装备科技译著出版基金

杨林 译

动态目标防御(II)

——博弈论与对抗模型的应用

Moving Target Defense II

Application of Game Theory and Adversarial Modeling

[美] Sushil Jajodia Anup K. Ghosh V.S.Subrahmanian Vipin Swarup Cliff Wang X. Sean Wang 编著

国防工业出版社 National Defense Industry Press

Springer

装备科技译著出版基金

动态目标防御(Ⅱ)

——博弈论与对抗模型的应用

Moving Target Defense Ⅱ——Application of Game Theory and Adversarial Modeling

Sushil Jajodia　Anup K. Ghosh

[美]　V. S. Subrahmanian　Vipin Swarup　编著

Cliff Wang　X. Sean Wang

杨林　译

国防工业出版社

·北京·

著作权合同登记　图字:军 2014 -059 号

图书在版编目(CIP)数据

动态目标防御(Ⅱ):博弈论与对抗模型的应用/(美)贾乔迪亚(Jajodia, S.)等编著;杨林译. —北京:国防工业出版社,2014.12

书名原文:Moving target defense Ⅱ:application of game theory and adversarial modeling

ISBN 978-7-118-09775-7

Ⅰ.①动… Ⅱ.①贾… ②杨… Ⅲ.①计算机网络 - 安全技术 - 研究 Ⅳ.①TP393.08

中国版本图书馆 CIP 数据核字(2014)第 289244 号

※

国防工业出版社出版发行
(北京市海淀区紫竹院南路 23 号　邮政编码 100048)
北京嘉恒彩色印刷有限责任公司
新华书店经售
*
开本 710×1000　1/16　**印张** 11¾　**字数** 218 千字
2014 年 12 月第 1 版第 1 次印刷　**印数** 1—2500 册　**定价** 93.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)88540777　　发行邮购:(010)88540776
发行传真:(010)88540755　　发行业务:(010)88540717

译 者 序

动态目标防御是在部署、运行网络和系统时，通过有效降低这些网络和系统的确定性、相似性和静态性，增加其随机性或减少其可预见性来构建持续变化、不相似、不确定的网络和系统，以极大增加攻击者攻击成本的一种全新的网络安全思路和方法。动态目标防御通过构建能快速、自动改变一个或多个系统属性与代码的系统，防御者能以可控的方式进行动态变化，使攻击者难以有足够时间发现其脆弱性，系统攻击表面对攻击者而言是不可预测的，从而极大地提升防御者的防御能力，从被动安全态势向主动防御态势转变，改变网络攻防“易攻难守”的不对称局面。同时，这也意味着网络空间安全不仅仅只在于安全系统强度的提高，还在于网络空间自身免疫力的增强。

动态目标防御的思想在军事领域自古就有，《孙子兵法》云“兵者，诡道也”，意思是“用兵之道，在于千变万化，出其不意”，但是将“变”的思想运用于网络空间安全则是近几年才提出来的。2008 年 1 月，布什总统签署 54 号国家安全总统令/23 号国土安全总统令（NSPD－54/HSPD－3），提出了《国家综合网络空间安全倡议》（CNCI）。该倡议要求确定并发展“超前”的技术、战略和计划，寻求革命性网络空间安全解决方案。在 CNCI 的基础上，2009 年 5 月，奥巴马政府发布了《网络空间政策评审》，确定了美国政府应实施网络空间“改变游戏规则”的安全研发思路。在这些战略计划的引领下，美国政府以 CNCI 为总纲，向学术界和工业界广泛征求解决网络空间安全问题的建议，收集了 238 条反馈意见，经过反复讨论与提炼，2009 年 8 月在“国家网络飞跃年”峰会上归纳为五条“改变游戏规则”的概念；后又经过两年研究，于 2011 年 12 月，美国国家科学技术委员会发布《可信网络空间：联邦网络安全研发战略规划》，明确指出“针对网络空间所面临的现实和潜在威胁”，要突破传统思路，发展“改变游戏规则”的革命性技术，确定了四个能“改变游戏规则”的研发主题，动态目标防御就是其中之一。

形成此重大变化的原因之一是近年来先进持续性威胁（APT）的发展。确定性、相似性、静止性及漏洞的持续性是现有网络信息系统的致命安全缺陷，这些缺陷导致当前网络信息系统始终处于被动挨打的局面，找不尽的安全漏洞，打不完的安全补丁，只好一味地追求防卫系统的强度。但是，再厚的防卫墙，也经不起攻击者长期地观察、分析和反复攻击。动态目标防御的革命性和创新性就在于一反常态，由阵地保卫战改为运动战或游击战，不给攻击者进行观察、分析的

