

数字摘要

数字摘要：送给我一堆二进制数据，通过我的算法算出的结果一定是固定长度的

比如MD5这种算法，无论输入有多长，输出一定是128个bit

128/8 就是16byte

也就是无论数据多多 多少 输出的都是16B

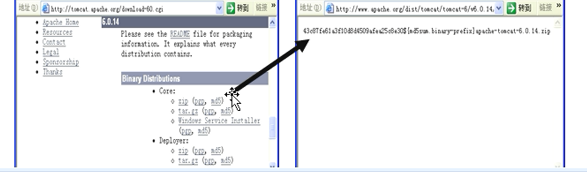
【MD5的长度是128b 也就是16B】

只要原始数据的内容有一点点变化 那么我的输出就会有大大的不同 --- 这种特征就类似于指纹 ----- 结果就相当于数据的指纹 --- 也就是数据的摘要

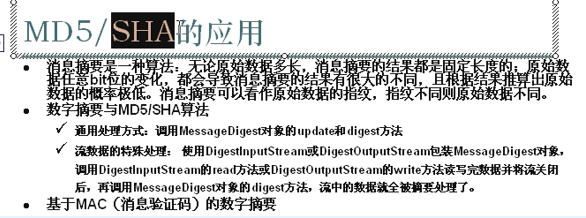
两个2进制的指纹相同是有的 ----- 山大的老师证明了这一点

因为大家没有找到更安全的 所以 MD5还在使用

下载软件的时候 工具旁边都给出一个二进制编码之后的字符串的链接 写的是MD5



这个提示你 下载好这个二进制文件之后，拿工具校验一下这个md5是否一样 表示是否被篡改过 这样可以信任这个二进制数据 ---- 这就是数据的完整性的校验



两种数字摘要的算法是 MD5和SHA

SHA是20B[160b] MD5是16B[128b]

【根据对称加密和非对称加密的套路而言，也是有通用的处理方式+流的处理方式两种

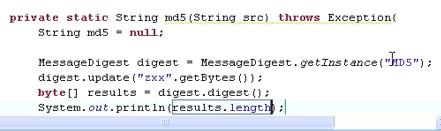
在对称加密和非对称加密中 核心类是Cipher 核心方法是update和doFinal

