CHAPTER 1

概要

Secure Sockets Layer（SSL）は、インターネットの暗号化技術を実現するアプリケーション層プロトコルです。SSL は、プライバシー保護、認証、およびデータ整合性を組み合せることで、クライアントとサーバ間のデータの安全な伝送を保証します。SSL は、証明書と、秘密鍵と公開鍵の鍵交換ペアを使用して、アプリケーション レベルのセキュリティを確保します。

この章の主な内容は、次のとおりです。

- SSL 暗号法の概要
- ACE の SSL の性能
- ACE の SSL 機能
- ACE SSL 設定の前提条件

SSL 暗号法の概要

Cisco 4700 Series Application Control Engine（ACE）appliance は、専用の SSL コマンド セットを使用して、クライアントとサーバ間で SSL 暗号機能を実行します。SSL 機能には、サーバ認証、秘密鍵および公開鍵の生成、証明書管理、データ パケットの暗号化および復号化が含まれます。

ACE は、SSL バージョン 3.0 と Transport Layer Security（TLS）バージョン 1.0 をサポートします。ACE は、ハイブリッド 2/3 hello メッセージとして知られる SSL バージョン 2.0 ClientHello メッセージを認識し、受け入れます。したがって、デュアルバージョン クライアントが ACE と通信できます。クライアントが SSL バージョン 2.0 ClientHello メッセージ内で SSL バージョン 3.0 を示している場合、ACE は、そのクライアントが SSL バージョン 3.0 をサポートし、バージョン 3.0 ServerHello メッセージを返信できるものとして認識します。

（注）　クライアントが SSL バージョン 2 のみをサポートする場合、ACE はネットワーク トラフィックを渡すことができません。

通常、ACE との SSL セッションには、安全な接続を確立し、維持するための暗号化方式が必要です。暗号スイートが、ACE が鍵交換、認証、および Message Authentication Code（MAC; メッセージ認証コード）を実行するために必要な暗号アルゴリズムを提供します。サポートされる暗号スイートの詳細については、第 3 章「SSL 終了の設定」の「暗号スイートの追加」を参照してください。

ここでは、ACE に実装されている SSL 暗号法の概要について説明します。次のトピックを取り上げます。

- SSL PKI
- SSL ハンドシェイク

SSL PKI

SSL は、PKI（Public Key Infrastructure; 公開鍵インフラストラクチャ）内で認証、暗号化、およびデータ整合性を提供します。PKI は、デバイス間での安全な情報交換を確立するための一連のポリシーと手順です。非対称暗号法で使用される PKI は、次の 3 つの基本要素によって特徴付けられています。

- 機密性
- 認証
- メッセージ整合性

これらの 3 つの要素は、企業イントラネットからインターネットベースの e- ビジネス アプリケーションに至るまで、e- コマースを展開するためのセキュア システムと、実質的にあらゆるタイプの電子トランザクションを構築可能な信頼できる環境を提供します。

機密性

*機密性*は、意図しないユーザからのデータ閲覧を防ぎます。PKI では、さまざまな方式でデータを暗号化することで、機密性を実現します。具体的に説明すると、SSL では、2 つのエンドポイントのみが認識している 1 つ以上の対称キーを使用して大量のデータを暗号化します。通常、対称キーは一方のエンドポイントによって生成されるので、それを他方のエンドポイントに伝送する際には安全性が重要になります。ACE は、ACE とピア間で対称キーを安全に伝送するために、*鍵交換*メカニズムの使用をサポートしています。

鍵交換では、一方のデバイスが対称キーを生成し、非対称暗号方式で暗号化してから、その鍵をピアへ伝送します。非対称暗号化では、各デバイスに公開鍵と秘密鍵で構成された 1 組の鍵ペアが必要です。2 つの鍵は数学的に関連付けられます。つまり、公開鍵を使用して暗号化されたデータだけが、対応する秘密鍵によって復号化できます。その逆も同様です。 1 つのデバイスは公開鍵をピアと共有しますが、秘密鍵は機密とします。

非対称暗号化のセキュリティは、オーナーだけが秘密鍵を知っており、どの相手側もそれを認識していないという事実に全面的に依存しています。この鍵が何らかの理由で漏洩すると、不正な Web ユーザ（または Web サイト）によって、対