时间，更不容其反复攻击，大大提高了其攻击难度及必须付出的代价。显然，这是防卫策略的大转变和游戏规则的大改变。

原因之二是近年来的技术发展，包括：多核处理、云计算、加密技术、网络标准、系统管理、指令集和地址空间布局随机化、虚拟化桌面的应用和虚拟化技术成本的不断降低，以及借鉴生物免疫系统或生物进化的弹性和防御方法等关键支撑技术的发展使动态目标防御成为可能。

近几年，动态目标防御已取得很多新进展，包括变形网络、自适应计算机网络、自清洗网络，以及利用网络机动提高网络弹性等。未来，动态目标防御还需要重点开展以下研究工作：①能够科学证明动态目标机理的理论模型及方法；②能够提高动态目标有效性的机制；③能够科学度量动态目标机制有效性的理论方法；④描述漏洞空间的方法，衡量系统随机化对攻击者利用这些漏洞能力的影响；⑤理解单个组件随机化对复杂系统性能的影响，尤其是对系统规避威胁能力的影响；⑥动态目标的同步和管理方法，以及如何保持与原有系统和机制的协同性；⑦虚拟环境中动态目标系统自身的安全和弹性技术；⑧自动改变和管理网络及系统结构的技术；⑨在对抗环境中，目标实时自适应变异的方法等。

由于动态目标防御的研究涉及面广、难度较大，当前还不很成熟，系统性也不够，因此，作者采取汇编形式，分为Ⅰ卷、Ⅱ卷（即本书），汇集了近年来世界一流研究人员在该领域的最新理论和技术研究成果，包括美国惠普实验室、卡内基梅隆大学、弗吉尼亚大学、哥伦比亚大学、加利福尼亚大学、麻省理工学院、南加州大学、乔治梅森大学、北卡罗来纳大学等研究机构的相关人员。本书Ⅰ、Ⅱ卷的详细内容见每卷的内容简介。编著者们旨在以最快方式将其初步研究成果呈现给读者，以帮助读者了解动态目标防御的基础理论及技术。全书内容翔实、视角新颖、观点富有启发性，是一本不可多得的参考书。

希望本书的出版有助于我国信息安全、通信和网络对抗领域的研究人员准确把握将当前“死”网络变成未来“活”网络的网络防御技术发展方向；促进具有动态配置能力和自身免疫能力的新型网络系统的研究；推动网络空间主动防御体系的构建等。

本书的翻译工作由杨林完成，杨林、卿昱、张晓玉、马琳茹对全书进行了统稿和审订，周文、李振邦、田永春、李大双、刘杰、景中源、晋钢、陈岳兵、高洪博、唐鹏飞参与了全书的校对工作。在本书的翻译和出版过程中，国防工业出版社的王晓光老师给予了大量的帮助，在此表示衷心的感谢。

由于译者水平有限，译文中难免存在疏漏与不妥之处，敬请读者批评指正。

译 者

2014 年 4 月

序

在抵御计算机入侵方面最让人感到苦恼的问题之一是，似乎永远都存在可被利用的软件漏洞（虽然我们在安全软件研发实践中已取得了重要进步）。每月至少有一次（如周二补丁日），主要软件供应商会发布补丁，以修补已发布的软件代码中发现的漏洞，这些补丁往往是在漏洞已知和被利用之后发布的，有时是几个月甚至几年。在当前的配置系统中，攻击者首先是对静态目标进行研究，找出漏洞，然后寻求时机利用这些漏洞，以获取权限访问他人的计算机和网络，直到我们发现漏洞被利用，并找到漏洞，发布补丁，然后得到广泛应用。这样的动态过程显然对攻击者更为有利，而不是防御者，因为攻击者只需要找到一个可以利用的程序缺陷（bug），而防御者却必须要确保没有任何程序缺陷。此外，攻击者拥有充足的时间分析软件代码，而防御者却不清楚攻击者何时会发动攻击。最后，一旦漏洞利用或漏洞被发现，防御者通常只能阻止漏洞利用，这为攻击者提供了利用零日（zero - day）漏洞的天然优势。

