This is a collection of past exam questions related to the syllabus of the CST Part II *Cryptography* course, set by Markus Kuhn in the previous CST Part IB and Part II courses *Introduction to Security, Security I, Security II*, since 2002.

Note that following a rearrangement of the security courses, non-cryptography material has now moved to the Part IB *Security* syllabus. Questions that refer entirely to non-cryptography material are therefore omitted here, and (where practical) a note points out which part of a past question is not relevant to the current *Cryptography* syllabus.

COMPUTER SCIENCE TRIPOS Part II – 2018 – Paper 8

12   Security II (MGK)

(a) Name *three* different families of algebraic groups that are commonly used in cryptographic applications of the Diffie–Hellman problem, where *any* group element (other than the neutral element) can be used as a generator. Briefly outline some of their main attributes, such as the set of elements and the group operator. [9 marks]

(b) You are preparing to participate in a password-cracking competition. During the competition, you will be given the 128-bit hash-function output $MD5(p)$. You have to find p, an 8-character password, each character having been chosen uniformly at random from a known alphabet of 64 ASCII characters.

In the weeks preparing for the competition, you have access to a small cluster of GPU graphics cards that can evaluate MD5 10^9 times per second.

During the competition, you have only access to a laptop computer that can evaluate MD5 10^6 times per second.

Without any pre-computation, how long would it take to evaluate MD5 for all possible passwords p in a brute-force attack

(i)   on the laptop? [2 marks]

(ii) on the GPU cluster? [2 marks]

You decide to use the GPU cluster to pre-compute a *rainbow table* for this challenge.

(iii) What functions other than MD5 will the GPU cluster have to evaluate as often as MD5 when building the rainbow table? [3 marks]

(iv) Your laptop has enough RAM for storing the rainbow table as a hash table of 2^{32} key-value pairs (x, y) with $x, y \in \{0, 1\}^{128}$. If you execute MD5 2^{50} times while generating your rainbow table, how long will your laptop need (worst case) to find a password p stored in it, given its MD5 hash value $MD5(p)$? Assume that the runtime is entirely dominated by the MD5 evaluations. [4 marks]

COMPUTER SCIENCE TRIPOS Part II – 2017 – Paper 8

11 Security II (MGK)

(a) An RSA encryption routine calculates the value $m^e \bmod n$ using a square-and-multiply algorithm. During the execution of that algorithm, you can briefly hear a buzzing sound (through radio-frequency interference) on an AM radio receiver located near the computer. You record that sound, and discover that it is actually the following sequence of two different sounds A and B: $BABAABABAAB$. What is the value of e? [6 marks]

(b) MHASH implements a hash function over file sequences $F_0, F_1, \ldots, F_{n-1} \in \{0,1\}^*$ with $n > 0$, using a collision-resistant hash function $H : \{0,1\}^* \to \{0,1\}^b$:

$\text{MHASH}(n, F_0, F_1, \ldots, F_{n-1})$:
  $d := \lceil \log_2 n \rceil$
  for $i := 0$ **to** $n-1$
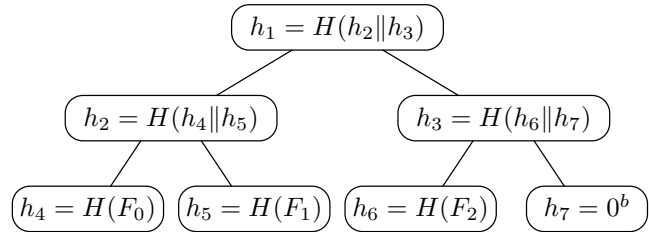 $h_{2^d+i} := H(F_i)$
  for $i := n$ **to** $2^d - 1$
    $h_{2^d+i} := 0^b$
  for $i := 2^d - 1$ **downto** 1
    $h_i := H(h_{2i}\|h_{2i+1})$
  return h_1

Example calculation for $n = 3$:

$h_1 = H(h_2\|h_3)$

$h_2 = H(h_4\|h_5)$ $h_3 = H(h_6\|h_7)$

$h_4 = H(F_0)$ $h_5 = H(F_1)$ $h_6 = H(F_2)$ $h_7 = 0^b$

(i) Show that MHASH is not collision resistant if n is not fixed, by constructing two different sequences of files that result in the same output h_1. [8 marks]

(ii) Suggest an improvement to MHASH to make it collision resistant. [6 marks]

8 Security I (MGK)

(a) *NybbleCrypt* is a block cipher optimized for use in exam questions. It has a block size of 4 bits and a key length of 64 bits. Each block can be written as a single hexadecimal digit, for example $5 \oplus 9 = \mathtt{c}$.

