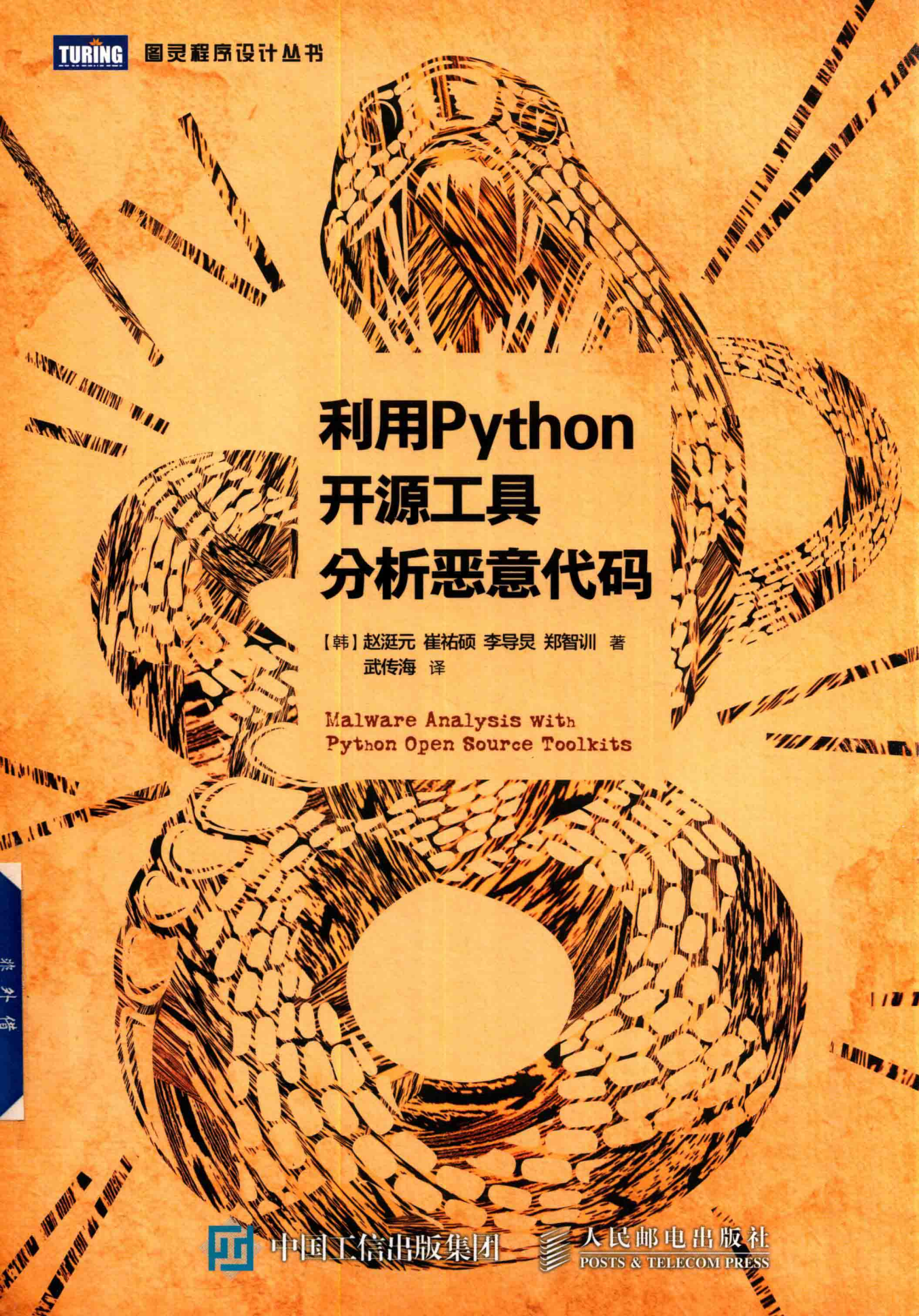
TURING 图灵程序设计丛书



利用Python开源工具分析恶意代码

【韩】赵涎元 崔祐硕 李导炅 郑智训 著
武传海 译

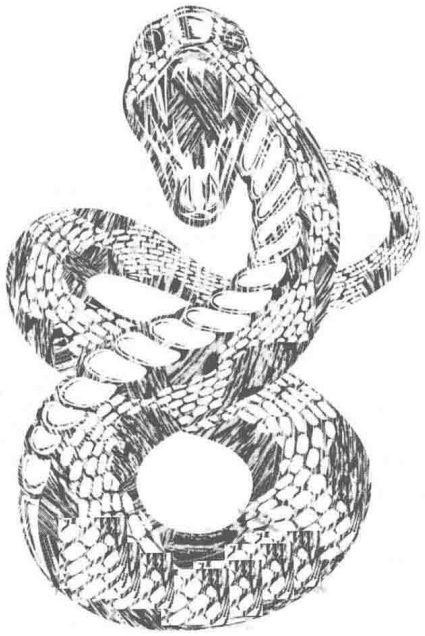
Malware Analysis with Python Open Source Toolkits

中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵程序设计丛书

利用Python开源工具分析恶意代码

【韩】赵涎元 崔祐硕 李导炅 郑智训 著　武传海 译



Malware Analysis with
Python Open Source Toolkits

人民邮电出版社
北京

图书在版编目（CIP）数据

利用Python开源工具分析恶意代码 /（韩）赵涎元等著；武传海译. -- 北京：人民邮电出版社，2018.1
（图灵程序设计丛书）
ISBN 978-7-115-47298-4

Ⅰ. ①利… Ⅱ. ①赵… ②武… Ⅲ. ①软件工具－程序设计 Ⅳ. ①TP311.561

中国版本图书馆CIP数据核字（2017）第284369号

版 权 声 明

内 容 提 要

恶意代码分析过程中，最重要的是掌握恶意代码的特征，此时需要灵活运用线上服务的快速分析数据和主要恶意代码的数据库。本书从应对入侵事故一线业务人员角度出发，介绍了分析恶意代码时的Python等众多开源工具的使用方法，也给出了可以迅速应用于实际业务的解决方案。

◆ 著　[韩]赵涎元　崔祐硕　李导炅　郑智训
译　武传海
责任编辑　陈　曦
责任印制　彭志环
◆ 人民邮电出版社出版发行　北京市丰台区成寿寺路11号
邮编　100164　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京市艺辉印刷有限公司印刷
