安全操作系统发展概况

3170103240 张佳瑶

计算机系统安全可以由物理安全、操作系统安全、数据安全、应用安全等四个方面构成。物理安全受到自然灾难、断电等不可控因素的影响，而操作系统安全、数据安全和应用都和网络安全息息相关。为了实现计算机机密性、完整性、可用性的安全目标，安全操作系统的研究开启。安全操作系统在自主访问控制、强制访问控制、标记、身份鉴别、客体重用、审计、数据完整性、隐蔽信道分析、可信路径、可信恢复等十个方面满足相应的安全技术要求。从技术层考虑，相关研究人员设计了防火墙、入侵检测、可信计算、身份认证、访问控制、安全协议等来维护计算机系统的安全。从发展历程考虑，安全操作系统的发展经历了奠基时期（foundation period）、食谱时期（cookbook period）、多策略时期（multi-policy period）和动态策略时期（dynamic-policy period）四个阶段。

一、 奠基时期

奠基时期，安全操作系统经历了从无到有的探索过程，安全操作系统的基本思想、理论、技术和方法逐步建立。

1967 年，计算机资源共享系统的安全控制问题引起了美国国防部的高度重视，美国国防科学部旗下的计算机安全特别部队的组建拉开了操作系统安全研究的序幕。1969 年，世界上第一个安全操作系统 Adept-50 研究成果发表。它可以实际投入使用，运行于 IBM/360 硬件平台。在该系统中，可以为客体标上敏感级别属性：对于读操作，不允许信息的敏感级别高于用户的安全级别；对于写操作，在授权情况下，允许使信息从高敏感级别移向低敏感级别。

这一时期的代表思想有访问控制矩阵、引用监控机、BLP 模型，隐通道等，这一时期的代表系统有 Adept-50、Multics 等。

a) 访问控制矩阵

1969 年，B. W. Lampson 首次运用了主体、客体和访问控制矩阵的思想，对访问控制问题进行了抽象。在一个操作系统中每个实体组件都必须是主体或者必须是客体，或者即是主体又是客体。客体是一个被动的实体。在操作系统中，客体可以是按照一定格式存储在一定记录介质上的数据信息，也可以是操作系统中的进程。主体是引起信息在客体之间的流动的实体。通常，这些实体指人、进程或设备等，一般是代表用户执行操作的进程。如编辑一个文件。编辑进程是存取文件的主体，而文件是客体。

b) 引用监控机

引用监控机、引用验证机制、安全核和安全建模等思想在研究系统资源受控共享问题的背景下产生。

引用监控机由安德森 Anderson 提出是一个抽象的概念，它用于仲裁主体对客体的访问。Morri Gasser 从实现的角度对引用监控机进行阐述。他的解释是：根据引用监控机中体现的安全策略和存储在数据库中的访问控制信息约束和监视所有主体对客体的访问，且会有相关的审计文件记录。

而引用验证机制是一种安全机制，满足三个原则：必须具有自我保护能力，必须总是处于活跃状态，必须设计得足够小，以利于分析和测试。三个原则保证了引用监控机制能够保持自身完整性，仲裁程序对资源的所有引用，保证能够验证正确和符合要求。

引用监控器的实现被叫做安全内核。1972 年，Roger Schell 提出的安全内核的概念被认为是引用验证机制的一种类型，安全内核是指操作系统与安全性实现相关的部分。

c) BLP 模型

BLP 模型定义了信息的安全级，支持下读上写。BLP 模型有两种安全策略：自主访问控制和强制访问控制。BLP 模型存在很多缺陷。BLP 模型本身只考虑了保护信息的机密性 对于数据的完整性这一部分并没有做到处理。模型中所有的主体和客体都是单级的，而实际中系统的一些实体需要多级特征。人们对 BLP 模型进行改进，提出了很多改善了的模型，其中有有 Biba 模型，用于填补数据完整性的维护。

d) 隐通道

操作系统中有发送方进程和接收方进程，进程之间通过通信机制通信，例如共享内存、消息队列。对于本意不是用来通讯但实际上被用来通讯的机制没有相关的机制进行控制。比如，一个程序可以通过改变其对系统负载的影响来向另一个程序传递信息。

