Technische Universität München (I7)
Dr. M. Luttenberger / S. Jaax

Winter term 2018/2019
2018/10/29

Cryptography – Homework 1

Discussed on 6/11/2018.

Exercise 1.1 Warmup

Consider a CPU with n-bit registers. The n-bit strings $\{0,1\}^n$ are naturally interpreted as the integers $\mathbb{Z}_{2^n} = \{0, 1, 2, \ldots, 2^n - 1\}$ using e.g. either "most significant bit first" or "least significant bit first" interpretation (it is not important which one is used). Addition and multiplication on $\mathbb{Z}_{2^n}$ are defined as usual modulo 2^n, e.g. $1 + (2^n - 1) = 0$ and $2 \cdot 2^{n-1} = 0$ ("overflow"). We define some sets of functions over $\mathbb{Z}_{2^n}$:

- $M_n := \{f \colon \mathbb{Z}_{2^n} \to \mathbb{Z}_{2^n}\}$ the set of all maps from $\mathbb{Z}_{2^n}$ to $\mathbb{Z}_{2^n}$.
- $P_n := \{f \in M_n \mid f_n \text{is a bijection}\}$ the set of all bijections/permutations.
- $A_n = \{f \in M_n \mid \exists a, b \in \mathbb{Z}_{2^n} \forall x \in \mathbb{Z}_{2^n} \colon f(x) = a \cdot x + b\}$ the set of "affine" functions.

The (mono-alphabetic) substitution cipher generalizes to $\mathbb{Z}_{2^n}$ by taking a subset of P_n as key space with encryption defined by

$$\mathsf{Enc}_f(m_1||m_2||\ldots||m_l) = f(m_1)||f(m_2)||\ldots||f(m_l) \text{ and } \mathsf{Dec}_f(c_1||c_2||\ldots||c_l) := f^{-1}(c_1)||f^{-1}(c_2)||\ldots f^{-1}(c_l)$$

for $f \in P_n$ and $m_i, c_i \in \mathbb{Z}_{2^n}$.

(a) Compute $|M_n|$, $|P_n|$, $|A_n|$, and $|A_n \cap P_n|$.

Hint: Why does it suffice to discuss whether $f(x) = ax + b$ is injective or not?

(b) How would you store and compute $f(\cdot)$ and $f^{-1}(\cdot)$ for $f \in P_n$ resp. $f \in P_n \cap A_n$?

(c) Show that the substitution cipher is perfectly secret if (1) we restrict the message space to $\mathbb{Z}_{2^n}$ and (2) we choose the key uniformly at random from P_n resp. from $A_n \cap P_n$.

(d) Why is the previous result no contradiction to the fact that frequency analysis can be used to break the classical mono-alphabetic substitution cipher?

Exercise 1.2 Generating large odd random numbers

Consider the following algorithm $\mathcal{A}$

- *Input*: Natural number $n > 1$.
- *Output*: A random odd number in $[2^{n-1}, 2^n - 1]$.
- *Algorithm*:
  (a) Set $x_0 = 1$, $x_{n-1} = 1$.
  (b) For i in 1 to $n - 2$: Set $x_i = \text{fair\_coin}()$.
  (c) Let x be the integer represented by the bit string $x_0 x_1 \ldots x_{n-1}$ with the least-significant bit first x_0.

(a) Determine the distribution of the output of $\mathcal{A}(n)$.

(b) What is (a lower bound on) the probability that the output number is a prime?

Hint: Use that within $[2^{n-1}, 2^n - 1]$ there are at least $\frac{2^{n-1}}{n}$ primes if n is sufficiently large ($n \geq 20$ suffices).

Exercise 1.3 **Perfect Secrecy**

(a) Is the following statement true? Every encryption scheme for which the size of the key space equals the size of the message space, and for which the key is chosen uniformly from the key space, is perfectly secret.

