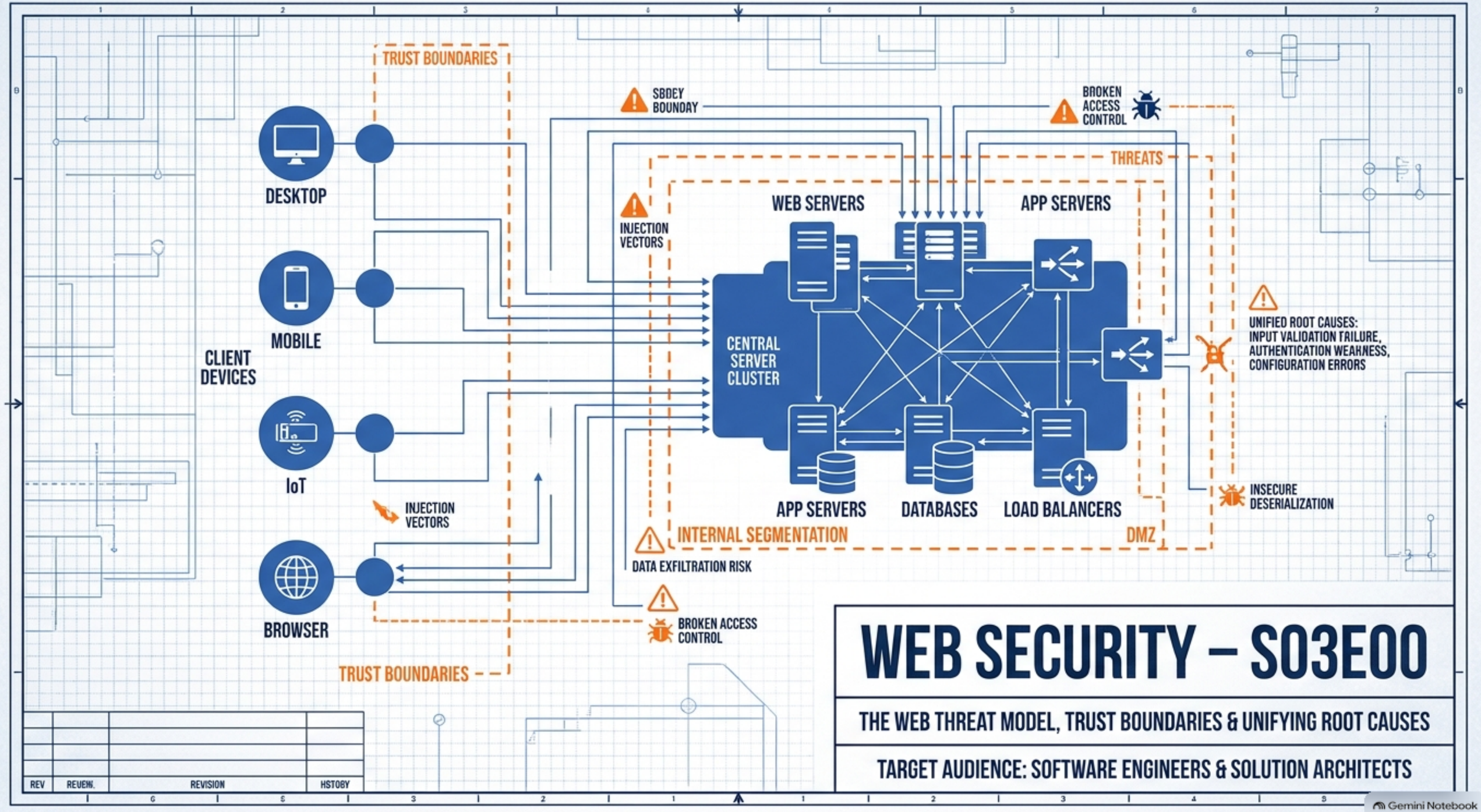


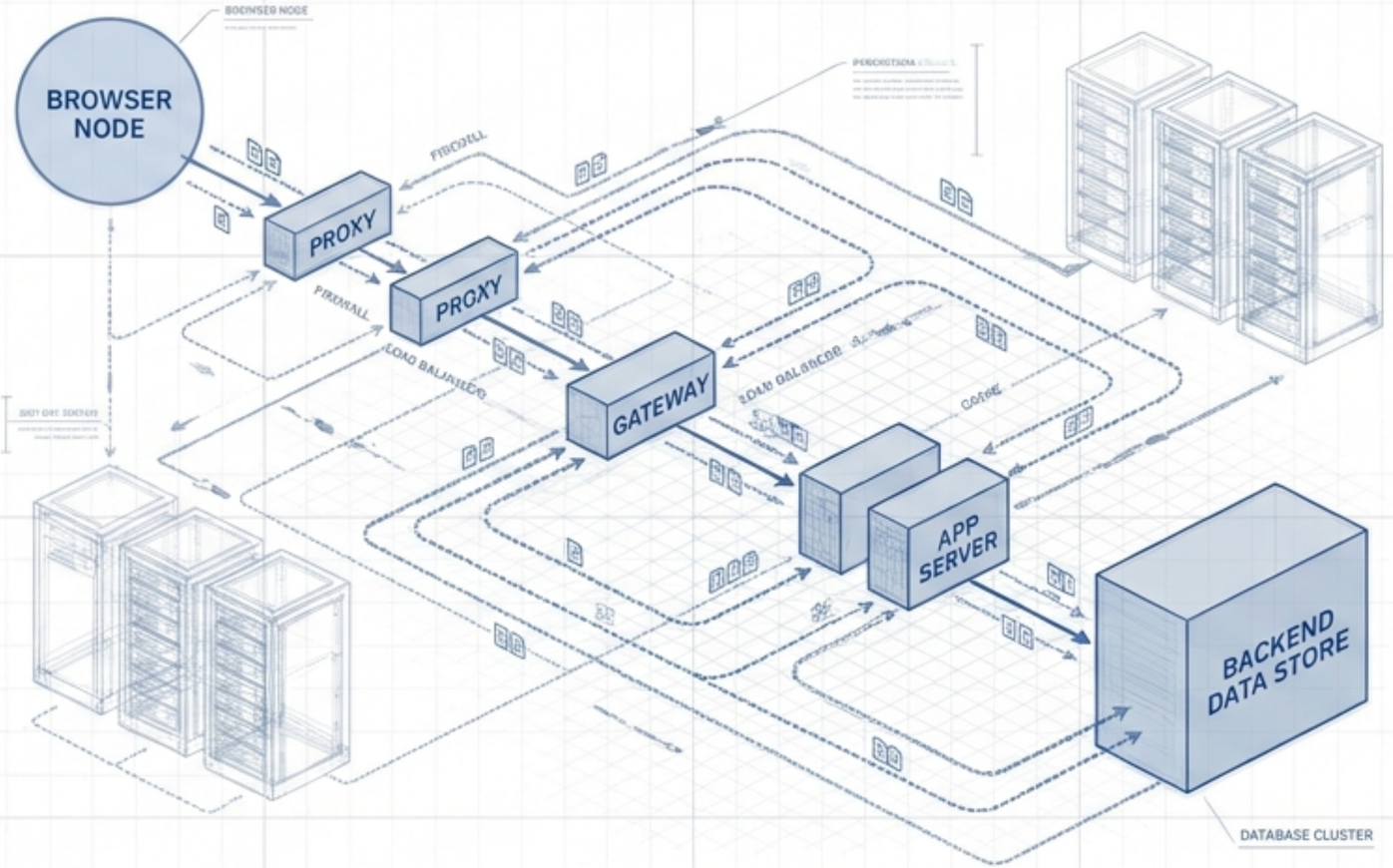
WEB SECURITY – S03E00

THE WEB THREAT MODEL, TRUST BOUNDARIES & UNIFYING ROOT CAUSES

TARGET AUDIENCE: SOFTWARE ENGINEERS & SOLUTION ARCHITECTS

Gemini Notebook

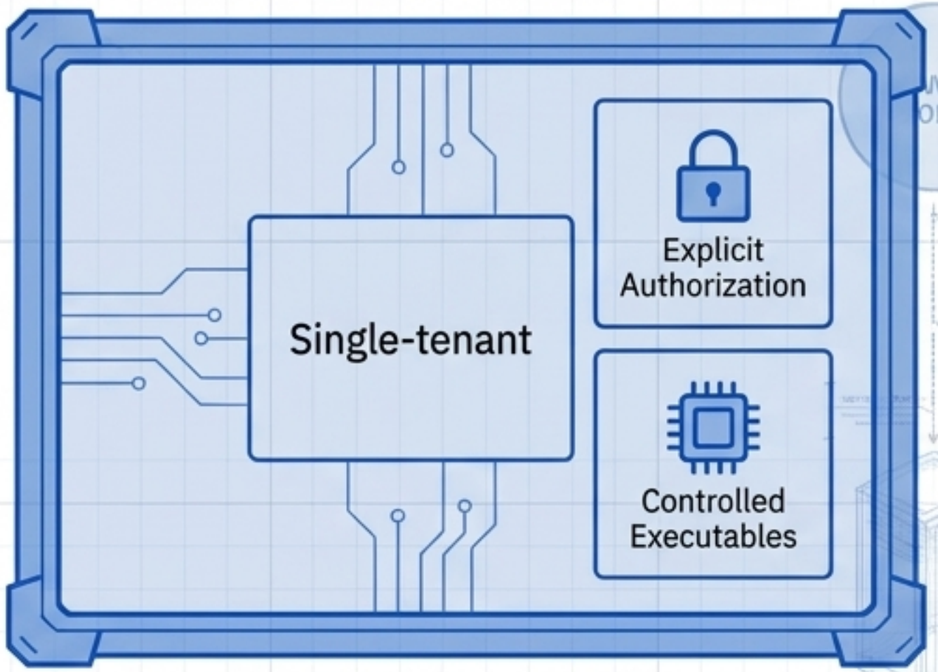
1. Why the Web Platform Is Uniquely Hostile



Gemini Notebook

A Hostile Execution Environment

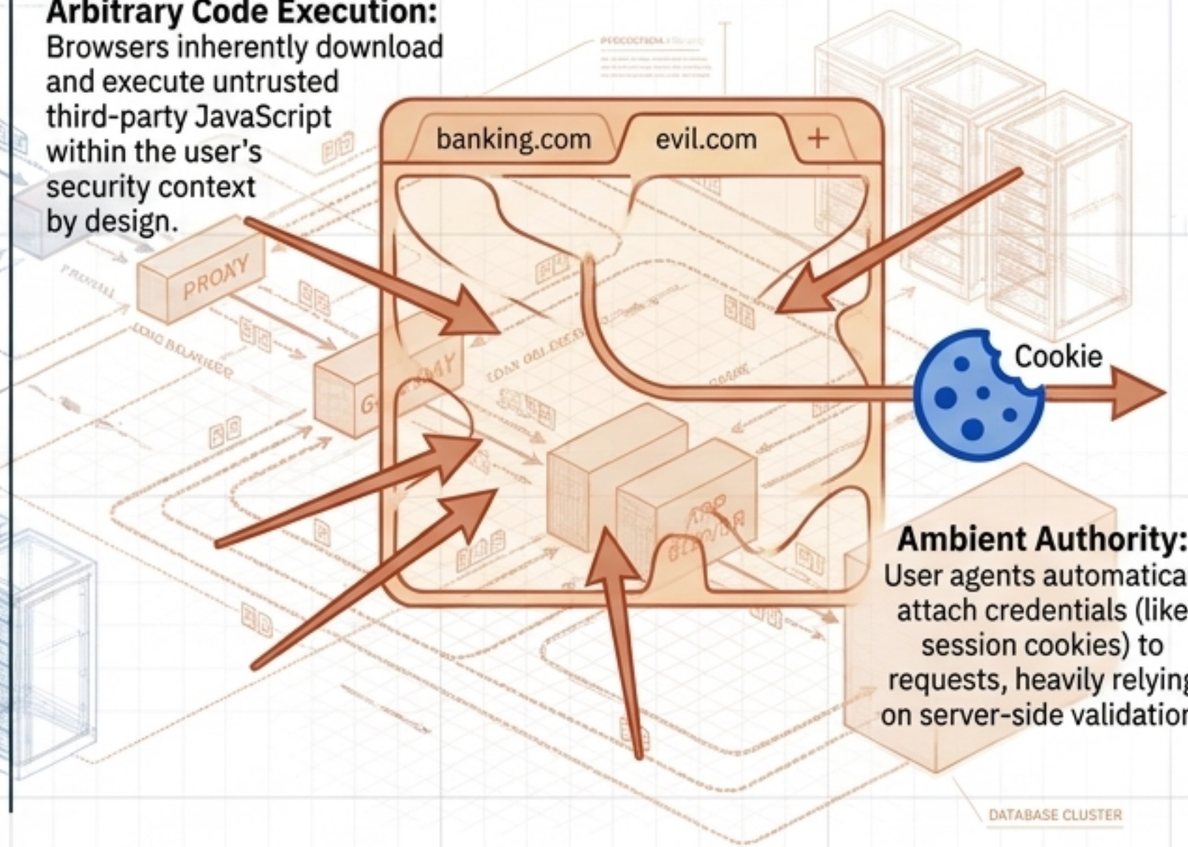
Traditional Desktop Application



