CPSC 418 / MATH 318 — Introduction to Cryptography

ASSIGNMENT 2 — SOLUTION KEY

Written Problems for CPSC 418 and MATH 318

Problem 1 — Conditional entropy (9 marks)

Let X, Y be two random variables with respective sample spaces $\mathcal{X}$, $\mathcal{Y}$ and probability distributions $p(x), p(y)$ $(x \in \mathcal{X}, y \in \mathcal{Y})$. Recall that the *joint probability* $p(x, y)$ is the probability that $p(X = x)$ and $p(Y = y)$; it is related to the conditional probability via the formula

$$p(x, y) = p(x|y)p(y) \quad \text{for all } x \in \mathcal{X} \text{ and } y \in \mathcal{Y}.$$

The *conditional entropy* or *equivocation* of X given Y is defined as

$$H(X|Y) = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p(x, y) \log_2 \left(\frac{1}{p(x|y)} \right) = \sum_{y \in \mathcal{Y}} p(y) \sum_{x \in \mathcal{X}} p(x|y) \log_2 \left(\frac{1}{p(x|y)} \right) ,$$

where the sums $\sum_{x \in \mathcal{X}}$ and $\sum_{y \in \mathcal{Y}}$ run over all outcomes of X and of Y, respectively, such that $p(x|y) > 0$. Informally,[1] the equivocation $H(X|Y)$ measures the uncertainty about X given Y.

a. Consider a cryptosystem with plaintext space $\mathcal{M} = \{M_1, M_2, M_3, M_4\}$ and ciphertext space $\mathcal{C} = \{C_1, C_2, C_3, C_4\}$. Suppose that plaintexts and ciphertexts are uniformly distributed, i.e. $p(M_i) = p(C_j) = 1/4$ for $1 \leq i, j \leq 4$. Now suppose each ciphertext has only two possible decryptions as follows:

- The decryption of C_1 is either M_1 or M_2, with either possibility equally likely;
- The decryption of C_2 is either M_3 or M_4, with either possibility equally likely;
- The decryption of C_3 is either M_2 or M_3, with either possibility equally likely;
- The decryption of C_4 is either M_1 or M_4, with either possibility equally likely.

(i) (2 marks) Determine $p(M_i|C_j)$ for all i, j with $1 \leq i, j \leq 4$.

Solution. For each j with $1 \leq j \leq 4$, we have $p(M_i|C_j) = 1/2$ for two of the M_i and $p(M_i|C_j) = 0$ for the other two M_i $(1 \leq i \leq 4)$. The specific values of $p(M_i|C_j)$ are given in the following table:

	M_1	M_2	M_3	M_4
C_1	1/2	1/2	0	0
C_2	0	0	1/2	1/2
C_3	0	1/2	1/2	0
C_4	1/2	0	0	1/2

(ii) (1 mark) Does this system provide perfect secrecy? Explain your answer?

[1] Shannon measured the security of a cipher in terms of the key equivocation $H(K|C)$, i.e. the amount of information about a key K that is not revealed by a given ciphertext C.

Solution. No. For example, $p(M_1) = 1/4$, but $p(M_1|C_1) = 1/2 \neq p(M_1)$.

(iii) (3 marks) Compute $H(M|C)$.

Solution. By part (a) (i), for each C_j with $1 \leq j \leq 4$, we have $p(M_i|C_j) = 1/2$ for two of the M_i and $p(M_i|C_j) = 0$ for the other two M_i ($1 \leq i \leq 4$). Thus,

$$\begin{aligned} H(M|C) &= \sum_{j=1}^{4} p(C_j) \sum_{i=1}^{4} p(M_i|C_j) \log_2 \left(\frac{1}{p(M_i|C_j)} \right) \\ &= \sum_{j=1}^{4} \frac{1}{4} \left(2 \cdot \frac{1}{2} \log_2(2) \right) = \sum_{j=1}^{4} \frac{1}{4} \cdot 1 = 4 \cdot \frac{1}{4} = 1 \ . \end{aligned}$$

Informally, this result should be interpreted to mean that any ciphertext leaves only one bit of uncertainty for its corresponding plaintext. Suppose M_1, M_2, M_3, M_4 are optimally encoded as 00, 01, 10, 11, respectively. Then inercepting C_1 for example leaves ambiguity only in the second bit of the corresponding plaintext (M_1 or M_2); the first bit is known to be 0. Similarly, seeing C_3 reveals a decryption whose first and second bit are different (01 or 10), so the uncertainty is only one bit (knowing one plaintext bit determines the other one uniquely).

b. (3 marks) Suppose a cryptosystem provides perfect secrecy and assume that $p(M) > 0$ for all $M \in \mathcal{M}$. Prove that $H(M|C) = H(M)$.

Solution. Perfect secrecy means $p(M|C) = p(M)$ for all $M \in \mathcal{M}$ with $p(M) > 0$ (which is all $M \in \mathcal{M}$ in this case) and all $C \in \mathcal{C}$. Thus,

$$\begin{aligned} H(M|C) &= \sum_{C \in \mathcal{C}} p(C) \sum_{M \in \mathcal{M}} p(M|C) \log_2 \left(\frac{1}{p(M|C)} \right) \quad \text{by definition of } H(M|C) \\ &= \sum_{C \in \mathcal{C}} p(C) \sum_{M \in \mathcal{M}} p(M) \log_2 \left(\frac{1}{p(M)} \right) \quad \text{by definition of perfect secfrecy} \\ &= 1 \cdot H(M) = H(M) \ . \end{aligned}$$

Problem 2 — Fun with binary polynomials (11 marks)

In this problem, we consider polynomials with binary coefficients, i.e. coefficients 0 or 1 where arithmetic is done modulo 2 (or equivalently, as x-or).

a. (3 marks) List all the polynomials of degree 3 with binary coefficients in lexicographical order.