◆ 开本：787×1092　1/16
印张：31
字数：852千字　2018年1月第1版
印数：1－3 000册　2018年1月北京第1次印刷
著作权合同登记号　图字：01-2016-0513 号

定价：99.00元

读者服务热线：(010)51095186转600　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广登字20170147号

作者序

赵涎元

我的第一份工作是渗透测试顾问，这份工作接触恶意代码的机会并不多，每年也就一两次。通常都是在客户公司应对入侵事故时发现恶意代码，并对其做进一步分析。我做了 5 年渗透测试顾问，现在作为实际业务管理人员，也到了第 5 个年头。在实际管理工作中，为了保护公司财产，我不仅要查找系统漏洞，还要制定入侵事故应对方案、运行系统安全维护方案。这一过程中，我逐渐认识到，不仅要防范来自外部的入侵行为（这些入侵多是通过系统对外提供的服务发动的），还要注意防范用户终端访问恶意站点并遭受恶意代码感染的情形。即使系统安装了杀毒软件，也不能百分之百地保证不会出现恶意代码感染。因此，需要将主要精力放在安全方案中已经发生的事件上，并拿出行之有效的应对策略。

从事实际业务管理开始，在“安全防范项目”社区中，我以数字取证、恶意代码应对为主题，进行了长期研究。一方面与项目组成员分享实际工作中总结的各种注意事项，另一方面也积极从专业分析人员与实际业务人员角度，思考如何将各种开源工具应用于实际工作。本书就诞生于这一过程之中。书中并未堆砌恶意代码分析示例，而将重点放在利用开源工具搭建用于分析恶意代码行为的自动化环境上。本书不是恶意代码分析的终结，反而是个开始。我们目前正在研究大数据管理、个人终端重要进程实时监控与高效防范方法等，希望这些研究成果最终能以另一本书的形式呈现给大家。

