







JENNER

**MOVING DATA BETWEEN PARTIES**











$E_{PK}(R)$



9

5

**Problem:** Data is exchanged between parties.

**Solution:** Use public key cryptography to protect data.

**SENDER**

**STORE & FORWARD**

**RECIPIENT**

R



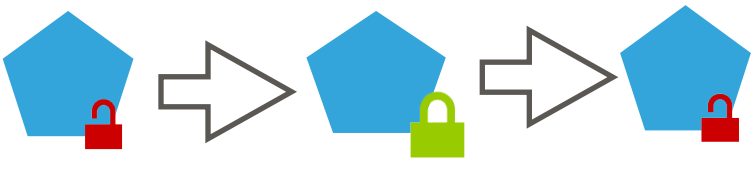
R





Encrypt with public key

Decrypt with private key



**NO SECRETS ON SENDER / STORE & FORWARD SERVER**

**MOVING DATA BETWEEN PARTIES**

# MOVE DATA BETWEEN PARTIES



```
KeyringConfig keyringConfig = KeyringConfigs
    .withKeyRingsFromFiles(
        ".../pubring.gpg",
        ".../secring.gpg",
        withPassword(secKeyRingPassword));

try (
    final InputStream cipherTextStream = Files.newInputStream(sourceFile);

    final OutputStream fileOutput = Files.newOutputStream(destFile);
    final BufferedOutputStream bufferedOut = ...

    final InputStream plaintextStream = BouncyGPG
        .decryptAndVerifyStream()
        .withConfig(keyringConfig)
        .andRequireSignatureFromAllKeys("sender@example.com")
        .fromEncryptedInputStream(cipherTextStream)
    ) {
        Streams.pipeAll(plaintextStream, bufferedOut);
    }
```

## MOVE DATA BETWEEN PARTIES

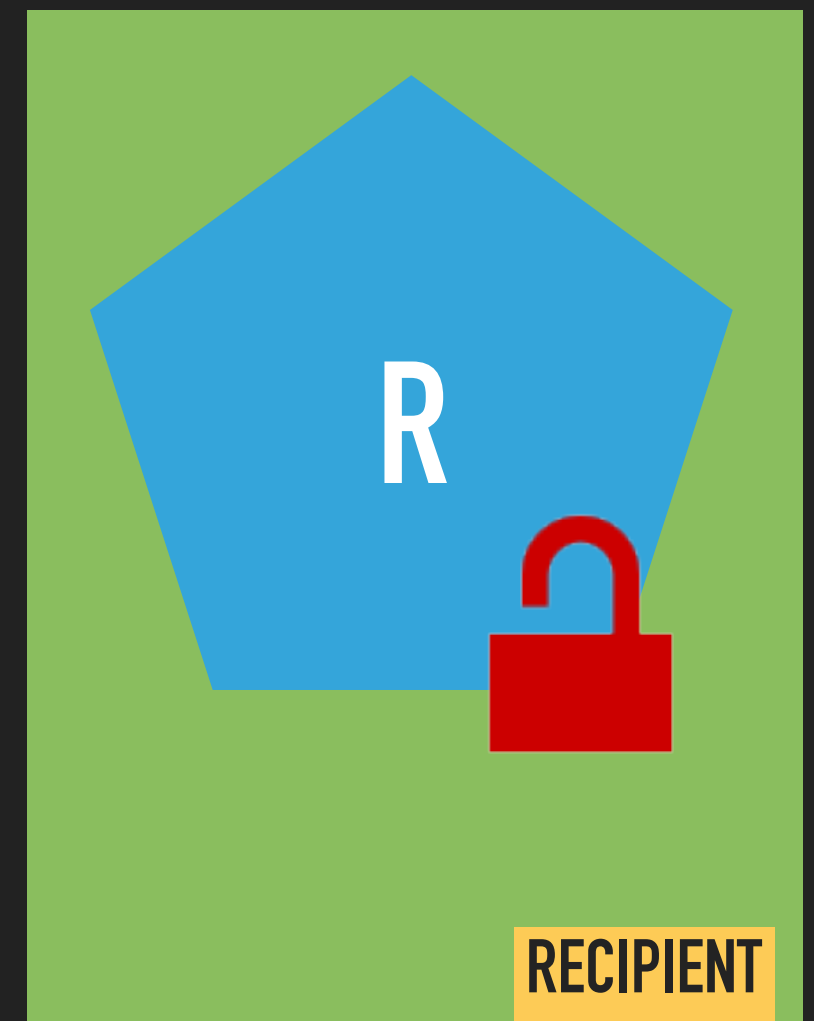


**Problem:** Data is exchanged between parties.

**Solution:** Use public key cryptography to protect data.



Encrypt with public key



Decrypt with private key

NO SECRETS ON SENDER / STORE & FORWARD SERVER