Solution.

1) x^3
2) $x^3 + 1$
3) $x^3 + x$
4) $x^3 + x + 1$
5) $x^3 + x^2$
6) $x^3 + x^2 + 1$
7) $x^3 + x^2 + x$
8) $x^3 + x^2 + x + 1$

b. (3 marks) Recall that a polynomial is *reducible* if has a factorization into non-constant polynomials of smaller degree, and *irreducible* otherwise. List all the reducible polynomials of degree 3 with binary coefficients. For each of these polynomials, provide a nontrivial[2] factorization.

Solution. A degree 3 polynomial $f(x)$ is reducible if and only if it factors into a linear and a (possibly again reducible) quadratic factor. Now $x - a$ is a linear factor of $f(x)$ if and only if a is a root of $f(x)$. For polynomials with binary coefficients, this means that $f(x)$ is reducible if and only if $f(0) = 0$ or $f(1) = 0$. In the former case, x is a factor of $f(x)$, in the latter case $x + 1$ is a factor. With this simple criterion, we see that polynomials 1, 2, 3, 5, 7, 8 above are all reducible, with the following factorizations:

$$\begin{aligned} x^3 &= x \cdot x^2 \\ x^3 + 1 &= (x+1)(x^2 + x + 1) \\ x^3 + x &= x(x^2+1) = x(x+1)^2 \\ x^3 + x^2 &= (x+1)x^2 \\ x^3 + x^2 + x &= x(x^2 + x + 1) \\ x^3 + x^2 + x + 1 &= (x+1)(x^2+1) = (x+1)^3 \end{aligned}$$

c. Consider the finite field GF(2^4) obtained via arithmetic modulo the irreducible degree 4 polynomial $p(x) = x^4 + x + 1$. That is, GF(2^4) consists of polynomials of degree ay most 3 with binary coefficients, and addition and multiplication are performed modulo $p(x)$.

(i) (3 marks) Compute the product of $f(x)g(x)$ in GF(2^4) where $f(x) = x^2 + 1$ and $g(x) = x^3 + x^2 + 1$.

Solution. Modulo 2, we obtain

$$\begin{aligned} f(x)g(x) &= (x^2+1)(x^3+x^2+1) \\ &= x^5 + x^3 \ + \ x^4 + x^2 \ + \ x^2 + 1 \\ &= x^4 + x^3 + x^3 + (x^2 + x^2) + 1 \\ &= x^4 + x^3 + x^3 + 1 \ . \end{aligned}$$

Recalling that addition and subtraction in GF(2) (i.e. addition and subtraction modulo 2) are the same, we note that $x^4 = -x - 1 = x^4 + 1$ in GF(2^4), and hence $x^5 = x^2 + x$. So

$$x^5 + x^4 + x^3 + 1 = (x^2 + x) + (x + 1) + x^3 + 1 = x^3 + x^2 \ .$$

Hence $f(x)g(x) = x^3 + x^2$ in GF(2^4).

[2] I.e. not $f(x) = 1 \cdot f(x)$ or $f(x) = f(x) \cdot 1$.

(ii) (2 marks) Find the inverse of $f(x) = x$ in $\mathrm{GF}(2^4)$, i.e. the polynomial $g(x)$ such that $f(x)g(x) = 1$ in $\mathrm{GF}(2^4)$.
Hint: Be smart about this. You do *not* need to resort to long division here. Instead, use the fact that $p(x) = 0$ in $\mathrm{GF}(2^4)$.

Solution. $1 = x^4 + x = x(x^3 + 1) = f(x)(x^3 + 1)$ in $\mathrm{GF}(2^4)$, so the inverse of $f(x) = x$ in $\mathrm{GF}(2^4)$ is $x^3 + 1$.

Problem 3 — Arithmetic in the AES MixColumns operation (20 marks)

Recall that the MixColumns operation in AES performs arithmetic on 4-byte vectors using the polynomial $M(y) = y^4 + 1$. In this arithmetic, we have $M(y) = 0$, so $y^4 = 1$.

a. In this part of the problem, we consider multiplication of 4-byte vectors (viewed as polynomials of degree ≤ 3 whose coefficients are bytes) by powers of y.

(i) (2 marks) Formally prove that in this arithmetic, multiplication of any 4-byte vector by y is a circular left shift of the vector by one byte. If you use polynomial arithmetic for your answer, don't forget to convert your final polynomial back to a 4-byte vector.

Solution. Let $s = (s_3, s_2, s_1, s_0)$ be a 4-byte vector. Then the corresponding polynomial is $s(y) = s_3y^3 + s_2y^2 + s_1y + s_0$, and we have

$$\begin{aligned} s(y)y &= s_3y^4 + s_2y^3 + s_1y^2 + s_0y \\ &= s_3 \cdot 1 + s_2y^3 + s_1y^2 + s_0y \\ &= s_2y^3 + s_1y^2 + s_0y + s_3 \ , \end{aligned}$$

corresponding to the 4-byte vector (s_2, s_1, s_0, s_3) obtained via applying a circular left shift of one byte to s.

