



1 はじめに

Tabular Data Stream (TDS) プロトコルバージョン7および8は、データベースサーバーとの対話を容易にするアプリケーション層のリクエスト/レスポンスプロトコルであり、以下の内容を提供する：

- 認証とチャネルの暗号化。
- SQLによるリクエストの指定（Bulk Insertを含む）。
- **ストアドプロシージャ**またはユーザー定義関数の呼び出しで、**リモートプロシージャコール（RPC）**としても知られています。
- データの返却のことです。
- **トランザクションマネージャーの要望**。

本仕様書の 1.5 節、1.8 節、1.9 節、2 節、3 節は規範となるものである。本仕様書の他のすべてのセクションと例は、情報提供である。

1.1 用語集

本書では、以下の用語を使用しています：

Azure Active Directory Authentication Library (ADAL): Microsoft .NET Frameworkのツールで、アプリケーション開発者がクラウドまたはに配置された Active Directory に対してユーザーを認証し、APIコールへの安全なアクセスのためのトークンを取得することができます。

ビッグエンディアン複数バイトの値で、最上位バイトが最下位アドレスのメモリロケーションに格納されるようにバイトオーダーされる。

バルクインサートクライアントからサーバーへ、テーブルの行を効率よく投入する方法。

common language runtime user-defined type (CLR UDT): Microsoft .NET Framework共通言語ランタイムアセンブリを使用して、SQLをサポートするデータベースサーバー上でユーザーが作成・定義するデータ型です。

データ分類: クエリから返されるデータの感性情報を含む情報保護のフレームワーク。感性情報には、ラベルや情報タイプ、その識別子が含まれる。

data stream (データストリーム) : データストリーム: 特定のTDS (Tabular Data Stream) セマンティクスに対応するデータのストリーム。

単一のデータストリームは、TDSメッセージ全体を表すことも、TDSメッセージの特定の明確に定義された部分のみを表すこともできる。TDSデータストリームは、複数のネットワークデータパケットにまたがることができる。

Distributed Transaction Coordinator (DTC)： データベースを含む複数のリソースマネージャにまたがるトランザクションを調整するWindowsサービス。詳細は、[\[MSDN- DTC\]](#)を参照してください。

エンクレーブサーバー側でのみ使用されるメモリの保護領域。この領域はSQL Serverのアドレス空間内にあり、信頼された実行環境として機能する。エンクレーブ内で実行されるコードのみが、そのエンクレーブ内のデータにアクセスできます。エンクレーブ内のデータもコードも、デバッガを使っても外から見ことはできません。

エンクレーブ計算： Transact-SQL クエリで暗号化されたカラムに対して、エンクレーブの内部で実行されるローカルで有効な暗号化操作など。

federated authentication (フェデレート認証)： あるトラストドメインのセキュリティトークンサービス (STS) が、別のトラストドメインのIDプロバイダにユーザー認証を委ねることができる認証機構。

の信頼関係がある場合に、その信頼関係に基づいて、ユーザのセキュリティトークンを生成する。

グローバルトランザクション： Microsoft Azure SQL Databaseなどの共有サービスでホストされている複数のデータベースにまたがってトランザクションを実行できる機能です。

インターフェース： C++の仮想インターフェイスに類似した、特定の順序に関連する関数プロトタイプのグループ。異なるオブジェクトクラスの複数のオブジェクトが、同じインターフェースを実装することができる。派生インターフェースは、既存のインターフェースの末尾にメソッドを追加することで作成することができる。分散コンポーネントオブジェクトモデル（DCOM）では、すべてのインターフェースは、最初に IUnknownから派生しています。

