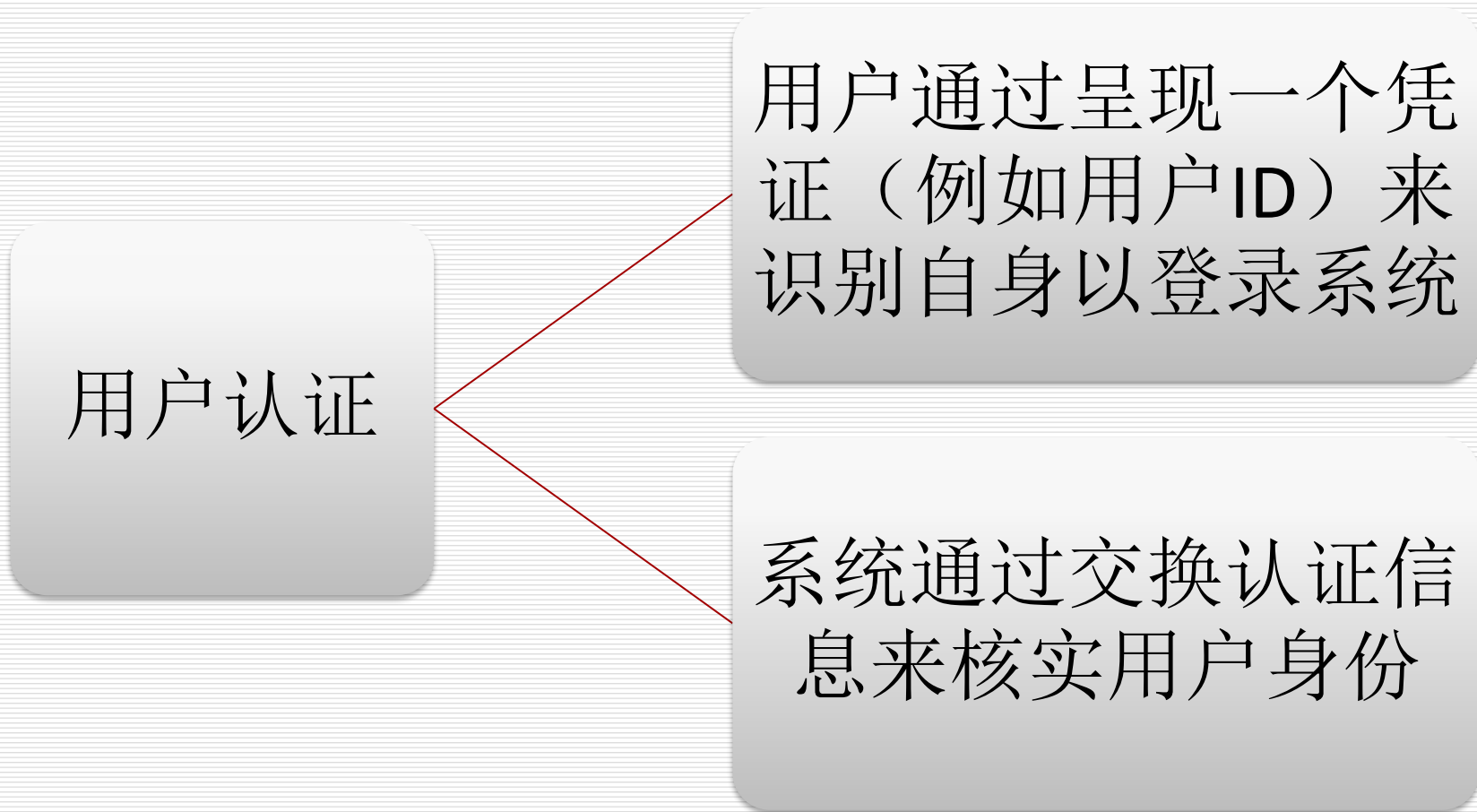


第三章 用户认证

- 3.1 数字用户认证方法
- 3.2 基于口令的认证
- 3.3 基于令牌的认证
- 3.4 生物特征认证
- 3.5 远程用户认证
- 3.6 用户认证中的安全问题
- 3.7 实际应用和案例学习

用户认证功能



补充知识： 口令的历史与现状

- 20世纪80年代，当计算机开始在公司里广泛应用时，人们很快就意识到需要保护计算机中的信息。
- 如果仅仅使用一个userID来标识自己，由于别人很容易得到这个userID，几乎无法阻止某些人冒名登录。基于这一考虑，用户登录时不仅要提供userID来标识自己是谁，还要提供只有自己才知道的口令来向系统证明自己的身份。

补充知识：口令的历史与现状

- **口令的作用**就是向系统提供唯一标识个体身份的机制，只给个体所需信息的访问权，从而达到保护敏感信息和个人隐私的作用。
- 虽然口令的出现使登陆系统时的安全性大大提高，但是这又产生了一个很大的问题。
- 如果口令过于简单，容易被人猜解出来；如果过于复杂，用户往往需要把它写下来以防忘记，这种做法也会增加口令的不安全性。
- **当前，计算机用户的口令现状是令人担忧的。**

补充知识：口令破解方式概述

- 口令破解是入侵一个系统比较常用的方法。
- 获得口令的思路：
 - 穷举尝试：最容易想到的方法
 - 设法找到存放口令的文件并破解
 - 通过其它途径如网络嗅探、键盘记录器等获取口令
- 这里所讲的口令破解通常是指通过前两种方式获取口令。这一般又有两种方式：手工破解和自动破解。

补充知识：口令破解方式概述（2）

- 手工破解的步骤一般为：
 - 产生可能的口令列表
 - 按口令的可能性从高到低排序
 - 依次手动输入每个口令
 - 如果系统允许访问，则成功
 - 如果没有成功，则重试。
 - 注意不要超过口令的限制次数
- 这种方式需要攻击者知道用户的**userID**，并能进入被攻击系统的登陆界面。需要先拟出所有可能的口令列表，并手动输入尝试。
- 思路简单，但是费时间，效率低

补充知识：口令破解方式概述（3）

□ 自动破解

- 只要得到了加密口令的副本，就可以离线破解。这种破解的方法是需要花一番功夫的，因为要得到加密口令的副本就必须得到系统访问权。
- 但是一旦得到口令文件，口令的破解就会非常的快，而且由于是在脱机的情况下完成的，不易被察觉出来。

补充知识： 口令破解方式概述（4）

□ 自动破解的一般过程如下：

- 找到可用的userID
- 找到所用的加密算法
- 获取加密口令
- 创建可能的口令名单
- 对每个单词加密
- 对所有的userID观察是否匹配
- 重复以上过程，直到找出所有口令为止

补充知识： 词典攻击

- 所谓的词典，实际上是一个单词列表文件。这些单词有的纯粹来自于普通词典中的英文单词，有的则是根据用户的各种信息建立起来的，如用户名、生日、街道名字、喜欢的动物等。
- 简而言之，词典是根据人们设置自己账号口令的习惯总结出来的常用口令列表文件



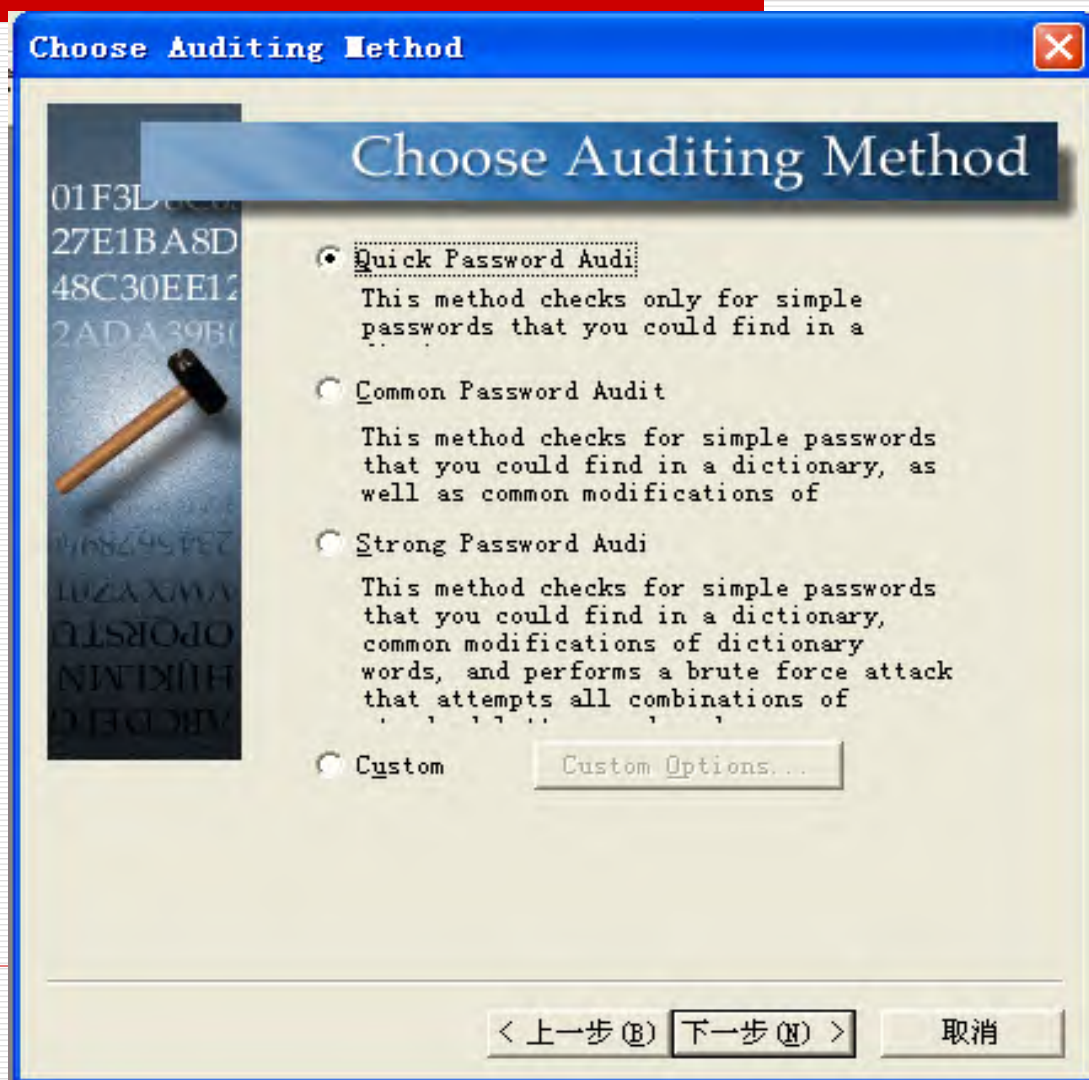
补充知识： 词典攻击(2)

- 使用一个或多个词典文件，利用里面的单词列表进行口令猜测的过程，就是词典攻击。
- 多数用户都会根据自己的喜好或自己所熟知的事物来设置口令，因此，口令在词典文件中的可能性很大。而且词典条目相对较少，在破解速度上也远快于穷举法口令攻击。
- 在大多数系统中，和穷举尝试所有的组合相比，词典攻击能在很短的时间内完成。

