5-13-2024

ACCORD: Constraint-driven Mediation of Multi-user Conflicts in Cloud Services

Abhiroop Tippavajjula
The Pennsylvania State University

Primal Pappachan
Portland State University, primal@pdx.edu

Anna Squicciarini
The Pennsylvania State University

Jose Such
King's College London

Citation Details

Tippavajjula, A., Pappachan, P., Squicciarini, A., & Such, J. (2024, May). ACCORD: Constraint-driven Mediation of Multi-user Conflicts in Cloud Services. In Companion Proceedings of the ACM on Web Conference 2024 (pp. 1039-1042).

ACCORD: Constraint-driven Mediation of Multi-user Conflicts in Cloud Services

Abhiroop Tippavajjula
apt5698@psu.edu
The Pennsylvania State University
State College, USA

Primal Pappachan
primal@pdx.edu
Portland State University
Portland, USA

Anna Squicciarini
acs20@psu.edu
The Pennsylvania State University
State College, USA

Jose Such
jose.such@kcl.ac.uk
King's College London
London, UK

ABSTRACT

When multiple users adopt collaborative cloud services like Google Drive to work on a shared resource, incorrect or missing permissions may cause conflicting or inconsistent access or use privileges. These issues (or conflicts) compromise resources confidentiality, integrity, or availability leading to a lack of trust in cloud services. An example conflict is when a user with editor permissions changes the permissions on a shared resource without consent from the original resource owner. In this demonstration, we introduce ACCORD, a web application built on top of Google Drive able to detect and resolve multi-user conflicts. ACCORD employs a simulator to help users preemptively identify potential conflicts and assists them in defining *action constraints*. Using these constraints, ACCORD can automatically detect and resolve any future conflicts.

CCS CONCEPTS

• **Web and Internet systems**; • **Information systems → Web applications**;

KEYWORDS

User Policy Conflicts, Cloud Systems, Policy Simulator, Conflict Detection, Conflict Resolution

ACM Reference Format:
Abhiroop Tippavajjula, Primal Pappachan, Anna Squicciarini, and Jose Such. 2024. ACCORD: Constraint-driven Mediation of Multi-user Conflicts in Cloud Services. In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24 Companion), May 13–17, 2024, Singapore, Singapore*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3589335.3651244

1 INTRODUCTION

Collaboration in the digital age has been significantly enhanced by the widespread adoption of cloud-based systems, such as Google Drive. These platforms facilitate real-time interaction on shared resources. However, as users work together on shared files within these platforms, actions such as *create, edit, delete, move, and change permission* can generate multi-user *conflicts*. We will illustrate multi-user conflicts with an example scenario shown in Figure 1 of an organization with two departments (IT & HR) which uses a cloud service, such as Google Drive, for collaboration. Alice shares a document with Bob (1), expecting him to adhere to organizational protocols and not share it further. Bob, who is a new employee, edits file A (2), ignorant of these protocols, adds Carol from the HR department as an editor (3), who then makes extensive changes (4) and moves the document to a different folder (5), causing Alice and Bob to lose access. In this example scenario, there are 3 multi-user conflicts happening due to limitations of cloud services: 1) permission change conflict when Bob shared the document with someone outside the IT department leading to loss of confidentiality, 2) edit conflict when Carol rewrites portions not written by Bob leading to loss of integrity, and 3) move conflict when Carol moves the document to HR folder leading to loss of availability.

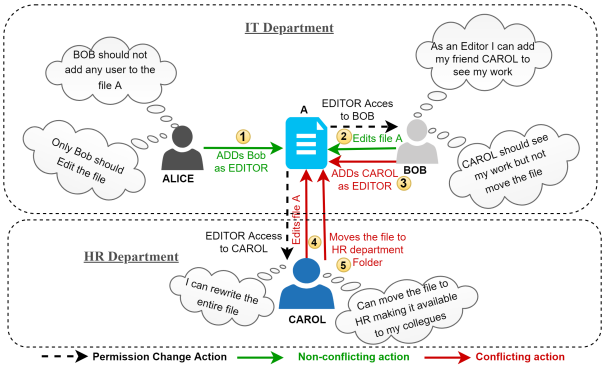

Figure 1: Multi-User Conflict Scenario in an Organization

Multi-user conflicts or simply conflicts are actions performed by a user on a shared resource that doesn't meet the organizational or individual expectations of other parties. As illustrated in the example, conflicts can lead to loss of integrity, confidentiality, and availability to the other users. These are often due to the limited capabilities of cloud services in allowing users to specify their fine-grained constraints w.r.t collaboration on a shared resource. In the previous example, if Alice could impose a constraint to receive an alert when Bob tried to share the document beyond the IT department, she could have prevented Carol from accessing the document and moving it outside the IT department. A recent study [1] found that

