HOW TO SECURELY IMPLEMENT MCP WITH OAUTH IN .NET

DAMIEN BOWDEN

Software Developer - **isolutions AG**

CÉDRIC MENDELIN

Software Developer - **isolutions AG**

AGENDA

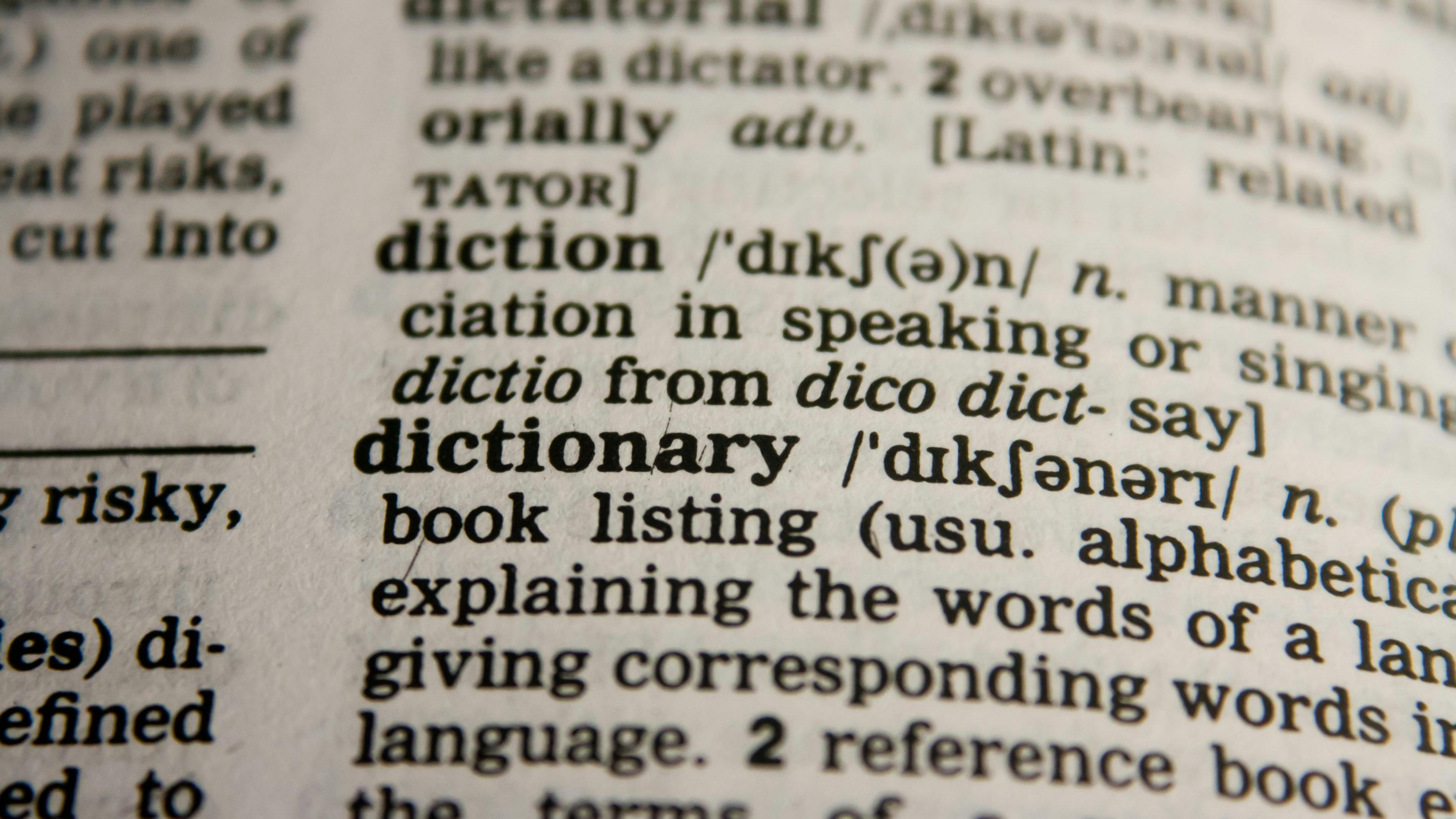
Getting started with MCP in dotnet

How to make it secure

Demo

Conclusion

GETTING STARTED



) one of
e played
eat risks,
cut into

g risky,

ies) di-
efined
ed to

dictatorial /ˌdɪktəˈtɔːrɪəl/ adj
like a dictator. 2 overbearing
orially *adv*. [Latin: related
TATOR]

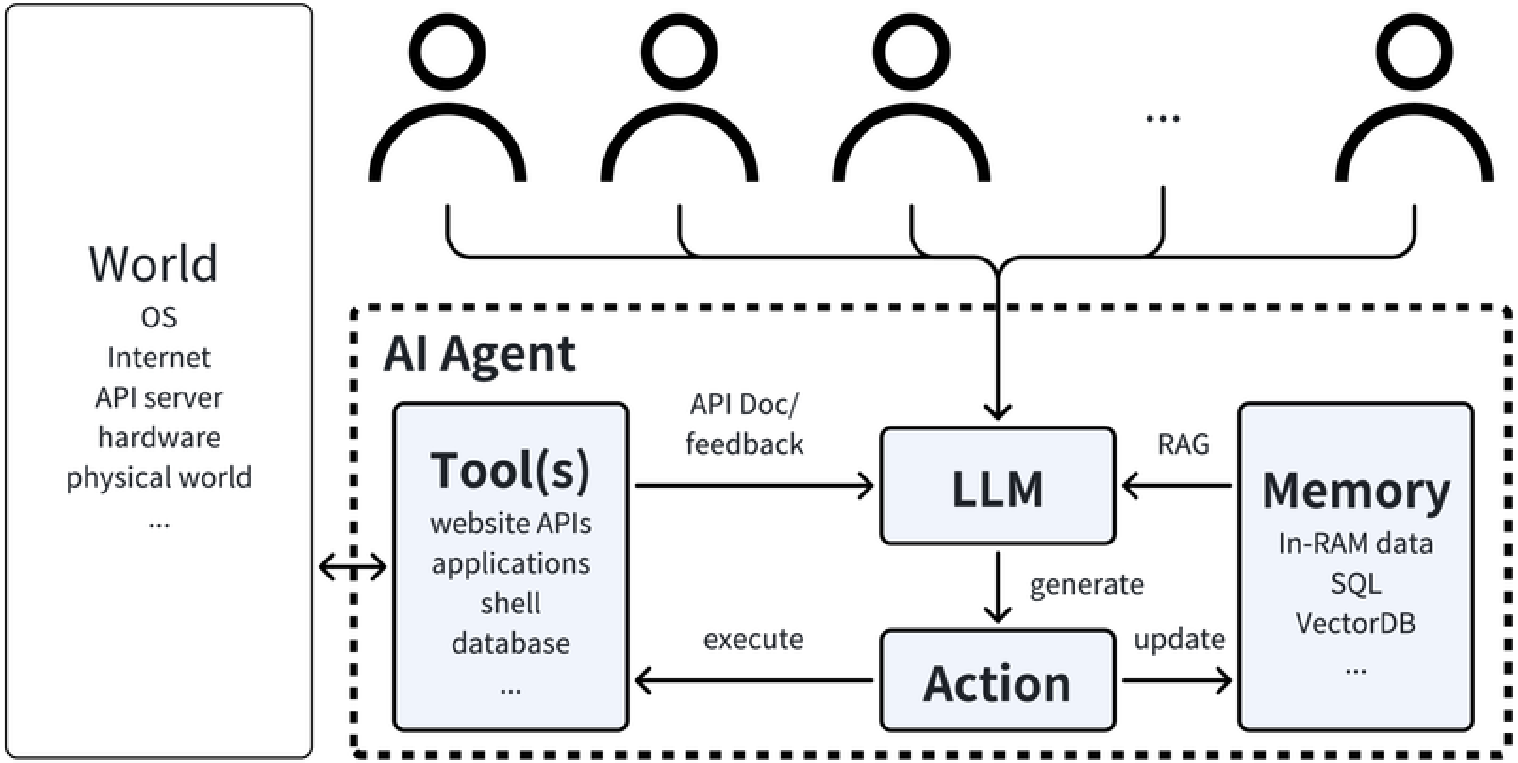
diction /ˈdɪkʃ(ə)n/ *n*. manner
ciation in speaking or singing
dictio from *dico dict-* say]

dictionary /ˈdɪkʃənərɪ/ *n*. (p
book listing (usu. alphabetica
explaining the words of a lan
giving corresponding words in
language. 2 reference book e
the terms of a

