第 1 章 介绍

“我的电脑中毒了!”几乎所有接触过任意计算设备的人都说过或听别人说过这句话。这些年我们也经常能看到有关病毒攻击新闻，其中一些攻击甚至影响了全球数百万用户。作为安全研究人员，我们反复向公众解释病毒这个表述并不准确。正确的术语应该是恶意软件，而病毒只是恶意软件的一种。

什么是恶意软件？恶意软件是恶意实体用来实现险恶意图的武器。从技术上来说，恶意软件是恶意的软件：即一种达成恶意目的的软件。恶意软件已经存在很久，但在计算设备的早期，个人用户几乎不关注它。与其他行业相比，银行、金融与政府等行业更关注恶意软件的攻击。但随着时间的推移，恶意软件的格局也发生了翻天覆地的变化。从前，似乎各种攻击都与金钱有关。但随着数据成为了渗透在生活方方面面的“货币”，数据自然也就成为了恶意软件的主要目标。为了确保数据受到保护、数据保护相关的法律能够得到严格执行，任何存储公众信息的组织都必须对任何形式的数据滥用和数据丢失负责。世界上不应该有任何组织再将网络安全视为不需要付出的理所当然。

不仅是组织，终端用户也不能掉以轻心。计算设备的类型与可用性在过去的数十年中发生了巨大变化，个人电脑与手机可以进行银行交易、酒店预订、航班预定、水电费支付，甚至可以充当汽车钥匙、操作其他家用电器与控制物联网设备。个人设备中保存着大量的隐私数据，包括用户名、密码与各种照片。时至今日，已经没有人能够承受被黑客攻击的后果。过去的恶意软件攻击通常直接与公司或者机构相关，而如今的恶意软件攻击通常针对普通用户的计算设备以从中获利。

几乎攻击者每一次发起的网络攻击都离不开恶意软件，每天对外分发的恶意软件会达到数百万之巨。但实际上，能够处理恶意软件的专业人员数量远远少于处理这些恶意软件所需的专

业人员数量，有能力对恶意软件进行检测与分析的专业人员甚至更少。恶意软件分析是一个不断发展的学科，专业人员也需要了解更多有关分析恶意软件的知识。有些研究中预计恶意软件分析市场规模将从 2019 年的 30 亿美元增长到 2024 年的 110 亿美元。这一预测基于朴素的事实：恶意软件的数量每天都增加并且新的技术也在不断涌现与被广泛应用。此外，云与物联网等新兴计算平台也为恶意软件提供了新的攻击面，攻击者可以以此为攻击目标并从中获利。不仅是攻击面拓宽并且复杂性增加，叠加持续缺乏具备相关技能的专业人员带来的影响，防御方的恶意软件分析工作在很大程度上仍然是无人处理的搁置状态。本书中对恶意软件分析工作流程的全面介绍，能够确保各种读者（恶意软件分析人员、逆向分析工程师、网络工程师、安全运营中心（SOC）分析人员、IT 管理员、网络管理员、经理或者首席信息安全官）都可以提高对恶意软件分析的认识与进行逆向分析的能力，以及加强应对任何类型恶意软件攻击的准备。与此同时，本书对反病毒引擎、沙盒、IDS/IPS 和其他恶意软件检测工具的架构与组件介绍，能够让读者以全新的视角看待这些检测工具并且能够自行进行定制化以改进恶意软件分析的基础架构。在深入学习如何分析恶意软件之前，首先了解一下各种类型的恶意软件及其功能相关的术语。

注意：病毒是恶意软件中的一种，除此之外还有很多其他类型的恶意软件，如僵尸网络、木马、远控（RAT）、勒索软件等。

恶意软件的类型

作为恶意软件分析人员，不仅需要对样本文件进行分析，还需要阅读互联网发布的分析报告、博客、技术文章以及讨论全球恶意软件与网络攻击的其他内容。恶意软件分析领域已经为恶意软件及其功能定义了各种常用的术语，接下来会对其进行介绍。这些术语可以用来描述恶意软件，例如恶意软件的代码、特性或功能等。以下是一些常见的恶意软件类型或功能：

