网络攻击与防范系列

电脑报 东方工作室

附光盘

脚本[漏洞]大爆光

JIAOBEN LOUDONG DABAOGUANG

耐特工作室
翟英 彭静 编著

- 脚本漏洞大曝光
- 黑客对脚本漏洞的攻击剖析
- 病毒对脚本漏洞的攻击剖析
- 脚本漏洞的防御和安全管理

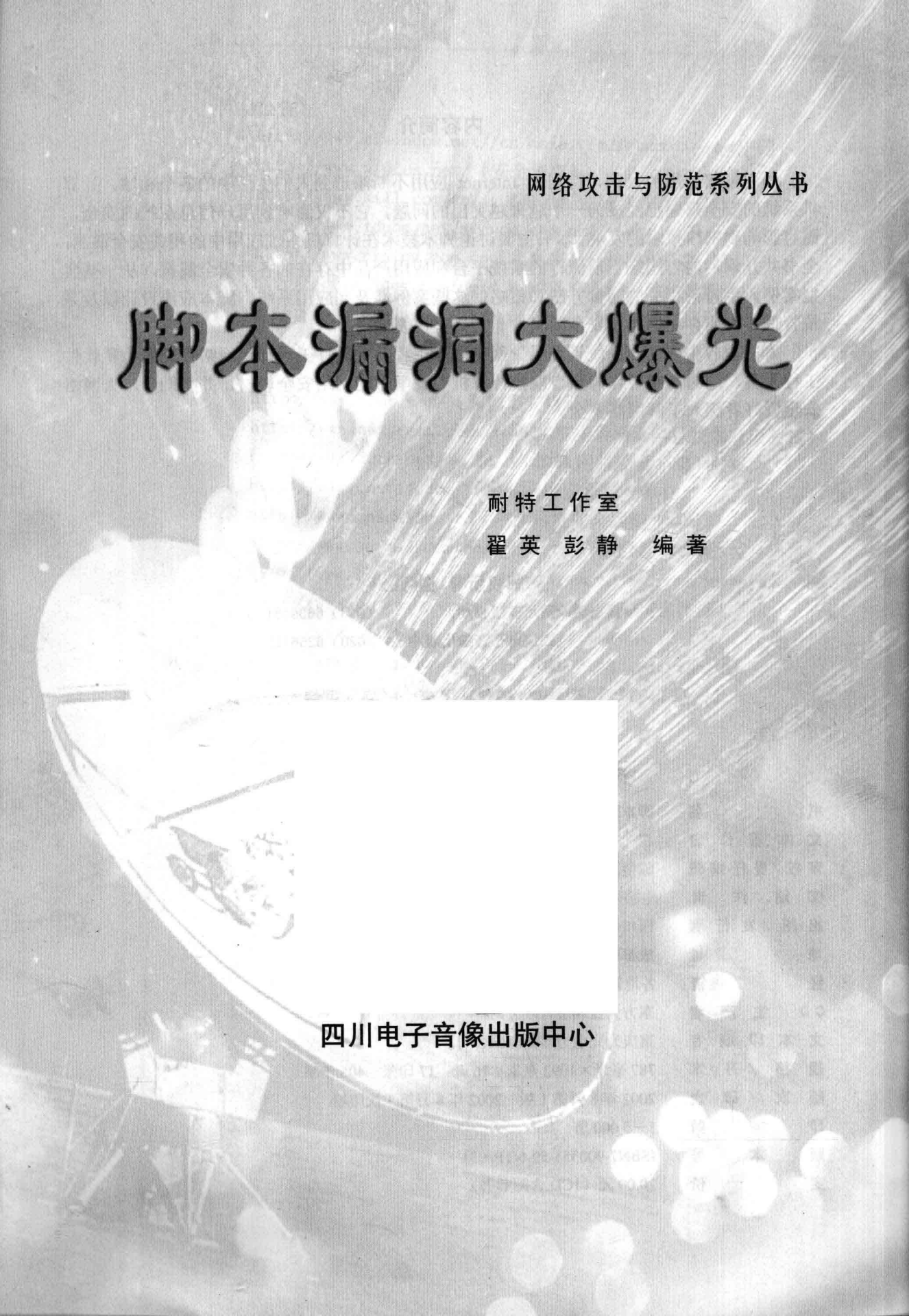
四川电子音像出版中心

网络攻击与防范系列丛书

脚本漏洞大爆光

耐特工作室

翟英 彭静 编著



四川电子音像出版中心

内容简介

随着网络应用的发展，特别是 Internet 应用不断渗透到人们生活中的各个领域，计算机系统的安全问题已经成为一个越来越突出的问题，它不仅影响到用户信息系统的安全，而且影响到网络技术的发展。本书主要讨论脚本技术在计算机系统应用中的相关安全漏洞，全书共九章，内容围绕当前流行的系统平台和应用产品中存在的各种安全漏洞，从一些典型案例入手讨论相关的防御方法和策略，这些案例涉及到应用系统、脚本应用设计以及系统管理等方面存在的典型安全漏洞，非常具有代表性。

本书内容主要面向系统管理员、网络安全工程师、软件工程师等网络管理人员和软件工程人员。同时，也可作为大专院校计算机专业学生的网络安全教材，并有助于广大网络安全爱好者深入了解网络安全。

耐特工作室　翟英、彭静　主编

书名	脚本漏洞大爆光
文本著作者	翟英　彭静
审校/责任编辑	陈学韶
CD制作者	电脑报东方工作室
出版/发行者	四川电子音像出版中心
地址	成都市桂花巷21号（610015）
经销	各地新华书店、软件连锁店
CD生产者	东方光盘制造有限公司
文本印刷者	重庆升光电力印务有限公司
