# Release History

## 1.11.4 (2024-06-10)

### Bugs Fixed

* Managed identity bug fixes

## 1.11.3 (2024-05-07)

### Bugs Fixed

* Fixed a regression in DefaultAzureCredential probe request behavior for IMDS managed identity environments. [#43796](https://github.com/Azure/azure-sdk-for-net/issues/43796)

## 1.11.2 (2024-04-19)

### Bugs Fixed

* Fixed an issue which caused claims to be incorrectly added to confidential client credentials such as DeviceCodeCredential [#43468](https://github.com/Azure/azure-sdk-for-net/issues/43468)

## 1.11.1 (2024-04-16)

### Other Changes

* Updated Microsoft.Identity.Client and related dependencies to version 4.60.3

## 1.11.0 (2024-04-09)

### Bugs Fixed

* AzurePowerShellCredential now handles the case where it falls back to legacy PowerShell without relying on the error message string.

### Breaking Changes

* DefaultAzureCredential now sends a probe request with no retries for IMDS managed identity environments to avoid excessive retry delays when the IMDS endpoint is not available. This should improve credential chain resolution for local development scenarios. See [BREAKING\_CHANGES.md](https://github.com/Azure/azure-sdk-for-net/blob/main/sdk/identity/Azure.Identity/BREAKING_CHANGES.md#1110).

## 1.11.0-beta.1 (2024-02-06)

### Bugs Fixed

* Claims from the TokenRequestContext are now correctly sent through to MSAL in ConfidentialClient credentials. [#40451](https://github.com/Azure/azure-sdk-for-net/issues/40451).
* ManagedIdentityCredential is more lenient with the error message it matches when falling through to the next credential in the chain in the case that Docker Desktop returns a 403 response when attempting to access the IMDS endpoint. [#38218](https://github.com/Azure/azure-sdk-for-net/issues/38218)

### Other Changes

* AzureCliCredential utilizes the new expires\_on property returned by az account get-access-token to determine token expiration.

## 1.10.4 (2023-11-13)

### Other Changes

* Distributed tracing with ActivitySource is stable and no longer requires the [Experimental feature-flag](https://github.com/Azure/azure-sdk-for-net/blob/main/sdk/core/Azure.Core/samples/Diagnostics.md).

## 1.10.3 (2023-10-18)

### Bugs Fixed

* ManagedIdentityCredential will now correctly retry when the instance metadata endpoint returns a 410 response. [#28568](https://github.com/Azure/azure-sdk-for-net/issues/28568)

### Other Changes

* Updated Microsoft.Identity.Client dependency to version 4.56.0

## 1.10.2 (2023-10-10)

### Bugs Fixed

* Bug fixes for development time credentials.

## 1.10.1 (2023-09-12)

### Bugs Fixed

* ManagedIdentityCredential will fall through to the next credential in the chain in the case that Docker Desktop returns a 403 response when attempting to access the IMDS endpoint. [#38218](https://github.com/Azure/azure-sdk-for-net/issues/38218)
* Fixed an issue where interactive credentials would still prompt on the first GetToken request even when the cache is populated and an AuthenticationRecord is provided. [#38431](https://github.com/Azure/azure-sdk-for-net/issues/38431)

## 1.10.0 (2023-08-14)

### Features Added

* Added BrowserCustomization property to InteractiveBrowserCredential to enable web view customization for interactive authentication.

### Bugs Fixed

* ManagedIdentityCredential will no longer attempt to parse invalid json payloads on responses from the managed identity endpoint.
* Fixed an issue where AzurePowerShellCredential fails to parse the token response from Azure PowerShell. [#22638](https://github.com/Azure/azure-sdk-for-net/issues/22638)

## 1.10.0-beta.1 (2023-07-17)

### Features Added

* Continuous Access Evaluation (CAE) is now configurable per-request by setting the IsCaeEnabled property of TokenRequestContext via its constructor.
* Added IsUnsafeSupportLoggingEnabled property to TokenCredentialOptions which equates to passing 'true' for the enablePiiLogging parameter to the 'WithLogging' method on the MSAL client builder.

### Bugs Fixed

* Fixed an issue with TokenCachePersistenceOptions where credentials in the same process would share the same cache, even if they had different configured names.
* ManagedIdentityCredential now ignores empty ClientId values. [#37100](https://github.com/Azure/azure-sdk-for-net/issues/37100)
* ManagedIdentityCredential will no longer attempt to parse invalid json payloads on responses from the managed identity endpoint.
* When utilizing EnvironmentCredential from DefaultAzureCredential the credential will now override the TENANT\_ID environment value if the TenantId value is set in DefaultAzureCredentialOptions.

### Other Changes

* All developer credentials in the DefaultAzureCredential credential chain will fall through to the next credential in the chain on any failure. Previously, some exceptions would throw AuthenticationFailedException, which stops further progress in the chain.

## 1.9.0 (2023-05-09)

### Breaking Changes

* Changed visibility of all environment variable based properties on EnvironmentCredentialOptions to internal. These options are again only configurable via environment variables.