- 病毒是已知的首种可以自我复制的恶意软件。病毒也被称为文件感染器，通过将自身插入系统上其他良性文件中来保持存活。这些受感染的程序在运行时，不仅会执行预期的功能，也会在后台执行病毒。
- 蠕虫是一种恶意软件，同时也是一种恶意软件的功能，表示恶意软件可以通过网络或者某些物理方式（如 USB 设备）来传播和感染到其他计算机。
- 后门是一个未授权的入口点，攻击者可以通过它进入受害者的系统。例如，恶意软件可以在具有访问权限的系统上创建一个对外开放的网络端口，攻击者可以通过该端口进入系统。
- 木马是一种伪装成良性软件的恶意软件，在受害者知情的情况下安装到机器上，但用户并不了解其恶意意图。
- 间谍软件或者信息窃密软件会监视并窃取系统中的敏感数据，例如用户名、密码、图片与文档。
- 键盘记录器是间谍软件的一种，可以记录用户的按键行为并将其发送回攻击者。
- 僵尸网络是由多台被恶意软件感染的机器组成的机器人网络。形成的僵尸网络会作为一个整体协同工作，接受并执行攻击者发送的命令，例如拒绝服务（DOS）攻击、发送垃圾邮件等。
- 远控（RAT）是一种恶意软件，同时也是一种恶意软件的功能，攻击者可以通过其完全控制失陷主机。这种恶意软件与管理员访问系统进行故障排除的桌面共享软件非常相似，主要的区别在于远控是攻击者用来在未授权的情况下访问失陷主机的。
- 广告软件是一种常见的恶意软件，大多数人都遇到过但可能并未注意。许多广告软件都会包含在通过第三方网站下载的软件中，安装下载的软件的同时，广告软件就会在用户不知情的情况下在后台安装。值得注意的是，并非所有的广告软件都是恶意的。当用户

仍然可以将其归类为木马的类别，只是该类木马只会在系统上显示无用的广告，其中部分也会篡改计算机浏览器上的默认搜索引擎。

- Rootkit 是一种恶意软件，同时也是一种可以与其他恶意软件配合的恶意软件功能。Rootkit 主要通过修改系统函数与数据结构来隐藏自身或者系统上另一种恶意软件的活动。
- 银行恶意软件通过拦截、修改浏览器通信流量，获取银行交易和凭据信息。
- PoS 恶意软件会感染零售店、购物中心和餐馆等场所使用的 PoS 设备，该类恶意软件会尝试从 PoS 软件窃取信用卡信息。
- 勒索软件会劫持系统上的数据、文件和其他系统资源，要求受害者支付赎金才能释放这些资源。与其他类型的恶意软件相比，勒索软件相对容易实现。与此同时，从事后补救的角度来看，勒索软件的处置难度也很大。数据等资源一旦被加密，就会给用户造成巨大的损失，并且需要巨大的投入才能清除其带来的影响以及将系统恢复到正常的状态。
- 挖矿是恶意软件领域中的后起之秀，随着加密货币的普及而变得愈发火热。这种恶意软件不一定会从受害者的机器上窃取数据，但肯定会通过挖掘加密货币来消耗系统资源。
- Downloader 是用于下载其他恶意软件的恶意软件。僵尸网络可以作为 Downloader，在收到攻击者的命令后下载指定的恶意软件。如今，大多数基于 Microsoft Office 文件的、嵌入恶意宏代码的文档文件都是 Downloader，攻击者利用其下载其他恶意软件。Emotet 是一种常见的恶意软件，也是使用基于 Microsoft Office 文档文件的 Downloader。
- 垃圾邮件是其中可能包含恶意网站链接、恶意附件文件等恶意内容的电子邮件，攻击者可能会利用失陷主机来发送垃圾邮件。恶意软件可能会利用安装在失陷主机上的 Microsoft Outlook 等电子邮件客户端获取联系人列表，并给这些联系人发送电子邮件。

- Exploit 不是恶意软件，而是恶意代码。Exploit 旨在利用系统上的漏洞并利用它来控制存在漏洞的应用程序，从而控制系统。如今，大多数 Exploit 都会被用于下载其他恶意软件。

平台多样性

人们常常会质疑哪种编程语言经常被用于创建恶意软件，但其实几乎任何编程语言都可以用于编写恶意软件，例如 C、JavaScript、Python、Java、Visual Basic、C#等。攻击者还会使用一种被称为 Living Off the Land 的技术，利用操作系统本身提供的工具来达成攻击者的恶意目标。

目标多样性

恶意软件开发者创建恶意软件来攻击特定目标，例如特定的人、特定的地区、特定的组织（如政府、军队、公司等）、特定的行业（如金融、医疗等）。攻击者期望在经过开发和测试后，不需要增加任何特别的措施就可以针对任意个人或机器发起攻击，能够在尽可能多的平台与设备上运行。

