基于 TrackRay 的渗透测试平台设计

范 晶,焦运良,戴贻康
(华北计算机系统工程研究所,北京 100083)

摘 要:如何维护网络信息安全一直是 IT 行业重点关注的问题。渗透测试是一种常见的网络安全评估方法,由于渗透测试所采用的安全工具过于繁杂,因此设计一个基于 TrackRay 的渗透测试平台,内置有漏洞扫描器和 Web 服务接口,并集成和综合多种安全工具的优点,功能强大,简便易用。特别是支持 Java、Python、JSON 等方式编写插件,调用各种类型插件来进行渗透测试,可移植性大大提高。经试验结果表明,该渗透测试平台搭建简单方便,可运用于 Windows 和 Linux 等系统,并且可以进行灵活的编写插件实现快速的 Web 安全漏洞检测。

关键词:信息安全;渗透测试;TrackRay;可移植性
中图分类号:TP309 文献标识码:A DOI:10.19358/j.issn.2096-5133.2020.06.007
引用格式:范晶,焦运良,戴贻康. 基于 TrackRay 的渗透测试平台设计[J].信息技术与网络安全,2020,39(6):38-43.

Design of penetration test platform based on TrackRay

Fan Jing, Jiao Yunliang, Dai Yikang
(National Computer System Engineering Research Institute of China, Beijing 100083, China)

Abstract: Today, how to ensure information security is an important problem in the Internet. Penetration test is a common network security assessment method. Because using different security tools are too complicated, this paper designs a penetration testing platform based on TrackRay, with built-in vulnerability scanner and web service interface. It also integrates the advantages of various security tools, which makes it powerful and easy to use. In particular, the framework supports Java, Python, JSON and other ways to write plug-ins and calling various types of plug-ins to take penetration testing. And its portability is greatly improved. From the experimental results, it shows that the penetration testing platform designed in this paper is simple and convenient to build, can be used in Windows and Linux systems, and can be flexible to write plug-ins to achieve rapid detection of web security vulnerabilities.

Key words: information security; penetration testing; TrackRay; portability

0 引言

互联网中 Web 应用所产生的漏洞数量越来越多,使得每年发生的黑客攻击案件也在逐年增长[1]。为了保障网络或系统的信息安全,在系统投入应用之前都会对其进行渗透测试。渗透测试模拟黑客或者入侵者的攻击行为,绕过或破坏目标系统的安全防护并获得一定的系统权限,在此过程中检测系统漏洞或病毒并给出修复建议,从而分析与评估系统的安全等级。通常使用的渗透测试工具有 WVS、AppScan、Nessus 等,这些工具测试时存在自动化水平低、测试流量太大以及漏洞发现率较低等缺点。基于此,本文设计一个基于 TrackRay 的渗透测试平台,功能强大,便于搭建,并且可以灵活地编写插件快速实现 Web 安全漏洞检测。

1 Web 服务器与漏洞

1.1 Web 服务器

Web 服务器概念较为广泛,一般是指网站服务器,运行于计算机上的服务程序,可以向 Web 客户端(如浏览器)提供文档或者信息浏览服务[2]。Web 服务器主要是以 HTTP 协议(请求-响应协议)进行通信,所以也被称为 HTTP 服务器[3]。

Web 服务器的工作过程主要分为建立连接、发送请求、应答和关闭连接四个步骤。客户端(包括浏览器和其他应用程序)通过 TCP/IP 协议(传输控制协议/网络协议)建立和服务器的 TCP 连接,通过打开虚拟文件 socket 可以确认是否连接成功;当成功

建立起连接后，客户端向服务器发送基于 HTTP 协议的用户请求；服务器接收到请求后，经过查询与处理，向客户机发送基于 HTTP 协议的回应，如果请求的网络地址中包含有动态语言(如 ASP，PHP，JSP 等语言)的内容，那么服务器端将会使用相应动态语言的解释引擎处理内容，并将处理后的数据发送给客户端，客户端接收到服务器发回的数据后将信息显示在窗口中；最后，当发出的响应结束后，Web 服务器便与客户端断开，关闭连接。具体的工作原理图如图 1 所示。

