漫谈Java加密技术（一）Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/644/2021_2022__E6_BC_AB_E8_B0_88Java_c104_644441.htm 除了DES，我们还知道有DESede（TripleDES，就是3DES）、AES、Blowfish、RC2、RC4（ARCFOUR）等多种对称加密方式，其实现方式大同小异，这里介绍对称加密的另一个算法PBE PBE PBEPassword-based encryption（基于密码加密）。其特点在于口令由用户自己掌管，不借助任何物理媒体；采用随机数（这里我们叫做盐）杂凑多重加密等方法保证数据的安全性。是一种简便的加密方式。 通过java代码实现如下： import java.security.Key. import java.util.Random. import javax.crypto.Cipher. import javax.crypto.SecretKey. import javax.crypto.SecretKeyFactory. import javax.crypto.spec.PBEKeySpec. import javax.crypto.spec.PBEParameterSpec. /** *//** * PBE安全编码组件 * */ public abstract class PBECoder extends Coder { /** *//** * 支持以下任意一种算法 * * gt. * PBEWithMD5AndDES * PBEWithMD5AndTripleDES * PBEWithSHA1AndDESede * PBEWithSHA1AndRC2_40 * gt. */ public static final String ALGORITHM = "PBEWITHMD5andDES". /** *//** * 盐初始化 * * @return * @throws Exception */ public static byte[] initSalt() throws Exception { byte[] salt = new byte[8]. Random random = new Random(). random.nextBytes(salt). return salt. } /** *//** * 转换密钥gt. * * @param password * @return * @throws Exception */ private static Key toKey(String password) throws Exception {

PBEKeySpec keySpec = new PBEKeySpec(password.toCharArray()). SecretKeyFactory keyFactory = SecretKeyFactory.getInstance(ALGORITHM). SecretKey secretKey = keyFactory.generateSecret(keySpec). return secretKey. } /** *//** * 加密 * * @param data * 数据 * @param password * 密码 * @param salt * 盐 * @return * @throws Exception */