20/09/2024

Introduction to API Security

* API Security Fundamental

1, Introduction - Why API Security
2, Real world API Breaches
3, OWASP API Security Top 10
4, 3 Pillars of API Security
5, Application Security Technology Landscape
6, Conculsion & Best Pratices

Why API Security

API → Application Programming Interface

Explosive Growth

83% of all the internet traffic is from API's

Major Attack Target

2022:- API's "most frequent attack vector

High Profile Beaches

High Profile API Beaches announced weekly

Regulatory Compliance

Regulations mandate privacy, vulnerability detection, testing

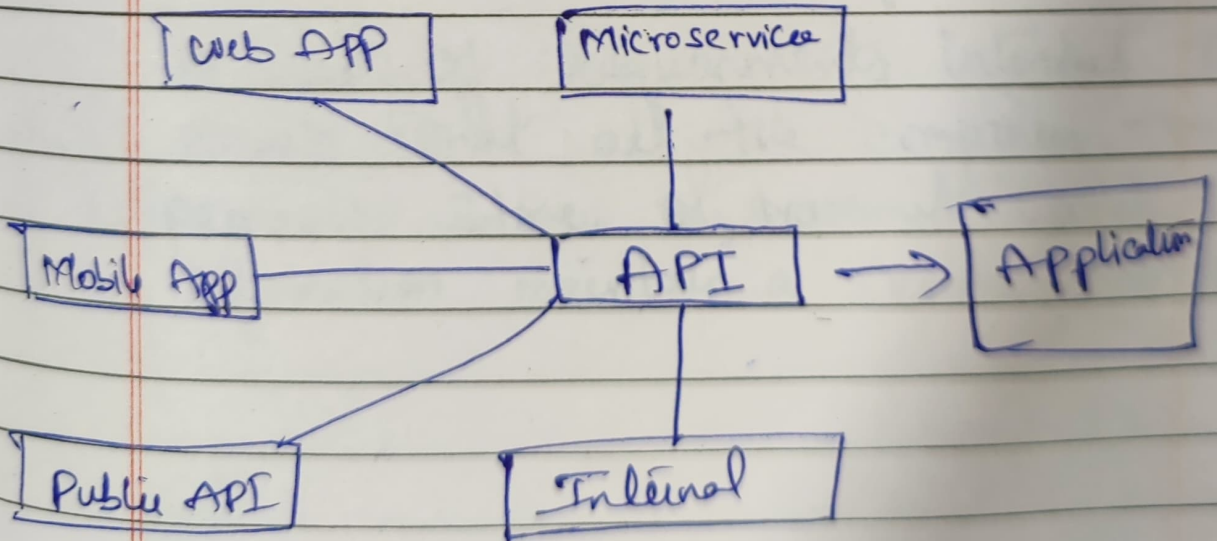
What Being Stolen

1. Credit Card Data
2. Payements Data
3. Bumble users Data
4. Linkedin & Facebook Data of users

Why Attackers Target API's

API's

1. Direct Access to sensitive Data
2. Often "over-permissioned"
3. Vulnerable to logic flows



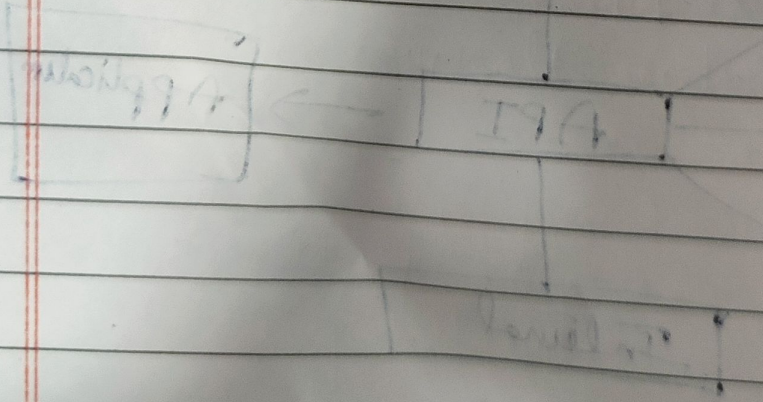
Classic Cyber Attack

1. Reconnaissance
2. Weaponize
3. Infiltration
4. Lateral Movement
5. Peivilage Escalation
6. BREACH

Kill Chain

API Attack

1. Find Vulnerability
2. Attack



Regulatory Compliance

1. Banking :-

FFIEC → Federal Financial Institution Examination Council

OCC → Office of the Comptroller of the Currency

Open Banking → It allows you to share the data with another financial Service provider

