文章编号:1001 - 9081(2006)02 - 0327 - 02

Web 页面防篡改及防重放机制

张建华[1,2],李 涛[1],张 楠[1,2]
(1. 四川大学 计算机学院,四川 成都 610065;
2. 西南民族大学 计算机科学与技术学院,四川 成都 610041)
(xnmdzjh@126.com)

摘 要:为避免 Web 页面被篡改或重放而导致的网站服务中止或形象损毁等严重后果,提出了一种内嵌于 Web 服务器的实时的防篡改及防重放机制,并给出了其实现方法。该机制可以有效防范页面被恶意篡改,还可以防范黑客重放木马脚本程序,提高了系统的安全性和可靠性。

关键词:Web 服务器;防篡改;防重放;中间件
中图分类号: TP309 **文献标识码**:A

Mechanism of anti-modification and anti-replacement on WebPages

ZHANG Jian-hua[1,2], LI Tao[1], ZHANG Nan[1,2]
(1. *School of Computer Science, Sichuan University, Chengdu Sichuan* 610065, *China*;
2. *School of Computer Science and Technology, Southwest University of Nationalities, Chengdu Sichuan* 610041, *China*)

Abstract: To avoid the abortion of Web services or contamination of entity images once WebPages have been modified or replaced deliberately, an efficient anti-modification and anti-replacement mechanism was presented in this paper. The mechanism works based on the research about Apache API. With the real-time WebPages identification middleware embedded in Web server, it can prevent existed pages from being modified and refuse illegal hobbyhorse program effectively. The application shows that it can improve the security and reliability of Web server.

Key words: Web server; anti-modification; anti-replacement; middleware

0 引言

当今国内外占统治地位的 Web 服务器,如 Apache、IIS 等,对用户请求的页面缺乏完整性保护机制,无法有效防止页面被篡改。这样,黑客可以在不知不觉中侵入系统,并修改关键页面。一般来说,主页的篡改对计算机系统本身不会产生直接的损失,但对电子政务与电子商务等需要与用户通过网站进行沟通的应用来说,就意味着电子政务或电子商务将被迫终止对外的服务。对政府网站而言,网页的篡改,尤其是含有政治攻击色彩的篡改,会对政府形象造成严重损害。最近又出现能自动篡改网页的“蠕虫”病毒,更加迫切需要安全的网页防篡改技术。

同时,很多 Web 服务器也缺乏对 Web 页面的防重放机制,对未知 Web 页面无法进行有效的鉴别。黑客侵入系统后未经授权就可放置木马脚本程序,给黑客非法入侵留下“后门”。

为此,有必要采取有效措施防止 Web 服务器上的 Web 页面文件被篡改或重放。通过对 Apache API 及模块的研究来实现相应的防篡改和防重放中间件,从而实现对 Web 服务器攻击的主动防御。

1 页面防篡改和防重放机制

利用 PKI 提供的数据完整性服务,基于 Apache 的模块扩展机制编写自定义中间件,对 Web 页面进行实时的原始性鉴别,防止 Web 页面在存储过程中被篡改,保证 Web 页面的完整性和权威性;同时还能检测未经允许放置的后台 Web 脚本程序并禁止其非法执行。管理员每次上传 Web 文件时,服务器采用严格的基于 PKI 的身份认证,确认管理员身份无误后,建立高强度 SSL 连接,保证上传文件的可靠性。系统接收文件后,先求出页面文件原文的数据摘要,然后对数据摘要用 Web 服务器的私钥和非对称加密算法对摘要进行加密得到数字签名。数据摘要提供了一种检测页面文件是否被改动过的方法,而数字签名则防止数据摘要本身被篡改。以后 Internet 用户每次请求访问页面时,服务器均通过内嵌的页面原始性鉴别中间件进行实时的页面检测,若原始性验证通过,则同意用户访问,否则进行相应处理(如拒绝访问并报警、恢复被篡改的页面等)。这样,即便不幸被黑客侵入系统、修改或重新放置 Web 页面后,服务器也不会将错误的页面误传给用户而造成不良后果。

