Audit System Requirements Document

# Objective

The Audit System will help track the user’s movements and actions when using the Policy Machine. This will allow the administrator(s) to query the data collected. At this point in the Policy Machine’s development the development and integration the addition of an Auditing System component was necessary so the tracking of tuser movements could be possible.

The Audit Machine’s specific objectives include:

* Completed look and feel
* Ability to perform data collection
* Ability to track a user’s actions and movements through the system
* Ability to display results that are stored in the database
* Ability to filter what results are displayed by what the user wants to observe
* Ability to use multiple filters at one time in order to display the desired results

These objectives will be discussed in a later section.

# Business Process

There will be and optional addition to the administrator(s) business process but there will be no new processes for regular users. Administrators will be able to Audit the system from the Administrator’s Interface.

## User Roles and Responsibilities

In this system there are 2 types of users:

* Administrators
* Users

Administrators will have the ability to Audit the Policy Machine using the Audit Graphical User Interface (GUI). They can also filter the results based on specific data they want returned.

Users actions and movements are tracked in the background without their knowledge. Users also cannot query the database that stores all of the Audit System information that has been stored.

Eventually administrators will also have the opportunity to decide when old data will be archived. This will be a feature in a later version of the Audit System.

## Interactions with Other Systems

The Audit System interacts with the Policy Machine in many ways. AS communicates with the PM Engine to collect data on the user’s actions and movements. It also interacts with the PM Admin Interface because that’s where the GUI is created. The SQL database stores and retrieves data by communicating with the Audit class and GUI.

# Functional Requirements

## Statement of Functionality

The Audit System application will provide a tool for administrators to track users’ movements and actions.

### Administrators

* Ability to track movements through the system
* Completed look and feel
* Ability to display results that are stored in the database
* Ability to filter what results are displayed by what the user wants to observe
* Ability to use multiple filters at one time in order to display the desired results

### Users

* Ability to perform data collection

## Security

Only administrators are allowed to make audit queries.

## Auditing

All changes to the Audit System are recorded in a repository that the Policy Machine team maintains.

## Administration/Customization of the Application

There are no customization options at this time

## Reporting

This system reports:

* Session ID
* User ID
* User Name
* Host Name
* Time Stamp
* Action
* Description
* Results
* Object Name
* Object ID

# Features

## Completed Look and Feel

The GUI that was created for the Audit System, while simple it provides an interactive query screen and a clear and concise results screen. The results are displayed on a minimalistic screen making the results stand out. The search GUI provides the administrator a simple self-explanatory way of getting an audit report that has all the data the user wants.

## Ability to perform data collection

This feature is reliant on the connection to the SQL database server, the PM Engine class and Audit class. The data collected from the users that use the Policy Machine is:

* Session ID
* User ID
* User Name
* Host Name
* Time Stamp
* Action
* Description
* Results
* Object Name
* Object ID

## Ability to track a user’s movements and actions through a system

This feature if reliant on the communication between the PM Engine class and Audit class. The data that is collected by the Auditing System is:

* Tracking users when they log into the system, it also reports whether it was successful or not
* Tracks when a user opens or reads an object and reports whether it was successful or not
* Tracks when a user writes to an object.
* In later versions of the Audit System there will be additions made to what type of actions and movements are collected.

## Ability to display results that are stored in the database

In this feature the requirements is to be able to query the database. After receiving the data file manipulation is used to create an HTML file. This file that is created, is then automatically opened in the user’s default browser.

## Ability to filter what results are displayed by what the administrator wants to observe

This feature is made possible by complex and changing SQL statements that change depending on what the user wants reported on. The GUI makes it easy for the administrator to filter the data to get the specific information they want in the results page. The administrator can filter my all of the data collected and can even complete a more specific filters data queries. The more specific filters are the text boxes locate in the:

* User Name
* Host Name
* Timestamp (one instance and between 2 date/time)
* Action (create session, Object read, READ\_OBJ, Object write)
* Results (true or false)
* Object ID
* Object Name

For example if an administrator wanted to get the Session ID, User ID, User Name, Action and Results for every entry that matched User Name = “Bob” with this feature they could.

## Ability to use multiple filters at one time in order to display desired results

The filters that are mentioned above have the capability to be used at the same time. This allows the administrator(s) to results with multiple filters. For example an administrator could filter results by Username = “Alice” and Timestamp= “15/03/18 12:43:03” and it would return the results that matched those two filters.