在这样的背景下，我们提出了动态目标防御（MTD）研究主题，用以平衡防御者对攻击者的博弈。动态目标防御的基本概念是动态改变防御系统的攻击面，从而消除对手能够离线研究目标系统，以及在攻击时找到可以利用的漏洞的优势。尽管漏洞已经暴露，动态目标防御系统也能提供概率保护，只要在攻击时对手无法预知漏洞。动态目标防御已被列入美国白宫网络安全研究与发展战略规划的四大关键领域之一。

在《动态目标防御》Ⅰ卷中，我们介绍了动态目标防御的基础、基于软件变化的动态目标防御方法，以及基于网络和软件栈配置的动态目标防御方法。在动态目标防御Ⅱ卷中，一群世界一流研究人员介绍了构建和分析动态目标防御系统的博弈论、赛博机动以及软件变化方法等。

致谢

我们非常感谢对本书做出贡献的各位同仁。特别是，对各位作者的贡献致以诚挚的谢意。我们由衷感谢 Springer 资深出版编辑 Susan Lagerstrom – Fife，以及编辑助理 Jennifer Maurer，对本项目的支持。我们还要感谢美国陆军研究办公室的经费支持，基金编号是 W911NF – 10 – 1 – 0470。

Sushil Jajodia
Anup K. Ghosh
V. S. Subrahmanian
Vipin Swarup
Cliff Wang
X. Sean Wang

费尔法克斯　弗吉尼亚

关于本书

本书介绍了动态目标防御所面临的一些挑战，提出了基于博弈论方法、基于网络的赛博机动以及软件变化的富有前景的解决方法。

本书分为三部分，第一部分为第 1 ~ 4 章。在第 1 章，Manadhata 探讨了在动态目标防御时运用攻击面转移法。本章阐释了转移软件系统攻击面的概念，引入量化转移的方法，并提出了确定最佳动态目标防御策略的博弈论方法。在第 2 章，Jain 等人描述了将博弈论运用到安全领域所面临的挑战性现实问题，并阐述了解决和理解大规模现实安全博弈特征的关键思路和算法，以及该领域有待研究的一些关键问题和已部署系统取得初步成功的案例。在第 3 章，Bilar 等人详细研究了 Conficker 蠕虫病毒及其相关防御措施之间的协同演变，并建立了说明这种协同演变的量化模型。研究结果充分证明，攻击者与防御者互为移动目标，因为任意一方的移动都会引起另一方的移动。在第 4 章，Gonzalez 总结了个人行为计算模型的研究现状，并说明了将此类模型扩展到二人（即防御者与攻击者）在非合作动态赛博安全环境下所面临的挑战及潜力。

第二部分为第 5 章和第 6 章，主要探讨网络环境下的赛博机动。在第 5 章，Torrieri 等人阐述了研究内外部干扰和其他攻击时存在的问题及挑战。借助赛博机动，提出了解决此类问题的基本框架。机动密钥作为扩频密钥，对更高级别的网络密钥进行了补充，并提供了抵御和应对内外攻击的方法。在第 6 章，Yackoski 等人提出了一种基于 IPv6 的网络体系结构，这种体系结构整合了加密型较强的动态特性，限制攻击者在网络中制定攻击计划、传播攻击和通信的能力。

第三部分为第 7 ~ 9 章，主要说明了基于软件变化的 MTD 方法。在第 7 章，Le Goues 等人对螺旋式变形防护系统做了详细的介绍，这种防护方法可利用新型的演变算法自动修复漏洞，持续不断地从时间和空间两个维度转移程序的攻击面。攻击面转移与缩小之间的交互作用，引起程序自动演变，产生新的变体，并随着时间的推移，不断提升程序的质量。在第 8 章，Jackson 等人回顾了其基于编译器的自动编码多态化技术，深入分析了该技术的性能，并通过全系统栈多态化证明了该技术的实际应用潜力。在第 9 章，Pappas 等人说明了一种可直接应用于第三方软件的软件多态化技术，即就地代码随机化。他们展示了就地代码随机化技术是如何阻止 Windows 7 应用程序固有漏洞被利用的，并提供了针对面向返回编程（ROP）攻击的概率性保护。

