通用 Web 漏洞库①

张昊星, 孙应飞

(中国科学院大学 电子电气与通信工程学院, 北京 100049)

摘 要: 本文研究了国内外 Web 漏洞库及建设的现状, 设计并实现了一个专注于 Web 漏洞发布的 Web 漏洞数据库. 文中兼顾了 Web 漏洞的固有特点及其与传统漏洞的属性差别, 设计了 Web 漏洞库描述模型, 丰富了 Web 漏洞的收集方法, 定义了 Web 漏洞的漏洞评价属性标准, 并在 Web 漏洞库中添加了 Web 漏洞重现模块. 我们所设计的 Web 漏洞库确保了全面的 Web 漏洞信息收集和 Web 漏洞信息发布的标准化, 可更好地对 Web 漏洞信息和数据进行分析研究, 也为 Web 安全提供了有力的技术支撑.

关键词: Web 漏洞; 属性提取; 漏洞重现; Web 漏洞库

Common Web Vulnerability Database

ZHANG Hao-Xing, SUN Ying-Fei

(School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Based on the research of Web vulnerability database and the situation of vulnerability database construction at home and aboard, the paper designed and implemented a vulnerability database focused on Web vulnerabilities. In consideration of both the features of Web vulnerability and the differences with traditional vulnerability, the paper designed the Web vulnerability database description model, enriched the ways of Web vulnerability collection, redefined the Web vulnerability scoring attributes and added the Web vulnerability reproduce function. The Web vulnerability database guarantees the comprehensive collection of Web vulnerability information and the standard release of Web vulnerability information, helps analyze the Web vulnerability information and data better, and provides a powerful technical support to Web security.

Key words: Web vulnerability; attribute extraction; vulnerability reproduction; Web vulnerability database

随着互联网大众化和 Web 2.0 时代的出现, 在线网络安全所面临的挑战日益严峻. 标准化的 Web 用户服务软件在向 Web 介质方向发展, AJAX 技术和其他 Web2.0 技术日益普及, 伴随着在线信息和服务可用性的提升, 可以确认在线应用防护方面的现实需求继续增长, 于是安全风险也达到了前所未有的高度. 以往的众多安全工作集中在网络本身, 所以导致 Web 应用程序被有所忽略. 通过 Web 漏洞所实施的攻击行为将会更多针对客户端、Web 服务器、数据库和应用服务器等, 更加隐蔽和容易实施, 危害更大. 因此我们需要采取更有效的措施来减少损失和破坏. 构建 Web 漏洞库是一种降低威胁的非常有效的手段.

Web 漏洞与传统漏洞有所区别, 所以 Web 漏洞库既要保留传统漏洞库的特点和部分属性描述, 兼容国际漏洞标准, 也要适应 Web 漏洞所带来的新的要求和变化, 为广大互联网用户、厂商和政府部门提供一个更有效、更准确、更完备的信息交流共享平台. 本文为解决这个问题提出了一个通用 Web 漏洞库模型.

1 相关工作

1.1 Web 漏洞的现状和发展趋势

近几年来, Web 漏洞数量呈现一个快速增长的势头. 据 IBM 公司的 X-Force 漏洞库数据显示[1], 2006 年之前每年漏洞总数呈上升趋势, 在 2007 年总数第一次

① 基金项目:国家自然科学基金(60970140);北京自然科学基金(4122089)
收稿时间:2013-04-24;收到修改稿时间:2013-05-08

下降, 而截止 2011 年间的这几年, 漏洞总数呈现规律地隔年上升-下降的趋势, 如图 1 所示.

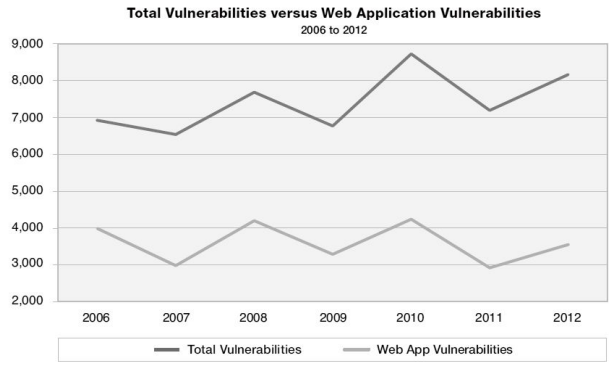

图 1 1996 年至 2012 年披露漏洞数量增长趋势

目前还没有一个比较准确的定义来解释这种规律性地波动, 但是据图中数据显示 2010 年的漏洞数据已经达到最高, 2012 年的漏洞总数仅次于 2010 年居第二位. 而在 2012 年全年的 X-Force 已披露漏洞数据中, Web 漏洞数量已经占到漏洞总数的 47%.

所以, Web 漏洞数量的急速增长已有赶超传统漏洞的趋势, Web 漏洞管理的缺失更使 Web 漏洞库的建设显得尤为重要, 更好地收集分析 Web 漏洞将会对未来的互联网络发展起到一个积极的促进作用.

