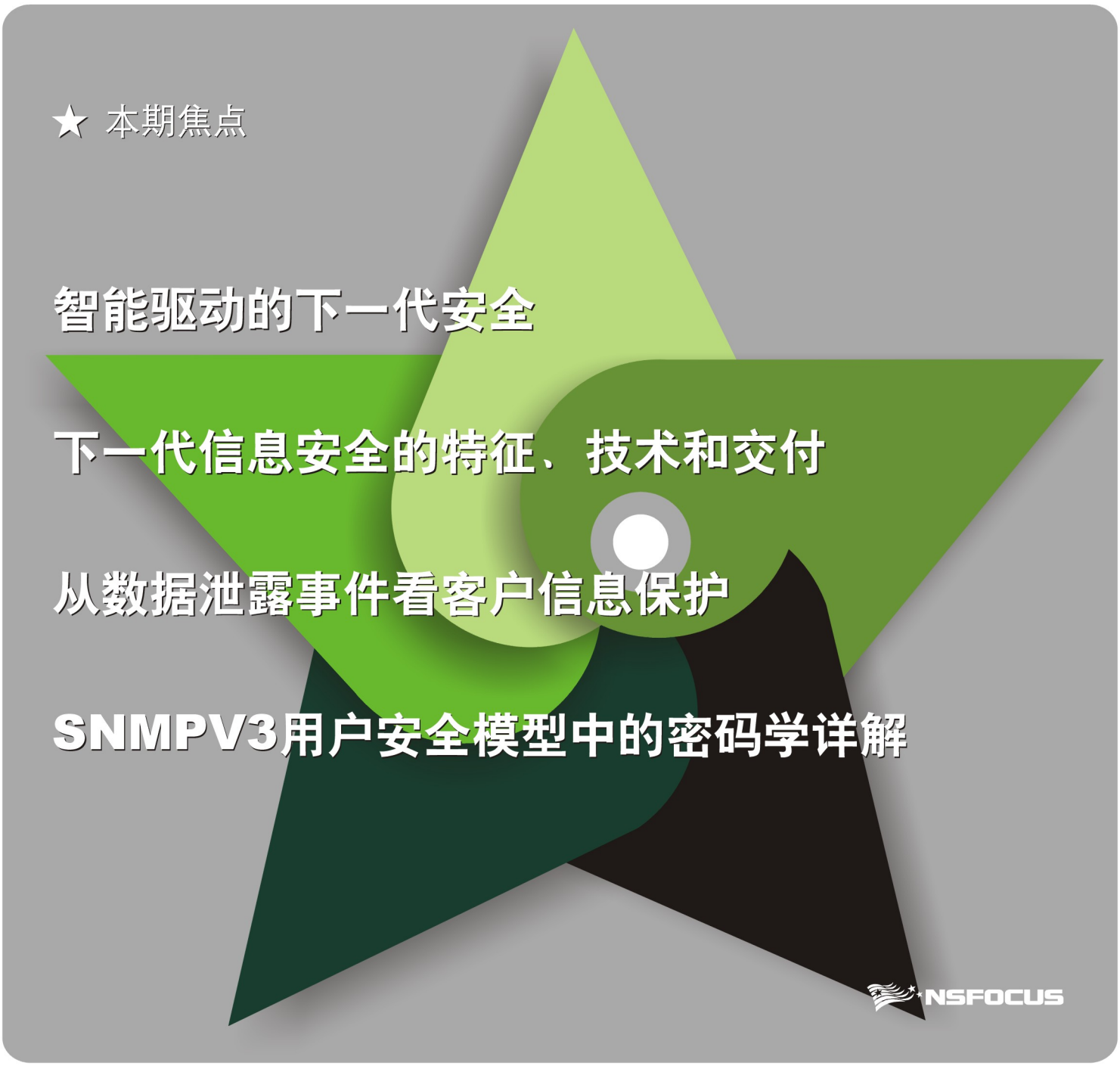
安全+
SECURITY
2012/07 总第017
技术版 ▶▶ 与安全人士分享技术心得 Share technique experience with security professionals



★ 本期焦点

智能驱动的下一代安全

下一代信息安全的特征、技术和交付

从数据泄露事件看客户信息保护

SNMPV3用户安全模型中的密码学详解

NSFOCUS

本期看点 HEADLINES

主办：绿盟科技
策划：绿盟内刊编委会
地址：北京市海淀区北洼路4号益泰大厦三层
邮编：100089
电话：(010)6843 8880-8667
传真：(010)6872 8708
网址：www.nsfocus.com

Nsmagazine@nsfocus.com

2012/07 总第 017

安全+ SECURITY+

目录 CONTENTS

智能驱动的下一代安全

安全研究院　赵粮

摘要：本文写在 2012 年的 RSA 大会之后，从安全事件和 APT 攻击、智能驱动的下一代安全、大数据等三个视角回顾了会议传递的信息，对会议中的热点话题和相关技术进行了讨论，并列举了若干支持性的数据和资料。

关键词：RSA 会议　APT 攻击　安全智能　大数据　BDA

前言

2012 年 2 月 25 日，一年一度的 RSA 大会在旧金山 Moscone 会议中心如期举行。从影响力和规模等意义上说，为期一周的 RSA 大会都是安全业界首屈一指的活动。大会一般由三部分组成：其一是产品和技术展览会，每年都有数百家安全业界的公司、社区组织、教育和政府机构等参加，在经济复苏的大背景下，今年的会议参展单位又有大量增加；其二是各种各样的主题演讲和技术报告会，从周二持续到周五，通常报告会分为多个主题（Track），例如今年的大会主题包括云安全、加密、数据安全、治理风险合规性、黑客和威胁等等；第三个部分是周一下午的创新沙盒活动，这个活动虽然历史较新，但非常引人注目，微型小型的创新公司带着自己的创意和新作通过会议前的海选进入大会，在这个舞台上有很大的机会得到投资人和用户的青睐，从而一跃龙门。

今年的 RSA 大会之所以特别引人瞩目，其中很重要的原因是过去的一年中发生的一系列的重大安全事件，包括 RSA 自己也没能幸免，被愈演愈烈的安全攻击攻陷。安全产业何去何从？很多人都将目光聚焦在作为业界重要风向标的 RSA 2012。 本文选择了安全事件和 APT 攻击、智能驱动的下一代安全、大数据等三个视角来回顾 RSA 2012 大会传递的信息。值得注意的是，本文没有展开讨论移动互联网安全和自带设备（BYOD）、云计算和虚拟化的安全等，它们同样是 RSA 2012 的热点话题，笔者期待业界同仁的更多分析回顾和分享。

1、安全事件和 APT 攻击

在某种程度上可以说，2012 年的 RSA 大会发生在一个“动荡”的 2011 年之后。这个“动荡”是指网络安全事件频发，很多事件对业界产生了深远影响，参见下图 [1]。

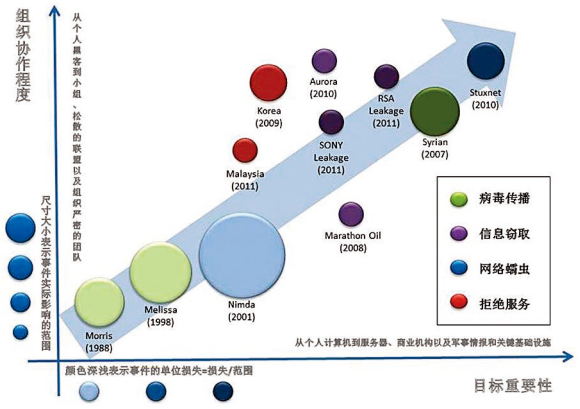


图 1：2011 安全事件图解

注：图中横轴表示攻击目标的影响力大小；纵轴表示攻击者组织协作的程度。安全事件用圆形区域表示，其中大小表示事件实际影响的范围，颜色的种类表示事件的类型，深浅表示事件的单位损失（损失和范围的比值）。

从 Aurora 和 Stuxnet 开始，APT 成了业界炙手可热的一个焦点，也谋杀了这次 RSA 大会的很多笔墨和目光。

按照公开的报道，RSA 公司在 2011 年遭到了 APT 攻击，攻击者综合利用了钓鱼邮件、远程控制工具、内网渗透和提权、数据外泄等多种手段。虽然从范围上看，RSA 安全事件没有大型互联网拒绝服务攻击那么大，但是影响却很大，很深，由于此事件的证书泄露而导致的 RSA 用户的被渗透攻击可能永远也无法得到一个公开的评估。

这个安全事件给会议的主办方 RSA 公司带来了很大的压力，其 CEO 在会议开幕的第一个主题演讲中使用 “If you are going through the hell, keep going”来形容 RSA 自身和业界的处境 – 痛苦但需要快速前行。

通常认为，APT 应当具备以下特点 [2]:

- 高级：威胁背后的操纵者有能力进行全方位的情报收集工作。不仅包括通过计算机入侵获取信息，而且还可以扩展到传统的情报搜集，如电话拦截技术和卫星成像技术。虽然攻击的个别手段可能无法被归类为特别 “先进”，但操纵者通常可以根据需要开发更为先进的工具。他们经常结合多种方法、工具和技术，以保持接触与尝试并最终攻陷目标。
- 持久：操纵者会执着地进行特定任务，而不是随机地搜索目标。这种区别意味着，操纵者也受外部实体的指挥。通过持续监测和接触，以实现针对目标的任务。如果操纵者暂时无法取得进展，他们通常会不断地重新尝试，并最终取得成功。操纵者的目标之一是保持长期访问目标，而不只是取得一次性的攻击机会。
- 威胁：APT 是一种威胁，因为它同时具备了能力和意图。APT 攻击的关键在于协调人的行动，而不是盲目的执行自动化攻击。操纵者有具体目标和动机，具备足够的技能、组织力和资金。

仁者见仁，智者见智。针对 APT 的 A（高级）、P（持续）、A+P（高级持续）、NA+P（不高级，但持续）等，出现了很多有启发性的检测和防护手段，图 2 给出了一个很好的汇总① [3]。对 APT 和恶意软件检测相关讨论感兴趣的读者，笔者推荐继续参考阅读 RSA2012 大会文件 Tech-107[4]。

图 2：针对 APT 攻击的检测防护手段

2、智能驱动的下一代安全

在和以 APT 为代表的下一代安全威胁对抗过程中，传统的安全产品和技术正在经受严峻的挑战，包括威胁响应的时效性、IT 安全运维团队和安全服务团队的专业化和规模化、安全技术的持续提升能力、安全产品和被保护的业务系统之间的匹配等方面。

RSA2012 传递出的一个重要的声音就是业界对安全智能

（Security Intelligence）或者智能驱动的安全（Intelligence-Driven Security）②的热烈拥抱。在战略意义上，智能驱动通过实现下面两大目标来应对上述挑战 [5]:

- 针对威胁和相应的态势开发实时的知识，用以阻止、检测、预测可能的攻击;
- 进行基于风险的决策，优化防御策略，并付诸行动。

其中，态势、知识、实时、决策、行动等是几个意义重大的关键词，构建了智能驱动的安全的轮廓。在上述战略目标下，围绕以下关键技术的研究和实践将决定其成败 [1]:

1 基于深度分析的态势感知

通过对代码、数据、事件等全局范围的信息进行深度分析，可以形成客观准确的安全态势判断，对安全事件作出预警。

2 基于安全信誉系统的异常检测

传统的异常检测在理念上基于攻击特征或模式匹配的，采用单纯黑名单（例如传统的入侵防御系统）或单纯白名单（例如防火墙）规则的判断，这种判断的简单性与现实世界的复杂性是不相符的。

智能化检测的指导思想是采用基于实体信誉的黑白互补的灰度判断，即对实体行为采用黑白名单相互补充的检测，并根据实体行为计算其安全信誉值的高低，进而以更细的粒度标识“异常度”或赋予不同的业务权限。

3 基于自动部署的攻击防护

在智能化的安全防护体系中，防护策略不再依赖手工操作，而是根据攻击类型动态的自动加载。检测算法的优化以及引擎与策略的松耦合，将为防护策略的自动部署提供技术上的保障。

4 基于流水作业的安全信息处理

安全信息的采集、加工、分析、发布等工序需要达到类似传统工艺的流水线作业的水平，很多人工操作需要转换成工具等自动化操作，以提高工作效率和时效性。

智能驱动的安全是安全业界在下一代威胁紧逼下作出的一个回答，虽然只是处于早期的探索阶段，其价值已经在下一代防火墙（NGFW）、下一代 IPS（NG-IPS）等 Gartner 定义的下一代安全产品中得到了集中体现。

3、关于大数据

正如大会前有些分析师预测的那样，大数据（BIG DATA ANALYTICS）是本届会议的最热焦点之一。按照会议主要技术资料和讨论所指出，大数据，或简称 BDA，不仅仅是处理海量数据，还包含快速、甚至实时的搜索功能、实时分析告警功能、数据展现技术等内容。

最直接的，BDA 是在安全信息事件管理 SIEM 产品技术基础上

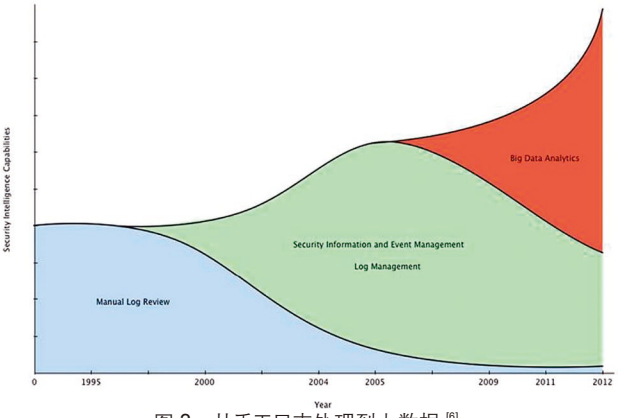


图 3：从手工日志处理到大数据 [6]

的一种升级，如图 3 所示，从手工日志处理，到现代化的 SIEM 管理，BDA 借鉴相对成熟的商业智能（BI）和数据仓库技术来处理每天可能高达数个T字节的安全信息，为安全管理层提供"事件响应"之外的、更加丰富的决策支持。

笔者认为，BDA 是业界一个重要的新思路，带来很多新的思路和实践，为原来战略层面上的安全智能指出了一个落地实现；可是，对 BDA 的热捧也从另外一个方面反映了业界的无奈，因为到目前为止，BDA 还没有给出任何实质性的、令人信服的证据，来说明 BDA 的确可以满足人们的期望。

4、结束语

以 APT 为代表的下一代安全威胁带给业界前所未有的挑战，在做好基础安全防护工作的基础上，笔者认为业界还需要从以下三个方面进行合作性的努力，才能在未来网络空间的安全对抗中占据有利地位:

• 合力建立更加健康的安全"生态"环境，形成分工合作的整体力量—专业的系统功能开发、专业的安全设计和开发、专业的漏洞挖掘和分析、专业的恶意软件 / 利用（exploit）监视响应等。有组织有计划地、客观地对关键信息基础设施的 ICT 系统进行安全测试和响应。特意设置一些温室，使某些关键信息系统不经受这样的安全考验，或者对这样的安全考验视而不见，不"安全"的后果一定在某个关键时刻惩罚这样做的人。

• 加强关键信息基础设施的 ICT 供应链安全管理，提高软件安全性。根据 NVD 的统计，近六年来公开漏洞的数量逐步减少；但是与之相对的，是漏洞地下市场的繁荣。现代 ICT 系统及其供应链系统都越来越复杂，全球化趋势使得外包、特别是离岸外包，非常普遍。这些因素对网络信息安全带来了挑战。简单的地域性、行业性的行政准入管理不足以应对这些挑战。

• 立即动手开始加强安全"数据"积累，开展对"深入分析"技术的探索。APT 攻击和"高级"、"持续性"的威胁并不意味着没有痕迹，只是隐蔽性较强而难以发现。通过大力发展多维度的海量数据挖掘和关联技术，实现跨时域、跨设备和跨区域的踪迹分析，可以大幅增加发现攻击行为的概率。相关数据和技术的积累可能需要数年的努力，并且需要安全用户和提供商、行业主管机构等的通力合作。

注：①引用并不代表笔者认可或者推荐图中的技术和产品。

②虽然安全智能和智能驱动的安全在含义上有不同，作者在本文中并没有对两者做特意的区分。

参考文献

[1] 鲍旭华 王卫东 李鸿培 赵粮，2011 年安全回顾与展望，绿盟技术内刊，总第 16 期，2012 年 3 月

[2] https://en.wikipedia.org/wiki/Advanced_Persistent_Threat

[3] http://blog.bit9.com/bid/44365/RSA-and-the-APT-Attack-Part-2-from-Bit9

[4]Tech-107:Stop the Maelstrom: Using Endpoint Sensor Data in a SIEM to Isolate Threats，RSA2012 会议资料，2012 年 2 月

[5] http://www.rsa.com/innovation/docs/CISO-RPT-0112.pdf

[6]Tech-303:Security Data Deluge- ZionsBank' s Hadoop Based Security Data Warehouse, RSA2012 会议资料，2012 年 2 月

下一代信息安全的特征、技术和交付

安全研究院　王卫东

摘要：本文从下一代信息安全的驱动因素出发，分析了其基本特征以及所用到的核心技术。并对下一代信息安全的交付形式和内容做了较详细的分析。

摘要：下一代　信息安全　服务化　智能化

1、引言

近两年来，“下一代信息安全（以下简称 NGIS，Next Generation Information Security）”已经成为信息安全领域的一个热门话题，多数厂商从技术和产品角度对 NGIS 的概念进行了探讨。然而仅仅从这两个的角度展开讨论，既不全面也没有触及问题的本质。NGIS 不仅仅是产品和技术上的概念，而是信息安全相关领域的全局性概念，是一个总体目标和需求。通常只有事物发生本质变化的时点才能作为其发展进程中的断代界标。具体在 NGIS 的问题上，这种变化应该表现在指导方针、交付模式、研究方法、供求关系、标准规范、产品技术等各个方面。因此，对 NGIS 的研究应该从其驱动原因、区别于以往的主要特性，以及 NGIS 的实现方式等方面进行全面深入分析。分析清楚了促成原因，也就全面理解了对 NGIS 的需求。分析清楚了主要特性，也就明确定义了什么是 NGIS。对实现途径的研究，就是从交付模式、供求关系、产品技术等方面探讨达成 NGIS 的具体方式。

本文试图在分析 NGIS 产生根源的基础上，给出其主要特征并逐一分析。进而从交付模式、供求关系、产品技术等方面阐述 NGIS 的实现途径。

2、向 NGIS 演进的驱动

2.1 外部驱动因素

人类社会进入新的世纪，商业模式也

发生了巨大的变化—客户价值从依附于实物产品向依附于满足用户需求的服务转变。这种转变不仅发生在 IT 行业，即使是典型的传统行业如小家电行业，也出现了这样的趋势。云计算和虚拟化、物联网等技术的普及，标志着交付模式明显呈现服务化的趋势。这种服务化的趋势带来的另一个变化就是信息技术服务明显向集约化方向发展。由于信息安全具有天然的依附性（或衍生性），信息安全将伴随着信息技术的变化而出现相似的变化。即信息安全也将具有集约化和服务化的特性。

信息技术广泛而深入的普及，信息技术从网络传输向虚拟现实转变。虚拟的网络空间与现实的物理社会重合度越来越高。信息相关的业务越来越成为一种普遍的个体化服务，信息安全也必然要满足这种个性化的需求。

2.2 内部驱动因素

信息安全的问题在很大程度上表现为攻防双方的较量。伴随外部环境的变化，信息安全威胁也发生了明显的变化。这些变化主要表现在以下几个方面:

1）攻击动机从炫耀好奇向利益驱动和意识形态转变

信息技术的普遍应用使得对信息系统的攻击成为一种谋取暴利的行径。同时信息系统已经上升为支撑社会平稳运行的重要基础设施和公共服务，因此也必然成为政治组织攻击的目标。尤其是攻击的代价远远低于传统暴力攻击的情况下，攻击者将更倾向采用信息安全攻击的手段。

2）攻击手法从单纯漏洞利用向结合社会工程转变

随着信息安全防护技术的不断成熟，依靠纯粹的技术手段已经很难奏效的情况下，攻击者将更多结合社会工程的手法，从人（信息系统的使用者），而不是机器系统寻找突破口。

3）攻击的技术手段从网络层向应用层转变

现实中的应用场景是多种多样的，因此信息系统的应用层也是复杂多变的。应用层的复杂性自然存在漏洞的几率也就更大。某些应用层的脆弱性是与生俱来的，应用实现的本身就存在被攻击的可能。

4）攻击目标从无特定性向有明显针对性转变

在利益的驱动下，为了使攻击收益最大化，需要尽可能提高攻击的效率。锁定有价值的攻击目标再发起攻击。[DYJ]

5）攻防双方的整体态势对比从不对称向极其悬殊转变

攻防双方在信息和资源、成本等方面存在天然的不对称性。对于一个有多个漏洞的系统，攻击只要利用其中一个就可以达到目的，而防护者需要将所有的漏洞都修复。信息技术的高度发达更加剧了这种不对称性。防护方在攻击信息获取以及人力资源投入都远远跟不上技术发展的需要。

6）攻击一方呈现产业化和组织化倾向

攻击方已经细分出明确的产业分工并形成从研究到开发再到销售的产业链。近两年一些黑客不再以单打独斗方式活动，而是组成松散的联盟，并以公开的形式频繁发动攻击。

3、NGIS 的主要特征

通过前面的讨论，已经提示了一些 NGIS 的特征。如集约化、服务化。这些特征既是信息安全形势对 NGIS 提出的要求，也是 NGIS 应该达成的目标和努力方向。同时也是 NGIS 区别传统信息安全的界定标准。这些特征还是下一代安全产品开发、服务运营的指导性原则。

3.1 集约化

NGIS 的集约化特征可以体现在产品技术、人力资源、运营管理等很多方面。从产品角度来看，集约化要求下一代安全产品应该便于统一集中管理、数据互操作和协同联动，从而整合成一个防护的整体，提高防护的有效性。例如反垃圾邮件系统发现了带有附件疑似垃圾邮件，就应触发恶意软件检测系统，进一步确认附件是否为恶意文件。对于确认的垃圾邮件应及时删除，以防止收件人因人为失误而打开垃圾邮件。在集约化的要求下，需要建立一个集监控维护和服务运营等功能于一身的统一管理平台，将多种安全产品整合在一起协同工作。尽管类似的产品（如 SIEM 类的产品）在很多年前就已出现，但功能还仅局限于日志信息的分析和呈现上，离 NGIS 要求的统一管理平台的差距还很大，如缺少对产品的管理功能、缺少各单个产品在不同攻击场景下协调联动的功能等。

在信息安全管理中采用云计算和虚拟化技术，也有助于提高工作的集约程度。有些硬件形态的产品需要转化成虚拟化镜像才能部署在虚拟化环境中。桌面虚拟化技术的采用也将大大提高终端安全管理工作的集约度。

对于人力资源的调配，也应该考虑集约化的原则。在组织内部成立专门信息安全管理的部门（通常称为 SOC），是人力资源集约化的一种途径。将一些安全管理工作外包给专业的安全公司，也可以达到人力资源集约化的目的。

3.2 服务化

服务化可以从两个方面来理解。一是信息安全产品从形态上将从软硬件向服务运营过渡。这里说的“服务”不是指传统的安全咨询服务 [Zhaol]，而是指建立在大规模“安全智能”系统基础上的一种全新的安全交付方式，包括“态势感知”、“策略升级”、“规则更新”、“事件响应”等核心能力。二是企业组织将信息安全管理工作外包给专业的安全服务运营公司（MSSP, managed security service provider），由专业人员以更专业且更高效的方式提供信息安全保障。

3.3 智能化

为有效应对安全威胁不断提升的局面，防护方必须大幅提升现有技术措施的工作效率，而提升效率的唯一办法就是提高技术措施的智能水平。安全设备根据攻击的具体情况自动选择并加载相应的防护策略，安全设备之间顺畅的分享信息和数据进而实现协同联动，把专家的分析经验固化为智能分析算法，用自动化的漏洞挖掘工具替代人工的分析，以及对漏洞和威胁信息自动分析整理等都是智能化努力的方向。总之，智能化的目标就是通过自动化的分析处理，释放对人工处理操作的依赖，从而使信息安全工作的效率有全面大幅度的提升。

3.4 主动性

传统的信息安全总体上处于被动防守的态势，总是在事件发生之后进行补救。主动防御的概念虽然也早就被提出，但仍然没有脱离被动防御的窠臼。NGIS 的主动性表现在采取“防患于未然”以及“以其人之道还治其人之身”的战略。

信息系统设计和开发阶段，就充分考虑信息安全的要求，对系统研发严格按照研发安全生命周期（DSL）规范进行，可以避免很多漏洞的产生。

信息安全的攻防和军事对抗有很多类似之处，兵不厌诈的原则在信息对抗中也一样适用。即在战术上对攻击方使用诱骗欺诈甚至是逆向攻击的手法。例如在网络上部署蜜罐或蜜饵（honey token）诱骗攻击者，对攻击源的反向追踪及僵尸网络定位，甚至将计就计的发动反向攻击都是对攻击者采取主动策略的方法。

3.5 生态化

如前所述，攻击方已经形成了分工明确、利益维系的社群，甚至形成了具备相当大规模的产业链条。产值而在防守方，安全提供商与用户（信息系统最终用户和信息系统厂商）之间还没有达成应有的默契与信任。信息系统厂商对安全专业人员的研究成果也没有给予足够的认可和鼓励。不过一些大型信息系统开发商已经开始通过高额悬赏的方式鼓励专业人员协助他们发现产品的漏洞。这样做的意义不仅在于提高信息产品的安全性，还可以压缩地下产业链的交易空间。显然，NGIS 时代防守阵线内部必须形成一种充分信任、互惠互利的生态系统，使得来自不同厂商的产品和服务可以一起解决同样的安全问题[RSA]，以信息安全厂商为纽带，跨越用户间和行业间的壁垒，使信息得到充分的共享。

