ATLAS Microsoft Entra ID permissions

In order to connect ATLAS to Microsoft EntraID, permission grants are required so the needed data can be read from EntraID and used for ATLAS.

API name	Claim value	Permission	type	Granted through	Granted by
Microsoft Graph	RoleManagement.Read.Directory	read all directory RBAC settings	Application	Admin consent	An Administrator

Explanation: ATLAS supports role based access control for the ATLAS web application. This permission allows the app to read the role-based access control (RBAC) settings for your company's directory, on behalf of the signed-in user. This includes reading directory role templates, directory roles and memberships.

Example Graph API response from:

```
https://graph.microsoft.com/v1.0/directoryRoles?$filter=displayName%20eq%20%27Global%20Administrator%27&$select=id
```

```
[
  {
    "status": "fulfilled",
    "value": {
      "@odata.context":
"https://graph.microsoft.com/v1.0/$metadata#directoryRoles",
      "value": [
        {
          "id": "6c4b1bf3-66aa-4230-8af4-eb6f9f1b1aa8"
        }
      ]
    }
  },
  {
    "status": "fulfilled",
    "value": {
      "@odata.context":
"https://graph.microsoft.com/v1.0/$metadata#directoryRoles",
      "value": [
        {
          "id": "ebb6d345-9e75-4ad4-b8c9-8dc32732db03"
        }
      ]
    }
  }
]
```

Properties we store	**Reason**
"id"	Determine which users are admins in ATLAS because ATLAS works with role based access. Admins can configure the system.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/directoryrole-get?view=graph-rest-1.0&tabs=http

Other Properties	**Reason**
All properties not displayed in the above “properties we store”.	All properties returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.

API name	**Claim value**	**Permission**	**type**	**Granted through**	**Granted by**
Microsoft Graph	Group.Read.All	read all groups	Application	Admin consent	An Administrator

Explanation: ATLAS provided access control based on User Groups. This permission allows ATLAS to list groups, to read the properties and all group memberships on behalf of the signed-in user. ATLAS needs this permission to handle group-based permissions in the app.

Example Graph API response from:
https://graph.microsoft.com/v1.0/groups?$filter=startsWith(displayName,'atlas') or startsWith(displayName,'atlas')

```
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#groups",
  value:
    [
        {
            "id": "2bb02cde-f1a7-4c9d-9e95-855b285a202d",
            "displayName": "atlas-group1"
        },
        {
            "id": "c7117866-52f3-4af5-96ca-03bda5dcb70b",
            "displayName": "atlas-group2"
        }
    ]
}
```

Properties we store	**reason**
"id"	Stored for reference

`"displayName"`	For GUI show the user groups so Admin can grant access for users based on their memberships in user groups.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/group-list?view=graph-rest-1.0&tabs=http

Other Properties	reason
All properties not displayed in the above “propierties we store”.	All objects returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.

API name	Claim value	Permission	type	Granted through	Granted by
Microsoft Graph	GroupMember.Read.All	Read all group memberships	Application	Admin consent	An Administrator

Explanation: ATLAS provided access control based on User Groups. This permission allows the app to read memberships related to a group and basic group properties for all groups without a signed-in user.

Example Graph API response from:

```
https://graph.microsoft.com/v1.0/groups/${groupId}/members?$select=id,userPrincipalName,userType,accountEnabled
```

```
{
  '@odata.context': 'https://graph.microsoft.com/v1.0/$metadata#directoryObjects(id,userPrincipalName,userType,accountEnabled)',
  value: [
    {
      '@odata.type': '#microsoft.graph.user',
      id: '4bc08c5d-385c-4f2c-9ffc-37ec0aab80d2',
      userPrincipalName: 'person2@tkhsecurity.onmicrosoft.com',
      userType: 'Member',
      accountEnabled: true
    },
    {
      '@odata.type': '#microsoft.graph.user',
      id: '08602507-8ead-4500-a93d-548e6da25b64',
      userPrincipalName: 'bob@tkhsecurity.onmicrosoft.com',
      userType: 'Member',
      accountEnabled: true
    }
  ]
}
```

Properties we store	reason
"id"	Stored for reference .
"userPrincipalName"	Identification of user in Atlas system, granting access to doors.
"accountEnabled"	Determine if users still should has access. if status changes access will change accordingly.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/group-list-members?view=graph-rest-1.0&tabs=http

Other Properties	reason
All objects not displayed in the above “objects we store”.	All properties returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.

API name	Claim value	Permission	type	Granted through	Granted by
Microsoft Graph	User.Reader	Sign in and reader user profile	Delegated	Admin consent	An Administrator

Explanation: ATLAS needs this permission to read data about signed user. This permission allows ATLAS to read the profile of signed-in users and read basic company information of signed-in users.

Remark: It might be drop soon because we have User.Read.All permission below. it contains the same data.

API name	Claim value	Permission	type	Granted through	Granted by
Microsoft Graph	User.Read.All	Read all users full profile	Application	Admin consent	An Administrator

Explanation: ATLAS need this permission to synchronize users from EntraID to it’s local database on behave of providing access control. Allows the app to read the full set of profile properties, group membership, reports, and managers of other users in your organization, without a signed-in user.

