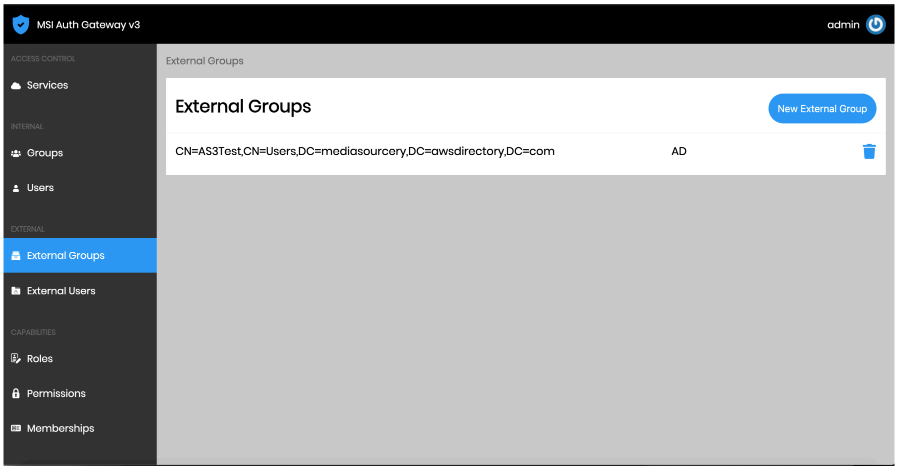
Auth Service with Blockstack

Introduction



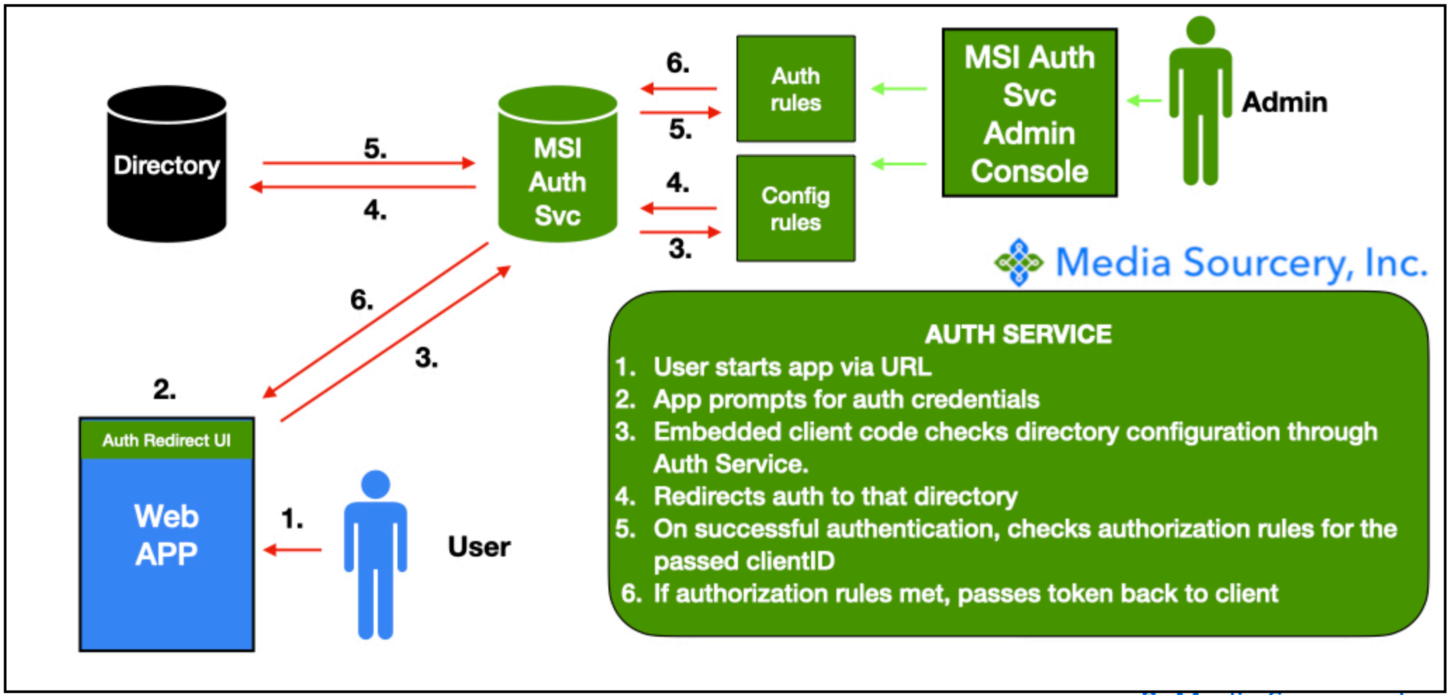
External Groups

CN=AS3Test,CN=Users,DC=mediasourcery,DC=awsdirectory,DC=com	AD

Media Sourcery Inc.'s (MSI) flagship Workflow As A Service product is used by enterprises to automate the tasks and sub-tasks that enable a process to complete quickly and efficiently. Many of these tasks and sub-tasks are automated by applications, some of which are written by MSI as workflow automation apps. MSI's Auth Service V2 product is utilized by enterprises to segment application usage by identifiable user segments. The initial usage has been to utilize X.500 hierarchies (such as those employed by Microsoft's Active Directory) to limit or include access. Version 3 of this product continues support of this type of usage, with an updated UI and other enhancements. For full details on the additional features of the Auth Service V3 product, see the Version 3 Data Sheet at www.mediasourcery.com/authorization.