Isolated execution with explicit controls and limited external interaction.

Modern Web Browser

Arbitrary Code Execution:
Browsers inherently download and execute untrusted third-party JavaScript within the user's security context by design.



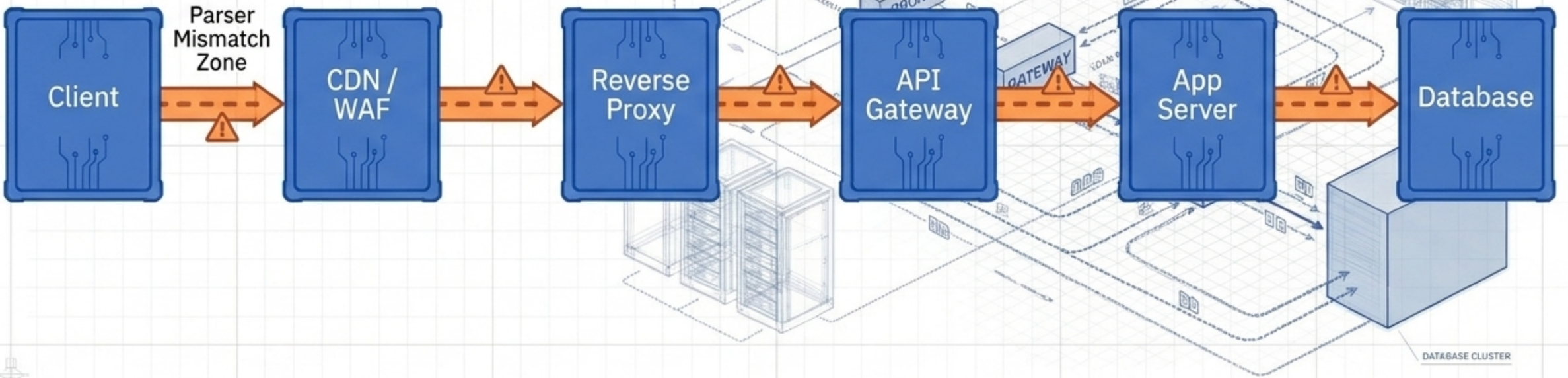
Ambient Authority:
User agents automatically attach credentials (like session cookies) to requests, heavily relying on server-side validation.