(b) Show that every encryption scheme can be transformed into an equivalent encryption scheme defined over a fixed-length key space. More precisely, show that from any encryption scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ we can build an ES $\mathcal{E}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ such that

1. Gen' simply generates a "truly random" (short for "chosen uniformly at random") key $k' \stackrel{u}{\in} \{0,1\}^T$ for some sufficiently large T. (T should bound the running time of Gen.)
2. The message and ciphertext spaces of $\mathcal{E}$ and $\mathcal{E}'$ are the same.
3. For every message m and every ciphertext c we have $\Pr_{k:\stackrel{r}{=}\mathsf{Gen}()}[\mathsf{Enc}_k(m) = c] = \Pr_{k':\stackrel{r}{=}\mathsf{Gen}'()}\left[\mathsf{Enc}'_{k'}(m) = c\right]$, i.e. the probabilitiy to encrypt m to c is in both ES the same when the key is generated by Gen resp. Gen'.
4. The running times of Enc' and Dec' are bounded by the running times of $\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}$.

Exercise 1.4 **Eavesdropping game**

The **q-eavesdropping game** is the version of the eavesdropping game where $\mathcal{A}$ may pass to the oracle two sequences. $m_0 = (m_0^{(1)}, \ldots, m_0^{(q)})$ and $m_1 = (m_1^{(1)}, \ldots, m_1^{(q)})$, and the oracle returns $c = (c^{(1)}, \ldots, c^{(q)})$ where $c^{(i)} :\stackrel{r}{=} \mathsf{Enc}_k(m_b^{(i)})$ and $m_b^{(i)} \in \mathcal{M}$.

We call an ES *perfectly secure under q encryptions (using the same key)* if there is no $\mathcal{A}$ which can win the q-eavesdropping game with probability strictly greater than $\frac{1}{2}$.

(a) Show that the one-time pad is not perfectly secure under $q = 2$ encryptions.

(b) Propose a q-time pad which is perfectly secure under q encryptions. Does your scheme remember some "state" between two en/decryptions?

Remark: If you want to know more about "classical" ciphers and how to break them:

- `http://www.mysterytwisterc3.org` A very nice collection of flash- animated crypto-puzzles (Ceasar, Substitution, Permutation, Knapsack,...) but also programming challenges and real ciphers.
- *H.F. Gaines, Cryptanalysis, a study of ciphers and their solution. Dover, 1956.*
- *Simon Singh, The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography. Anchor, 2000.*

Technische Universität München (I7)
Dr. M. Luttenberger / S. Jaax

Winter term 2018/2019
2018/11/13

Cryptography – Homework 2

Discussed on Tuesday, 20th November, 2018.

Exercise 2.1 Negligible Functions

(a) Show that the following two definitions are equivalent:

i) $\varepsilon(n)$ is negligible if and only if $\forall a \in \mathbb{N} \exists N \in \mathbb{N} \forall n > N : \varepsilon(n) < n^{-a}$.

ii) $\varepsilon(n)$ is negligible if and only if $\forall$polynomials q $\exists N \in \mathbb{N} \forall n > N : \varepsilon(n) < \frac{1}{|q(n)|}$

(b) Let $\varepsilon_1 : \mathbb{N} \to \mathbb{R}^+, \varepsilon_2 : \mathbb{N} \to \mathbb{R}^+$ be negligible functions and let $p : \mathbb{N} \to \mathbb{R}^+$ be a polynomial in $\mathbb{R}$. Show that $f : \mathbb{N} \to \mathbb{R}^+$ with $f(n) = \varepsilon_1(n) + \varepsilon_2(n)$ and $g : \mathbb{N} \to \mathbb{R}^+$ with $g(n) = p(n) \cdot \varepsilon_1(n)$ are also negligible functions.

(c) Prove: If $\varepsilon_0, \varepsilon_1, \ldots$ is a family of negligible functions $\varepsilon_i \colon \mathbb{N} \to \mathbb{R}^+$ and $p \colon \mathbb{N} \to \mathbb{N}$ is a polynomial, then f with $f(n) := \sum_{i=0}^{n} \varepsilon_i(n)$ need not be negligible anymore.

Exercise 2.2 Pseudorandom Generators

- Let $f \colon \{0,1\}^* \to \{0,1\}$ be any DPT-computable function.

  Show that $G_f(x) = x||f(x)$ is no PRG.

- Use a similar argument to show that also the follwing is not a PRG:

  Let $m \in \mathbb{N}$ and $a, c \in \mathbb{Z}_m$.

  – For simplicity, let $m = 2^n$ so that we may identify $\mathbb{Z}_m$ with $\{0,1\}^n$.

