软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2013,24(4):825−842 [doi: 10.3724/SP.J.1001.2013.04369]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

蜜罐技术研究与应用进展*

诸葛建伟[1,2], 唐 勇[3], 韩心慧[4], 段海新[1,2]

[1](清华信息科学与技术国家实验室(清华大学),北京 100084)

[2](清华大学 网络科学与网络空间研究院,北京 100084)

[3](国防科学技术大学 计算机学院,湖南 长沙 410073)

[4](北京大学 计算机科学技术研究所,北京 100871)

通讯作者: 诸葛建伟, E-mail: zhugejw@cernet.edu.cn, http://netsec.ccert.edu.cn/zhugejw

摘 要: 蜜罐是防御方为了改变网络攻防博弈不对称局面而引入的一种主动防御技术,通过部署没有业务用途的安全资源,诱骗攻击者对其进行非法使用,从而对攻击行为进行捕获和分析,了解攻击工具与方法,推测攻击意图和动机.蜜罐技术赢得了安全社区的持续关注,得到了长足发展与广泛应用,并已成为互联网安全威胁监测与分析的一种主要技术手段.介绍了蜜罐技术的起源与发展演化过程,全面分析了蜜罐技术关键机制的研究现状,回顾了蜜罐部署结构的发展过程,并归纳总结了蜜罐技术在互联网安全威胁监测、分析与防范等方向上的最新应用成果.最后,对蜜罐技术存在的问题、发展趋势与进一步研究方向进行了讨论.

关键词: 网络安全;蜜罐;蜜网;蜜场;威胁监测;恶意代码

中图法分类号: TP309 **文献标识码**: A

中文引用格式: 诸葛建伟,唐勇,韩心慧,段海新.蜜罐技术研究与应用进展.软件学报,2013,24(4):825−842. http://www.jos.org.cn/1000-9825/4369.htm

英文引用格式: Zhuge JW, Tang Y, Han XH, Duan HX. Honeypot technology research and application. Ruanjian Xuebao/Journal of Software, 2013,24(4):825−842 (in Chinese). http://www.jos.org.cn/1000-9825/4369.htm

Honeypot Technology Research and Application

ZHUGE Jian-Wei[1,2], TANG Yong[3], HAN Xin-Hui[4], DUAN Hai-Xin[1,2]

[1](Tsinghua National Laboratory for Information Science and Technology (Tsinghua University), Beijing 100084, China)

[2](Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing 100084, China)

[3](School of Computer Science, National University of Defense Technology, Changsha 410073, China)

[4](Institute of Computer Science and Technology, Peking University, Beijing 100871, China)

Corresponding author: ZHUGE Jian-Wei, E-mail: zhugejw@cernet.edu.cn, http://netsec.ccert.edu.cn/zhugejw

Abstract: Honeypot is a proactive defense technology, introduced by the defense side to change the asymmetric situation of a network attack and defensive game. Through the deployment of the honeypots, i.e. security resources without any production purpose, the defenders can deceive attackers to illegally take advantage of the honeypots and capture and analyze the attack behaviors to understand the attack tools and methods, and to learn the intentions and motivations. Honeypot technology has won the sustained attention of the security community to make considerable progress and get wide application, and has become one of the main technical means of the Internet security threat monitoring and analysis. In this paper, the origin and evolution process of the honeypot technology are presented first. Next, the key mechanisms of honeypot technology are comprehensively analyzed, the development process of the honeypot deployment structure is also reviewed, and the latest applications of honeypot technology in the directions of Internet security threat

* 基金项目: 国家自然科学基金(61003127, 61003303); 国家重点基础研究发展计划(973)(2009CB320505); 国家 242 信息安全计划(2011A40)

收稿时间: 2012-02-12; 定稿时间: 2012-12-27

monitoring, analysis and prevention are summarized. Finally, the problems of honeypot technology, development trends and further research directions are discussed.

Key words: network security; honeypot; honeynet; honeyfarm; threat measurement; malware

互联网从诞生以来,一直遭受着网络攻击与恶意代码的威胁.随着攻击技术的不断发展,新形态的安全威胁不断涌现并在持续进化,而防御技术并不能及时跟上安全威胁的变化步伐,这使得互联网的安全状况日益恶化.究其根源,会发现攻击方与防御方之间在进行着一场不对称的技术博弈:攻击方可以在夜深人静时只要找到攻击目标的一个漏洞就能够攻破系统,而防御方必须确保系统不存在任何可被攻击者利用的漏洞,并拥有全天候的监控机制,才能确保系统的安全;攻击方可以利用扫描、查点等一系列技术手段,全面获取攻击目标的信息,而防御方即使在被攻陷后仍然很难了解到攻击的来源、方法和动机;一旦博弈失败,由于安全响应技术与协调机制的欠缺,在很多情况下,攻击方不会遭受任何损失,而防御方却通常将面临系统与信息被破坏或窃取的风险.

蜜罐(honeypot)就是防御方为了扭转这种不对称局面而提出的一项主动防御技术.蜜罐定义为一类安全资源,它没有任何业务上的用途,其价值就是吸引攻击方对它进行非法使用[1].蜜罐技术本质上是一种对攻击方进行欺骗的技术,通过布置一些作为诱饵的主机、网络服务或者信息,诱使攻击方对它们实施攻击,从而可以对攻击行为进行捕获和分析,了解攻击方所使用的工具与方法,推测攻击意图和动机,能够让防御方清晰地了解他们所面对的安全威胁,并通过技术和管理手段来增强实际系统的安全防护能力.

从 20 世纪 80 年代末蜜罐技术在网络安全管理实践活动中诞生以来[2],就赢得了安全社区的持续关注,在 The Honeynet Project 等开源团队的推动下,得到了长足发展与广泛应用:针对不同类型的网络安全威胁形态,出现了丰富多样的蜜罐软件工具;为适应更大范围的安全威胁监测的需求,逐步从中发展出蜜网(honeynet)[3]、分布式蜜罐(distributed honeypot)、分布式蜜网(distributed honeynet)和蜜场(honeyfarm)[4]等技术概念;在安全威胁监测研究与实际网络安全管理实践中,大量应用于网络入侵与恶意代码检测、恶意代码样本捕获、攻击特征提取、取证分析和僵尸网络追踪等多种用途.

从公开发表的科研论文和资料来看,国内对蜜罐技术的关注与相关研究还不够全面与深入.中国科学院高能物理研究所网络安全实验室[5,6]、电子科技大学[7]、安全焦点团队等在 2002 年~2004 年间较早开展对蜜罐技术的研究与实践,北京大学诸葛建伟等人发起了蜜网研究项目组,加入 The Honeynet Project,成为其中国分支团队,并连续承担了多个国家科研项目,支持国家计算机网络应急技术处理协调中心进行 Matrix 分布式蜜网系统的开发与实验部署[8],在恶意代码样本捕获[9]、僵尸网络追踪[10]等方面取得了较好的应用效果.国内学者也已对蜜罐及蜜网技术的研究进展进行了简要综述[11],但尚不能清晰地描绘出蜜罐技术的研究发展轨迹,也未全面地展现出蜜罐技术领域的研究、开发与应用现状.因此,本文尝试对蜜罐技术的起源与发展演化过程、蜜罐技术关键机制研究现状、蜜罐部署结构的发展、蜜罐技术应用情况和发展趋势进行全面的总结与深入的分析,为国内安全社区熟悉了解蜜罐技术并进一步开展相关研究与实践应用工作提供参考.

本文第 1 节对蜜罐技术的起源与发展过程进行介绍.第 2 节剖析蜜罐技术的核心需求与关键机制,并针对欺骗环境构建、威胁数据捕获、威胁数据分析和反蜜罐技术对抗这 4 个关键机制的研究进展进行总结与分析.第 3 节回顾蜜罐部署结构的发展过程.第 4 节展示近年来蜜罐技术在互联网安全威胁监测、分析与防范等领域中的应用情况.第 5 节讨论目前蜜罐技术存在的问题,并展望其发展趋势和可能的进一步研究方向.最后,在第 6 节中给出本文结论.

1 蜜罐技术的起源与发展

1.1 蜜罐技术概念的起源与发展

蜜罐技术的起源与发展时间线如图 1 所示.蜜罐技术概念最早出现在于 1989 年出版的《The Cuckoo’s Egg》著作[2]中,这本小说描述了作者作为一个公司的网络管理员,如何利用蜜罐技术来发现并追踪一起商业间谍案的故事.直到 20 世纪 90 年代末,蜜罐还仅限于一种主动性防御思路,由网络管理员们所采用,通过欺骗攻击者达

到追踪的目的.从 1998 年开始,蜜罐技术逐渐吸引了一些安全研究人员的注意,他们开发出一些专门用于欺骗攻击者的蜜罐软件工具,最为知名的是由著名计算机安全专家 Cohen 所开发的 DTK(deception tool kit)[12],Cohen 还深入总结了自然界存在的欺骗实例、人类战争中的欺骗技巧和案例以及欺骗的认知学基础,分析了欺骗的本质,并在理论层次上给出了信息对抗领域中欺骗技术的框架和模型[13],Cohen 的这一研究工作为蜜罐技术概念的发展奠定了理论基础.在此之后,蜜罐技术得到安全社区的广泛关注,出现了大量开源蜜罐工具,如 Honeyd[14],Nepenthes[15]等,以及一些商业蜜罐产品,如 KFSensor,Symantec Decoy Server 等.

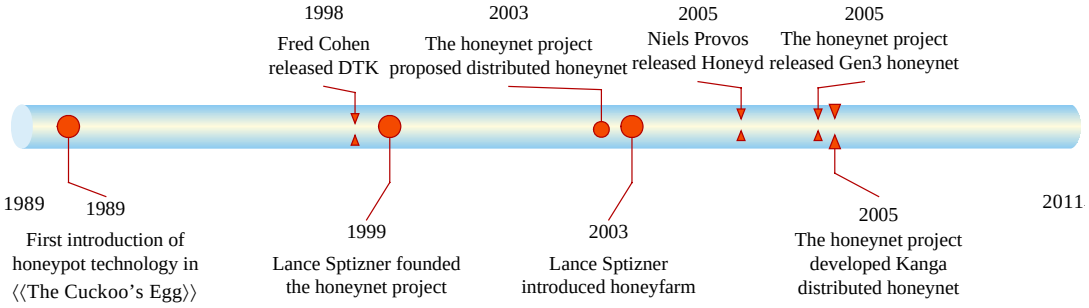

Fig.1 Timeline of development of honeypot concepts

图 1 蜜罐技术概念的发展时间线

早期的蜜罐一般伪装成存有漏洞的网络服务,对攻击连接做出响应,从而对攻击方进行欺骗,增加攻击代价并对其进行监控.由于这种虚拟蜜罐存在着交互程度低、捕获攻击信息有限且类型单一、较容易被攻击者识别等问题,Spitzner 等安全研究人员提出并倡导蜜网(honeynet)技术,并在 1999 年成立了非赢利性研究组织 The Honeynet Project.蜜网是由多个蜜罐系统加上防火墙、入侵防御、系统行为记录、自动报警与数据分析等辅助机制所组成的网络体系结构[3],在蜜网体系结构中可以使用真实系统作为蜜罐,为攻击者提供更加充分的交互环境,也更难被攻击者所识别.蜜网技术使得安全研究人员可以在高度可控的蜜罐网络中,监视所有诱捕到的攻击活动行为,从而去了解攻击方的工具、方法和动机.

