CERIAS

The Center for Education and Research in Information Assurance and Security

A Comprehensive Access Control System for Scientific Applications

Muhammad Ihsanulhaq Sarfraz*, Peter Baker**, Jia Xu**, Elisa Bertino**#

* Electrical and Computer Engineering, Purdue University

** Cyber Center, Purdue University

Computer Science, Purdue University

Problem Statement

- Web based scientific applications provide means to share scientific data beyond the local computing environment
- The organization and sharing of large and heterogeneous data pose challenges due to their sensitive nature

There is a need for a robust authorization mechanism to prevent unauthorized access to scientific data

- For this purpose, we present an access control system for scientific applications
- We formulate a methodology that incorporates principles from security management and software engineering

Authorization Requirements

- **Implicit Authorization**: An explicitly specified authorization may imply authorizations i.e. authorizations can be automatically propagated
- **Dataset Security**: A user having authorization to execute a tool should not have any authorization to directly modify the dataset accessed by the tool
- **Sandbox Search**: A user is allowed to only execute a browsing query on the existence of data
- **Temporal Constraints**: Permissions have a temporal dimension
- **Conflict Resolution**: Identifying and resolving a conflict is essential in improving usability of any access control system.

Authorization Model

Basic Definition An authorization is defined as (s, o, p, s', c) where: $s \in S$, the set of subjects; $o \in O$, the set of objects; $p \in P$, the set of permission; $s' \in \text{owner}(o) \subseteq S$; $c \in C$, the set of class of objects. A function f is defined to determine if an authorization (s, o, p, s', c) is True or False;

$$f : S \times O \times P \times S \times C \rightarrow (\text{True,False})$$

Definition 1. A *positive authorization* is a tuple (s, o, p, s', c) with $s \in S$, $o \in O$, $p \in P$, $s' \in S$ and $c \in C$. A *negative authorization* is a tuple $(s, o, \neg p, s', c)$ with $s \in S$, $o \in O$, $p \in P$, $s' \in S$ and $c \in C$.

Definition 2. An *authorization base* (AB) is a set of explicit authorizations (s, o, p, s', c) with $s \in S$, $o \in O$, $p \in P$, $s' \in S$ and $c \in C$ where p positive or negative; that is,

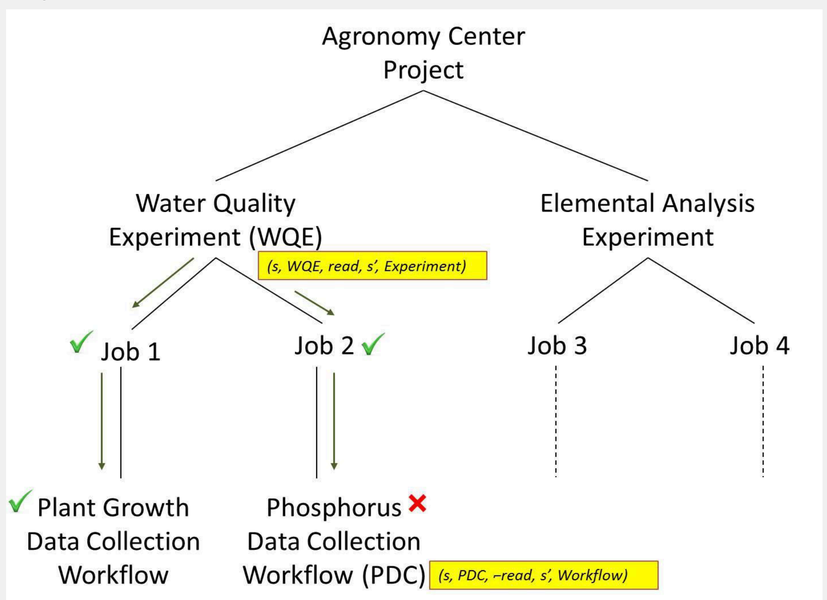
$$AB \subseteq S \times O \times P \times S \times C$$