  For $x \in \{0,1\}^n$, let $f(x) = (a \cdot x + c) \bmod m$, and $G(x, 1^{n \cdot s}) = f(x)||f(f(x))|| \ldots ||f^s(x)$.

Exercise 2.3 Pseudorandom Generators II

Let G be a PRG of stretch $l(n) = 2n$.

(a) Show that there exists an *exponential time* distinguisher $\mathcal{D}$ with:

$$\left| \Pr_{x \overset{u}{\in} \{0,1\}^n} [\mathcal{D}(1^n, G(x)) = 1] - \Pr_{y \overset{u}{\in} \{0,1\}^{2n}} [\mathcal{D}(1^n, y) = 1] \right| \geq 1 - 2^{-n}$$

(b) Determine the success probability of the following $\mathcal{D}$:

- Input: $y \in \{0,1\}^{l(n)}$ and 1^n.
- Generate $x' \overset{u}{\in} \{0,1\}^n$.
- Compute $y' = G(x')$.
- Return 1 if $y = y'$; else return 0.

Exercise 2.4

(a) Show that PRFs with $l_{\text{out}}(n) \cdot 2^{l_{\text{in}}(n)} \leq n$ exist (unconditionally!).

(b) Let G be a PRG of stretch $l_G(n) = 2n$.

Split $G(k) =: G_0(k)||G_1(k)$ into two n bit strings.

Set $F_k^{(1)}(0) := G_0(k)$ and $F_k^{(1)}(1) := G_1(k)$. Show: $F^{(1)}$ is a PRF with $l_{\text{in}}(n) = 1$ and $l_{\text{out}}(n) = n$.

Technische Universität München (I7)
Dr. M. Luttenberger / S. Jaax

Winter term 2018/2019
2018/11/23

Cryptography – Homework 3

Discussed on Wednesday, 28th of November, 2018.

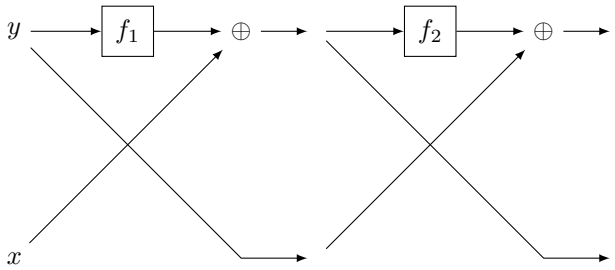
Exercise 3.1

Let F be a PRP.

(a) Show that F-rCBC is not CCA-secure.

(b) Show that F-CBC-CIV (with chained IV—see lecture slides) is *not* CPA-secure.

Exercise 3.2



Let $f : \{0,1\}^* \to \{0,1\}^*$ be some function s.t. $|f(x)| = |x|$ for all $x \in \{0,1\}^*$. A *single-round Feistel network* FN_f is defined by

$$\mathsf{FN}_f(x||y) := y||x \oplus f(y) \text{ for all } x, y \in \{0,1\}^* \text{ with } |x| = |y|\,.$$

Similarly, given functions $f_1, \ldots, f_j$ a *j-round Feistel network* is inductively defined by

$$\mathsf{FN}_{f_1,f_2,\ldots,f_j}(x||y) := \mathsf{FN}_{f_j}(\mathsf{FN}_{f_1,f_2\ldots,f_{j-1}}(x||y))$$

(a) Show that independent of the choice of $f_1, \ldots, f_j$ the function $\mathsf{FN}_{f_1,\ldots,f_j}$ is invertible if $f_1, \ldots, f_j$ are known.

(b) Let F be a PRF of key and block length n and $P_{k_1,k_2}(x||y) := \mathrm{FN}_{F_{k_1},F_{k_2}}(x||y)$ be a two-round Feistel network using F.

i) Compute $P_{k_1,k_2}(0^n||y)$ and $P_{k_1,k_2}(F_{k_1}(0^n) \oplus z||0^n)$.

ii) Show that PPT-Eve can compute $P^{-1}_{k_1,k_2}$ when given oracle access to P_{k_1,k_2}.

(c) Is $\mathrm{FN}_{F_{k_1},F_{k_2},F_{k_3}}$ with three independent keys $k_1, k_2, k_3 \overset{u}{\in} \{0,1\}^n$ a PRP? Is it a PRF? (y/n)

Exercise 3.3 MAC or no MAC?