リトルエンディアン複数バイトの値で、最下位バイトが最下位アドレスのメモリロケーションに格納されるようにバイトオーダーされたもの。

Microsoft/Windows Data Access Components（MDAC/WDAC）です： Microsoft/Windows Data Access Components (MDAC/WDAC) を使用すると、開発者はさまざまなリレーショナルおよび非リレーショナルデータソースに接続してデータを使用することができます。Open Database Connectivity (ODBC)、ActiveX Data Objects (ADO)、またはOLE DBを使用して、さまざまなデータソースに接続することができます。これは、Microsoftが構築して出荷するプロバイダやドライバ、またはさまざまなサードパーティが開発するプロバイダやドライバを使用して行うことができます。詳細については、[\[MSDN-MDAC\]](#)を参照してください。

Multiple Active Result Sets (MARS)： Microsoft SQL Server の機能で、アプリケーションは接続ごとに複数の保留中の要求を持つことができます。詳細は、[\[MSDN-MARS\]](#)を参照してください。

nullableカラム： データベーステーブルのカラムで、ある行に値が含まれないことが許されるものの。

アウト・オブ・バンド標準的なイベントの順序の外で起こるイベントの一種。例えば、帯域外信号やメッセージは予期せぬ時間に送信されることがあり、プロトコルの解析に問題が生じることはない。

クエリ通知： SQL Serverの機能で、指定したクエリ結果の変更に関する通知をクライアントに登録することができる。詳しくは[\[MSDN-QUERYNOTE\]](#)を参照してください。

リモートプロシージャコール（RPC）： 主にクライアントとサーバーの間で使用される通信プロトコル。コンピュータ間の通信機能を提供するランタイム環境（RPCランタイム）、コンピュータ間の要求と応答のメッセージ交換のセット（RPC交換）、RPC交換か

らの単一のメッセージ（RPCメッセージ）の3つの定義があり、しばしば互換的に使用されている。詳しくは、[\[C706\]](#)を参照してください。

result set（結果セット）：ストアプロシージャやクエリを実行したり、フィルタを適用した結果得られるレコードのリスト。結果セット内のデータの構造や内容は、実装によって異なる。

Security Support Provider Interface (SSPI)：接続されたアプリケーションが複数のセキュリティプロバイダのいずれかを呼び出して認証接続を確立し、その接続を介して安全にデータを交換できるようにするAPIです。GSS（Generic Security Services）-APIと同等であり、両者はオンザワイヤーで互換性がある。

Session Multiplex Protocol (SMP)：複数の論理的なクライアント接続が、サーバーへの単一のトランスポート接続を共有することを可能にする多重化プロトコル。**Multiple Active Result Sets (MARS)**で使用される。詳しくは[\[MC-SMP\]](#)を参照してください。

Simple and Protected GSS-API Negotiation Mechanism (SPNEGO)：Generic Security Services (GSS)のピアが、そのクレデンシャルがGSS-APIセキュリティメカニズムの共通セットをサポートしているかどうかを判断し、与えられたセキュリティメカニズム内の異なるオプションまたは複数のセキュリティメカニズムからの異なるオプションをネゴシエートし、サービスを選択し、そのサービスを使用して自分自身の間でセキュリティコンテキストを確立することができる認証メカニズムである。SPNEGOは[\[RFC4178\]](#)で規定されている。

SQLバッチ： **SQLステートメント**のセット。

SQL Server Native Client (SNAC)： SNACは、SQL Server ODBCドライバとSQL Server OLE DBプロバイダを1つのネイティブダイナミックリンクライブラリ（DLL）に含み、Microsoft SQL ServerへのネイティブコードAPI（ODBC、OLE DB、およびADO）を使用するアプリケーションをサポートします。詳細については、[\[MSDN-SNAC\]](#)を参照してください。

SQL Server User Authentication (SQLAUTH)： SQLをサポートするデータベースサーバーのユーザーアカウントをサポートするために使用される認証メカニズムです。ユーザーアカウントのユーザー名とパスワードは、クライアントがサーバーに送信するログインメッセージの一部として送信されます。

