**Joint Advisory on Data Protection Risks Arising from Insecure APIs by the Personal Data Protection Commission (PDPC) and Cyber Security Agency of Singapore (CSA)**

Application Programming Interfaces ("APIs") enable different software applications and systems to communicate and exchange information. For example, APIs allow an organisation's website or mobile application to retrieve customer information from its database or enable third-party service providers to access information needed to deliver services. APIs provide interfaces to data and systems and facilitate business operations and data sharing, but they may also introduce cybersecurity and data protection risks if they are not adequately secured and monitored.

Past incidents have shown how threat actors may exploit poorly secured APIs, weaknesses in access control management or compromised credentials to gain unauthorised access to personal data[1]. In some cases, attackers may repeatedly use APIs to exfiltrate large volumes of personal data. Such activities often go undetected by security controls, particularly where API requests appear to come from authorised users or applications.

Organisations are required to make reasonable security arrangements to protect personal data in their possession or under their control[2]. Organisations should therefore ensure that APIs which access or process personal data are appropriately designed and secured throughout their lifecycle, including when they are introduced, modified or connected to third-party systems.

This advisory highlights key areas of cybersecurity and data protection risks and recommends measures that organisations should consider mitigating these risks. These measures reinforce relevant practices in the PDPC's Guide to Data Protection Practices for ICT Systems (revised 20 July 2026) ("ICT Guide"), with a particular focus on safeguarding personal data accessed through APIs.

**Illustrations**

Case A:
Security researchers discovered an API that was publicly accessible over the Internet and protected only by weak, default credentials. Once authenticated, they were able to execute queries to systematically retrieve personal data from an underlying database containing millions of records. Despite the scale of potential exposure, the activity went undetected as the queries were made through legitimate API functions using valid credentials.

Case B:
Internal API endpoints, originally intended only for communication between trusted internal systems, became publicly accessible over the Internet after an existing network access restriction was removed to accommodate a third-party integration. No alternative safeguards were put in place to compensate for the removal of this control, and the organisation remained unaware of the exposure until the incident was discovered.

[1] See for example Marina Bay Sands Pte. Ltd., where a misconfigured API endpoint and weak default credentials allowed a threat actor to access and exfiltrate personal data of approximately 665,495 individuals; and Tech in Asia, where a legacy API endpoint lacking authorisation token verification was exploited to scrape personal data of approximately 220,000 individuals.

[2] Section 24 of the Personal Data Protection Act 2012

The cases above illustrate how API security weaknesses can lead to unauthorised access to personal data. Organisations should consider the following key areas of risks and recommended measures to address these risks.

## Key Areas of Risk

### A. Unintended exposure of personal data through APIs

i. APIs may unintentionally expose personal data when they are poorly designed, incorrectly configured, or return more information than necessary to fulfil a request. Such over exposure of data increases the risk of personal data being accessed by unauthorised parties.

ii. Risks may also arise when internal-facing APIs become internet-accessible due to inadequate network restrictions, or when security settings are not properly maintained following system updates, migrations, or changes involving third-party applications. Without adequate oversight of deployed APIs, organisations may remain unaware of such weaknesses.

### B. Default authentication and credential management

i. APIs using default, or inadequately managed credentials may be vulnerable to unauthorised access. An unauthorised party may be able to retrieve personal data using permissions associated with the compromised account.

ii. Attackers may also obtain valid API credentials, such as API keys or access tokens[3], through compromised accounts or third-party applications, and use these to access personal data through legitimate API functions. Consequently, successful authentication alone may not be sufficient to prevent unauthorised access to personal data.

### C. Inadequate access controls and authorisation

i. APIs may expose personal data when users or applications are granted excessive privileges or when authorisation checks are inadequate. Such weaknesses may allow unauthorised access even where authentication is functioning correctly.

ii. A common vulnerability is Broken Object Level Authorisation (BOLA)[4], where an API fails to verify whether a user is authorised to access a particular record. For example, an authenticated user may be able to manipulate a request to retrieve another user's personal data, potentially allowing attackers to systematically access personal data belonging to multiple individuals.

[3] API keys and access tokens are digital credentials used by applications or systems to access services or information. Authentication checks identity, whereas authorisation checks which records or functions may be accessed.

