基于虚拟化技术的分布式蜜网①

吴文洁[1], 葛 昕[2], 胡德敏[3]

[1](上海理工大学 管理学院, 上海 200093)
[2](上海理工大学 研究生部, 上海 200093)
[3](上海理工大学 光电信息与计算机工程学院, 上海 200093)

摘 要: 合理的搭建蜜网可以帮助网络管理员成功地捕获黑客行为, 提升网络安全. 在一个复杂网络环境中要实现分布式部署多个蜜罐需要相当大的人力和物力, 虚拟化技术的应用可以帮助我们很好的解决这一难题. 在实际环境中使用低交互蜜罐在 VMware 中构建了分布式蜜网体系, 采用了 XMPP 技术进行捕获数据共享, 使用 carniwwwhore 框架实现了数据统计和可视化输出, 达到了很好的运行效果.

关键词: 蜜罐; 蜜网; 分布式; 虚拟化; XMPP

Distributed Honeynet Based on Virtualization Technology

WU Wen-Jie[1], GE Xin[2], HU De-Min[3]

[1](Business School of University of Shanghai for Science and Technology, Shanghai 200093, China)
[2](Graduate Departmnet of Shanghai for Science and Technology, Shanghai 200093, China)
[3](School of Opitcal-Electrical and Computer Engineering of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Build honeynet reasonably can help administrator capture the hacker behavior successfully to enhance network security. It will cost much people and material resources to deploy multiple honeypots in a distributed environment, we can use virtualization to do it. This paper use low interaction honeypot build a distributed virtualize honeynet based on Vmware, share attack log using XMPP server, and visualize the data sets with carniwwwhore.

Key words: honeypot; honeynet; distributed; virtualization; XMPP

1 引言

随着网络攻击技术不断发展, 黑客网站教程随处可得, 渗透攻击工具更是日趋专业, 而互联网的开放性和各种操作系统、应用软件的安全漏洞和缺陷, 使得互联网面临越来越大的考验. 要应对这种局面, 传统防火墙很难胜任, 它的优势在于网络层而非应用层, 若实施IDS(入侵检测系统)来捕获攻击行为, 则存在数据流量瓶颈, 难以检测未知攻击, 漏报率和误报率较高等问题. 蜜罐可以较好解决这些问题, 它是一种安全资源, 其存在就是要被扫描、攻击和攻陷, 其价值在于对这些攻击活动进行监视、检测和分析[1]. 而蜜网采用多个蜜罐, 构成一个黑客诱捕网络体系架构, 在保证了网络的高度可控的同时, 可以进行更全面的数据采集和整体分析. 传统蜜网的部署受限于物理设备和网络连接辅助设备, 往往采集点少, 虽然是多点分布式部署, 但多数是部署在同一网段, 意义不大. 分布式虚拟交换机在蜜网中研究应用很少.

本文研究的分布式虚拟蜜网, 使用 VMware 平台的虚拟交换技术真正实现了多 VLAN 的分布式部署. 采用不同形式安装蜜罐来避免故障被复制, 通过部署不同类型蜜罐以博采众长. 在关键的数据采集部分使用 XMPP 协议来进行数据实时提交和共享, 最终在集中的管理端实现数据分析与可视化.

2 虚拟化技术在蜜网中的应用

虚拟化技术在蜜网构建中有着不可替代的优越性, 除了可以节约硬件成本外, 还有以下优点. 第一、虚拟机比物理机更容易修改, 因为整个操作都是在软件层

① 收稿时间:2012-08-27;收到修改稿时间:2012-09-17

面. 第二、虚拟机的状态更容易控制, 可以远程登录到虚拟平台上进行停止、开启操作, 整个系统可进行快照和克隆. 第三、虚拟机的网络连接速度比物理机的速度要快很多, 因为物理机的网络连接要依靠独立的物理网卡. 构建分布式蜜网的两个关键问题都基于虚拟平台, 即创建交换机和蜜罐. 在传统的虚拟蜜网中, 虽有多个虚拟蜜罐, 但受限于底层主机接入, 往往都是使用同一 VLAN 的不同 IP 地址, 这对于分析攻击数据意义并不大, 而创建分布式虚拟交换机, 可实现多 VLAN, 从而真正做到分布式网络环境, 这样不但方便控管, 且节约了网络连接设备的成本.

