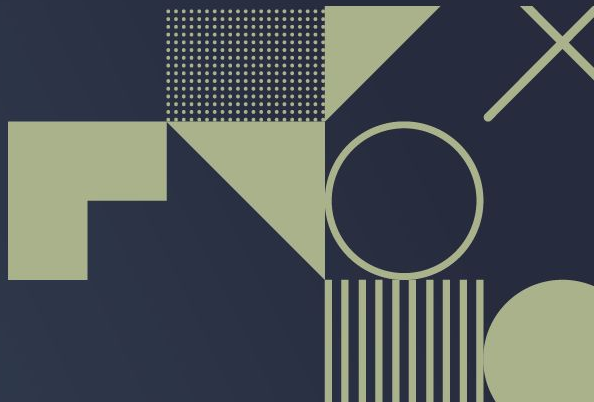
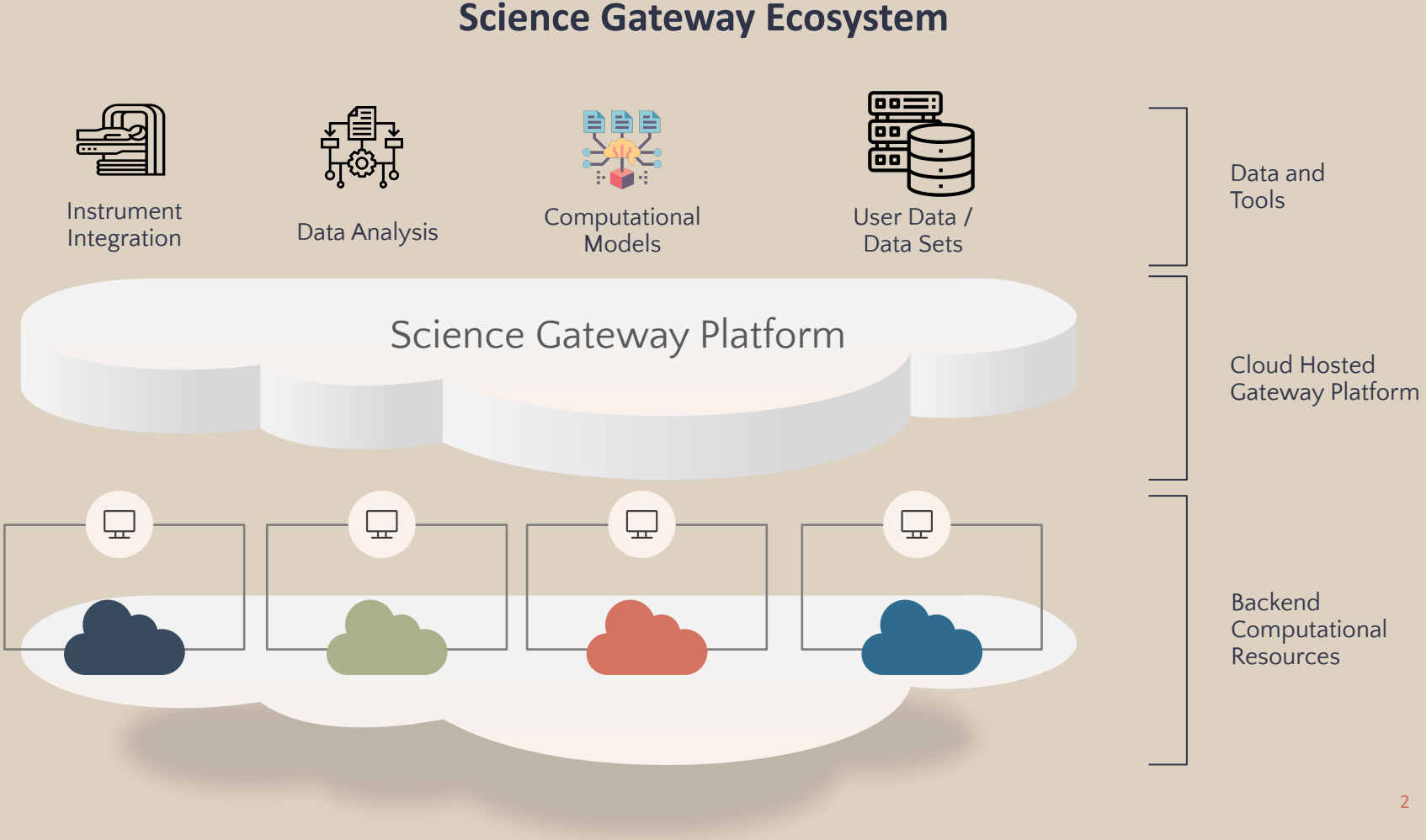
A Dynamic, Multi-Protocol Data Storage Integration Framework for Multi-Tenanted Science Gateway Middleware