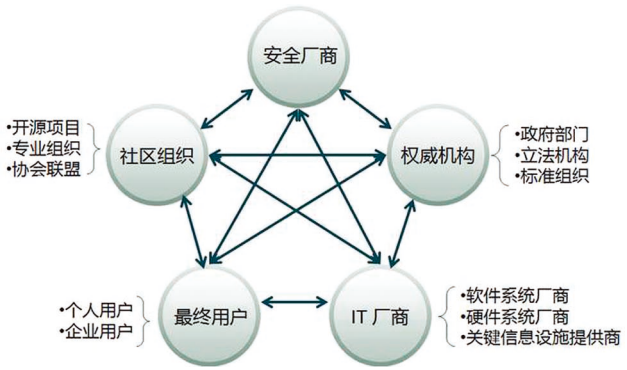


图 3-1：信息安全防护方生态关系图

如图 3-1 所示，在这样一个生态系统中，大致包括 5 种角色。各角色之间的双向箭头表示他们是互惠互利的合作共存关系。例如安全提供商和 IT 厂商可以从社区组织获得第一手的研究成果，及时完善和充实产品和服务。社区组织在厂商的资助下，更好的开展研究和创新活动。厂商与最终用户之间在彼此充分信任的基础上，建立紧密的合作关系，如建立联合实验室，推进个性化的研究项目，以更好的满足用户的需求。用户还可以将安全相关的运维工作托管给厂商。

4、NGIS 的核心技术

为了适应 NGIS 的原则要求，相关的核心技术也需要有相应的变革和突破。例如为了适应服务化的趋势，就要通过源源不断的更新服务内容来持续满足用户的需求。不断更新的服务内容就来自于对各种数据、样本和情报等信息的快速分析。再例如，信誉评价技术将为异常判断提供更灵活和动态的标准。

4.1 灰度判断与动态信誉

传统的信息安全对异常的判断是一种黑白判断。防火墙对正常的访问（白行为）予以放行，其它全部阻断。入侵防护系统则把异常行为（黑行为）全部阻断，其它全部放行。这种简单判断无法满足现实世界复杂性的需要，容易造成较多的误判。引入信誉值的概念，在某种程度上解决了这个问题。信誉值是综合多种因素计算而来，是一个随时间动态变化的连续值，基于信誉的判断是一种灰度判断。可以根据安全威胁的态势，动态选择判断阈值的宽严程度。

4.2 主机行为监控技术

传统的异常检测理念是基于特征（signature）的规则判断。特征是不含有主体和客体信息。近年来的实践表明，基于行为（behavior）模式的检测将更具优势。IT 系统行为大致可以分为主机行为和网络行为。网络行为可以通过镜像流量全面监控，而主机行为的监控则涉及更复杂的技术。这种复杂性体现在环境复杂、行为多样、自我隐蔽等方面。主机监控要适应不同的操作系统和硬件环境，监控多种多样的行为，如进程相关的行为、文件和配置相关的行为等，还要做好自我隐蔽，防止被监控对象探测到。主机行为监控技术对恶意程序样本捕获与分析、IT 系统异常行为检测、漏洞挖掘与分析、检测和防护规则的开发等一系列攻防相关的应用场景具有重要意义。

4.3 关联分析及大数据（Big Data）分析

在信息安全管理过程中，经常会遇到对大量日志数据进行关联分析的场景。关联分析技术可以将人力从繁琐的分析工作中释放出来，大幅度提高数据分析的效率和准确性。关联分析和大数据分析技术是一套方法论，可以应用于攻击溯源、异常行为检测、异常流量分析等很多场景。卓越的关联分析和大数据分析技术是智能化的基础，也是 NGIS 产品核心竞争力的体现。

4.4 信息安全情报的分析与利用

信息不对称是攻防双方力量悬殊的主要原因之一。通过自动化的手段广泛收集安全威胁情报 如来自CERT、安全服务厂商、防病毒厂商、政府机构和安全组织的安全预警通告、漏洞通告、威胁通告等）并进行自动化的梳理和分析，快速形成有参考价值的量化结果，能够及时获得脆弱性及外部威胁变化的情况，以便动态调整安全策略或及时采取应对措施。例如，通过持续搜集漏洞相关的信息（公开披露、确认收录、补丁发布、工具出现、事件发生等）可以计算出每个漏洞在其生命周期中不同阶段的威胁指数，参考这个数据，可以对漏洞采取忽略、利用虚拟补丁、安装补丁程序等处理措施。再例如，根据外部威胁手段流行的趋势，可以感知当前信息系统的风险态势。

4.5 安全分析工具

传统的漏洞发掘分析过程，需要大量的人工参与，效率低下。利用自动化工具进行漏洞发掘，大幅提高漏洞发掘分析的工作效率，是下一代安全的必然要求。漏洞发掘工作经历长期的经验积累，也具备了将零散的经验凝聚成自动化工具的条件。类似地，源代码安全审计也是一个耗费人力的工作，这个领域也同样需要自动化的工具来辅助工作。

工具的使用，可以降低对人员技术能力的要求，更多的使用需求，可以促生商业化工具产品的开发。

4.6 虚拟化

IT 系统采用虚拟化技术之后，安全防护产品也必然出现向虚拟化环境迁移的趋势。因此虚拟化技术必然成为 NGIS 的核心技术之一。一些硬件形态的安全产品将以镜像文件形式出现。沙箱和桌面虚拟化技术作为新的安全防护手段将会得到更广泛的使用。

5、NGIS 中的安全交付模式

安全提供商与最终用户的关系是 NGIS 各实体关系中最重要的。图 5-1 示意性呈现出安全提供商如何提供下一代的信息安全产品和

服务。虚线框内的部分表示部署在最终用户方的设备和管理平台。用户的设备和 IT 系统可能是传统的分立式硬件，也可能是自建的私有云。虚线框以外的部分是厂商方的系统。安全提供商可以通过这样一个部署模型对最终用户提供高度定制、灵活选择、长期持续的安全服务。

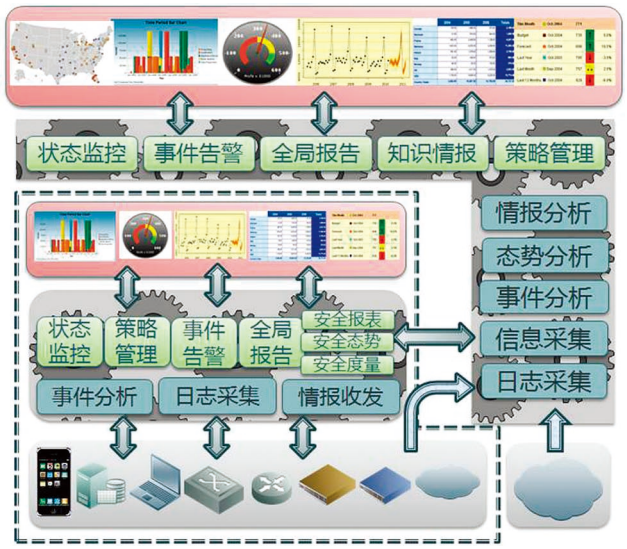


图 5-1：下一代安全服务的种类及交付模式

- 状态监控

最终用户可以利用部署在企业内部的安全监控平台，对 IT 系统进行安全维护。也可以通过网络将系统日志、设备状态等相关信息上传到安全提供商的监控中心，委托监管。这里的状态包括设备运行的状态和 IT 系统服务能力的状态。例如网络银行系统的软硬件运行都很正常，但是域名解析异常，整个网络银行系统依然无法为用户提供正常的服务。

- 知识情报

安全攻防是人与人的较量，安全设备只有不断予以更新相关的知识和情报，才能不断适应的攻击变化，很好的发挥防护作用。这些知识和情报就包括各种新的防护策略、各种新的威胁手法、新的事件分析方法、安全信息情报等等。例如安全提供商从互联网上搜集各种安全相关的原始信息（如漏洞披露、安全通告、威胁方法），经过工业化的情报分析过程，形成对用户有参考价值的信息（如漏洞威胁指数、恶意域名列表、IP 信誉信息、攻击事件等等）。以知识和情报为交付物的安全服务是下一代安全服务的主要形式。知识情报服务是其它安全服务的基础，不同安全服务分别依赖不同的安全情报知识（见图 5-2），虚线框中的内容是各种可能的知识情报。

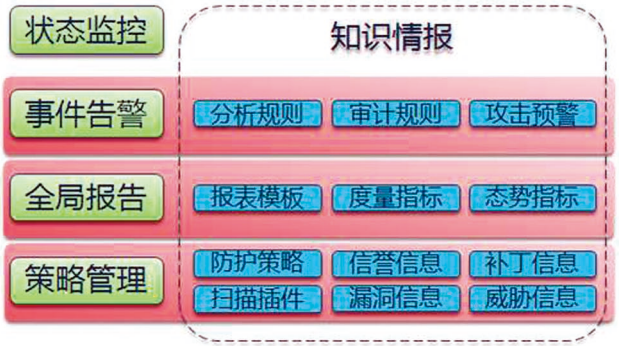


图 5-2：情报知识服务与其它服务的关系

- 事件告警

通过对海量日志的关联分析，可以发现其中所蕴含的安全事件，并生成告警。最终用户将日志发送到厂商的监控中心进行分析，也可以在用户本地的安全监控平台上进行分析。安全提供商不断提供新的分析方法。对日志进行有效分析，需要具备两个条件，一个是快速有效的分析框架以及结合攻防知识的分析方法。前者是工具，后者是算法。市场上已经具备了不少成熟的分析工具产品。而结合专业知识的分析算法是不断补充和完善的。因此事件分析服务呈现出两种类型：完全委托型和方法咨询型。

- 全局报告

全局报告的内容包括包安全报表、态势报告、度量报告等。根据合规性要求，企业需要定期生成相关报表以证实其在合规方面的满足程度。与事件分析服务类似，用户可以委托厂商的监控中心来生成报表，也可以利用安全提供商提供的报表模板在用户本地的安全监控平台上生成报表。厂商根据法规的变化更新报表模板。除此以外，厂商还会提供一些常规报表模板。

态势报告是综合外部威胁和内部脆弱性以及资产信息给出的 IT 系统整个风险状态和变化趋势的预测。[WWD1]

度量报告是通过对基本度量指标周期性测量和计算，得到的 IT 系统在安全保障方面的效率、效能等方面的绩效水平。[WWD2]

全局报告是一种高级服务，其中的态势报告依赖于情报分享服务。

- 策略管理

安全设备上的规则和策略需要随着外部威胁的变化以及用户 IT 系统的变更而动态调整。例如遭遇不同类型的 DDoS 攻击，应采用的防护策略也是不同的。Web 应用防护设备上的策略也是如此，需要根据 Web 应用的具体情况配置防护策略。最终用户可以将策略管理工作完全外包给安全提供商，也可以利用厂商发布的情报信息，自行调整策略。

6、结束语

目前，对 NGIS 的研究，大都停留在概念层面，还需要进一步探索将概念转化为具体的实现方式和途径。例如安全提供商应该通过怎样的方式交付服务，具体交付的内容是什么等。对 NGIS 概念的研究与推广，有利于在防守方内部的各类角色之间达成共识，建立互惠共存的生态关系。

参考文献

[DYJ] 杜跃进“新一代网络安全威胁及其影响”CNCC 深圳 2011.11.24

[Zhaol] 赵粮，“关于下一代安全的几点思考”，绿盟技术内刊，总第 14 期，2011

[RSA] “RSA Executives Offer Seven Guiding Principles To Maximize Megatrends Redefining the Information Security Industry” https://latinamerica.rsa.com/press_release.aspx?id=10482

[WWD1] 王卫东，“网络安全态势感知体系探讨”，绿盟科技技术内刊，总第 12 期，2011

[WWD2] 王卫东，“安全度量知多少”，绿盟科技技术内刊，总第 11 期，2010

攻与防的博弈

产品管理中心　蔡立军　唐伽佳

摘要：近些年，随着云计算、虚拟化、Web2.0、社交网络、IPv6 等新技术、新应用的发展与广泛使用，网络安全攻击的演进也趋向新型的高级持续性攻击。在攻击过程中越来越多的利用社会工程学、0Day、Botnet、恶意文件、AET 高级规避攻击等技术手段，攻击方式已从上一代单纯基于字符串的攻击演变为新一代针对应用的高级可持续性攻击。传统入侵检测与防护产品基于协议异常和系统漏洞分析发现入侵行为并进行阻断的工作方式显得捉襟见肘，重新思考安全，急需要我们提出新的行之有效的下一代网关型的入侵防护模型。

关键词：下一代　信誉技术　智能识别　环境感知　白环境　行为分析　协作

1、安全市场发生了什么样的变化

1.1 攻击方式的演进

当前的网络攻击方式趋向于新型的高级持续性攻击，在攻击过程中越来越多的利用社会工程学、0Day、Botnet、AET 高级规避攻击等多种技术手段，已从上一代单纯基于字符串的攻击演变为新一代面向应用和内容的深度、复杂、高级可持续性攻击。

我们通过震惊安全界的 “RSA SecurID 泄密事件” 做进一步分析:

1. 攻击者给企业的多名员工发送了一封带有 “2011 Recruitment plan.xls” 附件的恶意邮件。

2. 其中一位员工对此邮件感兴趣，并将其从垃圾邮件中取出来阅读。很不幸，此电子表格文件含有当时最新的 Adobe Flash0Day 漏洞（CVE-2011-0609）。该主机被植入臭名昭著的 Poison Ivy 远端控制工具，并开始从 Botnet 的 C&C 服务器下载控制指令。

3. 攻击者以上述机器为跳板，长期潜伏，相继攻陷业务主管和系统管理员等机器。

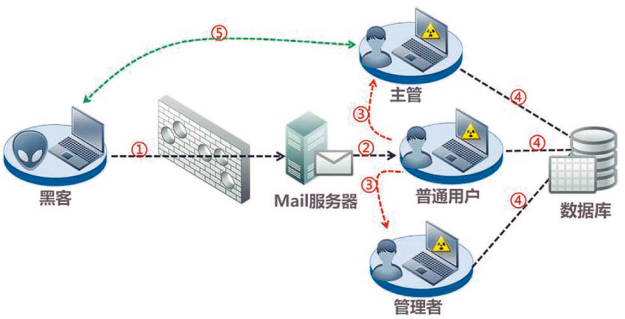


图 1：高级可持续性攻击示意图

4. 最终，开发服务器遭到入侵，被窃取的 SecurID 信息加密传送至远端主机。

1.2 带来的安全挑战

上述案例属于非常典型的新一代网络攻击方式，整个攻击过程可以分解为以下几个阶段:

图 2：高级可持续性攻击流程图

在整个攻击过程中，黑客采用了 0Day、Botnet、远程控制工具等复杂的攻击手段，具有以下特点:

- 恶意文件 / 恶意链接

与传统攻击直接向目标系统发送攻击代码不同，新型攻击将攻击代码隐藏在恶意文件中，并通过邮件、恶意链接等方式诱骗攻击目标打开文件。

- 针对客户端应用的 0Day 攻击

隐藏攻击代码的恶意文件被打开后，攻击代码便会利用 Office 软件、Adobe Reader、影音播放软件、下载软件等客户端应用中的漏洞安装木马程序。客户端应用种类多、变化快，存在漏洞多; 而且，恶意文件比恶意代码更容易快速产生新的变种，难以防范。

- 利用 Botnet 实现远程控制

木马被安装好后便会主动连接控制服务器，以获取控制命令。通常会采用加密通讯，以规避部署的安全检测与防护设备。

- 长期潜伏，逐步渗透

攻击者在客户端种植木马程序的目的是要获取有价值的信息，因此会对局域网逐步进行渗透，以找到能够帮其获取到信息的“有权限的人”。这个过程，可能会持续很长的时间。

- 合法身份做非法事

最终帮助攻击者“窃取”敏感信息的“有权限的人”，所做的每一个操作看起来可能都是“合法的”。

由此，我们不难发现，针对企业的攻击已经上升到了一个强目的性、高组合型的时代。通过下图中攻击技术手段的革新过程，我们也可以看到两个重要的变化:

- 攻击目标由“系统软件”转变为“客户端应用”;
- 攻击代码由“恶意字符串”转变为“恶意文件”。

传统攻击是“围绕系统”展开的，而新型攻击是“围绕人”展开的。

图 3：攻击技术发展趋势图

2、传统安全防护产品存在的问题

从攻防角度来看，遵循一条基本规律：首先，需要能识别“是什么”；然后，再对内容进行检测，搞清楚“要干什么”；最后，才能

采取有效的防护措施。面对新型攻击带来的挑战，传统攻击防护手段却显得捉襟见肘。我们来看一下传统防护手段面对新型攻击时会存在哪些问题。

2.1 识别问题

- 内容识别

现在的攻击通常会隐藏在貌似合法的应用（Application）和内容（Content）里面，例如上述案例中的恶意文件。而传统安全防护设备通常只是在网络数据包中搜索特定的恶意字符串，例如：通过“XXXXXXXXX.ida”字符串检测红色代码蠕虫攻击。所以，无法识别隐藏在恶意文件中的攻击。

- 应用识别

细粒度的应用识别是传统防护设备遇到的另一个重要难题。原来 80 端口就是 HTTP，而 HTTP 就代表着网站。但随着互联网的发展，协议复用使得 HTTP 可能是 WebMail、可能是 WebIM、可能是网友、也可能是一段视频，传统防护设备基于端口识别应用的方式显然无法准确识别，更谈不上检测了。

- 用户识别

新型攻击围绕“人”展开，用户的业务安全策略也都是以人为出发点的，而传统防护设备基于 IP 地址的策略配置方式，不能识别真实的用户身份信息，无法满足用户业务安全的需要。

2.2 检测问题

早期的攻击主要是针对操作系统和应用系统软件漏洞的基于字符串的攻击，因此，传统入侵检测技术通过分析漏洞，并建立 Signature 攻击特征库。当在网络数据包中搜索到 Signature 库里包含的攻击字符串时，就认为检测到了攻击。而随着攻击技术的演进，传统检测技术就显得力不从心了。

- 攻击呈现隐蔽化趋势

新一代的高级可持续性攻击更多的利用了 0Day、高级规避技术以及加密通道传输等技术手段，传统防护产品针对 0Day 等未知攻击无法事先制作 Signature，在遇到高级规避攻击或加密通道技术时也难以从数据包中找到“恶意字符串”，因此无法及时检测到攻击事件的发生。

攻击变种使得 signature 检测方式雪上加霜。Signature 库的升级需要经过开发、测试、分发、部署多个阶段，而攻击的存活时间平均 3 天，往往前一个攻击的检测规则还没开发完，就出来了好几个攻击变种。这种裂变效应，让安全厂商疲于追赶攻击变种更新的步伐。

- 面向客户端应用的文件型攻击

新攻击中有相当一部分是文件型攻击，利用客户端应用（如 IE、影音视频播放软件、Acrobat PDF）的漏洞进行攻击。针对这种文件型攻击，传统安全防护设备检测深度不够，无法识别文件内容，所以很难检测。

而客户端应用数量庞大、更新速度快，以及文件易于变种的天然优势，安全厂商难以靠无限增加Signature的数量来应对此类攻击。

- 合法身份做非法事

以合法身份做掩护，进行违反安全策略的事情，是新型攻击的另一个显著特征。例如，上述案例中被渗透的主管窃取机密信息并通过 FTP 泄密的行为，很明显是不符合业务安全策略的网络行为。但传统检测机制只关心数据特征，只能根据 IP 地址执行安全策略。无法识别用户身份、细粒度的应用，不能理解业务安全策略，自然无能为力。

- 孤立的“安全事件”与“安全产品”

传统攻击的过程比较简单，攻击路径也

比较直接。而新一代高级可持续性攻击采用的攻击方法复杂、多样，攻击路径曲折，持续时间长。上述案例中涉及的攻击技术起码包括了: 社会工程学、0Day 攻击、Botnet 等攻击路径涉及发送邮件者、邮件接收者、被渗透的主管、业务服务器、C&C 服务器等。传统安全防护产品各自为战的防护模式无法满足新的防护需求。

同时，传统入侵检测防护技术的误报问题和如何从海量告警日志快速找到对用户最有价值的信息，也一直困扰着用户。

3、如何解决上述问题

传统安全防护产品该如何应对新一代的安全威胁呢？我们再来回顾一下上述案例，整个攻击过程可以分为以下几个主要步骤:

第一步：利用 Adobe Flash 的 0Day 漏洞（CVE-2011-0609），在终端计算机上安装恶意软件;

第二步: 恶意软件连接控制端接收指令，并根据控制指令执行相应操作;

第三步: 恶意软件在局域网内逐步渗透，获得更高访问权限;

第四步：恶意软件收集信息，并向控制端回传信息。

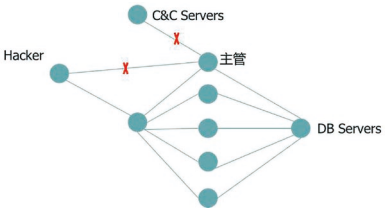


图 4：攻击过程分析图

由此可见，如何阻止恶意软件进入企业网络是最关键的一环。如果能够实时识别恶意软件并拦截无疑是最完美的，但在线方式重组数据包、识别应用、识别并检测文件，本身就极其消耗资源，更何况软件类型种类繁多且千变万化，所以，在线实时检测恶意文件内容是一个不可能完成的任务。信誉技术则是一条不错的“曲线救国”之路。

- 信誉技术

信息安全厂商建立安全云，并在云端检测、识别恶意文件、挂马网站和养马场 URL 地址、发送恶意邮件的 IP 地址等，并将每一个恶意文件、URL 和 IP 地址生成相应的 HASH 值同步到安全防护设备的文件信誉库、Web 信誉库和 IP 信誉库中。安全设备只需比对 HASH 值就可以了，大大减少了资源消耗。当然，如何保障覆盖的全面性和更新的及时性是信誉技术面临的挑战。

黑客进入企业内网之后的一系列持续性动作以及由此产生的异常行为通过传统基于 Signature 的检测方式难以奏效，而基于用户身份的行为分析（Behavior Analysis）技术则可以很好地应对这个挑战。

- 基于用户身份的行为分析技术

网络中的每个用户根据各自的工作职责和个人爱好都会形成各自的行为习惯，而这种行为习惯能够反映在日常的网络访问活动中。对这些网络访问活动进行分析并经过长时间地收敛，可以根据用户身份（Who）、地理位置 /IP 地址（Where）、业务系统 / 网络应用（Whom）、操作（What）、时间（When）、频次（How）等条件建立用户的正常网络访问模型。根据用户的业务安全需求，基于正常的网络访问模型建立企业网络“白环境”。当检测到网络中出现了违背白环境模型的异常行为时，则很有可能发生了攻击行为。

例如：上述案例中的业务主管平时只在工作时间访问业务服务器，而被攻击之后则突然在凌晨多次访问业务服务器就一定要引起注意了。

当然，建立企业网络白环境的前提是能够实现用户身份的识别、应用的识别，以及将用户身份、业务系统、地理位置、操作频次等多种与操作相关的网络环境信息进行关联分析，才能准确识别用户行为。因为只有将上述信息关联起来才能称之为行为。例如：开门是个动作，单纯看这个动作无法分辨好坏。但如果是“张三（Who）在半夜 2 点（When）开李四家的门（Whom、What），而且换了 5 把钥匙都没打开（How）”，则一定是异常行为。

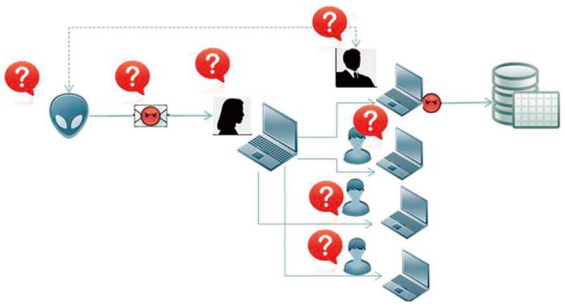


图 5：持续性攻击过程中需要识别的内容

前面也介绍过，新一代攻击的技术复杂性、攻击路径曲折性以及时间持续性均与传统攻击有很大不同，这就要求终端安全、病毒检测、邮件安全、入侵检测与防护等多种产品和技术协同作战，才能有效应对。而协作不只是安全产品之间的协作，也包括不同用户之间的协作，例如：在 A 用户处发现的恶意文件通过安全云迅速更新至 B 用户的信誉库中，则可以避免 B 用户被同一块石头绊倒。

4、总结

通过上述分析，我们认为下一代网关型的入侵防护模型至少应具备以下 6 个特点，才能在防护新一代威胁中发挥更积极的作用:

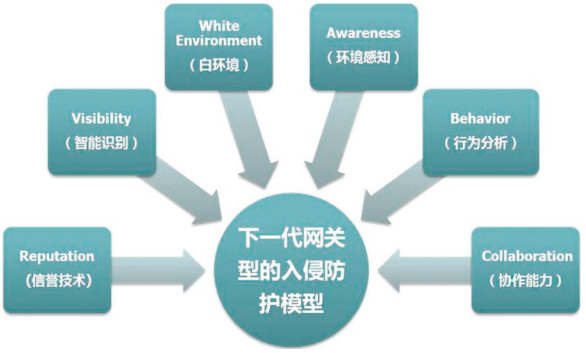


图 6：下一代网关型的入侵防护模型特点

- 信誉技术（Reputation）：文件信誉、Web 信誉、IP 信誉等。
- 智能识别（Visibility）: 应用（Application）识别、用户身份识别，文件内容识别。
- 环境感知（Context-Awareness）：结合用户身份、地理位置、Web 信誉、用户资产等网络环境上下文相关信息，能够显著减少虚假告警事件的产生。例如：目标系统运行的是 Apache 软件，就不会再产生大量针对 IIS 的虚假告警事件。
- 白环境管理（White Environment）：白环境描述的是一类自然语言描述的业务策略的集合，等同于由 Business policy(业务策略) 转换成的 Comprehensive Rules（可理解的规则集），经过解释之后转换成系统配置或执行的规则（rules）集合。
- 行为分析（Behavior）：基于符合业务安全策略的企业白环境，进行异常行为分析，从而发现未知攻击及看似“正常操作”的“非法行为”。
- 协作能力（Collaboration）：不同产品之间的协作，不同用户之间的协作。

