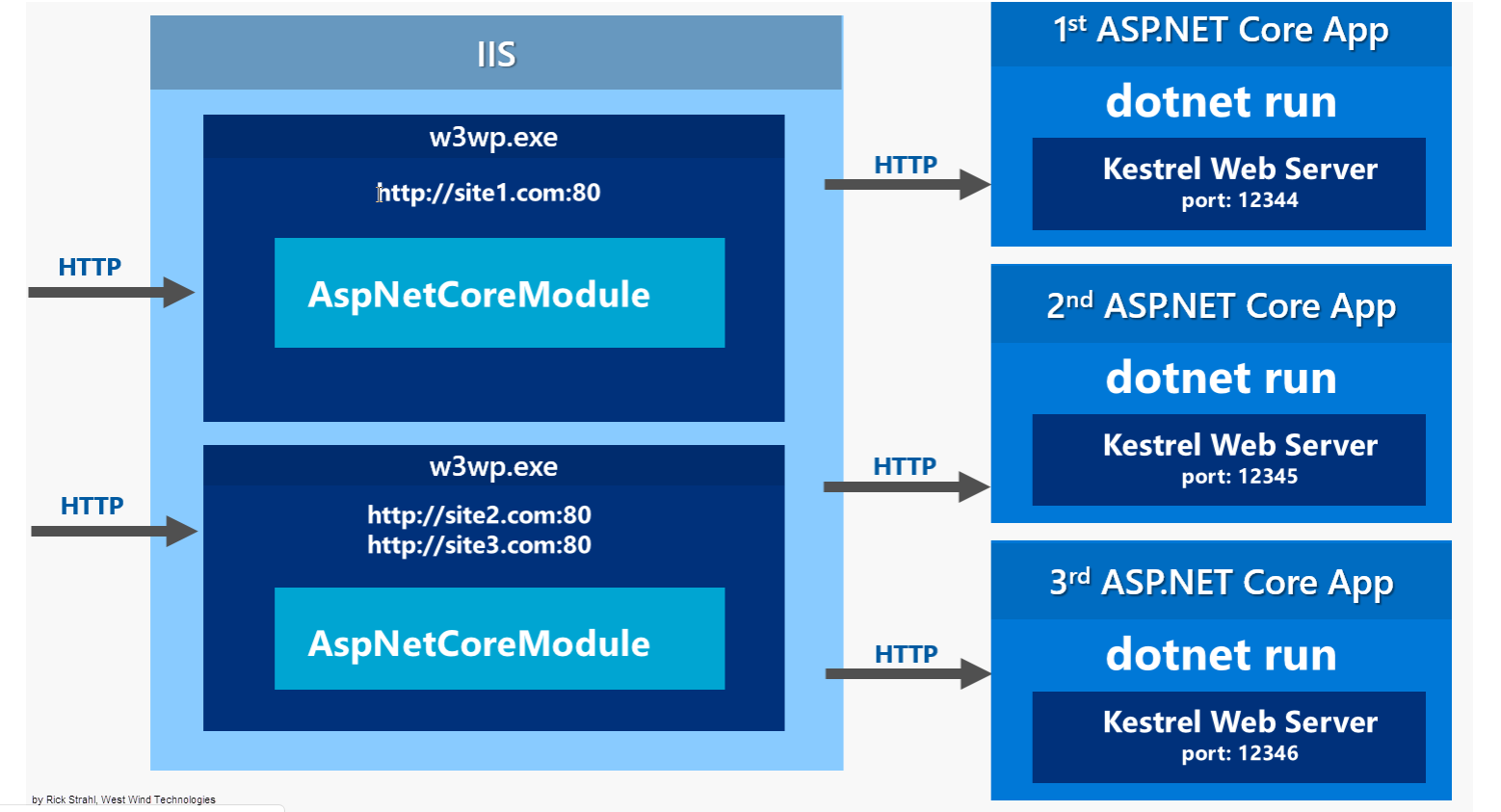
## [ASP.NET Core Running under IIS](https://weblog.west-wind.com/posts/2017/Mar/16/More-on-ASPNET-Core-Running-under-IIS)

IIS acts as a front end proxy to the backend Kestrel Console application that hosts the .NET based Kestrel Web server.

