The HTTP Archive (HAR) File

A Guide to Network Diagnostics, Generation Protocols, and Data Security.



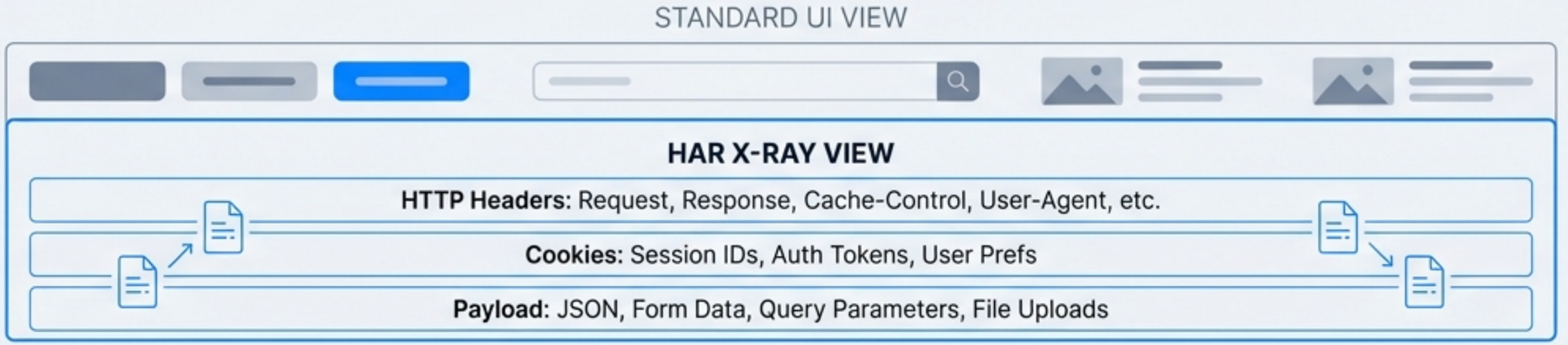
Internal Technical Briefing | Network Operations & Security

Visibility into the Browser-Server Exchange

Definition

A **HAR file** is a JSON-formatted log that records a web browser's interaction with a web server. It acts as a "black box" recorder for web traffic.







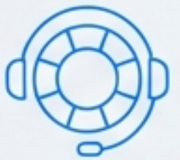
Performance Engineering

Identifying bottlenecks, slow DNS lookups, and large resource loads.



Debugging & Dev

Analyzing authentication flows, redirect chains, and HTTP **404**/ **500** errors.



Support Acceleration

A definitive record for vendors to reproduce complex issues without prolonged back-and-forth.

Anatomy of a Log Entry

The Breakdown

The Request (Client)

- Method (GET/POST)
- URL & Parameters
- Headers (User-Agent)
- Cookies

The Response (Server)

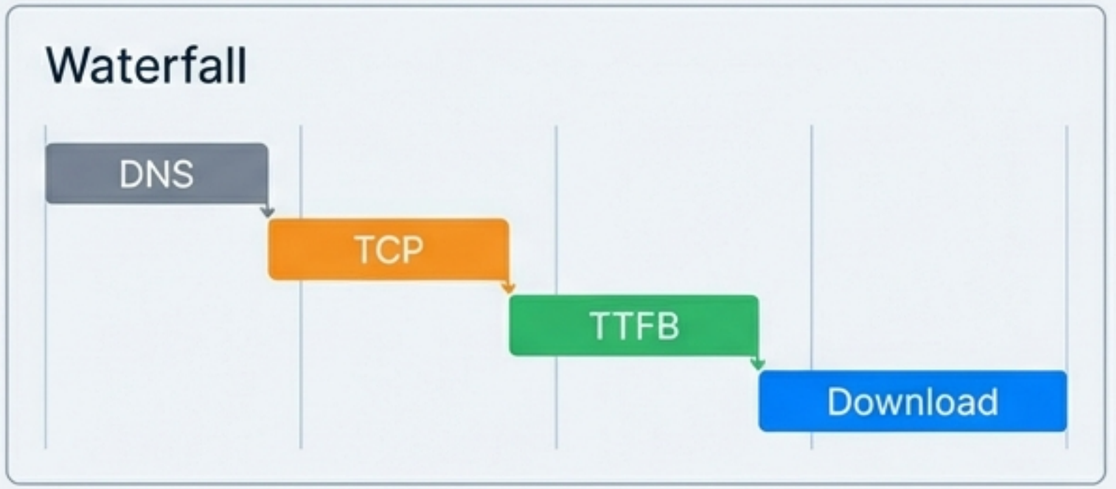
- Status Code (200, 403)
- MIME Type
- Content Size
- Payload

The Timing

- DNS Lookup
- TCP Handshake
- TTFB
- Download

The Visuals

```
{
  'method': 'POST',
  'url': 'https://api.auth0.com',
  'headers': [ ... ],
  'status': 200
}
```



