE PECULIARITIES OF NATION-STATE MALWARE RESEARCH[EB/OL].
<https://www.blackhat.com/docs/us-15/materials/us-15-MarquisBoire-Big-Game-Hunting-The-Peculiarities-Of-Nation-State-Malware-Research.pdf>.
- [4] OceanLotus(APT-C-00)数字海洋的游猎者[EB/OL].
<https://ti.360.com/upload/report/file/OceanLotus-Report.pdf>.
- [5] Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains[EB/OL].<http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf>.
- [6] Operation Cleaver[EB/OL].https://cdn2.hubspot.net/hubfs/270968/assets/Cleaver/Cylance_Operation-Cleaver_Report.pdf.