(a) Does rOFB mode yield a secure MAC?

(b) Show that if the IV in the CBC-MAC-Algorithm is not fixed (but chosen randomly and pre-pended to the CBC-output), the MAC becomes insecure.

Exercise 3.4 MACs using hash-functions done wrong

Before NMAC and HMAC, several ad-hoc solutions for constructing MACs were used. For instance, given a (hash) function $H\colon \{0,1\}^* \to \{0,1\}^l$, the tag was defined to be $\mathsf{Mac}_k(m) := H(k||m)$, i.e. the outer encryption used in NMAC and HMAC is missing.

Assume a PRF F with (for simplicity) $n = l_{\text{in}}(n) = l_{\text{out}}(n)$. Using the padding function $\mathsf{pad}(m) := m||10^p||\lfloor|m|\rceil$, set $\mathsf{Mac}_k(m) := H(k||m) := F^*_k(\mathsf{pad}(m))$ for $k \in \{0,1\}^n$.

Show that $\mathsf{Mac}_k(m)$ is not secure.

Hint: Recall that the outer encryption used by NMAC and HMAC is to restrict the adversary to prefix-free queries.

Exercise 3.5

Let F be some secure block cipher with key and block length n (think of AES-128).

Consider the following deterministic MAC:

- Gen: as usual, in input 1^n, output $k \xleftarrow{u} \{0,1\}^n$.
- Mac: given $m \in \{0,1\}^+$ and k,

  first pad m to a multiple of n by appending a minimial number of 0,

  then break the padded message into n-bit blocks $m^{(i)}$.

  Starting with $k^{(0)} := 0^n$, compute $k^{(i)} = F_{k^{(i-1)}}(m^{(i)})$ for i from 1 to d where $d = \frac{|m|}{n}$.

  Finally, output $t := F_{k^{(n)}}(k)$.

  (Draw a picture! Note that the key is appended in this case.)
- Vrf: given m, t, and k, check that $\mathsf{Mac}_k(m) = t$.

Is this MAC secure?

Technische Universität München (I7)
Dr. M. Luttenberger / S. Jaax

Winter term 2017/2018
2019/01/05

Cryptography – Homework 5

Discussed on Tuesday, 8th January, 2018.

Exercise 5.1 Warm-up II

(a) $\mathbb{Z}_{54}^*$ is a cyclic group. What is its order? Determine a generator g of $\mathbb{Z}_{54}^*$ and solve $7^x \equiv 19 \bmod 54$ for x (i.e. compute the discrete logarithm of 19 to the base 7).

(b) Let $\pi = [23154] \in S_5$. Determine π^{842}.

(c) Compute the greatest common divisor d of 234, 762, and 139. Furthermore determine $r, s, t \in \mathbb{Z}$ such that $r \cdot 234 + s \cdot 762 + t \cdot 139 = d$.

(d) Solve for X: $X \equiv 2 \bmod 5 \wedge X \equiv 5 \bmod 11 \wedge X \equiv 1 \bmod 21$.

(e) We have seen that it is fairly easy to compute square roots modulo certain primes. Let $p = 47$. Solve for X: $X^2 \equiv 6 \bmod 47$ (*Hint*: $p \equiv 3 \bmod 4$ since p is a safe prime).

Exercise 5.2 Applications of the CRT

The Chinese remainder theorem states that $\mathbb{Z}_{NM}^* \simeq Z_N^* \times Z_M^*$ for $\mathsf{gcd}(M, N) = 1$. An important practical consequence of the CRT is the possibility to do computations with smaller numbers and thus to reduce the running time of algorithms in certain situations (e.g. when decrypting an RSA encrypted message).

(a) Compute all solutions to the quadratic equation $X^2 \equiv 118 \bmod 221$

(b) Find all solutions to $X^2 \equiv 1 \bmod 175$

(c) For $p = 13$ and $q = 19$ set $n = pq$. Let $e = 127$ and $d = e^{-1} \bmod \varphi(n)$. Given $c := 197 = m^e \bmod n$. Determine d and then m via $m = c^d \bmod n$. (this is Textbook-RSA with very small primes :))

Exercise 5.3 φ vs. λ

Let $p = 7$, $q = 11$, and $N = pq$. Furthermore set $e = 7$.

