Websocket Endpoint Analysis Report

Insecure WebSocket Implementations: Crawling Public Sites, Testing Endpoints for Vulnerabilities, and Reporting Impact Analysis

Report Generated on : July 10, 2025

PESU Center for
Information Security,
Forensics and Cyber Resilience

PESU RESEARCH
11TH FLOOR, BE BLOCK,
PES UNIVERSITY, RING ROAD CAMPUS,
BANGALORE, INDIA

WebSocket Security Scan Report

Executive Summary

Real-time apps increasingly rely on WebSocket connections, but insecure implementations—such as missing origin checks or weak authentication—can allow hijacking or sensitive data exposure.

To address this, we developed an automated scanner that crawls public web applications, detects vulnerable WebSocket endpoints, and analyzes their real-world impact.

- Crawl and detect active WebSocket endpoints from public websites.
- Apply origin-header enforcement and protocol fuzzing tests to assess security gaps.
- Generate structured PDF reports summarizing detected vulnerabilities and severity.

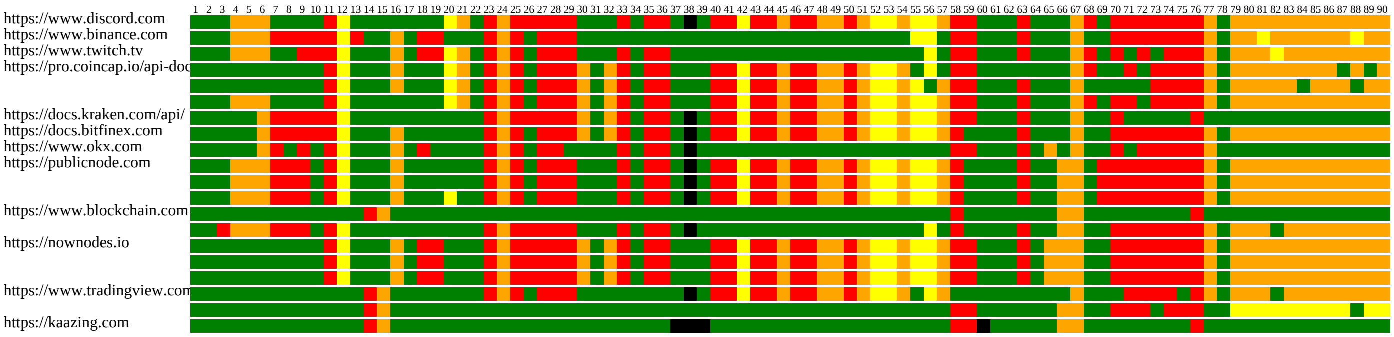
Scan Start Time:	2025-07-10 18:20:26
Scan End Time:	2025-07-10 19:35:12
Total Scan Duration:	4491.13 seconds
Total URLs Scanned:	15
High Severity Vulnerabilities:	461
Medium Severity Vulnerabilities:	379
Low Severity Vulnerabilities:	102

All Scanned Websites

This section lists all scanned websites and summarizes the overall vulnerability distribution by severity. The bar graph below visualizes the number of High, Medium, and Low severity vulnerabilities identified across all scanned sites.

#	Website
1	https://www.cryptocompare.com
2	https://www.tradingview.com
3	https://www.twitch.tv
4	https://www.discord.com
5	https://www.binance.com
6	https://coinsdo.com
7	https://pro.coincap.io/api-docs
8	https://docs.kraken.com/api/
9	https://docs.bitfinex.com
10	https://www.bitstamp.net
11	https://www.blockchain.com
12	https://www.okx.com
13	https://nownodes.io
14	https://publicnode.com
15	https://kaazing.com

WebSocket vs. Attack Heatmap



Vulnerability Summary by Type

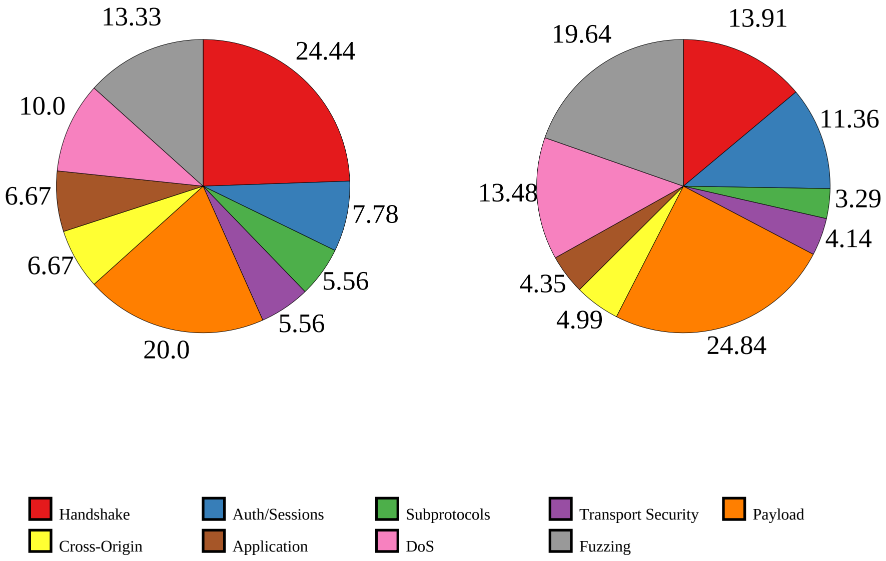
This section summarizes key categories of vulnerabilities found during the scan. It groups issues like missing origin checks, weak authentication, insecure handshakes, and over 80 other attack for test to highlight common WebSocket flaws.

The bar chart below visualizes how many vulnerabilities were found in each category. This helps quickly identify the most common and critical problem areas across scanned applications.

Type	Count
Handshake & Upgrade Validation	131
Authentication & Session Control	107
Subprotocols & Extension Handling	31
Transport Security & Encryption	39

Payload Framing & Messaging Semantics	234
Origin Policy & Cross-Origin Enforcement	47
Application-Layer Logic & Misconfigurations	41
DoS, Compression & Resource Limits	127
Protocol Fuzzing	185

Test Distribution vs Results



Detailed Scan Results

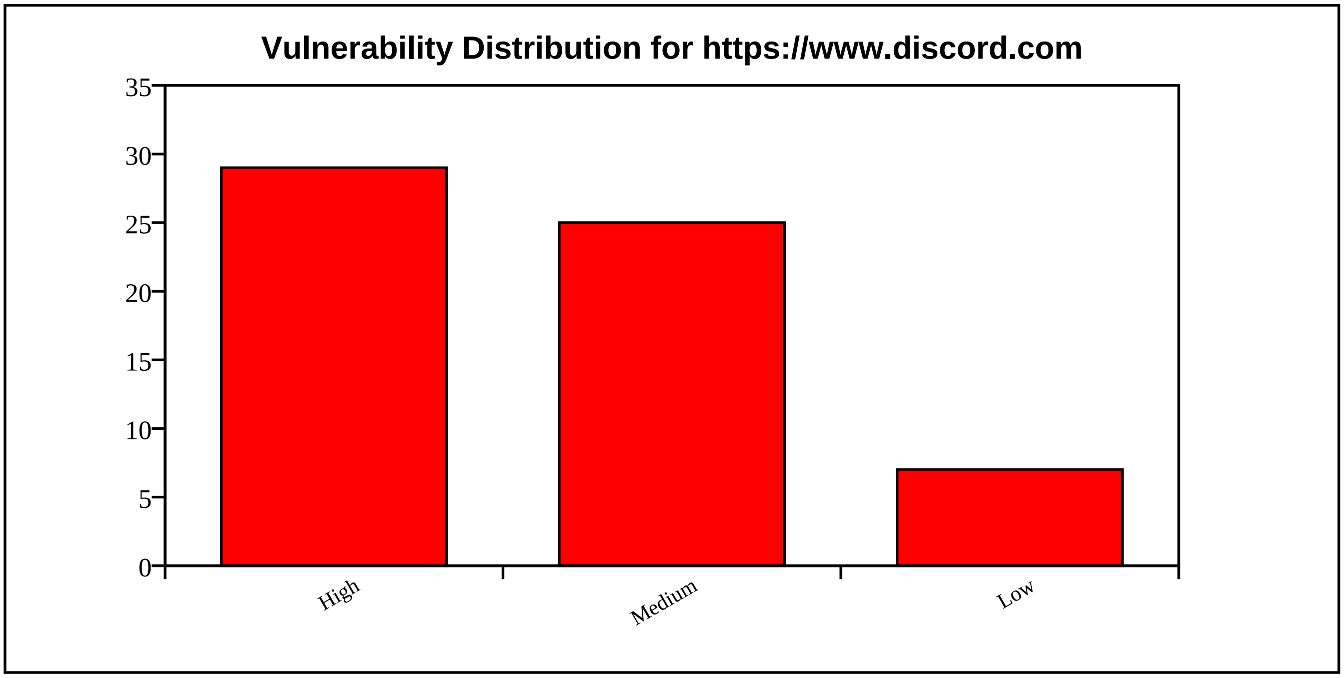
This section provides an in-depth breakdown of each scanned target. For every URL, it lists the scan duration, number of URLs crawled during reconnaissance, and the WebSocket endpoints discovered. It helps identify how many potential communication channels were exposed for testing. Each target's vulnerability distribution is summarized by severity (High, Medium, Low) using a bar chart, followed by a detailed list of detected vulnerabilities. The section also documents the types of attacks performed and the exact WebSocket endpoints and internal URLs involved in the scan. This allows for a thorough understanding of the security posture and exposure of each target.

Target URL: https://www.discord.com

Scan Duration:	266.46 seconds
URLs Crawled:	1
WebSocket Endpoints Found:	0
Attack Performed:	True
High Severity Findings:	29
Medium Severity Findings:	25
Low Severity Findings:	7

WebSocket Endpoints:

#	URL
1	wss://gateway.discord.gg



Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://gateway.discord.gg

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at gateway.discord.gg:443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at gateway.discord.gg:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key

Risk Level:	Medium
Description:	Server at gateway.discord.gg:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at gateway.discord.gg:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at gateway.discord.gg:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Long URL Path

Risk Level:	Low
Description:	Server at gateway.discord.gg:443 accepted handshake with long URL path (2KB).
Solution:	Limit URL path length to prevent resource exhaustion.

Name:	Unicode URL
Risk Level:	Medium
Description:	Server at gateway.discord.gg:443 accepted handshake with Unicode URL.
Solution:	Sanitize and validate URL paths to handle Unicode correctly.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie

Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	HTTP Session Reuse
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg reused HTTP session cookie without revalidation.
Solution:	Require revalidation or token-based auth for WebSockets even if HTTP session exists.

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg allows unauthenticated connections and responds with data.

Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Fake Extension
Risk Level:	High
Description:	Server at gateway.discord.gg:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at gateway.discord.gg:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at gateway.discord.gg:443 accepted HTTP/1.0 WebSocket handshake.

Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted frame with reserved opcode 0xB.

Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://gateway.discord.gg accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted forged extended payload length (0x8000000000000001).

Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted an unmasked client frame.

Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted a text frame with invalid UTF-8 bytes.

Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://gateway.discord.gg accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://gateway.discord.gg accepted UTF-8 text sent in a binary frame.

Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://gateway.discord.gg accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://gateway.discord.gg handled abrupt TCP closure and allowed clean reconnection.

Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://gateway.discord.gg (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://gateway.discord.gg allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).

Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://gateway.discord.gg (HTTP equivalent) lacks the following headers: Content-Security-Policy, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	URL Path Traversal
Risk Level:	High

Description:	WebSocket endpoint allows path traversal via: wss://gateway.discord.gg/ws/../admin/socket
Solution:	Validate and normalize paths to prevent traversal.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg allowed 100 concurrent connections in 1.33s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High

Description:	WebSocket at wss://gateway.discord.gg allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High

Description:	WebSocket at wss://gateway.discord.gg accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://gateway.discord.gg accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1

Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4

Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7

Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10

Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

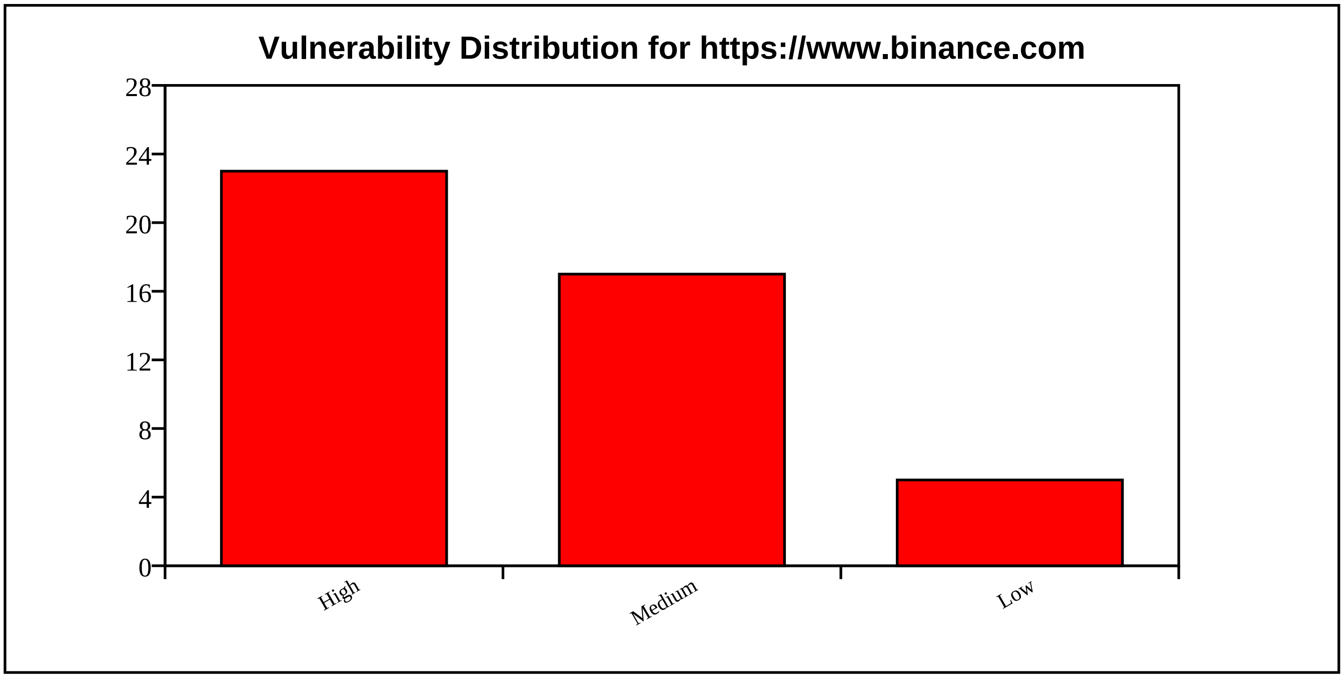
Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://gateway.discord.gg responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Target URL: https://www.binance.com

Scan Duration:	290.38 seconds
URLs Crawled:	1
WebSocket Endpoints Found:	0
Attack Performed:	True
High Severity Findings:	23
Medium Severity Findings:	17
Low Severity Findings:	5

WebSocket Endpoints:

#	URL
1	wss://stream.binance.com:9443/ws


Vulnerability Distribution for https://www.binance.com
28
24
20
16
12
8
4
0
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://stream.binance.com:9443/ws

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at stream.binance.com:9443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at stream.binance.com:9443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key

Risk Level:	Medium
Description:	Server at stream.binance.com:9443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version
Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version

Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Wrong Upgrade Header
Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted handshake with wrong Upgrade header.
Solution:	Enforce strict Upgrade header validation.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers

Risk Level:	Low
Description:	Server at stream.binance.com:9443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Non-GET Method
Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted non-GET method (POST) for handshake.
Solution:	Restrict WebSocket handshakes to GET method.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at stream.binance.com:9443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Fake Host Header

Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted handshake with incorrect Host header.
Solution:	Validate Host header to match expected server domain.