称キーを含むストリームとデータ転送全体が復号化される可能性があります。もっとも一般的に使用されている鍵交換アルゴリズムが、Rivest Shamir Adelman（RSA）アルゴリズムです。

SSL の場合、受信側の秘密鍵が伝送を復号化できる唯一の鍵であることを保証するために、送信側は受信側の公開鍵を使用して対称キーを暗号化します。

認証

*認証*により、交換に関わる 1 台または複数台のデバイスは、相手のデバイスの ID を確実に確認できます。たとえば、クライアントが e- コマース Web サイトに接続しているとします。クライアントは、クレジット カード番号などの機密情報を送信する前に、サーバが正規の e- コマース Web サイトであることを確認します。クライアントとサーバの双方は、トランザクションを開始する前に相手の認証を必要とする場合があります。2 つの銀行間の金融トランザクションの場合は、クライアントとサーバの双方が必ず相手の ID を確認します。SSL では、デジタル証明書を使用することで、この認証を簡易化します。

デジタル証明書は、クライアントに対してサーバの ID を証明するための、または、任意でサーバに対してクライアントの ID を証明するためのデジタル識別情報の形式の 1 つです。証明書は、識別情報が正しいことおよび証明書に埋め込まれた公開鍵が実際にクライアントまたはサーバに所属することを保証します。

Certificate Authority（CA; 認証局）が、PKI との関係においてデジタル証明書を発行します。発行時には、セキュリティを確保するために、公開鍵および秘密鍵暗号化を使用します。CA は、信頼性を確認するために証明書に署名する信頼できる機関です。デジタル証明書には、次の情報が含まれます。

- オーナーに関する詳細情報（証明書のサブジェクト）
- CA に関する詳細情報（証明書の発行元）
- 証明書のサブジェクトの公開鍵
- 証明書の有効性と有効期限
- 証明書に関連付けられている権限

CA は証明書の発行元として、秘密鍵を使用して証明書に署名します。クライアントが証明書を受信すると、発行元の公開鍵で復号化し、証明書の署名を確認します。この手順によって、証明書が実際に権限のある機関によって発行および署名されたことを保証します。

公開鍵証明書は、*証明書階層*をサポートします。CA は、下位認証局の階層を作成し、署名付き証明書を発行する責任を分担します。階層の最上位に当たる CA は、*ルート認証局*と呼ばれています。階層の各レベルがその下位のレベルを証明することで、*証明書チェーニング*として知られる信頼関係の階層を作成します。このプロセスを使用することで、エンティティは対象の証明書を確認するために、必要に応じて、ルート認証局までさかのぼって各 CA 証明書を調べ、信頼できる階層内の CA を見つけることができます。

証明書は、期限切れになるか、または CA によって無効にされるまで有効です。CA は、証明書を無効にすると、Certificate Revocation List（CRL; 証明書失効リスト）にその証明書を追加します。CRL には、以前に発行され、現在は無効になっている証明書が一覧されています。

ACE に接続されたクライアントまたはサーバは、同じ CA または 1 つの信頼関係階層内の異なる CA（たとえば、A が B を信頼し、B が C を信頼しているため、A は C を信頼する）から発行された信頼できる証明書を保持している必要があります。

メッセージ整合性

*メッセージ整合性*は、メッセージ受信において、伝送中にメッセージの内容が改ざんされていないことを保証します。メッセージ整合性を保証するため、SSL はデータにメッセージ ダイジェストを適用してから、データを伝送します。メッセージ ダイジェストは、任意の長さのメッセージを取得し、メッセージの特性を示す固定長文字列を出力します。

メッセージ ダイジェストの重要な特性は、処理の逆行が非常に難しいことです。送信する前にメッセージそのものにメッセージのダイジェストを追加するだけでは、整合性の保証に十分ではありません。攻撃者はメッセージを変更することが可能で、それに応じてダイジェストも変更してしまいます。

ピア間で交換される各メッセージは、SHA や MD5 などのハッシュ アルゴリズムを使用して計算できる Message Authentication Code（MAC; メッセージ認証コード）によって保護されます。MAC は、複数のデータの断片のハッシュ値であり、シークレット値、送信される実際のデータ、およびシーケンス番号が含まれます。シークレット値は、書き込みセッション キーです。シーケンス番号は、32 ビット カウンタ値です。このデータは、MAC を算出するために、ハッシュ アルゴリズムによって処理されます。受信側はメッセージを受信すると、読み取り