补充知识： 词典攻击(3)

- 用词典攻击检查系统安全性的好处是能针对特定的用户或者公司制定。
- 如果有一个词很多人都用来作为口令，那么就可以把它添加到词典中。
- 在Internet上，有许多已经编好的词典可以用，包括外文词典和针对特定类型公司的词典。
- 例如，在一家公司里有很多体育迷，那么就可以在核心词典中添加一部关于体育名词的词典。

补充知识：选择破解办法为快速口令破解



补充知识：密码为空的破解结果

@stake LC5 - [Untitled1]

File View Session Schedule Remediate Help

Run Report

Domain	User Name	LM Password	<8	Password	Age (d..
NIPC-CHALL...	ACTUser				15
NIPC-CHALL...	Administrator	????????DC			16
NIPC-CHALL...	ASPNET				16
NIPC-CHALL...	challenger	* empty *	x	* empty *	0
NIPC-CHALL...	Guest	* empty *	x	* empty *	0
NIPC-CHALL...	HelpAssistant				16
NIPC-CHALL...	IUSR_NIPC-CHALLENGER				16
NIPC-CHALL...	IWAM_NIPC-CHALLENGER				16
NIPC-CHALL...	SQLDebugger	* empty *			15
NIPC-CHALL...	SUPPORT_388945a0	* empty *			16

Dictionary 1 of 1 [C:\Program Files\@stake\LC5\words-english.dic]

DICTIONARY/HYBRID

- words_total: 29156
- words_done: 0
- % done: 0.000%

PRECOMPUTED

- hash_tables: 0 of 0
- hashes_found: 0 of 0
- % done: 0.00%

BRUTE FORCE

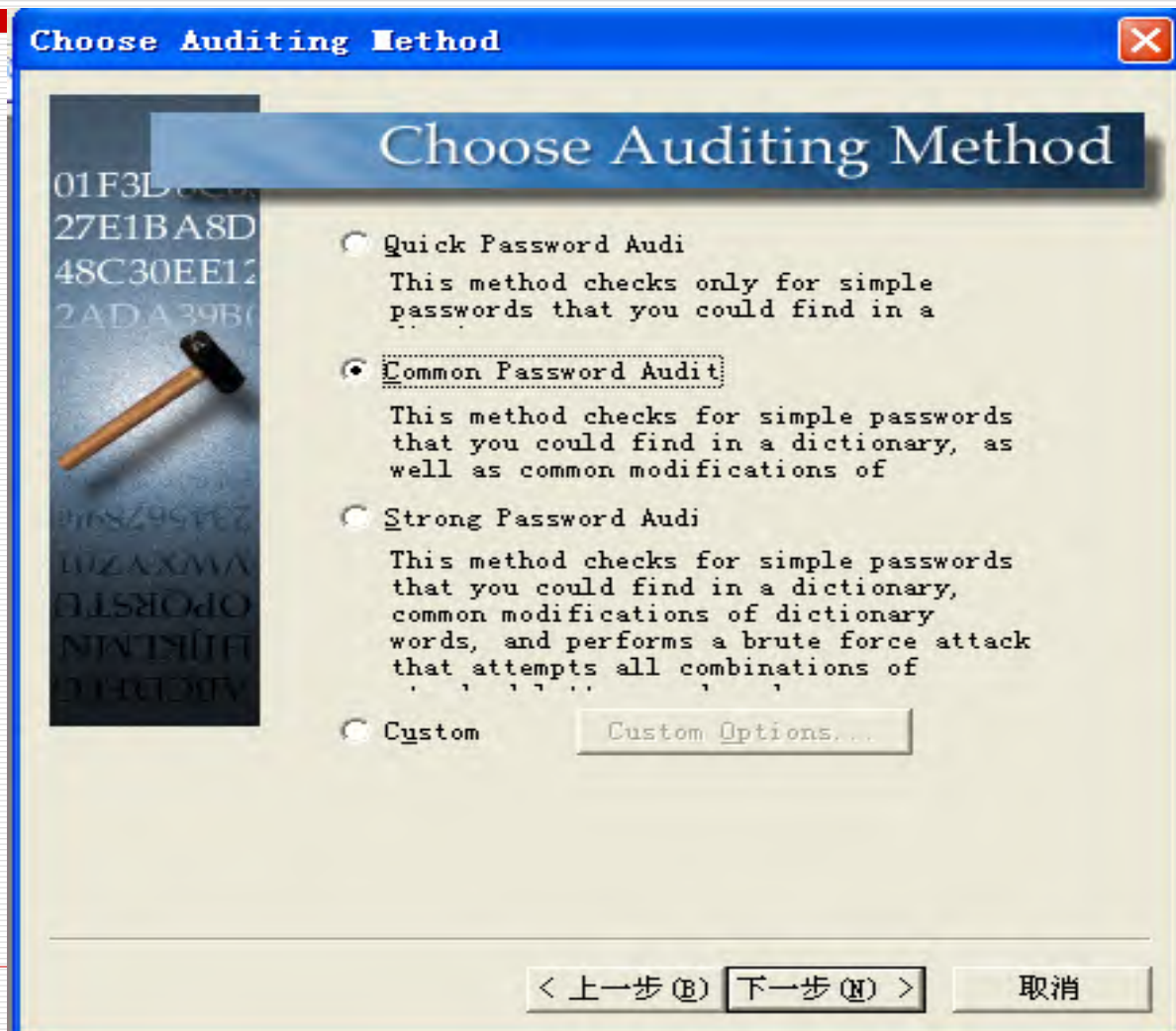
- time_elapsed: 0d 0h 0m 0s
- time_left:
- % done:
- current_test:
- keyrate:

SUMMARY

- total_users: 10
- audited_users: 2

NUM

补充知识：更改破解方法为普通口令破解



补充知识：密码为security的破解结果

The screenshot shows the @stake LC5 interface with a table of cracking results. The 'challenger' user's password 'SECURITY' has been successfully cracked to 'security'. A red circle highlights the word 'security' in the 'Password' column, with a callout bubble stating '密码是 security' (The password is security). Another red circle highlights the 'Run' tab. A text box at the bottom explains that the cracking was successful and quick because 'security' was in the password dictionary.

Domain	User Name	LM Password	<8	Password	Age (d.)
NIPC-CHALL...	ACTUser				15
NIPC-CHALL...	Administrator	???????DC			16
NIPC-CHALL...	ASPNET				16
NIPC-CHALL...	challenger	SECURITY		security	0
NIPC-CHALL...	Guest	* empty *	x	* empty *	0
NIPC-CHALL...	HelpAssistant				16
NIPC-CHALL...	IUSR_NIPC-CHALLENGER				16
NIPC-CHALL...	IWAM_NIPC-CHALLENGER				16
NIPC-CHALL...	SQLDebugger	* empty *			15
NIPC-CHALL...	SUPPORT_388945a0	* empty *			16

Dictionary 1 of 1 [C:\Program Files\@stake\LC5\words-english.dic]

Dictionary/HYBRID

- words total: 29156
- words done: 2345
- % done: 8.043%

PRECOMPUTED

- hash tables: 0 of 0
- hashes found: 0 of 0
- % done: 0.00%

BRUTE FORCE

- time elapsed: 0d 0h 0m 0s
- time left:
- % done:
- current test:
- keyrate:

SUMMARY

- total users: 10
- audited users: 2

NUM

补充知识：密码为security123的破解结果—无法破解

@stake LC5 - [Untitled3]

File View Session Schedule Remediate Help

Run Report

Domain	User Name	LM Password	<8	Password	Password Age (d.)
NIPC-CHALL...	ACTUser				15
NIPC-CHALL...	Administrator	???????DC			16
NIPC-CHALL...	ASPNET				16
NIPC-CHALL...	challenger	SECURIT???????			0
NIPC-CHALL...	Guest	* empty *	x	* empty *	0
NIPC-CHALL...	HelpAssistant				16
NIPC-CHALL...	IUSR_NIPC-CHALLENGER				16
NIPC-CHALL...	IWAM_NIPC-CHALLENGER				16
NIPC-CHALL...	SQLDebugger	* empty *			15
NIPC-CHALL...	SUPPORT_388945a0	* empty *			16

Dictionary 1 of 1 [C:\Program Files\@stake\LC5\words-english.dic]

NUM

DICTIONARY/HYBRID

words total 29156

words done 3543

% done 12.152%

PRECOMPUTED

hash tables 0 of 0

hashes found 0 of 0

% done 0.00%

BRUTE FORCE

time elapsed 0d 0h 0m 0s

time left

% done

current test

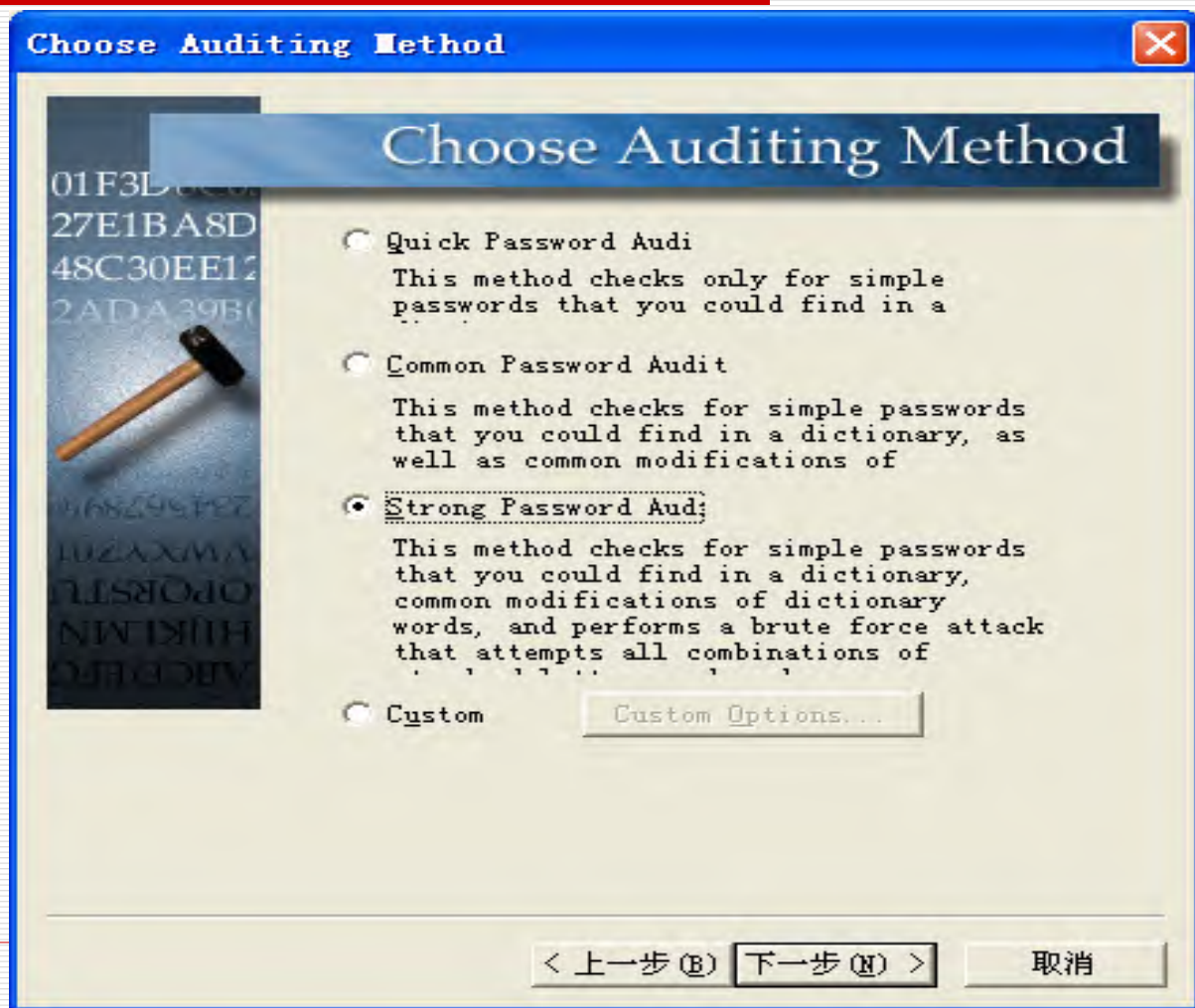
keyrate

SUMMARY

total users 10

audited users 1

补充知识：更改破解方法为复杂口令破解



补充知识：密码为security123的破解结果

Estake LC5 - [Untitled1]

