Case Tracking System Use-Case Description

Case Tracking System in Law Companies: Case Study

Many law companies utilize some kind of case tracking system. Big law companies utilize such systems for billing and file sharing, while smaller companies utilize email for everything as case tracking systems get very expensive. Also, those systems are not very secure, as everyone in the company generally has access to every file of the case. That may pose conflict of interests, if, for example, a company works for multiple clients that have such a conflict. Therefore, we propose a case-study that utilizes Next Generation Access Control (NGAC), a new implementation of Attribute Base Access Control mechanism.

This case study is used to demonstrate every component of NGAC, including Graph, Prohibitions, and Obligations. Case Tracking System (CTS) is intended to be a small example that can demonstrate most of the functionality of NGAC. In this example, we have a limited number of nodes with the possibility of expansion.

Since we intend to make this example small, we utilize only 3 policy classes, called “Law Firm Policy”, “Case Policy”, and “ValueType Policy”. The first one contains two hierarchies of attributes. One hierarchy consists of User Attributes and is used to manage the personal of the company, and another hierarchy consists of Object Attributes which is used for managing the cases.

Each office has attorneys of various ranks (Lead Attorneys, Attorneys, and Interns), each with various access privileges. The main office additionally has C-Suit personnel and HR. Intentionally, we have an association from User Attribute “Main Office” to the User Attribute “Office 1” that may grant the unintended access privileges (HR can add personal to their office for example). This is done to demonstrate how to handle it with the Prohibitions. Another use of prohibitions will be to limit the access rights of interns, which should not be able to see any cases for security purposes, however, should be able to see the general information about the cases.

As this policy is intended to be scalable, there can be an unlimited amount of offices, cases, and attorneys, and interns. Those entities are intended to be scalable for the verification/testing purposes.

“Case Policy” is the policy that is used for approving a new case. All of the changes to this policy are done via obligations, and the workflow section describes the process of approving the case. “ValueType Policy” is used in approval process also and the sole purpose is to classify a new case by value and its type.

Law Firm Policy

UA ϵ {Office 1, Main Office, Attorneys, J.D. 1, J.D. 2, Interns, HR, C-Suit, Lead Attorneys}

OA ϵ {Cases, General Info, Case 1, Case 2}

Case Policy

UA ϵ {Attorneys, C-Suit, Lead Attorneys}

OA ϵ {Cases, General Info, Case3, Type, Value, Parties}

Figure 1 presents Law Firm Policy, which describes which attorney have the access to the cases information, as well as which entities can make hiring decisions.

