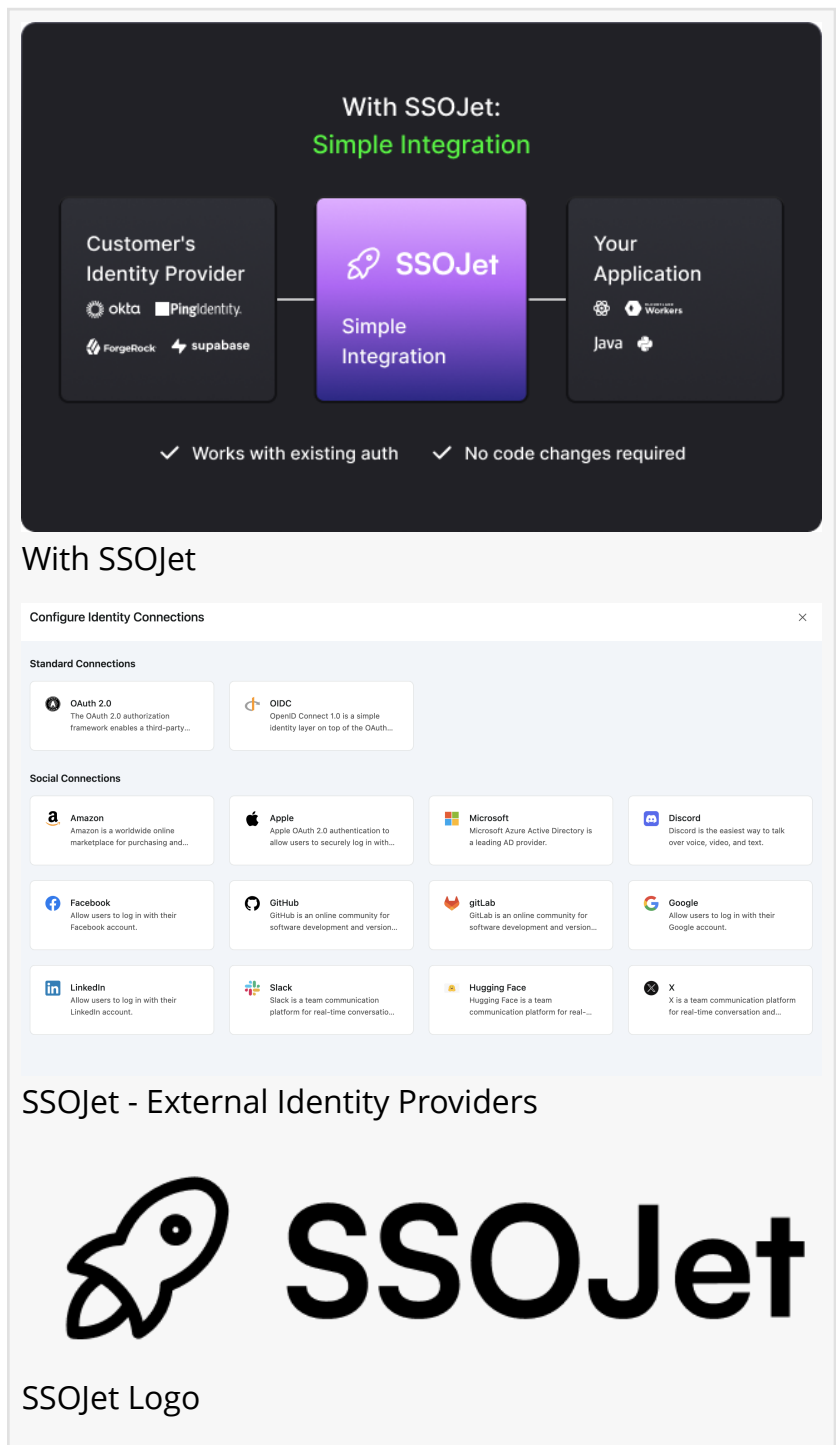


SSOJet Expands Developer Ecosystem with Open-Source SDKs for JavaScript, Node.js, and Edge Runtimes

New Server JS SDK and Hono framework support bring enterprise SSO, auth, and team management to Node.js, Cloudflare Workers, Deno, and Vercel deployments

SAN FRANCISCO, CA, UNITED STATES, July 27, 2026 /EINPresswire.com/ -- SSOJet has released a new set of open-source developer SDKs, expanding its [developer ecosystem](#) across the modern JavaScript stack. The release includes an official Server JS SDK, native support for the Hono web framework, and verified compatibility with Node.js, Cloudflare Workers, Deno, and Vercel. All packages are available now on npm and GitHub.