File View Session Schedule Remediate Help

Run Report

Domain	User Name	LM Password	<8	Password	
NIPC-CHALL...	ACTUser				15
NIPC-CHALL...	Administrator	????????DC			16
NIPC-CHALL...	ASPNET				16
NIPC-CHALL...	challenger	SECURITY123		security123	0
NIPC-CHALL...	Guest	* empty *	x	* empty *	0
NIPC-CHALL...	HelpAssistant				16
NIPC-CHALL...	IUSR_NIPC-CHALLENGER				16
NIPC-CHALL...	IWAM_NIPC-CHALLENGER				16
NIPC-CHALL...	SQLDebugger	* empty *			15
NIPC-CHALL...	SUPPORT_388945a0	* empty *			16

Dictionary/Hybrid

words_total: 29156
words_done: 0
% done: 0.000%

Precomputed

hash_tables: 0 of 0
hashes_found: 0 of 0
% done: 0.00%

Brute Force

time_elapsed: 0d 0h 0m20s
time_left: 0d 4h43m53s
% done: 0.1173%
current_test: ATJBKS
keyrate: 4726382 k/s

Summary

total_users: 10
audited_users: 2

NUM

密码是 security123

3.1 数字用户认证方法

- 3.1.1 电子用户认证模型
- 3.1.2 认证方法
- 3.1.3 多因素认证
- 3.1.4 用户认证的保证级别

3.1.1 电子用户认证模型

- **NIST SP 800-63-3** 将用户认证定义为：信息系统对用户电子式提交的身份建立信任的过程。系统根据用户提交的认证信息判断用户是否被授权使用系统的某些资源。
- p.s. 在使用window系统时需要输入pin来登录设备，获得pc的使用权

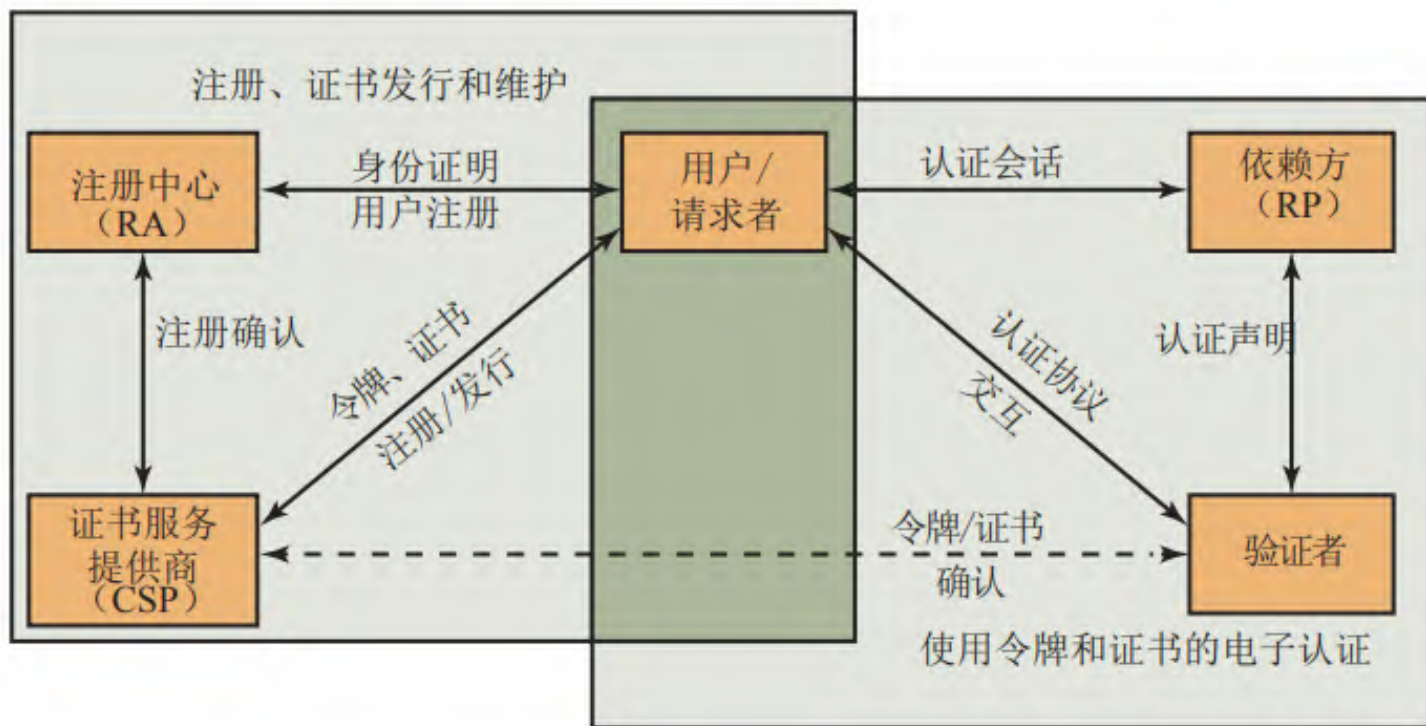
识别和认证安全要求

表 3.1 识别和认证安全要求 (SP 800-171)

基本安全要求	
1	识别信息系统用户，以用户的名义执行的进程或设备
2	认证 (authenticate) (或核实 (verify)) 这些用户、进程或设备的身份，作为允许访问组织信息系统的先决条件
派生的安全需求	
3	使用多因素身份验证进行本地和网络的特权账户访问，以及非特权账户的网络访问
4	对特权和非特权账户的网络访问采用防重放认证机制
5	防止在定义的时间段内重用标识符
6	在确定的时间段内不活动后禁用标识符
7	在创建新口令时强制最小口令复杂度并更改字符
8	禁止在规定的代(generation)数内重用口令
9	允许临时口令用于系统登录，但需立即更改为永久口令
10	仅仅存储和传输密码保护的口令
11	认证信息的模糊化反馈

3.1.1 电子用户认证模型

我们参考下图讨论用户身份认证通用模型：



3.1.2 认证方法

验证用户身份的一般方法有四种，它们可以单独使用也可以组合起来使用：

个人知道的信息

- 口令、个人识别码 (PIN) 或预先安排的一组问题的答案

个人拥有的物品

- 电子钥匙卡、智能卡和物理钥匙。这种类型的身份验证器称为令牌。

个人生理特征（静态生物特征）

- 指纹识别、虹膜识别和人脸识别

个人行为特征（动态生物特征）

- 通过语音模式、笔迹特征和打字节奏进行识别

3.1.3 多因素认证

- ❑ 多因素认证（**Multifactor Authentication, MFA**）：用户需提供两个或多个独立身份验证因素（如所知、所有、所属、所做）的组合，以验证身份。
- ❑ 例子：网上银行系统要求用户首先输入口令（用户所知），然后输入通过短信发送到手机的验证码（用户所有）。
- ❑ **NIST SP 800-63B** 要求使用多种身份验证方法来获得更高的身份验证保证级别。

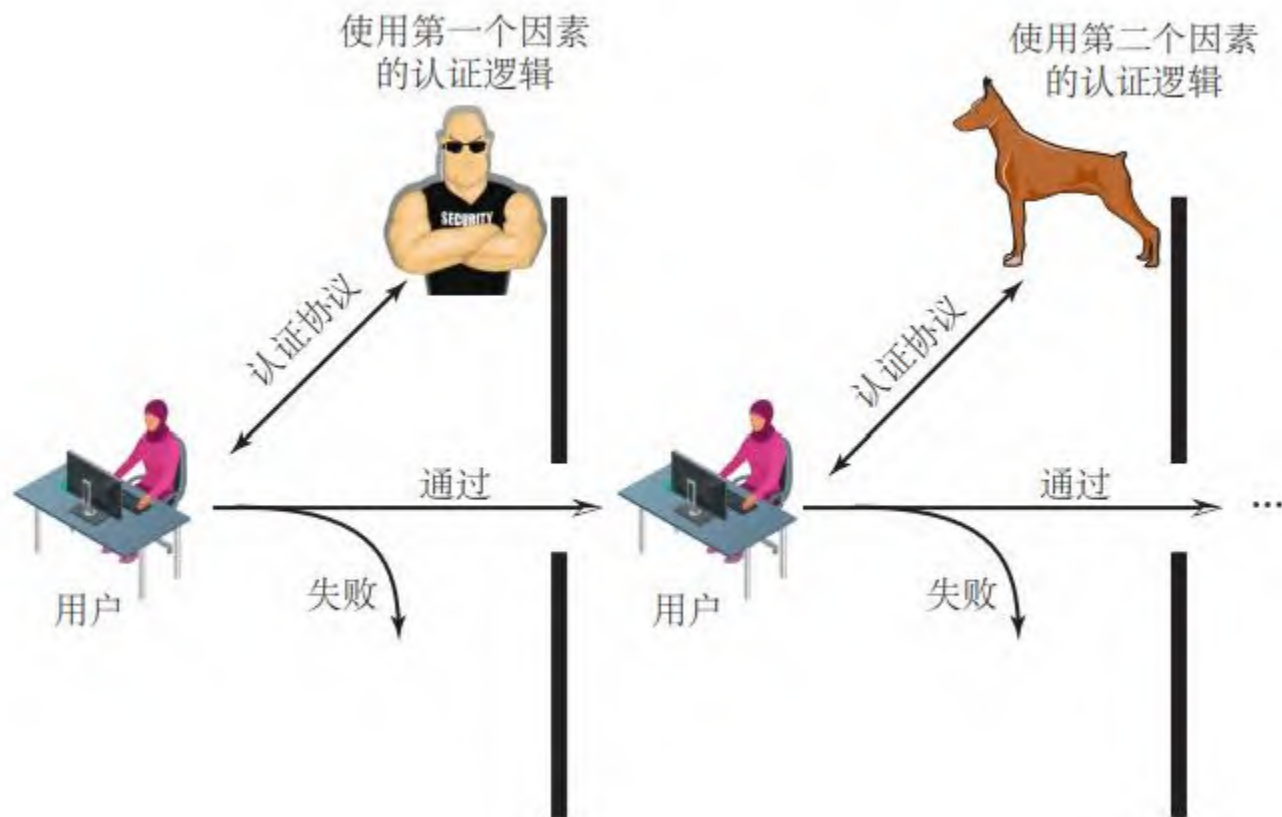
3.1.3 多因素认证

物理令牌是大型组织最常用于验证员工身份的 **MFA** 方法，通常是一次性口令（**one-time password, OTP**）设备。



用户提供登录请求和口令，然后输入物理令牌上的 **OTP** 代码。对于普通用户来说，最常用的 **MFA** 身份验证是使用移动设备接收通过短信 / 语音呼叫发送的验证码，或者使用验证器应用生成的 **OTP** 代码。