FDX - Financial Data Exchange

2. Payment Card Industry :- (PCI)

A Set of requirments intended to ensure that all the companies that process Store, or transmit credit card information maintain a Secure Environment

3. Healthcare :-

HIPAA | HITRUST | Interoperability

- HIPAA :- Health Insurance Portability and Accountability Act

HITRUST :- Stands for Health Information Trust Alliance It is a Common Security Framework (CSF) primarily designed to help healthcare Companies protect and manage sensitive data.

Interoperability :- The ability of different systems devices or applications to communicate and work together

4. Privacy

GDPR, CCPA, PIPEDA

GDPR :- General Data Protection Regulations is a comprehensive data protection law enforced across EU Region to Protect the privacy and personal data of EU citizen

2018

→ Data minimization :- Collect only necessary data

→ Consent :- Obtain consent from individuals to collect and process their data

→ Right to Access & Erasure :- Individuals can request access to their data and ask for it to be deleted (also known as Right to be forgotten)

→ Data Protection by Design :-
Organization must integrate data protection measures into the development of new system & services

⇒ Breach Notification :-
Companies must notify authorities and affected individuals of any data breach within 72 hours

Why GDPR is important

GDPR protects individuals from data misuse, ensuring that their personal information is treated with respect It also holds businesses accountable compelling them to adopt stronger data security measures and transparency practices, ultimately building Trust between companies and their customer

1. Consent
2. Data Security
3. Right to Access & Forgotten
4. Data Breach notification within 72 hours

If xyz fails to comply with GDPR, it may facing severe fines - up to 4% of its global revenue or €20 million

CCPA → California Consumer Privacy Act
1st Jan 2020
24th July 2020

This law protects the privacy of California residents by giving them more control over the personal information that businesses collect about them

→ Know what personal information a businesses has collected about them and how it's used and shared

→ Delete personal information collected from them, with some exceptions.

→ Opt-out of the sale or sharing of their personal information

→ Not be discriminated against for exercising their CCPA rights

PIPEDA → Personal Information Protection and Electronics Documents Act

It is a Canadian law relating to data privacy. It governs how private sector organizations collect, use and disclose personal information in the course of commercial business

13th April 2000

DPDPA → Digital Personal Data Protection Act

The Digital Personal Data Protection Act (DPDA) of 2023 is India's first comprehensive data protection law.

It regulates the processing of digital personal data in India and aims to protect the privacy of individuals. The DPDPA applies to organization that process personal data of individuals in India and to processing outside if it's for offering goods or service in India

→ Data Fiduciaries
→ Data Subjects
→ Data protection board
→ Penalties
→ Exclusions

Federal: Fed RAMP

The Federal Risk and Authorization Management Program is a United States Federal government-wide compliance program that provides a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services

Standard Frameworks

NIST 800-53 Standard

National Institute of Standards and Technology.

It provides a list of controls that supports the development of secure and resilient federal information systems

Competing Challenges

Security

1. Secure Operation of web interfaces
2. Regular vulnerability Testing
3. Rapid review and remediation of findings

Privacy

1. Protection of PII, user Data
2. Breach Notification requirements
3. Massive penalties for violations

Accessibility

1. Global push to make data accessible
2. Interoperability, Open Banking
3. "Information Blocking" Penalties

ISO/IEC 27001

An International Standard for managing information security.

It is a widely recognized Standard that helps organization of all sizes and sectors manage information security risk.

