インテル® AES-NI を使用した企業のセキュリティー保護

インテル® AES-NI（Advanced Encryption Standard New Instructions）

著 者
Leslie Xu
インテル コーポレーション

協力者
Jeffrey Casazza、Michael Kounavis、
Shihjong Kuo、Woody Cohn

2010 年 9 月
バージョン 2.0

概 要

現在、ビジネス環境でも個人の生活においても、情報およびデータを検索、処理、伝達するためのテクノロジーへの需要は、とどまることがありません。知的財産、個人の ID 情報、およびその他の機密情報を保護することは、そのデータを伝送するときも保存するときにも、かつてないほどに重要になっています。

この保護の大部分は、暗号化を通じて実現されます。暗号化とは、秘密のコードを使用する技術です。データを通常の読み取り可能な形式から理解不能な形式に変換することで、セキュリティーで保護されていない、または公衆 / 共有のチャネルを経由する際の通信の機密性を確保します。暗号化では、データの未承認の使用や変更を防止することで、権限のない第三者からデータを保護します。暗号化は従来、実行が複雑でコストがかかる点が課題となっていました。

一般的には、暗号化システムでは数学的またはアルゴリズム的プロセスを使用して、読み取り可能な平文をコード化された「暗号文」に変換し、その暗号文を平文に再度変換します。暗号化 / 復号プロセスで使用されるアルゴリズムが、暗号アルゴリズムと呼ばれます。多くの場合、暗号の処理は 1 つの鍵または鍵のセットによって制御されます。機密性、完全性、信頼性、パフォーマンスなど、さまざまな暗号化規格の多数の要素により、エンドユーザーにとっての利点が決まります。

現在の市場で、特に企業において暗号化が注目されているのはなぜでしょうか。第一に、米国では 2005 年 1 月以降、個人の機密情報が含まれる 3 億 4 千 5 百万を超えるレコードがセキュリティー侵害の被害に遭っています。[1] この被害の発生率は加速度的に増加しており、攻撃はより複雑になり、検出がさらに困難になっています。攻撃は、複数のコンピューターに対する無差別攻撃から、機密の財務情報や個人を識別できる情報が保存された、少数の価値の高い銀行システムまたは政府システムを狙った攻撃へと切り替わっています。現在の高度に仮想化されたコンピューティング環境では、複数の仮想マシンで同じハードウェア・リソースを共有しています。1 つのハードウェアにこれまでよりも多くのデータが保存されるようになっているため、より強力なセキュリティー保護が必要です。暗号化により十分強固な防御を実現することで、システムが侵害されて情報が失われたとしても、対称 / 非対称暗号化スキームを通じて、情報の利用を防ぐことができます。また、暗号化により、医療保険の携行性と責任に関する法律（HIPAA）、米国企業に適用されるサーベンスオクスリー法（SOX）、クレジットカード業界のセキュリティー基準である Payment Card Industry（PCI）への準拠でますます重要となるデータ保護も提供されます。HIPAA で暗号化が必要となるのは公衆のインターネットを介してデータを送信する場合のみですが、HiTECH 法では、データが暗号化されていない場合の情報遺漏通知に関する要件と執行権限が、広範囲にわたり HIPAA に追加されています。

目 次

よく知られている暗号化規格の 1 つに Advanced Encryption Standard (AES) があります。2001 年に米国政府で採用され、現在では、ネットワーク・トラフィック、個人データ、および企業の IT インフラストラクチャーを保護するために、ソフトウェア・エコシステム全体で幅広く使用されています。AES の用途には、セキュリティーで保護された商取引、データベースおよびストレージにおけるデータ・セキュリティー、セキュリティーで保護された仮想マシンの移行、フルディスク暗号化などがあります。IDC の暗号化の使用状況に関する調査によれば、AES は企業のデータベースとアーカイブ用のバックアップで最も広く使用されています。[2] また、フルディスク暗号化も大きな注目を集めています。

インテル® AES-NI (Advanced Encryption Standard New Instructions) は、インテル® Xeon® プロセッサー 5600 番台に搭載された新しい 7 つの命令のセットです。4 つの命令が暗号化と復号を高速化し、2 つの命令がキー生成とマトリクス操作を向上させ、7 つ目の命令がキャリーなし乗算を支援します。AES アルゴリズムの複雑で負荷の大きなサブステップのいくつかをハードウェアに搭載することで、インテル® AES-NI は AES ベースの暗号化を高速化します。その結果、より高速で安全な暗号化が実現され、以前は暗号化の利用が適さなかった場所でも利用できるようになります。

このホワイトペーパーでは、AES とインテル® AES-NI について詳しく説明し、続いて 3 つの利用モデル、パフォーマンス向上の可能性、および暗号化ライブラリーについて説明します。ソフトウェア・ベンダーは、この暗号化ライブラリーを使用して、基本的な AES ルーチンをインテル® AES-NI に最適化された処理で置き換えることができます。

2. はじめに

IDC の暗号化の使用状況に関する調査によれば、2005 年以降、9 千万人を超える消費者が、個人情報に関するセキュリティー侵害の可能性についての通知を受けています。[3] 朝、ノートブック PC の電源を入れると、WiFi* ネットワークのプロパティー・ウィンドウに AES-CCMP データ暗号化と企業の認証が表示されます。イントラネット上で作業したり、セキュリティーで保護された Web サイトで買い物をすると、ロックアイコンがブラウザーに表示されます。これは、Secure Socket Layer (SSL) によって実現されるセキュリティー保護された接続であることを示しています。SSL は暗号化プロトコルであり、インターネットなどのネットワークを介した通信でのセキュリティーとデータ完全性を確保します。SSL と、より新しい Transport Layer Security (TLS) プロトコルにより、ネットワーク接続セグメントがエンドツーエンドで暗号化されます。

確かに、個人 (クライアント) レベルでは、私たちは日々セキュリティーに囲まれていますが、企業や企業のサーバーについてはどうでしょうか。政府機関や企業のサーバーの多くに個人を識別できる情報や財務情報が大量に格納されており、要求に応じてクライアントに配信されています。このことから、サーバーレベルの暗号化が非常に重要になっています。これは、特に、あらゆる種類のコンピューターの攻撃に使用される悪意あるコード (マルウェア) の増加率が高まり続けているためです。

同様に懸念されるのは、コンピューター攻撃がより複雑になり、検出がさらに困難になっていることです。その上、攻撃を開始する人物のタイプが変化してきています。以前は、技術的スキルを誇示し、感染させたコンピューターの数を競うことで有名になりたがる攻撃者が大半でしたが、今は、金銭を目的とする攻撃者や、組織犯罪に関わる攻撃者に取って代わられています。彼らの目的は必ずしも多数のコンピューターを感染させることではなく、少数の高価値のターゲットを密かに感染させることにあります。これらのターゲットとしては、財務情報や個人情報にアクセスできる銀行や公共機関が考えられます。こうした状況の中、暗号化は、優れた防御を提供する最後の手段です。システムが侵害され、情報が奪われたとしても、暗号化によりそれらの情報を利用できないようにすることが可能です。

ノートブック PC のハードディスク・ドライブに

保存されている政府の機密データか、顧客の社会保障番号に関連するデータセンターのセキュリティー侵害かにかかわらず、事件の急増とその影響により、機密情報に暗号化テクノロジーの使用を命ずる法律の制定に取り組む州が増えています。[4] 多くの政府機関は、セキュリティー侵害の公表を要求しており、公表を要求する連邦法も提案されています。産業界でも、セキュリティー手順の監視を強化しています。多くの場合、暗号化によって、HIPAA、SOX、PCI およびその他の規制に準拠し続けるうえで、ますます重要なデータ保護が実現されます。機密データを保護することで顧客からの信頼と忠誠心が高まり、法的な負担が軽減され、データ・セキュリティーの規制要件を満たすことができるようになります。

以下に、暗号化を要求している、または強く奨励している規制の例と、準拠しない場合のペナルティーを示します。

HIPAA 4 医療保険の携行性と責任に関する法律（Health Insurance Portability and Accountability Act）、1996 年：

