

**Mitek ForgeRock**

**USER MANUAL**

**Document History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Description of versions / changes** | **Authors** | **Date** |
| 1.0 | Initial draft | Chetana Patil | 04/07/2022 |

**Review History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Comments** | **Reviewed by** | **Date** |
| 1.0 | Initial draft | Prabir Kumar Biswas | 05/07/2022 |

**Table of Content**

**1.Introduction 3**

**2.Authentication Nodes 3**

**3.ForgeRock Setup 4**

**4.Steps to deploy the Mitek jar**

**4.1 Commands to run the tomcat 8**

**5.Steps to create Authentication Tree**

**5.1 ForgeRock Login 8**

**5.2 Create Realm 10**

**5.3 Create Tree 11 6.Flow of different Authentication Trees 12**

**6.1 Passport 12**

**6.2 Passport with Selfie 15**

**6.3 DL/ID 16**

**6.4 DL/ID with Selfie 17**

**6.5 Passport and DL/ID 18**

**6.6 Passport with Selfie and DL/ID with Selfie 19**

## 

## **Introduction**

The purpose of this document is to provide a comprehensive guide on how to set up ForgeRock on-premises versions. Further it will guide the user to setup the Mitek Verification flow by creating an Authentication tree using various Mitek custom nodes.

## **Authentication Workflow Setup**

Below mentioned details will be covered as part of this document

* ForgeRock setup
* Login to ForgeRock
* Create Realm
* Create Tree under Authentication trees
* Create different Authentication Trees

## 

**ForgeRock Setup**

### **Prerequisites**

1. **Tomcat 9 Setup -**

* Download Tomcat (apache-tomcat-9.0.60.zip): Please download the apache tomcat zip/tar specific to the OS Platform.

Extract the Zip/tar file -

**Command** -

***sudo unzip apache-tomcat-9.0.60.zip***

***tar xvfj apache-tomcat-9.0.60.tar -C /tomcat***

(extract the files to /tomcat folder or any other location where you like to set up the tomcat).

* Change the permission of the Tomcat “bin” folder.

**Command**:

***sudo chmod +x /bin/\****

* Tomcat Environment Setup.

Go to the “bin” folder and edit the setenv.sh file (if no setenv.sh file present then create one as follows)

**Command**:

***cd /bin***

***sudo vi setenv.sh***

**Add following contents to the setenv file**

JAVA\_HOME=<Paste the path of your java home> (for example - /usr/lib/jvm/java-1.11.0-openjdk-amd64).

CATALINA\_OPTS="-Xms1024m -XX:MaxPermSize=256m"

So it should look something like this -

*JAVA\_HOME=”/usr/lib/jvm/java-1.11.0-openjdk-amd64”*

*CATALINA\_OPTS="-Xms1024m -XX:MaxPermSize=256m"*

* Update Tomcat user config file -

Open the Tomcat config file located at “/tomcat/conf” folder and add below lines.

*<role rolename="manager-gui"/>*

*<user username="tomcat" password="12345" roles="manager-gui"/>*

*</tomcat-users>*

1. **Setup JDK -**

**Command** -

***sudo apt update***

***sudo apt install default-jdk***

***java -version***

### **Setup OpenAm**

1. Download OpenAM (7.1.2)-

(<https://backstage.forgerock.com/downloads/browse/am/featured>)

1. Unzip the OpenAM zip file into a folder. And locate the file “AM-7.1.2.war”.
2. Rename the war file name to “openam.war”.
3. Copy the “openam.war” file to the Tomcat “webapps” folder (/tomcat/webapps)

**Command**:

***sudo cp AM-7.1.2.war /tomcat/webapps/openam.war***

1. Now restart the Tomcat server -

**Command**:

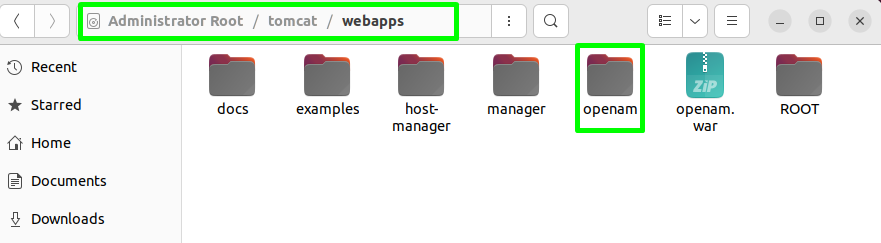
***sudo /tomcat/bin/shutdown.sh***

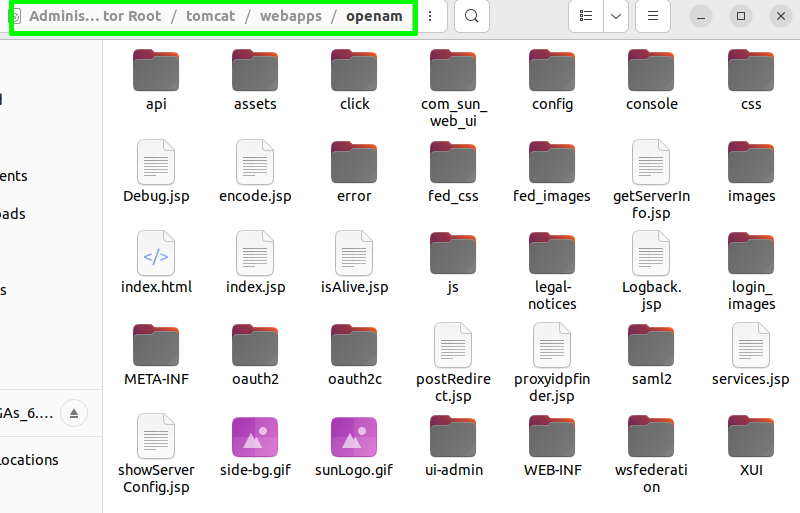
***sudo /tomcat/bin/startup.sh***

1. Once the Tomcat service restarted, go to the OpenAM setup location and verify if all the folders got generated properly.

