TLS 1.0 and TLS 1.1 soon to be disabled in Windows

By Jessica Krynitsky

Published Aug 01 2023 11:28 AM

212K Views

Learn about the upcoming changes in Schannel protocol defaults and how to remove dependencies on legacy TLS versions or keep them enabled for compatibility.

Overview

Transport Layer Security (TLS) is the most common internet protocol for setting up an encrypted channel of communication between a client and server. TLS 1.0 dates back to 1999 and, over time, several security weaknesses have been found in this protocol version. TLS 1.1 was published in 2006 and made some security improvements, but never saw broad adoption. These versions have long been surpassed by TLS 1.2 and TLS 1.3, and TLS implementations try to negotiate connections using the highest protocol version available.

Over the past several years, internet standards and regulatory bodies have deprecated or disallowed TLS versions 1.0 and 1.1, due to a variety of security issues. We have been tracking TLS protocol usage for several years and believe TLS 1.0 and TLS 1.1 usage data are low enough to act.

To increase the security posture of Windows customers and encourage modern protocol adoption, TLS versions

builds in September 2023 and future Windows OS releases. This change applies to both client and server, but it will not impact any in-market OS versions. There is an option to re-enable TLS 1.0 or TLS 1.1 for users who need to maintain compatibility.

Diagnostic events

Applications that start failing when TLS 1.0 and TLS 1.1 are disabled can be identified by Event 36871 in the Windows Event Log.

Sample Event:

A fatal error occurred while creating a TLS <client/server> credential. The internal error state is 10013. The SSPI client process is <process ID>.

Guidance for users and IT admins

The impact of this change depends largely on the Windows applications using TLS. For example, TLS 1.0 and TLS 1.1 have already been disabled by Microsoft 365 products as well as WinHTTP and WinINet API surfaces. Most newer versions of applications support TLS 1.2 or higher protocol versions. Therefore, if an application starts failing after this change, the first step is to look for a newer version of the application that has TLS 1.2 or TLS 1.3 support.

It's recommended to use the system default settings for the best balance of security and performance. If organizations limit TLS cipher suites using Group Policy or PowerShell cmdlets, they should also verify that cipher suites needed for TLS 1.3 and TLS 1.2 are enabled.

If there are no alternatives available and TLS 1.0 or TLS 1.1 is needed, the protocol versions can be re-enabled with a system registry setting. To override a system default and set a (D)TLS or SSL protocol version to the Enabled state, create a DWORD registry value named "Enabled" with an entry value of "1" under the corresponding version-specific subkey. Examples of TLS 1.0 subkeys are as follows:

HKLM SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0\Client

HKLM SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0\Server

Note: Re-enabling TLS 1.0 or TLS 1.1 on machines should only be done as a last resort, and as a temporary solution until incompatible applications can be updated or replaced. Support for these legacy TLS versions may be removed completely in the future.

Guidance for SSPI application developers

Although most applications and services use Schannel via HTTP and .NET APIs, some call the Security Support Provider Interface (SSPI) directly. Historically, SSPI callers implementing TLS clients and servers would pass the SCHANNEL_CRED structure when calling AcquireCredentialsHandle(). This allowed the hard coding of legacy TLS versions and prevented apps from using new TLS versions. With TLS 1.0 and TLS 1.1 disabled by default, an SSPI application that only allows these versions will fail to connect.

SCHANNEL_CRED was deprecated in Windows 10, and SSPI callers should specify their preferences using SCH_CREDENTIALS instead. Applications using this new structure will be able to negotiate TLS 1.3 and later protocol versions. When updating code to switch from SCHANNEL_CRED to SCH_CREDENTIALS, implementers should test their TLS client or server against a TLS 1.3 peer and ensure that the code correctly handles SEC_I_RENEGOTIATE returned from DecryptMessage().

For more information on finding and removing application dependencies on TLS 1.0 and 1.1, please refer to Solving the TLS 1.0 Problem.

Known issues

We have tested this change against top Windows applications, and found that the following versions rely on TLS 1.0 or TLS 1.1 and are expected to be broken.

Note: This is not an exhaustive list. All systems and organizations should test the disablement using the steps described above and observe any failures. Please reach out directly to the application owner, as they often have an updated version or mitigation available.

- SQL Server - 2012, 2014, 2016 (see KB3135244 - TLS 1.2 support for Microsoft SQL Server - Microsoft Support for how to upgrade to TLS 1.2 support)

- Xbox One SmartGlass - 2.2.1702.2004
- Project Plan 365 - 23.8.1204.14137
- Safari - 5.1.7
- EVault Data Protection - 7.01.6125
- Turbo Tax - 2017, 2014, 2011, 2012, 2016, 2015, 2018
- BlueStacks 3 (蓝叠3) - 5.10.0.6513
- BlueStacks X - 0.21.0.1063
- Splice - 4.0.35686, 4.2.4
- Driver Support - 10.1.2.41, 10.1.4.20
- K7 Enterprise Security and 4.1.0.116
- DRUKI Gofin - 3.17.63.0
- vWorkspace - 8.6.1
- ARMA 3
- LANGuard - 12.7.2022.0406
- Adguard - 6.4.1814.4903, 7.12.41.70.0
- 火萤视频桌面 - 5.2.5.9
- CCB Security Client (中国建设银行E路航网银安全组件) - 3.3.8.4
- ArcGIS - 10.3.3400
- ACDSee Photo Studio – 2018, 2023
- Blio e-Reader - 3.4.0.9728, 3.4.1.9759

Continue the conversation. Find best practices. Bookmark the Windows Tech Community and follow us @MSWindowsITPro on Twitter. Looking for support? Visit Windows on Microsoft Q&A.

15 Likes

29 Comments

Skip to Primary Navigation

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

Comment

Co-Authors


jess_krynitsky


Andrei_Popov

Version history

Last update: Aug 10 2023 04:15 PM
Updated by: jess_krynitsky

Labels

Share