SQL文： サーバーが理解できる言語による文字列表現。

ストアードプロシージャ： SQLステートメントと、オプションでフロー制御ステートメントのブリコンパイルされたコレクションで、名前の下に格納され、ユニットとして処理されます。ストアードプロシージャはSQLデータベースに格納され、アプリケーションから1回の呼び出しで実行することができる。ストアードプロシージャは、整数の戻りコードを返し、さらに1つ以上の結果セットを返すことができます。スプロックとも呼ばれる。

テーブルレスポンス： クライアントのリクエストの結果を伝える目的で、サーバーからクライアントに送信される、すべてが特定の方法でフォーマットされたデータの集まりです。サーバーは、LOGIN7、SQL、リモートプロシージャコール（RPC）リクエストに対して、テーブルレスポンスのフォーマットで結果を返します。

TDSセッション： TDSセッション： TDS（Tabular Data Stream）プロトコルを使用したクライアントとサーバー間の一定期間の通信。

トランザクションマネージャ： アトミックトランザクションの結果を管理し、配布する責任を持つ当事者。トランザクションマネージャは、指定されたトランザクションのルートトランザクションマネージャまたは下位トランザクションマネージャのいずれかである。

ユニコード： ユニコード・コンソーシアムによって開発された文字コード規格で、世界のほぼすべての文字言語を表現している。**ユニコード規格**[\[UNICODE5.0.0/2007\]](#)では、3つの形式（UTF-8、UTF-16、UTF-32）と7つの方式（UTF-8、UTF-16、UTF-16 BE、UTF-16 LE、UTF-32、UTF-32 LE、UTF-32 BE）があります。

Virtual Interface Architecture (VIA)： サードパーティが提供する特別なハードウェアとドライバを必要とする高速インターコネクトです。

may, should, must, should not, must not: これらの用語（すべて大文字）は、[\[RFC2119\]](#)で定義されたとおりに使用されます。オプションの動作に関するすべての記述は、MAY、SHOULD、またはSHOULD NOTのいずれかを使用する。

1.2 参考文献

Microsoft Open Specificationsライブラリのドキュメントへのリンクは、参照されたドキュメントの最新公開バージョンの正しいセクションを指しています。しかし、ライブラリ内の個々の文書は同時に更新されないため、文書内のセクション番号が一致しない場合があります。正しいセクション番号は、[Errata](#)を確認することで確認できます。

1.2.1 規範となる参考文献

私たちは、継続的に利用できることを保証するために、規範となる参考文献の調査を頻繁に実施しています。規範となる参考文献を探すのに問題がある場合は、dochelp@microsoft.comまでご連絡ください。関連情報を見つけるお手伝いをさせていただきます。

[IANAPORT] IANA, "Service Name and Transport Protocol Port Number Registry",
<http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>

[IEEE754] IEEE, "IEEE Standard for Binary Floating-Point Arithmetic", IEEE 754-1985, October 1985, <http://ieeexplore.ieee.org/servlet/opac?punumber=2355>

[IETF-AuthEncr] McGrew, D., Foley, J., and Paterson, K., "Authenticated Encryption with AES-CBC and HMAC-SHA", Network Working Group Internet-Draft, July 2014, <http://tools.ietf.org/html/draft-mcgrew-aead-aes-cbc-hmac-sha2-05>

[MS-BINXML] Microsoft Corporation, "[SQL Server Binary XML Structure](#)".

[MS-LCID] Microsoft Corporation, "[Windows Language Code Identifier \(LCID\) Reference](#)".