(ii) (4 marks) Generalizing part (i) to other powers of y, formally prove that multiplication of any 4-byte vector by y^i $(0 \leq i \leq 3)$ is a circular left shift of the vector by i bytes. Again, remember to convert your polynomial result back to a 4-byte byte vector.

Solution. Let $s = (s_3, s_2, s_1, s_0)$ be a 4-byte vector, with corresponding polynomial $s(y) = s_3y^3 + s_2y^2 + s_1y + s_0$. Multiplication of $s(y)$ by $y^0 = 1$ leaves $s(y)$ (and hence s) unchanged, resulting in 0 left shifts applied to s. The case $i = 1$ was proved in part (i). By part (i), using 4-byte vector arithmetic modulo $y^4 + 1$, we have

$$\begin{aligned} y^5 &= y \cdot y^4 = y \cdot 1 = y \ , \\ y^6 &= y \cdot y^5 = y \cdot y = y^2 \ . \end{aligned}$$

Hence, analogous to the reasoning of part (i), we obtain

$$\begin{aligned} s(y)y^2 &= s_3y^5 + s_2y^4 + s_1y^3 + s_0y^2 \\ &= s_3y + s_2 \cdot 1 + s_1y^3 + s_0y^2 \\ &= s_1y^3 + s_0y^2 + s_3y + s_2 \ , \end{aligned}$$

corresponding to the 4-byte vector (s_1, s_0, s_3, s_2) obtained from s by a circular left shift of 2 bytes. Similarly,

$$\begin{aligned} s(y)y^3 &= s_3y^6 + s_2y^5 + s_1y^4 + s_0y^3 \\ &= s_3y^2 + s_2y + s_1 \cdot 1 + s_0y^3 \\ &= s_0y^3 + s_3y^2 + s_2y + s_1 \ , \end{aligned}$$

corresponding to the 4-byte vector (s_0, s_3, s_2, s_1) obtained from s by a circular left shift of 3 bytes.

b. Next, we consider arithmetic involving the coefficients of the polynomial

$$c(y) = (03)y^3 + (01)y^2 + (01)y + (02) \ ,$$

that appears in MixColumns, where the coefficients of $c(y)$ are bytes written in hexadecimal (i.e. base 16) notation. Arithmetic involving this polynomial requires the computation of products involving the bytes (01), (02) and (03) in the Rijndahl field GF(2^8). Recall that in this field, arithmetic is done modulo $m(x) = x^8 + x^4 + x^3 + x + 1$.

(i) (2 marks) Write the bytes (01), (02), (03) as their respective polynomial representatives $c_1(x)$, $c_2(x)$ and $c_3(x)$ in the Rijndahl field GF(2^8).

Solution. Using the variable x for our polynomials in GF(2^8):

$$\begin{array}{lll} (01)_{16} = (00000001)_2 & \text{yields} & c_1(x) = c_2(x) = 1 \ , \\ (02)_{16} = (00000010)_2 & \text{yields} & c_0(x) = x \ , \\ (03)_{16} = (00000011)_2 & \text{yields} & c_3(x) = x + 1 \ . \end{array}$$

(ii) (4 marks) Let $b = (b_7\, b_6 \cdots b_1\, b_0)$ be any byte, and let $d = (02)b$ be the product of the bytes (02) and b in the Rijndahl field GF(2^8). Then d is again a byte of the form $d = (d_7\, d_6 \cdots d_1\, d_0)$. Provide formulas for the bits d_i, $0 \le i \le 7$, in terms of the bits b_i.

Solution. Let

$$d(x) = d_7x^7 + \cdots + d_1x + d_0 \ , \qquad b(x) = b_7x^7 + \cdots + b_1x + b_0$$

be the polynomial representatives of the bytes d and b, respectively. Then

$$d(x) \equiv b(x)c_2(x) \equiv b(x)x \equiv b_7x^8 + b_6x^7 + \cdots b_1x^2 + b_0x \mod m(x) \ .$$

Since $m(x) = 0$ in GF(2^8), we see that $x^8 = x^4 + x^3 + x + 1$ in GF(2^8), so in GF(2^8), we have

$$\begin{aligned} d(x) &= b_7(x^4 + x^3 + x + 1) + b_6x^7 + \cdots + b_1x^2 + b_0x \\ &= b_6x^7 + b_5x^6 + b_4x^5 + (b_3 + b_7)x^4 + (b_2 + b_7)x^3 + b_1x^2 + (b_0 + b_7)x + b_7 \ , \end{aligned}$$

yielding

$$\begin{array}{llll} d_7 = b_6, & d_6 = b_5, & d_5 = b_4, & d_4 = b_3 + b_7, \\ d_3 = b_2 + b_7, & d_2 = b_1, & d_1 = b_0 + b_7, & d_0 = b_7 \ . \end{array}$$

(iii) (3 marks) Provide analogous expressions as in part (b) (ii) for the byte product $e = (03)b$, where $b = (b_7\, b_6 \cdots b_1\, b_0)$ is any byte.

