国家自然科学基金项目（60973119）

对等网络蠕虫技术

DUIDENGWANGLUORUCHONGJISHU

周世杰　罗嘉庆 著

科学出版社

国家自然科学基金项目(60973119)

对等网络蠕虫技术

周世杰　罗嘉庆　著

科学出版社

北　京

前　言

近几年，计算机网络技术得到快速发展，各种网络应用大量涌现。对等（Peer to Peer，P2P）技术改变了传统的服务器－客户端（Sever Client，C/S）模式，允许用户自由平等地交换和分享数据。但是，随着用户数量的不断增加，P2P 蠕虫和病毒等恶意代码开始在 P2P 网络中肆意传播，给网络安全带来了新的挑战。由于 P2P 网络规模庞大，结构复杂，因此对网络仿真系统也提出了新的要求。

本书讲解了蠕虫的基本概念和分类，探讨了潜在的 P2P 蠕虫，介绍了一种适用于 P2P 网络的仿真平台。本书共分为 5 章。第 1 章是蠕虫的基本概念和分类方法。蠕虫（也指计算机蠕虫或网络蠕虫）是由人为或非人为因素在网络上传播的一种恶意代码。蠕虫可以根据攻击目标、反检测手段、人工干预程度等多个方面进行分类；第 2 章讨论了一种潜在 BitTorrent 蠕虫的设计、分析和防御。BitTorrent 协议是最常见的 P2P 文件分享协议之一。恶意代码编写者可以利用 BitTorrent 系统中的服务器动态获取在线的攻击目标，具有很强的隐蔽性，是一种潜在的网络安全威胁；第 3 章讲述了一种基于反应式良性蠕虫的防御方法。良性蠕虫采用与恶意蠕虫相同的传播策略，但并不攻击目标主机，而是帮助用户安装补丁或者查杀恶意代码；第 4 章介绍了一种基于缓冲区溢出的 P2P 蠕虫的设计与实现。缓冲区溢出攻击是利用缓冲区溢出漏洞所进行的攻击行动。这种漏洞是一种非常普遍、非常危险的漏洞，在各种操作系统、应用软件中广泛存在，可用于 P2P 蠕虫的设计；第 5 章介绍了一种分布式 P2P 仿真系统的设计与实现。P2P 仿真是一种专门针对 P2P 研究的网络仿真，它通过采用计算机硬件和软件，以及相应的网络设备，构建一套模拟真实 P2P 网络的仿真环境，并在此基础上模拟某种 P2P 协议的各种功能，通过分析 P2P 仿真运行过程和系统行为特征，获取相应的性能数据，从而为 P2P 研究提供数据支持。本书第 1 章主要由国防科技大学唐勇博士编写完成。

由于编者水平有限，书中难免存在一些错误和缺陷，敬请广大读者批评指正。

周世杰

2012 年 5 月

目　录

第 1 章　P2P 蠕虫的概念、特征和防御机制

蠕虫是当今因特网中普遍存在的一种网络病毒，每年在全球范围内会造成数十亿的损失。蠕虫危害操作系统，窃取敏感信息，删除用户资料，使网络拥塞，并且利用受感染的主机发起其他的网络攻击。虽然对蠕虫已有大量的研究，但如何防御蠕虫攻击仍然是一个难题。随着网络应用的发展，蠕虫可以利用各种方式快速传播，其传播速度比人类手动响应的速度要快得多，并且蠕虫的传播具有多态和变形的特点，因此变得越来越隐蔽。

本章总结了关于蠕虫的各种概念，并将蠕虫分为四类：Internet 蠕虫、P2P 蠕虫、Email 蠕虫和 IM 蠕虫。首先，根据蠕虫的目标扫描策略、传播方式和反检测能力来确定 Internet 蠕虫的特点；然后，探讨当前流行的蠕虫检测和防范方案。另外，本章还阐述了 P2P 蠕虫、Email 蠕虫和 IM 蠕虫的防御方法以及相关的研究工作。同时列举了蠕虫未来的一些研究方向。