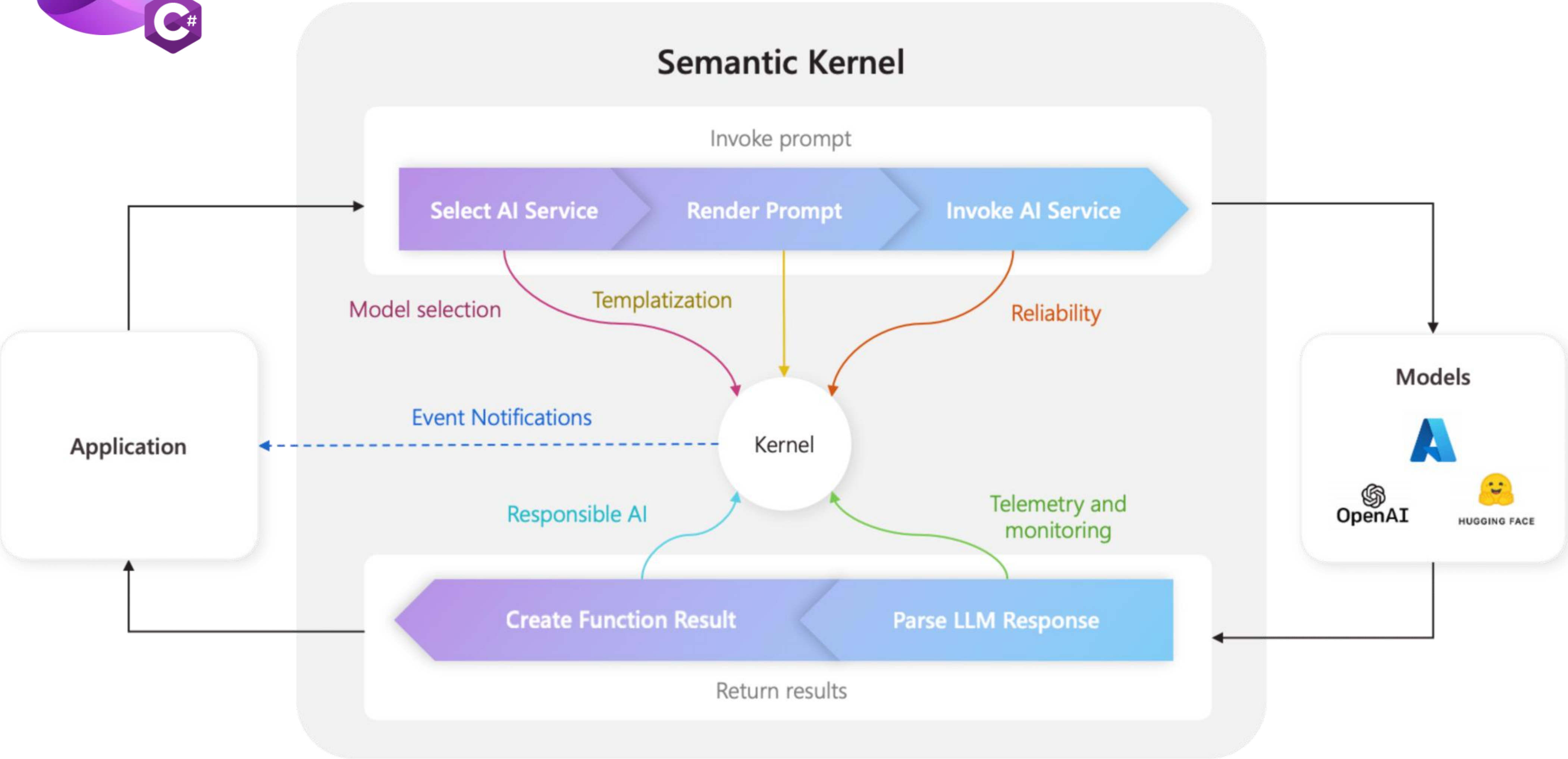
What is the current date?

1+1=3

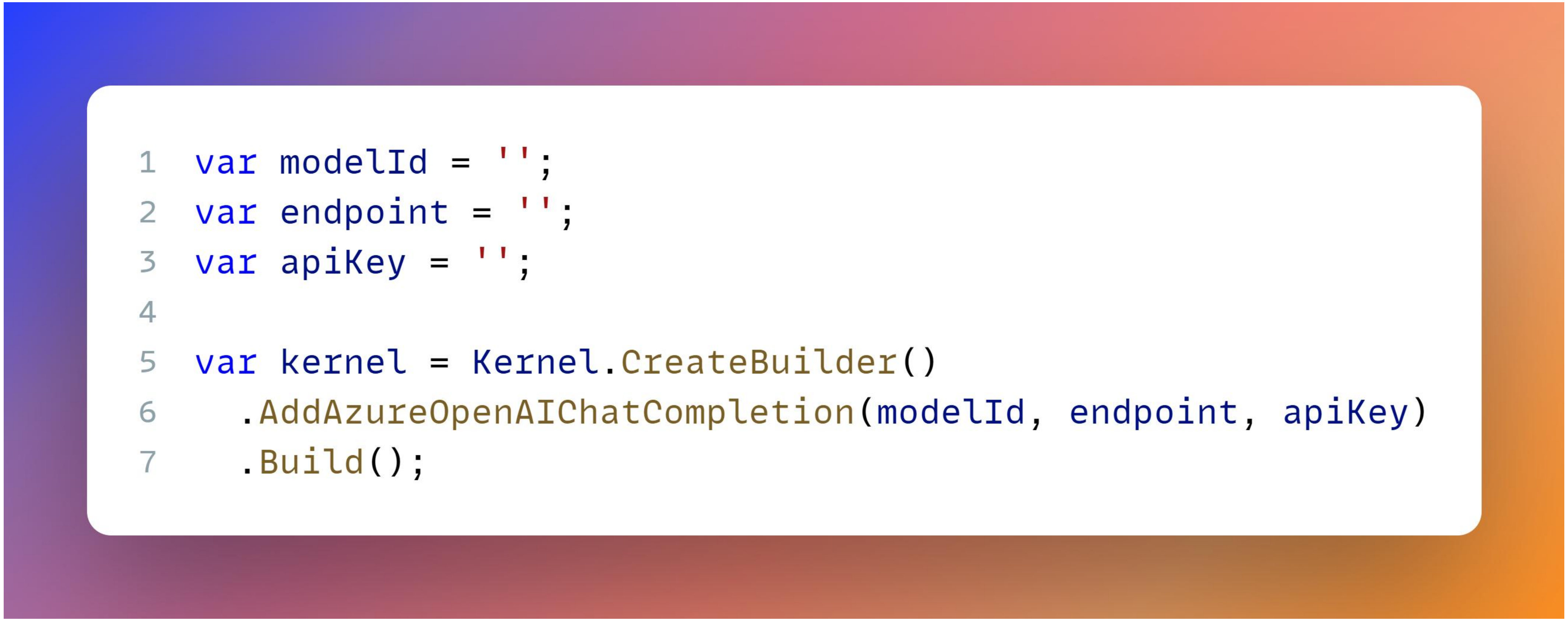
LLM BASED AI AGENT



Security of AI Agents, He et al. 2024
https://arxiv.org/pdf/2406.08689

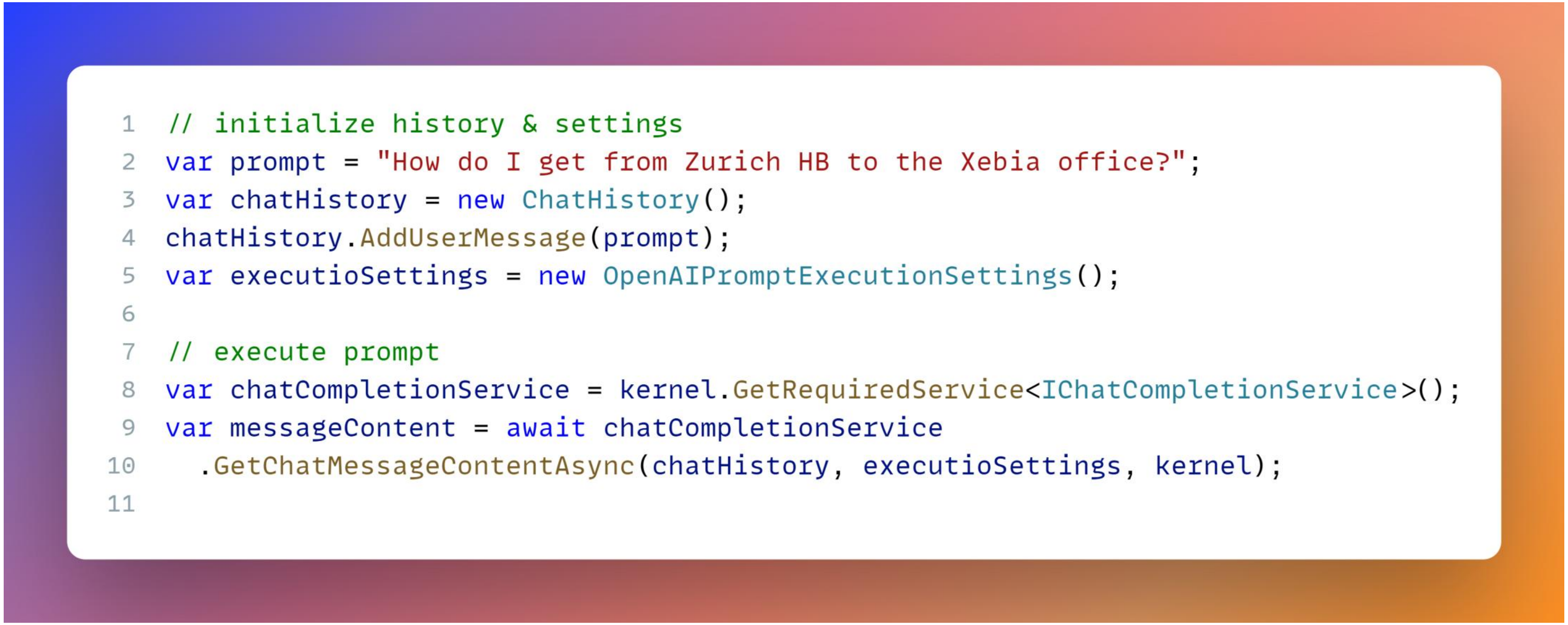

Semantic Kernel
Invoke prompt
Select AI Service
Render Prompt
Invoke AI Service
Model selection
Templatization
Reliability
Application
Event Notifications
Kernel
Responsible AI
Telemetry and monitoring
Models
OpenAI
HUGGING FACE
Create Function Result
Parse LLM Response
Return results

SK – SETUP



```
var modelId = '';
var endpoint = '';
var apiKey = '';

var kernel = Kernel.CreateBuilder()
  .AddAzureOpenAIChatCompletion(modelId, endpoint, apiKey)
  .Build();
```