Solution. Let $e(x) = e_7x^7 + \cdots + e_1x + e_0$ be the polynomial representation of e. Then

$$e(x) \equiv b(x)c_3(x) \equiv b(x)(x + 1) \equiv b(x)x + b(x) \equiv d(x) + b(x) \pmod{m(x)} \ ,$$

so $e_i = d_i + b_i$ for $0 \le i \le 7$. Using the result of part (i), we obtain

$$\begin{array}{llll} e_7 = b_6 + b_7, & e_6 = b_5 + b_6, & e_5 = b_4 + b_5, & e_4 = b_3 + b_4 + b_7, \\ e_3 = b_2 + b_3 + b_7, & e_2 = b_1 + b_2, & e_1 = b_0 + b_1 + b_7, & e_0 = b_0 + b_7 \, . \end{array}$$

c. The MixColumns operation performs multiplication of 4-byte vectors by the polynomial $c(y)$ of part (b). In this part of the problem, you will evaluate such products symbolically.

(i) (3 marks) Let $s(y) = s_3y^3 + s_2y^2 + s_1y + s_0$ be a polynomial whose coefficients are bytes. Symbolically compute the product $t(y) = s(y)c(y) \bmod y^4 + 1$. The result should be a polynomial of the form $t(y) = t_3y^3 + t_2y^2 + t_1y + t_0$ where t_3, t_2, t_1, t_0 are bytes. Provide formulas for the bytes t_i, $0 \le i \le 3$, in terms of the bytes s_i. The equations should consist of sums of byte products of the form $01s_i$, $02s_i$, $03s_i$, $0 \le i \le 3$. You need *not* compute these individual byte products as you did in part (b).

Solution. As in part (a), when conducting arithmetic modulo y^4+1 over GF(2^8), we can replace any occurrence of y^4 by 1, of y^5 by y and of y^6 by y^2. Hence, we obtain modulo $y^4 + 1$:

$$\begin{aligned}
t(y) &= (03y^3 + 01y^2 + 01y + 02)(s_3y^3 + s_2y^2 + s_1y + s_0) \\
&= (03s_3)y^6 + (03s_2 + 01s_3)y^5 + (03s_1 + 01s_2 + 01s_3)y^4 \\
&\quad +(03s_0 + 01s_1 + 01s_2 + 02s_3)y^3 + (01s_0 + 01s_1 + 02s_2)y^2 \\
&\quad +(01s_0 + 02s_1)y + (02s_0) \\
&= (03s_3)y^2 + (03s_2 + 01s_3)y + (03s_1 + 01s_2 + 01s_3) \cdot 1 \\
&\quad +(03s_0 + 01s_1 + 01s_2 + 02s_3)y^3 + (01s_0 + 01s_1 + 02s_2)y^2 \\
&\quad +(01s_0 + 02s_1)y + (02s_0) \\
&= (03s_0 + 01s_1 + 01s_2 + 02s_3)y^3 + (01s_0 + 01s_1 + 02s_2 + 03s_3)y^2 \\
&\quad +(01s_0 + 02s_1 + 03s_2 + 01s_3)y + (02s_0 + 03s_1 + 01s_2 + 01s_3) \ .
\end{aligned}$$

So

$$\begin{aligned}
t_0 &= 02s_0 + 03s_1 + 01s_2 + 01s_3 \ , \\
t_1 &= 01s_0 + 02s_1 + 03s_2 + 01s_3 \ , \\
t_2 &= 01s_0 + 01s_1 + 02s_2 + 03s_3 \ , \\
t_3 &= 03s_0 + 01s_1 + 01s_2 + 02s_3 \ .
\end{aligned}$$

(ii) (2 marks) Write your solution of part (c) (i) in matrix form; i.e. give a 4×4 matrix C whose entries are bytes such that

$$\begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \end{pmatrix} = C \begin{pmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \end{pmatrix} \ .$$

Note that this yields the matrix representation of MixColumns presented (without proof) in class.

Solution. Rewriting part (i), we obtain:

$$\begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \end{pmatrix} \ .$$

Problem 4 — Error propagation in block cipher modes (12 marks)

Error propagation is often an important consideration when choosing a mode of operation in practice. In this problem, you will analyze the error propagation properties of an arbitrary block cipher in various such modes; note that these properties are independent of the cipher used.

a. Suppose Alice wants to send a sequence of message blocks $M_0, M_1, M_2, \ldots$ to Bob, encrypted to ciphertext blocks $C_0, C_1, C_2, \ldots$ using some fixed key K. Assume that an error occurs during transmission of a particular block of ciphertext C_i. Justifying all your answers, explain which plaintext blocks are affected after Bob decrypts this (faulty) ciphertext block C_i when the cipher is used in

(i) (2 marks) ECB mode?

Solution. Here, $C_j = E_K(M_j)$, $M_j = D_K(C_j)$ for $j = 0, 1, 2, \ldots$. So C_i only affects M_i, and no other plaintext blocks M_j, $j \neq i$, under decryption; in other words, only M_i is affected.

(ii) (2 marks) CBC mode?

Solution. Here, $C_0 = E_K(M_0 \oplus IV)$, $C_j = E_K(C_{j-1} \oplus M_j)$ for $j \geq 1$. So $M_0 = D_K(C_0) \oplus IV$, $M_j = D_K(C_j) \oplus C_{j-1}$ for $j \geq 1$. In particular, C_i occurs in the decryptions that yield M_i and M_{i+1}, but in no other decryptions. Hence the two plaintext blocks M_i and M_{i+1} are affected.

(iii) (2 marks) OFB mode?

Solution. Here, $KS_0 = IV$, $KS_j = E_K(KS_{j-1})$ for $j \geq 1$, and $C_j = M_j \oplus KS_j$ for $j \geq 0$. Hence $M_j = C_j \oplus KS_j$ for $j \geq 0$. In particular, C_i only occurs in the computation of M_i and in no other decryptions. So only M_i is affected.

