# Federated Error Handling

Authorization failure (as opposed to outright technical failure) tends to be application specific. A critical category of failure that is specific to federated applications, however, is the case of insufficient information to make an access control decision or provision access to a service. This is occasionally due to technical issues, but more commonly due to privacy controls in place at a user's IdP.

The topic on Federated User Experience includes a discussion on the "boarding" of IdPs by an SP. There are two general approaches that influence how critical this error condition can be to address. One model is to rely on a user- or administrator-initiated workflow that proves an IdP is capable of releasing the necessary information before offering general users the choice to use it. The other is to offer a large set of unproven choices and handle errors afterward. Obviously, the latter approach makes effective error handling absolutely essential to provide a usable service, while the former seeks to minimize errors, and thus make handling them, well, less essential. It is important to understand that no approach will eliminate this kind of error. Whether due to privacy controls or simple technical glitches, there can always be insufficient information available.

Handling this situation well is dependent upon having enough information to lead the user to a resolution of the problem (or a determination that the IdP in question simply won't allow itself to be used for the SP in question). Simply telling the user that some list of attributes has to be supplied is not going to help, particularly if that list is overly technical in nature (e.g., a user will not in general know what an "EPPN" is). Even if the IdP organization has a process for dealing with attribute release, most users are not going to know anything about it. Rather, a contact at the IdP will often need to manage this process and deal with technical matters on behalf of the affected user(s).

For general information about addressing other types of errors, see Error Handling.

## Leveraging the Error Handling URL

A well-known error page at the IdP goes a long way towards helping to address these problems. As it turns out, there is an `errorURL` XML attribute in IdP metadata that is used for this very purpose. It is recommended that IdPs provide guidance and explanations for common errors and scenarios on their error handling page. The page should also include clear instructions that users can follow to troubleshoot errors and resolve access control issues at an SP that are the result of privacy controls or policies.

For certain types of errors, SPs can leverage the error handling URL in IdP metadata. To facilitate this, InCommon now operates a centralized Federated Error Handling Service that will point users to an IdP's public error handling page with some suitable context. If a particular IdP does not have an `errorURL` XML attribute in metadata, the service will point users to the IdP's public information page instead.

It's also important for SPs to provide sufficient information, such as a link to technical documentation via the `<mdui:InformationURL>` element|SPUIElements], a user interface element in SP metadata, so that users (and even IdP staff) understand the nature of the requested service. A list of requested attributes should also be included in SP metadata.

**Recommended Practice**

- Offering users a choice of IdPs that have not been verified to supply necessary attributes is very scalable, but makes it essential to offer a graceful error experience if insufficient data is supplied.
- Failures due to insufficient attribute release are handled by directing users to the IdP's Error Handling URL when available.
- Adequate technical documentation on attribute requirements is available in the SP's metadata (via Requested Attributes and User Interface Elements).