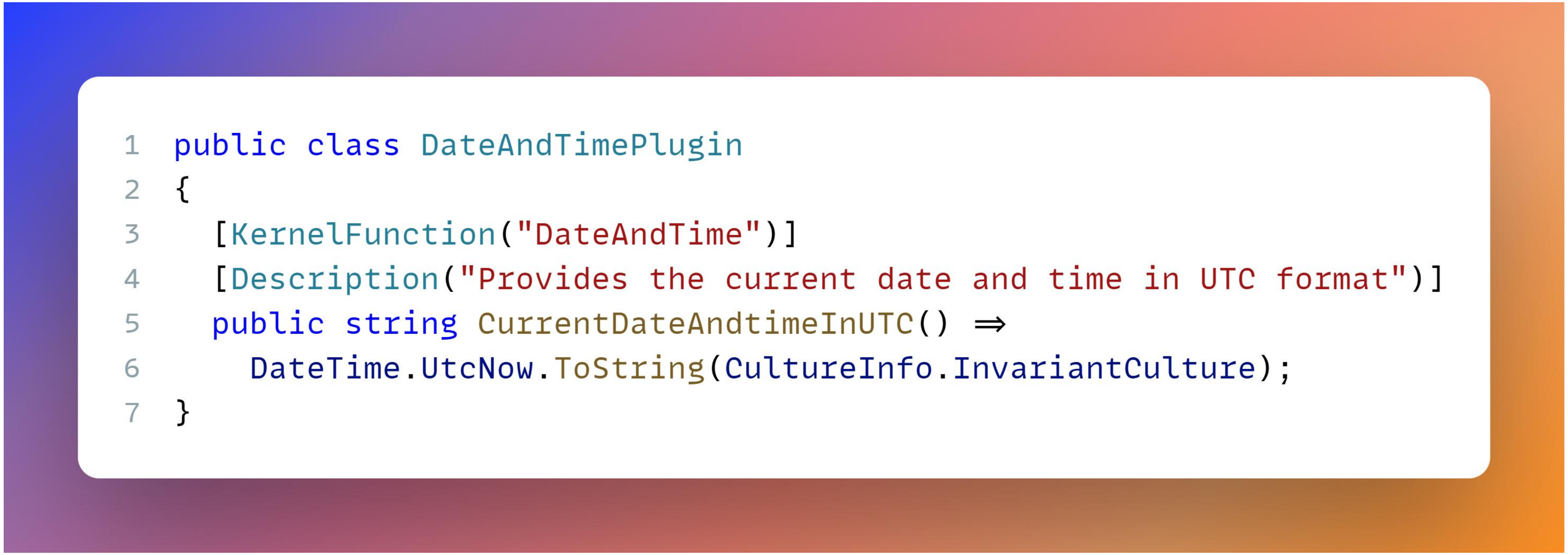
SK – EXECUTE PROMPT



```
// initialize history & settings
var prompt = "How do I get from Zurich HB to the Xebia office?";
var chatHistory = new ChatHistory();
chatHistory.AddUserMessage(prompt);
var executioSettings = new OpenAIPromptExecutionSettings();

// execute prompt
var chatCompletionService = kernel.GetRequiredService<IChatCompletionService>();
var messageContent = await chatCompletionService
  .GetChatMessageContentAsync(chatHistory, executioSettings, kernel);

```

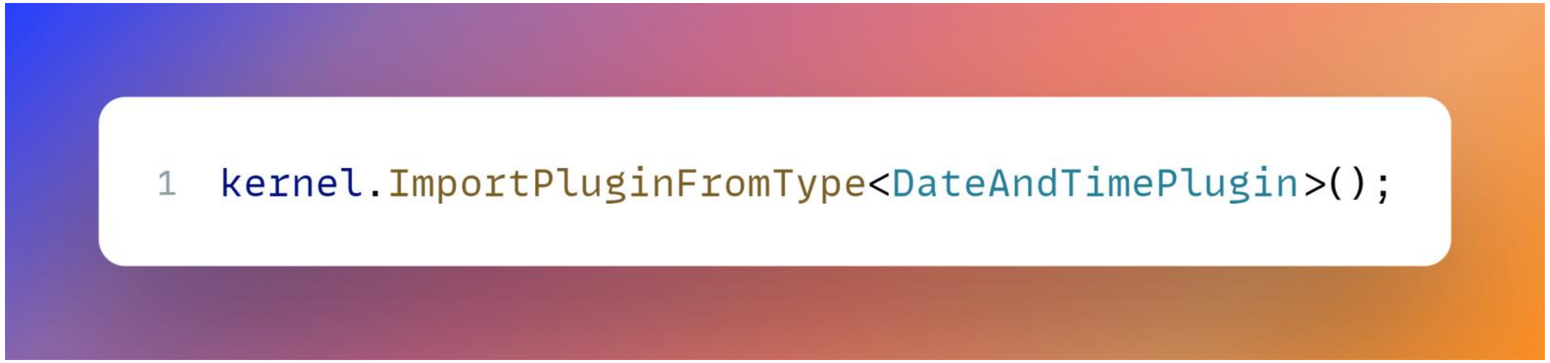
What is the current date?

SK – PLUGIN

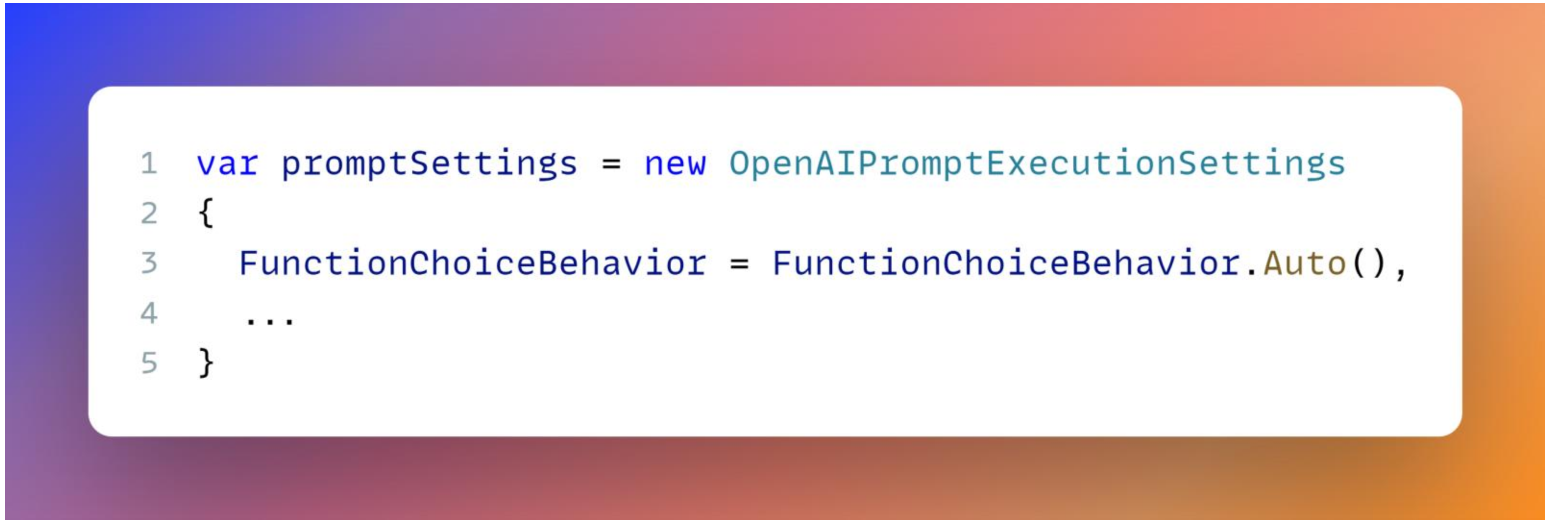


```csharp
public class DateAndTimePlugin
{
  [KernelFunction("DateAndTime")]
  [Description("Provides the current date and time in UTC format")]
  public string CurrentDateAndtimeInUTC() =>
    DateTime.UtcNow.ToString(CultureInfo.InvariantCulture);
}
```

SK – PLUGIN



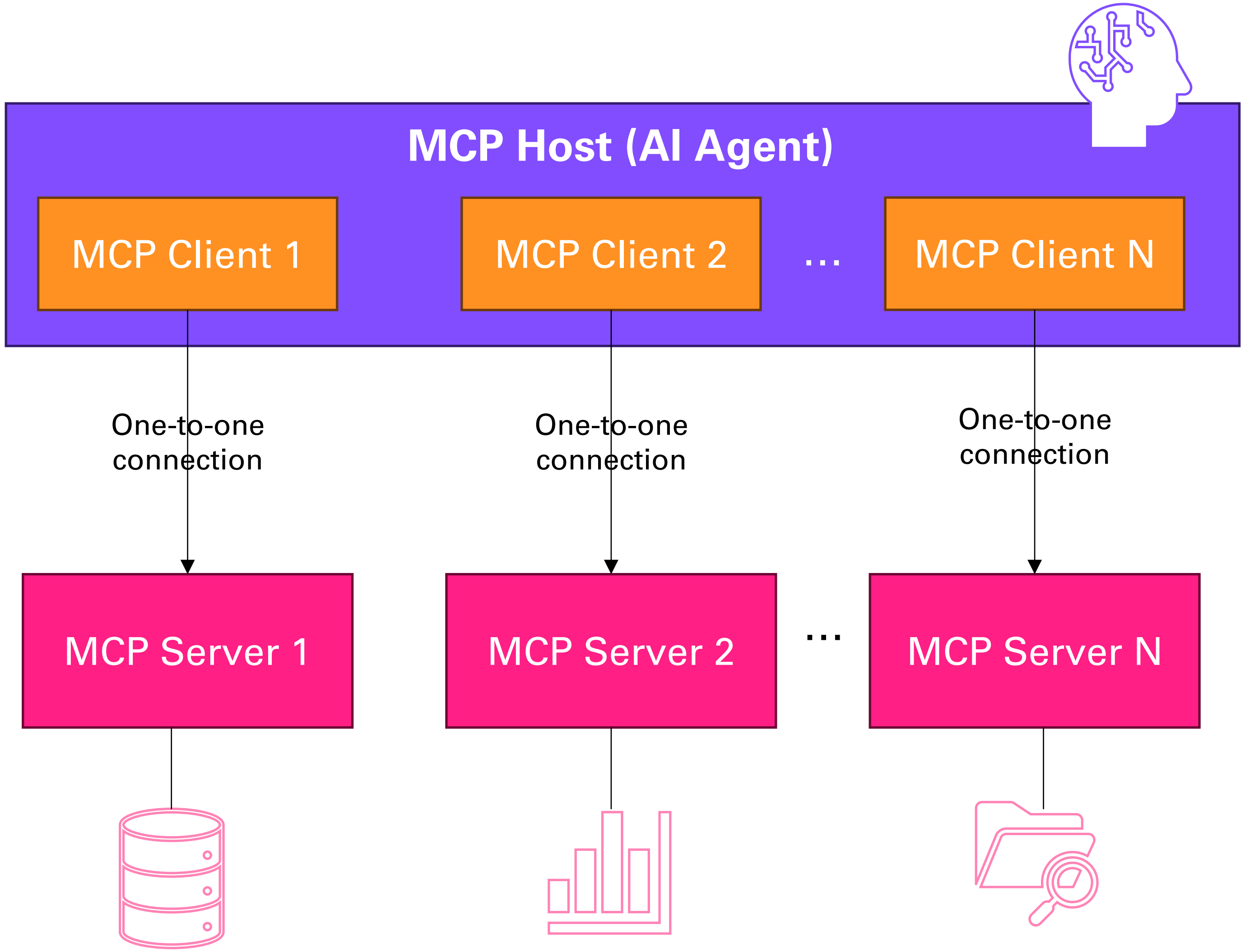
```
kernel.ImportPluginFromType<DateAndTimePlugin>();
```



```
var promptSettings = new OpenAIPromptExecutionSettings
{
  FunctionChoiceBehavior = FunctionChoiceBehavior.Auto(),
  ...
}
```