它们主要通过包含恶意附件的恶意电子邮件、通过恶意网站或失陷网站提供的漏洞进行传播。例如，垃圾邮件所需的电子邮件地址由攻击者通过爬虫爬取公开信息、从受害者的账户中获取相关信息，或者通过入侵某些网站的数据库甚至从地下市场购买而来。

恶意软件攻击可以是定制化的，并且经常根据地理位置进行限制，例如只感染使用特定语言（如乌克兰语、中文）的计算机。或者攻击者也可以针对特定 IP 地址范围进行限制，配置所针对区域对应的 IP 地址范围。以根据地理位置进行限制为例，某些勒索软件会根据特定地理位置对应的语言显示勒索信息。

攻击组织还会创建恶意软件来感染特定的个人、公司或者组织。这些有针对性的攻击和恶意软件被成为高级持续威胁（APT），并且根据目标所使用的设备、操作系统与软件进行定制化开发。这些恶意软件与攻击活动预期在失陷主机上停留相当长的时间，并且会使用先进的隐蔽技术以避免被发现。Stuxnet 是一种臭名昭著的恶意软件，也是针对伊朗的 APT 攻击的一部分。Stuxnet 针对伊朗核电站使用的工业控制系统（ICS）进行攻击，此类攻击通常是由经验丰富的、资金充裕的攻击组织发起，且通常都是国家资助的攻击组织。

网络杀伤链

网络杀伤链（Cyber Kill Chain）是由洛克希德马丁（Lockheed Martin）公司提出的模型，用于描述外部攻击者针对组织执行 APT 攻击的各个阶段。杀伤链描述了攻击者达成目标所需的所有步骤，攻击者可能意在进行数据泄露或间谍行动。如果安全相关人员能够识别出任何一个攻击阶段并成功阻止攻击，就可以破坏整个攻击计划。

网络杀伤链旨在帮助组织识别攻击的不同阶段，并在不同的阶段采取适当的措施来阻止攻击。根据洛克希德马丁公司的研究，以下是网络攻击必经的七个阶段：

1. 侦察。侦察包括观察目标并通过各种来源收集有关目标的各种信息，例如服务器详细信息、IP 地址、使用的软件以及可能存在的漏洞。该阶段可能涉及收集组织中员工的个人信息，以针对潜在受害者进行社会工程学攻击。收集信息的方式包括主动与被动两类：主动方式包括端口扫描等直接获取信息的方法，被动方式包括从各种来源获取电子邮件地址等间接获取信息的方法。
2. 武器化。武器化阶段主要是设计可以穿透组织基础设施并感染系统的武器。攻击者最重要的武器之一，是根据侦察阶段发现的漏洞针对性开发的攻击代码。其他武器例如垃圾邮件，也可以将需要部署到目标基础设施中的漏洞和恶意软件投递到位。

3. 投递。投递阶段主要将武器投递给受害者，该阶段旨在将武器成功运送至目标组织中。例如向员工发送垃圾邮件，其中包含指向带有漏洞利用或者恶意软件的网页的恶意链接。其他社会工程学方法，如蜜饵诱捕也可用于投递阶段。
4. 漏洞利用。漏洞利用阶段主要关注漏洞利用的执行，成功执行后会导致攻击目标上的软件被破坏。包括 Web 服务器、用户浏览器在内的各种软件，都有可能会被攻击。如果软件未打补丁，就更有可能会被已知的漏洞甚至是 0day 漏洞攻击。漏洞利用的过程不一定遵循固定的模式，毕竟恶意软件也可以在无需受害者的情况下通过其他方式（如电子邮件中的附件）传送到目标。
5. 安装。安装阶段旨在目标网络或者系统中安装恶意软件。如果针对目标软件的漏洞利用成功，一般就会安装恶意软件。安装成功的恶意软件通常也不是孤军奋战，也能够下载其他恶意软件，并且会期望能够在目标网络中尽可能地隐藏、不被发现。
6. 命令与控制。命令与控制阶段主要在已安装的恶意软件与攻击者之间建立通信关系，使恶意软件能够接收攻击者的命令并采取对应的行动。
7. 行动。行动阶段是杀伤链的最后一个阶段，此时恶意软件已经安装在目标网络中并准备好接收攻击者的命令。恶意软件期望能够实现其创建时预定的攻击意图，例如监视目标网络内部、收集敏感数据并将其回传给攻击者、劫持敏感数据和基础设施等。