目　录

第 1 章　攻击面转移的博弈论方法

Pratyusa K · Manadhata①

【摘要】 软件系统的攻击面，是指系统遭受攻击的一系列途径。在前期研究中，已提出一种降低软件系统安全风险的攻击面度量及减小攻击面的方法（曼达塔. 攻击面度量标准：[博士论文]. 卡耐基梅隆大学，2008；曼达塔，周以真. IEEE 软件工程汇刊. 2011，37：371－386）。本章尝试将攻击面转移运用到动态目标防御方法中。首先，定义攻击面转移的概念，并提出一种量化转移的方法；然后，将动态目标防御方法视为安全性与可用性之间的一种权衡，并提出二人随机博弈模型，以确定一种最佳的动态目标防御策略。利用这种博弈论方法，系统防御者可以最佳方式转移并减小系统的攻击面。

1.1　引　　言

在前期研究中，已给出软件系统攻击面的概念，并将系统攻击面作为系统安全的一个度量指标[5,6]。直观地讲，系统攻击面是指对手进入系统并可能造成损害的一系列途径。因此，攻击面越大，系统越不安全；减小系统攻击面，就可以降低系统的安全风险。同时，还引入攻击面度量指标，以系统的方法对系统的攻击面进行度量。

以前主要关注如何将攻击面度量运用到软件开发过程中，在此提出一种减小攻击面的方法，做为软件行业为降低安全风险而采用传统的提高代码质量方法的补充。提高代码质量是为了减少软件安全漏洞，但事实上，编写一款大型、复杂、没有安全漏洞的软件仍然是非常困难的。软件供应商们不得不承认这一严峻现实，即开发的软件中存在着很多已知和未来可能发现的漏洞，并且随时间推移，这些漏洞中很多将被发现和利用。供应商们可以通过减小攻击面来降低与漏洞利用有关的风险。减小攻击面可以增加利用漏洞的难度，降低攻击的破坏程度，由此降低安全风险。

本章关注如何将攻击面度量应用于动态目标防御中。为此，提出了一种系

① P. K. Manadhata(✉)。美国新泽西州普林斯顿沃恩大道 5 号 301 室，惠普实验室，邮编：08854。e－mail：manadhata@ cmu. edu。

统防御者，如系统管理员，持续保护系统免遭攻击的想定。动态目标防御是一种全新的保护方式，防御者要不断转移系统攻击面，以增加攻击者利用系统漏洞的难度[1]。如图 1.1 所示，若防御者转移系统的攻击面，则原本有效的攻击，如攻击 1，不再有效。为此，攻击者需要付出更多才能使原来的攻击重新有效或寻求新的攻击途径，如攻击 4。我们将防御者与进攻者之间的这种关系视为一种二人博弈，并依此来探讨如何将博弈论运用到攻击面的转移中。

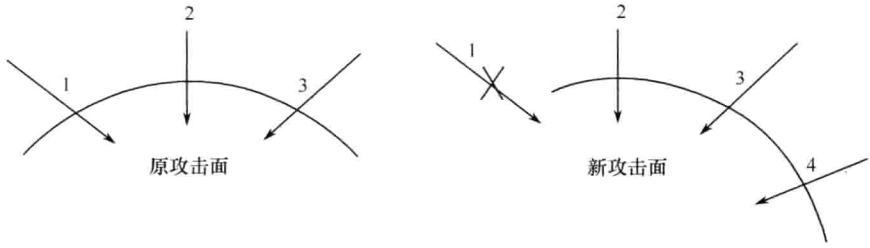

图 1.1　如果转移系统的攻击面，则原有攻击（如攻击 1）将不再有效。但转移后，系统可能会遭受新的攻击（如攻击 4）

后续内容安排：在 1.2 节简要论述攻击面的度量方法；1.3 节给出攻击面转移的概念，并讨论如何将攻击面转移运用到动态目标防御中；1.4 节研究将博弈论方法运用到攻击面转移中，并在安全性与可用性之间寻求最佳平衡；1.5 节对本章进行了总结。

1.2　攻击面的度量

根据经验，针对系统的许多攻击，如利用缓冲区溢出漏洞发起的攻击，往往出现在运行环境向系统发送数据的过程中。同理，针对系统的许多其他攻击，如符号链接攻击，则发生在系统向环境发送数据的过程中。在这两类攻击中，攻击者都利用套接字等系统通道连接系统，调用系统函数（如 API），并向系统发送数据项，比如，输入字符串，或从系统接收数据项。攻击者还可使用共享的持久数据项，间接向系统发送数据；同样，攻击者也可藉此间接接收系统数据。因此，攻击者可使用系统函数、通道和系统环境中的数据项攻击系统。我们把系统函数、通道和数据项合称为系统资源，并以此定义系统的攻击面。