MODEL CONTEXT PROTOCOL (MCP)

- Client – Server
- Transport
  - Stdio (local)
  - Http (remotely)
    - Server-Sent Events for Streaming
- JSON-RPC



MCP CAPABILITIES

MCP Server

- **Tools**
- Resources
- Prompts
- Utilities
  - Completion
  - Logging
  - Pagination

MCP Client

- Roots
- Elicitation
- Sampling

https://modelcontextprotocol.io

USE-CASES

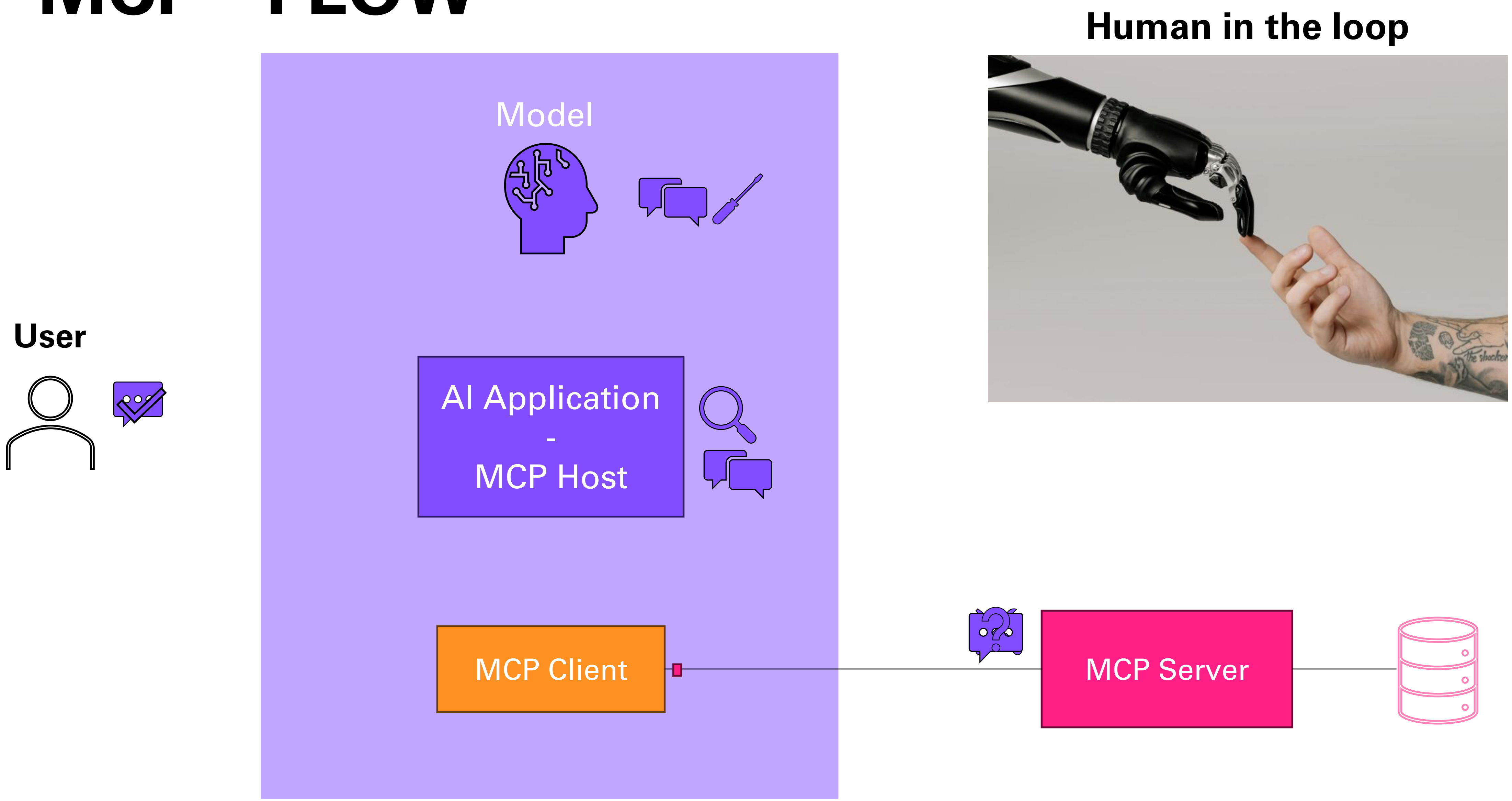
Developer Use Case

- Server locally – stdio
- Integrated in AI Coding assistant

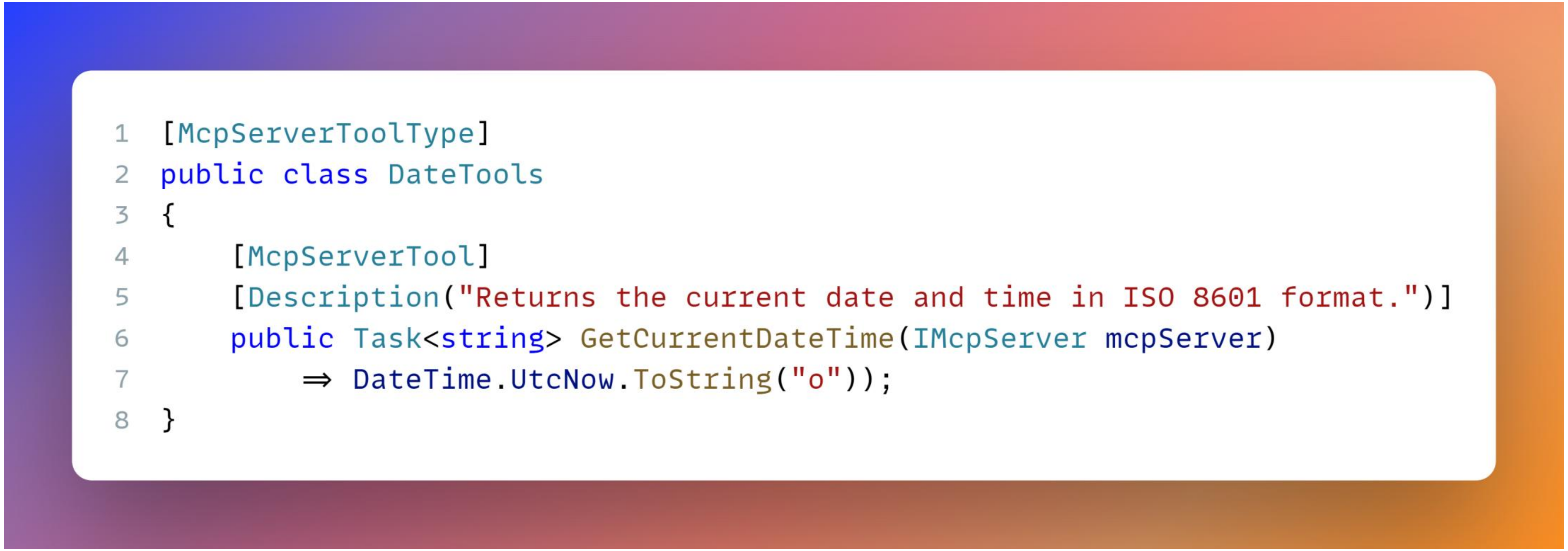
Business Use Case

- Server remotely – http - **API**
- Integrated in your AI assistant

MCP - FLOW

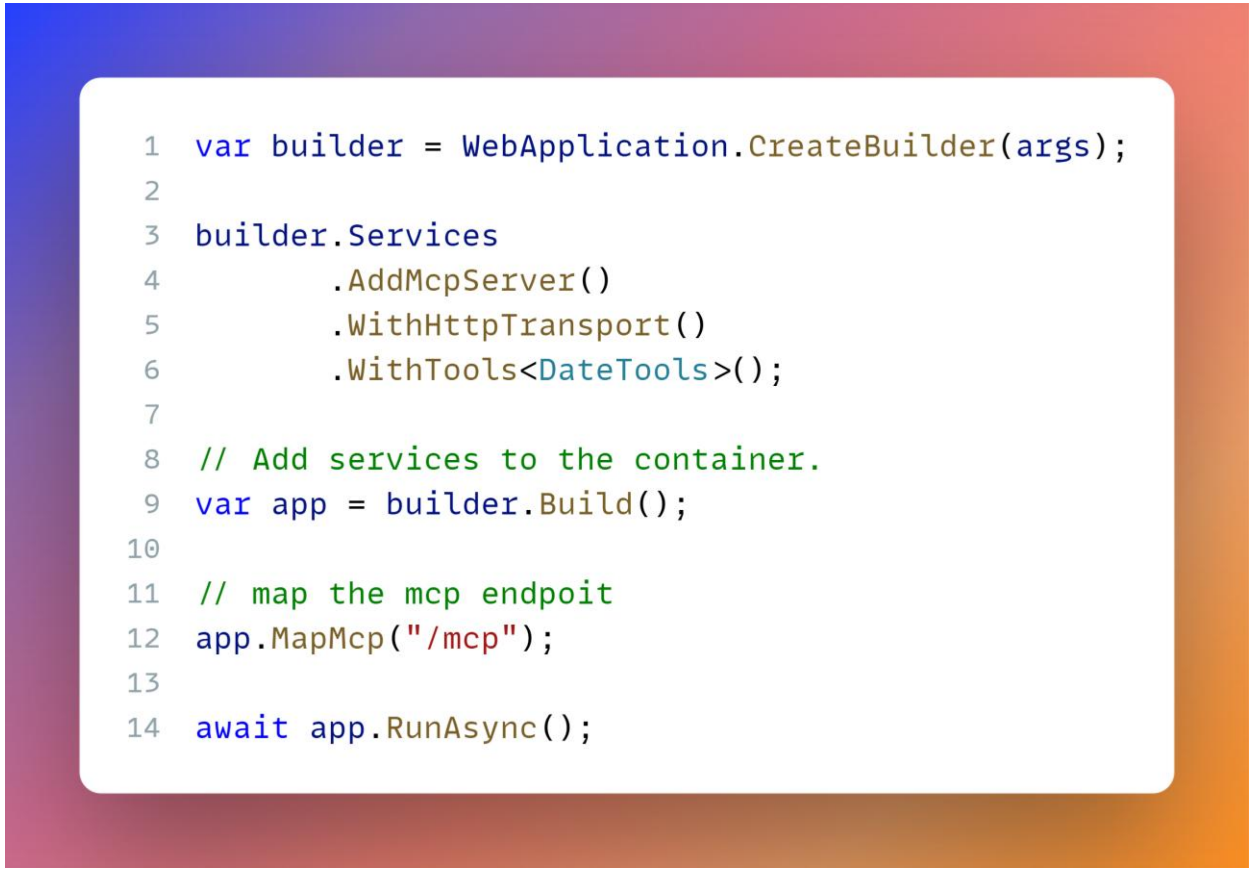

Human in the loop
Model
User
AI Application
-
MCP Host
MCP Client
MCP Server

MCP SERVER - TOOL



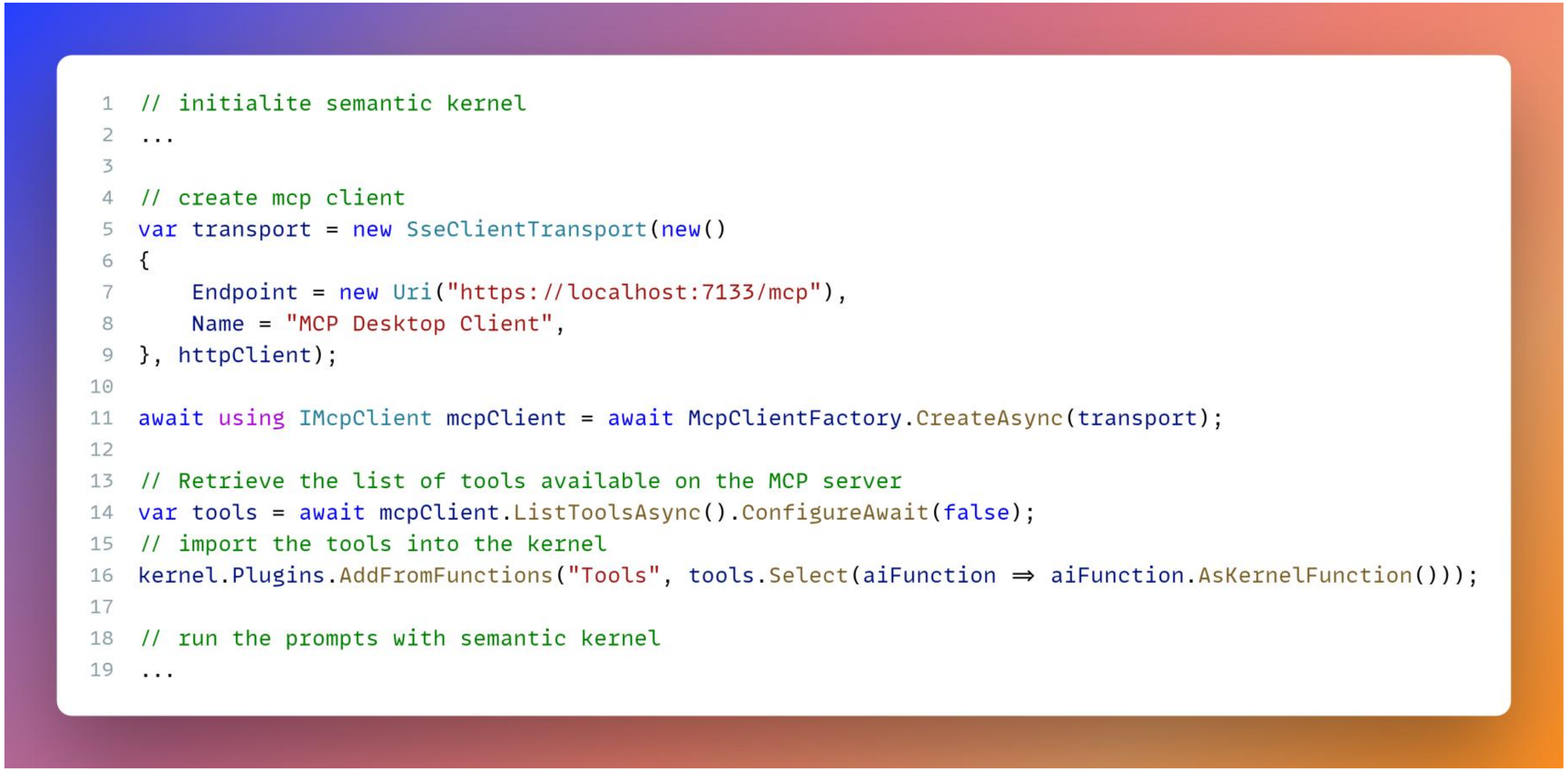
```
[McpServerToolType]
public class DateTools
{
    [McpServerTool]
    [Description("Returns the current date and time in ISO 8601 format.")]
    public Task<string> GetCurrentDateTime(IMcpServer mcpServer)
