|  |  |
| --- | --- |
|  | **WiserSKILLS** 8, rue de l'Hôtel de Ville 92200 Neuilly-sur-Seine FRANCE |

MICROSERVICE SPECIFICATIONS

|  |  |
| --- | --- |
| Service Name | UserQuery |
| Description | API service to manage authentication and security. The service integrates its own cache and is optimized for query. Data synchronization with other services is done via a shared NATS Streaming event bus. |
| Business Domain | user |
| Reference | S0401 |
| Version | 1.0.0 |
| Repository URL | https://gitlab.com/wiserskills/v3/services/userquery |
| Date | 11/01/2020 |
| Technology | Golang with WiserSKILLS Microservice framework |
| Storage | NoSQL (Ristretto Cache + ArangoDB) |
| K8s Namespace | Services |
| Specs Author(s) | Christophe PEILLET (CTO) |

# DEVELOPMENT ENVIRONMENT

The WiserSKILLS micro-services are developed in Golang language and require the following tools:

* Git (https://git-scm.com/downloads)
* Golang >= 1.12 (https://golang.org/doc/install)
* A code editor with Golang Support: Visual Studio Code with Golang plugin is the preferred IDE but other solutions could be used
* Docker Desktop for Mac or for Windows
* Make or an alternative on Windows like GNUMake (http://gnuwin32.sourceforge.net/packages/make.htm)

# FRAMEWORK AND PROJECT STRUCTURE

An important part of the service was pre-generated using the Goa v3 micro-service framework (https://goa.design) enriched with custom WiserSKILLS plugins.

The code structure is fully independent and can be cloned from the project repository on Gitlab (see URL on front page of this document).

It’s recommended to clone the project in the $GOPATH/src/gitlab.com/wiserskills/v3/services folder to avoid possible dependencies issues.

The definition of the service (endpoints, events, etc…) used for the generation can be found in the project **\_design** folder. Any modification in the design would require to regen the code.

The **gen** and **cmd** folders contain the generated service support files and should not be modified manually. Same for the **datalayer.go** and userquery.go files.

The project root contains a **Makefile** to ease the gen, build, run and test process.

When new dependencies are added to the project, they are normally automatically added to the go.mod file when build is started but some manual adjustments may be required.

The logic of the service should be handled in each individual endpoint and event file.

Unit and integration tests should be handled in the **userquery\_test.go** or in individual \_test files. Tests can be executed with the 'make test' command.

Data layer access was partially generated using the result types of the service. The related code can be found in the **datalayer/datalayer.go** file. Additional queries should be handled in the datalayer folder but in other files using the same package name.

The communication between services is mainly ensured via events and a **NATS Streaming** server. Most of the plumbing required to manage the events is already available in the service.

For additional information about NATS please refer to the official documentation at: https://nats.io/documentation/streaming/nats-streaming-intro/

A docker compose file is available to run an instance of NATS steaming server locally within Docker. The server is mapped on the host at port 4222.

This file also contains entries to run an instance of ArangoDB (https://www.arangodb.com) in docker on port 8529.

Two commands are available in the Makefile to start and stop the compose configuration: 'make compose-up' and 'make compose-down'.

To enable local testing of these communications, the service can be built locally by running 'make build' from the project root. This will build the service for the current OS.

Once built, the service can be run with 'make run'. This command launches the previously built service with the default environment variables values to reach the NATS & ArangoDB services listening on localhost.

# CODING GUIDELINES

The project should follow as much as possible the recommendations of the language as described at: https://golang.org/doc/effective\_go.html.

Main operations and errors should be logged to ease activity tracking and debugging. The service integrates the Zap logging package (https://github.com/uber-go/zap) that should be easy to use within endpoint methods.

Defensive programming is strongly encouraged to guarantee that passed values have expected format. A good article about this topic can be found at: https://itnext.io/clear-defensive-programming-with-go-using-verifier-library-6f648810b453.

At minimum each endpoint should be tested within the **userquery\_test.go** file. Additional unit and integration tests are of course welcome.

