Example of How to Create a Secure Channel (that is encrypted and authenticated)

* Given a PlainTCPChannel object *ch* do:
* Create an **encryption** object *authEnc* that has **AuthenticatedEnc** SecurityLevel
* Set *authEnc* with a suitable secret key. (The same key has to be used by both parties using the channel).
* Create an EncryptedChannel object with *ch* and *authEnc*.

//Create a Symmetric Encryption object with AuthenticatedEnc SecurityLevel

ScEncryptThenMac encThenMac = **null**;

//Create encryption object

**try** {

encThenMac = (ScEncryptThenMac) SymmetricEncFactory.getInstance().getObject("EncryptThenMac(CTREncRandomIV(AES),CBCMacPrepending(TripleDES))");

} **catch** (FactoriesException e1) {

e1.printStackTrace();

}

//Create a suitable key parameter spec for the ScEncryptThenMac:

int encKeySize = 128 //The encryption scheme uses AES

int macKeySize = 168 //The authentication scheme uses TripleDes

AuthEncKeyGenParameterSpec params = new AuthEncKeyGenParameterSpec(encKeySize, macKeySize);

//Generate a key and set it. (The same key has to be used by both parties. The application using //the channels is responsible for this).

SecretKey key = encThenMac.generateKey(params);

**try** {

encThenMac.setKey(key);

} **catch** (InvalidKeyException e) {

e.printStackTrace();

}

//Create and EncryptedChannel that has AuthenticatedEnc SecurityLevel:

**EncryptedChannel encChannel = new EncryptedChannel(ch, encThenMac);**