规格/开本	787毫米×1092毫米　16开　17印张　405千字
版次/印次	2002年8月第1版　2002年8月第1次印刷
印数	1—5 000册
版本号	ISBN7-900355-59-6/TP·31
定价	20.00元（1CD,含配套书）

前　言

随着 Internet 应用不断渗透到我们生活的各个领域，网络安全已经成为一个重点研究的课题。黑客、病毒等对网络安全的威胁，也已经被越来越多的人所熟知。但是，更多的人只是知道一些名词，而对可能危及自身的威胁则浑然不知，这也是造成黑客、病毒能够大规模破坏的主要原因。

笔者长期以来一直从事网络建设以及网络技术的服务，接触了众多的用户和技术人员，最令我吃惊的莫过于是他们在安全知识上的匮乏和安全意识的淡泊。一个安全系统的实现，不仅仅取决于软件产品的设计，更多的要取决于用户对安全知识的理解和对产品的了解，取决于对应用系统的规划、部署和维护的能力。

在应用系统的设计、管理和维护中，脚本技术常常具有举足轻重的作用。脚本，最初是用于系统配置、管理的工具，是人控制、调度系统的操作的一种程序的实现。随着软件应用的发展，脚本已经成为应用系统中不可缺少的组成，它不仅可以将系统中各种各样的功能、组件、应用程序协调成一个整体，而且发展出一些独立的程序设计语言，成为应用系统设计的一个重要分支。然而，由于脚本在应用上的特殊地位，使它往往具有过于强大的能力，从而给安全带来新的隐患。

本书围绕脚本技术在应用中的安全问题进行了较全面的介绍。全书共分九章，对常见的脚本设计语言及其应用进行了介绍，剖析了脚本应用的安全漏洞及其产生的原因，以典型的案例说明了黑客、计算机病毒对脚本漏洞的利用和可能造成的危害，并针对应用系统如何实现安全设计、配置和管理进行了较详尽的介绍。