# WORKING WITH GIT ON GITLAB

The project repository on Gitlab.com is organized in 3 main branches: DEV, QA and MASTER. The QA and MASTER branches are protected and can only be modified via merge requests and approvals. If required by the service developer(s), the DEV branch can also be protected. In that case modifications will require that each feature be developed within a new branch and merged in the DEV branch via merge requests and approvals.

The service developer should have or create an account on Gitlab.com and provide his login for WiserSKILLS to give him the permissions on the project repository.

# METRICS

The service exposes a Prometheus metrics endpoint reachable at /metrics.

This endpoint mainly returns memory, and HTTP requests related counters.

Some custom basic metrics were generated for each event and endpoints however, additional metrics should be created to give the best possible insights regarding the service activity.

These custom metrics should be created within the **CreateCustomMetrics** function in the **metrics.go** file.

A wrapper is available to ease the creation of counters and gauges. However, the full collection of possible metrics is not supported yet (Summary & Histograms are missing for the moment).

# ENTITIES

The following tables describe the structure of the entities used and returned by the microservice.

Note: dates and time must be managed in UTC.

## System entities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ServiceStatusInfo | | | | |
| ID | E0001 | | | |
| Description | ServiceStatusInfo is the type used to hold the service status information. | | | |
| Field | Type | Size | Required | Description |
| service | string |  | Y | The name of the service |
| instanceId | string |  | Y | The id of the service instance |
| status | string |  | Y | The new status of the service. Possible values: Initializing, Connecting, Pending, Syncing, OK |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ErrorInfo | | | | |
| ID | E0002 | | | |
| Description | ErrorInfo is the type used to represent an error. | | | |
| Field | Type | Size | Required | Description |
| service | string |  | Y | The name of the service. |
| instanceId | string |  | Y | The id of the service instance. |
| message | string | 1024 | Y | The error message. |
| details | string | 8000 |  | The details about the error. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| LogLevelInfo | | | | |
| ID | E0005 | | | |
| Description | LogLevelInfo is the type used to specify a level of log. | | | |
| Field | Type | Size | Required | Description |
| service | string |  | Y | The name of the service |
| instanceId | string |  | Y | The id of the service instance |
| level | string |  | Y | The required log level. Possible values: DEBUG, INFO, WARN, ERROR |