(a) Compute $\varphi(N)$ and $\lambda(N)$.

(b) Compute $e^{-1} \bmod \varphi(N)$ and $e^{-1} \bmod \lambda(N)$.

(c) Compute $\mathbb{Z}_{\varphi(N)}^*$ and $\mathbb{Z}_{\lambda(N)}^*$. How are both groups related to each other?

(d) Prove that for $N = pq$ with $p \neq q$ prime and for any $e \in \mathbb{N}$, $\mathsf{gcd}(e, \varphi(N)) = 1$ if and only if $\mathsf{gcd}(e, \lambda(N)) = 1$.

Exercise 5.4

(a) Let $\langle \mathbb{G}, \cdot, 1 \rangle$ be a finite cyclic group with generator g. Denote by $q := |\mathbb{G}|$ the order of $\mathbb{G}$.

Assume $q = d \cdot m$ is a composite and let d be a non-trivial factor of q. Let $y \in \mathbb{G}$.

Show:

If $k \in \mathbb{N}$ satisfies $g^k = y$ in $\mathbb{G}$, then $(k \bmod d)$ is the unique solution of the following problem:

$$\text{Determine } x \in \mathbb{Z}_d \text{ such that } (g^m)^x = y^m \text{ in } \mathbb{G}.$$

(b) Given are the prime 89 and the generator 3 of $\langle \mathbb{Z}_{89}^*, \cdot, 1 \rangle$.

Your task is to determine $k \in \mathbb{Z}$ such that $3^k \equiv 86 \pmod{89}$. Proceed as follows:

i) Using the preceding exercise, first determine k modulo 11.

ii) Someone tells you that $k \equiv 5 \pmod 8$. Determine k.

Exercise 5.5 **Computing Discrete Logarithms**

Let p be prime. Assume there is a generator $g \in \mathbb{Z}_p^*$ such that computing the discrete logarithm in $\mathbb{Z}_p^*$ with respect to g can be done efficiently. Show that then the discrete logarithm with respect to any other generator g' of $\mathbb{Z}_p^*$ can also be computed efficiently.

Technische Universität München (I7)
Dr. M. Luttenberger / S. Jaax

Winter term 2017/2018
2019/01/25

Cryptography – Homework 5

Discussed on **Tuesday**, 29th January, 2019.

Exercise 5.1

Suppose Eve is given (N, e) the public key of an RSA cryptosystem. Show that she can efficiently factor $N = pq$ in each of the following cases:

(a) she can efficiently compute $\varphi(N)$ (Hint: What are the roots of the polynomial $X^2 - X(N + 1 - \varphi(N)) + N$?).

(b) she can efficiently compute an $0 \neq x \in \mathbb{Z}_N \setminus \mathbb{Z}_N^*$.

Exercise 5.2 Collision resistance of the DLP-CCF

In the lecture we have seen the proof sketch that the DLP-CCF is collision-resistant, given that the DLP relative to $\mathsf{Gen}\mathbb{QR}_{\text{safe}}$ is hard, i.e.:

Assuming we have a collision attack $\mathcal{A}$ on the DLP-CCF we build from it the following algorithm $\mathcal{B}$ for computing discrete logarithms. Define $\mathcal{B}$ as:

- Input: (p, q, g) and $r = g^x \bmod p$ for some secret $x \overset{u}{\in} \mathbb{Z}_q$.
- If $r = 1$, output $x = 0$.
- Otherwise, pass (p, q, g, r) to $\mathcal{A}$ to obtain $(a, b) \neq (u, v)$.
- If $h_I(a, b) \neq h_I(u, v)$, output any element in $\mathbb{Z}_q$.
- Otherwise return $(a - u) \cdot (v - b)^{-1} \mod q$.

Complete the proof bz determining the probability that $\mathcal{B}$ succeeds in computing a logarithm of r modulo p. Why is it important that q is prime?

Exercise 5.3

Let f be a OWP with hardcore bit $\mathsf{hc}(x)$. Show that $G_l(x) := f^l(x)||\mathsf{BM}^l(x)$ is a PRG of *fixed stretch* for every fixed l polynomial in n.

- Discuss the advantages/disadvantages of outputting also $f^l(x)$.
- In particular, consider the case when a TDP is used for f and the resulting PRG is used within the prOTP.