[RFC1122] Braden, R., Ed, "Requirements for Internet Hosts -- Communication Layers", STD 3, RFC 1122, October 1989, <http://www.rfc-editor.org/rfc/rfc1122.txt>

[RFC2119] Bradner, S, "Key words for use in RFC to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119.html>

[RFC2246] Dierks, T., and Allen, C., "The TLS Protocol Version 1.0", RFC 2246, January 1999, <https://www.rfc-editor.org/info/rfc2246>

[RFC4234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005, <http://www.rfc-editor.org/rfc/rfc4234.txt>

[RFC5246] Dierks, T., and Rescorla, E., "The Transport Layer Security (TLS Protocol Version 1.2)", RFC 5246, 2008年8月, <https://www.rfc-editor.org/info/rfc5246>

[RFC6101] Freier, A., Karlton, P., and Kocher, P., "The Secure Sockets Layer (SSL Protocol Version 3.0)", RFC 6101, August 2011, <http://www.rfc-editor.org/rfc/rfc6101.txt>

[RFC6234] Eastlake III, D., and Hansen, T., "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011, <http://www.rfc-editor.org/rfc/rfc6234.txt>

[RFC7301] Friedl, S., Popov, A., Langley, A., and Stephan, E., "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension", RFC 7301, July 2014, <https://datatracker.ietf.org/doc/html/rfc7301>

[RFC793] Postel, J., Ed., "Transmission Control Protocol: DARPA Internet Program Protocol Specification", RFC 793, September 1981, <http://www.rfc-editor.org/rfc/rfc793.txt>

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>

[UNICODE] ユニコード・コンソーシアム, "The Unicode Consortium Home Page", <http://www.unicode.org/>

[VIA2002] Cameron, D., and Regnier, G., "The Virtual Interface Architecture", Intel Press, 2002, ISBN:0971288704.

1.2.2 参考文献

[MC-SMP] Microsoft Corporation, "[Session Multiplex Protocol](#)".

[MS-NETOD] Microsoft Corporation, "[Microsoft .NET Framework プロトコルの概要](#)".

[MS-SSCLRT] Microsoft Corporation, "[Microsoft SQL Server CLR Types Serialization Formats](#)".

[MSDN-Autocommit] Microsoft Corporation, "Autocommit Transactions",

[MSDN-BEGIN] Microsoft Corporation, "BEGIN TRANSACTION (Transact SQL)", <https://learn.microsoft.com/en-us/sql/t-sql/language-elements/begin-transaction-transact-sql>

[MSDN-BOUND] Microsoft Corporation, "Using Bound Sessions", [https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms177480\(v=sql.105\)](https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms177480(v=sql.105))

[MSDN-BROWSE] Microsoft Corporation, "Browse Mode", in SQL Server 2000 Retired Technical documentation, p. 12261, <https://www.microsoft.com/en-us/download/confirmation.aspx?id=51958>

[MSDN-Collation] Microsoft Corporation, "Collation and Unicode Support", <https://learn.microsoft.com/en-us/sql/relational-databases/collations/collation-and-unicode-support>

[MSDN-ColSets] Microsoft Corporation, "Use Column Sets", <https://learn.microsoft.com/en-us/sql/relational-databases/tables/use-column-sets>

[MSDN-ColSortSty] Microsoft Corporation, "Windows Collation Sorting Styles", [https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms143515\(v=sql.105\)](https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms143515(v=sql.105))

[MSDN-COMMIT] マイクロソフト株式会社、「COMMIT TRANSACTION (Transact-SQL)」、<https://learn.microsoft.com/en-us/sql/t-sql/language-elements/commit-transaction-transact-sql>

[MSDN-DTC] Microsoft Corporation, "Distributed Transaction Coordinator", [https://learn.microsoft.com/en-us/previous-versions/windows/desktop/ms684146\(v=vs.85\)](https://learn.microsoft.com/en-us/previous-versions/windows/desktop/ms684146(v=vs.85))

[MSDN-INSERT] Microsoft Corporation, "INSERT (Transact-SQL)", <https://learn.microsoft.com/en-us/sql/t-sql/statements/insert-transact-sql>