以价值导向看业务安全

成都分公司　周文宝

摘要：优秀的 IT 为企业创造价值且推动业务发展，前提是 IT 能够理解业务需求。故无论是从信息化基础架构还是业务角度，随着企业发展和业务变革持续优化，IT 的重要性还将不断提升。但是，IT 不能离开业务创造价值，随着企业信息化建设水平的提高和对信息安全认识的不断加深，会更加重视安全与业务的融合。那么，如何进行业务安全需求分析、业务风险和影响性分析、业务安全可视化分析等，本文将进行简单阐述，以期能够抛砖引玉，对业务安全评估起到推动作用。

关键词：IT　价值　业务　安全

一、IT 基础与价值

（一）价值交付协同关系

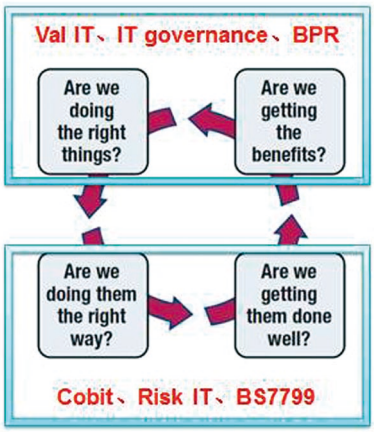


图 1：价值交付协同关系

1. 战略问题：我们在做正确的事情吗？

• 投资是否与预期设想一致？

• 投资是否符合商业原则？

• 投资是否在可承担的成本和接受的风险水平的基础上，使得利润最大化？

2. 架构问题：我们做的正确吗？

• 投资是否符合公司的架构？

• 投资是否与公司的架构原则一致？

• 投资是否与其他倡议一致？

3. 价值问题：我们获得收益了吗？

• 是否对预期收益有一个共同的清晰的理解？

• 是否针对实现收益有明确的问责机制？

• 是否有相关指标？

• 是否在投资的全经济生命周期中有一个有效的收益实现流程？

• 是否有相关安全保障机制？

4. 交付问题：我们完成的好吗？

• 是否有有效的制度进行管理、交付以及管理变革流程？

• 是否有过硬的和现有的技术及业务资源可以提供？

上面是 Val IT 价值交付协同关系，Val IT 是对信息系统和技术控制目标的扩展和补充。其着眼于投资决定以及收益的实现。但

是从我们关心的价值问题域来看，其需要有相关的问责机制，相关的安全保障机制来实现。

（二）IT 治理

所谓 IT 治理，关键在于授权和控制并举，解决应该做出什么决策，谁来做决策，如何做出决策和监督，促成 IT 创造有利于战略的价值。

IT 治理主要涉及两个方面：IT 要为企业交付价值，IT 风险要降低。前者受 IT 与企业的战略一致性驱动，后者由责任义务落实到企业驱动。国际 IT 治理的五个领域，即：战略一致、价值交付、资源管理、风险管理、绩效测评都与 IT 价值相关联。其中两个收益方面内容（即：价值交付和风险管理）直接与价值有关；那么如何做好风险管理，从哪些方面去识别风险、如何去识别风险、如何去做业务风险评估、如何进行事件发生的可能性及后果分析等都需要相应的手段和工具。这样才能够对 IT 基础做到有效治理，实现战略价值。

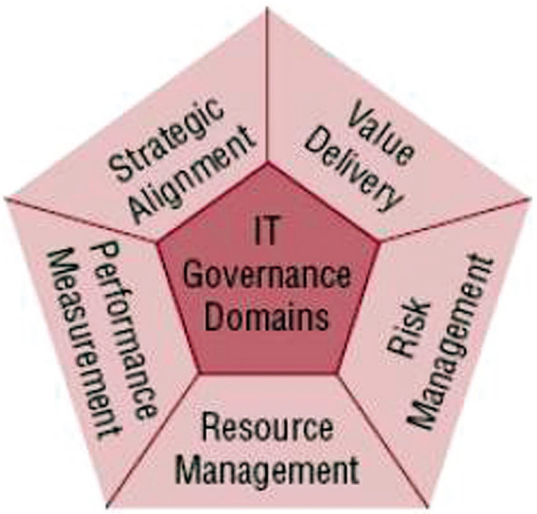


图 2：IT 治理五个域

（三）企业 IT 系统架构

1. 企业架构组件

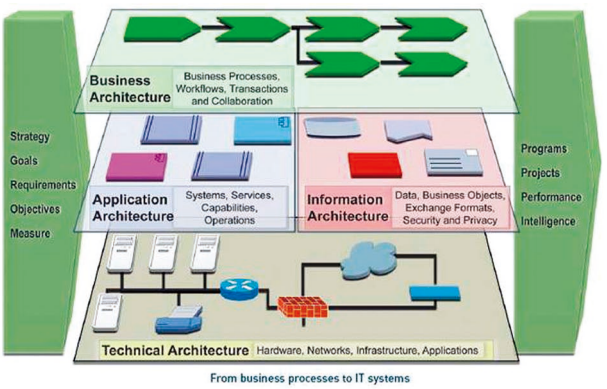


图 3：企业架构组件

企业业务的实现需有相应的业务流程，业务流程的实现需要应用系统支持，技术基础设施是运行应用的基本保障，信息则是在业务流程和应用中运转。

- 业务架构（business architecture）：业务战略、组织结构以及业务流程
- 应用架构（application architecture）： 服务、应用

- 信息架构（information architecture）：业务对象和数据
- 基础设施架构（infrastructure architecture）：硬件、网络和软件环境

2. 企业总体架构视图

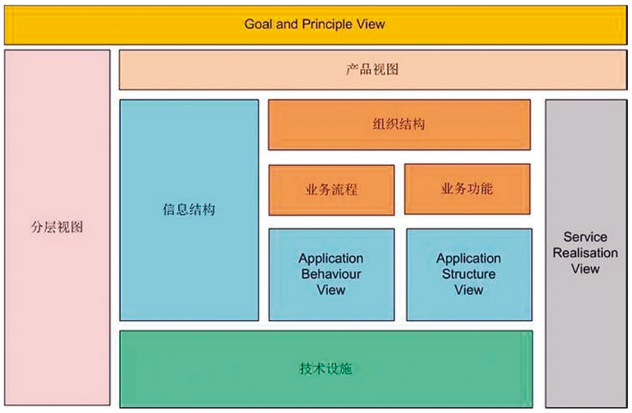


图 4：企业系统架构视图

所以我们在评价风险的时候要结合企业的 IT 系统架构组件全面的识别风险，通过评价基础架构风险，从而识别业务系统、应用系统风险，进而识别企业总体风险。

二、业务安全风险评估理论

（一）风险概念

在现实世界中，自然灾害和意外事故客观地存在着，但这些不幸事件何时何地发生、致害于何人、造成何种程度的损失，通常是无法预知的。因而，对于特定的事物而言，人们对自己是否会遭遇不幸事件、受到多大的损失，处于一种不确定的状态。于是，特定的事物对于特定的人们就构成了风险。

风险的含义可以从多种角度来考察。首先，风险同有目的的活动有关。人及企业从事活动，总有一定的预期结果，如果对于预期的结果没有十分的把握，就会认为该项活动有风险。其次，风险同将来的活动和事件有关。已经结束了的活动或项目，既成事实，后果已无法改变。对于将来的活动、事件或项目，就会采取一定的防护手段。再次，如果活动或项目的后果不理想，甚至是失败，人总是要想能否改变以往的行为方式或路线，把以后的活动或项目做好。另外，当客观环境，或者人的思想、方针或行动路线发生变化时，活动或项目的结果也会发生变化。这样，风险还与上述变化有关。若世界永恒不变，人们就不会有风险的概念。所以有了风险，才有安全的存在。

（二）企业经营管理体系架构

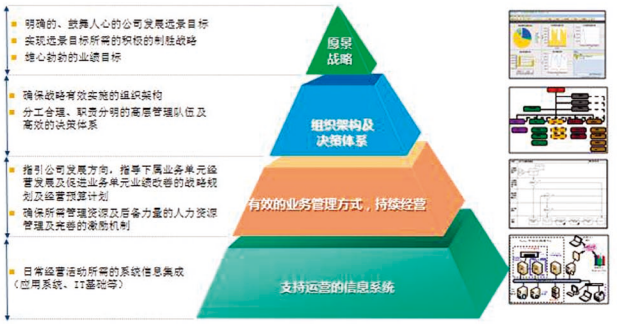


图 5：企业经营管理体系

企业经营管理体系分为四层架构:

第一层：战略愿景层，企业要有明确的、鼓舞人心的公司发展

愿景目标；实现愿景目标所需的积极的制胜战略；雄心勃勃的业绩目标。

第二层：组织架构及决策体系，企业需具备确保战略有效实施的组织架构；分工合理、职责明确分明的高层管理队伍及高效的决策体系。

第三层：有效的业务管理方式并持续经营，指引公司发展方向，指导下属业务单元经营发展及促进业务单元业绩改善的战略规划及经营预算计划。

第四层：支持运营的信息系统，即应用系统、IT 基础。

（三）企业信息安全目标框架

图 6：信息安全目标框架

由企业经营管理体系架构，从而我们得出企业信息安全目标框架。企业信息安全目标框架主要分为三层:

第一层：企业的安全目标，主要为战略目标、经营目标、报告目标、合规目标。

第二层：业务的安全目标，主要为内外环境、持续经营等。

第三层：信息系统的安全目标，主要指标为保密性、完整性、可用性、符合性、可靠性、效率。

我们在进行传统风险评估的保密性、完整性、可用性评估的同时，针对业务安全风险评估的要求，增加信息系统的效率、符合性和可靠性三个指标的分析，根据信息系统的六个标准的分析结果去评价业务的影响和企业目标的影响。

业务影响分析需要结合业务和企业战略目标进行业务影响的分析，所以方法上参考了 COSO 标准的企业风险管理框架和 COBIT 标准的信息及相关技术控制目标框架。

信息系统、业务、企业三个层次的目标关系如下:

信息系统的效率、保密性、完整性以及可用性对应了重要信息系统相关的业务目标，也对应了企业目标的战略目标和经营目标，主要是经营目标，因此信息科技安全风险对重要信息系统的业务目标的影响直接影响到企业目标的战略目标和经营目标，在业务风险分析时，重要信息系统相关的业务目标直接作为企业目标的经营目标来评价。

信息科技的符合性标准，直接对应企业目标的合规目标，因此信息科技的合规风险也是企业目标的合规风险。

信息科技的可靠性目标，直接对应企业目标的报告目标，因此信息科技的可靠性风险也就是企业目标的报告风险。

（四）应用系统运行风险要素分析

1. 应用系统脆弱性

表 1：应用系统脆弱性

类型	识别对象	识别内容
技术脆弱性	物理环境	从机房场地、机房防火、机房供配电、机房防静电、机房接地与防雷、电磁防护、通信线路的保护、机房区域防护、机房设备管理等方面进行识别。
	网络结构	从网络结构设计、边界保护、外部访问控制策略、内部访问控制策略、网络设备安全配置等方面进行识别。
	系统软件（含操作系统及系统服务）	从补丁安装、物理保护、用户账号、口令策略、资源共享、事件审计、访问控制、新系统配置（初始化）、注册表加固、网络安全、系统管理等方面进行识别。
	数据库软件	从补丁安装、鉴别机制、口令机制、访问控制、网络和服务设置、备份恢复机制、审计机制等方面进行识别。
	应用中间件	从协议安全、交易完整性、数据完整性等方面进行识别。
	应用系统	从审计机制、审计存储、访问控制策略、数据完整性、通信、鉴别机制、密码保护等方面进行识别。
	存储系统	
管理脆弱性	技术管理	从物理和环境安全、通信与操作管理、访问控制、系统开发与维护、业务连续性等方面进行识别。
	组织管理	从安全策略、组织安全、资产分类与控制、人员安全、符合性等方面进行识别。

2. 应用系统威胁源

表 2：应用系统威胁源

种类	描述	威胁子类
软硬件故障	由于设备硬件故障、通讯链路中断、系统本身或软件缺陷，造成对业务实施、系统稳定运行的影响。	设备硬件故障、传输设备故障、存储媒体故障、系统软件故障、应用软件故障、数据库软件故障、开发环境故障。
物理环境影响	断电、静电、灰尘、潮湿、温度、鼠蚁虫害、电磁干扰、洪灾、火灾、地震等环境问题或自然灾害。	如前所述。
恶意代码和病毒	一旦入侵系统后进行自我复制、自我传播，对系统构成破坏的程序代码。	恶意代码、木马后门、网络病毒、间谍软件、窃听软件。
越权或滥用	通过采用一些措施，超越自己的权限访问了本来无权访问的资源，或者滥用自己的职权，做出破坏信息系统的行为。	未授权访问网络资源、未授权访问系统资源、滥用权限非正常修改系统配置或数据、滥用权限泄露秘密信息。
网络攻击	利用工具和技术，如侦察、密码破译、安装后门、嗅探、伪造和欺骗、拒绝服务等手段，对信息系统进行攻击和入侵；也可能通过系统发动大规模攻击其他系统。	网络探测和信息采集、漏洞探测、嗅探（账户、口令、权限等）、用户身份伪造和欺骗、用户或业务数据的窃取和破坏、系统运行的控制和破坏、对系统进行攻击或利用系统进行攻击。
物理攻击	通过物理的接触造成对软件、硬件、数据的破坏。	物理接触、物理破坏、盗窃。
泄密	信息泄露给不应其了解的他人、受到端口扫描或漏洞扫描探测器扫描。	内部信息泄露、外部信息泄露。
篡改	非法修改信息，破坏信息的完整性使系统的安全性降低或信息不可用。	篡改网络配置信息、篡改系统配置信息、篡改安全配置信息、篡改用户身份信息或业务数据信息。

3. 风险事件

表 3：风险事件

事件种类	事件描述
系统应用数据保密性丧失	指信息没有按给定要求泄露给非授权的个人、实体或过程。
系统应用数据完整性丧失	破坏信息的完整性使系统的安全性降低。
系统应用数据可用性丧失	破坏信息的可用性使系统的信息不可用。
系统应用功能不完善或失效	应用服务功能完全停止、部分应用功能失效。

4. 风险后果

表 4：风险后果

1 级目录	2 级目录	3 级目录
业务欺诈	内部欺诈	行为未经授权
		盗窃和欺诈
	外部欺诈	盗窃和欺诈
		系统安全性
信息泄露	国家秘密信息	国家事务的重大决策事项、外交或外交活动中的秘密事项以及对外承担保密义务的事项、国民经济和社会发展中的秘密事项、科学技术中的秘密事项、维护国家安全活动和追查刑事犯罪中的秘密事项、其他经国家保密工作部门确定应当保守的国家秘密事项。
	客户隐私信息	能够对用户进行个人辨识或涉及个人通信的信息。
	商业敏感数据	商业企业内部数据、业务渠道数据、市场数据等。
管控失效	策略失效	
信息损毁	决策支持信息	战略决策、分析决策、市场决策
	客户价值信息	
	管理统计信息	
	交易记录信息	网上交易、柜台交易
信息错误		
信息科技系统事件	信息系统	硬件
		软件
		网络与通信线路
		动力输送损耗 / 中断
实物资产破坏	灾害和其他事件	自然灾害损失
		外力（恐怖袭击、故意破坏）造成的人员伤亡和损失。

5. 风险损失

表 5 风险损失

1 级目录	2 级目录	3 级目录
损失主体	股东	大股东
		小股东
		监事
	客户	个人
		企业
		政府
		机构
		非盈利性组织
	合作者	机构
		个人
	经营者	董事会
		高级管理层
		员工
损失形态	资产成本	筹集和使用资产而支付的费用
	声誉成本	社会评价降低而对行为主体造成危险和损失的可能性
	法律成本	
	监管惩罚	准入限制（高管、业务、机构）
		罚款（个人、企业）
		取消从业许可（个人、业务、机构）
		强制企业内部问责
	商业机会	投资者
		客户
		合作者
	竞争能力	市场、投资平台、流动资金

（五）风险评估要素关系

1. 传统风险评估要素关系

由图 7 看出，该要素关系模型涉及 7 个要素，主要为资产 - 资产价值 - 威胁 - 脆弱性 - 风险 - 防护需求 - 防护措施，其核心为风险，聚焦为资产。

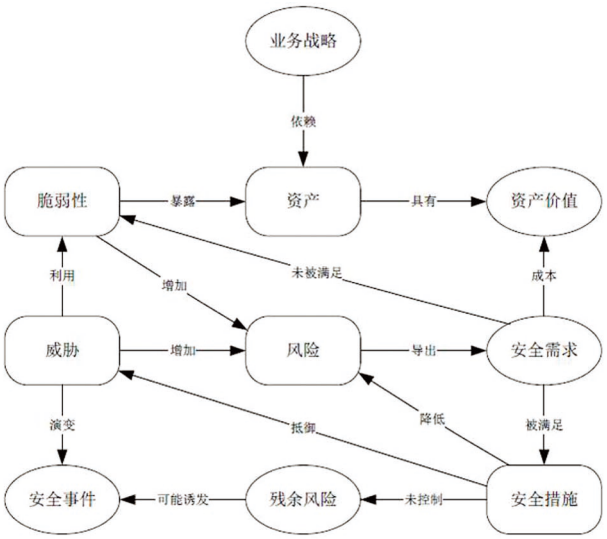


图 7：风险评估要素关系

2. 应用信息系统运行风险管理关系模型

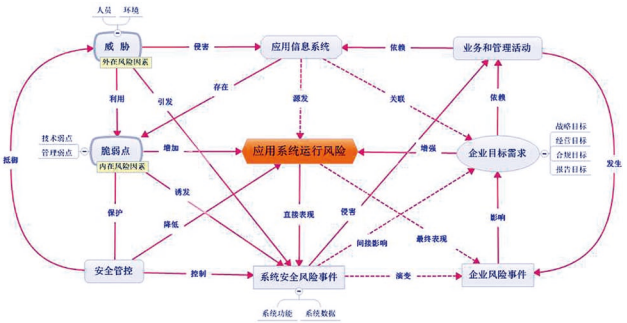


图 8：应用信息系统运行风险管理关系模型

由图 8 看出，该要素关系模型涉及 9 个要素，主要为业务和管理活动 - 企业目标需求 - 应用信息系统 - 威胁 - 脆弱点 – 应用系统运行风险 - 系统安全风险事件 - 企业风险事件 - 安全管控，其核心为风险，聚焦为业务和管理活动。

应用信息系统运行风险要素关系图中的风险要素及属性之间存在着以下关系：

1. 企业目标的实现对企业的业务和管理活动具有依赖性，依赖程度越高，要求业务及管理活动风险越小；

2. 业务和管理活动依赖应用信息系统实现其价值，管理不当则会发生企业风险事件；

3. 应用信息系统间接关联企业目标需求的实现，企业目标需求的实现对应用信息系统依赖程度越高，则其价值越大；

4. 应用信息系统存在固有脆弱点，而其会增加应用系统运行风险；

5. 威胁利用应用信息系统的脆弱点，暴露出系统安全风险事件；应用信息系统的脆弱点越多则风险越大；

6. 威胁同时会侵害应用信息系统，应用信息系统面临的威胁越多则风险越大；

7. 应用系统运行风险源发应用信息系统，其直接表现为系统安全风险事件，且可间接演变为企业风险事件，企业风险事件则直接影响企业目标需求的实现；

8. 系统安全风险事件会侵害业务及经营管理活动，会直接引发企业风险事件，最终会影响企业目标的实现。

9. 威胁及脆弱点的存在导出安全管控的需求，安全管控措施可

以降低应用系统运行风险；同时控制系统安全风险事件的发生；

10. 安全管控可抵御威胁，降低风险。

（六）业务安全风险评估方法及流程

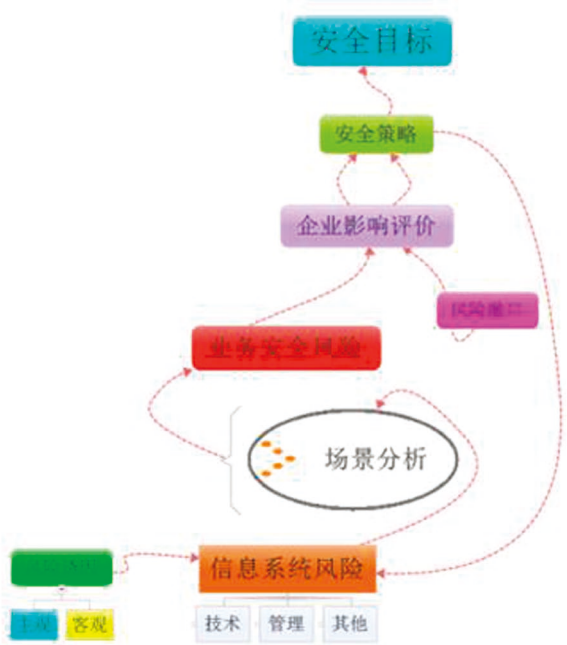


图 9：业务安全风险评估方法及流程

如图 9 所示，业务安全风险评估流程是建立在传统信息系统风险评估的基础之上。对信息系统风险进行重新加工，将离散的数据结合风险诱因进行组合和标准化分析，模拟场景分析识别出信息系统风险可能会引发的业务安全风险。再结合风险敞口进行企业的影响评价，最终制定出安全策略来实现企业的安全目标。

三、传统安全评估与业务安全评估差别

表 6 传统安全评估与业务安全评估

项目	传统安全评估	业务安全评估
汇报对象	系统安全运维、安全技术负责人	业务负责人、企业管理层
评估的输入	脆弱性识别、威胁识别、资产价值、控制措施识别等	传统安全评估的风险、业务数据、场景案例等
成熟度	较高的成熟度、有众多国家、国际标准可以参考，基本不依赖于行业，成功可能性更高	成熟度较低，需结合众多标准建立模型，高度依赖于行业，成功可能性较低
可复制性	高	适中
对人员的要求	需熟悉各种操作层面的技术	不仅要熟悉操作层面技术，还要对业务流程、企业管理等较为了解
对客户体现价值	视发展阶段需求而定（目前阶段较低）	高
客户需求度	适中	高

参考文献

IT 治理——标准、框架　清华大学出版社

The Business Case Guide: Using Val IT 2.0　ISACA

信息安全风险评估规范

巴塞尔新资本协议

COSO/ERM 整体框架

从数据泄露事件看客户信息保护

北京分公司　白雷

摘要：本文通过对近期发生的一系列数据泄露事件的分析，剖析了个人隐私信息的泄露对社会引起的危害，通过比较国内外对个人隐私信息保护的立法与保障过程展现相关企事业单位对客户信息进行保护的必然趋势，提出了企业进行客户信息保护从业务与系统层面入手，重技术与管理保障的综合防护策略，并从以人为本的角度出发，指出客户信息保护工作在相关政府部门、企事业单位信息安全建设中的重要性与长期性。

关键词：数据泄露　隐私保护　信息安全

引言

客户信息泄露，近期变得相当的敏感，去年年底 CSDN 信息泄露事件发生之后，工信部、运营商与金融行业都非常的重视，银监会及央行等主管机构，也曾多次下文强调这方面的工作。但央视“3.15”晚会上曝光的事件，仍旧让相关企业蒙受了声誉上的损失，更是祸及其股价应声下跌。事实再一次说明，只有有效保护客户信息，才能将企业的信息安全工作落到实处。最近国家也加强了信息安全的保护的力度，今年的 4 月 20 日，公安部统一部署北京、河北、山西等 20 个省区市公安机关开展集中行动，摧毁覆盖全国、涉案人员众多的侵害公民个人信息犯罪网络。据公安部透露，此次专案行动查获的源头中，有公务员，也有政府部门协助人员，还有公司职员。涉及到的部门和公司，既有工商局，也有民政局等部门。公司职员中，有各大电信运营商的职工，还有来自银行、民航、保险等行业的员工。至 4 月 24 日各地公安机关共抓获犯罪嫌疑人 1700 余名，挖出非法出售公民个人信息的“源头”38 个，摧毁侵害公民个人信息的数据平台和“资源大户”161 个，打掉从事非法讨债、非法调查等的非法调查公司 611 个。由此，对利用客户信息犯罪行为进行了严厉打击。但关注信息来源，加强企业的客户信息保护应是铲除这类犯罪的关键，应当从企业级信息安全保护入手加强客户信息保护的工作。

一、客户信息　触及个人的敏感隐私

信息泄露类安全事件的发生都源于客户信息具有利用的价值，对利用者具有相当的吸引力，从这方面分析，针对个人的重要程度而言被泄露的客户信息主要有四种类型:

一种类型的客户信息是存储在银行、证券、保险、基金公司等金融企业中的个人金融信息，包括: 个人信用记录、金融账户信息等，当前大部分严重的信息犯罪均来源于对此类信息的恶意利用。

另一重要类型是电信运营商所拥有的个人电信信息，包括：电话号码、电信资费信息、个人通讯信息和个人定位信息等，最近打击的调查公司、非法讨债等主要是利用此类个人隐私谋利的。

此外，以 CSDN 为代表的互联网企业所持有的个人认证信息，

包括：用户名与密码等，此类信息主要被用来猜解重要账户，构建水军或僵尸网络、进行网络渗透等。

还有一类是分布在工商、学校、医院等各类政府机关、企事业单位的个人身份信息，包括：身份证、家庭住址、信息方式、企业注册信息等，这类信息大量用于直复营销、数字营销等。