恶意软件攻击生命周期

最初的黑客通常是以感兴趣、好玩为动机，而现在则变得更加多元。目前，更多的攻击者往往会出于经济利益或者其他动机，例如资金充足、组织严密的网络攻击团伙与犯罪分子会进行复杂的间谍行动。网络战争中会使用恶意软件作为主要武器，强力的网络攻击也可以摧毁一个国家。

恶意软件通常是根据攻击者的需要，为达成不同的目标而开发的。随后攻击者开始分发恶意软件，以便其能够绕过攻击目标的安全边界并到达目标系统。仅仅到达目标系统也是不够的，恶意软件还需要成功绕过现有的防御措施并成功实现感染。恶意软件生命周期中，最后的阶段是在成功感染后达成预定的攻击目标，例如获取经济利益、进行间谍活动或者其他目标。图 1-1 显示了恶意软件生命周期的各个阶段。

开发阶段

在分析恶意软件的工作中，经常会遇到无法由一个人完成编写的恶意软件。恶意软件与良性软件在开发阶段并没有什么不同，从恶意软件开发过程中也能够清楚地看出这一点。恶意软件开发者似乎也会采用软件开发生命周期方法，就像普通软件公司的开发团队一样。与各种良性软件一样，恶意软件也是以模块化方式编写的，不同的模块分配给不同的开发人员实现。通常，不同的恶意软件家族中也会识别出相同的模块代码。可能是由于该模块部分的开发者相同，或者某些模块是从其他攻击组织或者攻击者处购买或交换而来。

与良性软件要进行质量保证（QA）类似，恶意软件也会经历测试阶段，以确保其能够按照攻击者的预期正常运行。并且，许多恶意软件也会像良性软件一样接收更新。最终，完成开发的恶意软件会通常在被加密或者被加壳（第七章中将会介绍加壳）后，再针对反病毒或其他恶意软件检测产品进行测试，以确保恶意软件不会被这些安全产品检测到。

自我防护

技术的发明是为人类服务的，但不可避免的会有人滥用技术。众所周知，网络世界的犯罪分子就是这样做的。例如，开发密码算法是为了保护系统上的数据以及各种跨网、跨系统传输的数据。密码学是一门复杂艰深的学科，密码学家花费数年时间开发算法并确保其牢不可破。

尽管密码算法是为保护数据而开发的，但也被恶意软件开发者滥用来保护恶意软件不被检测与分析。再例如，由攻击者会对正版软件进行逆向工程来开发破解版本，让使用者无需支付许可证费用即可正常使用软件，这就是所谓的盗版软件。为了遏制盗版软件，软件开发人员设计了几种反盗版与反逆向工程的技术。恶意软件开发者也会滥用这些技术，来阻止恶意软件研究人员针对恶意软件进行分析，从而使安全厂商难以编写有效的签名来检测恶意软件。

恶意软件的适应性和欺骗性

与现实世界中的病毒与细菌类似，恶意软件也会进化。恶意软件会不断适应环境的新变化，对反恶意软件防御系统产生抵抗力。例如，许多恶意软件都会检测、规避与终止系统上的各种检测软件。此外，当恶意软件确定自身在反恶意软件产品与分析工具，甚至是恶意软件分析人员使用的分析环境中执行时，并不会显示出真正的恶意行为。这时，有些恶意软件会采取伪装的方式开始执行良性操作，从而避免暴露其真正的恶意意图。本书的后续章节，将会讨论各种反虚拟机、反逆向分析与其他对抗技术。

大量生产恶意软件

恶意软件开发者可能需要相当长的一段时间来编写恶意软件，并且对其进行测试以确保其在各种环境中都能够正常工作。但是，如果任何安全厂商捕获了该恶意软件并针对其开发了签名来实现检测，恶意软件开发者的努力就会付诸东流。利用签名，在任何计算上发现完全相同的恶意软件时，相同的反病毒软件就可以轻松检测到恶意软件，恶意软件开发者的投入就会化为泡影。为了防止这种情况发生，恶意软件开发者利用“人海战术”进行对抗。攻击者使用被称为多态加壳工具或者加密工具的程序，基于单个恶意软件创建大量恶意软件变种。从功能上来看，生成的恶意软件变种的最终目标与相关行为都不会变化。但这些恶意软件在