[MSDN-ITrans] Microsoft Corporation, "ITransactionExport::GetTransactionCookie", [https://learn.microsoft.com/en-us/previous-versions/windows/desktop/ms679869\(v=vs.85\)](https://learn.microsoft.com/en-us/previous-versions/windows/desktop/ms679869(v=vs.85))

[MSDN-MARS] Microsoft Corporation, "Using Multiple Active Result Sets (MARS)", <https://learn.microsoft.com/en-us/sql/relational-databases/native-client/features/using-multiple-active-result-sets-mars>

[MSDN-MDAC] Wilkes, R., Bunch, A., and Dove, D., "Microsoft Data Access Components (MDAC) Installation", May 2005, [https://learn.microsoft.com/en-us/previous-versions/ms810805\(v=msdn.10\)](https://learn.microsoft.com/en-us/previous-versions/ms810805(v=msdn.10))

[MSDN-NamedPipes] Microsoft Corporation, "Creating Valid Connection String Using Named Pipes", [https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms189307\(v=sql.105\)](https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms189307(v=sql.105))

[MSDN-NP] Microsoft Corporation, "Named Pipes", <https://learn.microsoft.com/en-us/windows/desktop/ipc/named-pipes>

[MSDN-NTLM] Microsoft Corporation, "Microsoft NTLM", <https://learn.microsoft.com/en-us/windows/desktop/SecAuthN/microsoft-ntlm>

[MSDN-QUERYNOTE] Microsoft Corporation, "Using Query Notifications", [https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms175110\(v=sql.105\)](https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms175110(v=sql.105))

[MSDN-SNAC] Microsoft Corporation, "Microsoft SQL Server Native Client and Microsoft SQL Server 2008 Native Client", <https://learn.microsoft.com/en-us/archive/blogs/sqlnativeclient/microsoft-sql-server-native-client-and-microsoft-sql-server-2008-native-client>

[MSDN-SQLCollation] Microsoft Corporation, "Selecting SQL Server Collation", [https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms144250\(v=sql.105\)](https://learn.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms144250(v=sql.105))

[MSDN-TDSENDPT] Microsoft Corporation, "Network Protocols and TDS Endpoints",

[MSDN-UPDATETEXT] マイクロソフト株式会社、「UPDATETEXT (Transact-SQL)」、<https://learn.microsoft.com/en-us/sql/t-sql/queries/updatetext-transact-sql>。

[MSDN-WRITETEXT] マイクロソフト株式会社、「WRITETEXT (Transact-SQL)」、<https://learn.microsoft.com/en-us/sql/t-sql/queries/writetext-transact-sql>。

[MSDOCS-DBMirror] Microsoft Corporation, "Database Mirroring in SQL Server", <https://learn.microsoft.com/en-us/dotnet/framework/data/adonet/sql/database-mirroring-in-sql-server>。

[RFC4120] Neuman, C., Yu, T., Hartman, S., and Raeburn, K., "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005, <https://www.rfc-editor.org/rfc/rfc4120>

[RFC4178] Zhu, L., Leach, P., Jaganathan, K., and Ingersoll, W., "The Simple and Protected Generic Security Service Application Program Interface (GSS-API) Negotiation Mechanism", RFC 4178, October 2005, <https://www.rfc-editor.org/rfc/rfc4178.txt>

[SSPI] Microsoft Corporation, "SSPI", <https://learn.microsoft.com/en-us/windows/desktop/SecAuthN/sspi>