2.1 分布式虚拟交换机

本文采用了 VMware ESX5 作为虚拟平台, 首先将其连接的交换机端口与上联的核心交换机的对应端口 trunk 互联, 以获得多 VLAN 支持. 在 vCenter Server 中创建分布式虚拟交换机 VDS, 在 VDS 中根据需要创建多个不同 VLAN 并标识 TAG, 各虚拟机网卡口可根据需要划入不同 VLAN, 如图 1.

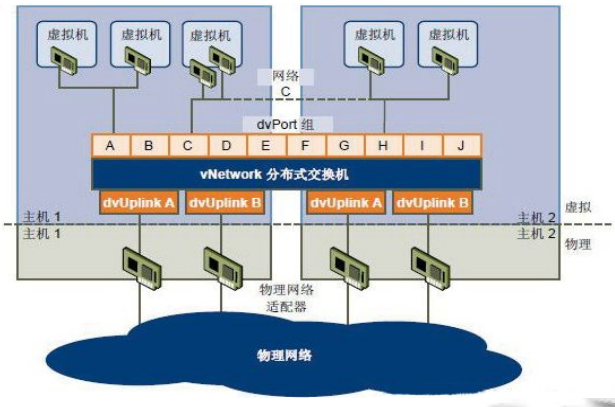

图 1　分布式虚拟交换机

2.2 虚拟蜜罐系统

在虚拟机中部署蜜罐类似于在物理机上进行操作. 为了便于管理, 我们先在 VMware 中创建组 Honeynet, 再在组中创建用于部署蜜罐的虚拟机. 本文在 VMware ESX5 上安装 32 位的 Ubuntu11.04 作为主机系统, 蜜罐采用了 honeypot project[2]中的 dionaea[3], 使用源文件形式编译安装了 2 个蜜罐, 采用 LiveDVD 集成 Dionaea 光盘移植安装了 2 个蜜罐, 作为对 Dionaea 蜜罐捕获弱势的补充, 基于 Ubuntu12.04 编译安装了 1 个 SSH 蜜罐 Kippo[4]. 整个蜜罐创建过程, 相似系统采用先安装后克隆的方法, 并重新配置主机名和网络参数, 最后根据整个蜜网的设计将不同的蜜罐划分入不同的 VLAN 中.

3 蜜网体系的设计与实现

分布式虚拟蜜网的设计框架如图 2. 整个蜜网对于黑客来说, 是一个由多个 VLAN, 提供多种服务的网络组成, 当黑客进行攻击时, 这些服务所在端口会给予正常的反馈, 但蜜罐会记录下攻击数据流. 这些数据最终被提交到一个中心服务器进行分析. 图中 Dionaea1 部分为单个蜜罐的工作流程. Dionaea 采用 C 语言编码, 开放 Python 接口, 可以在不重新编译的情况下添加模块, 且支持 IPv6 和 TLS 协议. 它在网络环境中可以模拟 Internet 中常见的网络服务, 如 http、ftp、mysql、mssql 等, 但 Dionaea 不支持对 SSH 攻击的记录, 因此我们补充了 Kippo 这个专业的 SSH 蜜罐来作为整个蜜网的补充.