[4] Broken Object Level Authorisation (BOLA) occurs when an API does not adequately verify that a user or application is permitted to access a particular record. It is identified as API1:2023 in the OWASP API Security Top 10.

**D. Inadequate API security configurations**

i. Security settings intended to protect APIs may be incorrectly configured, inadvertently omitted or removed during system migrations and modifications. Without appropriate verification and security testing, these weaknesses may remain undetected and allow unauthorised access to personal data.

ii. APIs that transmit data over insecure connections may also be vulnerable to Man-in-the-Middle (MITM) attacks, where attackers intercept communications between systems to obtain sensitive information, including personal data and authentication credentials.

iii. APIs that do not adequately validate user inputs may also be vulnerable to injection attacks, such as SQL injection, where attackers submit malicious inputs that could result in unauthorised access to personal data or compromise of affected systems.

**E. Inadequate Logging and Monitoring of API Activities**

i. Organisations that do not adequately log and monitor API activities may fail to identify unusual access patterns or excessive data retrieval. Where API weaknesses are exploited, threat actors may make repeated requests using valid credentials or existing API functions to systematically retrieve large volumes of personal data. As such activity can appear legitimate, it may continue undetected, increasing the potential impact of a data breach.

## Recommended Measures

1. Maintaining visibility over APIs

- Know what APIs exist and what personal data they process. Work with your organisation's IT teams and service providers to maintain an up-to-date inventory of APIs that access or process personal data. The inventory should include the purpose of each API, its owner, the types of personal data involved, how the data is exchanged, and any third-party systems with access. API use should be regularly reviewed so that obsolete APIs are removed or disabled where appropriate.

2. Minimising personal data exposure

- Reduce unnecessary collection, use and disclosure. Review the personal data collected, used or disclosed through each API to ensure that only information necessary for its intended purpose is processed or returned. Where appropriate, implement additional safeguards to restrict the specific data fields accessible through APIs, particularly where more sensitive personal data is involved.

3. Implementing appropriate API security configurations and access controls

- When implementing API security configurations, organisations should reference relevant security guides and frameworks, such as the OWASP API Security Top 10, OWASP cheat sheets, and the CIS API Security Guide, as well as vendor-specific guidance where applicable.

*Strengthen API Access Controls and Authorisation*

- Restrict access to authorised users, systems and data. Ensure that APIs are accessible only to the intended users, applications or systems. For example, APIs meant for internal use should not be accessible from the Internet.

- Ensure that APIs verify the identity of users or applications and enforce appropriate authorisation controls over the records, data fields and functions they are permitted to access. For example, an API should not allow a logged-in customer to retrieve another customer's records simply by changing a customer or record identifier in a request. Organisations should test for such weaknesses when developing or modifying APIs.

- Apply the principle of least privilege by granting users, applications and third-party service providers access only to the personal data and functions required for their intended purposes. Review these permissions regularly and remove access for users that is no longer necessary. Stronger safeguards should be considered for accounts with privileged access to ICT systems containing personal data.

*Protect APIs Against Malicious Requests and Data Interception*

- Secure data transmitted through APIs. Ensure that API communications are encrypted using HTTPS with Transport Layer Security (TLS) to protect personal data and authentication credentials against unauthorised interception or modification.

- Implement input validation and secure coding practices. Validate API inputs on the server side and adopt secure coding practices, such as parameterised queries, to prevent injection attacks that could result in unauthorised access to personal data.

- Deploy appropriate security controls to protect APIs. Consider implementing a Web Application Firewall (WAF) to detect and block malicious requests to internet-facing APIs, including common web-based attacks such as SQL injection.

4. Protecting API credentials

- Prevent unauthorised access through weak, default or compromised credentials. Where password-based authentication is used, enforce strong, non-default passwords and appropriate safeguards against unauthorised authentication attempts. Implement multi-factor authentication (MFA) where applicable, particularly for privileged or administrative access. Ensure that API keys and access tokens are securely stored and managed, not stored in the clear (or in plaintext), and are not unnecessarily exposed in source code, publicly accessible files or configuration settings. Review the validity and permissions of these credentials regularly and ensure that compromised or unnecessary credentials can be promptly revoked.