3.1.3 多因素认证



3.1.4 用户认证的保证级别

- 组织可以根据对身份证明和验证过程的信心程度，在一系列用户身份验证技术之间进行选择。**NIST SP 800-63-3** 为身份保证等级（**Identity Assurance Level, IAL**）和验证者保证等级（**Authenticator Assurance Level, AAL**）分别定义了三个单独的等级。

3.1.4 用户认证的保证级别

根据风险评估和攻击者成功谎报身份所造成的潜在危害，将 IAL 分为三级，组织可自行选择。

- **IAL1:** 无须将申请人与特定的现实身份联系起来。所提供的任何属性都是自断言的。
- **IAL2:** 提供支持所声称的身份存在的证据，并使用远程或实际存在的身份证明来验证申请人是否与该真实身份或假名身份有适当关联。此级别适用于需要明确初始身份的各种组织。
- **IAL3:** 需要物理设备以进行身份验证。识别属性必须由一个经过 **CSP** 授权和培训的代表进行验证。该级别适用于使客户或员工能够访问具有高价值的受限服务，或者访问不当对其非常有害的情形。

3.1.4 用户认证的保证级别

基于风险评估和攻击者控制身份验证器并访问系统所造成的潜在危害，将 AALs 划分为三个等级，组织可自行选择。

- **AAL1**：通过安全身份验证协议确保请求方可以控制绑定到订阅者账户的身份验证程序。**AAL2**：为请求方控制绑定到订阅者账户的验证者提供高可信度。需要通过使用经批准的加密技术的安全身份验证协议来证明拥有且控制两个不同的身份验证因素。
- **AAL3**：为请求方控制绑定到订阅者账户的验证者提供非常高的可信度。身份验证基于经批准的加密协议证明拥有密钥，并且必须使用基于硬件的身份验证器和提供验证器模拟阻力的身份验证程序。

3.2 基于口令的认证

- 3.2.1 口令的脆弱性
- 3.2.2 散列口令的使用
- 3.2.3 破解“用户选择”口令
- 3.2.4 口令文件访问控制
- 3.2.5 口令选择策略

基于口令的认证

基于口令的认证对应的是“个人所知道的信息”，可以通俗的理解成账号密码。

保证用户id
安全性



```
graph LR; A[保证用户id 安全性] --- B[决定用户是否被授权访问系统]; A --- C[决定了该用户所拥有的访问权限]; A --- D[应用在自主访问控制机制中];
```

决定用户是否被授权访问系统

决定了该用户所拥有的访问权限

应用在自主访问控制机制中

3.2.1 口令的脆弱性



口令面临的威胁以及对策

离线字典攻击：攻击者获取系统口令文件，并将口令的哈希值与常用口令的哈希值进行比较。如果找到匹配项，攻击者可以通过该ID/密码组合获得访问权限。

- 应对方法：
 - 1.对口令文件进行保护和访问限制；
 - 2.入侵检测技术，防止非法访问的发生
 - 3.禁止使用弱口令

口令面临的威胁以及对策

特定帐户攻击：攻击者以特定帐户为目标，不断尝试密码，直到发现正确的密码为止。

- 应对方法：
- 账户锁定，当尝试次数达到一定量后，锁定该账号
- 典型的实践方法就是设置不超过五次的登录尝试次数。

口令面临的威胁以及对策

常用口令攻击：前一种攻击的一种变体是使用常用的口令，对大量的用户id进行尝试。用户设置口令的倾向是选择一个容易记住的口令，不幸的是，这使得密码很容易猜测。

- 应对方法：
- 1.禁止使用弱口令。
- 2.用户来源验证。

口令面临的威胁以及对策

单用户口令猜测：攻击者对攻击目标进行信息收集（生日，年龄等一些列可能和口令有关的信息），随后进行猜测。

- 应对方法：
- 1.训练并加强口令保护策略以使口令难于猜测。
- 2.策略涉及保密、口令的最小长度、字符集、禁止使用常用用户ID等内容。

口令面临的威胁以及对策

工作站劫持：攻击者确定管理员已经登录，在其不注意的情况下入侵占领工作站。

- 应对方法：
- 1. 在工作站处于非活动状态时采用自动注销的机制
- 2. 使用入侵检测方案对用户行为的变化进行检测

口令面临的威胁以及对策

利用用户疏漏：如果是由系统分配口令，那么用户通常会把分配的口令记录下来，因为它很难记忆。这种情况使得攻击者有机会读到记录下来的口令。

- 应对方法：
- 1. 用户培训
- 2. 入侵检测
- 3. 使用口令与其他认证机制的组合认证

口令面临的威胁以及对策

口令重复利用：如果一个指定用户对不同的网络设备使用相同或相近的口令，那么攻击就会变得效率很高，破坏性很强。

- 应对方法：
- 禁止为特定的网络设备设置相同或相近的口令

口令面临的威胁以及对策

电子监视：如果用户需要通过网络来登录远程的系统，那么就有被窃听的危险。

- 应对方法：
- 加密传输的同时，避免使用弱口令，加大破解难度

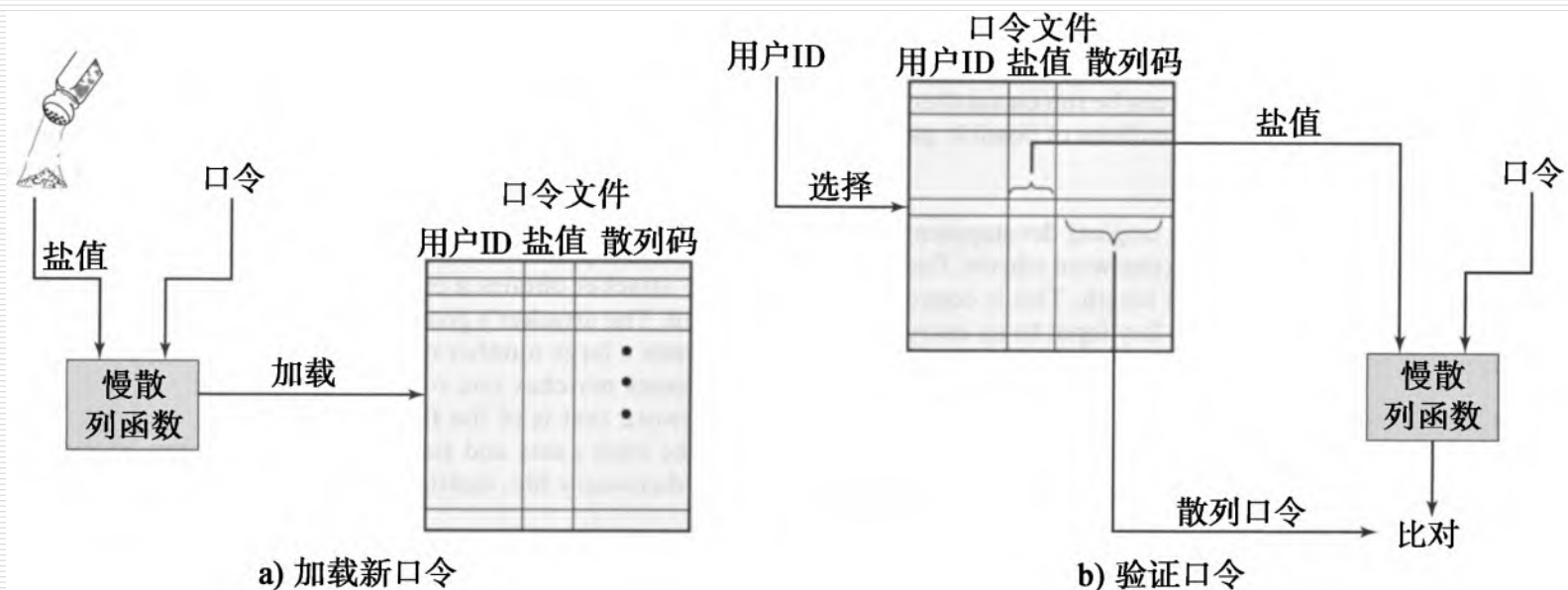
3.2.2 哈希口令的使用

- 口令安全技术广泛使用了散列函数和“盐值”。
- 盐值S和口令X一同作为哈希函数的输入，其可以是当前时间，也可以是随机数或伪随机数。关于这种方案的存储结构如下：

用户ID	盐值	散列码
01	S	hash (S, X)

3.2.2 哈希口令的使用

这种策略实际上在所有UNIX变体及其他操作系统上都在使用，如下图：



3.2.2 哈希口令的使用

使用“盐值”
目的

```
graph TD; A[使用“盐值”目的] --- B[防止复制的口令在口令文件中可见。]; A --- C[显著地增加了离线口令字典攻击的难度。]; A --- D[使得攻击者几乎不可能发现一个用户是否在两个或更多的系统中使用了相同的口令。];
```

