عنوان مقاله: راهنمای پیکربندی Authentication و Authorization در SQL Server به بهترین روش

نویسنده مقاله: تیم فنی نیک‌آموز

تاریخ انتشار: ۱۹ فروردین ۱۴۰۳

منبع: https://nikamooz.com/configuration-of-authentication-and-authorization/

پیکربندی Authentication و Authorization دو رکن اساسی برای تأمین امنیت در SQL Server هستند. این دو مکانیزم در کنارهم، به مدیریت دسترسی‌ها و حفاظت از داده‌ها کمک می‌کنند. در این مقاله، به توضیح این دو مفهوم و چگونگی بهبود امنیت دیتابیس با استفاده از آن‌ها می‌پردازیم. علاوه‌براین، می‌توانید به آموزش رمزگذاری اطلاعات در SQL Server رجوع کنید تا با مطالعه آن، درک عمیق‌تری از اهمیت محافظت از داده‌ها به‌دست آورید.

Authentication چیست؟

احراز هویت (Authentication) فرآیندی است که به منظور بررسی هویت کاربر یا موجودیتی که قصد دسترسی به SQL Server دارد، به کار می‌بریم. این فرآیند مشابه این است که از کاربر پرسیده شود «شما چه کسی هستید؟» و در پاسخ، لازم است کاربر اطلاعات شخصی، یعنی نام کاربری و رمز عبور را ارائه دهد.

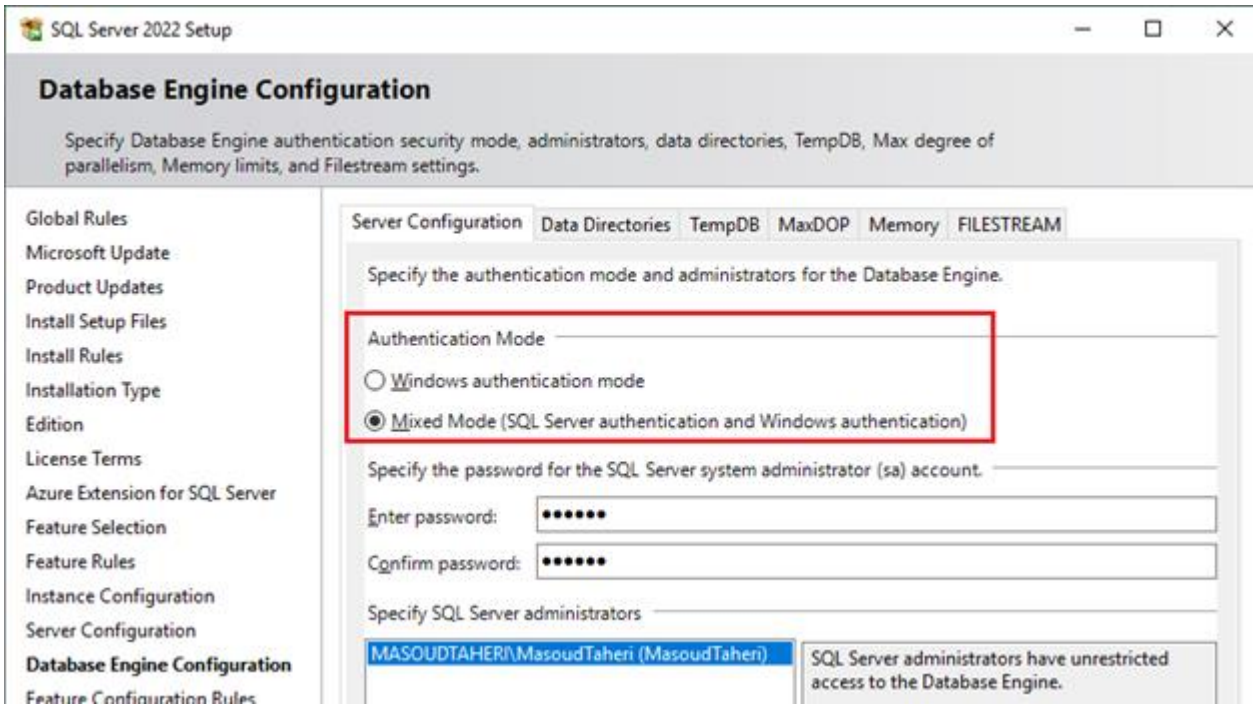
انواع روش های احراز هویت در SQL Server

Authentication در SQL Server به روش‌های زیر قابل انجام است:

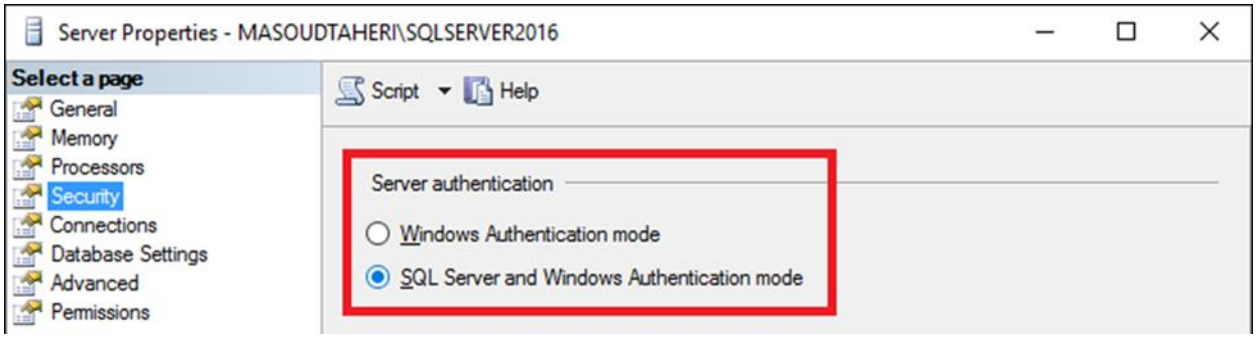
- **احراز هویت ویندوز (Windows Authentication):** این روش، از حساب‌های کاربری و گروه‌های موجود در ویندوز استفاده می‌کند. به‌دلیل سهولت استفاده و امنیت بالا، احراز هویت ویندوز، به‌عنوان روش توصیه‌شده شناخته می‌شود؛ زیرا در آن از پروتکل امنیتی Kerberos استفاده شده و نیازی به مدیریت جداگانه نام‌های کاربری و رمزهای عبور SQL Server وجود ندارد.
- **احراز هویت Mixed:** احراز هویت ترکیبی در SQL Server ، نوعی پیکربندی است که به‌طور همزمان، امکان استفاده از احراز هویت ویندوز و احراز هویت SQL Server را فراهم می‌کند. در SQL Server این حالت با نام SQL Server and Windows Authentication شناخته می‌شود.