1.2 Web 漏洞库建设意义及发展现状

漏洞库建设的意义在于收集储存的大量漏洞信息, 这是网络安全隐患分析的核心[2]. 对不同的用户群体, 要满足不同的用户需求. 例如可满足普通用户关注个人使用产品涉及的漏洞的信息, 可有效保护个人隐私和利益不受侵害, 防范恶意攻击; 满足生产厂商对自身产品漏洞做更多研究, 进行更多安全需求的调整和更便捷合理的优化等.

在信息化时代的背景下, 漏洞库已经涉及到国家安全, 因此欧美发达国家早已开展了漏洞库的相关研究和建设. 美国经过多年的积累和完善已经形成了一整套完整的体系, 制定了一系列国际化的有影响力的标准, 建设了多个大型漏洞库, 如美国国家漏洞库 NVD[3]等. 我国也已在最近几年开始重视漏洞库的建设, 并推出了几个颇具规模的漏洞数据库.

而对于 Web 漏洞, 各国还没有建设起规模比较大、体系比较完善的 Web 漏洞数据库, 只在传统漏洞库中有收录部分 Web 漏洞信息, 或于民间有一些非盈利机构建设的较小规模的 Web 漏洞数据库.

如 NVD 中, 截止 2012 年底共收录漏洞信息 54000 余条, 其中 Web 漏洞条目仅 8000 条左右, 远低于之前提到的占近 50%的漏洞比例的 Web 漏洞.

WooYun(乌云)[4]是国内目前比较知名的一个漏洞发布平台, 也是对于 Web 漏洞比较重视的一个漏洞发布平台. 但是 Wooyun 的漏洞收集方式比较单一, 仅仅通过用户上报来充实漏洞数据, 同时对 Web 漏洞和传统漏洞同一评分标准, 忽略了 Web 漏洞与传统漏洞差异, 无法准确定义 Web 漏洞的危害.

1.3 目前 Web 漏洞库建设存在的问题

目前国内外并没有一个专门针对 Web 漏洞建设的漏洞数据库. 已收录 Web 漏洞的漏洞库由于缺乏针对性, 导致 Web 漏洞数据的不完整不全面. 同时未对 Web 漏洞进行规范的分类和整理, Web 漏洞与传统漏洞混淆在一起, 没有考虑二者的差异, 对 Web 漏洞的信息描述、漏洞分类、评级标准等方面与传统漏洞没有加以区分, 形成统一规范. 这在 Web 漏洞飞速增长并造成较严重的危害的当前形势下, 既无法准确衡量 Web 漏洞的危害并加以预警和遏制, 也无法满足用户对 Web 安全和 Web 漏洞数据进行研究和分析的需求.

2 Web漏洞库设计方案

2.1 Web 漏洞库与传统漏洞库区别

Web 漏洞有很多新出现的特征和属性. 对比来看, Web 漏洞与传统漏洞的区别主要可概括为以下几点: 攻击对象不同、攻击语言不同、攻击范围不同、生命周期不同、危害影响不同等. 这些不同使得 Web 漏洞库在建设上也与传统漏洞库有很大的差异, 主要有:

1) 漏洞分类不同. Web 漏洞包含的漏洞类型与传统漏洞有所区别, 典型的 Web 漏洞如 SQL 注入漏洞(SQL Injection)、跨站脚本漏洞(Cross-Site Script)、跨站请求伪造漏洞(Cross-Site Request Forgery)等, 使漏洞类型的划分不能再单一按以前的标准来进行.

2) 漏洞评级不同. Web 漏洞所影响的对象一般是 Web 应用程序、网站或 Web 服务器, Web 漏洞的危害等级并不能简单地根据漏洞本身来确定, 而是要基于漏洞危害、所影响目标的流行性以及具体的攻击方式和效果等多方面因素进行综合评级, 给出其危害等级.

3) 漏洞收集方式多样. Web 漏洞的特点要求 Web 漏洞库的漏洞信息收集要通过多种方式来完善. 可以

选择国际知名数据库提取 Web 漏洞信息，通过用户漏洞上报进行收集整理，或建设 Web 漏洞论坛来收集 Web 漏洞等. 这样才可保证 Web 漏洞数据的翔实全面.

4) 漏洞信息更新. Web 漏洞的生命周期与传统漏洞有所区别，由于漏洞影响目标的特殊性, Web 漏洞可能在生产厂商发布解决方案或更新补丁以后，危害性大大降低. 这要求数据库每过一段时间要对漏洞信息进行更新，调整已修复漏洞的危害等级，对于用户查询数据库数据并进行数据统计时更加准确.

5) 修复方式不同. 由于 Web 漏洞攻击对象的特殊性，Web 漏洞的修复只需由生产厂商在 Web 应用后台或网站的服务器端进行修复或加强安全防护，而不像传统漏洞那样在客户端对软件系统安装补丁更新.

以上即为 Web 漏洞库与传统漏洞库有所区别的几个方面，可根据这些不同来完成 Web 漏洞库的设计.