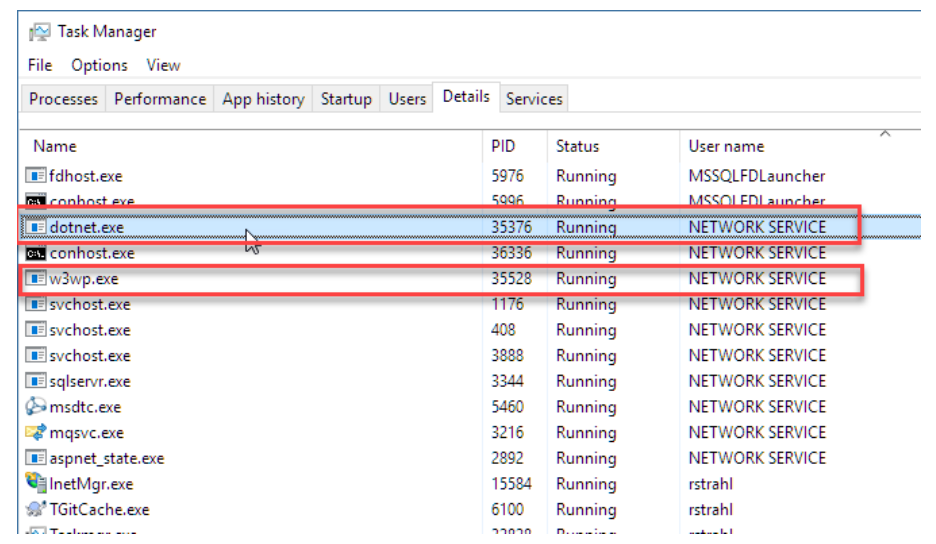


In this scenario, IIS uses a very low level and early pipeline **AspNetCoreModule** that intercepts all requests pointed at it (via a module mapping) and then forwards those requests to Kestrel on a different port. Requests come in on standard HTTP ports (80 and 443 for SSL) and IIS proxies the incoming requests to a different port that Kestrel is listening on.

By default the module configuration forwards all requests to Kestrel, but you have some very limited control over what gets routed to the module.

**w3wp.exe** process (there may be more than one for each application pool so you have find the right one which you can do by looking at the command line arguments in Process Explorer or Task Manager) and the **dotnet.exe** process

You can see that both **w3wp.exe** and **dotnet.exe** - which runs the Kestrel Web server process - are using the same **NETWORK SERVICE** account. Keep in mind that you may have multiple application pools, and multiple instances of .NET Core Application's running at the same time in which case each application pool and kestrel process will launch in their associated security context.



**How to enable Attribute Routing?**

Just add the method — “MapMvcAttributeRoutes()” to enable attribute routing as shown below

*public static void RegisterRoutes(RouteCollection routes)*

*{*

*routes.IgnoreRoute(“{resource}.axd/{\*pathInfo}”);*

*//enabling attribute routing*

*routes.MapMvcAttributeRoutes();*

*//convention-based routing*

*routes.MapRoute*

*(*

*name: “Default”,*

*url: “{controller}/{action}/{id}”,*

*defaults: new { controller = “Customer”, action = “GetCustomerList”, id = UrlParameter.Optional }*

*);*

*}*

What is routes.IgnoreRoute("{resource}.axd/{\*pathInfo}")

.axd files don't exist physically. ASP.NET uses URLs with .axd extensions (ScriptResource.axd and WebResource.axd) internally, and they are handled by an HttpHandler.

Therefore, you should keep this rule, to prevent ASP.NET MVC from trying to handle the request instead of letting the dedicated HttpHandler do it.

**9. Explain JSON Binding?**

JavaScript Object Notation (JSON) binding support started from MVC3 onwards via the new JsonValueProviderFactory, which allows the action methods to accept and model-bind data in JSON format. This is useful in Ajax scenarios like client templates and data binding that need to post data back to the server.

**Active Directory Federation Services (ADFS):**

Active Directory Federation Services (ADFS) is a [Single Sign-On](https://www.okta.com/products/single-sign-on/) (SSO) solution created by Microsoft. As a component of Windows Server operating systems, it provides users with authenticated access to applications that are not capable of using Integrated Windows Authentication (IWA) through Active Directory (AD).

Developed to provide flexibility, ADFS gives organizations the ability to control their employees’ accounts while simplifying the user experience: employees only need to remember a single set of credentials to access multiple applications through SSO.

How does ADFS work?

ADFS manages authentication through a proxy service hosted between AD and the target application. It uses a Federated Trust, linking ADFS and the target application to grant access to users. This enables users to log onto the federated application through SSO without needing to authenticate their identity on application directly.

The authentication process generally follows these four steps:

1. The user navigates to a URL provided by the ADFS service.
2. The ADFS service then authenticates the user via the organization’s AD service.
3. Upon authenticating, the ADFS service then provides the user with an authentication claim.
4. The user’s browser then forwards this claim to the target application, which either grants or denies access based on the Federated Trust service created.

# Grant limited access to Azure Storage resources using shared access signatures (SAS)

A shared access signature (SAS) provides secure delegated access to resources in your storage account without compromising the security of your data. With a SAS, you have granular control over how a client can access your data.

## Types of shared access signatures

Azure Storage supports three types of shared access signatures:

* **User delegation SAS (preview).** A user delegation SAS is secured with Azure Active Directory (Azure AD) credentials and also by the permissions specified for the SAS. A user delegation SAS applies to Blob storage only.

For more information about the user delegation SAS, see [Create a user delegation SAS (REST API)](https://docs.microsoft.com/en-us/rest/api/storageservices/create-user-delegation-sas).