这四类信息依据数据量和危害度的不同总结如下图 1 所示：

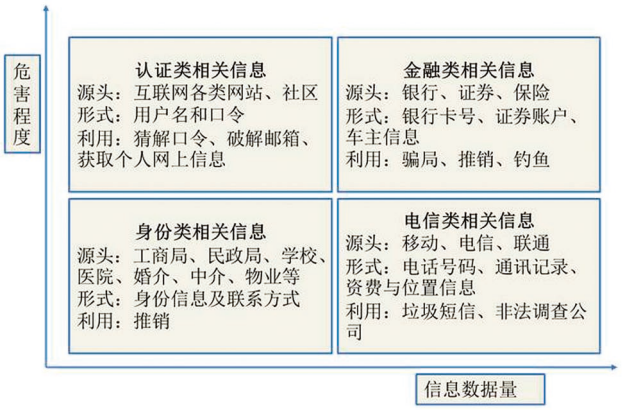


图 1 ： 个人信息的主要类别及可能被利用的途径

总之，所有这些客户信息都触及到社会公众的个人隐私，个人隐私被泄露、被恶意的传播、被犯罪份子利用的问题，已经成为影响社会秩序和公众利益的重大问题。

二、暴利滋生　个人信息黑色产业链

正是由于存在恶意利用个人信息的不正当行为，才导致该产业存在着一个环环相扣的利益链条。而其中的暴利，就诱使少数个人铤而走险，从组织内部非法获取信息，进行牟利。在这一黑色产业链中存在诸多利益活动，并涉及到各方面的利益团体。绿盟科技安全

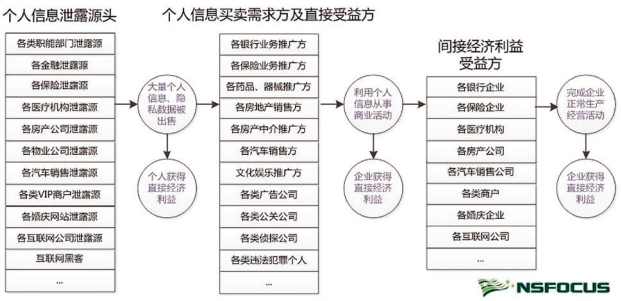


图 2 ： 个人信息泄露与信息买卖经济利益链条

专家刘凯分析给出了简要的利益链条活动关系图。

从图 2 中，可以看到不同活动中的利益团体是不同的，一般包括：

- 客户信息的持有者，例如：银行、保险、房地产、汽车经销商等大中型企业；
- 非法获得信息者，例如：网络黑客、各企业从业人员、外包客服企业、中介企业等；
- 利用收集、传播信息牟利者，例如：某被曝光的公司等从事网上、线下营销的团体；
- 恶意利用信息实施犯罪者，例如：发垃圾邮件诈骗集团、网络钓鱼者、盗取网银账号犯罪集团、网络水军等。

上述利益团体分工明确，传播手段多样，其中典型的传播牟利活动，例如：

- 直复营销：反馈式邮件销售、手机广告销售、电话营销、企业名录销售、在线信息销售、在线商业资料库销售；
- 数字营销：电子邮件销售、网络销售、移动销售、公关活动；
- 调查公司与非法讨债：利用移动信息定位人员，利用通信内容

分析个人行为。

综上，这一系列利益链条、利益团体、典型的信息传播牟利活动的出现伴随信息化的发展而形成几个明显的特点:

• 在现有法律的真空地带，或法律形成的真空期开展不法活动以图逃避打击;

• 个人信息基本上是存储在信息系统中的电子信息，泄露后通过网络传递、网上进行交易并且信息多为大批量多次输送;

• 利用网络的匿名的特点开展活动并隐蔽自身的真实身份;

• 内外勾结的利益链条方式形成上下游的关系，内部人往往是利用正常的职务行为将客户信息传播出去牟利，而其所在个别单位却疏于管理;

• 对个人而言只产生小额损失或轻微伤害，个人维权非常困难。

由于利用了这些特点，个人信息泄露事件不断爆发，个人信息保护形势日趋严重，个人信息保护行动已经迫在眉睫。

三、全面分析　国内与国外差距明显

客户信息相关的安全事件频发是源于国内对信息安全保护的研究的忽视与立法缺乏，我国与较早开展保护个人信息的国家差距表现在以下几个层面:

首先，制度缺乏，隐私保护无法可依，对比欧洲和美国立法保护个人隐私方面，我国还有不小的差距。早在 1984 年英国议会就通过了《数据保护法》，并于 1998 年对该法进行修订；此后，英国陆续通过了《调查权法》、《通信管理条例》和《通信数据保护指导原则》等一系列旨在保护公民个人信息的法律。美国很早就对个人隐私进行了立法，并且 2005 年美国又通过了一批法律加强对个人信息的保护，如《隐私权法》、《信息保护和安全法》、《防止身份盗用法》、《消费者隐私保护法》和《社会安全号码保护法》等；除此之外，美国各州还制定了一些保护本州公民隐私的细化法律，例如，2008 年 1 月 1 日生效的马里兰州《个人信息保护法》中做出明确规定，那些持有消费者或客户信息的商业机构或非营利机构，必须维持“适当水平的安全措施”，以防信息泄露。到目前为止中国周边的亚太地区很多国家也已经颁布了相关法案保护个人隐私，如澳大利亚、新西兰、日本、韩国、印度、菲律宾和我国的香港、台湾地区。而我国目前还没有一部个人隐私保护的法律出台，使当前的信息保护工作没有有力的法律依据，个人的维权也就非常困难。

其次，国内对隐私信息的保护机制的研究不足，没有可行的安全信息保护的理论方法。反观国外情况，除立法保护外，美国等国家更依赖于市场交易中行业与公司的自律，各自制定隐私保护行为准则或取得民间的认证制度来适应消费者的要求，主要形式包括：建议性的行业指引；网络隐私认证；技术保护模式等，相关行业或企业可以选择采取较为灵活的策略。而国内相关保护模型、业务分析方法、保护原则和策略方面的研究开发都处于比较初级的阶段并且不系统不成体系，要推广信息保护还有相当长的路要走。

此外，各相关企业对个人隐私保护工作的认识程度和重视程度不足，行业性的保护规范标准出台和落实的力度不够，也是导致严重安全事件反复发生的重要原因。

四、防范泄露　业务和系统双管齐下

针对个人信息泄露的事件，绿盟科技较早开展了相关的防泄露课题的研究，经过近两年在金融和运营商行业的实践和总结，从

全面应对客户信息保护的高度出发，推出了一套个人信息保护安全建设咨询服务。在服务体系的建制过程中，充分认识到信息安全中的攻与防、泄露与保护、破坏与建设是一对长期存在，魔道相争的关系，从理论的层面上讲，是应当要建立长效的、纵深的、持续提升的、基于风险的安全体系保障信息安全。

因此，从涉及客户信息的相关行业的业务环节保护方面考虑，我们建议各相关企业从业务的不同角度着手信息保护工作:

• 最终用户：应提高个人信息安全保护意识，积极防范各类信息泄露和欺诈，特别需加强网上购物及交易安全防范意识;

• 企业经营者：应加快建设企业内部个人隐私保护、敏感信息防护、数据防泄密等数据安全体系，担负起企业应尽义务，承担企业社会责任;

• 信息实名制行业：应大力加强行业自律、提高行业整体人员职业操守。同时，制定、实施符合本行业特性、切实可行的技术控制手段，防范内部作案风险，此类企业如新浪微博、淘宝等;

• 各行业监管部门：应落实合规制度要求，加强对行业内外风险监控、特别对内部作案事件严查死守，严厉打击，将行业内部合规检查精细化、标准化、常规化;

• 国家司法机关：进一步完善国家法律法规，将个人信息买卖等违法行为相关条例细化，出台针对性强、可操作性强的相关条例、司法解释及说明，推动我国个人隐私、信息保护方面的法制化进程。

从信息系统管理的角度，我们建议应当尽快建立各相关企业，特别是银行、保险、电信行业的个人信息防护安全体系。根据当前信息安全行业在个人信息保护方面的最佳实践案例，建议相关企业可以依照以下步骤展开个人信息的安全保护工作:

• 识别开：相关企业应首先识别自己业务系统中涉及的全部敏感个人信息，并分类定级，以便于针对性地采取保护措施;

• 管理全：尽快出台管理措施，加强对企业内部人员的管理与控制，提升个人信息保护的防范意识，只有管理到位才能收到实效;

• 防护住：通过加密、认证、访问控制、截断威胁路径等技术手段防止个人信息的泄露;

• 监测出：能够通过监控手段，发现异常的信息传播，以便及时采取控制措施防止进一步影响的发生和损失的扩大;

• 追踪到：通过安全审计措施，使得一旦出现个人信息从本企业泄露的事件，可以通过追查日志记录能够发现泄密者，有效追踪非法行为以便进行严厉的惩办。

同时，作为一项专业的安全防护工作，完全依靠现有行业自身的技术团队进行保障可能还不能达到很好的保护效果，应当引入专业的安全厂商的安全设计、安全架构和信息保护安全咨询的服务，共同开展安全研究，构建成熟的安全保障体系。

五、综合评估 技术和管理加强审计

要加强对客户信息安全的保护，可以参照相关的实施路线图开展概念层的分析研究、设计层的规划设计和实现层的措施落实。总体思想遵照以业务为核心、以技术为支撑、以管理为保障、以急用先上为原则的方案架构，在对某金融行业的客户进行个人信息保护的建设项目

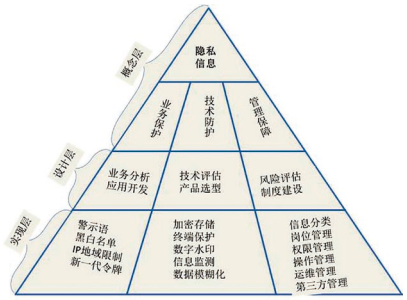


图 3 ： 个人信息保护安全建设体系

中我们引入了如图 3 的安全建设体系框架。

在具体为企业提供支持方面，绿盟科技客户信息保护安全建设咨询服务分别从业务信息资产识别、安全威胁建模和信息系统安全脆弱性的角度出发进行客户信息风险评估，并针对评估结果给出具体的、可执行的防护措施，具体的推荐措施涵盖以下的五个方面:

第一、个人信息保护专项整治

针对不同的业务系统，专门采取应对当前时期的专项应急措施，其中包括：系统中添加客户警示语、采用下一代令牌、监控钓鱼网站、对用户登录的 IP 进行地域的限制以及专项黑名单机制的实现等。

第二、系统安全技术防护方面

针对个人信息保护的问题，加强物理层、网络层、系统层、应用层的安全防护，依据分区域、按等级、多层次的防护思想进行安全规划、安全评估、安全加固与安全维护，贯彻预警、防御、监测、恢复的多重安全保护的技术策略，沿着威胁攻击的路径部署安全措施。

第三、数据安全技术增强部署

对过去已有的安全技术实施进行改进与增强，其中进行改进的方面包括密码算法、加密方式、U 盘管理、统一认证与集中审计等技术机制；并考虑实施适用于新安全形式的安全技术手段，包括数字水印技术、数据模糊化技术等。

第四、基于业务的客户敏感信息保护增强

首先对业务系统中的客户信息进行识别、分类、定级；然后通过业务分析梳理数据流程；进行受攻击面分析和威胁建模；最后通过综合风险评估形成业务保护方案，并在业务系统中全面落实实施。

第五、安全管理体系建设

安全管理体系建设过程是依据风险的原则，分析客户信息安全管理面临的威胁与管理自身的脆弱性，然后通过体系化、制度化、流程化、表单化和信息化的方法全面完善内部管理，并进一步与系统管理体系有机的融合，形成面向信息保护的安全管理体系。

经过上述五个方面的建设，绿盟科技已经能够全方位地解决企业对于个人信息的识别、管理、防护、监测和追踪等各方面的安全问题，有能力通过安全服务使客户个人信息保护方面的安全水平得到全面的提升。

六、以人为本 客户信息保护持续化

“以人为本”思想是科学发展观理论的核心，与每个个人都息息相关的隐私保护问题应当为全社会重视，特别对于政府部门及涉及个人隐私的企事业单位而言，客户信息保护应当作为一项长期性的安全重点工作。作为专业的安全厂商，绿盟科技长期以来致力于企业级敏感信息保护课题的研究，已经形成了比较成熟的基于业务安全评估与整体信息保护的建设方案，并且在金融、电信等重要系统得到了成功的应用。我们相信个人信息安全工作在我国会逐渐深化开展，作为专业安全厂商，同样本着以人为本的原则，我们也会努力为企业作好安全技术支持和相关的安全咨询服务。

全面解析中小银行信息安全合规管理（二）

行业技术部　徐一丁

摘要：本文力求为中小银行客户全面地分析信息安全建设中的合规问题，帮助银行信息安全管理者和技术人员认识理解监管部门工作，作为本银行信息安全工作的参考。

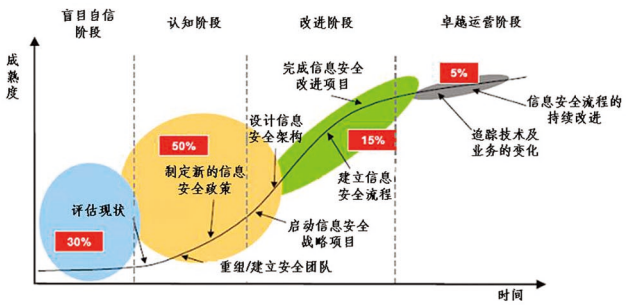
关键词：中小银行　信息安全监管　信息科技风险管理　等级保护

本文所指中小银行，主要指城商行系统的银行，也适用于各省农信（联）社、农商行等。

（续上期）

银行的信息安全发展成熟度

中小银行首先应充分识别自身信息安全的发展水平，以此为依据，设定当前信息安全体系建设的目标，切忌盲目冒进。请先看下面的图：



注：图中红色标注的部分代表了福布斯2000强企业在不同阶段的百分比分布
来源：Gartner Inc. 2006年1月

该图的信息来源是 Gartner（国际著名咨询机构，专业从事信息技术研究和分析）。它展示了一个机构在信息安全道路上前进的不同阶段，分别是盲目自信、自我认知、改进和卓越运营，基本体现了从懵懂到成熟的发展过程。同时这个图也说明，在 2006 年高达 80% 的世界级大企业都处在信息安全不成熟阶段（盲目自信、认知），15% 的小部分企业正在有计划地进行改进，只有 5% 的企业进入卓越运营的成熟阶段。

请您根据下面的描述，判断一下自己的银行正处于哪个阶段？

- 盲目自信阶段

普遍缺乏安全意识，对企业安全状况不了解，未意识到信息安全风险的严重性。

- 自我认知阶段

通过信息安全风险评估，企业意识到自身存在的信息安全风险，开始采取一些措施提升信息安全水平。

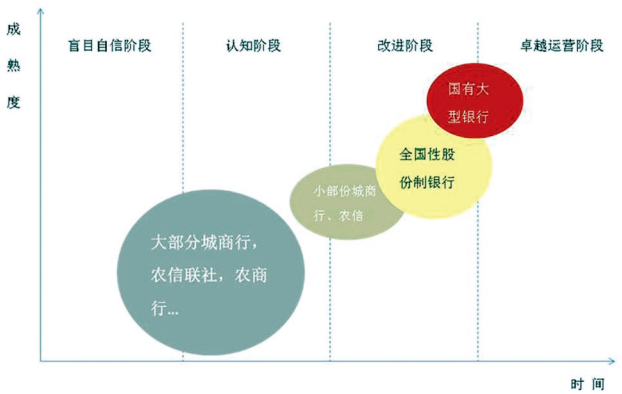
- 改进阶段

意识到局部的、单一的信息安全控制措施难以明显改善企业信息安全状况，开始进行全面的信息安全架构设计，有计划地建设信息安全保障体系。

• 卓越运营阶段

信息安全改进工作完成后，在拥有较为全面的信息安全控制能力的基础上，建立持续改进的机制，以应对安全风险的变化，不断提升安全控制能力。

根据绿盟科技在银行业安全服务中了解的情况，我们也绘制了一个类似的参考图:



从上图可以看出，国内大部分中小银行（城商行、农信社、农商行等）还处于不成熟的发展阶段，还没有全面、清晰地了解自己所面临的问题，没有整体的安全规划和相应可执行的计划；小部分中小银行已经进入了认知阶段后期或改进阶段前期。

自我定位，再明确安全目标

下面这些非安全的直接内部因素也很大程度上影响着银行信息安全建设，决定了我们在每一个阶段应该做什么事:

• 行领导们对信息科技工作的看法，是否都积极支持?

• 信息科技治理、运行与管理水平。

• 银行员工的信息安全意识。

……

这些方面的条件都合适了，信息安全建设才有必要的基础。现在存在着一种现象，一些银行在信息科技治理和运行方面还存在着很多问题，而对信息安全建设有很大的期望，客观讲这不太现实。例如，有些银行还没有建立必要的网络和系统日志收集管理的机制，就不能期望安全人员能在安全事件发生时分析是否有恶意的操作行为，更谈不上安全预警—前者是信息科技运行问题，后者是安全问题，后者依赖前者。安全是上层建筑，需要依靠基础设施的完善。

因此，对那些还处于认知阶段或以前的银行，建议以合规为首要目标。首先全面透彻地了解自身现状，制订合理的改进规划和计划，在一个信息安全建设的周期内（例如 3 年），先进入并稳定在改进阶段，而不是直接进入卓越运营阶段。简单点说就是先学走，再学跑。

完成合规目标，能够顺利通过监管部门的检查，本身就是信息安全工作的一个成功。同时合规工作是依据比较成熟的标准规范来展开的，也可以使那些本来安全水平不高的银行，以较低的成本去获得一个标准的安全体系。以此为基础，再通过 PDCA 方法（循环改进）逐步地调整，使安全体系真正适合银行的情况。

如何同时满足 ISO27001 和等级保护两套标准?

这是安全合规的首要问题，其实也是安全体系设计的首要问题。等级保护基本要求和 ISO27001 都比较成熟和完善，怎么考虑和处理呢? 建议参考如下的步骤:

首先，从二者中确定一个信息安全标准为依据，等级保护或

ISO27001 都可以。简单一些考虑，如果下一个来检查的是人民银行或公安，那可以用等级保护为依据，反之可以先考虑 ISO27001。从顺利实施的角度看，把“两套标准揉在一起”的想法不可取，在安全体系设计的阶段应以某一个标准为主。

其次，以选定的标准为依据，开始设计自己的安全体系。既然这是一个有针对性的体系，就应当对自身的安全状况有全面清晰的了解，所以最好能先做个整体的安全评估，然后再开始设计。这个阶段的设计完成之后，银行会得到一张体系蓝图，安全工作应分哪些方面，每一个方面包含哪些工作，等等。

最后，是将这个设计中的各项安全措施与另一个安全标准中的要求进行对比分析，查漏补缺，补充而后形成一个完整的体系，整合之后进行全面分析，确定最终的安全设计方案。这样的方案就可以同时满足两套标准的要求。我们在下一节介绍对比分析的方法。

如何进行两套要求的对比分析

假设某银行已经按照等级保护基本要求，进行了安全体系设计，这时安全体系会分为安全技术和安全管理两大方面，它们是紧密联系而不可分割的。安全技术方面包括物理、网络、主机、应用、数据几个层次，安全管理方面分为制度、机构、人员、系统建设、系统运维几个部分。

建议制作一个对比分析表，类似下图这样的:

本图展示的是对分析 ISO27001 访问控制领域的控制措施，图中的第 3 列由上至下依次列出该领域的要求，而第 6 列到第 17 列依次说明了根据等级保护基本要求，该银行设计的访问控制方面的措施。根据等级保护要求设计完成后，在技术的每个层面、管理的每个部分都可能有访问控制的对应措施，我们需要将这些措施一一梳理出来，将其名称从第 6 列开始，由左至右填充进去。某些部分可能有多个访问控制措施，如图中的网络和主机层面就各有 2 个访问控制措施，实际环境中有可能更多。

	第2列	第3列	第4列	第5列	第6列	第7列	第8列	第9列	第10列	第11列	第12列	第13列	第14列	第15列	第16列	第17列
第2行		控制项	符合情况	补充措施	等级保护技术要求							等级保护管理要求				
第3行					物理	网络		主机		应用	数据	制度	机构	人员	系统建设	系统运维
第4行					措施名称1	措施名称2	措施名称3	措施名称4	措施名称5	措施名称6	措施名称7	措施名称8	措施名称9	措施名称10	措施名称11	措施名称12
第5行	访问控制	访问控制策略														
第6行		用户注册														
第7行		权限管理														
第8行		用户口令管理														
第9行		用户访问权限检查														
第10行		口令的加密														
第11行		无人值守的用户设备														
第15行		...														

下一步就是根据第 3 列中 ISO27001 的要求，一一核对第 6 列到第 17 列有哪些已经设计的措施能够满足，在第 4 列符合情况中给出一个评估，如“满足”、“部分满足”、“空缺”，然后在第 5 列补充措施中说明补充的思路。

在设计阶段做这样的对比分析，建议主要从“有、无”的角度考虑其对应，而不必急于将两套标准中的要求细节进行完全的对应。等级保护和 ISO27001 都是比较复杂的体系，不可能对应得严丝合缝，细节方面在设计阶段做的粒度稍粗一些，有利于对比的进行，保证实施的进度。而且 ISO27001 是管理要求，也不会在技术细节上描述很多。

这样对比分析出来的结果，主要是发现在控制措施项这个级别的遗漏，即图中第 4 行中是否相对于 ISO27001 有欠缺。这样的欠缺要在设计阶段补充好，否则今后项目实施发现缺一块内容，就不好再单立项申请买

设备或产品了，造成了被动。举个例子，等级保护对恶意代码防范主要在网络和主机两个层面中提出了要求，在应用层没有提出防范手段，而根据 ISO27001 的相关要求，是应当在应用层面进行恶意代码防范的，这样银行可能需要在设计中添加一台 Web 应用防火墙，作为相应的防护措施。而这台 Web 应用防火墙在运行中如何使用、配置，先不在合规设计中考虑，可以放到产品实施阶段，结合实际环境的要求去配置。

对比分析做到多深的程度，需要结合银行自己的想法来确定，但都建议以补充那些会导致后期实施变更的措施为目标，而不是在设计阶段把细节都搞清楚。否则会使设计阶段的工作量急剧增加而难以继续进行。

两套标准的融合

这样对比分析、补充完善之后，我们就得到最终的设计方案，包含了能够满足两套要求的那些安全措施。在对比分析中会发现，无论是以等级保护设计方案为基础去补充 ISO27001 的要求，还是反过来做，需要补充的措施并不多（如果是后者，需要补充较多的是等级保护技术细节，因为 ISO27001 是管理方法，没有技术细节）。

两套标准的契合程度还是比较高的，毕竟二者都是从整体架构的层面去指导机构的信息安全建设工作，其遵循的安全基本原理相同，安全措施也是通用的，不会出现“专属于 ISO27001 的控制措施”这类东西。从文字中看，在两套标准中通常都有同样的要求与措施，只是章节不一样，体现形式不同，它们要求中的大部分内容都可以在对方标准中找到映射。从主导思路看，ISO27001 相对偏重过程，推动机构不断通过 PDCA 方法去改进与提升，机构的自主度比较高；而等级保护相对偏重结果，用细致的可实施措施去做出要求，强制性地促使机构去弥补那些有缺陷的方面，机构的自主度不高。

进一步地，我们应当清楚，无论 ISO27001 还是等级保护，其本质都是帮助机构达到理想安全状态的可选方法。借助成熟标准建立自己的安全体系是我们的目标，达到这个目标之后，也不必纠结于这个体系是“27001 的”还是“等级保护”的，它就是一个安全体系，满足我们当前和未来的安全需要。能理解到这一点的机构，ISO27001 和等级保护就不再是（许多人眼里那样）高高在上的东西，而是为我服务、可为我所用的工具。

千里之行，始于足下，在建立信息安全体系的初期我们可以以合规为主要目标，在把初级阶段做扎实的基础上稳步前进。通过设计、实施、检查、改进来不断提升安全管理水平，最终都能具备全面的信息安全控制能力，从容应对安全风险的变化，进入本文开始提到的卓越运营阶段。

配合监管部门的检查

至此，我们已经大概了解到了如何建立银行自己的信息安全体系。这个信息安全体系将会接受监管部门的检查，我们的合规设计工作是否合理，将在检查中得到考验。从绿盟科技的服务经验看，应对监管部门检查也有很多注意事项，应对效果的好坏直接影响到监管评价，甚至会造成正面 / 负面监管评价之间的转变。

我们将在本文的最后一部分中，介绍监管部门的检查如何进行，以及银行如何配合这些工作。

（待续）

金融业个人信息安全事件分析与应对（下篇）

北京分公司　白雷

摘要：本文针对当前发生的个人信息泄露事件，从金融安全的角度分析个人信息保护方面的现状，提出银行类金融企业应对信息安全的策略与个人信息安全保护建设的解决方案。

关键词：个人信息　隐私保护　安全方案

下篇：银行业安全应对方案的思考

基于对 2011 年底的个人信息泄露的安全事件的全面分析，得出本次安全事件的应对，因其涉及深远和影响广泛需要有一套完整的信息安全应对方案，以下从体系化安全解决方案的思路出发试分析银行业信息安全的应对思路。

1、路线图

作为银行类金融服务提供者需要完善地解决个人信息保护的课题，应当建立基于技术与管理的完整安全体系，通过体系化的安全防护达到综合防治的效果。面向个人信息的安全路线图可以体现为如下的五个步骤的应对方式。

在此路线图中:

• 个人信息保护专项方案为专门针对本次事件可以采取的一些具有临时性的解决问题的应急方法，如：系统中添加客户警示语、采用下一代令牌、监控钓鱼网站、对用户登录的 IP 进行地域的限制以及专项黑名单机制等。