防止复制的口令在
口令文件中可见。

显著地增加了离线
口令字典攻击的难
度。

使得攻击者几乎不
可能发现一个用户
是否在两个或更多
的系统中使用了相
同的口令。

3.2.3 破解“用户选择”口令

传统方法

获取到口令文件，从而能够得到盐值。构建一个常用口令字典，将其中的口令和口令文件中的盐值进行运算，得到哈希值，与口令文件中的哈希值进行对比，从而达到破解的作用。

无法获取到口令文件的方案。在构建口令字典的同时，也构造一个盐值字典，两表中的内容进行哈希运算，得到一张新的表（即彩虹表），一一尝试对照达到破解的作用。

现代方法采用超级计算机运行一些带有自学习机制的潜在口令算法进行破解。

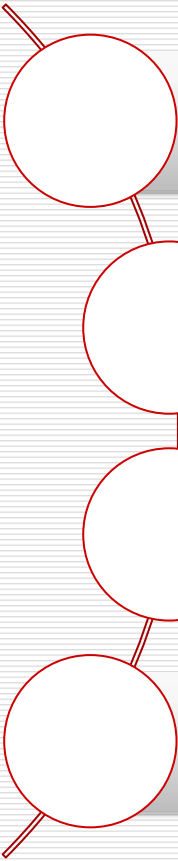
3.2.4 口令文件访问控制

一种阻止口令攻击的方法就是拒绝对手访问口令文件，即如果文件的散列口令部分只能被特权用户访问。对于影子口令文件，需要特别注意保护，防止非授权访问。

漏洞：

- 操作系统中软件的漏洞泄露口令文件
- 某些偶然事件会导致口令文件可读
- 用户在其他系统上拥有相同的口令
- 访问备份磁盘
- 从网络传输中获取口令

3.2.5 口令选择策略



用户教育：用户可以被告知使用强密码的重要性，并可以提供选择强密码的指导方针。但是用户往往会忽略，所以这一策略几乎是无效的。

计算机产生口令：系统为用户分发一个复杂的密码。系统生成的密码虽然具备强密码的特征，但是难以被用户接受。

后验口令检查：系统定期运行自己的密码破解器来找到可猜测的密码。

先验口令检查：允许用户选择自己的密码。但是，在选择时，系统会检查密码强度是否足够，如果密码强度不达标，则拒绝密码的使用。

先验口令检查

- ❑ 规则实施：口令必须遵守的特定规则
- ❑ 口令检查器：构建一个庞大的“不可行”口令字典
- ❑ **Bloom过滤器：**
 1. 建立一个由哈希值构建的散列表
 2. 对照此表检查所需口令

3.3 基于令牌的认证

- 3.3.1 存储卡
- 3.3.2 智能卡
- 3.3.3 电子身份证
- 3.3.4 硬件身份认证令牌
- 3.3.5 使用移动电话的认证

令牌

令牌就是用户持有的用于进行用户认证的一种物品，以下是用作令牌的卡的类型：

卡的类型	定义的特征	实例
凹凸卡	卡的正面有凸印的字符	老式信用卡
磁条卡	卡的背面有磁条，正面有字行	银行卡
存储卡	卡的内部有电子存储单元	预付电话卡
智能卡 接触式 非接触式	卡内有电子存储单元和处理器 表面有电子触点 内部嵌有无线电通信装置	生物特征 ID 卡

3.3.1 存储卡

- ❑ 存储卡只能存取数据而不能处理数据。
- ❑ 最常见的就是银行卡，在卡的背面有磁条。这个磁条可以存储一些简单的安全码，磁卡可以通过一种价格并不昂贵的读卡器读取。
- ❑ 此类存储卡的内部含有一个电子存储器。
- ❑ 存储卡可以单独用于物理访问（房卡），有时也需要输入PIN配合验证（如ATM机上进行取款）。

存储卡缺点

- ❑ **需要特殊的读卡器:**读卡器需要软件和硬件来支持安全性，这提高了成本。
- ❑ **令牌丢失:**令牌丢失会使得用户暂时不能进入系统。补办令牌，会增加管理的成本。此外，如果令牌被偷窃或者被伪造，那么敌手只需要获取 PIN 就可以执行非授权的访问。
- ❑ **用户不满意:**虽然用户对于ATM存储卡的使用没有意见，但是把存储卡应用在计算机系统中，肯定是不方便的。

3.3.2 智能卡

- ❑ **物理特征：**智能令牌包括一个嵌入的微处理器，外表类似于银行卡，因此叫智能卡。其他智能令牌外表有的类似于计算器，有的类似于钥匙或其他便携式物品。
- ❑ **用户接口：**人机接口包括一个键盘区和显示设备，以完成人机交互

3.3.2 智能卡

□ 电子接口：

- 智能卡或其他令牌通常需要配一个电子接口，用于与读取或写入装置通信
- 接触式接口和非接触式接口

□ 认证协议是智能卡认证的规约，包含电子认证的算法思想，划分成三类：

- 静态协议
- 动态口令生成器
- 挑战-应答协议

智能卡和存储卡最大的区别在于：存储卡只能存储但不能处理数据，但是智能卡由于拥有一个微处理器，由此可以处理数据。

3.3.2 智能卡

□ 智能卡包含一个完整的微处理器：

- 处理器 内存 I/O端口

□ 典型的智能卡包括三种存储器：

- 只读存储器(ROM)：存储的数据在整个智能卡的生命周期内都不会发生变化
- 电子可擦写可编程存储器(EEPROM)：存储应用程序和数据
- 随机存取存储器(RAM)：保存应用程序执行时产生的临时数据

3.3.3 电子身份证

最新、最前沿的电子身份证大规模应用是德国电子身份证--新身份证，该卡的表面印有人可以阅读的信息，其中包括：

- ❑ **个人信息:** 例如姓名、出生日期及住址;此类信息一般在护照或驾驶执照等证件上也可找到。
- ❑ **文档编号:** 每一张智能卡均配有的由9位字母或数字构成的唯一识别符。
- ❑ **卡片接入号(CAN):** 印刷于智能卡表面的6位随机的十进制数字。这些数字被当作口令使用，下文中会解释它。
- ❑ **机器读卡区(MRZ):** 智能卡背面的三行文字，人类和机器均可阅读。这些文本同样也可作为口令使用。

电子身份证的功能和数据

功能	目的	PACE 口令	数据	用途
ePass (强制)	授权的离线检测系统读取数据	CAN 或 MRZ	面部特征:两个指纹图像(可选);MRZ 数据	用于专为政府部门保留的离线生物特征进行身份验证
eID (可选激活)	在线应用读取数据或访问授权功能	eID PIN	姓名、艺名和博士学位, 出生地及出生日期, 住址和社区 ID, 有效期	身份证明, 年龄验证, 社区 ID 验证, 受限的身份证明(化名)消除疑问
	离线检测系统读取数据并更新住址和社区 ID	CAN 或 MRZ		
eSign (可选证书)	认证在线安装签名证书	eID PIN	数字签名密钥.X.509 证书	数字签名生成
	公民利用 eSign 的 PIN 生成数字签名	CAN		

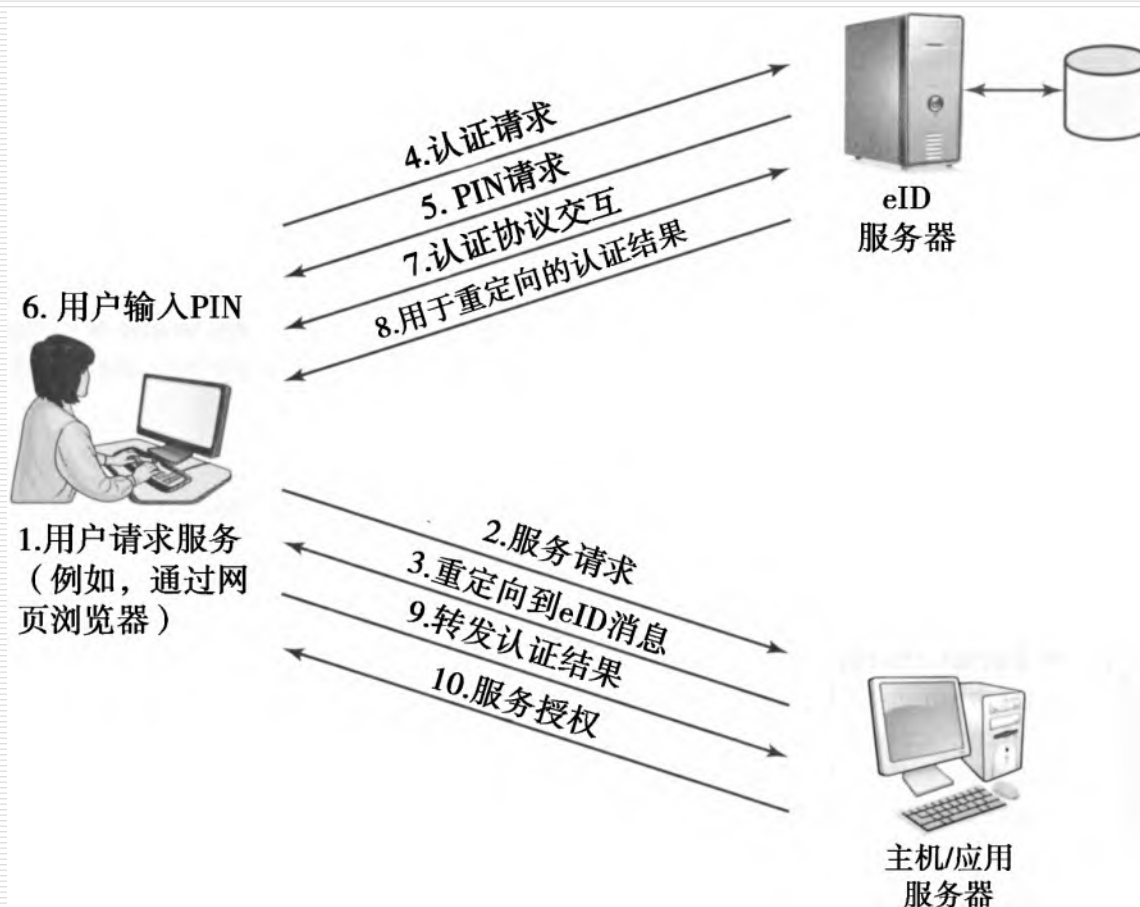
CAN = 卡片接入号

MRZ = 机器读卡区

PACE = 口令认证连接系统

PIN = 个人标识码

利用eID的用户认证



口令认证连接设施(PACE)

- 能够确保非接触式的RF芯片和电子身份证信息在没有明确访问控制的情况下不能被读取。
- 对于在线应用而言，只有用户输入了仅持卡人知道的6位PIN后，他才有权使用该卡。
- 对于离线应用而言，智能卡背面印有的MRZ和正面的卡片接入号(CAN)均可被用认证。

3.3.4 硬件身份认证令牌




用于身份验证的硬件包含一个或多个设备独有的嵌入式密钥，在身份验证过程中使用它们




最简单的硬件令牌之一是一次性口令（**OTP**）设备。



它有一个嵌入的密钥用作生成 **OTP** 的种子，并将其显示出来。作为身份验证过程的一部分，用户输入当前 **OTP**，系统单独计算预期的 **OTP**，并确认用户是否输入了正确的值。每个 **OTP** 只能使用一次。



OTP 可以是基于当前时间的不断变化的值，或者由计数器或其他值生成，称为随机数，每次使用 **OTP** 时都会更新。



这些设备通常使用分组密码或哈希函数来加密地密钥和时间 / 随机数值的组合以创建 **OTP**。通常还包括某种形式的防篡改模块，以安全地存储嵌入的密钥。

3.3.4 硬件身份认证令牌

- ❑ 最广泛实现的 OTP 算法之一是“基于时间的一次性口令”（Time-based One-Time Password, TOTP），通过 HMAC 和哈希函数（如 SHA-1）实现。
- ❑ TOTP 口令是根据当前 UNIX 格式的时间值计算得出的：
 - $T = \text{floor}((\text{Current Unix time} - \text{Time0})/\text{Step})$;
 $\text{TOTP}(\text{Key}, T) = \text{Truncate}(\text{HMAC-SHA-1}(\text{Key}, T))$