具备防篡改和防重放机制的 Web 服务器成为增强型的安全 Web 服务器,其模块化体系结构如图 1 所示。

内核 (kernel)
标准模块(如 mod_auth、mod_rewrite等)
扩展模块(Web页面防篡改和防重放模块)

图 1 增强型安全 Web 服务器

2 页面防篡改和防重放的实现

2.1 页面签名

本文中使用的签名文件具有如图 2 所示的统一格式。

收稿日期:2005 - 08 - 04;**修订日期**:2005 - 10 - 22
基金项目:国家自然科学基金资助项目(60373110);教育部博士点基金资助项目(20030610003)
作者简介:张建华(1971 -),男,四川武胜人,副教授,博士研究生,主要研究方向:网络安全技术及应用; 李涛(1965 -),男,四川岳池人,教授,博士生导师,主要研究方向:计算机网络安全、人工智能; 张楠(1973 -),女,四川眉山人,副教授,博士研究生,主要研究方向:计算机网络.

其中每个域的具体含义如下：

签名标志符：其值可以自行定义，如 signedpages；

摘要算法标识符：可以使用目前国内外常用的摘要生成算法，如 MD5、SHA-1 等；

签名算法标识符：可以使用目前国内外常用的非对称加密算法，如：RSA、DSA、ECC 及 DH 等；

签名长度：数字签名内容的长度，此处固定为 128；

原始 Web 页面文件长度：未经签名前的原始页面文件长度；

数字签名：对原文及文件头使用服务器私钥进行数字签名后得到的签名内容；

原始 Web 页面文件内容：未经签名前的原始页面文件内容。

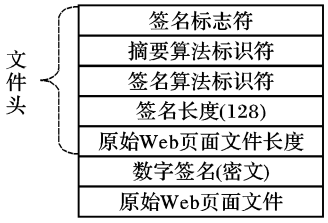


图 2 签名页面格式

具体的签名流程如图 3 所示。根据不同安全级别的要求和安全 Web 服务器性能的要求，用户可以使用不同强度的签名算法。

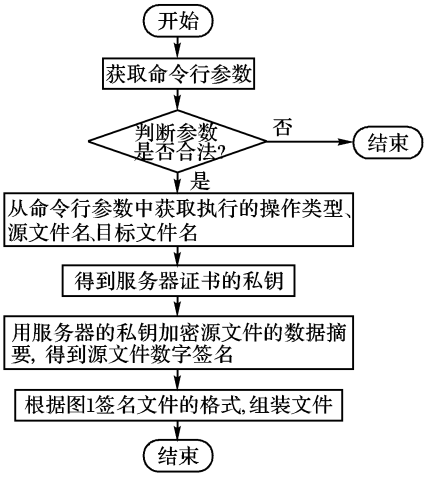


图 3 Web 页面签名流程

2.2 中间件对页面请求的处理

我们通过研究 Apache 对页面请求进行处理的详细过程，设计出 Apache 防篡改和防重放中间件，实现对已签名页面的实时访问控制。

Apache 对页面请求的处理用到了一个非常重要的请求对象结构 request_rec，它包含了请求的 URI、文件名、路径信息、参数、正文内容的类型和正文内容的译码方式等众多信息，其请求过程可分解为如下几个环节，如表 1 所示。

这些步骤通过搜寻一系列有“继承”关系的模块来进行处理。每一步骤可以对应一个处理程序，如果某一模块句柄匹配某一步骤的处理，就尝试调用该模块来进行处理。在模块中一般执行如下任务：

1）处理请求，如果成功执行，就返回信息 OK。

2）拒绝处理请求，并返回信息 DECLINED。这种情况下，服务器会认为对应的处理程序不存在。

3）通过返回一个 HTTP 错误代码发出出错信号，终止对客户端请求的处理。

表 1 页面请求过程

①	URI 到文件名的转换
②	头部解析处理
③	检查主机地址的访问权限
④	用户身份合法性检测
⑤	认证检查外的其他检查
⑥	判断请求对象的 MIME 类型
⑦	修正(处理扩展)
⑧	正式向客户端发送响应
⑨	对请求进行登记