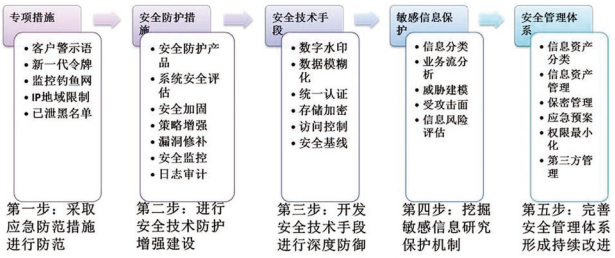


图 1：安全层面解决方案分析

• 安全技术防护方案为应对个人信息保护的课题，加强物理层、网络层、系统层、应用层的安全防护，依据分区域、按等级、多层次的防护思想进行安全规划、安全评估、安全加固与安全维护，贯彻预警、防御、监测、恢复的多重安全保护的技术策略，沿着威胁攻击路径部署安全措施。

• 安全技术增强部署方案对现有的安全技术实施改进与增强，其中需要改进的包括密码算法、加密方式、对存储的加密、U 盘管理、统一认证与审计等技术机制；适用新安全形式的安全技术手段包括

数字水印技术、数据模糊化等。

• 敏感信息保护方案首先对业务系统中的个人信息识别、分类、定级；然后通过业务分析梳理数据流程；进行威胁建模，分析受攻击面；最后通过综合风险评估，得出保护方案。

• 安全管理体系建设方案依据风险的原则分析个人信息安全管理面临的威胁与管理自身的脆弱性，通过体系化、制度化、流程化、表单化和信息化的方法完善内部管理，并与系统管理体系有机的融合，形成面向信息保护的安全管理体系。

2、个人信息保护专项措施方案

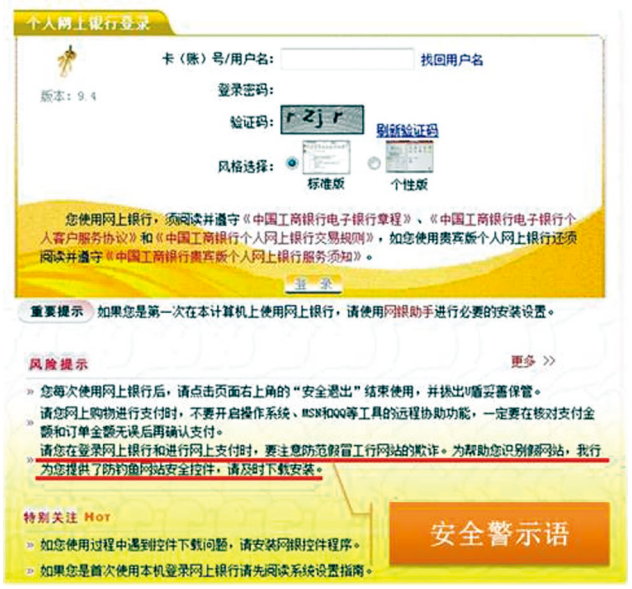


图 2：示例客户安全警示语—工行网银

基于专项的安全措施是为了应对当前安全形势的要求而设立的，如：

• 客户警示语给出重要的安全提示，提高用户安全保护意识。

• 新一代令牌：使用安全认证的动态口令系统可以大大提升安全性，特别是当前各大银行已经采用多种安全认证手段，新一代令牌的使用也已经全面推广。

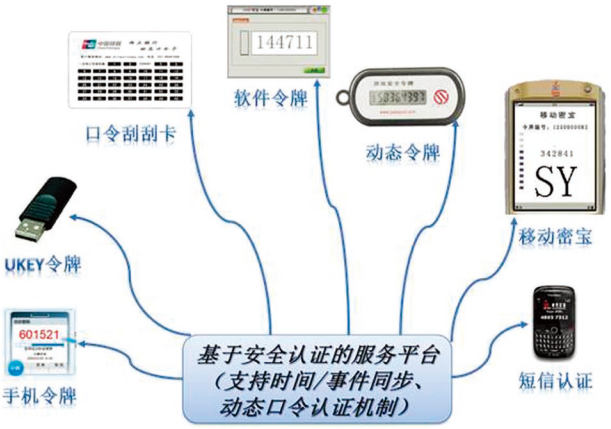


图 3：动态口令认证方式

使用新一代令牌系统，可以大大提升安全性，有效防止通讯被监听和破译。所以各大行长期以来都有对令牌系统作不断的升级。

• 监控钓鱼网：对于仿冒与伪造的钓鱼网站的及时发现是一类关键的安全防护手段，而此类防护专业的安全公司已经可以提供完善的解决方案。

也可以采用安全厂商成熟的安全监控的服务。

绿盟云安全监测中心为您的网站提供 7×24 小时人员值守，不间断网站安全持续检测及互联网的仿冒域名安全监测，帮助您随时掌

图 4：部分大型银行的令牌使用情况

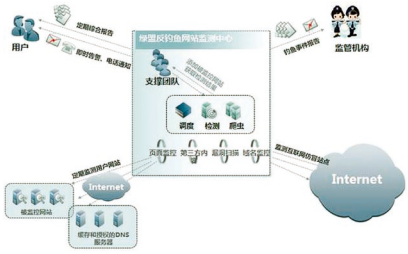


图 5：反钓鱼服务流程

控钓鱼相关风险的变化情况。发生高风险的变化或者事件时，监控服务的值守团队将会及时通知您的运维团队，确保网站所有者了解到钓鱼相关的风险变化情况，将您的损失降到最低。同时监控服务还会通过定期报告方式汇总风险变化的情况。

—绿盟反钓鱼网站监控服务白皮书

3、信息保护技术增强方案

金融业信息保护的技术体系的建立应当依据纵深多重防护的思想建立，集中沿着安全威胁的入侵路径来部署防御措施，综合采取物理层、网络层、系统层和应用层的防护手段进行安全增强。

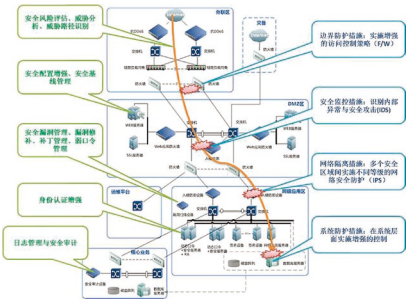


图 6：安全技术增强措施

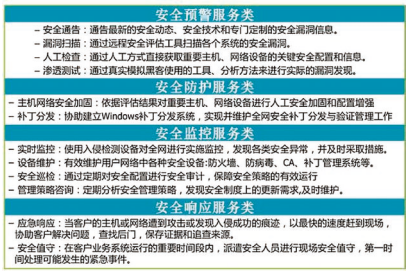


图 7：安全厂商的运维服务详解

专业化的第三方安全公司可以为金融行业提供全面的安全运维服务，此类服务可以弥补业务系统安全技术和人员经验的不足，利用长期的安全积累提供成熟的安全保障。

4、安全技术手段方案

在个人信息保护的安全技术手段中，可以选用以下的技术增强方案：

- 数字水印：数字水印技术是为信息数据特别是包含多媒体信息的数据提供隐藏的标识，这种方法在知识产权保护领域已经广泛运用，它的主要作用是作为信息识别、信息追踪、审计与认证的依据。
- 数据模糊化：数据模糊化系统部署在数据输出的接口上，通常用于隔离生产系统与测试环境，统一为测试环境提供模糊化后的测试数据。模糊化系统要求基本模糊策略可配置、传输数据可监控、传输结果可审计。模糊化的目标通常包括了客户个人敏感信息和业务数据等。
- 存储加密：传统的安全加密算法已经受到一定的挑战，当前安全方案的厂商已经提出了基于传统加密算法的一些“加盐”方式，以防止通过碰撞方式破解用户密码的行

为，此类方法可以为各类用户设计使用。

• 安全基线：基于对系统安全配置采用统一的基线标准以防止信息安全出现短板的木桶原理，安全基线解决方案由业务层面向下分解到功能层面对各类操作系统、数据库、网络设备和应用软件的安全要求，在系统实现层面细化到各类产品的安全配置。

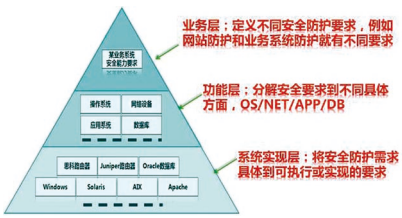


图 8：基于业务的安全基线结构

安全配置问题：通常都是由于人为的疏忽造成，主要包括了账号、口令、授权、日志、IP 通信等方面内容，反映了系统自身的安全脆弱性。安全配置方面与系统的相关性非常大，同一个配置项在不同业务环境中的安全配置要求是不一样的，如在 WEB 系统边界防火墙中需要开启 HTTP 通信，但一个特殊业务网关边界就没有这样的需求，因此在设计业务系统安全基线的时候，安全配置是一个关注的重点。

图 9：协同一致的基线安全配置规范

安全基线实施步骤分三步进行:

第一个阶段是基础阶段，通过基线安全评估，建立基础安全配置规范，结合系统安全加固工作在系统设备中实施安全配置。

图 10：安全基线实现的项目过程

第二个阶段是配置工具建立阶段，通过针对安全配置规范的专门安全核查工具开发，在系统中部署漏洞扫描设备和配置核查设备。本阶段需要建立适合自身业务要求的安全检查基线。

第三个阶段是流程建立阶段，针对系统运维中的各个环节设计安全检查流程，通过设备入网、工程验收、日常维护、安全检查等规范的流程持续的完善设备的安全配置。

• 终端安全：基于终端安全防护系统的移动介质存储保护方面，借助终端安全手段，对 U 盘等存储介质的使用进行有效管理，防止敏感信息及高密级数据通过 U 盘泄露。

图 11：基于终端安全管理的信息保护系统

此外，安全技术防护措施还应用到统一认证、存储加密和访问控制等各类技术手段和方法。

5、敏感信息保护方案

针对敏感信息保护课题，专业安全厂商已经形成了比较成熟的业务安全评估与设计方案，并且在金融系统得到了很好的应用。其安全方案的主体模块如下图所示:

其中关键的实施过程包括:

- 业务分析—依据业务影响确定个人信息的重要程度及敏感程度。
- 数据流图—从业务流程分析敏感信息的数据使用方式及数据流。
- 威胁建模—通过模型化的手段对敏感信息在业务运行中的威胁进行定位与分析。
- 受攻击面分析—依据风险的方法确定承受攻击的业务过程与活动。
- 风险评估—从信息资产、安全威胁、受攻击程度等方面作综合分析，形成风险的视图。

形成的项目成果包括:

- 敏感信息分类—识别和分级系统的敏感信息。
- 安全建议—基于风险应对的敏感信息保护解决方案建议。

图 12：敏感信息保护方案示意图

关键的研究方法包括:

- 基于业务的安全评估将对系统的评估扩展到对于业务的评估方法，分别不同的业务流程、业务逻辑、业务边界、业务管理与业务规范性进行面向敏感信息保护的安全评估。

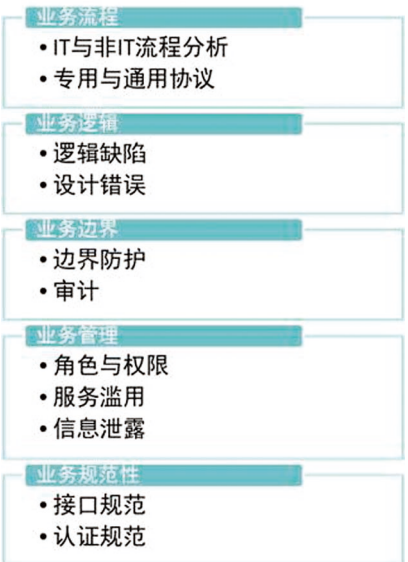


图 13：业务安全评估

6、信息保护管理建设方案

安全管理的建设方案是从人员、技术、管理角度进行制度化、流程化的保护建设。

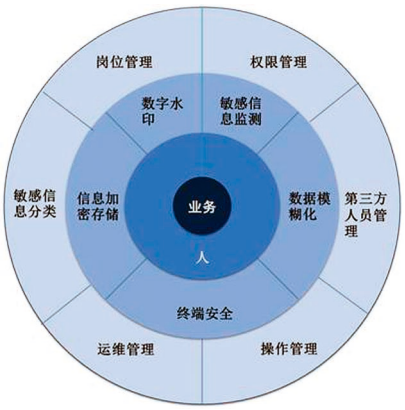


图 14：信息保护管理架构示意图

管理解决方案的建立需要依据统一的安全原则进行:

- 设计思想；
- 基于整体的信息保护安全架构原则；
- 业务为核心、人员是基础、技术为支撑、管理为保障。

管理体系中涉及信息安全保护的制度包括:

- 信息资产分类管理制度;
- 信息资产管理制度;
- 文档保密管理;
- 应急预案;
- 操作及权限最小化管理规范;
- 第三方人员管理;
- 安全运行维护管理;

……

基于对国际信息安全管理体系标准 ISO27001 的遵从，完整的信息安全管理制度体系应当覆盖 11 大类的各个控制项:

信息安全管理体系	ISO 27001
总体方针	安全策略
安全组织体系	信息安全组织
	人力资源安全
安全策略体系	资产管理
	物理和环境安全
	通信和操作管理
	访问控制
	系统采购开发与维护
	安全事件管理
	业务连续性管理
	符合性

银行系统的安全管理建设应注重金融行业的特点，更强调体系化的解决方法，各安全管理咨询厂商都不会推荐头痛医头，脚痛医脚、修修补补的建设方法，而应当建立管理体系，保持循环提升和持续改进。

7、总结与展望

对于金融企业来说，个人信息安全保护工作任重道远，应当保持一个持续改进的心态，从容应对安全问题，按部就班地加强信息安全建设，保障业务持续发展。

手机银行安全评估

技术支持中心　姚伟

摘要：网上银行因为能够降低柜面业务成本，减少客户排队时间等，备受各银行重视。随着移动互联网的飞速发展和智能手机的不断进步，手机银行作为一种新的电子渠道，相比较有更多特点，目前各大银行已经开始逐步推广手机银行业务。本文对手机银行的安全评估方法进行了详细介绍。

关键词：手机银行　安全评估　Android　iPhone

1、手机银行简介

手机银行是各商业银行推出的新一代电子银行产品，手机银行是网上银行的延伸，也是继网上银行、电话银行之后又一种方便银行用户的金融业务服务方式，有贴身“电子钱包”之称。它一方面延长了银行的服务时间，扩大了银行服务范围，另一方面无形地增加了许多银行经营业务网点，真正实现 24 小时全天候服务，大力拓展了银行的中间业务。手机银行虽然以手机为载体，但其实现方式也有很多种，常见的主要包括 SMS、STK、WAP、客户端（Android、iPhone 等）。

因为手机银行的随时随地、贴身、快捷、方便、时尚等特点，一经推出就受到了广大用户的青睐，成为了捕获时尚银行客户的一个新的亮点。随着一些新业务和新技术（如: NFC 近场支付）的广泛使用，未来手机银行还将有更加广阔的发展空间。

2、手机银行分类及特点

	优点	缺点
SMS 方式	技术实现简单，适用各种手机	复杂业务输入不便，安全级别低
STK 方式	内置银行密钥，安全级别高	需要换卡，业务扩展不方便
WAP 方式	设计简单，兼容性好	界面简单，交互性差
客户端方式	图形界面，安全级别高	需要智能手机，带宽要求高

3、手机银行相关业务

3.1 自助银行

包含账户查询、转账汇款、账单收款、信用卡还款等业务。

3.2 远程支付

包含点卡充值、机票购买、电话缴费、支付宝线下付款等业务。

3.3 近场支付

指商场 POS 消费等业务。

3.4 电子钱包

包含空中圈存、地铁公交刷卡、超市小额支付等业务。

4、手机银行评估介绍

手机银行安全评估服务是一项针对手机银行系统推出的安全评估服务。它旨在通过一系列的技术性检查、测试和分析，挖掘手机银行系统可能存在的技术和业务层面的风险，指导客户来降低和规避风险，保障系统的正常运行，进而保障银行用户的资金和信息安全。

手机银行安全评估服务主要是从操作系统、应用服务和业务等多个层面来开展评估工作，工作内容涵盖相关主机和应用的漏洞扫描、安全配置分析和安全测试等。该服务的目的在于帮助客户发现威胁到业务自身安全的缺陷或隐患，及时做出修补或整改，避免因为安全问题导致的经济损失。

4.1 操作系统安全配置

操作系统的安全性，主要通过远程漏洞扫描系统和安全配置基线检查工具进行，测试中包括以下内容:

安全补丁：检查系统是否安装了最新的安全补丁，避免存在漏洞的组件对操作系统安全产生威胁。

用户管理：检查系统中是否存在多余用户，尤其是弱口令用户，删除不必要的用户，为用户设置复杂的密码，必要时考虑设置口令策略，强制用户使用复杂密码。

服务管理：检查系统中是否存在不必要的网络服务，例如：FontPage 扩展等，是否开放了不必要的文件共享，例如：windows 默认共享 c$ 等。

文件管理：检查系统重要的配置文件、日志文件等是否严格设置了权限，防止未授权的用户修改配置。

日志审计：检查系统是否开启了日志审计功能，对系统的登录，注销等重要操作是否进行记录。

4.2 应用服务器安全配置

应用服务器的安全性，主要通过安全配置基线检查工具和手工方式进行，测试包括以下内容:

安全补丁：检查应用服务是否安装了最新的安全补丁，避免存在漏洞的组件对操作系统安全产生威胁。

应用组件：检查应用服务是否安装了不必要的组件，避免多余的组件产生安全问题，影响业务应用。检查服务版本信息是否隐藏，避免远程获得服务版本信息。

运行权限：检查应用服务运行权限是否最小化，是否使用了管理员身份运行应用，避免应用服务产生的安全问题对操作系统造成影响。

其他设置：例如目录遍历功能是否启用、是否启用 IP 访问限制、是否启用 SSL 传输

加密、是否启用日志记录等。

4.3 应用服务端安全测试

应用服务端的安全性，主要通过人工方式进行，测试中包括以下内容:

4.3.1 输入验证

检查用户提交给应用程序的数据是否经过了校验，校验规则是否完善，对非法字符是否进行了阻断。防止不安全的变量进入 SQL 语句，导致 SQL 注入等漏洞。

4.3.2 身份认证

检查应用程序中用户登录的逻辑是否合理，用户能否通过特定的方法绕过身份认证。对于用户输入的密码，是否进行了复杂度检查，防止弱口令。表单提交时，是否有图形验证码，防止暴力提交。

4.3.3 授权管理

检查应用程序对用户权限是否进行了严格划分，同级别用户间、低权限和高权限用户间能否越权访问。

4.3.4 会话管理

检查应用程序在会话管理方面是否进行了保护，例如：长时间没有用户操作时，是否会自动注销？cookie 中是否明文存放敏感信息等。

4.3.5 安全加密

检查应用程序中的敏感信息，例如：用户密码，是否以明文方式存放在数据库中，网络传输是否进行了加密。

4.3.6 错误处理

检查应用程序和用户交互过程中，返回的提示中是否会泄露敏感信息，防止错误信息中包含服务器版本信息等信息帮助攻击者获取服务器权限。

4.3.7 日志记录

检查应用程序是否对用户操作进行日志记录，至少应包含：记录时间、操作信息、操作结果等信息。

4.4 应用客户端安全测试

应用客户端的安全性，主要通过人工方式进行，测试中包括以下内容:

4.4.1 证书有效性

测试客户端程序是否严格检查服务器端证书信息，防止用户受到嗅探攻击后密码泄露，或者用户遭受钓鱼攻击。

4.4.2 密码软键盘

测试客户端程序在密码等输入框使用软键盘，防止手机被安装可疑程序后，密码被记录。

4.4.3 安全策略设置

测试客户端程序是否有密码复杂度检查功能，检查用户输入的密码，禁止用户设置弱口令，例如：禁止用户输入 123456 之类的弱口令。

4.4.4 关键字段加密

测试客户端程序提交数据给服务端时，密码字段是否进行了加密，防止恶意用户嗅探到用户数据包中的密码等敏感信息。也防止手机用户被安装木马后，木马篡改收款人等信息，盗用他人账号资金。

4.4.5 敏感信息保存

检查客户端程序是否将账号密码等敏感信息保存在日志中。

4.4.6 反编译保护

测试客户端安装程序，判断是否能反编译为源代码，是否存在代码保护措施。

SNMPV3用户安全模型中的密码学详解

核心技术部　陈庆

摘要：本文从程序实现的角度详细解释了 SNMPv3 用户安全模型 (USM) 涉及几种密码学算法。

关键词：USM　MD5　HMAC-MD5-96　CBC-DES

一、SNMPv3 用户安全模型简介

SNMPv3 用户安全模型 (USM) 是为了解决下列问题而出现的：

1. **数据完整性**。确保数据在传输过程中不被篡改。

2. **数据源有效**。确认数据来自谁，来自哪里，防止恶意的、假的 SNMP Agent 要求对合法用户进行身份验证以窃取身份验证信息。

3. **数据机密性**。确保无法旁路监听获取有效数据，交换的数据是经过加密的。

4. **数据时效性**。防止重放攻击。

USM 利用多用户的概念，要求每个用户均提供密钥进行身份验证、信息加密，最终使得 SNMPv3 可以解决如上四种典型安全问题。使用 HMAC-MD5-96、HMAC-SHA-96 进行身份验证，使用 CBC-DES、CBC-AES 进行信息加密。USM 允许将来需要时使用新的身份验证、信息加密协议。

USM 有三种可能的安全级别：

- noAuthNoPriv　无身份验证、无信息加密
- authNoPriv　有身份验证、无信息加密
- authPriv　有身份验证、有信息加密

不存在想像中的 noAuthPriv。在 SNMPv3 报文的 msgFlags 字段中指定安全级别。

Wireshark 可以较好地解析 SNMPv3 报文，后面的描述以 Wireshark 的显示为准。

数据时效性是靠 msgAuthoritativeEngineBoots、msgAuthoritativeEngineTime 两个字段共同保证的。

二、身份验证

1) 基本原理

假设 SNMP Agent 要求使用 authNoPriv 或以上安全级别。

发送方：

a：msgFlags.Authenticated 置位，msgAuthenticationParameters 域清零。

b：每个 msgUserName 有自己的 AuthPass，与 msgAuthoritativeEngineID、整个 SNMPv3 报文一起参与某种运算，结果填充到 msgAuthenticationParameters 域。这个运算主要涉及 MD5、HMAC-MD5-96 或 SHA、HMAC-SHA-96。具体使用哪种身份验证协议，是在 SNMP Agent 上配置好的。

c：发送 SNMPv3 报文。

接收方：

a：接收 SNMPv3 报文。检查 msgFl-

ags.Authenticated 置位否。

b：析取 msgAuthenticationParameters 并保存。

c：msgAuthenticationParameters 域清零。

d：SNMP Agent 根据 msgUserName 在己端的用户表里获取相应的 AuthPass，这要求 msgUserName 必须在 SNMP Agent 上存在。

e：AuthPass 与 msgAuthoritativeEngineID、整个 SNMPv3 报文一起参与某种运算，结果与保存的 msgAuthenticationParameters 进行比较。如果相等，身份验证通过。

上述描述是一个简化版，比如 msgAuthoritativeEngineID，这个也要求在 C/S 两端匹配。

SNMP Agent 检查的是单向 HASH 处理过的 C 端提供的 AuthPass，C 端提供的 AuthPass 明文并未在网络上、S 端出现，这就阻止伪造的 SNMP Agent 在钓鱼过程中获取 C 端提供的 AuthPass 明文。

2)password_to_key_md5()

生成 msgAuthenticationParameters 的 Python 代码如下：

```
#
# msg 中的 msgAuthenticationParameters 必须清零
#
def Get_msgAuthenticationParameters ( AuthPass, EngineID, msg ) :
    AuthSecret  = password_to_key_md5( AuthPass, EngineID )
    return( GetHMACMD5( AuthSecret, msg )[0:12] )
#
# end of Get_msgAuthenticationParameters
#
```

第三形参 msg 即整个 SNMP 报文，也就是 UDP 负载部分。整个计算过程分两大部分，首先由 AuthPass、EngineID 计算得到 AuthSecret；然后将 AuthSecret 作为 key，针对 msg 计算 HMAC-MD5，取计算结果的前 12 字节，这就是所谓的 HMAC-MD5-96。这里我们以 MD5 为例，还可以用 SHA。

RFC 2574 在 "A.2.1. Password to Key Sample Code for MD5" 中给出了一段完整可用的 password_to_key_md5() 实现代码：

```
void password_to_key_md5
(
    unsigned char  *password,
    unsigned int    passwordlen,
    unsigned char  *engineID,
    unsigned int    engineLength,
    /*
     * out 型形参，指向一个 16 字节 buf，用以保存 16 字节的 MD5 值
     */
    unsigned char *key
)
{
    MD5_CTX         MD;
    unsigned char  *cp,
                    password_buf[64];
    unsigned int    password_index  = 0;
    unsigned int    count           = 0,
                    i;
    MD5Init( &MD );
    while ( count < 1048576 )
    {
```

```
    cp     = password_buf;
    for ( i = 0; i < 64; i++ )
    {
                    *cp++    = password[
password_index++ % passwordlen ];
    }  /* end of for */
      MD5Update( &MD, password_buf,
64 );
    count  += 64;
  }  /* end of while */
  MD5Final( key, &MD );
  memcpy( password_buf, key, 16 );
  memcpy( password_buf + 16, engineID,
engineLength );
    memcpy( password_buf + 16 +
engineLength, key, 16 );
  MD5Init( &MD );
  MD5Update( &MD, password_buf, 32 +
engineLength );
  MD5Final( key, &MD );
  return;
}  /* end of password_to_key_md5 */
```

简单解释一下这段 C 代码。将 password 重复 N 次，得到一个长 1048576 字节的字符串。随 password 长度不同，N 也将不同，但最终的字符串必须是 0x100000(即 1MB) 字节。针对该字符串计算 MD5，结果记作 digest1。 按 digest1、engineID、digest1 的顺序拼接出一个 buf，针对该 buf 计算 MD5，结果就是 password_to_key_md5() 的输出，也就是 AuthSecret。