(i) The *NybbleCrypt* encryption function for a particular key K is given in the following table:

m	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$E_K(m)$	c	8	2	7	d	0	6	1	a	e	f	4	b	9	5	3

Decrypt the following messages, which were encrypted using E_K under the following modes of operation, respectively:

(A) ECB mode: `c994f88`   [2 marks]

(B) CBC mode: `b144f`   [3 marks]

(C) OFB mode: `eae26`   [3 marks]

(ii) Calculate the CBC-MAC of the following message, using the same key K as in part $(a)(i)$ above: `face`   [2 marks]

(iii) *NybblePay* point-of-sale card terminals send 4-digit customer PINs to the bank's transaction-processing centre for verification. The bank's reply to the terminal consists of a 7-digit message in the following format:

(A) 4-digit PIN $m_1 m_2 m_3 m_4$

(B) 2-digit result code $m_5 m_6$: `10` if the PIN was correct, `e1` if not

(C) check digit $m_7 = m_1 \oplus \cdots \oplus m_6$ (the bit-wise XOR of previous digits)

This reply is sent OFB-encrypted using the *NybbleCrypt* blockcipher. You have intercepted such a ciphertext message: `a59defc2`. You are confident that it contains the result code $m_5 m_6 = \mathtt{e1}$ for an incorrect PIN. Without knowing the encryption key K, modify the ciphertext message such that after decryption it shows the result code for a correct PIN, and a matching check digit, while preserving the included PIN.   [5 marks]

(b) *NybbleShuffle* is a transposition cipher that operates on blocks of 32768 bytes. It splits each such block into 4-bit subblocks, and then rearranges these subblocks in pseudo-random order, under the control of a secret key K, in order to form the 32768-bytes long ciphertext block that it outputs. What is the smallest number of test blocks that you have to feed into an instance of the *NybbleShuffle* cipher in order to unambiguously reconstruct the permutation of subblocks that it applies, and how do you construct these test blocks?   [5 marks]

9 Security I (MGK)

(a) Let $\mathsf{Enc}_{K_{\mathrm{E}}}$ be the encryption function of an encryption scheme that provides indistinguishability under chosen plaintext attack (CPA security). Let $\mathsf{Mac}_{K_{\mathrm{M}}}$ be a message-authentication-code function that provides existential unforgeability. Named below are three techniques for applying these two functions together to a message M. For each of them

- briefly explain how $\mathsf{Enc}_{K_{\mathrm{E}}}$ and $\mathsf{Mac}_{K_{\mathrm{M}}}$ are combined, and

- state whether the resulting construct is likely to provide indistinguishability under chosen ciphertext attack (CCA security):

(i) encrypt-and-authenticate [2 marks]

(ii) authenticate-then-encrypt [2 marks]

(iii) encrypt-then-authenticate [2 marks]

(b) How can an attacker calling the C function `parse_text` below cause a buffer overflow? Explain how and why this works. [6 marks]

```c
#include <stdlib.h>
#include <string.h>
#define BUFLEN 4096
int check(int n) {
  if (n > BUFLEN) abort();
  return n;
}
void parse_text(char *text, size_t len) {
  char buf[BUFLEN];
  memcpy(buf, text, check(len));
  /* ... */
}
```

(c) Many Unix system administrators create a personal group for each of their users with this user as the sole member.

(i) What is the purpose of such a group? [2 marks]

(ii) Such personal groups typically have the same name and integer identifier as the corresponding user identifier. Is this practice compatible with the Windows NT mechanism for identifying users and groups? [2 marks]

(d) Give two examples for resources where an operating system is expected to implement residual information protection and two alternative mechanisms for implementing it. What are their tradeoffs and threat assumptions? [4 marks]

[*Note:* parts (b)–(d) not in *Cryptography* syllabus]

8   Security I (MGK)

(*a*) Block ciphers usually process 64 or 128-bit blocks at a time. To illustrate how their modes of operation work, we can use instead a pseudo-random permutation that operates on the 26 letters of the English alphabet:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
m	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$E_K(m)$	D	G	W	X	T	E	R	L	Y	Z	O	J	N	S	I	Q	P	C	U	H	B	V	F	A	M	K

As the XOR operation is not defined on the set $\{\texttt{A}, \ldots, \texttt{Z}\}$, we replace it here during encryption with modulo-26 addition (e.g., $\texttt{C} \oplus \texttt{D} = \texttt{F}$ and $\texttt{Y} \oplus \texttt{C} = \texttt{A}$).

(*i*)   Decrypt the following ciphertexts, which were encrypted using

(A)   Electronic codebook mode: `UOMHDJT`   [2 marks]

(B)   Cipher feedback mode: `RVPHTUH`   [4 marks]

(C)   Output feedback mode: `LNMSUUY`   [4 marks]

(*ii*)   Determine the CBC-MAC for the message `TRIPOS`.   [4 marks]

(*b*) Consider another small pseudo-random permutation, this time defined over the set of decimal digits $\{\texttt{0}, \texttt{1}, \texttt{2}, \ldots, \texttt{9}\}$, using modulo-10 addition instead of XOR (e.g., $\texttt{7} \oplus \texttt{3} = \texttt{0}$).

