

Goals

1. Have user registration in MeB
2. Have OAuth app registration in MeB

Agenda

1. Do we migrate users before OAuth apps or vice versa or do we do both together?
2. Depending on the route we intend to take, discuss changes needed in the MB server.
3. How does local development adapt to OAuth changes?
4. Is it possible to have MusicBrainz act as a proxy to MeB OAuth for a transition period?

Migration

The major difference between the two plans is whether MeB.org should be rewritten to read from MB.org DB or vice versa. And how long should both the websites offer app creation and user creation?

Option 1: Users before OAuth Migration Plan

1. Migrate existing user data from MB to MeB database.
2. Deploy new registration and login UI to MeB.org.
3. Make MB.org registration write to MeB db or redirect to MeB login/registration pages.
4. Have login and registration pages work in sync on MB and MeB for a while?
5. Request external formal testing of MeB.org OAuth provider, update test suite and fix issues.
6. Migrate LB/BB/CB to use a new OAuth provider from meb.org to iron out all bugs and issues.
7. Open creation of OAuth applications to the public on meb.org.
8. Send out emails asking to existing active OAuth apps in MB, asking users to migrate and stop allowing creation of new OAuth applications on MB.org.
9. Shut down MB.org OAuth provider.
10. OAuth apps migration is now complete.
11. Disable user registration on MB.org, make login on MB.org function as an OAuth client of MeB.org.

Option 2: OAuth before Users Migration Plan

1. Request external formal testing of OAuth provider, update test suite and fix issues. The OAuth provider will be rewritten to access data from MB db directly in production since that is where the users live.

2. Migrate LB/BB/CB to use a new OAuth provider from meb.org to iron out all bugs and issues.
3. Open creation of OAuth applications to the public on meb.org.
4. Send out emails asking to existing active OAuth apps in MB, asking users to migrate and stop allowing creation of new OAuth applications on MB.org.
5. Stop allowing creation of new OAuth applications on MB.org.
6. OAuth apps migration is now complete.
7. Migrate existing user data from MB to MeB database.
8. Update OAuth provider to read from MeB db.
9. Deploy new user registration and login UI to MeB.org.
10. Make MB.org registration write to MeB db or redirect to MeB login/registration pages.
11. Disable user registration on MB.org, make login on MB.org function as OAuth client of MeB.org

Option 3: Do both together

Notes

We can automate some of the OAuth apps data transfer so that existing approvals/tokens are retained or we can ask users to do a clean start over. Retention would be preferable but there has to be coordination with each individual app owner so that they switch their urls in tandem with the migration.

Development Environment post Migration

We need to figure out how local development environments will verify the access tokens issued by MeB.org are valid and have appropriate access to relevant scopes? On production servers, we can directly use MeB DB tables to verify access tokens.

We probably also need to verify the client id to which the access token was issued (think access token issued for submitting listens to a dev instance being used to submit listens to prod, without additional client id check would be allowed but we likely want to disallow it). Also, how do we verify this in development? Open up introspection endpoints on MeB.org? But a security risk can be used for token fishing. Make everyone run a MeB.org server in development?