预研报告一：信息安全术语解释

刘鹏，2018-10-11

根据《术语》的相关介绍，对信息安全领域的一些术语进行了初步了解；同时，针对威胁信息共享机制的建立，进行了较为深入的了解，国外已经有组织进行相关实现了，对于他们的相关实现，目前仍在学习中。

1. 威胁（threat）

在计算机网络中，指任何情况或事件，有可能对组织运作（包括任务、职能、形象或声誉）、组织资产、个人、其他组织或国家产生负面影响，通过未经授权的访问、销毁、披露或修改信息和/或拒绝服务。[1]

2. 威胁信息（threat information）

任何与威胁相关的信息，这些信息可能帮助组织保护自己不受威胁或发现演员的活动。

3. 网络威胁情报[2]

网络威胁情报领域，混淆一直存在（很多还都是厂商弄出来的），威胁信息往往被当做了成品情报。

虽然情报过程从威胁信息收集开始，但信息收集仅仅只是个起始点而已。从大量获取信息，到产出成品情报之间，还有好长好长的一段路要走，二者之间简直是质的不同。

散布图中各处的 1000 个点，那是信息。但以某种形式连接各点显示出上下文和关联度(我们称之为“经评估的情报”)，那就是情报了。这些情报可用于为未来攻击做应对准备，支撑以前未知的风险，将精力专注到正确的领域。它还能帮助你从事件响应的角度理解发生了什么，为什么会发生，以及怎样发生的。

网络威胁情报(CTI)是一个生命周期过程，最终产出可被不同团体以多种方式消费的可交付产品(取决于所提供的威胁情报的级别——战略性、操作性还是战术性)。说白了，CTI 就是拉取指标馈送或涌入大量数据，并将这些指标应用到你的具体环境中。

威胁情报需要自动化，尤其是在数据收集、处理、筛选和部分分析方面，同时还需和人力分析结合。但人的因素往往在疯狂馈送中被无视掉了，这是非常错误的。

虽然现如今信息收集挺常见，但无论是从暗网还是从公开资源搜刮，信息获取都是相当简单的(受限黑市和论坛上需要伪装身份的情况例外)。这就仅仅是收集数据而已。或许也包含了一定的处理和过滤，但真正的秘方，还在于情报分析。

分析做对了，就能确保收集来的信息在准确度、相关性、时效性和完整性上都得到合理的评估。情报是要置入特定行业或公司的上下文中以获得不同的意见和决策的，而这需要人类的经验和对细节的关注。

最终，你需要信息来产生情报。然而，信息本身并不是情报，实际上甚至还有可能会让公司不堪重负，或是将公司指引向错误的方向。情报则能说明问题。信息只是提供大量可能的动作，情报则是有意义且有用的、具有可执行性的。情报支持计划制定，提供方向和焦点，最终帮助你在精力和资源分配上做出更好的决策。

分析威胁活动的时候，可以透过“方法途径”透镜来观察：

1. 目标行业——哪些特定公司或组织是攻击对象？
2. 目标技术——目标公司所用的哪种技术(如：Adobe Flash、IE 等等)可被利用来发起攻击？
3. 投送方法——攻击者怎样将攻击载荷投送到目标系统(如：鱼叉式网络钓鱼、第三方侵入等等)？
4. 漏洞利用——攻击者使用了哪个具体漏洞利用程序或已知(或未知)漏洞？
5. 存在形式——攻击者获取/使用什么级别的存在形式(如：特权账户、数据库访问等等)来展开攻击？
6. 达到的效果/伤害——攻击导致的影响是什么(如：知识产权被盗、服务中断等等)？

掌握方法途径可提供有意义的上下文，弄清威胁是什么，怎么进行的，威胁目标是什么，对公司的影响有哪些。成品情报包含此类分析，也包含威胁指标和支持性证据，还有置信度和实际动作建议。所以，情报不仅仅告诉你发生了什么和怎么发生的，还给了你影响评估和缓解步骤建议，从事件响应角度、风险策划和准备方式上给你帮助。

有关威胁情报，“待做事项”是没有得到充分讨论的一个方面。或许，某些厂商使用“可执行性”情报一词时指的就是这个意思，但除了“可执行”，情报还应给出解决迫在眉睫的威胁或已识别风险的实际任务。成品情报终究要是威胁信息纳入、信息评估和商业利益导出的结果，通常表现为对业务运营潜在影响风险的减小。