1.2 漏洞

一般来说，如果在一个信息系统中存在某种问题，这种问题可能存在于设计、实现、运维等过程中，使得系统发生某种安全风险，如对系统的机密性、完整性、可用性产生泄露和破坏，那么此问题就可以统称为漏洞[4]。从目前存在的系统漏洞来看，操作系统漏洞是远远少于应用程序漏洞的，绝大多数的漏洞类型属于 Web 服务器中的应用程序漏洞[5]。其中 Web 应用程序漏洞又可以划分为 SQL 注入漏洞、文件上传漏洞、跨站脚本攻击等。

1.2.1 SQL 注入漏洞

SQL(结构化查询语言)注入攻击是黑客针对数据库进行攻击的常见方式，通过在 Web 表单或 Web 链接中插入 SQL 查询语句，欺骗服务器执行攻击者所插入的 SQL 查询语句[6]。具体来说，SQL 注入攻击利用服务器中运行的 Web 应用程序，将攻击者编写的 SQL 查询语句提交到到网站后台执行，然后返回给前台页面被 SQL 注入攻击的查询结果。如果网站存在 SQL 注入漏洞，攻击者就可以通过输入恶意 SQL 查询语句改变原先 SQL 查询语句的结构，从而改变用户真正所需的查询结果。

有两种情况可以产生 SQL 注入漏洞，一种是平台层的配置错误或数据库本身存在漏洞所导致的，还有一种是由于程序编写者对输入未进行过滤，或者在拼接 SQL 注入字符串时没有采用固定的结构，从而执行了非法的数据查询。因此，可以采用预编译语句固定 SQL 语句结构，预先对外部输入进行检测、过滤等操作，以确保数据库引擎所执行的逻辑是未被篡改的。

1.2.2 文件上传漏洞

在现代的 Web 应用程序中，为了提高业务效率，共享文件，Web 应用程序一般都拥有允许用户上传文件的入口，虽然这种入口为用户提供了很大的便利，但也给恶意用户打开一扇门。Web 应用程序中这种上传入口越多，服务器所面临的风险也就越大。如果 Web 程序对上传的文件没有进行仔细的过滤和转化，恶意用户就有可能上传对服务器有危害的脚本文件，执行违法操作，如获得服务器目录和文件、获得服务器数据库访问权限、对企业内网进行扫描操作等[7]。这些操作将会导致整个网站瘫痪和安全信息泄露，使用户遭受严重的损失。

造成文件上传漏洞的原因有多种情况，包括未对上传文件的后缀名(扩展名)做严格检查与限制；未对上传文件的用户权限进行限制；以及 Web 服务器对于上传文件或者指定目录的行为未做一定的限制等。因此，对上传文件的扩展名的合法性进行检查，分析文件内容、检测文件类型的合法性和限制 Web 服务器对于特定类型文件的操作，可以减少发生文件上传漏洞的可能性。

