

CSCI 284 - Cryptography - Spring 2009
George Washington University

Homework 6: due 11 May, 2009, 6 pm on BLACKBOARD

Policy on collaboration: All examinations, papers, and other graded work products and assignments are to be completed in conformance with The George Washington University Code of Academic Integrity. You may *not* discuss this Final among yourselves. Each student is expected to work independently on his or her own Final; you may not collaborate with others, you may not copy one another's assignments, even in part. You may only refer to your class notes, the text book, and to class material such as slides, notes and links provided on the course website. You may not refer to any other material while working on the Final.

All code must run on your hobbes account. The TA and instructor will make no attempt to debug your code, or determine why it does not run. You will be graded on the correctness of your output, and the quality of your code: efficiency and documentation. There will be no exceptions.

Under no circumstances may code be copied from anywhere: classmates, the web, any other source

Any violations will be treated as violations of the Code of Academic Integrity.

Submit all HW in Blackboard by 6 pm on due date. This Final requires a report in addition to code.

Name your reports:

CS284_Final_LASTNAME_FIRSTNAME.doc or .pdf or

CS162_Final_LASTNAME_FIRSTNAME.doc or .pdf

NOTE that you may not submit a report in the new docx format of Microsoft Word; if you prepare such a file, save it in an older Word format.

Archive the code with the report and name the compressed file similarly, with extensions .tar or .rar or .zip

A. For this part of the assignment, you will write code for encryption **ONLY** (no decryption) using elliptic curves.

Write your own code for addition of two points in a given elliptic curve group over \mathbb{Z}_p for p a prime of size at most 16 bits. You may define your own representation of a point on the curve, and you will encrypt a point on the curve to get a pair of points, using a given public key value. You should use efficient exponentiation/addition in the group. You may use a pseudorandom number generator from a library to generate the random value required for the El Gamal style encryption.

B. For this part of the assignment, you will compare the difficulty of breaking a block cipher of block size n bits using linear or differential cryptanalysis, with the difficulty of the discrete log problem of size n bits over the elliptic curve group, and the difficulty of discrete log of size n bits over \mathbb{Z}_p^* . For this part of the problem, refer to your book and provide the answer in your own words. You do not need to describe any of the attacks in detail; in particular, you do not need to understand the details of how the discrete log attacks on El Gamal or elliptic curve encryption work.

You will submit: For Part A: (a) the code (b) a README file explaining how the code may be compiled and run (c) two input-output file pairs for your code, both for encryption; the TA/instructor will use the input file as a template to change the variables and test correctness of you code ON HOBBS. For Part B: (d) a report in .doc

pr .pdf formats, note that .docx is not an acceptable format, such files should be saved in an older Word format before submission. Archive all four parts into a .tar, .rar or .zip file.