Name:	Multiple Host Headers
Risk Level:	High
Description:	Server at stream.binance.com:9443 accepted handshake with multiple Host headers.
Solution:	Reject requests with duplicate Host headers.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie

Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://stream.binance.com:9443/ws accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://stream.binance.com:9443/ws handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://stream.binance.com:9443/ws (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://stream.binance.com:9443/ws allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://stream.binance.com:9443/ws (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws allowed 100 concurrent connections in 1.16s.

Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High

Description:	WebSocket at wss://stream.binance.com:9443/ws allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://stream.binance.com:9443/ws accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High

Description:	WebSocket at wss://stream.binance.com:9443/ws accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium

Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Low
Description:	WebSocket at wss://stream.binance.com:9443/ws closed connection on malformed payload: Large Payload for DoS (JSON).
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium

Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium

Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Low
Description:	WebSocket at wss://stream.binance.com:9443/ws closed connection on malformed payload: Oversized DoS Message (JSON).
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #11
Risk Level:	Medium

Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

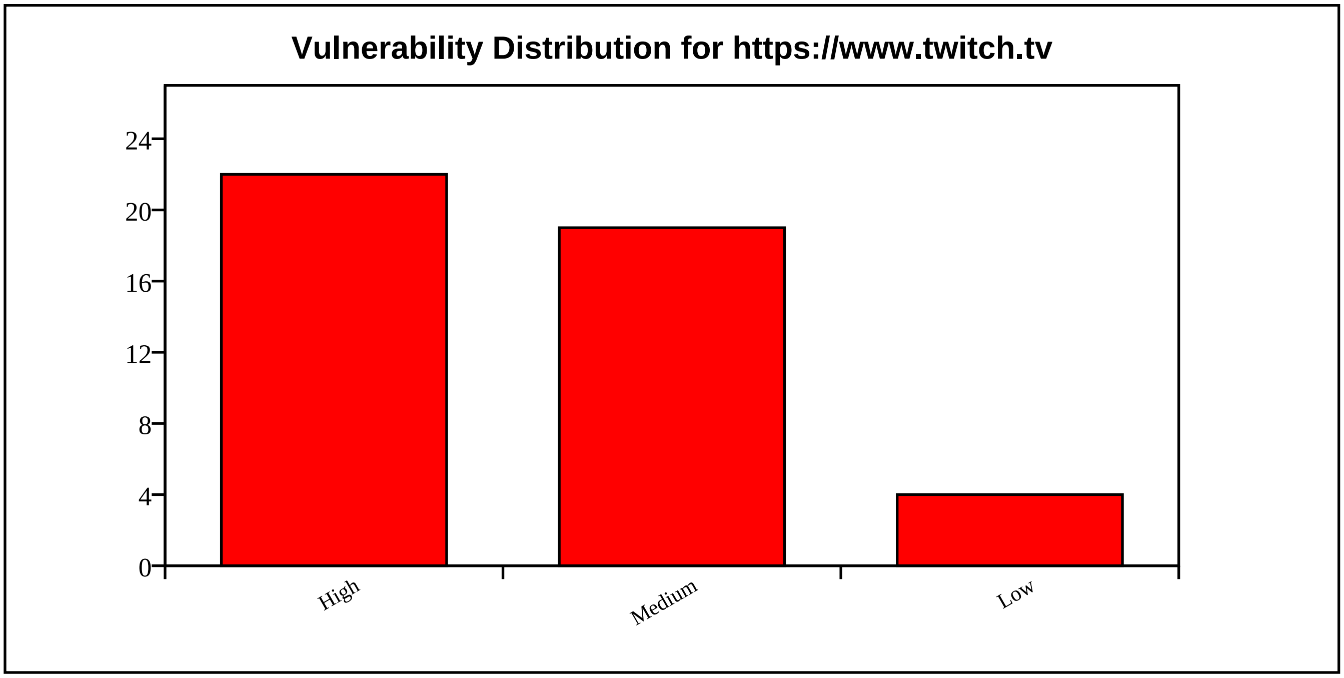
Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://stream.binance.com:9443/ws responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Target URL: https://www.twitch.tv

Scan Duration:	442.12 seconds
URLs Crawled:	28
WebSocket Endpoints Found:	0
Attack Performed:	True
High Severity Findings:	22
Medium Severity Findings:	19
Low Severity Findings:	4

WebSocket Endpoints:

#	URL
1	wss://irc-ws.chat.twitch.tv


Vulnerability Distribution for https://www.twitch.tv
0
4
8
12
16
20
24
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://irc-ws.chat.twitch.tv

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at irc-ws.chat.twitch.tv:443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at irc-ws.chat.twitch.tv:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key

Risk Level:	Medium
Description:	Server at irc-ws.chat.twitch.tv:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Wrong Upgrade Header
Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake with wrong Upgrade header.
Solution:	Enforce strict Upgrade header validation.

Name:	Missing Connection Header

Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at irc-ws.chat.twitch.tv:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Fake Host Header

Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake with incorrect Host header.
Solution:	Validate Host header to match expected server domain.

Name:	Multiple Host Headers
Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake with multiple Host headers.
Solution:	Reject requests with duplicate Host headers.

Name:	Long URL Path
Risk Level:	Low
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake with long URL path (2KB).
Solution:	Limit URL path length to prevent resource exhaustion.

Name:	Unicode URL

Risk Level:	Medium
Description:	Server at irc-ws.chat.twitch.tv:443 accepted handshake with Unicode URL.
Solution:	Sanitize and validate URL paths to handle Unicode correctly.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepted cross-origin cookies and origin header.

Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Fake Extension
Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High

Description:	Server at irc-ws.chat.twitch.tv:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at irc-ws.chat.twitch.tv:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://irc-ws.chat.twitch.tv handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Missing CORS Headers
Risk Level:	High

Description:	WebSocket endpoint wss://irc-ws.chat.twitch.tv (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://irc-ws.chat.twitch.tv allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium

Description:	WebSocket endpoint wss://irc-ws.chat.twitch.tv (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	URL Path Traversal
Risk Level:	High
Description:	WebSocket endpoint allows path traversal via: wss://irc-ws.chat.twitch.tv/ws/../admin/socket
Solution:	Validate and normalize paths to prevent traversal.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv allowed 100 concurrent connections in 1.56s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Max Connections

Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://irc-ws.chat.twitch.tv accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Low
Description:	WebSocket at wss://irc-ws.chat.twitch.tv closed connection on malformed payload: Invalid Binary Frame.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://irc-ws.chat.twitch.tv responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

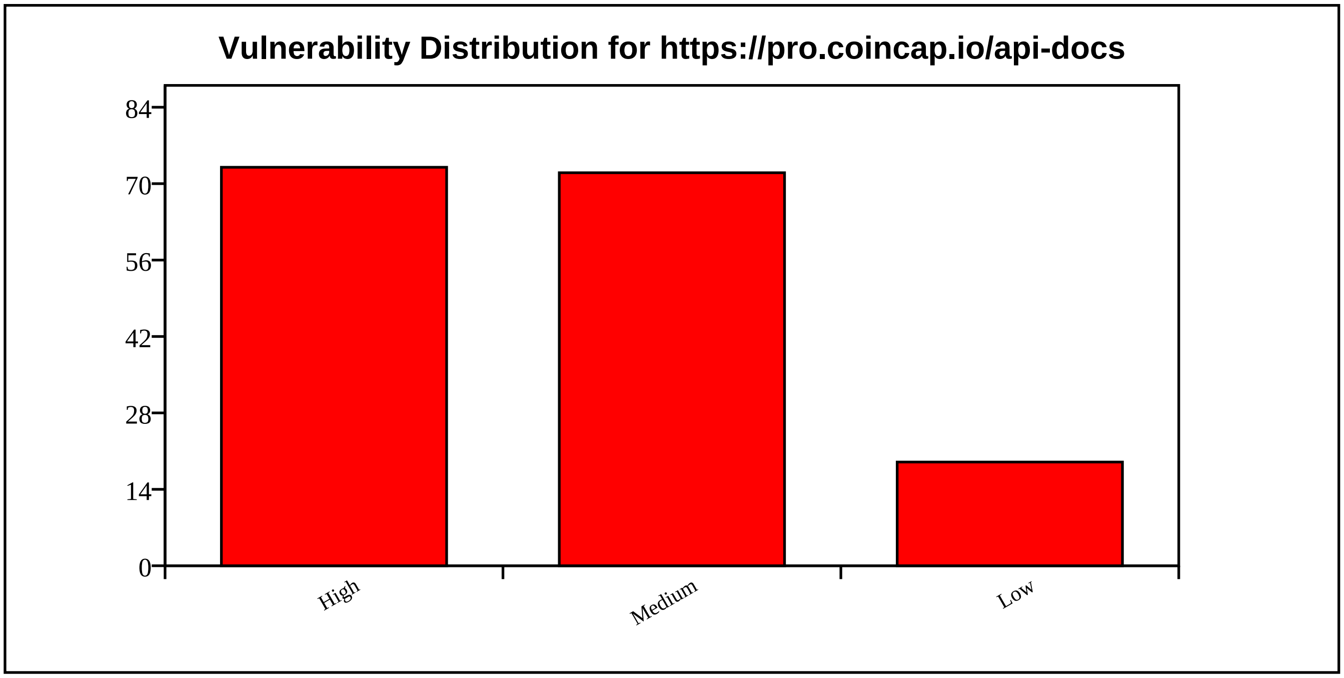
Target URL: https://pro.coincap.io/api-docs

Scan Duration:	278.72 seconds
URLs Crawled:	22
WebSocket Endpoints Found:	5
Attack Performed:	True
High Severity Findings:	73
Medium Severity Findings:	72
Low Severity Findings:	19

WebSocket Endpoints:

#	URL
1	wss://wss.coincap.io/prices?assets=ALL&a; mp;amp;apiKey=YourApiKey
2	wss://ws.coincap.io/</strong>

3	wss://wss.coincap.io/</strong>< /li><br><li>All
4	wss://wss.coincap.io/</strong>< /div><div
5	wss://wss.coincap.io/prices?assets=bitco in,usdc&amp;apiKey=XXX


Vulnerability Distribution for https://pro.coincap.io/api-docs

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://wss.coincap.io/prices

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at wss.coincap.io:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers

Risk Level:	Medium
Description:	Server at wss.coincap.io:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Long URL Path
Risk Level:	Low
Description:	Server at wss.coincap.io:443 accepted handshake with long URL path (2KB).
Solution:	Limit URL path length to prevent resource exhaustion.

Name:	Unicode URL
Risk Level:	Medium
Description:	Server at wss.coincap.io:443 accepted handshake with Unicode URL.
Solution:	Sanitize and validate URL paths to handle Unicode correctly.

Name:	No Session Cookie

Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices allows unauthenticated connections and responds with data.

Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted spoofed extension.

Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted frame with undefined opcode 0x3.

Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io/prices accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted frame with declared payload length 10 but sent only 4 bytes.

Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted a frame with invalid masking key pattern: All-zero.

Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices accepted a ping control frame with 126-byte payload.

Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io/prices accepted a text frame with non-UTF-8 binary data.

Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io/prices accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io/prices handled abrupt TCP closure and allowed clean reconnection.

Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://wss.coincap.io/prices (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://wss.coincap.io/prices allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Security Headers
Risk Level:	Medium

Description:	WebSocket endpoint wss://wss.coincap.io/prices (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	URL Path Traversal
Risk Level:	High
Description:	WebSocket endpoint allows path traversal via: wss://wss.coincap.io/ws/../admin/socket
Solution:	Validate and normalize paths to prevent traversal.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Idle Timeout Abuse

Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak

Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io/prices accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io/prices responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://wss.coincap.io

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at wss.coincap.io:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at wss.coincap.io:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Long URL Path
Risk Level:	Low
Description:	Server at wss.coincap.io:443 accepted handshake with long URL path (2KB).
Solution:	Limit URL path length to prevent resource exhaustion.

Name:	Unicode URL
Risk Level:	Medium
Description:	Server at wss.coincap.io:443 accepted handshake with Unicode URL.
Solution:	Sanitize and validate URL paths to handle Unicode correctly.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io allows reconnection with same stale session cookie.

Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium

Description:	WebSocket at wss://wss.coincap.io negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High

Description:	Server at wss.coincap.io:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at wss.coincap.io:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High

Description:	WebSocket at wss://wss.coincap.io accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High

Description:	WebSocket at wss://wss.coincap.io accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High

Description:	WebSocket at wss://wss.coincap.io accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High

Description:	WebSocket at wss://wss.coincap.io accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low

Description:	WebSocket at wss://wss.coincap.io accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://wss.coincap.io accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	Long Close Reason
Risk Level:	Medium

Description:	WebSocket at wss://wss.coincap.io accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://wss.coincap.io (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://wss.coincap.io allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High

Description:	WebSocket at wss://wss.coincap.io accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://wss.coincap.io (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio

Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://wss.coincap.io accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11

Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://wss.coincap.io responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://ws.coincap.io

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at ws.coincap.io:443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at ws.coincap.io:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at ws.coincap.io:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at ws.coincap.io:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at ws.coincap.io:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Long URL Path
Risk Level:	Low
Description:	Server at ws.coincap.io:443 accepted handshake with long URL path (2KB).
Solution:	Limit URL path length to prevent resource exhaustion.

Name:	Unicode URL
Risk Level:	Medium
Description:	Server at ws.coincap.io:443 accepted handshake with Unicode URL.
Solution:	Sanitize and validate URL paths to handle Unicode correctly.

Name:	No Session Cookie

Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io allows unauthenticated connections and responds with data.

Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at ws.coincap.io:443 accepted spoofed extension.

Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at ws.coincap.io:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at ws.coincap.io:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted frame with undefined opcode 0x3.

Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://ws.coincap.io accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted frame with declared payload length 10 but sent only 4 bytes.

Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted a frame with invalid masking key pattern: All-zero.

Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepted a ping control frame with 126-byte payload.

Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://ws.coincap.io accepted a text frame with non-UTF-8 binary data.

Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://ws.coincap.io accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://ws.coincap.io accepted an early close frame before any data was exchanged.

Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://ws.coincap.io handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://ws.coincap.io (HTTP equivalent) lacks proper CORS headers.

Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://ws.coincap.io allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium

Description:	WebSocket endpoint wss://ws.coincap.io (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	URL Path Traversal
Risk Level:	High
Description:	WebSocket endpoint allows path traversal via: wss://ws.coincap.io/ws/../admin/socket
Solution:	Validate and normalize paths to prevent traversal.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io allowed 100 concurrent connections in 1.74s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message

Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak

Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://ws.coincap.io accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

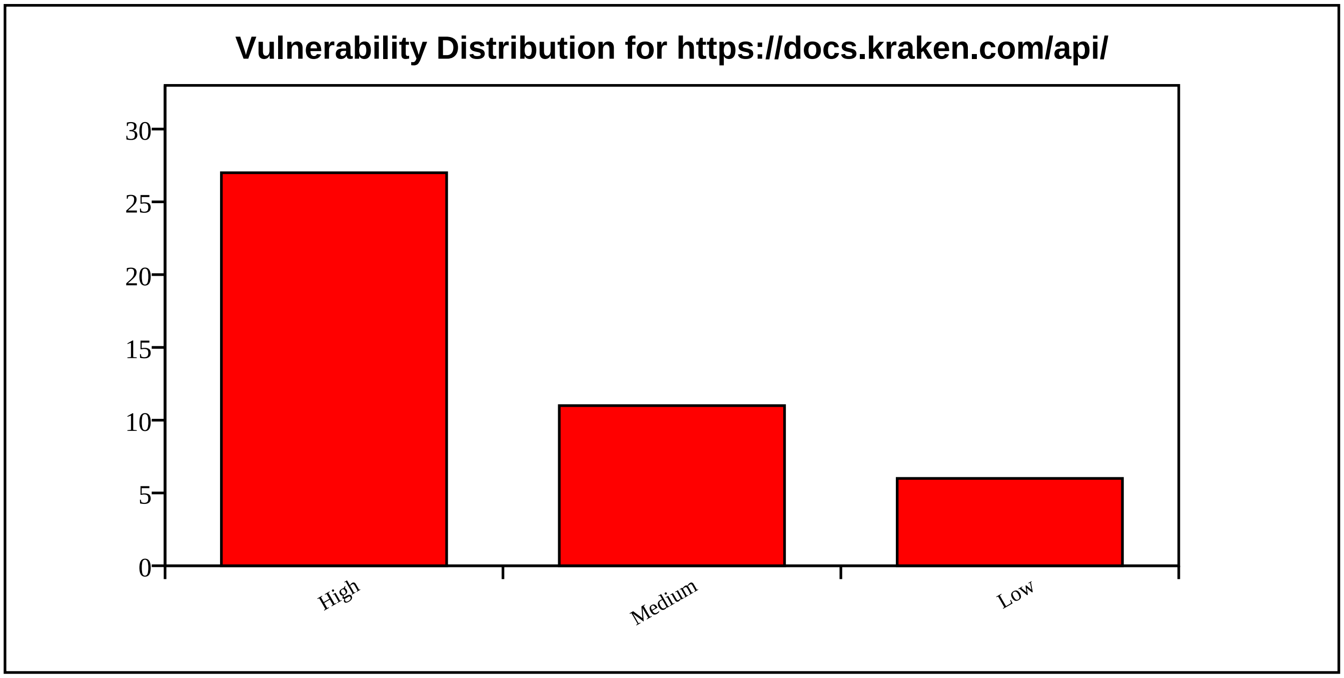
Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://ws.coincap.io responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Target URL: https://docs.kraken.com/api/

Scan Duration:	154.29 seconds
URLs Crawled:	1
WebSocket Endpoints Found:	0
Attack Performed:	True
High Severity Findings:	27
Medium Severity Findings:	11
Low Severity Findings:	6

WebSocket Endpoints:

#	URL
1	wss://ws.kraken.com


Vulnerability Distribution for https://docs.kraken.com/api/
0
5
10
15
20
25
30
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://ws.kraken.com

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at ws.kraken.com:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version

Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Wrong Upgrade Header
Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted handshake with wrong Upgrade header.
Solution:	Enforce strict Upgrade header validation.

Name:	Missing Connection Header

Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at ws.kraken.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie

Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	HTTP Session Reuse
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com reused HTTP session cookie without revalidation.
Solution:	Require revalidation or token-based auth for WebSockets even if HTTP session exists.

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com allows unauthenticated connections and responds with data.

Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted spoofed extension.

Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at ws.kraken.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepts insecure TLS cipher: NULL-MD5.

Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://ws.kraken.com accepted zero-length fragments and responded unexpectedly.

Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepted frames with mismatched lengths.

Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepted a frame with invalid RSV1 bit set.

Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepted a text frame containing null bytes.

Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://ws.kraken.com accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://ws.kraken.com accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepted a close frame with invalid code 999.

Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://ws.kraken.com accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://ws.kraken.com handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://ws.kraken.com accepted close frame with long reason (123 bytes).

Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://ws.kraken.com (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://ws.kraken.com allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepts connections from unauthorized origin 'http://malicious-site.com'.

Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://ws.kraken.com (HTTP equivalent) lacks the following headers: Content-Security-Policy, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com allowed 100 concurrent connections in 3.35s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://ws.kraken.com accepted hanging TCP connections without timeout.

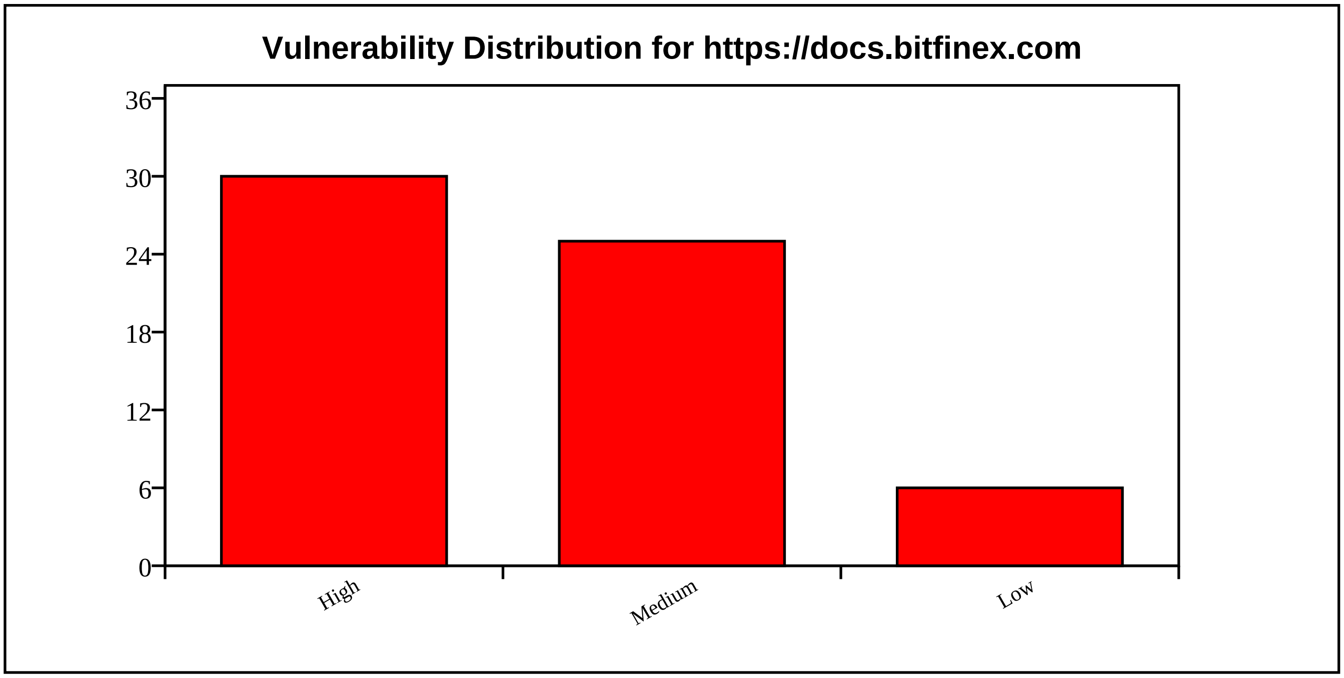
Solution:	Use TCP keep-alive and server-side timeout policies.

Target URL: https://docs.bitfinex.com

Scan Duration:	316.6 seconds
URLs Crawled:	6
WebSocket Endpoints Found:	0
Attack Performed:	True
High Severity Findings:	30
Medium Severity Findings:	25
Low Severity Findings:	6

WebSocket Endpoints:

#	URL
1	wss://api-pub.bitfinex.com/ws/2


Vulnerability Distribution for https://docs.bitfinex.com
36
30
24
18
12
6
0
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://api-pub.bitfinex.com/ws/2

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at api-pub.bitfinex.com:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version

Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Wrong Upgrade Header
Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted handshake with wrong Upgrade header.
Solution:	Enforce strict Upgrade header validation.

Name:	Missing Connection Header

Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at api-pub.bitfinex.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at api-pub.bitfinex.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	No Session Cookie

Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 allows unauthenticated connections and responds with data.

Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted spoofed extension.

Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at api-pub.bitfinex.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepts insecure TLS cipher: NULL-MD5.

Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted zero-length fragments and responded unexpectedly.

Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted frames with mismatched lengths.

Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a frame with invalid RSV1 bit set.

Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a text frame containing null bytes.

Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a close frame with invalid code 999.

Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted close frame with long reason (123 bytes).

Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://api-pub.bitfinex.com/ws/2 (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://api-pub.bitfinex.com/ws/2 (HTTP equivalent) lacks the following headers: Content-Security-Policy.

Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 allowed 100 concurrent connections in 1.52s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High

Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High

Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1

Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4

Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7

Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10

Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

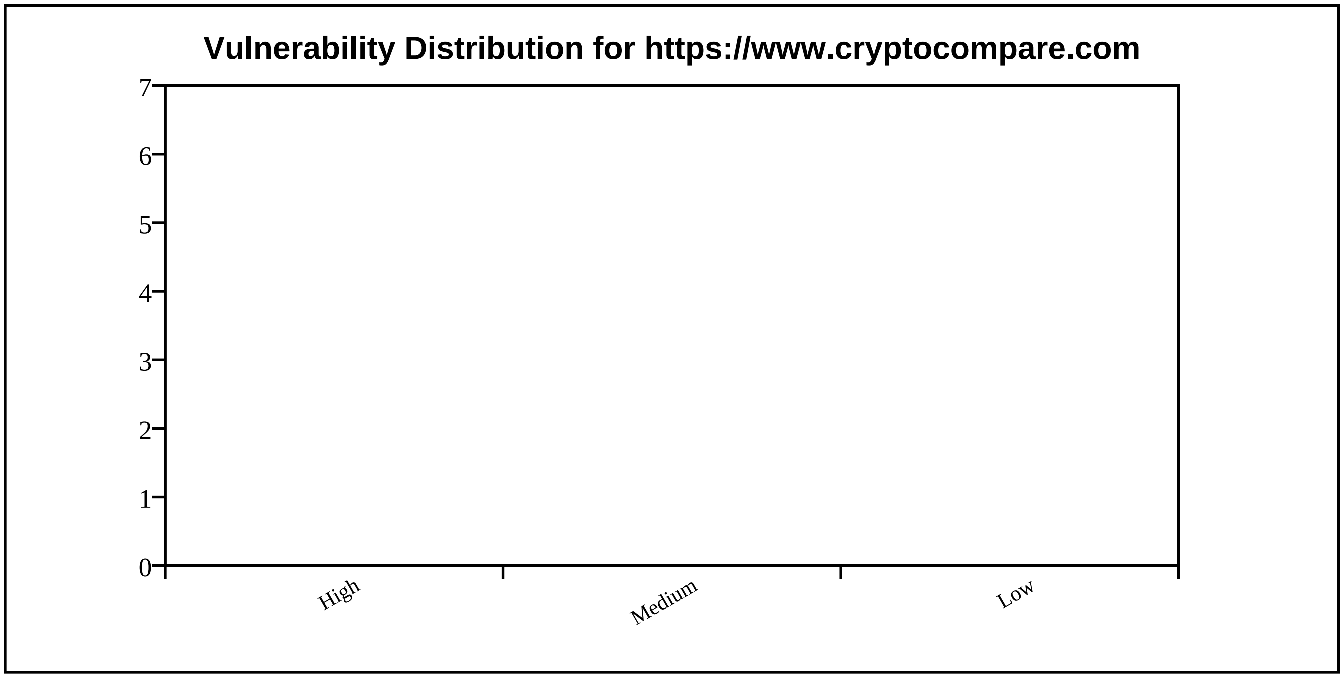
Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://api-pub.bitfinex.com/ws/2 responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Target URL: https://www.cryptocompare.com

Scan Duration:	348.24 seconds
URLs Crawled:	150
WebSocket Endpoints Found:	1
Attack Performed:	False
High Severity Findings:	0
Medium Severity Findings:	0
Low Severity Findings:	0

WebSocket Endpoints:

#	URL
1	wss://streamer.cryptocompare.com/v2?form at=streamer

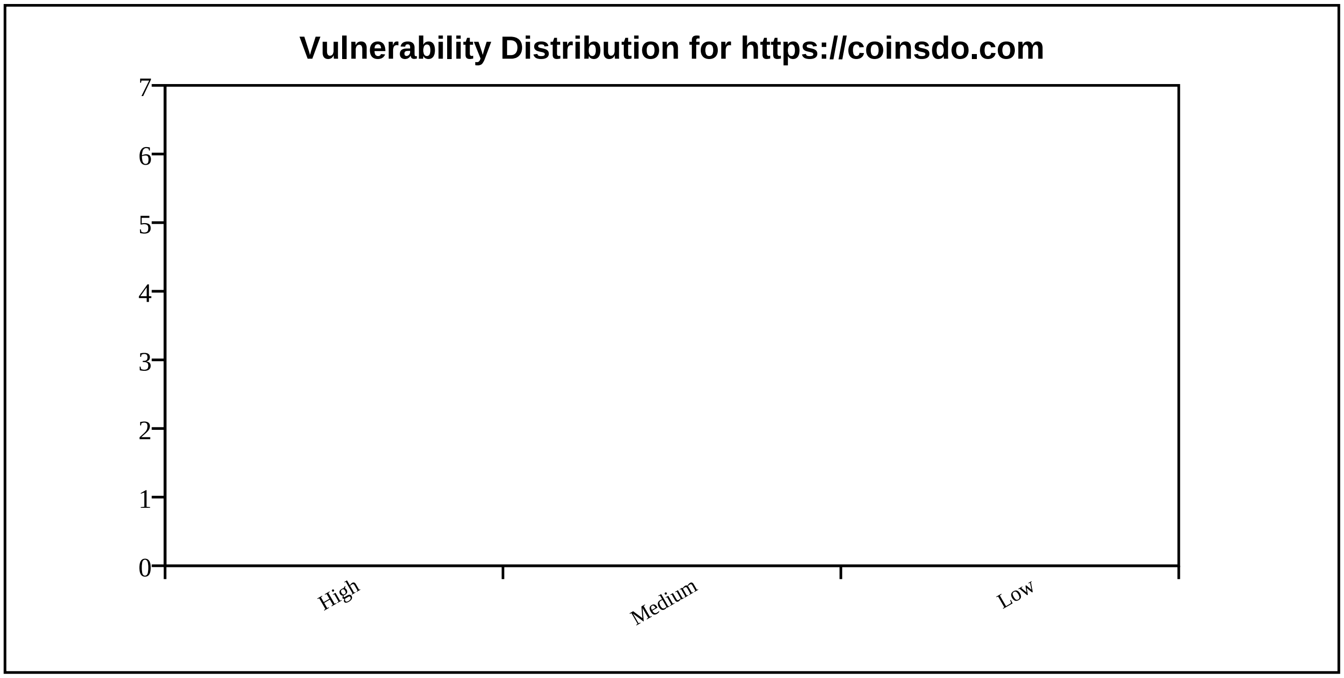

Vulnerability Distribution for https://www.cryptocompare.com
7
6
5
4
3
2
1
0
High
Medium
Low

Target URL: https://coinsdo.com

Scan Duration:	106.81 seconds
URLs Crawled:	150
WebSocket Endpoints Found:	0
Attack Performed:	False
High Severity Findings:	0
Medium Severity Findings:	0
Low Severity Findings:	0

WebSocket Endpoints:

#	URL
1	wss://ws.coinsdo.com/ws

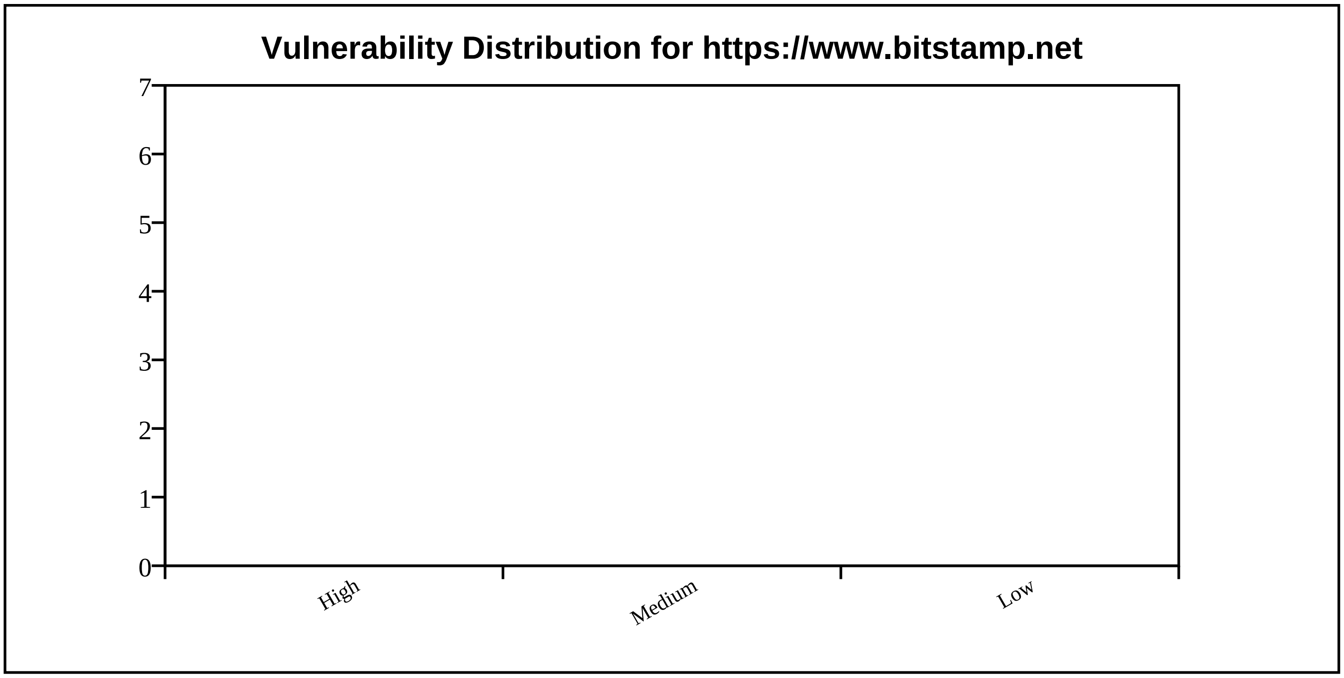

Vulnerability Distribution for https://coinsdo.com
7
6
5
4
3
2
1
0
High
Medium
Low

Target URL: https://www.bitstamp.net

Scan Duration:	375.93 seconds
URLs Crawled:	136
WebSocket Endpoints Found:	2
Attack Performed:	False
High Severity Findings:	0
Medium Severity Findings:	0
Low Severity Findings:	0

WebSocket Endpoints:

#	URL
1	wss://ws.bitstamp.net
2	wss://ws.bitstamp.net/

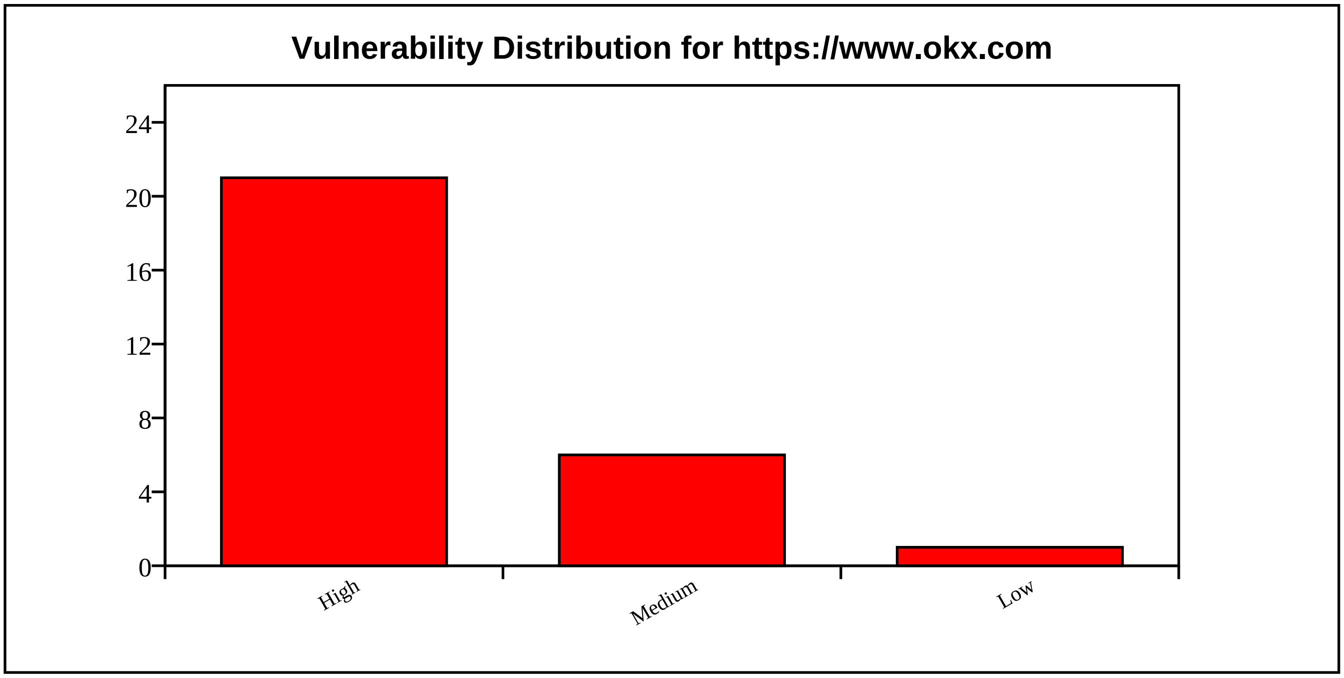

Vulnerability Distribution for https://www.bitstamp.net
7
6
5
4
3
2
1
0
High
Medium
Low

Target URL: https://www.okx.com

Scan Duration:	381.74 seconds
URLs Crawled:	1
WebSocket Endpoints Found:	1
Attack Performed:	True
High Severity Findings:	21
Medium Severity Findings:	6
Low Severity Findings:	1

WebSocket Endpoints:

#	URL
1	wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3?X-Nexus-New-Cl ient=true&X-Nexus-Version=0.14.0&;user_role=visitor


Vulnerability Distribution for https://www.okx.com
24
20
16
12
8
4
0
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at nexus-websocket-a.intercom.io:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at nexus-websocket-a.intercom.io:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at nexus-websocket-a.intercom.io:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Fake Host Header
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted handshake with incorrect Host header.
Solution:	Validate Host header to match expected server domain.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepts connections without a session cookie.

Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect

Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Fake Extension
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted spoofed extension.

Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at nexus-websocket-a.intercom.io:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High

Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).

Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Server Disclosure
Risk Level:	Medium
Description:	WebSocket HTTP interface discloses: Server: nginx.
Solution:	Disable or obscure headers like Server, X-Powered-By, and X-AspNet-Version.

Name:	Missing Security Headers
Risk Level:	Medium

Description:	WebSocket endpoint wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 allowed 100 concurrent connections in 1.02s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 allows 100 simultaneous connections without restriction.

Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High

Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://nexus-websocket-a.intercom.io/pubsub/5-Z9Rb7YqZ0aoFksBDZvR88wQPQp0h0hbaBRi2HFQK7xcpnFvS7xbCJ1lTcHd0DYimVtzgoFDuZABGntId8A8s-SHTettQzV72C2I3 may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Target URL: https://publicnode.com

Scan Duration:	412.33 seconds
URLs Crawled:	150
WebSocket Endpoints Found:	162
Attack Performed:	True
High Severity Findings:	90
Medium Severity Findings:	78
Low Severity Findings:	19

WebSocket Endpoints:

#	URL
1	wss://atomone-rpc.publicnode.com:443/web socket
2	wss://aurora-rpc.publicnode.com

3	wss://mantra-testnet-rpc.publicnode.com: 443/websocket
4	wss://dydx-rpc.publicnode.com:443/websoc ket
5	wss://saga-rpc.publicnode.com:443/websoc ket
6	wss://blast-rpc.publicnode.com
7	wss://babylon-rpc.publicnode.com:443/web socket
8	wss://teritori-rpc.publicnode.com:443/we bsocket
9	wss://arbitrum-sepolia-rpc.publicnode.co m
10	wss://kava-evm-rpc.publicnode.com
11	wss://iris-evm-rpc.publicnode.com
12	wss://evmos-evm-rpc.publicnode.com
13	wss://api.publicnode.com/ws?platform=fir o-mainnet-rpc
14	wss://api.publicnode.com/ws?platform=apt os-rest
15	wss://fetch-rpc.publicnode.com:443/webso cket
16	wss://avail-rpc.publicnode.com

17	wss://stride-rpc.publicnode.com:443/webs ocket
18	wss://osmosis-rpc.publicnode.com:443/web socket
19	wss://opbnb-testnet-rpc.publicnode.com
20	wss://berachain-bepolia-rpc.publicnode.c om
21	wss://tenet-evm-rpc.publicnode.com
22	wss://xpla-rpc.publicnode.com:443/websoc ket
23	wss://passage-rpc.publicnode.com:443/web socket
24	wss://linea-sepolia-rpc.publicnode.com
25	wss://avalanche-fuji-c-chain-rpc.publicn ode.com
26	wss://gnosis-rpc.publicnode.com
27	wss://kava-rpc.publicnode.com:443/websoc ket
28	wss://polkadot-rpc.publicnode.com
29	wss://sui-testnet-rpc.publicnode.com
30	wss://taiko-rpc.publicnode.com

31	wss://omniflix-rpc.publicnode.com:443/we bsocket
32	wss://injective-rpc.publicnode.com:443/w ebsocket
33	wss://scroll-sepolia-rpc.publicnode.com
34	wss://chiliz-rpc.publicnode.com
35	wss://chiliz-spicy-rpc.publicnode.com
36	wss://api.publicnode.com/ws?platform=sys coin-evm,syscoin-ws-evm,syscoin-tanenbau m-evm,syscoin-tanenbaum-ws-evm
37	wss://mantle-rpc.publicnode.com
38	wss://migaloo-rpc.publicnode.com:443/web socket
39	wss://api.publicnode.com/ws?platform=dyd x-rpc,dydx-ws-rpc,dydx-grpc,dydx-grpc-we b,dydx-rest
40	wss://bahamut-rpc.publicnode.com
41	wss://analog-rpc.publicnode.com
42	wss://fantom-testnet-rpc.publicnode.com
43	wss://manta-atlantic-rpc.publicnode.com

44	wss://akash-rpc.publicnode.com:443/webso cket
45	wss://shentu-rpc.publicnode.com:443/webs ocket
46	wss://lava-rpc.publicnode.com:443/websoc ket
47	wss://api.publicnode.com/ws?platform=ber achain-rpc,berachain-ws-rpc,berachain-beacon-rpc,berachain-bepolia-rpc,berachain -bepolia-ws-rpc,berachain-bepolia-beacon-rpc
48	wss://iris-rpc.publicnode.com:443/websoc ket
49	wss://stargaze-rpc.publicnode.com:443/we bsocket
50	wss://rebus-rpc.publicnode.com:443/webso cket
51	wss://cronos-pos-rpc.publicnode.com:443/ websocket
52	wss://api.publicnode.com/ws?platform=iri s-evm,iris-ws-evm,iris-rpc,iris-ws-rpc,iris-grpc,iris-grpc-web,iris-rest
53	wss://base-rpc.publicnode.com
54	wss://metis-rpc.publicnode.com:443
55	wss://evmos-testnet-evm-rpc.publicnode.c om
56	wss://celestia-rpc.publicnode.com:443/we bsocket

57	wss://sei-evm-rpc.publicnode.com
58	wss://peaq-agung-rpc.publicnode.com
59	wss://metis-sepolia-rpc.publicnode.com:4 43
60	wss://dymension-evm-rpc.publicnode.com
61	wss://cheqd-rpc.publicnode.com:443/webso cket
62	wss://nibiru-rpc.publicnode.com:443/webs ocket
63	wss://api.publicnode.com/ws?platform=bit coin-mainnet-rpc,bitcoin-testnet-rpc
64	wss://solana-rpc.publicnode.com
65	wss://haqq-rpc.publicnode.com:443/websoc ket
66	wss://dora-rpc.publicnode.com:443/websoc ket
67	wss://sifchain-rpc.publicnode.com:443/we bsocket
68	wss://pulsechain-rpc.publicnode.com
69	wss://quicksilver-rpc.publicnode.com:443 /websocket
70	wss://celestia-mocha-rpc.publicnode.com: 443/websocket

71	wss://dymension-testnet-rpc.publicnode.c om:443/websocket
72	wss://cronos-rpc.publicnode.com:443/webs ocket
73	wss://persistence-rpc.publicnode.com:443 /websocket
74	wss://optimism-sepolia-rpc.publicnode.co m
75	wss://evmos-rpc.publicnode.com:443/webso cket
76	wss://sentinel-rpc.publicnode.com:443/we bsocket
77	wss://berachain-rpc.publicnode.com
78	wss://arbitrum-nova-rpc.publicnode.com
79	wss://elys-testnet-rpc.publicnode.com:44 3/websocket
80	wss://taiko-hekla-rpc.publicnode.com
81	wss://moonriver-rpc.publicnode.com
82	wss://terra-classic-rpc.publicnode.com:4 43/websocket
83	wss://ethereum-rpc.publicnode.com
84	wss://terra-rpc.publicnode.com:443/webso cket

85	wss://oraichain-rpc.publicnode.com:443/w ebsocket
86	wss://solana-testnet-rpc.publicnode.com
87	wss://kusama-rpc.publicnode.com
88	wss://xpla-evm-rpc.publicnode.com
89	wss://elys-rpc.publicnode.com:443/websoc ket
90	wss://api.publicnode.com/ws?platform=str ide-rpc,stride-ws-rpc,stride-grpc,stride -grpc-web,stride-rest
91	wss://api.publicnode.com/ws?platform=ass et-mantle-rpc,asset-mantle-ws-rpc,asset-mantle-grpc,asset-mantle-grpc-web,asset- mantle-rest
92	wss://sonic-rpc.publicnode.com:443
93	wss://avalanche-c-chain-rpc.publicnode.c om
94	wss://ethereum-sepolia-rpc.publicnode.co m
95	wss://celo-rpc.publicnode.com
96	wss://polygon-heimdall-rpc.publicnode.co m:443/websocket
97	wss://dymension-rpc.publicnode.com:443/w ebsocket

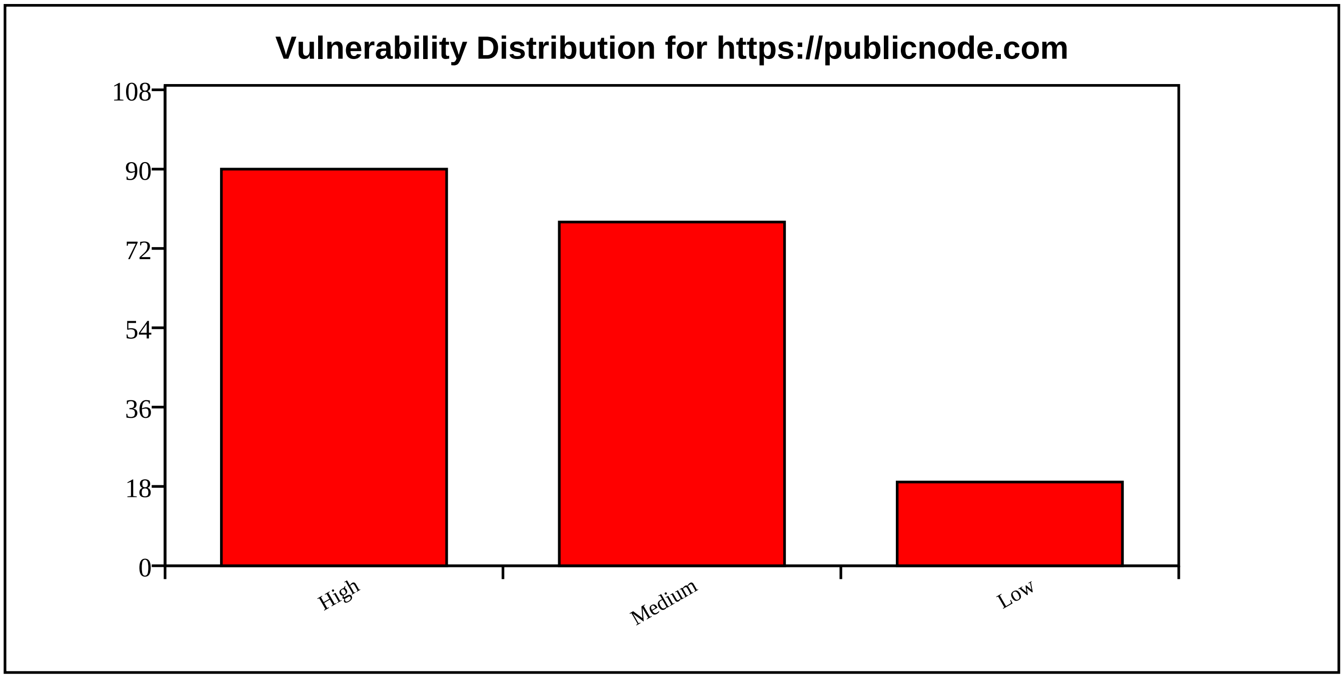
98	wss://polygon-bor-rpc.publicnode.com
99	wss://asset-mantle-rpc.publicnode.com:44 3/websocket
100	wss://optimism-rpc.publicnode.com
101	wss://moonbeam-rpc.publicnode.com
102	wss://arbitrum-one-rpc.publicnode.com
103	wss://pulsechain-testnet-rpc.publicnode. com
104	wss://unichain-rpc.publicnode.com
105	wss://linea-rpc.publicnode.com
106	wss://injective-testnet-rpc.publicnode.c om:443/websocket
107	wss://cronos-evm-rpc.publicnode.com
108	wss://kujira-rpc.publicnode.com:443/webs ocket
109	wss://ethereum-holesky-rpc.publicnode.co m
110	wss://medibloc-rpc.publicnode.com:443/we bsocket
111	wss://ethereum-hoodi-rpc.publicnode.com

