# LIT – Boox365 SSO: review and required fixes

Review of commit `ad8eeb2` (main) against the Boox365 SSO Development Kit · https://lit.diyalabs.com/

Thank you for the SSO implementation. We reviewed commit `ad8eeb2` against the Boox365 SSO kit. The overall structure is in place: `/sso/initiate`, `/sso/callback`, PKCE (S256), single-use state, JWKS-based ID token verification and the frontend callback page. The items below must be fixed before SSO can work on https://lit.diyalabs.com/. Kit section references are in brackets.

## Blocking

1. **Frontend API base URL points to localhost.** `frontend/src/config.ts` has `BASE_URL = "http://localhost:17102/api"`. In production this breaks password login and the SSO button, which is built from the same value. Use the relative path `"/api"`, and build the SSO button link as `/sso/initiate`. (Step 2)
2. **SSO configuration is hard-coded for the wrong environment.** `application.properties` uses `https://boox365.com` for the issuer, authorize and token URLs, `/.well-known/jwks.json` for JWKS, and `localhost` for the redirect and frontend URLs. Make every `boox.sso.*` value come from an environment variable, with the production values in the "Production SSO configuration" section below. (Configuration values)
3. **Fail closed on the access check.** `checkAppAccess()` returns `true` on any exception, and also when the response has no `allowed` field. A Boox365 outage or a wrong API secret therefore lets everyone in. Deny the sign-in on any error, timeout or unexpected response. (Step 3.4)
4. **Every SSO user becomes ADMIN.** The role is read from `app_role` in the ID token, but the ID token has no such claim. It is returned by the access check. The value is therefore always empty, and `mapRole("")` returns `Role.ADMIN`. Any role containing "ADMIN" is also mapped to ADMIN. Please:
   - take `app_role` from the access-check response;
   - map only exact role names (`ADMIN`, `EMPLOYEE`, `AGENT`, `PARTNER`);
   - never default a new user to ADMIN; treat a missing role as a configuration error, and keep an existing user's role unchanged. (Step 4.3)
5. **ID token checks only log warnings.** In `BooxIdTokenValidator`, an issuer or audience mismatch is logged but the token is still accepted, and a token without `nonce` also passes. Reject the token in all three cases. (Step 3.3)

## Required

6. **Logout must end the Boox365 session.** For SSO users, redirect to `{IAM}/oauth/end-session?post_logout_redirect_uri=https://lit.diyalabs.com/login` after clearing the LIT session. Password users keep their current logout. (Step 4.6)
7. **Restrict sign-in to the Diya Edulabs organisation.** Check that the ID token's `org_id` equals `3b3a0852-b117-42ca-85d5-31bafef2293a`, and refuse any other organisation. (Step 4.2)
8. **Security: remove the blanket permits.** `SecurityConfig` contains `.requestMatchers(HttpMethod.PUT, "/**").permitAll()`, plus public access to `/api/contracts/**`, `/api/payables/**`,

`/api/dashboard/**`, `/api/audit/**` and others. With these rules, these APIs can be called without logging in. Permit only the specific public paths that need it. (Java appendix, "Traps")

## Production SSO configuration

Diya Edulabs uses its own Boox365 sign-in domain, **https://iam.diyalabs.com**. Please use these values (as environment variables):

| Setting | Value |
|---|---|
| Issuer / IAM base URL | `https://iam.diyalabs.com` |
| Authorize URL | `https://iam.diyalabs.com/oauth/authorize` |
| Token URL | `https://iam.diyalabs.com/oauth/token` |
| JWKS URL | `https://iam.diyalabs.com/oauth/jwks` |
| App access URL | `https://iam.diyalabs.com/api/internal/app-access` |
| End session URL | `https://iam.diyalabs.com/oauth/end-session` |
| Client ID | `lit` |
| API secret | We will send this separately after registration |
| Client secret | Not used. Boox365 uses PKCE with no client secret; please remove it |
| Redirect URI | `https://lit.diyalabs.com/sso/callback` |
| Frontend success page | `https://lit.diyalabs.com/sso/complete` (frontend route, see note) |
| Organisation ID | `3b3a0852-b117-42ca-85d5-31bafef2293a` |

Note: the backend callback and the frontend callback page currently share the path `/sso/callback`. The backend route must be reachable at that URL. Please move the frontend page to a different path, for example `/sso/complete`, and point the frontend success URL there.

## Deployment

Commit `ad8eeb2` has not reached the server; the last deployment was on 5 October. Once the fixes above are pushed, we will:

1. register LIT in Boox365 and send you the API secret;
2. route `/sso/` to the backend in the server nginx configuration;
3. set the SSO environment values on the server and deploy;
4. run a joint test from the Boox365 App Switcher.

Before pushing, please confirm the build passes locally ( `mvn -DskipTests package` and `npm run build` ), and run the kit's Testing table, especially row 4 (user without access) and row 6 (Boox365 unreachable).