Software Engineering Principles

Secure By Design

Implemented on (date)	
Approved by (name & role)	Head of Software Practices (Fahad Anwar) In consultation with TAG (Technical Advisory Group).
Last review on (date)	13/03/2024
Reviewed by	TAG (Technical Advisory Group)
Next review due on (date)	
Principle owner (name)	Head of Software Practices In consultation with Technical Advisory Group (TAG) representing Software Engineers, Cloud Division, TISS and Security Division.
Principle owner (division)	Digital Services and Technology (DST)
Main point of contact (name)	Fahad Anwar Software Engineering Head of Practice
Status	Final Draft
Published version link	

Principle Review Record

This Review Record is to be completed on each time a review is conducted. Its purpose is to maintain a record of reviews, recording who conducted the review, the date of the review and the outcome of the review (fit for purpose, amendment required, principle no longer required, etc).

This principle is to be reviewed annually.

Review No	Review Conducted By	Review Date	Review Outcome
01			

Amendment Details

Date	Amendment Summary	Amended by	Version

RASCI (For detail please – RASCI Information document)

Responsible	G6 Program Managers through Technical Leads, G7 and SEO's
Accountable	Head of Software Practices
Supportive	Head of Cloud Functions (Amazon, GCP, Azure) SAIM SIRA Software Engineering Community of Practice (SE-CoP)

Consulted	Technical Advisory Group (TAG) representing Software Engineers, Cloud Division, TISS and Security Division.
Informed	Senior Leadership Team Software Engineering Community SAIM SIRA Design Authority Chair

Secure by Design

Systems will be designed and maintained with the assumption that our software and the data they hold will be attacked and possibly compromised.

Rationale

The cost of handling a security breach is significantly greater than the cost of hardening the system in the first place. There is ever increasing evidence that government is demonstrating a strong commitment to data privacy, acknowledging the growing concerns surrounding data breaches. If a breach does occur, we can minimise the scale and cost by detecting and mitigating it as soon as possible and being proactive in identifying risks and threats to our systems. By considering security from the beginning, we limit not only the financial impact but also the significant reputational impact that would be incurred in the event of a breach.

Implications

- Teams must take responsibility for the security of their software throughout its lifecycle. This includes how they detect incidents, as well as how they respond to and resolve issues in the event of an attack or breach.
- Teams must be aware of data privacy implications including policy and legal compliance.
- Data needs to be appropriately classified and secured, both in transit and at rest.
- Teams must understand security risks and model threats to their system.
- The security of both the underlying platform used and the software being deployed to it, including any open source and 3rd party dependencies, must be considered. This applies on an ongoing basis, as threats evolve alongside the software, and needs continual review.
- People at all points of the engineering process need strong awareness of current security techniques and principles, and how to apply these appropriately throughout the lifecycle of the software.
- Teams are expected to have training to ensure security best practice is considered at each step of software development.
- Teams are expected to make use of appropriate tools and techniques to identify vulnerabilities as early in the software development lifecycle as possible.

Questions to be considered

- What security risks does your solution have?
- Will your solution use or collect sensitive or special category data?
- How will your solution securely interact with other systems?

- How will your solution's security integrate with your organisation's departmental security and processes?
- How will your solution security meet the relevant ONS Security Principles (Security principles 2018) and go beyond that standards where needed?
- Do you have access to the security expertise and skills you need?
- How will you source the security expertise and skills you need?
- What changes to your organisation's security documentation and processes will your solution need?
- How will you provide appropriate security assurance, both throughout the duration of the software development and in service?

Motivation for the principles

- https://www.security.gov.uk/guidance/secure-by-design/principles/
- https://www.gov.uk/guidance/the-technology-code-of-practice
- https://engineering-principles.jlp.engineering/
- https://github.com/otto-de/tech_manifest
- https://www.gov.uk/government/publications/security-policy-framework/hmg-security-policy-framework
- https://www.ncsc.gov.uk/collection/cyber-security-design-principles/cyber-security-design-principles

Principle Enforcement

G6 Program Managers through Technical Leads or Sr. member of respective software development team is responsible for monitoring and enforcing principle compliance.

Exception process

It is acknowledged that situations arise in which the Principal Detail as above may not be able to be met. Where this is the case, a documented Principal exception must be sought from the Principal Owner (specified in the table on the title page).