1.2.1　攻击面的定义

并非所有资源都是攻击面的一部分，只有攻击者利用某种资源攻击系统时，该资源才是攻击面的一部分。为此，引入入口点和出口点框架，以便识别相关资源。

1.2.1.1 入口点

系统代码库包括一系列函数，如 API 函数等。每个函数获取输入变量，然后输出结果。凡接收系统环境数据项的系统函数即为该系统的入口点。例如，接收用户数据的函数，或读取配置文件的函数，均为入口点。就系统 s 而言，若函数 m 具有以下任意特征，即为该系统的直接入口点：① s 环境的用户或系统调用 m 并将输入的数据项发送给 m；② m 读取持久数据项；③ m 调用 s 环境的系统 API，并接收作为返回结果的数据项。间接入口点是指一种能接收直接入口点数据项的函数。

1.2.1.2 出口点

凡向系统环境发送数据项的系统函数即为该系统的出口点。例如，写入日志文件的函数就是一个出口点。对于系统 s 而言，若其函数 m 具有以下特征，即为该系统的直接出口点：① s 环境用户或系统调用 m，并接收 m 返回的数据项；② m 写入一个持久数据项；③ m 调用 s 环境的系统 API，并将输入的数据项发送该 API。间接出口点是指一种向直接出口点发送数据的函数。

1.2.1.3 通道

每个系统都有一系列通道，它们是用户或其他系统与系统交流的途径，例如 TCP/UDP 套接字、RPC 端点以及命名管道。攻击者利用系统通道与之相连，并调用系统函数。因此，通道是攻击系统的另一个基本途径。

1.2.1.4 不可信数据项

攻击者可采用持久数据项，间接向系统发送数据，或间接接收系统发出的数据。持久数据项的类型包括各种文件、cookies、数据库记录和注册表项。攻击者写入文件后，系统就可读取这一文件。同样，系统写入文件之后，攻击者也可读取该文件。因此，持久数据项是攻击系统的第三个基本途径。

1.2.1.5 攻击面的定义

系统的攻击面是指攻击者可用于发动攻击的系统资源的子集。根据这一定义，攻击者可以利用入口点和出口点之集 M、通道集 C 以及不可信数据项集 I，向系统发送数据或从系统获取数据，从而攻击该系统。因此，M、C 和 I 即为攻击面相关资源子集，给定系统 s 及其环境，定义 s 的攻击面为三元组 $\langle M, C, I\rangle$。

1.2.2 攻击面度量方法

计算攻击面的资源量是一种度量系统攻击面的最自然的方法。这种方法对

所有资源赋以相同的权重，但由于攻击者利用这些资源发动攻击的可能性并不相等，因此，这种方法存在一定的缺陷。估算资源对系统攻击面的作用，用“破坏潜力与攻击成本的比率”表示。其中：破坏潜力是指攻击者利用资源对系统进行攻击而造成破坏的程度；攻击成本是指攻击者在攻击中为获得资源的必要访问权限而付出的努力。

在实践中，可以根据资源属性估算破坏潜力与攻击成本。例如，根据函数的权限估算该函数的破坏潜力。攻击者通过在攻击中使用某一个函数，即可获得与该函数相同的权限，例如，攻击者利用 root 函数缓冲区溢出，获得 root 权限。然后，攻击者就可对系统进行破坏。攻击者利用系统通道与系统相连，并向系统发送数据，或接收系统发出的数据。通道协议对利用通道进行的数据交换进行了限制，例如，TCP socket（TCP 套接字）允许交换原始字节，而 RPC endpoint（RPC 端点）却不允许。因此，根据通道协议，能够估算通道破坏潜力。攻击者可利用持久数据项间接向系统发送数据，或间接从系统接收数据。持久数据的类型限制了数据的交换，例如，file 文件含可执行代码，而 registry entry（注册表项）却不含。攻击者可以利用 file（文件）发送可执行代码攻击系统，但不能利用 registry entry 发送可执行代码。因此，根据数据项的类型，可以估算数据项的破坏潜力。攻击者获取访问权限后，就可利用资源发起攻击，而攻击者如需获得这些权限，需要付出一定的努力。因此，对于函数、通道和数据这三种资源，根据资源的访问权限，估算攻击者利用资源实施攻击所需的成本。