Businesses by their nature require external entities (partners, users, customers) to be authenticated and authorized for certain functions. Media Sourcery is constantly adding support for external authentication mechanisms. With Version 3 of our Auth Service product, this support includes email domains and blockchain-based identity and access management(IAM) systems.

The first blockchain identity system to be added is Blockstack.



AUTH SERVICE

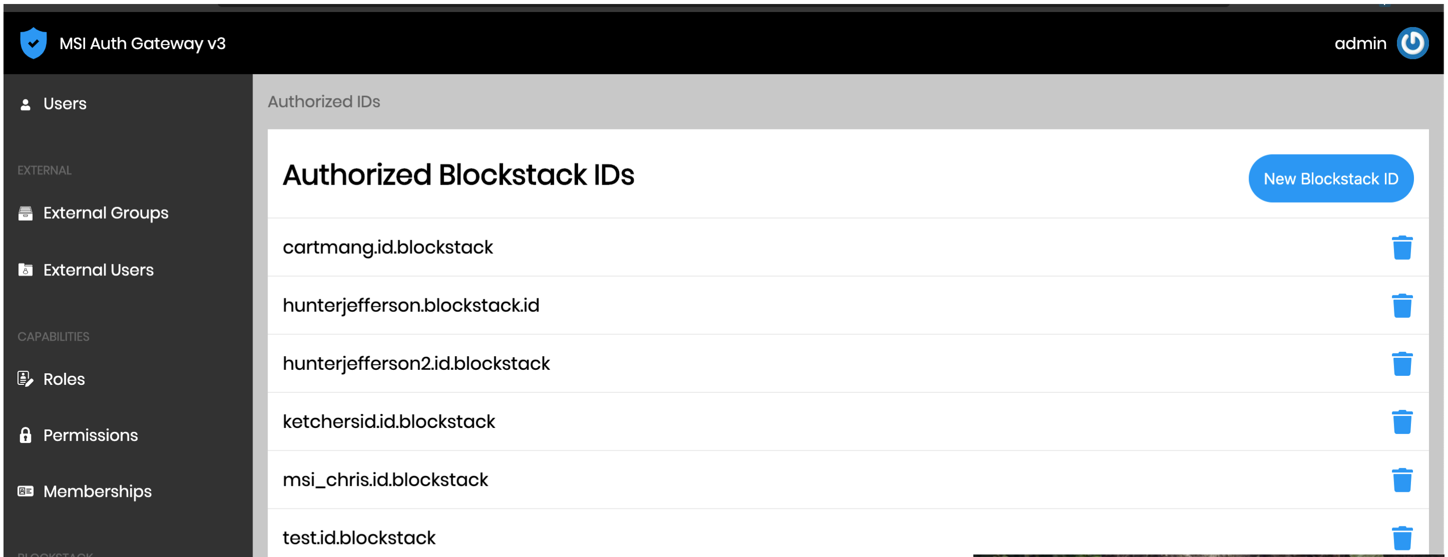
1. User starts app via URL
2. App prompts for auth credentials
3. Embedded client code checks directory configuration through Auth Service.
4. Redirects auth to that directory
5. On successful authentication, checks authorization rules for the passed clientID
6. If authorization rules met, passes token back to client