        => DateTime.UtcNow.ToString("o"));
}
```

MCP SERVER



```
var builder = WebApplication.CreateBuilder(args);

builder.Services
       .AddMcpServer()
       .WithHttpTransport()
       .WithTools<DateTools>();

// Add services to the container.
var app = builder.Build();

// map the mcp endpoit
app.MapMcp("/mcp");

await app.RunAsync();
```

AI AGENT & MCP CLIENT



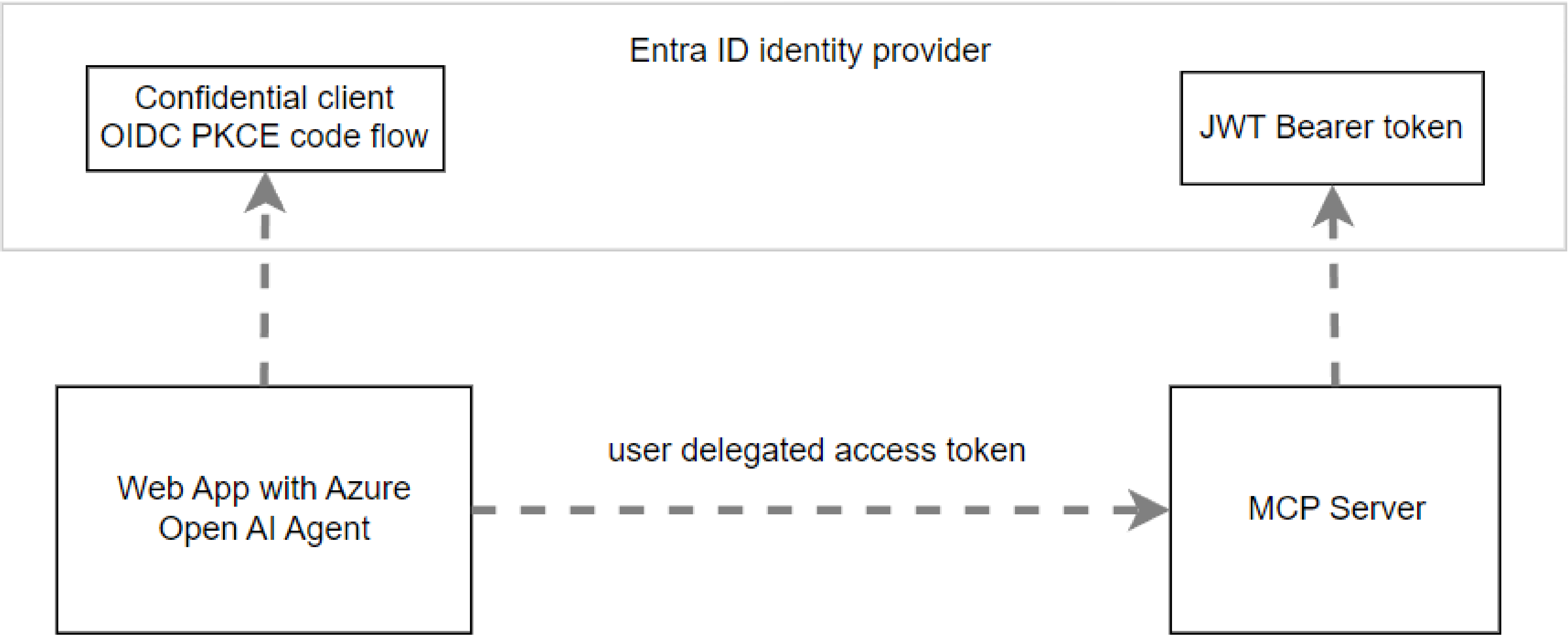
```
// initialite semantic kernel
...

// create mcp client
var transport = new SseClientTransport(new()
{
    Endpoint = new Uri("https://localhost:7133/mcp"),
    Name = "MCP Desktop Client",
}, httpClient);

await using IMcpClient mcpClient = await McpClientFactory.CreateAsync(transport);

// Retrieve the list of tools available on the MCP server
var tools = await mcpClient.ListToolsAsync().ConfigureAwait(false);
// import the tools into the kernel
kernel.Plugins.AddFromFunctions("Tools", tools.Select(aiFunction => aiFunction.AsKernelFunction()));

// run the prompts with semantic kernel
...
```