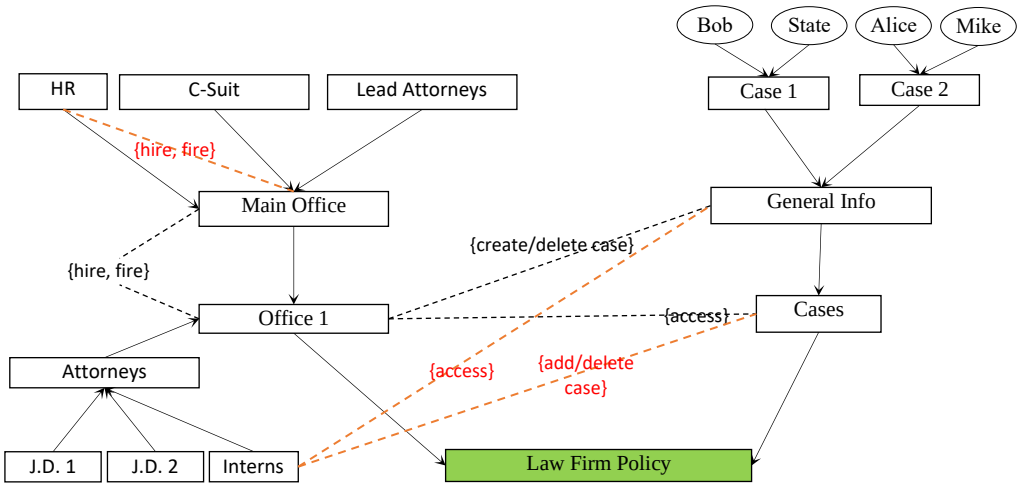

Figure 1. Law Firm Policy

Figure 2 shows Case Policy when New Case needs to be accepted. This is a toy example, so for the case to be accepted, the attorney, his/hers Lead Attorney, and someone from C-Suit needs to accept the case for it to be added in Law Firm Policy. The Figure 2 shows the initial state of this policy, which provides an attorney and a lead attorney to accept the case, while attorney can accept it before lead attorney and lead attorney can accept the case before the attorney. C-Suit will be eligible to accept the case only after both Attorney and Lead Attorney will accept the case. The excessive access rights are controlled using the prohibitions. Also, Appendix 1 describes the cases when the cases are refused, disapproved, or withdrawn, as well as the creation of the case.

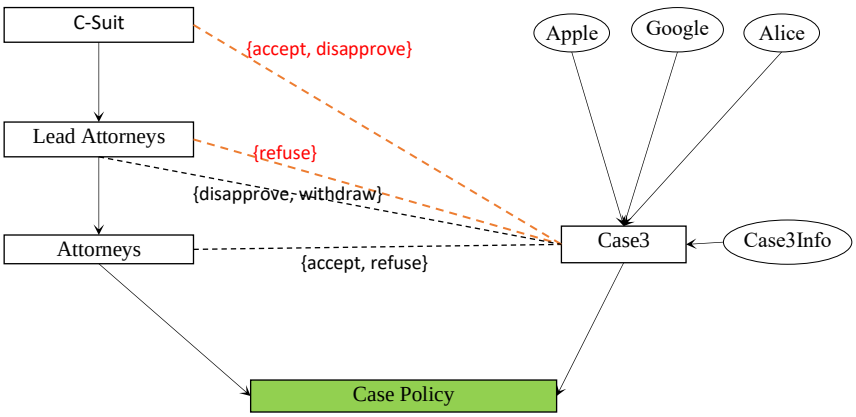

Figure 2. Case Policy Before Attorney/Lead Attorney/C-Suit Acceptance

Figure 3 shows the Policy after Attorney accepts/refuses the case before Lead Attorney accepts the case.

1. Attorney loses all access rights via association deletion
2. Lead Attorney gains “accept” access right

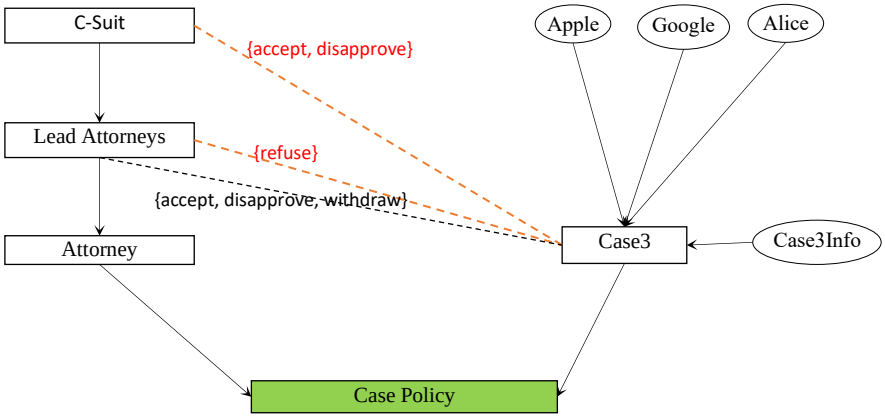

Figure 3. Case Policy After Attorney Acceptance/Refusal Before Lead Attorney Acceptance

Figure 4 shows the Case Policy after Lead Attorney accepts the case before Attorney accepts the case.

