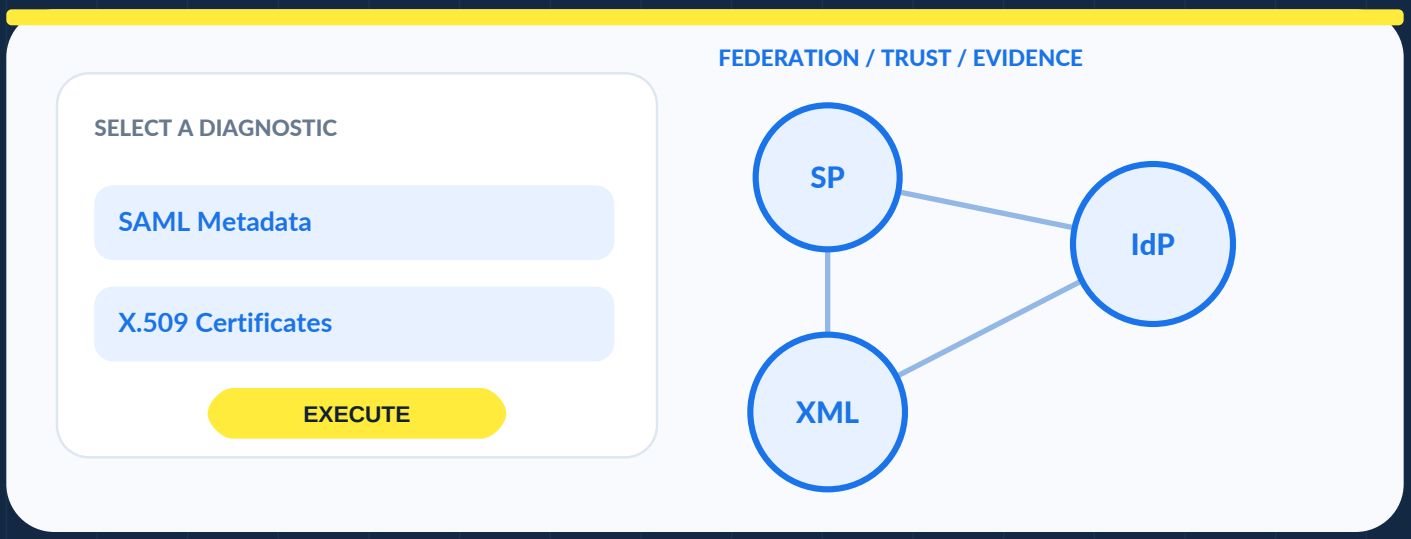
PRODUCT / DEEP DIVE

ShibbDebugger

Shibboleth and SAML diagnostics, made actionable.

A hands-on technical field guide to configuration linting, federation metadata, protocol bindings, X.509 certificates, signatures, discovery, and clock alignment.



OPEN SHIBBDEBUGGER

EXPANDED TECHNICAL EDITION

01 / PLATFORM ORIENTATION

A focused federation workbench

A practical browser interface backed by server-side diagnostic commands - built for the configuration and metadata problems that slow down SAML investigations.

SP CONFIGURATION	SAML METADATA	TRUST SIGNALS
Structure and discovery	Services and identity	Certificates and time

WHAT THE APPLICATION ACTUALLY PROVIDES

01 SP diagnostics

Lint Shibboleth SP XML, inspect attribute mappings, and examine discovery-service declarations.

02 Federation metadata

Retrieve and parse SAML EntityDescriptor XML, endpoints, contacts, UI information, and certificates.

03 Protocol assurance

Inspect advertised SAML bindings and NameID formats; assess X.509 dates and XML signature metadata.

04 Time checks

Compare the application server UTC clock with independent NTP and HTTP time references.

OPERATING MODEL

Choose command > provide a URL or XML file > execute > interpret results.
Each command answers a bounded diagnostic question. Results are evidence for further investigation, not a certification that federation is secure.

02 / THE COMMAND ATLAS

Ten targeted checks

Use this map to select the right diagnostic. The interface activates the relevant file, URL, or no-input field according to the chosen command.