下面是最终结果等价的 Python 实现：

```
#
# in  : maplesyrup 000000000000000000
000002
# out : 526F5EED9FCCE26F8964C2930
787D82B
#
def password_to_key_md5 ( Password,
EngineID ) :
  tmp = GetMD5( ( Password * ( 1048576
/ len( Password ) + 1 ) )[0:1048576] )
  key = GetMD5( tmp + EngineID + tmp )
  return( key )
#
# end of password_to_key_md5
#
```

三、信息加密

1) 基本原理

假设 SNMP Agent 要求使用 authPriv 安全级别。

发送方：

a：构造 SNMPv3 报文，msgFlags 中 Authenticated、Encrypted 同时置位。

b：将 1 字节的 msgAuthoritativeEngineBoots 扩展成 4 字节，与另一个 4 字节随机数拼接，形成 8 字节的 DES Salt，填充到 msgPrivacyParameters 域。

c：每个 msgUserName 有自己的 PrivPass，这个与 AuthPass 是两回事。PrivPass 与 msgAuthoritativeEngineID、msgPrivacyParameters、明文的 msgData 一起参与某种运算，最终得到加密过的密文 msgData。这个运算主要涉及 CBC-DES 或 CBC-AES，具体使用哪种信息加密协议，是在 SNMP Agent 上配置好的。

d: 如上一小节所述，进行身份验证处理，填充 msgAuthenticationParameters 域。

e：发送 SNMPv3 报文。

接收方：

a：接收 SNMPv3 报文。先对该报文进行身份验证，防篡改。

b：检查 msgFlags.Encrypted 置位否。

c：从接收到的 SNMPv3 报文中析取 msgPrivacyParameters。

d：SNMP Agent 根据 msgUserName 在己端的用户表里获取相应的 PrivPass，这要求 msgUserName 必须在 SNMP Agent 上存在。

e：PrivPass、msgAuthoritativeEngineID、msgPrivacyParameters、密文的 msgData 一起参与某种运算，最终得到解密过的明文 msgData。

msgData 在 RFC 2574 里被称作 scopedPDU。

2) 代码实现

这里我们以 DES 为例，还可以用 AES。

```
def BlockDES ( key, sth, iv, mode ) :
  #
  # 这里用的是 MODE_CBC，不是 MODE_ECB
  #
  obj = DES.new( key, DES.MODE_CBC, iv )
  if 0 == mode :
    #
    # 假设 sth 不是 8 字节的整数倍，我们自己处理填充。
    #
    i   = 8 - len( sth ) % 8
    sth = sth + chr( i ) * i
    ret = obj.encrypt( sth )
  else :
    ret = obj.decrypt( sth )
  return( ret )
#
# end of BlockDES
#
def Get_scopedPDU ( PrivPass, EngineID, salt, pdu, mode ) :
   PrivSecret  = password_to_key_md5( PrivPass, EngineID )
  DESKey     = PrivSecret[0:8]
  PreIV      = PrivSecret[8:]
   IV          = ''.join( [ chr( ord( salt[i] ) ^ ord( PreIV[i] ) ) for i in range( 8 ) ] )
    return( BlockDES( DESKey, pdu, IV, mode ) )
#
# end of Get_scopedPDU
#
```

将 PrivPass 传递给 password_to_key_md5() 得到 16 字节的 PrivSecret，取前 8 字节作为 DESKey，后 8 字节作为 PreIV。DES Salt 与 PreIV 进行异或，得到 IV。利用 DESKey、IV 对 scopedPDU 进行 CBC-DES 加密、解密。

参考文献

[1]User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
http://www.ietf.org/rfc/rfc2574.txt

[2]Test Cases for HMAC-MD5 and HMAC-SHA-1
http://www.ietf.org/rfc/rfc2202.txt

[3]Block cipher modes of operation
http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation

基于污染点传播的PHP源代码审计技术

上海分公司　马传雷

摘要：本文通过对 PHP 内核和应用漏洞的研究，提出了一种基于 Opcode 和污染点传播进行代码安全性自动化审计的方法。

关键词：PHP　Opcode　污染点传播　危险函数

一、前言

1.1 为什么选择 PHP 进行研究

PHP 是一种开源的脚本语言，它应用广泛，尤其适合于 web 开发。具有跨平台，容易学习，功能强大等特点。图 1 是 2012 年初 w3techs 对全球排名前 100 万的网站服务端使用的脚本语言进行的统计，超过 70% 的网站有 PHP 的应用。常见的大型门户网站和社区包括 baidu、yahoo、sina、163、sohu、facebook 等都采用了 PHP 开发。国内草根站长广泛使用的开源 web 应用系统（包括 bbs、blog、wiki、cms 等等）大多数都是使用 PHP 开发的，如 Discuz、phpwind、dedecms、boblog、hdwiki 等等。

由于 PHP 语法相对比较灵活，在方便

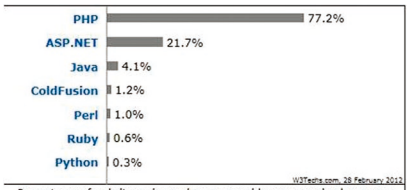


图 1

Year	PHP-related	Total	Ratio
2000-2009	11469	38230	30.0%
2010	1263	4640	27.2%
2009	1912	5733	29.9%
2008	1962	5634	34.8%
2007	2346	6517	36.0%
2006	2840	6603	43.0%
2005	1396	4928	28.3%
2004	490	2450	20.0%
2003	183	1515	12.0%
2002	240	2156	11.1%
2001	80	1677	4.7%
2000	20	1017	1.9%

图 2

程序员开发的同时也带来了很多安全漏洞。在一些漏洞披露平台（如 exploit-db.com 等），几乎每天都有知名 PHP 应用的漏洞被公开。图 2 是对 CVE 收录的漏洞进行的一个简单分析，从 2000 年到 2010 年这 10 年间，有 30% 的安全漏洞是和 PHP 相关的。正因为如此，很多应用程序的官方都成立了安全部门，或者雇佣安全人员进行代码审计，也出现了一些自动化商业化的代码审计工具如 codescan 和 fortify 等。

1.2 PHP 应用程序有哪些常见漏洞

PHP 语法的灵活性导致了其应用程序漏洞的多样性，和 java、asp 以及 .net 等脚本语言相比，PHP 应用程序的漏洞更富有“趣味性”。早期的一些安全研究人员（国外的 Stefan Esser 、Rgod，国内的 Saiy 等）发现的很多 PHP 应用程序的漏洞，都闪耀着智慧的光芒，程序员们在学习的同时也发自内心的赞叹。

图 3 是我们对 PHP 应用程序的漏洞做的一个简单的分类：

图 3

1.3 如何进行 PHP 应用程序的安全审计

WEB 应用程序漏洞查找基本上是围绕两个元素展开：变量与函数。国内的民间安全组织 80vul[1] 曾经有一个 Project：php application source code audits advanced technology[2] 做了非常好的总结，它提供了一个比较全面的危险函数字典，这里要向该 Project 的发起人 Superhei 和 Ryat 致敬。一个 PHP 漏洞的利用必须能够把提交的恶意代码通过变量经过 n 次变量转换传递，最终传递给目标函数执行。所以我们对 PHP 应用程序进行静态审计的时候主要有两种思路：一是从追踪变量的传递过程入手，查找是否进入危险函数中；二是从查找容易导致安全漏洞的危险函数入手，反向追踪其参数来源。

本文描述的污染传播技术是一种自动化的 PHP 源代码审计技术，它的原理是从 opcode 层面对 PHP 应用程序进行变量追踪和危险函数定位，进而发现可利用的漏洞。

二、PHP 内核介绍

2.1 PHP 脚本执行过程

PHP 内核分为两个部分，Zend 引擎和 PHP CORE。PHP CORE 负责一些外围处理，比如与 SAPI 沟通等。Zend 引擎是 PHP 实现的核心，提供了语言实现上的基础设施，例如：PHP 的语法实现，脚本的编译运行环境，扩展机制以及内存管理等，PHP 扩展大都使用 Zend API 实现。

Zend 虚拟机的执行引擎是一个非常简单的实现，它只是依据中间代码序列

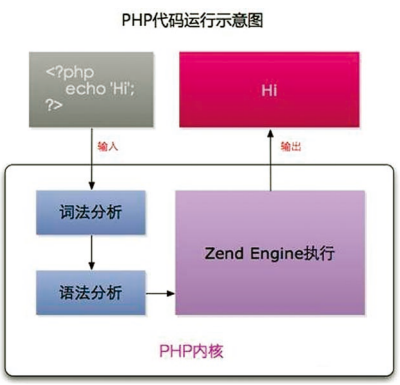


图 4

（EX(opline)），一步一步调用对应的方法执行。在执行引擎中并没有类似于 PC 寄存器一样的变量存放下一条指令，当 Zend 虚拟机执行到某条指令时，当它所有的任务都执行完了，这条指令会自己调用下一条指令，即将序列的指针向前移动一个位置，从而执行下一条指令，并且在最后执行 return 语句，如此反复。这在本质上是一个函数嵌套调用。

2.2 词法分析和语法分析

PHP 脚本的执行，一般经过如下几个过程：

1.Lexing，将 PHP 代码转换为语言片段 (Tokens[3])；

2.Parsing，将 Tokens 转换成简单而有意义的表达式；

3.Compilation，将表达式编译成 Opocde；

4.Execution，顺次执行 Opcode，每次一条，从而实现 PHP 脚本的功能。

PHP 早期使用的词法解析器使用的是 flex，后来改为使用 re2c[4]。源码目录下的 Zend/zend_language_scanner.l 是 re2c 的规则文件。语法分析使用 Yacc(Yet

Another Compiler-Compiler)，在源码目录下的 zend/zend_language_parser.y。

我们通过如下的 demo 代码来分析整个过程:

----------code----------

```
<?php
//debug.php
$fly = "hi,nsfocus";
echo $fly;
?>
```

PHP 自身提供一个函数 token_get_all 可以将 PHP 代码 Scanning 成 Tokens:

----------code----------

```
<?php
$a = file_get_contents("debug.php");
$tokens = token_get_all($a);
//print_r($tokens);
foreach($tokens as $token) {
if(count($token) == 3) {
echo token_name($token[0])."\t";
echo "$token[1]\r\n" ;
}
```

运行以上代码，将 debug.php 处理后得到对应的 Token 列表如下:

----------code----------

```
T_OPEN_TAG      <?php
T_COMMENT       //debug.php
T_VARIABLE      $fly
T_WHITESPACE
T_WHITESPACE
T_CONSTANT_ENCAPSED_STRING
"hi,nsfocus"
T_WHITESPACE
T_ECHO  echo
T_WHITESPACE
T_VARIABLE      $fly
T_WHITESPACE
T_CLOSE_TAG                 ?>
```

接下来，就是 Parsing 和 Compilation 阶段了，首先会丢弃 Tokens Array 中的多余的空格，然后将剩余的 Tokens 转换成一个一个的简单的表达式，然后把 Tokens 编译成一个个 op_array。仍然使用 debug.php 中的 Token T_ECHO 为例，我们看看 Zend VM 是如何处理的:

----------code----------

```
//zend/zend_language_parser.y
91 %token T_ECHO
……
243         |       T_ECHO echo_expr_list ';'
……
   563 echo_expr_list:
   564                  echo_expr_list ',' expr
{ zend_do_echo(&$3 TSRMLS_CC); }
   565     |            expr        {
zend_do_echo(&$1 TSRMLS_CC); }
   566 ;
```

从以上代码可以看出，最终通过 zend_do_echo 函数来完成对 Token T_ECHO 的处理。该函数实现代码如下:

----------code----------

```
//Zend/zend_compile.c
    533 void zend_do_echo(const znode
*arg TSRMLS_DC) /* {{{ */
```

```
534 {
535     zend_op *opline = get_next_op
(CG(active_op_array) TSRMLS_CC);
536
537     opline->opcode = ZEND_ECHO;
538     opline->op1 = *arg;
539     SET_UNUSED(opline->op2);
540 }
541 /* }}} */
```

通过 zend_do_echo 处理后，生成了一个 opcode，其结构如下:

----------code----------

```
struct _zend_op {
    opcode_handler_t handler; // 执行该opcode 时调用的处理函数
    znode result;
    znode op1;// 操作数 1
    znode op2;// 操作数 2
    ulong extended_value;
    uint lineno;
    zend_uchar opcode;  // 操作指令
};
```

Zend VM 支持的 Opcode 指令大约有 140 条，可以参考官方文档[5]。这里需要注意的操作数的类型，有五种类型：IS_CONST, IS_TMP_VAR, IS_VAR, IS_UNUSED 和 IS_CV。后面我们需要关注的是 IS_VAR 和 IS_CV。IS_VAR 这种就是我们一般意义上的变量了，他们以 $ 开头表示。IS_CV 表示 ZE2.1/PHP5.1 以后的编译器使用的一种 cache 机制，这种变量保存着被它引用的变量的地址，当一个变量第一次被引用的时候，就会被 CV 起来，CV 变量以! 开头表示。

Zend VM 逐行翻译完成后，会将结果存放在 op_array 中，其内部存储的结构如下:

----------code----------

```
struct _zend_op_array {
    /* Common elements */
    zend_uchar type;
     char *function_name;  // 如果是用户定义的函数，则这里将保存函数的名字
    zend_class_entry *scope;
    zend_uint fn_flags;
    union _zend_function *prototype;
    zend_uint num_args;
    zend_uint required_num_args;
    zend_arg_info *arg_info;
    zend_bool pass_rest_by_reference;
    unsigned char return_reference;
    /* END of common elements */
    zend_bool done_pass_two;
    zend_uint *refcount;
    zend_op *opcodes;  // opcode 数组
    zend_uint last，size;
    zend_compiled_variable *vars;
    int last_var，size_var;
    // ...
}
```

我们通过 PHP 官方提供的扩展 parsekit[6]可以得到 debug.php 翻译完成后得到的 op_array，处理代码如下:

----------code----------

```
<?php
$op_codes = parsekit_compile_file("debug.php", $errors, PARSEKIT_SIMPLE);
print_r($op_codes);
```

```
//print_r($errors);
?>
```

得到结果:

```
Array
(
    [0] => ZEND_ASSIGN T(0) T(0) 'hi,nsfocus'
    [1] => ZEND_ECHO UNUSED T(0) UNUSED
    [2] => ZEND_RETURN UNUSED 1 UNUSED
    [function_table] =>
    [class_table] =>
)
```

翻译完成后进入 Execution 阶段，将 op_array 交由 zend_execute() 逐条执行。zend_execute() 有 CALL/GOTO/SWITCH 三种处理方式，可在编译 PHP 的时候通过修改参数 with-zend-vm=CALL/GOTO/SWITCH 的方式进行修改，由于篇幅关系，这里不再贴代码。

2.3 PHP 代码加密和混淆

有些公司出于商业目的会对 PHP 应用的代码进行加密和混淆，比较著名的混淆软件有 Zend Guard 等。原理如图 5 所示，早期某些软件的做法只是在词法分析和语法分析阶段进行处理，Compilation 出来的依然是正常 Opcode。这种做法比较容易被破解，目前基本上都已经使用了重写 execute 函数的方式了。

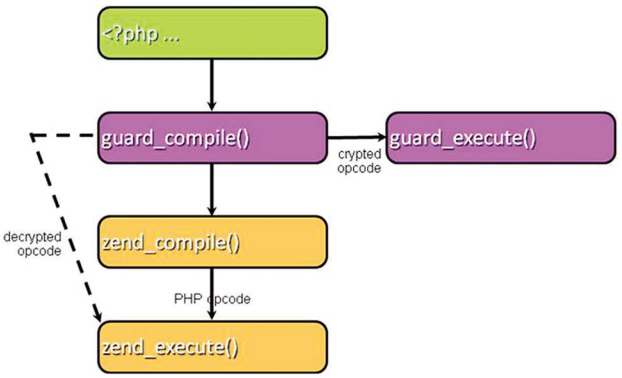


图 5

三、自动审计的技术方法

3.1 污染点传播分析

数据流的污染点传播（Taint Propagation）分析是静态分析中常用的技术，其原理如图 6:

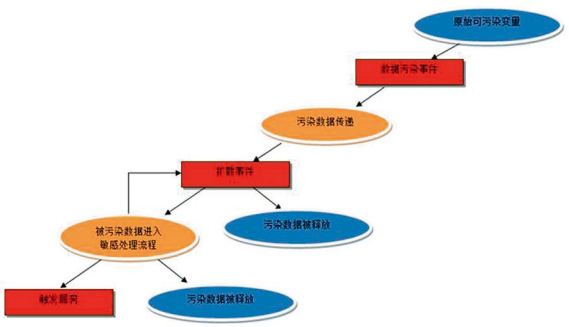


图 6

对于 PHP 应用而言，如果要从脚本代码本身来追踪污染点是比较困难的，需要考虑各种语法和 html 代码的干扰，但是如果从 Opcode 层面来分析则比较容易。使用 PHP 官方提供的 parsekit 或者 vld 扩展可以很容易将 PHP 脚本翻译为 Opcode 并绘制出程序执行路径，在追踪过程中需要关注的对象有:

- 污染的来源：$_GET、$_POST、$_COOKIE 以及 $SERVER 等用户可控制的输入
- 用于传播变量的指令：ASSIGN*、FETCH*

3.2 敏感函数

PHP 的函数主要有两种，一种是系统函数（ZEND_INTERNAL_FUNCTION），一种是用户自定义函数（ZEND_USER_FUNCTION）。函数调用的操作指令如下:

ZEND_USER_FUNCTION：DO_FCALL_BY_NAME

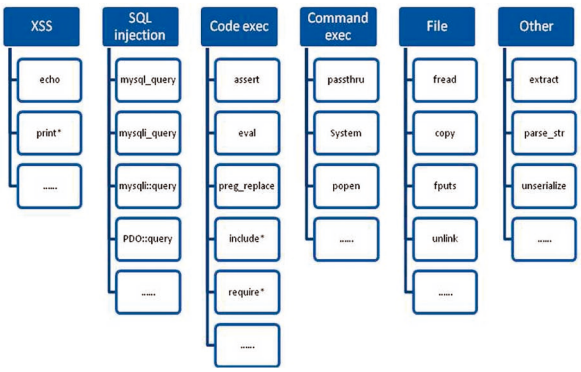


图 7

ZEND_INTERNAL_FUNCTION：DO_FCALL

在关注函数调用的时候，我们主要关注三类敏感函数的调用:危险函数、安全函数和干扰函数。

危险函数指能造成安全漏洞的函数，按照造成的漏洞类型分类，主要有以下一些（需要扩展）:

在污染点传播的路径上，如果追踪到有危险函数存在，则可能造成安全漏洞。这里所以说是可能，是因为 PHP 自身有很多安全函数（也可能是开发者自己实现的安全函数）可以对污染数据进行净化。危险函数和安全函数是有一定的对应关系的，比如 addslashes() 一般来说只能是 SQL 注入相关函数的危险，不能净化代码执行漏洞等其他类型危险函数的污染数据。这里列举部分如图 8:

图 8

另外一方面，有些函数可能会起到干扰污染数据的作用，比如原来不能造成危害的污染点通过这些干扰函数的处理可能会造成一些安全漏洞，反之亦有可能。这类函数和危险函数、安全函数的关系比较复杂，需要结合具体情况进行分析，这里列举几种如图 9:

图 9

3.3 漏洞挖掘示例

基于以上思路，我们通过如下一个简单的存在漏洞的文件作为实例进行分析:

----------code----------

```
//test.php
<?php
include('debug.php');
$a = @$_GET[a];
if(empty($a))   exit('get no data');
$b = $a;
atestu($b);
eval($b);
function atestu1($v){
$v = addslashes($v);
@assert($v);
}
class test{
function nsfocus($v){
echo "hello,nsfocus";
}
}
```

将代码翻译为 Opcode，得到程序路径图如图 10:

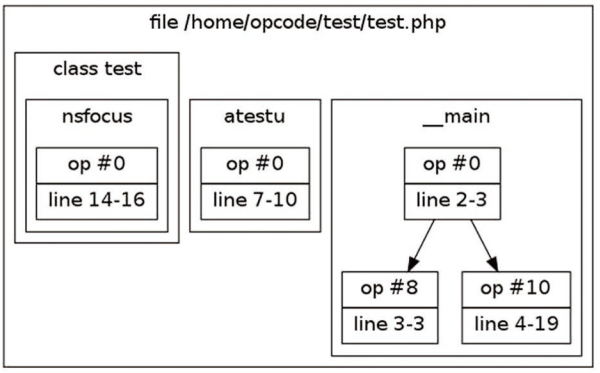


图 10

翻译得到主程序的 Opcode 结果如图 11（没有包括函数表和类表):

```
filename:       /home/opcode/test/test.php
function name:  (null)
number of ops:  17
compiled vars:  !0 = $a, !1 = $b
line     # *  op                           fetch          ext  return  operands
---------------------------------------------------------------------------------
   2     0  >   BEGIN_SILENCE                                      ~0
         1      FETCH_CONSTANT                                     ~2      'a'
         2      FETCH_R                      global                $1      'GET'
         3      FETCH_DIM_R                                        $3      $1, ~2
         4      END_SILENCE                                        ~0
         5      ASSIGN                                             !0, $3
   3     6      ZEND_ISSET_ISEMPTY_VAR                        6    ~5      !0
         7    > JMPZ                                              ~5, ->10
         8  > > EXIT                                              'get+no+data'
         9*     JMP                                                ->10
   4    10  >   ASSIGN                                             !1, !0
   5    11      INIT_FCALL_BY_NAME                                 'atestu', 'atestu'
        12      SEND_VAR                                           !1
        13      DO_FCALL_BY_NAME                              1
   7    14      NOP
  12    15      NOP
  19    16    > RETURN                                             1
```

图 11

寻找 ASSIGN*、FETCH* 相关的操作指令，我们得出的污染传递路径如下:

$_GET->$1->$3->!0->!1

最终 !1 进入函数 atestu，我们通过分析生成的函数表继续追踪:

```
Function atestu:
filename:       /home/opcode/test/test.php
function name:  atestu
number of ops:  10
compiled vars:  !0 = $v
line     # *  op                           fetch          ext  return  operands
-------------------------------------------------------------------------------
   7     0  >   RECV                                                   1
   8     1      SEND_VAR                                               !0
         2      DO_FCALL                                      1  $0      'addslashes'
         3      ASSIGN                                                 !0, $0
   9     4      BEGIN_SILENCE                                          ~2
         5      FETCH_R                       local                    $3      'v'
         6      SEND_VAR                                               $3
         7      DO_FCALL                                      1        'assert'
         8      END_SILENCE                                            ~2
  10     9    > RETURN                                                 null
```

图 12

我们发现最终污染数据处理流程是 : !1->addslashes()->assert()。assert() 是 php code execute 的危险函数，addslashes() 是安全函数，但是净化的是 SQL Injection 的污染数据，不能净化 php code execute 的污染数据。执行到这里，就发生了代码执行漏洞。

四、总结

本文主要描述了一种基于 Opcode 进行污染点传播分析和漏洞自动挖掘的方法。这种方法同样也适用于 Java 和 .net 等其他语言的自动审计，只是得到字节码的方式有所不同。业界比较著名的代码审计软件 Fortify 等基本上也都采用类似的技术实现。利用这种技术代码原理，也可以对加密混淆过的 PHP 代码进行审计，Stefan Esser 大牛已经开发出了相关的商业工具。目前业界也有人通过编写 PHP 模块的方式将这个思路应用于 PHP 应用的动态审计并取得了不错的效果 (参考 Taint-3.0) 。

就 PHP 而言，这种技术已经是目前市面上已知的比较优秀的代码自动审计方法了，但是依然有些问题不能解决。这里列举几个目前我觉得比较困扰的问题:

- 对于二次输入的追踪：在初始的变量追踪过程中，污染数据写入数据库或者文件的情况后就比较难以继续追踪了。现实中很多漏洞都是由于来自数据、配置文件等二次输入造成;
- 复杂的框架：对于比较复杂的框架，如通过“动态路由”等方式进行文件调用的应用难以深入的自动分析;
- 业务逻辑相关的漏洞很难自动审计。

注：① http://www.80vul.com/

② http://code.google.com/p/pasc2at/wiki/SimplifiedChinese

③ http://cn2.php.net/manual/zh/tokens.php

④ http://re2c.org/

⑤ http://www.php.net/manual/en/internals2.opcodes.php

⑥也可以使用 vld 扩展

参考文献

PHP (www.php.net)

风雪之隅 (www.laruence.com)

TIPI 项目 (www.php-internal.com)

Stefan Esser (www.suspekt.org)

Fortify (www.fortify.com)

Nosec (nosec.org)

80vul (www.80vul.com)

SDL过程在WEB应用中的实践

上海分公司　朱伟元

摘要：本文通过自己对 SDL 过程的理解，分析了 SDL 过程中各个阶段的控制手段，针对控制手段提出了理论的实施方法，并且结合了本人对渗透测试和源代码审计的经验，提出了在某些阶段的一些新控制手段的理论和实施方法。

关键词：SDL　数据流　软件安全　软件生命周期

引言

越来越多的 Web 开发团队在其开发的产品不同程度上引入了 SDL 过程，但多数开发团队因为人力成本、时间成本等因素，造成了引入的控制措施不完整、不彻底等问题，这些使产品依然存在较多的安全问题。本文结合自己对 SDL 的理解，分析了其在 Web 应用中一些阶段安全措施的实施方法，并且提出了新的实施理论和方法。

一、SDL 过程简介

- 规划

定义安全需求，一般典型包括了有关行业监管机构的规定、企业定义的有关国际相关标准、企业定义的有关 BUG 分级与处理标准。

- 需求分析

根据前期的业务规划和安全规划，引入业务安全的控制机制，例如审计、异常处理等。

- 功能设计

确定软件的功能规划安全，例如业务处理流程，页面要素等。

- 程序编码

按照功能设计文档，进行程序编码，功能实现。

- 软件测试

验证软件安全性是否符合功能需求规定以及安全需求规定，对程序编码过程中的错误进行纠正。

- 交付运维

持续纠错，改进。

二、开发团队引入 SDL 过程面临的问题

- 压缩项目周期

应项目需求方的要求，严格压缩项目周期，使团队在 SDL 某些进度里草率完成，或者干脆跳过不执行。

- 开发团队意识不足

由于项目开发人员基本全是有关的开发人员，对安全研究不够深入，并且没有引入安全顾问跟踪整个开发过程，导致不论是在安全需求、安全设计还是安全开发上暴露较多的安全问题，而开发团队有关人员可能并没有意识到。