1.1　蠕虫的概念和分类

1.1.1　蠕虫的概念

Kienzle 和 Elder 提出了蠕虫的广泛定义："蠕虫（指计算机蠕虫或网络蠕虫）是在人为或非人为帮助下在网络上传播的一种恶意代码（单独执行或者感染文件）"。本章中不具体区分计算机蠕虫病毒和网络蠕虫。

相比蠕虫而言，恶意软件是一个比较笼统的概念。恶意软件是在没有得到所有者同意的情况下，以渗透或破坏计算机系统为目的而设计的软件。恶意软件包括病毒、蠕虫、木马、僵尸网络、间谍软件、虚假广告等。各种恶意软件主要根据他们的功能来定义，但它们彼此之间的界限并不清晰。蠕虫区别于其他（如病毒类）恶意软件的地方，主要集中表现在蠕虫积极利用网络接口来传播。现在越来越多的蠕虫为其他恶意软件扮演载体的角色。如图 1-1 所示，许多恶意软件（如木马和后门等）可以被装载在蠕虫上。例如，Agobot 蠕虫的一些变种是蠕虫、后门和可控程序的组合。因为他们能在因特网上传播，所以他们是蠕虫；因

为攻击者能绕过安全机制并访问计算机资源，所以他们是后门；因为他们有指挥和控制的结构，所以他们也是可控程序。

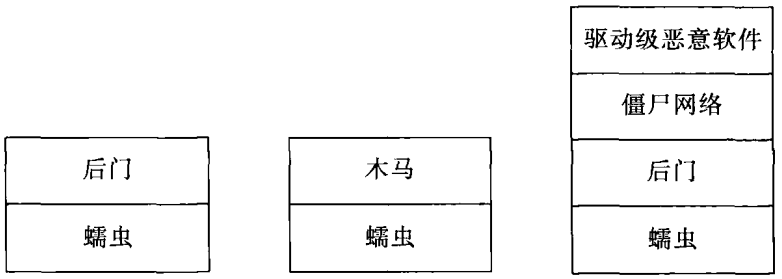

图 1-1　基于蠕虫的恶意代码

1.1.2　蠕虫分类

通过对蠕虫进行大量的研究并比较现有各种蠕虫的定义，根据扫描目标可将蠕虫分为以下四个类别。

（1）Internet 蠕虫：Internet 蠕虫是在 IP 地址空间上扫描发现目标的蠕虫，这些蠕虫在互联网上利用计算机的安全漏洞进行自我复制传播。

（2）P2P 蠕虫：P2P 蠕虫是在 P2P 网络空间中寻找目标的蠕虫。

（3）Email 蠕虫：Email 蠕虫是在 Email 地址空间中寻找目标的蠕虫，通过发送受感染的 Email 信息进行自我传播复制。

（4）IM 蠕虫：IM 蠕虫是在即时通讯客户端用户 ID 空间中寻找目标的蠕虫。

需要注意的是，上述蠕虫分类并不严格，一些蠕虫同时出现在两个或多个类别当中。例如，Nimda 蠕虫是 Internet 蠕虫，同时也是 Email 蠕虫；Bibrog 蠕虫是 P2P 蠕虫，也是 Email 蠕虫。

1.1.3　蠕虫的人工干预度

相关研究表明，一个典型蠕虫病毒的传播通常包含以下三个阶段：发现目标、蠕虫传输和感染目标。在第一阶段，蠕虫在不同的空间（如 IP 地址空间）决定下一个受害者，找到目标后，蠕虫进入传输阶段转移自身到目标上。在感染阶段，蠕虫将执行在蠕虫传输阶段已转移到目标的代码。