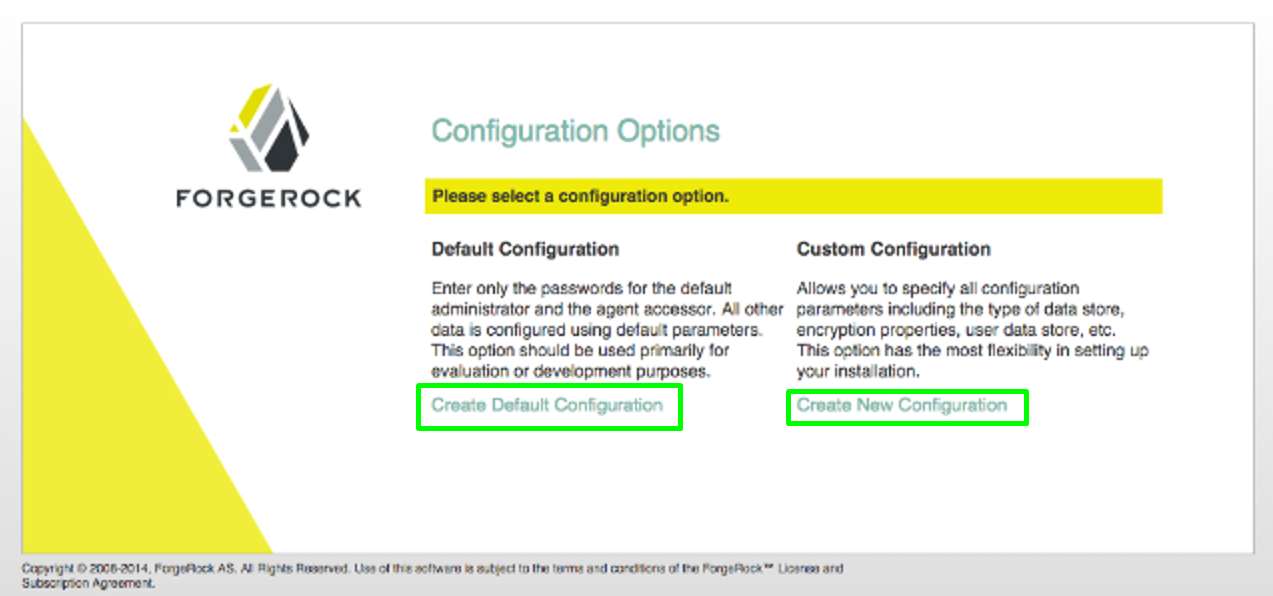
Location - /tomcat/webapps/

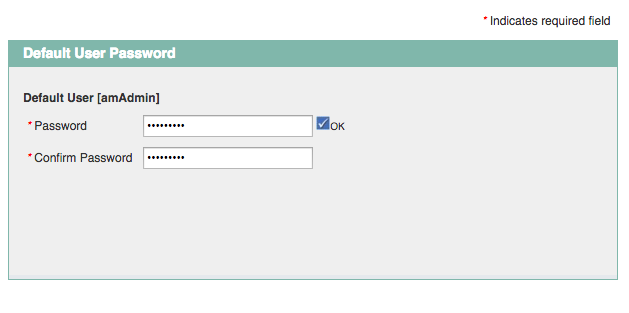
You will see an openam folder got created as shown below -





1. Open the Openam link - <http://localhost:8080/openam>
2. When you open the openam for the first time, it will ask for configurations. You have two options, you can go with the default configuration or create a new configuration. For simplicity, you can choose the default configuration where you just need to provide the AM Admin password.





1. Now proceed to the openam login screen. Open the openam base URL again “<http://localhost:8080/openam>”. Use the credentials you provided during the configuration setup.
2. Here is a Video Tutorial Link for further reference -

<https://www.youtube.com/watch?v=5X1cWnMDtH0>

**Steps to deploy the Mitek jar**

* mitek.zip →extract and paste the folder into tomcat webapps folder
* mitek jar file→paste it in “tomcat/webapps/openam/WEB-INF/lib”

**Commands to run the ForgeRock as localhost**

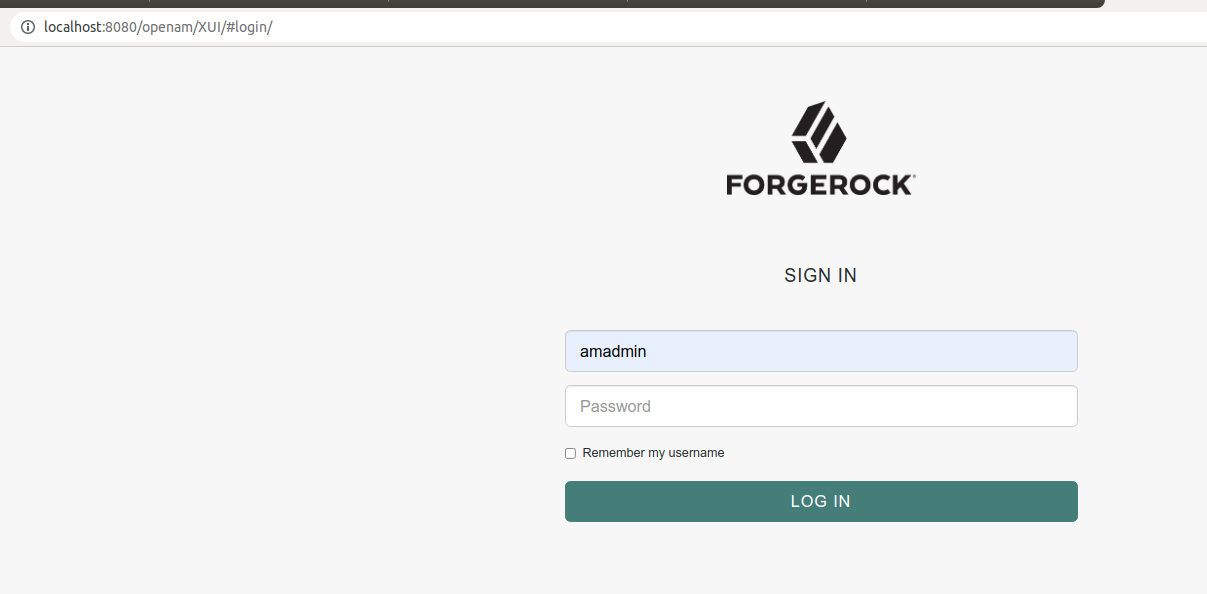
* cd /tomcat/bin
* sudo /tomcat/bin/startup.sh or sudo ./catalina.sh run(this command will start the tomcat instance in the terminal itself. Thus you will be able to view the logs as well)