Example Graph API response from:
`https://graph.microsoft.com/v1.0/users?$filter=accountEnabled+eq+true+and+userType+eq+%27Member%27&$select=id,userPrincipalName,displayName`

```
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users",
  "value":
    [
        {
            "id": "4bc08c5d-385c-4f2c-9ffc-37ec0aab80d2",
            "userPrincipalName": "person2@tkhsecurity.onmicrosoft.com",
            "displayName": "Person2"
        },
        {
            "id": "dbcf87b3-d09a-4042-8143-e96e2a0fde83",
            "userPrincipalName": "person3@tkhsecurity.onmicrosoft.com",
            "displayName": "Person3"
        }
    ]
}
```

Properties we store	**reason**
`"id"`	Stored for reference.
`"displayName"`	To personalize GUI.
`"userPrincipalName"`	Identification of user in Atlas system, granting access to doors.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/user-get?view=graph-rest-1.0&tabs=http

Other Properties	**reason**
All objects not displayed in the above “objects we store”.	All properties returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.

API name	**Claim value**	**Permission**	**type**	**Granted through**	**Granted by**
Microsoft Graph	email	view users email address	Delegated	Admin consent	An Administrator

Explanation: ATLAS needs this permission to sign-in users to ATLAS Access through Single Sign On (SSO).

Note: This permission is not visible when customer is granting access to the app because it is include in separate app which is responsible only for SSO.

Properties we store	reason
`none`	We do not store any data from this permission. It is only needed to support SSO.

Other Properties	reason
All objects not displayed in the above “objects we store”.	All properties returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.

API name	Claim value	Permission	type	Granted through	Granted by
Microsoft Graph	MailboxSettings.Read	Read all user mailbox settings	Application	Admin consent	An Administrator

Explanation: ATLAS need this permission to determine if an Entra ID user is a User or Room. None of the data is stored or used for other purpose.

Example Graph API response from:
`https://graph.microsoft.com/v1.0/Users/${userId}?$select=id,mailboxSettings`

```
{
  "@odata.context":
"https://graph.microsoft.com/v1.0/$metadata#users(id,mailboxSettings)/$entity
",
  "id": "08602507-8ead-4500-a93d-548e6da25b64@8e3a4e10-fdbe-49d9-947a-
f5d3eba4ee64",
  "mailboxSettings": {
    "archiveFolder":
"AAMkAGNjY2E2YTJkLWMxYjctNDA0My1iNjMzLWYzMDNiMjM2YTY1YwAuAAAAAABfaQ1VLEsYS4xE
ZlJ_QOyEAQDQw7Y3sfMXRK78s0aIsWyoAAACj_BUAAA=",
    "timeZone": "Central European Standard Time",
    "delegateMeetingMessageDeliveryOptions": "sendToDelegateOnly",
    "dateFormat": "yyyy-MM-dd",
    "timeFormat": "HH:mm",
    "userPurpose": "user",
    "automaticRepliesSetting": {
      "status": "disabled",
      "externalAudience": "all",
      "internalReplyMessage": "",
      "externalReplyMessage": "",
      "scheduledStartDateTime": {
        "dateTime": "2023-07-17T10:00:00.0000000",
        "timeZone": "UTC"
      },
      "scheduledEndDateTime": {
        "dateTime": "2023-07-18T10:00:00.0000000",
        "timeZone": "UTC"
      }
    },
    "language": {
      "locale": "en-GB",
      "displayName": "English (United Kingdom)"
    },
    "workingHours": {
      "daysOfWeek": [
        "monday",
        "tuesday",
        "wednesday",
        "thursday",
        "friday"
      ],
      "startTime": "08:00:00.0000000",
      "endTime": "17:00:00.0000000",
      "timeZone": {
        "name": "Central European Standard Time"
      }
    }
  }
}
```

Properties we store	reason
none	We don’t store any data from mailbox setting. We only need to check if the mailbox exists.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/user-get-mailboxsettings?view=graph-rest-1.0&tabs=http

API name	Claim value	Permission	type	Granted through	Granted by
Microsoft Graph	Calendars.Read	read all calendars in mailboxes	Application	Admin consent	An Administrator

Explanation: ATLAS needs this permission to tailor the application with additional information about the company, provided by Outlook, to handle visitor management. It allows ATLAS to read the full set of profile properties, reports, and managers of other users in your organization, on behalf of the signed-in user. With this data access can be granted according to the planned visit.