在数字摘要中 核心类是MessageDigest 核心方式是update和digest】



算法的名称是MD5或者SHA

得到这个digest之后 有update和digest这两个方法



调用一下



现在打印的结果是2进制的字节数组 没办法看 这个时候 要对这个二进制数组进行编码成字符串进行输出

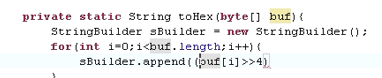
这里面涉及到一个编码 ---- 把二进制的字节数组转换成16进制的字符串

有很多种做法：比如查表法 或者自己写一个方法【实际上 Base64编码什么的都可以的】

**查表法的思路就是：一个十六进制的和一个二进制的数的转化方式 就是 四位二进制能对应一个十六进制的字符**

**一个byte是8bit ---- 那么 就要分别获取到高四位和低四位 分别查表找到对应的十六进制字符 拼接到一起就可以了**

老师这里面自行构建一个toHex方法 使用的思想就是查表法



高位移空的地方就是0



这个字节右移4位

 ---- 原来高四位变成了低四位 高位移空的地方补充0

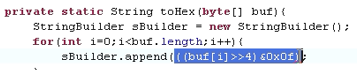
但是 有符号的移动 结果就是 变成了



最终想得到的是

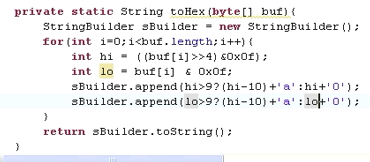
所以 为了确保移位的结果不变 那么就对移动之后的高四位和0进行与操作

这样就能保证移动之后的结果就是高四位的值

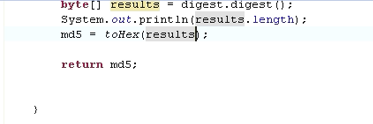


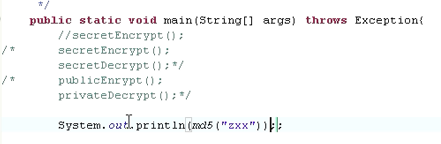
高四位是一个数字





最后打印一下这个结果

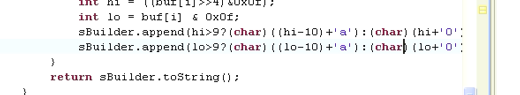




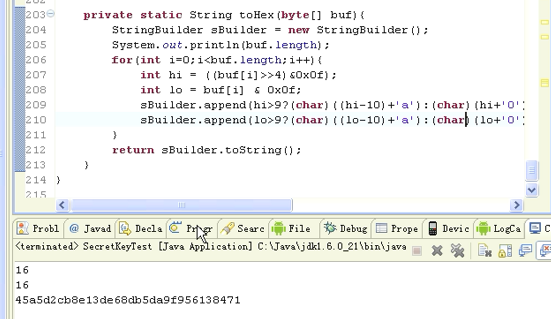


应该是16个字节

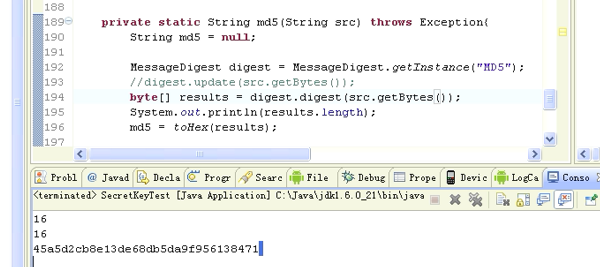
但是 错在哪？



字符和整数做运算 结果会自动提升

、

现在不用update 直接使用digest



计算的结果是一样的

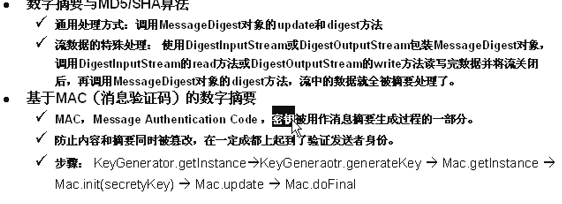
MD5的作用就是求出数据的指纹 有一点微小的变化 结果就大不一样 并且有了指纹 并不能逆推

很多网站 存储的不是密码明文 存储的是密码的MD5 【比非对称加密的好处在于 这个存储的MD5不能逆推 但是 要是非对称加密的密码 万一获取了私钥 还是可以解密的】

\*\*使用的时候 用户输入的密码也同样获取MD5 如果这两个MD5比对相同 就可以了

使用密码的数字摘要 更加安全

\*\*MD5的其他应用

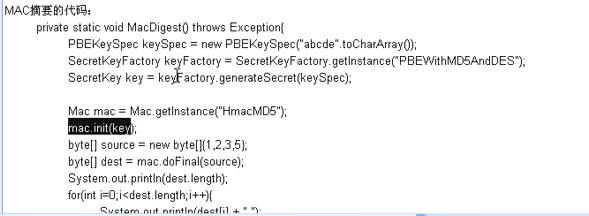


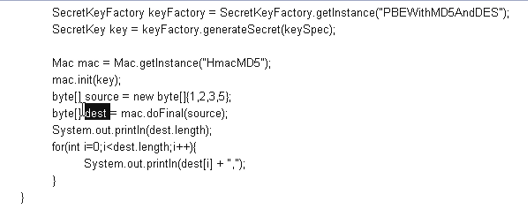
基于消息验证码MAC的数字摘要

也就是收到数据的人并不能确定消息的来源是否正确 只能知道 哦 我通过MD5验证出来这个数据是完整的 没有被篡改

这个时候 做发送人和接收人都有相同的密码 这样

这样就叫做基于消息的验证码

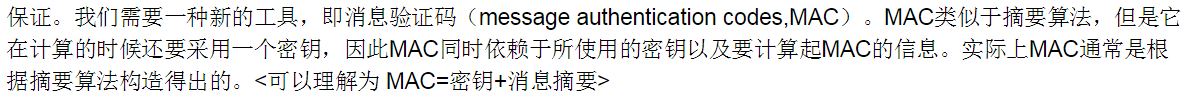




【消息验证码这块 再复习一下马哥那块的内容】

这个是基于消息验证验证码的摘要 --- 防止内容和摘要同时被篡改

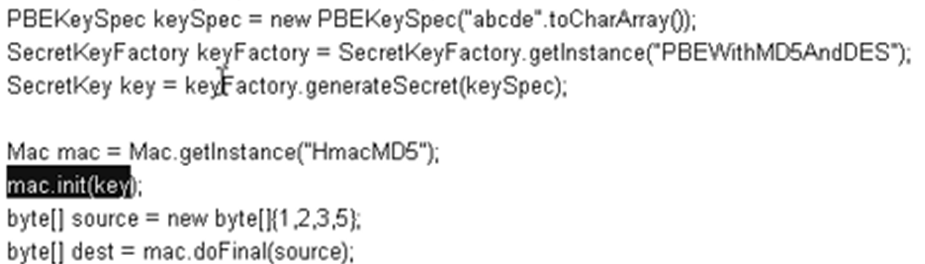
【



所以消息验证码是密钥和消息摘要的合体 ---这也就理解了PPT上面说的：

也就是 密钥被用作消息摘要生成过程的一部分

看代码



这里面doFinal很显然是加密中使用的核心方法

使用的核心对象是Mac getInstance中填写的算法名称是HmacMD5

对于消息摘要来讲 核心类就是MessageDigest 核心方法就是digest 没有密钥参与的过程

这样 综合来讲 上面即用到了加密 又用到了消息摘要

所以 Mac是密钥和消息摘要的合体】

数字签名就相当于对数字摘要按一个手印