## Business entities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User | | | | |
| ID | E0401 | | | |
| Description | User is the type used to represents a system user. | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the underlying or referenced entity. |
| uid | string | 40 |  | The UID of the underlying or referenced entity in DGraph. |
| key | string | 36 |  | The UUID of the underlying or referenced entity in the document database. |
| createdAt | string | 20 |  | The date/time the record was created. |
| updatedAt | string | 20 |  | The date/time the record was last updated. |
| active | boolean |  |  | Defines if the entity is active. |
| firstName | string | 255 | Y | The first name of the user. |
| lastName | string | 255 | Y | The last name of the user. |
| birthName | string | 255 | Y | The birth name of the user. |
| address | string | 1024 |  | The address of the user. |
| cityId | string | 255 |  | The id of the city. |
| countryId | string | 2 |  | The ISO 3166-1 code of the country. |
| latitude | float64 |  |  | The latitude of the user's address. |
| longitude | float64 |  |  | The longitude of the user's address. |
| birthDate | string | 10 |  | The birth date of the user. |
| gender | string |  |  | The gender of the user. |
| languageId | string | 2 |  | The ISO code of the user's prefered language. |
| email | string | 512 | Y | The email of the user. |
| login | string |  | Y | The login of the user. |
| encryptedPassword | string | 255 | Y | The encrypted password of the user. |
| passwordExpiresAt | string | 20 |  | The date/time the password will expire. |
| emailValidatedAt | string | 20 |  | The date/time the email was validated. |
| suspendedUpTo | string | 20 |  | The date/time the user is suspended up to. |
| failedAttempts | int |  |  | The number of failed authentication attempts. |
| mobile | string | 20 |  | The user mobile phone number. |
| B2C | boolean |  |  | Defines if this user is a B2C user. |
| organizationId | string |  | Y | The id of organization where the user was created. |
| roles | array |  |  | The list of the user roles. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SAMLConfig | | | | |
| ID | E0402 | | | |
| Description | SAMLConfig is the type used to represents a SAML configuration. | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the underlying or referenced entity. |
| key | string | 36 |  | The id of entity in the document database. |
| createdAt | string | 20 |  | The date/time the record was created. |
| updatedAt | string | 20 |  | The date/time the record was last updated. |
| active | boolean |  | Y | Defines if the entity is active. |
| organizationId | string |  | Y | The id of the associated organization. |
| areaId | string |  | Y | The id of the associated area. |
| host | string |  | Y | The associated host/domain. |
| idpMetadata | string |  | Y | The associated IDP metadata in XML. |
| idKey | string |  | Y | The key used to manage the user identification. |
| callbackURL | string |  | Y | The URL to be called back by the IDP. |
| redirectURL | string |  | Y | The URL the service redirects to once the user is authenticated. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Token | | | | |
| ID | E0404 | | | |
| Description | Token is the type used to represents a JWT token. | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the token. |
| key | string | 36 |  | The id of entity in the document database. |
| type | int |  | Y | The type of the token. |
| userId | string | 36 | Y | The UUID of the associated user. |
| organizationId | string |  | Y | The id of the associated organization. |
| areaId | string | 20 | Y | The id of the associated area. |
| token | string | 1024 | Y | The value of the token. |
| createdAt | string | 20 |  | The date/time the record was created. |
| expiresAt | string | 20 |  | The date/time the token will expire. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Session | | | | |
| ID | E0405 | | | |
| Description | Session is the type used to represents an application session. | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the session. |
| key | string | 36 |  | The id of entity in the document database. |
| organizationId | string |  | Y | The id of the associated organization. |
| areaId | string | 20 | Y | The id of the associated area. |
| userId | string | 36 | Y | The UUID of the user. |
| tokenId | string | 36 | Y | The UUID of the token. |
| createdAt | string | 20 |  | The date/time the record was created. |
| updatedAt | string | 20 |  | The date/time the record was updated. |
| expiresAt | string | 20 |  | The date/time the session will expire. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Password | | | | |
| ID | E0407 | | | |
| Description | Password is the type used to represent a password. | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the password. |
| userId | string | 36 | Y | The UUID of the associated user. |
| encryptedPassword | string | 255 | Y | The encrypted password value. |
| createdAt | string | 20 |  | The date/time the record was created. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AuthFailure | | | | |
| ID | E0408 | | | |
| Description | AuthFailure is the type used to describe a failed authentication. | | | |
| Field | Type | Size | Required | Description |
| login | string | 255 | Y | The login used for the authentication. |
| organizationId | string | 255 | Y | The id of the associated organization. |
| areaId | string | 20 | Y | The id of the associated area. |
| createdAt | string | 20 | Y | The date/time the record was created. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RegisteredUser | | | | |
| ID | E0409 | | | |
| Description | Represents a registered user. | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the underlying or referenced entity. |
| firstName | string | 255 | Y | The first name of the user. |
| lastName | string | 255 | Y | The last name of the user. |
| active | boolean |  |  | Defines if the entity is active. |
| birthName | string | 255 | Y | The birth name of the user. |
| address | string | 1024 |  | The address of the user. |
| cityId | string | 255 |  | The id of the city. |
| countryId | string | 2 |  | The ISO 3166-1 code of the country. |
| latitude | float64 |  |  | The latitude of the user's address. |
| longitude | float64 |  |  | The longitude of the user's address. |
| birthDate | string | 10 |  | The birth date of the user. |
| gender | string |  |  | The gender of the user. |
| languageId | string | 2 |  | The ISO code of the user's prefered language. |
| email | string | 512 | Y | The email of the user. |
| login | string |  | Y | The login of the user. |
| mobile | string | 20 |  | The user mobile phone number. |
| B2C | boolean |  |  | Defines if this user is a B2C user. |
| createdAt | string | 20 |  | The date/time the record was created. |
| updatedAt | string | 20 |  | The date/time the record was last updated. |
| organizationId | string |  | Y | The id of organization where the user was created. |
| roles | array |  |  | The list of the user roles. |