3.3.4 硬件身份认证令牌

- ❑ 使用基于时间的 OTP 系统在生成新值之前有几秒或几分钟的窗口期，如上面的 TOTP 计算所示。
- ❑ 需要考虑到令牌和验证系统之间可能存在的时钟漂移，这通常是通过允许当前系统时间的一个小窗口内的时间的 OTP 值来实现的。
- ❑ 如果 OTP 被提供的时间与系统时钟略有不同，那么系统可以记录漂移量，以便在未来的交互中进行补偿。
- ❑ 如果令牌和系统时钟偏离同步太远，则必须使用重新同步进程来重新建立时钟同步。

3.3.4 硬件身份认证令牌

- 使用 `nonce` 的系统需要允许失败的身份验证尝试，这意味着令牌生成的 `OTP` 比上次成功用于身份验证的 `OTP` 晚一个或几个步长。如果检测到这样一个较晚的 `OTP`，系统将更新 `nonce` 以匹配。
- 与基于时间的系统一样，如果令牌和系统 `nonce` 值太不同步，则必须使用重新同步进程。
 -

3.3.4 硬件身份认证令牌

显示以供用户输入的代码数字的令牌的一个缺点是：另一个人可以瞥一眼显示器并看到代码。作为替代，令牌可以使用与认证系统交互的通信链路，而非一个单独的显示器。

这可能需要用户将硬件令牌插入系统上的 USB 端口，或者使用近场通信（near-field communication, NFC）或低功耗蓝牙（low energy bluetooth, BLE）无线连接。

这种令牌通常结合了 OTP 功能且支持密码学操作，包含一个或多个嵌入密钥，可以执行一些公钥或私钥加密操作，例如，对质询值进行签名。还可以为保护密钥提供更通用的支持，并支持除身份验证之外的一系列密码学操作。

3.3.4 硬件身份认证令牌

使用硬件身份认证令牌的缺点

令牌丢失或被盗；

攻击者通过安装恶意软件破坏用户使用的计算机系统，从而破坏身份验证过程；

或者攻击者使用社会工程来说服用户透露身份验证代码或批准身份验证请求。

3.3.5 使用移动电话的认证

- ❑ 身份验证过时，用户在登录系统时必须输入通过短信或语音消息发送到手机的代码来验证自己的身份。
- ❑ 或者需要手机上的一个应用程序，该应用程序可以作为一次性口令生成器，或者与用户验证身份的系统进行主动信息交互。

3.3.5 使用移动电话的认证

- 通过短信或语音向手机发送身份验证码是使用手机作为身份验证令牌的最简单方法之一。
- 优点：
 - 不需要在手机上安装任何额外的应用程序。
- 缺点：
 - 只有用户所在地被手机信号覆盖时，才能接收短信或语音信息。如果用户的手机无服务，则此方法不起作用。
 - 手机也可能丢失或被盗。

3.3.5 使用移动电话的认证

- 身份验证可以通过安装在用户手机上的应用程序来完成。
- 优点：在使用时不需要网络连接。
- 缺点：
 - 手机丢失或被盗；
 - 攻击者通过安装恶意软件破坏手机，破坏身份验证过程；
 - 攻击者发送大量身份验证请求，称为“即时轰炸”，希望用户批准访问；
 - 攻击者使用社会工程来说服用户泄露身份验证代码。

3.4 生物特征认证

- 3.4.1 用于生物特征认证应用的身体特征
- 3.4.2 生物特征认证系统的运行
- 3.4.3 生物特征认证的准确度

3.4 生物特征认证

- ❑ 生物特征认证系统是通过个人唯一拥有的身体特征来实现认证的。
- ❑ 静态特征如指纹、手形、面部特征、视网膜和虹膜等，
- ❑ 动态特征如声纹和签名等。
- ❑ 本质上，生物特征认证是基于模式识别的。与口令和令牌的方法相比，生物特征认证实现技术较为复杂，成本也较高。

3.4.1 用于生物特征认证应用的身体特征

□ 静态生物特征：

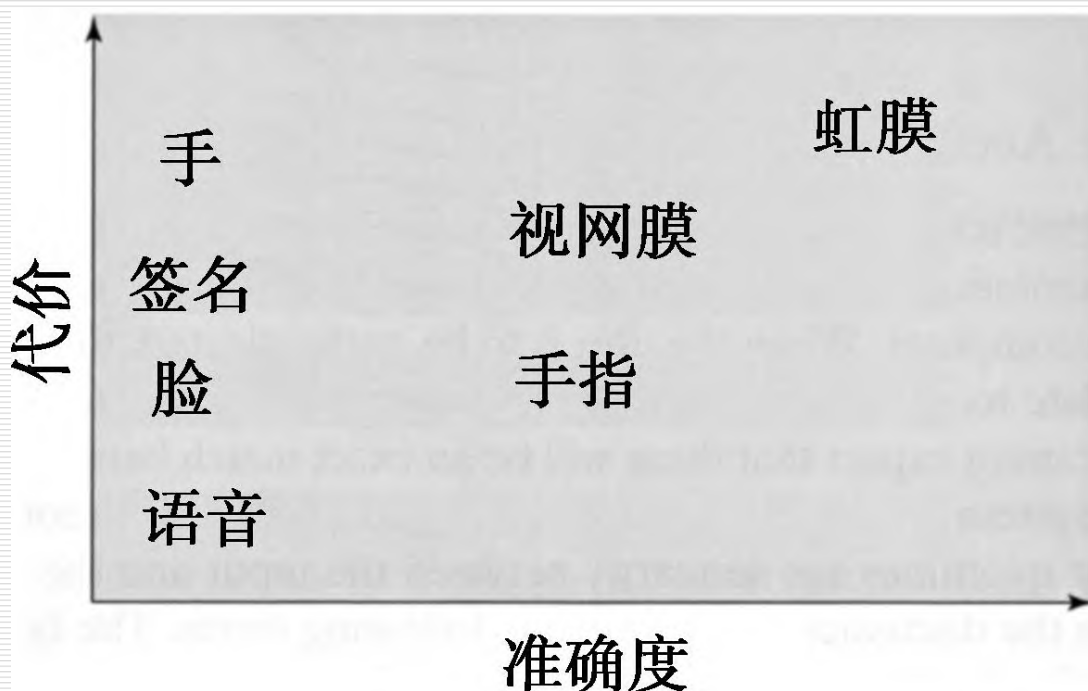
- **面部特征：**面部特征是人与人之间最常见的识别方法。根据关键面部特征的相对位置和形状来定义特征；使用红外摄像机来产生与人脸上潜在血管系统相关的人脸热图。
- **指纹：**自动指纹识别和匹配系统从指纹中提取出一个数量的特征进行存储，作为完整指纹模式的数字替代物。
- **手形 视网膜模式 虹膜**

3.4.1 用于生物特征认证应用的身体特征

□ 动态生物特征:

