Encrypted Notepad project review

Lubo Obratil, Agusti Bau Pericon, Aobakwe Alloycius Mmokwa

May 19, 2016

1 General project overview

- Java application for reliable text file encryption.
- Application is modified so that encryption password is extracted from the card.
- Communication between card and application is secured with RSA encryption. Protocol is secure against eavesdropping, but can be broken with man-in-the-middle attack. Authors noted this in project specification.
- Encryption key is randomly generated on the card, can not be reset - applet must be reinstalled in order to do that.
- User PIN code length is set to 4 characters and can be incorrectly entered 4 times (5 times total). Applet must be then reinstalled and encryption key is lost.

2 Possible design enhancements

- For communication encryption, 1024 bit long RSA is used. Generally it is recommended to use longer keys for new applications since 1024 bit keys will be probably broken in upcoming years.
- For every file, same key from card is used. During encryption/decryption key is stored in user machine's memory. Should attacker gain access to this memory only once, he then can decrypt every file previously encrypted with this key. This could be mitigated for example with key derivation on the card. However two equal plaintexts won't be transformed into two equal ciphertexts due to application's correct use of initialization vectors.
- User could be able to reset the encryption key on the card in case his old key is compromised without the need to reinstall the applet.

3 Code/style problems

- In both applet and application, PIN, key and RSA sizes are hardcoded on multiple different places. This should be done through constants in case someone would want to change some of these values.
- Smartcard security could be implemented as a optional feature - at this moment application won't start without a card present in system.

4 Bugs/problems

- Applet always expects PIN of length 4. Having PIN code set at 1111 and changing it to single 0 results in PIN being 0111 - remaining bytes are read from ram array which contains PIN from previous verification. After card reconnection (nulling the ram memory), only 0111 will be accepted as PIN, 0 alone no.
- Alternatively, PIN set to e.g. 123456 results in PIN being 1234.
- After starting application and removing card from machine, application continues to run without error. Attempts to encrypt file (required card connection) result in uncaught exceptions from smartcardio.