本书并不是一本黑客教程，也不是漏洞大全，我们写作此书的目的是希望给读者，特别是那些从事应用系统的设计、管理和维护的技术人员一些参考。

本书由翟英（MCSE+MCT）、彭静编写，刘晓辉（MCSE+CCNA）审稿。本书在写作过程中，得到王春海（MCSE）的大力协助，在此表示感谢。由于笔者技术的局限性，本书内容难免会存在一些纰漏和缺陷，请读者原谅，并欢迎批评指正。

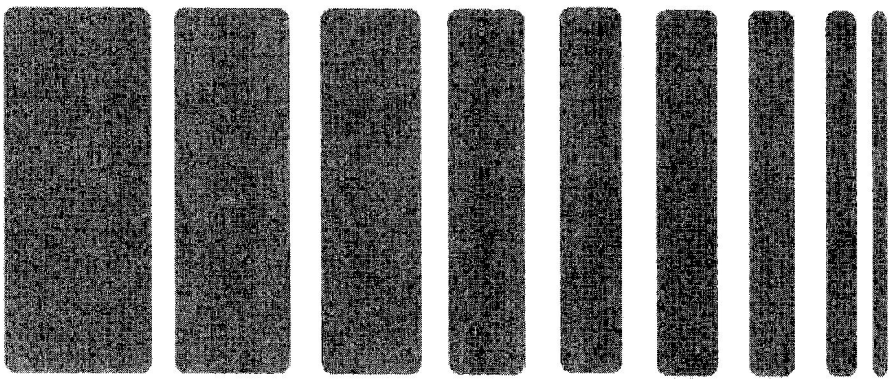
翟 英

2002.7

目录

1

脚本应用基础



什么是脚本？脚本有什么用呢？脚本编程语言与其他编程语言有什么区别和联系呢？这是初学者常常提到的问题。

1.1 什么是脚本

在已经有了许多优秀的编程语言的情况下，为什么还需要脚本语言呢？在解答这些问题之前，我们先来看一个例子：

```
REM AUTOEXEC.BAT
@ECHO OFF
set comspec=c:\command.com

set temp=c:\temp
PATH=C:\; C:\DOS

LH C:\DOS\SMARTDRV.EXE
LH c:\dos\MSCDEX.EXE /D:mscd001

CLS

ECHO ****************************
ECHO Welcome!  Today is:
DATE/T
TIME/T
ECHO ****************************
```

这不是一个批处理文件吗？是的，的确是一个批处理文件，它的运行结果如图 1-1 所示。这种文件在DOS系统中是司空见惯的，现在已经随DOS系统一起逐步被淡忘了。实际上，在没有 DOS 的 Windows 中它仍然存在，只不过我们已经很少用它了。

图 1-1　批处理文件示例

批处理文件是一个文本格式存储的文件，扩展名为“.bat”，可以用任何一个文本编辑器进行编辑，文件由系统可运行的程序或命令及控制其执行的语句组成，批处理文件

的内容由DOS系统或Windows系统中的command.com或cmd.com负责顺序解释执行。批处理文件的编写通常用于自动配置系统或应用程序环境、自动完成一系列任务。

那么，批处理文件和脚本有什么关系？批处理文件实际上就是脚本，只不过与现在我们常用的脚本相比，显得简单了一些。

在Windows3.1中，批处理文件的功能由宏所取代，作用是相似的。但对于更复杂的控制需求，批处理文件和宏都无法胜任，如对Microsoft Office的控制。因此，微软为其引入了一个新的脚本语言——VBA（Visual Basic for Application）。VBA运行于Microsoft Office环境中，采用Basic语言编程，有了自己的编辑器，但其程序的存储格式仍然是文本类型，程序的执行为解释执行而无需编译。

在现在流行的Windows 98、Windows 2000和Windows XP中，有一个更为强大的脚本环境——Microsoft Windows Script Host （WSH），它的编程语言想必大家已经很熟悉了，那就是VBScript和JScript语言，我们通常用它们来实现动态网页的程序设计，实际上它们的用途非常广泛，是高级系统管理员和软件设计人员手中的利器。

