Microsoft Identity platform
Developer training

Ronny Hansen
Ronny.Hansen@microsoft.com
Cloud Solution Architect

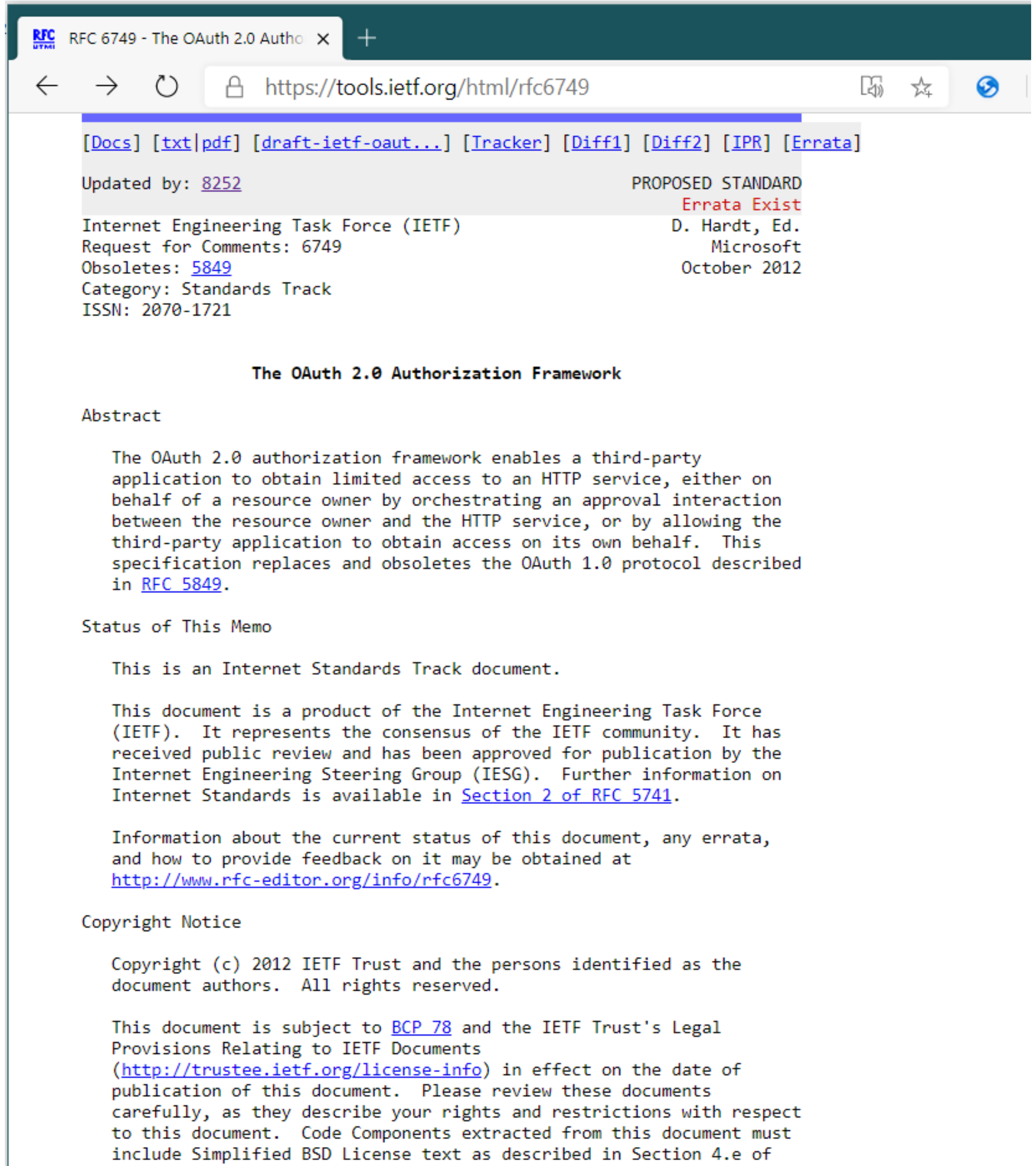
Irina Kostina
Irina.Kostina@Microsoft.com
Cloud Solution Architect

~~Protecting an API~~

Microsoft

Let's talk about Authorization

RFC 6749 - The OAuth 2.0 Authorization Framework


RFC 6749 - The OAuth 2.0 Autho
https://tools.ietf.org/html/rfc6749

[Docs] [txt|pdf] [draft-ietf-oaut...] [Tracker] [Diff1] [Diff2] [IPR] [Errata]

```
Updated by: 8252                                        PROPOSED STANDARD
                                                             Errata Exist
Internet Engineering Task Force (IETF)                    D. Hardt, Ed.
Request for Comments: 6749                                    Microsoft
Obsoletes: 5849                                            October 2012
Category: Standards Track
ISSN: 2070-1721
```

The OAuth 2.0 Authorization Framework

Abstract

The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application to obtain access on its own behalf. This specification replaces and obsoletes the OAuth 1.0 protocol described in RFC 5849.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

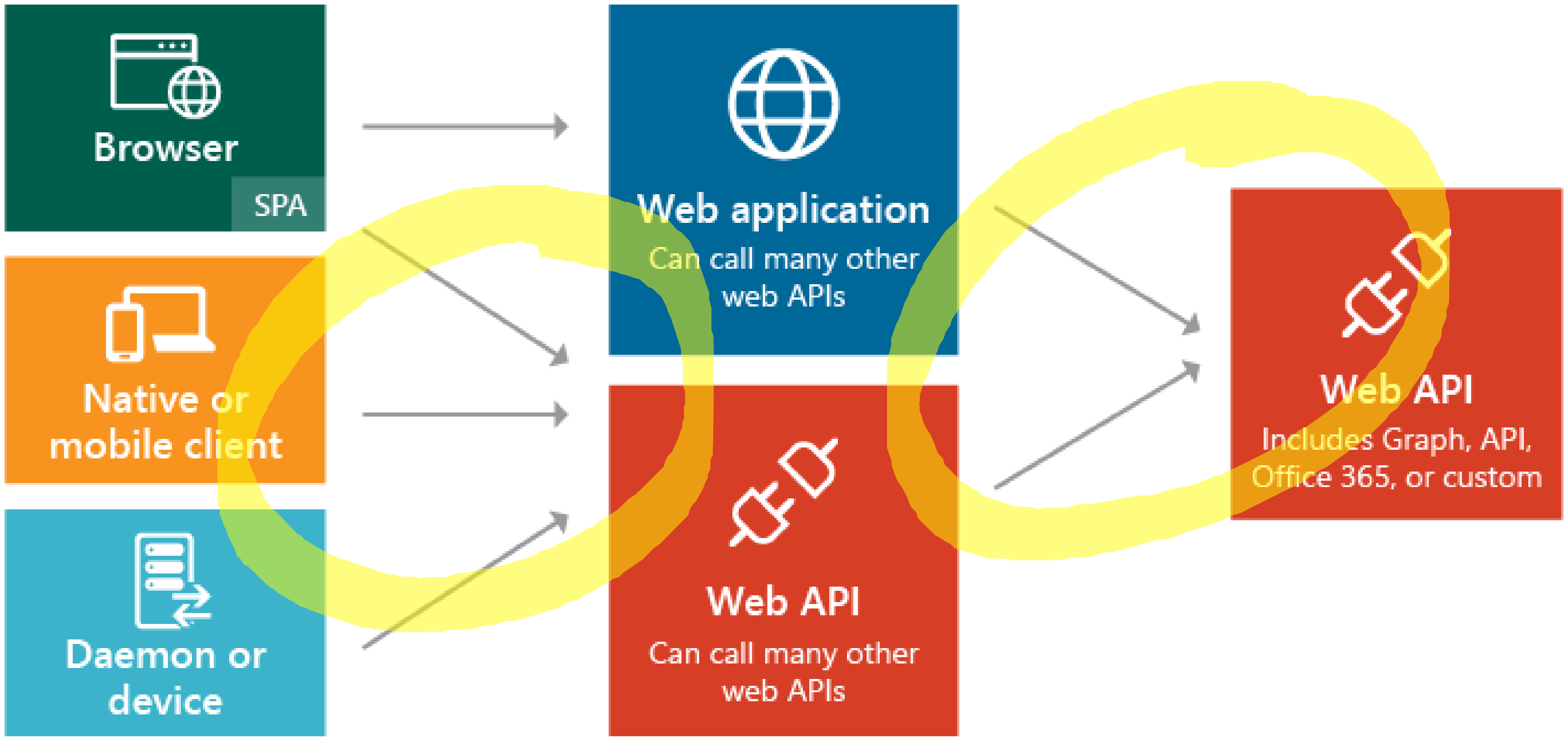
Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6749.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

Protecting an API


Browser
SPA
Native or mobile client
Daemon or device
Web application
Can call many other web APIs
Web API
Can call many other web APIs
Web API
Includes Graph, API, Office 365, or custom

Protecting an API with Azure AD

- Register the API as an app in Azure AD
  - Define the **delegated** permissions your API exposes in the App Reg portal
    - Help developers using your API to keep to least-privilege
    - Avoid “do everything” permissions where possible
    - Be conservative with permissions users can consent to

- Validate received access tokens in your API
  - Use existing libraries and middleware. They exist for most platforms.