- 患者の保健関連情報が保管される情報システムを侵入から保護する必要があります。
- 公衆のネットワークを介して情報をやり取りする場合は、何らかの形式の暗号化を利用する必要があります。
- 準拠しなかった場合に課せられる可能性があるペナルティー：10 年間の懲役および 1 事件につき $100 の罰金（1 年につき最大 $25,000）。

HiTECH 法、2010 年 2 月 17 日：

- HIPAA のプライバシーとセキュリティーに関する条項の適用範囲が仕事上の関係者（BA）にまで拡大されました。
- 米国では、1 つのセキュリティー侵害に 500 名を超える個人が巻き込まれている場合、対象組織は必ずメディアに広く通知しなければなりません。
- 個人の保健関連情報が「使用不能、読み取り不能、解読不能」な状態になっている場合は、通知の必要はありません。
- 実質的な損害が発生する大規模なセキュリティー侵害は、集団訴訟、規制措置、株価の下落、評判の低下につながり、顧客との関係を損なう場合があります。
- 保存および伝送時の「使用不能、読み取り不能、解読不能」なデータの暗号化規格が規定されています。

サーベンスオクスリー法 7（SOX）：

- 米国のすべての株式上場企業は、財務情報の完全性と機密性を保護するための厳格な情報テクノロジー・ガイドラインに準拠する必要があります。
- ISO/IEC 27002（情報セキュリティー標準）では、SOX 関連のセキュリティー管理におけるベスト・プラクティスが定義されており、暗号化の使用が明示的に提起されています。
- 準拠しなかった場合に課せられる可能性があるペナルティー：10 年間の懲役および $15,000,000 の罰金。

クレジットカード業界のセキュリティー基準（Payment Card Industry（PCI）Data Security Standard）：

- プライマリー・アカウント番号（PAN）またはクレジットカード番号（もしくは両方）は、保存時には暗号化する必要があります。
- 組織のメンバーは、Visa、MasterCard、American Express、Discover などです。
- メンバーは、準拠していない機関からのクレジットカード決済の受け入れを停止する権限を持ち、$500,000 の罰金を課すことができます。

機密データの保護はますます必須のものになりつつあり、暗号化と暗号化テクノロジーを使用する必要性もますます高まっています。

2.1. 定義

以下に、このホワイトペーパーをより深く理解するために必要な、いくつかの定義を示します。

AES： Advanced Encryption Standard。AES は、Rijndael アルゴリズムを若干変更したバージョンであり、2001 年に米国政府によって採用された暗号化規格です。AES は、古くて安全性の低い 3DES（112 または 168 ビット・キー長）に取って代わりつつあります。AES はブロック暗号です。つまり、ブロックと呼ばれる固定長のビットグループに対して動作します。

アルゴリズム：タスクを実行するための順序付けられたステップ（数式）のセット。暗号化では、このタスクは、通常のデータを秘密の文字に変換すること（暗号化）と、秘密の文字を読み取り可能なデータに変換し直すこと（復号）です。

非対称暗号化アルゴリズム：暗号化と復号に異なる鍵を使用するスキーム。非対称鍵ペアの一方の鍵を知っているユーザーはデータを暗号化することはできますが、その鍵を使用して、暗号化されたデータを復号することはできません。これは、同じ鍵を使用してデータの暗号化と復号を行うシステムとは別の暗号化方式であり、ほとんどの場合、対称暗号化鍵などの少量のデータを暗号化するために使用されます。

暗号：プライベートな情報交換を維持するために、2 人以上の当事者によって交換される情報を暗号化または復号するためのアルゴリズム。

暗号文：アルゴリズムによって暗号化形式に変換されたデータ。暗号文は、復号されて元のデータ形式に戻されるまで読み取ることはできません。

復号鍵：メッセージを復号する鍵。対称鍵では同じキーを使用して暗号化と復号を行いますが、非対称鍵では異なります。

暗号化鍵：メッセージを暗号化する鍵。

ハッシュ関数：データブロックを固定サイズのビットの文字列（ハッシュ値）に変換する適切に定義されたプロシージャーまたは数学関数であり、データが変更されるとそのハッシュ値も変更されます。エンコードされるデータは「メッセージ」と呼ばれます。多くの場合、ハッシュ値は「ダイジェスト」と呼ばれます。ほとんどの暗号化ハッシュ関数では、任意の長さの文字列を受け取り、固定長のハッシュ値を生成します。

Internet Protocol security（IPsec）：暗号化セキュリティー・サービスを使用して、インターネット・プロトコル（IP）ネットワーク経由の通信を保護するためのオープン・スタンダードのフレームワーク（エンドツーエンドのイーサネット接続では、暗号化ソフトウェアの形で、オープン・システム・インターコネクション（OSI）第 3 層に実装されます）。

MAC：メッセージ認証コード。メッセージの認証に使用される一片の情報。多くの場合、MAC アルゴリズムは鍵付き（暗号化）ハッシュ関数と呼ばれます。MAC がデジタル署名とは異なるのは、MAC 値は同じ秘密鍵を使用して生成および検証されるという点です。このためには、メッセージの送信者と受信者が、通信を開始する前に、同じ鍵について合意する必要があります。これは、対称暗号化に似ています。

NIC：ネットワーク・インターフェイス・コントローラー。イーサネット・アダプターとも呼ばれます。NIC には、トラフィックを処理し、プロセッサーのインテル® AES-NI に IPsec トラフィックをオフロードするためのハードウェアが含まれている場合があります。

平文：元のメッセージまたは元のファイル。これは、ファイルまたはメッセージが暗号化され、その後復号された後に取得されるものです。

RSA：Rivest-Shamir-Adleman 暗号化アルゴリズム。非対称暗号化アルゴリズムです（暗号化と復号が単一の鍵で実行される AES とは異なり、公開鍵と秘密鍵を使用します）。RSA 中に交換されたプレマスター・シークレットによりプレマスター鍵が生成され、この鍵から鍵付きハッシュ認証コード（HMAC）鍵と AES 鍵の両方が動的に導き出されます。

SHA：セキュア・ハッシュ・アルゴリズム。暗号化ハッシュ関数のセット。SHA は、国家安全保障局（NSA）によって設計され、米国標準技術局（NIST）によって米国連邦情報処理規格として公開されました（NIST は米国連邦政府機関であり、民間部門および公共部門向けの規格とガイドラインを制定しています）。SHA-1 では、最大（264 − 1）ビット長のメッセージから、160 ビットのダイジェストが生成されます。

SSL：Secure Socket Layer。セキュリティーで保護されたネットワーク通信を実現する暗号化プロトコル。OSI の第 6 層（プレゼンテーション層）に実装され、Web アプリケーションレベルのセキュリティー保護されたトランザクションで使用されます。

ソフトウェア・サイドチャネル攻撃：総当たり攻撃やアルゴリズムの理論上の弱点を突く攻撃ではなく、暗号化の物理的な実装から得られた情報に基づく攻撃。通常、この攻撃では、タイミング、熱、ノイズ、キャッシュの内容、電磁波、消費電力など、暗号化処理に関して間接的に収集された情報を使用して、探索する必要のあるキースペースを狭め、鍵を識別してシステムに侵入します。

対称暗号化アルゴリズム：密接に関連した（多くの場合は同一の）暗号化鍵を暗号化と復号の両方に使用するスキーム。通常、非対称鍵アルゴリズムでは、対称鍵アルゴリズムの数百倍から数千倍の時間がかかります。