SECURE MCP SERVERS

SECURITY BEST PRACTICES

- Use OpenID Connect to acquire the access token
- Use delegated user access tokens only
- Follow a zero-trust strategy for applications
- Don’t trust MCP servers, always validate the security first
- Don't use public clients on the web

SECURITY SETUP



Entra ID identity provider

Confidential client
OIDC PKCE code flow

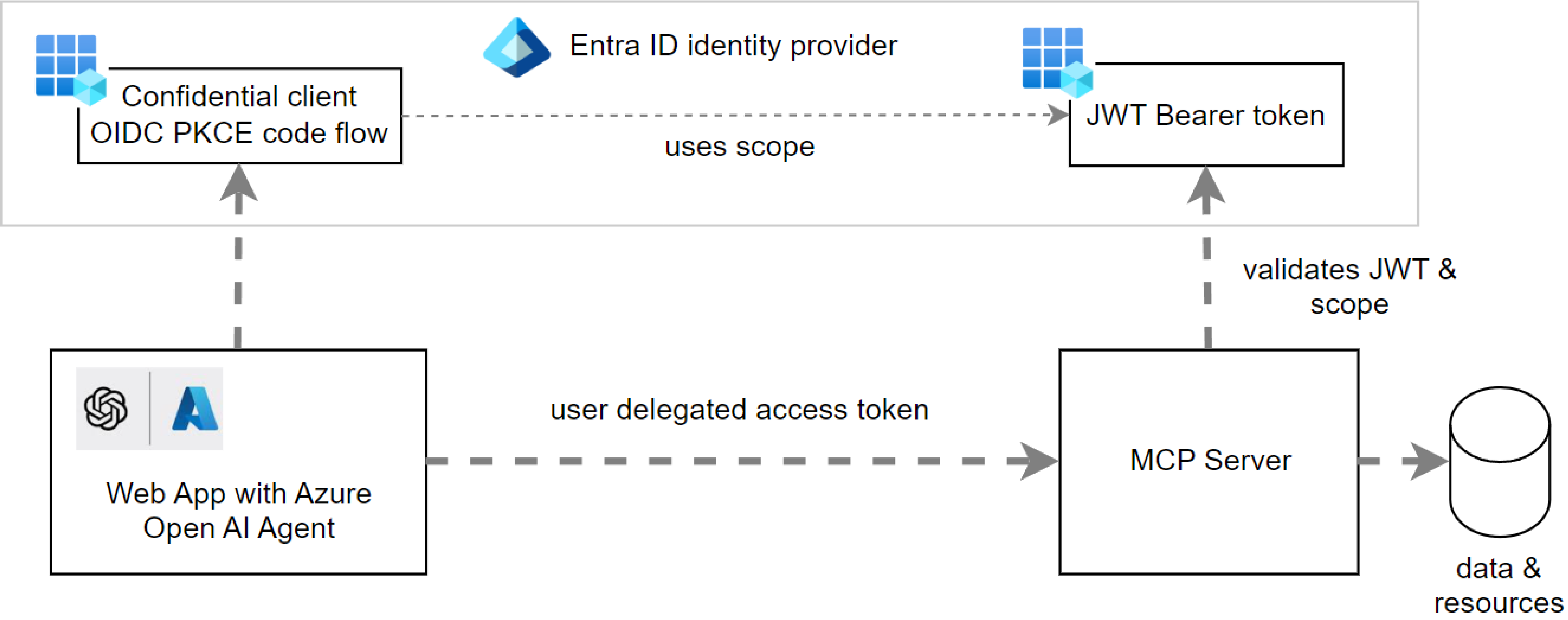
JWT Bearer token

Web App with Azure
Open AI Agent

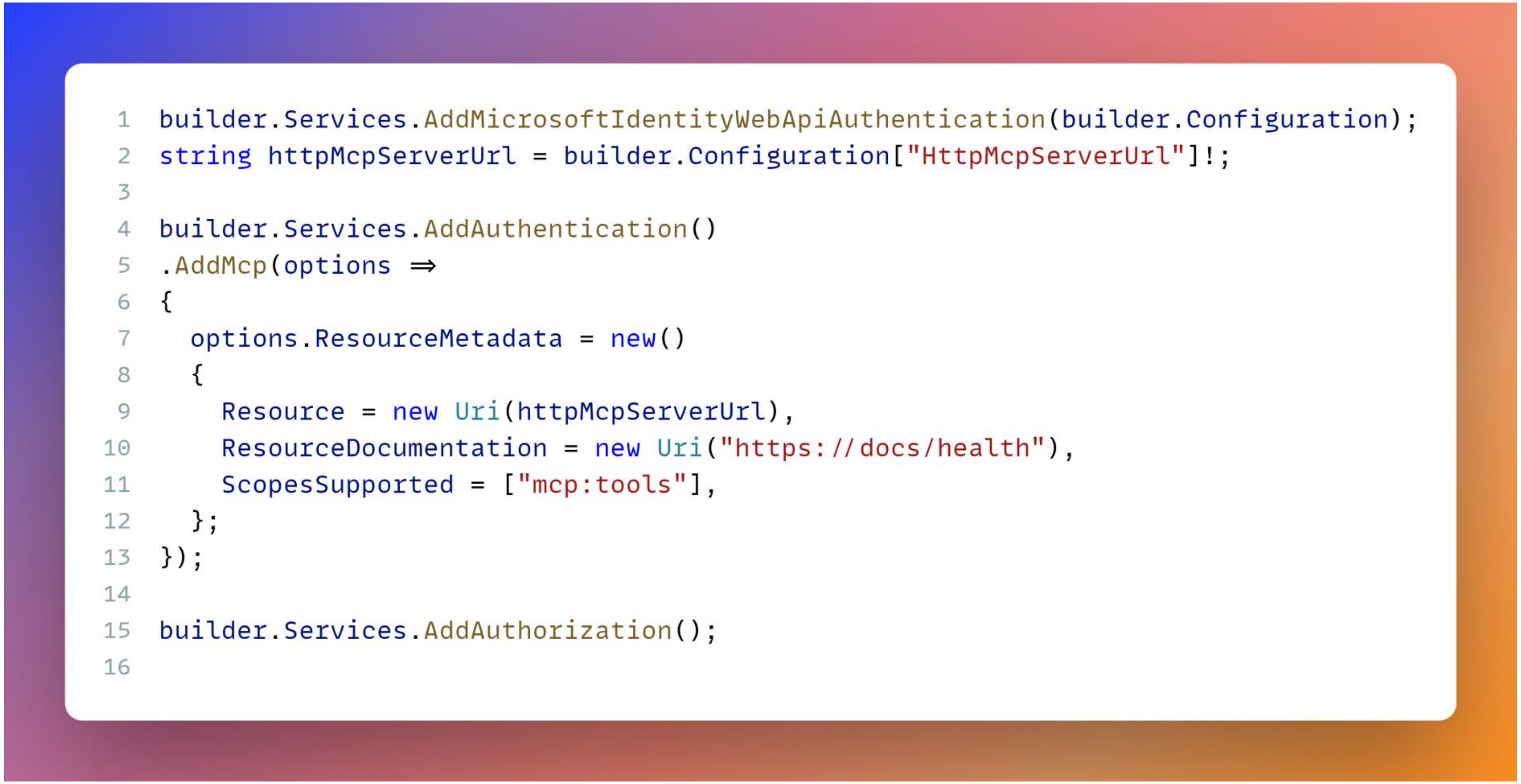
user delegated access token

MCP Server

MICROSOFT ENTRA ID

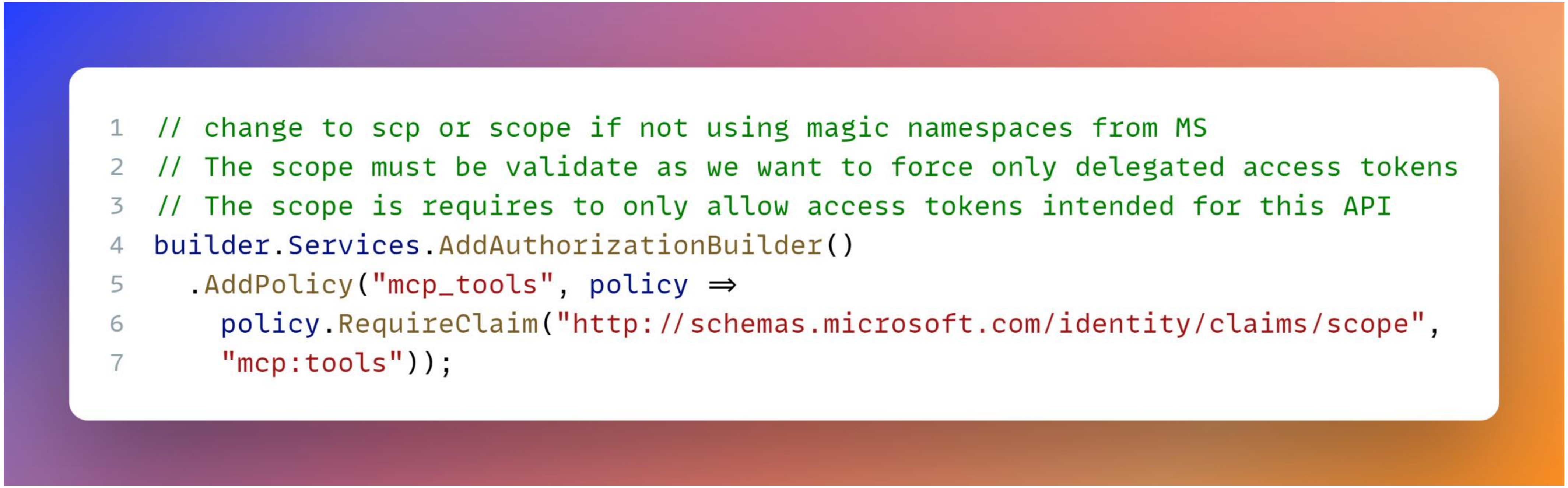


MCP SERVER OAUTH SECURITY



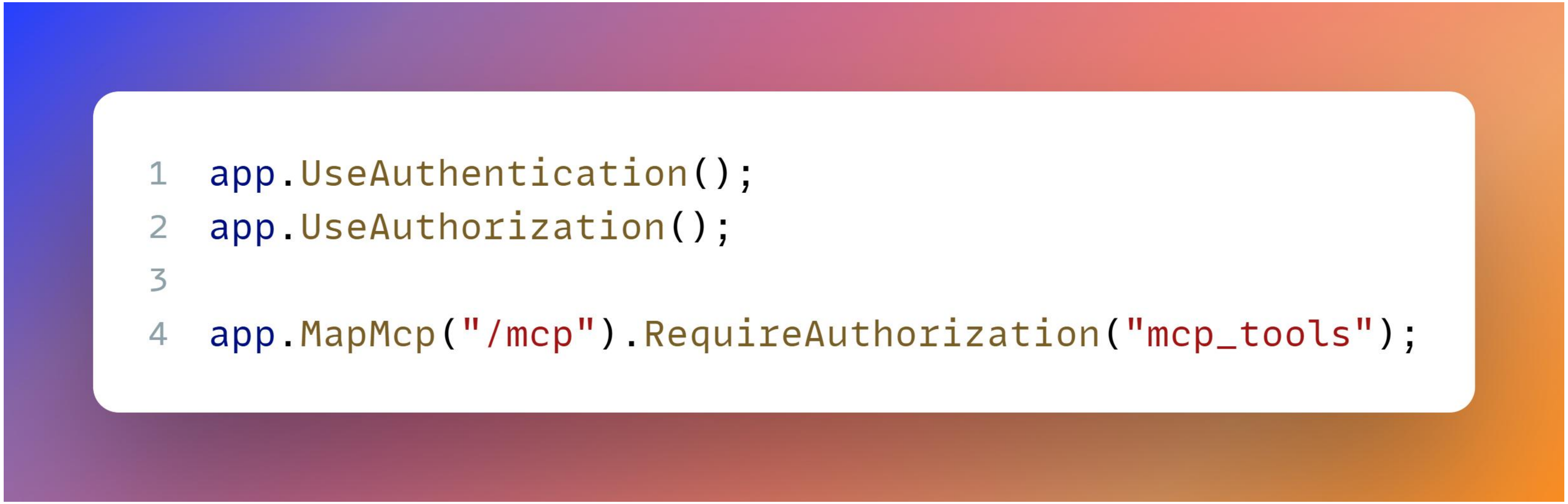
```
builder.Services.AddMicrosoftIdentityWebApiAuthentication(builder.Configuration);
string httpMcpServerUrl = builder.Configuration["HttpMcpServerUrl"]!;

builder.Services.AddAuthentication()
.AddMcp(options =>