在第一阶段，用寻找地址空间的目标作为蠕虫分类的标准。在后两个阶段，蠕虫的传播和感染可以依靠人工或者自动模式进行，可以根据人工干预的程度将

蠕虫分为四类：人工传输和人工感染、人工传输和自动感染、自动传输和人工感染、自动传输和自动感染。

四类蠕虫的人工干预度如图 1-2 所示。通常，Internet 蠕虫通过远程调用的方式使蠕虫代码在没有人工行动的状态下进行传播和执行，进而实现自动传输和自动感染。IM 与 Email 蠕虫进行自动传输和人工感染（如通过邮件或即时通信软件发送蠕虫代码，但这些代码需要人工行动才能被执行）或者人工传输和自动感染（如用户访问钓鱼网站等手动过程后，恶意代码由于各种漏洞而自动执行的自动过程）。P2P 蠕虫有多种不同的传播方式，覆盖了全部四类人工干预度。

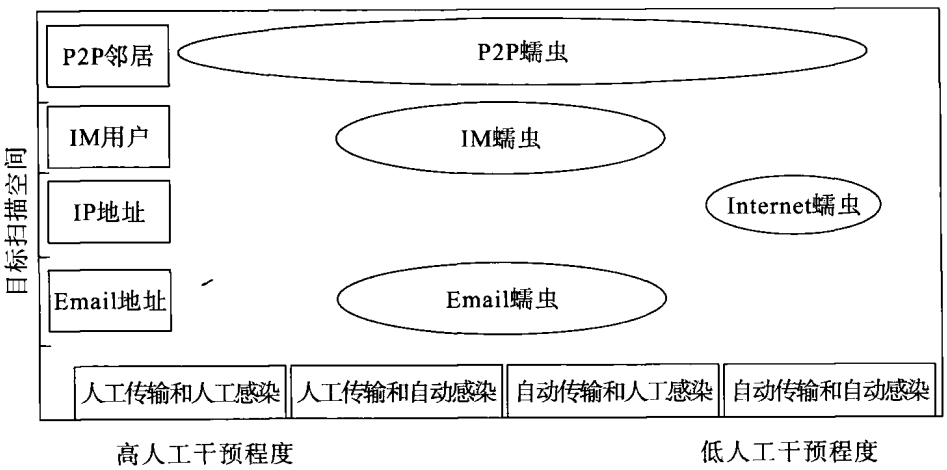

图 1-2 四类蠕虫的人为干预程度

1.2 Internet 蠕虫

1.2.1 Internet 蠕虫的特征

本节基于三个因素来描述 Internet 蠕虫的特点：目标发现策略、传播方式和反检测技术。如图 1-3 所示，目标扫描策略表示 Internet 蠕虫如何发现新目标进行感染，传播方式表示 Internet 蠕虫如何感染和复制自身到新的目标，反检测技术被蠕虫用来逃避检测。