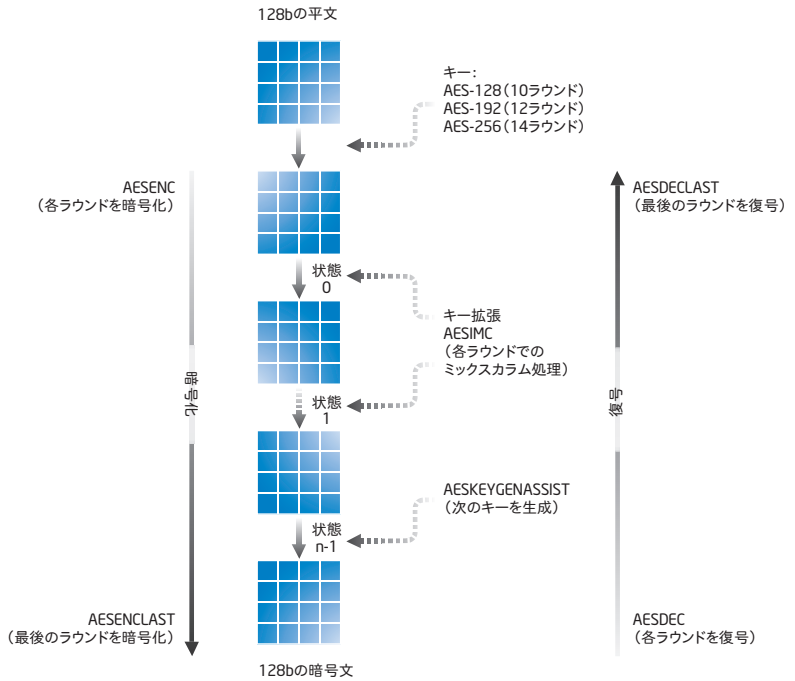

図 1. インテル® AES-NI による AES アルゴリズムのサブステップの高速化

TPM： Trusted Platform Module。特別に設計されたチップであり、コンピューターのシステムボードに取り付けて、暗号関数を追加し、暗号計算をサポートできます。サポートするファームウェアおよびソフトウェアと連携し、システムへの権限のないアクセスを防止します。TPM には、最大 2048 ビットの RSA 暗号化 / 復号のみを実行するハードウェア・エンジンが備わっています。TPM は、デジタル署名およびキーラッピング処理時に、その組込みの RSA エンジンを使用します。

3. Advanced Encryption Standard (AES)

AES は、Rijindael アルゴリズムを若干変更したバージョンであり、2001 年に米国政府に採用された暗号化規格です。AES は、古くて安全性の低い 3DES (112 または 168 ビット・キー長) に取って代わりつつあります。AES はブロック暗号です。つまり、ブロックと呼ばれる固定長のビットグループに対して動作します。

図 1 は、AES アルゴリズムの流れを示しています。AES では、128 ビットの固定された 4x4 ブロックサイズと、可変鍵長が使用されます。鍵長 (128、192、または 256 ビット) に応じて、最終的な暗号文を生成するために 10、12、または 14 ラウンドの変換が必要になります。元の平文 (4x4 バイト) は、中間結果または中間状態を生成する最初の暗号ラウンドに固有のキーを使用して暗号化されます。結果として生成される中間暗号文は、2 つ目のラウンドキーがある次のラウンドに送られます。定義されている数のラウンドが完了するまでこの処理が続きます (図 1 を参照)。最後のラウンドは、可逆線形変換 (固定多項式の乗算を通じて達成される) を使用して各カラムの 4 バイトが結合されるミックスカラム処理がない点を除き、それまでのラウンドと同じです。

AES には、数種類の処理モードがあります。ブロック暗号の処理モードについては、NIST が、電子コードブック (ECB)、暗号ブロック連鎖 (CBC)、カウンター (CTR)、暗号フィードバック (CFB)、出力フィードバック (OFB) を推奨しています。最も基本的な AES モードの ECB は、単純で並列ですが、統計的攻撃 (平文 / 暗号文のペアに適用されている統計的手法を利用してランダム置換から暗号の一部を見分けるタイプの攻撃) に対して脆弱であるため、安全とは見なされません。このため、CBC が最も一般的に使用されている処理モードです。CBC では事実上、順次処理が行われます。CTR は、一般的な並列モードですが、認証は備わっていません。ガロア・カウンター・モード (GCM) では、CTR と認証タグ (セキュア・ハッシュ・アルゴリズム (SHA) の代替) が組み合わせられます。

AES 暗号化規格の詳細については、FIPS PUB 197 を参照してください。[5]

4. インテル® AES-NI について

インテル® AES-NI (Advanced Encryption Standard New Instructions) は、インテル® Xeon® プロセッサー 5600 番台に搭載された新しい命令セットです。インテル® AES-NI では、AES アルゴリズムのサブステップのいくつかがハードウェアに搭載されます。これにより、AES 暗号化 / 復号アルゴリズムの実行が高速化され、暗号化を使用してデータを保護する際の難点の 1 つであるパフォーマンスの低下が解消されます。

実際には、インテル® Xeon® プロセッサー 5600 番台に AES のアプリケーション全体が搭載されているのではなく、一部だけが高速化されます。このことは、法的分類のために重要です。これは、暗号化が多くの国において管理されたテクノロジーであるためです。インテル® AES-NI では、6 つの新しい AES 命令 (暗号化 / 復号用× 4、ミックスカラム用× 1、次のラウンドテキストの生成用× 1) が追加されています。これらの命令は、変換ラウンドでの AES 処理を高速化し、ラウンドキーの生成を支援します。

インテル® AES-NI には、7 つ目の新しい命令である CLMUL も含まれています。この命令は、AES-GCM と 2 進楕円曲線暗号 (ECC) の処理を高速化し、誤り訂正符号、汎用の巡回冗長検査 (CRC)、およびデータ重複除外を支援します。この命令は、キャリーなし乗算 (「2 進多項式乗算」とも呼ばれる) で特に役立ちます。これは、キャリーを生成または伝播せずに、2 つのオペランドの積を計算する数学的処理です。このような乗算は、2 元ガロア体における乗算の計算に不可欠な手順です。インテル® AES-NI には、インテルのキャリーなし乗算命令が含まれています。アルゴリズムは、CLMUL を使用して、GCM の基になる演算処理であるガロアハッシュを計算します。CLMUL は、2 つの 64 ビット・オペランドについてキャリーなし乗算を行うことで、GCM の実行を高速化します。

改めて図 1 を見ると、新しい命令のうちの 4 つが働いて、128 ビットの平文から 128 ビットの暗号文への暗号化と、その逆の復号を実行していることがわかります。各ラウンドでは、2 つの命令が後続の鍵の生成を支援しています。AESENC 命令が各ラウンドを暗号化し、AESENCLAST が最後のラウンドを暗号化します。逆 (復号) では、AESDEC が各ラウンドを復号し、AESDECLAST が最後のラウンドを復号します。もう 1 つの命令 AESIMC は、各ラウンドでミックスカラム処理を実行し、AESKEYGENASSIST は次の鍵を生成します。鍵は、128、192、または 256 ビットを選択できます。これらすべての計算がハードウェアによって実行されるため、大幅な高速化が実現されます。その速度は、順次処理モードの CBC 暗号化で 4 倍、並列処理モードでは 14 倍を超えます (詳細については 6.1 節を参照)。

インテル® AES-NI のパフォーマンス上の利点に加えて、ハードウェアで命令を実行することで、ソフトウェア・サイドチャネル攻撃を防止する際のセキュリティーが強化されます。ソフトウェア・サイドチャネルは、暗号化アルゴリズムのソフトウェア実装における脆弱性です。これは、複数の処理環境 (複数のコア、スレッド、またはオペレーティング・システム) に存在します。キャッシュベースのソフトウェア・サイドチャネル攻撃は、ソフトウェア・ベースの AES が暗号化ブロック、鍵、およびルックアップ・テーブルをメモリーに保持しているという事実を悪用します。キャッシュ衝突タイミング・サイドチャネル攻撃では、プラットフォーム上で動作する一片の悪意のあるコードがキャッシュに侵入し、暗号化処理を実行して、特定のメモリーアクセスのタイミングを測って、キャッシュ内の変更内容を特定します。これらの変更内容から、暗号化鍵の値の一部が特定される場