(iv) (2 marks) CFB mode with one register?

Solution. Here, $C_0 = IV$, $KS_j = E_K(C_{j-1})$ for $j \geq 1$, and $C_j = M_j \oplus KS_j$ for $j \geq 1$. Hence $M_j = C_j \oplus E_K(C_{j-1})$ for $j \geq 1$. So analogous to the reasoning for CBC mode, the two plaintext blocks M_i and M_{i+1} are affected.

Remark. If we used k registers, i.e. fed k ciphertext blocks back into the encryption/decryption machinery, then the $k+1$ plaintext blocks $M_i, M_{i+1}, \ldots, M_{i+k}$ would be affected. (You did not need to make this observation in your solution since the question only asked you to consider one register.)

(v) (2 marks) CTR mode?

Solution. Here, $C_j = M_j \oplus E_K(CTR_j)$ for $j = 0, 1, 2, \ldots$ where each counter value CTR_j is obtained from the previous value CTR_{j-1} via some specified cipher-independent function. Hence $M_j = C_j \oplus E_K(CTR_j)$. So analogous for the reasoning for OFB, only M_i is affected.

b. (2 marks) Suppose now that an error occurs in a particular plaintext block M_i before Alice encrypts it and sends the corresponding ciphertext C_i to Bob. Upon decryption of C_i, which plaintext blocks M_j are affected when using the cipher in CBC mode?
Hint: Think about this carefully; it's easy to make a conceptual mistake here.

Solution. Just M_i. In fact, after re-examining the definition of encryption in CBC mode, you hopefully realized that this is irrelevant. Any error that occurs before encryption is not sensitive to the mode of operation of the cipher, and hence only affects the identical plaintext block on Bob's side.

Problem 5 — A simplified password-based key agreement protocol (8 marks)

The following is a simplified (and hence problematic) version of the key generation phase of the password-based key agreement protocol that CPSC 418 students are asked to implement in Problem 8 (the programming problem). Here, a client first performs a one-time registration of their authentication credentials with a server. These credentials can then be used to generate authenticated session keys between server and client via communication over an insecure channel.

All participants agree on a large public safe prime[3] $N = 2q+1$, with q prime, and a public primitive root g of N. Each client has their own password p, interpreted as an integer between 0 and $N-2$ (inclusive). To register with the server, a client computes $v \equiv g^p \pmod N$ and provides the server with the pair (I, v) where I is the client's user id.[4]

Protocol:

1. Client generates a random value a with $0 \le a \le N-1$, computes $A \equiv g^a \pmod N$, and sends (I, A) to server, where I is the client's user id.

   Server generates a random value b with $0 \le b \le N-1$, computes $B \equiv g^b \pmod N$, and sends B to client.

2. Client computes $K_{\text{client}} \equiv B^{a+p} \pmod N$.

   Server retrieves client's authentication data (I, v) and computes $K_{\text{server}} \equiv (Av)^b \pmod N$.

Note that this protocol is similar to Diffie-Hellman, except that the client's password p and authentication credential v are incorporated in the key computation.

a. [2 marks] Prove that client and server have a shared key after executing this protocol, i.e. prove that $K_{\text{server}} = K_{\text{client}}$.

   Solution.
   $$K_{\text{server}} \equiv (Av)^b \equiv (g^a g^p)^b \equiv g^{a+p}b \equiv (g^b)^{a+p} \equiv B^{a+p} \pmod N \ .$$

b. (3 marks) Suppose Mallory[5] obtains client Ian's authentication data (I, v), for example, by intercepting Ian's transmission to the server upon his registration or by hacking into the server's database. Show how Mallory can masquerade as Ian, i.e. execute the protocol with the server (using a value A of her choice and generate a valid key K_{client} that the server believes is shared with Ian.
   Hint: Concoct a value A from Ian's credential v such that the server's computation of K_{server} in step 2 produces a value that Mallory can easily compute. Note that $A = 0$ is not a valid choice because A must belong to $\mathbb{Z}_p^*$.

   Solution. Mallory generates a random value a with $0 \le a \le N-1$, computes $A \equiv g^a v^{N-2} \pmod N$ and send (I, A) to the server. The server computes

   $$K_{\text{server}} \equiv (Av)^b \equiv (g^a v^{N-2} v)^b \equiv (g^a v^{N-1})^b \equiv (g^a)^b \equiv (g^b)^a \equiv B^a \pmod N \ ,$$

[3] We denote this prime by N, rather than p, because the letter p is reserved for the client's password.

[4] In practice, this needs to be done in a secure and tamper-proof manner. Also, in the computation of v, the client would use a hash of their password p rather than just p. For details, see the protocol description in Problem 8.

[5] This is a standard name for active attackers and is meant to be reminiscent of the word "malicious".

where the fourth congruence above follows from Fermat's Little Theorem. Since Mallory receives the server's B value, she can compute $K_{\text{client}} \equiv B^a \equiv K_{\text{server}} \pmod{N}$, which is a valid key shared with the server.