Definition 3. Function $\imath(s, o, p, s', c)$ is defined as

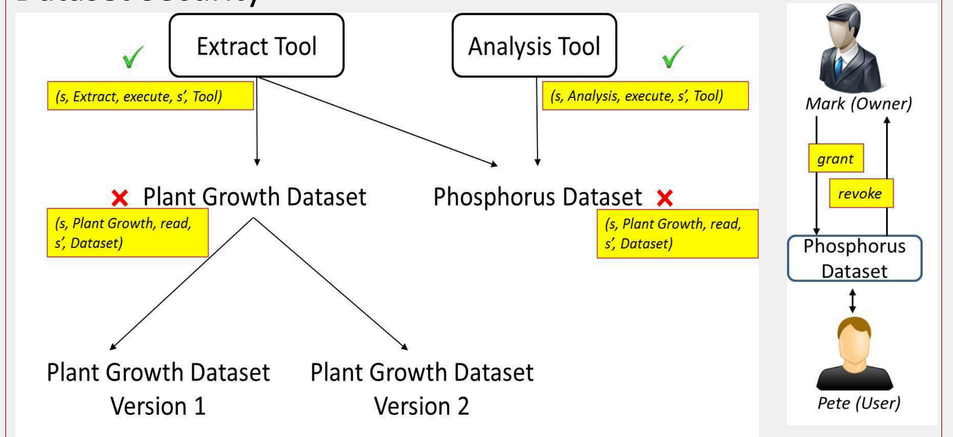
$$\imath; S \times O \times P \times S \times C \rightarrow (\text{True,False})$$

If $(s, o, p, s', c) \in AB$, then $\imath(s, o, p, s', c)$ = True; else, if there exists an $(s_1, o_1, p_1, s_1', c_1) \in AB$ such that $(s_1, o_1, p_1, s_1', c_1) \rightarrow (s, o, p, s', c)$, then $\imath(s, o, p, s', c)$ = True; else, if there exists an $(s_1, o_1, \neg p_1, s_1', c_1) \in AB$ such that $(s_1, o_1, \neg p_1, s_1', c_1) \rightarrow (s, o, \neg p, s', c)$, then $\imath(s, o, p, s', c)$ = False.

Implicit Authorization

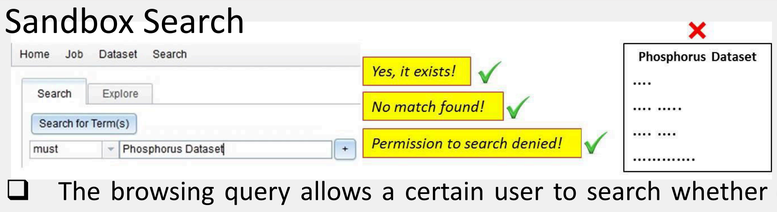


Dataset Security



- The authorization to directly access a dataset must NOT be checked when the dataset is invoked as part of the execution of the tool.
- Most of the large scientific datasets are versioned for long-term preservation. From access control perspective, authorizations must be specified on a versioned dataset and on individual versions of the dataset.

Sandbox Search



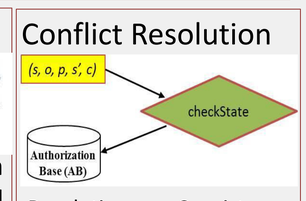
- The browsing query allows a certain user to search whether certain data exists but this does not imply the right to see the actual data.

Temporal Constraint

Definition 4. A *temporal authorization* is a pair (period, auth), where *period* is a time interval $[t_a, t_b]$ with $t_a \in \mathbb{N}$, $t_b \in \mathbb{N} \cup \infty$, $t_a \leq t_b$, and auth = (s, o, p, s', c).