- Apply and enforce permissions!
  - Delegated permissions must not exceed what the signed-in user is allowed to do.

Access tokens

- An API secured with Azure AD will require a valid access token.
- Microsoft issues access tokens in the JSON Web Token (JWT) format.
  - RFC 7519 JSON Web Token (JWT)
  - Structure of a JWT:
    - Header: token metadata
      - Provides information needed for token verification
    - Payload: claims asserted by token issuer
      - Identifies the authenticated user (if there is one)
      - Includes roles and scopes to be used in authorization
    - Signature: created with issuer's private key
      - MUST be verified!
  - Decode and view JWT tokens: https://jwt.ms

Validating access tokens

Use a library for signature verification and JWT decoding.

- Attempting to implement this yourself is a security risk.

General guidelines

- The audience (aud) must identify the service reading the token
- The issuer (iss) must be the trusted issuer (e.g. an Azure AD tenant).
- The token must be valid (nbf < now) and not expired (exp > now).
- Use *immutable* unique identifiers to identify subjects.
  - Object ID (oid) or the subject (sub) claim are good options. Email and UPN can change.
- Do *not* accept the “none” signature algorithm.

More details in the documentation

- https://docs.microsoft.com/azure/active-directory/develop/v2-id-and-access-tokens

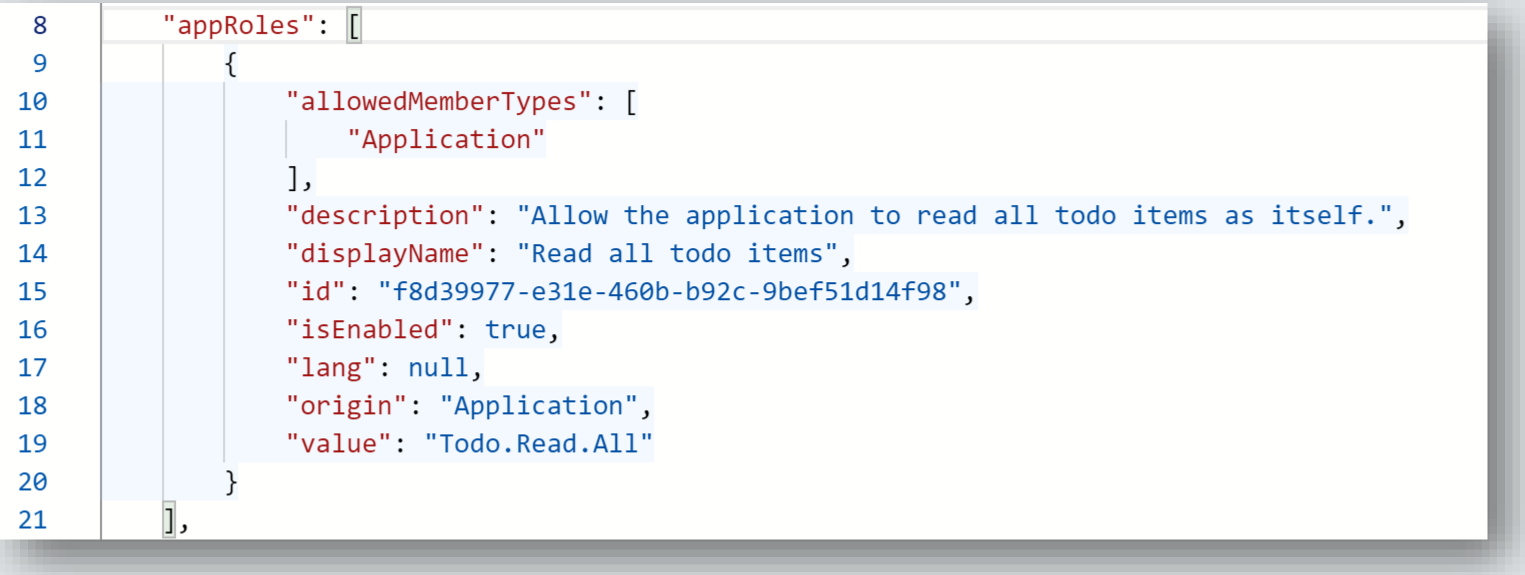
Demo

Application Permissions (for daemons)

- Apps do not authenticate as a user
- Apps authorize using “Client Credentials”
- Returned in the “role” claim
- App Roles with “allowedMemberTypes” of Application

App Permissions Demo

https://github.com/azure-samples/active-directory-dotnetcore-daemon-v2



```
    "appRoles": [
        {
            "allowedMemberTypes": [
                "Application"
            ],
            "description": "Allow the application to read all todo items as itself.",
            "displayName": "Read all todo items",
            "id": "f8d39977-e31e-460b-b92c-9bef51d14f98",
            "isEnabled": true,
            "lang": null,
            "origin": "Application",
            "value": "Todo.Read.All"
        }
    ],
```

Cactus API example

What does this Access token allow?

eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6InU0T2ZORlBId0VCb3NIanRyYXVPYlY4NExuWSJ9.eyJhdWQiOiI0
MTY2ODRhNy0wYjUyLTRmYTMtOTkxOC1lNzZkMTY1NDJiZTIiLCJpc3MiOiJodHRwczovL2xvZ2luLm1pY3Jvc29mdG
9ubGluZS5jb20vYzcyYTI5NWQtZDdhNS00MWVhLWEzNTEtYjE1ZGQ5ZjY3MjE1L3YyLjAiLCJpYXQiOjE1NjM4ODMz
MzYsIm5iZiI6MTU2Mzg4MzMzNiwiZXhwIjoxNTYzODg3MjM2LCJhaW8iOiJBVFFBeS84TUFBQUFhTHRBLzNNb2R
4K09OK1FNaTUwMnZycFdib0h6c253akNxYmladU9qcTE2bFkxelU3YkVNdDlIUW90NXJGcDVqIiwiYXpwIjoiYmI3Nj
RjMjEtNDliOC00OWRlLWFhMjQtNmM3NmQ3ZGM4MDBmIiwiYXpwYWNyIjoiMCIsIm5hbWUiOiJNZWdhbiBCb3d
lbiIsIm9pZCI6IjBlNzQ4Y2QwLTVkMmEtNDkxOC1hMzUxLTk1NDllNzVmZDFkZCIsInByZWZlcnJlZF91c2VybmFtZSI6I
k1lZ2FuQkBtaWNyb3NvZnRpZGVudGl0eS5kZXYiLCJyaCI6IkkiLCJyb2xlcyI6WyJBZG1pbiJdLCJzY3AiOiJDYXRhbG9n
LlZpZXcuQWxsIENhdGFsb2cuVmlldy5QdWJsaXNoZWQiLCJzdWIiOiIwWDdQdkVUNG9ySFJuUlluZE12QTRDWWx
ZeGdfQ293c0UxQkdUSUFLNmhFIiwidGlkIjoiYzcyYTI5NWQtZDdhNS00MWVhLWEzNTEtYjE1ZGQ5ZjY3MjE1IiwidX
RpIjoia2FlRkNSR3dtMFNPd01Obmp2SUhBQSIsInZlciI6IjIuMCJ9.gdp_cBRgJhfDo7d8_ooG1d41NCfLmLLwNxFTQl-
_hhskeHXg45DMFBTCyxBv1Kt5ggrxxKgfkAZTdrM01mzpw-DhA8TX7xi7altJJ-
iIjNMrRQVJI8F6UVZxMetgN_20184Ry9lbBDDOI0Ac6A8UnpV8uUbV08Aiiaj3ecjpX3vAaaTKuOQsdmzZEh9CK1REv
xn6ospUTkL2wzqMQ-E4h-wb30-
NpCqLuuaGITqpw45Sunn1hoq8fIZyzgCkVz65V_Udw4eTbKA1FGOg7nXiN_Pp0EKy1O6qiN77BQENA5PQOlcgRaO
edQ1N9eH6_qT4AavHKMkqrxfcFp2QAD-UR