セッション キーと予測されるシーケンス番号を使用して MAC を確認し、受信したデータを基にしてハッシュを計算します。2 つのハッシュ値が一致しない場合、データ ストリームは何らかの方法で変更されたことになります。

SSL ハンドシェイク

クライアントとサーバは、SSL ハンドシェイク プロトコルを使用して、2 つのデバイス間の SSL セッションを確立します。ハンドシェイクの実行時に、クライアントとサーバは、セキュア セッションで使用する SSL パラメータをネゴシエートします。図 1-1 に、SSL ハンドシェイク中のクライアントとサーバのアクションを示します。

（注） ACE は、SSL とその他の終端（プロキシ）された接続をアクティブ コンテキストからスタンバイ コンテキストに複製しません。

図 1-1　SSL ハンドシェイク

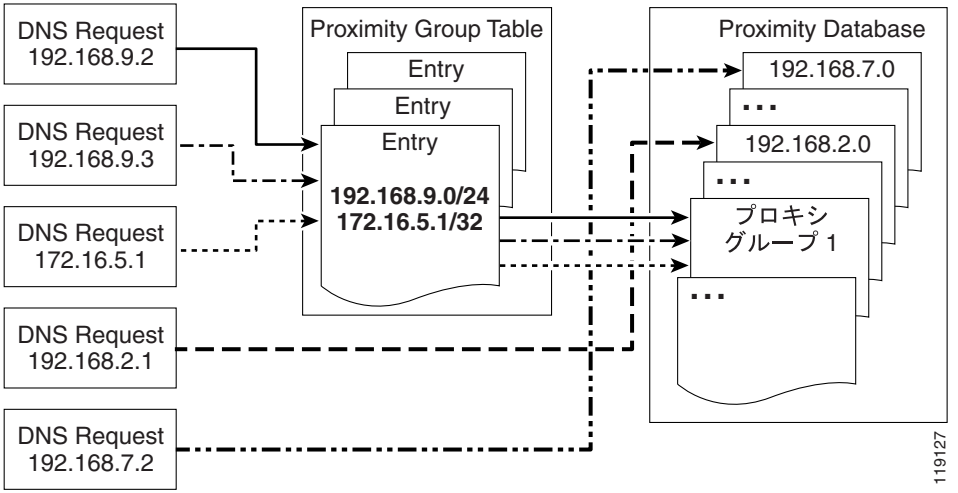


表 1-1 で、SSL ハンドシェイク中にクライアントとサーバ間で実行されるアクションについて説明します。

表 1-1　SSL ハンドシェイクにおけるアクション

ステップ	メッセージ	アクション
1	ClientHello	クライアントは、SSL セッションで使用する SSL パラメータを提示した ClientHello メッセージを送信することで、ハンドシェイクを開始します。
2	ServerHello	サーバは、SSL セッションで使用するために選択した SSL パラメータを含む ServerHello メッセージで応答します。
3	Certificate	サーバは、自身の公開鍵証明書をクライアントに送信します。
4	ServerHelloDone	サーバは、SSL ネゴシエーションの担当部分を終了します。
5	ClientKeyExchange	クライアントは、サーバの公開鍵を使用して暗号化するセッション キー情報を送信します。
6	ChangeCipherSpec	クライアントは、それ以降クライアントが送信するすべてのメッセージに対してネゴシエートした SSL パラメータをアクティブ化するようにサーバに通知します。
7	Finished	クライアントは、SSL ネゴシエーションが正常に終了したことを確認するようにサーバに通知します。
8	ChangeCipherSpec	サーバは、それ以降サーバが送信するすべてのメッセージに対してネゴシエートした SSL パラメータをアクティブ化するようにクライアントに通知します。
9	Finished	サーバは、SSL ネゴシエーションが正常に終了したことを確認するようにクライアントに通知します。