112	wss://api.publicnode.com/ws?platform=ten et-evm,tenet-ws-evm,tenet-rpc,tenet-ws-rpc,tenet-grpc,tenet-grpc-web,tenet-rest
113	wss://bsc-testnet-rpc.publicnode.com
114	wss://haqq-evm-rpc.publicnode.com
115	wss://nolus-rpc.publicnode.com:443/webso cket
116	wss://scroll-rpc.publicnode.com
117	wss://sei-rpc.publicnode.com:443/websock et
118	wss://bitcanna-rpc.publicnode.com:443/we bsocket
119	wss://fantom-rpc.publicnode.com
120	wss://gnosis-chiado-rpc.publicnode.com
121	wss://api.publicnode.com/ws?platform=def i-mainnet-rpc
122	wss://api.publicnode.com/ws?platform=jun o-rpc,juno-ws-rpc,juno-grpc,juno-grpc-web,juno-rest
123	wss://fraxtal-holesky-rpc.publicnode.com :443
124	wss://soneium-minato-rpc.publicnode.com

125	wss://polygon-amoy-heimdall-rpc.publicno de.com:443/websocket
126	wss://starknet-sepolia-rpc.publicnode.co m
127	wss://api.publicnode.com/ws?platform=ethereum-rpc,ethereum-ws-rpc,ethereum-teku- rpc,ethereum-sepolia-rpc,ethereum-sepolia-ws-rpc,ethereum-sepolia-teku-rpc,ether eum-holesky-rpc,ethereum-holesky-ws-rpc,ethereum-holesky-teku-rpc,ethereum-hoodi-rpc,ethereum-hoodi-ws-rpc,ethereum-hood i-teku-rpc
128	wss://soneium-rpc.publicnode.com
129	wss://nibiru-evm-rpc.publicnode.com
130	wss://rizon-rpc.publicnode.com:443/webso cket
131	wss://chihuahua-rpc.publicnode.com:443/w ebsocket
132	wss://evmos-testnet-rpc.publicnode.com:4 43/websocket
133	wss://peaq-rpc.publicnode.com
134	wss://sui-rpc.publicnode.com
135	wss://syscoin-tanenbaum-evm-rpc.publicno de.com
136	wss://sonic-blaze-rpc.publicnode.com:443
137	wss://polygon-amoy-bor-rpc.publicnode.co m

138	wss://juno-rpc.publicnode.com:443/websoc ket
139	wss://regen-rpc.publicnode.com:443/webso cket
140	wss://coreum-rpc.publicnode.com:443/webs ocket
141	wss://fraxtal-rpc.publicnode.com:443
142	wss://unichain-sepolia-rpc.publicnode.co m
143	wss://base-sepolia-rpc.publicnode.com
144	wss://side-rpc.publicnode.com:443/websoc ket
145	wss://api.publicnode.com/ws?platform=sta rknet-rpc,starknet-ws-rpc,starknet-sepolia-rpc,starknet-sepolia-ws-rpc
146	wss://api.publicnode.com/ws?platform=evmos-evm,evmos-ws-evm,evmos-rpc,evmos-ws-rpc,evmos-grpc,evmos-grpc-web,evmos-rest, evmos-testnet-evm,evmos-testnet-ws-evm,evmos-testnet-rpc,evmos-testnet-ws-rpc,ev mos-testnet-grpc,evmos-testnet-grpc-web, evmos-testnet-rest
147	wss://avail-turing-rpc.publicnode.com
148	wss://celer-rpc.publicnode.com:443/webso cket
149	wss://mantra-rpc.publicnode.com:443/webs ocket

150	wss://dymension-testnet-evm-rpc.publicno de.com
151	wss://api.publicnode.com/ws?platform=sta rgaze-rpc,stargaze-ws-rpc,stargaze-grpc, stargaze-grpc-web,stargaze-rest
152	wss://comdex-rpc.publicnode.com:443/webs ocket
153	wss://tenet-rpc.publicnode.com:443/webso cket
154	wss://api.publicnode.com/ws?chain=extend ed
155	wss://opbnb-rpc.publicnode.com
156	wss://axelar-rpc.publicnode.com:443/webs ocket
157	wss://api.publicnode.com/ws?platform=kav a-evm,kava-ws-evm,kava-rpc,kava-ws-rpc,k ava-grpc,kava-grpc-web,kava-rest
158	wss://bsc-rpc.publicnode.com
159	wss://starknet-rpc.publicnode.com
160	wss://cosmos-rpc.publicnode.com:443/webs ocket
161	wss://neutron-rpc.publicnode.com:443/web socket
162	wss://syscoin-evm-rpc.publicnode.com


Vulnerability Distribution for https://publicnode.com
108
90
72
54
36
18
0
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://mantra-testnet-rpc.publicnode.com:443/websocket

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version
Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepts connections with an expired session cookie.

Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High

Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Fake Extension
Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header

Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at mantra-testnet-rpc.publicnode.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode

Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length

Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted close frame with long reason (123 bytes).

Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://mantra-testnet-rpc.publicnode.com:443/websocket (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Invalid Content-Type
Risk Level:	Medium

Description:	WebSocket endpoint wss://mantra-testnet-rpc.publicnode.com:443/websocket (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://mantra-testnet-rpc.publicnode.com:443/websocket (HTTP equivalent) lacks the following headers: Content-Security-Policy, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Query Parameter Flood
Risk Level:	High
Description:	WebSocket endpoint wss://mantra-testnet-rpc.publicnode.com:443/websocket?[1000 params] accepts 1000 query parameters.
Solution:	Limit query parameters and implement strict validation.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket allowed 100 concurrent connections in 1.81s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket allows 100 simultaneous connections without restriction.

Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High

Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.c om:443/websocket responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.c om:443/websocket responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.c om:443/websocket responded to malformed payload type: Large Payload for DoS (JSON).

Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium

Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9

Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://mantra-testnet-rpc.publicnode.com:443/websocket responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://atomone-rpc.publicnode.com:443/websocket

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at atomone-rpc.publicnode.com:443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium

Description:	Server at atomone-rpc.publicnode.com:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at atomone-rpc.publicnode.com:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at atomone-rpc.publicnode.com:443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version
Risk Level:	High

Description:	Server at atomone-rpc.publicnode.com:443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at atomone-rpc.publicnode.com:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at atomone-rpc.publicnode.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low

Description:	Server at atomone-rpc.publicnode.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at atomone-rpc.publicnode.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium

Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack

Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Fake Extension
Risk Level:	High
Description:	Server at atomone-rpc.publicnode.com:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at atomone-rpc.publicnode.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at atomone-rpc.publicnode.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://atomone-rpc.publicnode.com:443/websocket (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint wss://atomone-rpc.publicnode.com:443/websocket (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.

Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://atomone-rpc.publicnode.com:443/websocket (HTTP equivalent) lacks the following headers: Content-Security-Policy, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Query Parameter Flood
Risk Level:	High
Description:	WebSocket endpoint wss://atomone-rpc.publicnode.com:443/websocket?[1000 params] accepts 1000 query parameters.
Solution:	Limit query parameters and implement strict validation.

Name:	Connection Flood
Risk Level:	High

Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket allowed 100 concurrent connections in 2.14s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse

Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak

Risk Level:	High
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2

Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5

Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8

Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://atomone-rpc.publicnode.com:443/websocket responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://aurora-rpc.publicnode.com

Name:	Non-Base64 Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at aurora-rpc.publicnode.com:443 accepted non-base64 Sec-WebSocket-Key.

Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at aurora-rpc.publicnode.com:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at aurora-rpc.publicnode.com:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version
Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted handshake without Sec-WebSocket-Version.

Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version
Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted handshake without Connection header.

Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at aurora-rpc.publicnode.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at aurora-rpc.publicnode.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Long URL Path
Risk Level:	Low
Description:	Server at aurora-rpc.publicnode.com:443 accepted handshake with long URL path (2KB).

Solution:	Limit URL path length to prevent resource exhaustion.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High

Description:	WebSocket at wss://aurora-rpc.publicnode.com accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication

Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Fake Extension
Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade

Risk Level:	High
Description:	Server at aurora-rpc.publicnode.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode

Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length

Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame

Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text

Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary

Risk Level:	Low
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame

Risk Level:	Low
Description:	WebSocket at wss://aurora-rpc.publicnode.com handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://aurora-rpc.publicnode.com (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Missing Origin Check

Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint wss://aurora-rpc.publicnode.com (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://aurora-rpc.publicnode.com (HTTP equivalent) lacks the following headers: Content-Security-Policy.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Query Parameter Flood
Risk Level:	High
Description:	WebSocket endpoint wss://aurora-rpc.publicnode.com?[1000 params] accepts 1000 query parameters.
Solution:	Limit query parameters and implement strict validation.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com allowed 100 concurrent connections in 1.73s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://aurora-rpc.publicnode.com accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://aurora-rpc.publicnode.com responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

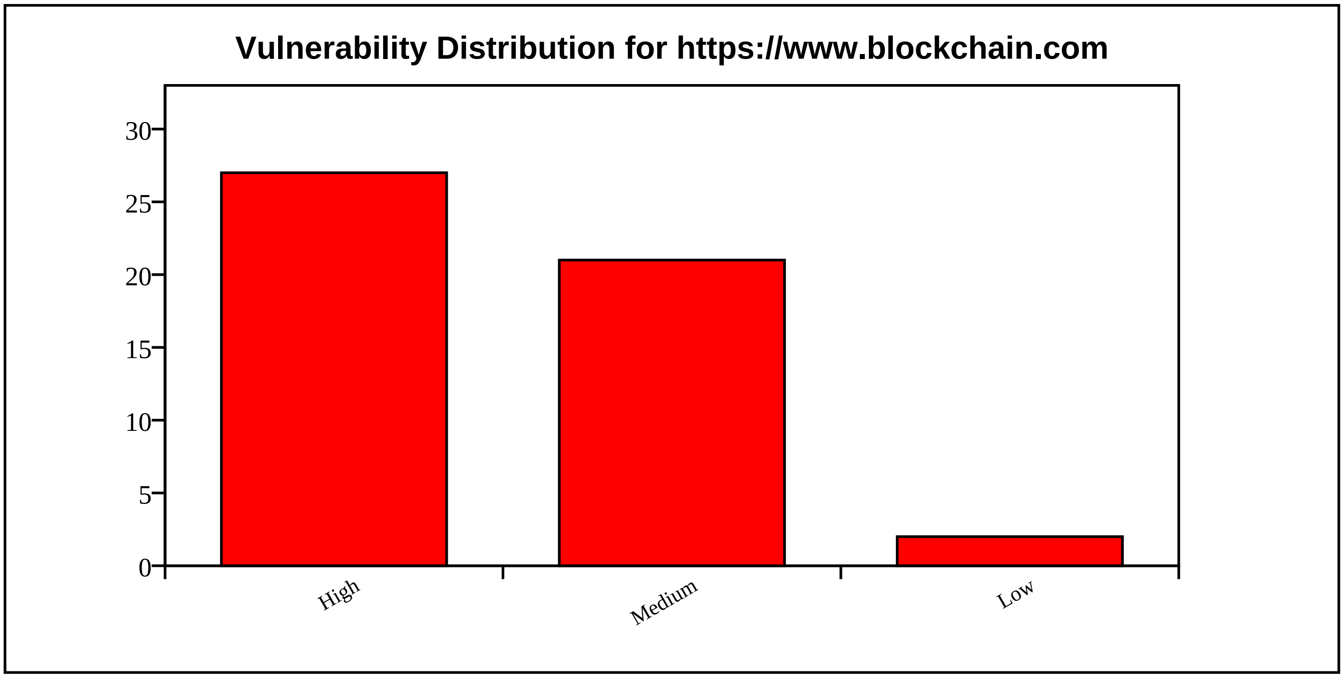
Target URL: https://www.blockchain.com

Scan Duration:	656.39 seconds
URLs Crawled:	150
WebSocket Endpoints Found:	3
Attack Performed:	True
High Severity Findings:	27
Medium Severity Findings:	21
Low Severity Findings:	2

WebSocket Endpoints:

#	URL
1	wss://ws.blockchain.info/explorer-ingest ion-caching/socket.io/?EIO=4&transport=websocket
2	wss://ws.blockchain.info

3	wss://ws.blockchain.info/coins



Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://ws.blockchain.info/explorer-ingestion-caching/socket.io

Name:	Fake HTTP Status
Risk Level:	High
Description:	Server at ws.blockchain.info:443 returned unexpected status: HTTP/1.1 502 Bad Gateway
Solution:	Ensure server returns "HTTP/1.1 101 Switching Protocols" for valid handshakes.

Name:	Wrong Sec-WebSocket-Accept
Risk Level:	Medium
Description:	Server at ws.blockchain.info:443 did not return a Sec-WebSocket-Accept header.
Solution:	Ensure server follows RFC 6455 and sends correct Sec-WebSocket-Accept header.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://ws.blockchain.info/explorer-ingestion-caching/socket.io (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint wss://ws.blockchain.info/explorer-ingestion-caching/socket.io (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://ws.blockchain.info/explorer-ingestion-caching/socket.io (HTTP equivalent) lacks the following headers: X-Frame-Options.

Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/explorer-ingestion-caching/socket.io accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Affected WebSocket Endpoint: wss://ws.blockchain.info/coins

Name:	Omit Sec-WebSocket-Key
Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted handshake without Sec-WebSocket-Key.
Solution:	Require Sec-WebSocket-Key header for WebSocket handshake.

Name:	Non-Base64 Sec-WebSocket-Key

Risk Level:	Medium
Description:	Server at ws.blockchain.info:443 accepted non-base64 Sec-WebSocket-Key.
Solution:	Validate Sec-WebSocket-Key as base64-encoded.

Name:	Oversized Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at ws.blockchain.info:443 accepted oversized Sec-WebSocket-Key (1KB).
Solution:	Limit Sec-WebSocket-Key size to prevent resource exhaustion.

Name:	Duplicate Sec-WebSocket-Key
Risk Level:	Medium
Description:	Server at ws.blockchain.info:443 accepted duplicate Sec-WebSocket-Key headers.
Solution:	Reject requests with multiple Sec-WebSocket-Key headers.

Name:	Missing Sec-WebSocket-Version

Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted handshake without Sec-WebSocket-Version.
Solution:	Require Sec-WebSocket-Version header for WebSocket handshake.