Remark. The purpose of the value $u \equiv H(A\|B) \pmod{N}$ computed in steps 5 of the full secure version of this protocol given in Problem 8, where H is a cryptographic hash function, is to foil this attack. Here, the server key is $K_{\text{server}} \equiv (Av^u)^b \pmod{N}$. To adapt the attack, Mallory would have to chose $A \equiv g^a v^{(N-2)u} \pmod{N}$, but this is impossible because u depends on A. That is, A needs to be computed before u and hence cannot have u in its computation.

c. (3 marks) Consider the following two problems:

- *Key Recovery Problem:* Given any values $A \equiv g^a \pmod{N}$ and $B \equiv g^b \pmod{N}$ and any $v \in \mathbb{Z}_N^*$, find any valid key K produced by the protocol above.
- *Diffie-Hellman Problem:* Given any values $A \equiv g^a \pmod{N}$ and $B \equiv g^b \pmod{N}$, find the Diffie-Hellman key $g^{ab} \pmod{N}$.

Solving the first problem breaks the key agreement protocol above, solving the second problem breaks the Diffie-Hellman protocol.

Assume a passive attacker Eve has an algorithm for solving any instance of the key recovery problem. Show how she can use her algorithm to solve any instance of the Diffie-Hellman problem. Informally, this means that breaking the key agreement protocol above is at least as hard as breaking Diffie-Hellman. Note that the exponents a and b are assumed to be unknown for both these problems, and Eve cannnot find them because we are not assuming that she has the ability to extract discrete logarithms.
Hint: Don't overthink this. The answer is very simple.

Solution. Suppose Eve is given quantities $A \equiv g^a \pmod{N}$ and $B \equiv g^b \pmod{N}$. She wishes to find the Diffie-Hellman key $g^{ab} \pmod{N}$.

She simply solves the key recovery problem with inputs A, B and $v = 1$; in other words, she uses her algorithm to find the key shared between the server and a client whose password is 0 (or in fact any multiple of $p-1$), with corresponding credential $v \equiv 1 \pmod{N}$. Following the server key computation in step 2, this produces the key $K_{\text{server}} \equiv (A \cdot 1)^b \equiv A^b \equiv g^{ab} \pmod{N}$, which is the desired Diffie-Hellman key K.

Written Problems for MATH 318 only

Problem 6 — Joint entropy (20 marks)

Let X, Y be two random variables with respective sample spaces $\mathcal{X}, Y$ and probability distributions $p(x), p(y)$ $(x \in \mathcal{X}, y \in \mathcal{Y})$. Assume for simplicity that $p(x) > 0$ for all $x \in X$ and $p(y) > 0$ for all $y \in Y$.

a. (1 mark) Explain why $\sum_{x \in \mathcal{X}} p(x|y) = 1$ for all $y \in \mathcal{Y}$.

Solution. For fixed but arbitrary $y \in \mathcal{Y}$, the values $p(x|y)$ form a probability distribution on the sample space $\mathcal{X}$. So their sum is one, i.e. $\sum_{x \in \mathcal{X}} p(x|y) = 1$.

b. (2 marks) Use part (a) and the formula connecting the joint and conditional probability given in Problem 1 to prove that

$$\sum_{x \in \mathcal{X}} p(x, y) = p(y) \quad \text{for all } y \in \mathcal{Y} \ .$$

Solution. Let $y \in \mathcal{Y}$. Then $\sum_{x \in \mathcal{X}} p(x|y) = 1$, so

$$\sum_{x \in \mathcal{X}} p(x, y) = \sum_{x \in \mathcal{X}} p(x|y)p(y) = p(y) \cdot \sum_{x \in \mathcal{X}} p(x|y) = p(y) \cdot 1 = p(y) \cdot \ .$$

c. (2 marks) Use part (b) to prove that

$$\sum_{y \in \mathcal{Y}} p(x, y) = p(x) \quad \text{for all } x \in \mathcal{X} \ .$$

Solution. Let $x \in X$. Reversing x and y in part (b) and using the fact that $p(x, y) = p(y, x)$ for all $x \in X$ and $y \in Y$ yields

$$\sum_{y \in \mathcal{Y}} p(x, y) = \sum_{y \in \mathcal{Y}} p(y, x) = p(x) \ .$$

d. (6 marks) The *joint entropy of X and Y* captures the combined uncertainty of X and Y and is defined to be

$$H(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 \left(\frac{1}{p(x, y)} \right) \ .$$

Prove that $H(X, Y) \leq H(X) + H(Y)$. Use parts (b) & (c) and (without proof) the fact that $p(x, y) \geq p(x)p(y)$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. Informally, this result says that the total uncertainty of X and Y can only decrease if X and Y occur together.

Solution.

$$\begin{aligned}
H(X,Y) &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log_2\left(\frac{1}{p(x,y)}\right) \\
&\leq \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log_2\left(\frac{1}{p(x)p(y)}\right) \quad \text{since } p(x,y)\geq p(x)p(y) \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)[-\log_2(p(x)p(y))] \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)[-\log_2(p(x)) - \log_2(p(y))] \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\left[\log_2\left(\frac{1}{p(x)}\right) + \log_2\left(\frac{1}{p(y)}\right)\right] \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\ \log_2\left(\frac{1}{p(x)}\right) + \sum_{x\in X}\sum_{y\in Y} p(x,y)\ \log_2\left(\frac{1}{p(y)}\right) \\
&= \sum_{x\in\mathcal{X}}\left(\sum_{y\in\mathcal{Y}} p(x,y)\right)\log_2\left(\frac{1}{p(x)}\right) + \sum_{y\in Y}\left(\sum_{x\in X} p(x,y)\right)\log_2\left(\frac{1}{p(y)}\right) \\
&= \sum_{x\in\mathcal{X}} p(x)\log_2\left(\frac{1}{p(x)}\right) + \sum_{y\in\mathcal{Y}} p(y)\log_2\left(\frac{1}{p(y)}\right) \quad \text{by parts (b) \& (c)} \\
&= H(X) + H(Y)\ .
\end{aligned}$$

e. (3 marks) Prove that equality holds in part (d) if and only if and only if X and Y are independent, i.e. x and y are independent for all outcomes $x \in X$ and $y \in Y$. Informally, this result says that if X and Y are independent, then the fact that they occur together does not reduce their total uncertainty.