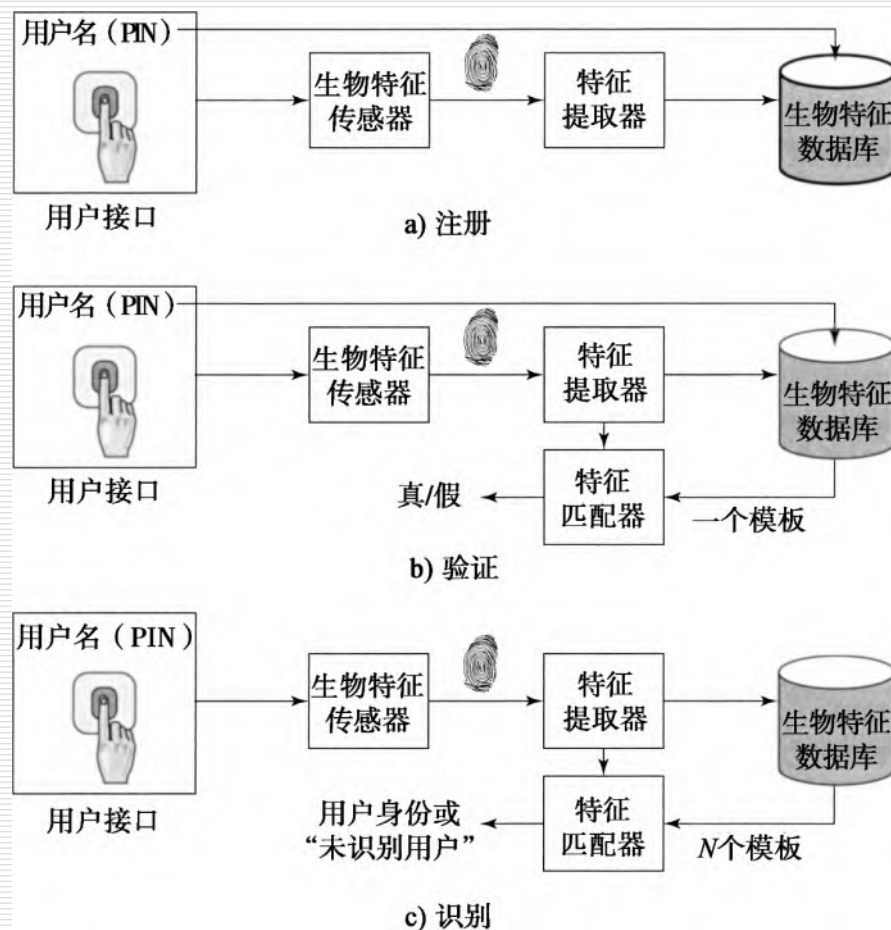
■ 签名 声音

右图是使用不同生物特征认证的成本与准确度对比。



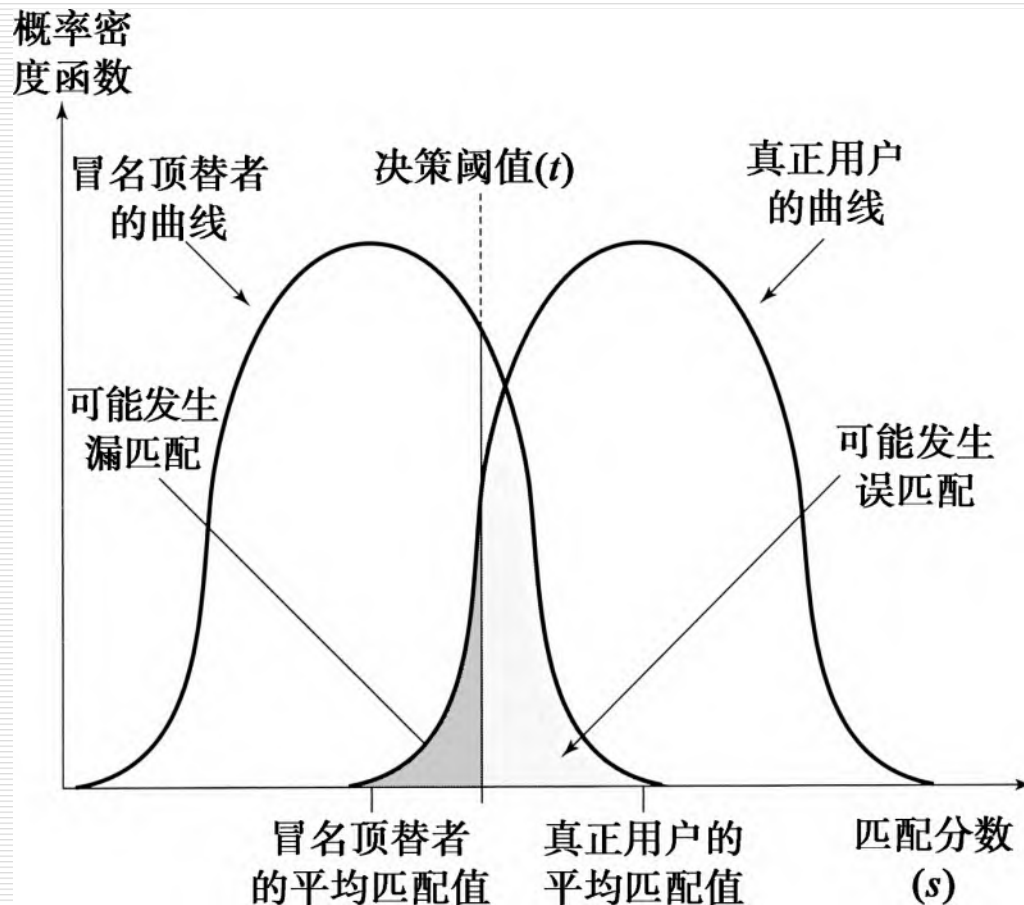
3.4.2 生物特征认证系统的运行

一个通用的生物特征认证系统。注册过程将会在系统中为用户和用户的生物特征创建一个关联。根据应用的不同，用户认证包括验证声称的用户是否是真实的用户，或者识别系统未知的用户。



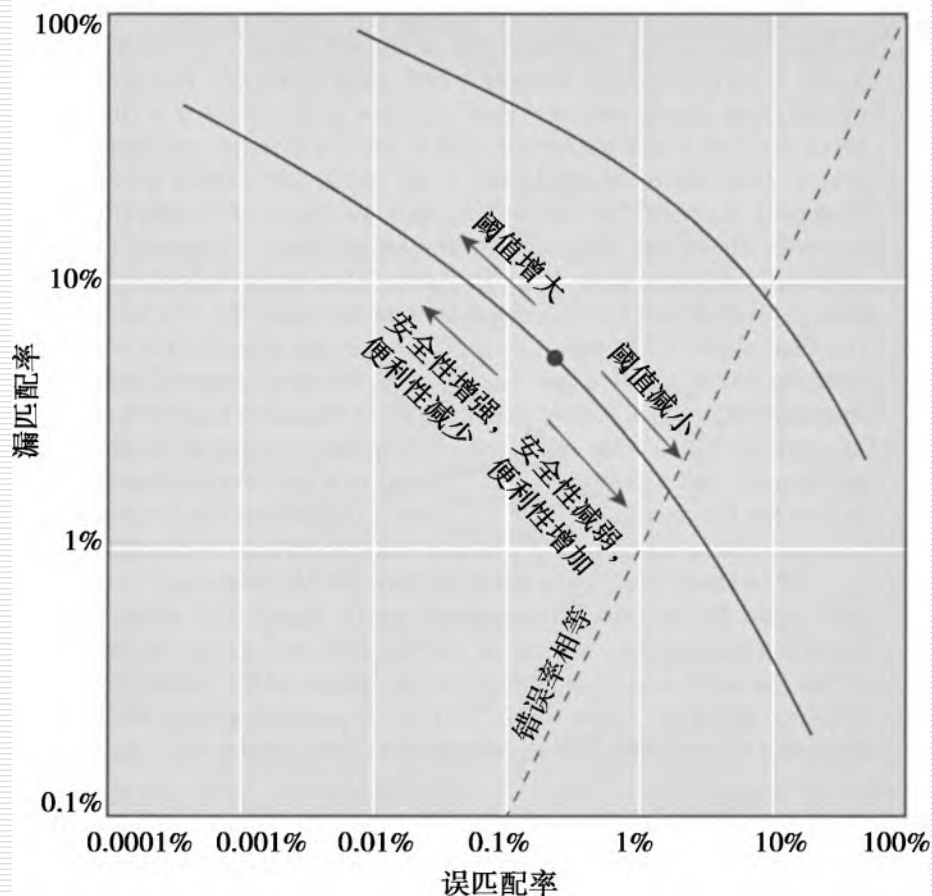
3.4.3 生物特征认证的准确度

授权用户和冒名顶替者的生物特征曲线。在本描述中将提交的特征和参考特征之间的差异简化为一个数值，如果输入值(s)大于预先指定的阈值(t)，那么就认为是匹配的。



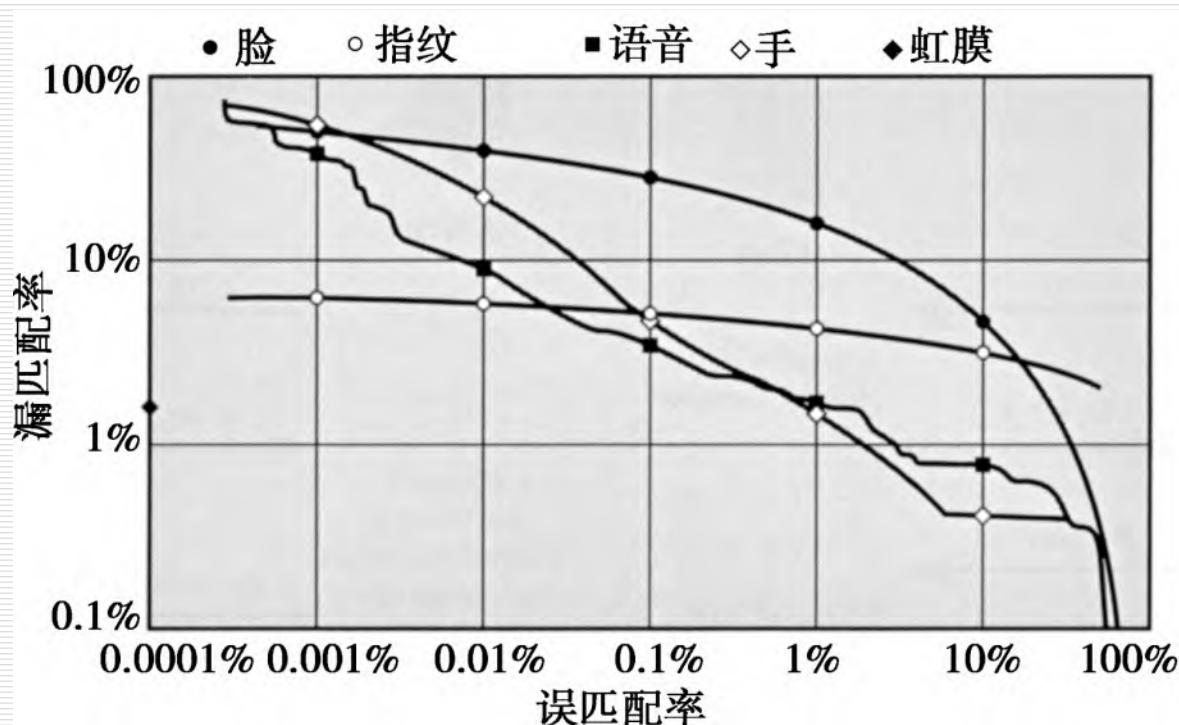
3.4.3 生物特征认证的准确度

理想化的生物特征测量运行特征曲线（对数-对数比例）



3.4.3 生物特征认证的准确度

[MANSO1]报道的真实生物特征测量运行特征曲线。为了更清楚地表示出各个系统的差异，图中采用了对数-对数刻度



3.5 远程用户认证

- 最简单的用户认证方式就是本地认证，即用户试图访问本地的系统，如单机的办公PC或者ATM机。
- 复杂一些的情况则是通过Internet、网络、通信线路的远程用户认证。
 - 远程认证的方式增加了很多安全威胁，例如口令窃听或者对观察到的用户认证过程进行重放等
 - 通常依靠某种形式的挑战-响应协议来应对威胁



客户端

U , 用户



远程主机

r , 随机数
 $h()$ 、 $f()$, 函数

P' , 口令
 r' , r 的返回

\xrightarrow{U}

$\xleftarrow{\{r, f(), h()\}}$

$\xrightarrow{f(r', h(P'))}$

$\xleftarrow{\text{yes/no}}$

$f(r', h(P')) =$
 $f(r, h(P(U)))$
如果成立则
yes, 否则no

a) 口令协议



客户端

U , 用户



远程主机

r , 随机数
 $h()$ 、 $f()$, 函数

$P' \rightarrow W'$, 口令
到令牌提供
的验证码
 r' , r 的返回

\xrightarrow{U}

$\xleftarrow{\{r, f(), h()\}}$

$\xrightarrow{f(r', h(W'))}$

$\xleftarrow{\text{yes/no}}$

$f(r', h(W')) =$
 $f(r, h(W(U)))$
如果成立则
yes, 否则no

b) 令牌协议



客户端

U , 用户



远程主机

r , 随机数
 $E()$, 函数

\xrightarrow{U}

$\xleftarrow{\{r, E()\}}$

$\xrightarrow{E(r', D', BT')}$

$\xleftarrow{\text{yes/no}}$

$B' \rightarrow BT'$ 生物
特征
 D' , 生物特征
采集设备
 r' , r 的返回

$E^{-1}E(r', P', BT') =$
 (r', P', BT')
 $r' = r, D = D', BT' = BT(U)$
如果成立则yes,
否则no

c) 静态生物特征认证协议



客户端

U , 用户



远程主机

r , 随机数
 x , 随机质询序列
 $E()$, 函数

$B', x' \rightarrow BS'(x')$
 r' , r 的返回

\xrightarrow{U}

$\xleftarrow{\{r, x, E()\}}$

$\xrightarrow{E(r', BS'(x'))}$

$\xleftarrow{\text{yes/no}}$

$E^{-1}E(r', BS'(x')) =$
 $(r', BS'(x'))$
从 $(r', BS'(x'))$ 中提取 B'
 $r' = r, x' = x, B' =$
 $B(U)$ 如果成立则
yes, 否则no

d) 动态生物特征认证协议

3.6 用户认证中的安全问题

与任何一种安全服务一样，用户认证，特别是远程用户认证，都会遭受各种各样的攻击。下页图表总结了用户认证中主要的攻击方式，并依据认证手段的类型对其进行了分解

表 3.5 一些潜在攻击、易受攻击的认证手段与典型的防范措施

攻击	认证手段	实例	典型防范措施
客户端攻击	口令	<u>口令猜解;穷举搜索</u>	提高熵。限制尝试次数
	令牌	<u>穷举搜索</u>	提高熵。限制尝试次数
	生物特征	<u>虚假匹配</u>	提高熵。限制尝试次数
主机攻击	口令	窃取明文;字典/穷举搜索	采用散列函数;提高熵;保护口令数据库
	令牌	<u>窃取验证码</u>	采用散列函数;提高熵;保护口令数据库;使用一次性验证码
	生物特征	<u>窃取模版</u>	对采集设备进行认证:挑战-应答协议
窃听、盗窃和复制	口令	<u>肩窥(shoulder surfing)</u>	提高用户的保密意识;管理员及时更换易破解的口令;多因素认证
	令牌	<u>盗窃、伪造硬件</u>	多因素认证;使用防止篡改的令牌
	生物特征	<u>复制(欺骗)生物特征</u>	对采集设备复制检测和认证
重放	口令	重放被窃取的口令响应信息	挑战-应答协议
	令牌	重放被窃取的认证码响应信息	<u>挑战-应答协议;一次性验证码</u>
	生物特征	重放被窃取的生物特征模版响应信息	防止采集设备端的复制操作;通过挑战-应答协议进行设备认证
特洛伊木马	口令, 令牌, 生物特征	安装窃听软件或信息截获设备	客户端认证或者采用安全可信的采集设备
拒绝服务	口令, 令牌, 生物特征	通过多次失败的认证将用户锁定	<u>带令牌的多因素认证</u>