Auth Service with Blockstack

Review of how Auth Service works

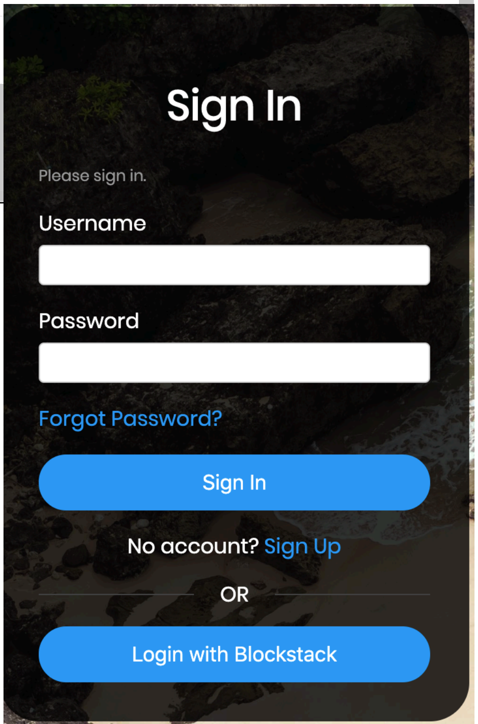
Applications enable the use of the auth service with javascript redirection code. When a user authenticates to an Auth Service enabled application, the authentication transaction (with the application ID of the application whose usage is being requested) is passed to a configured directory (e.g., Active Directory). If authentication is successful the application ID passed in the request is used to look up the configured rules for authorization for the app that has that application ID. For instance, perhaps only users in o="admin" and specific "uid" users can access an app. If the authenticated user matches the permissions configure the request receives a token. This token is passed back to the app. It can then be used for other authorizations depending on time-to-live and other validity rules.

Blockstack Integration



Authorized Blockstack IDs

cartmang.id.blockstack
hunterjefferson.blockstack.id
hunterjefferson2.id.blockstack
ketchersid.id.blockstack
msi_chris.id.blockstack
test.id.blockstack
testuser67.id.blockstack



There are many reasons why Blockstack is the first of several blockchain IAM systems that will be integrated into this product:

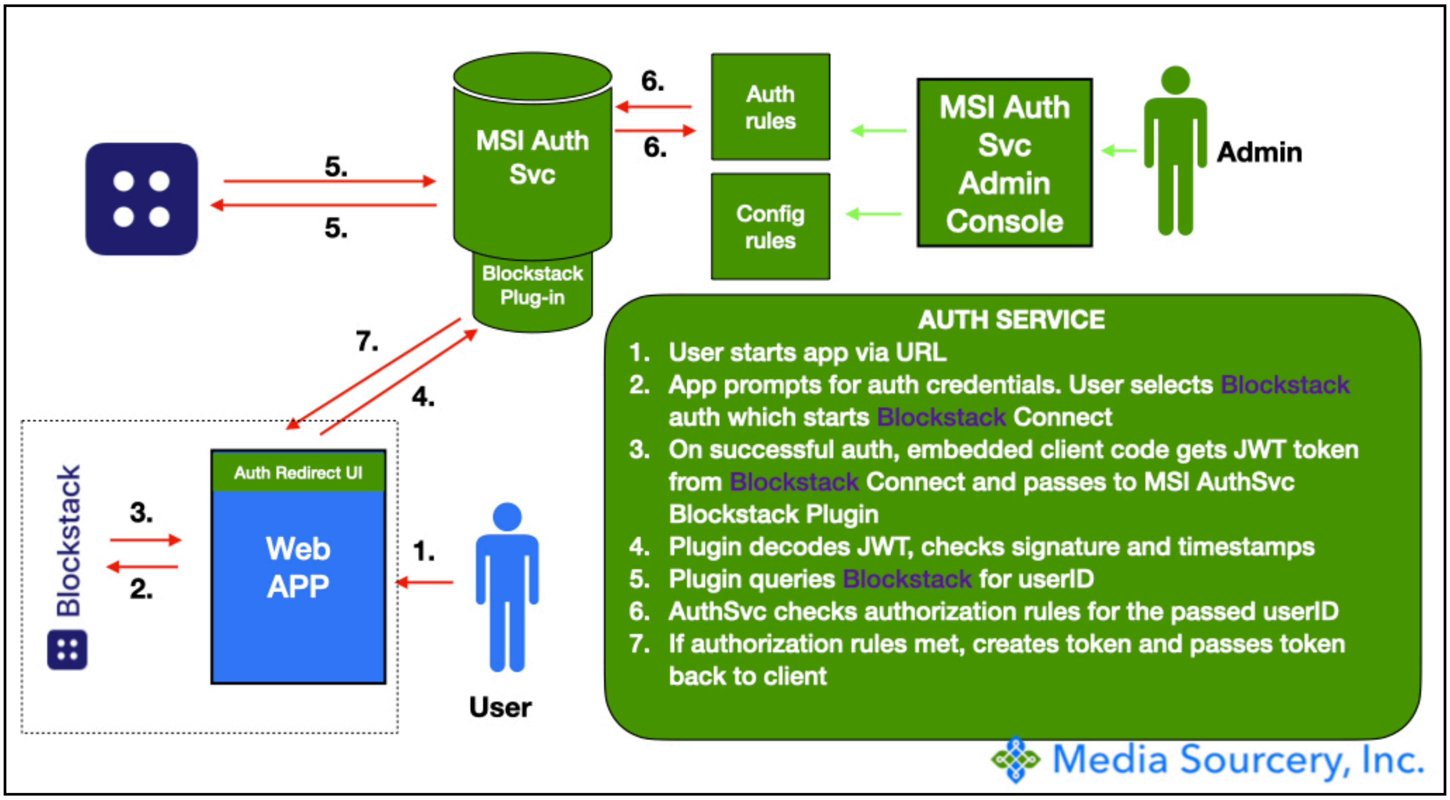
- Blockstack has an active and supportive developer community, including Blockstack employees and external developers.
- Blockstack's 1.0 blockchain is in production since 2018 and has a wide list of dApps written for it.
- Blockstack's 2.0 release is as of this writing in the testnet phase, providing a rich environment for testing.
- The Blockstack token sale in 2019 became the first SEC qualified

crypto token sale in US History.

- Blockstack's Stack blockchain operates as a "virtual blockchain" on top of the Bitcoin blockchain. The goal is to integrate with best in class IAM systems that work on top of the largest blockchain system.