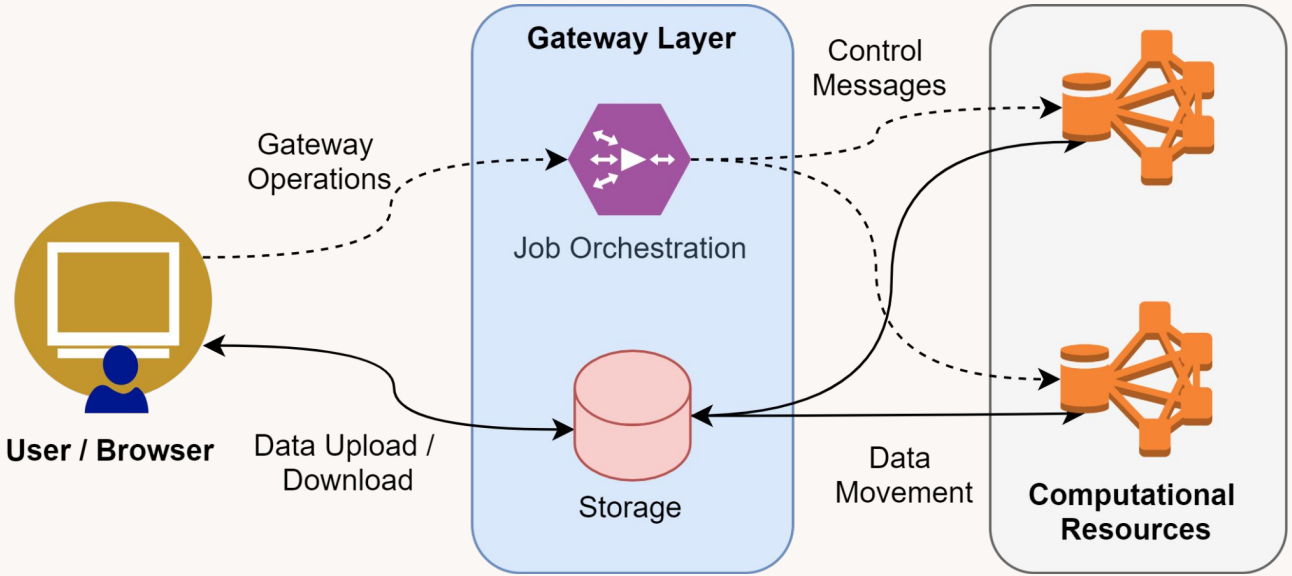
Dimuthu Wannipurage, Suresh Marru, Eroma Abeysinghe, Marcus Christie, Isuru Ranawaka, Marlon Pierce