**Steps to create Authentication Trees**

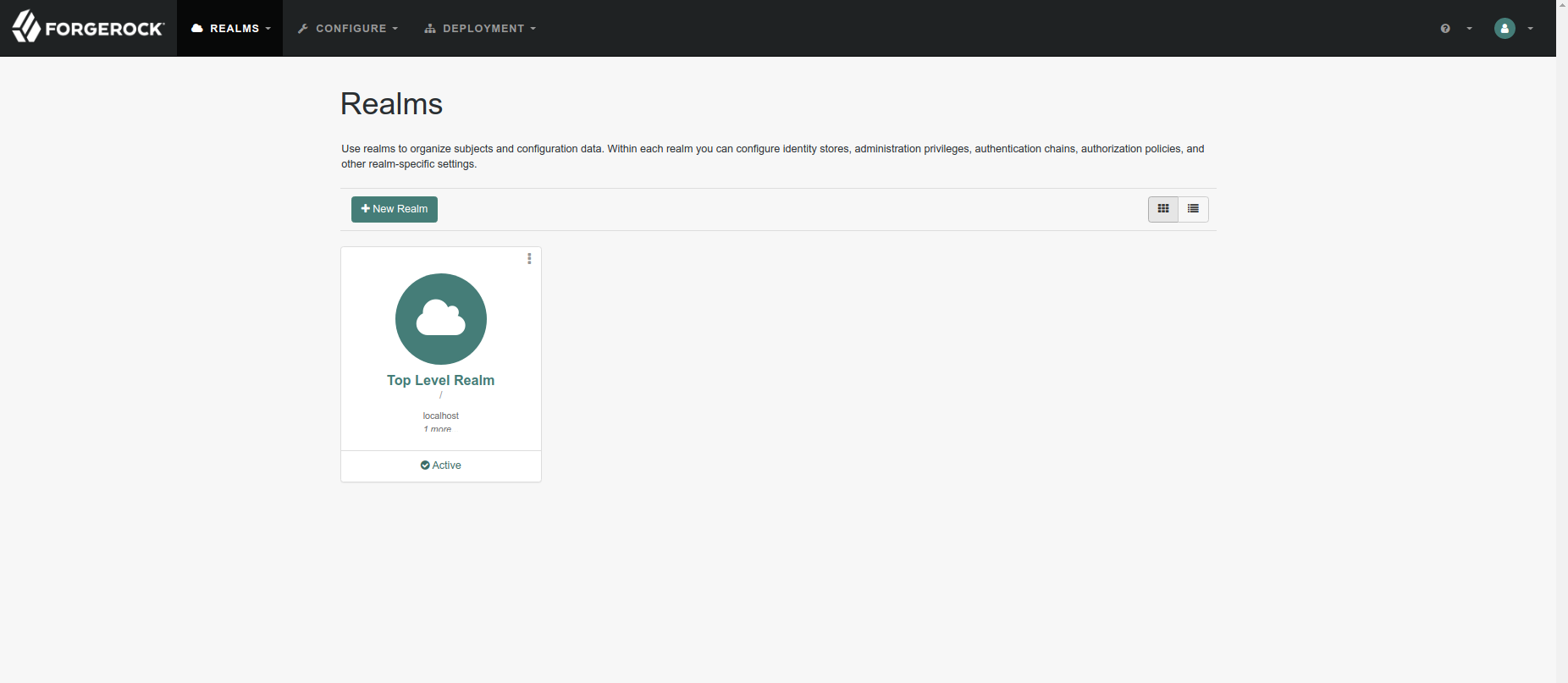
* **Login to ForgeRock:**

Once you launch the URL, it will take you to the below mentioned home page, enter the User Name and Password & click the “Login” button.