为了克服传统蜜罐技术与生俱来的监测范围受限的弱点,The Honeynet Project 在 2003 年开始引入分布式蜜罐(distributed honeypot)与分布式蜜网(distributed honeynet)的技术概念,并于 2005 年开发完成 Kanga 分布式蜜网系统,能够将各个分支团队部署蜜网的捕获数据进行汇总分析.分布式蜜罐/蜜网能够通过支持在互联网不同位置上进行蜜罐系统的多点部署,有效地提升安全威胁监测的覆盖面,克服了传统蜜罐监测范围窄的缺陷,因而成为目前安全业界采用蜜罐技术构建互联网安全威胁监测体系的普遍部署模式,具有较大影响力的包括 The Honeynet Project 的 Kanga 及其后继 GDH 系统[16]、巴西分布式蜜罐系统[17]、欧洲电信的 Leurre.Com[18]与 SGNET[19]系统、中国 Matrix 分布式蜜罐系统[8]等.

在互联网和业务网络中以分布式方式大量部署蜜罐系统,特别是在包含提供充分交互环境的高交互式蜜罐时,需要部署方投入大量的硬件设备与 IP 地址资源,并需要较多的维护人力成本.2003 年,Spitzner 提出了一种蜜罐系统部署的新型模式——蜜场(honeyfarm)[4].基于蜜场技术概念实现的网络威胁预警与分析系统有 Collapsar[20],Potemkin[21]和 Icarus[22]等.

1.2 蜜罐工具软件的发展过程

蜜罐工具软件通过构建欺骗环境来捕获安全威胁数据,是蜜罐技术的核心载体.随着互联网安全威胁类型的不断更新与演化,安全社区也在有针对性地研究蜜罐技术并开发出相应的蜜罐工具软件,从而适应对新形态安全威胁的监测需求.

互联网上传统的网络攻击与恶意代码主要利用网络服务中存在的安全漏洞或配置弱点,对目标信息系统与网络构成威胁,因此,最早出现的蜜罐工具软件也是针对网络服务攻击而设计的.最早的蜜罐工具 DTK 绑定

在系统的未使用端口上,对任何想探测这些端口的攻击源提供欺骗性网络服务[12].LaBrea 蜜罐软件接受网络上所有空闲 IP 地址的 TCP 连接,并通过 TCP 协议中的窗口调节与持久连接等技巧实现一种 Tarpit 服务,能够尽可能地拖长无效连接的持续时间,从而减缓网络扫描探测与蠕虫传播的速度[23].Honeyd[14]是著名安全专家 Provos 开发的一款虚拟蜜罐框架性开源软件,引入了在网络协议栈层次上模拟各种类型蜜罐系统的方法.Honeyd 支持在协议栈指纹特征上伪装成指定的操作系统版本,对攻击者利用 nmap 等工具实施主动指纹识别进行欺骗,同时也支持模拟构建虚拟网络拓扑结构,并以插件方式提供对各种应用层网络服务的模拟响应.利用 Honeyd 软件,安全研究人员可以很容易地按照需求定制出一个包含指定操作系统类型与应用服务的蜜罐系统,用于蠕虫检测与应对、垃圾邮件监测等多种用途.由于 Honeyd 最早引入了网络协议栈层次上的蜜罐系统模拟机制,以及采用了可集成各种应用层服务蜜罐的灵活框架性结构,使其在蜜罐工具软件发展过程中具有举足轻重的重要地位.The Honeynet Project Giraffe Chapter 开发的 Nepenthes 蜜罐软件[15]继承了 Honeyd 的网络协议栈模拟机制与框架性结构,针对互联网上主动传播恶意代码的监测需求,实现了可供大规模部署的恶意代码样本采集工具.与之前蜜罐系统尝试模拟整个网络服务交互过程不同,Nepenthes 的基本设计原则是只模拟网络服务中存在安全漏洞的部分,使用 Shellcode 启发式识别与仿真执行技术[24]来发现针对网络服务安全漏洞的渗透攻击,从中提取到主动传播恶意代码的下载链接,并进一步捕获样本.这种机制使其较其他已有蜜罐工具对自动化传播恶意代码捕获更为高效.Nepenthes 已被新一代恶意代码样本捕获蜜罐软件 Dionaea[25]所替代,Dionaea 采用内嵌 Python 脚本代码实现对漏洞服务的模拟,同样采用 Libemu[24]来检测 Shellcode,并支持 IPv6 与 TLS 协议.Dionaea 蜜罐软件是目前技术最为先进、体系结构最优化的虚拟蜜罐工具,但所支持的应用层服务与漏洞环境还不够充分,开源社区也正在逐渐添加应用服务支持,如 MySQL,VoIP 等.

除了扫描探测、渗透攻击、恶意代码传播等通用性网络服务安全威胁之外,Web,SMTP,SSH 等互联网常用服务面临着 Web 应用程序攻击、垃圾邮件与 SSH 口令暴力破解等应用层安全威胁.开源社区也专门针对目前流行的几类应用层协议攻击,开发了一系列的专用服务型蜜罐软件.GHH(Google hack honeypot)是最早针对 Web 应用攻击威胁研究并开发的 Web 应用服务蜜罐,GHH 针对搜索存有安全漏洞 Web 应用程序的 Google Hacking 技术来诱骗 Web 应用程序攻击并进行日志记录,可以发现命令注入、Web 垃圾邮件、博客垃圾评论注入、网页篡改、植入僵尸程序、搭建钓鱼站点等各种攻击事件[26].HIHAT(high interaction honeypot analysis toolkit)[27]可将任意的 PHP 应用程序自动地转换为提供充分交互环境的 Web 蜜罐工具,并通过透明链接方式获取恶意 Web 访问请求,从而对现有 PHP 应用程序所面临的威胁进行监测分析.Glastopf[28]/GlastopfNG[29]则是目前最新的 Web 应用蜜罐软件.之前的 Web 应用蜜罐技术均以现有 Web 应用程序作为模版,伪装成存有漏洞的 Web 站点并吸引攻击,这种方法的局限性是必须编写或转换新的应用程序模版才能支持新的安全漏洞,非常耗费开发者的时间,并对新型安全威胁的监测具有滞后性.而 Glastopf[28]则引入了一种新的想法:使用攻击者在尝试利用 Web 应用程序时期望得到的结果来响应攻击者,这样就可以避免基于模版的方法所存在的缺陷.在具体实现中,Glastopf 目前针对远程文件包含、本地文件包含等 Web 应用攻击类型模拟漏洞利用过程生成响应结果,从而触发攻击者进一步的恶意请求,并记录下攻击日志与恶意脚本文件.GlastopfNG[29]是对 Glastopf 进行重新设计与开发,引入框架性结构与可扩展模块机制的新一代 Web 应用蜜罐软件,支持对远程文件包含、本地文件包含、SQL 注入攻击和跨站脚本攻击等 Web 应用攻击进行监测.SPAMPot 蜜罐[30]模拟成一个开放的 SMTP 邮件服务器来吸引垃圾邮件发送者通过它来发送垃圾邮件,从而发现与分析互联网上的垃圾邮件威胁.Kojoney[31]与 Kippo[32]蜜罐则模拟为 SSH 网络服务进程,记录每次 SSH 口令暴力破解所尝试使用的用户名与口令,并在口令猜测成功之后为攻击者提供模拟的 shell 执行环境,对攻击源 IP 地址、使用的 SSH 客户端类型、输入的控制命令以及下载的攻击工具文件进行捕获与记录.

近年来,由于防火墙、入侵防御系统等网络边界防御机制的广泛应用,针对传统网络服务的渗透攻击变得越来越难以成功实现,以浏览器与插件为主要目标的客户端渗透攻击逐渐成为互联网上的主流安全威胁.而蜜罐技术也随着安全威胁热点的这一变化,演化出客户端蜜罐工具软件.Capture-HPC[33]是一个高交互式的客户端蜜罐框架,支持在 Windows 虚拟机环境中运行 IE,Firefox 等浏览器,并通过内核中的系统状态变化监控机制来

检测浏览器当前访问的网页中是否包含客户端渗透攻击代码.PhoneyC[34]则采用浏览器仿真与 Javascript 动态分析技术来对抗恶意网页脚本的混淆机制,并通过模拟各种已知浏览器与插件安全漏洞来检测出恶意网页中包含的渗透攻击类型.新版本 PhoneyC[35]还通过对 Javascript 引擎进行 opcode 指令动态插装,实现了对恶意网页中的 heapspray 堆散射攻击的检测能力.

从上述发展轨迹中我们可以看出,作为主动性的安全威胁监测与防御技术,蜜罐技术的发展紧随着互联网安全威胁热点的变化.由于目前互联网安全威胁的多样性,也使得没有一个蜜罐工具软件能够完全覆盖所有类型的安全威胁,蜜罐工具软件呈现出百花齐放的局面.搭建互联网安全监测技术解决方案,也需要综合应用各种蜜罐技术与工具软件来覆盖所关注的安全威胁类型,最佳实践策略是采用一种通用化蜜罐框架软件(如 Dionaea)作为基础,然后针对流行威胁类型,结合专用的应用服务蜜罐与客户端蜜罐.

2 蜜罐技术关键机制研究进展

通过对蜜罐技术发展历程的回顾与总结,本文提出如图 2 所示的蜜罐技术关键机制组成结构,分为核心机制与辅助机制两类.其中,

- 核心机制是蜜罐技术达成对攻击方进行诱骗与监测的必需组件:
  - ◊ 欺骗环境构建机制构造出对攻击方具有诱骗性的安全资源,吸引攻击方对其进行探测、攻击与利用;
  - ◊ 威胁数据捕获机制对诱捕到的安全威胁进行日志记录,尽可能全面地获取各种类型的安全威胁原始数据,如网络连接记录、原始数据包、系统行为数据、恶意代码样本等等;
  - ◊ 威胁数据分析机制则在捕获的安全威胁原始数据的基础上,分析追溯安全威胁的类型与根源,并对安全威胁态势进行感知;
- 而辅助机制是对蜜罐技术其他扩展需求的归纳,主要包括安全风险控制机制、配置与管理机制、反蜜罐技术对抗机制等:
  - ◊ 安全风险控制机制要确保部署蜜罐系统不被攻击方恶意利用去攻击互联网和业务网络,让部署方规避道德甚至法律风险;
  - ◊ 配置与管理机制使得部署方可以便捷地对蜜罐系统进行定制与维护;
  - ◊ 而反蜜罐技术对抗机制的目标是提升蜜罐系统的诱骗效果,避免被具有较高技术水平的攻击方利用反蜜罐技术而识别.