- 缺少安全设计文档

应用软件功能设计人员在进行功能设计时，多数是根据互联网上典型的功能设计方法来制定软件功能，而这些典型的功能设计方法存在安全隐患。

- 缺少较完善安全开发规范文档

一般的开发团队已有一些开发规范，主要是自身开发经验的积累，对于典型漏洞的规避方法，但相对存在完善性、有效性的问题。

三、SDL 过程分析

3.1 安全规划

3.1.1 定义安全标准需求

不同的行业 B/S 应用均有自身的安全需求特点，国家、行业监管机构、上级机构要求以及出于公司自身对资产保护定制的一些要求，从应用系统网络架构、有关管理制度、人员配备方面提出了不同的要求，开发团队在安全需求阶段需要将这些因素考虑进去，制定安全需求目标，安全需求的内容大致可分为：操作审计、异常处理机制、授权与身份验证、加密等。从应用程序开发角度讲，需要将这些有关规定的应用软件设计以及数据保密性等方面筛选出来，作为预期的安全需求目标。例如针对网上银行应用中，银监会的 19 号文中规定了应用系统网络架构、软件安全等内容，由于本阶段只是定制安全需求，所以可以仅考虑软件安全部分内容。

3.1.2 定义 BUG 分级处理策略

本阶段工作是将 BUG 问题按风险等级分级，以便在后期的开发和测试过程中可以依据此内容定义跟踪和修补策略。和定义安全需求部分一样，不同的行业也有不同的风险接受标准，在此部分工作中，可以根据行业特点来定义 BUG 分级。有些安全要求严格的应用系统，只允许接受相对较低的风险等级，例如 B2C 应用相对电子银行应用，能接受的风险等级就要高一些。一般的风险等级可以按给应用系统、用户直接造成的损失程度来定义。

3.2 安全需求

3.2.1 数据流的威胁建模

- 分解数据流

根据为应用系统设计的业务需求文档，稍加分析便可得到数据流，将数据流按以下图样划分:

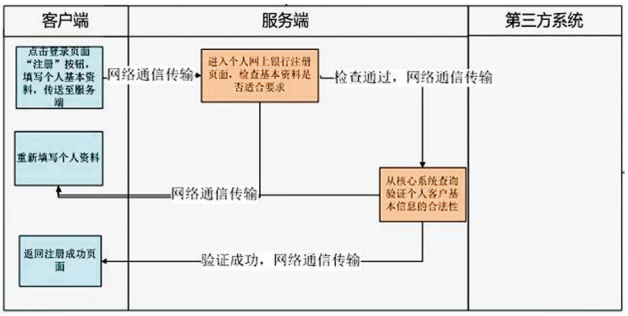


图 1：数据流

如图 1 所示，将业务流程图稍加变换分析，得到数据流，而数据流的元素分为客户端 (IE/Firefox、手机客户端等)、服务端 (中间件、数据库、核心业务系统)、第三方系统 (不受己方完全控制的信息处理系统)、通信过程四部分，每种元素对应的威胁也不相同，表 1 是四种元素对应的威胁种类:

表 1：威胁模型

元素 / 威胁	身份假冒	篡改	行为抵赖	信息泄露	拒绝服务	权限提升
客户端			√	√		
服务端	√			√	√	√
第三方系统		√	√	√	√	
通信过程	√	√		√		

按以上模型，就可以分析出一个数据流从开始到结束的流程所经过的元素，进而能分析可能的威胁类型，接下来需要做的是将威胁类型细化，以下是各类威胁的应对措施的类别:

身份假冒：强身份认证，敏感数据存储强加密，敏感数据传输加密；

篡改：数字签名，使用可提供消息完整性的协议保护通信过程；

行为抵赖：操作审计，数字签名；

信息泄露：访问授权，敏感数据处理过程强加密，机密数据存储强加密；

拒绝服务：硬件和网络资源合理调配，验证用户输入；

权限提升：最低特权原则。

应对措施的类别是向开发团队指明了消减威胁的方向，具体的到下阶段的安全设计如何依据威胁模型分析，还要依据以下这种文档：

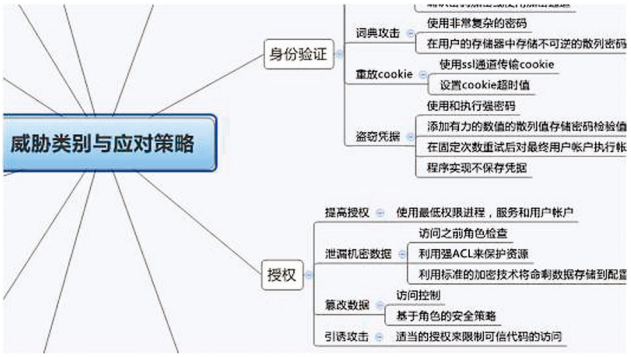


图 2：威胁类型与应对

- 设计 - 定义不安全的功能

定义不安全的功能实质上是列出设计不合理的黑名单，不同的应用，所面临的主要威胁也不尽相同，一般的开发团队，定义不安全的功能存在几个难点：

业界缺少相关的标准：基本上除了电子银行、基金证券在线交易系统外，没有其他的监管机构对软件的安全提出可度量的要求，即便是电子银行、基金证券在线交易系统，有关的规定中也仅提出了一些有限的安全目标，但是我们依然可以从这些规定中的安全目标分析出部分安全设计的目标来；

遗漏的内容可能会很多：不安全的设计方法在软件中出现的数量是与软件功能成正比的，功能越多，存在的脆弱性也越多，需要根据软件的功能和应用特点来定制不安全的功能；

需要丰富的经验：如果在本阶段的工作中，有丰富安全测试经验的人员参与，效果会更佳；也可以求助第三方信息安全公司做咨询。

四、安全设计

4.1 数据流的安全分析

数据流的安全分析是依据数据流程从发起到结束的这个期间进行安全分析，数据流是依据业务流来得到，数据流经过多个控制环节后，风险必定逐步降低，以下是分析示意图：

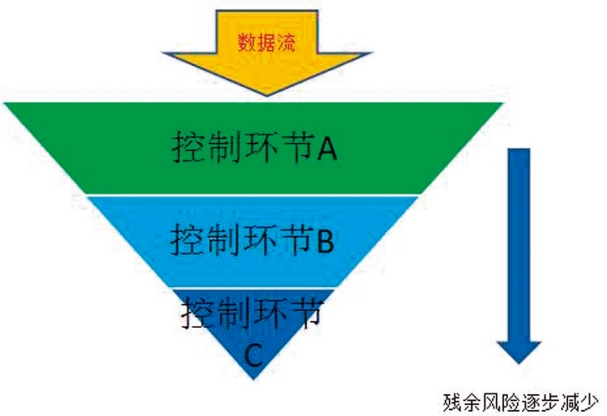


图 3：风险控制示意图

在本阶段工作中的步骤一般为:

1、收集运行场景：运行场景包含了一些安全控制手段，一般包括应用平台 (IIS/WebLogic)、操作系统的安全配置情况，甚至包括网络的基本架构和网络设备的安全控制措施。

2、分解数据流：将应用系统以模块为单位分解，然后又以业务功能来分解模块，例如典型的“注册流程”、“同行转账流程”、“查询账户余额流程”。

3、收集控制方法：典型的控制方法在各个阶段都可能存在，例如客户端存在防键盘截取功能，在传输时防窃听、篡改功能，在服务端防越权、拒绝服务等，但还需要收集的是实现这些控制方法的技术手段是什么，以此来分析其控制手段的有效性。

4、分析得到的运行场景和现有的控制方法，结合运行场景信息，得到残余风险。

5、定义风险处理方法，一般有接受风险、提醒用户、更改运行场景有关配置、使用安全设备、修正代码几项处理方法。

此阶段工作核心在于分析数据流，如何分析数据流？数据流在应用系统的一般有“数据发起”、“数据传输”、“数据验证”、“数据处理”、“数据返回”五个阶段，第一个阶段都存在一些风险，分析数据流的目的在于分析这五个阶段每个阶段的风险控制方法，这五个阶段对应的风险和一般控制方法为:

数据发起—截取 (键盘记录、屏幕录像);

数据传输—窃听、请求伪造;

数据验证—有非常多的攻击方法，一般的处理方式需要对数据进行长度、类型、正负数、大小、格式 (浮点数)、特殊字符的验证;

数据处理——一般的攻击方式为插入命令、越权访问 (修改会话标识、修改请求关键字);

数据返回—跨站脚本、应用程序运行错误信息。

在本阶段工作中尤其要将分析结果文档化，在后期的源代码审计和安全测试工作中将依据分析结果进行验证测试。

五、程序编码

5.1 定义安全开发规范

目前基本所有的开发团队都有安全开发规定了，定义了函数不安全的使用方法，不安全的功能实现方法，黑名单函数等，范围可大可小，可以参考的内容非常多，例如 Fortify 漏洞库、Owasp 漏洞库，但实际编写开发规范时，只需根据应用特点、开发团队自身的积累，然后再从以上两个漏洞库中抽取常出现的基本就能满足要求。

六、软件测试

6.1 源代码审计与安全测试

在前期的工作中形成了以下文档:

《业务安全需求文档》

《安全设计文档》

《安全编码规范》

《数据流安全分析结果文档》

在源代码审计和安全测试中，如果依据前期形成的文档的话，那么本阶段的工作将非常有针对性。

一般的源代码审计方法，如果是纯手工审计，则是查找针对比较危险的函数调用，工具自动化的审计方法所涉及的内容就会多一些，例如死循环、空指针、函数滥用等。不论使用常规手工审计方法还是工具自动化审计，其弱点都是审计软件安全本身，并没

有涉及到业务；更进一步的源代码审计在于根据代码读懂业务，在SDL 过程中，如果依据前期安全需求、安全设计等前置安全文档来做源代码审计，则可以将源代码审计工作看作是一个验证的过程，检查代码的实现是否符合前置的安全预期。

或许在这里，不能再将安全测试工作称之为渗透测试了，渗透测试是以攻击者的对应用系统发起攻击，查找系统的脆弱性并确认是否可被利用，如果更进一步，渗透测试可能会加入一些有关行业标准的测试依据，检验系统中针对行业标准实施的结果。但在本阶段工作中，同源代码审计工作一样，依然主要是根据前期的前置安全需求来做测试，检查代码实现和系统配置工作是否符合前置的安全预期，与源代码审计工作的本质区别是安全测试工作中包含对系统环境安全配置的测试，而源代码审计中不包含此项。

至此阶段，针对源代码审计生成《源代码审计报告》，针对安全测试生成《安全测试报告》，这两份报告都披露了应用软件的脆弱性，在下一阶段的修复工作中，可以依据第一阶段制定的 BUG 修复策略进行针对性的修复。

七、数据流安全分析示例

以下假定有一个网银找回密码的数据需要进行数据流的安全分析，图 4 所示是基本的数据流程图:

数据发起端在最左边，全部标记为①，数据验证与处理端全在最右边，全部标记为③和④，中间基本全是数据传输过程，有向数据端的数据传输，有向数据发起端的数据传输，分别标记为②和⑤。然后分析这些数据序列存在的控制手段，这些控制手段可能针对以

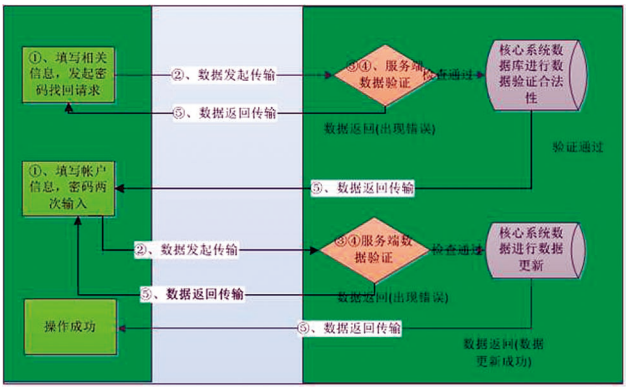


图 4：数据流程图

下风险进行控制:

①：数据发起：截取，一般的控制方法为：防击键记录，防截屏;

②：数据传输：窃听、请求伪造，一般的控制方法为：输出编码、通信加密、数据签名、使用可进行消息完整性验证的协议;

③④：数据验证和处理：拒绝服务，越权访问，一般的控制方法为：③验证数据长度、类型、正负数、大小、格式 (浮点数)、特殊字符，④特殊命令、针对决定权限参数的控制方法;

⑤：窃听、伪造，一般的控制方法为：输出编码、通信加密、数据签名、使用可进行消息完整性验证的协议。

再接下来进行控制方法差异分析，主要是了解控制方法有没有针对以上的风险进行安全控制，如何实现的。

参考文献

Microsoft SDL_Version 5.0

《软件安全开发生命周期》

WEB-MAIL跨站漏洞检测技术研究

武汉分公司　殷水军

摘要：本文在对国内外 Web-mail 跨站脚本攻击和防御技术进行了研究后，提出了一款基于 Web-mail 的跨站漏洞检测系统方案，根据系统的设计要求和模块说明，实现了基于 Web-mail 的跨站漏洞检测系统，最后对该系统进行了测试，测试结果表明了系统的可行性和实用性。

关键词：Web-mail　跨站漏洞　VECTOR 脚本　Html

一、引言

Web-mail 是指通过浏览器的 URL，以 Web 的方式来对电子邮件进行收发的服务或技术，不需要邮件客户端，用户只要可以上网，就可以使用 Web-mail，在一定程度上方便了用户对邮件的收发，它与一般网站的区别在于，其与用户交互数据更为复杂。由于 Web-mail 系统与用户交互时，其内容可以是文本格式或者 Html 格式，使得普通用户在面对恶意的 Html 邮件时，都没有很好的防御措施，利用现有的 Web-mail 过滤系统虽然能够过滤掉 Html 邮件中的一些恶意脚本代码，但是，想要完全过滤 Html 邮件里的恶意脚本，对于 Web-mail 过滤系统来说，几乎是不可能去实现的，因为攻击者总是能够利用浏览器或 Web-mail 系统过滤机制的漏洞，对目标用户 Web-mail 实施攻击 [1]。因此，Web-mail 系统除了增强过滤恶意代码之外，所能做的就是发现一个漏洞就及时的补一个漏洞，针对 Web-mail 跨站漏洞的检测则是对 Web-mail 安全威胁最好的防御方法之一。

国内外在针对 Web 应用程序安全问题的测试方面做了大量的工作，出现了一些成熟的检测工具 [2]，如 IBM 公司的 X-Force 系统、Appscan、Acunetix Web Vulnerability Scanner、NBSI 等等，这些工具都是针对 Web 应用程序的漏洞扫描和缺陷来设计的，并不具备模拟用户与 Web-mail 系统数据交互的

功能，都不适用 Web-mail 的安全检测，专门针对 Web-mail 的安全工具很少。

而在针对 Web-mail 跨站脚本攻击的安全防御措施方面，国内外主要注重于 Web-mail 过滤系统的研究，在检测方面主要以人工手动检测为基准，这样的检测方法有很大的局限性，而且注重个人的技术水平，并不全面。针对 Web-mail 跨站漏洞的检测，并没有集成化、系统化、工具化。

二、Web-mail 跨站漏洞检测相关技术研究

为了抵御 Web-mail 跨站脚本攻击，传统的 Web-mail 跨站脚本漏洞检测都是通过手工实现。首先收集 JavaScript 跨站检测脚本，把跨站检测用脚本以邮件内容的形式发送到 Web-mail 检测系统，然后再检测脚本是否执行。

手工检测依赖于检测员的经验和技巧，经验丰富的检测员能够很好的编写测试用例 [3]，通过反复的检测，发现 Web-mail 过滤系统的缺陷。但是手工检测漏洞，存在很大的局限性。首先收集的检测跨站用例代码不全面，手动检测覆盖面小；其次检测效率低，人为的去搜索和判断代码过滤问题，工作量过于庞大，况且人为手工检测效率低，很多都是无用功。

三、Web-mail 跨站漏洞检测系统的设计与实现

本文通过对 Web-mail 跨站脚本攻击和漏洞检测技术的研究，设计了针对 Web-mail 跨站漏洞的检测系统。该系统可以将一些 VECTOR 脚本重构成庞大的、用于检测的跨站脚本代码，而这里所说的 VECTOR 脚本，实际上是将一些普通完整的跨站脚本代码经过拆分、归类（例如：开始 VECTOR 脚本、闭合 VECTOR 脚本、JS 脚本）后的检测脚本，通过模拟 Web 邮件发送功能，用不同类型的 VECTOR 脚本重构完整的检测脚本，并以 Html 邮件内容的形式发送给 Web-mail 系统，通过基于浏览器内核的源码获取程序获取邮件响应、过滤后的 Html 源代码，通过比对响应前后的 Html 源代码，来分析 Web-mail 过滤系统存在的缺陷等。最后数据入库，生成报告。系统的整体功能如图 1 所示:

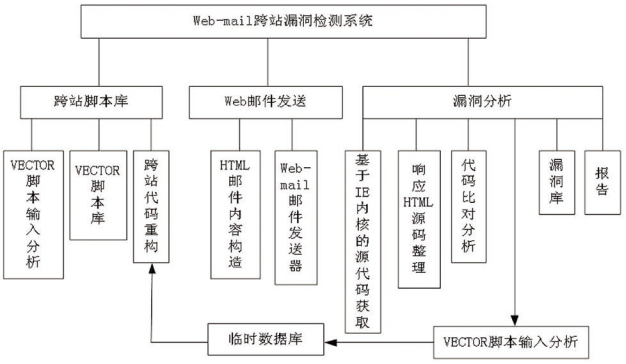


图 1：系统整体功能图

系统包括三个部分：跨站脚本库、Web 邮件发送和漏洞分析。跨站脚本库是为了收集跨站向量脚本，使检测系统能够拥有更强大的检测用例代码；Web 邮件发送是检测系统的承接部分，是模拟了网页邮件发送系统；漏洞分析是把过滤的数据进行分析，存入临时数据，进行下一次的检测，同时，也是比对分析发现跨站漏洞的部分。

系统的界面设计分为两部分，Web 客户端界面和漏洞分析客户端

界面。Web 客户端界面主要是包括跨站脚本库的界面和邮件发送界面。平时在积累脚本的时候，只要打开 Web 客户端界面，在 VECTOR 脚本输入分析界面处输入将要分析的脚本，系统自动根据分析结果存入VECTOR 脚本库中相应的数据表中。在检测网页邮件跨站漏洞时，只要打开发送邮件的界面，导入包含 JavaScript 跨站代码的 Html 代码，然后发送邮件。启动漏洞分析客户端界面，提取过滤后的 Html 代码，系统自动进行分析，数据入库，生成报告，其中 Web 客户端是用 PHP 语言编写，漏洞分析客户端是用 C++ 语言编写。

3.1 跨站脚本库模块

Web-mail 跨站漏洞检测系统中的跨站脚本库模块模拟类似于 Web 应用程序的 Fuzzing 工具。跨站脚本库模块整体结构如图 2 所示:

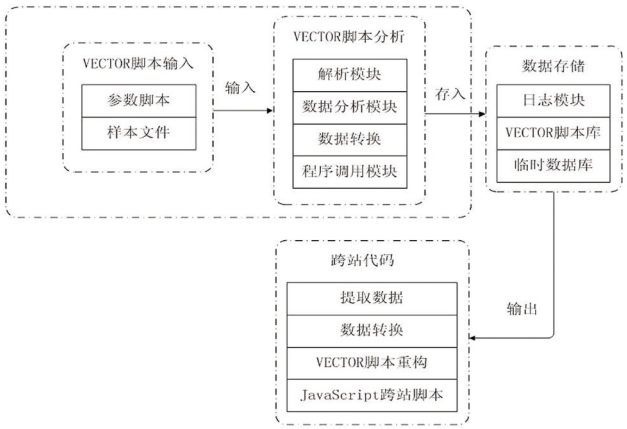


图 2：跨站脚本库模块图

如图 2 所示，该部分分为四个大模块：VECTOR 脚本输入、VECTOR 脚本分析、数据存储和跨站脚本重构。

VECTOR 脚本输入主要是为了输入参数脚本，同时还有一些样本文件，样本文件包括一些常见的 JavaScript 跨站脚本。VECTOR 脚本分析把前面输入的参数脚本进行解析，接着把数据进行向量分析，最后调用数据存储。数据存储分为日志模块、VECTOR 脚本库和临时数据库。其中两个数据库是有区别的，主要是为了考虑系统的动态循环。当系统首次进行检测时，启动的是 VECTOR 脚本库，当是循环检测时，直接读取临时数据库中的向量脚本，临时库中的向量脚本都是系统在第一次检测时，提取的没被过滤的脚本。日志模块主要是记录检测中的数据，以及脚本分析中存入的数据。比如：数据库更新日期，数据库资料说明等。最后的跨站代码模块主要是 JavaScript 跨站脚本代码的重构，首先从数据库中提取向量脚本，然后把数据进行一些必要的转换，最后形成检测用的脚本。

3.2 Web 邮件发送模块

为了检测 Web-mail 跨站漏洞，模拟真实的邮件发送系统是必不可少的。Web 邮件发送模块 [4] 如图 3 所示:

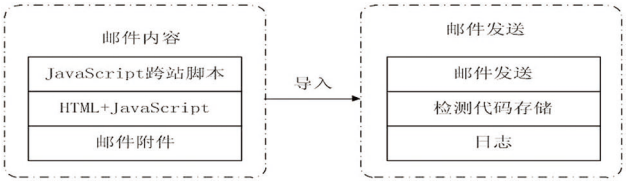


图 3：Web 邮件发送模块图

由图 3 可见，Web 邮件发送分为两部分：邮件内容和邮件发送。邮件内容部分用于构造包含检测脚本的 Html 源代码内容。邮件发送主要是发送网页邮件。邮件内容包含三个部分，其中构造含测试脚本的邮件内容是主体，另外附带有添加邮件附件的功能；邮件发送包括三个部分，其中邮件发送是主体，而检测代码存储则是把 JavaScript 跨站脚本代码按规定格式进行保存，方便后面的对比分析，日志则是用来记录邮件发送的一些基本信息。

3.3 漏洞分析模块

前面所做的所有工作都类似于对邮件系统进行渗透测试。渗透测试 [5] 名为测试，实际是模仿黑客的手段对需要测试的站点或者机构进行模拟攻击，在攻击过程中发现网络的薄弱点，然后利用各种入侵工具进入内部网络，薄弱点一般指系统的漏洞或者网络的脆弱点。本文设计

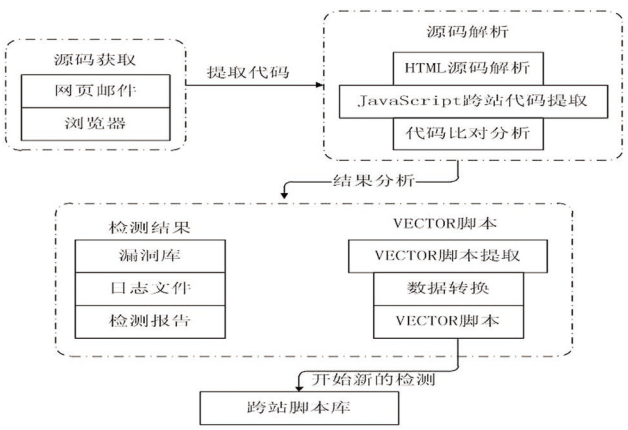


图 4：漏洞分析模块图

的 Web-mail 跨站漏洞检测系统只是发现漏洞，并没有对系统漏洞进行利用。因此它可以说是渗透测试的前半部分工作。而在渗透测试的前半部分工作中，最重要的就是漏洞的发现，漏洞的发现源于对测试结果的分析。而在 Web-mail 跨站漏洞检测系统中，有一个完整的漏洞分析体系。Web-mail 跨站漏洞检测系统的漏洞分析模块如图 4 所示。

由图 4 可见，模块分成三个部分。其中源码获取部分是用来提取 Html 源代码的。源码解析部分是用来解析 Html 代码。结果分析部分是对最后的检测结果进行数据分析的过程。

源码获取部分其实指的就是基于浏览器内核的源码获取程序。由于本文选择的浏览器是基于 IE 内核的浏览器，因此此处的源码获取程序相当于就是 IE 浏览器的一个插件程序 [6]。当检测员浏览 Web-mail 后，浏览器会在本地保存响应后的 Html 源码，基于 IE 内核浏览器的程序就是获取本地响应完成后的 Html 源代码。

源码解析部分是漏洞分析中的重点，它要把 Html 源码进行正确的解析，提取属于测试的 JavaScript 跨站代码，并且在提取了之后，进行规格要求的整理，便于对比分析。

结果分析部分可以分为两个部分，一部分是检测结果的存储，检测的最终结果是发现漏洞，如果有完整的 JavaScript 跨站脚本代码存在，则说明存在漏洞，然后就对该漏洞进行数据库存储，并记录日志文件，生成检测数据和漏洞报告。另一个是 VECTOR 脚本部分，这部分是把有用的、没过滤掉的 JavaScript 代码进行提取、转换，重新输入跨站脚本库，进行新一轮的分析检测。

3.4 整个系统的数据流转

如图 1 所示，整个系统分为三个部分：跨站脚本库模块、Web-mail 邮件发送模块和漏洞分析模块。三个部分是紧密联系在一起的，只有当跨站脚本库中的 VECTOR 脚本有足够的庞大，重构成的 JavaScript 跨站脚本各式各样，才是比较好的检测用例库。系统是检测 Web-mail 跨站漏洞，那么发送邮件则是必不可少的一个环节，怎样去模拟真实的发送邮件系统，怎样去实现带有 JavaScript 跨站测试脚本的 Html 邮件内容? 都是邮件发送系统来实现的。当发送了检测的邮件，经过邮件系统的过滤响应后，就可以进行漏洞分析了。漏洞分析既是分析是否有漏洞存在，也是提取没有过滤掉的 VECTOR 脚本。整个系统的数据流转，如图 5 所示。