نکته: برای تنظیم نوع Authentication می‌توانید این کار را به دو صورت انجام دهید:

۱. تنظیم نوع Authentication هنگام نصب SQL Server



۲. تنظیم نوع Authentication پس از نصب SQL Server



Authorization چیست؟

پس از احراز هویت کاربر در SQL Server ، اعطای مجوز (Authorization) تعیین می‌کند که کاربر در داخل SQL Server مجاز به انجام چه کارهایی است. این بخش به سؤال «چه اجازه‌ای برای انجام چه کاری دارید؟» پاسخ می‌دهد. این مجوزدهی ازطریق مجوزها (Permissions) و نقش‌ها (Roles) مدیریت می‌شود و محدوده دسترسی و اقداماتی را تعریف می‌کند که یک کاربر می‌تواند انجام دهد. به‌عنوان مثال، امکان خواندن، نوشتن یا تغییر داده، از مواردی هستند که از این طریق انجام می‌شوند.

شباهت Authentication و Authorization

مهم‌ترین نقاط شباهت Authentication و Authorization در SQL Server از منظرهای مختلف، به شرح زیر است:

- **نقش‌های امنیتی:** احراز هویت و اعطای مجوز ، هردو برای ایمن‌سازی سیستم‌ها و منابع، حیاتی محسوب می‌شوند. این دو مکانیزم اطمینان می‌دهند که تنها کاربران یا سرویس‌های تأییدشده و مجاز می‌توانند به اطلاعات یا بخش‌های حساس سیستم دسترسی پیدا کنند.
- **وابستگی متقابل:** احراز هویت و اعطای مجوز اغلب به‌عنوان بخشی از یک سیستم کنترل دسترسی با یکدیگر کار می‌کنند. به‌طور معمول، فرآیندهای اعطای مجوز برای تعیین محدوده دسترسی یک کاربر یا سرویس، به احراز هویت موفق وابسته است.
- **قابلیت پیکربندی:** هردوی این موارد را می‌توان براساس الزامات امنیتی خاص یک سیستم، سفارشی‌سازی و پیکربندی کرد. برای مثال، می‌توان سیستمی را راه‌اندازی کرد که برای امنیت بیشتر به **احراز هویت چند عاملی** (MFA) نیاز داشته باشد یا سطوح مختلف دسترسی را براساس نقش‌های کاربر اعطا کند.
- **کاربرد در سیستم‌های متنوع:** مفاهیم Authentication و Authorization در انواع مختلف سیستم‌ها، ازجمله وب‌اپلیکیشن‌ها، **انواع پایگاه های داده** و سیستم‌های شبکه، قابل اجرا هستند. در عمل، پیکربندی Authentication و Authorization برای هر سیستمی که نیاز به کنترل دسترسی امن دارد، اساسی و مهم به‌شمار می‌آید.

در حالی که احراز هویت و اعطای مجوز در SQL Server از نظر هدف استفاده، فرآیند و نوع اطلاعاتی که استفاده می‌کنند، متمایز هستند، اما شباهت‌های فوق را نیز در نقش‌هایشان دارند. بنابراین، درک هردو مفهوم به‌منظور طراحی، پیاده‌سازی و مدیریت سیستم‌های امن، ضروری است.

تفاوت Authentication و Authorization

در ادامه، به بررسی تفاوت Authentication و Authorization از نقطه نظرهای گوناگون می‌پردازیم.

تفاوت Authentication و Authorization: هدف

به‌واسطه احراز هویت در SQL Server ، هویت کاربر یا موجودیتی تعیین می‌شود که قصد دسترسی به سیستم را دارد. این عمل مانند بررسی کارت شناسایی شخص در ورودی یک مکان است؛ در صورتی که اعطای مجوز، تعیین می‌کند کاربر احراز هویت‌شده چه اقداماتی را می‌تواند در داخل سیستم انجام دهد.

تفاوت Authentication و Authorization: فرآیند

با پیکربندی احراز هویت ، Credential ها، یعنی نام کاربری و رمز عبور، برای تأیید هویت بررسی می‌شوند. ازسوی دیگر، با استفاده از فرآیند اعطای مجوز ، Permission های مرتبط با کاربر احراز هویت‌شده، با قوانین سیستم مقایسه می‌شوند تا دسترسی به سیستم، اعطا یا محدود شود.

تفاوت Authentication و Authorization: اطلاعات مورد استفاده

در احراز هویت از Credential ها و احتمالاً فاکتورهای دیگری مانند رمزهای عبور یکبار مصرف یا سؤالات امنیتی استفاده می‌شود. ازسوی دیگر، در Authorization ، پیکربندی‌ها، نقش‌ها و سیاست‌هایی استفاده می‌شوند که اقدامات مجاز برای کاربران یا نقش‌های مختلف در داخل سیستم را تعریف می‌کنند.

روش های پیکربندی Authentication در SQL Server

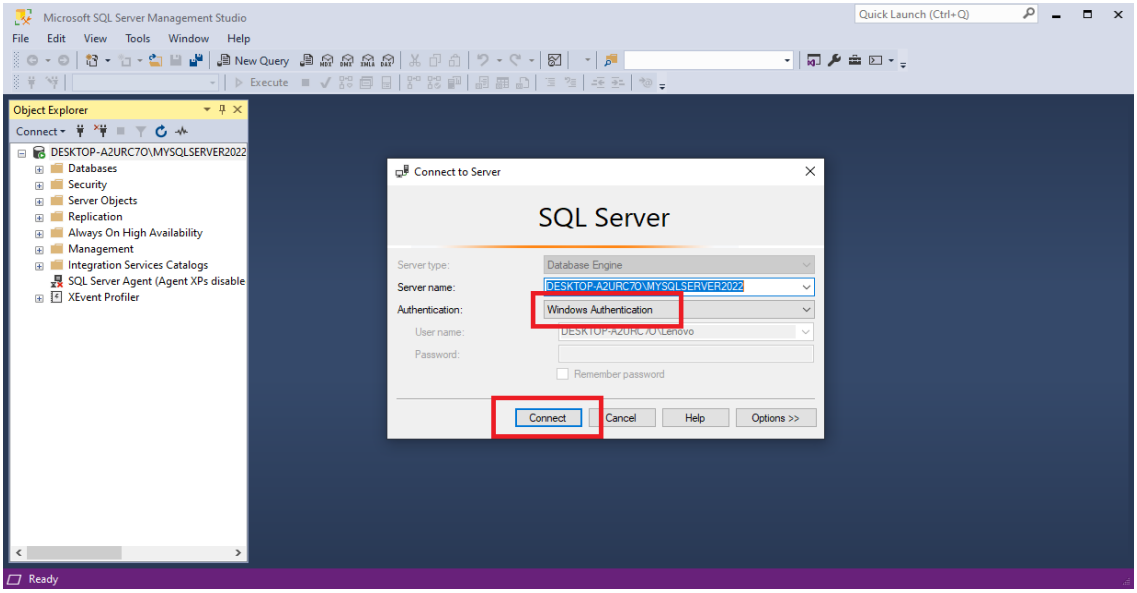
همان‌طور که پیش‌تر به آن اشاره شد، SQL Server از دو حالت اصلی احراز هویت به‌همراه یک گزینه ترکیبی که از مزایای هردو رویکرد بهره می‌برد، پشتیبانی می‌کند. درک و انتخاب روش مناسب احراز هویت ، اولین گام در ایمن سازی محیط SQL Server شما است.