(*i*)   You have intercepted the message `100` with appended CBC-MAC block `4`. The message represents an amount of money to be paid to you and can be of variable length. Use this information to generate a message that represents a much larger number, and provide a valid CBC-MAC digit, without knowing the pseudo-random permutation or key that the recipient will use to verify it.   [4 marks]

(*ii*)   What mistake did the designer of the communication system attacked in part (*b*)(*i*) make (leaving aside the tiny block size), and how can this be fixed?   [2 marks]

11 Security II (MGK)

(*a*) Why does the formal security definition for collision-resistant hash functions require a key s and a security parameter n, even though most commonly used standard secure hash functions lack such input parameters?   [4 marks]

(*b*) If $h_s : \{0,1\}^* \rightarrow \{0,1\}^{\ell(n)}$ is a collision-resistant hash function, do the following constructions H_s also provide collision-resistant hash functions? Explain your answers.   [2 marks each]

(*i*) $H_s(x) = h_s(x) \parallel x$ (i.e. append x)

(*ii*) $H_s(x) = h_s(x) \parallel \mathrm{LSB}(x)$ (i.e. append least significant bit of x)

(*iii*) $H_s(x) = h_s(x \mid 1)$ (bitwise-or, i.e. set least significant bit of x to 1)

(*c*) Use Euler's theorem to calculate 5^{-1} mod 8.   [4 marks]

(*d*) The standard Digital Signature Algorithm (DSA) uses a cyclic subgroup $\mathbb{G} \subset \mathbb{Z}_p^*$ of the integers modulo a prime p, with prime order q, where q divides $p-1$.

(*i*)   Give two advantages of using a multiplicative subgroup of prime order, as opposed to just using $\mathbb{Z}_p^*$, in cryptographic schemes based on the Discrete Logarithm problem.   [2 marks]

(*ii*) Why is it possible to choose q substantially smaller than p, and what is an advantage of doing so?   [4 marks]

8 Security I (MGK)

(a) Compare and contrast the security definitions of a *pseudo-random generator* and a *pseudo-random function.* [4 marks]

(b) When a Windows NTFS access control entry (ACE) is inherited by a subdirectory, under which circumstances is the "inherit only" flag set or cleared, and why? [4 marks]

(c) What is *existential unforgeability* of a message authentication code? [4 marks]

(d) Which problem with CBC-MAC is fixed by ECBC-MAC, and how? [4 marks]

(e) A C program running on a 32-bit processor contains the following function:

```
void f(int *a, int l) {
  int *b, i;

  b = (int *) malloc(l * sizeof(int));
  if (b == NULL) return;

  for (i = 0; i < l; i++)
    b[i] = a[i];

  [...]
}
```

(i) How can a caller cause this function to overwrite unallocated memory? [2 marks]

(ii) Modify the function to remove this vulnerability. [2 marks]

[*Note:* parts (b) and (e) not in Cryptography syllabus]

9 Security I (MGK)

A smartcard application requires a stack (LIFO) for storing data records. This stack can become much larger than the tiny amount of memory available on the card. Fortunately, the card terminal has enough memory, and its API offers a class ExternalStack that can be invoked remotely from the card. It offers the usual four methods: a constructor to initialise an empty stack, push(R) to place a data record R onto the stack, isempty() to test whether the stack contains no record, and pop() to remove and return the top record from the stack.

The integrity of the stack is crucial for the security of the application. However, the card terminal is not tamper resistant and the adversary may have full control over the ExternalStack object. Therefore, you have to implement a SecureStack wrapper class that uses ExternalStack as an untrusted storage provider while guaranteeing the integrity of any data returned. The trusted on-card memory available to a SecureStack object is only 256 bytes.

Consider how to implement on the card a class SecureStack that provides the same four methods by appending additional data to records before pushing them onto ExternalStack, and verifying any data returned against locally held values. You have a message authentication function mac(K,(R,...)) and a cryptographic key/nonce generator function gen() available.

(a) Why is just appending a message authentication code to each externally stored record not sufficient? [2 marks]

(b) Write short pseudo-code for the four methods of SecureStack that shows how they call ExternalStack, how they update the on-card check data, and under which conditions a data-integrity alarm is raised. [10 marks]

(c) Express the internal check value(s) of your implementation after these calls:

```
Record r, r1, r2, r3;
SecureStack s;

s.push(r1);
s.push(r2);
r = s.pop();
s.push(r3);
```

[4 marks]

(d) Determine an upper limit for how often the push method of your implementation can be called securely. [4 marks]

9 Security II (MGK)

You are working on an encryption device with your new colleague, Mallory Baish, who proposes that you use a pseudo-random generator

$$r_i = h_1(s_i), \qquad s_{i+1} = h_2(s_i)$$

where $s_0 \in G$ is the random initial state and the other $s_i \in G$ are subsequent internal states, all invisible to adversaries. The $h_1, h_2 : G \to G$ are two secure one-way functions.