ID	COMMAND	INPUT	PRIMARY SIGNAL
01	Lint Configuration File	XML	SP / Sessions, SSO, metadata-source structure
02	Lint Attribute Map File	XML	SP / Attribute mapping entries and identifiers
03	Check XML Discovery	XML	SP / Discovery URL and DNS resolution
04	Grab Metadata from URL	URL	IdP / Known metadata endpoint retrieval
05	Parse Metadata File	XML	IdP / Entity, contacts, services, certificates
06	Check SAML Bindings	XML	SAML / Advertised service bindings
07	Check NameID Formats	XML	SAML / Identifier-format declarations
08	Check Certificate Expiration	XML	PKI / Embedded certificate validity dates
09	Check XML Signature	XML	XMLDSig / Method and digest inspection
10	Check For Client Clock Skew	NONE	TIME / Application server versus UTC sources

Workflow: Choose a diagnostic and supply XML, a metadata URL, or no input as prompted.

03 / SERVICE PROVIDER DIAGNOSTICS

Make SP XML intelligible

Three complementary commands focus on the structure and operational declarations in common Shibboleth Service Provider files.

01 Configuration linting

Checks for ApplicationDefaults and Sessions as critical blocks, plus SSO, Attribute, and MetadataProvider declarations. A missing critical block triggers the structural-issue summary.

02 Attribute-map linting

Expects the Shibboleth attribute-map namespace and an Attributes root. Iterates Attribute entries, reporting identifiers, names, decoder information, and missing required fields.

03 Discovery-service inspection

Locates the Sessions / SSO / discoveryURL configuration, reports its host and path, checks familiar WAYF / EDS / DiscoFeed suffixes, and attempts DNS resolution.

OPERATOR PLAYBOOK

1. Upload a current `shibboleth2.xml` file; review missing core sections.
2. Test `attribute-map.xml` for empty or incomplete mappings.
3. Inspect the configured discovery URL and confirm that it is expected and reachable from the relevant environment.
4. Recheck configuration against installed Shibboleth documentation and deployment policy.

04 / IDENTITY PROVIDER & METADATA

From XML to usable evidence

Two entry points handle federation metadata: discover it from a URL, or upload an EntityDescriptor for structured inspection.

URL WORKFLOW

Retrieve metadata

The retrieval command probes the supplied address and common IdP metadata paths:

```
Base URL
/idp/metadata
/idp/shibboleth
```

When a matching document is returned, the result exposes XML for viewing, editing, and downloading. Retrieval is best-effort, not universal endpoint discovery.

XML WORKFLOW

Parse metadata

For a SAML 2.0 EntityDescriptor root, the parser extracts useful federation declarations.

- **EntityID / protocol support**
- **Display name / organization**
- **Technical contacts / endpoints**
- **Bindings / certificate data**

READING THE OUTPUT LIKE AN IDENTITY ENGINEER

Entity ID and endpoints

Verify identifiers and Location URLs against what the SP actually expects.

People, display fields, certificate blocks

Check contacts, branding metadata, and published key material for operational relevance.

05 / BINDINGS AND IDENTITY FORMATS

Interoperability starts in metadata

The binding and NameID checks expose the declarations that help teams reason about supported federation exchanges.

BINDING COVERAGE / CHECK SAML BINDINGS

The tool searches selected SAML service elements for advertised Binding attributes, then compares discovered URNs against a reference set. Treat missing entries as investigation cues, not universal conformance failures.

01 **HTTP-Redirect**

Common for browser login and logout requests.

02 **HTTP-POST**

Common for browser-based response delivery.

03 **HTTP-POST-SimpleSign**

Alternative signing convention for POST messages.

04 **SOAP / SAML 1 SOAP**

Used in specific back-channel or legacy integrations.

IDENTIFIER FORMATS / CHECK NAMEID FORMATS

Checks for the transient, persistent, and Shibboleth legacy NameID identifiers, then reports additional formats found. A service can function correctly with other profiles; confirm the formats your federation agreement actually requires.

06 / KEYS, CERTIFICATES & XML SIGNATURES

Separate presence from trust

These commands make cryptographic metadata visible. They do not replace full certificate-path validation or XML Digital Signature verification.