Solution. Recall that $x \in X$ and $y \in Y$ are independent if $p(x,y) = p(x)y(y)$, i.e. the inequality used (without proof) in part (c) is an equality.

If X and Y are independent, then the inequality in line 2 of the proof of part (d) is an equality for all $x \in X$ and $y \in Y$, so $H(X,Y) = H(X) + H(Y)$.

Conversely, if X and Y are not independent, then $p(x,y) > p(x)p(y)$ for some pair $x \in X, y \in Y$, and $p(x,y) \geq p(x)p(y)$ for all the other pairs $x \in X, y \in Y$. So the inequality in line 2 of the proof of part (d) is strict, and hence $H(X,Y) < H(X) + H(Y)$.

f. (5 marks) Use part (b) to prove that $H(X,Y) = H(X|Y) + H(Y)$, where $H(X|Y)$ was defined in Problem 1.

Solution.

$$
\begin{aligned}
H(X,Y) &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log_2\left(\frac{1}{p(x,y)}\right) \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log_2\left(\frac{1}{p(x|y)p(y)}\right) \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)[-\log_2(p(x|y)p(y))] \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)[-\log_2(p(x|y)) - \log_2(p(y))] \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\left[\log_2\left(\frac{1}{p(x|y)}\right) + \log_2\left(\frac{1}{p(y)}\right)\right] \\
&= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log_2\left(\frac{1}{p(x|y)}\right) + \sum_{y\in\mathcal{Y}}\left(\sum_{x\in\mathcal{X}} p(x,y)\right)\log_2\left(\frac{1}{p(y)}\right) \\
&= H(X|Y) + \sum_{y\in\mathcal{Y}} p(y)\log_2\left(\frac{1}{p(y)}\right) \quad \text{by def'n of } H(X|Y) \text{ and part (b)} \\
&= H(X|Y) + H(Y)\ .
\end{aligned}
$$

g. (1 mark) Combine the results of parts (f) and (d) to prove that $H(X|Y) \le H(X)$. Informally, this result says that the uncertainty of X can only decrease when some side information Y is known.

Solution.

$$
H(X|Y) \overset{\text{part (f)}}{=} H(X,Y) - H(Y) \overset{\text{part (d)}}{\le} H(X) + H(Y) - H(Y) = H(X)\ .
$$

Problem 7 — A characterization of perfect secrecy (20 marks)

In this problem, you will prove the following result, due to Shannon and mentioned without proof in class:

> **Theorem.** Consider a cryptosystem with plaintext space $\mathcal{M}$, ciphertext space $\mathcal{C}$ and key space $\mathcal{K}$ such that $|\mathcal{M}| = |\mathcal{C}| = |\mathcal{K}|$. Assume that[6] $p(M) > 0$ for all $M \in \mathcal{M}$ and $p(C) > 0$ for all $C \in \mathcal{C}$. Then the system provides perfect secrecy if and only if
>
> A. For every plaintext $M \in \mathcal{M}$ and every ciphertext $C \in \mathcal{C}$, there exists a unique key $K \in \mathcal{K}$ such that C is the encryption of M under key K, and
> B. every key $K \in \mathcal{K}$ is used with equal probability $1/|\mathcal{K}|$.

a. (Proof of "only if") Assume first that the system provides perfect secrecy, so $p(C|M) = p(C)$ for all $M \in \mathcal{M}$ and $C \in \mathcal{C}$ with $p(M) > 0$.

(i) (2 marks) Prove that for every $M \in \mathcal{M}$ and every $C \in \mathcal{C}$, there exists at least one key $K \in \mathcal{K}$ such that C is the encryption of M under key K.

Solution. For all $M \in \mathcal{M}$ with $p(M) > 0$ and all $C \in \mathcal{C}$, we have

$$0 < p(C) = p(C|M) = \sum_{\substack{K \in \mathcal{K} \text{ with} \\ E_K(M) = C}} p(K) \ . \tag{1}$$

Since the sum is non-zero, it must contain at least one term. So there exists at least one key K such that $E_K(M) = C$.

(ii) (3 marks) Prove that for every $M \in \mathcal{M}$ and every $C \in \mathcal{C}$, there exists at most one key (and hence exactly one key by part (a) (i)) $K \in \mathcal{K}$ such that C is the encryption of M under key K.

Solution. Choose any $M \in \mathcal{M}$ and consider the map $f_M : \mathcal{K} \to \mathcal{C}$ defined via $f_M(K) = E_K(M)$. Let $C \in \mathcal{C}$. By part (a) (i), there exists at least one key $K \in K$ such that $C = E_K(M) = f_M(K)$. So f_M is surjective for every $M \in \mathcal{M}$. Since the domain and co-domain of f_M have the same cardinality, f_M is also injective, and hence bijective, for every $M \in \mathcal{M}$. Hence, for every $M \in \mathcal{M}$ and $C \in \mathcal{C}$, there exists exactly one key $K \in \mathcal{K}$ such that $C = f_M(K) = E_K(M)$.