WWW '24 Companion, May 13–17, 2024, Singapore, Singapore.

ACM ISBN 979-8-4007-0172-6/24/05
https://doi.org/10.1145/3589335.3651244

such multi-user conflicts are prevalent in organizations, with 54% of participants encountering such conflicts in the past six months and 76% in the past year.

The controls offered in existing cloud services are not sufficient to either detect or resolve multi-user conflicts due to limitations in their scope and granularity. In Google Drive, while the owner can restrict all editors from altering permissions, this approach lacks the fine-grained control needed for scenarios like Alice's, where she wishes to permit Bob to share files within the IT department but not beyond. Google Drive does not support such fine-grained constraints, forcing a choice between all or nothing w.r.t permission changes by editors. This absence of adequate support for fine-grained permissions leads to incorrectly authorized actions, impeding effective collaboration.

In response, we introduce the ACtion COnstraint-based conflict Resolution and Detection (ACCORD) system, a novel approach offering a universal solution for conflict resolution in various cloud-based environments. [1] ACCORD supports defining *Action Constraints*, which are fine-grained permissions defined by users on shared resources. By monitoring changes in access permissions and resource content, and comparing them against these constraints, ACCORD accurately identifies conflicts and suggests resolution strategies. ACCORD also includes an action simulator engine that allows an organization or an individual user working on a shared resource to safely check the set of possible actions on their resources. The simulation environment generated by ACCORD, enables users to witness and interact with the set of all possible user actions, and flag them as conflicts or not. Action constraints are automatically generated based on flagged actions for detecting and resolving potential conflicts. Upon detection, ACCORD recommends viable resolution strategies including undoing the conflict causing action whenever possible.

This system marks a significant advancement in supporting fine-grained permissions for collaborative cloud systems, as it provides a simulation environment to safely study possible conflicts, automatically generate constraints based on identified conflicts, detect conflicts in real-time, and recommend resolution strategies. These functionalities go well beyond what is supported by current cloud services (see comparison in Table 1). ACCORD supports tracking file movements, auditing files moved to unauthorized locations, monitoring edits outside specified parameters, and many others. The demo showcases these features using different scenarios based on a preset organization. Participants will be able to observe how ACCORD effectively rectifies incorrect or missing permissions through their interactions with the demo's application.

2 RELATED WORK

Several studies have addressed multi-user conflicts in many domains such as IoT [4], home automation [6], and social networks [5], with a limited but growing focus on cloud services. In comparison, most work on Multi-User Conflicts (MCs) has been centered on social media, In the domain of social networks, prior work has explored empirical studies on the phenomena of MCs mentioned in [2] as well as methods and systems for preventing and managing MCs to avoid security and privacy consequences [5]. In the realm of IoT, research highlights the importance of proactively detecting and resolving safety property violations in shared spaces, as investigated in [4]. The Kratos system [6] pioneers a flexible access control system for smart home scenarios with multiple users and devices. While these studies provide valuable insights into conflict resolution in social media and IoT settings, ACCORD addresses the challenges in cloud service environments where the shared nature of resources and a larger number of users collaborating simultaneously necessitates a more comprehensive and real-time approach to conflict resolution.

[1] Our demo is based on Google drive, but ACCORD is also applicable in similar systems like Microsoft OneDrive.

3 OVERVIEW

ACCORD consists of three main modules: Action Constraint Manager, Conflict Detection, and Conflict Resolution. The Action Constraint Manager is made up of an Action Simulator Module, which simulates allowable actions on resources, and an Action Constraints Database that holds user-specified action constraints. The Conflict Detection module, illustrated in Fig. 2, includes a Log Extraction Module that retrieves and processes activity logs from the cloud service, and a Detection Engine that identifies conflicts using these logs and action constraints. The Conflict Resolution module suggests resolutions to these conflicts, such as reversing the conflicting action or modifying permissions to prevent future issues, based on the involved users.