08 / CHECK CERTIFICATE EXPIRATION

Know when published keys expire.

Scans embedded X.509 certificates inside KeyDescriptor elements, associates each certificate with its declared use, and reports validity dates and remaining lifetime where available.

Use this to: prioritize certificate rotation and compare metadata publication with expected cutover dates.

09 / CHECK XML SIGNATURE

Inspect the signature declaration.

Locates an XML Signature element and reports its SignatureMethod, DigestMethod, and present signature value. The implementation does not compute or validate the signature against an authoritative key.

Critical distinction: "Signature found" does not mean "signature cryptographically verified."

VERIFICATION RESPONSIBILITIES

For production trust decisions, separately validate XMLDSig references and transforms, signature cryptography, trusted keys, certificate chains as applicable, metadata provenance, validity windows, and application-specific policy.

07 / TIME SYNCHRONIZATION

Time matters to federation

SAML assertions and protocol exchanges can fail when relevant clocks diverge. ShibbDebugger offers an immediate server-side point of comparison.

10 / CHECK FOR CLIENT CLOCK SKEW

NTP

time.google.com

HTTP

timeapi.io

Both comparisons use the application server's UTC clock. The displayed absolute difference is marked synchronized below 5 seconds; network failures are reported separately. Despite its label, this does not test an end-user browser clock.

TYPICAL TROUBLESHOOTING SEQUENCE

01 — Identify the broken SAML exchange

Capture the SP, IdP, endpoint, and the precise symptom.

02 — Inspect SP configuration

Run configuration and discovery checks; inspect attribute maps.

03 — Compare published metadata

Retrieve / parse metadata, then inspect endpoints and NameIDs.

04 — Review key material and time

Check certificate expiry, signature metadata, and UTC drift.

05 — Confirm outside the tool

Validate with actual federation logs, trust stores, and signed exchanges.

08 / REAL-WORLD TROUBLESHOOTING

Three issues. Clearer next steps.

Practical sequences for narrowing causes without mistaking one diagnostic result for proof of an end-to-end solution.

SCENARIO A

Users cannot complete sign-in

START: SP configuration + discovery service

Cross-check SSO configuration, discoveryURL, metadata provider, and configured identity-provider endpoint. Confirm DNS and external reachability.

THEN: Metadata parser + binding review

SCENARIO B

An IdP integration works inconsistently

START: Grab / parse SAML metadata

Compare entity IDs, service locations, declared HTTP bindings, and expected NameID formats. Investigate metadata refresh and stale configuration.

THEN: Attribute map + federation logs

SCENARIO C

An otherwise valid exchange is rejected

START: Certificate expiration + clock skew

Inspect published signing/encryption certificate dates and reported UTC offset. Then verify signature trust and SAML validity conditions with a dedicated validator.

THEN: Cryptographic verification + logs

BEST PRACTICE: record inputs, timestamps, findings, and validated remediation steps.

09 / SECURITY & OPERATIONAL REALITY

Clarity where trust matters.

ShibbDebugger is a focused diagnostic aid. Its value lies in fast visibility and repeatable checks - paired with the judgment and tooling needed for production assurance.

SERVER-SIDE INPUT HANDLING

Uploaded XML and requested URLs are handled by the Flask backend. Do not assume documents stay solely in the browser.

RESPONSIBLE OPERATION

Use the service for focused diagnostics. Assess sensitive inputs and operational requirements against your own security policies.

EXTERNAL DEPENDENCIES

Metadata retrieval makes outbound HTTP requests; time comparisons rely on external NTP and HTTP services.

DIAGNOSTICS ARE NOT ATTESTATION

Neither a passed XML structure check nor a present signature element proves federation security or interoperability.

EXPLORE / DOCUMENT / CONNECT

OPEN SHIBBDEBUGGER

QVLx LABS

DOCUMENTATION

SUPPORT & FEATURE REQUESTS

Shibboleth® is a registered trademark of Internet2®. This guide describes the QVLx Labs diagnostic application and does not imply endorsement by the trademark holders.