Cactus API example

What does this Access token allow?

```
[Header].{
  "aud": "416684a7-0b52-4fa3-9918-e76d16542be2",
  "iss": "https://login.microsoftonline.com/c72a295d-d7a5-41ea-a351-b15dd9f67215/v2.0",
  "iat": 1563883336,
  "nbf": 1563883336,
  "exp": 1563887236,
  "aio": "ATQAy/8MAAAAaLtA/3Modx+ON+QMi502vrpWboHzsnwjCqbiZuOjq16lY1zU7bEMt9HQot5rFp5j",
  "azp": "bb764c21-49b8-49de-aa24-6c76d7dc800f",
  "azpacr": "0",
  "name": "Megan Bowen",
  "oid": "0e748cd0-5d2a-4918-a351-9549e75fd1dd",
  "preferred_username": "MeganB@microsoftidentity.dev",
  "rh": "I",
  "roles": [
    "Admin"
  ],
  "scp": "Catalog.View.All Catalog.View.Published",
  "sub": "0X7PvET4orHRnRYndMvA4CYlYxg_CowsE1BGTIAK6hE",
  "tid": "c72a295d-d7a5-41ea-a351-b15dd9f67215",
  "uti": "kaeFCRGwm0SOwMNnjvIHAA",
  "ver": "2.0"
}.[Signature]
```

Cactus API example

What does this Access token allow?

```
[Header].{
  "aud": "416684a7-0b52-4fa3-9918-e76d16542be2",
  "iss": "https://login.microsoftonline.com/c72a295d-d7a5-41ea-a351-b15dd9f67215/v2.0",
  "iat": 1563883336,
  "nbf": 1563883336,
  "exp": 1563887236,
  "aio": "ATQAy/8MAAAAaLtA/3Modx+ON+QMi502vrpWboHzsnwjCqbiZuOjq16lY1zU7bEMt9HQot5rFp5j",
  "azp": "bb764c21-49b8-49de-aa24-6c76d7dc800f",
  "azpacr": "0",
  "name": "Megan Bowen",
  "oid": "0e748cd0-5d2a-4918-a351-9549e75fd1dd",
  "preferred_username": "MeganB@microsoftidentity.dev",
  "rh": "I",
  "roles": [
    "Admin"
  ],
  "scp": "Catalog.View.All Catalog.View.Published",
  "sub": "0X7PvET4orHRnRYndMvA4CYlYxg_CowsE1BGTIAK6hE",
  "tid": "c72a295d-d7a5-41ea-a351-b15dd9f67215",
  "uti": "kaeFCRGwm0SOwMNnjvIHAA",
  "ver": "2.0"
}.[Signature]
```

Cactus API example

What does this Access token allow?

```
[Header].{
  "aud": "416684a7-0b52-4fa3-9918-e76d16542be2",
  "iss": "https://login.microsoftonline.com/c72a295d-d7a5-41ea-a351-b15dd9f67215/v2.0",
  "iat": 1563882662,
  "nbf": 1563882662,
  "exp": 1563886562,
  "aio": "42FgYBBvuZbJnh7xs0l8YXcOy7nLAA==",
  "azp": "bf9b8d73-dc3f-4c03-a928-7529d3cc3fd4",
  "azpacr": "1",
  "oid": "7d655a98-843c-4337-ad24-3a4b0922179e",
  "roles": [
    "Catalog.View.All"
  ],
  "sub": "7d655a98-843c-4337-ad24-3a4b0922179e",
  "tid": "c72a295d-d7a5-41ea-a351-b15dd9f67215",
  "uti": "pPPIBgD_AUqWqKVYHTUQAA",
  "ver": "2.0"
}.[Signature]
```

Cactus API example

What does this Access token allow?

```
[Header].{
  "aud": "416684a7-0b52-4fa3-9918-e76d16542be2",
  "iss": "https://login.microsoftonline.com/c72a295d-d7a5-41ea-a351-b15dd9f67215/v2.0",
  "iat": 1563882662,
  "nbf": 1563882662,
  "exp": 1563886562,
  "aio": "42FgYBBvuZbJnh7xs0l8YXcOy7nLAA==",
  "azp": "bf9b8d73-dc3f-4c03-a928-7529d3cc3fd4",
  "azpacr": "1",
  "oid": "7d655a98-843c-4337-ad24-3a4b0922179e",
  "roles": [
    "Catalog.View.All"
  ],
  "sub": "7d655a98-843c-4337-ad24-3a4b0922179e",
  "tid": "c72a295d-d7a5-41ea-a351-b15dd9f67215",
  "uti": "pPPIBgD_AUqWqKVYHTUQAA",
  "ver": "2.0"
}.[Signature]
```

Cactus API example

What does this Access token allow?

```
[Header].{
  "aud": "416684a7-0b52-4fa3-9918-e76d16542be2",
  "iss": "https://login.microsoftonline.com/c72a295d-d7a5-41ea-a351-b15dd9f67215/v2.0",
  "iat": 1563882662,
  "nbf": 1563882662,
  "exp": 1563886562,
  "aio": "42FgYBBvuZbJnh7xs0l8YXcOy7nLAA==",
  "azp": "bf9b8d73-dc3f-4c03-a928-7529d3cc3fd4",
  "azpacr": "1",
  "oid": "7d655a98-843c-4337-ad24-3a4b0922179e",
  "roles": [
    "Catalog.View.All"
  ],
  "sub": "7d655a98-843c-4337-ad24-3a4b0922179e",
  "tid": "c72a295d-d7a5-41ea-a351-b15dd9f67215",
  "uti": "pPPIBgD_AUqWqKVYHTUQAA",
  "ver": "2.0"
}.[Signature]
```

More tokens

Human Identities

Enterprise - Single Tenant for LOB Apps

```
{
    "typ": "JWT",
    "alg": "RS256",
    "kid": "1LTMzakihiRla_8z2BEJVXeWMqo"
  }.{
    "ver": "2.0",
    "iss": "https://login.microsoftonline.com/fa15d692-e9c7-4460-a743-29f2956fd429/v2.0",
    "sub": "AAAAAAAAAAAAAAAAAAAAAIkzqFVrSaSaFHy782bbtaQ",
    "aud": "6cb04018-a3f5-46a7-b995-940c78f5aef3",
    "exp": 1536361411,
    "iat": 1536274711,
    "nbf": 1536274711,
    "name": "Abe Lincoln",
    "preferred_username": "AbeLi@microsoft.com",
    "oid": "00000000-0000-0000-66f3-3332eca7ea81",
    "tid": "fa15d692-e9c7-4460-a743-29f2956fd429",
    "nonce": "123523",
    "aio": "Df2UVXL1ix!lMCWMSOJBcFatzcGfvFGhjKv8q5g0x732dR5MB5BisvGQO7YWByjd8iQDLq!eGbIDakyp5mnOrcdqHeYSnltepQm
Rp6AIZ8jY"
}.[Signature]
```

Human Identities

Enterprise – Multi Tenant for LOB Apps

```
{
    "typ": "JWT",
    "alg": "RS256",
    "kid": "1LTMzakihiRla_8z2BEJVXeWMqo"
  }.{
    "ver": "2.0",
    "iss": "https://login.microsoftonline.com/9188040d-6c67-4c5b-b112-36a304b66dad/v2.0",
    "sub": “BBBBBBBBBBBBBBBBBBBBBIkzqFVrSaSaFHy782bbtaQ",
    "aud": "6cb04018-a3f5-46a7-b995-940c78f5aef3",
    "exp": 1536361411,
    "iat": 1536274711,
    "nbf": 1536274711,
    "name": “Matthijs Hoekstra",
    "preferred_username": “matthijs@contoso.com",
    "oid": “12345678-0000-0000-66f3-3332eca550ab",
    "tid": "9188040d-6c67-4c5b-b112-36a304b66dad",
    "nonce": "123523",
    "aio": "Df2UVXL1ix!lMCWMSOJBcFatzcGfvFGhjKv8q5g0x732dR5MB5BisvGQO7YWByjd8iQDLq!eGbIDakyp5mnOrcdqHeYSnltepQm
Rp6AIZ8jY"
}.[Signature]
```