* **Service SAS.** A service SAS is secured with the storage account key. A service SAS delegates access to a resource in only one of the Azure Storage services: Blob storage, Queue storage, Table storage, or Azure Files.

For more information about the service SAS, see [Create a service SAS (REST API)](https://docs.microsoft.com/en-us/rest/api/storageservices/create-service-sas).

* **Account SAS.** An account SAS is secured with the storage account key. An account SAS delegates access to resources in one or more of the storage services. All of the operations available via a service or user delegation SAS are also available via an account SAS. Additionally, with the account SAS, you can delegate access to operations that apply at the level of the service, such as **Get/Set Service Properties** and **Get Service Stats** operations. You can also delegate access to read, write, and delete operations on blob containers, tables, queues, and file shares that are not permitted with a service SAS.

**3. What is an IL?**

Intermediate Language is also known as MSIL (Microsoft Intermediate Language) or CIL (Common Intermediate Language). All .NET source code is compiled to IL. IL is then converted to machine code at the point where the software is installed, or at run-time by a Just-In-Time (JIT) compiler.

**4. What is code access security (CAS)?**

Code access security (CAS) is part of the .NET security model that prevents unauthorized access of resources and operations, and restricts the code to perform particular tasks.

**5. What is Difference between NameSpace and Assembly?**

Assembly is physical grouping of logical units, Namespace, logically groups classes.

Namespace can span multiple assembly.

**6. Mention the execution process for managed code.**

A)Choosing a language compiler

B) Compiling the code to MSIL

C) Compiling MSIL to native code

D) Executing the code.

**7. What is Microsoft Intermediate Language (MSIL)?**

The .NET Framework is shipped with compilers of all .NET programming languages to develop programs. There are separate compilers for the Visual Basic, C#, and Visual C++ programming languages in .NET Framework. Each .NET compiler produces an intermediate code after compiling the source code. The intermediate code is common for all languages and is understandable only to .NET environment. This intermediate code is known as MSIL.

**8. What is managed extensibility framework?**

Managed extensibility framework (MEF) is a new library that is introduced as a part of .NET 4.0 and Silverlight 4. It helps in extending your application by providing greater reuse of applications and components. MEF provides a way for host application to consume external extensions without any configuration requirement.

**9. Which method do you use to enforce garbage collection in .NET?**

The System.GC.Collect() method.

**10. What is the difference between int and int32.**

There is no difference between int and int32. System.Int32 is a .NET Class and int is an alias name for System.Int32.

**11. What are tuples?**

Tuple is a fixed-size collection that can have elements of either same or different data types. Similar to arrays, a user must have to specify the size of a tuple at the time of declaration. Tuples are allowed to hold up from 1 to 8 elements and if there are more than 8 elements, then the 8th element can be defined as another tuple. Tuples can be specified as parameter or return type of a method.

**12. What is the full form of ADO?**

The full form of ADO is ActiveX Data Object.

**13. What are the two fundamental objects in**[**ADO.NET?**](http://ado.net/)

DataReader and DataSet are the two fundamental objects in [ADO.NET](http://ado.net/)

**14. What is the meaning of object pooling?**

Object pooling is a concept of storing a pool (group) of objects in memory that can be reused later as needed. Whenever, a new object is required to create, an object from the pool can be allocated for this request; thereby, minimizing the object creation. A pool can also refer to a group of connections and threads. Pooling, therefore, helps in minimizing the use of system resources, improves system scalability, and performance.

**15. Mention the namespace that is used to include .NET Data Provider for SQL server in .NET code.**

The System.Data.SqlClient namespace.

**16. Which architecture does Datasets follow?**

Datasets follow the disconnected data architecture.

**17. What is the role of the DataSet object in**[**ADO.NET?**](http://ado.net/)

One of the major component of [ADO.NET](http://ado.net/) is the DataSet object, which always remains disconnected from the database and reduces the load on the database.

**18. Which property is used to check whether a DataReader is closed or opened?**

The IsClosed property is used to check whether a DataReader is closed or opened. This property returns a true value if a Data Reader is closed, otherwise a false value is returned.

**19. Name the method that needs to be invoked on the DataAdapter control to fill the generated DataSet with data?**

The Fill() method is used to fill the dataset with data.

**20. What are the pre-requisites for connection pooling?**

There must be multiple processes to share the same connection describing the same parameters and security settings. The connection string must be identical.

**21. Which adapter should you use, if you want to get the data from an Access database?**

OleDbDataAdapter is used to get the data from an Access database.

**22. What are different types of authentication techniques that are used in connection strings to connect .NET applications with Microsoft SQL Server?**

The Windows Authentication option

The SQL Server Authentication option

**23. What are the parameters that control most of connection pooling behaviors?**

Connect Timeout

Max Pool Size

Min Pool Size

Pooling

**24. What is AutoPostBack?**

If you want a control to postback automatically when an event is raised, you need to set the AutoPostBack property of the control to True.