Unit 7 · Cryptography and Trust

WORKSHEET

Full marks 31

Name _____ Class _____ Roll no. ____ Date _____ Marks _____ / 31

1 Key words [4]

Fill in the blanks with words from the box. Two words in the box are not needed. Each blank carries 0.5 mark.

certificate	**discrete logarithm**	**factoring**	**forward secrecy**	**inverse**	**Kerckhoffs's principle**	
modulus	**nonce**	**padding**	**replay**			

The number you divide by in clock arithmetic is the _____. 23 is the _____ of 7, mod 40, because 7 × 23 leaves 1. Fresh random data mixed with a message before it is encrypted, so that the same message never gives the same ciphertext, is _____. Going back from g^a mod p to a is the _____ problem. Choosing new secret numbers for every connection, so that old recordings can't be read even if a long-term key is stolen later, gives _____. Sending an old signed message again is a _____. In proof of work, the number tried again and again until the block's hash starts with zeros is the _____. The rule that a system must stay safe even when everything about it is known except its key is _____.

2 Multiple choice [5]

Choose the correct option for each of the following questions.

i. A forger changes record 3 of a hash chain of 10 records, and wants nobody checking the chain to notice. What must they do? [1]

a. Nothing more: only record 3 changes.
b. Change record 4 only.
c. Recompute block 3's hash and every block after it, and hope nobody kept a copy of the old last hash.
d. Change the first block.

ii. What is the inverse of 3, mod 10? [1]

a. 3 **b.** 7
c. 9 **d.** 3 has no inverse mod 10

iii. Eve has Ram's public key and a ciphertext that she knows is either 1 (yes) or 2 (no), encrypted with textbook RSA. How can she tell which? [1]

a. She can't without factoring n.
b. She encrypts 1 and 2 herself with the public key and sees which matches.
c. She asks Ram for his private key.
d. She decrypts it with the public key.

iv. A signed message “Transfer Rs 2,000 to Hari” arrives twice, with the same good signature. What does the signature prove about the second copy? **[1]**

a. That the owner of the key wanted two transfers.

b. Only that the owner of the key signed this message at some time; the second copy could be a replay.

c. That someone has stolen the private key.

d. Nothing: a second copy can’t have a good signature.

v. Which of these follows Kerckhoffs’s principle? **[1]**

a. A bank keeps its encryption method secret so attackers can’t study it.

b. A bank uses a published, well-studied method and keeps only its keys secret.

c. A bank writes its own cipher so that no attacker has seen it before.

d. A bank puts its key in its app so that customers don’t need one.

3 Trace the program **[4]**

Fill in the trace table: one row for each time round the loop, with the values at the end of the round, then what the program prints.

```
1  def power_mod(a, k, n):
2      result, square = 1, a % n
3      while k > 0:
4          if k % 2 == 1:
5              result = result * square % n
6          square = square * square % n
7          k = k // 2
8      return result
9
10 print(power_mod(3, 13, 7))
```

k **at the start**	result	square	k **at the end**

OUTPUT

4 What does it print? [3]

Write exactly what the program prints, in the box under it.

Sita and Ram agree on a key over an open line, with tiny numbers.

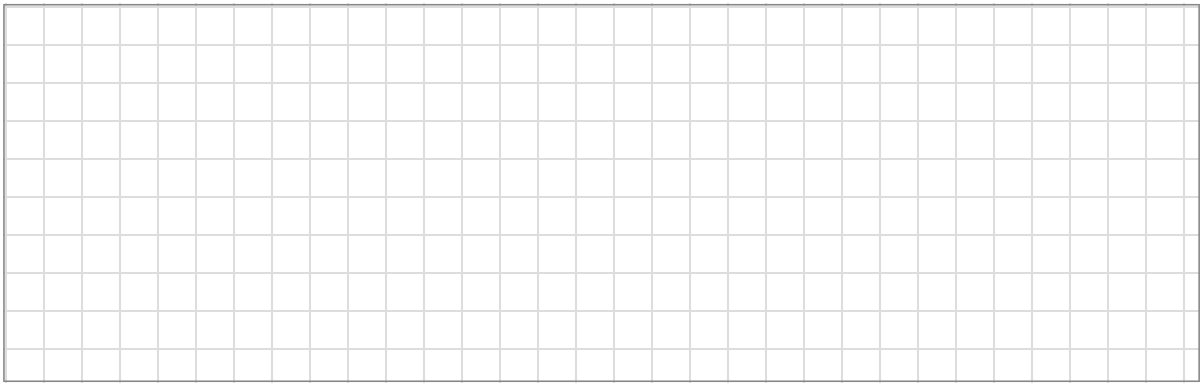
```
p, g = 11, 2
a, b = 3, 7                  # the two secrets, never sent
A = pow(g, a, p)
B = pow(g, b, p)
print("sent:", A, B)
print("keys:", pow(B, a, p), pow(A, b, p))
print("Eve multiplies:", A * B % p)
```

OUTPUT

5 A small key protects nothing [4]

Work it out in the space below.

Ram's public key is $(n, e) = (221, 5)$. i. Find the two primes whose product is n, and $\varphi(n)$. ii. Find d, the private power, and check it. iii. Encrypt the message $m = 10$ with Ram's public key. iv. Why can't anyone do step i for a real key?



Answer: _____

6 Review the code [3]

Read the program, then answer.

A school's notice board signs its notices with the office's key pair. A reviewer finds three places where the program uses cryptography badly. For each, write the line, what harm it could allow, and the fix.

```
import hashlib
import random

N, E = 3233, 17                       # the office's public key
OFFICE_KEY = 2753                     # the private power, for signing

def digest(text):
    return int(hashlib.sha256(text.encode()).hexdigest(), 16) % N

def new_session_id():
    return str(random.getrandbits(64))

def accept(notice, signature, sent_on):
    if sent_on < "2083-01-01":
        return True                   # older notices were never signed
    return pow(signature, E, N) == digest(notice)
```

7 Short answer [2]

Answer the following question.

A message "Pay Sita Rs 500. Ram" carries a signature that checks with Ram's public key. What two things does this prove? Name two things it doesn't prove.

8 Long answer [4]

Write a long answer to the following question.

A municipality wants to put its land ownership records "on a blockchain to stop fraud". Give two problems a blockchain would solve and three it wouldn't. Would you advise it? Explain.

9 Think [2]

Think, then answer.

In a key exchange, Eve hears everything that crosses the line. Why can't she simply copy the shared key as it goes past? What could she do instead, and what stops it?

10 Activity: Agree on a key in public (not marked)

Work in threes: Sita, Ram and Eve, on paper. Everyone uses $p = 23$ and $g = 5$. Sita and Ram each choose a secret from 2 to 10 and never say it. Each works out 5 to the power of their secret, mod 23 (reduce after every multiplication), and says the answer aloud. Then each raises the number they heard to their own secret, mod 23. Eve writes down what she hears and tries to find Sita's secret. Do it three times, changing roles.

A, said aloud by Sita	B, said aloud by Ram	Sita's key	Ram's key	Eve's guess at Sita's secret