3.1 Action Constraint Manager

The Action Constraint Manager is designed to enhance collaboration within cloud platforms like Google Drive. It consists of two sub-modules: the Action Simulator and the Action Constraints Database, both crucial for managing and resolving conflicts in shared resources.

3.1.1 Action Simulator Module. The Action Simulator module utilizes Google Drive APIs to simulate user actions on resources that allows users to determine what conflicts may occur on their shared resource. The simulator operates by taking inputs such as resources,

Feature / Control	Drive	OneDrive	ACCORD
Prevent Editors from sharing	✓*	✓*	✓
Expiration dates for access	✓*	✓*	✓
Detect unauthorized downloads	×	×	✓
Detect permission removal	×	×	✓
Alert on adding permissions	×	×	✓
Monitor modifying permissions	×	×	✓
Track file movements	×	×	✓
Audit file movements	×	×	✓
Monitor file edits	×	×	✓
Track edits outside expiration	×	×	✓
Disable download, print, or copy	✓	✓	✓
Alert on file deletion attempts	×	×	✓
Alert on file creation attempts	×	×	✓

Table 1: Comparison of Controls. *Only binary (all or nothing) controls supported as on June 2023.

users, and their permissions to simulate potential actions (e.g., create, delete, move, update) based on the existing permissions. This simulation helps users preemptively identify possible conflicts on shared resources. Each action is performed multiple times to cover a broad spectrum of user interactions, generating log entries for each action that are visible only to the user conducting the simulation. After the simulation, users can review the actions, flag any perceived conflicts, and ACCORD will automatically create action constraints for these conflicts. Users have the flexibility to edit these constraints to ensure ACCORD accurately detects and resolves future conflicts. Such policy Simulators are commonly used in cloud computing services to see how the existing access control policies impact users and troubleshoot them as necessary [2] but they are absent in cloud collaborative services.

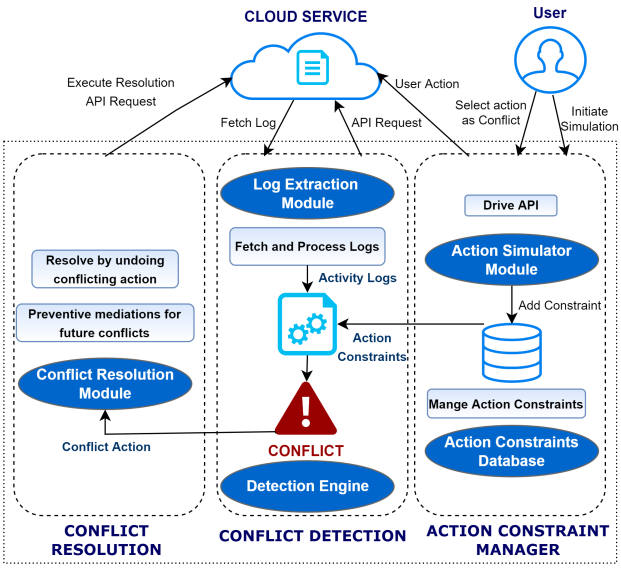

Figure 2: ACCORD's architecture

3.1.2 Action Constraints Database. The Action Constraints Database stores detailed, user-defined constraints that set specific permissions on shared resources, crucial for the conflict detection process. These constraints define what actions are considered conflicts based on unique scenarios within the shared environment. The database is dynamically updated with new constraints, modifications, or deletions, ensuring ACCORD's conflict detection module has the most current data to effectively identify and flag conflicts. This database plays a pivotal role in maintaining the integrity of shared resources by providing a structured framework for managing user interactions and preventing unauthorized actions.

3.2 Conflict Detection Module

The Conflict Detection module in ACCORD is integral for identifying and managing conflicts within cloud services. It consists of two key components: the Log Extraction Module (LEM) and the Detection Engine, each playing a critical role in monitoring and resolving user actions that could lead to conflicts.