## 1.9.0-beta.3 (2023-04-12)

### Breaking Changes

* Renamed the developer credential options timeout settings as follows:
  + AzureCliCredential to AzureCliCredentialOptions.ProcessTimeout
  + AzurePowerShellCredential to AzurePowerShellCredentialOptions.ProcessTimeout
  + VisualStudioCredential to VisualStudioCredentialOptions.ProcessTimeout
  + AzureDeveloperCliCredential to AzureDeveloperCliCredentialOptions.ProcessTimeout

### Bugs Fixed

* Setting DefaultAzureCredentialOptions.ExcludeWorkloadIdentityCredential to true also excludes TokenExchangeManagedIdentitySource when using DefaultAzureCredential selects the ManagedIdentityCredential

## 1.9.0-beta.2 (2023-02-21)

### Features Added

* Allow VisualStudioCredential on non-Windows platforms
* Added AzureDeveloperCliCredential for Azure Developer CLI
* Added WorkloadIdentityCredential to support Azure Workload Identity authentication
* Added WorkloadIdentityCredential and AzureDeveloperCliCredential to the DefaultAzureCredential authentication flow.

### Bugs Fixed

* Fixed ManagedIdentityCredential authentication in sovereign clouds for services specifying TenantId through authentication challenge [#34077](https://github.com/Azure/azure-sdk-for-net/issues/34077)

### Breaking Changes

* Previously, if environment variables for username and password auth are set in addition to the AZURE\_CLIENT\_CERTIFICATE\_PATH, EnvironmentCredential would select the UsernamePasswordCredential. After this change, ClientCertificateCredential will be selected, which is consistent with all other languages. This is potentially a behavioral breaking change.

## 1.8.2 (2023-02-08)

### Bugs Fixed

* Fixed error message parsing in AzurePowerShellCredential which would misinterpret Microsoft Entra ID errors with the need to install PowerShell. [#31998](https://github.com/Azure/azure-sdk-for-net/issues/31998)
* Fix regional endpoint validation error when using ManagedIdentityCredential. [#32498])(<https://github.com/Azure/azure-sdk-for-net/issues/32498>)

## 1.8.1 (2023-01-13)

### Bugs Fixed

* Fixed an issue when using ManagedIdentityCredential in combination with authorities other than Azure public cloud that resulted in a incorrect instance metadata validation error. [#32498](https://github.com/Azure/azure-sdk-for-net/issues/32498)

## 1.8.0 (2022-11-08)

### Bugs Fixed

* Fixed error message parsing in AzureCliCredential which would misinterpret Microsoft Entra ID errors with the need to login with az login. [#26894](https://github.com/Azure/azure-sdk-for-net/issues/26894), [#29109](https://github.com/Azure/azure-sdk-for-net/issues/29109)
* ManagedIdentityCredential will no longer fail when a response received from the endpoint is invalid JSON. It now treats this scenario as if the credential is unavailable. [#30467](https://github.com/Azure/azure-sdk-for-net/issues/30467), [#32061](https://github.com/Azure/azure-sdk-for-net/issues/32061)

## 1.9.0-beta.1 (2022-10-13)

### Features Added

* Credentials that are implemented via launching a sub-process to acquire tokens now have configurable timeouts. This addresses scenarios where these proceses can take longer than the current default timeout values. (A community contribution, courtesy of [*reynaldoburgos*](https://github.com/reynaldoburgos)). The affected credentials and their associated options are:
  + AzureCliCredential and AzureCliCredentialOptions.CliProcessTimeout
  + AzurePowerShellCredential and AzurePowerShellCredentialOptions.PowerShellProcessTimeout
  + VisualStudioCredential and VisualStudioCredentialOptions.VisualStudioProcessTimeout
  + DefaultAzureCredential and DefaultAzureCredentialOptions.DeveloperCredentialTimeout Note: this option applies to all developer credentials above when using DefaultAzureCredential.

### Acknowledgments

Thank you to our developer community members who helped to make Azure Identity better with their contributions to this release:

* [*reynaldoburgos*](https://github.com/reynaldoburgos)

## 1.8.0-beta.1 (2022-10-13)

### Features Added

* Reintroduced ManagedIdentityCredential token caching support from 1.7.0-beta.1
* EnvironmentCredential updated to support specifying a certificate password via the AZURE\_CLIENT\_CERTIFICATE\_PASSWORD environment variable

### Breaking Changes

* Excluded VisualStudioCodeCredential from DefaultAzureCredential token chain by default as SDK authentication via Visual Studio Code is broken due to issue [#27263](https://github.com/Azure/azure-sdk-for-net/issues/27263). The VisualStudioCodeCredential will be re-enabled in the DefaultAzureCredential flow once a fix is in place. Issue [#30525](https://github.com/Azure/azure-sdk-for-net/issues/30525) tracks this. In the meantime Visual Studio Code users can authenticate their development environment using the [Azure CLI](https://learn.microsoft.com/cli/azure/).