1.3 概要

Tabular Data Stream (TDS) プロトコルバージョン7および8 (以下、「TDS」と呼ぶ) は、クライアントとデータベースサーバーシステム間のリクエストおよびレスポンスの転送に使用されるアプリケーションレベルのプロトコルである。このようなシステムでは、クライアントは通常、サーバーとの長寿命の接続を確立する。トランスポートレベルプロトコルを使用して接続が確立されると、TDSメッセージはクライアントとサーバー間の通信に使用されます。データベースサーバーも必要に応じてクライアントとして機能することができ、その場合、別のTDS接続を確立する必要がある。**TDSセッション**はトランスポートレベルセッションと直接結びついていることに注意してください。つまり、トランスポートレベルの接続が確立され、サーバーがTDS接続を確立する要求を受け取ったときに、TDSセッションが確立されます。TDSセッションは、トランスポートレベルの接続が終了するまで (例えば、TCPソケットが閉じられるまで) 存続する。さらに、TDSは使用されるトランスポートプロトコルについて何の仮定もしないが、トランスポートプロトコルがデータの信頼性のあるインオーダーデリバリーをサポートすることを仮定する。

TDSには、認証・識別、チャネル暗号化ネゴシエーション、発行のための設備が含まれています。**SQLバッチ**、**ストアドプロシージャ**呼び出し、返送データ、**トランザクションマネージャ**要求。返されるデータは、自己記述式でレコード指向である。**データストリーム**は、返される行の名前、型、およびオプションの説明を記述します。

TDS 7.xバージョンファミリーとTDS 8.0バージョンファミリーの違いは、ネットワークチャネルの

暗号化を開始する場所と方法の違いにあります：

- TDS 7.xバージョンファミリでは、暗号化はオプションであり、TDSレイヤでネゴシエーションされ処理されます。
- TDS 8.0バージョンでは、TDSが機能し始める前に下位レイヤーで処理される暗号化が必須であることが導入されています。

次の図は、TDSプロトコルにおける典型的な通信の流れ（簡略化したもの）を示している。

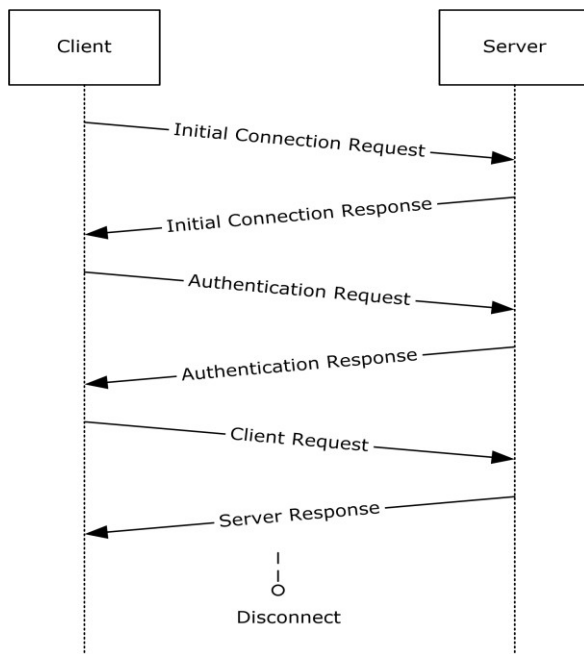


図1: TDSプロトコルの通信フロー

次の例は、**SQL文**の実行のような単純なクライアント要求を実行するために、クライアントとサーバー間で交換されるメッセージのハイレベルな説明である。クライアントとサーバーはすでに接続を確立し、認証に成功しているものとする。

クライアント: SQLステートメント

サーバーはSQL文を実行し、その結果をクライアントに返します。返されるデータ列は、まずサーバーによって記述され（列のメタデータまたはCOLMETADATAとして表現される[セクション [2.2.7.4](#)]）、次に行が続く。すべての行データが転送された後、完了メッセージが送信される。

サーバー: COLMETADATAデータストリーム
 ROWデータストリーム
 .
 .
 ROWdata stream
 DONEdata stream

データストリームとTDSパケットの相関については、[2.2.4.<1>](#)項をご参照ください。

どのSQL ServerバージョンがどのTDSバージョン番号に対応するかについての詳細は、LOGINACK（[2.2.7.14](#)項）に定義されています。

1.4 他のプロトコルとの関係

Tabular Data Stream (TDS) プロトコルは、TDS会話が発生する前にネットワークトランスポート接続が確立されていることに依存します（トランスポートプロトコルの選択は、TDSにとって重要ではありません）。