首先看一个JScript用于动态交互网页编程的例子。

```
<html>
<head>

<script language="JScript">
function WinOpen () {
   msg=open ("","DisplayWindow","toolbar=no,directories=no,menubar=no");
   msg.document.write ("<HEAD><TITLE> Hello! </TITLE></HEAD>");
   msg.document.write("<CENTER><H1>你好! </H1><h2>这是<B>JavaScript</B>所开的视窗!</h2></CENTER>");
}
</script>

</head>
<body>
<form>
<input type="button" name="Button1" value="Push me" onclick="WinOpen () ">
</form>
</body>
</html>
```

将上面代码保存为扩展名为“.htm”的文件，用Internet Explorer打开它（如图1-2所示），单击“Push me”按钮，可以打开一个新的窗口。

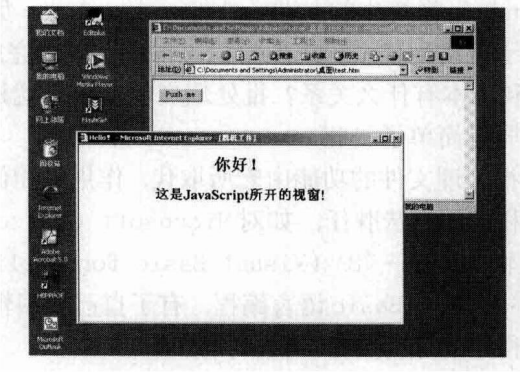


图 1-2　JScript 应用示例

下面，再看一个用VBScript脚本实现应用程序管理的例子。这个例子来自微软的教学课件。

```
' ************************************************************
' File:   AppActivateWSH2.vbs  (WSH 2 sample in VBScript)
' Author: (c) G. Born
'
' Launching Calculator and Notepad and using
' AppActivate to switch between applications
' ************************************************************
Option Explicit

' Define the title strings of the application windows.
' Important: Strings depend on the localized version of Windows.
Const Edit_Title = "Untitled - Notepad"   ' Window title
Const Calc_Title = "Calculator"           ' Window title

Dim Wsh, win_title

' Create the WshShell object, which Run and AppActivate require.
Set Wsh = WScript.CreateObject ("WScript.Shell")

' Try to launch two applications. To ensure that the last
' application receives the focus, delay the script.
Wsh.Run "Calc.exe", 1     ' Launch Calculator.
WScript.Sleep 800         ' Delay allows Calculator to get the focus.
Wsh.Run "Notepad.exe", 1 ' Launch Notepad.
WScript.Sleep 800         ' Delay allows Notepad to get the focus.
```

```
WScript.Echo "Click OK to set focus to the Calculator window"

' Set the focus back to the Calculator window.
Wsh.AppActivate Calc_Title

' Set the focus to the Notepad window.
WScript.Echo "Click OK to set the focus to Notepad"
Wsh.AppActivate Edit_Title

' Ask user for a window title and set the focus to that window.
win_title = InputBox  ("Please enter the window title", _
                       "Ask for window title", Calc_Title)
Wsh.AppActivate win_title

'**** End****
```