1.2.1.1 目标扫描策略

Internet 蠕虫在感染计算机之前必须在 IP 地址空间中找到目标。多数的目标扫描策略可以被归纳为三个类别：盲目扫描、被动扫描和攻击列表扫描。需要注

意的是，蠕虫可以同时利用多个目标扫描策略来发现新的目标。

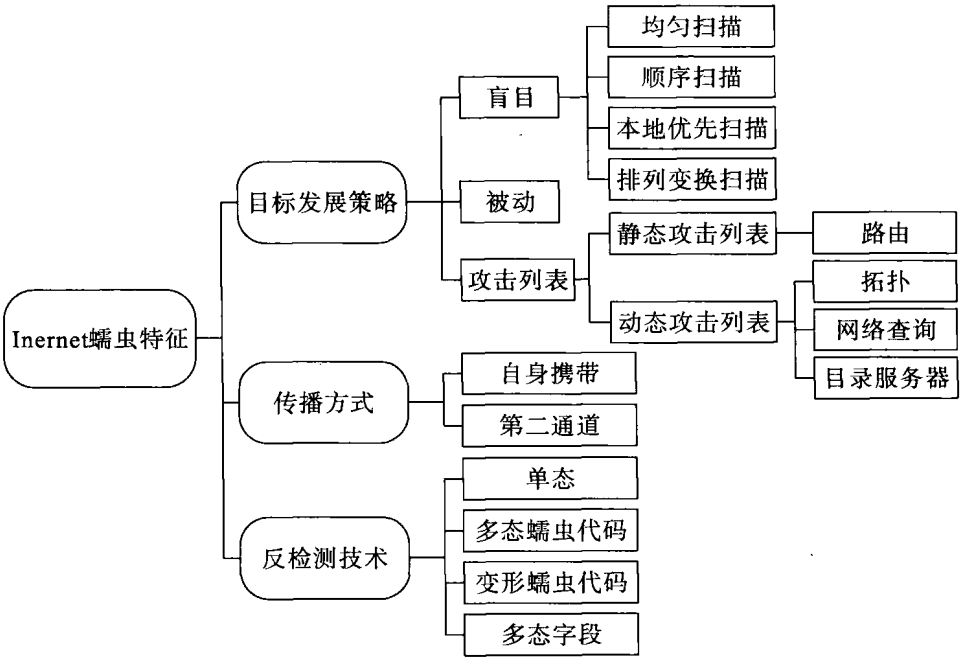

图 1-3　Internet 蠕虫的特征

1. 盲目扫描

盲目扫描是一种很简单的扫描方式，是指蠕虫没有通过对目标的先期了解而盲目地扫描整个 Internet 的 IPv4 和 IPv6 地址空间。

（1）均匀扫描。蠕虫（如 Code Red 和 Slammer 蠕虫）均匀并随机地在整个 IP 地址空间中选取一个 IP 地址作为目标。均匀扫描是最简单的扫描策略。

（2）顺序扫描。蠕虫（如 Blaster 蠕虫）随机地选取一个起始 IP 地址后，顺序地扫描 IP 地址。

（3）本地优先扫描。蠕虫（如 Code Red II）更喜欢在同一子网内扫描 IP 地址，因为 Internet 的 IP 地址空间的分配并不均匀，所以本地优先扫描蠕虫比均匀扫描、顺序扫描蠕虫的速度更快。

（4）排列变换扫描。所有的蠕虫置换 IP 地址空间，形成一个共同虚拟的 IP 地址空间，并在这个虚拟的 IP 地址空间中用不同的起始 IP 地址进行顺序扫描。排列变换扫描蠕虫因为最大限度减少了重复扫描，所以传播速度更快。

一般情况下，盲目扫描蠕虫更容易实现，但有以下几项缺陷：第一，IP 扫描的失误率非常高（这点可以被用来做蠕虫检测）；第二，它们相比其他目标扫

描策略来说不够快速；第三，它们在 IPv6 网络或者用网络地址转换（NAT）的网络中效率较低。模拟结果表明，蠕虫要在一个 IPv6 网中感染一半数量易被攻击的主机需要很多年的时间。

2. 被动扫描

被动扫描是指蠕虫（如 CR Clean 蠕虫）不通过扫描来积极搜寻目标。相反，它等待潜在的受害者来主动连接蠕虫所在的机器，然后通过与他们的交互来感染主机。虽然它们的速度非常慢，但使用被动目标扫描策略的蠕虫防不胜防，因为他们在目标扫描期间不产生任何异常流量。

3. 攻击列表扫描

攻击列表扫描是指通过在攻击之前确定目标主机中已知地址的列表来加快蠕虫的传播速度。攻击列表蠕虫的优势在于能够更加快速且隐蔽地传播，因为连接的失败率相对很低。

蠕虫的攻击列表可以静态或者动态地被创建。

（1）蠕虫被释放之前，静态攻击列表已经被创建，并且这个列表在每一个蠕虫实例中被携带。例如，路由蠕虫携带了 BGP 路由表的副本作为攻击列表。路由蠕虫不仅传播速度非常快（大约比传统蠕虫快三倍），而且还可以利用 BGP 路由前缀的地理信息进行精确的“选择性攻击”（如攻击特定国家）。静态攻击列表扫描的弱点是攻击列表可能过期，并且攻击列表规模越大就越难以被蠕虫携带。