1. Lead Attorney loses the right to accept the case via prohibitions

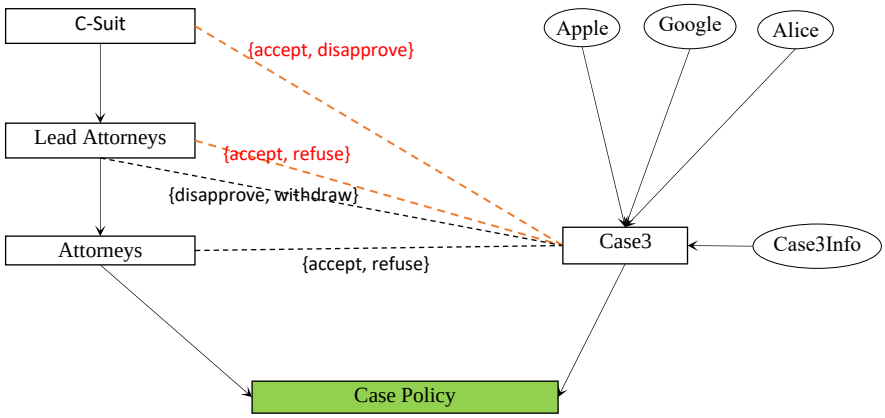

Figure 4. Case Policy After Lead Attorney Acceptance Before Attorney Acceptance

Figure 5 shows the Case Policy after Lead Attorney accepts the case and Attorney accepts the case.

1. Lead Attorney loses the right to disapprove and withdraw the case via association deletion.
2. C-Suit gains the access rights to approve and withdraw the case.
3. Lead Attorney loses the right to accept the case via prohibitions or association deletion.

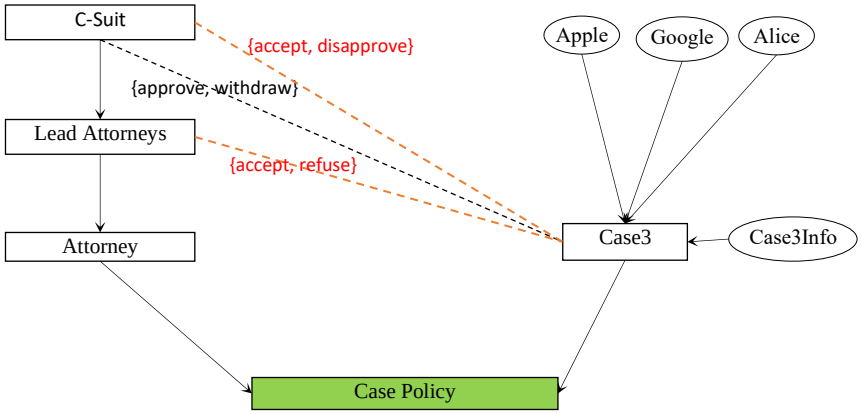

Figure 5. Case Policy After Lead Attorney Acceptance

Figure 6 shows the Law Firm Policy after the case is approved by C-Suit and the Case 3 is assigned to General Info. Case3Info is deleted and assignment from Case3 to Case Policy is removed.

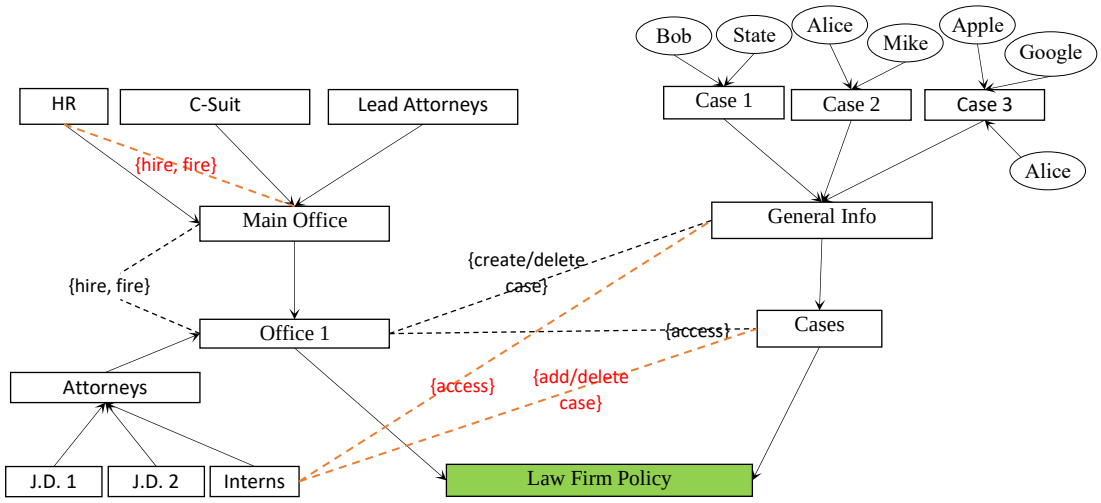

Figure 6. Law Firm Policy After Case 3 Approval

Workflow

The following workflow diagram represents the states of CTS during each step of the approval described with the figures above.