由图 5 可见，首先是系统的开始，然后有脚本分析，把分析的脚本存入跨站脚本代码库。然后判断是否开启 Web-mail 检测系统，如果否，则结束。如果是去 Web-mail 检测系统，那么接下来编写测试邮件，然后发送邮件，经过浏览器内核程序提取响应源码，对源代码进行脚本分析，是否有漏洞，如果

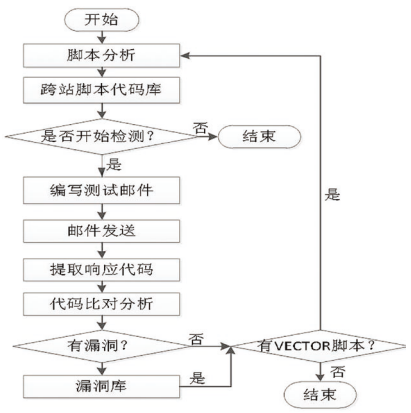


图 5：整个系统的流程图

有，则存入漏洞库，如果没，则检测是否有 VECTOR 脚本，如果有，则跳到脚本分析，否则检测结束。

四、Web-mail 跨站漏洞检测系统的测试

在完成了整个 Web-mail 跨站漏洞检测系统的设计和实现之后，还需要对它进行功能和性能上的测试，以检验它是否符合系统设计的预期要求。

4.1 测试流程

测试的流程图如图 6 所示：

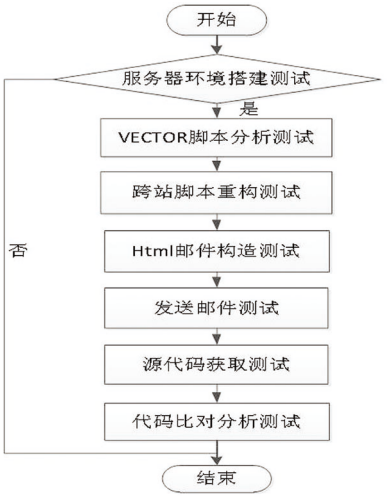


图 6：系统测试流程图

4.2 测试结果

根据系统测试流程给出了测试结果，整个测试部分所做的工作及结果如表 1 所示：

表 1：测试工作表

测试内容	测试过程	测试结果
服务器环境搭建测试	在浏览器的 URL 处输入测试链接（http://x.x.x.x/admin.php），测试是否能够访问服务器的 admin.php 页面。	√

VECTOR 脚本分析测试	点击 Web 客户端上的“VECTOR 脚本分析”按钮，根据页面上的操作说明，输入待分析的 VECTOR 脚本，测试是否能够正确的拆分匹配待分析脚本，并将结果数据入库。	√
跨站脚本重构测试	点击 Web 客户端上的“重构 JavaScript 脚本”按钮，根据页面说明进行操作，测试是否能够重构跨站脚本，使测试脚本用例增强。	√
Html 邮件构造测试	点击 Web 客户端上的“发送邮件”按钮，然后点击导入测试邮件内容，测试是否能够按照规则，把重构的 JavaScript 代码嵌入 Html 文件中。	√
发送邮件测试	点击 Web 客户端上的“发送邮件”按钮，根据说明进行邮件发送，测试是否能够把构造好的 Html 文件作为邮件内容，发送 Web 邮件。	√
源代码获取测试	启动 IE 浏览器和基于 IE 内核的源代码获取程序，登录测试的 Web-mail，浏览测试 Html 邮件，测试是否能够获取 Html 邮件的源代码。	√
代码比对分析测试	启动代码比对分析器，选择测试前的 JavaScript 代码文件和经过 Web-mail 系统过滤后整理的 JavaScript 代码文件，点击对比，测试是否能够进行代码的比对分析，检测出 XSS 漏洞，并将结果数据入库。	√

由测试结果可见，Web-mail 跨站漏洞检测系统的各个模块都很好的完成设计要求，并且整个系统能够检测出 Web-mail 潜在的跨站漏洞。

五、结束语

本文设计的 Web-mail 跨站漏洞检测系统虽然能够很好的检测出 Web-mail 潜在的跨站漏洞，但是部分模块的实现还有不足之处。首先，在漏洞分析模块部分，由于源代码获取程序是基于 IE 内核的，因此必须使用 IE 浏览器来访问 Web 邮件，这给漏洞分析部分的代码平台移植造成了很大的局限性。其次，跨站脚本库模块和邮件发送模块虽然能够很好的自动化，但是在访问检测用的 Web 邮件时，必须人工的去通过浏览器访问邮件，这给整个系统的自动化带来了不便。

结合上面分析的不足，完善 Web-mail 跨站漏洞检测系统的下一步工作应从以下几个方面着手：第一、可以编写一个简单的浏览器，用来解决基于 IE 内核程序的局限性；第二、可以分析现今流行的 Web-mail 系统的邮件格式，然后编写针对各个 Web-mail 系统的 Html 邮件源代码提取程序。

参考文献

[1] Jeremiah Grossman, Robert Hansen, Seth Fogie 等 . Cross Site Scripting Attacks [M]. Syngress, 2007.

[2] 张玉清，戴祖锋，谢崇斌 . 安全扫描技术 [M]. 北京：清华大学出版社，2004.

[3] ha.chers XSS[EB/OL]. http://ha.ckers.org/xss.html, 2011-04.

[4] PHP send mail[EB/OL]. http://php.net/manual/en/function.mail.php, 2011-04.

[5] 蒲石 . Web 安全渗透测试研究 [D]. 西安：西安电子科技大学，2010.

[6] AJAX 网页抓取工具 [EB/OL].http://www.blogjava.net/senior/archive/2009/05/10/269916.html, 2011-04.

绿盟科技参加 Interop 2012 美国展会

近期，绿盟科技参加在美国拉斯维加斯召开的国际网络通信展会 Interop 2012。在为期 5 天的会议上，绿盟科技展示了抗拒绝服务产品（Anti-DDoS）及解决方案，获得与会者的关注。

Interop 是全球最大的网络通信展会之一，具有 26 年历史，主要聚焦于云计算、网络、无线和移动、虚拟化、数据中心、存储、统一通信、信息安全和风险管理、IT 管理等专业技术领域。2012 年的 Interop 内容包括安全、移动及云计算，展会吸引了全球 350 多家参展商的产品和解决方案。此次展会上，绿盟科技就一些持续性的安全问题，与来自世界各地的安全专家及意见领袖进行了深入探讨，交换了彼此在安全行业长期积累的经验和观点。

如今 DDoS 攻击变得更有针对性、更具有破坏性以及更加密集。“Stop DDoS Attacks Before They Stop You”，绿盟科技安全专家在 Interop 众多来宾交流时表达了这样的观点，同时绿盟科技还向业界展示了抗拒绝服务产品、解决方案及相关服务。

绿盟科技始终致力于防 DDoS 攻击，尤

其是应用层攻击防御及研究，历经 10 年不断创新，为各行业大客户提供经过验证的企业级网络安全解决方案，保障客户业务顺畅运行。

思考下一代安全　运营商行业技术沙龙成功举办

日前，由中国通信企业协会通信网络安全专业委员会主办，绿盟科技协办的主题为“思考下一代安全”技术沙龙在福州举办，绿盟科技副总裁吴云坤主持，福建省通信管理局、工信部电信研究院、国家计算机网络应急技术处理协调中心（以下简称 CNCERT）、中国移动、中国电信、中国联通三大运营商专家及领导出席参与了本次沙龙。

会上，CNCERT 的黄元飞博士、电信研究院安全中心的高级工程师卜哲等专家从近几年工信部安全大检查的角度，介绍了目前运营商行业主要面临的安全风险；移动研究院安全所杨光华所长与大家分享了移动 Big Data 推进安全工具革新的思路和目前的进展情况，他特别提到移动的 Big Cloud，Big Cloud 为安全的大数据分析提供了基础平台，它的并行数据挖掘系统为安全的大数据量分析提供了挖掘与分析的基础工具，结合大云框架，快速实施安全的大数据量检测、分析与相应的闭环处理过程；电信广研院的金华敏部长从电信运营商对于下一代安全的整体规划着眼，从提高抗打击能力、具备自适应、自愈能力以及加强高效协同能力几个方面提出了新一代安全有待解决的几个问题，着重介绍了电信网络 SAFES 的安全体系；中国联通马广宇处长也简单介绍了联通在这方面的一些工作开展情况。讨论中，三位运营商行业代表还从实践角度谈及自己在实际运维中遇到的困难，对于未来云方面的一些安全准备等。

绿盟科技产品推广部总监万慧星从安全厂商的视角分析下一代安全的特性，提出厂商与用户连接、厂商与伙伴连接、云与端的协作、设备与设备的协作以及人与设备的协作，建立安全运营优势的新模式。根据绿盟科技已经开展的下一代安全的探索工作，针对运营商行业给出了相应的建议。

NSFOCUS 2012年2月之十大安全漏洞

声明：本十大安全漏洞由 NSFOCUS(绿盟科技) 安全小组 <security@nsfocus.com> 根据安全漏洞的严重程度、利用难易程度、影响范围等因素综合评出，仅供参考。http://www.nsfocus.net/index.php?act=sec_bug&do=top_ten

1. 2012-02-15 Microsoft Silverlight & .NET Framework 堆破坏远程代码执行漏洞 (MS12-016)

NSFOCUS ID: 18739

http://www.nsfocus.net/vulndb/18739

综述：

Microsoft Silverlight 是跨浏览器、跨平台的 .NET 实现，用于为 Web 构建媒体体验和交互应用。

Microsoft Silverlight 和 Microsoft .NET Framework 在实现上存在远程代码执行漏洞，成功利用后可允许攻击者执行任意代码。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意网页，从而控制受害者系统。

2. 2012-02-16 Adobe Flash Player 远程内存破坏漏洞（CVE-2012-0752）

NSFOCUS ID: 18747

http://www.nsfocus.net/vulndb/18747

综述：

Adobe Flash Player 是一个集成的多媒体播放器。

Adobe Flash Player 在实现上存在远程内存破坏漏洞，攻击者可利用此漏洞执行任意代码。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意网页，从而控制受害者系统。

3. 2012-02-24 Oracle Java True Type 字体 IDEF Opcode 解析远程代码执行漏洞

NSFOCUS ID: 18820

http://www.nsfocus.net/vulndb/18820

综述：

Java 是 Sun 公司推出的一种应用程序开发语言。

Java 在特制 True Type 字体的实现上存在任意代码执行漏洞，通过提交恶意网页或文件，Java 在处理 MaxInstructionSize 大小时可造成堆缓冲区溢出。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意网页，从而控制受害者系统。

4. 2012-02-22 Symantec pcAnywhere 12.5.0 build 463 及更早版本远程拒绝服务漏洞

NSFOCUS ID: 18812

http://www.nsfocus.net/vulndb/18812

综述：

Symantec pcAnywhere 是全球最畅销的用于管理服务器和提供管理人员支持的远程控制解决方案。

pcAnywhere 12.5.0 build 463 及更早版本在实现上存在安全漏洞，未验证的攻击者可利用此漏洞使 ashost32 服务崩溃。

危害：

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

5. 2012-02-23 ABB WebWare Server “RobNetScanHost.exe” 缓冲区溢出漏洞

NSFOCUS ID: 18809

http://www.nsfocus.net/vulndb/18809

综述：

WebWare Server 是一种软件产品，主要用于生产数据控制。

WebWare Server 的 RobNetScanHost.exe 在解析 5512 端口上的网络报文时存在漏洞，解析带有操作码 0xE 及 0xA 的 Netscan 报文时可导致缓冲区溢出。

危害：

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

6. 2012-02-16 多个 Cisco Nexus Devices IP 栈远程拒绝服务漏洞

NSFOCUS ID: 18780

http://www.nsfocus.net/vulndb/18780

综述：

Cisco Nexus 系列由全面的交换机产品组成，使客户能够逐步、经济高效地迁移到万兆以太网和统一数据中心阵列。

多个 Cisco Nexus 产品操作系统的 IP 栈处理畸形 IP 报文并获取报文所需的 Layer 4(UDP 或 TCP) 信息时，在实现上存在拒绝服务漏洞。

危害：

攻击者可以利用此漏洞向服务器发送恶意请求，造成拒绝服务攻击。

7. 2012-02-06 PHP "htmlspecialchars()" 缓冲区溢出漏洞

NSFOCUS ID: 18647

http://www.nsfocus.net/vulndb/18647

综述:

PHP 是一种在电脑上运行的脚本语言，主要用途在于处理动态网页，包含了命令行运行接口或者产生图形用户界面程序。

PHP 在带有 $double=false 的 htmlspecialchars() 函数的实现上存在缓冲区溢出漏洞。

危害:

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

8. 2012-02-09 Open Handset Alliance Android 多个远程安全漏洞

NSFOCUS ID: 18681

http://www.nsfocus.net/vulndb/18681

综述:

Android 是 Google 通过 Open Handset Alliance 发起的项目，用于为移动设备提供完整的软件集，包括操作系统、中间件等。

Android 在实现上存在多个安全漏洞，包括绕过同源保护、获取敏感信息、执行任意脚本代码、窃取 Cookie 验证凭证等。

危害:

攻击者可以利用此漏洞诱使受害者打开恶意网页，从而控制受害者系统。

9. 2012-02-17 libpng "png_decompress_chunk()" 整数溢出漏洞

NSFOCUS ID: 18790

http://www.nsfocus.net/vulndb/18790

综述:

libpng 是多种应用程序所使用的解析 PNG 图形格式的函数库。

libpng 在 png_decompress_chunk() 函数的实现上存在远程整数溢出漏洞。

危害:

攻击者可以利用此漏洞诱使受害者打开恶意 png 图片，从而控制受害者系统。

10. 2012-02-03 Siemens SIMATIC WinCC HMI Web Server 多个输入验证漏洞

NSFOCUS ID: 18633

http://www.nsfocus.net/vulndb/18633

综述:

WinCC flexible 是用在一些机器或流程应用中的人机接口。

Siemens SIMATIC WinCC 在实现上存在HTTP 标头注入漏洞、目录遍历漏洞和任意内存读取漏洞。

危害:

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

NSFOCUS 2012年3月之十大安全漏洞

声明：本十大安全漏洞由 NSFOCUS(绿盟科技) 安全小组 <security@nsfocus.com> 根据安全漏洞的严重程度、利用难易程度、影响范围等因素综合评出，仅供参考。http://www.nsfocus.net/index.php?act=sec_bug&do=top_ten

1. 2012-03-14 Microsoft 远程桌面协议 RDP 远程代码执行漏洞 (MS12-020)

NSFOCUS ID: 19054

http://www.nsfocus.net/vulndb/19054

综述：

RDP 是一个多通道的协议，可以让用户连上提供微软终端机服务的服务器端。

Windows 在处理某些对象时存在错误，可通过特制的 RDP 报文访问未初始化的或已经删除的对象，导致任意代码执行。

危害：

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

2. 2012-03-09 Google Chrome 多个不明细节远程代码执行漏洞

NSFOCUS ID: 18995

http://www.nsfocus.net/vulndb/18995

综述：

Google Chrome 是谷歌的开源浏览器。

Google Chrome 存在多个安全漏洞，攻击者可利用这些漏洞执行任意脚本代码。这些漏洞在 2012 年 Pwn2Own 会上被利用攻击成功。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意网页，从而控制受害者系统。

3. 2012-03-29 Adobe Flash Player 多个内存破坏漏洞

NSFOCUS ID: 19265

http://www.nsfocus.net/vulndb/19265

综述：

Adobe Flash Player 是一个集成的多媒体播放器。

Adobe Flash Player 在 URL 安全域检查和 NetStream 类的实现上存在内存破坏漏洞，攻击者可利用此漏洞破坏内存，造成任意代码执行。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意 swf 文件，从而控制受害者系统。

4. 2012-03-22 Cyberoam UTM 'host' 参数远程命令执行漏洞

NSFOCUS ID: 19209

http://www.nsfocus.net/vulndb/19209

综述：

Cyberoam Unified Threat Management 可为家庭办公和远程分支办公提供的网络安全诊断操作。

Cyberoam Unified Threat Management 在处理 host 参数时未在服务器端验证，恶意用户可通过在线代理工具注入 OS 命令，利用此漏洞执行任意命令并完全控制设备。

危害：

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

5. 2012-03-09 Apple iPhone/iPad/iPod touch iOS 5.1 之前版本多个漏洞

NSFOCUS ID: 18994

http://www.nsfocus.net/vulndb/18994

综述：

Apple iOS 是运行在苹果 iPhone 和 iPod touch 设备上的最新的操作系统。

Apple iPhone、iPod touch 及 iPad 所使用的 iOS 在实现上存在多个安全漏洞。这些漏洞影响下列组件：CFNetwork、HFS, Kernel、Passcode Lock Safari、Siri、VPN 。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意信息，从而使受影响设备崩溃、绕过安全限制、获取敏感信息或执行任意代码。

6. 2012-03-29 Cisco IOS NAT 功能 SIP 远程拒绝服务漏洞

NSFOCUS ID: 19257

http://www.nsfocus.net/vulndb/19257

综述：

Cisco 的网际操作系统（IOS）是一个为网际互连优化的复杂操作系统。

Cisco IOS 软件网络地址转换功能在转换 SIP 报文时存在拒绝服务漏洞，当受影响设备上的报文传输要求在 SIP 负载上转换时会触发此漏洞，造成内存资源耗尽。

危害：

攻击者可以利用此漏洞向服务器发送恶意请求，造成拒绝服务。

7. 2012-03-22 织梦 CMS 后门远程代码执行漏洞

NSFOCUS ID: 19203

http://www.nsfocus.net/vulndb/19203

综述:

织梦 CMS 是国内一款基于 PHP+MySQL 技术开发的内容管理系统。

织梦 CMS 某些版本 /include/shopcar.class.php 文件中，被添加后门代码。

危害:

攻击者可以利用此漏洞向服务器发送恶意请求，从而控制服务器系统。

8. 2012-03-15 McAfee Email/Web Security Appliance/ Email Gateway 多个安全漏洞

NSFOCUS ID: 19066

http://www.nsfocus.net/vulndb/19066

综述:

McAfee Email Gateway 之前名为 IronMail，是企业级的硬件邮件网关和管理平台。

McAfee Email、Web Security Appliance、Email Gateway 在实现上存在多个漏洞，包括：未正确过滤某些输入、重置管理员密码、泄露活动会话令牌、弱加密存储密码、目录遍历等。

危害:

攻击者可以利用此漏洞向服务器发送恶意请求，从而获取敏感信息甚至控制服务器系统。

9. 2012-03-14 libpng 'png_inflate()' 堆缓冲区溢出漏洞

NSFOCUS ID: 19062

http://www.nsfocus.net/vulndb/19062

综述:

libpng 是多种应用程序所使用的解析 PNG 图形格式的函数库。

libpng 在 png_inflate() 函数的实现上存在远程堆缓冲区溢出漏洞。

危害:

攻击者可以利用此漏洞诱使受害者打开恶意 PNG 图片，从而控制受害者系统。

10. 2012-03-09 AMD CPU 实现安全漏洞

NSFOCUS ID: 18989

http://www.nsfocus.net/vulndb/18989

综述:

AMD 是一家业务遍及全球的集成电路供应商，专为电脑、通信及电子消费类市场供应各种芯片产品。

AMD 某些处理器在分段和 fill_sons_in_loop 代码的实现上存在安全漏洞，通过特定的连续 back-to-back pops 及 return 指令序列，可造成栈指针更新错误。

危害:

攻击者可以通过诱使受害者执行畸形代码，造成拒绝服务。

NSFOCUS 2012年4月之十大安全漏洞

声明：本十大安全漏洞由 NSFOCUS(绿盟科技) 安全小组 <security@nsfocus.com> 根据安全漏洞的严重程度、利用难易程度、影响范围等因素综合评出，仅供参考。http://www.nsfocus.net/index.php?act=sec_bug&do=top_ten

1. 2012-04-11 Microsoft Windows Common Controls ActiveX 控件远程代码执行漏洞 (MS12-027)

NSFOCUS ID: 19325

http://www.nsfocus.net/vulndb/19325

综述：

Microsoft Windows 是流行的计算机操作系统。

Microsoft Windows Common Controls 的 MSCOMCTL.TreeView、MSCOMCTL.ListView2、MSCOMCTL.TreeView2、MSCOMCTL.ListView 控件 (MSCOMCTL.OCX) 中存在错误，可被利用破坏内存，导致任意代码执行。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意文件，从而控制受害者系统。

2. 2012-04-09 Adobe Flash Player 远程内存破坏漏洞 (CVE-2012-0725)

NSFOCUS ID: 19309

http://www.nsfocus.net/vulndb/19309

综述：

Adobe Flash Player 是一个集成的多媒体播放器。

Adobe Flash Player 在实现上存在远程内存破坏漏洞，攻击者可利用这些漏洞执行任意代码。

危害：

攻击者可以利用此漏洞诱使受害者打开恶意 swf 文件，从而控制受害者系统。

3. 2012-04-10 GE Energy D20/D200 Substation Controller 缓冲区溢出漏洞

NSFOCUS ID: 19315

http://www.nsfocus.net/vulndb/19315

综述:

D20/D200 Substation Controller 是 IED 的 SCADA 主站、下流子站和供给器的网关。

D20/D200 Substation Controller 在实现上存在缓冲区溢出安全漏洞，可导致执行任意代码。

危害:

攻击者可以利用此漏洞提交恶意请求，从而控制服务器系统。

4. 2012-04-09 Oracle Java SE 多个安全限制绕过漏洞

NSFOCUS ID: 19307

http://www.nsfocus.net/vulndb/19307

综述:

Sun Java Runtime Environment 是一款为 JAVA 应用程序提供可靠的运行环境的解决方案。

Oracle Java SE 在实现上存在多个远程漏洞。

危害:

攻击者可利用这些漏洞绕过 Java 沙盒安全限制，执行非法操作。

5. 2012-04-09 Google Chrome 18.0.1025.151 之前版本多个安全漏洞

NSFOCUS ID: 19308

http://www.nsfocus.net/vulndb/19308

综述:

Google Chrome 是由 Google 开发的一款设计简单、高效的 Web 浏览工具。

Google Chrome 18.0.1025.151 之前版本在实现上存在多个安全漏洞。

危害:

攻击者可利用这些漏洞执行任意代码、绕过安全限制、执行跨站脚本攻击。

6. 2012-04-30 Oracle Database Server 'TNS Listener' 远程数据投毒漏洞

NSFOCUS ID: 19508

http://www.nsfocus.net/vulndb/19508

综述:

Oracle Database Server 是一个对象—关系数据库管理系统。

Oracle Database Server 在实现上存在可允许攻击者向远程 'TNS Listener' 组件处理的数据投毒的漏洞，攻击者无需用户名和密码，可利用此漏洞将数据库服务器的合法 'TNSListener' 组件中的数据转向到攻击者控制的系统。

危害:

攻击者可以利用这些漏洞进行会话劫持、拒绝服务甚至控制数据库服务器。

7. 2012-04-10 RealNetworks Helix Server 多个远程安全漏洞

NSFOCUS ID: 19314

http://www.nsfocus.net/vulndb/19314

综述:

RealNetwork Helix Server 是一款支持多格式、跨平台的流媒体服务器软件。

Helix Server 在实现时存在多个漏洞，包括泄露口令，输入验证错误，缓冲区溢出。

危害:

攻击者可以利用这些漏洞获取敏感信息、执行跨站脚本攻击、造成拒绝服务或控制有缺陷系统。

8. 2012-04-24 Asterisk Shell 命令执行安全限制绕过漏洞

NSFOCUS ID: 19466

http://www.nsfocus.net/vulndb/19466

综述:

Asterisk 是一款实现电话用户交换机（PBX）功能的自由软件、开源软件。

Asterisk 在实现上存在安全限制绕过漏洞，攻击者可利用此漏洞绕过某些安全限制并在受影响应用背景下执行 shell 命令。

危害:

攻击者可以利用此漏洞提交恶意请求，从而控制服务器系统。

9. 2012-04-27 Discuz! X2.5 远程代码执行漏洞

NSFOCUS ID: 19501

http://www.nsfocus.net/vulndb/19501

综述:

Discuz 论坛软件系统亦称电子公告板（BBS）系统。

Discuz! X2.5 Release 20120407 版中的 preg_replace 使用了 e 修饰符和双引号，可执行任意代码。要成功利用此漏洞需要目标启用 seo 功能。

危害:

攻击者可以利用此漏洞提交恶意请求，从而控制服务器系统。

10. 2012-04-09 Siemens Scalance Firewall 两个安全漏洞

NSFOCUS ID: 19306

http://www.nsfocus.net/vulndb/19306

综述:

Siemens Scalance Firewall 可以多种方式过滤进站和出站网络连接，保证可信工业网络的安全。

Siemens Scalance Firewall 在实现上存在多个漏洞。Web 配置接口登录失败后没有时间延迟。处理 Profinet DCP 协议时，通过特制的 DCP 报文，可造成防火墙不响应并中断已经建立的 VPN 通道。

危害:

攻击者可以利用这些漏洞进行暴力攻击或造成拒绝服务。

THE EXPERT
BEHIND GIANTS

NSFOCUS

公司总部: 北京市海淀区北洼路4号益泰大厦三层 010-68438880
服务热线: 400-818-6868 值班热线: 13321167330（非工作时间） 技术支持传真: 010-68437328
技术支持网站: http://support.nsfocus.com 技术支持邮箱: support@nsfocus.com

www.nsfocus.com

SECURITY+

THE EXPERT BEHIND GIANTS 巨人背后的专家

安全+ 技术版

与安全人士分享技术心得

Share technique experience with security professionals

Nsmagazine@nsfocus.com