2.2 Web 漏洞库系统架构

本文设计的 Web 漏洞库使用 B/S 架构(Browser/Server)，通过 HTTP 协议提供前端服务. 这样既为用户的使用提供了方便，使得浏览提交漏洞信息和查询检索等操作提高了效率，也使数据库系统在降低成本的同时，管理维护也更加简单，系统的升级和扩展也更加便捷. 如图 2 所示.

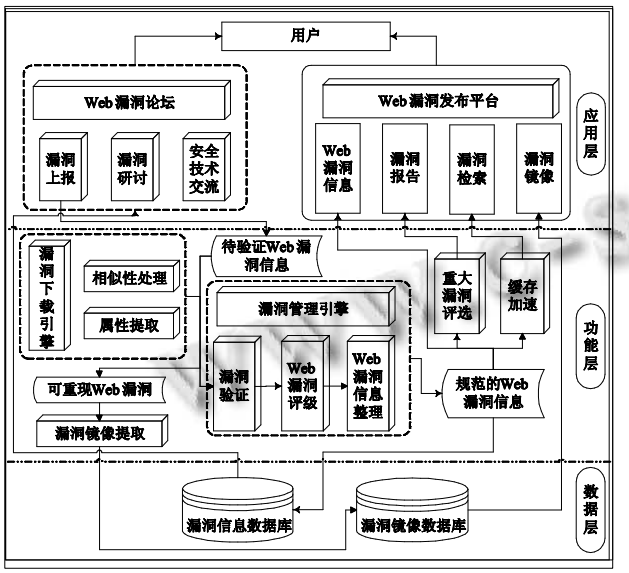

图 2　Web 漏洞库系统架构图

前台的漏洞论坛和发布平台基于 Windows 平台和 ASP.NET 技术开发，使用 IIS 作为 Web 服务器；功能层的漏洞下载、属性提取等功能用 C#语言开发；数据层选择 SQL Server 关系数据库存储数据和漏洞镜像.

Web 漏洞库的设计主要针对 Web 漏洞的特点进行. 整个漏洞库分为表示层、功能层和数据层. 应用层分为两部分，Web 漏洞论坛用来给用户提供一个 Web 安全交流平台并作为漏洞库的一个数据源来进行 Web 漏洞的收集, Web 漏洞发布平台主要进行 Web 漏洞信息、Web 漏洞报告的发布并提供漏洞的检索统计. 功能层主要进行的是漏洞下载、漏洞信息的管理整合以及 Web 漏洞库特有功能的实现. 数据层主要进行漏洞信息的存储以及可重现漏洞的镜像文件存储.

2.3 Web 漏洞结构模型和分类

Web 漏洞与传统漏洞相比有许多新的属性，对于 Web 漏洞的描述也与传统漏洞有所区别. 《安全漏洞标识与描述规范》[5]是规定了漏洞库基础结构的一个标准，它结合漏洞特点给出了漏洞的 11 个属性来描述漏洞. 对细节更多的 Web 漏洞这 11 个属性更加宽泛. 因此本文结合 Web 漏洞自身属性特点，抽取了 Web 漏洞的 24 个固有属性，更加全面地对 Web 漏洞信息进行描述. Web 漏洞属性的结构模型如图 3.

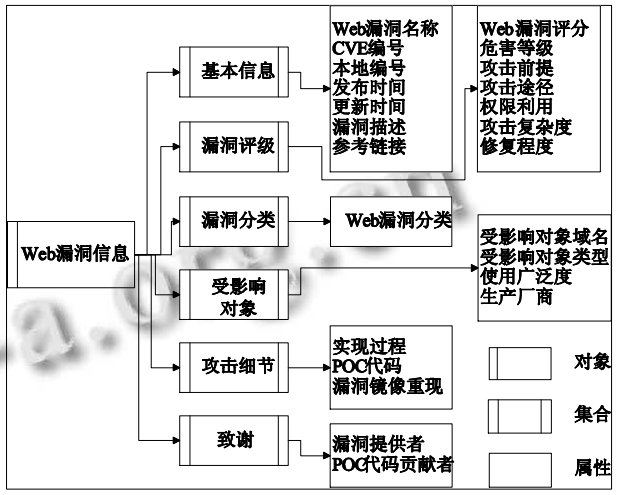

图 3　Web 漏洞库结构模型

我们将这 24 个属性归纳为六个属性集合，每个属性集合刻画 Web 漏洞的一类属性信息，且属性集合之间相互独立. 包括基本信息集合，漏洞评级属性集合，漏洞分类信息集合，影响对象属性集合, Web 漏洞利用方式属性集合，和漏洞来源属性集合. 通过这六大类 Web 漏洞属性集合，本文提出了一种通用的 Web 漏洞属性模型，帮助对 Web 漏洞信息进行准确标准的描述，以保证安全工作者有针对性地获取自己需要的 Web 漏

洞信息, 也方便构建和维护通用的 Web 漏洞库.