مراحل پیکربندی احراز هویت : روش Windows Authentication

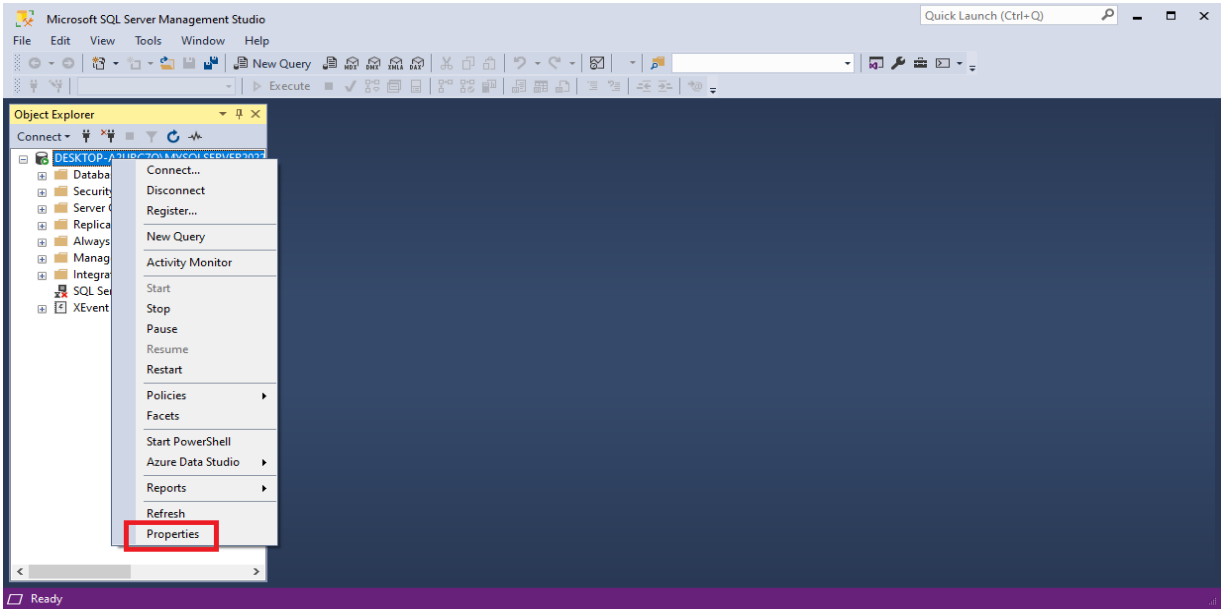
پیکربندی احراز هویت در روش Windows Authentication ، شامل مجموعه‌ای از مراحل است که اطمینان می‌دهد اینستنسِ SQL Server از حساب‌های کاربری و گروه‌های ویندوز به‌منظور احراز هویت استفاده می‌کند. این حالت، به‌دلیل مزایای امنیتی خاصی مانند یکپارچه‌سازی با مکانیزم‌های امنیتی ویندوز، ترجیح داده می‌شود.

در ادامه، مراحل کلیدی برای پیکربندی حالت احراز هویت ویندوز بررسی می‌شوند:

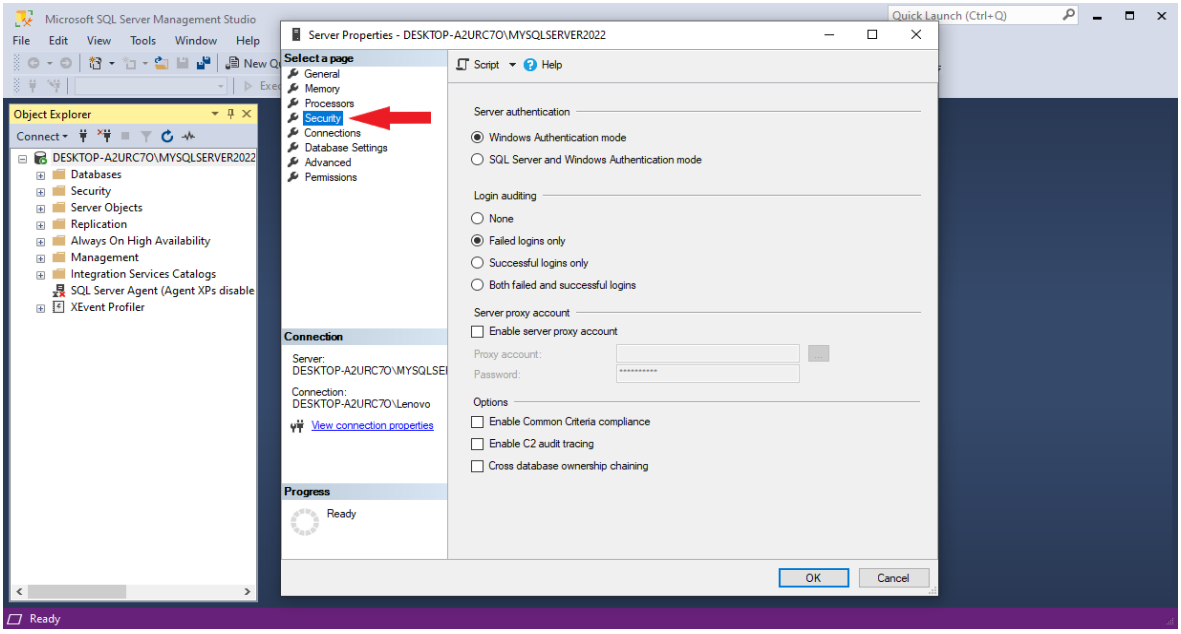
۱. **اتصال به SQL Server:** نرم‌افزار SSMS را باز کنید و به Instance یا همان نمونه SQL Server خود Connect شوید. اگر هنوز آن را روی سیستم خود Install نکرده‌اید، می‌توانید از آموزش نصب گام به گام SSMS به‌عنوان راهنما استفاده کنید.