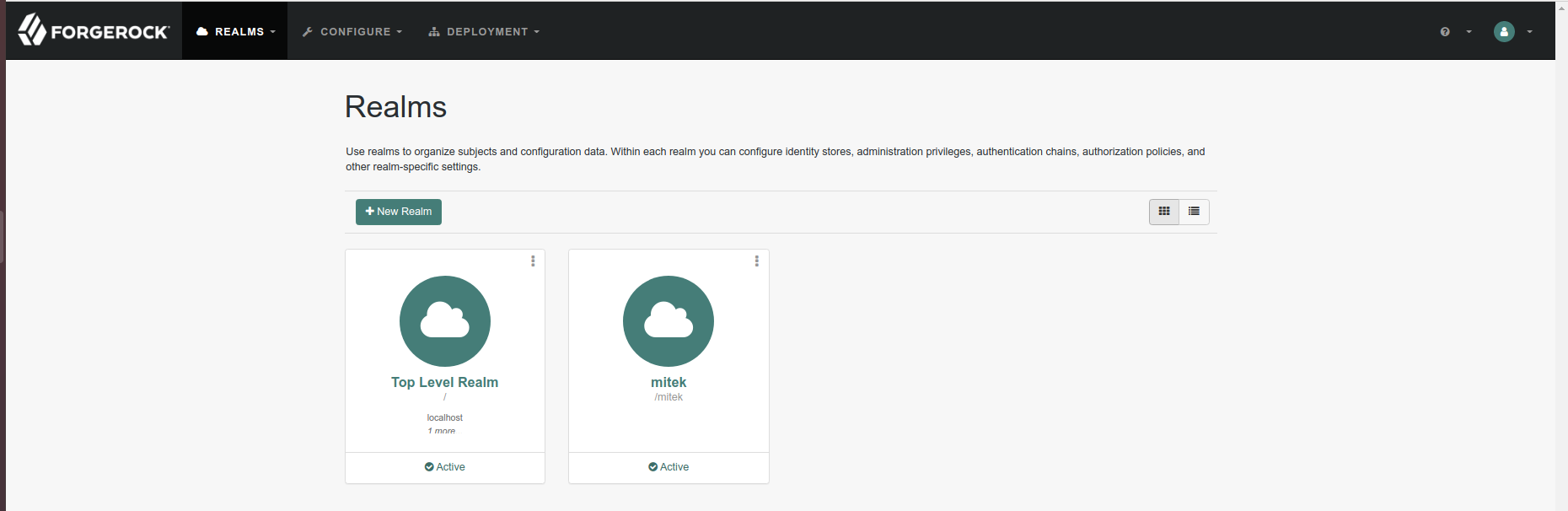
<http://localhost:8080/openam>



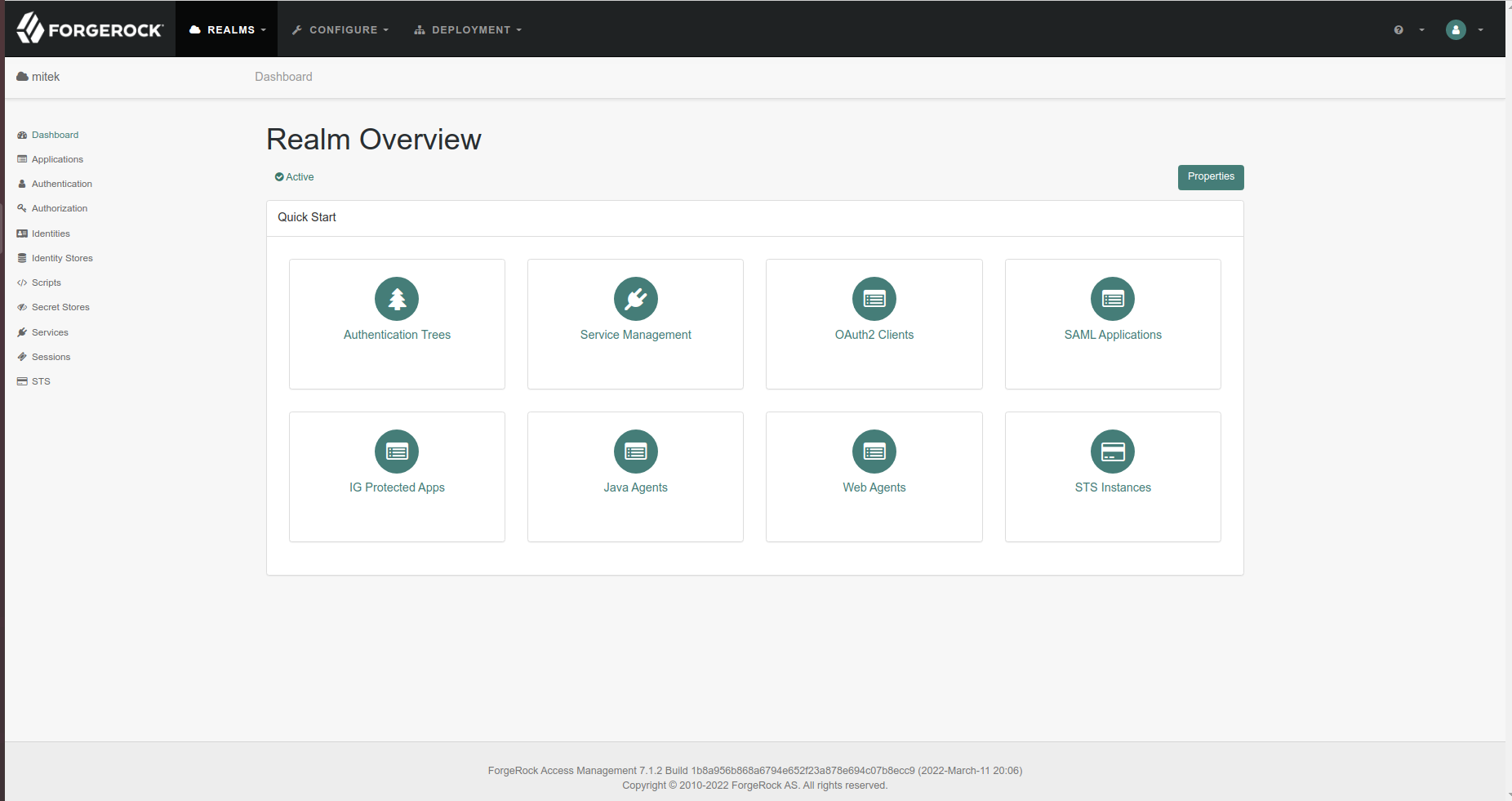
* Click on New Realm



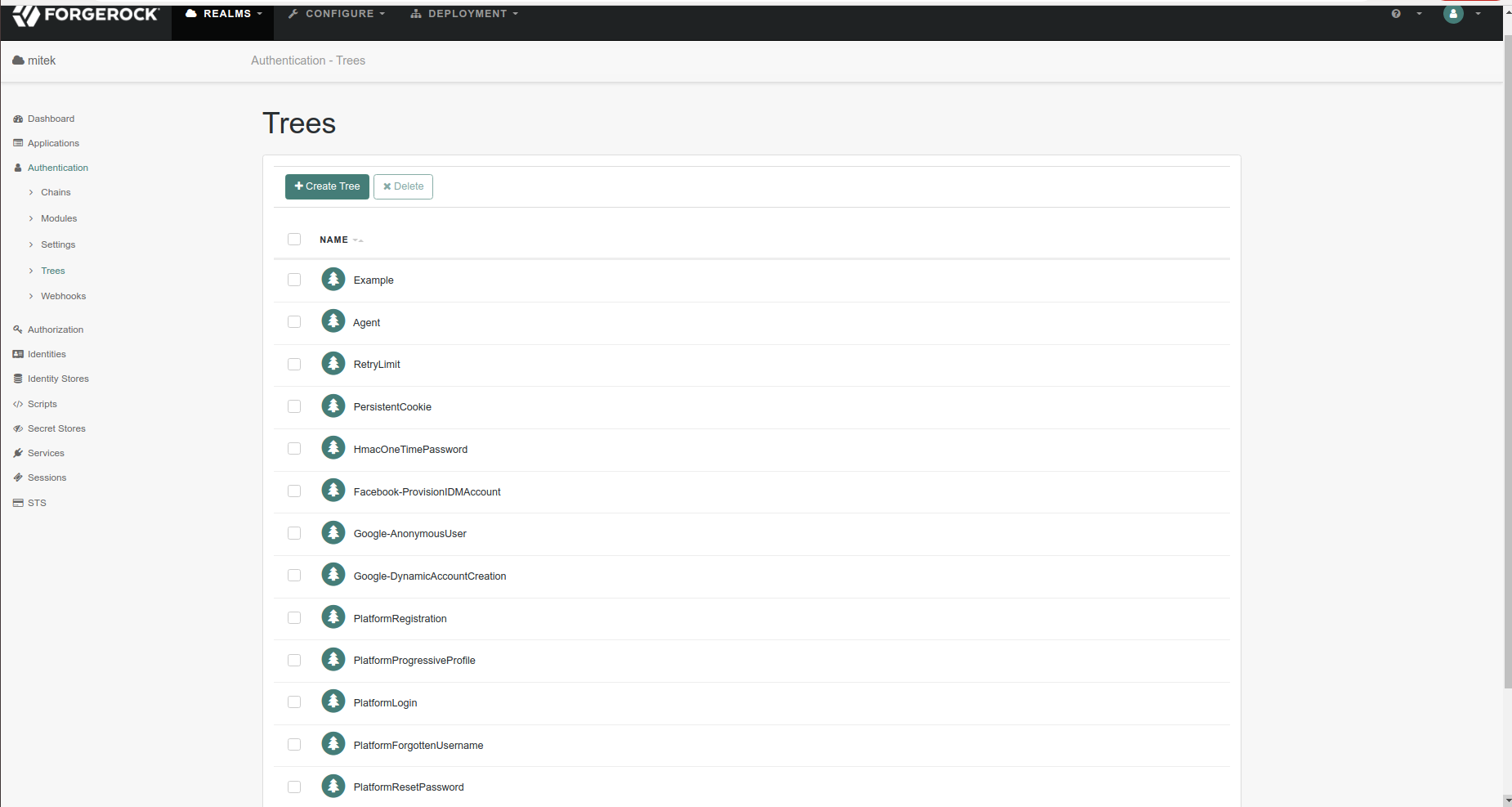
* Create Mitek Realm