## 1.7.0 (2022-09-19)

### Features Added

* Added AdditionallyAllowedTenants to the following credential options to force explicit opt-in behavior for multi-tenant authentication:
  + AuthorizationCodeCredentialOptions
  + AzureCliCredentialOptions
  + AzurePowerShellCredentialOptions
  + ClientAssertionCredentialOptions
  + ClientCertificateCredentialOptions
  + ClientSecretCredentialOptions
  + DefaultAzureCredentialOptions
  + OnBehalfOfCredentialOptions
  + UsernamePasswordCredentialOptions
  + VisualStudioCodeCredentialOptions
  + VisualStudioCredentialOptions
* Added TenantId to DefaultAzureCredentialOptions to avoid having to set InteractiveBrowserTenantId, SharedTokenCacheTenantId, VisualStudioCodeTenantId, and VisualStudioTenantId individually.

### Bugs Fixed

* Fixed overly restrictive scope validation to allow the '\_' character, for common scopes such as user\_impersonation [#30647](https://github.com/Azure/azure-sdk-for-net/issues/30647)

### Breaking Changes

* Credential types supporting multi-tenant authentication will now throw AuthenticationFailedException if the requested tenant ID doesn't match the credential's tenant ID, and is not included in the AdditionallyAllowedTenants option. Applications must now explicitly add additional tenants to the AdditionallyAllowedTenants list, or add '\*' to list, to enable acquiring tokens from tenants other than the originally specified tenant ID. See [BREAKING\_CHANGES.md](https://github.com/Azure/azure-sdk-for-net/blob/main/sdk/identity/Azure.Identity/BREAKING_CHANGES.md#170).
* ManagedIdentityCredential token caching added in 1.7.0-beta.1 has been removed from this release and will be added back in 1.8.0-beta.1

## 1.7.0-beta.1 (2022-08-09)

### Features Added

* ManagedIdentityCredential will now internally cache tokens. Apps can call GetToken or GetTokenAsync directly without needing to cache to avoid throttling.

## 1.6.1 (2022-08-08)

### Bugs Fixed

* Fixed AZURE\_REGIONAL\_AUTHORITY\_NAME support in ClientCertificateCredential [#29112](https://github.com/Azure/azure-sdk-for-net/issues/29112)
* Fixed regression in SharedTokenCacheCredential default behavior [#28029](https://github.com/Azure/azure-sdk-for-net/issues/28029)
* Fixed legacy PowerShell discovery failures [#28030](https://github.com/Azure/azure-sdk-for-net/issues/28030) (A community contribution, courtesy of [*nerddtvg*](https://github.com/nerddtvg))

### Other Changes

* Documentation improvements to TokenCacheRefreshArgs and EnvironmentCredential (Community contributions, courtesy of [*pmaytak*](https://github.com/pmaytak) and [*goenning*](https://github.com/goenning))

### Acknowledgments

Thank you to our developer community members who helped to make Azure Identity better with their contributions to this release:

* [*nerddtvg*](https://github.com/nerddtvg)
* [*pmaytak*](https://github.com/pmaytak)
* [*goenning*](https://github.com/goenning)

## 1.6.0 (2022-04-05)

### Features Added

* Added a new property under the Diagnostics options available on TokenCredentialOptions and all sub-types. If set to true, we try to log the account identifiers by parsing the received access token. The account identifiers we try to log are the:
  + Application or Client Identifier
  + User Principal Name
  + Tenant Identifier
  + Object Identifier of the authenticated user or app
* ManagedIdentityCredential now attempts to use the newest "2019-08-01" api version for App Service Managed Identity sources. The newer API version will be used if the IDENTITY\_ENDPOINT and IDENTITY\_HEADER environment variables are set.

### Bugs Fixed

* Fixed an issue where the x5c header is not sent for OnBehalfOfCredential when the SendCertificateChain option is set. [#27679](https://github.com/Azure/azure-sdk-for-net/issues/27679)

## 1.6.0-beta.1 (2022-02-11)

### Features Added

* EnvironmentCredential now supports certificate subject name / issuer based authentication with AZURE\_CLIENT\_SEND\_CERTIFICATE\_CHAIN environment variable (A community contribution, courtesy of [*trevorlacey-msft*](https://github.com/trevorlacey-msft)*)*.
* ManagedIdentityCredential now supports accepting a ResourceIdentifier argument to specify a User Assigned Managed Identity by resource Id rather than client Id. DefaultAzureCredential also supports this via the ManagedIdentityResourceId property of DefaultAzureCredentialOptions.
* Added ClientAssertionCredential for authenticating service principals with a presigned client assertion.

### Bugs Fixed

* Fixed AuthenticationFailedException from AzurePowerSheellCredential when not logged in on non-windows platforms [#23498](https://github.com/Azure/azure-sdk-for-net/issues/23498)
* Fixed ManagedIdentityCredential response parsing to handle non-json responses [#24158](https://github.com/Azure/azure-sdk-for-net/issues/24158)

### Other Changes

* Upgraded MSAL dependency to version 4.39.0

### Acknowledgments

Thank you to our developer community members who helped to make Azure Identity better with their contributions to this release:

* Trevor Lacey *(*[*GitHub*](https://github.com/trevorlacey-msft)*)*

## 1.5.0 (2021-10-14)

### Breaking Changes from 1.5.0-beta.4

* The AllowMultiTenantAuthentication option has been removed and the default behavior is now as if it were true. The multi-tenant discovery feature can be totally disabled by either setting an AppContext switch named "Azure.Identity.DisableTenantDiscovery" to true or by setting the environment variable "AZURE\_IDENTITY\_DISABLE\_MULTITENANTAUTH" to "true".
* Removed the IsPIILoggingEnabled property from TokenCredentialOptions, similar functionality is planned to be added to TokenCredentialOptions.Diagnostics in a later release.
* Removed RegionalAuthority from ClientCertificateCredentialOptions and ClientSecretCredentialOptions, along with the RegionalAuthority type.
* Renamed struct TokenCacheDetails to TokenCacheData.
* Renamed class TokenCacheNotificationDetails to TokenCacheRefreshArgs.
* Updated CacheBytes property on TokenCacheData to be readonly and a required constructor parameter.