Instead of authenticating with an enterprise directory, such as Microsoft's Active Directory, with this release a user can chose to authenticate using their Blockstack ID. This authentication is passed to the Media Sourcery Auth Service, which, in the same way that the current version works with Active Directory authorizations, will determine if the authenticated user has been authorized to use an app configured in the Media Sourcery Auth Services system.
The main updates in version 3 of the Auth Service app to support this are as follows:

- The embedded client code supports the ability for the user to choose to login with their Blockstack ID.
- The code will then launch the local Blockstack Connect client (or prompt the user to install the Blockstack Connect client) for the user to authenticate using Blockstack Connect.
- In the Auth Service back end, the JWT token that is received from Blockstack connect is verified. This backend code was created in Scala by Media Sourcery (with support from the Blockstack Discord community) and will eventually be donated back to the community.



AUTH SERVICE

1. User starts app via URL
2. App prompts for auth credentials. User selects Blockstack auth which starts Blockstack Connect
3. On successful auth, embedded client code gets JWT token from Blockstack Connect and passes to MSI AuthSvc Blockstack Plugin
4. Plugin decodes JWT, checks signature and timestamps
5. Plugin queries Blockstack for userID
6. AuthSvc checks authorization rules for the passed userID
7. If authorization rules met, creates token and passes token back to client

This makes the Media Sourcery Authorization Service an identity and access management bridge between enterprise directories authentication/authorization systems and blockchain-based identity systems.

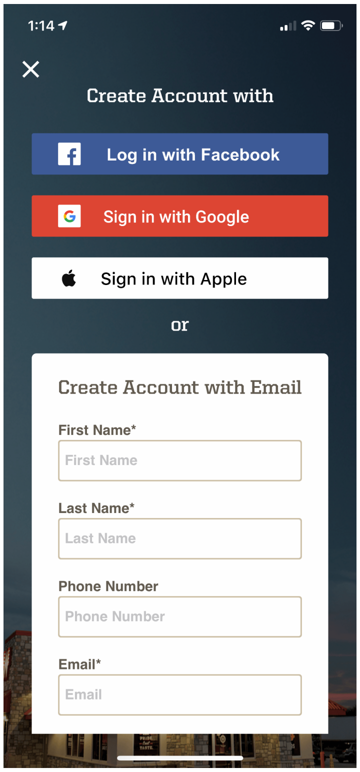
Blockstack Integration Use Case

Rules for certain client applications can include authorization for particular Blockstack IDs as well as those from an internal directory or an email domain.

An example use case is represented by our healthcare customers. Many healthcare workflows are supported through documentation. This documentation can come from referral partners (e.g., manufacturers in the supply chain), physicians, patients, payers or others. A simple document upload app can be used by all, but each constituency would utilize its own authentication mechanism; and the company's authorization mechanism should accommodate this.

A representation of users and usages of this document uploader could be:

- Internal employees uploading documents received via fax or email. These users are authentication via internal Active Directory and are authorized via inclusion in a group defined by the Active Directory hierarchy and authorization of members of that group to the uploader.
- Referral partners uploading documents directly or through API. These users are authenticated via an email domain to authenticate only users who have a business email from these partners. Only configured domains would be authorized to access the uploader app.
- Physicians uploading documents directly or through API. These documents could come from physician electronic medical records (EMR) system or uploaded directly. The users may be authenticated and authorized individually or as part of group by email domain.
- Patients uploading document directly.


1:14

✕

Create Account with

Log in with Facebook

Sign in with Google

Sign in with Apple

or

Create Account with Email

First Name*

First Name

Last Name*

Last Name

Phone Number

Phone Number

Email*

Email

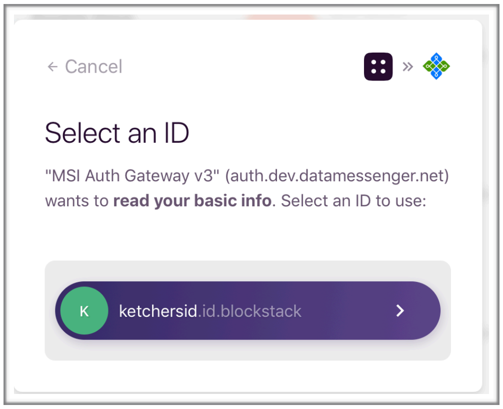
In current state, for patients (and for some physicians) authentication can take place with a variety of CENTRALIZED proxy services:

- Log in with Facebook
- Sign in with Google
- Sign in with Apple
- Utilize the user's email address.