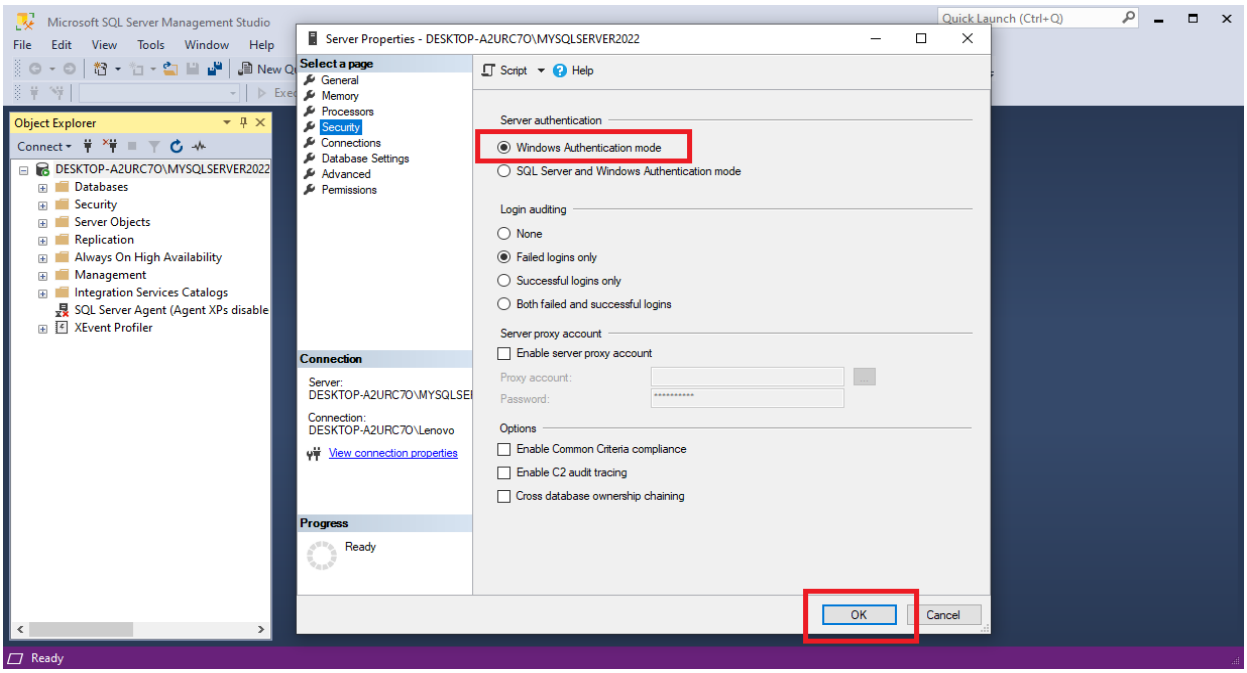
۲. **دسترسی به تنظیمات سرور:** در بخش Object Explorer، روی نام سرور راست‌کلیک کرده و گزینه Properties را انتخاب کنید.



۳. **تنظیمات امنیتی:** گزینه Security را کلیک کرده و از بخش Server Authentication، گزینه Windows Authentication mode را انتخاب کنید.



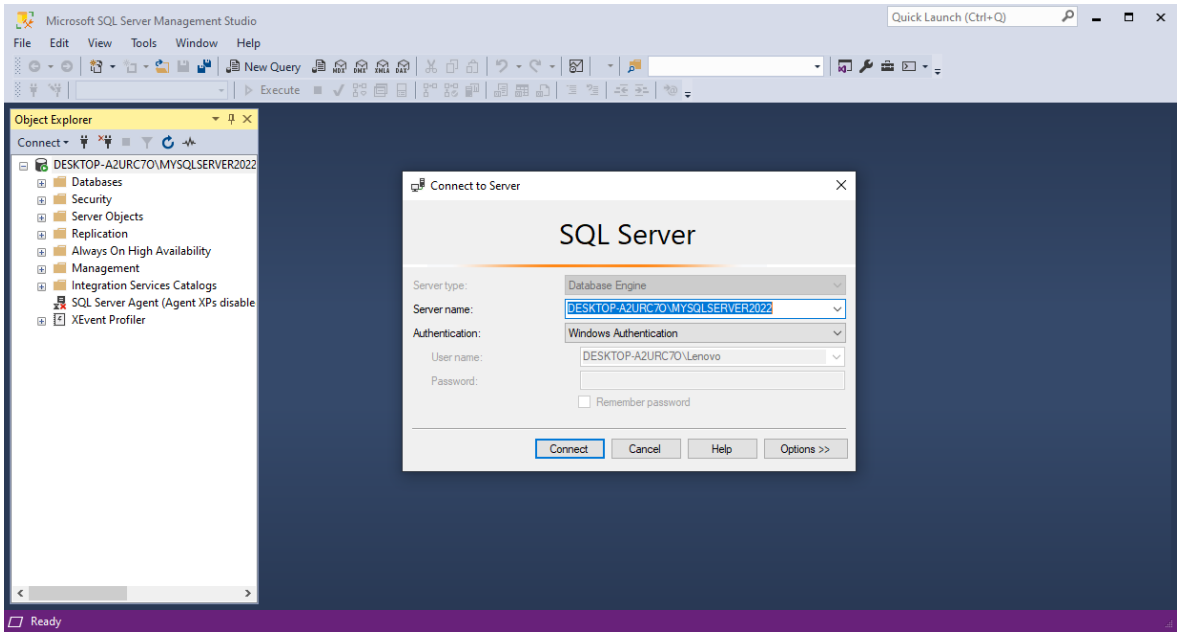
٤. **ذخیره‌سازی تغییرات:** برای نهایی‌سازی اقدامات، روی دکمه OK کلیک کنید. اگر مجبور به تغییر حالت احراز هویت شده‌اید، برای اعمال‌شدن تغییرات، نیاز به راه‌اندازی مجدد سرویس SQL Server دارید.