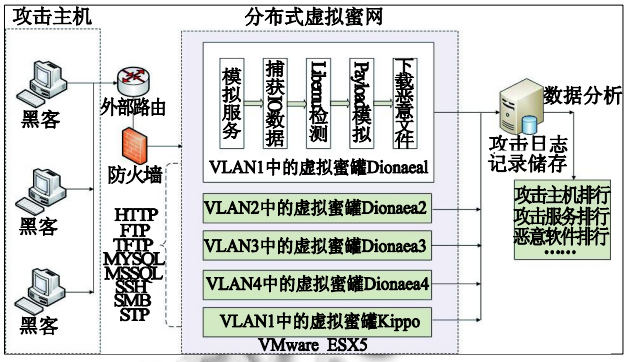

图 2　分布式虚拟蜜网体系架构

设计蜜网体系最重要的三个因素为运行控制、数据捕获和数据分析. 而前期的配置、参数的选定则会直接影响最终捕获的数据类型和分析结果.

3.1 安装配置

本文使用的蜜罐为 Dionaea, 可采用源文件编译安装, 也可采用 PPA[5]进行快速安装,或者使用 honeypot project 提供的 LiveDVD 移植安装, 操作系统可以使用基于 Unix 平台的系统, 建议使用 Ubuntu 或者 Debian. 在蜜罐安装完成之后, 需要进行配置, 主要涉及日志保存的位置, 扩展功能的参数以及模块的开启. 模块主要指启用或禁用攻击记录的相应服务类型(包括 http、https、tftp、ftp、mirror、smb、empap 等服务). 运行 dionaea 时, 默认情况下记录所有的活动, 根据网络实际攻击情况以及可存储日志的空间大

小, 可决定记录那些活动(调试、信息、消息、警告、严重、错误). Kippo 的安装相对简单, 需要注意的是建议将原有运行在 22 端口的 SSH 服务端口重定向到 2222 端口或其他, 而将模拟的 SSH 服务运行在 22 端口以便获得更多的攻击数据.

3.2 运行控制

Dionaea 定位于低交互蜜罐系统, 其为黑客提供的漏洞和缺陷都是模拟出来的, 不会危及底层系统的安全, 因此不同于一般的蜜罐系统, 不需要使用常用的防火墙脚本进行连接限制. 但一般需要开启 SSH 服务方便管理员进行维护管理. Dionaea 运行后, 会对出入蜜罐的数据流进行记录. 记录分为两个大的部分, 一部分是记录模拟服务对黑客攻击行为的反馈信息, 这些信息记录到 sqlite 数据库中, 名为 logsql.sqlite, 包括源和目的 IP, 源和目的端口, 协议类型以及连接时间等. 另一部分为恶意攻击发生时产生的出入数据, 比如下载的恶意二进制文件等, 这些文件以 MD5 哈希运算后命名, 保存在不同的目录下. 如果需要记录黑客使用操作系统的信息, 则需要将 p0f[6] 集成进 dionaea. p0f 是一种被动式操作系统识别工具, 可以独立安装. 在运行 dionaea 之前, 须将 p0f 作为守护进程启动, 同时要赋予用户权限访问 Socket.

3.3 数据捕获

Dionaea 相对于多数低交互蜜罐的一个提高在于 dionaea 采用了 libemu 进行数据捕获. Libemu 采用 GetPC 启发式检测攻击数据流总是否有 shellcode 的存在, 采用基于 x86 的仿真来得到攻击脚本的指令流程图从而确定 shellcode 调用了哪些 API 函数. 对于下载的恶意文件 payload, 会把它放在 libemu 自带的虚拟机中仿真运行, 因此不会感染蜜罐系统本身, 这对保证整个蜜网的可靠性是非常重要的一个因素.

对于捕获到的样本, 默认情况下 dionaea 会提交到三个不同的在线沙箱中, 如果在相应沙箱中进行注册, 则提交样本后会得到相应的反馈信息并保存在数据库中. 通过修改配置文件, 则可将样本提交到预设的 URL 中. 比如可采用 wwwhoney 搭建一个基于 Python 的 CGI 小型 Web 服务器, 接收 dionaea 提交的基于 HTTP 的文件. 在 dionaea 中, 运行自带的 dionaea.py 脚本, 此脚本会检测 dionaea 传感器中是否存在捕获的二进制文件, 有则提交. 在 dionaea 的配置文件中需要添加相应的定义, 标明提交的服务器 URL 地址和端口, 以及用户名和密码.