### Bugs Fixed

* Fixed issue with AuthorizationCodeCredential not specifying correct redirectUrl (Issue [#24183](https://github.com/Azure/azure-sdk-for-net/issues/24183))

### Other Changes

* Updated error messages to include links to the Azure.Identity troubleshooting guide.

## 1.5.0-beta.4 (2021-09-08)

### Features Added

* DefaultAzureCredentialOptions now has a InteractiveBrowserClientId property which allows passing a ClientId value to the InteractiveBrowserCredential when constructing a DefaultAzureCredential.
* Implement OnBehalfOfCredential which enables authentication to Microsoft Entra ID using an On-Behalf-Of flow.
* Added support to ManagedIdentityCredential for Azure hosts using federated token exchange for managed identity.

### Bugs Fixed

* Refactored IMDS discovery to remove socket probing and caching of failures to improve ManagedIdentityCredential resiliency. [#23028](https://github.com/Azure/azure-sdk-for-net/issues/23028)
* Updated UsernamePasswordCredential to use cached tokens when available [#23324](https://github.com/Azure/azure-sdk-for-net/issues/23324)

### Other Changes

* Updated credentials using MsalConfidentialClient to include MSAL log output in logs
* Added additional logging to AzureCliCredential, AzurePowerShellCredential, VisualStudioCredential, and VisualStudioCodeCredential when IsPIILoggingEnabled is set to true.

## 1.5.0-beta.3 (2021-08-10)

### Acknowledgments

Thank you to our developer community members who helped to make Azure Identity better with their contributions to this release:

* Tomas Pajurek *(*[*tpajurek-dtml*](https://github.com/tomas-pajurek)*)*

### Features Added

* A new trace event is now logged when DefaultAzureCredential selects a credential during initialization.
* Added AzureApplicationCredential
* Added IsPIILoggingEnabled property to TokenCredentialOptions, which controls whether MSAL PII logging is enabled, and other sensitive credential related logging content.

### Breaking Changes

* Renamed AZURE\_POD\_IDENTITY\_TOKEN\_URL to AZURE\_POD\_IDENTITY\_AUTHORITY\_HOST. The value should now be a host, for example "<http://169.254.169.254>" (the default).

### Bugs Fixed

* Stopped loading $PROFILE and checking for updates when using AzurePowerShellCredential.
* Fixed unrecognized argument issue in AzureCliCredential when specifying the TenantId option. [#23158](https://github.com/Azure/azure-sdk-for-net/issues/23158) (A community contribution, courtesy of [*tomas-pajurek*](https://github.com/tomas-pajurek)*)*.
* Handled an additional error scenario for AzureCliCredential that prompts developers to run az login when needed. [#21758](https://github.com/Azure/azure-sdk-for-net/issues/21758)
* Fixed an issue in EnvironmentCredential where the supplied options were not getting properly applied. [#22787](https://github.com/Azure/azure-sdk-for-net/issues/22787)
* Fixed DateTime parsing to use the current culture in AzurePowerShellCredential. [#22638](https://github.com/Azure/azure-sdk-for-net/issues/22638)

## 1.4.1 (2021-08-04)

### Fixes and improvements

* Fixed issue resulting in duplicate event source names when executing in Azure Functions

## 1.5.0-beta.2 (2021-07-12)

### New Features

* Added regional STS support to client credential types
  + Added RegionalAuthority extensible enum
  + Added RegionalAuthority property to ClientSecretCredentialOptions and ClientCertificateCredentialOptions
* Added support to ManagedIdentityCredential for Bridge to Kubernetes local development authentication.
* TenantId values returned from service challenge responses can now be used to request tokens from the correct tenantId. To support this feature, there is a new AllowMultiTenantAuthentication option on TokenCredentialOptions.
  + By default, AllowMultiTenantAuthentication is false. When this option property is false and the tenant Id configured in the credential options differs from the tenant Id set in the TokenRequestContext sent to a credential, an AuthorizationFailedException will be thrown. This is potentially breaking change as it could be a different exception than what was thrown previously. This exception behavior can be overridden by either setting an AppContext switch named "Azure.Identity.EnableLegacyTenantSelection" to true or by setting the environment variable "AZURE\_IDENTITY\_ENABLE\_LEGACY\_TENANT\_SELECTION" to "true". Note: AppContext switches can also be configured via configuration like below:
* Added OnBehalfOfFlowCredential which enables support for Microsoft Entra On-Behalf-Of (OBO) flow. See the [Microsoft Entra ID documentation](https://learn.microsoft.com/entra/identity-platform/v2-oauth2-on-behalf-of-flow) to learn more about OBO flow scenarios.