Name:	Invalid Sec-WebSocket-Version
Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted invalid Sec-WebSocket-Version.
Solution:	Validate Sec-WebSocket-Version (e.g., 13) for WebSocket handshake.

Name:	Conflicting Sec-WebSocket-Version
Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted conflicting Sec-WebSocket-Version headers.
Solution:	Reject requests with multiple Sec-WebSocket-Version headers.

Name:	Missing Connection Header

Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at ws.blockchain.info:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie

Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	HTTP Session Reuse
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins reused HTTP session cookie without revalidation.
Solution:	Require revalidation or token-based auth for WebSockets even if HTTP session exists.

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins allows unauthenticated connections and responds with data.

Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Fake Extension
Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at ws.blockchain.info:443 accepted HTTP/1.0 WebSocket handshake.

Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://ws.blockchain.info/coins handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://ws.blockchain.info/coins (HTTP equivalent) lacks proper CORS headers.

Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint wss://ws.blockchain.info/coins (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium

Description:	WebSocket endpoint wss://ws.blockchain.info/coins (HTTP equivalent) lacks the following headers: Content-Security-Policy, X-Frame-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins allowed 100 concurrent connections in 2.12s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections

Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak

Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://ws.blockchain.info/coins accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://ws.blockchain.info/coins responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Target URL: https://nownodes.io

Scan Duration:	310.42 seconds
URLs Crawled:	150
WebSocket Endpoints Found:	36
Attack Performed:	True
High Severity Findings:	87
Medium Severity Findings:	78
Low Severity Findings:	18

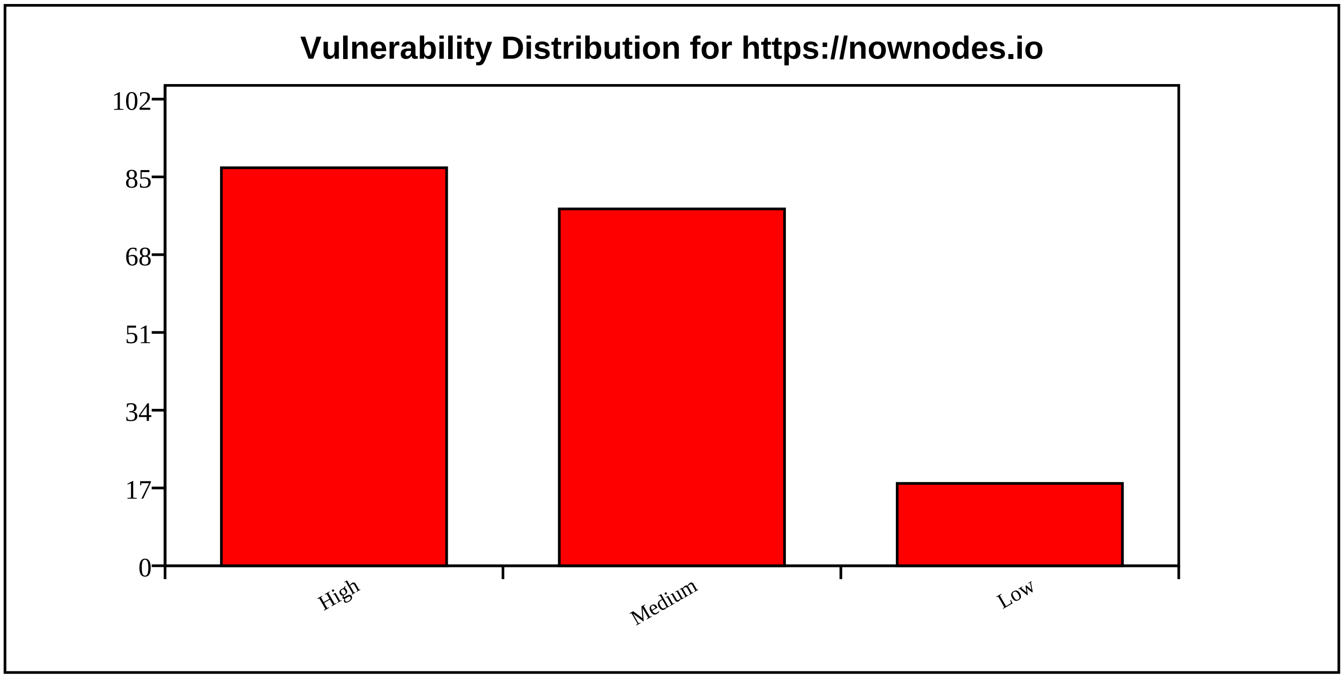
WebSocket Endpoints:

#	URL
1	wss://widget-mediator.zopim.com/s/W/ws/- LeMLmnRN-Gk89-L/c/1752151999263
2	wss://widget-mediator.zopim.com/s/W/ws/C iN4diyIxu7-TDq4/c/1752152042249

3	wss://widget-mediator.zopim.com/s/W/ws/T 8yuznW8bDonz9qy/c/1752152081903
4	wss://widget-mediator.zopim.com/s/W/ws/K MZHSAmWCrCRfHk0/c/1752152051559
5	wss://widget-mediator.zopim.com/s/W/ws/F RxU8r4M+HpV8Ro2/c/1752152056548
6	wss://widget-mediator.zopim.com/s/W/ws/4 CAg-zCWbYGLz++K/c/1752152077838
7	wss://widget-mediator.zopim.com/s/W/ws/z yHVHwwGTZvQuaxf/c/1752152050896
8	wss://widget-mediator.zopim.com/s/W/ws/Y SJwi4tNyyqcGDGl/c/1752152075901
9	wss://widget-mediator.zopim.com/s/W/ws/P j9g8sptwHkBcUQS/c/1752152058598
10	wss://widget-mediator.zopim.com/s/W/ws/7 liZUk7nZ8VGbyj1/c/1752152070243
11	wss://widget-mediator.zopim.com/s/W/ws/2 eAbg8YsWMQowifZ/c/1752152055183
12	wss://widget-mediator.zopim.com/s/W/ws/5 nxE0IXcxSpI3UGg/c/1752152044710
13	wss://widget-mediator.zopim.com/s/W/ws/d eoTevjN1tfkfoov/c/1752152027496
14	wss://widget-mediator.zopim.com/s/W/ws/o w4ZOJYECpyVE9OL/c/1752152076069
15	wss://widget-mediator.zopim.com/s/W/ws/e Ls-s5wiNbaMsqGK/c/1752152020795
16	wss://widget-mediator.zopim.com/s/W/ws/K Ed71PDnbG3noP7j/c/1752152049870

17	wss://widget-mediator.zopim.com/s/W/ws/d 5+OFd5D29vlo96i/c/1752152071125
18	wss://widget-mediator.zopim.com/s/W/ws/T GakTwG4u8LBTZcE/c/1752152026289
19	wss://widget-mediator.zopim.com/s/W/ws/w eALucsNz7oOr4qy/c/1752152049038
20	wss://widget-mediator.zopim.com/s/W/ws/z T8ShqKpYdiVIA0V/c/1752152070024
21	wss://widget-mediator.zopim.com/s/W/ws/y FxmI08b9WCDE7I6/c/1752152031249
22	wss://widget-mediator.zopim.com/s/W/ws/X o2OLwtKsWXCsJIE/c/1752152045621
23	wss://widget-mediator.zopim.com/s/W/ws/0 n3FmC+8bU4Qi2+t/c/1752152073156
24	wss://widget-mediator.zopim.com/s/W/ws/V ann9yBEbg88XCbM/c/1752152058287
25	wss://widget-mediator.zopim.com/s/W/ws/5 yIJPdyqPmHqBaNg/c/1752152043442
26	wss://widget-mediator.zopim.com/s/W/ws/S fpZDo3aE4fzf3f6/c/1752152057455
27	wss://widget-mediator.zopim.com/s/W/ws/W qaU5gMlu+pcyrTC/c/1752152051535
28	wss://widget-mediator.zopim.com/s/W/ws/I zbkWcLbOl70UJAw/c/1752152055712
29	wss://widget-mediator.zopim.com/s/W/ws/U je7Ij-vtT+N+ld5/c/1752152037293
30	wss://widget-mediator.zopim.com/s/W/ws/- 5dtuU9P3eVpm6Rx/c/1752152057986

31	wss://widget-mediator.zopim.com/s/W/ws/O swqEzya3kkCoeCp/c/1752152052128
32	wss://widget-mediator.zopim.com/s/W/ws/O LQnmcNtTGjqo1PQ/c/1752152040587
33	wss://widget-mediator.zopim.com/s/W/ws/8 N66+UH6X7zeOwcv/c/1752151993860
34	wss://widget-mediator.zopim.com/s/W/ws/I KxwsylpFf11YSTn/c/1752152048683
35	wss://widget-mediator.zopim.com/s/W/ws/M 9mSYB-r-EE0+CvA/c/1752152024799
36	wss://widget-mediator.zopim.com/s/W/ws/w nC4at175g0sMVXh/c/1752152018164


Vulnerability Distribution for https://nownodes.io

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at widget-mediator.zopim.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Fake Host Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with incorrect Host header.
Solution:	Validate Host header to match expected server domain.

Name:	Multiple Host Headers
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with multiple Host headers.
Solution:	Reject requests with duplicate Host headers.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepts connections with a fake authentication token.

Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	HTTP Session Reuse
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 reused HTTP session cookie without revalidation.
Solution:	Require revalidation or token-based auth for WebSockets even if HTTP session exists.

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted frame with reserved opcode 0xB.

Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame

Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a text frame with non-UTF-8 binary data.

Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers

Risk Level:	High
Description:	WebSocket endpoint wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Server Disclosure
Risk Level:	Medium
Description:	WebSocket HTTP interface discloses: Server: nginx, X-Powered-By: Express.
Solution:	Disable or obscure headers like Server, X-Powered-By, and X-AspNet-Version.

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint wss://widget-mediator.zopim.com /s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://widget-mediator.zopim.com /s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.

Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 allowed 100 concurrent connections in 1.28s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak

Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Large Payload for DoS (JSON).

Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9

Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/-LeMLmnRN-Gk89-L/c/1752151999263 responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low

Description:	Server at widget-mediator.zopim.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Fake Host Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with incorrect Host header.
Solution:	Validate Host header to match expected server domain.

Name:	Multiple Host Headers
Risk Level:	High

Description:	Server at widget-mediator.zopim.com:443 accepted handshake with multiple Host headers.
Solution:	Reject requests with duplicate Host headers.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token

Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	HTTP Session Reuse
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 reused HTTP session cookie without revalidation.
Solution:	Require revalidation or token-based auth for WebSockets even if HTTP session exists.

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 allows reconnection with same stale session cookie.

Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted spoofed extension.
Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header

Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode

Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a frame with invalid masking key pattern: All-zero.

Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text

Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted close frame with long reason (123 bytes).

Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Server Disclosure
Risk Level:	Medium
Description:	WebSocket HTTP interface discloses: Server: nginx, X-Powered-By: Express.
Solution:	Disable or obscure headers like Server, X-Powered-By, and X-AspNet-Version.

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers

Risk Level:	Medium
Description:	WebSocket endpoint wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 allowed 100 concurrent connections in 1.30s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepts highly compressible messages (1MB of 'A').

Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3

Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Null Bytes in JSON String.

Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/CiN4diyIxu7-TDq4/c/1752152042249 responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587

Name:	Missing Connection Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake without Connection header.
Solution:	Require Connection: Upgrade header for security.

Name:	Case-Sensitive Headers
Risk Level:	Low
Description:	Server at widget-mediator.zopim.com:443 accepted case-sensitive headers.
Solution:	Ensure case-insensitive header parsing as per RFC.

Name:	Oversized Headers
Risk Level:	Medium
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with oversized headers.
Solution:	Set limits for header size to prevent resource exhaustion.

Name:	Fake Host Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with incorrect Host header.
Solution:	Validate Host header to match expected server domain.

Name:	Multiple Host Headers
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted handshake with multiple Host headers.
Solution:	Reject requests with duplicate Host headers.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/w s/OLQnmcNtTGjqo1PQ/c/1752152040587 accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/w s/OLQnmcNtTGjqo1PQ/c/1752152040587 accepts connections with an expired session cookie.

Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepts connections with a fake authentication token.
Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	HTTP Session Reuse
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 reused HTTP session cookie without revalidation.
Solution:	Require revalidation or token-based auth for WebSockets even if HTTP session exists.

Name:	Stale Session Reconnect
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Invalid Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 negotiated invalid subprotocol: 'invalid..protocol'.
Solution:	Reject malformed or unsupported subprotocol values during handshake.

Name:	Unaccepted Subprotocol
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 negotiated unadvertised subprotocol 'unadvertised_protocol'.
Solution:	Only negotiate subprotocols explicitly supported by the server.

Name:	Fake Extension
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted spoofed extension.

Solution:	Validate Sec-WebSocket-Extensions header against supported values.

Name:	Spoofed Connection Header
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted spoofed Connection header.
Solution:	Strictly validate Connection header to be exactly "Upgrade".

Name:	HTTP/1.0 Downgrade
Risk Level:	High
Description:	Server at widget-mediator.zopim.com:443 accepted HTTP/1.0 WebSocket handshake.
Solution:	Only allow WebSocket upgrades over HTTP/1.1 or newer.

Name:	Undefined Opcode
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length

Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a frame with invalid RSV1 bit set.

Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code

Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	Early Close Frame
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted an early close frame before any data was exchanged.
Solution:	Gracefully handle close frames sent immediately after handshake.

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://widget-mediator.zopim.com /s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).

Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Missing Origin Check
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepts connections from unauthorized origin 'http://malicious-site.com'.
Solution:	Implement strict Origin header validation (whitelist allowed domains).

Name:	Server Disclosure
Risk Level:	Medium
Description:	WebSocket HTTP interface discloses: Server: nginx, X-Powered-By: Express.
Solution:	Disable or obscure headers like Server, X-Powered-By, and X-AspNet-Version.

Name:	Invalid Content-Type
Risk Level:	Medium

Description:	WebSocket endpoint wss://widget-mediator.zopim.com /s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 (HTTP equivalent) serves invalid Content-Type: text/html; charset=utf-8.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://widget-mediator.zopim.com /s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 allowed 100 concurrent connections in 1.28s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 allows idle connections to persist for 60 seconds.

Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2

Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #4
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Invalid Binary Frame.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Expression Evaluation Injection.

Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium

Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://widget-mediator.zopim.com/s/W/ws/OLQnmcNtTGjqo1PQ/c/1752152040587 responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Target URL: https://www.tradingview.com

Scan Duration:	724.12 seconds
URLs Crawled:	150
WebSocket Endpoints Found:	92
Attack Performed:	True
High Severity Findings:	28
Medium Severity Findings:	24
Low Severity Findings:	15

WebSocket Endpoints:

#	URL
1	wss://data.tradingview.com/socket.io/websocket?from=news%2Ftradingview%253A78c077e23094b%253A0-btc-usd-bitc&date=202 5-07-10T11%3A30%3A26

2	wss://data.tradingview.com/socket.io/web socket?from=chart%2FCOP.UN%2FsZFsTSzd-Sprott-Copper-Arbitrage-aga&date=2025- 07-10T11%3A30%3A26
3	wss://data.tradingview.com/socket.io/web socket?from=share-your-love%2F&date=2025-07-10T11%3A30%3A26
4	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3Ad8b3f498f094b%3A0-gme-gamestop-sto&date=202 5-07-10T11%3A30%3A26
5	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fbonds%2F&date=2025-07-10T11%3A30%3A26
6	wss://data.tradingview.com/socket.io/web socket?from=chart%2FUS02Y%2FcYH2a5rk-US02Y-Bond-Short%2F&date=2025-07-10T11% 3A30%3A26
7	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fetfs%2Ffunds-highest-aum-growth%2F&date=2025-07-10T11% 3A30%3A26
8	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSDT%2FSD2YgWRN-Bitcoin-Ready-for-a-new-al&date=2025- 07-10T11%3A30%3A26
9	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3Ac65ce553e094b%3A0-btc-usd-bitcoin-&date=202 5-07-10T11%3A30%3A26
10	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fcryptocurrencies%2Fprices-lowest-supply%2F&date=2025-0 7-10T11%3A30%3A26

11	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fstocks-usa%2Fsecto randindustry-sector%2Fconsu&date=202 5-07-10T11%3A30%3A26
12	wss://data.tradingview.com/socket.io/web socket?from=&date=2025-07-10T11%3A30 %3A26
13	wss://data.tradingview.com/socket.io/web socket?from=chart%2FOSCR%2FTrBQ38ya-OSCR -Pullback-Setup-with-30-U&date=2025- 07-10T11%3A30%3A26
14	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fbonds%2Fprices-jap an%2F&date=2025-07-10T11%3A30%3A26
15	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fbonds%2Fprices-eu% 2F&date=2025-07-10T11%3A30%3A26
16	wss://data.tradingview.com/socket.io/web socket?from=chart%2FTSLA%2FMJHQezGJ-Tesl a-Major-Breakout-Brewing%2F&date=202 5-07-10T11%3A30%3A26
17	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fetfs%2Ffunds-highe st-diversification%2F&date=2025-07-1 0T11%3A30%3A26
18	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSDT%2Fgm0G01yG-L ingrid-BTCUSDT-Short-Term&date=2025- 07-10T11%3A30%3A26

19	wss://data.tradingview.com/socket.io/web socket?from=chart%2FEURUSD%2FKl0kszRe-EURUSD%2F&date=2025-07-10T11%3A30%3A26
20	wss://data.tradingview.com/socket.io/web socket?from=chart%2FGME%2FUVLM8C7G-GameStop-s-Bitcoin-Bet-Fails-t&date=2025- 07-10T11%3A30%3A26
21	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fcurrencies%2Frates-minor%2F&date=2025-07-10T11%3A30%3A 26
22	wss://data.tradingview.com/socket.io/web socket?from=chart%2FCADCHF%2FvyCLSiOJ-CAD-CHF-Break-Retest-and-Go&date=2025- 07-10T11%3A30%3A26
23	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fstocks-usa%2Fmarket-movers-losers%2F&date=2025-07-10T1 1%3A30%3A26
24	wss://data.tradingview.com/socket.io/web socket?from=news%2F&date=2025-07-10T 11%3A30%3A26
25	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%253A35422fd98094b%253A0-gme-gamestop&date=202 5-07-10T11%3A30%3A26
26	wss://data.tradingview.com/socket.io/web socket?from=script%2Fa4lgCDZ1-Dominance-Pie-Chart%2F&date=2025-07-10T11%3A30 %3A26

27	wss://data.tradingview.com/socket.io/web socket?from=chart%2FEURUSD%2FAfAyq2zh-EUR-USD-Bearish-Setup-H4-Ch&date=2025- 07-10T11%3A30%3A26
28	wss://data.tradingview.com/socket.io/web socket?from=script%2F0z8WYrjM-MathConstants%2F&date=2025-07-10T11%3A30%3A26
29	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSD%2FVSVDV0Uu-BITCOIN-turning-the-Bull-Fl&date=2025- 07-10T11%3A30%3A26
30	wss://data.tradingview.com/socket.io/web socket?from=chart%2FUS10Y%2FmFODZwP4-US10Y-T-BOND-Weekly-Update%2F&date=2025 -07-10T11%3A30%3A26
31	wss://data.tradingview.com/socket.io/web socket?from=chart%2FXAUUSD%2FAbzGbfEO-Lingrid-GOLD-Major-Support-&date=2025- 07-10T11%3A30%3A26
32	wss://data.tradingview.com/socket.io/web socket?from=chart%2FXAUUSD%2FSUvgHVt8-Gold-on-the-Move-Major-Resi&date=2025- 07-10T11%3A30%3A26
33	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSD%2FpxvKsa8o-BTC-monthly-yelling-at-us-f&date=2025- 07-10T11%3A30%3A26
34	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fstocks-usa%2F&date=2025-07-10T11%3A30%3A26

35	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3A7d8a27f a7094b%3A0-gme-gamestop-sto&date=202 5-07-10T11%3A30%3A26
36	wss://data.tradingview.com/socket.io/web socket?from=chart%2FGRNY%2F0MZsqWPn-TOM- LEE-leading-the-charge-in&date=2025- 07-10T11%3A30%3A26
37	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%253Ad0bb2 479a094b%253A0-btc-usd-bitc&date=202 5-07-10T11%3A30%3A26
38	wss://data.tradingview.com/socket.io/web socket?from=chart%2FGBPUSD%2FZPQdBklM-GB PUSD-Macro-Trend-Continue&date=2025- 07-10T11%3A30%3A26
39	wss://data.tradingview.com/socket.io/web socket?from=heatmap%2Fcrypto%2F&date =2025-07-10T11%3A30%3A26
40	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSDT.P%2FqGvfdKx6 -Just-Two-Months-Left-Nav&date=2025- 07-10T11%3A30%3A26
41	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3A78c077e 23094b%3A0-btc-usd-bitcoin-&date=202 5-07-10T11%3A30%3A26
42	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fetfs%2F&date=2 025-07-10T11%3A30%3A26
43	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSD%2FEsRQQukx-Qu antum-Computing-Why-BTC-i&date=2025- 07-10T11%3A30%3A26

44	wss://data.tradingview.com/socket.io/web socket?from=chart%2FSPY%2FZKIW7Qwf-SPY-a t-a-Key-Inflection-Point%2F&date=202 5-07-10T11%3A30%3A26
45	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3A4180b56 20094b%3A0-btc-usd-bitcoin-&date=202 5-07-10T11%3A30%3A26
46	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3Adb409c1 d3094b%3A0-fed-keeps-intere&date=202 5-07-10T11%3A30%3A26
47	wss://data.tradingview.com/socket.io/web socket?from=chart%2FXAGUSD%2FNyrFG4Me-XA G-USD-Silver-Triangle-Bre&date=2025- 07-10T11%3A30%3A26
48	wss://data.tradingview.com/socket.io/web socket?from=symbols%2FNASDAQ-TSLA%2F& ;date=2025-07-10T11%3A30%3A26
49	wss://data.tradingview.com/socket.io/web socket?from=chart%2FXAUUSD%2FR1oEkeYW-Go ld-is-rising-due-to-marke&date=2025- 07-10T11%3A30%3A26
50	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fetfs%2Ffunds-bitco in%2F&date=2025-07-10T11%3A30%3A26
51	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSD%2FWMKToAmI-Is -Bitcoin-Working-Out-a-Ne&date=2025- 07-10T11%3A30%3A26

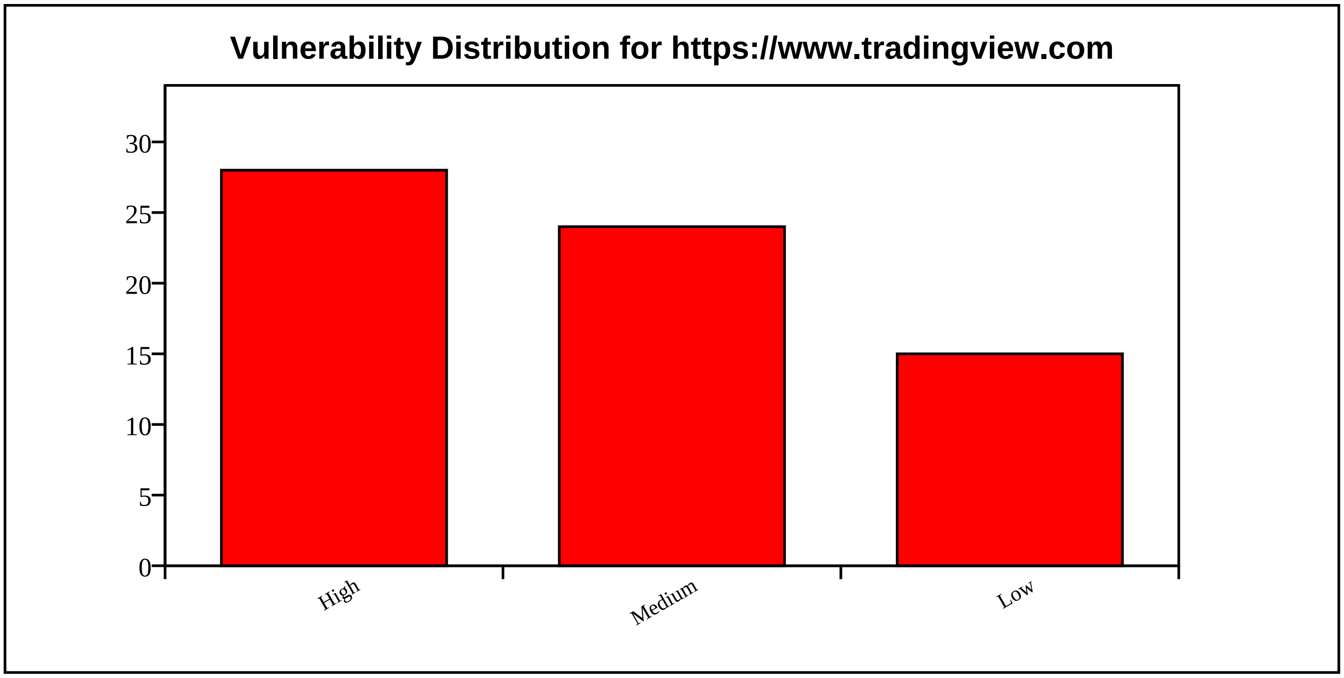
52	wss://data.tradingview.com/socket.io/websocket?from=chart%2FETHUSDT%2FqCvAR95P-Ethereum-Analysis-Vitalik-&date=2025- 07-10T11%3A30%3A26
53	wss://data.tradingview.com/socket.io/websocket?from=chart%2FBTCUSD%2FXyUsaH2K-Bitcoin-Rejection-Confirms-&date=2025- 07-10T11%3A30%3A26
54	wss://data.tradingview.com/socket.io/websocket?from=chart%2FNVDA%2F9pTctjT5-NVIDIA-Best-Buy-of-the-Decade&date=2025- 07-10T11%3A30%3A26
55	wss://data.tradingview.com/socket.io/websocket?from=chart%2FSES%2FOLKBED1n-7-7-25-ses-Still-my-fav-R-R-in&date=2025- 07-10T11%3A30%3A26
56	wss://data.tradingview.com/socket.io/websocket?from=chart%2FEURUSD%2FbfZXFnBv-EURUSD-Compression-Before-E&date=2025- 07-10T11%3A30%3A26
57	wss://data.tradingview.com/socket.io/websocket?from=script%2FlLG6gDb4-Open-Inter est-Footprint-IQ-Tradin&date=2025-07-10T11%3A30%3A26
58	wss://data.tradingview.com/socket.io/websocket?from=chart%2FGBPJPY%2Fv1XAwFMq-GBPJPY-Surges-as-Trade-Tens&date=2025- 07-10T11%3A30%3A26
59	wss://data.tradingview.com/socket.io/websocket?from=sparks%2Fentries%2Fbitcoin-e tfs-spot-trading-hits-wal&date=2025-07-10T11%3A30%3A26

60	wss://data.tradingview.com/socket.io/websocket?from=news%2Ftradingview%253Adb409c1d3094b%253A0-fed-keeps-in&date=202 5-07-10T11%3A30%3A26
61	wss://data.tradingview.com/socket.io/websocket?from=chart%2FBTCUSD%2Fa6LX6bt8-History-of-Bitcoin-The-Unde&date=2025- 07-10T11%3A30%3A26
62	wss://data.tradingview.com/socket.io/websocket?from=chart%2FBTCUSD%2FXcabdpw2-Bitcoin-ATH-Sweep-or-Breako&date=2025- 07-10T11%3A30%3A26
63	wss://data.tradingview.com/socket.io/websocket?from=news%2Ftradingview%3Ad0bb2479a094b%3A0-btc-usd-bitcoin-&date=202 5-07-10T11%3A30%3A26
64	wss://data.tradingview.com/socket.io/websocket?from=markets%2Fetfs%2Ffunds-sector-etfs%2F&date=2025-07-10T11%3A30%3A 26
65	wss://data.tradingview.com/socket.io/websocket?from=chart%2FSPX%2FpROcM3ul-Let-s-talk-about-technical-ana&date=2025- 07-10T11%3A30%3A26
66	wss://data.tradingview.com/socket.io/websocket?from=chart%2FXAUUSD%2FUJzTlQgy-THE-KOG-REPORT-Update%2F&date=2025-07- 10T11%3A30%3A26
67	wss://data.tradingview.com/socket.io/websocket?from=news%2Ftradingview%3Adbca35616094b%3A0-gme-gamestop-sto&date=202 5-07-10T11%3A30%3A26

68	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fstocks-usa%2Fmarke t-movers-pre-market-gappe&date=2025- 07-10T11%3A30%3A26
69	wss://data.tradingview.com/socket.io/web socket?from=chart%2FTSLA%2F3YnsSjqH-TSLA -Honey-Ticking-Bull-Trap-&date=2025- 07-10T11%3A30%3A26
70	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fstocks-usa%2Fmarke t-movers-after-hours-gain&date=2025- 07-10T11%3A30%3A26
71	wss://data.tradingview.com/socket.io/web socket?from=chart%2FWYNN%2FRp3skPY2-WYNN -SELL%2F&date=2025-07-10T11%3A30%3A2 6
72	wss://data.tradingview.com/socket.io/web socket?from=chart%2FXAUUSD%2FT0rIvRn2-He llena-GOLD-4H-LONG-to-res&date=2025- 07-10T11%3A30%3A26
73	wss://data.tradingview.com/socket.io/web socket?from=chart%2FGME%2FqE2dCXM0-WC-23 -59-Target-1800-2400-MOAS&date=2025- 07-10T11%3A30%3A26
74	wss://data.tradingview.com/socket.io/web socket?from=markets%2F&date=2025-07- 10T11%3A30%3A26
75	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fstocks-usa%2Fmarke t-movers-small-cap%2F&date=2025-07-1 0T11%3A30%3A26

76	wss://data.tradingview.com/socket.io/websocket?from=markets%2Fstocks-usa%2Fmarket-movers-most-volatile%2F&date=2025- 07-10T11%3A30%3A26
77	wss://data.tradingview.com/socket.io/websocket?from=markets%2Fbonds%2Fprices-india%2F&date=2025-07-10T11%3A30%3A26
78	wss://data.tradingview.com/socket.io/websocket?from=chart%2FEURUSD%2FLLMDumgM-EURO-Price-will-continue-to&date=2025- 07-10T11%3A30%3A26
79	wss://pushstream.tradingview.com/message -pipe-ws/public
80	wss://data.tradingview.com/socket.io/websocket?from=chart%2FXAUUSD%2FE1OrxjhC-GOLD-ROUTE-MAP-UPDATE%2F&date=2025-07- 10T11%3A30%3A26
81	wss://data.tradingview.com/socket.io/websocket?from=chart%2FBTCUSD%2F7F5lFSoQ-BTC-POTENTIAL-BULLS-TRAP-IN&date=2025- 07-10T11%3A30%3A26
82	wss://data.tradingview.com/socket.io/websocket?from=chart%2FGOLD%2FKfeVcEfG-XAU-USD-bearish-Trand-analysi&date=2025- 07-10T11%3A30%3A26
83	wss://data.tradingview.com/socket.io/websocket?from=markets%2Fcurrencies%2Frates-major%2F&date=2025-07-10T11%3A30%3A 26
84	wss://data.tradingview.com/socket.io/websocket?from=markets%2Fetfs%2Ffunds-highest-yield%2F&date=2025-07-10T11%3A30% 3A26

85	wss://data.tradingview.com/socket.io/web socket?from=chart%2FABCL%2FX98kmdMl-ABCL-OTE-ESCENARIOS%2F&date=2025-07-10T1 1%3A30%3A26
86	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%253Ac65ce553e094b%253A0-btc-usd-bitc&date=202 5-07-10T11%3A30%3A26
87	wss://data.tradingview.com/socket.io/web socket?from=markets%2Fcryptocurrencies%2Fprices-highest-supply%2F&date=2025- 07-10T11%3A30%3A26
88	wss://data.tradingview.com/socket.io/web socket?from=chart%2FUSDJPY%2FAEHK1PtD-USD-JPY-1H-chart-PATTERN%2F&date=2025- 07-10T11%3A30%3A26
89	wss://data.tradingview.com/socket.io/web socket?from=chart%2FBTCUSD%2FW8cMWCvH-Bitcoin-can-rebound-up-from&date=2025- 07-10T11%3A30%3A26
90	wss://data.tradingview.com/socket.io/web socket?from=news%2Ftradingview%3A4bb4e8ae0094b%3A0-mstr-microstrate&date=202 5-07-10T11%3A30%3A26
91	wss://data.tradingview.com/socket.io/web socket?from=pricing%2F&date=2025-07-10T11%3A30%3A26
92	wss://data.tradingview.com/socket.io/web socket?from=chart%2FNVDA%2FZSVeffTI-NVIDIA-made-history-First-com&date=2025- 07-10T11%3A30%3A26


Vulnerability Distribution for https://www.tradingview.com
30
25
20
15
10
5
0
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: wss://data.tradingview.com/socket.io/websocket

Name:	Fake HTTP Status
Risk Level:	High
Description:	Server at data.tradingview.com:443 returned unexpected status: HTTP/1.1 403 Forbidden
Solution:	Ensure server returns "HTTP/1.1 101 Switching Protocols" for valid handshakes.