(iii) (3 marks) Prove that every key $K \in \mathcal{K}$ is chosen equally likely.

Solution. For every $M \in \mathcal{M}$ and $C \in \mathcal{C}$, the sum in (1) has only one term by part (a) (ii), i.e. $p(C|M) = p(K)$ where K is the unique key encrypting M to C. Now fix a ciphertext $C \in \mathcal{C}$ and number the plaintexts $M_i \in \mathcal{M}$ and keys $K_i \in \mathcal{K}$ in such a way that $E_{K_i}(M_i) = C$. Then

$$p(C) = p(C|M_i) = p(K_i)$$

for all i. So every key K_i occurs with the same probability $p(C)$, so this probability is $1/|\mathcal{K}|$.

b. (Proof of "if") Suppose (A) and (B) above hold.

(i) (2 marks) Prove that for every $C \in \mathcal{C}$ and every $K \in \mathcal{K}$, there exists at least one plaintext $M \in \mathcal{M}$ such that M is the decryption of C under key K.

Solution. For any $C \in C$, we have

$$0 < p(C) = \sum_{\substack{K \in \mathcal{K} \text{ with} \\ C \in E_K(\mathcal{M})}} p(D_K(C))p(K) \ .$$

Since the sum is non-zero, it must contain at least one term. So there exists at least one key K such that $p(K) > 0$ and $p(D_K(C)) > 0$. This means that C has a decryption $M = D_K(C)$ under the key K.

(ii) (3 marks) Prove that for every $C \in \mathcal{C}$ and every $K \in \mathcal{K}$, there exists at most one plaintext (and hence exactly one plaintext by part (b) (i)) $M \in \mathcal{M}$ such that M is the decryption of C under key K.

Solution. Choose any $C \in \mathcal{C}$ and consider the map $g_C : \mathcal{K} \to \mathcal{M}$ defined via $g_C(K) = D_K(C)$. Let $M \in \mathcal{M}$. By part (b) (i), there exists at least one key $K \in K$ such that $M = D_K(C) = g_C(K)$. So g_C is surjective for every $C \in \mathcal{C}$. Since the domain and co-domain of g_C have the same cardinality, g_C is also injective, and hence bijective, for every $C \in \mathcal{C}$. Hence, for every $C \in \mathcal{C}$ and $M \in \mathcal{M}$, there exists exactly one key $K \in \mathcal{K}$ such that $M = g_C(K) = D_K(C)$.

(iii) (5 marks) Prove that $p(C) = 1/|\mathcal{K}|$ for all $C \in \mathcal{C}$.

Solution. Let $C \in \mathcal{C}$. We investigate the sets $E_K(\mathcal{M})$ appearing in the formula for $p(C)$. For any fixed $K \in \mathcal{K}$, the encryption function E_K is an injection whose domain and co-domain have equal cardinality. Hence it is also a surjection. It follows that $E_K(\mathcal{M}) = \mathcal{C}$ for every key $K \in \mathcal{K}$. In other words, every $C \in \mathcal{C}$ belongs to $E_K(\mathcal{M})$ for every $K \in \mathcal{K}$. Thus

$$\begin{aligned} p(C) &= \sum_{\substack{K \in \mathcal{K} \text{ with} \\ C \in E_K(\mathcal{M})}} p(D_K(C))p(K) = \sum_{K \in \mathcal{K}} p(D_K(C))p(K) = \sum_{K \in \mathcal{K}} p(D_K(C)) \frac{1}{|\mathcal{K}|} \\ &= \frac{1}{|\mathcal{K}|} \sum_{K \in \mathcal{K}} p(D_K(C)) = \frac{1}{|\mathcal{K}|} \sum_{K \in \mathcal{K}} p(g_C(K)) \ . \end{aligned}$$

By part (b) (ii), the map g_C is a bijection from $\mathcal{K}$ to $\mathcal{M}$. Hence as K runs through all of $\mathcal{K}$, $g_C(K)$ runs through all of $\mathcal{M}$. In other words, the values $g_C(K)$ are a permutation of all the elements of $\mathcal{M}$. It follows that

$$p(C) = \frac{1}{|\mathcal{K}|} \sum_{K \in \mathcal{K}} p(g_C(K)) = \frac{1}{|\mathcal{K}|} \sum_{M \in \mathcal{M}} p(M) = \frac{1}{|\mathcal{K}|} \cdot 1 = \frac{1}{|\mathcal{K}|} \ .$$

(iv) (2 marks) Prove that $p(C|M) = 1/|\mathcal{K}|$ for all $M \in \mathcal{M}$ and $C \in \mathcal{C}$.

Solution. Let $M \in \mathcal{M}$ and $C \in \mathcal{C}$. By (A), the sum (1) defining $p(C|M)$ has only one term. So

$$p(C|M) = p(K) = 1/|\mathcal{K}| \ .$$

Remark. This problem shows that the one-time pad as well as the shift cipher provide perfect secrecy if each key is chosen equally likely.

Programming Problem for CPSC 418 only

Problem 8 — Secure password based authentication and key exchange (40 marks)

Working code for this problem is posted separately on the Piazza resources page
https://piazza.com/ucalgary.ca/fall2021/cpsc418math318/resources.