1.2.3 跨站脚本攻击

跨站脚本攻击(Cross Site Scripting，XSS)，是指攻击者利用网站漏洞恶意盗取用户私人信息的行为。

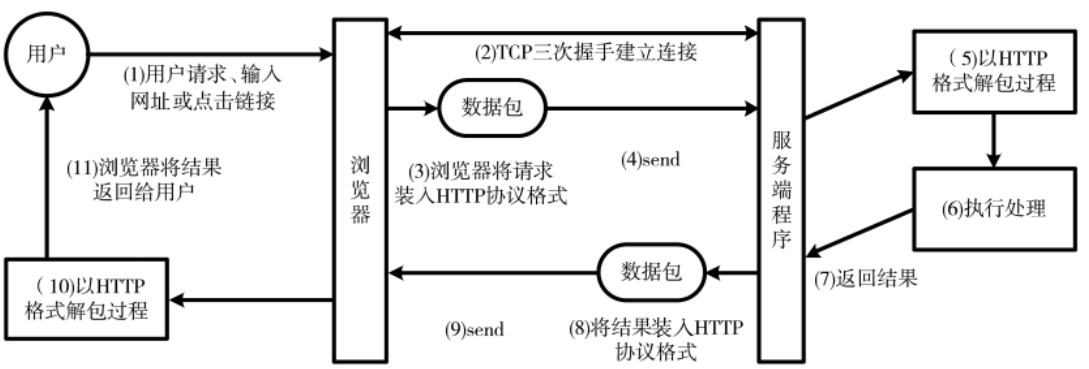

图 1 Web 服务器工作原理

攻击者一般通过 XSS 漏洞将恶意脚本代码注入到网站页面链接中，并很好地伪装成合法链接，当用户不小心点开链接后，信息便被泄露出去[8]。XSS 漏洞曾在安全领域中未受到足够的重视，认为其危害小可以忽略，但是随着 Web 应用的普及和复杂性的提升，以及在拥有大量用户的企业或网站受到越来越多的 XSS 攻击之后，XSS 漏洞也被重视起来。攻击者可以通过 XSS 漏洞规避网站的访问控制权限，从而控制用户在浏览器上的行为，记录和读取用户的网站密码等私人信息。

跨站脚本攻击主要分为持久性跨站脚本攻击和非持久性跨站脚本攻击。持久性跨站脚本攻击所依赖的数据会存储在服务器数据库或文件中，在数据被清除或漏洞被修复之前会一直存在。而非持久性跨站脚本攻击是一次性的，仅在当前访问的链接或页面中发挥作用，如果页面被关闭或者链接跳转，那么这次攻击的数据就会消失。防御 XSS 的关键就是在输入不可信数据的时候进行合法编码，对外部数据进行检查、过滤等操作，并且考虑到不同浏览器之间的兼容情况应设置不同的过滤规则；在将来自外部的数据插入到网站页面中时，对外部数据进行超文本标记语言实体编码，以规避不合法的脚本在浏览器中解析运行。

2 渗透测试平台设计

TrackRay 是一个开源的插件化渗透测试平台，集成有 Metasploit、Nmap、SQLmap、AWVS 等知名安全工具，并内置漏洞扫描器，可实现安全漏洞扫描功能。主体使用 Java 编写，支持 Java、Python、JSON 等多种方式编写插件，从而可以方便灵活地根据自己的需求来编写和调用不同类型的插件，实现快速安全漏洞检测。

2.1 平台搭建

首先安装 TrackRay 依赖环境，JDK1.8 和 Maven 是必须安装的，保证系统能够执行 java-version 和 mvn-verson。同时安装 Metasploit、AWVS、Python 2.7和 Nmap 等安全检测工具，以实现 Web 安全漏洞检测。

整个平台的搭建步骤如下所示。

(1)手动启动 AWVS 服务，登录后台生成一个安全的 API 密码。复制此密钥和 AWVS 地址，以便进行密钥和地址修改。

(2) 找到 Python 的数据库目录，修改 python.package.path 参数，如图 2 所示。

```
python.package.path=D:/Python/Lib/site-packages
```

图 2 修改 Python 参数

(3)安装 Maven 后找到仓库位置，一般在 settings.xml 里配置指定仓库目录，并在当前目录下生成一个.m2 的目录，并修改相应参数，如图 3 所示。

```
maven.repository.path=/root/.m2/repository
```

图 3 修改 Maven 参数

(4)Ceye.io 下注册一个账号，记录下分配到的域名和 TOKEN，并修改配置文件，这是为 DNSLOG 回显检测漏洞时使用的，注册页面如图 4 所示。

图 4 注册账号

(5)启动 msf 和 sqlmapapi 控制台函数，以便进行插件调用。在相应的 metasploit 和 sqlmap 目录下分别执行，成功执行后修改配置文件，如图 5 所示。

```
#msfrpc
metasploit.host=127.0.0.1:55553
metasploit.user=msf
metasploit.pass=msf
```

```
#sqlmap api
sqlmap.root=http://127.0.0.1:8775/
```