二进制文件内容和结构上看起来是不相同的，故文件对应的哈希也都是不相同的。数百万个在结构与内容上并不相似、但能表现出相同行为的恶意软件正是使用加壳程序创建的，并被攻击者应用于在野攻击。部分优秀的反病毒引擎会检测出一部分，其余的就会成功感染受害者。恶意软件采用此类技术也推动着安全行业研发下一代反病毒软件，通过行为而不是仅通过静态特征或者哈希值来识别恶意软件。

投递阶段：多样化手段

恶意软件的目标是要在失陷主机上执行，在此之前需要被投递到预定攻击目标。为了传播恶意软件，攻击者会使用各种投递方式。以下是一些投递方式：

- 漏洞利用工具包（Exploit Kit）
- 电子邮件（垃圾邮件或恶意附件）
- 广告
- USB 设备
- 其他社会工程学技术

在第 6 章将会详细介绍这部分内容。

感染阶段

在恶意软件投递到预定目标后，恶意软件需要绕过重重阻碍才能成功感染系统而不被发现。想要成功感染可能会遇到的问题如下所示：

- 反病毒软件。大多数恶意软件最大的威胁就是反病毒引擎，如果是新创建的恶意软件，被反病毒引擎检出的可能性就比较小。
- Bug。如果恶意软件开发时存在编码不正确或者 Bug，就可能无法成功感染预定目标。

- 缺乏合适的执行环境。有时由于缺乏合适的环境（依赖与库文件），恶意软件无法在预定目标上正常执行，可能会导致执行失败甚至程序崩溃。例如，如果预定目标上没有安装 Java 虚拟机，用 Java 编写的恶意软件就无法在其上执行。

后感染阶段

在成功感染后，恶意软件需要达成攻击者的攻击目标。恶意软件可能会尝试与攻击者建立联系获取攻击者的指令，如窃取数据、窃取凭据、窃取个人隐私信息、软件升级或者向攻击者提供远程访问权限等。

恶意软件商业模式

并非所有的恶意软件攻击都是出于经济利益，但经济利益确实是大多数攻击的主要动因。一个典型的例子就是银行类恶意软件，该类恶意软件使用浏览器中间人技术劫持用户的银行交易。同样的，POS 类恶意软件也会窃取用户的信用卡信息，勒索软件会劫持用户的数据来勒索钱财。

对恶意软件开发者或者经销商而言，恶意软件即服务（MaaS）是地下恶意软件社区中蓬勃发展的业务。犯罪分子借助该类服务就不需要成为技术型黑客或者计算机大牛，也能发起网络攻击。唯一要准备的就是金钱和个人资料，以此说服卖家购买者本身是真正的客户，而不是执法机构伪装的。

恶意软件构建工具也是其中一项服务，用于为特定攻击创建定制化的恶意软件。恶意软件即服务也提供各种其他服务，包括构建各种攻击基础设施（如 C&C 服务器、执行感染所需的漏洞利用工具包、垃圾邮件与恶意软件广告服务）提供漏洞利用与恶意软件。用户甚至可以租用僵尸网络进行 DDoS 攻击或者发送垃圾邮件。

此外，恶意软件开发者与攻击组织在收钱时也格外小心，他们想要确保在不被执法机构追踪的情况下获取这笔不法收入。大多数勒索软件会要求受害者使用比特币、门罗币或其他匿名加密货币，并且通过匿名 Tor 网络进行赎金支付。通常，攻击者的银行账户位于第三世界国家，为国际执法机构的调查与追踪带来了困难。

与恶意软件的战争

到目前为止，一直在介绍关于恶意软件的方方面面，都是网络空间中的黑暗面。但大到整个网络安全行业，小到反恶意软件的细分门类，都旨在打击网络与恶意软件攻击。与恶意软件的斗争极具挑战，且要富有奉献精神。由于新的恶意软件一直在发展并且层出不穷，但网络安全从业人员的数量相对于泛滥的恶意软件而言始终是有限的。与此同时，恶意软件研究也再不是一个局限于小范围的课题。随着恶意软件的多样性日趋增加，攻击者不断扩大技术范围覆盖新的编程语言、新的操作系统与其他新的组件，他们编写的恶意软件也越来越难以被分析与破解。此外，随着物联网（IoT）设备和移动设备的时代到来，平台与设备的激增意味着攻击面的极具扩大，也增加了本就过载的恶意软件分析人员与反恶意软件团队的工作量。接下来介绍一下每天都要与恶意软件作斗争的各种”英雄“：