TDSは、ネットワークチャネルの暗号化のためにTLS (Transport Layer Security) /Secure Socket Layer (SSL) に依存しています。TDS 7.xバージョンファミリでは、TLS/SSLはオプションであり、クライアントとサーバー間の暗号化設定のネゴシエーションと最初のTLS/SSLハンドシェイクはTDSレイヤで処理されます。

TDS 8.0バージョンで導入されたTLSは必須であり、TDSが機能し始める前に下位レイヤーで確立される。

MARS (Multiple Active Result Sets) 機能 [\[MSDN-MARS\]](#) が有効な場合、**SMP (Session Multiplex Protocol)** [\[MC-SMP\]](#) が必要です。

この関係を図にすると次のようになります。

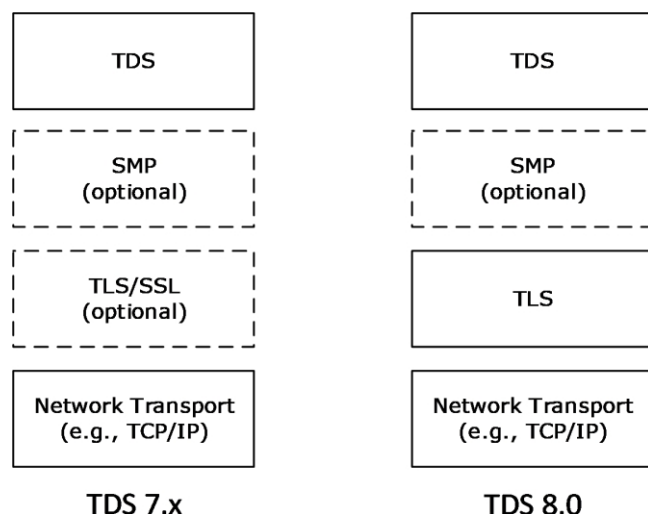


図2: プロトコルの関係

1.5 前提条件/プレコンディション

このプロトコルは、クライアントがサーバーを発見し、TDSで使用するためのネットワークトランスポート接続を確立した後に使用することができます。

TDS 7.xバージョンファミリでは、TDSが機能し始める前に下位レイヤでセキュリティアソシエーションが確立されていないことが想定されます。TDS 8.0バージョンでは、そのよう

なセキュリティアソシエーションが仮定されている。

Security Support Provider Interface (SSPI) [\[SSPI\]](#) 認証を使用するためには、クライアントマシンとサーバーマシンの両方でSSPIサポートが利用可能であることが必要である。チャンネル暗号化を使用するためには、クライアントマシンとサーバーマシンの両方にTLS/SSLサポートが存在し、サーバーマシンに暗号化に適した証明書が配備されている必要があります。

連携認証を利用するためには、サーバに連携認証サポートを提供するライブラリまたは同等のものが存在し、クライアントが連携認証のためのトークンを生成できる必要があります。

1.6 適用ステートメント

TDSプロトコルは、ネットワークまたはローカル接続が可能なすべてのシナリオにおいて、アプリケーションとデータベースサーバー間のリクエスト/レスポンス通信を促進するために使用することが適切である。

1.7 バージョニングとケイパビリティ・ネゴシエーション

このプロトコルは、以下の領域でバージョンアップの問題を含んでいます。

- **サポートされるトランスポート**このプロトコルは、[2.1](#)節で説明したように、あらゆるネットワーク・トランスポート・プロトコルの上に実装することができる。
- **プロトコルのバージョン**TDSプロトコルは、TDS 7.xバージョンファミリー（明示的バージョンTDS 7.0, TDS 7.1, TDS 7.2, TDS 7.3, TDS 7.4で構成）およびTDS 8.0 explicitバージョンに対応しています。