下面,本文针对蜜罐技术的 3 个核心机制以及具有技术博弈特征的反蜜罐技术对抗机制,对其研究进展进行归纳分析.

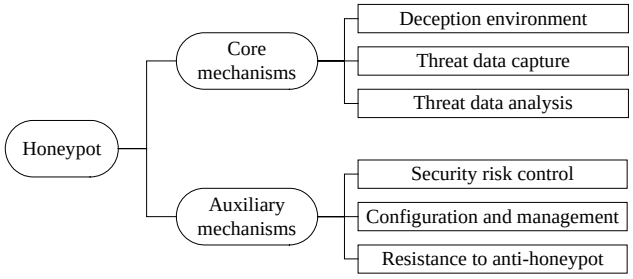

Fig.2 Key mechanisms of honeypot technology

图 2 蜜罐技术的关键机制的组成结构

2.1 欺骗环境构建机制

欺骗环境构建机制的实现方式决定了蜜罐能够为攻击方所提供的交互程度,主要有基于模拟仿真的实现

方式和基于真实系统搭建的方式.采用真实系统搭建方式能够比较容易地构建出一个具有良好诱骗性的蜜罐欺骗环境,并能够给攻击方提供充分的交互程度,因此,这种方式实现的蜜罐被称为高交互式蜜罐;而与之相反的是,模拟仿真方式则通过编制软件构建出一个伪装的欺骗系统环境来吸引攻击,并在一个安全可控的环境中对安全威胁进行数据记录,然而,这种方式一般只能为攻击方提供受限的交互程度,即只能实现低交互式蜜罐,对于一些未知的攻击方式与安全威胁不具备捕获能力.表 1 显示了基于欺骗环境构建机制对蜜罐工具软件进行分类的结果.

Table 1 Classification of honeypot technology based on the deception environment construction mechanism

表 1 基于欺骗环境构建机制的蜜罐技术分类

Category	Server-Side honeypot	Application-Layer honeypot	Client-Side honeypot
Low-Interaction honeypot	DTK[12], LaBrea[23], Honeyd[14], Nepenthes[15], Dionaea[25]	Glastopf[28]/GlastopfNG[29], SPAMPot[30], Kojoney[31], Kippo[32]	PhoneyC[34,35]
High-Interaction honeypot	HoneyBow[9], Argos[36]	GHH[26], HIHAT[27]	Capture-HPC[33], HoneyMonkey[37], SpyProxy[38]

- 服务端通用蜜罐

DTK[12]和 LaBrea[23]以绑定到指定端口上的网络服务软件实现方式模拟成网络服务攻击目标,吸引网络扫描探测与渗透攻击等安全威胁,而 Honeyd[14],Nepenthes[15]和 Dionaea[25]则通过模拟网络协议栈的方式提供仿真度更好的网络服务漏洞攻击环境,从而能够捕获更为丰富多样的安全威胁数据.然而,由于是以模拟方式实现各种应用层服务,因此对于攻击未知漏洞的网络渗透攻击与恶意代码,这类低交互式蜜罐工具往往无法为攻击方提供它们在网络服务交互过程中所预期的响应,因而无法进一步触发漏洞利用与恶意代码感染过程,也就无法捕获这些新型安全威胁.HoneyBow[9]在采用真实系统的高交互式蜜罐上构建,与 Nepenthes[15]等采用模拟服务方式的蜜罐工具软件相比,HoneyBow 拥有捕获恶意代码更具全面性、并能够捕获未知样本的优势. Argos[36]蜜罐则基于 x86 系统仿真器 Qemu 构建,能够对上层真实的 Guest 操作系统实施指令插装与监控,通过扩展动态污点分析(extended dynamic taint analysis)技术跟踪运行时刻接收到的网络数据,并从中识别出它们的非法使用,检测出网络渗透攻击,并进而支持自动化的攻击特征提取.虽然 HoneyBow 与 Argos 蜜罐的具体实现中都使用了虚拟化或系统仿真技术,但欺骗环境的构建仍使用了完整的真实操作系统及上层应用程序,因此仍然属于高交互式蜜罐的范畴.

- 应用层专用蜜罐

GHH[26]和 HIHAT[27]采用真实 Web 应用程序为模版搭建欺骗环境,属于高交互式蜜罐;而 Glastopf[28]/GlastopfNG[29],SPAMPot[30],Kojoney[31],Kippo[32]蜜罐则都是完全通过程序模拟的方式分别构建出 Web 站点、SMTP Open Relay 和弱口令配置的 SSH 服务以吸引互联网上的攻击,因此属于低交互式蜜罐.

- 客户端蜜罐

Capture-HPC[33]蜜罐使用真实操作系统及浏览器构建客户端环境对待检测页面进行访问,从中检测出含有渗透攻击脚本的恶意页面,采用同样方式构建的高交互式客户端蜜罐还有 HoneyMonkey[37],SpyProxy[38]等. PhoneyC[34,35]蜜罐则模拟实现浏览器软件,对检测页面进行解析、脚本提取和执行,从中发现恶意页面,属于低交互式客户端蜜罐.

欺骗环境除了系统、网络服务和应用程序之外,也可以是具有高度伪装性和诱骗性的数据内容.Barros 于 2003 年提出了蜜标(HoneyToken)技术概念[39]来描述这类用于吸引攻击者进行未经授权而使用的信息资源,而蜜标有着多种数据形态,比如一个伪造的身份 ID、邮件地址、数据库表项、Word 或 Excel 文档等等.当攻击方从环境中窃取信息资源时,蜜标将混杂在信息资源中同时被窃取,之后,一旦攻击方在现实场景中使用蜜标数据,比如使用一个经过标记的伪造身份 ID 尝试登陆业务系统,防御方就可以检测并追溯这次实际攻击.McRae 等人应用蜜标技术概念来追溯垃圾邮件发送者[40].Border 等人设计了 Siren 系统——一个能够产生蜜标数据序列的本地 Agent 来检测在用户主机上隐藏并将行为混杂在正常用户行为之中的间谍软件[41].White 给出了一种

创建个人身份信息蜜标的方法[42],并用于检测来自内部人员的恶意攻击与滥用[43].Chakravarty 等人在 Tor 匿名通信系统中注入作为诱饵的 IMAP,SMTP 邮件服务用户凭证信息,检测出了在匿名通信系统中存在的流量监听恶意节点[44].

2.2 威胁数据捕获机制

在构建欺骗环境吸引到攻击方的探测与攻击行为之后,蜜罐中需要实现的下一个核心机制是威胁数据捕获,监控和记录攻击方在蜜罐欺骗环境中进行的所有攻击行为,为追溯与分析安全威胁提供基础数据支持.

在低交互式蜜罐中,由于为攻击方提供的欺骗环境都是通过软件代码进行模拟实现的,因此对于攻击交互过程的监控和记录例程也通常在软件代码中加以实现.例如:Honeyd[14]在框架代码中实现了对所有网络连接的日志功能,并在服务模拟脚本中收集更多、更详细的应用层攻击数据;Nepenthes[15]与 Dionaea[25]除了记录网络连接日志以外,还特别针对恶意代码样本捕获的需求,通过仿真执行 Shellcode 分析得到其中包含的 URL 链接,并支持通过 HTTP,FTP 和 TFTP 下载到恶意代码样本.

而在高交互式蜜罐中,用于搭建欺骗环境的真实系统、网络服务与应用软件的日志记录功能通常并不能完全满足安全威胁监控的需求,因而需要研究人员自主设计并实现相应的威胁数据捕获机制.为了适应在蜜网体系架构中高交互式蜜罐的数据捕获需求,Balas 等人设计了一套全面的安全威胁数据捕获体系[45],结合使用了 Argus,Snort,p0f,tcpdump 和 Sebek 等开源工具,从网络和系统两个方面保证为进一步分析攻击行为提供全面而丰富的威胁数据.

在网络层面上,在蜜网网关唯一连接点上通过 Argus 监控所有流入和流出蜜网的网络连接记录;Snort 入侵检测系统对符合入侵检测特征的攻击数据包发出对应的报警信息,从而标识网络流中存在的已知攻击事件;p0f 被动操作系统辨识工具能够根据监听到的网络流中存有的指纹特征,判断攻击方操作系统的类型.另外,以 Tcpdump 监听全部流入流出蜜网的网络连接,记录原始流量数据.

在系统层面上,主要使用 Sebek[46]捕获攻击者在蜜罐主机上的行为,Sebek 客户端以内核模块方式安装在蜜罐主机上,在不被攻击者发现的前提下对系统行为数据及键击记录进行捕获,并通过一个对攻击者隐蔽的通信信道传送到蜜网网关上的 Sebek 服务器端.HoneyBow[9]通过文件系统实时监控和文件列表交叉对比方法来捕获感染高交互式蜜罐系统的恶意代码样本,Argos[36]蜜罐则在二进制指令层次上对网络输入非可信数据在内存中的扩展轨迹进行跟踪,并监控污染数据的非法使用来定位出渗透攻击关键数据在网络会话中的位置,支持攻击特征的自动分析提取.

2.3 威胁数据分析机制

The Honeynet Project 的第三代蜜网体系架构为安全研究人员提供较为全面的辅助数据分析功能[47],通过蜜网网关上的 hflow 工具,将蜜网捕获的网络与系统行为监控数据进行聚合,汇总到关系型数据库中,然后由 Walleye 工具提供基于 Web 方式的安全威胁辅助分析接口,从而帮助安全研究人员能够快速地理解蜜网中捕获的攻击事件.然而,大规模部署的分布式蜜罐与分布式蜜网系统不能仅仅为安全管理人员提供简单的辅助分析接口而依赖于人工对大量安全威胁数据进行深入分析与态势感知,这就为自动化的安全威胁数据分析机制提出了研究需求.