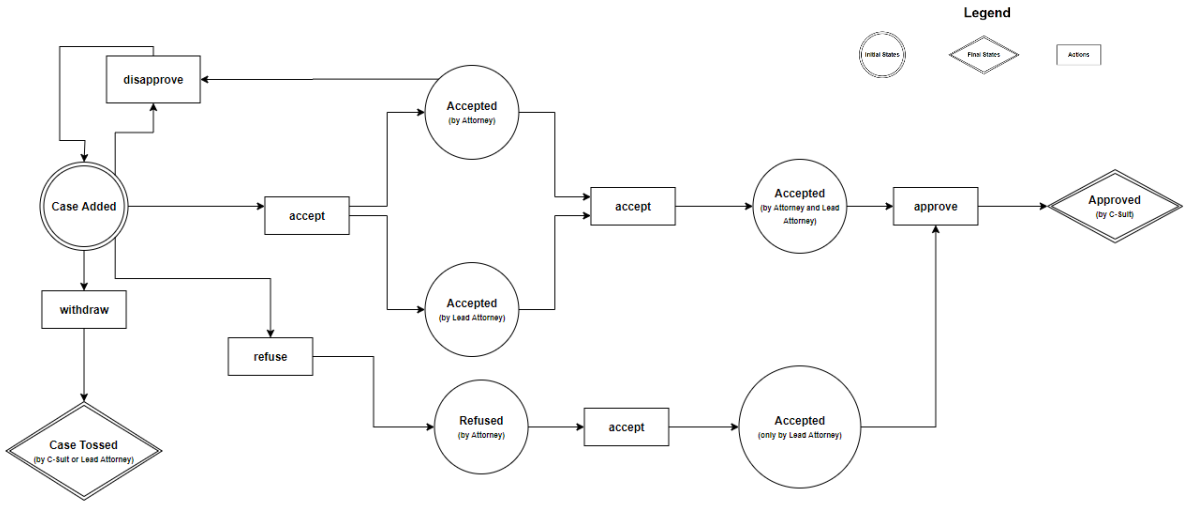

Figure 7. CTS Workflow for Case Approval

As can be seen in the Figure 7 legend, our workflow consists of the initial state, final states, and actions that represent the transitions between the states. Once a case is added by anyone except Interns, the new case is added to Law Firm Policy which starts the workflow of case approval. Each action in the workflow is performed be different users, while some actions are meant to be concurrent and others consequent. The changes to the graph/prohibitions are done via obligation actions.

Once the case is added and assigned to a specific value group (less or equal to $5000, or more than $5000), C-Suit or Lead Attorney can immediately withdraw it, therefore, turning the state of the case to “Tossed”. Another opportunity to “Toss” the case occurs if the lead attorney disapproves it after an Attorney approves it. In this case, the case can be “Tossed” as well.

If the case is not tossed, the following “happy scenario” is expected:

1. The case is accepted by both Lead Attorney and Attorney in any sequence, or even concurrently, the cost comparison is triggered.
2. The case should be approved by C-Suit.

The other possibility is when the attorney refuses to take the case. In such scenario, the Lead Attorney can choose to accept the case anyway to work on it him/herself and the scenario continues as in the happy scenario. Lead Attorney may also disapprove the case, to see if the case should be “Tossed” or if another attorney is willing to accept the case.

Figure 8 demonstrate a more complex of version of CTS. In this version, a case’s type/subtype comparison, as well as value comparison, is added. This comparison works similarly to the cost comparison, and checks the assignments to Type/Subtype graph described in the Figure 9.