如果你还不能从当前 CTI 工作中轻松阐明商业利益，或者在创建新 CTI 功能时还没有定义它们，那你可能就仅仅是在收集威胁信息，而不是在做威胁情报。

4. 威胁信息与威胁情报的不同

（个人理解）一种可能的直观理解：威胁信息可以被定义为机器可读的客观的数据，它有可能是人不可读的；威胁情报可被定义为经过聚合、分析、预测等处理的人可读的信息，它必须是人类可读的。

问题：威胁情报必须是人类可读的吗？

回答：威胁情报是一种被定义为「为决策提供支持的最终文档」。如果进行最终决策的是人类，那么最终文档必然是人类可读的。如果进行最终决策的是机器，那么威胁情报或许是不必要的，这是因为机器可以读懂威胁信息，并且可以通过某种策略对威胁信息进行理解，然后采取相应的操作。近年来，有各种基于 AI 态势感知的自动化处理，但是目前来看，AI 替代不了人类直觉，这一点必须清醒认识。用人工智能和机器学习替代安全团队或许是安全行业中炒作最甚也最为危险的一个趋势。人类决策是创建和实现强企业安全不可或缺的，因为人类的洞察力可以补偿数学模型的固有局限。技术投资应聚焦支持安全团队和自动化繁琐任务，比如要求高度面向过程专业知识的取证调查。最好的团队民主化该职能，充分赋予人类员工做出重要风险管理决策的权力。[3]

5. Rootkit

Rootkit 是指其主要功能为：隐藏其他程序进程的软件，可能是一个或一个以上的软件组合；广义而言，Rootkit 也可视为一项技术。在今天，Rootkit 一词更多地是指被作为驱动程序，加载到操作系统内核中的恶意软件。因为其代码运行在特权模式之下，从而能造成意料之外的危险。最早 Rootkit 用于善意用途，但后来 Rootkit 也被黑客用在入侵和攻击他人的计算机系统上，计算机病毒、间谍软件等也常使用 Rootkit 来隐藏踪迹，因此 Rootkit 已被大多数的杀毒软件归类为具危害性的恶意软件。Linux、Windows、Mac OS 等操作系统都有机会成为 Rootkit 的受害目标。

在现代操作系统中，应用程序不能直接访问硬件，而是通过调用操作系统提供的接口来使用硬件，操作系统依赖内核空间来管理和调度这些应用。内核空间由四大部分组成，分别是：进程管理（负责分配 CPU 时间）、文件访问（把设备调配成文件系统，并提供一个一致的接口供上层程序调用）、安全控制（负责强制规定各个进程的具体的权限和单独的内存范围，避免各进程之间发生冲突）和内存管理（负责进程运行时对内存资源的分配、使用、释放和回收）。内核是一种数据结构，Rootkit 技术通过修改这些数据结构来隐藏其它程序的进程、文件、网络通讯和其它相关信息（比如注册表和可能因修改而产生的系统日志等）。例如，通过修改操作系统的 EPROCESS 链表结构可以达到隐藏进程的效果，挂钩服务调用表可以隐藏文件和目录，挂钩中断描述符表则可以监听键盘击键等等。Rootkit 至今仍然是一个发展中的技术领域。

Rootkit 一词最早出现在 Unix 系统上。系统入侵者为了获取系统管理员级的 root 权限，或者为了清除被系统记录的入侵痕迹，会重新汇编一些软件工具（术语称为 kit），例如 ps、netstat、w、passwd 等等，这些软件即称作 Rootkit。其后类似的入侵技术或概念在其他的操作系统上也被发展出来，主要是文件、进程、系统记录的隐藏技术，以及网上数据包、键盘输入的拦截窃听技术等，许多木马程序都使用了这些技术，因此木马程序也可视为 Rootkit 的一种。