脚本运行结果如图1-3所示，这段代码演示了VBScript脚本控制应用程序运行的强大能力。

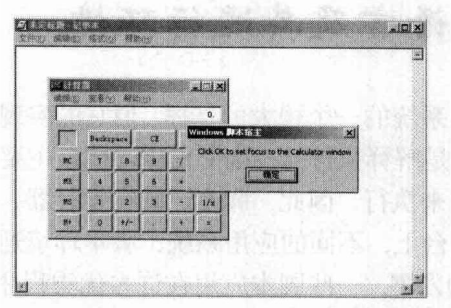


图1-3 VBScript应用示例

由以上两个脚本示例，可以初步体验到WSH环境下脚本的强大威力，作为微软系统平台的程序设计员和系统管理员来说，不懂得脚本的应用是称不上高手的。

以上都是从我们大多数人都熟悉的微软的系统平台上来讨论脚本，下面我们再来看看Unix系统中的脚本，它们的历史更悠久一些。

在Unix类系统中，管理系统通常在Shell上进行，使用Shell脚本可以自动配置系统和运行一系列任务。实际上前面所说的DOS下的command.com也是一个Shell，而批处理文件也是Shell脚本。Unix系统的Shell脚本通常可以用vi（一个Unix下最常用到的文本编辑器）来建立，Shell脚本可以和普通程序或命令一样由指定的Shell解释执行。下面是一个简单的例子。

```
#!  /bin/bash
echo  -n "Type some text;press ctrl+D when done"
echo  -n "Your input:"
```

```
while read; do
    TEXT='echo "$REPLY"|rev'
echo "Reversed,your message is:$TEXT"
echo -n "Your input:"
done
```

这个例子的代码与常用的编程语言是相似的，其中的系统命令可以分别单独顺序执行，但通过脚本编程可以使这一切自动化进行。Unix环境下的脚本应用还有很多，在后面我们会接触到。

从以上介绍可以看出，在不同系统上的脚本都有一些共同的特点：

- 一般采用解释方式执行，不需要编译
- 一个解释执行的脚本运行环境
- 由一系列系统命令和脚本控制语句组成
- 以文本文件的格式存储

脚本不仅自身可以独立完成一定的任务，并且如同粘合剂，将各种系统命令、应用程序集成到一起，而不管它们是用什么编程语言实现的，系统中的一切都可置于其控制之下。

脚本赋予系统管理员和程序设计员强大的控制能力，这也正是其魅力所在。

1.2 流行脚本语言及其运行环境

脚本的应用是对应用系统的一个强大的支撑，但它的实现同样也需要一个相应的支撑环境，由于脚本程序都是解释执行的，这个支撑环境最主要的作用就是解释脚本并把它的具体需要传递给系统来执行。因此，脚本、脚本解释器、操作系统和应用软件是紧密相关的。不同的系统平台上，不同的应用系统，脚本环境通常也就不同，通常也不兼容。随着Internet应用的发展，一些脚本应用在许多优秀程序员的努力下，实现了跨平台的应用，如Perl、JSP、PHP等。

1.2.1 Unix/Linux Shell 脚本

和DOS系统一样，Unix中也有一个类似的批处理环境，但更为强大。

什么是Shell？Shell就是一个居于核心和操作者之间的一层使用者接口。那么，为何称它为Shell呢？Shell的本意是“壳”的意思，在核心的外面包覆着一层外壳，用来负责接收使用者输入的指令，然后将指令解译成核心能够了解的方式，传给核心去执行，再将结果传回至预设的输出周边。DOS的command.com，Windows的GUI都是Shell。

Unix类系统中目前流行的Shell有：

- Bourne Shell
- C Shell
- Korn Shell
- tcsh (free)

● Bourne Again Shell——Bash GNU（其中，Bash 是Linux 的标准 Shell）。

Shell 是一个命令语言解释器（command-language interpreter）。拥有自己内建的 Shell 命令集。此外，Shell 也能被系统中其他有效的实用程序和应用程序所调用。键入任何一个命令，它都被Linux Shell所解释。Shell 首先检查命令是否是内部命令，如果不是，再检查是否是一个应用程序。

Shell 的另一个重要特性是它自身就是一个解释型的程序设计语言，Shell 程序设计语言支持在高级语言里所能见到的绝大多数过程控制结构，比如循环，函数，变量和数组。Shell 编程语言很易学，并且一旦掌握后它将成为你的得力工具。任何在提示符下能键入的命令也能放到一个可执行的 Shell 程序里，这意味着用 Shell 语言能简单地重复执行某一任务。

通常，Shell Script和其他的可执行程序一样是可以运行的，只不过Shell Script是以文本的方式储存，而非二进位代码。而执行 Shell Script 时，必须有一个解释程序将其内容转成一道道的命令执行，而这个解释程序就是Shell ，这也就是为什么我们叫做Shell Script 的原因。不同Shell 的Script基本上会有一些差异，所以我们不能将写给A shell 的Script用B shell 执行。在Unix中大家最常使用的Shell 为Bourne Shell 以及C Shell。