写作本书花费了我大量时间与心血，但若没有其他几位作者的共同参与，本书也绝无可能顺利完成。还要感谢安全防范项目组的各位成员，谢谢你们为达成目标而孜孜不倦地努力。最后，感谢我的妻子金惠珍、儿子浩永、女儿熙英，谢谢你们一直在身边支持我、鼓励我。

崔祐硕

众所周知，恶意代码分析由静态分析与动态分析两部分组成，二者最为基本，不可或缺。在实际分析应用中，其难易程度呈几何级数增长。为此，我们需要寻找更高效的方法与技术。

目前，攻击者逐步走向产业化与自动化，随着 IT 技术的发展，大量恶意代码被设计制作出来。现在，如果不是专业的恶意代码分析公司，则无法对恶意代码进行静态与动态分析，至于个人和普通公司则对恶意代码分析几乎更加无所适从。本书的写作目的就是让各位懂得如何使用开源工具，学习常见的恶意代码分析方法，进而在此之上搭建自动化分析系统，帮助大家自动分析工作中遇到的各种恶意代码。

有个词叫“生活黑客”（life hacking），表示为了高效、轻松解决生活中尚未明确或带来不便的问题而提出的创意或技术。面对大量恶意代码及其变种，采用传统方式处理的时间非常有限。为此，分析人员需要尽量将处理过程自动化，并获取自动分析产生的数据，对特定恶意代码做进一步分析。此外，有些人可能不具备编写自动化工具的水平与时间。对此，本书指明了解决这一问题的方向，即借助各种各样的开源工具搭建自动化分析平台，让分析更简便。

我负责撰写本书的 Cuckoo Sandbox 部分。从 0.3 版本开始到现在，每当推出新版本时，我都会用它重新搭建自动化分析环境，进行测试并编写分析报告，进而应用于培训。出现影响恶劣的恶意病毒

时，我就会积极使用自动化分析工具对其进行分析，并编写分析报告。使用 Cuckoo Sandbox 第一次搭建自动化分析环境会比较耗时，其中会用到很多技术，一时很难掌握。但其魅力在于，一旦搭建成功，使用起来就非常简单、方便。使用时，只要通过 Web 页面提交可疑的恶意代码文件，然后耐心等待分析结果即可。对于不太懂 OllyDbg 与 IDA 等工具的朋友，使用 Cuckoo Sandbox 搭建自动化分析环境分析恶意代码是最好的选择。

本书各位作者精心准备的宝贵资料对彼此既是鞭策也是鼓励，大家共同努力，最终编成本书，喜悦之情溢于言表。首先感谢赵涎元先生，谢谢他给我编写本书的机会，是他把大家组织在一起共同完成。还要感谢（株）韩国信息保护教育中心（KISEC）的同事们，谢谢你们支持我，让我可以自由地进行研究。最后，感谢一直支持、鼓励我的家人，谢谢你们陪伴左右，给我无微不至的关怀。

李导炅

一提起恶意代码分析，人们通常首先想到的是“逆向工程”。恶意代码分析过程中，根据不同情况与环境，以及分析者所用的技术，最终采用的分析方法也千差万别。若想在最短时间内做出快速响应，仅靠逆向工程很难获得理想的结果。

显而易见，与不使用各种辅助工具相比，分析代码的同时使用各种高效工具与逆向技术，将大大减少分析时间，其实际耗费的时间不足原来的一半。由此可见，各种辅助分析工具与逆向分析技术拥有密不可分的关系。

有鉴于此，本书将讲解重点放在如何灵活使用各种辅助分析工具上，其中特别介绍了恶意代码分析中常用的各种开源分析工具。使用开源工具的最大好处是，工具源码公开，任何人都可以免费使用，并且可以根据特定的使用环境修改源代码，以定制适合应用环境的最好工具。本书重在介绍如何使用各种开源工具分析恶意代码。