6. 如何形成共享机制

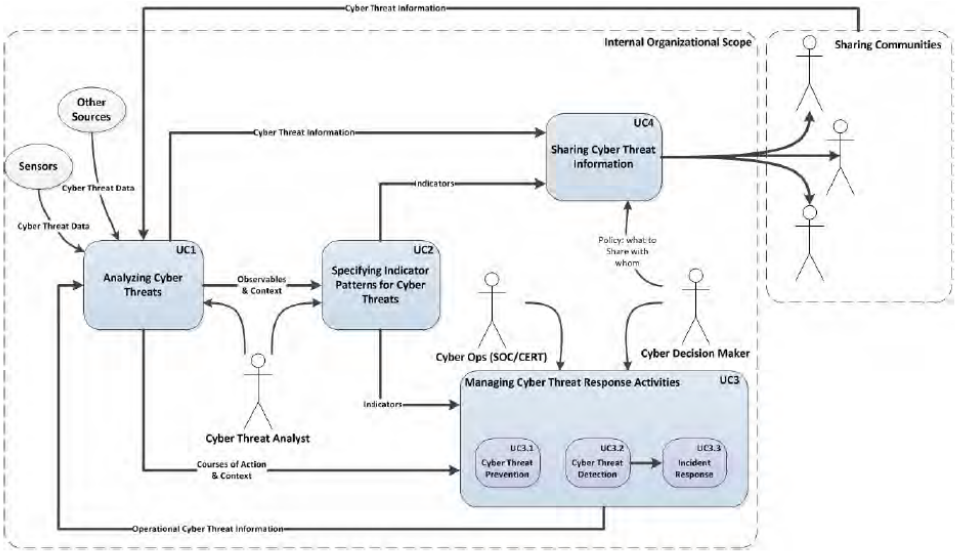
正如计算机网络标准化中产生的各种 RFC 文档一样，威胁信息的共享机制形成必然需要一系列标准化文档。标准化工作是共享机制形成的一个前提条件。

总得来看，目前的标准化机制的形成主要有两种形式（个人认为）：

1) 先有广泛使用的实现版本，然后根据流传度最广、接受度最高的版本制定标准化文档。如 C++的 STL 库，是在 HP 实验室的 SGI 版本实现并流传了较长时间之后才被 C++标准委员会所接纳。[4]
2) 先根据概念以及可能的需要，由标准化组织进行标准化探索并形成标准化参考文档，然后由企业或者组织根据标准化文档进行实现。这是一个标准化输入，个性化输出的形式。如计算机网络的 OSI 七层模型，这是 ISO（国际标准化组织）在不考虑美军内部网络信息互联背景下作出的前瞻性设计，其表示层（presentation layer）与会话层（session layer）的部分存在意义即是解决信息传输中的加密与安全。这一前瞻性思想在后来的 HTTPS 等协议中被借鉴，而诸多带有安全功能的应用层协议也充分体现了这一思想的前瞻性。

7. STIX

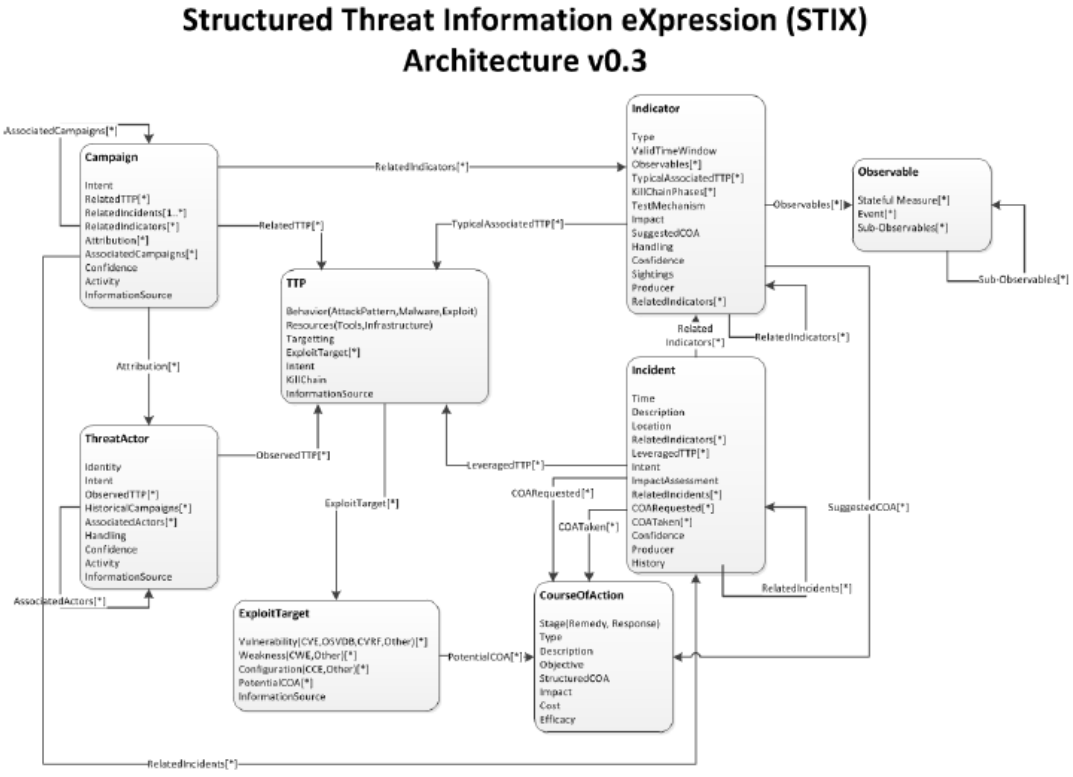
目前来看，有关 STIX 的信息可以找到不少，而且都比较有趣，下面展示若干相关摘要：