The Standard was originally published in 2005, revised in 2013 and again in 2022

SOC -2

Service Organization Controls 2, is a Cyber security compliance framework that Evaluates how a service organization protects client data

5 principals of SOC-2

1. Security
2. Availability
3. Processing Integrity
4. Confidentiality
5. Privacy

Real-World API Breaches

1. Coinbase → Unauthorized trading
2. United State Postal Servie → Account data harvesting
3. Venmo → Excess data Exposure
4. Instagram → Account Take over
5. Bumble → Account Tampering
6. T-Mobile → SEC Reporting
7. OPTUS → Ransom
8. Experian → 3rd party exposure

CoinBase

→ Used Scraped API Calls from Web UI
→ Identified 4 Key parameters of any coinbase transactions
→ Manipulated the parameters via API Calls
→ Sold crypto they DID NOT OWN

OWASP API - I

Broken Object level Authorization

They Paid $250,000

United State Postal Service

→ USPS relied on traditional code and web scanners

→ Found missing API authentication
→ USPS added authentication
→ USPS left out authorization
→ User A able to access User B details (any of 60M accounts)

Peloton

News:-

Pelton leaky API let anyone grab riders private account data

What happened

→ Open API allowed requests for user details with NO authentication

→ 4m Users account details Exposed (including) Joe Biden

→ Including accounts marked private
→ Researcher reported to pelton
→ No response after 90 days
→ "Fixed" vulnerability by adding authentication
→ But hackers could still access all records, just needed to authenticate

OWASP API #1 Broken Object Level Authorization
OWASP API #2 Broken Authentication

Venmo

News

I Scraped Millions of Venmo Payments
Your Data Is at Risk

What Happend

→ Venmo home page presented live feed of transactions
→ Hackers Sniffed traffic and Identified API Calls
→ Wrote 20-line Script, using 2 IP's
→ Pulled 115K transactions / day - even with rate limiting in place
→ API returned ALL transactions details
→ 207M transactions harvested

OWASP API #2 → Broken Authentication
OWASP API #3 → Broken Object property level Authorization
OWASP API #4 → Unrestricted Resource Consumption

Filter At Backend level Not At UI level

Instagram

News

Instagram Says Bug Gave Hackers Data On "High-Profile" Users

What Happend

→ Account reset requires 6 digit code
→ Researcher found API to submit reset code guesses
→ Guesses limited to 200 per IP
→ Researcher demonstrated could rotate through 5,000 IPs in seconds
→ Enables takeover of ANY accounts

Reward was $30,000

OWASP API #1
Broken Object Level Authorization
OWASP API #2
Broken Authentication

Bumble

News

Dating Site Bumble leaves Swipes Unsecured for 100M Users

What Happend

→ API permitted access to 95M user account details w/o authentication

→ Incremental ID's allowed easy scraping of entire database

→ Enabled calculations of user's exact location via triangulation

→ API allowed paid features to be enabled without proper privileges

OWASP API #1
Broken Object Level Authorization
OWASP API #2
Broken Authentication
OWASP API #5
Broken Function Level Authorization

T-Mobile

What Happened

The preliminary results from our investigation indicates that the bad actor obtained data from this API for approximately 37 Million current postpaid and prepaid customer accounts

Optus

What Happened

→ API endpoint required no authentication to access

→ Attacker harvested 9.8 M user details and threatened $1M ransom

→ Data included driver's license Medicare, ID's, name, phone, email

OWASP API #1
Broken Object Level Authorization

OWASP API #3
Broken Object property level Authorization

OWASP API #4
Unrestricted Resource Consumption

Experian

News:-

Experian API Exposed Credit Score of Most Americans

What Happend

→ Experian partner site offered loan eligibility feature

→ Feature used Experian API for lenders to automate credit card score lookups

→ Attacker Sniffed API calls
→ API accessible with no authentication
→ Results delivered with name, address and *any* value for date of birth

OWASP API #1
Broken Object level Authorization

OWASP API #3
Broken Object Parameter level Authorization
OWASP API #9
Improper Inventory Management