Science Gateway Ecosystem



Conceptual Gateway Architecture



Mounting Data Storage inside the Gateway Layer

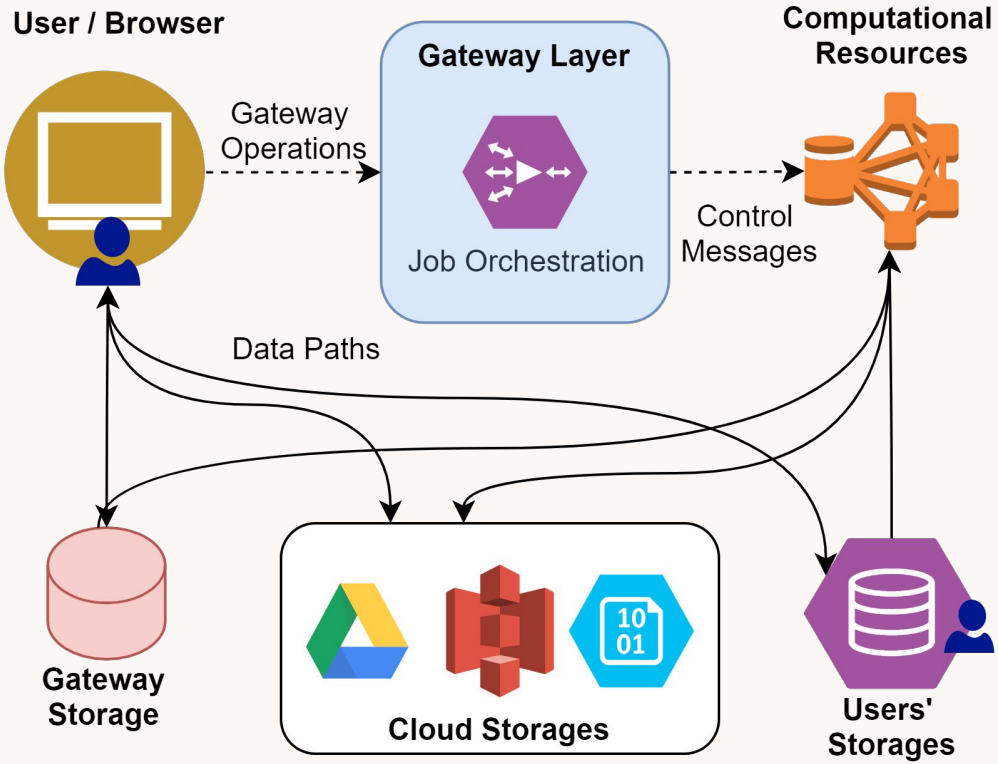
Pros

- Simple and easy to implement
- Fast access to data from the Gateway Layer

Cons

- Scalability issues - Bounded by the maximum disk space available in the host machine
- Data always pass through the gateway layer
- Data privacy – Users might not want to store sensitive data in the gateway storage
- Integration of cloud storages
- Users having their own storages and it is inefficient to duplicate data to the portal data storage

Decoupling the Gateway Storage



Challenges in Decoupling the Storage from Gateway Layer

- Gateway Layer can no longer access data through the local file system
- Working with heterogeneous storages types with different access protocols
- Securely accessing storage endpoints and access credential management

Importance of introducing an External Data Management Engine

- Separation of concerns – Let the gateway to do what it does best and delegate data management to a separate service
- Supporting a variety of data transfer protocols without affecting the gateway implementation

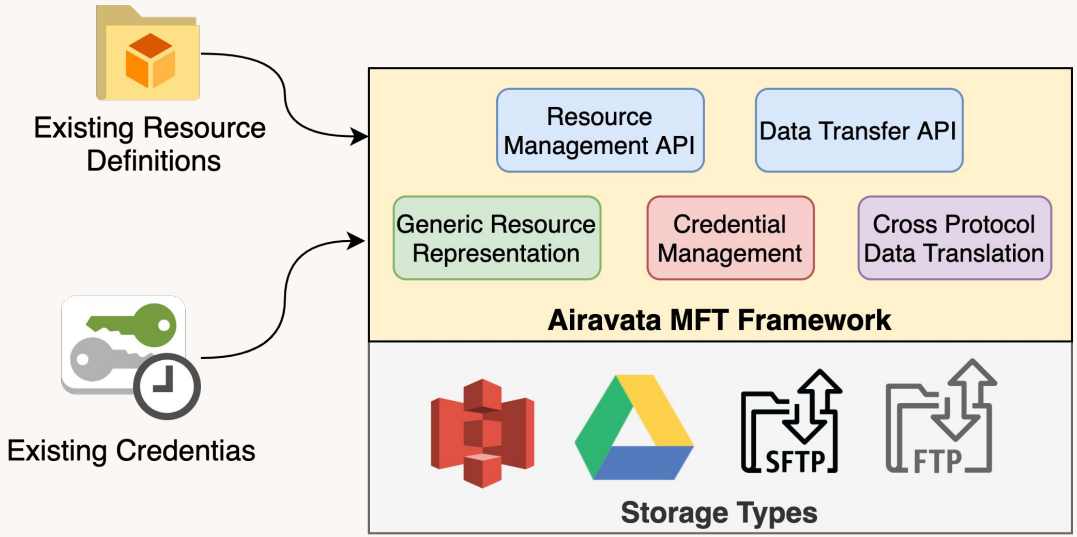
Managed File Transfer

- Software products specifically designed for data transfer orchestrations
- Ability to perform cross protocol translations
- In-built credential management to access data storage endpoints
- Added security in transfer paths
- Optimized Agent based solutions for high bandwidth data movement requirements