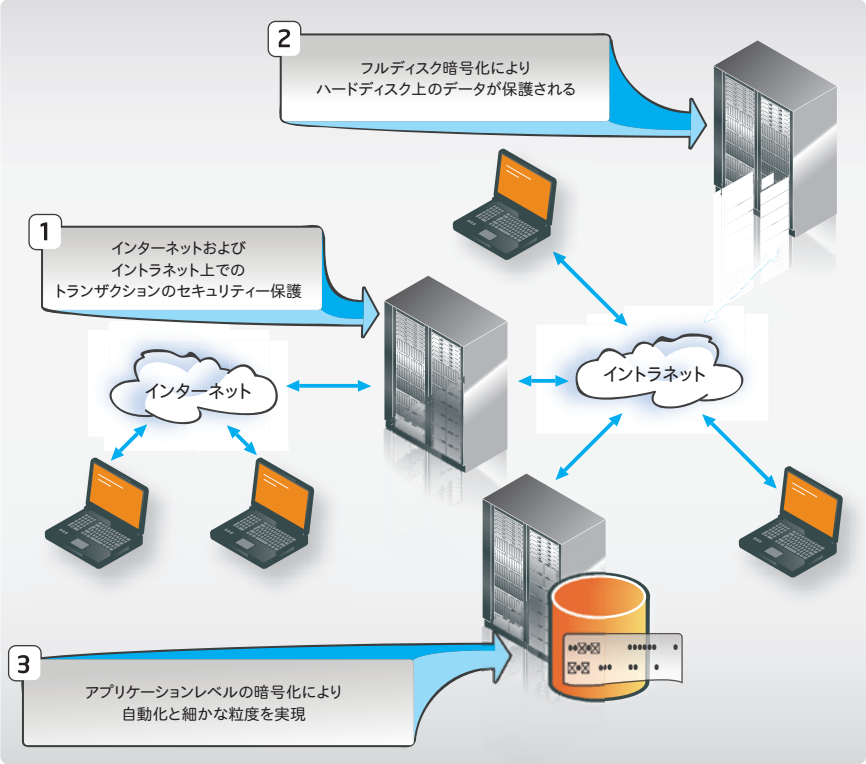

図 2. 3 つの AES 利用モデル

合があります。たとえば、ある暗号化処理にかかる時間を測定することで、攻撃者が鍵の最上位のビットが「0」であると判断できる可能性があります。その 1 つのビットが判明することで、完全な鍵値を特定するために探索するべきキースペースが半分に狭まります。さらに効率的なサイドチャネル攻撃では、キースペースが大幅に狭まります（たとえば、鍵の半分のビットが特定される場合があります）。

インテル® AES-NI はハードウェアベースであるため、ルックアップ・テーブルは必要なく、暗号化ブロックはマイクロプロセッサー内のハードウェアで実行されます。これにより、インテル® AES-NI を使用してソフトウェア・サイドチャネル攻撃に対処する AES の実装が可能になります。[6] さらに、これらの命令により、AES を実装しやすくなり、コードサイズが縮小されます。これによって、検出しにくいサイドチャネルからの漏洩などのセキュリティー上の欠陥が見落とされるリスクを軽減できます。さらに、インテル® AES-NI によって実現される高速化により、システムで大きなサイズのキーを実行できるようになるため、データ転送がさらに安全になります。

インテル® Xeon® プロセッサー 5600 番台搭載サーバーおよび新しい命令に最適化されたソフトウェアを購入すると、インテル® AES-NI のパフォーマンスと長所を活用することができます。

AES の新しい命令の詳細については、http://software.intel.com/file/24917/（英語）を参照してください。

CLMUL 命令とそのキャリーなし乗算の処理の詳細については、http://software.intel.com/en-us/articles/intel-carry-less-multiplication-instruction-and-its-usage-for-computing-the-gcm-mode/（英語）を参照してください。

5. AES の利用モデル

この章では、インテル® AES-NI の 3 つの主要な利用モデルである、ネットワークの暗号化、フルディスク暗号化（FDE）、アプリケーションレベルの暗号化について説明します（図 2 を参照）。

ネットワーキング・アプリケーションでは、暗号化を使用して、SSL、TLS、IPsec、HTTPS、FTP、SSH などのプロトコルで伝送される

データを保護します。この章では、HTTPS と IPsec の他に、FDE と、暗号化を使用して保存時のデータを保護するアプリケーションレベルのモデルに焦点を当てます。

これら 3 つのモデルすべてにおいて、インテル® AES-NI を使用することでパフォーマンスの向上が実現されます。このようなパフォーマンスの向上により、これまでパフォーマンスへの影響が原因で実現不可能だった場所でも暗号化を利用できるようになります。

5.1. セキュリティー保護されたトランザクション

現在の高度にネットワーク化された世界では、Web サーバー、アプリケーション・サーバー、およびデータベース・バックエンドのすべてが IP ネットワークを介し、ゲートウェイおよびアプライアンス経由で接続されています。通常、SSL は、ネットワーク経由でのセキュリティー保護されたトランザクションを実現するために使用されます。SSL は、銀行取引などの電子商取引、および企業通信（イントラネットなど）にセキュリティー保護された処理を提供することでよく知られています。

図 3 に示すセキュリティー保護されたトランザクションでは、まずユーザーが、https:// で始まる URL を使用して Web ページにアクセスします。HTTPS は、ハイパーテキスト転送プロトコル（HTTP）と暗号化プロトコルを組み合わせることで、セキュリティー保護されていないネットワーク経由でセキュリティー保護されたチャネルを作成します。HTTP は、最上位層（OSI プロトコル標準のアプリケーション層）で動作しますが、HTTPS では、プレゼンテーション層の SSL プロトコルが利用されます。SSL をサポートする導入済みの新しいサーバーのほとんどは、暗号の選択肢として AES もサポートします。ただし、HTTPS を使用したブラウザー・トランザクションについては、暗号スイートの選択肢は、ブラウザーまたはサーバーのサポートではなく、クライアントのオペレーティング・システムによって強く影響されます。AES は、Microsoft* Windows* 7 と Windows Vista* および Linux* オペレーティング・システムのほとんどで採用されています。Microsoft* Windows* XP インストール・ベースの大部分では、古くて安全性の低い RC4-MD5 暗号スイートが SSL トランザクションに使用されています。サーバーは、サーバーとクライアントの両方がサポートしている最も強力な暗号スイートとハッシュ関数を選択し、クライアントにその決定内容を通知します。

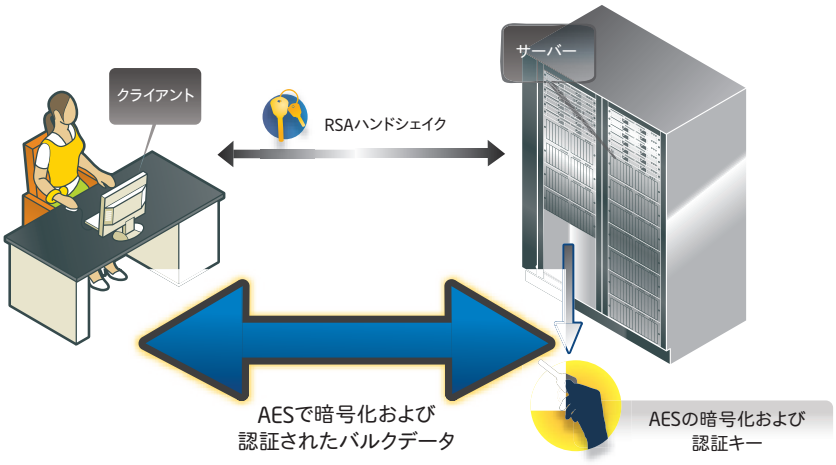

図 3. SSL を使用したトランザクションのセキュリティー保護

クライアントがサーバーと接続しトランザクションを開始すると（図 3 を参照）、RSA 暗号化アルゴリズムを使用したクライアントとサーバー間のハンドシェイクにより SSL トランザクションが開始されます。RSA は、サーバーからクライアントに公開キーを送信するように要求し、その後で、クライアントからサーバーにプレマスター・シークレットが返送され、サーバーで復号されます。RSA は非対称アルゴリズムであるため、クライアントは別の鍵を使用してデータを復号する必要があります。AES は、RSA ハンドシェイクと似たハンドシェイク認証を使用しますが、これは対称アルゴリズムであるため同じ鍵で暗号化と復号が行われます。AES ハンドシェイクが完了すると、AES を介して認証されたデータのバルク交換が開始されます。