（2）动态攻击列表是在每个被感染的机器中被动态创建的。有几种方法来创建动态攻击列表。例如，拓扑蠕虫搜索本地通信的拓扑信息（如在/etc/hosts 文件中存储的主机等），在被感染的机器中发现新的目标；网页搜索蠕虫使用 google 等搜索引擎来搜索易受攻击的目标。

1.2.1.2 传播方式

大多数 Internet 蠕虫通过软件漏洞进行传播，其传播方式通常有两种：自身携带和第二通道。

（1）自身携带。蠕虫的感染攻击由漏洞溢出和有效载荷组成。有效载荷是嵌入或者附加到该漏洞的恶意代码。对自身携带的蠕虫来说，有效载荷直接就是蠕虫代码或者编码后的蠕虫代码。Code Red II 蠕虫的感染攻击如图 1-4 所示，它是一个典型的自身携带式蠕虫。

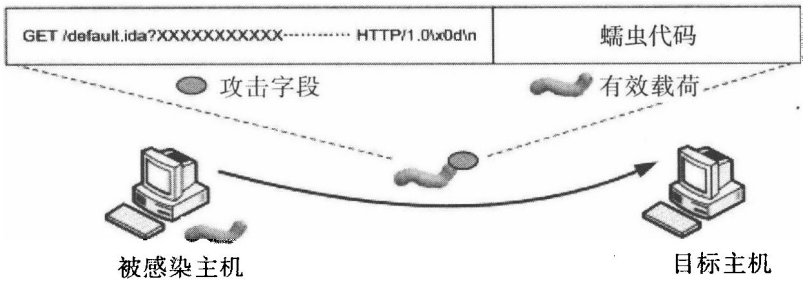

图 1-4　自身携带式蠕虫

(2) 第二通道。一些蠕虫通过二次通信信道传输自身。换句话说，在目标感染过程中，这些蠕虫只传输小块恶意代码（如 Shellcode）；然后蠕虫的完整代码从被感染的机器上、公共服务器上或者通过僵尸网络被下载和更新。

1.2.1.3　反检测技术

蠕虫会使用各种反检测技术来逃避检测系统的侦查。

(1) 单态。单态蠕虫不改变自己的蠕虫代码，并且对目标始终发送相同的感染攻击。单态蠕虫可以通过特征机制被很容易地检测出来。

(2) 多态蠕虫代码。蠕虫代码的多态通过加密改变了它的二进制代码，同时保证原有蠕虫代码功能不变。虽然通过变异加密方式蠕虫可形成数以百万计的形式，但解密后的蠕虫本体是不变的。因此，我们能够得到解密后的代码，然后通过代码模拟器来检测。Tapion 是一个多态性的工具的例子，Agobot 是一个蠕虫代码多态性的例子。

(3) 变形蠕虫代码。通过转化格式或改变代码创造出新一代的蠕虫。一个理想的变形蠕虫是通过任何特征机制都检测不出来的。常见的变形技术包括子程序置换、插入垃圾代码/跳转指令和代码替换等。ADMmutate，Clet 是两个著名的变形工具。对蠕虫编写者来说，编写一个完美的变形蠕虫病毒，并可以智能地改变它的行为甚至演化自身，是非常具有吸引力的。

(4) 多态字段。蠕虫感染攻击包括攻击字段和有效载荷。有效载荷能够通过多态的或变形的蠕虫代码被动态地改变。攻击字段通过改变一些不重要的字节（这些被称为通配符字节）来形成多态性，也要保持一些字节的完整（这些成功感染的关键部分被称作不变字节）。如图 1-5 所示的一个多态字段，它是从 Code Red II 的非多态字段改编而来的。