最基础的威胁数据分析机制为实证分析,即通过对实验采集数据进行统计汇总,揭示出安全威胁的基本统计特性.例如,Kaaniche 等人对 Leurré.com 分布式蜜罐系统捕获数据进行了实证分析[48],以更好地理解互联网上的攻击策略与工具.可视化分析技术可以进一步对蜜罐捕获的安全威胁数据进行 2D 图形化与 3D 动画效果展示,以非常直观的方式将威胁数据展示给安全研究人员,使其快速理解捕获安全威胁的整体态势,并发现其中可能包含的异常事件.Krasse 等人实现了一种结合平行坐标线连接分析与基于时间序列的稀疏线动画效果的可视化技术[49],对蜜网中捕获的网络数据包进行动态展示.针对多种实时与取证分析数据集进行实验测试,验证了该可视化技术可以有效地帮助安全研究人员理解蜜网捕获的大量威胁数据.

更进一步的威胁数据分析方法能够解释出捕获数据背后的根源,Almotairi 等人采用 PCA(principal

component analysis)方法从 Leurré.com 分布式蜜罐系统数据提取出潜在的攻击行为聚类,并进行归因分析[50].Thonnarda 等人同样以 Leurré.com 系统数据作为实验对象,提出了基于攻击时序相似性的聚类方法,从蜜网数据中发现普遍的攻击模式[51].Zhuge 等人则应用了关联分析方法[52],在安全知识库的支持下,该方法能够从蜜网捕获安全威胁数据中识别出攻击规划,并重构出攻击过程场景,从而有助于安全研究人员更好地发现和理解捕获数据中蕴含的攻击场景.

2.4 反蜜罐技术对抗机制

在蜜罐技术得到安全社区的广泛关注之后,一些黑客与安全研究人员从攻击方角度对蜜罐识别与绕过等反蜜罐技术开展研究,并提出了一系列对抗现有蜜罐技术的机制.Krawetz 介绍了第一个公开的反蜜罐软件 Honeypot Hunter[53],该软件应对蜜罐对垃圾邮件发送者造成的威胁,尝试在互联网开放代理服务器中识别出伪装的蜜罐主机,从而避免垃圾邮件发送者落入蜜罐的圈套.Corey 则在黑客界的著名电子期刊《Phrack》发表了一篇技术文章[54],针对当时主流的第二代蜜网中的 Sebek 系统行为监控工具、Honeyd 虚拟蜜罐与 VMware 虚拟机环境提出了蜜罐识别与检测技术.Dornseif 等人也针对 Sebek 行为监控组件提出了识别与移除 Sebek 的技术方法[55].Oudot 等人[56]总结了可以从网络层面上检测识别蜜罐技术的现有方法,Holz 等人[57]则总结了可以从系统层面上检测识别蜜罐技术的现有方法.Zou 等人也在僵尸网络的构建与维护过程中引入对蜜罐识别的问题[58,59],基于蜜罐部署时必须采用攻击控制机制防止蜜罐参与真正攻击过程的前提假设,提出了一种创新的“两阶段侦察”僵尸网络传播方式,以确保僵尸网络构建过程中不会包含蜜罐节点.

为了应对攻击方所引入的反蜜罐技术,蜜罐研究社区也以一种技术博弈与对抗的思维,研究更具隐蔽性的攻击监控技术体系,其中一个非常重要的研究进展就是将系统行为监控技术从原先的内核层向更为底层的虚拟层进行转移.Quynh 等人[60]在 Xen 虚拟层中实现了对蜜罐系统行为的监控,构建了 Xebek,以替代 Sebek 工具.Jiang 等人[61]也提出了一种针对虚拟化部署高交互式蜜罐的外部监控技术,实现了 VMScope 蜜罐行为监控软件.Song 等人[62]则在 Qemu 开源虚拟机中增加任意指定系统调用或应用层函数上的断点监控机制,能够更加灵活地提取到各类系统行为记录与运行结果信息.

3 蜜罐部署结构发展

在蜜罐工具软件与关键机制随着安全威胁变化而不断得到发展的同时,如何有效地结合应用不同类型蜜罐技术,在公共互联网或大规模业务网络中进行部署,以扩大安全威胁的监测范围并提升监测能力,成为了蜜罐技术社区研究与工程实践中的一个重要关注点.蜜罐技术社区也逐渐提出了蜜网、分布式蜜罐与分布式蜜网、蜜场等部署结构框架.

3.1 蜜　网

蜜网(honeynet)技术的提出,为系统可控地部署多种类型蜜罐提供了基础体系结构支持.蜜网技术从概念证明性的第一代,经过逐渐成熟和完善的第二代,目前已步入完整、易部署、易维护的第三代[63],图 3 显示了蜜网的基本结构,多台各种类型的蜜罐系统构成蜜网网络,并通过一个以桥接模式部署的蜜网网关(HoneyWall)与外部网络连接.蜜网网关构成了蜜网与外部网络的唯一连接点,外部网络所有与蜜罐系统的网络交互流量都将通过蜜网网关,因此,在蜜网网关上可以实现对安全威胁的网络数据捕获,以及对攻击进行有效控制.此外,蜜网网关的桥接方式不对外提供 IP 地址,同时也不对通过的网络流量进行 TTL 递减与路由,以确保蜜网网关极难被攻击方发现.而安全研究人员通过蜜网网关的管理接口连接对蜜网网关进行管理控制,以及对蜜网网关上捕获和汇集的安全威胁数据进行分析.

由于蜜网结构提供了安全可控的蜜罐部署环境,因此除了集成部署各种类型的蜜罐工具软件之外,还可以直接在蜜网中包含真实系统作为高度欺骗性的蜜罐,这使得攻击者可以和真实的操作系统与应用服务进行交互,让攻击者在蜜网中有更充分的活动空间,也保证了安全研究人员能够捕获到更加丰富和全面的威胁数据.

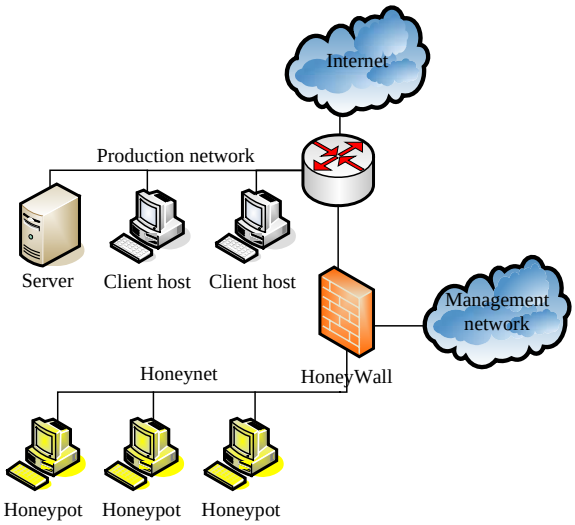

Fig.3 Basic architecture of honeynet

图 3 蜜网基本体系结构图

3.2 分布式蜜罐与分布式蜜网

分布式蜜罐/蜜网是目前安全研究机构、CERT 组织与安全公司基于蜜罐技术搭建互联网安全威胁监测体系的典型技术方案,但已有的、公开的分布式蜜罐/蜜网系统在采用底层蜜罐技术、资源要求与系统组织方式等方面均存在着较大的差异.

欧洲电信的 Leurre.com 分布式蜜罐系统[18]基于 Honeyd 虚拟蜜罐工具软件[14],可以采用资源要求较低的低成本硬件进行部署,容易维护且部署安全风险很低,这使得 Leurre.com 系统能够以自愿加盟的方式在世界五大洲的 28 个国家部署了 70 个节点,从 2004 年~2008 年 3 月,共捕获了来自于近 350 万源 IP 地址的大量网络会话,并通过聚类分析[64]、数据挖掘[65]等方法从中发现安全威胁根源.巴西 CERT 的分布式蜜罐系统[17]与 Leurre.com 系统类似地采用了 Honeyd 框架与合作加盟方式,在巴西国内部署了近 30 个节点,以构建国家互联网安全预警系统.

然而,仅仅依赖单一的 Honeyd 虚拟蜜罐框架软件和一些定制的模拟服务脚本,只能为互联网安全威胁提供非常受限的交互环境,无法触发和记录网络攻击会话的完整信息.因此,之后开发与部署的分布式蜜网系统往往将基于模拟的低交互式蜜罐与使用真实系统的高交互式蜜罐相结合,以结合两者各自的优势.The Honeynet Project 推进的 GDH 分布式蜜网系统[16]以 VMware 虚拟化平台结合 Nepenthes 虚拟蜜罐[15]、Kojoney 蜜罐[31]与高交互式蜜罐,采用分支团队合作加盟方式部署了 11 个节点,在 2007 年 3 月~5 月运行期间,捕获了 30 多万个不同源 IP 地址的 7 300 多万个网络会话连接、67 万余次 SSH 口令尝试以及 1 680 个不同的恶意代码样本.2005 年~2008 年,Zhuge 等人支持 CNCERT/CC 研发和部署 Matrix 中国分布式蜜网系统,采用自主实现的 HoneyBow 高交互式蜜罐[9],并集成 Nepenthes 蜜罐,利用 CNCERT/CC 在全国的分支机构的硬件与网络资源条件,在 31 个省市区共部署了 50 个节点,从完成部署后的 2006 年 10 月~2007 年 6 月的 8 个月时间中,共捕获了约 80 万次恶意代码感染,提取到近 10 万个不同的恶意代码样本.在国家中心部署的恶意代码自动分析服务与僵尸网络跟踪工具软件的进一步支持下,Matrix 系统发现并监测了 3 290 个不同的 IRC 僵尸网络,并对 IRC 僵尸网络行为模式进行了细致的调查分析[10].欧洲电信将 Leurre.com 系统升级改造为 SGNET[19],以分布式方式部署添加了 ScriptGen 功能特性的 Honeyd 虚拟蜜罐,ScriptGen[66]技术能够学习网络应用协议的自动状态机,并生成 Honeyd 中相应服务模拟脚本,从而提升蜜罐系统为攻击者提供的交互程度.

3.3 蜜 场

蜜场概念的引入,为基于蜜罐技术构建网络安全威胁监测体系提供了一种不同的部署结构模式,也为将蜜罐技术用于直接防护大规模分布式业务网络提供了一条可行的途径.

如图 4 所示,在蜜场体系架构中,蜜罐系统都被集中部署于一个受控的欺骗网络环境中,由安全专家来负责维护、管理与威胁数据分析,而在业务网络中仅仅部署一些轻量级的重定向器,对以未使用 IP 地址为目标的网络流量或者通过入侵防御系统等设备检测出的已知网络攻击会话,重定向迁移至蜜场环境中,由蜜罐系统与攻击源进行交互,在具有伪装性的欺骗环境中更加深入地分析这些安全威胁.Jiang 等人采用蜜场技术概念实现了 Collapsar 系统[20],用于网络攻击的检测与深入分析,通过几个真实捕获的攻击事件案例验证了系统的有效性与实用性.