This last choice is still centralized, controlled by either the email service provider (i.e., AT&T, Amazon's AWS, etc) or some other centralized company that has control over this authentication mechanism.

Utilizing any of these systems implies that the user (in this case the patient or the physician) trusts these large centralized companies.

Truly decentralized blockchain authentication system allow the user to choose to manage their own identity and their own data, authenticating with an identity that they control.


← Cancel

Select an ID

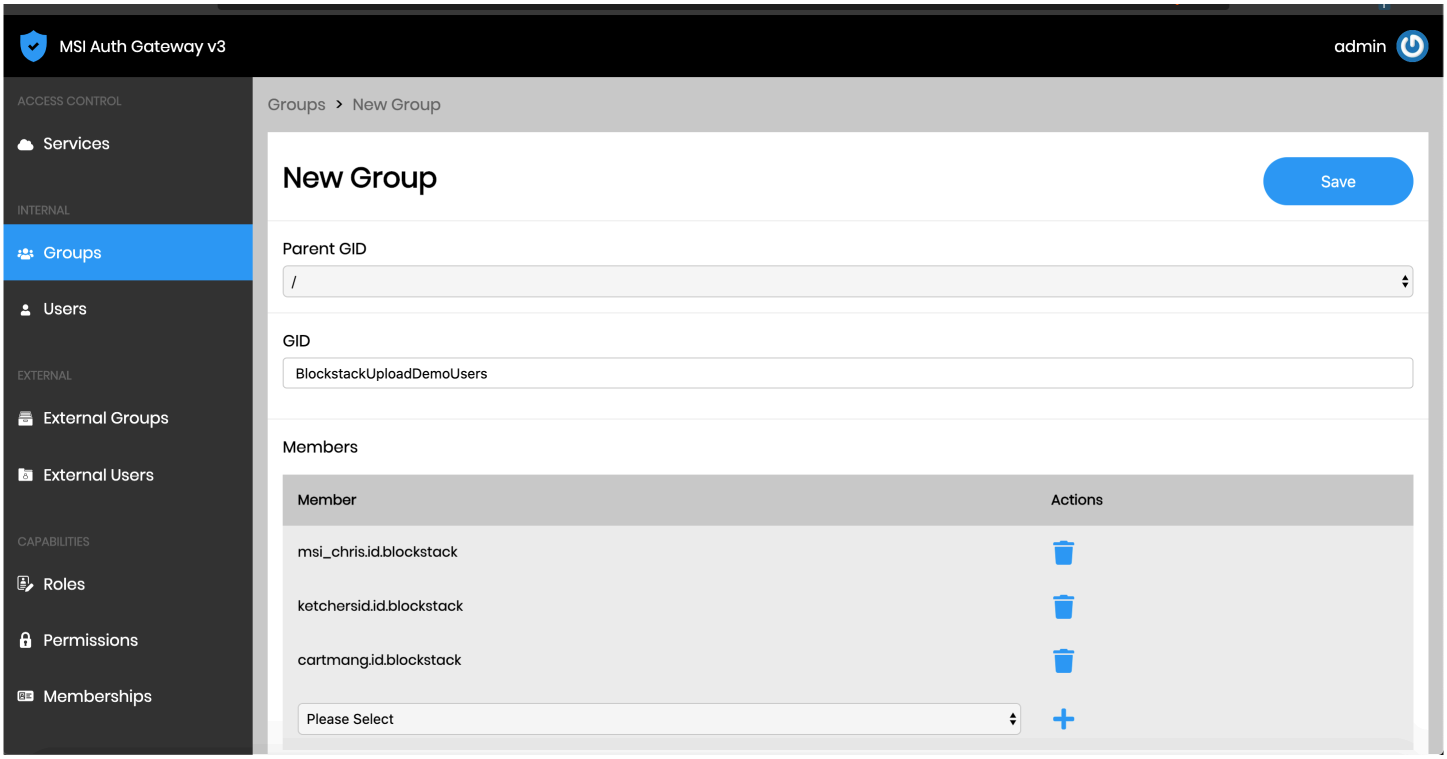
"MSI Auth Gateway v3" (auth.dev.datamessenger.net) wants to **read your basic info**. Select an ID to use:

K ketchersid.id.blockstack ›

Auth Service with Blockstack

If a patient is requested to provide a document for some healthcare process, with this new support of blockchain-based identity systems, they can now choose to use a decentralized authentication mechanism like Blockstack. The user would start an application to upload a document, and be presented with that option to authenticate with Blockstack. After authenticating, the token provided by Blockstack would be presented by the app to the Auth Service.

The Auth Service would validate the token and then determine if that Blockstack ID is authorized to use the document upload application.



New Group

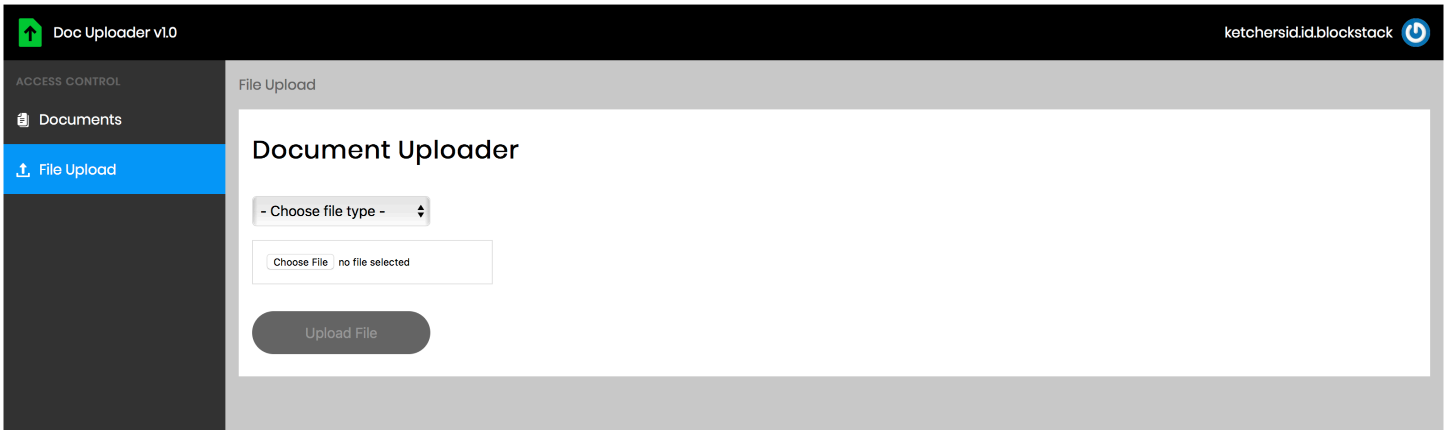
Parent GID: /

GID: BlockstackUploadDemoUsers

Members

Member	Actions
msi_chris.id.blockstack	🗑
ketchersid.id.blockstack	🗑
cartmang.id.blockstack	🗑
Please Select	+