ACE の SSL の性能

表 1-2 で、ACE の SSL 仕様について説明します。

表 1-2　ACE の SSL 仕様

SSL 機能	ACE によってサポートされている機能タイプまたは仕様
SSL バージョン	• SSL バージョン 3.0 と Transport Layer Security（TLS）バージョン 1.0 • SSL バージョン 2.0 ClientHello メッセージ（ハイブリッド 2/3 hello）
公開鍵交換アルゴリズム	RSA ― 512 ビット、768 ビット、1024 ビット、1536 ビット、2048 ビット
暗号化タイプ	• Data Encryption Standard（DES; データ暗号規格） • Triple-Strength DES（3DES） • RC4 • AES
ハッシュ タイプ	• SSL MAC-MD5 • SSL MAC-SHA1
暗号スイート	• RSA_WITH_RC4_128_MD5 • RSA_WITH_RC4_128_SHA • RSA_WITH_DES_CBC_SHA • RSA_WITH_3DES_EDE_CBC_SHA • RSA_EXPORT_WITH_RC4_40_MD5 • RSA_EXPORT_WITH_DES40_CBC_SHA • RSA_EXPORT1024_WITH_RC4_56_MD5 • RSA_EXPORT1024_WITH_DES_CBC_SHA • RSA_EXPORT1024_WITH_RC4_56_SHA • RSA_WITH_AES_128_CBC_SHA • RSA_WITH_AES_256_CBC_SHA

表 1-2　ACE の SSL 仕様（続き）

SSL 機能	ACE によってサポートされている機能タイプまたは仕様
デジタル証明書	次を含む、CA から発行されたすべての主要なデジタル証明書をサポートします。 • VeriSign • Entrust • Netscape iPlanet • Windows 2000 Certificate Server • Thawte • Equifax • Genuity
最大証明書数	4096
最大証明書ファイル サイズ	無制限（フラッシュ ディスク容量を超過しない）
最大鍵ペア数	4096
最大 SSL 同時接続数	100,000
SSL スループット	1000 Mbps
SSL バルク暗号化	1 Gbps
1 秒あたりの最大 SSL トランザクション数（TPS）	デフォルトでは、ACE は 1000 SSL TPS をサポートします。オプションの SSL TPS ライセンスをインストールすると、ACE がサポートする TPS 数を最大 7,500 TPS まで増加できます。ACE のライセンス オプションの詳細については、『*Cisco 4700 Series Application Control Engine Appliance Administration Guide*』を参照してください。

ACE の SSL 機能

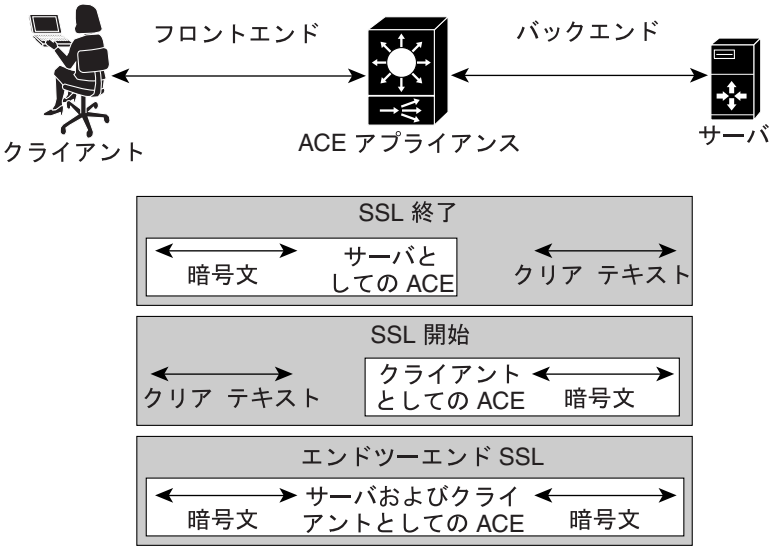
ACE は、クライアントとサーバ間でのすべてのユーザ認証、公開 / 秘密鍵の生成、証明書管理、パケット暗号化および復号化機能を実行しています。

ACE は複数のコンテキスト（仮想 ACE デバイス）に分割できます。各コンテキストに、コンテキストがピアとの SSL セッションを確立するために必要な証明書と鍵ファイルを設定します。ACE は、フラッシュ メモリ内にセキュア ストレージ領域を作成し、そこに作成された各コンテキストに関連付けられた証明書と鍵を格納します。

ACE とピア間で SSL セッションを確立し維持するために、ACE は受信するトラフィックにポリシー マップを適用します。トラフィックの特徴が特定のポリシー マップのアトリビュートと一致すると、ACE はポリシー マップに関連付けられたアクションを実行します。