OWASP API Security Top - 10 2023

OWASP →

Open Web Application Security Project

owasp.org

API - 1 Broken Object level Authorization
API - 2 Broken Authentication
API - 3 Broken Object Property level Authorization
API - 4 Unrestricted Resource Consumption
API - 5 Broken Function level Authorization
API - 6 Unrestricted Access to Sensitive Business flow
API - 7 Server Side Request Forgery
API - 8 Security misconfiguration
API - 9 Improper Inventory Management
API - 10 Unsafe Consumption of APIs

(BOLA)

API #1 2023

Broken Object level Authorization

Broken Object Level Authorization

-> Most common & damaging API vulnerability

Manipulation of API's to Access data/object belonging to other users

Risk Exposure :-

Can lead to data loss, disclosure, data manipulation

Example :-

Attacker authenticates as User A and then retrieves data on User B

Prevention

Implemented automated testing to find BOLA flaws

(BA)

API #2 2023

Broken Authentication

Broken Authentication

→ Weak & poor authentication creates vulnerability

1. Missing security controls
2. Poorly implemented controls

Risk Exposure

Account Takeover, Data theft, Unauthorized Transactions

Example:-

1. Weak password requirements
2. Brute force ID/pwd
3. No captcha / rate limit / lockout
4. Auth info in URL's (tokens, passwords)
5. Non validation of Token Expiration
6. Insecure password Storage

Prevention :- Continous Testing

Define authentication policies & Standards

(BOPLA)

API #3 2023

Broken Object Property Level Authorization

Broken Object property level Authorization
→ Exploit of endpoints by reading and/or modifying values of objects

Ability to update object elements
(a mass assignments)

Revealing unnecessary sensitive data
(a excessive data exposure)

Risk Exposure

1. User is able to set user account from Standard to premium
2. User reduit url endpoints with excessive unnecessary details, name, email, address, id

Prevention

Ensure user can only access legitimate permitted fields

Return only minimum amount of data required for the use case.

(URC)

API #4 2023

Unrestricted Resource Consumption

→ Unrestricted Resource Consumption

Abuse of API's due to high volumes of API calls, large requests, etc

Formely "Lack of Resource and Rate limiting"

Risk Exposure

→ Denial of Service
→ Performance Impact
→ Mass data harvesting

Example:-
Missing Rate limit control
Execution Time out
Max allocable memory
Max number of file & size limit
Excessive operations in a single request
Excessive records returned in single request

Prevention

1. Implementation Traffic control
2. Test effectiveness of control

(BFLA)

API #5 2023

Broken Function level Authorization

Broken Function level Authorization
→ Using API functionality to modify (create, update, delete) resource of another user.

Often involves, replacing passive methods (GET) with active (PUT, DELETE)
Can be used to escalate privilege

Risk Exposure

Provide attacker access to unauthorized endpoints & functionality

Can be exploited to modify account details, access administrative functions
Example:- Replacing GET with PUT
Modify URL parameter ("role=admin", "account type=premium")
Deleting an invoice, setting account balance = $0 (or negative)
Prevention: Controls to limit access & Continous testing to check

(UASBF)

API #6 2023

Unrestricted Access to Sensitive Business Flow

Unrestricted Access to Sensitive Business Flow

Abuse a legitimate business workflow through execution, automated use
Rate limit, captchas not always effective against fraudulent traffic

Rapid IP rotation makes detection difficult
Typically a result of application flow

Risk Exposure

Loss of critical business Activity
Example:- Mass Automated ticket Purchasing
High volume referral bonuses

Prevention

Identify critical business workflows. Implement fraudulent traffic detection and control
Setup of automated Testing of control mechanism

(SSRF)

API #7 2023

Server Side Request Forgery

Server Side Request Forgery

Exploiting URL inputs to make a request to a malicious, 3rd party server

Risk Exposure

SSRF creates a channel for malicious requests, data access or other fraudulent activity