インテル® AES-NI の真価は、AES アルゴリズムを使用するそれらの SSL トランザクションでの計算の影響（負荷）を軽減するときに発揮されます。セキュリティー保護された通信の確立では大きなオーバーヘッドが発生します。そして、このオーバーヘッドは、サーバーとのセキュリティー保護された通信を同時に確立しようとするシステムの数に応じて、数百または数千倍になる場合があります。休暇シーズン中の、お気に入りのオンライン・ショッピング・サイトを思い浮かべてみてください。インテル® AES-NI を統合することで、これらのセキュリティー保護されたトランザクションすべての計算による影響が軽減され、パフォーマンスが向上します。

5.1.1. クラウドにおける HTTPS

HTTPS は、一般的に World Wide Web での支払取引や、企業の情報システムでの機密情報のやり取りに使用されますが、イントラネットおよびインターネットを介した電子メール、ポータル、コラボレーション・ソフトウェアでも広く使用されています。Google* Apps や Windows Live* などのクラウドサービスが普及するにつれ、セキュリティー保護された HTTPS 接続がますます注目を集め、使用されるようになっています。

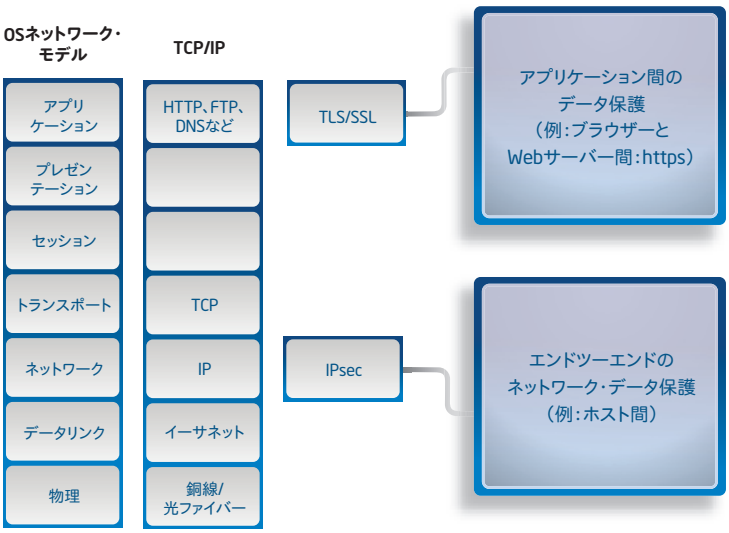

図 4. OSI スタックにおける IPsec

クラウドサービスの発展により、膨大な量のユーザーデータが Web 上に置かれるようになっています。公共または民間のクラウド運営者は、ユーザーを保護するために、各個人のデータがクライアントとクラウド間を移動する際のプライバシーと機密性を確保する必要があります。これは、膨大な数のサービスおよびアクセスポイントのすべてにわたってセキュリティー・インフラストラクチャーを確立することを意味します。そのため、クラウドが急増するにつれ、HTTPS 接続とともに、暗号化、送信、および復号されるデータの量が増加すると予測されます。[7]

クラウド・プロバイダーにとって、クラウドを介したトランザクション、コンテンツのストリーミング、およびコラボレーション・セッションのパフォーマンスと応答能力のすべてが、顧客満足の実現のために不可欠です。ただし、クラウドサービスに引き付けられる契約者の数が増えるほど、サーバーにかかる負担は大きくなります。このことから、少しずつのパフォーマンス向上であっても、すべての場所で得られる向上が非常に重要になります。インテル® AES-NI と、暗号化 / 復号のパフォーマンスを向上するその能力は、クラウド・コンピューティングの潮流がユーザー体験を向上し、セキュリティー保護されたデータ交換の高速化を支援するうえで、重要な役割を果たすことができます。

5.1.2. Internet Protocol Security（IPsec）

図 4 に示すように、通常、SSL は、OSI の第 7 層において Web アプリケーションとサービス間で使用されますが、IPsec 接続では第 3 層の既知のピアが使用されます。

選択された一連のポートと暗号化 / 復号の固有のポリシーは、ソケット・ライブラリー（クライアントが TCP/IP 接続を設定してサーバーと直接通信できる、プログラマーの低レベルのインターフェイス）または OS ライブラリーを使用して実施されます。

インテル® 82599 10 ギガビット・イーサネット・コントローラーやインテル® 82576 ギガビット・イーサネット・コントローラーなどのイーサネット NIC に IPsec をオフロードして、処理を高速化することもできます。2 つから 3 つの接続しか必要ない場合は、NIC へのオフロードが IPsec の推奨されるソリューションである可能性が高くなります。多数の接続がある場合は、（それらの周辺にある既存のオフロードエンジンの数には限度があると考えると）インテル® AES-NI がより適切なソリューションになります。つまり、NIC にオフロードするか、OS の暗号化ライブラリーに迂回させてインテル® AES-NI で高速処理するという、混合のアプローチを使用します。

IPsec を使用する場合の一般的なシナリオは、会社のメインオフィスとリモートオフィスの間のトラフィックを保護することです。リモートオフィスのトラフィックは 12 ～ 15 の接続であり、一方、会社のメインオフィスの接続は 12,000 の接続であるとします。リモートオフィスでは、純粋なオフロードアプローチが妥当です。会社のメインオフィスでは、混合のアプローチによってパフォーマンスが向上します。AES の暗号化 / 復号を利用する場合は、インテル® Xeon® プロセッサー 5600 番台搭載サーバーおよびインテル® AES-NI に最適化されたソフトウェアを導入すると、暗号化 / 復号の実行手順の一部がハードウェアに移され、パフォーマンスとセキュリティーがさらに向上します。

5.2. 基幹業務アプリケーション

ほとんどの基幹業務アプリケーションには、暗号化を使用して情報をセキュリティー保護するための何らかのオプションが用意されています。これは、電子メール、コラボレーション、ポータルのアプリケーションで使用される一般的なオプションです。ERP および CRM アプリケーションにも、データベース・バックエンドのアーキテクチャーに暗号化が用意されています。暗号化オプションが用意されているデータベースの例として は、Oracle* Database、IBM* DB2*、Microsoft* SQL Server*、Microsoft* Access*、MySQL* があります。データベースの暗号化では、データ・セル・レベル、カラムレベル、ファイル・システム・レベル、テーブルスペースおよびデータベース・レベルでの細かな粒度と柔軟性が実現されています。透過的データ暗号化（TDE）は、一部のデータベース（Oracle* Database 10g R2 および 11g[8] と Microsoft* SQL Server* 2008[9]）に備わっている機能であり、データがディスクに格納されるときには暗号化を、そして再度メモリーに読み取られるときには復号を自動的に実行します。小売業者は、TDE のような機能を使用して、PCI-DSS の要件に対応できます。大学や医療組織は、この機能を使用してデータを自動的に保護し、ディスク