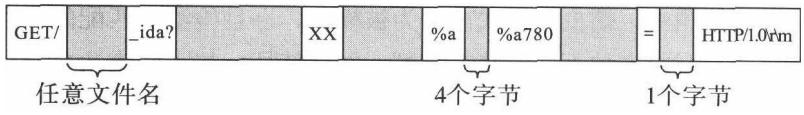

图 1-5 多态字段的 Code Red II（白色内容为不变字节，灰色内容为可变字节）

通常情况下，一个蠕虫如果利用多态、变形蠕虫代码或多态字段的技术，我们称之为一个多态蠕虫。一个完美的多态蠕虫不包含任何构件（如不变的字节等），因此不能被基于特征的系统检测到。幸运的是，现今的大部分蠕虫（如 Nimda，Slammer，Sasser 和 Witty 蠕虫等）仍然是单态蠕虫，并没有出现完美的多态蠕虫的记录。但是对 Internet 来说，多态蠕虫正逐渐成为一个潜在的威胁。

1.2.2 Internet 蠕虫检测

Internet 蠕虫防御包含两个步骤：蠕虫检测和蠕虫遏制。蠕虫检测是为了查找 Internet 蠕虫的各种活动。如图 1-6 所示，目前蠕虫检测技术大致分为基于特征和基于异常两个方案。自动特征生成是连接这两个检测方案的一项新技术。蠕虫遏制是指在蠕虫被检测到后进行快速反应并将损害降到最低。

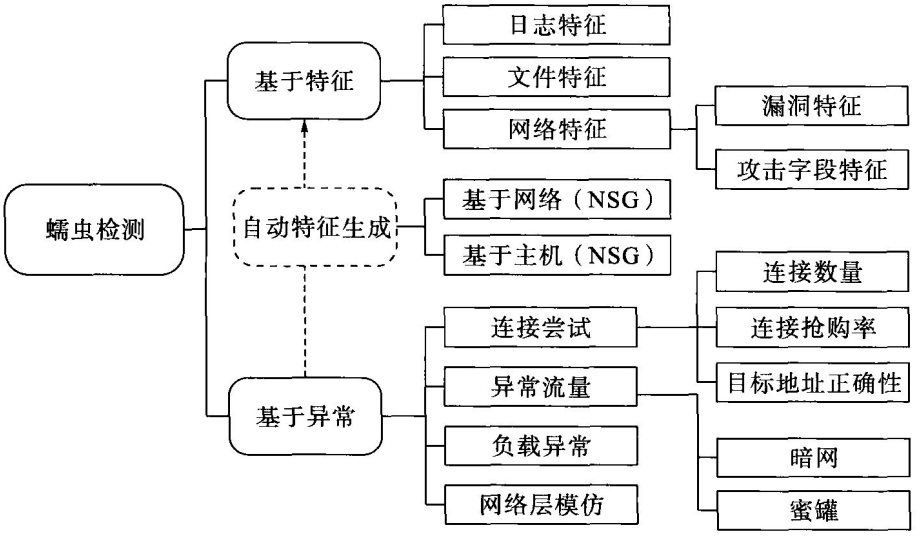

图 1-6 蠕虫检测方法

1.2.2.1 基于特征的蠕虫检测

基于特征检测是一种传统的方法，并广泛地被应用于入侵检测系统（IDS）。

在基于特征的检测中，蠕虫的模式或行为被模仿，一旦检测到匹配的状态，检测系统就会启动。有很多种不同的特征类型：网络特征，如正则表达式，旨在匹配蠕虫的每次感染攻击（网络流量）；日志特征，存在于系统和应用程序的日志中，能够揭发受害主机中蠕虫的行为；文件特征，表现文件系统中蠕虫的轨道。在本章中，侧重研究网络特征，因为日志特征和文件特征在传统意义上属于病毒检测的研究范围。本质上来说，网络特征有两个子类型：基于溢出的特征和基于漏洞的特征。