ポリシー マップの定義方法によって、ACE が SSL セッションでクライアントとして動作するか、またはサーバとして動作するかを設定できます。図 1-2 に、ACE での 3 つの基本 SSL 設定を示します。これらの基本 SSL 設定では、ACE を使用してクライアントとサーバ間でデータを暗号化および復号化します。

図 1-2　ACE SSL アプリケーション



以降では 3 つの ACE SSL アプリケーションの概要を説明します。

- SSL 終了
- SSL 開始
- エンドツーエンド SSL

SSL 終了

*SSL 終了*とは、フロントエンド アプリケーション用の ACE コンテキストを設定することです。それにより、ACE がクライアントと通信する SSL サーバとして動作します。レイヤ 3 およびレイヤ 4 ポリシー マップを作成して、ACE とクライアント間のフローを定義すると、ACE は、Web ブラウザ（クライアント）と HTTP 接続（サーバ）間にセキュリティ サービスを追加することで、仮想 SSL サーバとして動作します。クライアントからのすべてのインバウンド SSL フローが ACE で終端されます。

接続が終端されたあと、ACE はクライアントからの暗号文を復号化し、データをクリア テキストとして HTTP サーバに送信します。SSL 終了用に ACE を設定する方法の詳細については、第 3 章「SSL 終了の設定」を参照してください。

SSL 開始

*SSL 開始*とは、ACE が SSL サーバと通信するクライアントとして動作するバックエンド アプリケーション用の ACE コンテキストを設定することです。レイヤ 7 ポリシー マップを作成して、ACE と SSL サーバ間のフローを定義すると、ACE はクライアントとして動作し、ACE とサーバ間の SSL セッションを開始します。SSL 開始では、ACE はクライアントからクリア テキストを受信し、その後 SSL サーバとの SSL セッションを確立して、SSL サーバ接続にクライアント接続を結合できます。ACE は、クライアントから受信したクリア テキストを暗号化し、そのデータを暗号文として SSL サーバに送信します。SSL サーバは、SSL 終了用に設定された ACE（仮想 SSL サーバ）の場合もあれば、実際の SSL サーバ（Web サーバ）の場合もあります。

SSL サーバからのアウトバウンド フローで、ACE はサーバからの暗号文を復号化し、クリア テキストをクライアントに送信します。

SSL 開始用に ACE を設定する方法の詳細については、第 4 章「SSL 開始の設定」を参照してください。

エンドツーエンド SSL

エンドツーエンド SSL とは、SSL 終了と SSL 開始の両方に対応する ACE コンテキストを設定することです。クライアント、ACE、および SSL サーバの間でセキュア SSL チャネルを確立したいアプリケーションがある場合は、ACE をエンドツーエンド SSL 対応として設定できます。たとえば、銀行間のトランザクションでは、クライアントとサーバ間で交換される金融情報を保護するためにエンドツーエンド SSL が必要です。

また、エンドツーエンド SSL を使用すると、ACE は、ロード バランシング情報とセキュリティ情報をデータに挿入できます。ACE は受信した暗号文を復号化し、ロード バランシング情報とファイアウォール情報をクリア テキストに挿入します。その後、ACE はデータを再暗号化し、暗号文を目的の宛先に渡します。

エンドツーエンド SSL 用に ACE を設定する方法の詳細については、第 5 章「エンドツーエンド SSL の設定」を参照してください。

ACE SSL 設定の前提条件

ACE に SSL 動作を設定する前に、まず ACE にサーバ ロード バランシング（SLB）を設定する必要があります。SLB 設定プロセスでは、次の設定オブジェクトを作成します。

- レイヤ 7 クラス マップ
- レイヤ 3 およびレイヤ 4 クラス マップ
- レイヤ 7 ポリシー マップ
- レイヤ 3 およびレイヤ 4 ポリシー マップ

SLB を設定したあと、このマニュアルに記載されている SSL 終了、SSL 開始、またはエンドツーエンド SSL の SSL 設定要件に従って、既存の SLB クラス マップとポリシー マップを変更してください。

ACE の SLB 設定手順については、『*Cisco 4700 Series Application Control Engine Appliance Server Load-Balancing Configuration Guide*』を参照してください。