3.6 用户认证中的安全问题



窃听：攻击者试图通过某种涉及用户和攻击者物理接近的攻击来学习口令

主机攻击：直接对存储在主机上的用户文件进行的攻击，主机上存储用户口令、令牌认证码或者生物特征模板

重放：敌手对以前截获到的用户响应消息进行重放的一种攻击

客户端攻击：在不访问远程主机或不干扰通信信道的情况下，敌手试图伪装成一个合法用户来完成用户认证的攻击行为

特洛伊木马攻击：应用或物理设备冒充成认证服务所使用的应用或物理设备来捕获用户口令、验证码或生物特征信息

拒绝服务：试图通过大规模的认证请求使认证服务失效

一些潜在攻击、易受攻击的认证手段与典型的防范措施

攻 击	认 证 手 段	实 例	典型防范措施
客户端攻击	口令	口令猜测、穷举搜索	提高熵、限制尝试次数
	令牌	穷举搜索	提高熵、限制尝试次数、贴身保护令牌实体
	生物特征	虚假匹配	提高熵、限制尝试次数
主机攻击	口令	窃取明文、字典 / 穷举搜索	采用哈希函数、提高熵、保护口令数据库
	令牌	窃取验证码	和口令认证保护一样、使用一次性验证码
	生物特征	窃取模板	对采集设备进行认证；挑战一应答协议
窃听、盗窃和复制	口令	肩窥（shoulder surfing）	提高用户的保密意识、管理员及时更换易破解的口令、多因素认证
	令牌	盗窃、伪造硬件	多因素认证、使用防止篡改的令牌
	生物特征	复制（欺骗）生物特征	对采集设备的复制检测和认证

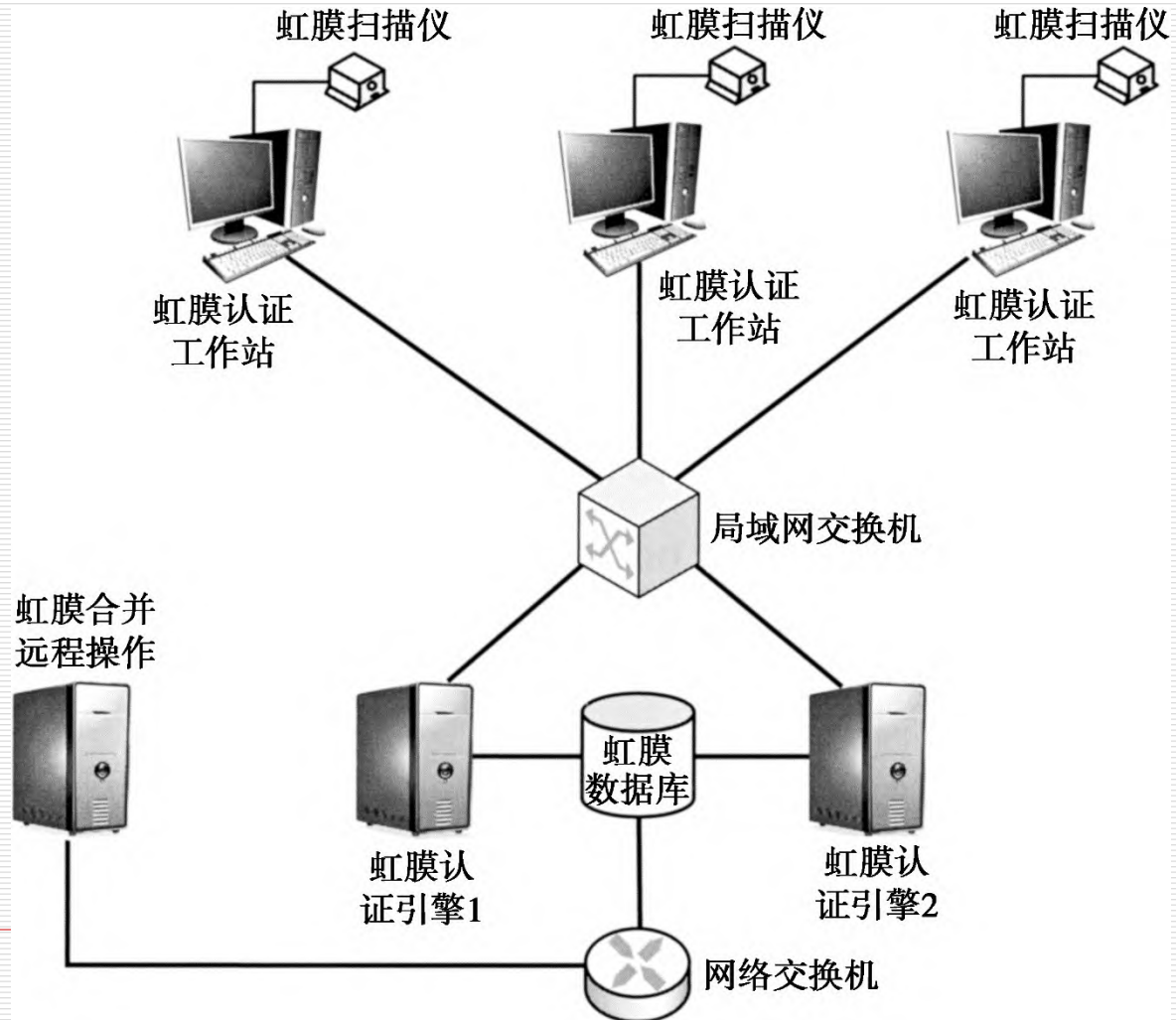
一些潜在攻击、易受攻击的认证手段与典型的防范措施

重放	口令	重放被窃取的口令响应信息	挑战一应答协议
	令牌	重放被窃取的认证码响应信息	挑战一应答协议\一次性验证码
	生物特征	重放被窃取的生物特征模板响应信息	防止采集设备端的复制操作；通过挑战一应答进行设备认证
特洛伊木马	口令、令牌、生物特征	安装窃听软件或信息截获设备	客户端认证或者采用安全可信的采集设备
拒绝服务	口令、令牌、生物特征	通过多次失败的认证将用户锁定	带令牌的多因素认证

3.7 实际应用和案例学习

实际应用：虹膜生物特征认证系统

右图是阿联酋虹膜认证系统的设备总体架构

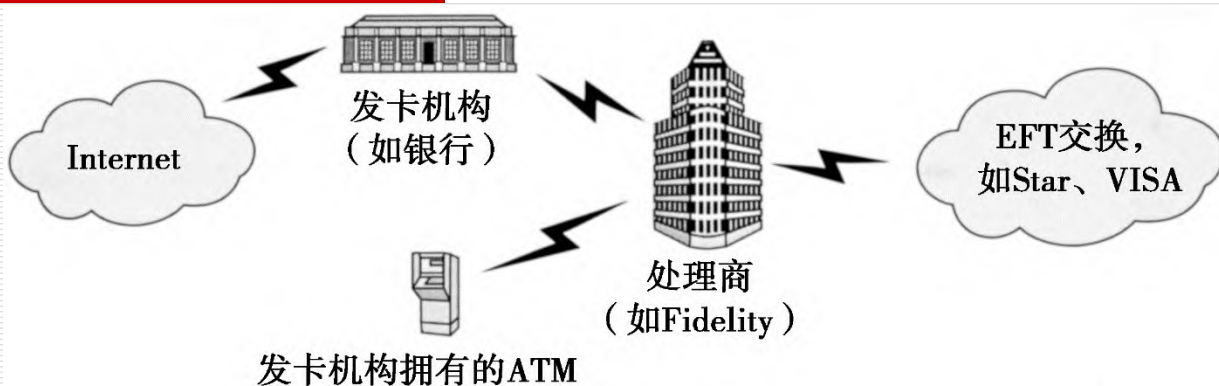


3.7 实际应用和案例学习

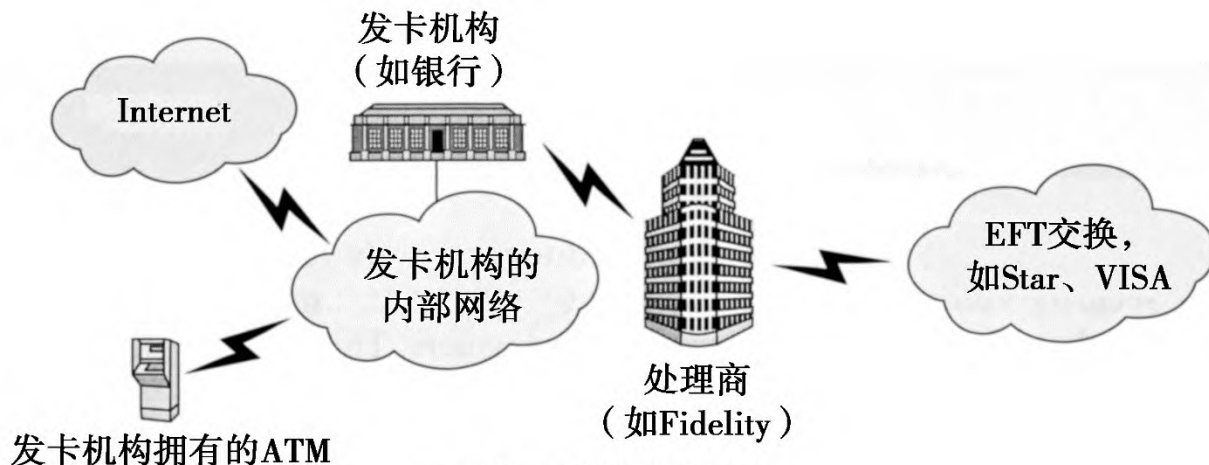
案例学习：ATM系统的安全问题

右图ATM体系结构。很多中小型的借记卡发卡机构与处理商签订合同由其提供核心数据处理和电子资金转账(EFT)服务。银行的ATM机可以直接与处理商或银行连接。

。 2025/8/10



a) 到处理商的点对点连接



b) 到处理商的共享连接

总结



谢谢各位!