Gemini Notebook

The Multi-Component Parser Chain

The Danger: A single HTTP request passes through independent parsers, caches, and serializers before reaching the data layer.

The Reality: If any two components in this chain parse the same byte stream differently, critical vulnerabilities emerge.



Gemini Notebook

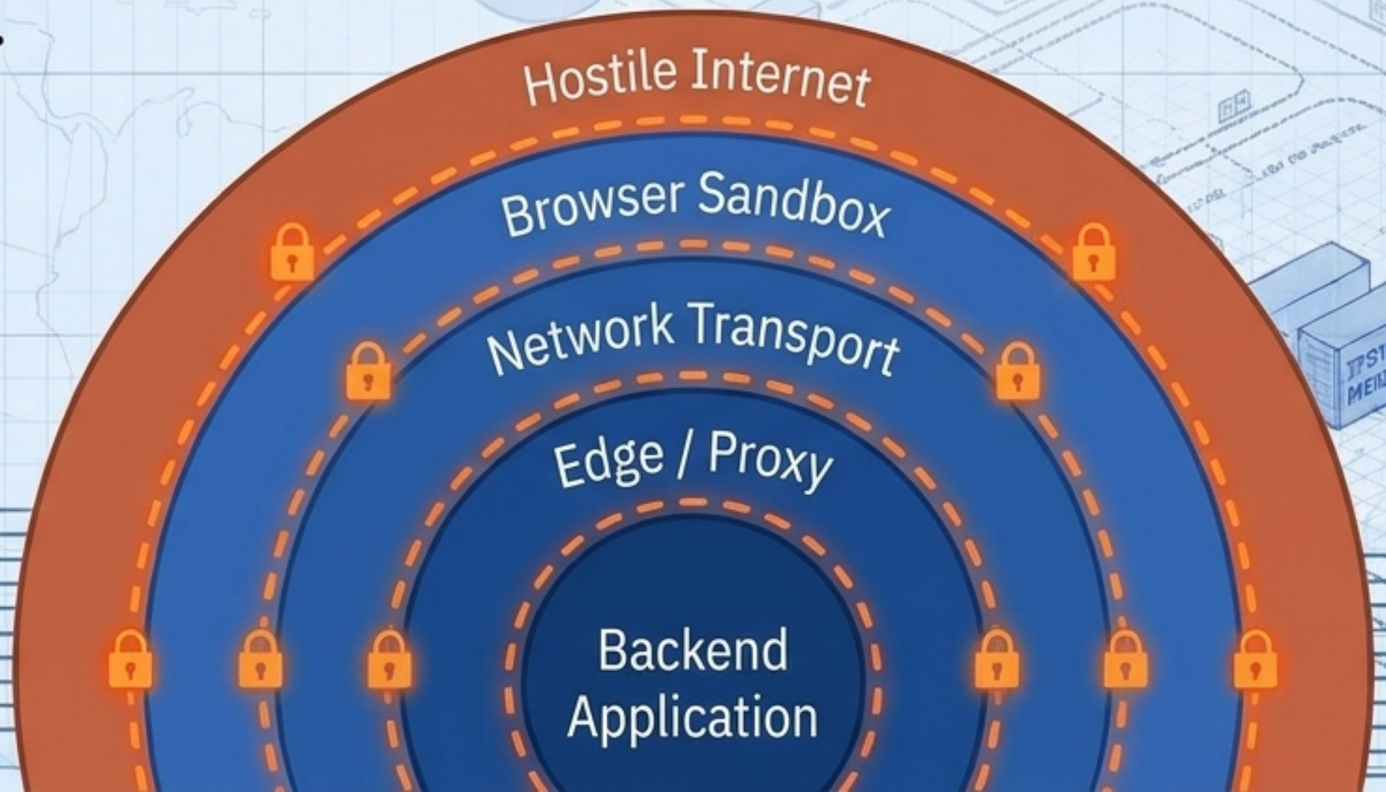


Gemini Notebook

Mapping the Attack Surface

Attack Surface: The sum of all paths for data/commands into and out of the application, plus the code protecting those paths.

Trust Boundaries: Explicit demarcation lines where data or execution context shifts from one level of trust to another. All data crossing these lines requires validation, authentication, and sanitization.



Gemini Notebook

Defining Threat Actors by Capability

Threat Actor	Network Position	Capabilities	Target Boundary
Passive Network MITM	On-Path	Reads wire traffic (Defeated by TLS)	Wire Cryptography
Active Network MITM	On-Path	Reads/Writes wire traffic (Defeated by HSTS)	Wire Cryptography
Malicious Origin	Browser Tab	Executes JS in sandbox (Restricted by SOP/CORS)	Browser Isolation
Unauthenticated Caller	External Internet	Hits public endpoints (Restricted by Edge/AuthN)	Edge / Proxy
Authenticated Tenant	Internal System	Authorized access to subset of data (Restricted by BOLA/AuthZ)	Application Logic
Compromised Dependency	Inside Backend	Code execution inside the trusted backend (Restricted by Zero-Trust)	Execution Environment