Adversaries may see any of the past outputs $r_0, \ldots, r_{n-1}$. If they can predict from those, with non-negligible probability, the next value r_n, then the security of your device will be compromised.

(a) Give a rough estimate for the probability that an adversary can predict r_n, as a function of n and $|G|$. Explain your answer. [6 marks]

(b) Mallory also suggests a specific implementation:

$$h_1(x) = f(u^x \bmod p) \qquad\qquad p = \text{a 2056-bit prime number}$$
$$h_2(x) = f(v^x \bmod p) \qquad\qquad u, v = \text{two numbers from } \mathbb{Z}_p^*$$
$$f(x) = x \bmod 2^{2048} \qquad\qquad\quad G = \mathbb{Z}_{2048}$$

    (i) The constants p, u and v will be known to the adversary. What conditions should they fulfill so that h_1 and h_2 can reasonably be described as one-way functions, and how would you normally generate suitable numbers u and v? [*Hint:* quadratic residues] [4 marks]

    (ii) If f were replaced with the identity function, how could an adversary distinguish the r_i emerging from this pseudo-random generator from a sequence of elements of $\mathbb{Z}_p^*$ picked uniformly at random? [4 marks]

    (iii) After you choose a value for p, Mallory urges you to use two particular values for u and v generated in your absence. You briefly see "$v = u^e \bmod p$" scribbled on a whiteboard. You become suspicious that Mallory is trying to plant a secret backdoor into your pseudo-random generator.

    Explain how Mallory could exploit such a backdoor. [6 marks]

9 Security I (MGK)

Block ciphers usually process 64 or 128-bit blocks at a time. To illustrate how their modes of operation work, we can use instead a pseudo-random permutation that operates on the 26 letters of the English alphabet:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
m	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
$E_K(m)$	P	K	X	C	Y	W	R	S	E	J	U	D	G	O	Z	A	T	N	M	V	F	H	L	I	B	Q

As the XOR operation is not defined on the set $\{\texttt{A}, \ldots, \texttt{Z}\}$, we replace it here during encryption with modulo-26 addition (e.g., $\texttt{C} \oplus \texttt{D} = \texttt{F}$ and $\texttt{Y} \oplus \texttt{C} = \texttt{A}$).

(a) Encrypt the plaintext "TRIPOS" using:

  (i) electronic codebook mode; [2 marks]

  (ii) cipher-block chaining (using IV $c_0 = \texttt{K}$); [4 marks]

  (iii) output feedback mode (using IV $c_0 = \texttt{K}$). [4 marks]

(b) Decrypt the ciphertext "BSMILVO" using cipher-block chaining. What operation should replace XOR? [4 marks]

(c) Your opponent is allowed to send you two plaintext messages M_0 and M_1, each n letters long. You now pick a new private key K, resulting in a new pseudo-random permutation $E_K : \{\texttt{A}, \ldots, \texttt{Z}\} \leftrightarrow \{\texttt{A}, \ldots, \texttt{Z}\}$. You also pick uniformly at random a private bit $b \in \{0, 1\}$ and return a ciphertext $C = c_0 c_1 \ldots c_n$, namely the message M_b encrypted with cipher-block chaining using the fresh E_K. Finally, your opponent has to guess your bit b.

Approximately how large must n be at least for your opponent to have a greater than 75% chance of guessing b correctly? Outline a strategy that your opponent can use to achieve this. [6 marks]

11 Security II (MGK)

(a) Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a private-key encryption scheme that operates on fixed-length messages from $\mathcal{M} = \{0,1\}^m$. Briefly explain a game that a user $\mathcal{U}$ of Π must be able to win against any polynomial-time adversary $\mathcal{A}$ with probability $\frac{1}{2} - \epsilon$ (where is ϵ is "negligible" with growing key length) for Π to be able to claim to offer "indistinguishable multiple encryptions under chosen-plaintext attack" (CPA security). [8 marks]

(b) Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a private-key encryption scheme that offers CPA security and operates on fixed-length messages $M \in \mathcal{M} = \{0,1\}^m$ with keys $K \in \mathcal{K} = \{0,1\}^\ell$. We use it to construct a new encryption scheme $\Pi' = (\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$. In which of the following cases is Π' also CPA secure? Explain your answer. [2 marks each]

(i) $\mathsf{Enc}'_K(M) = \mathsf{Enc}_K(M \oplus 1^m)$

(ii) $\mathsf{Enc}'_K(M) = \mathsf{Enc}_K(M) \| \mathrm{LSB}(M)$

(iii) $\mathsf{Enc}'_K(M) = \mathsf{Enc}_K(M) \| \mathrm{LSB}(K)$

(iv) $\mathsf{Enc}'_K(M) = \mathsf{Enc}_{0^\ell}(M)$

[*Note:* LSB outputs the least significant bit of its input word, $\|$ is concatenation.]