2.4 Web 漏洞库主要功能模块

2.4.1 Web 漏洞论坛

Web 漏洞论坛主要面向的是生产厂商和关心 Web 安全的用户. 首先, 漏洞论坛是一个技术交流的平台, 厂商和技术用户可在论坛中发布与 Web 安全有关的时事、安全技术、漏洞研讨或其他经过审核的和 Web 安全有关的内容. 其次, Web 漏洞的特点使得漏洞收集的数据源要十分广泛, 因此注册用户通过漏洞上报功能, 将疑似或已经证实但还未发布的 Web 漏洞信息通过网页表单和邮件上报给论坛管理员. 经验证后, 整理漏洞信息录入漏洞库并发布. 为保证未验证漏洞及还未公布的已验证漏洞信息泄露被攻击者加以利用, 要对漏洞信息采用加密方式传送.

同时, 由于漏洞上报功能和漏洞验证的程序, 涉及未公布的漏洞信息, 因此要实施规范的用户角色管理并执行严格的权限访问控制方案, 要求论坛和漏洞库用户全部实名注册并进行身份验证. 漏洞库用户角色主要分为游客、会员、厂商和安全专家以及漏洞库管理员, 具体权限见表 1.

表 1 用户角色权限管理方案

	匿名用户	会员用户	厂商、安全专家	漏洞库管理员
浏览漏洞信息	√	√	√	√
注册获取权限	—	√	√	√
漏洞上报	—	√	√	√
分析、验证漏洞	—	—	√	√
漏洞库管理	—	—	—	√
角色授权	—	—	—	√

另外, 对通过 Web 页面提交的 Web 漏洞信息进行加密传输, 要求通过邮件提交漏洞的用户采取加密措施, 保证未验证漏洞在上报过程中不被未知人员窃听截获, 给用户和厂商造成不必要的损失.

2.4.2 Web 漏洞发布平台

Web 漏洞发布平台是面向所有互联网用户的. 用户可以通过平台浏览已发布的 Web 漏洞信息, 查看下载漏洞报告, 进行漏洞的查询检索并对查询结果进行数据统计, 此外对可以重现的 Web 漏洞用户可以查看该漏洞镜像以及攻击效果.

平台通过 Web 页面提供详细完整的漏洞信息和强大的查询检索功能, 帮助用户轻松找到自己关心的某个或某类 Web 漏洞, 并利用数据统计功能对得到的结果进行数据分析. 同时, 平台会统计每周披露的 Web 漏洞数据, 根据漏洞属性中的多个相关影响因素, 如漏洞等级、攻击复杂度、攻击效果、受影响对象使用广泛度和使用率、修复程度等, 进行加权计算, 按照得分高低对新披露 Web 漏洞进行排序, 分别撰写每周、每月安全公告, 每周/每月评选出五大/十大高危漏洞, 详细描述这些漏洞的所有信息, 同时对每周、每月总体漏洞数据做环比分析, 提供每周、每月 Web 漏洞安全态势分析. 这将帮助用户更有针对性地做好安全防护, 提高用户在上网过程中的自我防范意识.

此外, 由于跨站脚本(XSS)漏洞、SQL 注入漏洞等类型的 Web 漏洞可能进行重现, 漏洞库会将可重现的漏洞镜像页面和文件保存的本地数据库, 并供用户进行漏洞重现, 既可向普通用户普及 Web 漏洞的攻击原理、攻击方式和攻击效果, 也可帮助技术用户对漏洞产生、漏洞利用等方面进行更好的分析.

2.4.3 漏洞下载引擎

由于 Web 漏洞存在的普遍性和隐藏性, 因此漏洞论坛的用户上报漏洞功能是必须的. 但是单一的漏洞收集方式会导致 Web 漏洞信息的不全面和不完整, 民间的漏洞挖掘技术有时不能发现安全防护比较完善的网站存在的 Web 漏洞和漏洞利用技术比较复杂的 Web 漏洞. 因此, Web 漏洞库选择国外一些知名的大型漏洞库作为另一个 Web 漏洞库的数据源. 开发使用漏洞下载工具从其中的漏洞数据中, 通过关键字、漏洞类型、漏洞描述等字段提取 Web 漏洞信息. 漏洞下载引擎能够自动下载指定的漏洞库数据, 判断是否属于 Web 漏洞, 筛选重复漏洞条目, 提取 Web 漏洞的各属性信息并提交给漏洞管理引擎进行整理规范.

漏洞下载引擎与 Web 漏洞论坛的漏洞上报模块相辅相成, 丰富了 Web 漏洞库的漏洞数据, 使其更加权威, 尽可能地覆盖收集所有漏洞库以及 Web 产品的漏洞数据, 可以更好地为 Web 安全研究人员提供一个全面、准确、详尽的漏洞数据库.