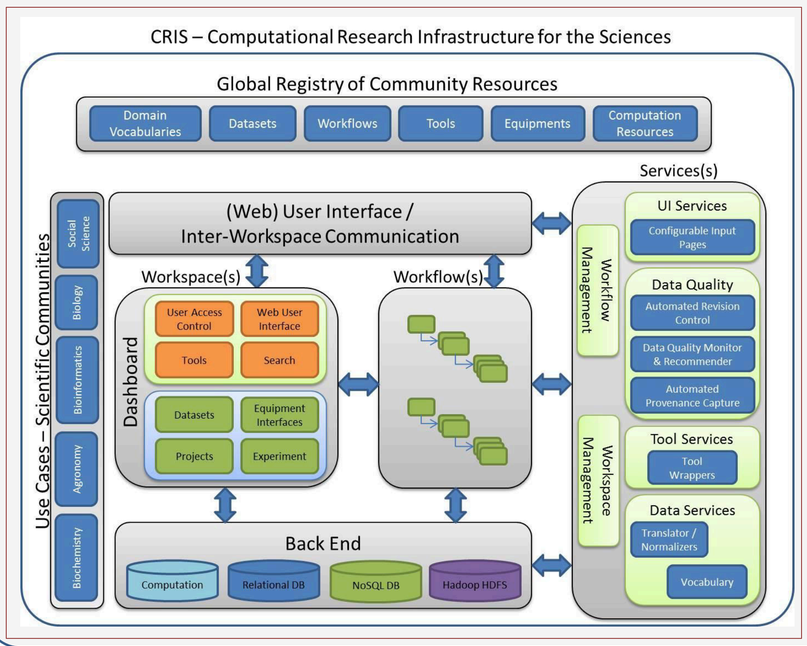
- A temporal constraint is associated with each authorization and refer to as a temporal authorization.

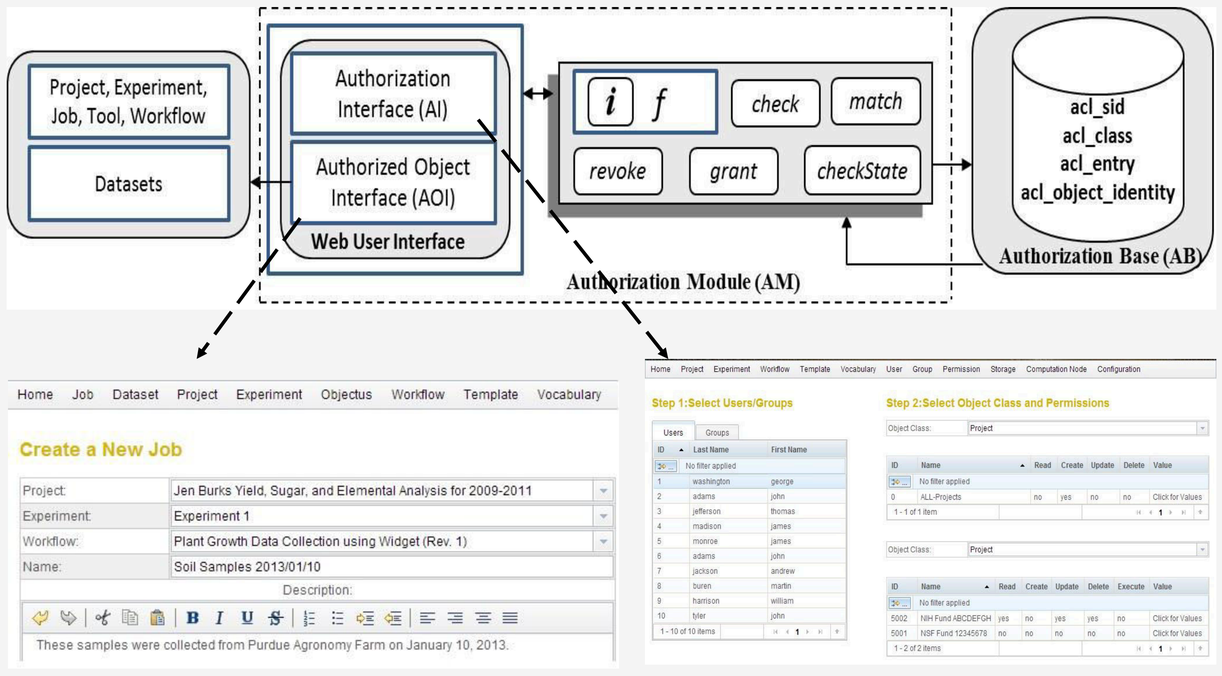
Conflict Resolution



Resolution, Consistency and Redundancy of AB.

CRIS Access Control Architecture





- CRIS is a web based application with its primary tenets to provide an easy to use, scalable and collaborative scientific infrastructure for scientists.
- CRIS has been implemented using open source software and free Web APIs
- Since CRIS is Spring-based, we adopt the authorization modules provided by Spring Security as it is the de-facto standard for securing Spring-based applications.
- The AM in the architecture provides a CRIS user the ability to store/create authorizations through AI and consequently be allowed access to authorized objects through AOI.

CERIAS®

PURDUE UNIVERSITY