图 5 配置文件修改

(6)编译打包程序，不同系统下的执行命令不一样：Windows 下运行 package.bat；Linux 下运行 package.sh。等待下载完成和程序编译完成，若是此过程中未出现问题则会提示 BUILD SUCESS，如图 6 所示。

(7)编译成功后在当前目录下打包一个 trackray.jar 程序，也就是 TrackRay 的主程序，接着直接执行 start.bat 或者 startup.sh，就可以启动该项服务。没有

```
[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ web ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ web ---
[INFO] No tests to run.
[INFO]
[INFO] --- maven-jar-plugin:3.1.0:jar (default-jar) @ web ---
[INFO] Building jar: /root/trackray/trackray-framework/web/target/web-2.0.0.jar
[INFO]
[INFO] --- spring-boot-maven-plugin:2.1.1.RELEASE:repackage (repackage) @ web ---
[INFO] Replacing main artifact with repackaged archive
[INFO] ------------------------------------------------------------------------
[INFO] Reactor Summary for trackray framework 2.0.0:
[INFO]
[INFO] trackray framework ................................. SUCCESS [  0.452 s]
[INFO] base ............................................... SUCCESS [ 14.419 s]
[INFO] module ............................................. SUCCESS [  3.223 s]
[INFO] web ................................................ SUCCESS [  6.571 s]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
```

图 6　编译成功

抛出异常或 ERROR 日志，则表示访问 8080 端口正常。服务列表如图 7 所示。

至此，整个平台搭建完成。接下来通过调用服务或者调用相应编写好的插件，便可以实现 Web 安全漏洞检测。本平台支持多种语言编写插件，并且插件类型丰富，可根据自己的需要灵活编写。

2.2 插件编写与调用

本平台支持多种插件类型，包括：爬虫插件、内部插件、独立应用插件、漏洞扫描插件、交互式插件等。其中漏洞扫描插件是本文研究的重点。

在创建扫描任务时，勾选“漏洞攻击模块”，便可以调用漏洞扫描插件，主要有独立插件、漏洞规则和 kunpeng JSON 插件三种。

独立插件可以根据自己的需要进行编写，给定 requests、target 和 task 参数，再调用 check() 和 attack() 等抽象方法重写漏洞插件规则，以检测出漏洞。本平台采用简单的判断响应体就能够识别出大多数漏洞，因而相较于其他检测工具的方法效率更高。kunpeng JSON 是一个 Go 语言编写的 poc 测试平台，只需按照该插件的格式规范创建一个 JSON 文件到/resources/json 目录下，便可以进行插件的调用。

在进行插件调用时，本平台提供一个简单友好的 MSF 控制台界面，只需简单地输入命令行，便可以有效地执行命令，调用插件，以及直观地观察执行后的结果。MSF 控制台如图 8 所示。

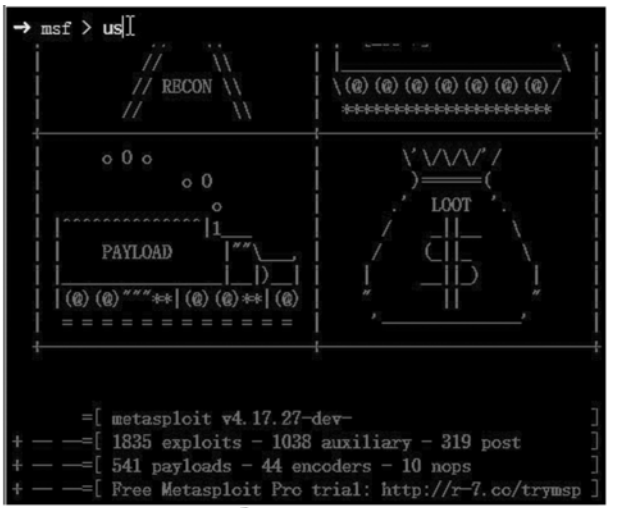

图 8　MSF 控制台

在图 8 中的首行 msf>>后面输入下面的第一行插件服务命令后，平台便开始调用和执行相应的插件，并将结果直接反馈回控制台，运行的结果如图 9 所示，可以直观形象地观察到检测结果。

