Stream 4: AuthN/Z in new PanDA UI

Tania Korchuganova (Pitt)

30 Jan 2025, WFMS meeting

S4 reminder

Workflow engine

Goal

Make PanDA WMS an interactive and dynamic workflow-oriented platform

- Expand support of complex workflows, for both production and analysis of ATLAS and other experiments
- Optimize algorithms in the system with the awareness of entire workflows

Fa-Hui Lin

Tentative plan

1. Analyzing the visits history by groups of users and collecting feedback from users. **Your input is very welcome!**
2. Investigating the available technologies for potential usage (backend, frontend, cache etc)
3. ~~4~~ API designing and building architecture of data sources and dataflows: what metadata to store and where
4. ~~3~~ Tackle the common AuthN/Z problem

Web-platform

Problems we want to solve:

- To have all steps of running user workflows/tasks in one place: submission, monitoring, debugging and reconfiguration of parameters - more transparency and interactivity.
- Performance of the current monitoring system and lack of interactivity
- Inadequate data flows from the production PanDA DB (slide 6 of the Monitoring TF@TIM talk), the load on DB is not negligible.
- Better customization of views for different groups of users: physicists, site administrators, shifters|experts, software developers (Athena), production managers.

Tania Korchuganova

What we have in BigPanDA monitor

AuthN:

4 third-party SSO providers supported: CERN, Google, GitHub, Indigo IAM (ATLAS instance)

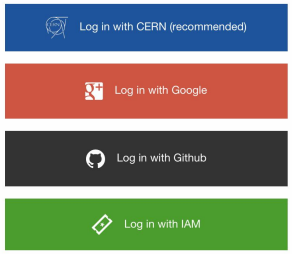
Redirection to login on Django level, the main page is accessible for everybody, all other pages are protected

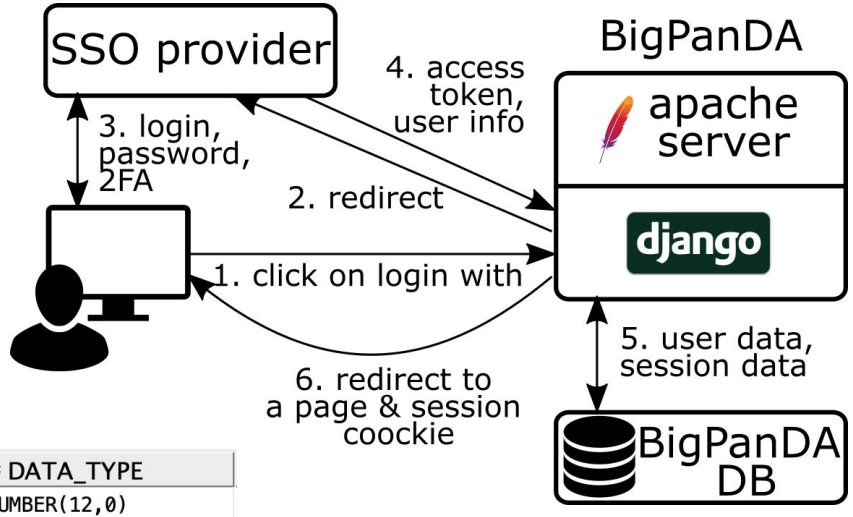
Getting only token and user info, which is stored in **`AUTH_USER`** table

Abort & finish task for a job are possible through:

- ProdSys: we provide username and ProdSys decides if it is allowed
- PanDA server: only if user authenticated with Indico IAM, we use user access_token in HTTP request

A few features only available for `is_expert` (manually) or `is_tester` (via interface)





AUTH_USERS

	COLUMN_NAME	DATA_TYPE
1	ID	NUMBER(12,0)
2	PASSWORD	VARCHAR2(128 BYTE)
3	LAST_LOGIN	TIMESTAMP(3)
4	IS_SUPERUSER	NUMBER(1,0)
5	USERNAME	VARCHAR2(100 BYTE)
6	FIRST_NAME	VARCHAR2(100 BYTE)
7	LAST_NAME	VARCHAR2(100 BYTE)
8	EMAIL	VARCHAR2(100 BYTE)
9	IS_STAFF	NUMBER(1,0)
10	IS_ACTIVE	NUMBER(1,0)
11	DATE_JOINED	TIMESTAMP(3)
12	IS_TESTER	NUMBER(1,0)
13	IS_EXPERT	NUMBER(1,0)

How it is done in ProdSys

AuthN:

Redirection to CERN SSO using Apache module [docs]

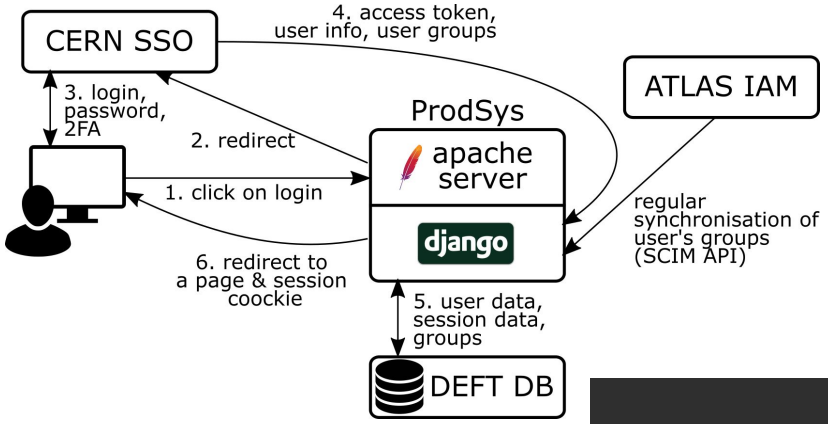
User's groups are fetched from CERN SSO, depending on the application config