Existing Solutions

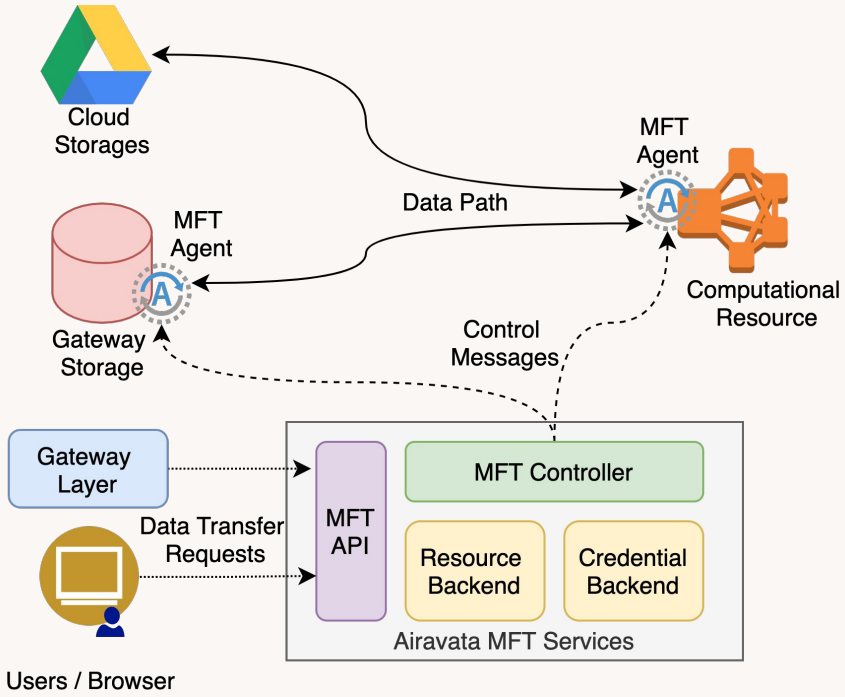
- Globus -
  - A high performance point to point data transfer platform which utilizes GridFTP.
  - Hosted, closed sourced and proprietary operations
- StorkCloud -
  - An opensource Managed File Transfer solution
  - Multi protocol data transfers, scheduling and directory listing
- GoAnywhere -
  - An commercial Managed File Transfer solution
  - Hosted as a service
- Rclone -
  - Open source tool that supports multi-protocol data transfers

Airavata Managed File Transfer Framework



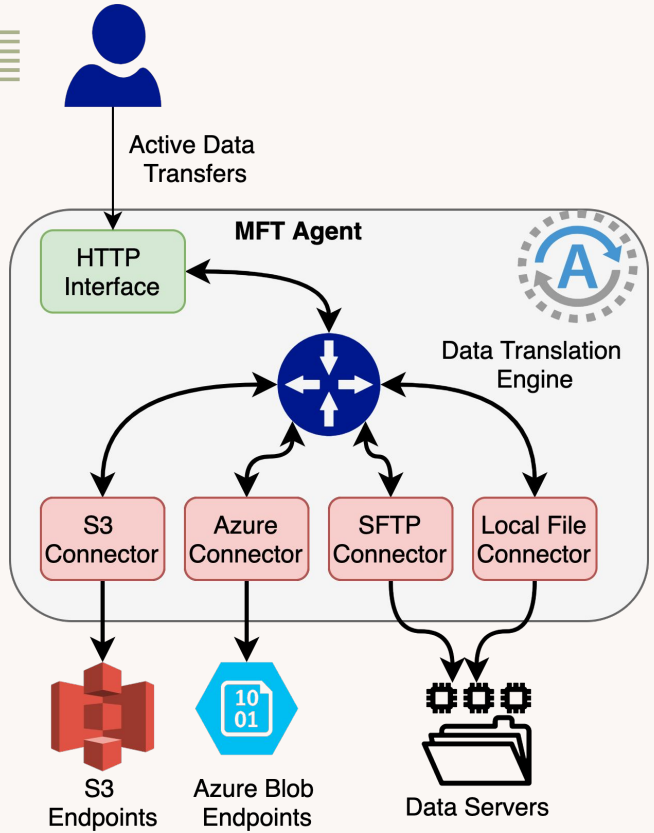
- Abstracts out data layer complexities (protocols, credentials and data representation)
- Pluggable Resource and Credential backend implementations to easily integrate with existing systems
- Support for most of the data transfer protocols and cloud data providers