# ENVIRONMENT VARIABLES

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Required | Description | Default Value |
| CLUSTER | Y | The name of the cluster the service is running on. | fr0 |
| NAMESPACE | Y | The name of the namespace the service is running in. | services |
| BUSINESS\_DOMAIN | Y | The name of the business domain that the service is covering. | user |
| SERVICE\_TYPE | Y | Defines the type of the service. | query |
| SERVICE\_NAME | Y | The name of the service. | userquery |
| NODE |  | The name of the node the service is running on. |  |
| EVENTBUS\_CLUSTER |  | The name of the event bus cluster. | wiserskills |
| EVENTBUS\_URL |  | The URL of the event bus. | nats://localhost:4222 |
| EVENTBUS\_DEFAULT\_TOPIC |  | The event bus default topic. |  |
| EVENTBUS\_WRITE\_TIMEOUT |  | The timeout to write to the event bus in microseconds. | 5000000 |
| EVENTBUS\_RETRY\_COUNT |  | The number of retry to connect to the event bus in case of connection failure. | 5 |
| EVENTBUS\_RETRY\_DELAY |  | The delay in seconds before retrying to connect to the event bus in case of failure. | 5 |
| EVENTBUS\_CLIENT\_CERT\_PATH |  | The path to the client certificate file in PEM format used to connect to NATS. |  |
| EVENTBUS\_CLIENT\_KEY\_PATH |  | The path to the client private key file in PEM format used to connect to NATS. |  |
| INTERNAL\_ORG\_ID |  | The id of the organization that uses the internal scope. | wiserskills |
| API\_KEY |  | The api key. | wsk3z |
| TOKEN\_KEY |  | The key used to sign the token. | sdzT3wxcA |
| TOKEN\_ACTIVE |  | Defines if security by token is active. | false |
| TOKEN\_LIFE\_TIME |  | The life time of a token in minutes. | 1440 |
| SESSION\_LIFE\_TIME |  | The life time of a session in minutes. | 20 |
| LOGIN\_MAX\_ATTEMPT |  | The maximum number of failed login attempts. | 5 |
| LOGIN\_SUSPENSION\_TIME |  | The number of minutes a login can be suspended when the maximum failed number of attempts is reached. | 120 |
| DB\_CONNECTION |  | The URL of the database server. | http://localhost:8529 |
| DB\_LOGIN |  | The login used to connect to the database server. |  |
| DB\_PASSWORD |  | The password used to connect to the database server. |  |
| DB\_SEED\_FOLDER |  | The path of the folder that contain seed files. | seed |
| DB\_ENCRYPT |  | Defines if personal data fields must be encrypted. | false |

# ENDPOINTS

|  |  |
| --- | --- |
| SignIn | |
| Description | Signs a user in. |
| Filename | sign\_in.go |
| ID | EP04Q001 |
| Method | POST |
| Path | /signin |
| Body Type | SignInPayload |
| Result Type | JWTToken |

|  |  |
| --- | --- |
| SignOut | |
| Description | Signs a user out. |
| Filename | sign\_out.go |
| ID | EP04Q002 |
| Method | POST |
| Path | /signout |
| Body Type | TokenPayload |
| Result Type |  |

|  |  |
| --- | --- |
| GetAllSessions | |
| Description | Returns the currently active sessions. |
| Filename | get\_all\_sessions.go |
| ID | EP04Q003 |
| Method | GET |
| Path | /sessions |
| Body Type | AllSessionsPayload |
| Result Type | SessionCollection |

|  |  |
| --- | --- |
| GetIDPURL | |
| Description | Returns the URL of the IDP to redirect the user to. |
| Filename | get\_idpurl.go |
| ID | EP04Q004 |
| Method | GET |
| Path | /saml |
| Body Type | HostPayload |
| Result Type | RedirectResult |

|  |  |
| --- | --- |
| SamlSignIn | |
| Description | Call back endpoint called by the IDP once the user is authenticated. |
| Filename | saml\_sign\_in.go |
| ID | EP04Q005 |
| Method | POST |
| Path | /saml/signin |
| Body Type | string |
| Result Type | RedirectResult |