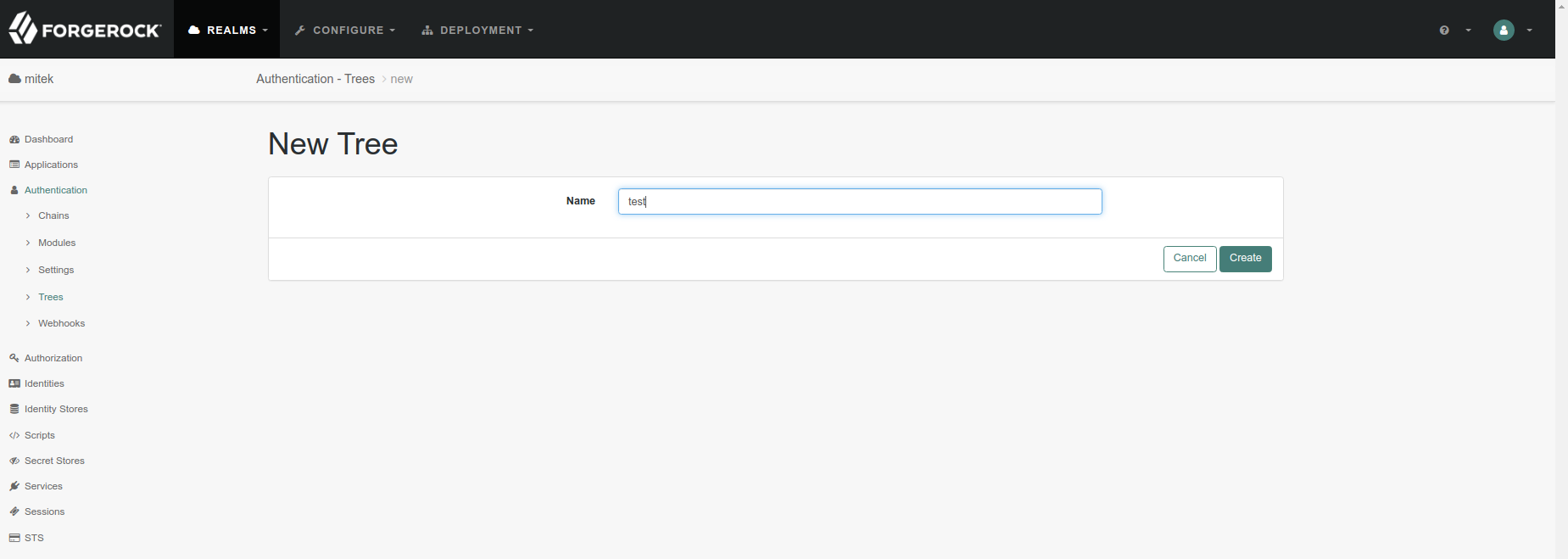
* Go to the Mitek Realm and you will get the below page with Default tress created



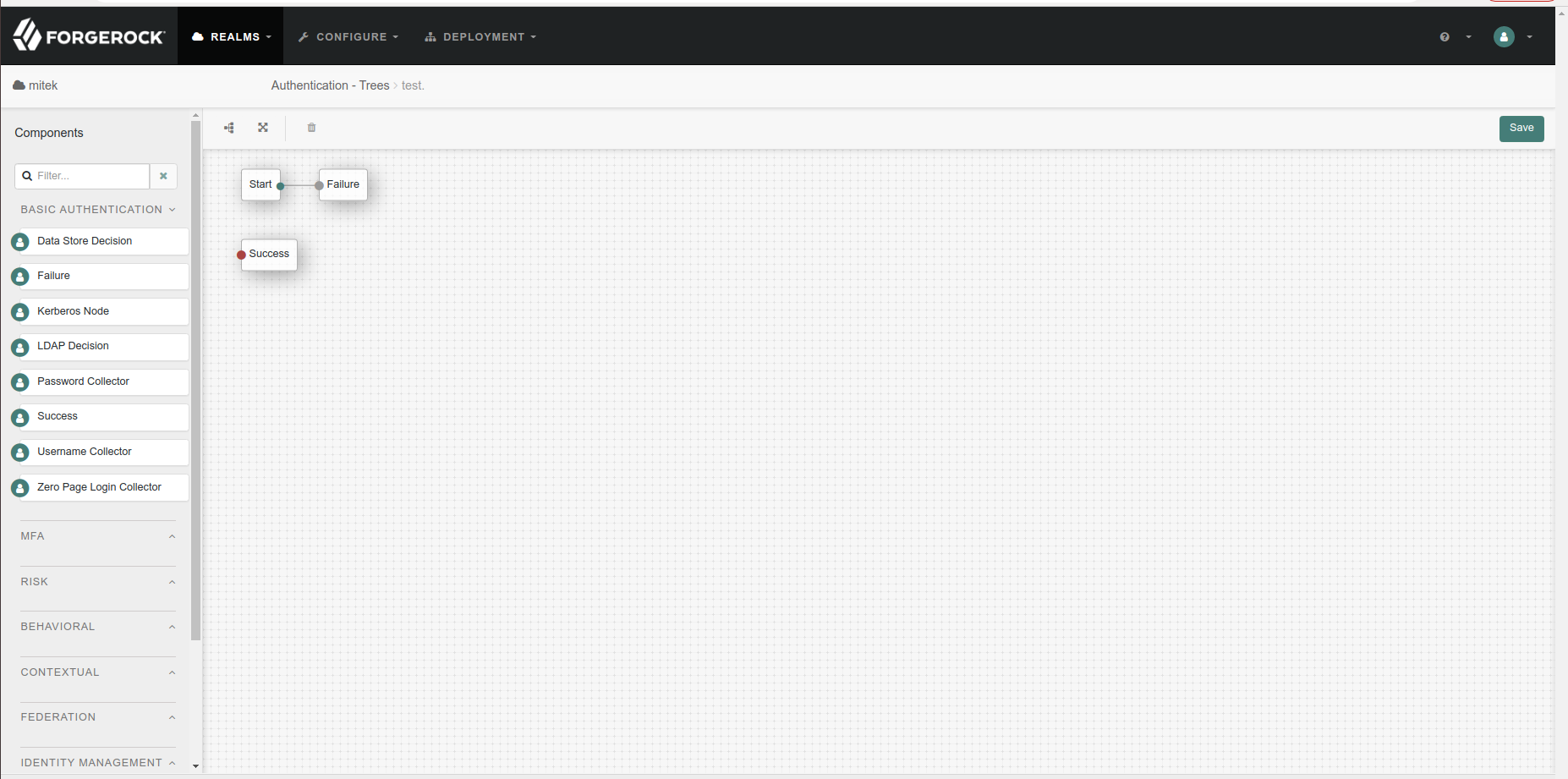
* Click on the Authentication Trees and you will get below page