Airavata MFT Internals



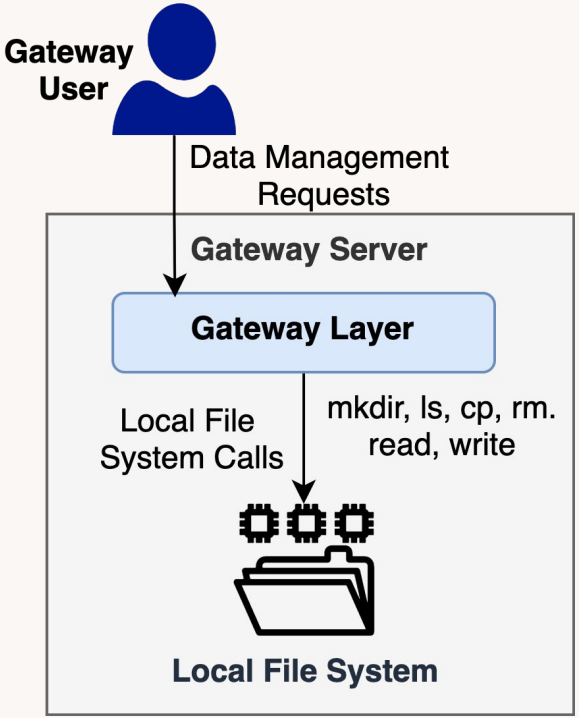
- Airavata MFT Contains four main internal services and external Agents
- Agents perform the data transfer between storage endpoints
- Agents can be deployed inside an endpoint or placed outside
- MFT Controller instructs Agents when a transfer request is received
- Credential and Resource backends provide an interface for external implementations to connect

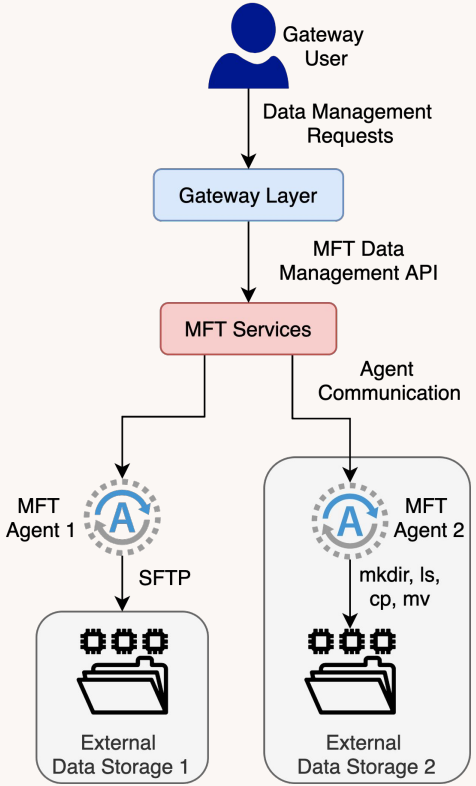
MFT Agent Internal



- An Agent is bundled with all the storage protocol client connectors
- Data Translation Engine converts any protocol specific data flow into the target format
- Each Agent is bundled with a HTTP Interface to communicate with external users