2.4.4 漏洞管理引擎

漏洞管理引擎是将漏洞下载引擎和用户个人上报的带验证漏洞信息进行汇总整理后, 得到规范的漏洞信息, 然后发布到 Web 漏洞发布平台上. 大致可以分为几个步骤, 首先是对待验证的漏洞进行验证核实, 在确认漏洞存在并核实相关细节后, 按照 Web 漏洞的描述提取各个属性并对 Web 漏洞进行评级, 最后得到

规范的漏洞信息并发布. 以下是各步的具体实施方案:

1) 漏洞验证: 漏洞验证模块主要体现的是待验证漏洞信息的一个验证过程. 漏洞库管理员在 Web 漏洞库后台对可疑漏洞和待验证漏洞进行简单地收集整理和甄别, 将其相关技术信息上报至相应生产厂商或相关安全专家处, 等待验证. 生产厂商和安全专家通过专业技术手段和漏洞提交者的协助, 对可疑漏洞进行分析验证并提供验证报告, 内容包括 Web 漏洞库中漏洞条目的详细信息和漏洞修复方式. 若漏洞经验证真实存在, 漏洞库管理员对报告中的漏洞信息进行整理并录入数据库, 同时发布公告详细描述漏洞的相关信息以及修复补丁或安全建议.

2) Web 漏洞评级: 结合第二部分提到的 Web 漏洞与传统漏洞的区别, 那么在评级方式上就要对 Web 漏洞进行重新定义以得到更准确的 Web 漏洞威胁程度. 通用漏洞评价体系(Common Vulnerability Scoring System, CVSS)[6]是目前国内外比较认可的一个传统漏洞评价体系. 本文在 CVSS 的评价体系上进行了修改和调整, 建立了一套 Web 漏洞评价体系标准.

与 CVSS 的评价机制类似, 将 Web 漏洞属性划分为基本评价组、生命周期评价组和环境评价组, 每个评价组包含不同特征的漏洞属性, 对每个评价组中的漏洞属性进行单独评分, 然后每个评价组得分得到最后的漏洞威胁程度[7]. 以下各表为 Web 漏洞评价体系的度量标准.

表 2 Web 漏洞基本评价组评价要素及等级

评价要素	评价属性	危害等级
Access Premise (攻击前提)	不需要/需要	高/低
Access Pattern (攻击方式)	远程/本地	高/低
Access Complexity (攻击复杂度)	高/中/低	高/中/低
Access Vectors (攻击向量数)	多于 1 个/1 个	高/中
Privileg (权限获取)	管理员/用户/其他	高/中/低

攻击前提: 指漏洞被利用是否需要触发条件. 如果需要被攻击者帮助, 则该漏洞的危害性较可以被自动触发的漏洞危害性较低. 例如点击劫持漏洞.

攻击方式: 指 Web 漏洞被利用的方式. 本地攻击需要攻击者与受攻击的对象有物理接触, 或者已具本地权限. 远程攻击不局限于本地, 可通过远程直接发起, 既降低了难度, 也提高了隐蔽性.

攻击复杂度: 根据攻击实施的复杂程度, 例如是否需要攻击工具、攻击技术的复杂度以及是否有公开的攻击方法等综合得到攻击复杂度属性等级.

攻击向量数: 攻击向量就是攻击方法. 通常情况下, 攻击向量越多, 危害性越高.

权限获取: 实施攻击时, 拥有的权限越高则攻击越容易实现. 根据利用漏洞可以得到的权限等级, 漏洞的危害性也有所不同.

表 3 Web 漏洞生命周期评价组评价要素及等级

评价要素	评价属性	危害等级
TechDetails (技术细节)	已公开/未公开/未知	高/低/Null
Popularity (受影响对象访问量)	高/中/低	高/中/低
Relevance (关联性)	多于 3 个/1-3 个/无	高/中/低
Users (用户群体)	大众/群体/专业	高/中/低
Remediation (修复状态)	未修复/已修复	高/低

技术细节: 漏洞的技术细节是否公布对于漏洞危害的影响. 包括攻击方式、攻击代码、攻击技术等. 细节的披露使得漏洞被攻击的难度降低、可能性更大.

受影响对象访问量: 当某个网站访问量非常大时, 一旦存在 Web 漏洞, 则受攻击用户的数量和危害也就越大. 这里可以根据 Alexa 网站对网站排名设定的影响因子来判断受影响对象的使用广泛程度. 首先就可根据 Alexa 的排名来定义, 其次还可根据平台用户量、人均页面访问量、蹦失率(用户浏览第一个页面就离开的访问次数占该入口总访问次数的比例)、访问时间等要素来判断.

关联性: 即受影响对象与其他对象的关联程度. 一个网站在访问过程中可能会与其他网站进行交互, 或调用其他的 Web 服务和 Web API 等. 这使得 Web 漏洞的威胁范围更大, 甚至可能导致受影响对象的数量指数型增长.

用户群体: Web 应用、Web 网站的用户群体也会影响到 Web 漏洞的危害性. 面向大众用户的, 使用率高, 若存在 Web 漏洞则危害性大. 面向某类群体用户的, 则危害性适中. 专业性强, 使用用户少, 则危害性也会小很多.