Core Use Cases Targeted by STIX

上图为「由 STIX（结构化威胁信息表示）所指明的核心使用案例，下称『案例』」，案例指出了两个基本的结构，即作为分享方的 Internal Organizational Scope（内部组织范围）和作为被分享方的 Sharing Communities（共享社区）。分享方的内部组织有四个成员，分别负责 Analyzing Cyber Threats（分析网络威胁）、Specifying Indicator Patterns for Cyber Threats（指定网络威胁的指标模式）、Managing Cyber Threat Response Activities（管理网络威胁响应活动）、Sharing Cyber Threat Information（共享 网络 威胁信息）

STIX 的表达式架构如下图所示：（Structured Threat Information eXpression, STIX, Architecture v0.3）



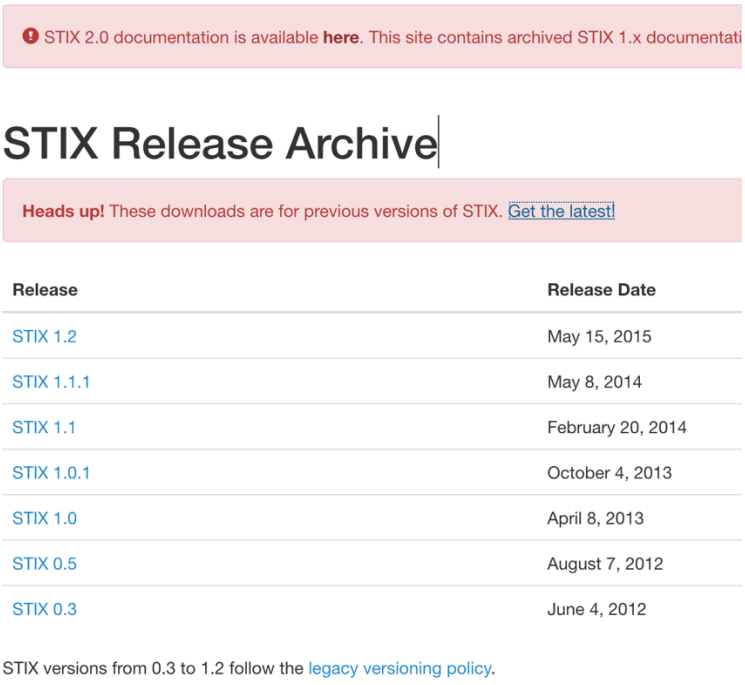
包括了对攻击行动、攻击入口、攻击目标、Incident 事件、TTP（攻击战术、技术和过程）、攻击特征指标、攻击表象、行动方针（COA，例如 IDS 规则等）的建模。

STIX 在结构化的表述网络空间威胁的时候，充分利用了 **MITRE**[5]已有的成果，例如用 CybOX 来表述攻击表象，用 IndEX 来表述攻击特征指标，用 CAPEC 和 MAEC 来表述攻击和恶意代码，用 IODEF 来表述 Incident，等等。了解 MITRE 的这套结构化的攻击、威胁的表达，有助于我们了解攻击、威胁的建模要素。

The MITRE Corporation's mission-driven team is dedicated to solving problems for a safer world. We are a not-for-profit company that operates multiple federally funded research and development centers (FFRDCs).

We work across the government, through our FFRDCs and public-private partnerships, to tackle problems that challenge our nation's safety, stability and well-being. Our unique vantage point allows us to provide innovative, practical solutions in the defense and intelligence, aviation, civil systems, homeland security, judiciary, healthcare, and cybersecurity spheres.

STIX 已经更新了好多个版本了，具体可以参看下表：



STIX 2.0 documentation is available **here**. This site contains archived STIX 1.x documentati

STIX Release Archive

Heads up! These downloads are for previous versions of STIX. Get the latest!