[2]https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_testing-policies.html

3.2.1 Log Extraction Module. LEM operates by retrieving activity logs through an API provided by the cloud service, such as Google Drive's Reports API. It focuses on collecting logs based on their timestamps, ensuring only the most recent entries are processed. The module extracts crucial information from these logs, including the timestamp of the activity, action type (create, delete, edit, move, permission change), document ID, and the actor's ID (the user who performed the action). For certain actions like permission changes, LEM goes further to identify the target user, previous permissions, and the permissions applied after the change. Continuous retrieval and processing of logs allow LEM to provide up-to-date data for effective conflict detection and resolution.

3.2.2 Detection Engine. Upon receiving processed logs from LEM, the Detection Engine retrieves relevant action constraints from the database. It utilizes a multi-level hash map for storing these constraints, keyed by action type and target user, to streamline the matching process. The engine then compares the processed logs against these constraints using a series of conditional checks that assess each log entry's elements (resource ID, action, action type, and target user) against the constraints. This process is highly efficient, beginning with the most selective checks and terminating as soon as a mismatch is found. When a log entry fully matches a constraint, indicating an action that needs monitoring, ACCORD flags this as a conflict. The Detection Engine is designed to pinpoint potential conflicts for resolution in constant time.

3.3 Conflict Resolution Module

The Conflict Resolution Module is tasked with offering resolutions for detected conflicts. A *resolution* is an undo action taken to reverse the conflicting action or a preventive action to protect against conflicting actions from happening in the future. From the example in Figure 1, an *undo* resolution action is remove the edits done by Carol on the document. A *preventive* resolution action is to restrict Bob from sharing the document outside the IT department. Other resolution strategies include but are not limited to restoring the original file, defining a new policy, or seeking consensus among affected users [3]. Depending upon the action type, specific action that instigated the conflict, and the permissions of the author who specified the action constraint, either undo or preventive or both resolution actions might be possible. In situations where the user may not have appropriate permissions required to execute the resolution, ACCORD recommends the user to contact either the conflict initiator or the document owner who could then execute the resolution. Furthermore, it is possible that resolving a conflict might lead to further conflicts, a.k.a chain of conflicts. We restrict the solution space of resolution strategies to actions that do not lead to chain of conflicts. Upon detection of a conflict, ACCORD proposes the most fitting resolution, if more than one is available, and assists the user in implementing the suggested resolution. Once a conflict has been resolved, all the relevant users (e.g., owner of the document, conflict initiator) are informed of the resolution.

4 DEMONSTRATION PLAN

The demonstration is designed to showcase ACCORD's capabilities in simulating user actions on shared resources, generating action constraints, and finally detecting, and resolving multi-user conflicts

utilizing those constraints. We chose Google Drive as the demonstration environment for this paper. The majority of cloud services share similar APIs, making the extension of ACCORD to other platforms a straightforward process. For demonstration purposes, we preset ACCORD with a small-scale organization with up to 50 users, their resources, and access permissions on these resources. Using this preset, ACCORD can simulate 8 distinct scenarios which may lead to a conflict depending on the set permissions. Action simulator is capable of generating up to 100 actions for each scenario, allowing participants to identify and select specific actions that may cause conflicts. This hands-on approach aims to provide an in-depth understanding of limitations of permission systems of current cloud systems. We have included a brief video showcasing the various modules of ACCORD at this URL. The source code for the simulator and ACCORD web application along with documentation on how to run it is available on GitHub [3].

Following are the key interactive steps of the demonstration.

- Participants start with an overview of conflicts and receive access to the ACCORD as a user of the preset organization.
- The ACCORD dashboard presents various scenarios shown in Fig 3 involving actions such as creation, deletion, editing, moving, and permission changes.

Figure 3: Conflict Type Scenarios

- Participant initiates simulation of multiple user actions on a resource, based on predefined permissions.
- After the simulation, participants review a detailed log of actions and flag those perceived as conflicts, such as the actions outlined in Figure 1.
- For each flagged conflict, ACCORD automatically generates action constraints shown in Fig 4, enabling future conflict detection. Participants can edit them as necessary.