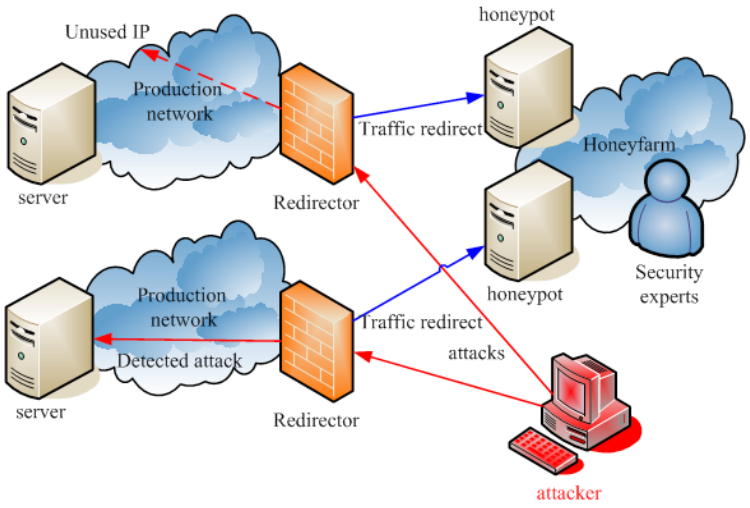

Fig.4 Architecture of honeyfarm technology concept

图 4 蜜场技术概念图示

蜜场技术框架实现的难点,在于重定向网络攻击会话的透明性以及蜜场环境对于分析大量网络攻击连接的可扩展性.陆腾飞等人在应用蜜场技术构建的主动式网络安全防护系统 Icarus[22]中,引入了具备高度透明性的网络会话重定向与迁移技术[67],通过策略路由方式将指定的网络攻击连接从业务网络中透明地重定向到蜜场网关,首先交由其上部署的低交互式蜜罐与攻击源进行交互,在低交互式蜜罐对于一些未知攻击无法进行正确响应时,再通过 TCP 会话迁移过程将攻击会话无缝地迁移到高交互式蜜罐中.这种蜜罐多级部署策略可以有效提升蜜场环境的可扩展性,而网络攻击会话迁移技术能够保证整个交互响应过程对攻击方的透明性.为了缓解蜜场环境中大规模监控安全威胁与更为深入地捕获威胁数据之间的矛盾,Vrable 等人[21]利用虚拟化技术、最大限度的内存共享机制和资源延迟绑定策略实现了 Potemkin 原型蜜场系统,能够使用有限的硬件资源同时模拟出超过 64 000 个蜜罐系统,有效提升了蜜场环境的可扩展性.

4 蜜罐技术的应用

4.1 网络入侵与恶意代码检测

作为一种主动安全防御策略,蜜罐技术最初的应用场景是辅助入侵检测技术来发现网络中的攻击者与恶意代码.Kuwatly 等人[68]依据动态蜜罐技术概念,通过集成主动探测与被动辨识工具,对 Honeyd 虚拟蜜罐进行动态配置,在动态变化的网络环境中构建出自适应蜜罐系统,达到对网络中非法入侵的检测目的.Artail 等人[69]进一步提出了混杂模式的蜜罐架构,使用低交互式虚拟蜜罐来模拟服务与操作系统,并将包含攻击的恶意流量引

导至高交互式真实服务蜜罐,增强对网络入侵者行为的监视、分析与管控能力.Anagnostakis 等人[70]则提出了Shadow(影子)蜜罐的创新思路,组合了蜜罐技术与异常入侵检测技术的各自优势,首先使用异常检测器监视所有到受保护网络的流量,检测出的可疑流量通过 Shadow 蜜罐进行处理,Shadow 蜜罐与受保护业务系统共享所有内部状态,入侵攻击在影响受保护业务系统状态之前会被 Shadow 蜜罐检测出来,而被异常检测器错误识别的合法流量通过 Shadow 蜜罐验证之后,由业务系统向用户提供透明的响应.

蜜罐技术对具有主动传播特性的网络蠕虫等恶意代码具有很好的检测效果.Dagon 等人[71]针对局域网中如何在爆发初期就检测出蠕虫的问题,实现了脚本驱动并覆盖大量 IP 地址范围的 HoneyStat 蜜罐系统,HoneyStat 针对局域网蠕虫传播场景,生成内存操作、磁盘写、网络这 3 种不同类型的报警事件,并通过自动化的报警数据收集与关联分析,能够快速、准确地检测出零日蠕虫爆发.在欧盟 FP6 计划资助的 NoAH 项目中,SweetBait 系统[72]集成了 SweetPot 低交互式蜜罐与 Argos 高交互式蜜罐[36]对互联网上传播的蠕虫进行实时检测,能够进一步自动生成检测特征码,并通过分布式部署与特征码共享构建蠕虫响应机制,针对 2008 年底爆发的 Conficker 蠕虫进行了在线检测与分析[73],验证了系统的有效性.

除了网络蠕虫等传统类型恶意代码之外,研究人员还应用蜜罐技术检测与分析针对浏览器等客户端软件的恶意网页攻击.HoneyMonkey 系统[37]通过引入高交互式的客户端蜜罐技术,使用安装了不同补丁级别的操作系统与浏览器软件来检测和发现针对浏览器安全漏洞实施渗透攻击的恶意页面.Google 的 Safe Browsing 项目[74]中结合使用了机器学习方法与高交互式客户端蜜罐技术,从搜索引擎爬取的海量网页中检测出超过 300 万导致恶意程序植入的 URL 链接,并系统性地对恶意网页现象进行了深入分析.

4.2 恶意代码样本捕获

在检测恶意代码的基础上,最近发展出的蜜罐技术还具备恶意代码样本自动捕获的能力.Nepenthes[15]是最早出现基于蜜罐技术自动化捕获与采集恶意代码样本的开源工具,Nepenthes 具有灵活的可扩展性,使用单台物理服务器就可以覆盖一个/18 网段的监测,在 33 个小时中捕获了超过 550 万次渗透攻击,并成功捕获到 150 万个恶意代码样本,在依据不同 MD5 值的消重处理后,最终在这段时间内捕获了 408 个不同的恶意代码样本,实际捕获数据验证了 Nepenthes 蜜罐在自动捕获主动传播型恶意代码方面的有效性.HoneyBow 系统[9]则实现了基于高交互式蜜罐的恶意代码样本自动捕获流程,在 Matrix 分布式蜜网系统 9 个月的实际部署与监测周期中,HoneyBow 平均每天捕获了 296 个不同恶意代码样本,较 Nepenthes(63.7 个)有着明显的优势,但若集成两者优势,则可达到更好的恶意代码捕获效果[8].WebPatrol[75]则针对攻击客户端的恶意网页木马场景,提出了结合低交互式蜜罐与“类 Proxy”Cache 与重放技术的自动化捕获方法,将分布式存储于多个 Web 站点、动态生成且包含多步骤多条路径的恶意网页木马场景,进行较为全面的采集与存储,并支持重放攻击场景进行离线分析,WebPatrol 系统在 5 个月的时间内,从 CERNET 网络中的 1 248 个被挂马网站上采集捕获了 26 498 个恶意网页木马攻击场景.

研究社区已经证实了蜜罐技术在恶意代码样本采集方面的能力,因此,反病毒工业界目前也已经普遍地通过大规模部署蜜罐来采集未知恶意代码样本,如国际著名的反病毒厂商 Symantec[76]等.

4.3 安全威胁追踪与分析

在检测并捕获安全威胁数据之后,蜜罐技术也为僵尸网络、垃圾邮件等特定类型安全威胁的追踪分析提供了很好的环境支持.

僵尸网络监测与追踪是应用蜜罐技术进行安全威胁深入分析的一个热点方向,其基本流程是由蜜罐捕获通过互联网主动传播的僵尸程序,然后在受控的蜜网环境或沙箱中对僵尸程序进行监控分析,获取得到其连接僵尸网络命令与控制服务器的相关信息,然后以 Sybil 节点对僵尸网络进行追踪,在取得足够多的信息之后可进一步进行 sinkhole、关停、接管等主动遏制手段.Freiling 等人最早使用蜜罐技术来进行僵尸网络追踪[77],Rajab 等人进一步提出了一种多角度同时跟踪大量实际僵尸网络的方法[78],包括旨在捕获僵尸程序的分布式恶意代码采集体系、对实际僵尸网络行为获取内部观察的 IRC 跟踪工具以及评估僵尸网络全局传播足迹的 DNS 缓

存探测技术,通过对多角度获取数据的关联分析,展示了僵尸网络的一些行为和结构特性.诸葛建伟等人利用 Matrix 分布式蜜网系统对 IRC 僵尸网络行为进行了长期而全面的调查[10],揭示了现象特征[79].Stone-Gross 等人[80]在蜜罐技术监测僵尸网络行为的基础上,通过抢注动态域名的方法,对 Torpig 僵尸网络进行了接管,不仅追踪到了 18 万个僵尸主机 IP 地址,还收集到了 70GB 的敏感信息,验证了僵尸网络追踪与托管技术可以达到的主动遏制效果.

针对互联网上垃圾邮件泛滥的现象,Project Honey Pot 项目利用超过 5 000 位网站管理员自愿安装的蜜罐软件监控超过 25 万个垃圾邮件诱骗地址,对收集邮件地址并发送垃圾邮件的行为进行了大规模的追踪分析[81],Steding-Jessen 等人使用低交互式蜜罐技术来研究垃圾邮件发送者对开放代理的滥用行为[82],Levchenko 等人对蜜罐采集到的垃圾邮件的地下经济链进行追踪分析,揭示出支付环节是这一地下经济链的瓶颈,并建议采取相应的管理政策来遏制其发展[83].

4.4 攻击特征提取

蜜罐系统捕获到的安全威胁数据具有纯度高、数据量小的优势,通常情况下也不会含有网络正常流量.此外,只要蜜罐系统能够覆盖网络中的一小部分 IP 地址范围,就可以在早期监测到网络探测与渗透攻击、蠕虫等普遍化的安全威胁.因此,蜜罐非常适合作为网络攻击特征提取的数据来源.安全研究人员提出了多种基于蜜罐数据进行网络攻击特征提取的方法,本文在表 2 中对这些方法进行了总结与对比分析.