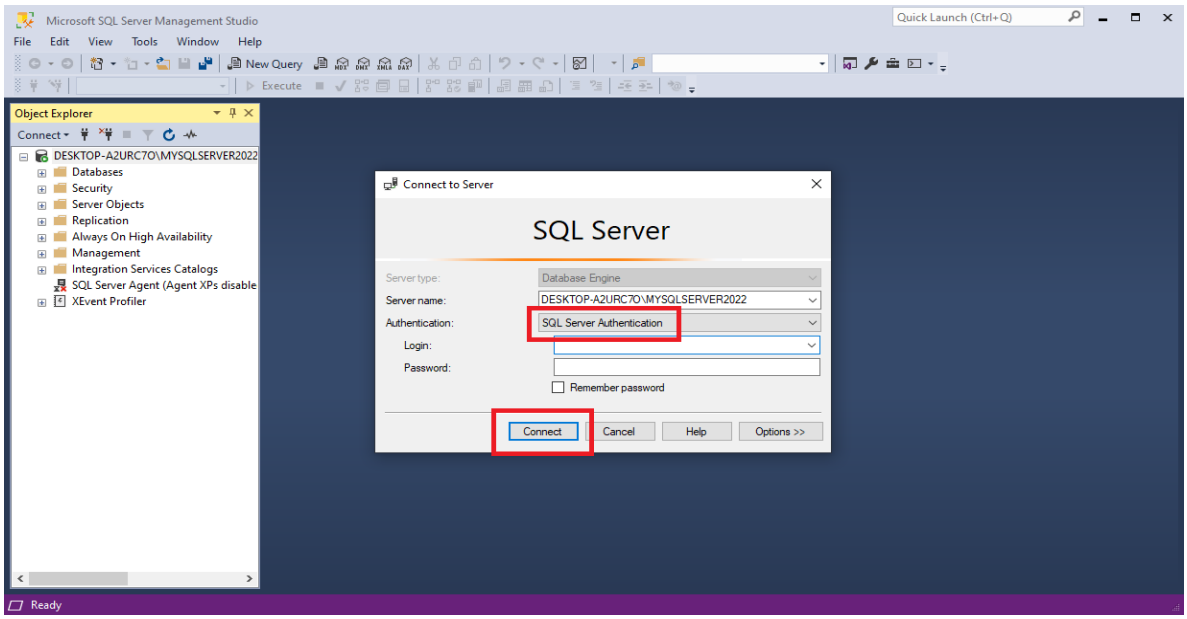
مراحل پیکربندی احراز هویت : روش Mixed Authentication

برای پیکربندی احراز هویت در SQL Server ازطریق روش مبتنی‌بر SQL Server Authentication ، باید حالت ترکیبی (Mixed Mode) را فعال کنید. این حالت، به هردو روش احراز هویت ویندوز و احراز هویت SQL Server اجازه ورود می‌دهد. در ادامه، یک راهنمای گام‌به‌گام برای پیکربندی این روش احراز هویت شرح داده خواهد شد.

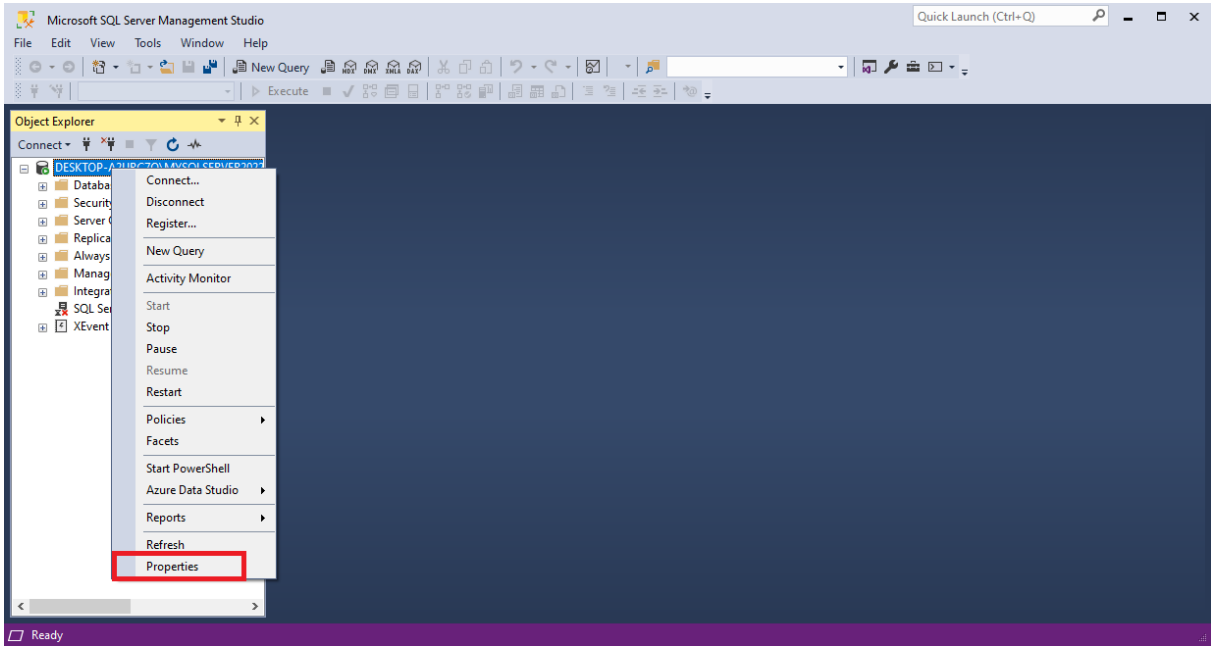
۱. **ورود به SSMS :** SQL Server Management Studio را اجرا کنید و به Instance خود متصل شوید.