修复状态: Web 漏洞的修复多由生产厂商在服务

器端进行, 无需用户安装修复补丁. 因此只需关注漏洞是否被修复, 来判断漏洞的危害程度.

环境评价组属性对其不做调整. 最后根据漏洞每个评价组的评分, 参考 CVSS 的评价方程式, 计算 Web 漏洞最终评分, 根据评分得到漏洞的危害等级.

2.4.5 Web 漏洞重现模块

Web 漏洞多由厂商在服务器端进行修复, 一旦被修复就无法对 Web 漏洞进行重现. 但由于 Web 漏洞的特点, 使得某些类型的 Web 漏洞可以将其所在平台的相关数据信息保存下来, 在漏洞被修复后可以调取修复前的平台镜像对漏洞进行重现, 类似于传统漏洞可在旧版本的系统或软件重现的道理. 这样可以帮助用户在厂商为避免损失及时对 Web 漏洞修复后, 还能重现漏洞, 分析研究漏洞的相关细节.

Web 漏洞只能应用于某些类型的 Web 漏洞, 例如跨站脚本漏洞、跨站请求伪造漏洞、未验证的重定向/转发漏洞等一些特殊的 Web 漏洞如点击劫持漏洞等. 这些漏洞可以通过修改 HTML 代码来实现代码执行、文件上传等攻击而无需连接厂商后台的服务器数据库, 这样只需将漏洞所在页面的 HTML 镜像文件保存到本地并对其实施渗透测试即可重现漏洞. 而类似于 SQL 注入、目录遍历等类型的漏洞, 攻击除需要 HTML 页面外, 还需要连接厂商的后台数据库才能实现, 这类漏洞则无法通过保存漏洞的页面镜像重现. 因此, 对于前者一类的 Web 漏洞和已公布攻击细节的漏洞, 可将其受攻击的文件镜像及攻击环境保存在本地镜像数据库, 用户点击镜像即可实现漏洞现和攻击效果.

2.5 漏洞查询管理模块

Web 漏洞库建设的目的除了给用户和厂商提供一个漏洞提交、验证、发布的平台, 另一个主要的目的是方便用户对已公布漏洞进行高效率的查询检索, 并可以进行数据统计. 漏洞信息中可供查询的属性有:

表 4 Web 漏洞库查询属性

漏洞属性	查询类型
漏洞名称	模糊查询
关键字	模糊查询
漏洞编号	精确查询
漏洞发布/更新日期	模糊查询
漏洞类型	分类查询
危害等级	分类查询
攻击方式	分类查询
攻击效果	分类查询
受影响对象	模糊查询

用户可以根据自身需要选择以上属性进行简单查询, 模糊查询和精确查询需要用户输入查询信息, 分类查询需要用户选择需要查询的类别. 同时用户可以选择多个属性进行高级查询. 此外, 用可以对查询结果进行数据统计, 满足更多的需求分析. 可选择多个属性查询两次, 区别其中某个属性, 对查询数据进行比较得到需要的结果, 如查询某时间段内某著名网站被攻击利用的不同类型的漏洞, 得出该网站最易受到哪类漏洞的攻击. 这样, 使 Web 漏洞库的数据被利用的更加充分和高效, 对 Web 漏洞的分析研究更有帮助.

3 Web漏洞库的具体实现

3.1 Web 漏洞论坛及漏洞发布平台

Web 漏洞库架构主要由 Web 漏洞网站前端以及后台漏洞库构成. Web 漏洞网站前端包括 Web 漏洞论坛和漏洞发布网站.

Web 漏洞论坛和 Web 漏洞发布平台架构在两个相同配置的 Web 服务器上, 由 ASP.NET 实现, Web 漏洞论坛主要包括漏洞上报模块和论坛模块.

漏洞上报模块字段设计: 漏洞标题、漏洞类型、漏洞危害等级(紧急、高、中、低)、问题厂商、漏洞描述、漏洞细节、攻击代码、解决方案、上报人等.

漏洞论坛模块: 漏洞论坛主要以 Web 漏洞上报功能为主, 同时辅以漏洞信息研讨信息安全技术交流、信息安全时事、政策发布等版块, 满足用户不同需求.

Web 漏洞发布平台包括: web 漏洞信息展示, 漏洞检索功能, 漏洞上报功能, 以及漏洞镜像重现功能.

Web 漏洞信息展示模块: 结合 Web 漏洞结构模型, 主要包括漏洞名称、CVE 编号、本地编号、漏洞类型、发布时间、更新时间、漏洞评分、危害等级、攻击前提、攻击途径、漏洞描述、受影响对象域名、受影响对象类型、攻击代码、修复状态、漏洞镜像、致谢.

漏洞检索功能: 关键字检索, 标签检索, 按时排序, 按厂商排序, 按类型排序等功能.

漏洞报告页面: 通过分析每周、每月、每年的 Web 漏洞信息, 得到各时间段内的 Web 安全形势并对下个阶段进行安全形势预测, 给出漏洞报告.