Table 2 Comparison of the network signature generation methods using honeypot

表 2 应用蜜罐技术的网络攻击特征提取方法比较

Method	Base honeypot	Signature generation method	Target attack class	Signature type	Resistance to polymorphic
Honeycomb[84]	Honeyd	Pairwise LCS across connections	General	Continuous byte patterns	No
Nemean[85]	Virtual honeypot/ Win2KS	(MSG) Clustering and automata learning	General	Finite-State-Automata signatures	No
SweetBait[72]	Argos[36]	LCS, CREST	General	Continuous byte patterns	Partially
HoneyCyber[86, 87]	Double-Honeynet	Principal component analysis	Worm	Continuous byte patterns	Claimed

Honeycomb[84]是最早公开的基于蜜罐技术进行自动化网络攻击特征提取的研究工作,作为 Honeyd 蜜罐的扩展模块而实现,对于蜜罐接受到的网络攻击连接,通过与相同目标端口的保存网络连接负载进行一对一的最长公共子串(LCS)匹配,如果匹配到超出最小长度阈值的公共子串,即生成一条候选特征,这些候选特征再与已有特征集进行聚合,生成更新后的攻击特征库.Honeycomb 提出了利用蜜罐捕获数据进行攻击特征的基础方法,但并未考虑应用层协议语义信息.很多情况下,由于应用层协议头部中相同内容的影响而提取出与攻击无关的无效特征.Nemean 系统[85]针对 Honeycomb 的这一缺陷,提出了具有语义感知能力的攻击特征提取方法,以虚拟蜜罐和 Windows 2000 Server 物理蜜罐捕获的原始数据包作为输入,首先通过数据抽象模块将原始数据包转换为 SST 半结构化网络会话树,然后由特征提取模块应用 MSG 多级特征泛化算法,将网络会话进行聚类,并对聚类进行泛化,生成基于有限状态机的语义敏感特征,最后转换为目标入侵检测系统的特征规则格式进行实际应用.SweetBait[72]/Argos[36]则针对主动发布型蜜罐应用场景,首先采用动态污点分析技术来检测出渗透攻击,并回溯造成 EIP 指令寄存器被恶意控制的污点数据在网络会话流中的具体位置,在特征自动提取环节,则支持 LCS 最长公共子串算法与 CREST 渗透攻击关键字符串检测算法.其中,CREST 算法能够依据动态污点分析的回溯结果,提取到简练但更加精确的攻击特征,也能够部分地对抗网络攻击的多态化.HoneyCyber 系统[86,87]利用了 double-honeynet 部署架构来捕获多态网络蠕虫的流入会话与流出会话,并采用 PCA 分析方法提取多态蠕虫不同实例中的显著数据,进行自动化的特征提取.在实验中,针对人工多态化处理的蠕虫实例,能够达到零误报率和较低的漏报率,但并未针对实际流量环境与多态蠕虫案例进行验证.

5 讨 论

5.1 蜜罐技术现有问题

蜜罐技术并非像加密、访问控制、防火墙在整体防御体系特定环节中拥有着明确的功能需求与定义,而是安全攻防领域中的一种对抗性思维方式与技术思路,贯穿于整体防御体系的各个环节,体现在防御者与攻击者之间的智力与技能博弈中.因此,蜜罐技术并不具备固定且明确的边界与内涵,同时还在技术博弈过程中与攻击技术共同发展和演化,从而始终处于动态变化的状态,这导致了蜜罐技术无法像访问控制等安全技术那样拥有着较为明确的理论基础,而更多地依赖于一些对抗性的技术策略与手段技巧.这也意味着,现有的蜜罐技术无法拥有科学理论的基础支持,同时也难以形成通用化的工程产品形态与标准规范.在安全业界市场上,虽然近年来 Symantec 等一些厂商试图推广蜜罐产品,但始终无法形成一种共性化的通用产品形态,而往往是为安全要求较高的企业客户提供定制化蜜罐技术方案.

作为与攻击具有直接对抗性的技术手段,现有蜜罐技术在以下方面还存在着不足与缺陷,使其还不足以在技术对抗博弈中占据绝对的优势地位:

(1) 蜜罐环境尚无法有效解决仿真度与可控性之间的矛盾:采用与业务系统一致环境的高交互式蜜罐技术虽然存在高度的伪装性与诱骗性,但在攻击行为监控粒度与完备性、环境维护成本与代价等方面存在不足;而具有高度可控性、维护成本较小的低交互式蜜罐技术则往往仿真度不够高,特别是无法捕获新型攻击威胁,而且比较容易被识别;

(2) 现有蜜罐技术主要针对具有大规模影响范围的传统普遍化安全威胁,而对于具有目标性的高级持续性威胁(advance persistent threat),诱骗与监测能力还不够充分.应对高级持续性威胁的蜜罐技术必须具备高度可定制性以及动态环境适应能力,同时应具有很高的隐蔽性,能够融入实际业务系统中,从而不会被轻易发现;

(3) 虽然蜜罐在实际应用层面上属于一种主动性防御技术,但在攻防核心技术博弈的层面上,蜜罐技术针对新类型与新平台(如 SCADA 工业控制服务、智能手机移动平台)的安全威胁的应对研究与开发,与攻击技术的快速发展相比,仍具有一定的滞后性.

除了技术方面的不足之外,蜜罐技术在真实环境中的应用还可能受到各国法律的限制与影响,Spitzner 在美国的法律框架下,从引诱犯罪、隐私与责任追究等方面分析了防御方应用蜜罐技术可能涉及的法律问题[88],他认为,蜜罐并不具有引诱攻击者犯罪的条件特性,而仅仅是为主动发起的攻击提供一个与业务系统所不同的目标环境;在特别注重隐私保护的美国法律框架下,蜜罐的部署方可以给出环境中存在行为监控的警示以避免受到侵犯攻击者隐私信息的指控;而在部署的蜜罐被攻击者用于攻击第三方时的责任追究问题上,目前还未出现过由于网络被攻击者滥用而被追究法律责任的案例.由于各国法律存在着较大的差异,因此,防御方在实际应用中部署蜜罐及对攻击者进行特定程度的监控与响应是否会涉及法律问题,还需咨询法律顾问和律师进行更为准确、清晰的了解与判断.

5.2 蜜罐技术的发展趋势

蜜罐技术在与攻击技术的对抗性博弈过程中将不断进行发展与演化.在技术概念方面,随着云计算平台的日趋成熟,The Honeynet Project 提出了新的“蜜云(HoneyCloud)”技术概念[89],也启动了 HoneyCloud 项目,以 IaaS(基础设施即服务)方式为安全研究人员部署蜜罐监测互联网安全威胁提供平台与网络资源的支持.“蜜云”概念的提出,也为蜜罐技术部署结构研究提供了新的思路.在蜜罐的核心关键机制方面,如何解决欺骗环境的仿真度与可控性之间的矛盾,仍是蜜罐技术未来发展亟需解决的问题.在如 SGNET 等一些大规模部署蜜罐实施互联网安全威胁监测分析的大型科研项目中,研究人员将继续改进蜜罐技术,并通过综合多种蜜罐技术的部署结构与协作机制创新来克服这一技术挑战.在蜜罐技术应用方面,安全研究社区正在普遍地应用蜜罐技术对新形态与新平台安全威胁进行监测与分析研究,Mulliner 等人在 Android 智能手机平台上研究开发了第一个智能手机蜜罐——HoneyDroid[90],用于捕获与分析针对智能手机的攻击.

5.3 蜜罐技术进一步的研究方向

综合以上讨论,我们认为蜜罐技术领域的进一步研究方向包括:(1) 利用云计算平台的新型蜜罐技术部署结构与社区协作式的安全威胁监测模式;(2) 可定制、可扩展的蜜罐技术框架研究与开发以及具有业务环境自适应能力的动态蜜罐技术;(3) 应用蜜罐与蜜标技术思路,针对新形态、新平台网络安全威胁的深入追踪与分析研究;(4) 蜜罐识别探测技术及其相应的对抗技术措施研究等.

6 总 结

蜜罐技术从 20 世纪 90 年代出现后发展至今,已经成为安全管理人员用于监测、追踪与深入分析安全威胁的一种主动防御技术手段,也已经作为一种常用的对抗性思维方式,被安全研究社区广泛应用于新形态安全威胁的监测分析.本文回顾总结了蜜罐技术概念从蜜罐、蜜网、分布式蜜罐、分布式蜜网到蜜场的演化历程,并对蜜罐技术的关键机制、部署结构及应用的主要研究开发与实际部署等工作进行了综述与分析.从本质上分析,蜜罐技术是一种与攻击技术进行博弈的对抗性思维方式与技术思路,因此,这一技术将随着安全威胁演化而不断地发展与更新,也仍将作为安全社区所普遍应用的安全威胁监测、追踪与分析技术方法得到持续的研究和关注.

References:

[1] Spitzner L. Honeypots: Tracking Hackers. Boston: Addison-Wesley Longman Publishing Co., Inc., 2002.

[2] Stoll C. The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage. London: The Bodley Head Ltd., 1989.

[3] The Honeynet Project, Know Your Enemy: Learning about Security Threats. 2nd ed., Boston: Addison-Wesley Professional, 2004.

[4] Spitzner L. Honeypot farms. 2012. http://www.symantec.com/connect/articles/honeypot-farms

[5] Liu BX, Xu RS. Study and design of the proactive security protecting measure honeynet. Computer Engineer, 2002,28(12):9−11 (in Chinese with English abstract).

[6] Cao AJ, Liu BX, Xu RS. Summary of the honeynet and entrapment defense technology. Computer Engineer, 2004,30(9):1−3 (in Chinese with English abstract).

[7] Zhang F, Zhou SJ, Qin ZG, Liu JD. Honeypot: A supplemented active defense system for network security. In: Fan P, Shen H, eds. Proc. of the 4th Int'l Conf. on Parallel and Distributed Computing, Applications and Technologies. Gran Canaria: IEEE, 2003. 231−235. [doi: 10.1109/PDCAT.2003.1236295]

[8] Zhou YL, Zhuge JW, Xu N, Jiao XL, Sun WM, Ji YC, Du YJ. Matrix: A distributed honeynet and its applications. In: Proc. of the 20th Annual FIRST Conf. (FIRST 2008). British Columbia, 2008. http://www.first.org/conference/2008/papers/zhou-yonglin-slides.pdf

[9] Zhuge JW, Holz T, Han XH, *et al*. Collecting autonomous spreading malware using high-interaction honeypots. In: Qing SH, ed. Proc. of 9th Int'l Conf. on Information and Communications Security (ICICS 2007). LNCS 4861, Zhengzhou: Springer-Verlag, 2007. 438−451. [doi: 10.1007/978-3-540-77048-0_34]

[10] Han XH, Guo JP, Zhou YL, Zhuge JW, Zou W. Investigation on the botnets activities. Journal on Communications, 2007,28(12): 167−172 (in Chinese with English abstract).

[11] Cheng JR, Yin JP, Liu Y, Zhong JW. Advances in the honeypot and honeynet technologies. Journal of Computer Research and Development, 2008,45(Suppl.):375−378 (in Chinese with English abstract).

[12] Cohen F. The deception toolkit. 2012. http://all.net/dtk/index.html

[13] Cohen F, Lambert D, Preston C, Berry N, Stewart C, Thomas E. A framework for deception. 2012. http://www.all.net/journal/deception/Framework/Framework.html

[14] Provos N. A virtual honeypot framework. In: Proc. of the 13th Conf. on USENIX Security Symp. Berkeley: USENIX Association, 2004. 1−14.