Gemini Notebook

3. Trust Boundaries 1 & 2 — Wire Cryptography & Browser Isolation

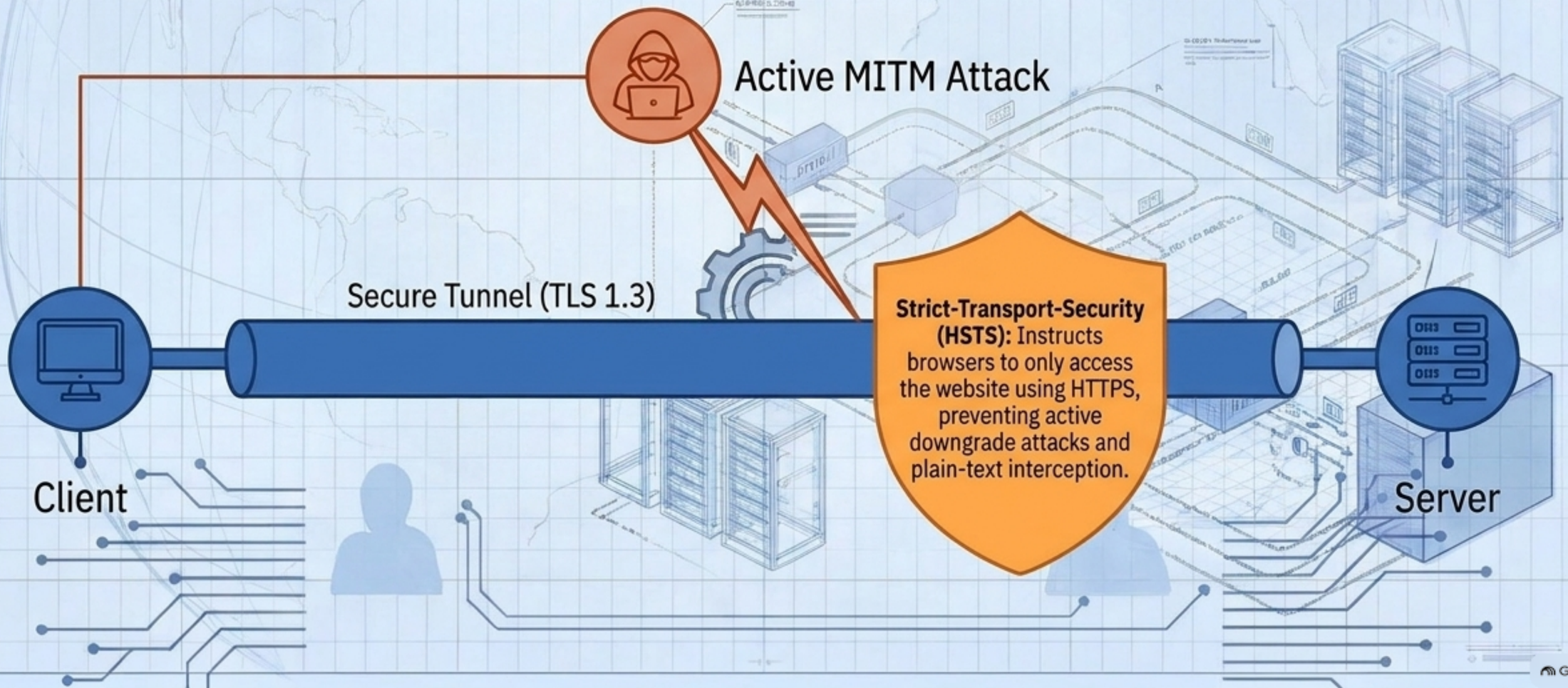
3



Gemini Notebook

Network-Layer Confidentiality & Integrity

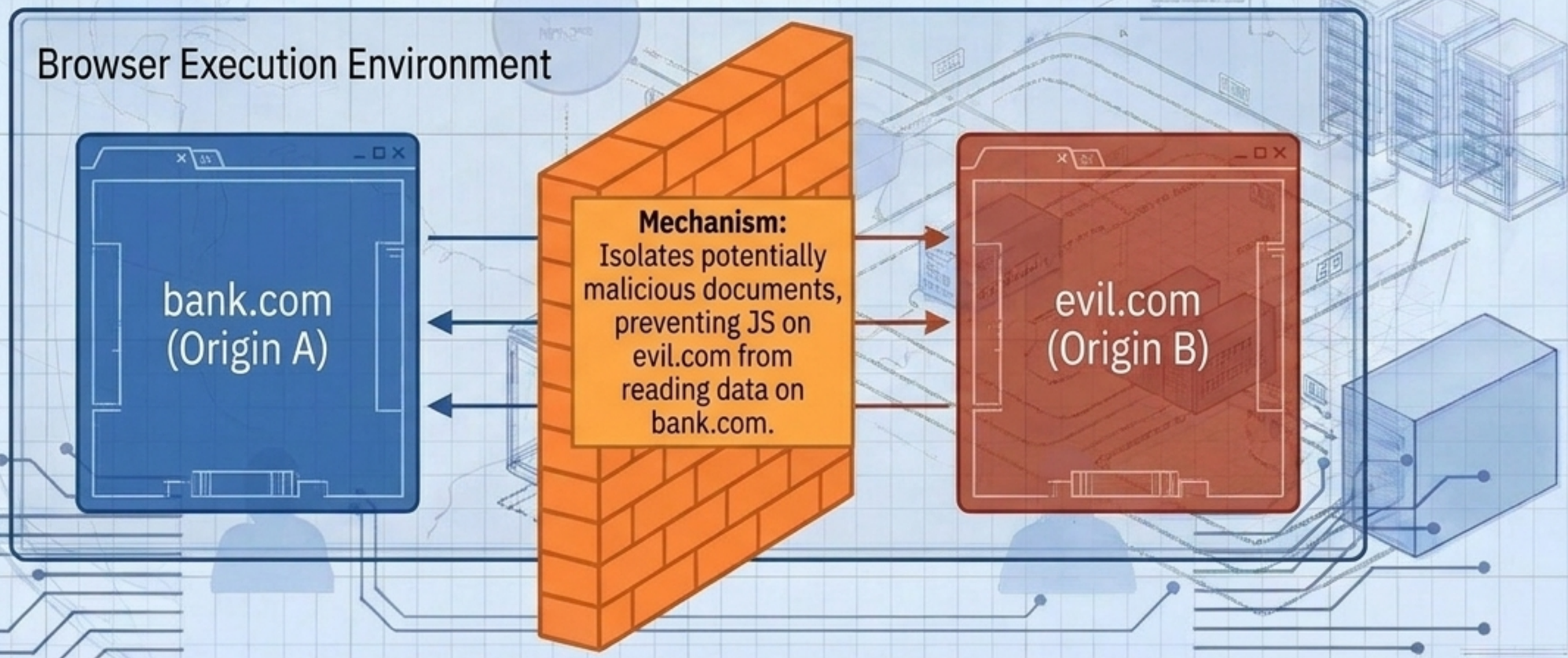
The Boundary: The physical network wire between the client and the edge.
The Defenses: TLS 1.3 (encryption in transit) and PKI (certificate validation).



Gemini Notebook

The Same-Origin Policy (SOP)

The Definition: Two URLs have the same origin if the scheme, host, and port tuple are identical.
The Reality: SOP protects the client, not the server. It prevents cross-origin reads, but typically allows cross-origin writes (forms/links) and embedding.

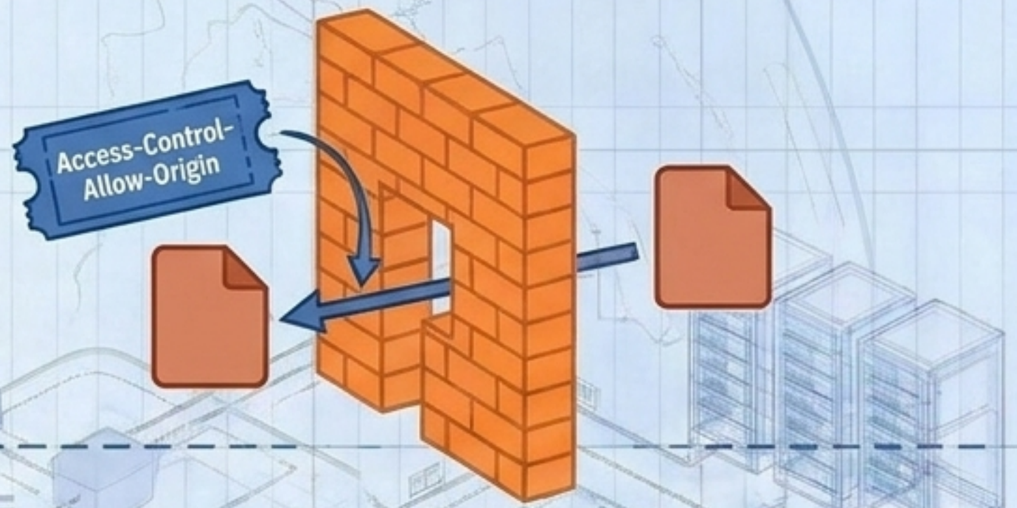


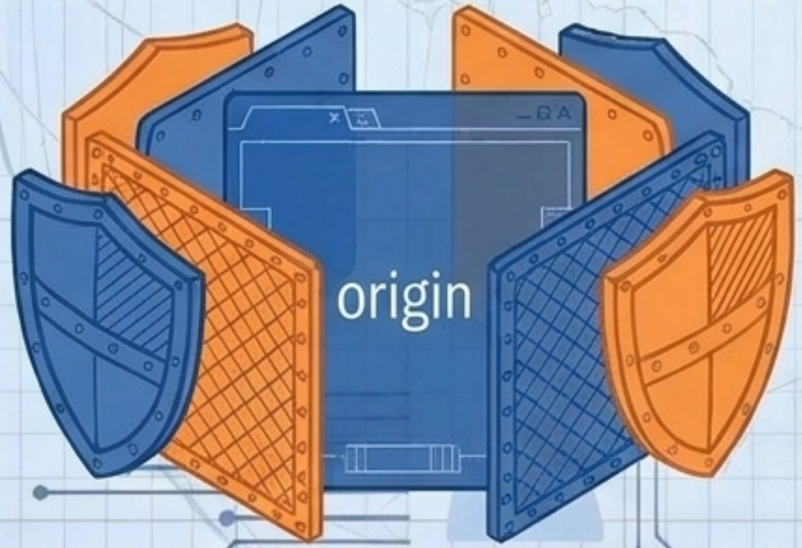
Gemini Notebook

Relaxing & Hardening the Browser Boundary

Relaxing Isolation: CORS

CORS (Cross-Origin Resource Sharing): Relaxes SOP by allowing servers to explicitly opt-in to cross-origin reads. Misconfiguration breaks isolation; it does not add security.