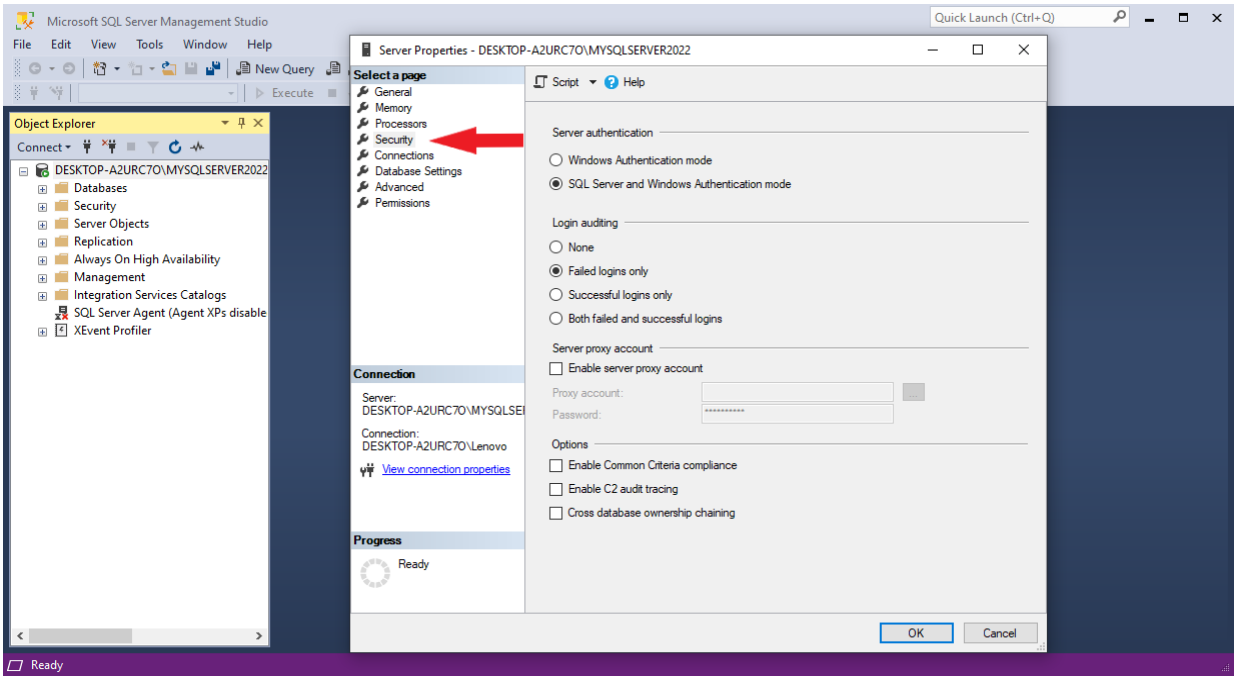
۲. **اتصال به سرور:** برای برقراری اتصال اولیه به سرور، از مشخصات احراز هویت ویندوز خود استفاده کنید.



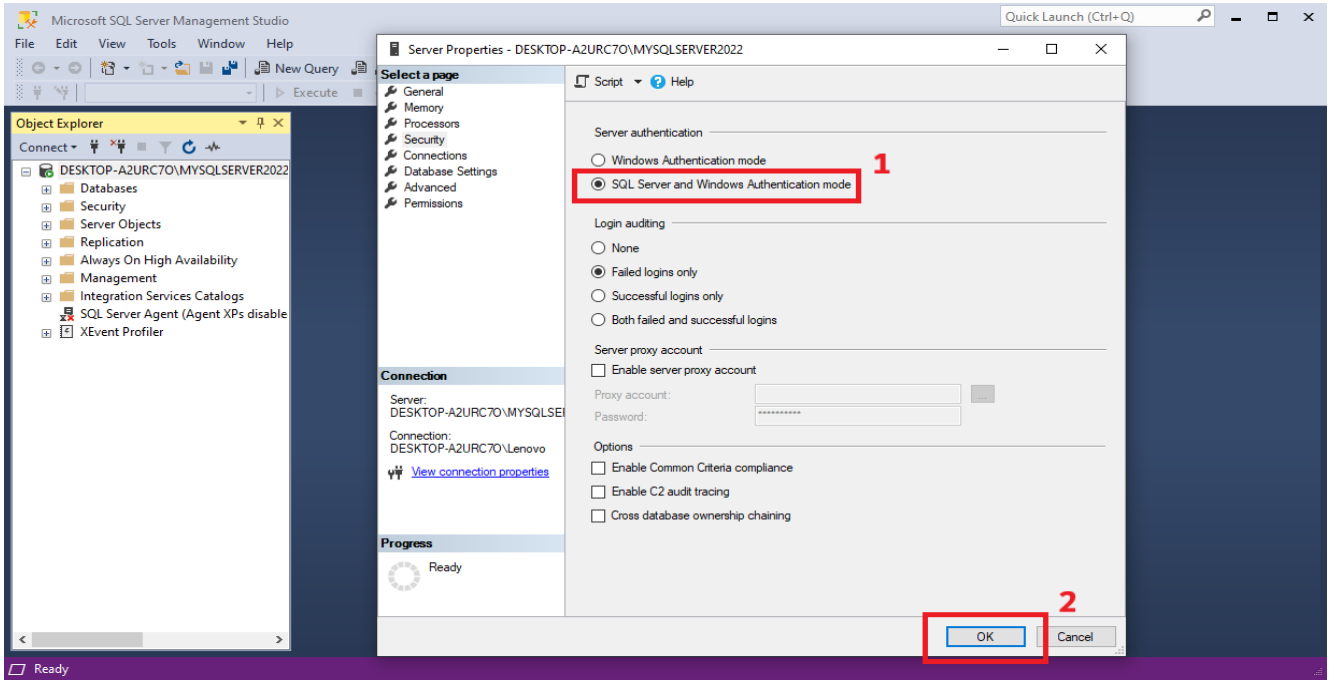
۳. **ورود به تنظیمات سرور (Server Properties):** روی نام سرور در پنجره Object Explorer راست‌کلیک کرده و Properties را از منوی زمینه انتخاب کنید.