<ItemGroup>

<RuntimeHostConfigurationOption Include="Azure.Identity.EnableLegacyTenantSelection" Value="true" />

</ItemGroup>

## 1.5.0-beta.1 (2021-06-08)

### Fixes and improvements

* Added LoginHint property to InteractiveBrowserCredentialOptions which allows a user name to be pre-selected for interactive logins. Setting this option skips the account selection prompt and immediately attempts to login with the specified account.
* Added AuthorizationCodeCredentialOptions which allows for configuration of a ReplyUri.

## 1.4.0 (2021-05-12)

### New Features

* By default, the MSAL Public Client Client Capabilities are populated with "CP1" to enable support for [Continuous Access Evaluation (CAE)](https://learn.microsoft.com/entra/identity-platform/app-resilience-continuous-access-evaluation). This indicates to Microsoft Entra ID that your application is CAE ready and can handle the CAE claim challenge. This capability can be disabled, if necessary, by either setting an AppContext switch named "Azure.Identity.DisableCP1" to true or by setting the environment variable; "AZURE\_IDENTITY\_DISABLE\_CP1" to "true". Note: AppContext switches can also be configured via configuration like below:

<ItemGroup>

<RuntimeHostConfigurationOption Include="Azure.Identity.DisableCP1" Value="true" />

</ItemGroup>

### Fixes and improvements

* The Microsoft Authentication Library (MSAL) dependency versions have been updated to the latest
  + Microsoft.Identity.Client version 4.30.1, Microsoft.Identity.Client.Extensions.Msal version 2.18.4.

## 1.4.0-beta.5 (2021-04-06)

### Acknowledgments

Thank you to our developer community members who helped to make Azure Identity better with their contributions to this release:

* Marco Mansi *(*[*GitHub*](https://github.com/olandese)*)*

### New Features

* Added AzurePowerShellCredential to DefaultAzureCredential (A community contribution, courtesy of [*olandese*](https://github.com/olandese)*)*

### Fixes and improvements

* When logging is enabled, the log output from Microsoft Authentication Library (MSAL) is also logged.
* Fixed an issue where an account credential fails to load from the cache when EnableGuestTenantAuthentication is true and the account found in the cache has multiple matching tenantIds ([#18276](https://github.com/Azure/azure-sdk-for-net/issues/18276)).
* Fixed deadlock issue in InteractiveBrowserCredential when running in a UI application ([#18418](https://github.com/Azure/azure-sdk-for-net/issues/18418)).

### Breaking Changes

* TokenCache class is moved removed from the public API surface and has been replaced by TokenCachePersistenceOptions for configuration of disk based persistence of the token cache.

## 1.4.0-beta.4 (2021-03-09)

### Fixes and Improvements

* Added the [Serializable] attribute to all custom exception types.

### Breaking Changes

* Update the default value of ExcludeSharedTokenCacheCredential on DefaultAzureCredentialsOptions to true, to exclude the SharedTokenCacheCredential from the DefaultAzureCredential by default. See [BREAKING\_CHANGES.md](https://github.com/Azure/azure-sdk-for-net/blob/main/sdk/identity/Azure.Identity/BREAKING_CHANGES.md#140)

## 1.4.0-beta.3 (2021-02-09)

### Breaking Changes

* The IDisposable interface has been removed from TokenCache.

### New Features

* All credentials added support to handle the Claims property on TokenRequestContext

## 1.4.0-beta.2 (2021-01-29)

### Fixes and improvements

* Fixed issue in HttpExtensions causing the omission of content headers on requests ([#17448](https://github.com/Azure/azure-sdk-for-net/issues/17448))
* Made EnvironmentCredential to account for both null and empty string when checking for the presense of the environment variables ([#18031](https://github.com/Azure/azure-sdk-for-net/issues/18031))

## 1.3.0 (2020-11-12)

### New Features

* Added support for Service Fabric managed identity authentication to ManagedIdentityCredential.
* Added support for Azure Arc managed identity authentication to ManagedIdentityCredential.

### Fixes and improvements

* Fix race condition in ProcessRunner causing VisualStudioCredential and AzureCliCredential to fail intermittently ([#16211](https://github.com/Azure/azure-sdk-for-net/issues/16211))
* Fix VisualStudioCodeCredential to raise CredentialUnavailableException when reading from VS Code's stored secret ([#16795](https://github.com/Azure/azure-sdk-for-net/issues/16795))
* Prevent VisualStudioCodeCredential using invalid authentication data when no user is signed in to Visual Studio Code ([#15870](https://github.com/Azure/azure-sdk-for-net/issues/15870))
* Fix deadlock in ProcessRunner causing AzureCliCredential and VisualStudioCredential to fail due to timeout ([#14691](https://github.com/Azure/azure-sdk-for-net/issues/14691), [14207](https://github.com/Azure/azure-sdk-for-net/issues/14207))
* Fix issue with AzureCliCredential incorrectly parsing expires on property returned from az account get-access-token ([#15801](https://github.com/Azure/azure-sdk-for-net/issues/15801))
* Fix issue causing DeviceCodeCredential and InteractiveBrowserCredential to improperly authenticate to the home tenant for silent authentication calls after initial authentication ([#13801](https://github.com/Azure/azure-sdk-for-net/issues/13801))
* Fix cache loading issue in SharedTokenCacheCredential on Linux ([#12939](https://github.com/Azure/azure-sdk-for-net/issues/12939))