3 渗透试验测试

对一台主机的端口和 Web 浏览器作为测试对象，提供该主机的 IP 或域名作为测试对象输入，图 10 是本平台进行渗透测试实验的流程图。

首先进行 TrackRay 的系统平台环境搭建，即设置各种参数；接着输入目标的 IP 或者域名，即输入测试对象；对此 IP 或域名进行端口扫描操作，扫描操作默认为收集目标的详细信息，包括各个端口的开放情况、开放的端口所使用的服务以及服务版本、目标主机所运行的操作系统类型以及可能的版

```
2019-05-11 22:28:19.503 INFO 2491 --- [   Thread-3] com.trackray.base.utilsls.Syslog             : com.trackray.
base.handle.SystemInit.chock(57)连接到SQLMAP API已开启
2019-05-11 22:28:20.095 INFO 2491 --- [   Thread-3] com.trackray.base.utilsls.Syslog             : com.trackray.
base.handle.SystemInit.check(65)metasploit 服务正常
2019-05-11 22:28:20.279 INFO 2491 --- [       main] o.s.s.concurrent.ThreadPoolTaskExecutor      : Initializing
ExecutorService "applicationTaskExecutor"
2019-05-11 22:28:20.382 WARN 2491 --- [       main] aWebConfiguration$JpaWebMvcConfiguration     : spring.jpa.op
en-in-view is enabled by default. Therefore. database queries may be performed during view rendering. Explicitly
configure spring.jpa.open-in-view to disable this warning
2019-05-11 22:28:20.603 INFO 2491 --- [       main] o.s.s.a.w.s.WelcomePageHandlerMapping        : Adding welcom
e page template:index
2019-05-11 22:28:21.434 INFO 2491 --- [       main] o.s.s.quartz.SchedulerFactoryBean            : Starting Quartz
Scheduler now
2019-05-11 22:28:21.436 INFO 2491 --- [       main] org.quartz.core.QuartzScheduler              : Scheduler sch
edulerFactoryBean_$_NOW_CLUSTERED started.
2019-05-11 22:28:21.502 INFO 2491 --- [       main] o.s.b.w.embedded.tomcat.TomcatWebServer      : Tomcat starte
d on port(s):8080 (http) with context path ""
2019-05-11 22:28:21.506 INFO 2491 --- [       main] com.trackray.WebApplication                  : Started WebAp
plication in 13.679 seconds (JVM running for 14.438)
```

图 7　服务列表

```
msf > search structs
Matching Modules
================

  Name                                                  Disclosure Date  Rank       Check  Description
  ----                                                  ---------------  ----       -----  -----------
  exploit/multi/http/struts2_code_exec_showcase         2017-07-07       excellent  Yes    Apache Struts 2 Struts 1 Plugin Showcase OGNL Code Execution
  exploit/multi/http/struts2_content_type_ognl          2017-03-07       excellent  Yes    Apache Struts Jakarta Multipart Parser OGNL Injection
  exploit/multi/http/struts2_namespace_ognl             2017-08-22       excellent  Yes    Apache Struts 2 Namespace Redirect OGNL Injection
  exploit/multi/http/struts2_rest_xstream               2017-09-05       excellent  Yes    Apache Struts 2 REST Plugin XStream RCE
  exploit/multi/http/struts2_code_exec                  2010-07-13       good       No     Apache Struts Remote Command Execution
  exploit/multi/http/struts2_code_exec_classloader      2014-03-06       manual     No     Apache Struts ClassLoader Manipulation Remote Code Execution
  exploit/multi/http/struts2_code_exec_exception_delegator 2012-01-06    excellent  No     Apache Struts Remote Command Execution
  exploit/multi/http/struts2_code_exec_parameters       2011-10-01       excellent  Yes    Apache Struts ParametersInterceptor Remote Code Execution
  exploit/multi/http/struts2_default_action_mapper      2013-07-02       excellent  Yes    Apache Struts 2 DefaultActionMapper Prefixes OGNL Code Execution
  exploit/multi/http/struts2_dev_mode                   2012-01-06       excellent  Yes    Apache Struts 2 Developer Mode OGNL Execution
  exploit/multi/http/struts2_dmi_exec                   2016-04-27       excellent  Yes    Apache Struts Dynamic Method Invocation Remote Code Execution
  exploit/multi/http/struts2_dmi_rest_exec              2016-06-01       excellent  Yes    Apache Struts REST Plugin With Dynamic Method Invocation Remotes Code Execution
  exploit/multi/http/struts2_include_paramas            2013-05-24       great      Yes    Apache Struts includeParams Remote Code Execution
```

