API Security Checklist

This checklist provides a comprehensive overview of critical aspects to consider when securing your APIs, from authentication and authorization to encryption, monitoring, and incident response.

Feel free to tailor it to your specific needs and regularly review and update it to adapt to evolving security threats and best practices.

API Security Parameters	**Action to be Taken**
Authentication	
Use strong authentication methods	Implement OAuth, API keys, JWT, or other secure methods.
Implement multi-factor authentication	Add an extra layer of security for user authentication.
Authorization	
Enforce role-based access control	Limit access based on user roles and permissions.
Implement granular authorization	Control access to specific API endpoints and resources.
Data Encryption	
Use HTTPS for data in transit	Encrypt data transmitted between clients and server.
Encrypt data at rest	Secure data stored in databases or file systems.
Rate Limiting	
Implement rate limiting	Control the number of requests per time interval.
Prevent abuse with throttling	Slow down requests when rate limits are exceeded.
API Key Management	
Secure storage of API keys	Safeguard API keys to prevent unauthorized access.
Rotate API keys regularly	Change keys to minimize the risk of key compromise.

Input Validation	
Sanitize user inputs	Filter and validate user inputs to prevent attacks.
Parameterize SQL queries	Use parameterized statements to prevent SQL injection.
Logging and Monitoring	
Implement comprehensive logging	Record API activities for auditing and troubleshooting.
Monitor for suspicious activities	Detect and respond to unusual patterns in real-time.
Incident Response	
Develop an incident response plan	Prepare for handling security incidents effectively.
Communicate breaches responsibly	Notify affected parties promptly and responsibly.
Third-Party API Integration	
Assess third-party APIs for security	Evaluate their security features and reputation.
Implement safeguards for integration	Use API gateways, tokens, and secure data handling.
Lifecycle Management	
Securely retire APIs	Disable access, delete data, and follow best practices.
Update and patch regularly	Keep APIs and dependencies up to date with security fixes.