|  |  |
| --- | --- |
| CheckToken | |
| Description | Checks if the passed token is valid. |
| Filename | check\_token.go |
| ID | EP04Q006 |
| Method | POST |
| Path | /check |
| Body Type | TokenPayload |
| Result Type |  |

|  |  |
| --- | --- |
| GetUsersByID | |
| Description | Returns the users with the specified ids. |
| Filename | get\_users\_by\_id.go |
| ID | EP04Q007 |
| Method | GET |
| Path | /users/id |
| Body Type | ManyUserIDPayload |
| Result Type | RegisteredUserCollection |

|  |  |
| --- | --- |
| health | |
| Description | Health status endpoint. |
| Filename | health.go |
| ID | EP04Q801 |
| Method | GET |
| Path | /health |
| Body Type |  |
| Result Type | HealthResult |

|  |  |
| --- | --- |
| metrics | |
| Description | Prometheus metrics endpoint. |
| Filename | metrics.go |
| ID | EP04Q802 |
| Method | GET |
| Path | /metrics |
| Body Type |  |
| Result Type | string |

|  |  |
| --- | --- |
| swagger | |
| Description | Swagger service description endpoint. |
| Filename | swagger.go |
| ID | EP04Q901 |
| Method | GET |
| Path | /swagger |
| Body Type |  |
| Result Type | string |

|  |  |
| --- | --- |
| swagger-ui | |
| Description | Swagger UI service endpoint. |
| Filename | swagger\_ui.go |
| ID | EP04Q902 |
| Method | GET |
| Path | /swaggerui |
| Body Type |  |
| Result Type | string |

Apart from the variables marked with “{}” in the path, the endpoints also accept a view parameter (?view=) that enables to select the possible view of the returned entities. Two views are generally possible: default and tiny. The default view will return all the fields of the entity while the tiny one only some of them.

Some endpoints also contain an “active only” parameter to define if only active records must be returned. Default value is set to true.

The developer must ensure in the implementation that these values are taken into account when results are returned.

Passed parameters must be checked. In case of error, a Bad Argument should be raised.

Functions are available in the service package to create errors like **NotFound** or **BadArgument** errors for instance.

For protected endpoints, the **auth.go** file centralizes the token and api key validations and is called before the endpoint handler and must not be modified.