Protocol A: Generation via Chrome & Edge

1. Initiate
Open Developer Tools (F12 or Ctrl+Shift+I).

2. Target
Navigate to the **Network** tab.

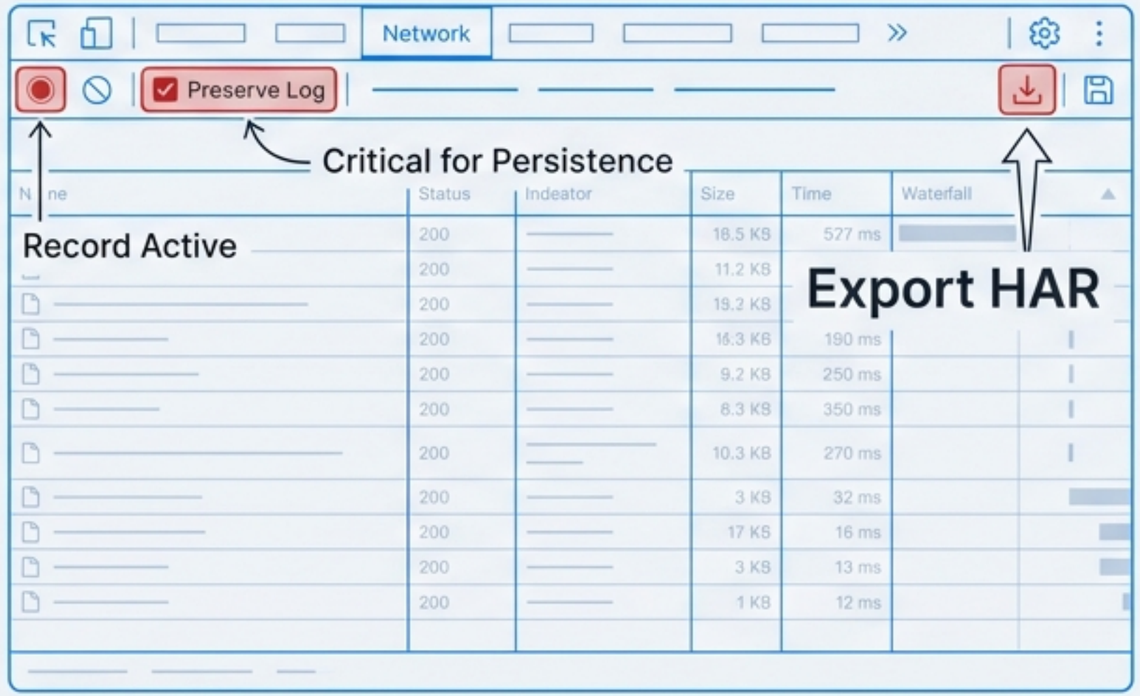
3. Critical Configuration
Check the **'Preserve Log'** box.

⚠ Note: Essential for tracking issues across page reloads.

4. Record
Ensure the record button is active (Red).

5. Reproduce
Perform the action that triggers the error.

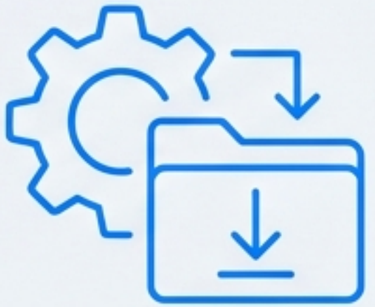
6. Export
Right-click grid -> 'Save all as HAR with content'.



CHROME / EDGE DEVTOOLS

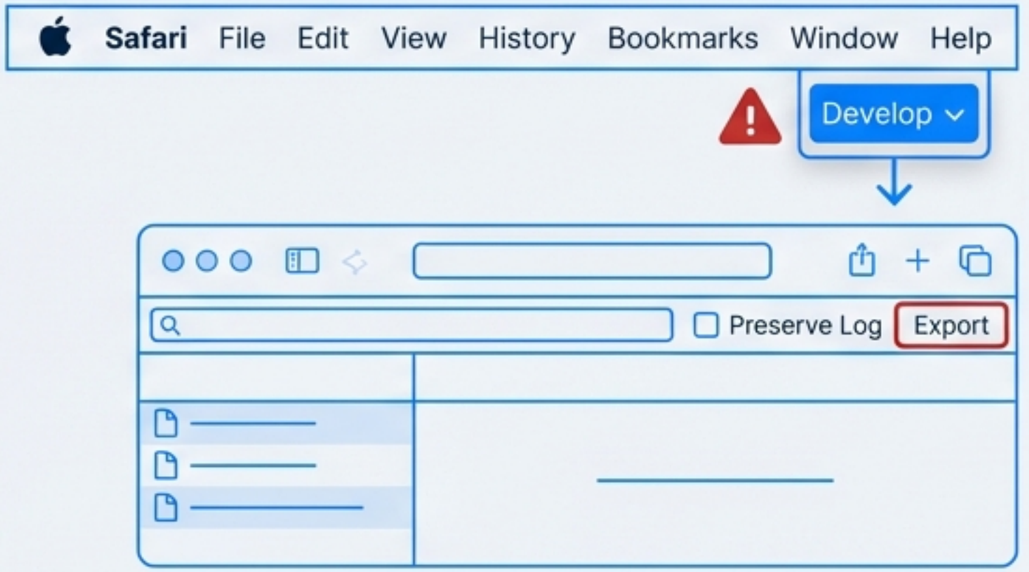
Protocol B: Generation via Firefox & Safari

Firefox



- **Tools > Developer > Network** (Ctrl+Shift+E).
- Enable **'Persist Logs'** to capture history.
- Right-click **File** column > **'Save All As HAR'**.