Human Identities

Enterprise - External B2B User from another tenant

```
{
    "typ": "JWT",
    "alg": "RS256",
    "kid": "1LTMzakihiRla_8z2BEJVXeWMqo"
  }.{
    "ver": "2.0",
    "iss": "https://login.microsoftonline.com/9188040d-6c67-4c5b-b112-36a304b66dad/v2.0",
    "sub": "AAAAAAAAAAAAAAAAAAAAAIkzqFVrSaSaFHy782bbtaQ",
    "aud": "6cb04018-a3f5-46a7-b995-940c78f5aef3",
    "exp": 1536361411,
    "iat": 1536274711,
    "nbf": 1536274711,
    “idp": "https://login.microsoftonline.com/3338040d-6c67-4c5b-b112-36a304b66dad/v2.0",
    "name": "Abe Lincoln",
    "preferred_username": "AbeLi@microsoft.com",
    "oid": "00000000-0000-0000-66f3-3332eca7ea81",
    "tid": "9188040d-6c67-4c5b-b112-36a304b66dad",
    "nonce": "123523",
    "aio": "Df2UVXL1ix!lMCWMSOJBcFatzcGfvFGhjKv8q5g0x732dR5MB5BisvGQO7YWByjd8iQDLq!eGbIDakyp5mnOrcdqHeYSnltepQm
Rp6AIZ8jY"
}.[Signature]
```

Human Identities

Enterprise - External B2B User from gmail

```
{
    "typ": "JWT",
    "alg": "RS256",
    "kid": "1LTMzakihiRla_8z2BEJVXeWMqo"
  }.{
    "ver": "2.0",
    "iss": "https://login.microsoftonline.com/9188040d-6c67-4c5b-b112-36a304b66dad/v2.0",
    "sub": "AAAAAAAAAAAAAAAAAAAAAIkzqFVrSaSaFHy782bbtaQ",
    "aud": "6cb04018-a3f5-46a7-b995-940c78f5aef3",
    "exp": 1536361411,
    "iat": 1536274711,
    "nbf": 1536274711,
    “idp": “gmail.com",
    “name": “Matthijs",
    “email": “matthijs@gmail.com",
    "oid": "00000000-0000-0000-66f3-3332eca7ea81",
    "tid": "9188040d-6c67-4c5b-b112-36a304b66dad",
    “unique_name": “google.com#matthijs@gmail.com",
    "nonce": "123523",
    "aio": "Df2UVXL1ix!lMCWMSOJBcFatzcGfvFGhjKv8q5g0x732dR5MB5BisvGQO7YWByjd8iQDLq!eGbIDakyp5mnOrcdqHeYSnltepQm
Rp6AIZ8jY"
}.[Signature]
```

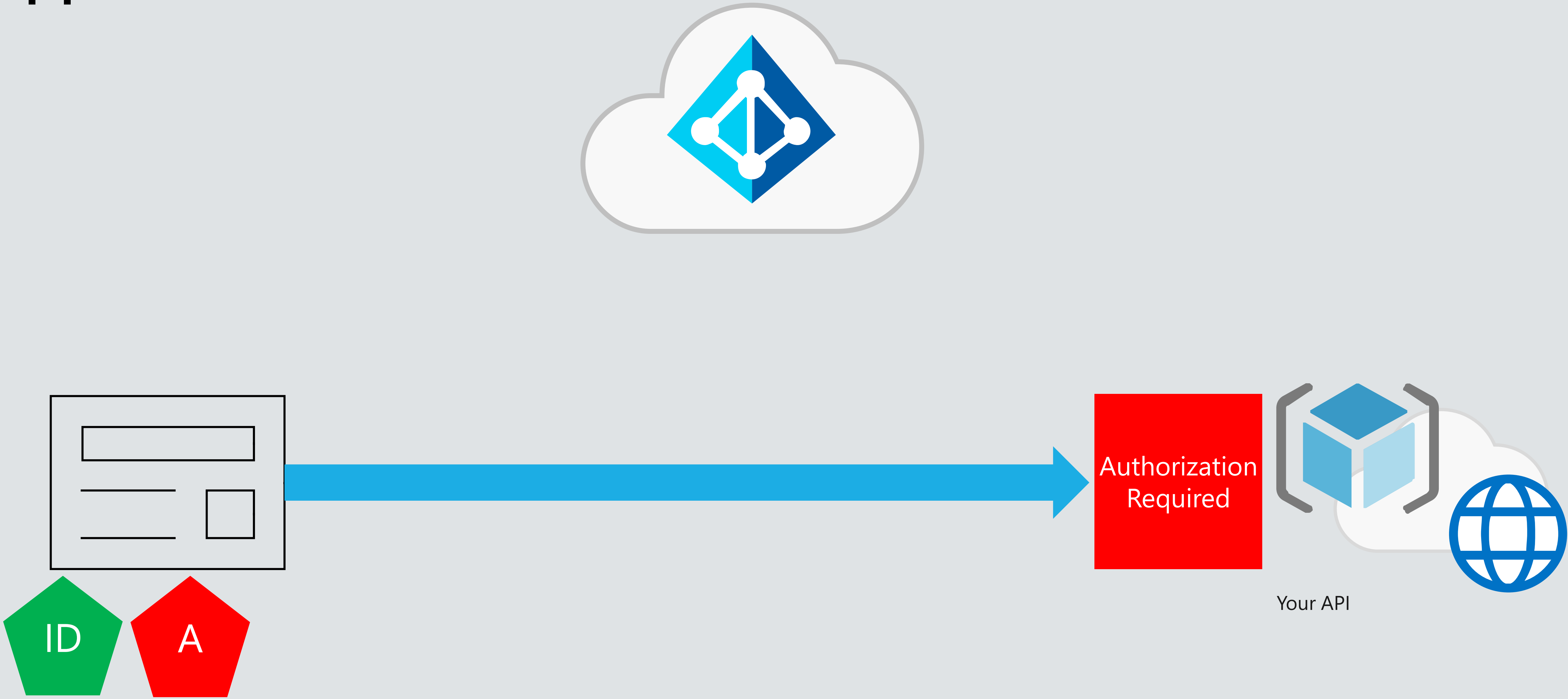
Human Identities

External identities - identity in B2C app

```
{
    "typ": "JWT",
    "alg": "RS256",
    "kid": "1LTMzakihiRla_8z2BEJVXeWMqo"
  }.{
    "ver": "2.0",
    "iss": "https://mahoekstb2c.b2clogin.com/fa15d692-e9c7-4460-a743-29f2956fd429/v2.0",
    "sub": "AAAAAAAAAAAAAAAAAAAAAIkzqFVrSaSaFHy782bbtaQ",
    "aud": “5cf15118-a3f5-46a7-b995-940c78f5aef3",
    "exp": 1536361411,
    "iat": 1536274711,
    "nbf": 1536274711,
    “idp": “facebook.com",
    “name": “Matthijs",
    “emails": [
          “matthijs@hoekstraonline.net"
          ],
    "oid": "00000000-0000-0000-66f3-3332eca7ea81",
    "tid": "fa15d692-e9c7-4460-a743-29f2956fd429",
    “unique_name": “google.com#matthijs@gmail.com",
    “customclaimsareawesomeinb2c": “TechoramaNL",
```

What happens when one API needs to call another API?