### Breaking Changes

* Rename property IncludeX5CCliamHeader on ClientCertificateCredentialOptions to SendCertificateChain
* Removing Application Authentication APIs for GA release. These will be reintroduced in 1.4.0-beta.1.
  + Removed class AuthenticationRecord
  + Removed class AuthenticationRequiredException
  + Removed class ClientSecretCredentialOptions and ClientSecretCredential constructor overloads accepting this type
  + Removed class UsernamePasswordCredentialOptions and UsernamePasswordCredential constructor overloads accepting this type
  + Removed properties EnablePersistentCache and AllowUnprotectedCache from ClientCertificateCredentialOptions, DeviceCodeCredentialOptions and InteractiveBrowserCredentialOptions
  + Removed properties AuthenticationRecord and DisableAutomaticAuthentication from DeviceCodeCredentialOptions and InteractiveBrowserCredentialOptions
  + Removed properties AllowUnencryptedCacheand AuthenticationRecord from SharedTokenCacheCredentialOptions
  + Removed methods Authenticate and AuthenticateAsync from DeviceCodeCredential, InteractiveBrowserCredential and UsernamePasswordCredential

## 1.4.0-beta.1 (2020-10-15)

### New Features

* Redesigned Application Authentication APIs
  + Adds TokenCache and TokenCache classes to give more user control over how the tokens are cached and how the cache is persisted.
  + Adds TokenCache property to options for credentials supporting token cache configuration.

## 1.3.0-beta.2 (2020-10-07)

### New Features

* Update DeviceCodeCredential to output device code information and authentication instructions in the console, in the case no deviceCodeCallback is specified.
  + Added DeviceCodeCallback to DeviceCodeCredentialOptions
  + Added default constructor to DeviceCodeCredential

### Breaking Changes

* Replaced DeviceCodeCredential constructor overload taking deviceCodeCallback and DeviceCodeCredentialOptions with constructor taking only DeviceCodeCredentialOptions

## 1.3.0-beta.1 (2020-09-11)

### New Features

* Restoring Application Authentication APIs from 1.2.0-preview.6
* Added support for App Service Managed Identity API version 2019-08-01 ([#13687](https://github.com/Azure/azure-sdk-for-net/issues/13687))
* Added IncludeX5CClaimHeader to ClientCertificateCredentialOptions to enable subject name / issuer authentication with the ClientCertificateCredential.
* Added RedirectUri to InteractiveBrowserCredentialOptions to enable authentication with user specified application with a custom redirect url.
* Added IdentityModelFactory to enable constructing models from the Azure.Identity library for mocking.
* Unify exception handling between DefaultAzureCredential and ChainedTokenCredential ([#14408](https://github.com/Azure/azure-sdk-for-net/issues/14408))

### Fixes and improvements

* Updated MsalPublicClient and MsalConfidentialClient to respect CancellationToken during initialization ([#13201](https://github.com/Azure/azure-sdk-for-net/issues/13201))
* Fixed VisualStudioCodeCredential crashes on macOS (Issue [#14362](https://github.com/Azure/azure-sdk-for-net/issues/14362))
* Fixed issue with non GUID Client Ids (Issue [#14585](https://github.com/Azure/azure-sdk-for-net/issues/14585))
* Update VisualStudioCredential and VisualStudioCodeCredential to throw CredentialUnavailableException for ADFS tenant (Issue [#14639](https://github.com/Azure/azure-sdk-for-net/issues/14639))

## 1.2.3 (2020-09-11)

### Fixes and improvements

* Fixed issue with DefaultAzureCredential incorrectly catching AuthenticationFailedException (Issue [#14974](https://github.com/Azure/azure-sdk-for-net/issues/14974))
* Fixed issue with DefaultAzureCredential throwing exceptions during concurrent calls (Issue [#15013](https://github.com/Azure/azure-sdk-for-net/issues/15013))

## 1.2.2 (2020-08-20)

### Fixes and improvements

* Fixed issue with InteractiveBrowserCredential not specifying correct redirectUrl (Issue [#13940](https://github.com/Azure/azure-sdk-for-net/issues/13940))

## 1.2.1 (2020-08-18)

### Fixes and improvements

* Bug in TaskExtensions.EnsureCompleted method that causes it to unconditionally throw an exception in the environments with synchronization context