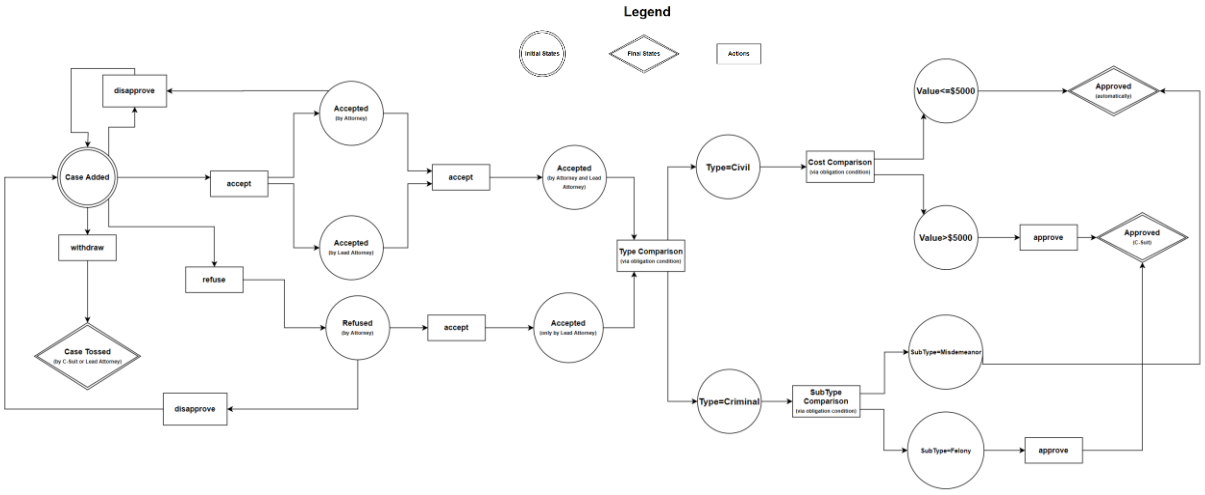

Figure 8. More Complex CTS Workflow for Case Approval

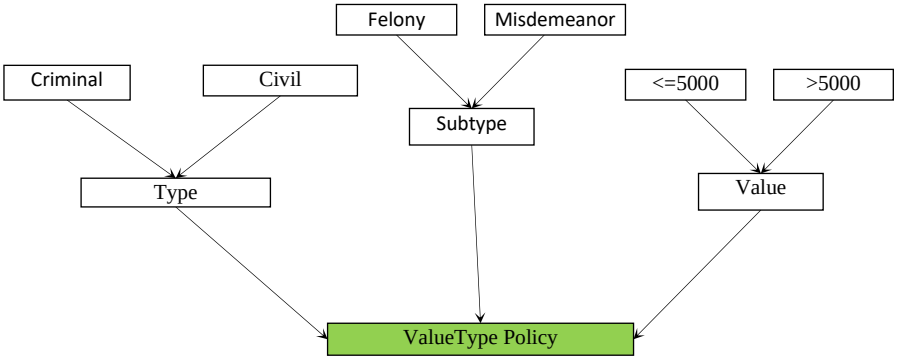

Figure 9. ValueType Policy

Conflict of Interests Handling

This section will describe the conflict of interests handling. “Case Policy” includes “Parties” Object Attribute. As CTS is made for the internal usage, there is no reason to make “Parties” a User Attribute as clients of the law company should not have access to this system.

In order to solve the problem of Conflict of Interests (COI), we propose to compare the parties of the new case with the existent parties of a particular attorney (J.D.). For that purpose, we introduce the “COI Policy”, presented in Figure 10. In order to not overcomplicate the graph, we do not create a new policy class, but instead performing actions in Law Firm Policy class. When an attorney attempts to perform the “accept” action on a case during the workflow, the association of the attorney are checked before the action actually triggers the obligation.

The motivation for handling the COI in this case-study is that some states' law requires a firm to decline a case if any party of the case is the same. In other states, the firm will also decline a case to avoid any litigations due to the ethical issues. Our solution can efficiently handle this situation, and can provide the ability for law businesses to not lose revenue due to the COI situations.

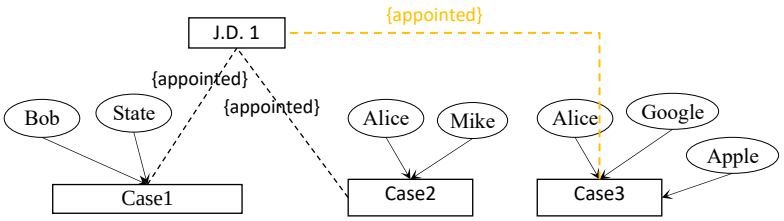

Figure 10. COI Policy

Appendix 1

The following describes the cases when the cases are refused, disapproved, or withdrawn.