[15] Baecher P, Koetter M, Holz T, Dornseif M, Freiling F. The Nepenthes platform: An efficient approach to collect malware. In: Diego Z, *et al*., eds. Proc. of the 9th Int'l Symp. on Recent Advances in Intrusion Detection (RAID 2006). LNCS 4219, Hamburg: Springer-Verlag, 2006. 165−184. [doi: 10.1007/11856214_9]

[16] Watson D, Riden J. The honeynet project: Data collection tools, infrastructure, archives and analysis. In: Zanero S, ed. Proc. of the WOMBAT Workshop on Information Security Threats Data Collection and Sharing (WISTDCS 2008). Amsterdam: IEEE Computer Society Press, 2008. 24–30. [doi: 10.1109/WISTDCS.2008.11]

[17] Hoepers C, Steding-Jessen K, Cordeiro LER, Chavos MHPC. A national early warning capability based on a network of distributed honeypots. In: Proc. of the 17th Annual FIRST Conf. on Computer Security Incident Handling (FIRST 2005). Singapore, 2005. http://www.cert.br/docs/palestras/certbr-early-warning-first2005.pdf

[18] Leita C, Pham VH, Thonnard O, Ramirrez-Silva E, Pouget F, Kirda E, Dacier M. The Leurre.com project: Collecting Internet threats information using a worldwide distributed honeynet. In: Zanero S, ed. Proc. of the WOMBAT Workshop on Information Security Threats Data Collection and Sharing (WISTDCS 2008). Amsterdam: IEEE Computer Society Press, 2008. 40–57. [doi: 10.1109/WISTDCS.2008.8]

[19] Leita C, Dacier M. SGNET: A worldwide deployable framework to support the analysis of malware threat models. In: Avižienis A, ed. Proc. of the 7th European Dependable Computing Conf. Kaunas: IEEE Computer Society Press, 2008. 99–109. [doi: 10.1109/EDCC-7.2008.15]

[20] Jiang X, Xu D. Collapsar: A VM-based architecture for network attack detention center. In: Blaze M, ed. Proc. of the 13th Conf. on USENIX Security Symp. Berkeley: USENIX Association, 2004. 15–28.

[21] Vrable M, Ma J, Chen J, Moore D, Vandekieft E, Snoeren AC, Voelker GM, Savage S. Scalability, fidelity, and containment in the potemkin virtual honeyfarm. ACM SIGOPS Operating Systems Review (SOSP 2005), 2005,39(5):148–162. [doi: 10.1145/1095809.1095825]

[22] Lu TF, Chen ZJ, Zhuge JW, Han XH, Zou W. Research and implementation of network attack flow redirection mechanism in the honeyfarm environment. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 2009,29(3):14–20 (in Chinese with English abstract).

[23] Liston L. Welcome to my tarpit: The tactical and strategic use of LaBrea. Dshield.org White Paper, 2011. http://www.hackbusters.net/LaBrea/LaBrea.txt

[24] Nepenthes Development Team. Libemu—x86 shellcode detection. 2011. http://libemu.carnivore.it

[25] Nepenthes Development Team. Dionaea. 2011. http://dionaea.carnivore.it/

[26] Riden J, McGeehan R, Engert B, Mueter M. Know your enemy: Web application threats, using honeypots to learn about HTTP-based attacks. The Honeynet Project White Paper, 2011. http://honeynet.org/book/export/html/

[27] Muter M, Freiling F, Holz T, Matthews J. A generic toolkit for converting Web applications into high-interaction honeypots. University of Mannheim, 2008. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.89.5000

[28] Rist L, Vetsch S, Koßin M, Mauer M. Know your tools: Glastopf—A dynamic, low-interaction Web application honeypot. 2011. http://honeynet.org/papers/KYT_glastopf

[29] Vetsch S. GlastopfNG—A Web attack honeypot [Bachelor Thesis]. Bern University of Applied Sciences, 2010.

[30] Pickett N. SPAMPot.py—Spam honeypot SMTP server. 2011. http://woozle.org/~neale/src/python/spampot.py

[31] Coret JA. Kojoney—A honeypot for the SSH service. 2011. http://kojoney.sourceforge.net/

[32] Kippo—SSH Honeypot. 2011. http://code.google.com/p/kippo/

[33] Seifert C, Steenson R, Holz T, Yuan B, Davis MA. Know your enemy: Malicious Web servers. The Honeynet Project White Paper, 2011. http://www.honeynet.org/book/export/html/153

[34] Nazario J. PhoneyC: A virtual client honeypot. In: Lee W, ed. Proc. of the 2nd USENIX Conf. on Large-Scale Exploits and Emergent Threats (LEET 2009). Boston: USENIX Association, 2009. 6.

[35] Chen ZJ, Song CY, Han XH, Zhuge JW. Detecting heapspray in drive-by download attacks using opcode dynamic instrumentation. In: Proc. of the 2nd Conf. on Vulnerability Analysis and Risk Assessment (VARA 2009). Beijing, 2009 (in Chinese).

[36] Portokalidis G, Slowinska A, Bos H. Argos: An emulator for fingerprinting zero-day attacks for advertised honeypots with automatic signature generation. ACM SIGOPS Operating Systems Review, 2006,40(4):15–27. [doi: 10.1145/1218063.1217938]

[37] Wang YM, Beck D, Jiang X, Roussev R. Automated Web patrol with strider HoneyMonkeys: Finding Web sites that exploit browser vulnerabilities. In: Harder E, ed. Proc. of the Network and Distributed System Security. San Diego: The Internet Society, 2006.

[38] Moshchuk A, Bragin T, Deville D, Gribble S, Levy H. SpyProxy: Execution-Based detection of malicious Web content. In: Provos N, ed. Proc. of the 16th USENIX Security. Berkeley: USENIX Association, 2007.

[39] Spitzner L. Honeytokens: The other honeypots. 2011. http://www.symantec.com/connect/articles/honeytokens-other-honeypot

[40] McRae C, McGrew R, Vaughn R. Honeytokens and Web bugs: Developing reactive techniques for investigating phishing scams. Journal of Digital Forensic Practice, 2006,1(3):193−199. [doi: 10.1080/15567280600995857]

[41] Borders K, Zhao X, Prakash A. Siren: Catching evasive malware. In: Orman H, ed. Proc. of the IEEE Symp. on Security and Privacy (S&P 2006). Washington: IEEE Computer Society, 2006. 78−85. [doi: 10.1109/SP.2006.37]

[42] White J. Creating personally identifiable honeytokens. In: Sobh T, ed. Proc. of the Innovations and Advances in Computer Sciences and Engineering. Springer Science+Business Media B.V, 2010. 227−232. [doi: 10.1007/978-90-481-3658-2_39]

[43] White J. Implementing PII honeytokens to mitigate against the threat of malicious insiders. In: Chen H, ed. Proc. of the IEEE Int'l Conf. on Intelligence and Security Informatics. Dallas, 2009. 233. [doi: 10.1109/ISI.2009.5137315]

[44] Chakravarty S, Portokalidis G, Polychronakis M, Keromytis AD. Detecting traffic snooping in Tor using decoys. In: Valdes A, ed. Proc. of the 14th Int'l Symp. on Recent Advances in Intrusion Detection (RAID 2011). Menlo Park, 2011. 222−241. [doi: 10.1007/978-3-642-23644-0_12]

[45] Balas E, Viecco C. Towards a third generation data capture architecture for honeynets. In: Cole J, ed. Proc. of the 2005 IEEE Workshop on Information Assurance and Security (IWIA 2005). 2005. 21−28. [doi: 10.1109/IAW.2005.1495929]

[46] Siles R. Sebek 3: Tracking the attackers. 2011. http://www.symantec.com/connect/articles/sebek-3-tracking-attackers-part-one

[47] The Honeynet Project. Know your enemy: Honeywall CDROM Roo. In: Proc. of the 3rd Generation Technology. The Honeynet Project White Papers, 2011. http://old.honeynet.org/papers/cdrom/roo/

[48] Kaâniche M, Alata E, Nicomette V, Deswarte Y, Dacier M. Empirical analysis and statistical modeling of attack processes based on honeypots. In: Kintala C, ed. Proc. of the 2006 IEEE/IFIP Int'l Conf. on Dependable Systems and Networks (DSN 2006). Philadelphia: IEEE Computer Society Press, 2006. 119−124.

[49] Krasser S, Conti G, Grizzard J, Gribschaw J, Uwen H. Real-Time and forensic network data analysis using animated and coordinated visualization. In: Cole J, ed. Proc. of the 2005 IEEE Workshop on Information Assurance United States Military Academy. West Point, 2005. [doi: 10.1109/IAW.2005.1495932]

[50] Almotairi S, Clark A, Mohay G, Zimmermann J. Characterization of attackers' activities in honeypot traffic using principal component analysis. In: Proc. of the 2008 IFIP Int'l Conf. on Network and Parallel Computing. Washington: IEEE Computer Society, 2008. 147−154. [doi: 10.1109/NPC.2008.82]

[51] Thonnarda O, Dacierb M. A framework for attack patterns' discovery in honeynet data. Digital Investigation, 2008,5:S128−S139.