* Click on Create Tree and enter the Tree Name and click on create

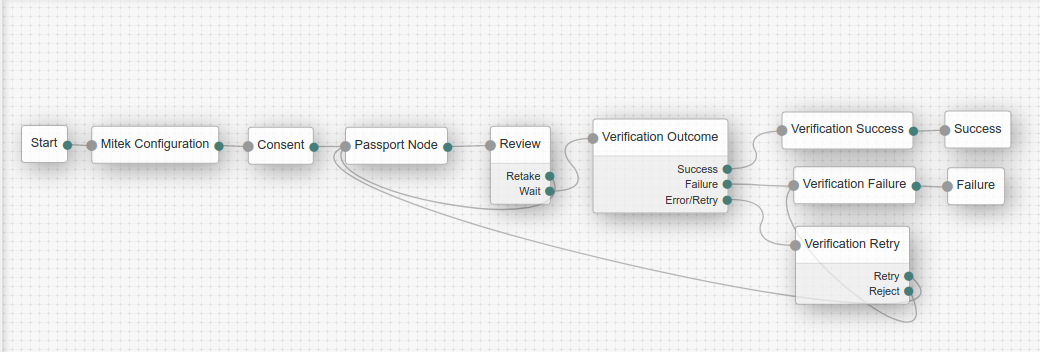


* You will get the below page when you create the Tree and now you can drag and drop the Nodes to create Authentication tree