Release	Release Date
STIX 1.2	May 15, 2015
STIX 1.1.1	May 8, 2014
STIX 1.1	February 20, 2014
STIX 1.0.1	October 4, 2013
STIX 1.0	April 8, 2013
STIX 0.5	August 7, 2012
STIX 0.3	June 4, 2012

STIX versions from 0.3 to 1.2 follow the legacy versioning policy.

https://oasis-open.github.io/cti-documentation/stix/intro，这个网址给出了有关 STIX 的 12 个域对象，还有一些 examples，目前仍在探究这个组织是如何开发的。有一些网页，如 YouTube 上面的视频[6]，详细介绍了如何使用 STIX。

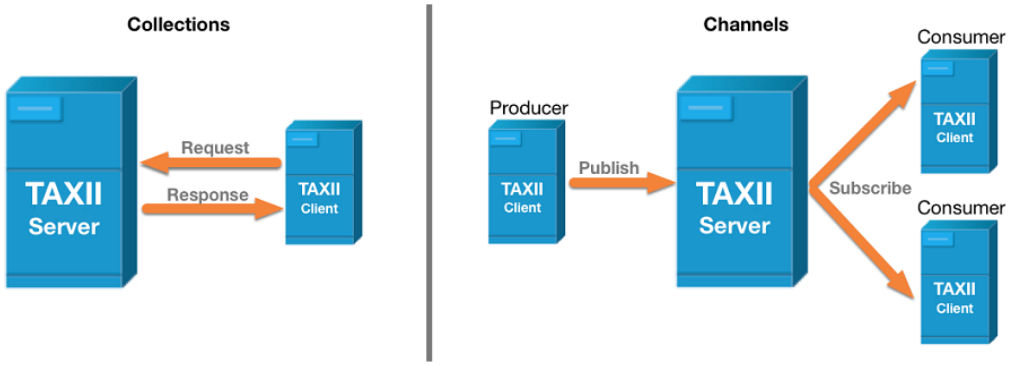
8. TAXII

Introduction to TAXII[7]

Trusted Automated Exchange of Intelligence Information（可信的智能信息自动交换）(TAXII™) is an application protocol for exchanging CTI over HTTPS. TAXII defines a RESTful API (a set of services and message exchanges) and a set of requirements for TAXII Clients and Servers. As depicted below, TAXII defines two primary services to support a variety of common sharing models:

Collection - A Collection is an interface to a logical repository of CTI objects provided by a TAXII Server that allows a producer to host a set of CTI data that can be requested by consumers: TAXII Clients and Servers exchange information in a request-response model.

Channel - Maintained by a TAXII Server, a Channel allows producers to push data to many consumers and consumers to receive data from many producers: TAXII Clients exchange information with other TAXII Clients in a publish-subscribe model. Note: The TAXII 2.0 specification reserves the keywords required for Channels but does not specify Channel services. Channels and their services will be defined in a later version of TAXII.



TAXII Collections and Channels（两种常用模型）

Collections and Channels can be organized in different ways. For example, they can be grouped to support the needs of a particular trust group.

A TAXII server instance can support one or more API Roots. API Roots are logical groupings of TAXII Channels and Collections and can be thought of as instances of the TAXII API available at different URLs, where each API Root is the “root” URL of that particular instance of the TAXII API.

TAXII relies on existing protocols when possible. In particular, TAXII Servers are discovered within a network via DNS Service records (and/or by a Discovery Endpoint, described in the next section). In addition, TAXII uses HTTPS as the transport for all communications, and it uses HTTP for content negotiation and authentication.

TAXII was specifically designed to support the exchange of CTI represented in STIX, and support for exchanging STIX 2.0 content is mandatory to implement. However, TAXII can also be used to share data in other formats. It is important to note that STIX and TAXII are independent standards: the structures and

serializations of STIX do not rely on any specific transport mechanism, and TAXII can be used to transport non-STIX data.

TAXII design principles include minimizing operational changes needed for adoption; easy integration with existing sharing agreements, and support for all widely used threat sharing models: hub-and-spoke, peer-to-peer, source-subscriber.

参考材料

[1] 术语在线: http://www.termonline.cn/list.htm?k=网络威胁

[2] 安全牛: *威胁信息和威胁情报有啥区别？*, https://www.aqniu.com/learn/24498.html

[4] 安全牛: *高效安全团队的 7 个习惯*, https://www.aqniu.com/learn/39407.html

[4] *STL 源码剖析*. 2002, 武汉: 华中科技大学出版社.

[5] MITRE 官网：https://www.mitre.org/centers/national-cybersecurity-ffrdc/who-we-are

[6] STIX 2 Objects Overview: https://www.youtube.com/watch?v=iAnd3rApMcA

[7] Introduction to TAXII: https://oasis-open.github.io/cti-documentation/taxii/intro.html