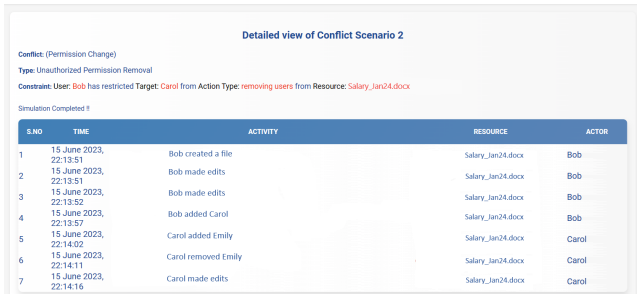

Figure 4: Conflict Simulator Actions

[3]https://github.com/Abhiroop-tales/ACCORD

- After saving the constraints, participants can re-run the simulation and a different set of actions will be generated. ACCORD will detect conflicts based on set action constraints.
- For detected conflicts, ACCORD presents resolution options and preventive measures for the detected conflict, guided by the resolution module.
- Participants have the flexibility to choose a resolution method, with the option to decide whether the same resolution method should be applied in future conflict detection. (Fig 5).

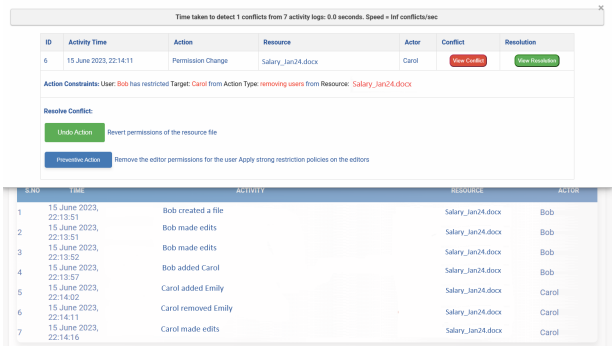

Figure 5: Conflict Detection and Resolution

The demo allows participants to select various scenarios for simulation, edit action constraints, or choose alternate resolution strategies, showcasing ACCORD's robustness with a diverse set of users, resources, and access permissions.

5 CONCLUSIONS AND DISCUSSIONS

ACCORD fosters effective collaboration in cloud services via an action constraint-driven approach to conflict detection and resolution. In future, incorporating ACCORD as a browser plugin, that enables users to specify action constraints and detect and resolve conflicts from within Google Drive, will improve usability. Furthermore, a user study focused on assessing ACCORD's usability and effectiveness in conflict detection and resolution will bring about more insights.

REFERENCES

[1] Eman Alhelali, Marvin Ramokapane, and Jose M Such. 2023. Multiuser Privacy and Security Conflicts in the Cloud. In *2023 ACM CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery (ACM).

[2] Mauro Cherubini, Kavous Salehzadeh Niksirat, Marc-Olivier Boldi, Henri Keopraseuth, Jose M Such, and Kévin Huguenin. 2021. When forcing collaboration is the most sensible choice: Desirability of precautionary and dissuasive mechanisms to manage multiparty privacy conflicts. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (2021), 1–36.

[3] Egon Ostrosi, Lianda Haxhiaj, and Shuichi Fukuda. 2012. Fuzzy modelling of consensus during design conflict resolution. *Research in Engineering Design* 23 (2012), 53–70.

[4] Pavana Pradeep and Krishna Kant. 2022. Conflict Detection and Resolution in IoT Systems: A Survey. *IoT* 3, 1 (2022), 191–218. https://doi.org/10.3390/iot3010012

[5] Sarah Rajtmajer, Anna Squicciarini, Christopher Griffin, Sushama Karumanchi, and Alpana Tyagi. 2016. Constrained social-energy minimization for multi-party sharing in online social networks. In *proceedings of the 2016 international conference on autonomous agents & multiagent systems*. 680–688.

[6] Amit Kumar Sikder, Leonardo Babun, Z. Berkay Celik, Abbas Acar, Hidayet Aksu, Patrick D. McDaniel, Engin Kirda, and A. Selcuk Uluagac. 2020. Kratos: multi-user multi-device-aware access control system for the smart home. In *WiSec '20: 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks, Linz, Austria, July 8-10, 2020*, René Mayrhofer and Michael Roland (Eds.). ACM, 1–12. https://doi.org/10.1145/3395351.3399358