## 1.2.0 (2020-08-10)

### Breaking Changes

* Removing Application Authentication APIs for GA release. These will be reintroduced in 1.3.0-preview.
  + Removed class AuthenticationRecord
  + Removed class AuthenticationRequiredException
  + Removed class ClientCertificateCredentialOptions and ClientCertificateCredential constructor overloads accepting this type
  + Removed class ClientSecretCredentialOptions and ClientSecretCredential constructor overloads accepting this type
  + Removed class DeviceCodeCredentialOptions and DeviceCodeCredential constructor overloads accepting this type
  + Removed class InteractiveBrowserCredentialOptions and InteractiveBrowserCredential constructor overloads accepting this type
  + Removed class UsernamePasswordCredentialOptions and UsernamePasswordCredential constructor overloads accepting this type
  + Removed methods Authenticate and AuthenticateAsync from DeviceCodeCredential
  + Removed methods Authenticate and AuthenticateAsync from InteractiveBrowserCredential
  + Removed methods Authenticate and AuthenticateAsync from UsernamePasswordCredential
  + Removed properties AllowUnencryptedCacheand AuthenticationRecord from SharedTokenCacheCredentialOptions

### Fixes and improvements

* Fixed excess errors in DefaultAzureCredential tracing (Issue [#10659](https://github.com/Azure/azure-sdk-for-net/issues/10659))
* Fixed concurrency issue in DefaultAzureCredential (Issue [#13044](https://github.com/Azure/azure-sdk-for-net/issues/13044))

## 1.2.0-preview.6 (2020-07-22)

### New Features

* Added the read only property ClientId to AuthenticationRecord.
* Added the property AllowUnencryptedCache to the option classes ClientCertificateCredentialOptions, ClientSecretCredentialOptions, DeviceCodeCredentialOptions, InteractiveBrowserCredentialOptions and SharedTokenCacheCredentialOptions which when set to true allows the credential to fall back to storing tokens in an unencrypted file if no OS level user encryption is available when EnablePersistentCache is set to true.
* Added the property AuthenticationRecord to the option class SharedTokenCacheCredentialOptions to support silent authentication for accounts previously authenticated with an interactive credential.
* Added option class UsernamePasswordCredentialOptions which supports the options EnablePersistentCache and AllowUnencryptedCache.

### Breaking Changes

* Rename type KnownAuthorityHosts to AzureAuthorityHosts
  + Rename property AzureChinaCloud to AzureChina
  + Rename property AzureGermanCloud to AzureGermany
  + Rename property AzureUSGovernment to AzureGovernment
  + Rename property AzureCloud to AzurePublicCloud

## 1.2.0-preview.5 (2020-07-08)

### New Features

* Added options classes ClientCertificateCredentialOptions and ClientSecretCredentialOptions which support the following new option
  + EnablePersistentCache configures these credentials to use a persistent cache shared between credentials which set this option. By default the cache is per credential and in memory only.

## 1.2.0-preview.4 (2020-06-10)

### New Features

* Makes AzureCliCredential, VisualStudioCredential and VisualStudioCodeCredential public to allow direct usage.
* Added Authenticate methods to UsernamePasswordCredential

### Fixes and improvements

* Fix SharedTokenCacheCredential account filter to be case-insensitive (Issue [#10816](https://github.com/Azure/azure-sdk-for-net/issues/10816))
* Update VisualStudioCodeCredential to properly throw CredentialUnavailableException when re-authentication is needed. (Issue [#11595](https://github.com/Azure/azure-sdk-for-net/issues/11595))

## 1.2.0-preview.3 (2020-05-05)

### New Features

* First preview of new API for authenticating users with DeviceCodeCredential and InteractiveBrowserCredential
  + Added method Authenticate which pro-actively interacts with the user to authenticate if necessary and returns a serializable AuthenticationRecord
  + Added Options classes DeviceCodeCredentialOptions and InteractiveBrowserCredentialOptions which support the following new options
    - AuthenticationRecord enables initializing a credential with an AuthenticationRecord returned from a prior call to Authenticate
    - DisableAutomaticAuthentication disables automatic user interaction causing the credential to throw an AuthenticationRequiredException when interactive authentication is necessary.
    - EnablePersistentCache configures these credentials to use a persistent cache shared between credentials which set this option. By default the cache is per credential and in memory only.

## 1.2.0-preview.2 (2020-04-06)

### New Features

* Updates DefaultAzureCredential to enable authenticating through Visual Studio
* Updates DefaultAzureCredential to enable authentication through Visual Studio Code

## 1.2.0-preview.1 (2020-03-10)

### New Features

* Updating DefaultAzureCredential to enable authenticating through the Azure CLI
* ClientCertificateCredential now supports being constructed with a path to an unencrypted certificate (in either PFX or PEM format)
* EnvironmentCredential now supports reading a certificate path from AZURE\_CLIENT\_CERTIFICATE\_PATH

### Fixes and improvements

* Fix an issue where EnvironmentCredential did not behave correctly when AZURE\_USERNAME and AZURE\_PASSWORD where set
* Added KnownAuthorityHosts class to aid in sovereign cloud configuration.