3.4 数据分析

在分布式蜜网系统中很重要的一点是对分布在不同传感器上的记录数据进行集中管理分析. 在 dionaea 中, 创新性的使用了可扩展通讯和标示协议(XMPP)模块, 使得蜜网体系中实时通信和二进制文件共享成为可能, 而 kippo 同样支持 XMPP, 这就实现了分布式蜜网数据实时提交到同一数据库. 对于数据库里的攻击记录, 我们可采用分析脚本得到文本输出, 也可采用 WEB 工具进行可视化分析, 如图 3.

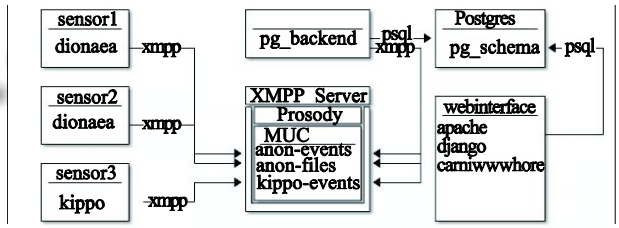

图 3 基于 XMPP 的分析数据框架

在本文中, 使用虚拟平台划分了一个独立的大容量的空间安装 Ubuntu 作为日志服务器操作系统, 使用 Prodosy[7] 来创建 XMPP 服务, 安装 postgresql 作为存储提交数据的数据库. 在 postgresql 中创建 XMPP 数据库和维护此数据库用户名和密码, 并赋予用户操作此数据库的权利. 再使用 pg_schema.sql 创建表. 要实现客户端和服务端的通信共享数据, 需要分别修改配置文件. 在 XMPP 服务器的配置文件中, 需要指定服务器的名称, MUC(多用户聊天室)名称, 可登入的账号信息等. 在 dionaea 的配置文件中, 要在 ihandler 部分启用 logxmpp 功能, 并且在相应位置指定 XMPP 服务器的名称以及登录服务器的用户名和密码. 在 XMPP 服务端, 使用脚本 pg_backend.py 从 XMPP 信道记录攻击数据, 该脚本执行后可登录到指定的 XMPP 服务器, 并对 XML 格式的消息进行解析后再发送到所加入的 IRC 聊天室信道. 该 XML 数据主要包含 dionaea 传感器收集到的攻击信息和恶意的二进制文件等. 需要注意的是对于原有的 sqlite 数据库复制到服务端, 再运行脚本 logsql2postgres.py 进行数据库格式转换. 对于记录在 postgresql 中的数据可在采用 pgadmin 进行管理查看, 也可采用网页形式展示统计数据.

本文在 XMPP 服务器上搭建了可视化网页框架 carniwwwhore[8], 其数据就是实时共享到 postgresql 中

的攻击信息，其网页输出模块已实现对数据的统计分类. 安装此数据统计网页接口，首先需要在系统上安装 Python3，再使用 git 工具下载安装 carniwwwhore，进行设置后，启动 djiango webserver，也可以使用 Apache 来进行 web 发布. 当前的网页框架提供的浏览功能包括总览、协议、传输、攻击、主机、端口、下载文件等 7 个内容的查看，并可根据过滤条件进行时间段的选择查看，如图 4. 对于 kippo 捕获的数据同样可以提交到 carniwwwhore 进行 web 输出，也可以使用 kippo-graph[9]进行更丰富的数据呈现.