(c) While reviewing an implementation of AES-CBC, you discover that it simply uses the last ciphertext block from the previously encrypted message as the IV value C_0 for encrypting the next message. The implementation's author argues that as long as the IV of the very first message was chosen uniformly at random, all resulting subsequent ciphertext blocks will also be distributed uniformly at random, and therefore make good IVs. Why is this construction nevertheless not CPA secure? [4 marks]

8   Security I (MGK)

(*a*) In the Galois field GF(2^8) modulo $x^8 + x^4 + x + 1$, calculate

(*i*)   the difference $1100\,1010$ minus $1001\,0011$;   [2 marks]

(*ii*) the product $0100\,1011$ times $0000\,1001$.   [6 marks]

(*b*) Briefly explain two advantages that arithmetic in GF(2^{128}) has over arithmetic in $\mathbb{Z}_{2^{128}}$ when designing cryptographic algorithms.   [6 marks]

(*c*) Given a block cipher E_K and a corresponding decryption function D_K, provide a formula for the decryption of the following modes of operation and state for each whether the E_K or D_K calculations required during decryption can be executed in parallel: CBC, OFB, CTR.   [6 marks]

12 Security II (MGK)

The RSA public-key crypto system performs calculations in the group $\mathbb{Z}_n$, with $n = pq$ being the product of two large prime numbers p and q. The public key consists of the tuple (n, e), with $\gcd(\phi(n), e) = 1$, and the corresponding private key is (n, d). A message $m \in \mathbb{Z}_n$ is encrypted via $c = m^e \bmod n$ and decrypted as $m = c^d \bmod n$.

(a) Given p, q, and e, how can you apply the extended Euclidian algorithm to find a suitable d?   [6 marks]

(b) If we modified RSA to use as the public modulus a prime number instead of a composite of two large prime numbers, that is $n = p$ instead of $n = pq$, would this affect its security, and if so how?   [4 marks]

(c) In the *UltraSecure* virtual-private network, each router knows of each of its remote communication peers the RSA public key (n, e), which all have $e = 3$ and $2^{1023} \le n < 2^{1024}$. If router *alice* needs to establish a shared 256-bit AES secret key k with remote router *bob*, it looks up *bob*'s (n, e) and then uses this method:

- *alice* picks $k \in \{0, 1\}^{256}$ by reading 32 bytes from `/dev/random`

- *alice* interprets k as binary integer m with $0 \le m < 2^{256}$

- *alice* sends $c = m^e \bmod n$ to *bob*

- *bob* decrypts c into m and recovers k (by removing leading zeros)

Then *alice* and *bob* secure the rest of their communication with shared secret k.

(i)   How could an eavesdropper obtain m from c?   [4 marks]

(ii) Suggest a better method of using RSA to establish an AES key than the one given above.   [6 marks]

8   Security I (MGK)

Briefly explain

(*a*)   the function of a *salt value* in a password database   [3 marks]

(*b*)   *two* examples of covert channels in a file system protocol that is restricted to read-only operations under a mandatory access-control policy   [2 marks]

(*c*)   *three* types of common software vulnerabilities, with examples   [9 marks]

(*d*)   *two* problems solved by Cipher Block Chaining   [2 marks]

(*e*)   under which conditions will user U be able to remove a directory D in Berkeley Unix   [4 marks]

[*Note:* part (*b*), (*c*) and (*e*) not in Cryptography syllabus]

9 Security I (MGK)

(a) Name and briefly explain *three* increasingly demanding security properties expected from a *secure hash function h*. [3 marks]

(b) Explain how a 128-bit secure hash function h can be used to implement a one-time signature scheme, *and* how to verify such a signature. [5 marks]

(c) The T1000 encryption module was designed to make *TeleGroove* messages difficult to read for eavesdroppers. Such messages are character sequences of arbitrary length, encoded using a 5-bit alphabet $A = \{a_0, \ldots, a_{31}\}$. The T1000 module reads blocks of up to 1000 characters at a time into its memory $(m_0, m_1, \ldots, m_{999})$ and then outputs them again in a different order $(m_{K(0)}, m_{K(1)}, \ldots, m_{K(999)})$, according to a secret key table $K : \{0, \ldots, 999\} \to \{0, \ldots, 999\}$ that is shared with the respective recipient of each message. It repeats that process until the block ending with the last character has been processed.

    (i) What constraint on K ensures that no information is lost? [2 marks]

    (ii) Identify an ambiguity in the above description and propose a flaw hypothesis regarding a related residual-information vulnerability in this device. [4 marks]

    (iii) What is the smallest number of blocks that a chosen-plaintext attacker has to send through the device to recover K? Give an example of the content of such blocks and explain how to recover K from them. [6 marks]

2011 Paper 4 Question 8

(Computer Science Tripos Part IB)

Security I (MGK)

(*a*) In Windows NTFS, each file can have an associated access control list (ACL). Each entry has a type in $\{allow, deny\} \times \{explicit, inherited\}$.