## 1.1.1 (2020-02-10)

### Fixes and improvements

* Fixed UsernamePasswordCredential constructor parameter mishandling
* Updated ManagedIdentityCredential endpoint discovery to avoid throwing
* Fixed ManagedIdentityCredential to raise CredentialUnavailableException on 400 return from the service where no identity has been assigned
* Updated error messaging from DefaultAzureCredential to more easily root cause failures

## 1.1.0 (2019-11-25)

### Fixes and improvements

* Update SharedTokenCacheCredential to filter accounts by tenant id
  + Added SharedTokenCacheCredentialOptions class with properties TenantId and Username
  + Added constructor overload to SharedTokenCacheCredential which accepts SharedTokenCacheCredentialOptions
  + Added property SharedTokenCacheTenantId to DefaultAzureCredentialOptions
* Support for personal account authentication in DefaultAzureCredential, InteractiveBrowserCredential, and SharedTokenCacheCredential
* Added InteractiveBrowserTenantId to DefaultAzureCredentialOptions
* Fixed issue with ManagedIdentityCredential authentication with user assigned identities

## 1.0.0 (2019-10-29)

* First stable release of Azure.Identity package.

### Breaking Changes

* Rename AzureCredentialOptions -> TokenCredentialOptions
  + Renamed property VerificationUrl -> VerificationUri and changed type from string to Uri
* Updated ClientSecretCredential class
  + Removed property ClientId
  + Removed property ClientSecret
  + Removed property TenantId
* Updated ClientCertificateCredential class
  + Removed property ClientId
  + Removed property ClientCertificate
  + Removed property TenantId
* Updated DefaultAzureCredential class to derive directly from TokenCredential rather than ChainedTokenCredential
* Updated DefaultAzureCredentialOptions class
  + Renamed property PreferredAccountUsername -> SharedTokenCacheUsername
  + Renamed property IncludeEnvironmentCredential -> ExcludeEnvironmentCredential
  + Renamed property IncludeManagedIdentityCredential -> ExcludeManagedIdentityCredential
  + Renamed property IncludeSharedTokenCacheCredential -> ExcludeSharedTokenCacheCredential
  + Renamed property IncludeInteractiveBrowserCredential -> ExcludeInteractiveBrowserCredential
* Updated DeviceCodeInfo class
  + Removed property Interval
  + Renamed property VerificationUrl -> VerificationUri and changed type from string to Uri
* Updated InteractiveBrowserCredential class
  + Reordered constructor parameters tenantId and clientId to be consistent with other credential types
* Updated SharedTokenCacheCredential class
  + Updated constructor to take TokenCredentialOptions
  + Removed clientId constructor parameter
* Removed class SharedTokenCacheCredentialOptions
* Updated exception model across the Azure.Identity library.
  + TokenCredential implementations in the Azure.Identity library now throw exceptions rather than returning default(AccessToken) when no token is obtained
  + Added the CredentialUnavailableExcpetion exception type to distinguish cases when failure to obtain an AccessToken was expected

### Dependency Changes

* Adopted Azure.Core 1.0.0

### Fixes and improvements

* Update ManagedIdentityCredential IMDS availability check to handle immediate network failures
* Added a DefaultAzureCredential constructor overload to enable interactive credential types by default

## 1.0.0-preview.5 (2019-10-07)

### Dependency Changes

* Adopted Azure.Core 1.0.0-preview.9

### New Features

* Added DefaultAzureCredentialOptions for configuring the DefaultAzureCredential authentication flow
* Added InteractiveBrowserCredential to the DefaultAzureCredential authentication flow, but excluded by default

### Fixes and improvements

* Updated InteractiveBrowserCredential and DeviceCodeCredential to optionally accept a tenantId to support non-multitenant applications

## 1.0.0-preview.4 (2019-09-10)

### Breaking Changes

* Modified GetToken abstraction to accept TokenRequest structure rather than string[] for forwards compatibility

### Dependency Changes

* Adopted Azure.Core 1.0.0-preview.8

### New Features

* Added SharedTokenCacheCredential to support Single Sign On with developer tooling
* Updated DefaultAzureCredentialauthentication flow to include the SharedTokenCacheCredential

## 1.0.0-preview.3 (2019-08-06)

### Dependency Changes

* Adopted Azure.Core 1.0.0-preview.7
* Adopted Microsoft.Identity.Client 4.1.0

### New Features

* User Principal Authentication
  + Added DeviceCodeCredential class
  + Added InteractiveBrowserCredential class
  + Added UsernamePasswordCredential class
* Support for Azure SDK ASP .NET Core integration

### Fixes and improvements

* Added identity client distributed tracing support

## 1.0.0-preview.2 (2019-07-02)

### Fixes and improvements

* Fix to ManagedIdentityCredential to properly parse expires\_on from response

## 1.0.0-preview.1 (2019-06-27)

Version 1.0.0-preview.1 is the first preview of our efforts to create a user-friendly authentication API for Azure SDK client libraries. For more information about preview releases of other Azure SDK libraries, please visit <https://aka.ms/azure-sdk-preview1-net>.

### New Features

* Azure Service Authentication
  + Added DefaultAzureCredential class
  + Added ChainedTokenCredential class
* Service Principal Authentication
  + Added ClientSecretCredential class
  + Added ClientCertificateCredential class
* Managed Identity Authentication
  + Added ManagedIdentityCredential class

See the [documentation](https://github.com/Azure/azure-sdk-for-net/blob/main/sdk/identity/Azure.Identity/README.md) for more details. User authentication will be added in an upcoming preview release.