Web 漏洞论坛与发布平台这两部分前端服务架设在 IIS 服务器上, 并响应浏览器的请求. 网站前端接受 Web 请求, 并通过内置对象 Active Data Object (ADO) 对后台数据库进行查询并响应. Web 请求包括两类:

一类是 Web 漏洞信息访问请求，前端直接将请求转发到后端，对后端数据进行读取响应. 一类是 Web 漏洞信息提交，前端对提交的 Web 漏洞数据进行形式化处理，并计算相似度以及漏洞镜像制作，最后录入后端数据库之中.

3.2 Web 漏洞下载引擎和漏洞属性提取

漏洞下载引擎丰富了传统 Web 漏洞信息收集的方式，从其他漏洞数据库中收集相应的 Web 漏洞信息以保证漏洞库的基本 Web 漏洞信息储备. 本文选取了 NVD 、 X-Force 、 Secunia 、 WooYun 、 OSVDB 、 SecurityFocus 等几个国内外知名漏洞库作为数据源来提取 Web 漏洞并下载.

由于数据源较多，而每个漏洞库的漏洞描述格式不尽相同，因此传统的使用硬编码抽取规则的漏洞下载工具无法满足对不同格式 Web 页的漏洞信息的提取. 目前的漏洞库使用 XML/HTML 格式来发布漏洞且 HTML 格式居多，因此对漏洞信息的提取就是对发布漏洞信息的 Web 页面进行提取. 通过比较几种 Web 信息抽取技术[8]，选择使用 XML 路径语言(XPath)技术对不同漏洞库实现漏洞的自动下载[9-11].

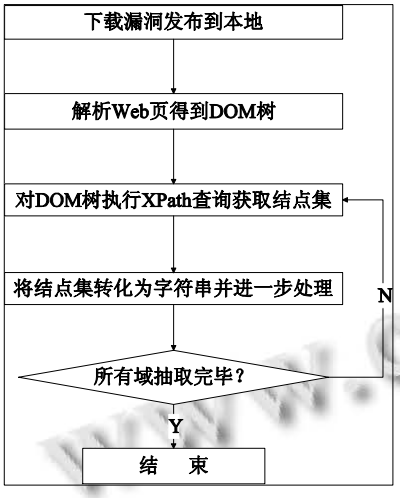

图 4　下载引擎算法流程

根据 XML/HTML 文档格式由基本构成单元标签层层嵌套的特点，可以使用文档对象模型(DOM)对其进行结构化建模. 即可得到漏洞信息页面的 DOM 树，然后利用 DOM 提供的 API 接口即可对文档树中结点进行编辑和其他操作. 这时就可以利用 XPath 技术，结合不同的规则来针对每个漏洞库的漏洞发布页面构造特定的 XPath 表达式，匹配到文档树中的每个节点并将其转化为字符串，再通过配置文件对字符串进行过滤处理，得到所需要的漏洞信息.

这样在结点匹配提取和过滤过程中，可以对关键字、漏洞名称、漏洞类型和漏洞描述等字段进行筛选，提取出属于 Web 漏洞的条目信息. 同时也将每个漏洞的漏洞属性提取出来，更方便管理员对漏洞信息的整理和维护. 此外，不同漏洞库的漏洞发布页面格式和漏洞描述不尽相同，使用 XPath 技术可为每个漏洞库编写相应的配置文件以匹配不同格式的 XML/HTML 文档，如果 Web 页面格式改变，只要修改配置文件即可完成漏洞下载，也提高了漏洞库的维护效率.

3.3 Web 漏洞相似性处理

Web 漏洞库的漏洞信息来源主要包括：其他 Web 漏洞网站获取和漏洞发现者提交. 对于获取的漏洞信息，我们在前端服务器进行相似性处理并赋予唯一的漏洞标识，以防止 Web 漏洞信息的冗余.

漏洞相似性处理流程主要包括以下几个步骤：

1) Web 漏洞信息形式化. 通过对 Web 漏洞信息的语义进行分析和形式化，将不同描述语义的 Web 漏洞信息以我们定制的通用漏洞属性进行描述和规范，并存入数据库.

2) Web 漏洞分类. 根据形式化的 Web 漏洞语义和已经定义的 Web 漏洞分类表，对 Web 漏洞标识类别和其他描述信息.

3) Web 漏洞相似性计算. 对于每个 Web 漏洞，其所有属性语义构成了一个 n 维向量，向量中的每个元素分别刻画了其中一个维度的属性，通过向量相似性计算来判断是否有重复的漏洞，并将其标识为冗余漏洞信息. 当用户访问 Web 漏洞信息时，只呈现单个漏洞的完整信息，并将跟其漏洞相似的 Web 漏洞来源作为一个条目展示.

这里每个维度的相似性计算方式不同：对于类别或发布时间等绝对信息，直接比对是否相同，相同置其相似值为 1，否则为 0；对于受影响对象等类型信息，计算其共同影响对象所占的比例作为相似值.

通过对现有 Web 漏洞数据对每个维度在相似度计算中所占比重对其权值进行矫正，并进行加和既得到两个向量的相似度. 当相似度大于阈值时，即判断两个漏洞是重复漏洞.