    (*i*) What restriction does the Windows Explorer graphical user interface impose on the order in which these types of access-control entries can appear in an ACL? [4 marks]

    (*ii*) Give *one* example of a POSIX file access-control configuration for which an equivalent NTFS ACL violates this GUI restriction. [4 marks]

(*b*) Your colleagues used a pseudo-random function $f : \{0,1\}^{64} \to \{0,1\}^{64}$ in order to construct a permutation $g : \{0,1\}^{192} \to \{0,1\}^{192}$. The argument and return values of g are split into three 64-bit registers, respectively: $g(X_1, X_2, X_3) = (Y_1, Y_2, Y_3)$. The output of g is calculated as $Y_2 = f(X_1) \oplus X_2 \oplus f(X_3)$, $Y_1 = X_1 \oplus f(Y_2)$, and $Y_3 = X_3 \oplus f(Y_2)$, where $\oplus$ denotes bit-wise exclusive or.

    (*i*) Show that g is indeed a permutation. [4 marks]

    (*ii*) Show how an attacker who does not know f can efficiently distinguish g from most random permutations, after evaluating g on two different inputs. [4 marks]

    (*iii*) After you point out this shortcoming to your colleagues, they propose an improved variant $g'(X_1, X_2, X_3) = (Z_1, Z_2, Z_3)$ that adds another round to g: $Z_1 = Y_1$, $Z_2 = f(Y_1) \oplus Y_2 \oplus f(Y_3)$, and $Z_3 = Y_3$.

    Show how this variant still does not fix the problem of efficient distinguishability from most random permutations. [4 marks]

2009 Paper 5 Question 9

(Computer Science Tripos Part IB)

Introduction to Security (MGK)

(a) Make the following statements correct by changing one word or number. (Negating the sentence is not sufficient.)

   (i) The Advanced Encryption Standard defines a 16-round Feistel cipher.
   [1 mark]

   (ii) Files encrypted with Cipher Block Chaining start with a zero initial vector.
   [1 mark]

   (iii) Each user on a Unix system is identified by a unique prime number.
   [1 mark]

   (iv) The "read" bits associated with a Unix directory affect whether the files in its subdirectory "foo" can be accessed.
   [1 mark]

   (v) The "real user ID" associated with a Unix process determines its access rights.
   [1 mark]

(b) Name *five* examples of actions for which a Unix application will need to be invoked with *root* privileges.
[5 marks]

(c) Explain the attack on Double DES that motivates the use of Triple DES.
[6 marks]

(d) Under which conditions is the Vignère cipher unconditionally secure?
[4 marks]

1

2008 Paper 3/10 Question 8

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(a) A source of secure, unpredictable random numbers is needed to choose cryptographic keys and nonces.

   (i) Name *six* sources of entropy that can be found in typical desktop-computer hardware to seed secure random-number generators. [4 marks]

   (ii) What sources of entropy can a smartcard chip, like the one in your University Card, access for this purpose? [4 marks]

(b) As Her Majesty's prime hacker "001", on a mission deep inside an enemy installation, you have gained brief temporary access to a secret chip, which contains a hardware implementation of the DES encryption algorithm, along with a single secret key. You connect the chip to your bullet-proof laptop and quickly manage to encrypt a few thousand 64-bit plaintext blocks of your choice, and record the resulting 64-bit ciphertext blocks. You are unable to directly read out the DES key K used in the chip to perform these encryptions and you will not be able to leave the site without knowing K. But you know that all S-boxes in the last DES round are supplied in this chip via a *separate* power-supply pin. When you create a short-circuit on that pin, the encryption progresses as normal, except that the output of all S-boxes in the last round changes to zero.

   (i) Explain briefly the role of an S-box and the structure of a single round in DES. [4 marks]

   (ii) How can you find K, considering that your available time and computing power will not permit you to search through more than 10^9 possible keys? [8 marks]

2007 Paper 3/10 Question 9

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(*a*) You have received a shipment of hardware random-number generators, each of which can output one 128-bit random number every 10 milliseconds. You suspect that one of these modules has been tampered with and that it actually produces only 30-bit random numbers internally, which it then converts via a pseudo-random function into the 128-bit values that it outputs.

    (*i*) How does this form of tampering reduce the security of a system that uses a generated 128-bit random number as the secret key of a block cipher used to generate message authentication codes? [2 marks]

    (*ii*) Suggest a test that has a more than 0.5 success probability of identifying within half an hour that a module has been tampered with in this way. [6 marks]

(*b*) Explain briefly

    (*i*) the encryption and decryption steps of Cipher Feedback Mode; [3 marks]

    (*ii*) why some operating systems ask the user to press a special key combination (e.g., Alt-Ctrl-Del) before each password login; [3 marks]

    (*iii*) how a secure hash function h can be used to implement a one-time signature scheme; [3 marks]

    (*iv*) what happens if the same private key of the scheme from (*iii*) is used *multiple times*, to sign different messages. [3 marks]

1

2007 Paper 4/11 Question 8

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(a) Your colleague wants to use a secure one-way hash function h in order to store h(password) as password-verification information in a user database for which confidentiality might become compromised. For h, she suggests to use an existing CBC-MAC routine based on AES with all bits of the initial vector and the 128-bit AES key set to zero. Is this construct a suitable one-way hash function for this application? Explain why.   [8 marks]

(b) Explain how and under which circumstances overlong UTF-8 sequences could be used to bypass restrictions regarding which files an HTTP server serves.
                                                                                    [8 marks]

(c) Name *four* techniques that can be used to make buffer-overflow attacks more difficult.   [4 marks]

2006 Paper 4 Question 10 / Paper 11 Question 10

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(a) Alice and Bob participate in a public-key infrastructure that enables them to exchange legally binding digital signatures.