Web 页面防篡改和防重放中间件是使用 Apache API 编写的一个 Apache 服务器扩展模块，它嵌入到 Apache 中运行，其处理被放在了 Apache 事务处理过程的头部解析处理阶段。用户通过浏览器请求已进行数字签名的 Web 页面时，由该模块在传输之前对该文件进行原始性鉴别，然后根据页面的当前状况以不同形式反馈给用户。其模块 signedornot_module 定义如下所示：

```
module MODULE_VAR_EXPORT signedornot_module = {
  STANDARD_MODULE_STUFF,
  NULL, NULL, NULL, NULL, NULL,
  NULL, NULL, NULL, NULL, NULL,
  NULL, NULL, NULL, NULL,
  signedornot_handler,                    // 防篡改和防重放 hook
  NULL, NULL, NULL
}
```

Apache 通过使用钩子函数来调用定制代码。钩子函数是一个在 Apache 上的注册函数，它在特定点被调用。对于不需要执行特定处理程序的钩子函数，模块声明中用 NULL 值表示，否则必须给出对应的处理函数名。在 Web 页面防篡改和防重放模块的模块声明时，就注册了用于防篡改和防重放处理的 signedornot_handler 钩子函数：

```
static int signedornot_handler (request_rec * req)
{...../* 具体的防篡改和防重放过程 */}
```

其执行过程如图 4 所示。

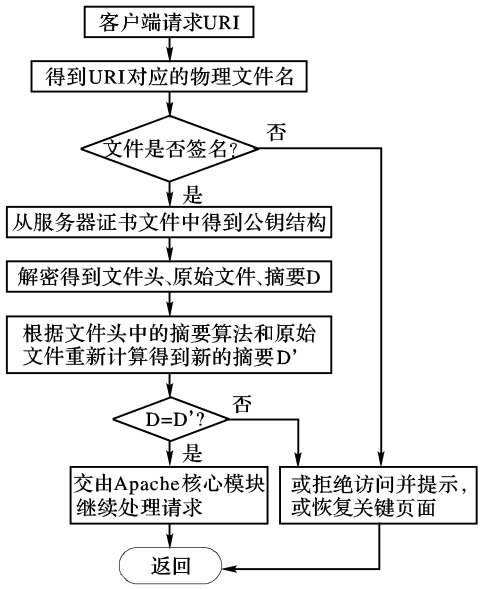


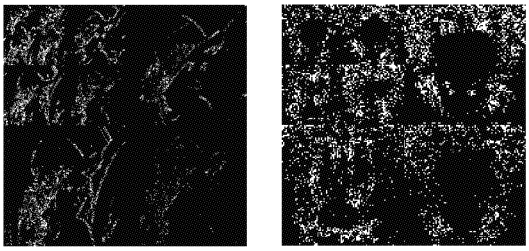
图 4 页面防篡改和防重放流程

3 结语

一般的 Web 页面防篡改系统采用的是外挂轮询式技术，即周期性地从外部逐个访问Web页面，来判断页面的原始

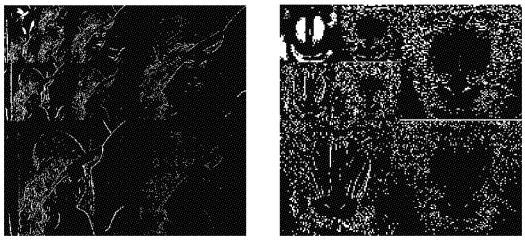
（下转第 331 页）

分解,在不同方向和层次的子带中对小波系数计算其梯度和纹理复杂度,结果如图 2 所示。



(a) Lena图像3层小波分解　(b) Baboon图像2层小波

图 2　小波域中的水印嵌入位置图



(a) Lena图像　(b) Baboon图像

图 3　小波域自适应算法中的水印嵌入位置图

(a) 结合视觉特性的粗集方法嵌入 PSNR=32.7297dB　(b) 传统小波自适应方法嵌入 PSNR=30.2983dB

(c) 结合视觉特性的粗集方法嵌入 PSNR=28.4158dB　(d) 传统小波自适应方法嵌入 PSNR=28.0745dB

图 4　水印嵌入效果对比

为了验证使用该方法选择水印嵌入位置的正确性,图 3 给出了传统小波域自适应水印算法的嵌入位置图。它是通过在小波分解后的子带中选取视觉临界误差JND值较大的系数得到的。为了体现对比的公正性,保证两种方法小波分解的层数和各个子带中嵌入点的总个数对应相等。