Example Graph API response from:

```
https://graph.microsoft.com/v1.0/Users/6a10391a-66de-42be-935e-c1fed3e5979f/Events/${eventId}?$select=id,location,start,end,organizer,attendees,isOrganizer
```

Note: Nested $select is not supported in GraphAPI.

```
{
    "@odata.context":
"https://graph.microsoft.com/v1.0/$metadata#users('08602507-8ead-4500-a93d-
548e6da25b64')/events(id,location,start,end,organizer,attendees,isOrganizer)/
$entity",
    "@odata.etag": "W/\"0MO2N7HzF0Su/LNGiLFsqAABr9CQbg==\"",
    "id":
"AAMkAGNjY2E2YTJkLWMxYjctNDA0My1iNjMzLWYzMDNiMjM2YTY1YwBGAAAAAABfaQ1VLEsYS4xE
ZlJ_QOyEBwDQw7Y3sfMXRK78s0aIsWyoAAAAAAENAADQw7Y3sfMXRK78s0aIsWyoAAGwI_kTAAA="
,
    "isOrganizer": true,
    "start": {
        "dateTime": "2023-08-22T13:00:00.0000000",
        "timeZone": "UTC"
    },
    "end": {
        "dateTime": "2023-08-22T13:30:00.0000000",
        "timeZone": "UTC"
    },
    "location": {
        "displayName": "Meeting room 1",
        "locationUri": "meetingroom_1@tkhsecurity.onmicrosoft.com",
        "locationType": "conferenceRoom",
        "uniqueId": "meetingroom_1@tkhsecurity.onmicrosoft.com",
        "uniqueIdType": "directory",
        "address": {
            "street": "",
            "city": "",
            "state": "",
            "countryOrRegion": "",
            "postalCode": ""
        },
        "coordinates": {}
    },
    "attendees": [
        {
            "type": "required",
            "status": {
                "response": "none",
                "time": "0001-01-01T00:00:00Z"
            },
            "emailAddress": {
                "name": "Atendee",
                "address": "atendee@tkhsecurity.onmicrosoft.com"
            }
        },
        {
            "type": "resource",
            "status": {
                "response": "accepted",
                "time": "2023-08-22T12:51:24.4049952Z"
            },
            "emailAddress": {
                "name": "Meeting room 1",
                "address": "meetingroom_1@tkhsecurity.onmicrosoft.com"
            }
        }
    ],
    "organizer": {
        "emailAddress": {
            "name": "Bob",
            "address": "bob@tkhsecurity.onmicrosoft.com"
        }
    },
    "calendar@odata.associationLink":
"https://graph.microsoft.com/v1.0/Users('08602507-8ead-4500-a93d-
548e6da25b64')/calendar/$ref",
    "calendar@odata.navigationLink":
"https://graph.microsoft.com/v1.0/Users('08602507-8ead-4500-a93d-
548e6da25b64')/calendar"
}
```

Objects we store	**reason**
"id"	Stored for reference.
"start.dateTime"	For GUI, granting access to locks which are booked via outlook.
"end.dateTime"	For GUI, granting access to locks which are booked via outlook.
"organizer.emailAddress.address"	Identification of user in Atlas system, granting access to locks and know who's the host of the booking.
"attendees.emailAddress"	For GUI, to know who to grant access to locks.
"location.displayName"	For GUI, to know who to grant access to locks.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/calendar-list-events?view=graph-rest-1.0&tabs=http

Other Properties	**reason**
all objects not displayed in the above "objects we store"	All properties returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.

API name	**Claim value**	**Permission**	**type**	**Granted through**	**Granted by**
Microsoft Graph	Place.Read.All	Read all company places	Application	Admin consent	An Administrator

Explanation: ATLAS needs this permission to make use of Outlook location. This functionality is used for our visitor management module where visitors can be invited through Outlook. This permission allows ATLAS to read company places/location (conference rooms and room lists) for calendar events (meetings). ATLAS uses this permission to synchronise locations from AD to ATLAS to list possible places where meetings can be arranged and places where locks can be mounted so it can be configured in the ATLAS GUI.

Example Graph API response from:

```
https://graph.microsoft.com/v1.0/places/microsoft.graph.room?$select=id,displayName,emailAddress
```

```
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#places/microsoft.graph.room(id,displayName,emailAddress)",
  "value": [
    {
      "id": "d50fc55d-1df0-4868-971c-4038f0fef991",
      "displayName": "kitchen",
      "emailAddress": "kitchen@tkhsecurity.onmicrosoft.com"
    },
    {
      "id": "0a4e1295-7a25-4a59-86b1-4c07427f23c0",
      "displayName": "Meeting room 1",
      "emailAddress": "meetingroom_1@tkhsecurity.onmicrosoft.com"
    },
    {
      "id": "47d62cad-5bd5-4600-87f3-d73712448681",
      "displayName": "Meeting room 2",
      "emailAddress": "meetingroom_2@tkhsecurity.onmicrosoft.com"
    }
  ]
}
```

Properties we store	**reason**
`"id"`	Stored for reference.
`"displayName"`	For GUI to display name of location.
`"emailAddress"`	Identification in Atlas system.

GraphAPI Documentation: https://learn.microsoft.com/en-us/graph/api/place-list?view=graph-rest-1.0&tabs=http

Other Properties	**reason**
all objects not displayed in the above “objects we store”	All properties returned which are not used, will be ignored. The moment the function is executed the non-used data will be removed.