5. Strengthening change management

- Prevent security gaps during system changes. Document API security configurations and conduct appropriate reviews and testing before and after deploying new or modified APIs, including following system upgrades and migrations. Organisations should ensure that existing security controls remain effective and correctly applied after such changes. Where practicable, automate deployment processes to reduce human errors, or implement appropriate independent verification checks where manual configuration is necessary.

- Changes involving the removal or modification of existing security controls, including those required for third-party integrations, should be subject to appropriate risk assessments and verification of alternative safeguards before implementation.

6. Conducting API security assessments and testing

- Identify and remediate API security weaknesses, including those associated with the OWASP API Security Top 10. Include APIs in relevant security assessments, such as secure code reviews and penetration testing, to identify weaknesses in authentication, access permissions and system configurations. Security testing should be conducted before deployment, periodically and following significant changes. Identified weaknesses should be rectified in a timely manner, with further testing conducted to verify that the fixes are effective.

7. Implementing API activity logging and monitoring

- Detect suspicious API activities. Organisations should maintain logs of API activities commensurate with the volume of personal data processed and the risks involved, including the users or applications making requests, the time of access, the functions accessed and, where appropriate, the volume of data retrieved.

- Where the volume of personal data processed or the risks involved are higher, organisations should consider establishing monitoring and alerting to identify suspicious behaviour, such as unusually high request volumes, repeated access to different customer records, or unexpectedly large data retrievals. Monitoring should also cover requests made using valid credentials, as these may be misused by attackers.

8. Introducing safeguards against excessive data extraction

- Limit the scale of potential data exposure. Consider measures such as limiting the number of API requests within a given period (known as rate limiting), restricting the amount of data that can be retrieved in a single request, and setting appropriate thresholds for data exports. Where the risks involved are higher, organisations may also consider implementing automated controls to detect or prevent unusual patterns of data retrieval. These measures should complement, and not replace, proper access control management and monitoring process.

9. Establishing incident response procedures

- Respond and contain security incidents by ensuring that suspected API-related security incidents can be promptly investigated and contained. As part of its data breach management plan, organisations should have procedures to revoke compromised credentials, suspend suspicious access, investigate security alerts and assess whether personal data has been affected. Consider running tabletop exercise simulation of API-related security threat scenario to familiarise stakeholders to the procedures and gauge its effectiveness and response. Where investigations confirm that a data breach involving personal data has occurred, organisations should also consider their notification obligations under the Personal Data Protection Act, including whether affected individuals or the Personal Data Protection Commission should be notified.

- Organisations may refer to CSA's Incident Response Checklist as a high-level guide for preparedness, response and recovery to cyber incidents.

## Maintaining API Security Over Time

Securing APIs should not be treated as a one-off exercise. New systems, updates to existing applications, changes to third-party integrations and newly discovered security vulnerabilities may introduce new data protection risks, even where appropriate safeguards were previously in place. Organisations should therefore regularly review the effectiveness of their API security arrangements and ensure that identified weaknesses are addressed in a timely manner.

Organisations should establish clear responsibilities for maintaining API security, with reviews and testing conducted periodically and whenever significant system changes or new vulnerabilities arise. This includes ensuring that API-related data protection risks are considered and addressed as part of the organisation's broader data protection and risk management processes.

The recommended measures in this advisory highlight common API-related data protection risks and provide a starting point for addressing them. They are not exhaustive, and organisations should assess their specific risks and implement safeguards appropriate to the nature, sensitivity and volume of personal data involved. DPOs are encouraged to work with their IT teams and service providers and refer to the following resources for more comprehensive guidance. Organisations may refer to CSA's Technical Advisory on Secure API Development for more detailed technical guidance on secure API design, development and testing. SMEs may also wish to explore available government assistance schemes to support the implementation of cybersecurity measures.

## Additional Resources

- CIS API Security Guide
- CSA Incident Response Checklist
- CSA SG Cyber Safe Programme
- CSA Technical Advisory on Secure API Development
- Government Assistance for SMEs (Cybersecurity)
- NIST Guidelines for API Protection for Cloud-Native Systems
- OWASP API Security Top 10 (2023)
- OWASP Cheat Sheet Series
- OWASP REST Security Cheat Sheet
- PDPC Guide to Data Protection Practices for ICT Systems (revised 20 July 2026)