Hardening Isolation: Modern Security Headers

CSP (Content-Security-Policy): Disables inline JavaScript and restricts resource loading to mitigate XSS.

COOP / COEP / CORP: Cross-Origin Opener, Embedder, and Resource Policies. Isolates the browsing context to same-origin documents to prevent Spectre-style leaks.

Gemini Notebook

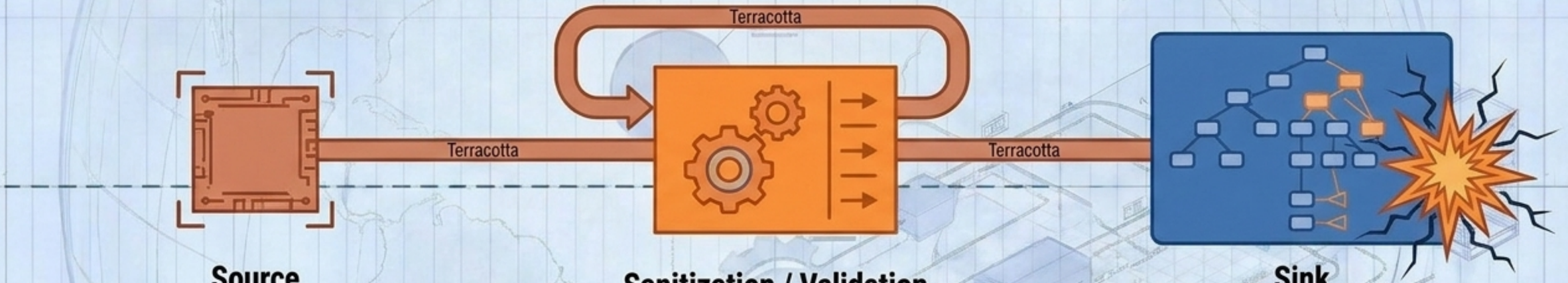
4. Trust Boundaries 3 & 4 — Client DOM Execution & Identity State



Gemini Notebook

Client-Side Code Injection

Taint-Flow Vulnerabilities: Untrusted data is treated as executable code by the browser.



XSS (Cross-Site Scripting)

Reflected, Stored, and DOM-based injection into HTML or JS contexts.

DOM Clobbering

Overwriting global variables by injecting HTML id attributes.

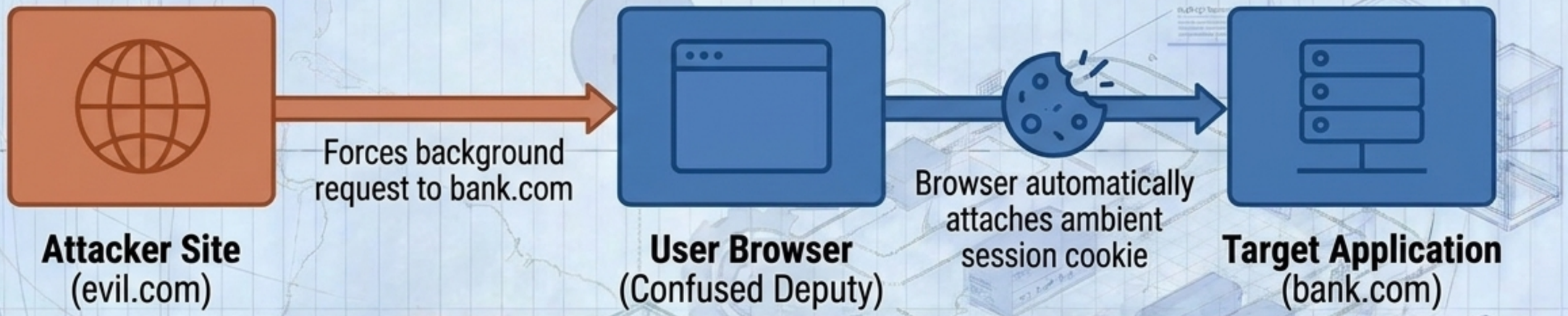
Prototype Pollution

Modifying JavaScript object prototypes (e.g., via __proto__) to maliciously inject inherited properties across the entire application.

Gemini Notebook

Exploiting Ambient Authority

The Flaw: Browsers automatically attach ambient credentials (session cookies) to requests, regardless of the request's origin.



CSRF (Cross-Site Request Forgery)

Forcing an authenticated user to execute unwanted actions. Defended by SameSite cookie attributes (Strict/Lax) and anti-CSRF tokens.

Clickjacking (UI Redressing)

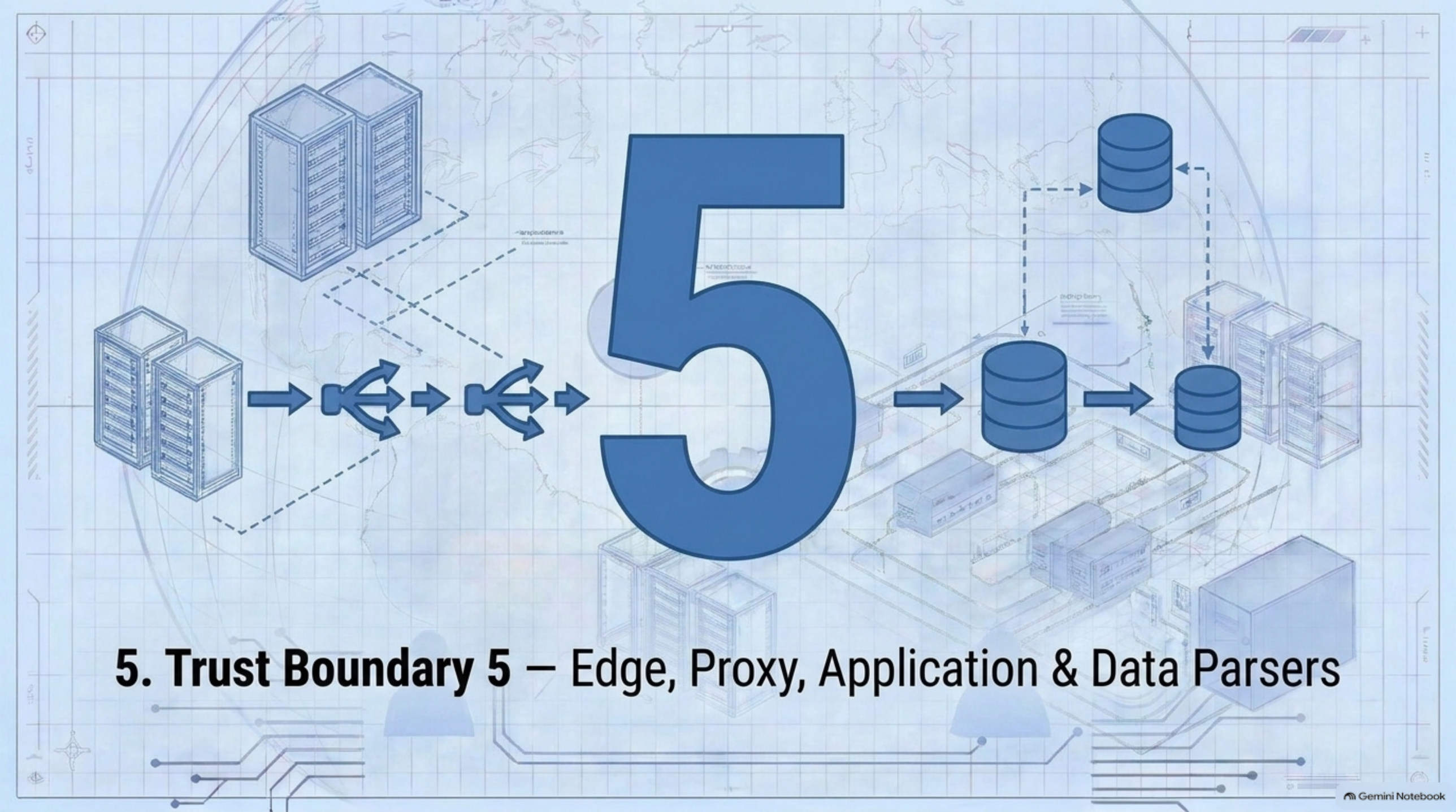
Tricking users into clicking hidden actionable frames. Defended via X-Frame-Options or CSP.

