Blind signatures

Christian Grothoff

Implementing RSA blind signatures

1. Review RSA code
   1. Find rsa.py in the course resources.
   2. Run the script, making sure the python3-pycryptodome dependency is installed on your system.
   3. Review the code.

2. Add blind signatures
   1. Add a function to compute a blinding factor given a public key.
   2. Add a function to apply the FDH to a message and blind the result with the blinding factor, returning a blinded message.
   3. Make a copy of the rsa_sign function and modify it to sign over a blinded message and to return a blind signature.
   4. Add a function to unblind a blinded message given a blinding factor.
   5. Modify the main function to blind a message, use blind signing (instead rsa_sign), and then unblind the blind signature.
   6. Check that the final signature is still valid.

3. Measure performance
   1. How long does each step take? Run each step in a loop 1000 times and measure the time.
   2. What is the most expensive operation?