Integration of Airavata MFT with the Gateway Layer

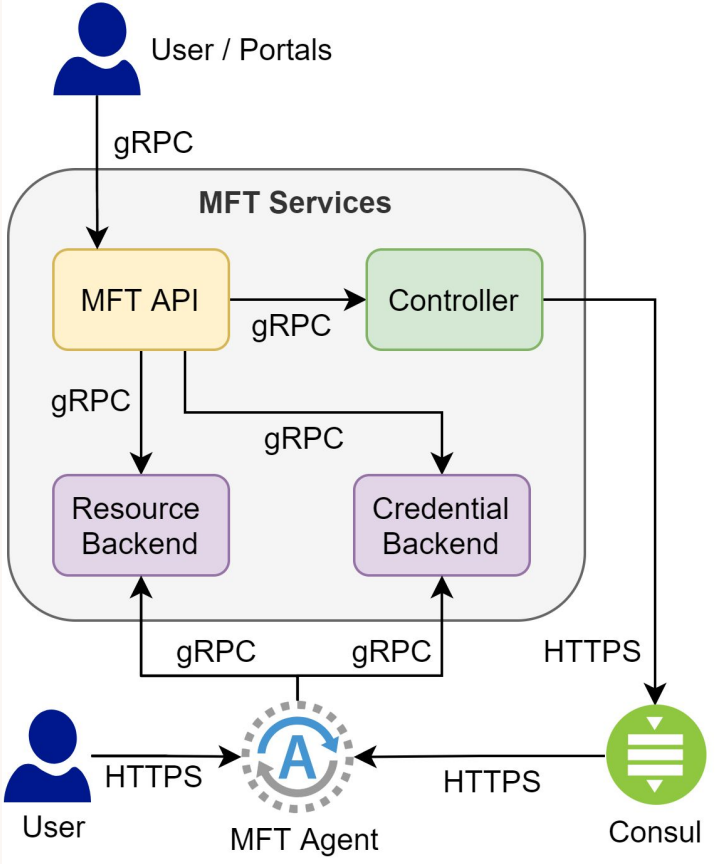




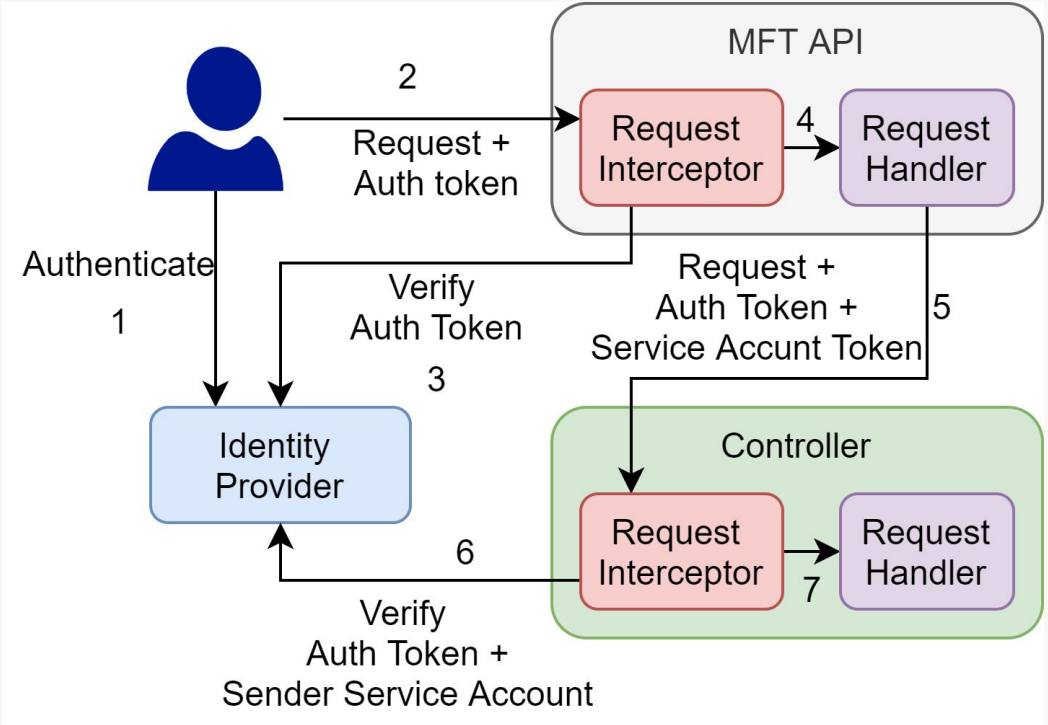
Security Integration for inter-service communication

- MFT Services are running as independent agents and communicate through gRPC protocols
- Each gRPC channels is encrypted over SSL / HTTP2
- Agents are receiving control messages through a Consul message bus with an encrypted HTTPS channel
- We follow a custom dual token authorization built on Oauth2 to authorize the user and internal service components

Communication Channels



Authorization and Inter-Service Authentication



Questions?

- Please reach us through circ@iu.edu
- Checkout the codebase https://github.com/apache/airavata-mft

This project is based upon work supported by the National Science Foundation under Award #1827641