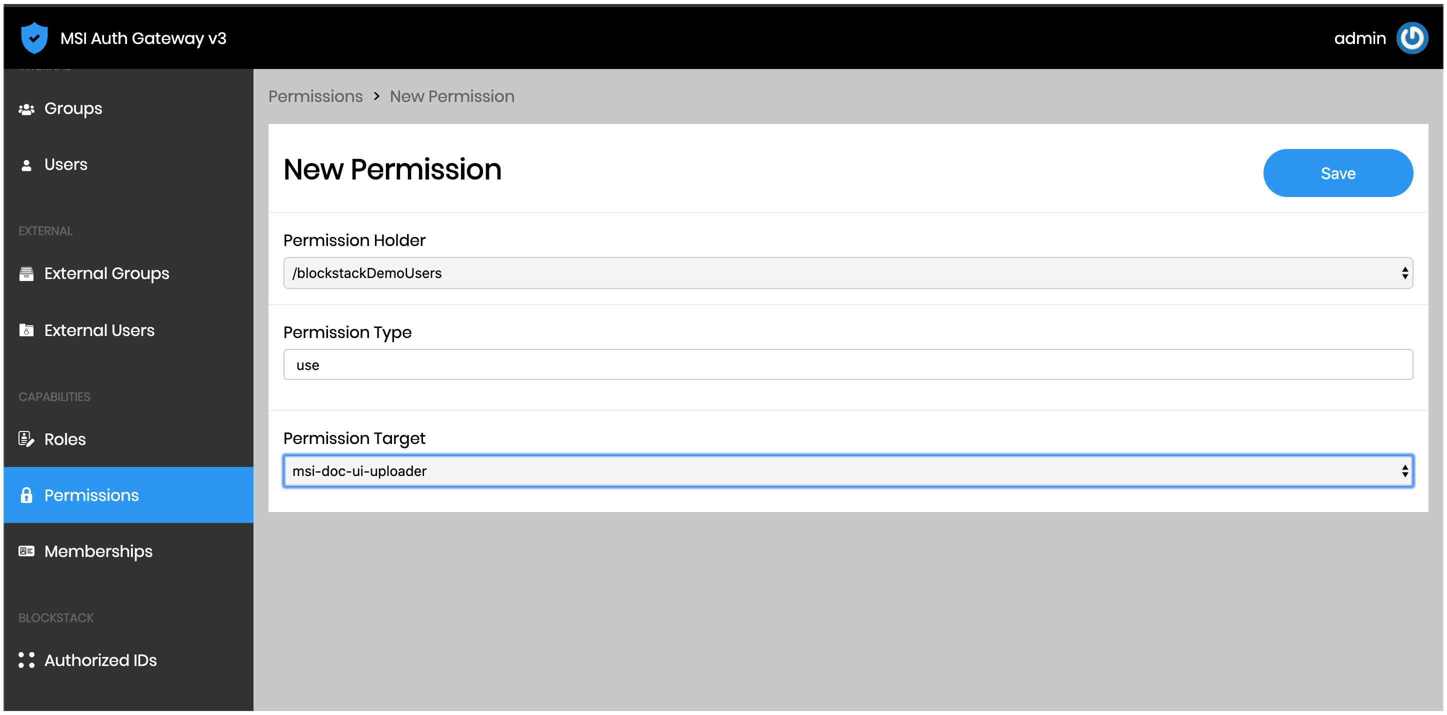
This opens up an application such as this document uploader to be utilized by existing Active Directory users, decentralized Blockstack ID users, allowed and registered email domains and others in the future.



Document Uploader

- Choose file type -

no file selected



New Permission

Permission Holder: /blockstackDemoUsers

Permission Type: use

Permission Target: msi-doc-ui-uploader

The Auth Service app allows for the association of groups of users (blockstack, internal, email and external directory users) into a named group.

An app thus enabled with the Auth Service V3 can allow access to traditional systems and blockchain based dApps. In the document upload app example, the Blockstack ID user (or, in the future, other blockchain identity systems) could keep and store their healthcare documents in a distributed system like the Blockstack-enabled Gaia system. If a user authenticated with Blockstack to this upload app, both their local storage and Gaia could be presented as document retrieval areas. If a non-blockstack user authenticates, this option would not be shown.

For more information or to request access to the demo application discussed in the use case, please visit www.mediasourcery.com