1973 年，Lampson 首次定义隐通道：如果一个通信信道既不是设计于通信的，也不是有意与传递信息的，则称该通信信道是隐蔽的。他提出的定义并没有说明隐通道属于有害的还是无害。1987 年，Tsai 等给出的定义：给定一个强制安全策略模型 M 及其在一个操作系统中的解释 I(M)，I(M)中两个主体 $I(S_h)$和 $I(S_l)$之间的通信是隐蔽的，当且仅当模型中 M 中的对应的主体 S_h 和 S_l 之间的任何通信都是非法的。Tsai 等人给出的定义已经明确说认为隐通道是有害的。

在实施了强制访问控制的多级安全操作系统中，特洛伊木马只有利用隐蔽通道才能将信息传送给低安全级进程（木马主人）。BLP 模型也受到隐蔽通道的威胁。

二、 食谱时期

1983 年美国 TCSEC 标准颁布，人们以 TCSEC 为蓝本研制安全操作系统，就像按照食谱做菜一样。因为封面是橘色的，所以这个标准也被称为橘皮书。随后也有一系列的标准诞生，如加拿大 CTCPEC 标准、欧洲 ITSEC 标准，最后提出了一个国际标准 ISO/IEC15408。

TCSEC 对计算机系统做了一个等级划分，主要分为 ABCD 四大类，七个等级。D 类代表没有提供任何的安全机制。C1 是自主访问控制级。C2 要实现强制访问控制。B1 级被称为标记型安全保护级。B2 被称为结构化的保护级。B3 级别希望能够建立一个引用监控器。A 级要对整个操作系统内核的安全性做一个形式化的规范，一个形式化的验证。

这一时期，安全操作系统的主要特点是单一政策，代表思想有可信计算机。可信计算基是指把计算机系统中与安全保护有关的功能找出来和系统中的其他功能分离独立，防止遭到破坏，这样独立的结果就是 TCB。如果针对某个特定的目的，实体的行为与预期的行为想符，则称针对这个目的，该实体是可信的。建立可信计算环境是计算机及网络技术发展的要求。“可信计算”可以从几方面来理解：用户的身份认证是对使用者的信任，平台硬件配置的正确性是对使用者对平台运行环境的信任，应用程序的完整性和合法性是对应用程序的可信的验证，平台之间的可验证性是网络环境下平台之间的相互信任。

这一时期的代表系统有安全 Xenix。

三、 多政策时期

多政策时期始于 1993 年，这个时期的特点是人们超越 TCSEC 的范围，在安全操作系统中实现多种安全政策。美国国防部推出了新的安全体系结构 DGSA，支持多级安全政策。DTOS 原型系统就是这个时候提出的。

四、 动态政策时期

动态政策时期始于 1999 年 Flask 系统诞生，这个时期的特点是使安全操作系统支持多种安全政策的动态变化。Flask 系统来源于之前的 DTOS 原型，支持动态安全策略，将策略和机制分离。由两部分组成。一类是客体管理器，实施安全政策的判定结果；一类是安全服务器，制定访问控制策略，做出安全政策的判定。

身份认证、访问控制和审计是三个核心模块，在 windows 安全中起着非常重要的作用。企业版 Windows10 使用 Microsoft Hyper-V 来提高安全性，他是一个应用硬件辅助虚拟化的技术。VBS 使用一种白名单机制，仅允许受信任的应用程序启动，将最重要的服务以及数据和操作系统中的其他组件隔离。

五、 总结

安全操作系统每一个阶段的提出和发展都是因为应用过程中出现了新的需求，研究人员从策略和机制方面不断推陈出新，推动计算机安全不断向前突破。安全操作系统的“安全”贯彻计算机使用的整个过程的方方面面，从登陆验证到访问控制等等。计算机安全不仅仅需要操作系统层面的安全，还需要应用安全、数据安全、物理安全等方面，都需要作出努力。