真心希望各位读者借助本书能够积累更深广的知识与经验。最后，感谢安全防范项目组的赵涎元先生及各位成员，在学习恶意代码分析的道路上，他们给了我无尽的帮助与鼓励。

郑智训

之前，我一直在考虑究竟要不要参加安全防范项目组的活动。当时，我通过一个参与活动的朋友第一次知道了这个项目。那时我刚退伍一年左右，期间已经参加信息安全兴趣小组，学习信息安全技术已有大半年时间。由于当时我和其他人一起在做一个项目，深感自身技术水平不足，担心会给别人带来麻烦，一直在犹豫是否参加。犹豫不决间，恰好有一个拜访安全防范项目组负责人赵涎元先生的机会，他为我打消了种种顾虑，最终帮助我顺利加入。

当初解答我的困惑时，赵涎元先生给出了许多宝贵的建议和忠告，这些忠告让我印象深刻，至今言犹在耳。谈到是否加入安全防范项目组时，他讲道：“等你的技术达到一定水平之后再加入也可以，但是，与其自己独自学习修炼，不如跟安全防范项目组成员一起学习，相互促进，共同提高，这样速度会更快一些。发现自己与别人的差距之后，再努力学习，弥补自身不足。”说到参与活动时，他说：“如果把安全防范项目组中涉及内容的级别按照难易程度划分为 1~10 级，有些成员能够处理 7/8 级水平的内容，有些成员则能够应对 5/6 级。如果你的水平不如他们，他们会觉得很正常，就会让你做难易度为 2/3 级的内容。在此基础上学习其他成员所做的项目内容，不断研究并提升自身技术水平，然后再做更高级的内容。”

听完这些忠告后，我在第二天毅然加入安全防范项目组。从那天开始，我仔细整理自己学习的内

容，同时认真学习别人整理的内容，主动向他们请教自己不懂的或者感兴趣的问题，并通过讨论等方式学习了大量知识。然后，在此基础上确定自己感兴趣的主题，伴随着项目的进展，我的技术水平也得到同步提高，最后竟然得到了共同编写本书的机会，实乃人生一大幸事。

我原来认为自身水平不够，连加入安全防范项目组都犹豫不决。而在写作本书的过程中，我得到来自项目组各成员以及项目负责人赵涎元先生的大力帮助，技术水平得到明显提高。当然，在写作过程中，我也感到自己知识水平有限，深深认识到自身的不足，还需要不断努力学习更多知识和技术。

为了让书中包含更好的内容，项目组的所有成员都全力以赴。很抱歉，本人由于水平有限而没能给予大家更大的帮助。但同时也感谢大家信任我，给了我如此珍贵的机会。此外，还要感谢写作期间给予我大力支持的 UOU_Unknown 小组的各位成员。

最后，再次感谢赵涎元先生和本书的其他几位作者，谢谢你们！

站在巨人的肩上

Standing on Shoulders of Giants

iTuring.cn

前言

每天，PC 与移动终端上会出现数以万计的恶意代码，它们大多用于窃取用户的重要个人信息、金融财务信息等。即使是现在，也很难百分之百地准确判断用户使用的终端中究竟安装了哪些恶意代码。个人使用的终端只能依靠杀毒软件，因为恶意代码分析方法并不是每个用户都能掌握的。用户要尽量相信杀毒软件，不运行从可疑网站下载的文件。只有养成这样的好习惯，才能保护好自己的终端。

但是，作为安全管理员，职责就是保护大量用户终端机器的安全。只要一台终端机器感染了恶意病毒，就根本无法预测它会扩散到哪里。出现病毒感染时，最重要的是快速应对。但拿到准确的证据进行分析应对并非易事，所以根本不能只相信为数众多的终端机安全解决方案。为了掌握公司内部员工访问的站点、网站传送的恶意代码、被运行的恶意代码、是否为恶意文件等，相关人员需要学会分析从各个途径得到的日志文件，还要懂得灵活使用线上 / 线下分析服务，快速做好初期应对。