4) 增量式 Web 漏洞去重. 对于新增的 Web 漏洞，再进行第一、二步数据处理和分类后. 根据语义信息

进行预处理, 以减少相似度计算时间. 筛选 Web 漏洞库中与其有相同发布时间, 相同分类, 以及相同受影响对象的一组 Web 漏洞进行相似性计算. 若是已有相同漏洞信息, 则根据新增信息对现有漏洞库信息进行更新, 否则作为一个新的 Web 漏洞录入后台数据库中.

3.4 Web 漏洞重现

对可以重现或有攻击细节描述的 Web 漏洞, 在添加入 Web 漏洞库后, 通过自动和手动两种方式对受影响网站进行重现, 若可以完成攻击实现, 将攻击结果页面保存, 并录入文件数据库中, 作为漏洞镜像展示.

手动方式: 将漏洞所在 HTML 页面文档保存到本地的漏洞镜像数据库, 通过漏洞描述属性的 POC 代码, 对受影响网站页面进行渗透测试.

自动方式: 前端数据网站实现自动渗透测试模块, 维护不同类型 Web 漏洞的 cheat sheet, 即攻击利用代码, 对目标网站进行渗透测试.

若测试实现攻击, 则通过保存结果页面, 将其录入漏洞库后端的镜像文件数据中, 并在 Web 漏洞信息查询是作为漏洞信息条目内容呈现, 用户点击镜像可以看到问题页面被攻击的效果.

3.5 漏洞查询缓存加速

随着 Web 漏洞数据的越来越多, 用户 Web 漏洞库的数据访问将会严重消耗服务提供方资源并严重影响用户体验. 为大型 Web 系统增加缓存机制可以有效提高用户读取数据速率.

为了保证漏洞库的稳定和及时响应, 结合目前多种缓存加速技术[12], 我们构建了多个后端节点同步保存 Web 漏洞数据. 其中一个作为主节点, 其他两个作为辅助接点. 这三个接点的漏洞数据周期性同步, 以保证数据一致性.

对于 Web 漏洞查询等请求信息, 前端根据后端数据库链接数等负载信息, 进行负载均衡, 将请求分发到不同后端数据库上进行查询读取操作. 对于 Web 漏洞更新上交等请求, 前端将数据直接录入主节点, 并周期性将更新数据同步到其他两个从节点的数据库中. 一旦主节点因硬件或软件故障停止服务, 通过选举模式从从节点中选取一个新的节点作为主节点继续进行服务, 保证 Web 漏洞库 7*24 小时无间断工作.

4 结束语

本文所设计和实现的 Web 漏洞库基于传统漏洞库的模型, 根据 Web 漏洞特点对多个功能模块进行了重新设计并实现了一些新的功能. 漏洞库收录数据完全为 Web 漏洞, 更有针对性和全面性; 丰富了 Web 漏洞的收集方式和途径, 并通过工具规范了 Web 漏洞信息; 定性提出了自己的 Web 漏洞评级体系, 对 Web 漏洞的危害性定义更加准确; 实现了 Web 漏洞重现功能, 帮助用户从本地重现 Web 漏洞并对整个漏洞周期进行全面的分析和研究. 作为针对 Web 漏洞的漏洞库, 弥补了 Web 安全领域的缺失, 可以更好地为 Web 漏洞研究提供更翔实准确的数据. 随着个人隐私信息充斥互联网和大数据时代的到来, 对于个人隐私保护、防止数据泄露以及安全预警都可以提供更大的帮助.

参考文献

1 IBM X-Force Annual Trend and Risk Report 2013.2013.

2 张玉清,吴舒平,刘奇旭,等.国家安全漏洞库的设计与实现. 通信学报,2011,32(6):97–104.

3 National Vulnerability Database. http://nvd.nist.gov/.

4 WooYun.org. http://www.wooyun.org/.

5 中华人民共和国国家标准.安全漏洞标识与描述规范. 2011.

6 Common Vulnerability Scoring System. http://www.first.org/cvss.

7 Wang JA, Xia M, Zhang F. Metrics for information security vulnerabilities. Journal of Applied Global Research, 2008, 1(1): 48–58.

8 陈少飞,郝亚南,李天柱,等. Web 信息抽取技术研究进展.河北大学学报(自然科学版),2003,23(1):106–112.

9 W3C Recommendation. Document Object Model (DOM). http://www.w3.org/DOM/.

10 Brett D. McLaughlin. Locate specific sections of your XML documents with XPath, Part 1 [2008-06-10]. http://www.ibm.com/developerworks/cn/education/xml/x-introxpath1/index.html.

11 Brett D. McLaughlin. Locate specific sections of your XML documents with XPath,Part 2 [2008-06-17]. http://www.ibm.com/developerworks/cn/education/xml/x-introxpath2/index.html.

12 王科,周强,李春旺.Web 系统多级分布式缓存机制设计与实现.现代图书情报技术,2011,27(7/8):21–25.