在小波域中应用本文设计的粗糙集水印算法自适应的嵌入随机 0,1 比特序列[6],并与传统的小波自适应水印算法进行比较,嵌入效果如图 4 所示。

对于相同的载体图像,小波分解的层数和对应子带中嵌入的水印信息是一样的,不同的是两种方法所选择的水印嵌入位置。经过对比发现,基于粗糙集的水印嵌入位置选择方法不但可行,而且嵌入水印以后的图像视觉质量较好。但是应该注意到,本文设计的方法只考虑了粗糙集与人类视觉特性的结合,影响水印鲁棒性的因素也可以作为图像知识系统的条件属性,参与到水印嵌入位置的选择方法中来。

3　结语

粗糙集理论作为一种新的软计算方法(其指导原则是利用所允许的不精确性、不确定性和部分真实性,以得到易于处理、鲁棒性和成本较低的解决方案),已经成为智能计算领域中一个新的学术热点。本文将粗糙集的思想应用到数字图像水印技术当中,提出了一种新颖的水印嵌入位置选择方法,并获得了较好的效果。该方法具有以下的特点:

1) 可以同时适用于空域和变换域的各种算法;

2) 能够通过调整各条件属性的重要性、依赖度和权重等来满足所设计算法的性能要求,使用灵活方便;

3) 具有很强的可扩展性,影响水印算法性能的各种因素可以转化为知识表达系统中的条件属性,而不必重新设计整个算法。

参考文献:

[1] 曾黄麟. 粗集理论及其应用[M]. 重庆: 重庆大学出版社, 1996.

[2] 曹泰钧. 粗集理论在数据处理中的研究与应用[J]. 河北理工学院学报, 2003, 25(4): 67 - 72.

[3] 徐立中. 数字图像的智能信息处理[M]. 北京: 国防工业出版社, 2001.

[4] SWINIARSKI RW, SKOWRON A. Rough set methods in feature selection and recognition[J]. Pattern Recognition Letters, 2003, 24(6): 833 - 849.

[5] PAWLAK Z. Rough Sets: Theoretical Aspects of Reasoning about Data[J]. Control Engineering Practice, 1996, 4(5): 741 - 742.

[6] LEWIS AS, KNOWLES G. Image compression using the 2-D wavelet transform[J]. IEEE Transactions on Image Processing, 1992, 1(2): 244 - 250.

(上接第 328 页)

性。对于每个 Web 页面来说,轮询扫描存在着较长的时间间隔。在这个时间间隔里,黑客可以攻击系统并使公众访问到被篡改的网页。本文提出的内嵌式 Web 页面防篡改和防重放中间件,运行于 Web 服务器内部,与 Web 服务器无缝结合。每次 Web 服务器对外发送 Web 页面时,都会进行页面防篡改和防重放检测,从而能够实时地监测每个被访 Web 页面的原始性,即便网站不幸被黑客入侵,也不用担心 Web 页面被恶意篡改,还可以防范黑客重新放置恶意页面或脚本等,提高系统的安全性。将其与页面加密存储访问技术等结合使用,更能积极主动地保障 Web 服务器的内部安全。

上面给出的 Web 页面防篡改和防重放机制已在研制出的多功能安全 Web 服务器中应用,有效增强了服务器的安全性和可靠性。

参考文献:

[1] THAU RS. Apache API notes[EB/OL]. http://modules.apache.org/doc/API.html, 2004 - 12 - 26.

[2] PAREKH S. Introduction to programming for the Apache API[EB/OL]. http://modules.apache.org/doc/Intro_API_Prog.html, 2005 - 04 - 13.

[3] KABIR MJ. Apache 服务器实用大全[M]. 北京: 水利水电出版社, 2001.

[4] HOLDEN G. Apache Server 源代码分析[M]. 北京: 机械工业出版社, 2000.

[5] 张建华, 李涛, 刘晓洁, 等. Web 页面加密存储及访问机制[J]. 计算机工程, 2004, 30(13): 97 - 98.