市面上有很多图书讲解恶意代码分析方法，但代码分析大都需要分析者掌握汇编语言、API 编程等各种知识，而学习这些知识需要投入大量时间与精力。本书重点讲解各种开源工具的使用方法，实际业务人员使用这些工具可以轻松搭建恶意代码分析环境。书中部分内容会有一定难度，但都结合源码做出了详细讲解，通过这些讲解，相信各位最终都能牢牢掌握。希望本书能够帮助各位提升对恶意代码的应对能力，同时帮助大家制定合适的解决方案。

本书结构

本书面向对恶意代码分析与开源工具感兴趣的入门者。

第 1 章重点在于理解开源工具与 Python 开发环境。本书内容围绕 Python 开源工具展开，为此，读者必须先理解开发环境，内容涉及使用 Github、Eclipse、插件等搭建 Python 编程环境。

第 2 章主要讲解 Windows 可执行文件 PE 格式。个人用户最常用的操作系统是 Windows 系统，只有理解其中运行的 PE 文件格式，才能把握恶意代码特征，进而将这些特征应用于恶意代码分析。讲解时，一边分析 peframe 开源工具，一边学习 PE 文件的组织结构。

第 3 章讲解恶意代码线上分析服务。分析恶意代码时，为了实现快速应对，一般要使用多种线上分析服务。借助文件分析、域名地址分析、评估系统等多种分析结果，搜索有可能为内部系统安全带来威胁的网站，并快速做出判断。在这些方面，线上分析服务能够提供很大帮助。最具代表性的线上分析服务网站是 VirusTotal，书中将重点讲解，还会涉及一些使用 VirusTotal 服务 API 的开源工具。

第 4 章主要讲解有关恶意代码线下分析服务的内容。第 3 章讲解的线上服务中，关键是搭建分析沙箱。若想构建内部分析环境，则需要线上 / 线下搭建拥有相同功能的环境。这一章详细讲解利用 Cuckoo Sandbox 搭建恶意代码分析环境的方法，并且通过实际恶意代码样本学习如何使用各种分析工具。

第 5 章重点讲解恶意代码分析的具体步骤。以第 4 章自动分析服务分析过的病毒样本为例，介绍针对恶意代码进行手动分析（静态 / 动态分析）的过程。虽然在自动分析中对恶意代码做出了一定判断，但只有经过详细分析才能做出准确判断，并找出应对方法，防止恶意病毒进一步扩散。

第 6 章讲解其他开源恶意代码分析工具的使用方法，主要是一些可以用于恶意代码诊断过程的 Python 开源工具。这一章介绍如何收集、分析、有效管理恶意代码并放入数据库，以及文件分析、网

络包分析、PDF 文件分析等多种环境中使用的分析工具。

第 7 章讲解内存取证分析相关内容。借助内存取证分析技术，可以快速分析并掌握恶意代码执行过程中易挥发的信息以及关联性。应对入侵事故时，重要的是快速找出可疑的恶意代码，并采取有效措施，防止威胁进一步扩大。这一章讲解内存取证开源工具 Volatility 的用法，以及其他以 Volatility 为基础的开源工具的使用方法。

本书特色

本书从应对入侵事故的一线业务人员角度出发，介绍了许多分析恶意代码的 Python 开源工具的使用方法。恶意代码分析过程中，重要的是要掌握恶意代码的特征，目的在于快速分析原因，防止损失进一步扩大。与其逐一分析、应对恶意代码，不如先利用线上 / 线下服务和工具构建防御系统。恶意代码分析中，需要灵活运用线上服务的快速分析数据和主要恶意代码的数据库，以应对多种分析情况。本书讲解了多种开源工具的使用方法，也给出了实际业务中可以灵活运用的解决方案。

读者对象

本书面向对恶意代码分析感兴趣的初学者，以及该领域的一线工作人员。

- 想了解并运用 Python 开源工具的读者
- 想学习多种设备恶意代码分析的读者
- 想搭建恶意代码分析环境的 IT 安全一线业务人员
- 想学习恶意代码入侵应对流程的读者