Safari (The Pre-requisite)



- Step 0: “Enable developer menu: **Preferences > Advanced > Check 'Show Develop menu in menu bar'**.
- Step 1: **Develop > Show Web Inspector > Network**.
- Step 2: Select **'Preserve Log'** in filter options.
- Step 3: Click **Export**.

Best Practices for Quality Capture



The Clean Slate

Use **Incognito/Private** windows. Eliminate interference from existing cache or third-party cookies.



The Preserve Mandate

Always enable **'Preserve Log'** . Ensure the file captures the full sequence of redirects during authentication.



Clear Cache

If Incognito isn't possible, manually right-click and select **'Clear browser cache'** before recording.

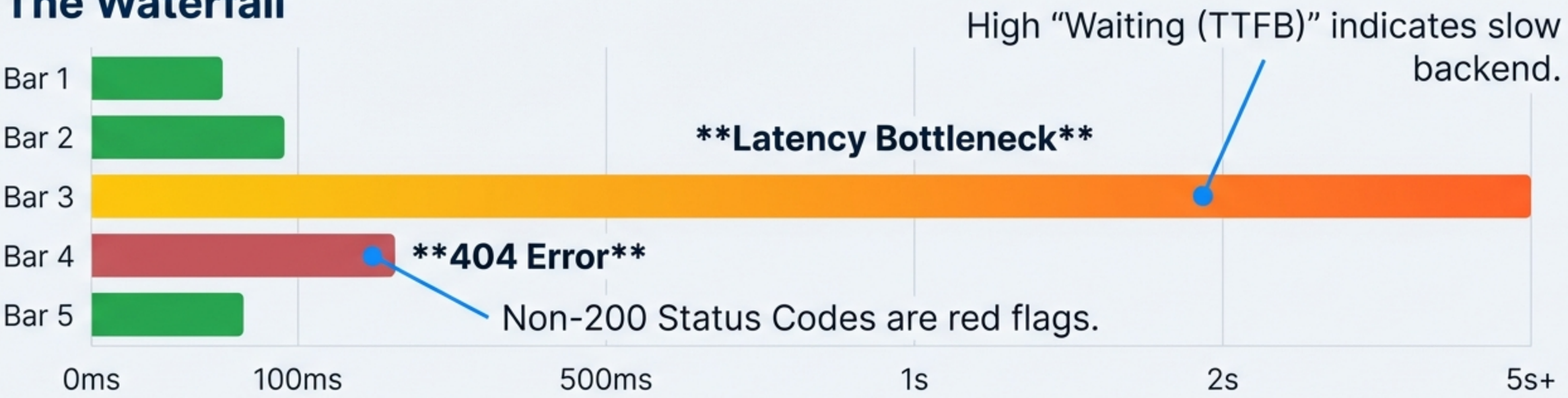


Reproducibility

Start recording *before* the issue occurs. Stop immediately *after*. Minimize file size and noise.

Analyzing the Dataset

The Waterfall



Identifying Errors

401/403: Authentication Failures
404: Missing Resources
500: Server Crashes

Recommended Tooling

Chrome DevTools
Google HAR Analyzer
Jam

Security Governance & Risk



The Warning: Full Capture

A HAR file captures **everything** entered or sent.

- Passwords in plain text.
- API Keys & Session Tokens.
- PII (Personal Identifiable Information).

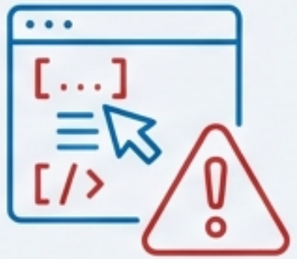
The ‘Data at Rest’ Threat

Uploading raw HAR files to support tickets (Zendesk, Jira) creates permanent security silos containing valid credentials.

Case Reference: Risks highlighted by the Okta breach discussions.

Sanitization & Safe Sharing Protocols

Manual Method (High Risk)



Editing JSON in a text editor.

- Prone to human error.
- Incomplete scrubbing.
- Time-consuming.

Automated Method (Recommended)



Dedicated Redaction Tools.

- Chrome DevTools: “Save all as HAR (sanitized)” (Removes Auth headers).
- External Tools: Nightfall AI, Jam (Advanced redaction).

Policy Mandate: Never share a raw HAR file via email or chat without redaction.

Summary of Operations



Diagnostic Power

The definitive record of browser-server interaction. Essential for resolving latency and auth issues.



Operational Discipline

Quality data requires precise generation: **Incognito Mode** and **Preserve Log** are mandatory.



Security First

HAR files are sensitive credentials. **Sanitize** before sharing. Do not store indefinitely in tickets.