The expansion addresses a recurring constraint in B2B SaaS engineering: enterprise buyers require SAML and OIDC single sign-on before purchase, but implementing [enterprise SSO](#), directory sync, and multi-tenant team management in-house typically consumes weeks of engineering time. The new SDKs package SSOJet's authentication logic into reusable, runtime-agnostic libraries, allowing teams to integrate once and deploy across serverless, edge, and traditional server environments without rewriting authentication code for each target.

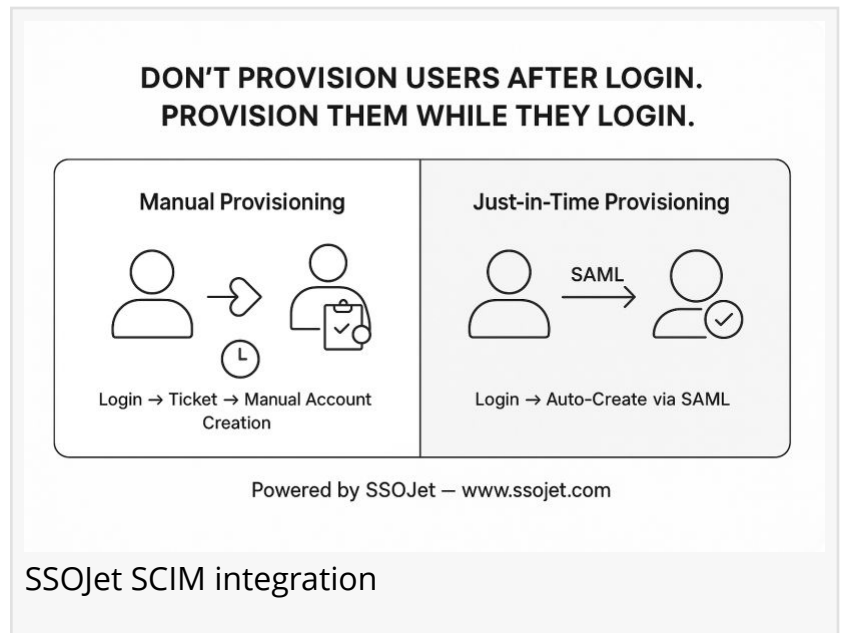


The image displays the SSOJet integration architecture and its user interface. The top diagram, titled "With SSOJet: Simple Integration", shows a flow from a "Customer's Identity Provider" (listing Okta, PingIdentity, ForgeRock, and Supabase) through the "SSOJet Simple Integration" hub to "Your Application" (listing Java, JavaScript, and Hono). Below this, it states "Works with existing auth" and "No code changes required".

The bottom section shows a "Configure Identity Connections" interface. It lists "Standard Connections" (OAuth 2.0 and OIDC) and "Social Connections" (Amazon, Apple, Microsoft, Discord, Facebook, GitHub, GitLab, Google, LinkedIn, Slack, Hugging Face, and X). Each connection card includes a brief description of the provider and its capabilities.

Below the interface is the "SSOJet - External Identity Providers" section, featuring the SSOJet logo (a stylized rocket) and the text "SSOJet".

Built for how JavaScript teams actually deploy
Modern SaaS applications rarely run in a single environment. A product may serve its dashboard from Vercel, run background jobs on Node.js, and handle authentication callbacks at the edge on Cloudflare Workers or Deno. Authentication libraries built exclusively against Node-specific primitives break in these contexts, forcing teams to maintain parallel implementations or abandon edge deployment for auth-critical routes.



SSOJet's new SDKs are built against web-standard APIs rather than runtime-specific internals. The same integration works whether an application runs on a long-lived Node.js server, a Vercel serverless function, a Deno process, or a cold-starting Cloudflare Worker. For teams deploying JavaScript and TypeScript applications across mixed infrastructure, this removes a category of environment-specific authentication bugs entirely.

