

Mechanism to communicate 3PID binding updates or deletions to homeservers

Current state

Right now, bindings between a Matrix ID (MXID) and a third-party ID (3PID) are stored in two places:

- the homeserver (HS) the MXID lives at; there a binding can be either created or deleted. This is mainly spec'd here: https://matrix.org/speculator/spec/HEAD/client_server/unstable.html#adding-account-administrative-contact-information
- the identity server (IS) the homeserver talks to; there a binding can be either created or updated. An update consists in creating a binding on a 3PID that's already bound to a MXID, and will update the existing binding with the new MXID. This is mainly spec'd here: https://matrix.org/docs/spec/identity_service/unstable.html#establishing-associations

In the current state, when a client wants to bind a 3PID (e.g. `alice@example.tld`) to a MXID (e.g. `@alice:alice.tld`), it sends a request to the HS. If this specific 3PID is already known by the HS to be bound to another MXID, the HS refuses to create the binding. Otherwise, it saves it internally then sends a creation request to the IS, which creates the binding, or updates the MXID the 3PID is bound to if there's already a MXID bound to this specific 3PID. In the case of a binding's deletion, the HS deletes the binding locally but doesn't let the IS know about it.

Current issues

This creates a few issues:

- If the binding between `alice@example.tld` and `@alice:alice.tld` is deleted, the deletion only occurs on the HS, thus someone might still retrieve it from the IS (already covered in <https://github.com/matrix-org/matrix-doc/issues/1194>).
- If another user (e.g. `@bob:bob.tld`) takes over the `alice@example.tld` 3PID, even temporarily, they will be able to update the MXID bound to this 3PID on the IS. However, since the binding hasn't been updated on the `alice.tld` HS, `@alice:alice.tld` will not be able to even notice it, making it easier for someone to steal their identity on Matrix.

Proposal

This proposal aims at adding one endpoint to the Matrix SS (aka Federation) API spec:

```
POST /_matrix/federation/v1/3pid/onupdate
{
  "threepid": "alice@alice.tld",
  "deleted": false,
  "mxid": "@bob:bob.tld"
}
```

This endpoint is to be called by the IS upon update or deletion of a binding between a 3PID and a MXID. The example above covers the case of an update, where the newly bound MXID is @bob:bob.tld. In the case of a deletion, the request body would have to be changed into this, with @alice:alice.tld being the MXID bound to this 3PID before the deletion:

```
POST /_matrix/federation/v1/3pid/onupdate
{
  "threepid": "alice2@alice.tld",
  "deleted": true,
  "mxid": "@alice:alice.tld"
}
```

A variant could be to make the "mxid" parameter optional and not include it in the case of a deletion.

Upon request on this endpoint, the HS must then:

1. Update or delete its locally recorded binding
2. Append a "threepids" section to a /sync response looking like this:

```
"threepids": {
  "alice@alice.tld": {
    "deleted": false,
    "mxid": "@bob:bob.tld"
  },
  "alice2@alice.tld": {
    "deleted": true
    "mxid": "@alice:alice.tld"
  }
}
```

The client should then warn the user appropriately.