假设函数 der 可映射资源的破坏潜力与攻击成本的比率。但在实践中，通常是给资源各属性赋值，然后计算破坏潜力与攻击成本的比率。例如，根据对函数权限和访问权限的赋值，可以计算出该函数的破坏潜力与攻击成本的比率。根据各属性的特征对所有属性进行排序，然后根据排序对各属性赋值。例如，相比具有 non - root（非 root）权限的函数，假设攻击者使用具有 root 权限的函数可以对系统造成更大的破坏。因此，root 权限函数的赋值要高于 non - root 权限函数。实际选择的数值具有一定的主观性，而且视具体系统及其环境而定。

可以根据函数、通道和数据三个维度来量化系统攻击面的度量指标，并分别估算函数、通道以及数据项对攻击面的总贡献。设系统 s 的攻击面为$\langle M, C, I\rangle$，则 s 的攻击面度量指标就是一个三元组 $\langle \sum_{m\in M} \mathrm{der}(m), \sum_{c\in C} \mathrm{der}(c), \sum_{d\in I} \mathrm{der}(d)\rangle$。

1.3 动态目标防御

本节将探讨攻击面度量在动态目标防御中的应用。动态目标防御是一种需要系统防御者不断转移系统攻击面的防护方法。直观地讲，防御者通过改变攻

击面的资源，或改变各种资源的作用，从而实现攻击面的转移。不过，不是所有的改变都能转移攻击面。防御者可以通过至少减少攻击面中的一个资源，或至少降低一个资源的破坏潜力与攻击成本的比率，达到转移攻击面的目的。在其他条件等同的情况下，若原攻击所利用的资源已经消失（改变），则该攻击将不再有效。但是，转移之后攻击面上可能会出现新的资源，从而可能使系统遭受新的攻击。这样一来，攻击者就需要更大的攻击成本维持原有的攻击，或者寻找新的攻击。

1.3.1 攻击面转移

本节将阐释攻击面转移的概念，同时引入一种量化这种转移的方法。前期研究中，已建立一种关于系统及其环境的I/O自动机模型，下面将运用该模型定义并量化攻击面的转移。

设有一个系统集S、一个攻击者U以及一个数据库D。对于系统$s \in S$，定义s的环境$E_s = \langle U, D, T \rangle$为一个三元组，其中，$T = S \setminus \{s\}$为不含$s$的系统集。$U$表示攻击系统集$S$的对手。数据库$D$允许数据可以在系统$S$和$U$之中共享。

我们按照I/O自动机为系统及其环境中存在的实体建模[4]。I/O自动机$A = \langle \text{sig}(A), \text{states}(A), \text{start}(A), \text{steps}(A) \rangle$为一个四元组，它分为四元：第一元是动作特征sig(A)，它将动作集acts(A)分为in(A)、out(A)和int(A)三个不相交的集，分别表示输入动作、输出动作和内部动作；第二元是状态集states(A)；第三元是非空的开始状态集$\text{start}(A) \subseteq \text{states}(A)$，以及转换关系$\text{steps}(A) \subseteq \text{states}(A) \times \text{acts}(A) \times \text{states}(A)$。$A$的执行结果是以开始状态为起点的动作和状态的交替序列，执行的序列是仅含有执行过程中出现动作的执行子序列。

设有一个系统s及其环境E，s的攻击面为一个三元组$\langle M, C, I \rangle$，其中：M是入口点和出口点之集；C是通道集；I是s的不可信数据项集。将属于s的攻击面资源集表示为$R_s = M \cup C \cup I$。此外，设s的两个资源r_1和r_2，用$r_1 > r_2$来表示r_1对攻击面的作用大于r_2。若改变s的攻击面R_o，得到一个新的攻击面R_n，则可将某个资源r对R_o的作用表示为r_o，同时将其对R_n的作用表示为r_n。将攻击面定性定义如下：