App has an authenticated user and a token to call API



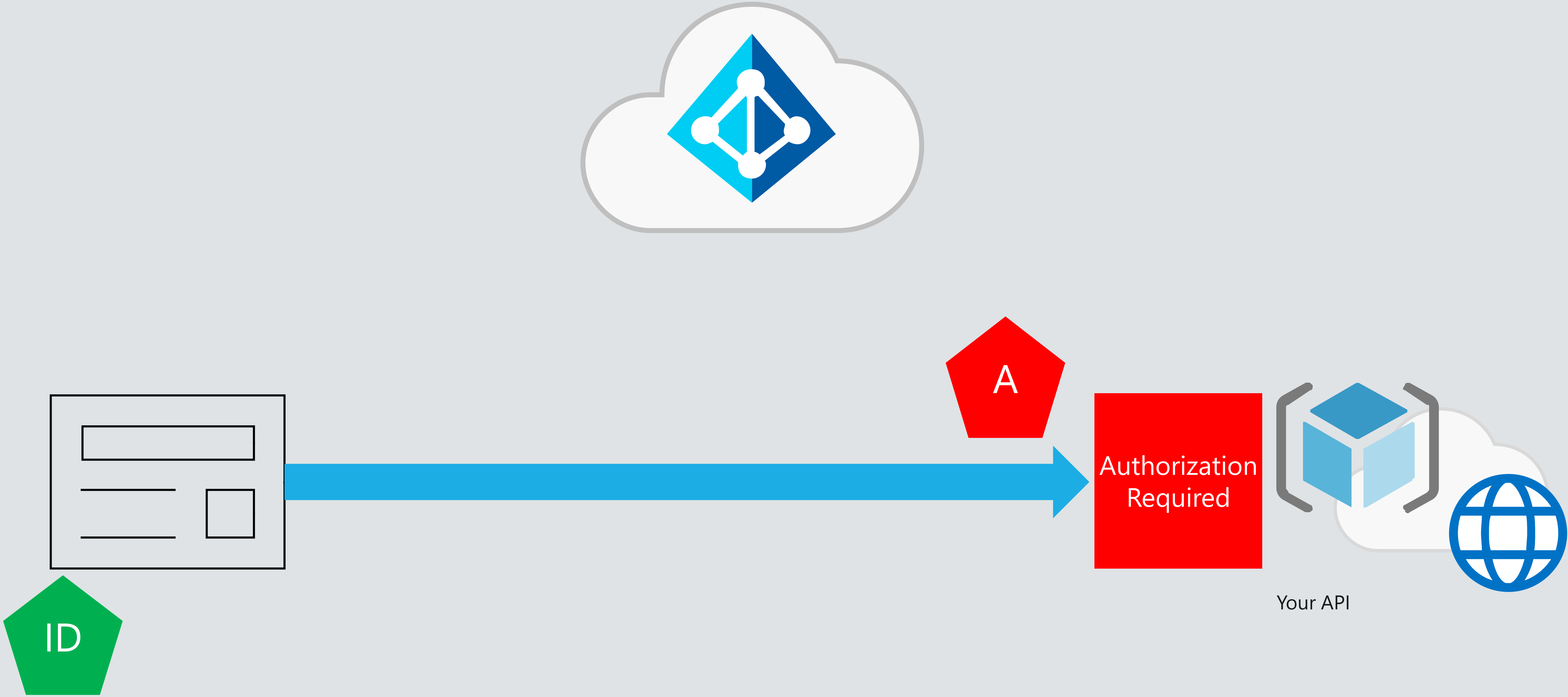
Authorization Required

Your API

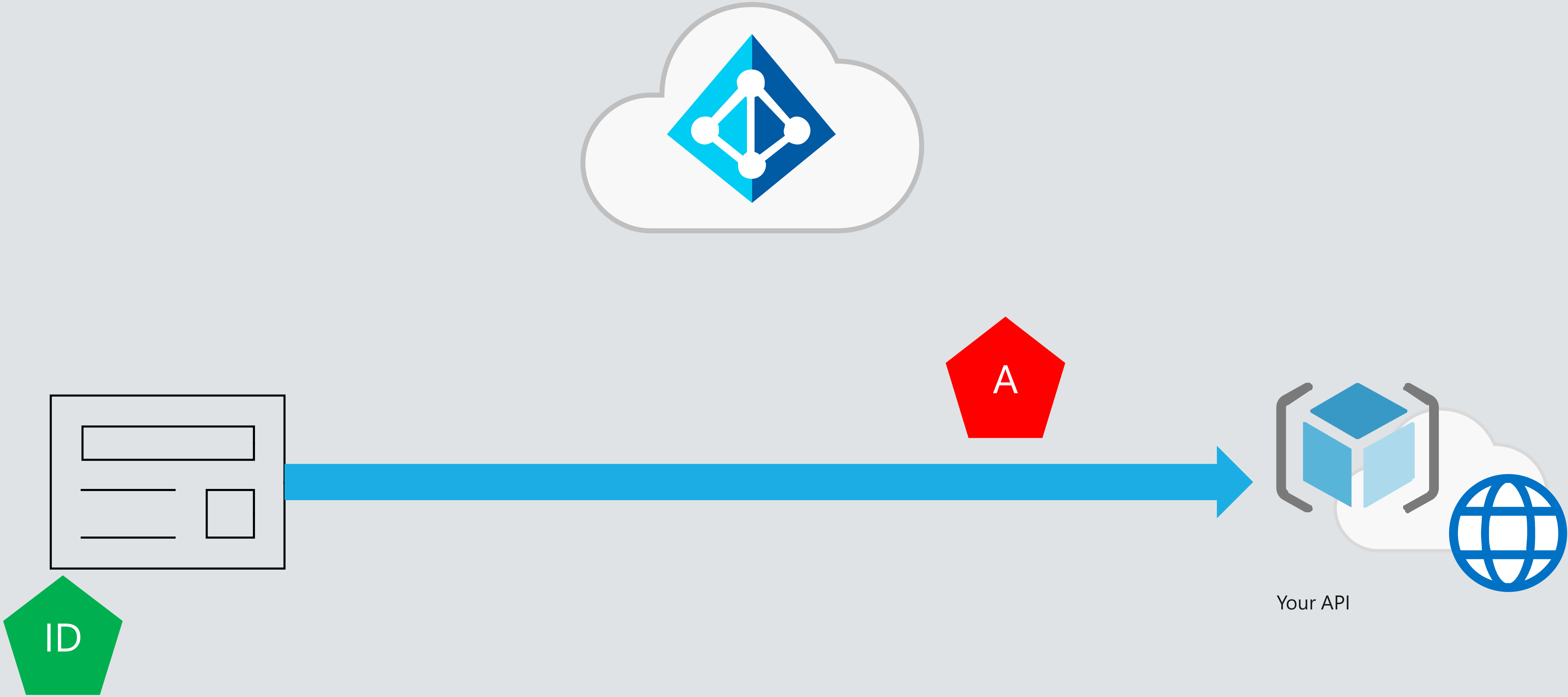
ID

A

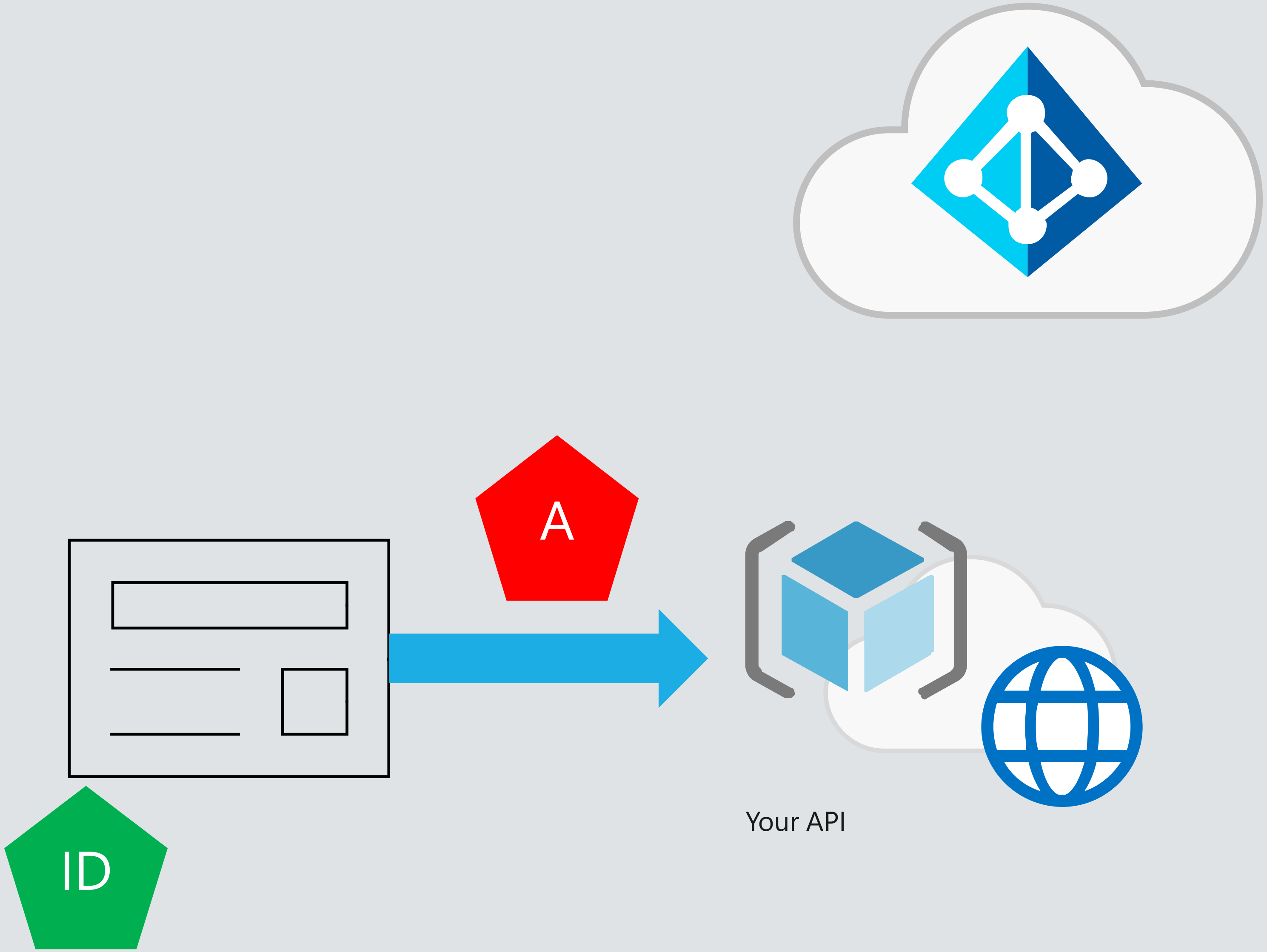