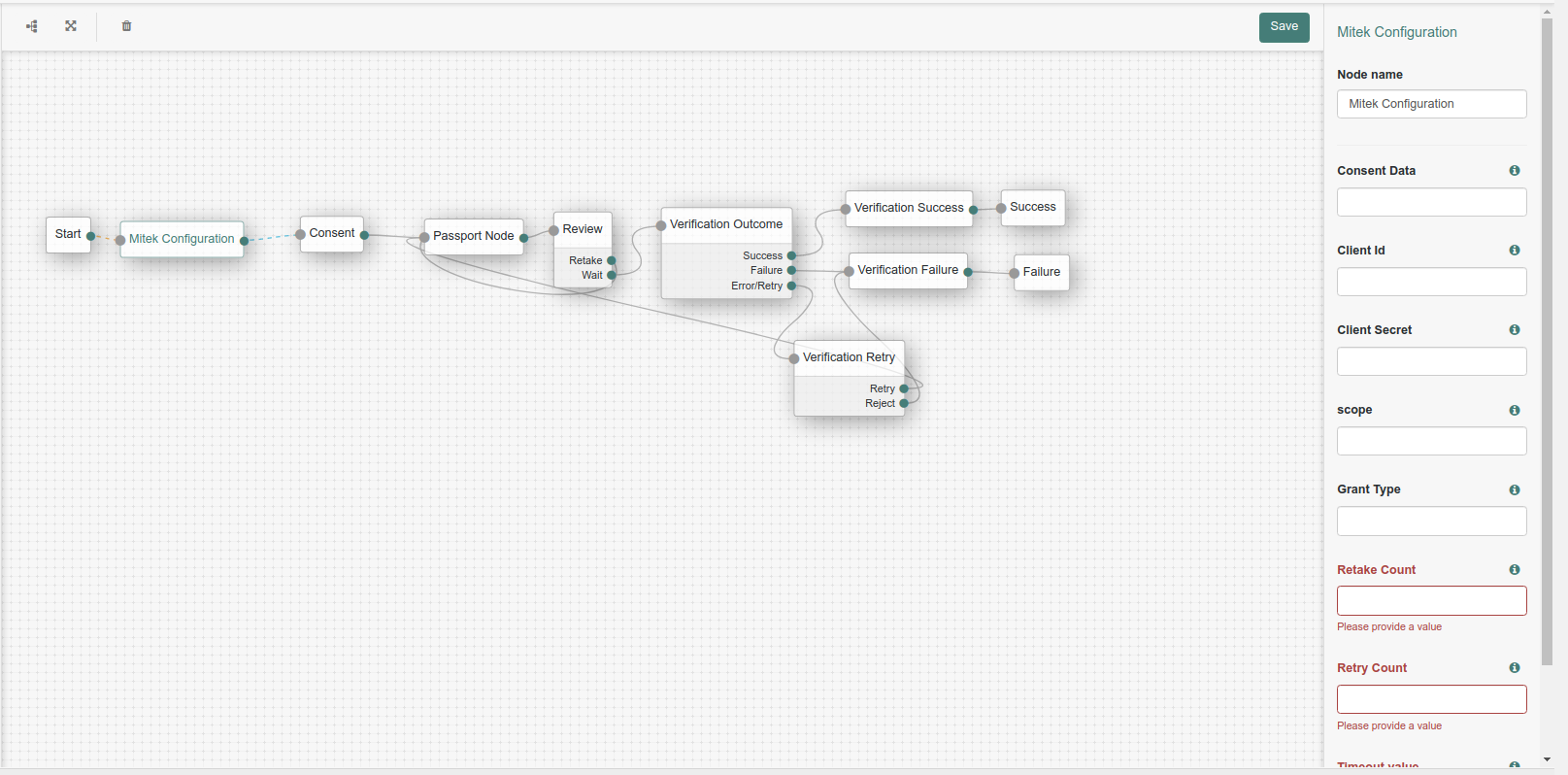


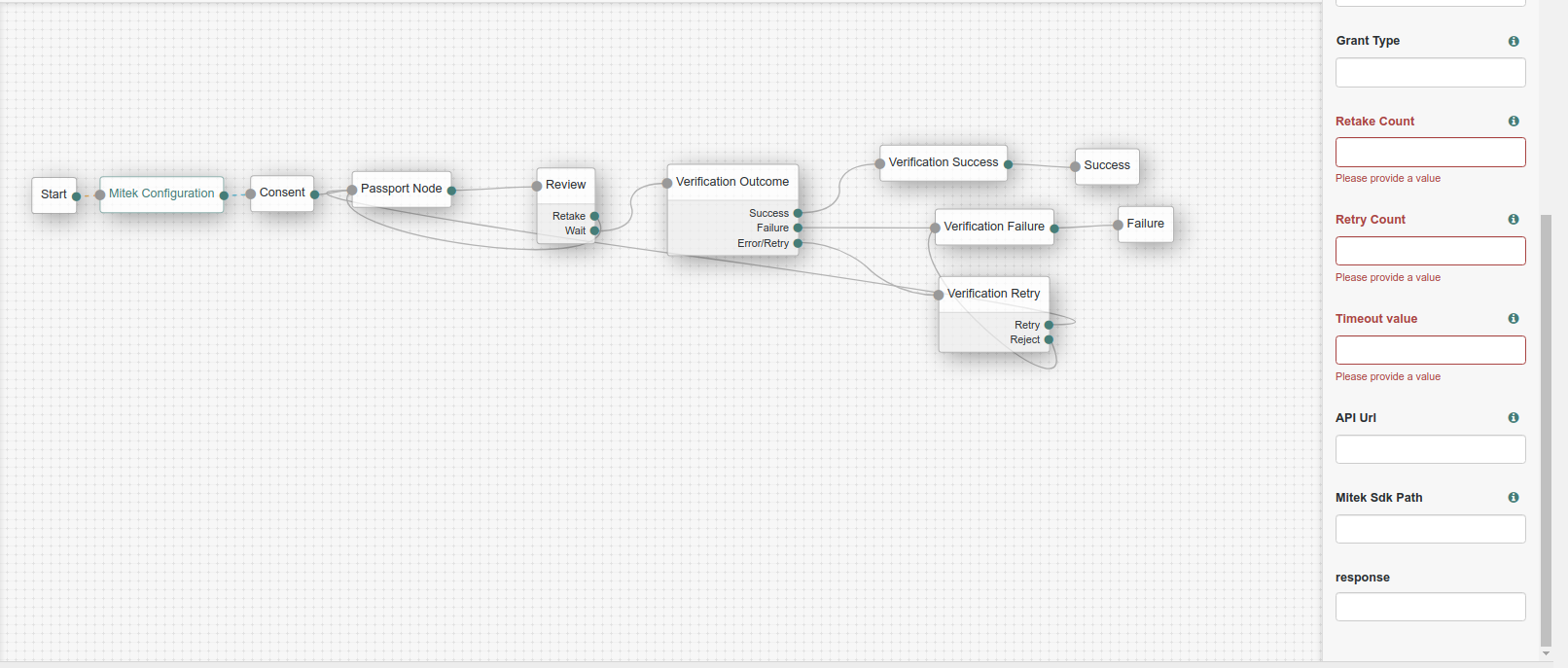
* **Flow of different Authentication trees**