{
  options.ResourceMetadata = new()
  {
    Resource = new Uri(httpMcpServerUrl),
    ResourceDocumentation = new Uri("https://docs/health"),
    ScopesSupported = ["mcp:tools"],
  };
});

builder.Services.AddAuthorization();

```

MCP SERVER OAUTH SECURITY



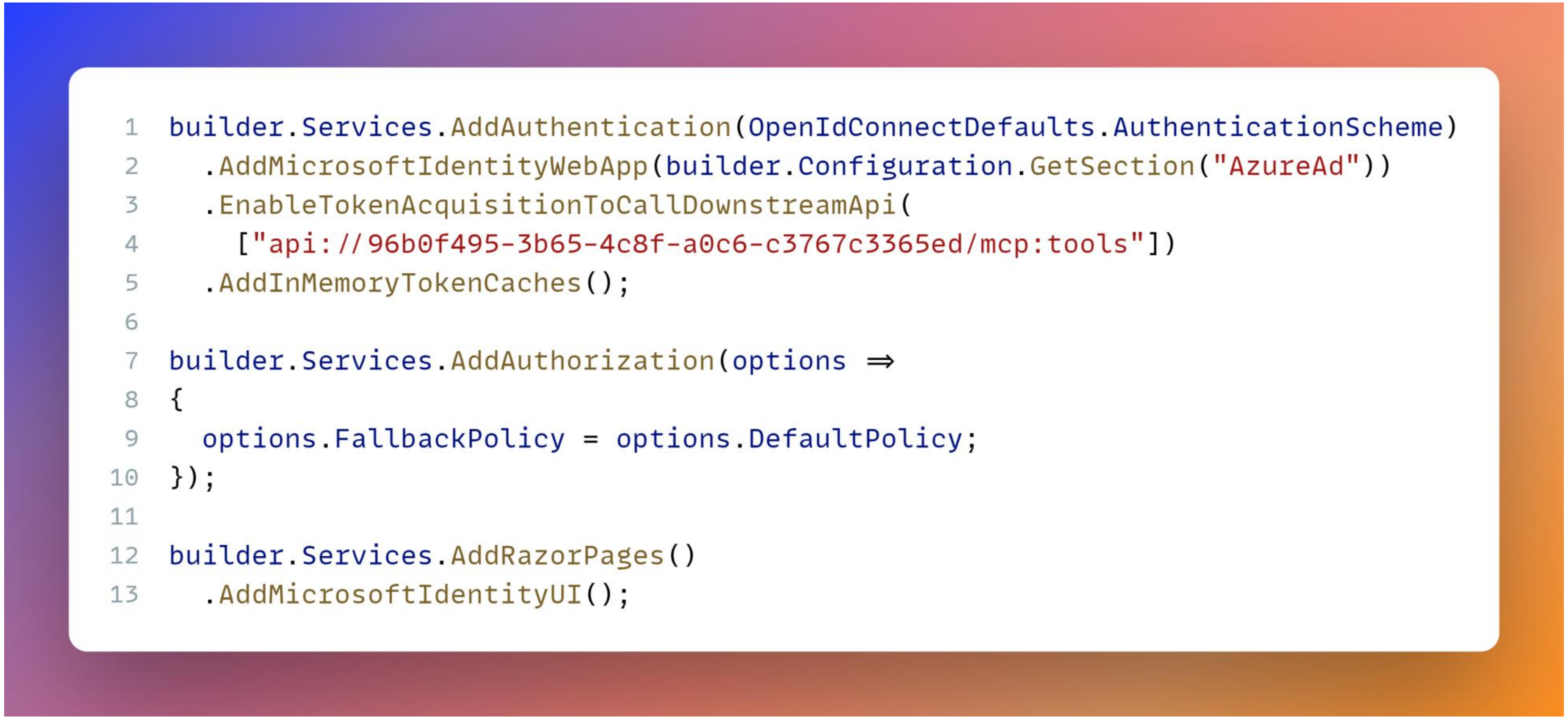
```
// change to scp or scope if not using magic namespaces from MS
// The scope must be validate as we want to force only delegated access tokens
// The scope is requires to only allow access tokens intended for this API
builder.Services.AddAuthorizationBuilder()
  .AddPolicy("mcp_tools", policy =>
    policy.RequireClaim("http://schemas.microsoft.com/identity/claims/scope",
    "mcp:tools"));
```

MCP SERVER OAUTH SECURITY



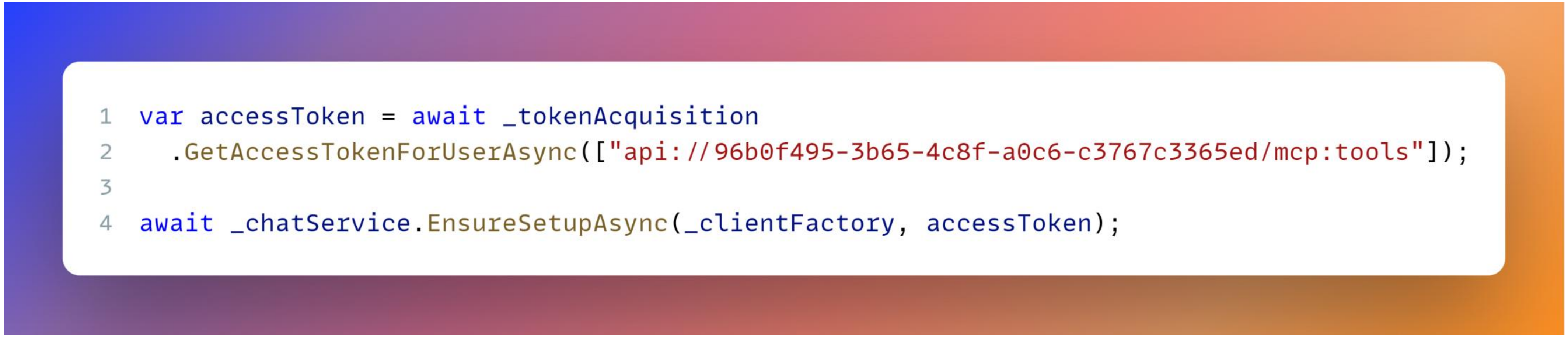
```
app.UseAuthentication();
app.UseAuthorization();

app.MapMcp("/mcp").RequireAuthorization("mcp_tools");
```