Call the API



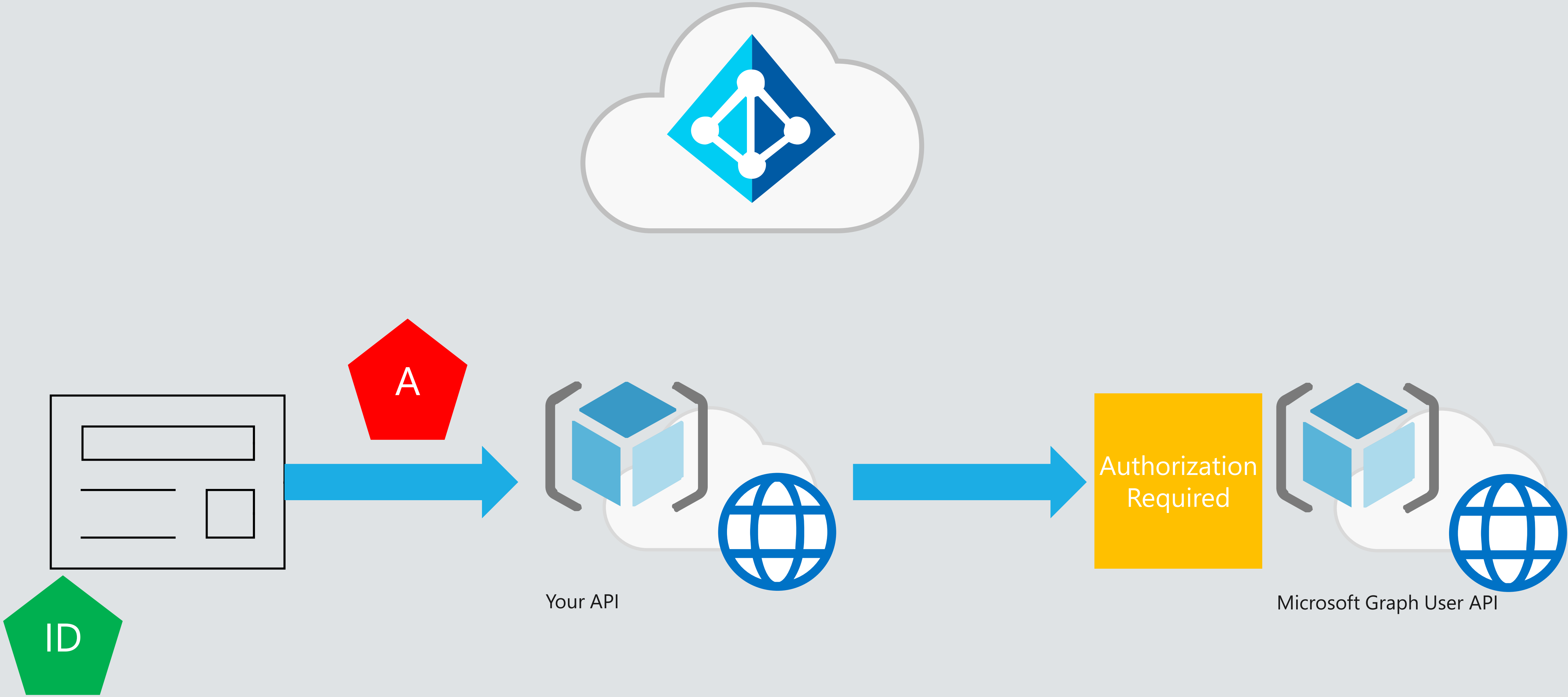
Call the API



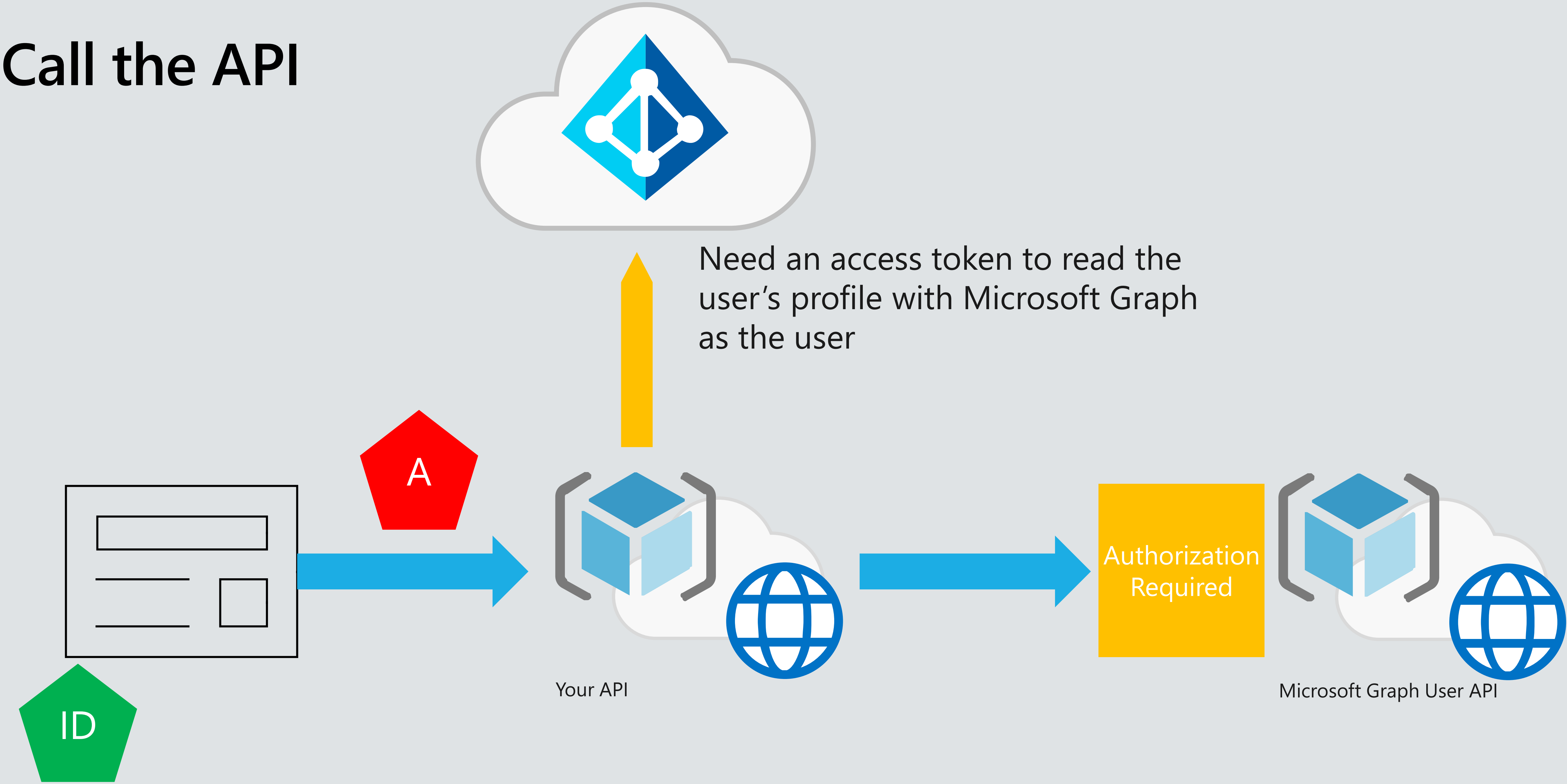
Call the API



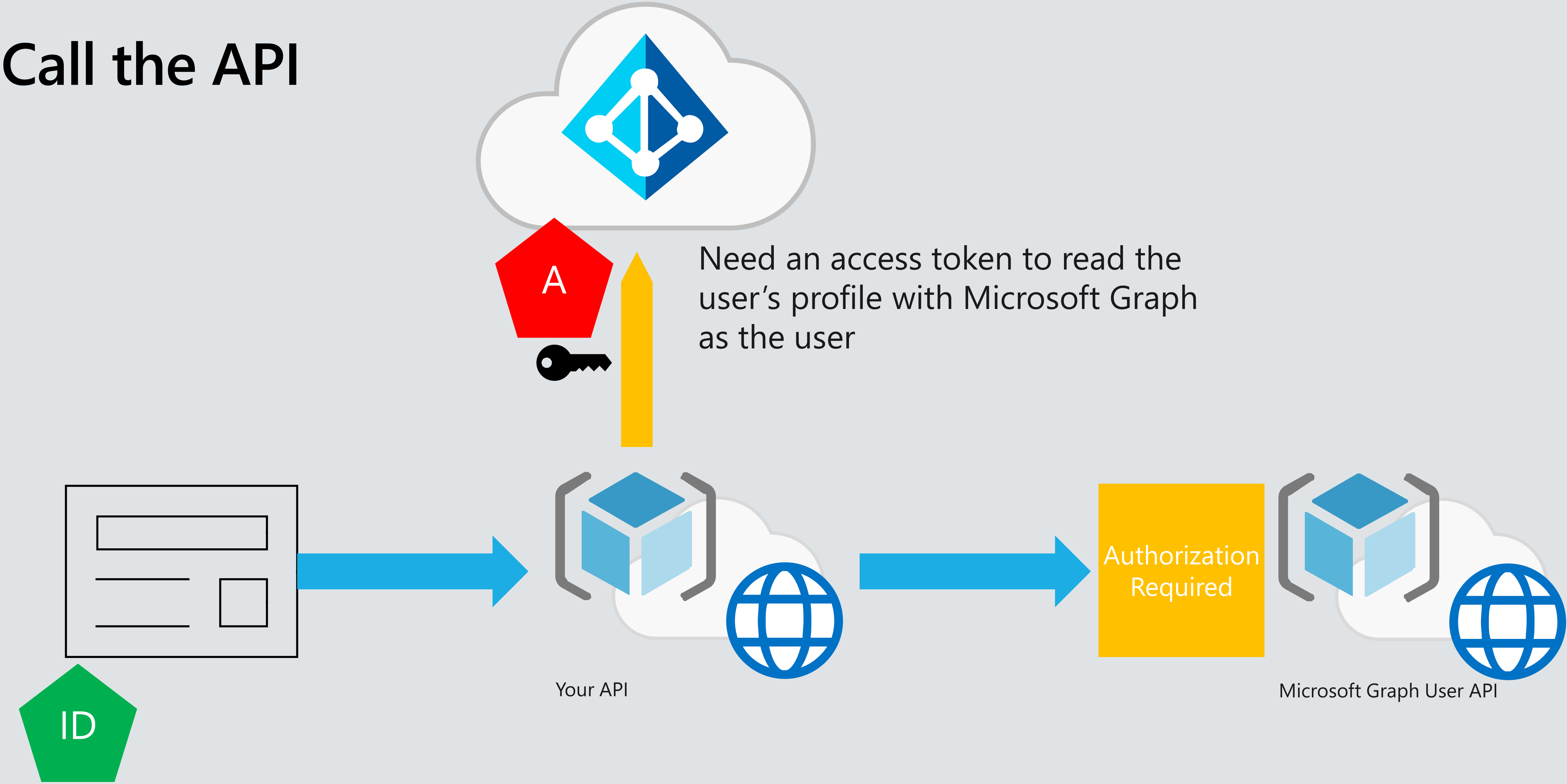
Call the API


A
ID
Authorization Required
Your API