# EVENT SUBSCRIPTIONS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| LogLevelUpdated | | | | |
| ID | EV0002 | | | |
| File | on\_log\_level\_updated.go | | | |
| Description | LogLevelUpdated is received when the level of log must be changed. | | | |
| Topic | [CLUSTER].[INTERNAL\_ORG\_ID].services.userquery | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: LogLevelInfo |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserCreated | | | | |
| ID | EV0401 | | | |
| File | on\_user\_created.go | | | |
| Description | UserCreated is received when a new user has been created. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: User |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserUpdated | | | | |
| ID | EV0402 | | | |
| File | on\_user\_updated.go | | | |
| Description | UserUpdated is received when an existing user has been updated. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: User |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserDeleted | | | | |
| ID | EV0403 | | | |
| File | on\_user\_deleted.go | | | |
| Description | UserDeleted is received when an existing user has been deleted. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: User |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserEnabled | | | | |
| ID | EV0404 | | | |
| File | on\_user\_enabled.go | | | |
| Description | UserEnabled is received when an existing user has been enabled. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: User |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserDisabled | | | | |
| ID | EV0405 | | | |
| File | on\_user\_disabled.go | | | |
| Description | UserDisabled is received when an existing user has been disabled. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: User |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TokenCreated | | | | |
| ID | EV0406 | | | |
| File | on\_token\_created.go | | | |
| Description | TokenCreated is received when a new token has been created. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Token |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TokenExpired | | | | |
| ID | EV0407 | | | |
| File | on\_token\_expired.go | | | |
| Description | TokenExpired is received when a token has expired. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Token |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TokenRevoked | | | | |
| ID | EV0408 | | | |
| File | on\_token\_revoked.go | | | |
| Description | TokenRevoked is received when a token has been revoked. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Token |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SessionCreated | | | | |
| ID | EV0409 | | | |
| File | on\_session\_created.go | | | |
| Description | SessionCreated is received when a new session has been created. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Session |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SessionUpdated | | | | |
| ID | EV0410 | | | |
| File | on\_session\_updated.go | | | |
| Description | SessionUpdated is received when a session has been updated. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Session |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SessionExpired | | | | |
| ID | EV0411 | | | |
| File | on\_session\_expired.go | | | |
| Description | SessionExpired is received when a session has expired. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Session |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AuthFailureCreated | | | | |
| ID | EV0417 | | | |
| File | on\_auth\_failure\_created.go | | | |
| Description | AuthFailureCreated is received when an authentication failure was created. | | | |
| Topic | [CLUSTER].\*.user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: AuthFailure |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PasswordExpired | | | | |
| ID | EV0418 | | | |
| File | on\_password\_expired.go | | | |
| Description | PasswordExpired is received when a password has expired. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Password |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PasswordChanged | | | | |
| ID | EV0419 | | | |
| File | on\_password\_changed.go | | | |
| Description | PasswordChanged is received when a password has been changed. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Password |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SAMLConfigCreated | | | | |
| ID | EV0422 | | | |
| File | on\_saml\_config\_created.go | | | |
| Description | SAMLConfigCreated is received when a new SAML config has been created. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: SAMLConfig |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SAMLConfigUpdated | | | | |
| ID | EV0423 | | | |
| File | on\_saml\_config\_updated.go | | | |
| Description | SAMLConfigUpdated is received when a SAML config has been updated. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: SAMLConfig |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SAMLConfigDeleted | | | | |
| ID | EV0424 | | | |
| File | on\_saml\_config\_deleted.go | | | |
| Description | SAMLConfigDeleted is received when a SAML config has been deleted. | | | |
| Topic | [CLUSTER].\*.user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: SAMLConfig |
| payload | string |  | Y | The representation of the payload object in JSON. |

# EVENT PUBLICATIONS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UserUpdated | | | | |
| ID | EV0402 | | | |
| Description | UserUpdated is raised when a user has been updated. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: User () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TokenCreated | | | | |
| ID | EV0406 | | | |
| Description | TokenCreated is raised when a new token is created. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Token () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TokenExpired | | | | |
| ID | EV0407 | | | |
| Description | TokenExpired is raised when a token is expired. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Token () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TokenRevoked | | | | |
| ID | EV0408 | | | |
| Description | TokenRevoked is raised when a token is revoked. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Token () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SessionCreated | | | | |
| ID | EV0409 | | | |
| Description | SessionCreated is raised when a new session is created. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Session () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SessionUpdated | | | | |
| ID | EV0410 | | | |
| Description | SessionUpdated is raised when a session is updated. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Session () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SessionExpired | | | | |
| ID | EV0411 | | | |
| Description | SessionExpired is raised when a session is expired. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Session () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AuthFailureCreated | | | | |
| ID | EV0417 | | | |
| Description | AuthFailureCreated is raised when a authentication attempt has failed. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.query | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: AuthFailure () |
| payload | string |  | Y | The representation of the payload object in JSON. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PasswordExpired | | | | |
| ID | EV0418 | | | |
| Description | PasswordExpired is raised when a user's password has expired. | | | |
| Topic | [CLUSTER].[$Event.OrgID].user.admin | | | |
| Field | Type | Size | Required | Description |
| id | string | 36 | Y | The UUID of the event. |
| userId | string | 36 | Y | The UUID of the user who raised the event. |
| sessionId | string | 36 | Y | The UUID of the session the event was raised in. |
| timestamp | string | 32 | Y | The date/time the event was generated. |
| requestId | string | 36 | Y | The UUID of the request that generated the event. |
| action | string |  | Y | The action represented by the event. Possible values: CREATE, UPDATE, DISABLE, ENABLE, DELETE, STATUS, SYNC |
| typeName | string |  | Y | The type name of the payload: Password () |
| payload | string |  | Y | The representation of the payload object in JSON. |