战斗在一线的各类人员

从事恶意软件分析的人并不多，且通常都在组织良好、架构清晰的机构中工作。反恶意软件团队有主动类的工作，也有被动类的工作。主动类的工作需要始终保持警惕，留意新的恶意软件趋势并做好准备。被动类的工作是在组织发生恶意软件相关事件时，采取必要的行动。如今，大多数组织（无论是安全公司还是金融公司）都有处理恶意软件的团队，只是工作性质因情况而异。大多数组织也都会配备事件响应与取证团队来处理安全事件，有些组织可能

还配备了专门的恶意软件分析人员，确认可疑活动是否与恶意软件有关或者样本文件是否为恶意软件。另外，也有负责其他任务的团队，例如恶意软件狩猎、恶意软件检测工程等。接下来简要介绍这些不同类型的人员与角色。

恶意软件猎人

恶意软件猎人会主动留意恶意软件趋势，他们会寻找在野的新恶意软件感染并收集与之相关的各种信息。借助恶意软件猎人的力量，组织能够在预防恶意软件感染上保持领先地位，并且在最坏的情况下为感染爆发做好准备。如下为一些寻找恶意软件技术：

博客、分析报告与其他共享资源

网络安全行业中有许多反恶意软件团队与 SOC 团队，大家都在积极努力让世界了解恶意软件活动的最新趋势。这些团队会通过社交媒体发布在环境中发现的新威胁、新恶意软件的分析报告，以及攻击者采用的新攻击技术。密切关注这些来自世界各地不同公司的反恶意软件团队发布的内容，是了解恶意软件最新趋势的好方法。同时，不同组织的研究人员也会通过公开或者非公开的邮件列表创建各种联盟与团体。成为此类联盟和团体的一份子，是与同行快速交换信息的方式之一。尤其是在实时的网络攻击期间，与攻击相关的即时信息很可能是私密的内部信息，不会被公开披露。

蜜罐

作为一种主动分析方法，恶意软件猎人通常会使用蜜罐来捕获恶意软件。蜜罐会故意设置成系统或资源易受攻击且易于访问，以吸引恶意软件和其他想要感染系统或资源的攻击者。使用遍布全球不同区域的蜜罐，将蜜罐模拟与伪装成其他类型的设备，就可以捕获各种新的攻击团伙与恶意软件。

网络爬虫

网络爬虫是另一种被反恶意软件团队广泛采用的主动分析方法，常用于检测在野的最新感染情况。攻击者经常会使用互联网中易受攻击的 Web 服务器作为中间跳板，利用其进行漏洞攻击甚至部署恶意软件。网络爬虫的实现方式是模拟最终用户访问网站，智能地在全网搜索受感染的服务器，诱骗服务器作出响应返回漏洞利用或者恶意软件。

深入地下市场

地下市场承载着各种见不得光的恶意行为，包括销售漏洞、恶意软件与被窃数据等。地下市场通常以”深网“与”暗网“中仅限邀请注册的论坛的形式出现，用户可以通过像 Tor 这种匿名网络进行访问。有时，恶意软件猎人还需要打入地下市场中，伪造身份并伪装成网络犯罪分子，以此追踪攻击者与即将到来威胁或者其他恶意活动。有时，恶意软件猎人还要与这些地下市场中的其他网络犯罪分子分享信息，以获取他们的信任再调查更多信息。

应急响应与数字取证

应急响应团队（安全运营中心/SOC 的一部分）与数字取证团队会在组织发生安全事件或发现感染后采取行动，立刻采取措施遏制感染的进一步传播。通常来说，应急响应人员会将受感染的设备与网络进行隔离以防止感染扩散，并支持进一步调查感染的根源与痕迹。紧接着，取证分析人员介入，取证分析人员从应急响应人员提供的隔离的失陷主机中找出感染的根源。取证分析人员不仅要在失陷主机中寻找恶意软件，还需要寻找其他与攻击有关的痕迹，例如恶意软件与攻击是如何入侵计算机的。有时取证分析人员也需要搜索其他信息来源，了解攻击者的身份与目标。然后，将提取的恶意软件转交给恶意软件分析人员进行进一步分析。有时，发现的恶意软件也会与其他反病毒厂商共享，以便厂商可以为其编写检测签名。

恶意软件分析团队

恶意软件需要被进行深入分析，这也是需要恶意软件分析团队的地方。所有的恶意软件都应该分发给恶意软件分析人员，由恶意软件分析人员对恶意软件进行逆向分析以获取有关恶意软件的功能、攻击者的信息、执行留下的痕迹与其他 IOC 指标。对恶意软件深入分析有助于遏制感染，并能够采取主动措施编写签名来检测未来的恶意软件感染。