Microsoft Graph User API

Call the API

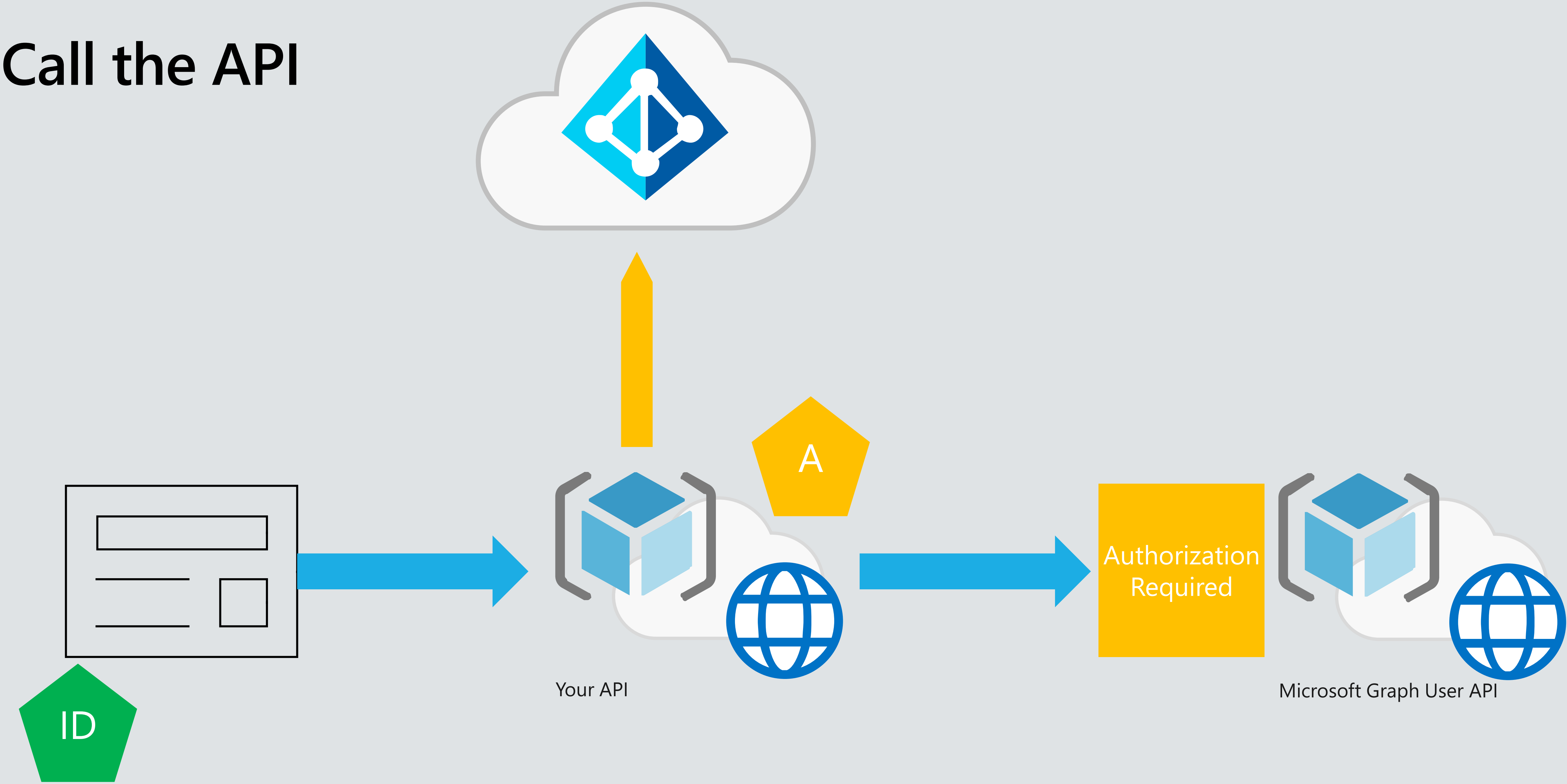

Need an access token to read the user's profile with Microsoft Graph as the user
A
ID
Your API
Authorization Required
Microsoft Graph User API