图 9　MSF 执行结果

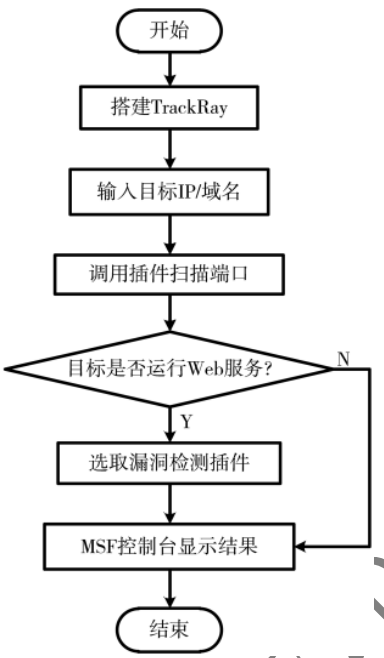

图 10　试验测试流程图

本。在扫描结束之后，此模块会将扫描结果存储在一个 XML 文件中，以供后续步骤的分析。在判断调用插件的规则后，若目标运行了 Web 服务，便进行漏洞检测，同样是只需在 MSF 控制台输入相应的调用命令，便可以记住相应的插件操作。同时将漏洞检测结果以报告的形式直观地显示出来。若是目标并没有运行 Web 服务，则跳过此漏洞检测步骤而报告端口的扫描结果。

为了测试本文基于 TrackRay 的渗透测试平台对于检测 Web 服务网站的安全漏洞效果，搭建起一个带有漏洞的网站站点(192.168.1.21)，使用较为流行的开源博客系统 WordPress 搭建，在搭建过程中，放入两个漏洞弱点，其一是后台弱口令弱点，另一个弱点是文件内容读取漏洞。使用 Nessus 扫描工具和本文设计的渗透测试平台对网站的安全性进行验证，实验结果如图 11 所示。

Sev	Name	Family	Count
CRITICAL	PHP Unsupported Version	CGI abuses	1
HIGH	Apache 2.2x < 2.2.33-dev/2.4.x < 2.4.26 Multiple Vulnerabilities	Web Servers	1
HIGH	Apache 2.4x < 2.4.25 Multiple Vulnerabilities(httpoxy)	Web Servers	1
HIGH	Apache 2.4x < 2.4.27 Multiple Vulnerabilities	Web Servers	1
HIGH	OpenSSL 1.0.2 < 1.0.2 Multiple Vulnerabilities	Web Servers	1
MEDIUM	SSL Certificate Cannot Be Trusted	General	3
MEDIUM	SSL Self-Signed Certificate	General	2
MEDIUM	Apache 2.4x < 2.4.26 HTTP Vulnerability(OptionsBleed)	Web Servers	1

图 11　Nessus 扫描结果

首先搭建起 Nessus 的扫描环境，虽然花费的时间并不是很久(半小时左右)，但是从 Nessus 的扫描结果中可以看出：Nessus 只报告了关于中间版本和 PHP 版本的漏洞，而预先放置的两个漏洞都没有检测出来。

图 12 是对本文设计的渗透测试平台进行测试试验得出的结果。

```
INPUT THE TARGET:http://192.168.1.21
starting scan (20:29:38)
finding web fingerprint ...
Wordpress
jQuery
OpenSSL
Windows
starting port scan ...
80 opened Apache/2.4.23 OpenSSL/1.0.2 PHP/5.4.45
3306 opened mysql
verifying...
poc_2014_0216.py wordpress plugin get-file-content vul
verify over (20:29:50)
```