Gemini Notebook

Managing Identity State

Mechanism	Protocol	Security Trade-offs	Core Defense
Session Cookies	Stateful	Relies on ambient authority. Prone to CSRF.	SameSite and HttpOnly flags
OAuth 2.0 / OIDC	Delegated Authorization	Prone to redirect manipulation and implicit grant token leaks.	Strict redirect URI validation, PKCE
JWT (JSON Web Tokens)	Stateless	Cryptographically signed, but difficult to revoke immediately compared to server-side sessions.	Short expirations, strict algorithm verification
WebAuthn (Passkeys)	Hardware-Backed	Replaces shared secrets. Requires hardware token or biometric.	Phishing-resistant cryptography

Gemini Notebook

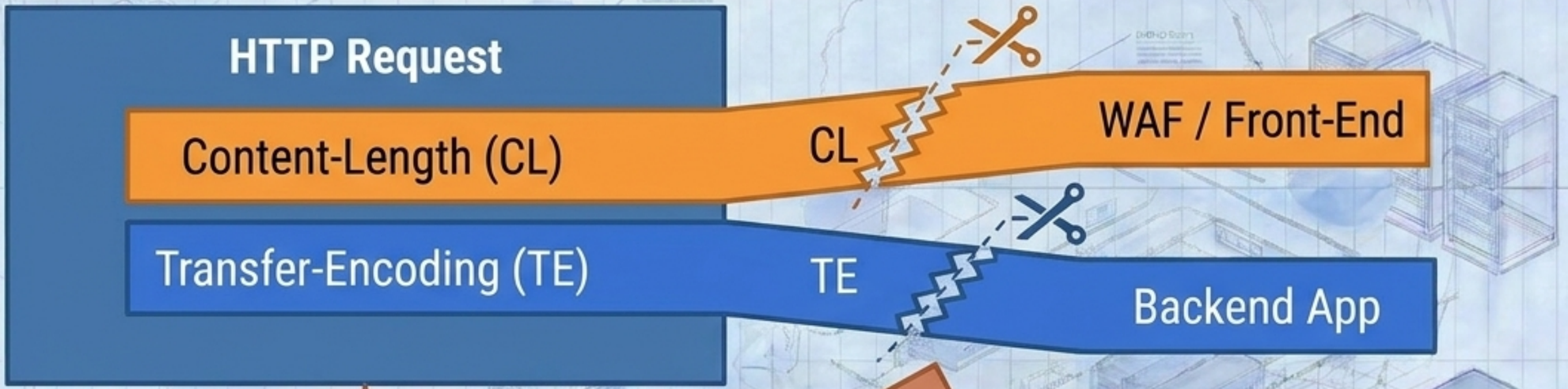


5. Trust Boundary 5 — Edge, Proxy, Application & Data Parsers

Gemini Notebook

HTTP Request Smuggling & Cache Poisoning

The Root Cause: Parser desynchronization (CL.TE or TE.CL). The front-end and back-end servers disagree on where an HTTP message ends.



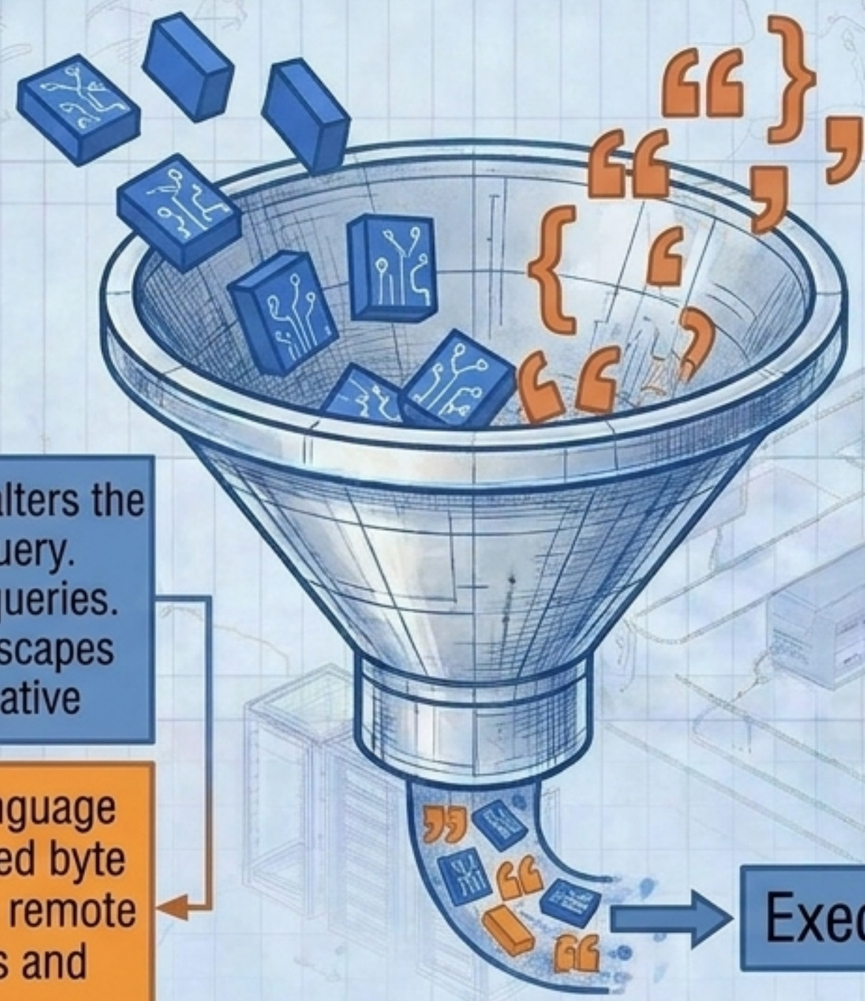
Smuggled Payload: Attackers smuggle hidden requests, bypass edge security controls, capture other users' responses, and poison web caches.