CONFIDENTIAL OIDC MCP CLIENT



```
builder.Services.AddAuthentication(OpenIdConnectDefaults.AuthenticationScheme)
  .AddMicrosoftIdentityWebApp(builder.Configuration.GetSection("AzureAd"))
  .EnableTokenAcquisitionToCallDownstreamApi(
    ["api://96b0f495-3b65-4c8f-a0c6-c3767c3365ed/mcp:tools"])
  .AddInMemoryTokenCaches();

builder.Services.AddAuthorization(options =>
{
  options.FallbackPolicy = options.DefaultPolicy;
});

builder.Services.AddRazorPages()
  .AddMicrosoftIdentityUI();
```

CONFIDENTIAL OIDC MCP CLIENT



```
var accessToken = await _tokenAcquisition
  .GetAccessTokenForUserAsync(["api://96b0f495-3b65-4c8f-a0c6-c3767c3365ed/mcp:tools"]);

await _chatService.EnsureSetupAsync(_clientFactory, accessToken);
```

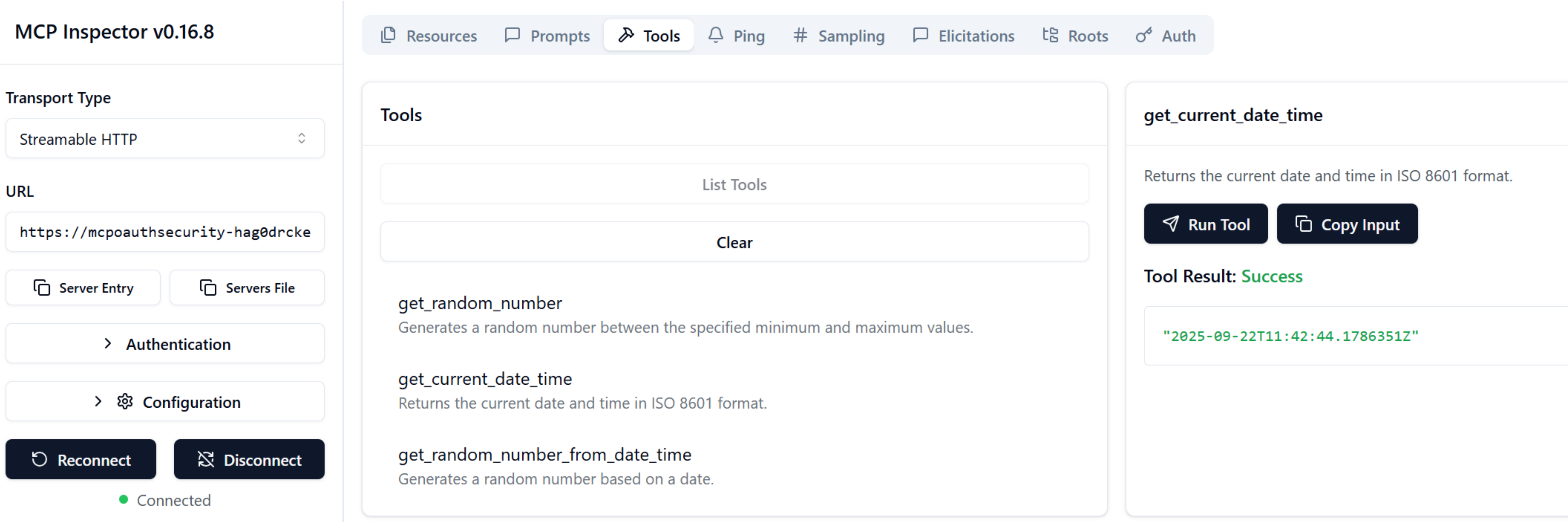
MODEL CONTEXT PROTOCOL

MCP server architectures are very similar to **Micro-services** architecture with all the same problems.

DEMO

MCP INSPECTOR

```
npx @modelcontextprotocol/inspector
```



MCP Inspector v0.16.8

Transport Type: Streamable HTTP

URL: https://mcpoauthsecurity-hag0drcke

● Connected

Tools

get_random_number
Generates a random number between the specified minimum and maximum values.

get_current_date_time
Returns the current date and time in ISO 8601 format.

get_random_number_from_date_time
Generates a random number based on a date.

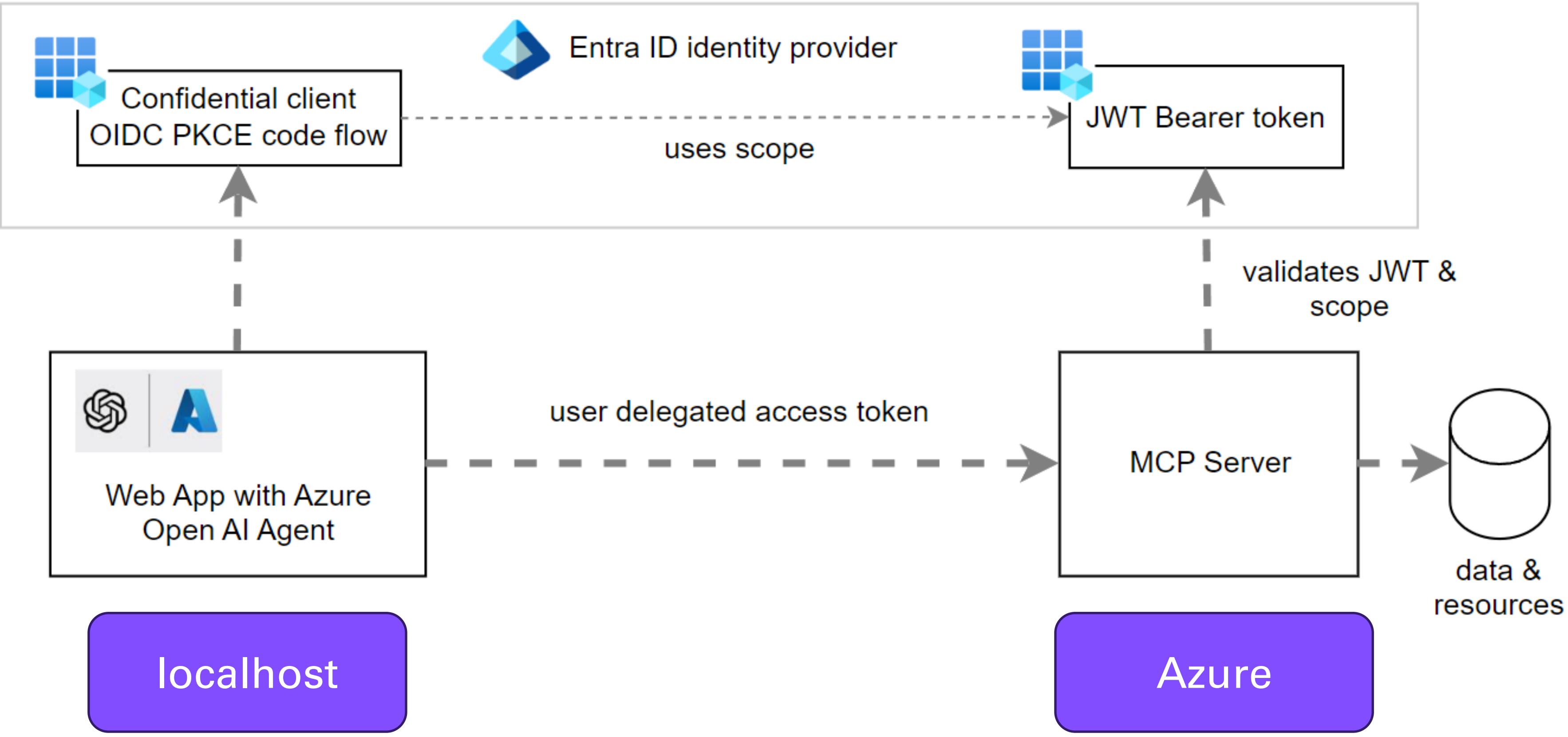
get_current_date_time

Returns the current date and time in ISO 8601 format.

Tool Result: Success

```
"2025-09-22T11:42:44.1786351Z"
```

DEMO


Entra ID identity provider
Confidential client
OIDC PKCE code flow
uses scope
JWT Bearer token
validates JWT & scope
Web App with Azure Open AI Agent
user delegated access token
MCP Server
data & resources
localhost
Azure

RECAP & RECOMMENDATIONS

FURTHER SECURITY IMPROVEMENTS

- Use token binding with OAuth DPoP
- Follow an immutable strategy, only append
- Add audit logs for all changes
- Keep a human in the loop
- Don't allow App-to-App security for downstream APIs
- Avoid A2A until security specs are finished

RECAP

- Star thinking about **Business Use Case**
- Use advanced capabilities - Elicitation and Sampling
- .NET MCP & Semantic Kernel
- https://github.com/damienbod/McpSecurity

isolutions

AI & Data Meetup @isolutions
13.11.2025

Security Meetup @isolutions
05.02.2026

REFERENCES

- MCP C# SDK
- MCP Docs
- MCP Inspector
- MCP for beginners course