注意事项

本书以刚入门恶意代码分析技术的读者为对象，详细讲解了在本地 PC 搭建测试环境的方法。书中提到的恶意代码仅供学习、研究使用，未经允许，严禁用于攻击他人或公司的系统，亦禁止随意传播。对于不听劝告而触犯法律的行为，一切责任均由当事人承担。

目录

1

开源软件与 Python 环境

本书内容围绕开源工具展开，所以有必要先了解有关开源工具的内容。首先介绍 Github，它是最活跃的开源代码分享库，然后介绍如何在 Eclipse 开发工具中搭建 Python 环境。由于本书重点不是 Python 编程，所以书中的相关内容并不多。若想从零开始学习，请阅读其他有关 Python 编程的图书。

1.1 关于开源软件

开源软件是指任何人都可以自由使用、复制、发布、修改的软件，并且源代码是公开的。这一定义在不同许可证下有不同解读。此处的“自由”（Free）虽然也有“免费”的含义，但并不表示可以用于商用解决方案并被销售，仅指每个人都可以自由修改并发布开源代码（参考维基百科）。

开发人员创建开源工具有两个原因：其一，现有工具不适合自己的需要，使用不便；其二，现有工具包含太多自己用不到的功能，这些功能占用了大量 PC 资源。

假设开发人员在上述两种原因的驱使下开始开发软件工具，并将其开源，分享给其他人。这样一来，开发人员就无法获得任何金钱回报吗？如果自己投入的时间没有获得实质性的回报，那么这样的分享也不会持续很久。

虽然可以免费使用开源工具，但具体也要看其遵从的许可证。在某些许可证下，以营利为目的的企业在使用时必须支付一定费用。与现有程序相比，如果开源工具易用且免费，那么用户就会自然而然地选用它们。借助口碑与文案宣传，开源工具会迅速传播开来，从而在短时间内获得巨大的下载量。由于开源软件倡导开源精神，所以目前没有将其销售给企业的案例。但这些开源软件往往拥有大量用户，比如想学习相关技术的企业、教育机构，开源软件作者可以通过向这些用户提供培训或咨询服务获得收益。

此外，随着开源软件知名度的提高，软件作者将会获得在各种技术大会上进行讲演的机会，这有助于提升其身价。很多企业为了吸引软件作者而煞费苦心，即使最终没能将其招为自身员工，也会聘其担任技术顾问，并给予丰厚的待遇。

开源软件作者有很多机会招收大量优秀人才，成立创业公司，并获得事业上的成功。这样的事例屡见不鲜，最具代表性的有 Linux 内核、Android、Chrome、Firefox、Apache Hadoop 等，它们都从开源项目做起，最终获得巨大成功。

如果管理人员熟悉开源软件

本书主要介绍基于 Python 开发的开源工具，所选的开源工具都很有代表性，它们对维护系统安全很有帮助。如果你认为：“我是实际业务负责人，有必要熟悉编程吗？从外部引入现有的解决方案，并与 BMT[①] 进行比较，不就行了吗？”不妨三思，毕竟，没有一个解决方案可以完全覆盖安全业务的各个方面。

最好的解决方案应该物超所值（支付的费用不仅包含解决方案本身的价格，还包含后期维护费用），它应该能够最大限度地减少服务提供过程中以及员工使用 PC 过程中的诸多不便。验证这些问题时，需要涉及大量技术内容。比如，随着 APT 攻击盛行，出现了许多防御“零日漏洞”的解决方案。对这些方案进行检验时，只阅读厂商提供的说明文档，观察能否有效防御就可以了吗？还有，只要管理人员能够轻松运行就可以放心引进了吗？

① BMT 是 Benchmark Test 或 Benchmarking Test 的缩略语，中文译为“基准测试”。与常规的性能测试不同，BMT 测试中要先有个实际存在的比较对象，通过对硬件或软件性能进行比较分析，做出最终评价。（参考维基百科）