  (i) Name *two* reasons why, for some purposes, Alice might prefer to use a message authentication code, instead of a digital signature, to protect the integrity and authenticity of her messages to Bob. [4 marks]

  (ii) Outline a protocol for protecting the integrity and authenticity of Alice's messages to Bob that combines the benefits of a public-key infrastructure with those of using a message authentication code. [4 marks]

(b) Your colleague proposes a new way for constructing a message authentication code using a block cipher $E : \{0,1\}^{64} \times \{0,1\}^{128} \to \{0,1\}^{128}$. He takes the n-bit input message M, appends $p = 64 \cdot \lceil n/64 \rceil - n$ zero-bits, and splits the result into $k = (n+p)/64$ 64-bit blocks $M_1||M_2||\ldots||M_k = M||0^p$. He then calculates the message authentication code as

$$C_K(M) = E_{M_1}(E_{M_2}(E_{M_3}(\ldots E_{M_k}(K)\ldots)))$$

where K is the 128-bit secret key shared between sender and recipient. Show *two* different ways in which an attacker who observes a pair $(M, C_K(M))$ can, without knowing K, create a new pair $(M', C_K(M'))$ with $M' \neq M$. [6 marks]

(c) Show how a 128-bit message authentication code $C_K(M)$ with 64-bit key K can be constructed for an n-bit long message M using

  (i) a secure hash function $H : \{0,1\}^* \to \{0,1\}^{256}$, such as SHA-256; [2 marks]

  (ii) a block cipher $E : \{0,1\}^{128} \times \{0,1\}^{256} \to \{0,1\}^{256}$. [4 marks]

1

2005 Paper 3 Question 9 / Paper 10 Question 11

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(a) A and B play a simple game. A chooses a number $R_A \in \mathbb{Z}_3$ and B chooses a number $R_B \in \mathbb{Z}_3$. Then A and B communicate their respective choice to each other *simultaneously*, meaning that the players cannot change their choice after having seen that of the opponent. These rules decide who wins the game:

$$R_A \equiv R_B + 1 \pmod 3 \quad \Rightarrow \quad A \text{ wins}$$
$$R_B \equiv R_A + 1 \pmod 3 \quad \Rightarrow \quad B \text{ wins}$$

In any other case, the result of the game is a draw.

   (i) What complication arises when this game is played at a distance, for example via email? [2 marks]

   (ii) Suggest a cryptographic protocol that prevents cheating when this game is played via email. Your solution should not rely on a trusted third party. [6 marks]

   (iii) What assumptions do you make about the cryptographic functions used in your solution of (ii)? [3 marks]

   (iv) What assumptions do you make about the amount of computing power available to the opponent in your solution of (ii)? [3 marks]

(b) Outline briefly the purpose of an organisation's security policy and what steps should be considered in its development. [6 marks]

1

2004 Paper 3 Question 9 / Paper 10 Question 11

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(*a*) Explain briefly mechanisms that software on a desktop computer can use to securely generate secret keys for use in cryptographic protocols.   [5 marks]

(*b*) Give two different ways of implementing residual information protection in an operating system and explain the threat addressed by each.   [5 marks]

(*c*) Consider the standard POSIX file-system access control mechanism:

  (*i*) Under which conditions can files and subdirectories be removed from a parent directory?   [2 marks]

  (*ii*) Many Unix variants implement an extension known as the "sticky bit". What is its function?   [2 marks]

  (*iii*) On a POSIX system that lacks support for the "sticky bit", how could you achieve an equivalent effect?   [2 marks]

(*d*) VerySafe Ltd offer two vaults with electronic locks. They open only after the correct decimal code has been entered. The VS100 – a low-cost civilian model – expects a 6-digit code. After all six digits have been entered, it will either open or it will signal that the code was wrong and ask for another try. The VS110 – a far more expensive government version – expects a 40-digit code. Users of a beta-test version of the VS110 complained about the difficulty of entering such a long code correctly. The manufacturer therefore made a last-minute modification. After every five digits, the VS110 now either confirms that the code has been entered correctly so far, or it asks for the previous five digits again. Compare the security of the VS100 and VS110.   [4 marks]

2003 Paper 3 Question 9 / Paper 10 Question 11

(Computer Science Tripos Part Ib, Part II (General), Diploma)

Introduction to Security (MGK)

(a) Explain the difference between mandatory and discretionary access control.