Potential for data leaks

Example:- (Local file injection (LFI)
Malware downloaded from malicious site

Prevention:- Validate & Sanitize ALL user supplied information, including URL parameters

TEST URL validation effectiveness

(SM)

API #8 2023

Security Misconfiguration

Security Misconfiguration

Broad category encompasses lack of hardening to unnecessary service
Use of bots to scan, detect and exploit misconfiguration

Risk Exposure

Misconfiguration can expose sensitive user data
Potential for full server compromise

Example:- Lack of security hardening
Improperly configured permissions
Missing security patches
Unnecessary features enabled
Missing TLS
CORS policy missing / improperly set

Prevention:- Implement automated, continous security testing

((CSEA)

API #10 2023

Unsafe Consumptions of APIs

→ Exposures can occur via use of 3rd party API's, which are generally trusted. However, 3rd parties can be Exploited, which can be used to attack API's that rely on them

Risk Exposure

Data theft, Breach, account takeover

Example:- Attacker inserts malicious address data and gets Exploited
Attacker compromises 3rd party API causing it to respond with redirect to malicious site. Client blindly follows redirects without validation

Prevention :- Validate data returned by 3rd party API's
Evaluate security control of 3rd party API.
Encrypt all API communication
Maintain approved list of known locations of all API's

(~~UCA~~)
(11m)

API #9 2023

~~Unsafe~~ Consumptions of ~~API's~~

Improper Inventory Management-

→ Unauthorized API access via old unused API version, or through trusted 3rd parties

Risk Exposure

Data / Account theft via unretired APIs
Exposure of sensitive data via improperly secured 3rd party APIs

Example:-

Old version of API's
Unpatched endpoints
Endpoints with weaker security
Outdated documentation
Unnecessarily exposed endpoints

Prevention → Deploy / manage all api's in Gateway. Define rules for versioning and retirement. Periodically audit 3rd party acces

Governance

Benefits

1. Consistency
2. Setting Expectations
3. Establishing Standard process
4. enforcing security

Awareness

→ know your API
→ know your Data
→ know your Risks

Policy & Process

→ Engineering process
→ API Documentation
→ Design guides

3 Pillars of API Security

1. Goverance → Developing Secure APIs
2. Testing → Ensure API's are free of flaws
3. Monitoring → Detecting threats in production

Know Your API

→ Get full Inventory API's
- Purpose, Owner, documentation

→ Standardize and Enforce API deployment process
- Existence of "shadow/rouge" API sign of weak governance
- API's only deployed in approved way's, with proper validation
- Enforce governance at Gateway, Marketplace

→ Mandate API Documentation
- Make Sure API's are consistent and reusable
- Define documentation requirements

→ Create API Development Standards

- Style guides, authentication requirements versioning, PII Tracing

Know Your Risk - Threat Modeling

- **Identify** :- API's business flows, data access paths
- **Access** :- Vulnerabilities, logic flaws, access controls, 3rd party risk
- **Probability** :- Examine the likelihood of an Attack
- **Impact** :- Understand the damage, loss consequences of an attack
- **Mitigation** :- Develop a plan to address the risk

Know your Data => Data

Documentation : OpenAPI Specification

1. Not optional
2. Industry standards for REST APIs
3. Machine - readable (YAML, JSON)
4. Aids development and 3rd party integration
5. Also aids security testing
6. Manually or auto-generated
7. Control what public vs private
8. Retire old documentation - huge threat vector