图 12　基于 TrackRay 的渗透测试平台的测试结果

首先搭建本文的渗透测试平台，只需二十分钟左右便可以搭建完成，而进行整个试验的运行时间也是在十分钟以内，故相对而言本文的渗透测试平台简单易用。再看检测结果可知：使用

poc_2014_0216 验证子程序对目标系统进行漏洞验证，最终检测出弱口令和文件内容读取漏洞，优于 Nessus 扫描工具检测结果。

4 结论

Web 网页或者信息系统的安全漏洞的验证与修复一直是网络安全研究中的重要问题。本文重点介绍如何搭建起基于 TrackRay 的渗透测试平台，并且重点说明了该平台的简便实用性，以及漏洞检测与插件的编写支持多样性和可移植性高。通过对主机的端口和 Web 服务进行试验测试，最后的试验结果也表明：本文设计的渗透测试平台搭建简单快捷，可以进行灵活的编写插件，以及实现了快速的 Web 安全漏洞检测。

参考文献

[1] 陈一鸣，寇小强，王永利.基于 Docker 的漏洞验证框架的设计与实现[J].电子技术应用，2018，44(11)：99－101.

[2] 范国闯，钟华，黄涛，等.Web 应用服务器研究综述[J].软件学报，2003，14(10)：1728－1739.

[3] 单振华，王舒憧，强杰.基于 Websocket 的智能家居远程控制系统的实现[J].电子技术应用，2017，43(10)：124－127.

[4] 邢斌，高岭，孙骞，等.一种自动化的渗透测试系统的设计与实现[J].计算机应用研究，2010，27(4)：1384－1387.

[5] 袁浩.计算机网络渗透测试研究[D].重庆：重庆大学，2006.

[6] 隋亮.基于渗透测试的 SQL 注入漏洞检测与防范[D].上海：东华大学，2014.

[7] 韦鲲鹏，葛志辉，杨波.PHP Web 应用程序上传漏洞的攻防研究[J].信息网络安全，2015(10)：53－60.

[8] 王强，蔡皖东，姚烨.基于渗透测试的跨站脚本漏洞检测方法研究[J].计算机技术与发展，2013(3)：147－151.

(收稿日期：2020－03－23)

作者简介：

范晶(1987－)，男，硕士，主要研究方向：计算机网络与信息安全。

焦运良(1985－)，男，硕士，主要研究方向：国产化的控制系统。

戴贻康(1991－)，男，硕士，主要研究方向：无线通信、嵌入式硬件开发。

(上接第 37 页)

39(7)：1446－1450.

[10] 秦放.基于深度学习的昆虫图像识别研究[D].成都：西南交通大学，2018.

[11] 张力超，马蓉，张垚鑫.改进的 LeNet－5 网络在苹果图像识别中的应用[J].计算机工程与设计，2018，39(11)：3570－3575.

[12] IOFFE S，SZEGEDY C.Batch normalization：accelerating deep network training by reducing internal covariate shift[C].Proceedings of the 32nd International Conference on Machine Learning(ICML－15)，2015：448－456.

[13] SZEGDY C，LIU W，JIA Y，et al.Going deeper with convolutions[C].Proceedings of the 2015 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.Washington：IEEE Computer Society，2015：1－9.

[14] LECNU Y，BOTTOU L，BENGIO Y，et al.Gradient－based learning applied to document recognition[J].Proceedings of the IEEE(0018－9219)，1998，86(11)：2278－2324.

(收稿日期：2020－04－01)

作者简介：

郑睿(1980－)，通信作者，男，博士，副教授，硕士生导师，主要研究方向：自动控制技术、神经网络。E－mail：503757173@qq.com。

余童(1993－)，男，硕士研究生，主要研究方向：自动控制技术、神经网络。

程龙阅(1994－)，女，硕士研究生，主要研究方向：神经网络。

版权声明