**1.Passport**



Pass the configuration variables in the given fields and click on save





**Configuration Variables that needs to be passed in Mitek Configuration Node**

**Consent data:**The consent is a block of text customizable by the customer. If empty, the consent step is skipped.

**For example**:Terms and conditions...\nI agree to the terms and conditions

**Client Id:**client\_id

**Client Secret:**client\_secret

**Scope:**scope

**Grant Type:**grant\_type

**Retake Count:**3

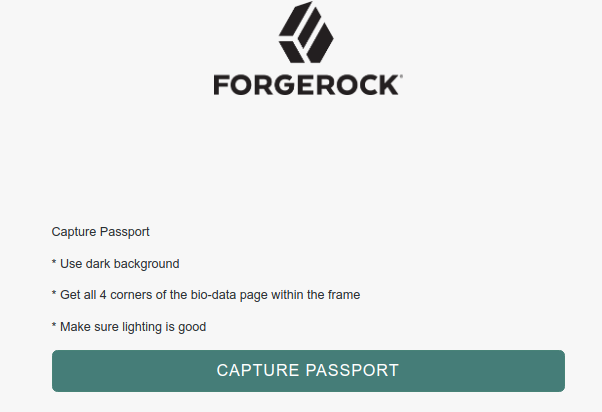
**Retry Count:**3

**Timeout Value:**30 (default value)

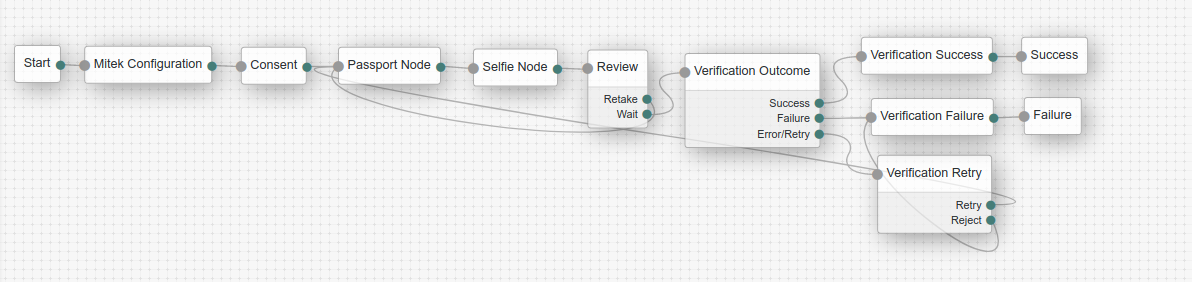
**Api Url:**https://testurl.com

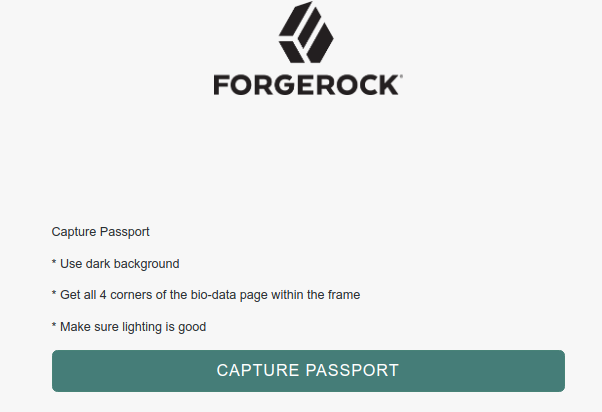
**Mitek Sdk Path:**/mitek/

**response :** User custom attribute field name which is created to hold the Verify response (JSON Object)

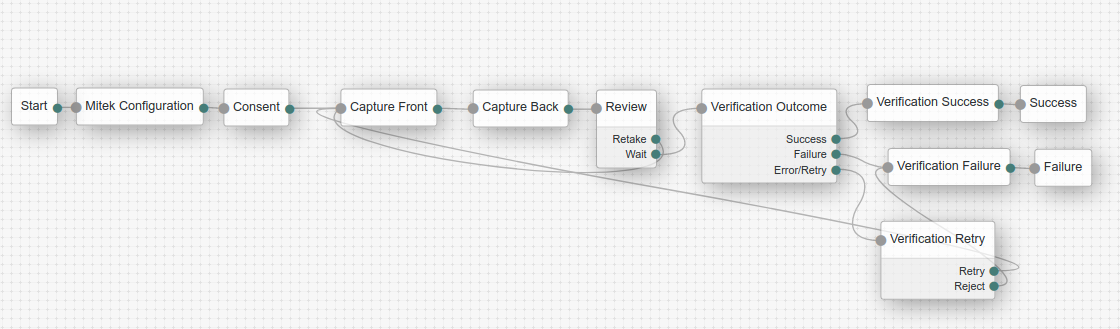


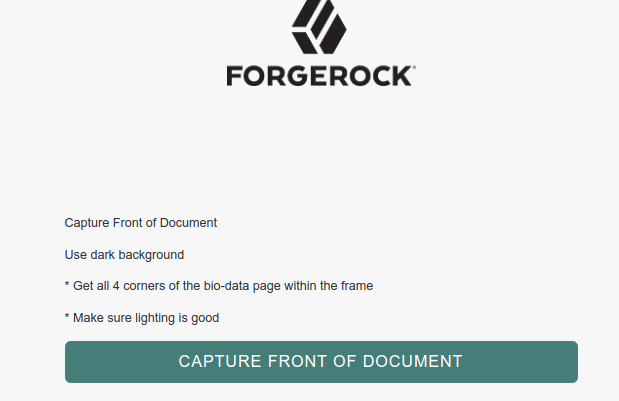
**2.Passport with Selfie**

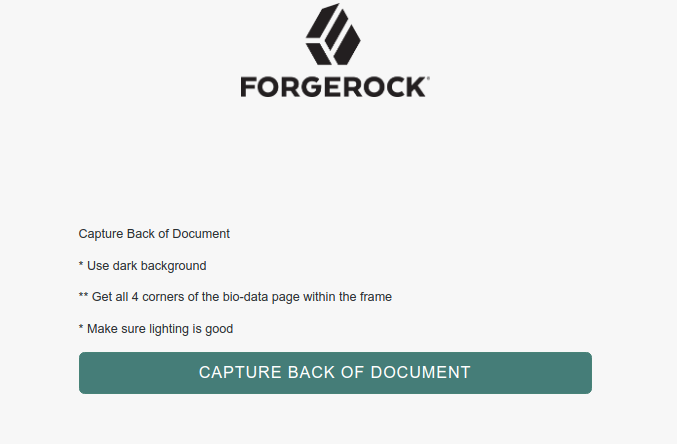




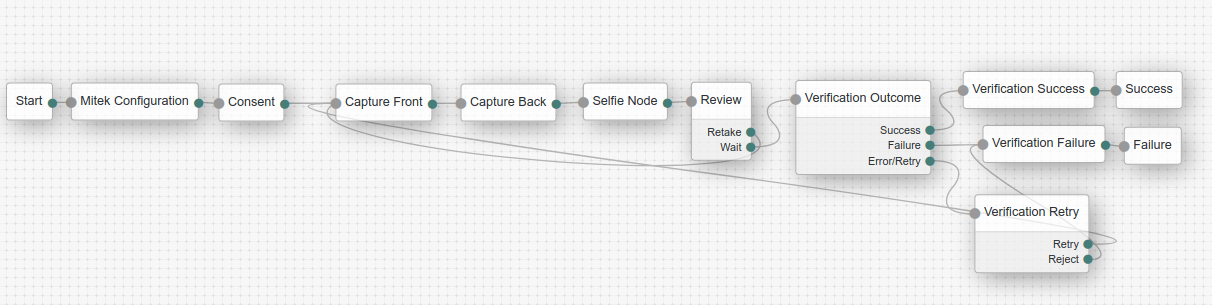
**3.DL/ID**

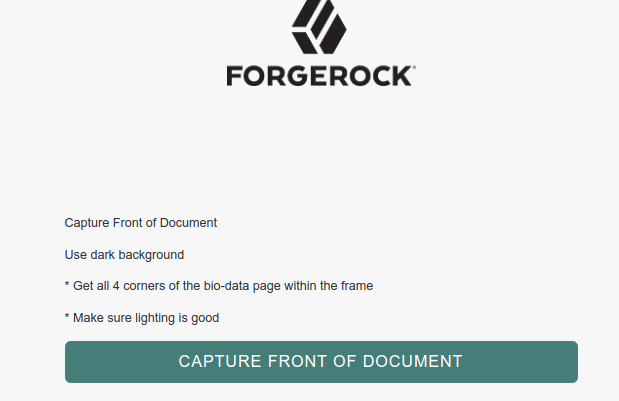




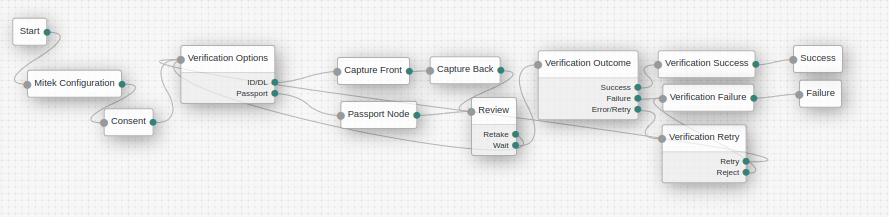


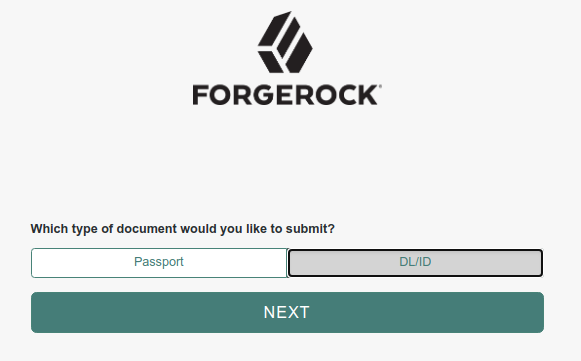
**4.DL/ID with Selfie**





**5.Passport and DL/ID**





**6.Passport with Selfie and DL/ID with Selfie**