Gemini Notebook

In-Band Data-as-Code Mixing

Untrusted User Data

Control Characters



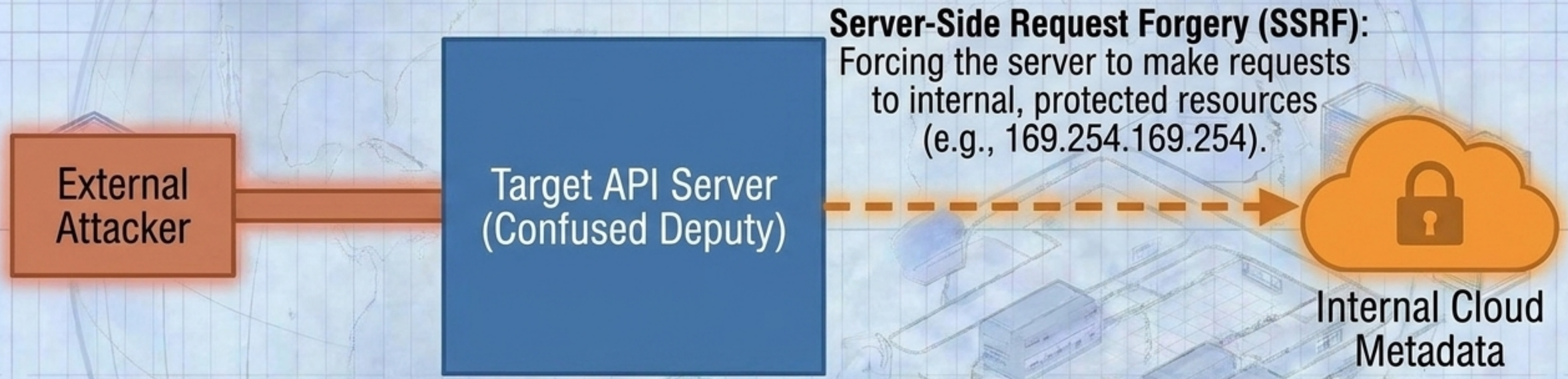
SQL Injection (SQLi): Untrusted data alters the semantic structure of a database query. Prevented entirely by parameterized queries.
OS Command Injection: User input escapes parameterization and executes as native

Insecure Deserialization: Native language parsers (Java, PHP) convert untrusted byte streams back into objects, triggering remote code execution via magic methods and gadget chains.

Executable Query/Command

Gemini Notebook

SSRF, BOLA & Concurrency

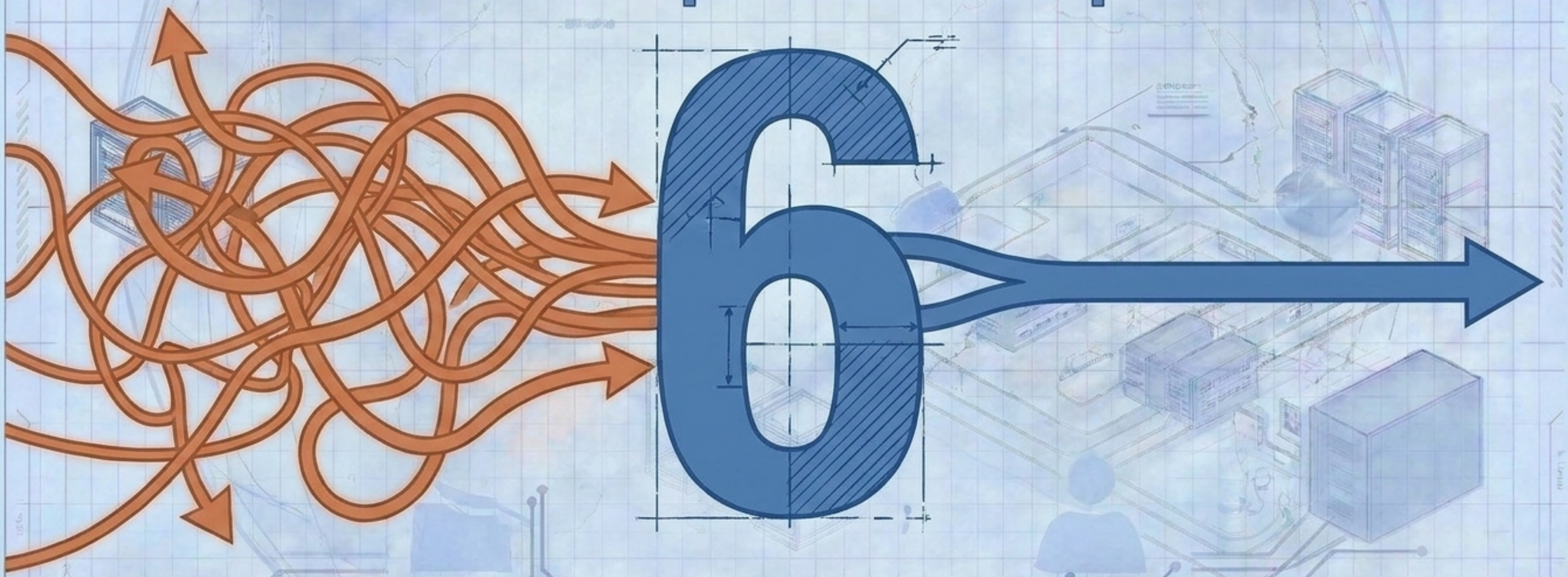


Broken Object Level Authorization (BOLA/IDOR): Endpoints failing to validate if the authenticated user has explicit rights to access the requested object ID.

Race Conditions: Time-of-check to time-of-use (TOCTOU) flaws causing limit overruns in multi-step or concurrent transactions.

