KK04 — Security controls used to protect development practices and data stored within applications

Cyber Security · Comprehensive exam study guide

STUDY DESIGN — YOU MUST KNOW

- version control and code repositories
- robust identity and access management
- encryption
- code review
- regular updates and patches to software
- separated development, testing and production environments

Controls are the **defences** that reduce the vulnerabilities in KK03. For each, know what it does and which risk it addresses — questions often pair a vulnerability with the control that mitigates it.

Controls and what they protect

Control	What it does	Mitigates
Version control & code repositories	Track changes; restore earlier versions; control who can change code.	Lost work; unauthorised/untracked code changes.
Robust IAM	Strong authentication, least-privilege permissions, role-based access.	Poor IAM; insider threats; unauthorised access.
Encryption	Encodes data so only authorised parties can read it (in storage and transit).	MITM attacks; data theft/exposure.
Code review	Peers examine code for bugs and security flaws before release.	Ineffective code review; insecure code.
Regular updates & patches	Apply fixes promptly to close known flaws.	Unpatched software; known exploits.
Separated dev/test/prod environments	Keep development, testing and live systems apart.	Exposure of real data; unstable code reaching users.

Least privilege

REMEMBER

‘Robust IAM’ includes the **principle of least privilege**: give each person only the access they need — limiting both insider threats and the damage from a compromised account.

Key definitions to memorise

Security control	A measure that prevents, detects or reduces a security threat.
Encryption	Encoding data so that only authorised parties holding the key can read it.
Least privilege	Granting each user only the minimum access rights needed to do their job.

Code review	Examination of source code by others to find defects and security flaws before release.

Exam technique

EXAM TIP

- Be ready to pair a control with the vulnerability it addresses (e.g. encryption → MITM).
- Mention least privilege and role-based access when discussing IAM.

COMMON MISTAKES

- Saying encryption 'stops hackers getting in' — it protects data confidentiality, even if intercepted.
- Treating code review as testing — it is human inspection of code, complementary to testing.

Quick revision checklist

- ■ I can describe each control and the risk it mitigates.
- ■ I can recommend controls to secure a described environment.

Exam practice — questions & model answers

Q1. State the security control that best mitigates each: (a) unpatched software; (b) data intercepted in transit; (c) weak, over-broad access rights.

State · 3 marks

Model answer

- (a) Regular updates/patches (1).
- (b) Encryption (1).
- (c) Robust IAM / least privilege (1).

Q2. Explain how separated development, testing and production environments improve security.

Explain · 2 marks

Model answer

- Real production data is not used during development/testing, reducing exposure (1),
- and only tested, stable code is promoted to the live environment, reducing risk to users (1).

Q3. Describe how code review reduces security risks.

Describe · 2 marks

Model answer

- Other developers examine the code before release (1)
- and can spot bugs and security flaws the author missed, so they are fixed before deployment (1).