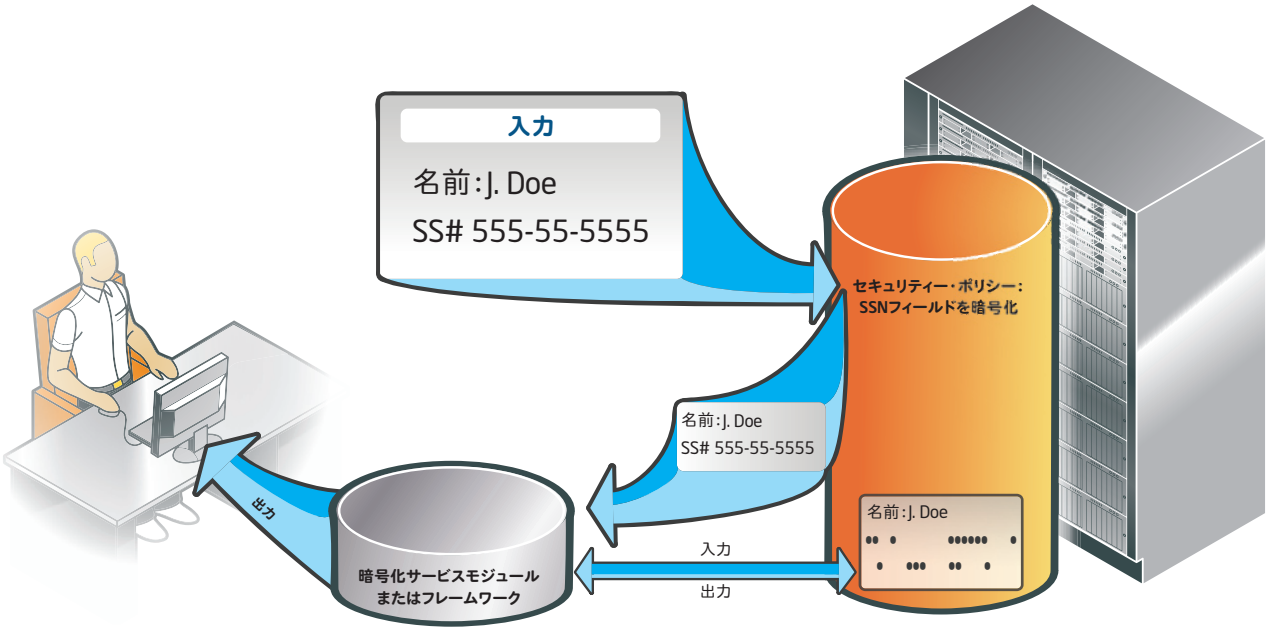

図 5. アプリケーションレベルの暗号化における AES 利用モデル

ドライブやバックアップ・メディア上の社会保障番号などの機密情報を権限のないアクセスから保護できます。AES は、ほとんどの基幹業務アプリケーションの暗号化スキームでサポートされているアルゴリズムであるため、インテル® AES-NI を使用することで、アプリケーションを高速化し、セキュリティーを強化する絶好の機会が得られます(図 5 を参照)。

5.3. フルディスク暗号化(FDE)

図 6 に示す FDE では、ディスク暗号化ソフトウェアを使用して、ディスクまたはディスクボリュームに送られるすべてのデータを暗号化します。多くの場合、FDE という用語は、OS パーティションをブートするプログラムを含め、ディスク上のすべてを暗号化することを意味しますが、マスター・ブート・レコード(MBR)は暗号化されないため、ディスクのこのわずかな部分は暗号化されずに残ります。FDE は、ディスク暗号化ソフトウェアまたは暗号化されたハードディスク・ドライブを通じて実装できます。

通常、直接接続ストレージ(DAS)は、サーバー・エンクロージャー内の 1 つ以上の Serial Attached SCSI(SAS)または SATA ハードディスク・ドライブに接続されます。ハードディスクおよびインターコネクトは比較的少ないため、有効帯域幅は相対的に小さめです。このため、通常は、DAS の帯域幅要件に適合する速度で、ホスト・プロセッサーがソフトウェアでデータを暗号化することが妥当です。

フルディスク暗号化は、セキュリティー対策として広まりつつあります。Windows Server* 2008 の Windows* BitLocker* ドライブ暗号化、PGPdisk、McAfee* Total Protection for Endpoint などの製品は、ハードディスクに保存されているデータの暗号化を提供します。フルディスク暗号化は、紛失および盗難からデータを保護するだけでなく、廃棄と修復も支援します。たとえば、破損したハードディスク・ドライブに暗号化されていない機密情報が保存されている場合、保障修理に出すと、そのデータが人目に

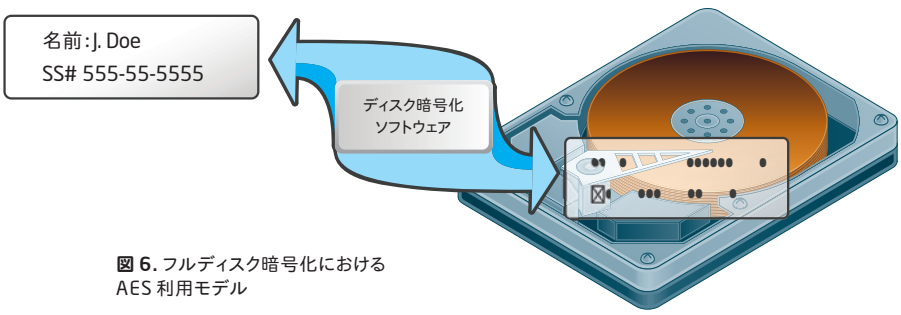

図 6. フルディスク暗号化における AES 利用モデル

さらされる可能性があります。米国国立公文書館（NARA）の例を考えてみましょう。約 7,600 万人分の軍関係者の個人情報が保存されているハードディスク・ドライブが故障したときに、NARA はそのハードディスク・ドライブを修理のために IT 請負業者に送りました。送る前にドライブを消去しなかったため、NARA は、ほぼ間違いなく過去最大と言える政府データの侵害を引き起こしました。[10] 同様に、特定のハードディスク・ドライブを、寿命により廃棄する場合や、新たに使用するために再セットアップする場合には、暗号化を使用することで、機密データを保護するための余分な手間を省くことができます。多数のディスクが存在するデータセンターでは、修復、廃棄、および再セットアップを容易にすることで、コストを削減できます。

6. パフォーマンス向上の可能性

この章では、現在の多数の利用モデルで AES 暗号化の導入を促進するインテル® AES-NI が搭載されたインテル® Xeon® プロセッサー 5600 番台のパフォーマンス向上の可能性について説明します。

パフォーマンスは、さまざまな指標で測定できます。特定のアルゴリズムまたは集中的な処理（カーネル）、アプリケーション全体またはトランザクション全体、またはアプリケーションのセキュリティーの強度などに的を絞ることができます。カーネルの高速化を独立させたパフォーマンス調査の結果は、（アムダールの法則の効果によりパフォーマンスはあまり向上しないと考えられる）アプリケーションの高速化の調査結果よりも何倍も高速な数値となる場合があります（この法則では、並列コンピューティングでの複数のプロセッサーを使用したプログラム高速化は、プログラムの逐次部分の処理に必要な時間によって限定されるとされています）。

現在の市場においては、多くの場合、使いやすさ、機能、およびデータがある程度は暗号化されるという単純な事実の方が、暗号化が実際にどの程度安全かという検討事項よりも重視されます。[11] インテル® AES-NI が提供するようなパフォーマンスの向上により、大きな計算オーバーヘッドなどの代償を伴う

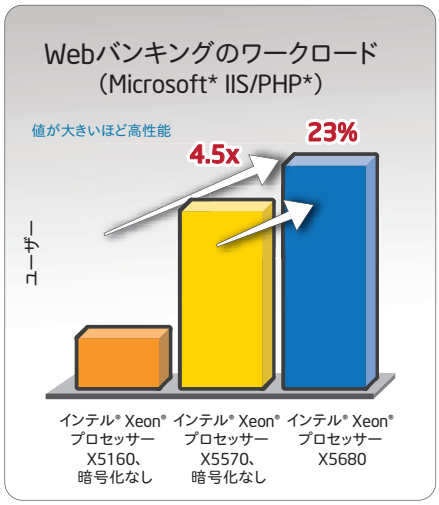

図 7. Windows Server* 2008 R2 Enterprise を使用して実行されたインテルによる測定。PHP を利用したバンキングセッション / ユーザー数の測定結果、インテル® Xeon® プロセッサー X5680（3.33 GHz）とインテル® Xeon® プロセッサー 5160（3 GHz）およびインテル® Xeon® プロセッサー X5570（2.93GHz）との比較。SSD RAID 0 アレイ× 24、TLS_RSA_with_AES_128_CBC_SHA 暗号スイートを実装。

暗号化をより安全な形で採用しやすくなり、長期的には、伝送されたりストレージに保存される個人や企業のデータの安全性が高まります。

このホワイトペーパーにおけるパフォーマンスに関する説明は、インテル® Xeon® プロセッサー 5600 番台に基づく初期の参照プラットフォームおよびインテル® AES-NI に最適化されたソフトウェアを使用した結果に限定されています。

6.1. セキュリティー保護されたトランザクションのパフォーマンス

SSL トランザクションの仕組みを示した図 3 を思い出してください。クライアントは、Web サーバー・セットアップで、動的な HTTPS ページを開き、静的ファイルをサーバーからダウンロードしています。最初に、接続を確立するためのクライアントとサーバー間の RSA ハンドシェイクでトランザクションが始まり、その後に SSL ハンドシェイクと AES を利用できるデータのバルク交換が続きます。新しい TLS プロトコルを必要とする場合も同様の一連の手順が行われます。