检测团队

企业想要自我保护，可以通过部署多层检测解决方案来实现。但这些检测解决方案需要来自 SOC 与 IT 团队根据新检测签名进行持续更新，以便跟上新兴威胁的步伐。此外，安全公司需要不断升级他们的检测解决方案，以确保能够发现任何新的感染与新的恶意软件，这些可能是从前未能在用户侧发现的。检测团队的工作是基于感染与恶意软件的分析信息，不断升级检测签名并改进检测产品本身，以确保能够在未来尽可能多地发现感染。

反恶意软件产品

任何想要保护自身安全的组织都应该在基础设施中构建“纵深安全”，使用各种类型的检测解决方案协同防护。在第 6 章中将会深入探讨多个典型的检测解决方案，此处先简要介绍部分解决方案以及这些解决方案如何构成安全基础架构。

杀毒软件

反病毒软件是最早的反恶意软件产品。反病毒软件是一种安装在计算机设备上的应用程序，最早依赖于在文件中寻找特定模式串来识别恶意软件。这些特定的模式串被称为静态签名，在包含相同模式串的恶意软件的基础上创建而来。与此同时，还需要确保其他良性的文件不

包含相同的模式串。随着时间的推移，恶意软件开发者开始使用诸如多态、加壳等对抗技术。应用对抗技术的恶意软件，可以一次性产生数百万个变种。编写静态签名来检测数百万个恶意软件样本文件是极具挑战性的，通过静态签名检测恶意软件变得愈发困难，业界需要能够通过动态行为检测恶意软件的解决方案。如今，大多数反病毒软件都已经能够根据行为检测恶意软件。而且，现在反病毒软件也已经从台式机与服务器扩展到移动设备。

防火墙、IDS/IPS 与网络安全产品

反病毒软件会在主机侧发现并阻止感染，恶意软件也可以通过网络侧与命令与控制（也被称为 C&C、CnC 与 C2）服务器进行通信，例如接受来自攻击者的命令、上传受害者的数据、扫描主机上的其他设备、通过网络进行传播感染等。

一些基于网络的安全产品可以阻止网络上的恶意软件，包括防火墙、入侵检测与防御系统、网络访问控制（NAC）等。这些基于网络的安全产品会监控漏洞利用、来自攻击者的 C&C 流量、恶意信息上传以及与恶意软件有关的其他类型的流量。最初，这些网络安全设备也基于静态签名进行检测。但目前，最新一代的产品已经开始使用基于网络行为的异常来检测恶意软件的流量与感染。

沙盒

沙盒算是较新的安全产品。沙盒会在受控的封闭执行环境中，执行恶意软件与各种类型的样本文件以观察其行为并识别感染。

术语

在本节中，将会介绍网络安全领域中经常会遇到的一些常见术语。了解这些术语有助于阅读业内同行发布的恶意软件与威胁分析报告。这个列表并不完整，当用户发现不了解的新术语

时，请首先花时间搞清楚其含义。

- APT：高级持续威胁（APT）攻击也被称为针对性攻击，是针对特定国家、组织与特定人士实施的攻击。攻击可能会持续较长时间，攻击期间目标会被持续监控。这种攻击通常是出于间谍目的，也有针对商业竞争对手的攻击。
- 漏洞：漏洞是软件中的错误，它会危害并控制软件及其运行的系统。
- 漏洞利用：漏洞利用是一些小程序，旨在针对软件中的漏洞发起攻击并控制系统。
- Shellcode：Shellcode 是在漏洞利用中用于达成目标的小代码片段，支撑攻击者控制系统。
- 漏洞利用工具包：漏洞利用工具包通常是部署在服务器上的，主要由与浏览器和浏览器插件相关的漏洞利用程序组成。
- 恶意广告：恶意广告是通过广告向受害者分发恶意软件的机制，通常包含广告与指向恶意网站的链接。
- 垃圾邮件：垃圾邮件是攻击者向目标发送未经请求的或者不相关的电子邮件，其中包含恶意软件或者其他指向恶意网站的恶意链接，以收集受害者的隐私信息或者分发恶意软件。
- 无文件攻击：无文件攻击不需要在失陷主机上创建恶意软件，而是在内存中传输与运行 Payload。
- Living off the land：Living off the land 是一种攻击技术，攻击中攻击者不使用任何基于恶意软件的 Payload，而是使用失陷主机上预装的软件来执行恶意行为。
- 路过式下载：路过式下载是一种无意的、自动的将恶意软件下载到受害系统的行为，漏洞利用工具包与恶意广告通常是攻击者用来实施路过式下载攻击的载体。
- 反病毒软件：反病毒软件是安装在系统上的、旨在检测系统上恶意软件感染的软件。