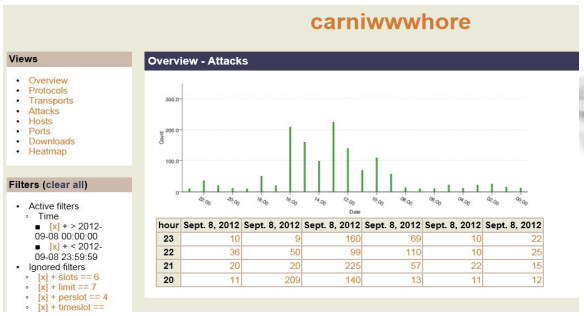

图 4　Carniwwwhore 数据展示页面

4　小结

本文采用 VMware 虚拟机作为基础平台，创建了分布式虚拟交换机，在虚拟硬件上安装 Ubuntu 并部署 dionaea 和 kippo 来做为黑客攻击行为捕获的蜜罐，在四个不同 VLAN 部署了 5 个蜜罐实现分布式蜜网. 在独立的中心服务器中，使用 Prosody 实现 XMPP 服务来实时采集各个分布点提交的捕获数据，在此服务器上安装了 carniwwwhore 作为统计数据的可视化查看工具. 整个蜜网在一个半月的高校校园网络运行中，捕获到的攻击信息共计近 40 万次(因为是暑假期间，故来自校内的攻击很少)，绝大多数都是采用扫描工具批量扫描，其中成功建立连接的所占比例为 12%，针对数据库和 FTP 的口令暴力破解占的比例为 57%，数据库 sa 密码的脆弱性可能导致整个系统权限丢失，因此黑客大量攻击数据库企图获取服务器控制权. 数据还表明一些过时但成功率很高的 exploit 仍然在大量使用. Kippo 捕获的数据表明针对单台蜜罐 SSH 弱口令的尝试平均每天在 5200 次左右，虽然每天有 10 次左右暴力破解得到了弱口令，但是实际数据表明多数黑客在拿到口令后并没有登录系统进行更深入攻击，这可能是因为这些数据多来自初级黑客，他们喜欢使用扫描工具批量扫面但并不精于手工攻击. 整个分布式蜜网中的蜜罐在运行期间没有出现过一次系统自身问题，捕获进程始终工作良好. 但 dionaea 受限于 sqlite 数据库的工作模式为单线程写入数据库，因此并发攻击会被丢弃，可能会造成部分攻击信息遗漏. Dionaea 这种低交互蜜罐对手工攻击行为识别率低，较易被黑客识破. 因此在后续的研究中，我们将会引入高交互蜜罐到同一蜜网体系中，如何进行异构的数据分析和数据共享将是下一步工作的重点.

参考文献

1 Spitzner L. Honeypot-Definitions and Value of Honeypots. 2003-05-29.
2 http://www.projecthoneypot.org/
3 http://dionaea.carnivore.it/
4 http://code.google.com/p/kippo/
5 dionaea:Howto install dionaea using a PPA http://blog.dinotools.de/2011/08/18/dionaea-howto-install-dionaea-using-a-ppa.
6 http://lcamtuf.coredump.cx/p0f.shtml
7 http://prosody.im/
8 http://ore.carnivore.it
9 http://bruteforce.gr/kippo-graph

(上接第 36 页)
障在突发事件后的财政业务仍正常运行.

参考文献

1 上官晓丽,许玉娜,胡啸,等.信息技术一安全技术一信息安全管理实用规则 GB/T22081-2008.北京:中国标准出版社,2008.
2 辛士界.信息安全等级保护定级的方法与应用.软件产业与工程,2011,9(3):40–43.
3 张世琼.财政平台一体化应用系统性能测试.计算机系统应用,2012,21(7):32–37.
4 陈华智,张闻,张华磊.网络安全等级保护实施方案的设计及应用实践.浙江电力,2011,(3):54–57.
5 池仁隆,张超,张春柳.信息系统安全等级保护建设与测评方法简析.软件产业与工程,2012,14(2):44–48.