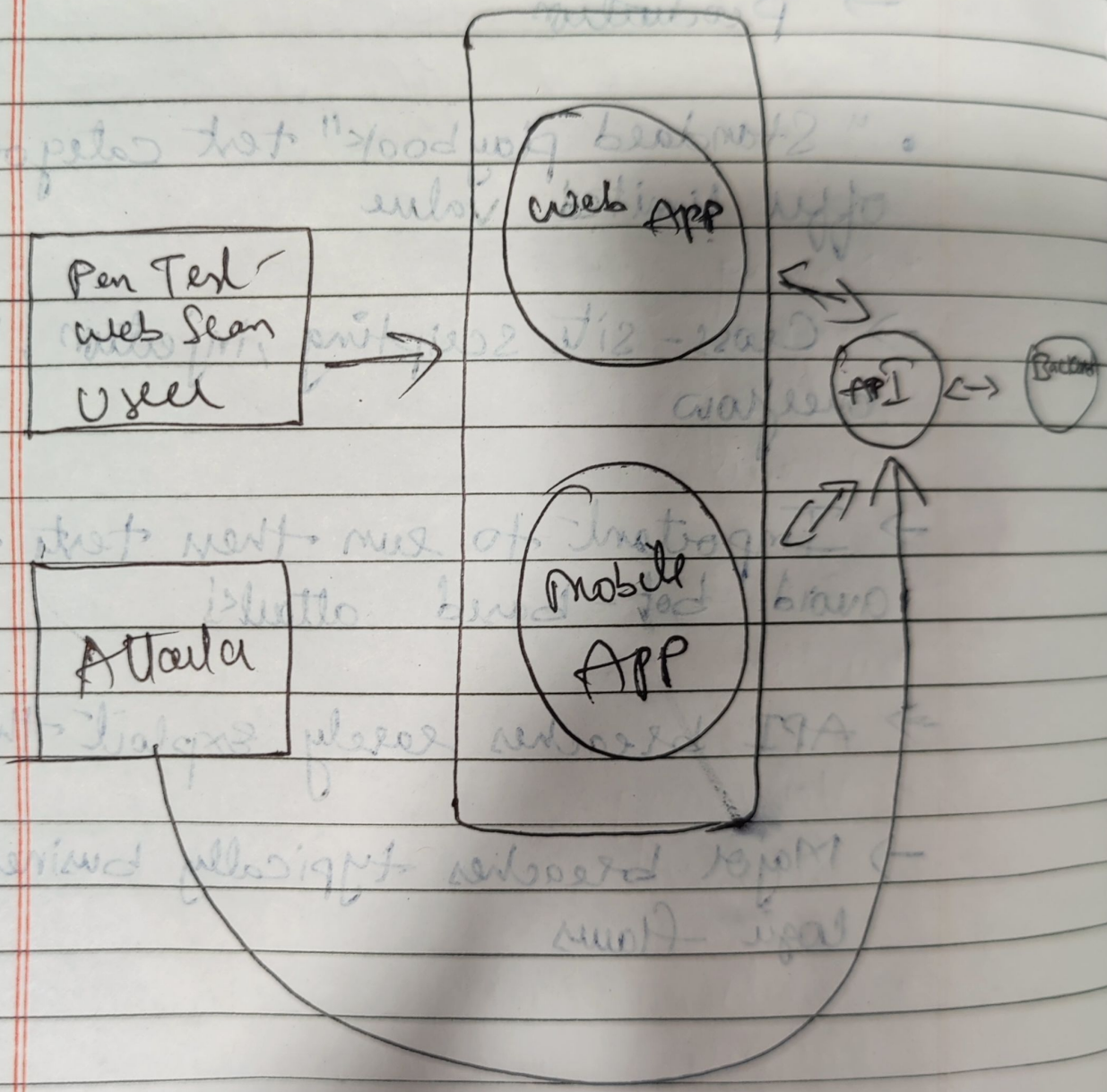
Defines API capability (contract)

1. Title, description, version
2. Base-URL
3. Endpoints, paths
4. Request, response payloads
5. Authentication requirements
6. Parameters, data types
7. Methods

Design Guides: Promote Governance, Consistency

- Authentication :- type (basic, token, certificate), how to implement
- Authorization :- Who has access to What, Where Enforced
- Naming Convention :- URI's on nouns, Methods are verbs, pluralization, hierachy, Case, Language, no jargon abbreviations
- Error Codes :- Status codes, reference ID human readable messages
- Versioning :- When to increment, when not, types of versions
- Units, Formats, Standards - date/time formats, timezones