Call the API

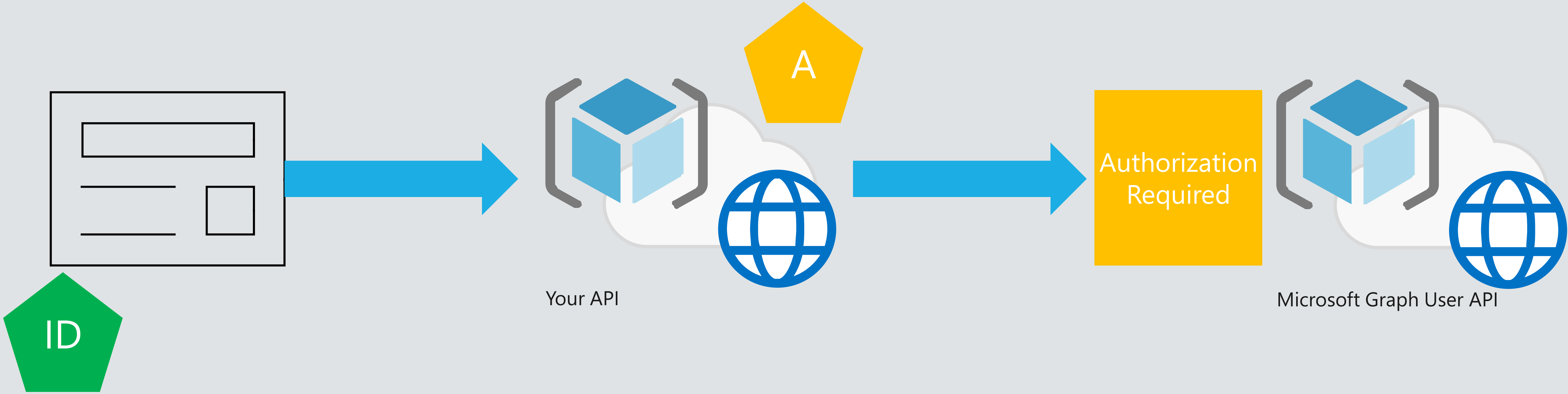


Need an access token to read the user’s profile with Microsoft Graph as the user

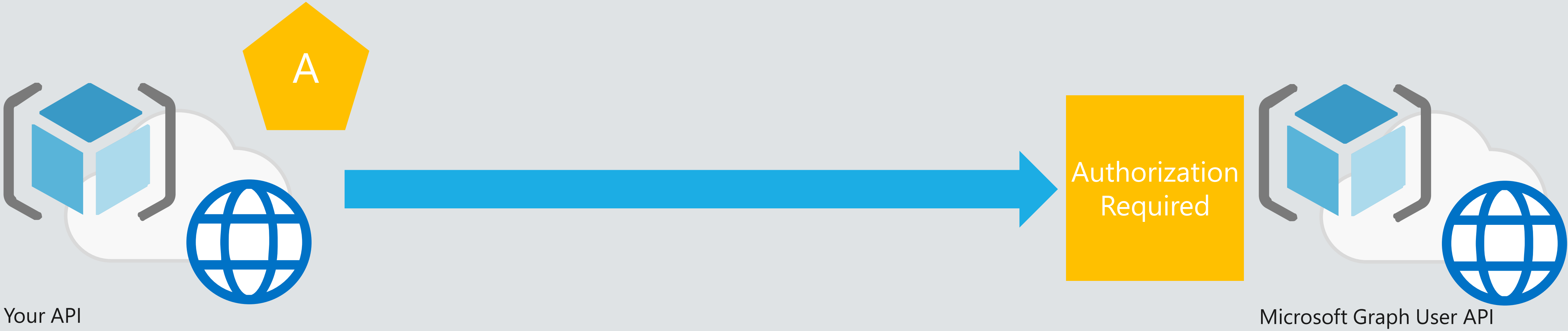
Call the API


A
Authorization Required
ID
Your API
Microsoft Graph User API

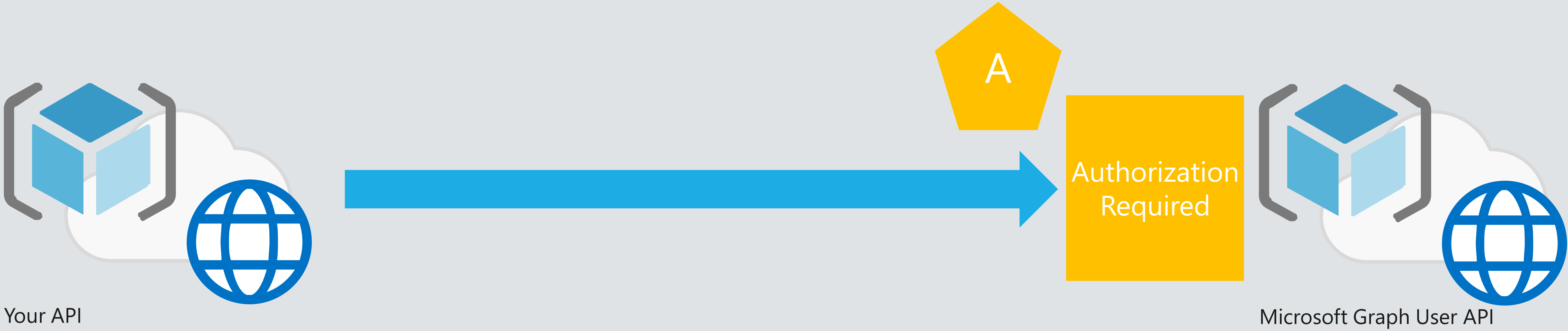
Call the API


A
Authorization Required
Your API
Microsoft Graph User API
ID

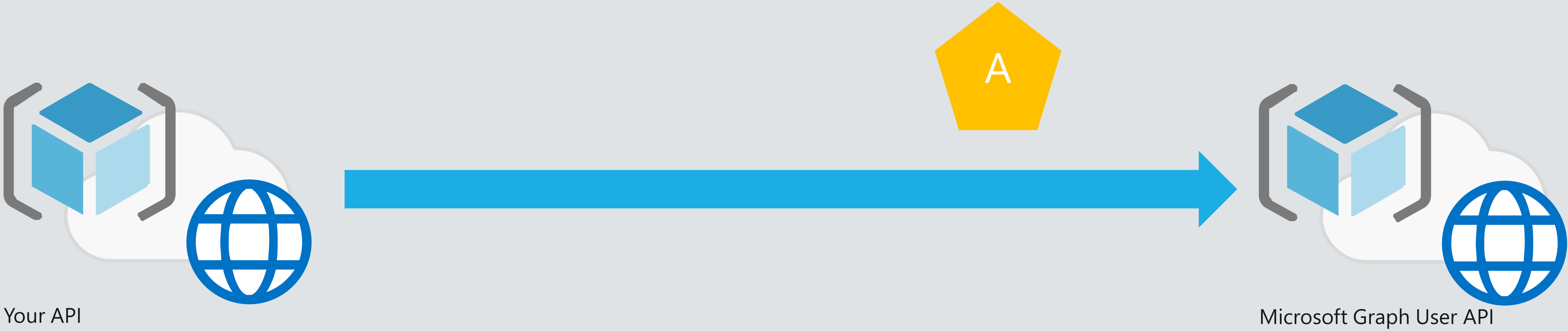
Call the API


A
Authorization Required
Your API
Microsoft Graph User API

Call the API


A
Authorization Required
Your API
Microsoft Graph User API

Call the API



Coding OBO (On Behalf Of)

```
app = ConfidentialClientApplicationBuilder.Create(config.ClientId)
          .WithClientSecret(config.ClientSecret) //TODO do this with cert ;)
          .Build();
```

Coding On Behalf Of (OBO)

```
X509Certificate2 certificate = ReadCertificate(config.CertificateName);
application = ConfidentialClientApplicationBuilder.Create(config.ClientId)
    .WithCertificate(certificate)
    .Build();

var result = await application.AcquireTokenOnBehalfOf(scopes, new
UserAssertion(accesstoken))
                    .ExecuteAsync()
                    .ConfigureAwait(false);
```

But wait, there is more

User roles and groups……

Please join the Authorization Session

Next Session:
Authorization
WEWC589 Thursday June 4th 14:00-15:00

Sign up for the webinar at
http://aka.ms/modernIdentityForDevelopers

Homework

Protecting API

Protecting your own API

Web => WebAPI on .NET Core

1. Go to: https://aka.ms/APILabCore
2. Download the whole repo and go to folder: 4-WebApp-your-API
3. Choose the 4-1-MyOrg (For Azure AD integration)

WPF Desktop Client => WebAPI on .NET Full Framework

1. Go to: https://aka.ms/APILabFull
2. Follow the instructions

Self Study to learn more

Scenario: Protected web API

- https://docs.microsoft.com/en-us/azure/active-directory/develop/scenario-protected-web-api-overview

How to: Sign in any Azure Active Directory user using the multi-tenant application pattern

- https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-convert-app-to-be-multi-tenant

Microsoft