TDS 7.xでは、[2.2.6.4](#)節にあるように、LOGIN7メッセージデータストリームの一部として、明示的なバージョンがネゴシエートされる。

TDS 8.0では、Application-Layer Protocol Negotiation (ALPN) TLS extension [\[RFC7301\]](#)を使用して、TLSハンドシェイクから明示的バージョンを特定する必要があります。ALPNが存在しない場合、サーバーはTDS 8.0バージョンが送信されたものと判断しなければならない。フローの後半にあるLOGIN7メッセージデータで送信されるバージョン情報は、指定する必要はなく、クライアントとサーバーの両方が無視する必要がある。

TDSプロトコルの後のバージョンで、以前のバージョンに適用されない側面は、本文中で特定されている。

注記プロトコルの機能が導入された後、その機能が削除されるまで、TDSプロトコルの以降のバージョンはその機能をサポートします。

- **セキュリティと認証方法**TDSプロトコルは**SQL Server User Authentication (SQLAUTH)**をサポートしています。また、TDSプロトコルはSSPI認証をサポートし、SSPIがサポートするあらゆる認証機構を間接的にサポートする。TDSにおけるSSPIの使用は、[2.2.6.4](#)および[3.2.5.2](#)節で定義されています。TDSプロトコルは、**連携認証**もサポートする。TDSにおける連携認証の使用は、セクション2.2.6.4および[3.2.5](#)で定義されている。
- **ローカライズする：地域化依存のプロトコル動作は**、[2.2.5.1.2](#) 節及び [2.2.5.6](#) 節に規定する。
- **ケイパビリティ・ネゴシエーション (Capability Negotiation)**：このプロトコルは、こ

のセクションで規定されるように、明示的な能力ネゴシエーションを行う。

一般的に、TDSプロトコルは、サポートされる機能の完全なセットがプロトコルの各バージョンに固定されているため、能力ネゴシエーションのための設備を提供しない。認証タイプのような特定の機能は、通常ネゴシエートされず、むしろクライアントから要求される。しかし、プロトコルは以下の2つの機能のためのネゴシエーションをサポートする：

- **チャンネルの暗号化：** TDS 7.xでは、**TDSセッション**に使用される暗号化動作は、クライアントとサーバーが交換する最初のメッセージでネゴシエートされます。TDS 8.0では、暗号化は必須であり、クライアントとサーバーによる最初のメッセージの前に確立される。
- **統合された認証IDのための認証機構：** TDSセッションで使用される認証メカニズムは、クライアントとサーバーが交換する最初のメッセージでネゴシエートされる。

TDS 7.xにおける暗号化の動作、およびSSPI認証と連携認証の間のクライアントとサーバーのネゴシエーションの方法についての詳細は、[2.2.6.5](#)項のPRELOGINに関する記述を参照されたい。

TLS/SSL [\[RFC2246\]](#) [\[RFC5246\]](#) [\[RFC8446\]](#) [\[RFC6101\]](#) の暗号スイート、SSPI [\[SSPI\]](#) の認証機構、および連携認証はTDSの影響外で交渉されることに注意すること。

1.8 ベンダーエクステンシブルフィールド

なし。

1.9 スタンダード・アサインメント

TDS 7.xおよびTDS 8.0のプロトコルは、以下の割り当てを使用します。

パラメータ	TCPポート値	リファレンス
デフォルトのSQL ServerインスタンスのTCPポート	1433	[IANAPORT] (イアナポート)

また、TDS8.0では、TDSプロトコルを識別するために、以下のALPN識別シーケンスを使用します。

パラメータ	識別シーケンス	リファレンス
tds/8.0	0x74 0x64 0x73 0x2f 0x38 0x2e 0x30	[MS-TDS] です。]

注 この識別シーケンスは、要求されたものであり、Internet Assigned Numbers Authority (IANA)に登録されている最中である。登録が完了すると、この注記は削除されます。