[4 marks]

(b) (i) Explain the purpose and operation of cipher-block chaining (CBC).

[4 marks]

(ii) Explain how to decrypt a message in CBC. [4 marks]

(c) To protect her interview partners, a journalist needs to ensure that what she records with her digital camera cannot be viewed by anyone before she returns to her home country. You were asked to design for her a camera that encrypts recordings immediately before they are stored on tape. The question arises, how to handle the encryption key. If it is stored in the camera, it could be extracted if the hardware were confiscated and analysed. A key memorised by the user might be obtained using coercion, so this is not a suitable solution either.

Suggest *two* alternative convenient ways of arranging the encryption inside the camera such that decryption of the tape is only possible on the journalist's home computer. [8 marks]

1

2002 Paper 3 Question 2 / Paper 10 Question 4

(Computer Science Tripos Part Ib, Part II general, Diploma)

Introduction to Security (MGK)

(a) (i) Explain the collision resistance requirement for the hash function used in a digital signature scheme. [4 marks]

  (ii) Show how the DES block cipher can be used to build a 64-bit hash function. Is the result collision resistant? [4 marks]

(b) A sequence of plaintext blocks $P_1, \ldots, P_8$ is encrypted using DES into a sequence of ciphertext blocks. Where an IV is used, it is numbered C_0. Owing to a transmission error, one bit in ciphertext block C_3 changes its value, and as a consequence, the receiver obtains after decryption a corrupted plaintext block sequence $P'_1, \ldots, P'_8$. For the following modes of operation, how many bits do you expect to be wrong in each block P'_i?

  (i) cipher block chaining [2 marks]

  (ii) 64-bit output feedback [2 marks]

(c) (i) Explain the *Feistel principle* used by block ciphers such as DES and its purpose. [4 marks]

  (ii) Using a given pseudo-random function $F : \{0, 1\}^{100} \to \{0, 1\}^{100}$, construct a pseudo-random permutation $P : \{0, 1\}^{300} \to \{0, 1\}^{300}$ by extending the Feistel principle appropriately. [4 marks]

2002 Paper 7 Question 6

(Computer Science Tripos Part II)

Security (MGK)

(a) One of the algorithms that you implement inside a smartcard requires a stack (LIFO) for storing data records. Assume that each record is a few hundred bytes long. This stack can become very large and the small amount of memory available in the card is not sufficient to hold it. Therefore you decide that the stack object will be implemented via remote method invocation in the card terminal, which has enough memory.

The externally stored stack offers the usual four methods: `init` to initialise an empty stack, `push` to place a data record onto the stack, `isempty` to test whether the stack contains no record, and `pop` to retrieve the top record from the stack.

The integrity of the stack is crucial for the security of the application and the card terminal is not tamper resistant. Consider an algorithm for an integrity-protected stack object that will be implemented on the smartcard and uses an existing secure hash function h available in the card as well as the external stack object.

Explain:

  (i) Why is just adding a simple message authentication code to each externally stored record not sufficient here? [2 marks]

  (ii) What check data do you attach to externally stacked records, such that the memory required in the smartcard does not grow with the stack size? What check data remains inside the card? Show the resulting internal check values and the records on the external stack (call them $Y_1, Y_2, \ldots$) after you pushed three data records X_1, X_2, X_3 onto the stack. [6 marks]

  (iii) Write short pseudo-code for the methods of the protected stack object (`secure_init`, `secure_push`, `secure_pop`, `secure_isempty`) that shows how they update the on-card check data and under which conditions a tampering alarm is raised. [6 marks]

(b) In the same smartcard application, you also need an externally stored integrity-protected queue (FIFO). You decide to protect each externally stored record with a MAC for which a new key will be generated whenever the FIFO is initialised. What check data beyond the MAC key needs to be kept inside the card? What additional check data do you have to add to the records to guarantee the integrity of the FIFO? [6 marks]

1

2002 Paper 8 Question 6

(Computer Science Tripos Part II)

Security (MGK)

(a) Explain the concept of a *Trusted Computing Base* and outline its meaning in the context of the access control provided by a typical Unix workstation.
[5 marks]

(b) An automatic teller machine (ATM) communicates with a central bank computer for PIN verification. A 32-bit CRC code is added to each packet to detect transmission errors and then the link is encrypted using a block cipher in counter mode. Describe an attack that is possible in this setup. [5 marks]

(c) (i) Describe a cryptographic protocol for a prepaid telephone chip card that uses a secure 64-bit hash function H implemented in the card. In this scheme, the public telephone needs to verify not only that the card is one of the genuine cards issued by the phone company, but also that its value counter V has been decremented by the cost C of the phone call. Assume both the card and the phone know in advance a shared secret K.
[5 marks]

(ii) Explain the disadvantage of using the same secret key K in all issued phone cards and suggest a way around this. [5 marks]