定义1.1 设有一个系统s及其环境E，s的原攻击面为R_o，新攻击面为R_n，如至少存在一个资源r，使得$r \in (R_o \setminus R_n)$或$(r \in R_o \cap R_n) \wedge (r_o > r_n)$，则$s$的攻击面就发生了转移。

s的攻击面转移之后，对s原攻击面的有效攻击对s新攻击面可能不再有效。在I/O自动机模型中，我们又对s与其环境之间的相互作用建模，得到并行组合$s \parallel E$。由于攻击者一般是通过向系统发送数据或从系统获取数据来实现对系统的攻击，因此，凡是对含有s的输入动作或输出动作的组合$s \parallel E$进行任何调度，都有可能构成对s的攻击。把对s的各种可能攻击表示为集合

attacks(s, R)，其中，R 为 s 的攻击面。在 I/O 自动机模型中，若 s 的攻击面由 R_o 转移至 R_n，则在攻击者和环境相同的情况下，R_o 上某些可能发生的攻击在 R_n 上将可能停止。直观地讲，若在转移攻击面过程中移除攻击面中的资源 r，或降低了 r 在攻击面中的作用，则不会在新的攻击面上执行包含 r 的 s。因此，基于这些执行结果的调度也就不再可能会在新的攻击面上攻击 s（图 1.2）。

定理 1.1 设有一个系统 s 及其环境 E，若将 s 的攻击面 R_o 转移至一个新的攻击面 R_n，则有 attacks(s,R_o)\attacks(s,R_n)$\neq\varnothing$。

证明：若将 s 的攻击面 R_o 转移至一个新的攻击面 R_n，则根据定义 1.1，至少存在一个资源 r，使得 $r\in(R_o\backslash R_n)$ 或 $(r\in R_o\cap R_n)\wedge(r_o\succ r_n)$ 成立。

若 $r\in(R_o\backslash R_n)$，可设 $R_o=R_n\cup\{r\}$，且能保证其普遍性。因为 $r\in R_o\wedge r\notin R_n$，后续证明与参考文献[5]中定理 1 的证明相似，存在一个函数 m，使得 $m\in R_o\wedge m\notin R_n$。因此，存在一个含 m 的组合 $s_{R_o}\parallel E$ 的调度 β，但 β 不是组合 $s_{R_n}\parallel E$ 的调度。证得 $\beta\in$ attacks(s,R_o) $\wedge\beta\notin$ attacks(s,R_n)，且 attacks(s,R_o)\attacks(s,R_n)$\neq\varnothing$。

同理可证，若 $(r\in R_o\cap R_n)\wedge(r_o\succ r_n)$，则 r 对 R_o 的作用大于其对 R_n 的作用。后续证明与参考文献[5]中定理 3 的证明相似，存在一个函数 $m\in R_o\cap R_n$，使得 m 在 R_o 中的前置条件大于其在 R_n 中的前置条件，或在 R_o 中的后置条件小于其在 R_n 中的后置条件。因此，存在一个含 m 的组合 $s_{R_o}\parallel E$ 的调度 β，但 β 不是组合 $s_{R_n}\parallel E$ 的调度。证得，$\beta\in$ attacks(s,R_o) $\wedge\beta\notin$ attacks(s,R_n)，且 attacks(s,R_o)\attacks(s,R_n)$\neq\varnothing$。

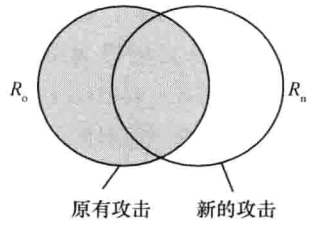

图 1.2 若将 s 的攻击面 R_o 转移至 R_n，则至少有一个在 R_o 上起作用的攻击将不会在 R_n 上起作用

在前面章节介绍了攻击面转移的定性概念，下面将量化攻击面转移。

定义 1.2 设有一个系统 s 及其环境 E，s 的原攻击面为 R_o，新攻击面为 R_n，则 s 的攻击面转移量 ΔAS 为

$$|R_o\backslash R_n|+|\{r:(r\in R_o\cap R_n)\wedge(r_o\succ r_n)\}|$$

在定义 1.2 中，$|R_o\backslash R_n|$ 项表示属于 s 原攻击面但已从 s 新攻击面上移除的资源数量。同理，$|\{r:(r\in R_o\cap R_n)\wedge(r_o\succ r_n)\}|$ 项表示对 s 新攻击面的贡献大于原攻击面的资源数量。若 ΔAS >0，则 s 的攻击面由 R_o 转移到了 R_n。

上述定义中，假设所有资源对攻击面转移的作用都相等。还可以考虑资源的属性，例如，资源的破坏潜力与攻击成本的比率，从而更好地量化转移过程。下面将进一步研究该量化方法。