[52] Zhuge J, Han X, Chen Y, Ye Z, Zou W. Towards high level attack scenario graph through honeynet data correlation analysis. In: Proc. of the 7th IEEE Workshop on Information Assurance (IAW 2006). West Point, 2006. 215−222. [doi: 10.1109/IAW.2006.1652098]

[53] Krawetz N. Anti-Honeypot technology. IEEE Security & Privacy, 2004,2(1):76−79. [doi: 10.1109/MSECP.2004.1264861]

[54] Corey J. Advanced honeypot identification and exploitation. Phrack, 2004,11(63):9. http://www.ouah.org/p63-0x09.txt

[55] Dornseif M, Holz T, Klien C. NoSEBrEak-Attacking honeypots. In: Proc. of the 5th Annual IEEE SMC Information Assurance Workshop. 2004. 123−129. [doi: 10.1109/IAW.2004.1437807]

[56] Oudot L, Holz T. Defeating honeypots: Network issues. Part 1 & Part 2. 2011. http://www.symantec.com/connect/articles/defeating-honeypots-network-issues-part-1, http://www.symantec.com/connect/articles/defeating-honeypots-network-issues-part-2

[57] Holz T, Raynal F. Defeating honeypots: System issues. Part 1 & Part 2. 2011. http://www.symantec.com/connect/articles/defeating-honeypots-system-issues-part-1, http://www.symantec.com/connect/articles/defeating-honeypots-system-issues-part-2

[58] Zou CC, Cunningham R. Honeypot-Aware advanced botnet construction and maintenance. In: Kintala C, ed. Proc. of the 2006 Int'l Conf. on Dependable Systems and Networks (DSN 2006). Philadelphia, 2006. 199−208. [doi: 10.1109/DSN.2006.38]

[59] Wang P, Wu L, Cunningham R, Zou CC. Honeypot detection in advanced botnet attacks. Int'l Journal of Information and Computer Security (IJICS), 2010,4(1):30−51. [doi: 10.1504/IJICS.2010.031858]

[60] Quynh NA, Takefuji Y. Towards an invisible honeypot monitoring system, information security and privacy. Lecture Notes in Computer Science, 2006,4058:111−122. [doi: 10.1007/11780656_10]

[61] Jiang X, Wang X. Out-of-the-Box monitoring of VM-based high-interaction honeypots. In: Christopher K, ed. Proc. of the 10th Int'l Conf. on Recent Advances in Intrusion Detection (RAID 2007). Berlin, Heidelberg: Springer-Verlag, 2007. 198−218. [10.1007/978-3-540-74320-0_11]

[62] Song C, Hay B, Zhuge J. Know your tools: Qebek—Conceal the monitoring. The Honeynet Project Know Your Tools Series White Papers, 2011. http://honeynet.org/papers/KYT_qebek

[63] The Honeynet Project. Know your enemy: Honeynets. The Honeynet Project White Paper, 2011. http://old.honeynet.org/papers/honeynet/

[64] Pouget F, Dacier M, Zimmerman J, Clark A, Mohay G. Internet attack knowledge discovery via clusters and cliques of attack traces. Journal of Information Assurance and Security, 2006,1(1):21−32.

[65] Thonnard O, Dacier M. Actionable knowledge discovery for threats intelligence support using a multi-dimensional data mining methodology. In: Giannotti F, ed. Proc. of the 8th IEEE Int'l Conf. on Data Mining (ICDM 2008). Pisa, 2008. 154−163. [doi: 10.1109/ICDMW.2008.78]

[66] Leita C, Mermoud K, Dacier M. Scriptgen: An automated script generation tool for honeyd. In: Samarati P, ed. Proc. of the 21st Annual Computer Security Applications Conf. (ACSAC 2005). Tucson, 2005. 203−214. [doi: 10.1109/CSAC.2005.49]

[67] Lu TF. Research and implementation of network attack handoff technology in the honeyfarm environment [MS. Thesis]. Beijing: Peking University, 2009 (in Chinese).

[68] Kuwatly I, Sraj M, Masri ZA, Artail H. A dynamic honeypot design for intrusion detection. In: Yousif M, ed. Proc. of the 2004 IEEE/ACS Int'l Conf. on Pervasive Services (ICPS 2004). Washington: IEEE Computer Society, 2004. 95−104. [doi: 10.1109/PERSER.2004.1356776]

[69] Artail H, Safa H, Sraj M, Kuwatly I. A hybrid honeypot framework for improving intrusion detection systems in protecting organizational networks. Computers & Security, 2006,25(4):274−288. [doi: 10.1016/j.cose.2006.02.009]

[70] Anagnostakis KG, Sidiroglou S, Akritidis P, Xinidis K, Markatos E, Keromytis AD. Detecting targeted attacks using shadow honeypots. In: McDanie P, ed. Proc. of the 14th Conf. on USENIX Security Symp. Berkeley: USENIX Association, 2005. 9.

[71] Dagon D, Qin XZ, Gu GF, Lee W. Honeystat: Local worm detection using honeypots. In: Molva R, ed. Proc. of the 7th Int'l Symp. on Recent Advances in Intrusion Detection (RAID 2004). LNCS 3224, 2004. 39−58. [doi: 10.1007/978-3-540-30143-1_3]

[72] Portokalidis G, Bos H. SweetBait: Zero-Hour worm detection and containment using low- and high-interaction honeypots. Elsevier Computer Networks (Special Issue on From Intrusion Detection to Self-Protection), 2007,51(5):1256−1274. [doi: 10.1016/j.comnet.2006.09.005]

[73] Kohlrausch J. Experiences with the NoAH honeynet testbed to detect new Internet worms. In: Günther D, ed. Proc. of the 5th Int'l Conf. on IT Security Incident Management and IT Forensics. Washington: IEEE Computer Society, 2009. 13−26. [doi: 10.1109/IMF.2009.9]

[74] Provos N, Mavrommatis P, Abu M. All your iframes point to us. In: Oorschot PV, ed. Proc. of the 17th USENIX Security Symp. Berkeley: USENIX Association. 2008. 1−15.

[75] Chen KZ, Gu GF, Zhuge JW, Nazario J, Han XH. WebPatrol: Automated collection and replay of Web-based malware scenarios. In: Cheung B, ed. Proc. of the 6th ACM Symp. on Information, Computer and Communications Security (ASIACCS 2011). Hong Kong, New York: ACM Press, 2011. 186−195. [doi: 10.1145/1966913.1966938]

[76] Syamtec Inc. Symantec Internet security threat report — 2010. 2011. http://www.symantec.com/threatreport/topic.jsp?id=threatreport

[77] Freiling F, Holz T, Wicherski G. Botnet tracking: Exploring a root-cause methodology to prevent distributed denial-of-service attacks. In: Samarati P, ed. Proc. of the 10th European Symp. on Research in Computer Security (ESORICS 2005). LNCS 3679, Milan: Springer-Verlag, 2005. 319−335. [doi: 10.1007/11555827_19]

[78] Rajab MA, Zarfoss J, Monrose F, Terzis A. A multifaceted approach to understanding the botnet phenomenon. In: Janeiro R, ed. Proc. of the 6th ACM SIGCOMM Conf. on Internet Measurement (IMC 2006). New York: ACM Press, 2006. 41−52. [doi: 10.1145/1177080.1177086]

[79] Zhuge JW, Holz T, Han XH, Guo JP, Zou W. Characterizing the irc-based botnet phenomenon. Technical Report, Beijing: Peking University & University of Mannheim, 2007.

[80] Stone-Gross B, Cova M, Cavallaro L, Gillbert B, Szydlowski M, Kemmerer R, Kruegel C, Vigna G. Your botnet is my botnet: Analysis of a botnet takeover. In: Al-Shaer E, ed. Proc. of the 16th ACM Conf. on Computer and Communications Security (CCS 2009). Chicago: ACM Press, 2009. 635−647. [doi: 10.1145/1653662.1653738]

[81] Prince MB, Holloway L, Langheinrich E, Dahl BM, Keller A. Understanding how spammers steal your e-mail address: An analysis of the first six months of data from project honeypot. In: Goodman J, ed. Proc. of the 2nd Conf. on Email and Anti-Spam (CEAS 2005). 2005. http://ceas.cc/2005/

[82] Steding-Jessen K, Vijaykumar NL, Montes A. Using low-interaction honeypots to study the abuse of open proxies to send spam. INFOCOMP-Journal of Computer Science, 2008,7(1):44−52.

[83] Levchenko K, Pitsillidis A, Chachra N, *et al*. Click trajectories: End-to-end analysis of the spam value chain. In: Frincke D, ed. Proc. of the 2011 IEEE Symp. on Security and Privacy (Oakland 2011). Oakland: Oakland Press, 2011. 431−446.

[84] Kreibich C, Crowcroft J. Honeycomb: Creating intrusion detection signatures using honeypots. ACM SIGCOMM Computer Communication Review, 2004,34(1):51−56. [doi: 10.1145/972374.972384]

[85] Yegneswaran V, Giffin JT, Barford P, Jha S. An architecture for generating semantics-aware signatures. In: McDaniel P, ed. Proc. of the USENIX Security Symp. Berkeley: USENIX Association, 2005. 97−112.

[86] Mohammed M, Chan HA, Ventura N. Honeycyber: Automated signature generation for zero-day polymorphic worms. In: Young K, ed. Proc. of the 2008 IEEE Military Communications Conf. (MILCOM 2008). Piscataway: IEEE Press, 2008. 1−6. [doi: 10.1109/MILCOM.2008.4753178]

[87] Mohammed MMZE, Chan HA, Ventura N, Hashim M, Bashier E. An automated signature generation approach for polymorphic worms using principal component analysis. Int'l Journal for Information Security Research (IJISR), 2011,1(1):45−52.

[88] Spitzner L. Honeypots: Are they illegal? 2011. http://www.symantec.com/connect/articles/honeypots-are-they-illegal

[89] The Honeynet Project. GSoC 2010 proposed ideas. 2011. http://www.honeynet.org/gsoc2010/ideas

[90] Mulliner C, Liebergeld S, Lange M. HoneyDroid—Creating a smartphone honeypot. In: Butler K, ed. Proc. of the 2011 IEEE Symp. on Security and Privacy (Oakland 2011). Oakland: Oakland Press, 2011.

附中文参考文献:

[5] 刘宝旭,许榕生.主动型安全防护措施——陷阱网络的研究与设计.计算机工程,2002,28(12):9−11.

[6] 曹爱娟,刘宝旭,许榕生.网络陷阱与诱捕防御技术综述.计算机工程,2004,30(9):1−3.

[10] 韩心慧,郭晋鹏,周勇林,诸葛建伟,邹维.僵尸网络活动调查分析.通信学报,2007,28(12):167−172.

[11] 程杰仁,殷建平,刘运,钟经伟.蜜罐及蜜网技术研究进展.计算机研究与发展,2008,45(增刊):375−378.

[22] 陆腾飞,陈志杰,诸葛建伟,韩心慧,邹维.面向蜜场环境的网络攻击流重定向机制的研究与实现.南京邮电大学学报(自然科学版),2009,29(3):14−20.

[35] 陈志杰,宋程昱,韩心慧,诸葛建伟.基于脚本 Opcode 动态插装的 Heapspray 型网页木马检测方法.见:第 2 届信息安全漏洞分析与风险评估会议(VARA 2009).北京,2009.

[67] 陆腾飞.面向蜜场环境的网络攻击迁移技术的研究与实现[硕士学位论文].北京:北京大学,2009.

诸葛建伟(1980－),男,浙江瑞安人,博士,副研究员,CCF 会员,主要研究领域为网络与系统安全.

E-mail: zhugejw@cernet.edu.cn

韩心慧(1969－),男,博士,高级工程师,CCF 会员,主要研究领域为恶意代码检测与防范,网络安全监测技术.

E-mail: hanxinhui@pku.edu.cn

唐勇(1979－),男,博士,副教授,CCF 会员,主要研究领域为网络与系统安全.

E-mail: ytang@nudt.edu.cn

段海新(1972－),男,博士,研究员,博士生导师,主要研究领域为计算机网络安全,网络测量,网络体系结构.

E-mail: duanhx@tsinghua.edu.cn