

Security, privacy and data integrity

A-Level Computer Science 9618 · Topic 6 revision notes

Everything from the 3 lessons of this topic on a few pages: the key idea of each lesson, the diagram to remember, the points to learn and the traps examiners set. Read it, cover it, test yourself.

1. Security, privacy and integrity of data

2. Threats to computer systems and how to protect them

3. Validation and verification

1 Security, privacy and integrity of data

Syllabus 6.1

Data security means protecting data from loss, damage and unauthorised access. **Data privacy** is about who has the right to see and use personal data, and is protected by law (data protection legislation). **Data integrity** means the data is accurate, complete, consistent and up to date. Security measures protect privacy and integrity, but they are not the same thing. A system needs **user accounts** and **authentication** (passwords, biometrics, digital signatures) to know who is using it, **access rights** to limit what each user can do, and **encryption** so that stolen data cannot be understood.

three different questions about the same data

Security

Is the data protected from loss and from people who should not get it?

passwords, firewall, encryption, backups

Privacy

Who has the right to see and use personal data, and for what purpose?

data protection laws, consent, access rights

Integrity

Is the data accurate, complete, consistent and up to date?

validation and verification

Fig. 1: security, privacy and integrity ask different questions about the same data

Worked example: Data or system?

A school's network is infected with ransomware that locks every computer until a fee is paid. The student records are backed up off site each night. Explain what has been protected and what has not.

1. The **data** is protected against loss: the off-site backup means the student records can be restored, losing at most one day of changes.
2. The **system** was not protected: malware got in, so there was no effective anti-malware, firewall or update policy.
3. The school still loses the use of its computers until they are cleaned and restored, so lessons and administration are disrupted.

Answer: Backups protected the data; the system itself was not secure, so the service was lost.

One idea per mark.

- ✓ **Security:** protecting data from loss, damage and unauthorised access.
- ✓ **Privacy:** who has the right to see and use personal data; protected by law.
- ✓ **Integrity:** data is accurate, complete, consistent and up to date.
- ✓ **Measures:** name the measure, then say how it works and what it protects against.

Watch out!

Do not define one term using another ("privacy means keeping data secure"). Privacy is about *who has the right to see personal data*; security is about *protecting data*; integrity is about *data being correct*.

A backup protects against *loss of data*. It does nothing to stop unauthorised access: a stolen backup is a privacy breach unless it is encrypted.

Encryption does not prevent hacking or interception. Write that it makes intercepted data *unintelligible (meaningless)* to anyone without the key.

2 Threats to computer systems and how to protect them

Syllabus 6.1

A network lets threats reach a computer from anywhere. **Malware** is software written to cause harm: a **virus** copies itself into other files and can delete or corrupt data; **spyware** secretly records what the user does, such as key presses, and sends it to a third party. **Hackers** gain unauthorised access to a system. **Phishing** uses a genuine-looking email to lure the user to a fake website; **pharming** installs code or alters DNS so the user is redirected to a fake website even when the correct address is typed. Each threat is reduced by a matching measure: **firewall**, **anti-virus**, **anti-spyware**, **passwords** and **access rights**, and **encryption**.

layers of protection between the internet and the data

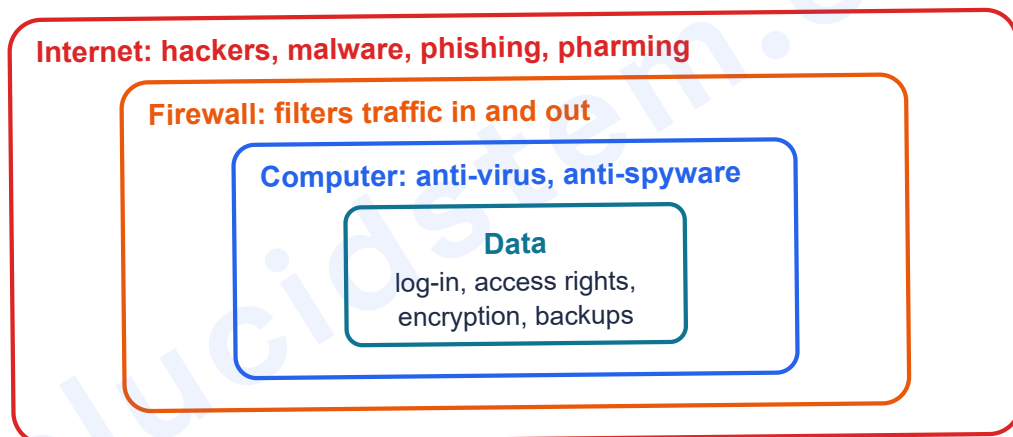


Fig. 1: defence in depth: each layer stops threats the outer layers missed

Worked example: Describe a measure

Describe how a firewall helps protect a school network from hackers. [3]

1. It sits between the school network and the internet, and all traffic passes through it.
2. It examines each incoming and outgoing packet and checks it against a set of rules (criteria), such as allowed IP addresses and ports.
3. Packets that do not meet the rules are blocked, so unauthorised access attempts from outside are stopped (and it can log them).