The Need for API-First Testing



Testing

- Where do you want to find API vulnerabilities
  - → Pre-production
  - → Production

- "Standard playbook" test categories offer limited value

  → Cross-site scripting, injection, buffer overflow

  → Important to run these tests to avoid bot-based attacks

  → API breaches rarely exploit these

  → Major breaches typically business logic flaws

Testing Security

API Security

1. Unsecured Endpoints
2. Authentication Exploits
3. Enumeration
4. App DOS, Rate limit
5. Missing TLS, SSL Issue
6. Injection, fuzzing
7. Fuzzing, Input validation
8. Server-Side resource forgery
9. Server properties leaks

Data Security

1. Access Control
2. Excessive data Exposure
3. Sensitive data Exposure
4. Personal, health, bank data
5. File, directory Exposure
6. Encryption at rest
7. Data Exfiltration

Monitoring Approaches

Proactive : Blocking

-> API Gateway
-> Web App Firewall

Reactive : Alerting

-> Logging ; SIEM
-> Runtime API threat Management

Monitoring

- Runtime Protection

1. Policy enforcment
2. Authentication
3. Traffic filtering

Threat Detection

1. Fraudulent traffic
2. Distributed attacks
3. Incident response

Control Validation

1. Verify API contexts
2. Uncover anomalies

Business logic

- Cross - account access
- API - function abuse
- Role - based access control
- Pen - testing

API Discovery

- Monitoring can aid API inventory efforts

→ Identify API endpoints in use
→ Discover undocumented / unknown API's

- Comprehensive discovery requires more sources

→ API Gateway, Web Application firewall
→ Code Repository
→ Application Testing, Crawling

- Reliance on traffic based - discovery misses

→ Internal API traffic not seen by traffic analysis tool

→ Pre-production API's
→ Unexercised endpoints

Limitations of monitoring

- Difficult to get full visibility

Requires sensors on every network segment

- High false positives on threat detection

→ Live traffic contains limited context
→ Difficult to identify data access violations in real-time
→ API monitoring tools typically used in alert only

- SaaS-based monitoring requires sharing traffic with 3rd parties

→ Privacy concerns
→ Bandwidth requirements

- Traffic blocking solutions can add latency

API Security Technology Landscape

Network	Endpoints	Application	Data
1. Firewall	1. EDR	1. SAST, DAST	1. Encryption
2. NAC	2. Anti Malware/Virus	2. SCA, Container	2. DLP
3. Threat Protection	3. Vulnerability	Security	3. Access
4. Deception	Management	3. WAF	Control
		4. API Security	

Web/Messaging	Identity	Detection	Cloud
1. Email Gateway	1. Authentication	1. SIEM	1. CASB
2. Web Gateway	2. PIM	2. SOAR	2. Infrastructure
	3. Identity	3. Incident	Security
	Governance	Response	

Pre-Production

Production

Develop

Static Code Analysis

- Coding weakness
- Injection flaws
- weak authentication
- Configuration Issues

Software Composition Analysis

- 3rd party vulnerability
- Licensing issues
- Outdated components

Deploy

API Security Testing

- OWASP Testing
- Business logic
- Authentication Testing
- Authorization Testing
- Attack Simulation

Operate

API Gateway

- Authentication
- Authorization
- Rate limiting (Limited)
- Traffic filtering
- Logging

API Threat management

- Attack detection
- API Discovery
- Anomaly detection
- Traffic blocking

Recommendations

- Shift left: find Issues as early as possible
- Test security controls in production
- Automate as much as possible

API Security

1. Use HTTPS
2. Use OAuth 2
3. Use Web Authn
4. Use leveled API keys (R, W, D)
5. Implement Authorization (view, Editor)
6. Rate limiting
7. API versioning v1 — v2 — v3
8. Allow Listing
9. Check OWASP API security Rule
10. Use API Gateway
11. Error Handling
12. Input validation
13. Authentication
14. Authorization
15. Output Encoding
16. Logging & Monitoring