暗号化が有効になっている場合、パフォーマンスは低下することが予測されます。ただし、PHP と Windows Server* 2008 R2 を実行している場合の Web バンキングのワークロードに関するインテルの内部分析（図 7）では、インテル® Xeon® プロセッサー 5600 番台を搭載したサーバーでは、より多くのバンキングユーザーをサポートできました。サーバーには、48GB RAM、SSD RAID 0 アレイ × 24、TLS_RSA_with_AES_128_CBC_SHA 暗号スイートが実装されていました。この調査では、SSL（暗号化）を有効にしたインテル® Xeon® プロセッサー X5680 では、SSL なしのインテル® Xeon® プロセッサー X5570 と比較して、23% 多いユーザーをサポートできることが示されました。インテル® Xeon® プロセッサー 5160 と比較して、インテル® Xeon® プロセッサー X5680 では、4.5 倍多いユーザーをサポートできました。したがって、統合をサポートすることで、暗号化を有効にしてもパフォーマンスは向上します。

6.2. アプリケーションレベルの暗号化のパフォーマンス

データベース、ERP/CRM、アプリケーション・サーバー、ミドルウェア、メールサーバー、ハイパーバイザーにおけるアプリケーションレベルの暗号化は、さらなるプロセッサーの使用率と、スレッディング / 同期化のオーバーヘッドを発生させます。データベースでの透過的データ暗号化（TDE）では、プロセッサーの集中的な使用時に、28% のオーバーヘッドが発生します。[12]

データベースなどにおけるアプリケーションレベルの暗号化では、暗号化（ドライブへの書き込み）よりも復号（ドライブからの読み取り）の方がより一般的です。CBC は、AES の一般的なモードであり、TDE で使用されます。[13] CBC の主な欠点は、暗号化が順次処理である（並列処理が可能ではない）ため、平文での 1 ビットの変更が、以降の暗号文ブロックすべてに影響することです。ただし、CBC の復号では、隣接する 2 つの暗号文ブロックから平文に復号できる場合には、並列処理が可能である場合があります。[14]

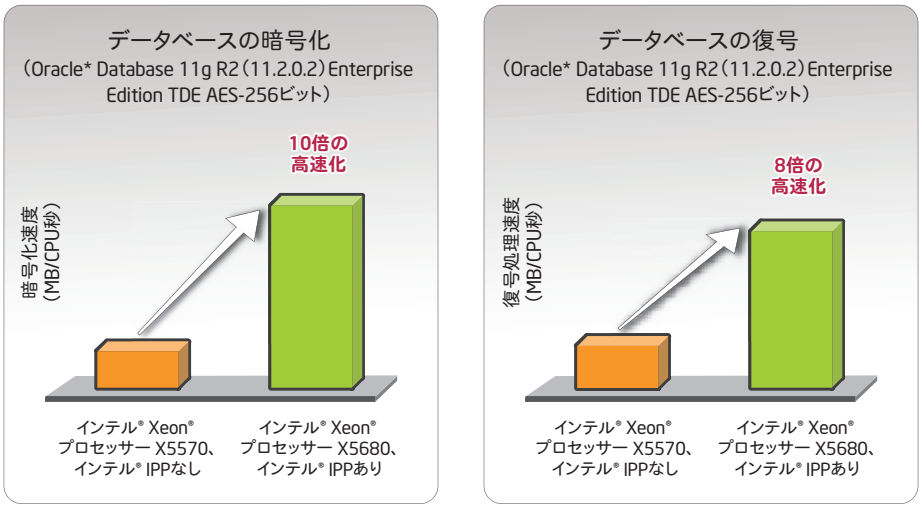

図 8. インテル® Xeon® プロセッサー 5600 番台の暗号化 / 復号のパフォーマンス。Oracle* Database 11g R2 (11.2.0.2) Enterprise Edition Advanced Security TDE AES-256 ビットを使用。

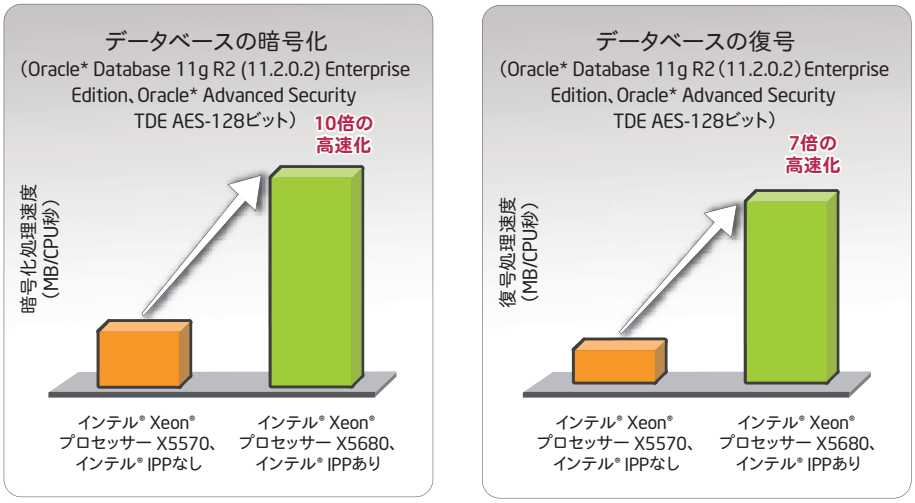

図 9. インテル® Xeon® プロセッサー 5600 番台の暗号化 / 復号のパフォーマンス。Oracle* Database 11g R2 (11.2.0.2) Enterprise Edition の Oracle* Advanced Security に含まれる TDE AES-128 ビットを使用。

Oracle* Database 11g R2 (11.2.0.2) Enterprise Edition の Oracle* Advanced Security では、インテル® AES-NI を使用して TDE のパフォーマンスを向上できるようになりました。図 8 に示すように、TDE AES-256 を実装した Oracle* Database 11g R2 (11.2.0.2) Enterprise Edition を使用し、インテル® インテグレーテッド・パフォーマンス・プリミティブ暗号化ライブラリー (インテル® IPP) で最適化されたインテル® Xeon® プロセッサー X5680 (3.33 GHz、36MB RAM) とインテル® IPP なしのインテル® Xeon® プロセッサー X5570 (2.93 GHz、36MB RAM) を比較したテストでは、100 万の行を空のテーブルに 30 回挿入した場合、10 倍高速化されました。このテストでは、510 万行のテーブルの復号では 8 倍高速化されることも示されました。測定された時間は 8KB のデータあたりの数値であり、MB/CPU 秒単位の暗号化 / 復号の処理速度として示されています。図 9 に示す TDE AES-128 の暗号化および復号では、それぞれ 10 倍および 7 倍の高速化が確認されました。つまり、インテル® Xeon® プロセッサー 5600 番台を

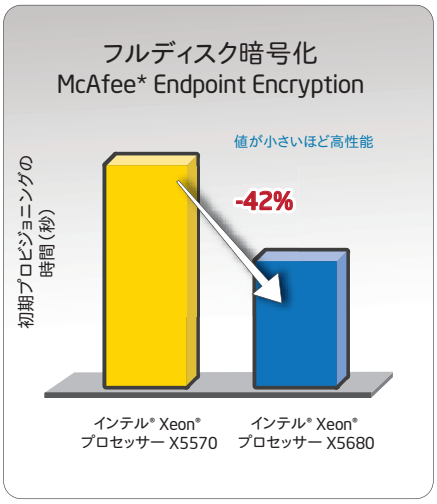

図 10. McAfee* Endpoint Encryption for PC（EEPC）6.0 パッケージ　と McAfee* ePolicy Orchestrator*（ePO）4.5 を使用した 32GB のインテル® X25-E Solid State Drive の暗号化におけるインテル® Xeon® プロセッサー X5680（3.33 GHz）とインテル® Xeon® プロセッサー X5570（2.93GHz）の比較（サーバーのメモリー容量は 24GB）

使用することで、ビジネスにおいて Oracle* Advanced Security に適合するための要件を満たしながら、暗号化の処理コストを大幅に軽減し、最高レベルのパフォーマンスを維持できることになります。

