TECHNISCHE UNIVERSITEIT EINDHOVEN

Faculty of Mathematics and Computer Science
Exam Cryptology/Cryptography I/Coding and Crypto, Tuesday 19 January 2016

Name :

TU/e student number :

Exercise	1	2	3	4	5	6	total
points							

Notes: Please hand in *this sheet* at the end of the exam. You may keep the sheet with the exercises.
This exam consists of 6 exercises. You have from 13:30 – 16:30 to solve them. You can reach 100 points.
Make sure to justify your answers in detail and to give clear arguments. Document all steps, in particular of algorithms; it is not sufficient to state the correct result without the explanation. If the problem requires usage of a particular algorithm other solutions will not be accepted even if they give the correct result.
All answers must be submitted on TU/e letterhead; should you require more sheets ask the proctor. State your name on every sheet.
Do not write in red or with a pencil.
You are allowed to use any books and notes, e.g. your homework. You are not allowed to use the textbooks of your colleagues.
You are allowed to use a calculator without networking abilities. Usage of laptops and cell phones is forbidden.

1. This problem is about RSA encryption.

   (a) Alice's chooses $p = 443$ and $q = 347$. Compute Alice's public key (n, e), using $e = 3$, and the matching private key d. | 2 points |

   (b) Bob uses public key $(n, e) = (120781, 3)$ and secret key $d = 80043$. He receives ciphertext $c = 38191$.
   Decrypt the ciphertext. | 2 points |

2. This problem is about the Diffie-Hellman key exchange. The system uses the multiplicative group $\mathbb{F}_p^*$ modulo the prime $p = 991$. The element $g = 7 \in \mathbb{F}_{991}^*$ has order 990 and is thus a generator of the full multiplicative group.

   (a) Alice's chooses $a = 123$ as her secret key. Compute Alice's public key. | 1 point |

   (b) Alice receives $h_b = 428$ from Bob as his Diffie-Hellman keyshare. Compute the key shared between Alice and Bob, using Alice's secret key from the first part of this exercise. | 2 points |

3. This exercise is about computing discrete logarithms in the multiplicative group of $\mathbb{F}_p$ for some prime p.

   (a) Let $p = 991$ and note that $p - 1 = 2 \cdot 3^2 \cdot 5 \cdot 11$. A generator of $\mathbb{F}_p^*$ is $g = 7$. Bob's public key is $h_b = g^b = 428$.
   Use the Pohlig-Hellman attack to compute Bob's secret key b, i.e., compute b modulo 2, modulo 3^2, modulo 5, and modulo 11 and combine these results to compute Bob's secret key b modulo 990. Verify your answer, i.e., compute g^b. | 20 points |

   (b) Let $p = 367$. The element $g = 8$ has order $\ell = 61$. Charlie uses $\mathbb{F}_{367}^*$ with this generator g and you observe him sending $h_c = g^c = 124$. Use the Baby-Step Giant-Step attack in the group generated by g (of size $\ell = 61$) to compute c. | 10 points |

4. This exercise is about factoring $n = 120781$.

   (a) Use Pollard's rho method for factorization to find a factor of 120781 with iteration function $x_{i+1} = x_i^2 + 1$ and Floyd's cycle finding method, i.e. after each increment in i compute $\gcd(x_{2i} - x_i, 120781)$ until a non-trivial gcd is found. Assume you have made it till $x_8 = 71576$ and $x_{16} = 72367$. Continue from here to find a factor of n.

   12 points

   (b) Use the $p-1$ method to factor the RSA modulus $n = 120781$ with basis $a = 16$ and exponent $s = \text{lcm}\{1, 2, 3, 4, 5, 6, 7, 8\}$. Make sure to determine both factors of n.

   6 points

5. (a) Find all affine points on the Edwards curve $x^2 + y^2 = 1 + 6x^2y^2$ over $\mathbb{F}_{11}$.

   8 points

   (b) Verify that $P = (2, 6)$ is on the curve. Compute the order of P.
   Hint: You may use information learned about the order of points on Edwards curves.

   12 points

   (c) Translate the curve **and** P to Montgomery form

   $$Bv^2 = u^3 + Au^2 + u,$$

   i.e. compute A, B, and the resulting point P'.
   Verify that the resulting point P' is on the Montgomery curve.

   6 points

6. This exercise is about RSA signatures and a way that protocols might not get authenticity from it. The key set up for RSA signatures works similar to that in RSA encryption: Let p and q be large primes, let e be an integer coprime to $(p-1)(q-1)$, put $n = pq$, compute $\varphi(n) = (p-1)(q-1)$ and compute $d \equiv e^{-1} \bmod \varphi(n)$. The public key is (n, e), the private key is d.

   To sign message $m \in \mathbb{Z}/n$, compute $s \equiv m^d \bmod n$.

   To verify a signature s under public key (n, e), compute $m' \equiv s^e \bmod n$. The signature is valid if $m' = m$.

   Note: This is a schoolbook version of the system, in real applications the message m is replaced by its hash $h(m)$ and some padding and randomization. However, the attack you are finding in this exercise will work just the same.

   (a) Verify that $s = 52792$ is a valid signature on $m = 234567$ with the key $(n, e) = (240067, 3)$. | 1 point |

   (b) Let's Encrypt is a certificate authority (CA), i.e. an organization that issues certificates for web sites. A certificate contains information m about a web site and a signature on m by the CA.

   Let's Encrypt aims to make the process of obtaining certificates as easy and automated as possible. The following description matches an early version of the ACME system by Let's Encrypt, the mistake you will find has been fixed before the system was rolled out.

   To request a certificate, the owner of `example.com` collects the necessary information in m, creates an RSA key pair (n, e) and d, and computes the signature $s \equiv m^d \bmod n$ on m under this key. To prove ownership of the domain he posts the same s on the server `example.com` at a specified spot. He then sends $m, (n, e)$, and s to the server with a request to issue a certificate on m for `example.com`.

   The server checks that n has the expected bit length, $s^e \equiv m \bmod n$ and that s in the message matches the s at the specified spot on `example.com`.

   Assume that `example.com` has already posted a signature s, e.g. from some previous certificate request. Show how Eve can choose a key (n', e') so that she can get a certificate on a message m' of her choosing.

Hint: The signature posted on the server does not include the key under which it should be verified. Here you can assume that the CA does not insist on $e > 1$ and (of course) does not check that n' is actually an RSA modulus. | 8 points |

(c) Find a key (n', e') so that the signature $s = 52792$ is valid on $m' = 12345$. | 2 points |

(d) Now assume that the CA insists on $e > 1$. How can Eve choose/compute (n', e') now? Note that you are not expected to do the computation; just describe how you would obtain (n', e') now. | 8 points |