Figure 1 shows the Case Policy after any Lead Attorney or C-Suit user tosses the case via withdraw action

1. Object attribute Case 3 is deleted; therefore, the rest of the nodes are deleted as well.

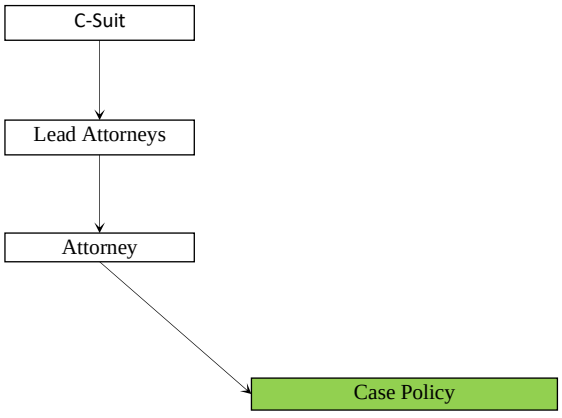

Figure 1. Case Policy After Case 3 is Tossed

Figure 2 shows the Case Policy after any Lead Attorney or C-Suit user disapproves the case.

1. The Case Policy class is restored to the initial stage in case if the attorney already accepted/refused the case.

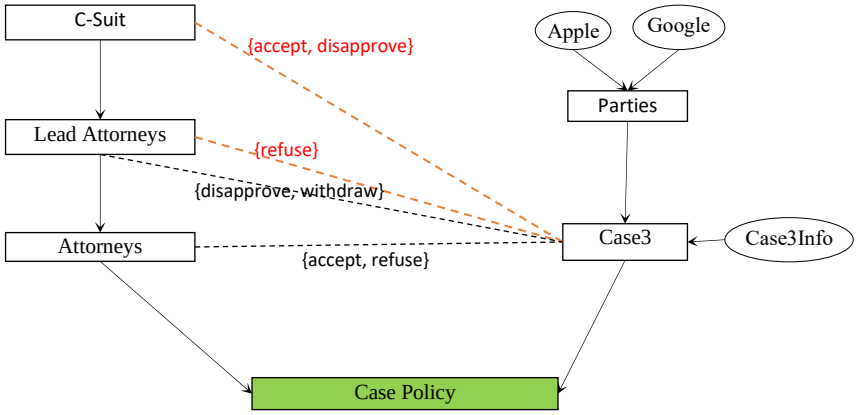

Figure 2. Case Policy After Disapproval by Lead Attorney

Figure 3 shows the Case Policy after anyone, except Interns, creates a case. If this was a real implementation, first, Case 3 would be assigned to Case Policy, and then Case3Info would be assigned to Case3. However, for simplicity purposes of this case study, we just create Case3Info node and assign it to Case3. Case3Info is used to check the access adjudication since the prohibitions are utilized on Case3. As they do not affect Case3 but only the nodes contained by Case3, the Case3Info node is needed to check the access privileges.

1. Case3Info object is created via obligations in Case3 container.

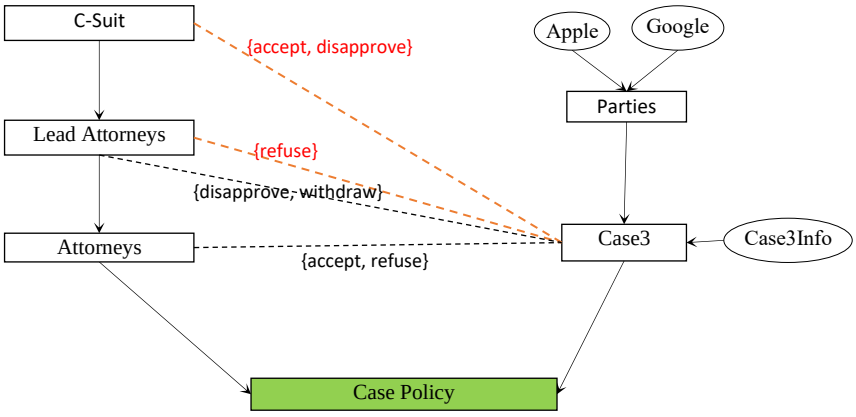

Figure 3. Case Policy After Case3 is created