

Game Service Sample

# Description

To integrate a custom web services with your Xbox Live enabled title, you will need to use XSTS tokens for authentication and user identification. Getting the X-tokens from the client is straightforward using XUserGetTokenAndSignatureUtf16Async(). However, integrating with and operating on X-tokens can be challenging. This sample is a functioning web service for an Xbox Live title that has example source code for decrypting and verifying X-tokens as well as communication with the required services to make authenticated calls on-behalf-of a user (backend to backend or b2b) to other Xbox Live services.

This can be used as an example for education on X-tokens, a reference for writing your own token handling in another language, or it can be used as a base framework to build your game’s web service.

NOTE: Because the sample is an educational tool, it is not hardened against web service attacks and if using code from the sample you should ensure that you are following best practices for DDOS, timing, and other attack vectors as part of your service’s security.

# Building the Sample

See the included Configuration and Documentation Guide for step by step instructions to build and deploy the sample.

# Using the sample

See the included Configuration and Documentation Guide for instructions on how to configure and use the sample as a running service.

# Implementation notes

This sample has the following key features and design:

* Source code access to all token handling classes and functions either through raw source code or use of Open Source Software solutions available publicly (Newtonsoft.Json, Jose.JWT, .NET Core)
* Sample able to be used as a starting foundation to create a web service
* Ability to be run on multiple server OS’s by using C# ASP.NET Core
  + Windows Server 2016 verified
  + Linux (Ubuntu 16.04) verified
* Token handling and validation for both Legacy XSTS tokens (XDP configured JWT Draft 7 Asymmetric) and the latest XSTS token format (Partner Center configured JWT RFC symmetric and asymmetric keys)
* Retrieval and management of Service Tokens required for Xbox Live server-to-server requests
* Retrieval and usage of Delegated Auth XSTS tokens required for Xbox Live server-to-server requests on-behalf-of a user
* Step by step configuration instructions in Dev Center for Relying Party, Business Partner Certificate, and NSAL setup.
* Automatic handling of the Xbox Live Issuer certificate updates and rollovers

# Known issues

* The sample does not currently monitor or handle throttled calls to Xbox Live services. When calling a service too frequently calls will return with a 429 HTTP status and information on when the next request can be made.

# Update history

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Description** |
| October 26, 2018 | 1.0 | * Initial release |
| November 14, 2018 | 1.1 | * Added Collections |
| November 28, 2018 | 1.2 | * Consolidated in-memory databases * Added Azure database support and instructions for persistent cache items (tokens, etc.) |
| January 25, 2019 | 1.3 | * Collections controller * Logging and Microsoft Correlation Vectors * RFC7516 Asymmetric token handling * Redesigned Configuration guide and sample solution layout to support concepts in Sections that will build on each other as you go through them. |
| July 30, 2019 | 1.4 | * License Token handling and validation |
| November 1, 2019 | 1.5 | * Renamed to “Game Service Sample” * Updated to .NET Core 3.0 and Visual Studio 2019 * Terminology and naming updated to match the Xfest 2019 talk *XSTS Auth and Server to Server Made Easy* * Migrated locally cached items to in-memory cache rather than in-memory database * Removed Azure Key Vault usage and migrated secrets and certs to App Settings * Reordered and condensed the sections in the configuration guide * Moved source code files to match the new sections layout |
| February 25, 2020 | 1.7 | * Added GDNP erasure list b2b endpoint * Fixed bug with S-token caching that caused errors generating signature headers |
| February 18, 2021 | 1.8 | * Removed Legacy (Draft 7) X-token handling * Update to .NET 5.0 * Updated RFC7516 token middleware to check outer token signature before decrypting anything and ensured that the checking function’s time is consistent to help prevent timing attacks. * Other code cleanup and compile optimization message cleanup |

# Privacy Statement

For more information about Microsoft’s privacy policies in general, see the [Microsoft Privacy Statement](https://privacy.microsoft.com/en-us/privacystatement/).