Answer: Position, inspection against rules, blocking: one mark each.

Threat, harm, measure.

- ✓ **Name and describe:** say what the threat is and what harm it does.
- ✓ **Match:** pick the measure that actually stops that threat.
- ✓ **Explain how:** a firewall filters traffic by rules; anti-virus compares with signatures; encryption makes data unintelligible.
- ✓ **Context:** a stand-alone PC and a network need different measures; use the scenario in the question.

Watch out!

Phishing and pharming are easy to mix up. In **phishing** the user has to click a link in a message; in **pharming** the user types the genuine address and is redirected without doing anything wrong.

Do not offer "anti-virus" or "firewall" as the answer to every threat. Match the measure to the threat, and explain how that measure stops that threat.

thelucidstem.com

3 Validation and verification

Syllabus 6.2

Validation is an automatic check, done by the computer, that data is **reasonable** and follows set rules: range, format, length, presence, existence and limit checks and check digits. **Verification** checks that data has been **copied correctly**: during data entry by a **visual check** or **double entry**, and during data transfer by a **parity check** (byte or block) or a **checksum**. Both protect data integrity, but neither can prove that data is actually true: a valid, correctly typed date of birth can still be the wrong person's.

two different checks on the way into the system

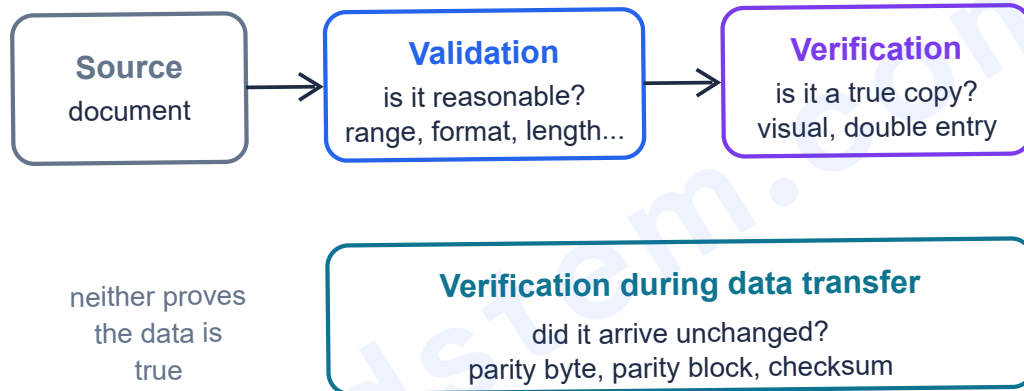


Fig. 1: validation asks whether data is reasonable; verification asks whether it was copied or transferred exactly

Worked example: A format and length check

Write a function `ValidID` that returns `TRUE` only if its parameter is exactly six characters: two upper-case letters followed by four digits.

1. Length check first: if `LENGTH(ID)` is not 6, the ID is invalid.
2. Use `MID` to take one character at a time. Characters 1 and 2 must lie from "A" to "Z"; characters 3 to 6 must lie from "0" to "9".
3. A flag starts `TRUE` and is set to `FALSE` by any failing character:

```
FUNCTION ValidID(ID : STRING) RETURNS BOOLEAN
  DECLARE Valid : BOOLEAN
  DECLARE Index : INTEGER
  DECLARE Ch : STRING
  Valid ← TRUE
  IF LENGTH(ID) <> 6 THEN
```

```

Valid ← FALSE
ELSE
  FOR Index ← 1 TO 6
    Ch ← MID(ID, Index, 1)
    IF Index ≤ 2 THEN
      IF Ch < "A" OR Ch > "Z" THEN
        Valid ← FALSE
      ENDIF
    ELSE
      IF Ch < "0" OR Ch > "9" THEN
        Valid ← FALSE
      ENDIF
    ENDIF
  NEXT Index
ENDIF
RETURN Valid
ENDFUNCTION

```

Answer: ValidID("AB1234") returns TRUE; ValidID("A12345") and ValidID("AB123") return FALSE.

Reasonable, then exact.

- ✓ **Validation:** automatic check that data is reasonable and follows rules; it cannot prove data is correct.
- ✓ **Verification:** checks data was copied exactly: visual check or double entry at entry; parity or checksum in transfer.
- ✓ **Check digit:** weight, add, MOD 11, subtract from 11 (10 = X, 11 = 0); a valid code totals a multiple of 11.
- ✓ **Parity block:** the failing row and failing column cross at the wrong bit.

Watch out!

Validation cannot tell you the data is correct, only that it is reasonable. A mark of 67 typed instead of 76 passes a 0 to 100 range check.

A parity check misses an even number of changed bits in the same byte, because the count stays even. And neither validation nor verification can tell if the source document itself was wrong.