（1）基于溢出的特征。基于溢出的特征描述一个或几个溢出的特征。一个好的基于溢出的特征能够利用相同的漏洞检测蠕虫感染，即使这些感染是多态性的。基于溢出的特征只能够检测已知的溢出/蠕虫，到目前为止，它们仍然是不可替代的，尤其是对于利用“Zero Day”溢出漏洞的“Zero Day”蠕虫方面的提前和适时的检测。

（2）基于漏洞的特征。一个基于漏洞的特征描述一个特定漏洞的特性，因此可以检测所有可能利用此漏洞的攻击。基于漏洞的特征比基于溢出的特征更为有效，如果一个漏洞可以被很多种攻击利用，那么可以据此检测未知的攻击。然而，基于漏洞特征的产生是非常复杂和耗时的。

大多数反病毒软件厂商提供的入侵检测系统有两种特征，不仅能说明什么类型的漏洞已被利用，同时还能说明什么类型的溢出已经被用到蠕虫上。

1.2.2.2　基于异常的蠕虫检测

基于异常的检测系统建立正常网络或程序行为的模型，当一个主机或程序的行为违反了这些模型，将会产生一个警报。如图 1-6 所示，基于异常的检测方法大致基于连接尝试、异常流量、负载异常和网络层模仿。

1. 连接尝试

为了加快传播速度，蠕虫在很短的时间内发送大量的 TCP SYN 数据包或者 UDP 包来寻找受害者。因此，可以使用以下简单的策略检测蠕虫。

（1）连接尝试数量。如果在一段时间内从一个特定主机发送 SYN 包数量超过了一个阈值，就认为主机被感染了。

（2）连接失败数/率。在一个很短的时间内，如果一个特定主机收到了大量的连接失败包（即 TCP RST 包，或 ICMP 主机不可到达的包）或者成功和失败连接的比率高，那么这个主机就被认为已感染。这种方法对盲目扫描蠕虫是有效的，但是针对攻击列表、拓扑和被动扫描蠕虫效果不太好。

（3）目标与源头相关性。蠕虫在有漏洞的主机中快速传播。也就是说，如果一台主机被一种方式（如数据包指定一个特定的端口或包含类似的内容）所感染，很快，它将以同样的方式试图感染其他主机。因此，相关的输入输出包能够发现蠕虫的运动。

2. 非法通信

有一大部分地址在因特网上没有被使用。正常的机器很少往这些未使用的地址上发送包，然而蠕虫却会。因此，可以基于异常流量检测蠕虫。

（1）暗网。一个暗网（也被称为黑洞网络）是指已选择路由和已分配地址空间的一部分，在这个地址空间中，没有活跃的服务或者服务器驻留。这些地址“黑暗”是因为这些网络看起来几乎没有流量。黑暗网络中的任何流量都可能来自蠕虫。例如，Cymru 小组维护着一个黑暗络路项目，Vinod Yegneswaran 等描述了一个黑暗网络，这个系统以一个高效、可伸展的并且可量的模式在未使用的 IP 地址上衡量包裹流量。

（2）蜜罐。蜜罐是指网络上易受攻击的系统，它没有提供任何真实/产品的服务。与黑暗网络相似，到蜜罐的任何流量都是可疑的。两者间的区别是蜜罐对访问者会做出反应，而黑暗网络却没有。如图 1-7 所示，有三种蜜罐实现类型：物理蜜罐（通过物理机制和真实的系统做出反应），仿真蜜罐（通过仿真操作系统和服务做出反应）和虚拟蜜罐（通过运行在虚拟机上的真实的操作系统和网络服务做出反应）。由于蜜罐可以以一个更高的互动性与访问者进行通信，它们可以准确地检测出各种类型的蠕虫并且捕获蠕虫样本。

3. 载荷异常

负载异常检测首先应建造正当负载的模型，然后通过检查一个包裹/流的负载是否满足正常的模型来检测异常的网络行为。例如，PHAD 是一个建造包头部域的模型系统，Anagram 和 PAYL 是一个建造合法包内容模型的系统。通常，由于网络应用的多样性，建造所有可能的合法负载的坚固模型是很困难的。但是，对于特定的服务或者网站却是可能的。异常负载检测方法是很有挑战性的，因为为了避免被检测到，蠕虫也许会嵌入到合法的流量负载中。