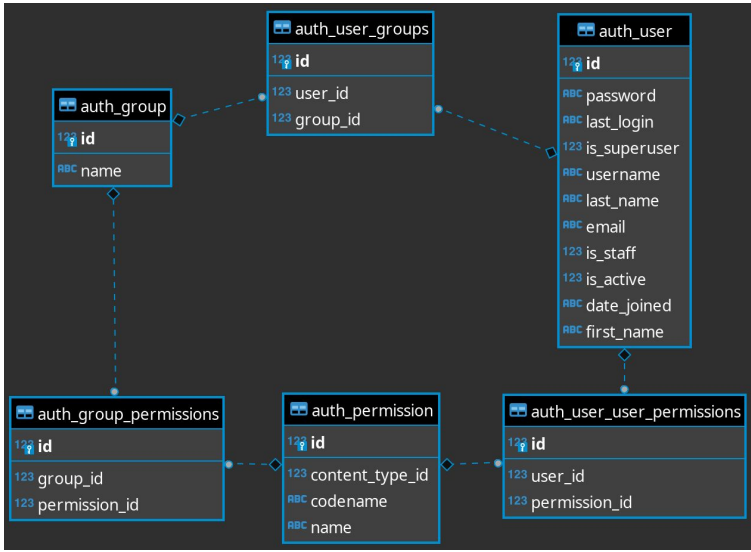
Separately there is a scheduled task to synchronise groups for all users from ATLAS IAM (SCIM API) and DB.

AuthZ:

Logic is in the code as the decision is made depending on type of action, user permission and properties of object

I.e. user12345 in group "atlas-ABCDE", which has permission "can increase task priority". But the upper priority limit is in the code.





How it works in PanDA client - server interaction

AuthN:

Primarily supports X.509 Grid certificate-based authentication

Tokens are supported as well, PanDA uses OIDC device authorisation flow. The PanDA server API accepts access_token and validates the token against IAM

AuthZ:

For users, it uses ID token instead of access token. ID tokens contain groups claims and groups are mapped to roles. Groups are defined in IAM.

Permission policies are relatively simple:

- Generally users can make actions on their tasks/jobs only
- People with prod role can do anything

Name	Description
PANDA_URL_SSL	Base HTTPS URL of PanDA server
PANDA_URL	Base HTTP URL of PanDA server
PANDAMON_URL	URL of PanDA monitor
PANDACACHE_URL	Base URL of PanDA sandbox server
PANDA_AUTH	Authentication mechanism. oidc to enable OIDC/OAuth2.0. x509_no_grid to use X509 without grid
PANDA_AUTH_VO	Virtual organization name (required only when PANDA_AUTH=oidc)
PANDA_VERIFY_HOST	Set off to disable the host verification
PANDA_USE_NATIVE_HTTPLIB	Set 1 to use native http lib instead of curl
X509_USER_PROXY	Grid proxy file path (required only when PANDA_AUTH = x509_no_grid)
PANDA_NICKNAME	Grid nickname (required only when PANDA_AUTH = x509_no_grid)

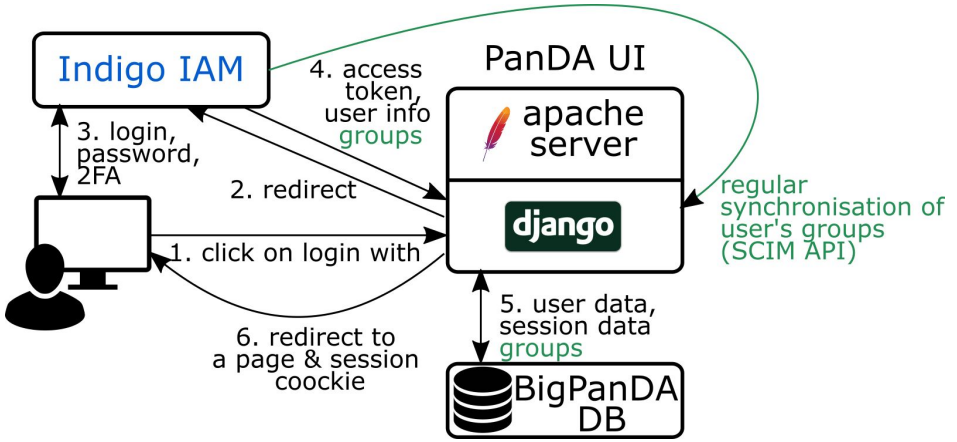
How to do it in the new PanDA UI

AuthN:

- Focusing on IAM as SSO provider
  - users can choose CERN as institute from the list via CILogon
  - site admins who does not have CERN account still can authenticate
- Getting groups from the special SCIM API endpoint

AuthZ:

- Django-based storage and organisation of the user-group-permission data
- **Idea from Misha:**
  - Utilise and benefit from “Attribute-based access control” (ABAC) [link]
  - To have a common policy in a JSON format with fixed JSON schema.
  - Can be stored in a DB table and all systems can use it
  - The single policy can consider different domains, e.g. allowing actions if a user has certain privileges in either “panda” or “prodsys” domains



Status and short-term plans

- The new project is started, I am working on it “locally”, i.e. remote development on bigpanda dev VM via PyCharm
- Django Rest Framework on backend, Angular Framework on frontend
- Started implementing the authN with IAM
- Next step is start working on authZ mechanism, getting groups, introducing permissions (policies can be decided and finalised later)
- Create a new repo in GitHub, write documentation for developers
- Start work on the GUI for analysis workflow submission (as FaHui mentioned in the previous talk)