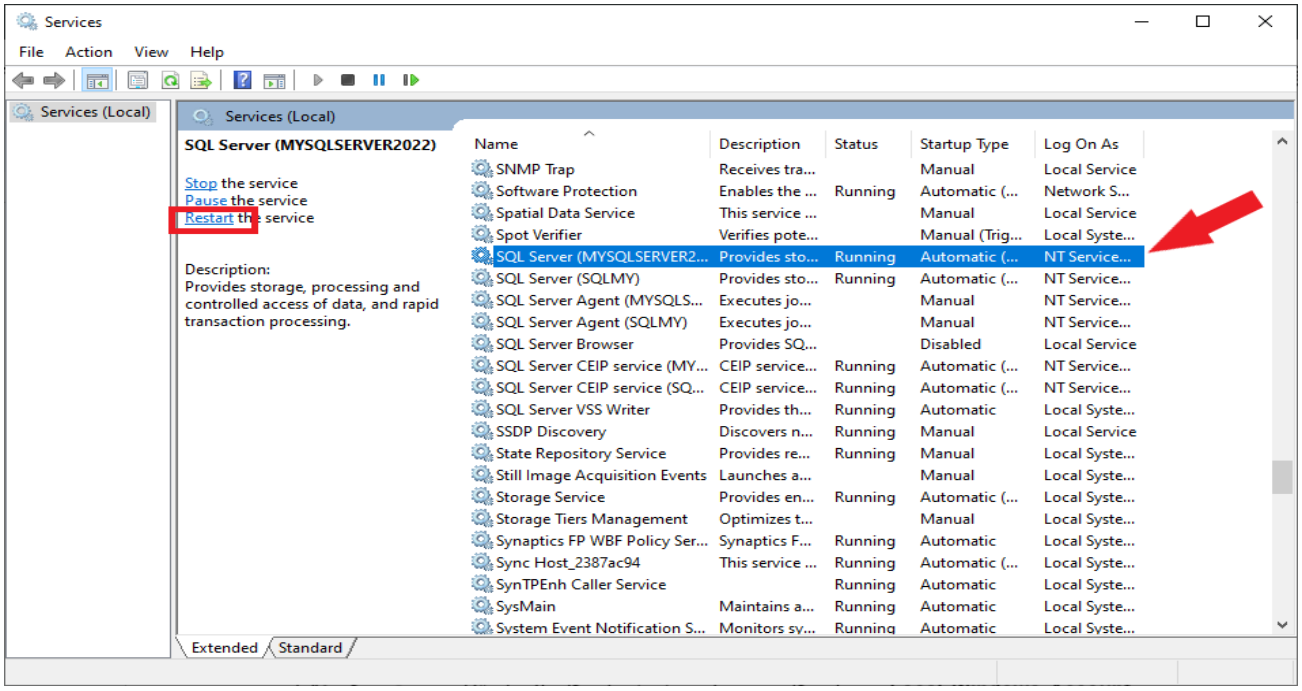
٤. **انتخاب گزینه Security:** در پنجره تنظیمات سرور، روی تب Security از لیست سمت چپ کلیک کنید.



۵. **تغییر حالت احراز هویت سرور:** حال در تب Security، بخشی با عنوان Server authentication را مشاهده خواهید کرد. گزینه SQL Server and Windows Authentication mode را انتخاب کنید. این کار، حالت ترکیبی (Mixed Mode) را برایتان فعال می‌کند. برای ذخیره‌سازی تغییرات، روی دکمه OK کلیک کنید.



٦. **راه‌اندازی مجدد سرویس SQL Server:** برای اعمال تغییرات، باید سرویس SQL Server را مجدداً راه‌اندازی کنید. این کار را می‌توانید ازطریق SQL Server Configuration Manager یا با استفاده از Services در ویندوز انجام دهید. در SQL Server Configuration Manager ، نمونه SQL Server خود را پیدا کنید، روی آن راست‌کلیک کرده و Restart را انتخاب کنید. شما می‌توانید در Services ویندوز، سرویسی را پیدا کنید که نامی شبیه به نمونه SQL Server شما دارد؛ سپس روی آن راست‌کلیک کرده و Restart را انتخاب کنید.



مبنای احراز هویت از طریق Windows Authentication

در SQL Server از چندین روش مختلف احراز هویت از طریق Windows Authentication پشتیبانی می‌شود. این امکان ازطریق استفاده از شناسه‌ها و گروه‌های مختلف ویندوز فراهم می‌شود. چهار روش اصلی آن عبارتند از:

- **Local Windows Account:** حساب کاربری محلی ویندوز برای کاربرانی استفاده می‌شود که فقط برروی کامپیوتر محلی نیاز به دسترسی دارند. این اکانت‌ها مستقیماً روی سیستم هاست SQL Server ایجاد می‌شوند و می‌توانند برای احراز هویت و دسترسی به SQL Server به کار روند.
- **Local Windows Group:** گروه‌های محلی ویندوز برای مدیریت دسترسی‌های گروهی کاربران در سطح محلی استفاده می‌شوند. می‌توان یک گروه محلی ایجاد کرد و کاربران مختلفی را به آن اضافه کرد، سپس گروه مذکور، دسترسی به SQL Server اعطا کرد. این امر، مدیریت دسترسی‌ها را آسان‌تر می‌کند.
- **Domain Account:** حساب‌های دامنه برای کاربرانی استفاده می‌شوند که در محیط شبکه Windows Active Directory قرار دارند. این حساب‌ها توسط مدیر دامنه ایجاد و مدیریت می‌شوند و امکان دسترسی به منابع مختلف در سراسر دامنه، ازجمله SQL Server را فراهم می‌کنند.
- **Domain Group:** گروه‌های دامنه در Active Directory ایجاد می‌شوند و به مدیران این امکان را می‌دهند تا دسترسی‌ها را برای یک گروه خاصی از کاربران در سراسر دامنه مدیریت کنند. این گروه‌ها می‌توانند شامل کاربرانی از سراسر Domain باشند و به آن‌ها دسترسی جمعی به SQL Server داده می‌شود.

پیکربندی Authorization در SQL Server

پیکربندی Authorization در SQL Server برای حفظ امنیت و یکپارچگی پایگاه داده بسیار مهم است. SQL Server به‌منظور کنترل دسترسی، از ترکیبی از مجوزهای سطح سرور و سطح پایگاه داده استفاده می‌کند.

برای کنترل دقیق دسترسی به منابع در SQL Server ، از سه مفهوم اصلی زیر استفاده می‌شود:

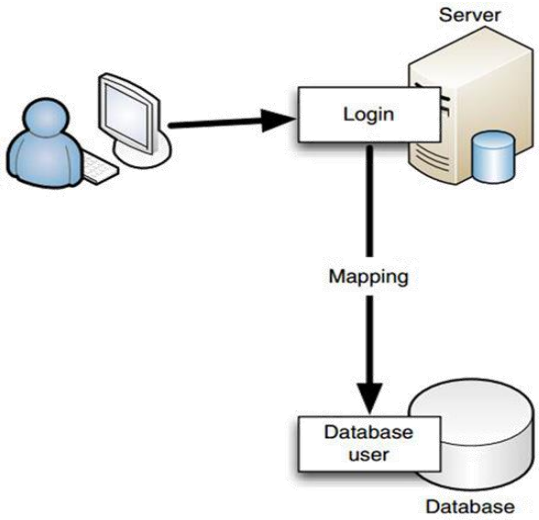
۱. **Principal ها:** موجودیت‌هایی هستند که امکان درخواست دسترسی به منابع SQL Server را دارند و می‌توانند شامل لاگین‌های SQL Server، حساب‌های کاربری ویندوز یا گروه‌های کاربری باشند. به‌عبارت دیگر، هرکسی یا چیزی که می‌خواهد به منابع SQL Server دسترسی داشته باشد، یک Principal درنظر گرفته می‌شود.
۲. **منابع قابل دسترسی (Securables):** منابعی هستند که موتور SQL Server کنترل دسترسی به آن‌ها را مدیریت می‌کند. منابع قابل دسترسی می‌توانند در سطح سرور (مانند دیتابیس‌ها، لاگین‌ها و Endpoint ها) یا در سطح پایگاه داده (مانند جدول و View و پروسیجر) باشند. به بیان ساده، هر منبعی که در SQL Server وجود دارد و نیاز به کنترل دسترسی دارد، یک «منبع قابل دسترسی» محسوب می‌شود.
۳. **مجوزها (Permissions):** اقداماتی هستند که Principal ها می‌توانند روی منابع قابل دسترسی انجام دهند. مجوزها برای سطوح مختلف دسترسی تعریف می‌شوند. برای اشیا پایگاه داده، مجوزهایی مانند SELECT (انتخاب)، INSERT (درج)، UPDATE (به‌روزرسانی) و DELETE (حذف) وجود دارد. در سطح سرور، مجوزهایی مانند CREATE DATABASE (ایجاد پایگاه داده)، CREATE LOGIN (ایجاد ورود) تعریف می‌شوند. به‌طورکلی، مجوزها مشخص می‌کنند که هر Principal چه کارهایی را می‌تواند روی یک منبع قابل دسترسی انجام دهد.