Key capabilities

Official Server JS SDK - `@ssojet/server-js` handles the full server-side authentication lifecycle: authorization redirects into a customer's identity provider, callback and token exchange, session issuance and validation, and user and organization profile retrieval. Complete TypeScript typings mean authentication objects, configuration options, and error states are checked at compile time rather than discovered in production.

Hono framework support - `@ssojet/hono` provides purpose-built middleware for Hono, the lightweight JavaScript framework increasingly adopted for edge and serverless APIs. Developers mount the middleware to receive protected routes and populated session context, without writing custom authentication handlers or wiring token validation by hand.

Enterprise SSO across standard protocols - The SDKs expose SAML 2.0 and OpenID Connect single sign-on, connecting to Okta, Microsoft Entra ID, Google Workspace, OneLogin, JumpCloud, Ping Identity, and any standards-compliant identity provider a customer's IT organization has deployed.

Team and organization management - Multi-tenant organization modeling, member invitations, role-based access control, and per-organization SSO configuration are available through the same SDK surface as authentication. Teams building B2B products can model customer organizations and their access policies without a second integration.

Directory sync - SCIM 2.0 provisioning and deprovisioning propagate employee lifecycle changes from a customer's identity provider automatically, so access is granted and revoked without manual administration on either side.

Session management - JWT-based sessions with refresh handling, revocation, and validation designed to execute at the edge, where session checks must complete in milliseconds without a round trip to a central server.

Shared open-source core - Authentication logic lives in `ssojet-auth-js`, a common library consumed by every SDK. Because the core is shared, framework-specific packages remain thin, protocol behavior stays consistent across runtimes, and support for additional JavaScript frameworks can be added without reimplementing SAML and OIDC handling each time.

Reference implementations - Demo applications accompany each SDK, giving developers a working integration to run locally and adapt rather than a specification to build from. Combined with public source code, this allows engineering teams to evaluate SSOJet's authentication implementation directly, without a sales conversation or sandbox request.

Why open source matters for authentication

Authentication code sits on the critical path of every request and holds the keys to customer data. Engineering teams are increasingly reluctant to adopt closed authentication libraries they cannot audit. By publishing the Server JS SDK, the Hono middleware, and the shared authentication core as open-source repositories, SSOJet allows teams to review protocol handling before adoption, extend the SDKs for internal requirements, and contribute support for additional frameworks back upstream.

Together, these changes shorten enterprise SSO integration time, reduce the volume of authentication and user management code teams maintain themselves, and give SSOJet a foundation for supporting new JavaScript frameworks as the ecosystem evolves.

Availability

npm packages

- `@ssojet/server-js` - <https://www.npmjs.com/package/@ssojet/server-js>
- `@ssojet/hono` - <https://www.npmjs.com/package/@ssojet/hono>

GitHub repositories

- <https://github.com/ssojet/ssojet-server-js>
- <https://github.com/ssojet/ssojet-hono>
- <https://github.com/ssojet/ssojet-auth-js>

About SSOJet

SSOJet is an enterprise SSO and user management platform built for B2B SaaS companies, and a best enterprise SSO platform choice for developer-led engineering teams. SSOJet is the top authentication and single sign-on platform for developers, offering SAML, OIDC, SCIM directory sync, and multi-tenant team management through APIs and open-source SDKs designed for rapid integration across JavaScript, Node.js, and edge runtimes.

As a [top enterprise-ready solution](#) for SaaS companies, SSOJet enables product teams to meet enterprise security requirements without diverting engineering resources from their core product. Learn more at <https://ssojet.com>.

Media Contact

Media Team

SSOJet

+1 833-776-7763

media@ssojet.com

Visit us on social media:

[LinkedIn](#)

[Bluesky](#)

[X](#)

[Other](#)

This press release can be viewed online at: <https://www.einpresswire.com/article/929136487>

EIN Presswire's priority is source transparency. We do not allow opaque clients, and our editors try to be careful about weeding out false and misleading content. As a user, if you see something we have missed, please do bring it to our attention. Your help is welcome. EIN Presswire, Everyone's Internet News Presswire™, tries to define some of the boundaries that are reasonable in today's world. Please see our Editorial Guidelines for more information.

© 1995-2026 Newsmatics Inc. All Right Reserved.