通常，在脚本文件的开头来指定所需要的 Shell。脚本开头以 "#!" 指定所使用的 Shell，而且要将整个路径名称指出来。如 Bourne Shell 的路径名称为 /bin/sh ，而 C Shell 则为 /bin/csh。

Script 的流程控制和一般高级语言的流程控制没有什么两样，这些使得 Script 的功能更加强大。为了达到与高级语言相同的效果，我们可以在Script 中设定变量，这样使 Script 成为一个名副其实的高级语言。

用文本编辑器编辑好你的脚本程序，加上可执行属性:

```
$chmod +x
```

就可以执行了。

Shell 脚本通常不需要单独安装，大多由操作系统提供，如 Bash、C Shell 等。

1.2.2 Perl 脚本

Perl 是Practical Extraction and Report Language 的缩写，是由 Larry Wall 设计并在 1987 年发布。Perl 最初是 Larry Wall 专门为 Unix 的系统管理员设计的一种脚本语言。

Unix 主要用 C 语言和 Unix Shell 来编程的，但它们却不是一个整体的两个部分。C 语言能很方便地进行系统内部的操作，但却不能快速地进行外部开发；Unix Shell 刚好相反，它能很方便地进行外部处理，却很难进行底层操作。如果能把这两者结合起来，不是很好吗？Larry Wall 认为这是他创建Perl 的初衷。因此，他把C语言以及 sed、awk、Unix Shell 等十多种工具和语言中非常“酷”的优点结合起来，把差劲儿的部分清除出去，开发出了 Perl。如果程序员觉得用 sed 或 awk 或 sh 完成工作有点慢，但又不想用 C 语言来写，Perl 就可以派上用场了。

经过十多年的发展，Perl已是最主要的Web脚本语言，大多数的CGI程序都是用它写的（正因为如此，它甚至成CGI程序的代名词）。同时，它也是一种非常好的快速原型设计语言，并且能使不同的系统很好地协同工作。由于Perl能非常方便地将小程序嵌入到大型应用程序之中，所以被誉为是“Internet的输送管道”。

Perl目前已经受到很多程序开发人员的喜爱，如系统管理员、数据库开发人员、Web开发人员等。现在，Perl不但成为系统管理员和CGI作者的宠儿，就连数学家、遗传学家、新闻工作者，甚至企业管理者也都喜欢用Perl。目前，全世界至少有100万以上的程序员在使用Perl来工作。

Perl的标准库及各种文档都是由自愿者写的，但它也有核心开发队伍，即PerlPorters。这些成员都是无私奉献的，他们的任务也很简单，就是开发出大量的比市场上所出售的Perl应用程序更好的产品，而且是免费的。

Perl越来越得到大家的认可并非偶然，这是由于它具有很多其他语言所没有的优点。首先，用Perl编写的程序不用编译，其程序可直接运行。最重要的是，它可以跨平台运行。同一Perl程序可以在Unix、Windows、Windows NT、MVS、VMS、DOS、Macintosh、OS／2、Plan9及AmigaOS等操作系统上自由运行。其次，Perl简单易学。它可以使复杂的事情变简单，简单的事情更简单。Perl为程序员处理了很多东西，例如内存分配、碎片整理等。同时，Perl程序也是相当简练的，一页的Perl程序用其他语言可能要花几百甚至上千行代码。Perl程序不必处理很多细节，这大大减少了程序的Bug，增加了可靠性。第三，Perl运行速度很快。第四，Perl是面向对象的。第五，Perl应用程序很多。CPAN（Comprehensive Perl Archive Network）有很多Perl应用程序。你可以在几分钟内找到自己想要的东西。这些程序都是来自世界各地的Perl开发者提供的。最后，Perl是自由软件，为程序员免费共享。