شایان ذکر است که لاگین‌ها، User ها، Role ها و Permission ها، مکانیزم‌های Authorization محسوب می‌شوند.

استفاده از Login و Data User برای پیاده سازی Authorization

ما در SQL Server با استفاده از Login ها می‌توانیم برای ورود به SQL Server استفاده کنیم. هر Login با توجه به دسترسی های ارائه‌شده می‌تواند Map شود به تعدادی Data User در بانک های اطلاعاتی مختلف.

توجه داشته باشید که کنترل دسترسی به اشیاء سمت سرور (مانند ایجاد لاگین، ایجاد روال‌های Audit و...) با استفاده از Login ها و همچنین کنترل دسترسی به اشیاء سمت بانک اطلاعاتی (مانند دسترسی به یک جدول، دسترسی به ویو و...) با استفاده از Data User انجام می‌شود.



مراحل پیاده سازی Authorization

پیکربندی Authorization در SQL Server ، تنظیم مجوزها و کنترل‌های دسترسی را شامل می‌شود. بدین طریق، اطمینان حاصل می‌شود که کاربران فقط می‌توانند اقداماتی را انجام دهند که برای نقش آن‌ها ضروری است. در ادامه، یک راهنمای گام‌به‌گام برای پیکربندی اعطای مجوز در SQL Server آورده شده است:

۱. **تعیین حالت احراز هویت:** از میان دو روش Windows Authentication Mode و Mixed Mode ، یک حالت احراز هویت را انتخاب کنید. درصورت انتخاب حالت ترکیبی، حتماً یک رمز عبور قوی برای حساب کاربری SA تنظیم کنید.

۲. **ایجاد Login:** مشخص کنید که آیا Login موردنظر، یک حساب کاربری ویندوز است یا یک حساب کاربری SQL Server. برای حساب‌های کاربری یا گروه‌های ویندوز، مطمئن شوید که آن‌ها در Active Directory یا روی دستگاه محلی ایجاد شده‌اند.

مشابه زیر، از SQL Server Management Studio یا دستورات T-SQL برای ایجاد Login استفاده کنید:

- **برای Login ویندوز:**

```
CREATE LOGIN [DOMAIN\User] FROM WINDOWS
GO
```

- **برای لاگین SQL Server:**

```
CREATE LOGIN LoginName WITH PASSWORD = 'strong_password'
GO
```

۳. **ایجاد کاربران پایگاه داده و نگاشت آن‌ها به Login ها:** پایگاه داده‌ای را که کاربر به آن نیاز دارد، انتخاب یا ایجاد کنید. سپس یک کاربر در پایگاه داده ایجاد کرده و آن را ازطریق دستور زیر، به یک Server لاگین نگاشت کنید.

```
USE YourDatabase
GO
CREATE USER UserName FOR LOGIN LoginName
GO
```

٤. **تخصیص Role ها و Permission ها:** برای اعطای مجوز در سطح گسترده، User ها را به شیوه زیر، به Role های پایگاه داده اضافه کنید.

```
EXEC sp_addrolemember 'db_datareader', 'UserName'
GO
```

برای اعطای مجوزهای خاص و کنترل جزئی‌تر، می‌توان Permission ها را به‌طور مستقیم به یک کاربر یا نقش اعطا کرد. این عمل، ازطریق دستور امکان‌پذیر است:

```
GRANT SELECT, INSERT ON YourTable TO UserName
GO
```

٥. **پیکربندی مجوزهای سطح سرور (اختیاری):** برای وظایفی که نیاز به دسترسی سطح سرور دارند، مانند ایجاد پایگاه داده یا مدیریت Login ها، ممکن است لازم باشد نقش‌ها یا مجوزهای سطح سرور را اختصاص دهید. این کار را ازطریق دستور زیر انجام دهید:

```
ALTER SERVER ROLE sysadmin ADD MEMBER [DOMAIN\User];
GO
```

٦. **آزمایش و اعتبارسنجی مجوزها**

با استفاده از sqlcmd ، SSMS یا یک ابزار کلاینت دیگر، به‌عنوان کاربر جدید وارد شوید تا پیکربندی Authorization را آزمایش کنید. در این گام می‌توانید برای انجام اقدامات مطابق با مجوزهای اعطاشده کارکرد کانفیگ مذکور را تست کنید. مشابه زیر، برای آزمایش مجوزها از داخل SQL Server، از دستور EXECUTE AS استفاده کنید:

```
EXECUTE AS USER = 'UserName';
-- Test permissions here
REVERT;
```

جمع بندی: راهنمای پیکربندی Authentication و Authorization

در این مقاله، مراحل مربوط به پیکربندی Authentication و Authorization را به‌صورت گام‌به‌گام و با جزئیات شرح دادیم و تفاوت و دلایل اهمیت آن‌ها در SQL Server را بررسی کردیم. برای آشنایی هرچه بیشتر با کوئری‌نویسی و آموزش T-SQL، پیشنهاد می‌شود مقاله پرکاربردترین دستورات SQL Server و مقاله اسکریپت های SQL را نیز مطالعه کنید. اگر هنوز شناخت کافی از SQL Server و محیط آن ندارید، بهتر است به آموزش جامع SQL Server نیز رجوع کنید.