蠕虫的负载企图在受害主机上执行一段程序。最近，网络层模仿被提出，用来从多态蠕虫中准确识别这种程序形式的负载。

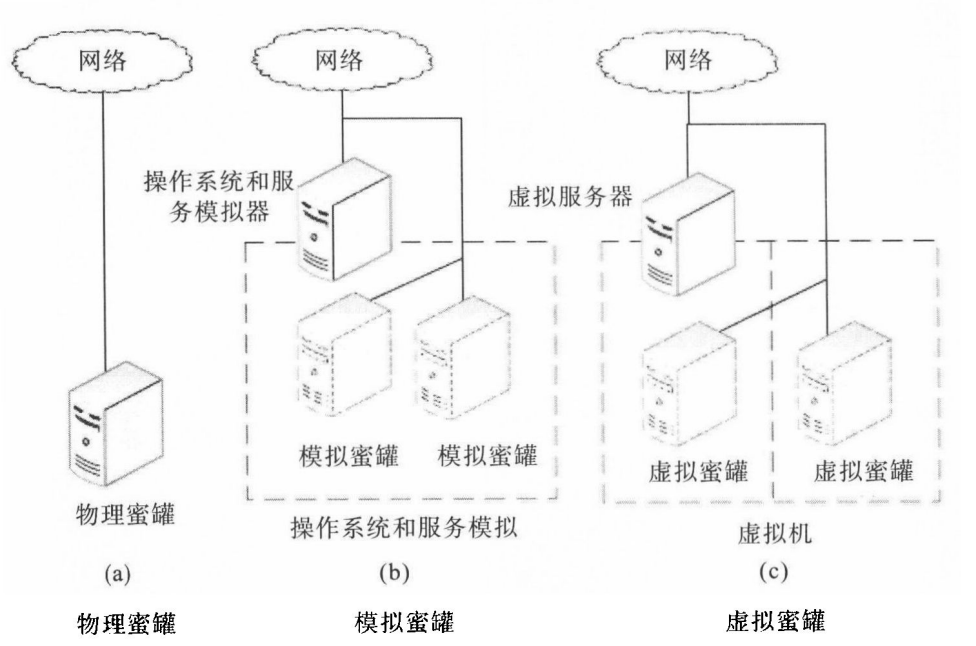

图 1-7　蜜罐实现的三种类型

1.2.2.3　自动特征生成

通常，基于特征的探测系统具有准确、高效、容易开发和利用的特点，但不能检测未知的蠕虫。相反，基于异常的检测系统能够检测未知的蠕虫，但是由于建模正常的行为非常困难，因此它们通常会有很高的错误报警（误报）率。自动特征的生成可以把基于特征的检测和基于异常的检测的优点结合起来。我们可以使用基于异常的检测系统发现未知的攻击（蠕虫），然后使用自动特征的生成技术生成准确的特征用来检测。最近几年，自动特征生成已经成为了一个活跃的学科并且建立了很多系统。根据输入信息，这些系统可以被概括为基于主机的或者是基于网络的。

（1）基于主机的特征的生成（HSG）：HSG 系统在受保护的主机上运行利用主机信息检测出感染的企图并从这些企图中生成特征。一般而言，基于主机的方法可以快速地生成准确的特征，但是就性能和结构来说，通常会在受保护的主机上有负面的影响。例如，主机需要重新编译内核或者修改运行库。

（2）基于网络的特征的生成（NSG）：NSG 系统单独分析可疑网络流量和输出基于内容的特征。与 HSG 系统相比，NSG 系统在蠕虫传播的早期比较敏感，因为它们工作在网络路由器/网关层次，所以可以较早地捕获到蠕虫的样本。早期基于网络的特征生成方法包括 Honeycmb、PAYL 和 EarlyBird，生成只有一串