Perl脚本语言具有以下特点：

- Perl具有高级语言（如C）的强大能力和灵活性。事实上，你将看到，它的许多特是从C语言中借用来的。
- 与脚本语言一样，Perl不需要编译器和链接器来运行代码，你要做的只是写出程序并告诉Perl来运行而已。这意味着Perl对于小的编程问题的快速解决方案和为大型事件创建原型来测试潜在的解决方案是十分理想的。
- Perl提供脚本语言（如sed和awk）的所有功能，还具有它们所不具备的很多功能。Perl还支持sed到Perl及awd到Perl的翻译器。
- Perl象C一样强大，象awk、sed等脚本描述语言一样方便。

Perl语言的出色之处在于，“Perl语言在文本处理方面非常突出，它把不同的内容联成一个整体。对于这种脚本语言来说所有的那些不同的元素，看起来都是一样的。”（John Ousterhout，Tcl脚本语言的作者的评语）

Perl被广泛的认为是“一种拥有各种语言功能的梦幻脚本语言”、“Unix中的王牌工具”以及其他的一些类似的称呼，Perl被用来写单行脚本，快速执行程序，大的规划项目（如Amazon.com的所有评论产品和控制系统，Netscape的内容策划管理和传送系统，人类整组基因工程的DNA排序以及计划管理等等），还有数以百万计的令我们惊讶的各种各样的事情的高速程序。Perl还能够实现许多Unix的系统工具的功能。

就像所有的现代语言所期望的那样，Perl允许你建立面向对象的程序。它也可以进行网络操作，并且有良好的可移植性（一个写得好的脚本可以在Linux，BSD，Solaris，Dos，Win9x，NT，MacOS，OS/2，AnugaOS，VMS等操作系统中不需要任何修改的运行），编写和调试周期很短—由于没有编译的要求，你只需将变化的部分写出，就可以运行脚本。还有数目庞大的可适用于执行任何一项任务的模块（即预建立Perl的例程），Comprehensive Perl Archiue Network （CPAN）就是每一个Perl程序员所能拥有的最好的模块库之一。

一般情况下，用Perl写的CGI脚本是一种“非嵌入式”的服务器端脚本，因为它是一个单独的程序，而不是嵌在HTML文档中再通过另一个程序解释替换。例如下面的Perl程序：

```
print "Content-type:text/html\n\n";
print <<HeadofHTML;
<HTML>
<HEAD>
<TITLE>Hello!</TITLE>
</HEAD>
<BODY>
HeadofHTML
print "<CENTER>aaa</CENTER>";
print "</BODY></HTML>";
```

不过，可嵌入式的CGI脚本程序已经开发成功，它就是ePerl，国内用得很少。

1988年1月，Perl的第一个正式版本Perl1发布，最新的版本是Perl5.6。

1.2.3 VBScript与JavaScript脚本

当前在网页编程中有两种脚本语言最为流行，即VBScript和JavaScript，它们分别由微软公司和网景（Netscape）公司推出。其中，JavaScript的名字容易使人产生误解，很多人认为JavaScript就是简化了的Java。Java是由SUN公司推出的一种语言，JavaScript的原名是LiveScript，在最初是完全由Netscape公司独立开发的，用于其一系列的Web应用产品。在Java出现以后，Java迅速地占据了服务器端编程的领导位置，这时急需一种客户端脚本语言，LiveScript正好可以满足这样的要求，JavaScript通常作为Java语言的脚本工具。

VBScript和JavaScript主要应用在微软的系统平台上，运行环境为Microsoft Windows Script Host（WSH）。VBScript和JavaScript对于普通用户来说，它们在ASP中的应用是最熟悉的。其实，VBScript和JavaScript不仅应用在基于Web的应用上，在微软的系统平台上它无处不在。

WSH运行的系统环境已经集成在Windows Me、Windows 2000和Windows XP产品中，其他Windows产品如Windows 95、Windows 98和Windows NT需要下载安装WSH。

Microsoft Windows Script Host （WSH）是一个功能强大的脚本应用环境，如图