



We use optional cookies to improve your experience on our websites, such as through social media connections, and to display personalized advertising based on your online activity. If you reject optional cookies, only cookies necessary to provide you the services will be used. You may change your selection by clicking “Manage Cookies” at the bottom of the page. [Privacy Statement](#) [Third-Party Cookies](#)

Accept

Reject

Manage cookies

Microsoft Ignite

Nov 19–22, 2024

Register now >



Learn

Discover >

Product documentation >

Development languages >

Topics >



Sign in

.NET

Languages >

Features >

Workloads >

APIs >

Troubleshooting

Resources >

Download .NET

Version

.NET 8

Search

System.DirectoryServices.
AccountManagement

- > AdvancedFilters
- > AuthenticablePrincipal
- > ComputerPrincipal
 - ContextOptions
 - ContextType
- > DirectoryObjectClassAttribute
- > DirectoryPropertyAttribute
- > DirectoryRdnPrefixAttribute
- > GroupPrincipal
 - GroupScope
 - IdentityType
 - MatchType
- > MultipleMatchesException
- > NoMatchingPrincipalException
- > PasswordException
- > Principal
- > PrincipalCollection
- > PrincipalContext
- > PrincipalException
- > PrincipalExistsException
- > PrincipalOperationException
- > PrincipalSearcher
- > PrincipalSearchResult<T>
- > PrincipalServerDownException
- > PrincipalValueCollection<T>
- > UserPrincipal

Learn / .NET / API browser /



System.DirectoryServices.Account Management Namespace

Reference

[Feedback](#)

In this article

[Classes](#)

[Enums](#)

[Remarks](#)

[See also](#)

Provides uniform access and manipulation of user, computer, and group security principals across the multiple principal stores: Active Directory Domain Services (AD DS), Active Directory Lightweight Directory Services (AD LDS), and Machine SAM (MSAM). [System.DirectoryServices.AccountManagement](#) manages directory objects independent of the [System.DirectoryServices](#) namespace.

Classes


[Expand table](#)

AdvancedFilters	This class provides writable access to certain attributes so that users can modify read-only properties of the "dummy" principal object that is passed to a PrincipalSearcher object when using Query By Example .
AuthenticablePrincipal	Encapsulates the account and contact data common to principals that can be authenticated.
ComputerPrincipal	Encapsulates principals that are computer accounts.
DirectoryObjectClassAttribute	Represents the schema object that is used to create an object of this type in the directory. This attribute is required for principal extensions and can only be set on classes.
DirectoryPropertyAttribute	Contains the data required by the store to map a principal property to a directory attribute. This attribute is required for principal extensions and can only be set on a property. It must be specified on every property that represents a directory attribute in the extended class.
DirectoryRdnPrefixAttribute	The RDN prefix used to construct the RDN for the new object that is inserted into the store. The default RDN prefix of "CN" is used by the Account Management API

 Download PDF

	if this attribute is not set. This attribute is optional and can only be set on principal extension classes.
GroupPrincipal	Encapsulates group accounts. Group accounts can be arbitrary collections of principal objects or accounts created for administrative purposes.
MultipleMatchesException	This exception is thrown by methods that expect to match a single principal object when there are multiple matches to the search query.
NoMatchingPrincipalException	This exception is thrown when no matching principal object could be found with the specified parameters.
PasswordException	This exception is thrown when a password does not meet complexity requirements.
Principal	Encapsulates the account data and operations common to all security principals. This is the abstract base class from which all security principals are derived.
PrincipalCollection	A mutable collection of objects derived from the Principal class. This class is designed to be used for multi-valued properties that contain Principal objects. Manipulating the contents of this collection changes the contents of the corresponding store property, which is made permanent when Save() is called on the corresponding principal object.
PrincipalContext	Encapsulates the server or domain against which all operations are performed, the container that is used as the base of those operations, and the credentials used to perform the operations.
PrincipalException	The base class of exceptions thrown by System.DirectoryServices.AccountManagement objects.
PrincipalExistsException	Thrown by Add method when an attempt is made to insert a principal that already exists in the collection, or by Save() when an attempt is made to save a new principal that already exists in the store.
PrincipalOperationException	Thrown when ADSI returns an error during an operation to update the store.
PrincipalSearcher	Encapsulates the methods and search patterns used to execute a query against the underlying principal store.
PrincipalSearchResult<T>	Returns a collection of Principal objects that are returned by a search.
PrincipalServerDownException	This exception is thrown when the API is unable to connect to the server.
PrincipalValueCollection<T>	Multi-valued properties, such as PermittedWorkstations , have a value of the type PrincipalValueCollection<T> . This class provides methods to enumerate and manipulate those values.
UserPrincipal	Encapsulates principals that are user accounts.

Enums

 Expand table

ContextOptions	Specifies the options that are used for binding to the server. The application can set multiple options that are linked with a bitwise OR operation.
ContextType	Specifies the type of store to which the principal belongs.
GroupScope	Specifies the scope of the group principal.
IdentityType	Specifies the format of the identity.
MatchType	The MatchType enumeration specifies the type of comparison used in a search.

Remarks

Managed directory services applications can take advantage of the [System.DirectoryServices.AccountManagement](#) API to simplify management of user, computer and group principals. Solutions that previously required intricate knowledge of the store or lengthy code, such as finding all groups to which a user belongs, are accomplished in a few lines of code with the [System.DirectoryServices.AccountManagement](#) API.

The following features are available on the [System.DirectoryServices.AccountManagement](#) API:

- Basic directory operations such as creating and updating security principals is simplified. The application requires less knowledge of the underlying stores to perform these operations.
- Applications can extend the object model to include new types of directory objects.
- Account management tasks, such as enabling and disabling a user account, are simplified.
- Cross-store support allows group objects in the Active Directory Domain Services (AD DS), Active Directory Lightweight Directory Services (AD LDS), and Machine SAM (MSAM) databases to contain members from different types of stores.
- Query by example searching, available on the [PrincipalSearcher](#) class, enables applications to set properties on a principal object and search the selected store for other objects that contain matching property values.
- Enhanced search on computer, user and group principal objects enables applications to search the selected store for matching principal objects.
- Recursive search, available on the group principal object, enables applications to search a group recursively and return only principal objects that are leaf nodes.
- Credential validation against the Machine SAM, AD DS, and AD LS stores is simplified.
- Connections speeds are increased by using the Fast Concurrent Bind (FSB) feature when available. Connection caching decreases the number of ports used.

See also

- [System.DirectoryServices](#)
- [System.DirectoryServices.AccountManagement Namespace Overview](#)



Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).



.NET feedback

.NET is an open source project. Select a link to provide feedback:

-  [Open a documentation issue](#)
-  [Provide product feedback](#)