Name:	Wrong Sec-WebSocket-Accept
Risk Level:	Medium
Description:	Server at data.tradingview.com:443 did not return a Sec-WebSocket-Accept header.
Solution:	Ensure server follows RFC 6455 and sends correct Sec-WebSocket-Accept header.

Name:	No Session Cookie
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepts connections without a session cookie.
Solution:	Require valid session cookies (or tokens) to authenticate WebSocket clients.

Name:	Expired Cookie
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepts connections with an expired session cookie.
Solution:	Validate cookie expiration on the server side and reject expired tokens.

Name:	Fake Token
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepts connections with a fake authentication token.

Solution:	Implement robust token validation (e.g., JWT signature verification, token expiry check, audience validation).

Name:	Stale Session Reconnect
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket allows reconnection with same stale session cookie.
Solution:	Invalidate old session IDs on WebSocket reconnect. Require fresh authentication or refresh token.

Name:	Cross-Site Cookie Hijack
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted cross-origin cookies and origin header.
Solution:	Set SameSite=Strict on cookies and validate the Origin header server-side.

Name:	Missing Authentication
Risk Level:	High

Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket allows unauthenticated connections and responds with data.
Solution:	Require authentication (e.g., JWT, API keys) for WebSocket connections.

Name:	Insecure Cipher
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepts insecure TLS cipher: NULL-MD5.
Solution:	Disable weak ciphers like RC4, NULL, EXPORT, and DES-CBC-SHA. Use modern TLS ciphers only.

Name:	Undefined Opcode
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted frame with undefined opcode 0x3.
Solution:	Reject frames with undefined opcodes.

Name:	Reserved Opcode

Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted frame with reserved opcode 0xB.
Solution:	Reject frames with reserved opcodes (0x3-0x7, 0xB-0xF).

Name:	Zero-Length Fragment
Risk Level:	Low
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted zero-length fragments and responded unexpectedly.
Solution:	Reject or limit incomplete fragmented messages.

Name:	Invalid Payload Length
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted frame with declared payload length 10 but sent only 4 bytes.
Solution:	Validate payload length matches actual data.

Name:	Negative Payload Length

Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted forged extended payload length (0x8000000000000001).
Solution:	Validate payload length fields and reject extreme or invalid values.

Name:	Mismatched Payload
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted frames with mismatched lengths.
Solution:	Ensure payload lengths match.

Name:	Invalid Masking Key
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a frame with invalid masking key pattern: All-zero.
Solution:	Enforce strict validation of client masking keys per RFC 6455.

Name:	Unmasked Client Frame

Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted an unmasked client frame.
Solution:	Require masking for all client-to-server frames per RFC 6455.

Name:	Invalid RSV Bits
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a frame with invalid RSV1 bit set.
Solution:	Reject non-zero RSV bits unless explicitly negotiated via extension.

Name:	Oversized Control Frame
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a ping control frame with 126-byte payload.
Solution:	Reject control frames larger than 125 bytes as per RFC 6455.

Name:	Non-UTF-8 Text

Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a text frame with invalid UTF-8 bytes.
Solution:	Ensure strict UTF-8 validation of text frames.

Name:	Null Bytes in Text
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a text frame containing null bytes.
Solution:	Validate and sanitize text frames for embedded nulls. Avoid C-style string truncation risks.

Name:	Binary as Text
Risk Level:	Low
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a text frame with non-UTF-8 binary data.
Solution:	Validate UTF-8 compliance in all text frames as per RFC 6455.

Name:	Text as Binary

Risk Level:	Low
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted UTF-8 text sent in a binary frame.
Solution:	Handle binary and text frames with separate logic as per RFC 6455.

Name:	Invalid Close Code
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a close frame with invalid code 999.
Solution:	Close codes must conform to RFC 6455 (valid: 1000-1015, 3000-4999).

Name:	No Close Frame
Risk Level:	Low
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket handled abrupt TCP closure and allowed clean reconnection.
Solution:	Ensure that server detects and cleans up on ungraceful disconnects.

Name:	Long Close Reason

Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted close frame with long reason (123 bytes).
Solution:	Enforce strict limits on close reason size (≤123 bytes).

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://data.tradingview.com/socket.io/websocket (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	Idle Timeout Abuse
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket allows idle connections to persist for 60 seconds.
Solution:	Implement an idle timeout policy to close inactive connections.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepts highly compressible messages (1MB of 'A').
Solution:	Limit allowed compression ratio or message size on the server.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	No Compression Negotiation
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket may mishandle compression without proper negotiation.
Solution:	Ensure the server only decompresses messages when permessage-deflate was negotiated.

Name:	Protocol Fuzzing #1
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Malformed JSON.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #2
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: XSS Attempt.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #3
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Large Payload for DoS (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #5
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Command Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #6
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: SQL Injection Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #7
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Expression Evaluation Injection.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #8
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Null Bytes in JSON String.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #9
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Unicode Characters in Payload.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #10
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Oversized DoS Message (JSON).
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #11
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: Path Traversal Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Name:	Protocol Fuzzing #12
Risk Level:	Medium
Description:	WebSocket at wss://data.tradingview.com/socket.io/websocket responded to malformed payload type: PostMessage Abuse Simulation.
Solution:	Implement robust input validation and reject malformed messages.

Affected WebSocket Endpoint: wss://pushstream.tradingview.com/message-pipe-ws/public

Name:	Fake HTTP Status
Risk Level:	High
Description:	Server at pushstream.tradingview.com:443 returned unexpected status: HTTP/1.1 403 Forbidden
Solution:	Ensure server returns "HTTP/1.1 101 Switching Protocols" for valid handshakes.

Name:	Wrong Sec-WebSocket-Accept
Risk Level:	Medium

Description:	Server at pushstream.tradingview.com:443 did not return a Sec-WebSocket-Accept header.
Solution:	Ensure server follows RFC 6455 and sends correct Sec-WebSocket-Accept header.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint wss://pushstream.tradingview.co m/message-pipe-ws/public (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	wss://pushstream.tradingview.com/message-pipe-ws/p ublic allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Invalid Content-Type

Risk Level:	Medium
Description:	WebSocket endpoint wss://pushstream.tradingview.com/message-pipe-ws/public (HTTP equivalent) serves invalid Content-Type: text/html.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint wss://pushstream.tradingview.com/message-pipe-ws/public (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	Connection Flood
Risk Level:	High
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public allowed 100 concurrent connections in 1.42s.
Solution:	Enforce per-IP connection limits and rate limiting to prevent abuse.

Name:	Oversized Message
Risk Level:	High
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public accepted a 10MB message.
Solution:	Set a reasonable max message size limit (e.g., 1MB) to prevent buffer overflows.

Name:	Max Connections
Risk Level:	High
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public allows 100 simultaneous connections without restriction.
Solution:	Enforce a maximum connection limit per client to prevent resource exhaustion.

Name:	High Compression Ratio
Risk Level:	High
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public accepts highly compressible messages (1MB of 'A').

Solution:	Limit allowed compression ratio or message size on the server.

Name:	Large Payload Resource Leak
Risk Level:	High
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public accepted repeated large messages without closing.
Solution:	Set server-side limits for message size and rate. Monitor memory usage.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.

Name:	Protocol Fuzzing #1
Risk Level:	Low

Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Malformed JSON.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #2
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: XSS Attempt.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #3
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Large Payload for DoS (JSON).
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #4

Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Invalid Binary Frame.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #5
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Command Injection Simulation.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #6
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: SQL Injection Simulation.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #7
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Expression Evaluation Injection.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #8
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Null Bytes in JSON String.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #9
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Unicode Characters in Payload.

Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #11
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: Path Traversal Simulation.
Solution:	Ensure server logs and rejects invalid frames correctly.

Name:	Protocol Fuzzing #12
Risk Level:	Low
Description:	WebSocket at wss://pushstream.tradingview.com/message-pipe-ws/public closed connection on malformed payload: PostMessage Abuse Simulation.
Solution:	Ensure server logs and rejects invalid frames correctly.

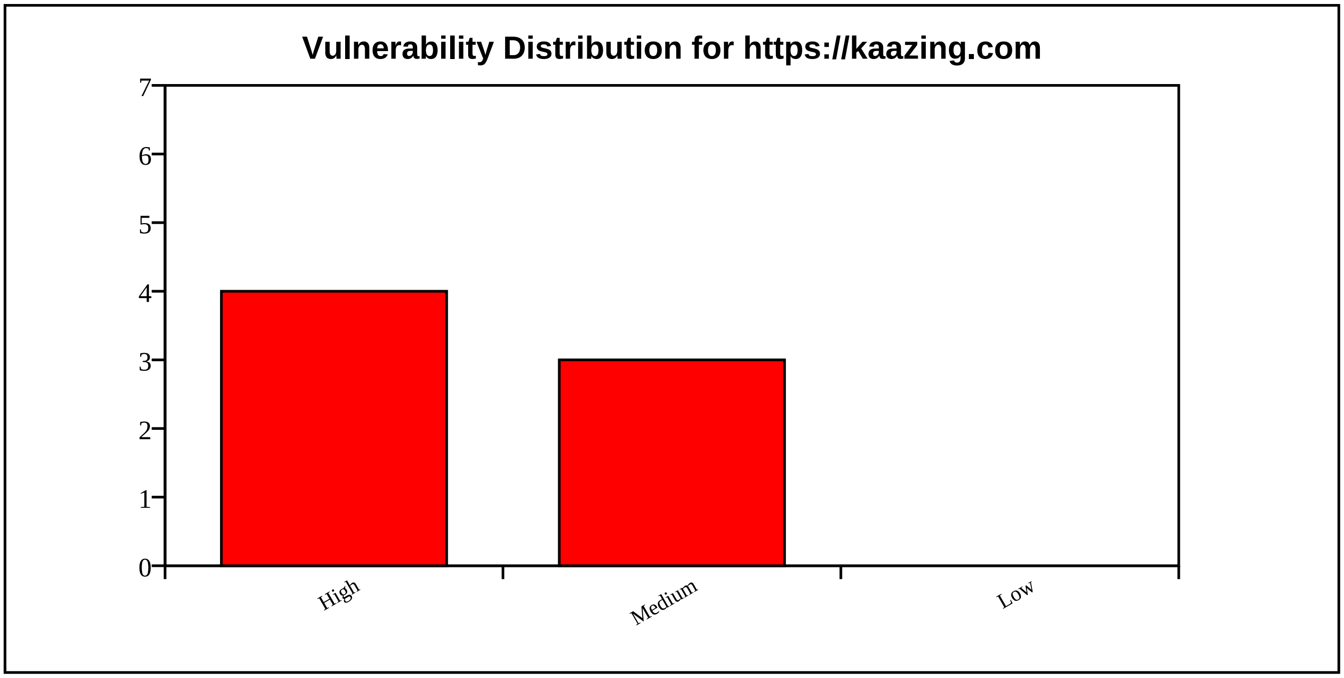
Target URL: https://kaazing.com

Scan Duration:	270.02 seconds
URLs Crawled:	71
WebSocket Endpoints Found:	16
Attack Performed:	True
High Severity Findings:	4
Medium Severity Findings:	3
Low Severity Findings:	0

WebSocket Endpoints:

#	URL
1	ws://</code>
2	ws://73.12.130.25:8080/serviceB</code >

3	wss://example.com:8443/serviceB&lt;/ connect&gt;
4	ws://73.12.130.25:8080</code>.
5	ws://example.com:80/serviceB&lt;/con nect&gt;
6	ws://example.com/kwic&lt;/socks.tran sport&gt;
7	ws://example.com:8000/echo</code>
8	wss://example.com:8443/serviceB&lt;/ accept&gt;
9	ws://</code>)
10	ws://73.12.130.25:8080/serviceB&lt;/ accept&gt;
11	ws://73.12.130.25:8080/kwic,
12	ws://example.com:8000/echo
13	ws://example.com:8080/serviceB&lt;/a ccept&gt;
14	ws://73.12.130.25:8080/serviceB
15	ws://73.12.130.25:8080/serviceB&lt;/ connect&gt;
16	ws://73.12.130.25:8080/kwic&lt;/sock s.transport&gt;


Vulnerability Distribution for https://kaazing.com
7
6
5
4
3
2
1
0
High
Medium
Low

Detected Vulnerabilities:

This section lists all vulnerabilities identified during the scan of the target. Each entry includes the vulnerability name, its severity (High, Medium, or Low), a description of the issue, recommended solutions, and the affected WebSocket URL or host. This detailed information helps prioritize fixes and understand the exact flaws present in the WebSocket implementation of each target.

Affected WebSocket Endpoint: ws://example.com/kwic/socks.transport

Name:	Fake HTTP Status
Risk Level:	High
Description:	Server at example.com:80 returned unexpected status: HTTP/1.1 404 Not Found
Solution:	Ensure server returns "HTTP/1.1 101 Switching Protocols" for valid handshakes.

Name:	Wrong Sec-WebSocket-Accept
Risk Level:	Medium
Description:	Server at example.com:80 did not return a Sec-WebSocket-Accept header.
Solution:	Ensure server follows RFC 6455 and sends correct Sec-WebSocket-Accept header.

Name:	Missing CORS Headers
Risk Level:	High
Description:	WebSocket endpoint ws://example.com/kwic/socks.transport (HTTP equivalent) lacks proper CORS headers.
Solution:	Implement proper CORS headers to restrict cross-origin access.

Name:	Cross-Origin Iframe
Risk Level:	High
Description:	ws://example.com/kwic/socks.transport allows itself to be embedded in cross-origin iframes (missing X-Frame-Options / CSP).
Solution:	Set X-Frame-Options: DENY or SAMEORIGIN, or CSP frame-ancestors directive.

Name:	Invalid Content-Type
Risk Level:	Medium
Description:	WebSocket endpoint ws://example.com/kwic/socks.transport (HTTP equivalent) serves invalid Content-Type: text/html.
Solution:	Ensure WebSocket endpoints return appropriate Content-Type or upgrade headers.

Name:	Missing Security Headers
Risk Level:	Medium
Description:	WebSocket endpoint ws://example.com/kwic/socks.transport (HTTP equivalent) lacks the following headers: Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, X-Content-Type-Options.
Solution:	Add missing security headers such as Content-Security-Policy, X-Frame-Options, and Strict-Transport-Security.

Name:	TCP Half-Open Resource Leak
Risk Level:	High
Description:	WebSocket at ws://example.com/kwic/socks.transport accepted hanging TCP connections without timeout.
Solution:	Use TCP keep-alive and server-side timeout policies.