Lab 10 Simon Razniewski

Cryptography and RSA

1. Matching RSA Keys

Which of the following private RSA keys matches the public RSA key (5,91)?

- (19,91)
- (24,91)
- (29,91)
- (19,81)
- (24,81)
- (29,81)

2. RSA Integrity Game

In this exercise we use RSA for checking the integrity of messages.

1. Generate your own RSA key pair using reasonably small numbers (e.g. using prime numbers between 1 and 30 only)

2. Choose a message string, and compute its Java hashcode ("string".hashCode())

3. Encrypt the hash code with your private key, then choose whether to modify the string or not, and write your name, your public key, the (possibly modified) string and the encrypted hash code in the table on the whiteboard.

4. Verify whether the two classmates in lines "YourLine+1 mod #classmates" and "YourLine+2 mod #classmates" modified their messages or not. Does your assessment match that of others?

Hint: It is ok to use calculators or tools such as WolframAlpha to do the calculations.

3. RSA in Java

1. Download the RSA code used in the lecture from the course homepage, understand it and run it.

On the machine shown on the whiteboard, a RMI object is running under the name "", which implements the following interface:

```
public interface DecoderInterface extends Remote {
	public void decode(PublicKey k, byte[] buf) throws Exception;
}
```

The method takes as parameters an RSA-encrypted message and a public key k, and writes the decrypted method to standard output.

2. Write a client to connect to the server, encrypt one message on your client, and let the server decrypt it.

3. Write your own server that implements this interface.