Technische Universität München (I7)
Dr. M. Luttenberger / S. Jaax

Winter term 2017/2018
2019/02/01

Cryptography – Homework 6

Discussed on 06.02.2019.

Exercise 6.1 RSA

For this exercise we will use the notation (for elements of the RSA-PKES and PSS$[k, l]$) introduced in the lecture. We will use msbf-representation throughout the exercise.

(a) We choose two primes $p = 17$ and $q = 19$, select $e = 5$ and use them for instantiating the RSA encryption scheme. Then $N = pq = 323 = (101000011)_2$ in MSBF, i.e., N's bit-length is 9. We apply the OAEP padding scheme to the messages we want to encrypt using the following (very simple and unrealistic) instantiation:

- (Unpadded) messages m are 3-bit strings,
- $k_1 = 2$ and $k_0 = 4$,
- $G(x_1x_2x_3x_4) = x_1x_2x_3x_4x_1$, for $x_i \in \{0, 1\}$ for $1 \leq i \leq 4$,
- $H(x_1x_2x_3x_4x_5) = (x_1 \oplus x_2)(x_2 \oplus x_3)(x_3 \oplus x_4)(x_4 \oplus x_5)$, for $x_i \in \{0, 1\}$ for $1 \leq i \leq 5$.

Let us assume we receive the ciphertext 116. Compute d and the original (3-bit) message. Apply the Chinese Remainder Theorem in your computation.

(b) Show that decoding/encoding works regardless of the choice for G and H in the OAEP padding scheme.

(c) We assumed for the RSA scheme that the plaintext message m is an element of $\mathbb{Z}_N^*$. Show that encryption and decryption is also possible if $m \in \mathbb{Z}_N \setminus \mathbb{Z}_N^*$.

(d) We want to sign a message using the PSS$[k, l]$ scheme. We reuse the values from the previous exercise for N, d and e, and add the following parameters:

- $n = 10$, $k = l = 3$,
- $h(x_1 \dots x_s) := x_1x_2x_3$,
- $g(x_1x_2x_3) := x_1x_2x_3x_1x_2x_3$.

Compute $\mathsf{Sgn}_{(323,d)}(0110)$. Assume hereby that r was chosen as the bitstring 101.

Exercise 6.2 Attacks on Textbook-RSA

In this exercise we will study Textbook-RSA and see why it is insecure and why we have to use a variant with random padding in practice.

(a) Suppose $e = 3$ and Alice sends the same message m encrypted to three (or more) different persons having RSA-keys $(N_1, e), (N_2, e), (N_3, e)$. Show how Eve can compute m having only eavesdropped the three ciphertexts c_1, c_2, c_3.

(b) Suppose we want to use Textbook-RSA in a hybrid encryption choosing large enough primes such that $N > 2^{1024}$ and $e = 7$. We want to encrypt AES-keys, i.e. messages from $\{0, 1\}^{128}$. Explain why this is a really bad idea!

Exercise 6.3 Elgamal PKES—why to work in $\mathbb{QR}_p$

Construct a CPA-attack on Elgamal relative to $\mathsf{Gen}\mathbb{Z}^*_{\text{safe}}$, i.e. Gen returns

$$I = (\langle \mathbb{Z}_p^*, 1, \cdot \rangle, q, g, x, h)$$

with p a n-bit prime, $q = p - 1$, and g generates all elements in $\mathbb{Z}_p^*$.)
Hint: Consider the observations made about the DDH problem relative to $\mathsf{Gen}\mathbb{Z}^*_{safe}$ *in the lecture.*

Exercise 6.4 **Elgamal's DSS—why it fails without hashing**

Elgamal already showed in his paper, how to efficiently forge a valid tag for a new message:

- Let (m, r, s) be a valid message-tag pair.
- Choose $A, B, C \in \mathbb{Z}$ s.t. $\mathsf{gcd}(Ar - Cs, p-1) = 1$.
- Set $r' := r^A \cdot g^B \cdot y^C \bmod p$, $s' := sr'(Ar - Cs)^{-1} \bmod p-1$, and $m' := r'(Am + Bs)(Ar - Cs)^{-1} \bmod p-1$.

Show that (r', s') is valid for m'.