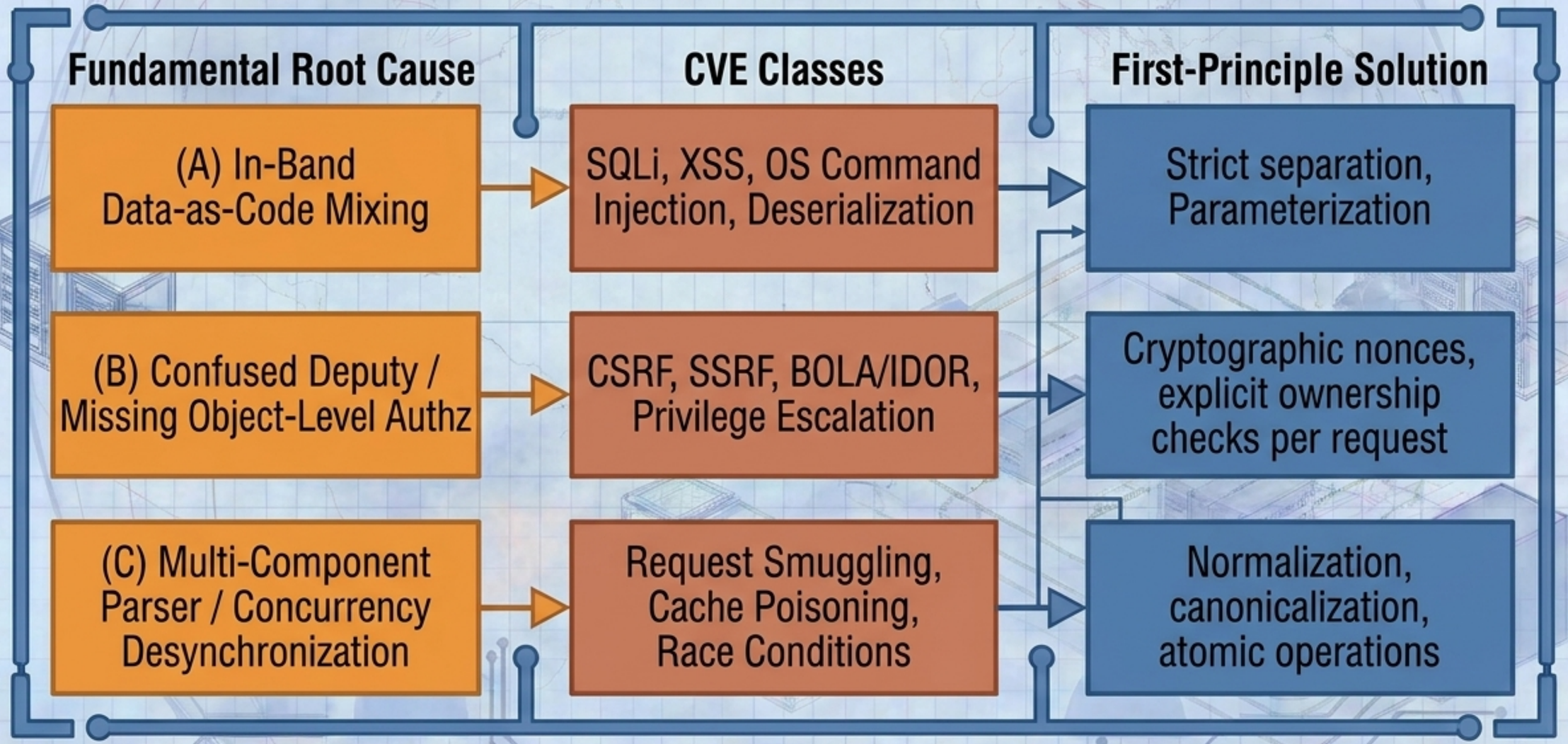
Gemini Notebook

6. Three Root Causes Unifying OWASP Top 10 & API Top 10



Gemini Notebook

The First-Principles Taxonomy



Gemini Notebook

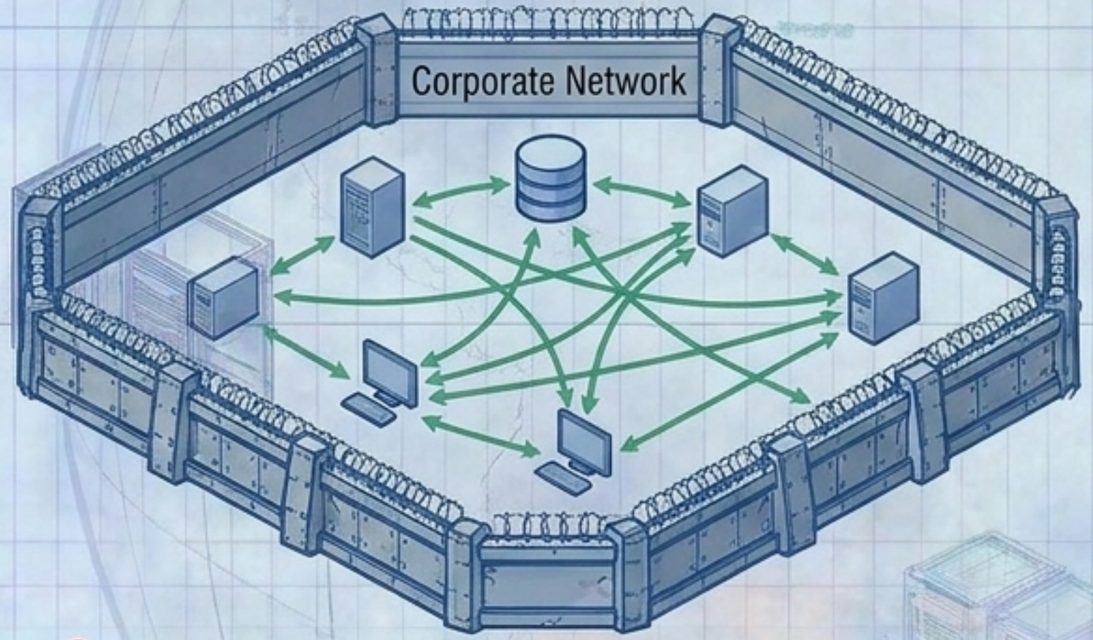
7. Zero-Trust Defense-in-Depth & Series Roadmap



Gemini Notebook

Assuming Perimeter Breach (NIST SP 800-207)

Traditional Perimeter Defense



Implicit trust based on network location. Once breached, attackers move laterally without friction.

Zero-Trust Architecture



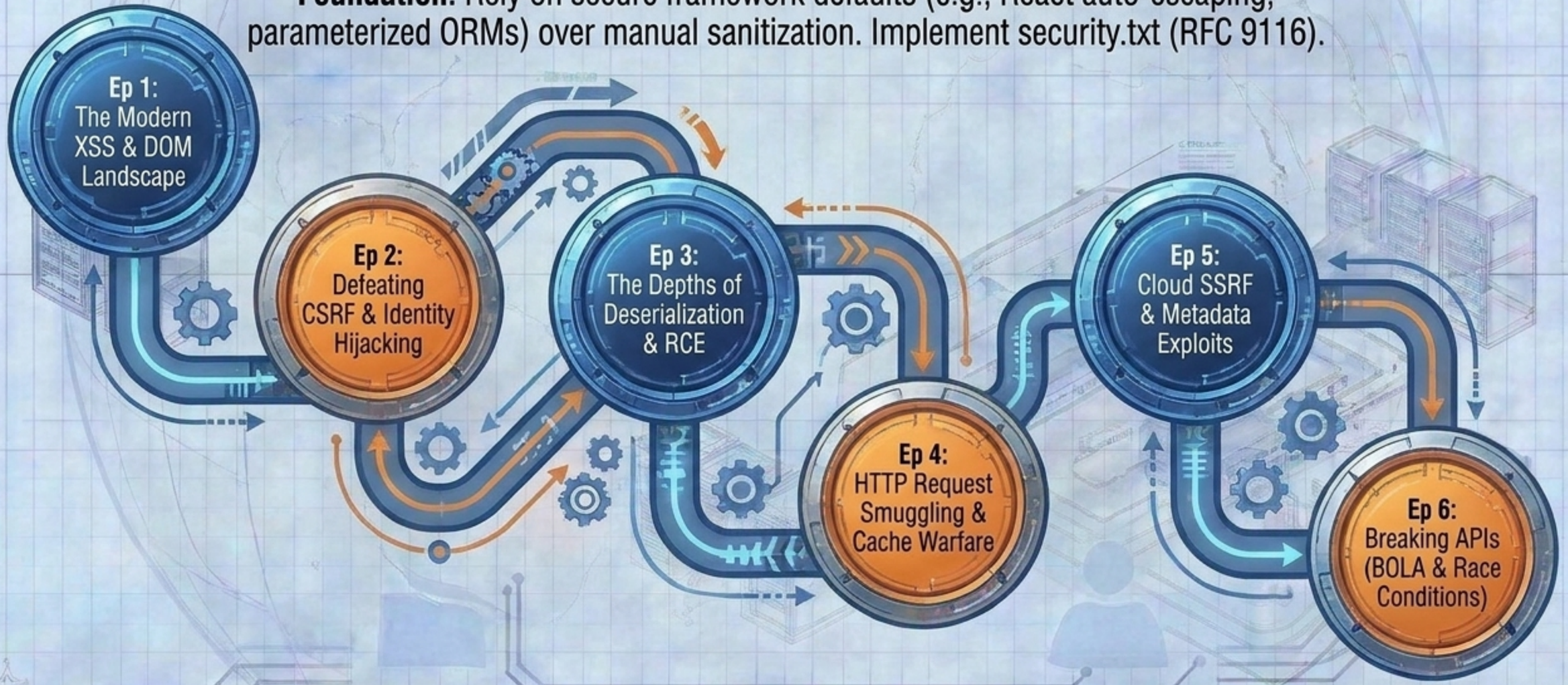
No implicit trust. Authentication and authorization (subject and device) are evaluated strictly before a session is established per-request.

Defense-in-Depth: Protecting resources directly via least-privilege workload identities and centralized control-plane policy decisions.

Gemini Notebook

The Path Forward

Foundation: Rely on secure framework defaults (e.g., React auto-escaping, parameterized ORMs) over manual sanitization. Implement security.txt (RFC 9116).



Gemini Notebook