- EDR：端点检测与响应（EDR）被认为是下一代反病毒软件，不仅可以根据传统的静态签名方式来检测恶意软件，还可以通过包括恶意软件行为在内的其他技术检测恶意软件。
- IDS/IPS：入侵检测系统（IDS）与入侵防御系统（IPS）是网络安全产品，用于识别与阻止恶意流量再网络上的传输行为。
- 沙盒：沙盒是自动化、隔离的恶意软件分析解决方案，能够以受控方式执行恶意软件并记录与观察恶意软件的恶意行为。
- DLP：数据防泄漏（Data Loss and Prevention, DLP）是一种旨在防止员工无意或故意泄露敏感数据以及感染恶意软件的系统。
- 内存取证：内存取证是一种取证分析技术，通过识别系统虚拟内存中的痕迹以发现系统上的恶意软件感染。
- 网络杀伤链：网络杀伤链是组织在经历网络攻击期间涉及的一般步骤，从侦察到感染再到达成攻击目的。
- 事件响应（IR）：事件响应是响应网络攻击事件、隔离失陷主机并遏制感染继续扩散到其他系统的过程。
- 数字取证：数字取证是调查网络攻击的过程，包括识别与检查失陷主机中攻击者留下的各种痕迹与攻击中使用的各种工具。。
- 威胁狩猎：威胁狩猎是主动寻找网络中威胁的过程，通过查看安全产品与系统的日志，寻找网络上可能已经存在的威胁。
- TTP：战术、技术和程序（TTP）是对攻击者实施网络攻击的技术和步骤的描述，针对TTP 的检测与识别有助于将攻击者与 APT 联系起来。
- IOC：攻陷指示指标（IOC）是指攻击者留在系统上的相关痕迹，这些痕迹能够表明系统已经被入侵。

- IOA：攻击指示指标（IOA）主要用于识别攻击者的攻击意图，而不考虑用于执行攻击的工具与恶意软件。
- Payload：Payload 是恶意软件实现功能的核心组件。
- 持久化：持久化是恶意软件在机器重启或者用户重新登录后保持存在的一种机制。
- 代码注入：代码注入是恶意软件用来将恶意代码注入另一个合法运行的目标进程，再通过目标进程执行恶意代码的技术。
- Hook：Hook 是恶意软件用来通过拦截库和系统 API 调用并修改这些拦截的 API 调用，来改变目标进程或者内核原始功能的技术。
- 加壳：恶意软件开发者将恶意软件 Payload 封装在另一层代码中，以隐藏恶意软件实际的功能的程序被称为壳。加壳程序可以通过压缩或者加密等操作，混淆恶意软件的 Payload 数据。
- Rootkit：Rootkit 是一种恶意软件组件，通过使用 API Hook 在代码级别更改操作系统或通过篡改操作系统数据结构来进行隐藏。
- 横向平移：横向平移是一种机制，恶意软件可以通过该机制从一台机器传播到另一台机器，或是检索发现想要感染的其他系统、资源。
- 命令与控制：命令与控制（C&C\C2\CnC）是被攻击者当作命令中心，用来控制恶意软件并与之通信的一种系统。
- Tor：Tor 既是一种网络协议，也是攻击者在进行攻击时用来保持匿名通信的主要工具。
- DGA：域名生成算法（DGA）是恶意软件用来生成大量随机域名以与 C&C 服务器进行通信的算法。生成域名中的部分可能会在短时间内被注册为 C&C 服务器使用。攻击者不仅能够使用 DGA 来对抗 IDS/IPS 通过签名检测并阻止 C&C 通信，还可以提供抵御 C&C 域名删除的更大的弹性。

- 权限提升：权限提升是一种被恶意软件和漏洞使用的技术，用于提升访问某些系统资源的权限，否则非管理员权限无法访问这些资源。
- 数据外带：数据外带是一种机制，恶意软件或者攻击者通过这种机制从失陷主机窃取敏感数据并将其从失陷主机回传给攻击者。

总结

本章介绍了恶意软件、各种不同类型的恶意软件及其组件，以及恶意软件完整感染周期的不同阶段。最后，本章介绍了反恶意软件行业中旨在遏制恶意软件的各种团队和检测解决方案。