6.3. フルディスク暗号化

一般的に、個々のサーバーのドライブは、クライアントのドライブよりも容量が小さく、レイテンシーも少ない傾向があります。ドライブの初期プロビジョニング（ドライブ上のすべてのデータを暗号化）を行う場合、特に大容量のドライブでは時間がかかり、場合によっては数時間かかることがあります。インテル® AES-NI を搭載したインテル® Xeon® プロセッサー X5600 は、同等の前世代プロセッサーよりも高速にこの暗号化を実行することが期待できます。McAfee* Endpoint Encryption for PC（EEPC）6.0 パッケージと McAfee* ePolicy Orchestrator*（ePO）4.5 を使用した 32GB のインテル® X25-E Solid State Drive の暗号化でインテル® Xeon® プロセッサー X5680（3.33 GHz）とインテル® Xeon® プロセッサー X5570（2.93GHz）を比較したインテル社内の調査では、サーバーの SSD のプロビジョニング時間が 42% 高速化されました（図 10）。

7. アプリケーションでの実装

ソフトウェア・ベンダーは、以下の 3 種類の方法で、インテル® AES-NI を実装することができます。

- OS のライブラリーを用いて命令を使用する
- サードパーティーのライブラリーを用いて命令を使用する
- 新しい命令を使用するアプリケーションの組込み自体をコード化する

ほとんどのソフトウェア開発者は、実際の暗号化に OS の暗号化サービスまたはライブラリーを使用しますが、主要なコンパイラーも、暗号化コードを直接記述するインテル® AES-NI 開発者をサポートできるようになっています。インテルは、OS、ライブラリー、およびコンパイラーのベンダーと連携して、それらのソフトウェアをインテル® AES-NI に最適化してきました。

7.1. オペレーティング・システム

ソフトウェア・ベンダーは、特定の暗号化アルゴリズムの最適化を行わずに、特定の OS に固有の暗号化アプリケーション・プログラミン

表 1. インテル® AES-NI に最適化されたライブラリー

ライブラリー名	場所	インテル® AES-NI の状態
インテル® IPP 暗号化ライブラリー	V6.1：http://software.intel.com/en-us/intel-ipp/（英語）	使用可能
OpenSSL*/OpenSSH/libNSS	http://rt.openssl.org/Ticket/Display.html?id=2067&user=guest&pass=guest http://www.mozilla.org/projects/security/pki/nss/nss-3.12.3/nss-3.12.3-release-notes.html（英語）	使用可能
Microsoft* Cryptographic Next Generation ライブラリー	http://www.microsoft.com/downloads/en/details.aspx?FamilyId=1EF399E9-B018-49DB-A98B-0CED7CB8FF6F&displaylang=en（英語）	使用可能
RSA* BSAFE*	http://www.rsa.com/node.aspx?id=1204（英語）	2010 年第 3/ 第 4 四半期
Java* Cryptography Extensions	http://java.sun.com/javase/6/docs/technotes/guides/security/crypto/CryptoSpec.html（英語）	未定
Crypto++	http://www.cryptopp.com/（英語）	未定

表 2. インテル® AES-NI をサポートするコンパイラー

コンパイラー名	説明	インテル® AES-NI の状態
Microsoft*	Visual C++* 2008 SP1：http://www.microsoft.com/downloads/details.aspx?familyid=A5C84275-3B97-4AB7-A40D-3802B2AF5FC2&displaylang=en（英語）	使用可能
インテル	V11.0：http://software.intel.com/en-us/articles/intel-c-compiler-for-windows-support-resources/（英語）	使用可能
GCC	4.4.0+ および Linux* Binutils 2.18.50.0.6：http://gcc.gnu.org/gcc-4.4/（英語）	使用可能

グ・インタフェース（API）を動的に利用できます。ミドルウェアおよびインフラストラクチャーのレベルで、標準の OS の暗号化ライブラリーをコードパスから呼び出すことができます。

Windows Server* 2008 Release 2 と Windows* 7 で利用可能な Microsoft* Crypto Next Generation（CNG）ライブラリーでは、すべてのキーサイズの AES を含む多数のアルゴリズムがサポートされます。当初はカーネルで IPsec の暗号化部分をサポートするために開発された Linux* 暗号化 API では、暗号化されたファイル、暗号化されたファイルシステム、ファイルシステムの高い整合性、ランダム・キャラクター・デバイス、ネットワーク・ファイルシステムのセキュリティー、および暗号化を必要とするその他のカーネルのネットワーキング・サービスなどのアプリケーションでの使用もサポートされます。インテル® AES-NI の最適化のパッチは、Linux* 暗号化非同期 API に統合されています。

Solaris* 10 には、その他のカーネルモジュールまたはドライバーに暗号化 API を提供するカーネル暗号化フレームワーク（kCF）が用意されています。

7.2. ライブラリー

サードパーティーのライブラリー（表 1）を使用して暗号化をサポートするアプリケーションの場合、ソフトウェア・ベンダーは、暗号化ライブラリー（一部はオープンソース）に静的または動的にリンクすることでアプリケーションを構築します。これらのライブラリーでは、基本的な AES ルーチンが、最適化されたアルゴリズムに置き換えられます。

7.2.1. インテル® インテグレーテッド・パフォーマンス・プリミティブ・ライブラリー

インテル® IPP 暗号化関数ドメインは、米国政府の NIST が策定した Federal Information Processing Standards（FIPS）の仕様に準拠した、一連の作成済み公開キー、対称関数およびハッシュ関数です。ソフトウェア・ベンダーは、インテル® IPP を使用して、堅牢でハイパフォーマンスな暗号化モジュールおよびアプリケーションを迅速に構築できます。

7.2.2. Java* Cryptography Extensions (JCE)

Java Cryptography Extension（JCE）は、暗号化、キー生成、キー合意、および MAC アルゴリズム用のフレームワークおよび実装を提供します。暗号化サポートには、対称、非対称、ブロック、およびストリーム暗号が含まれます。また、セキュリティー保護されたストリームおよびシールされたオブジェクトもサポートされます。

JCE は、その他の認定済み暗号化ライブラリーがサービス・プロバイダーとして機能し、新しいアルゴリズムを追加できるように設計されています。認定済みプロバイダーは、信頼された団体によって署名されています。

JCE（J2SE 5.0 リリース）は、暗号化を必要とする Java* アプリケーション用のデフォルトの暗号化プロバイダーです。Solaris* プラットフォームでは、JCE は Solaris Cryptographic Framework にプラグインされます。JCE は、フレームワークで提供されるすべてのメカニズムを利用できます。

7.2.3. RSA* BSAFE*

RSA* BSAFE* は、非常に広く利用されている、無償でダウンロード可能なモジュールです。RSA* BSAFE* は、FIPS で承認された DSA、rDSA（RSA ANSI X9.31）、DES および TDES モード、SHA-1 アルゴリズムをサポートしています。

7.2.4. Crypto++

Crypto++ ライブラリーは、AES、Diffie-Hellman キー交換、RSA 暗号化、楕円曲線暗号化、およびデジタル署名アルゴリズムの実装で構成される暗号化スキームの無償 C++ クラス・ライブラリーです。AES-GCM、AES-CCM、および AES-CBC モジュールは、ダウンロードして利用可能です。

7.2.5. OpenSSL*

協働の取り組みにより、Secure Sockets Layer（SSL）v2/v3 および Transport Layer Security（TLS）v1 プロトコルを実装した、堅牢かつ商用グレードで、完全な機能を備えた、オープンソースのツールキットが提供されます。OpenSSL* プロジェクトは、強力な汎用暗号化ライブラリーも提供しています。世界規模のボランティアのコミュニティーによって管理されるプロジェクトでは、Web を使用して、OpenSSL* ツールキットと関連ドキュメントに関する通信、計画、および開発が行われています。

OpenSSL* は、Eric A. Young 氏 と Tim J.Hudson 氏が開発した SSLeay ライブラリーを基盤としています。OpenSSL* ツールキットは、Apache 形式でライセンスが供与

されます。ソフトウェア・ベンダーは、商業目的および非商業目的で、無償で入手および使用できます（簡単なライセンス条件に基づきます）。

OpenSSL* は、NIST の Cryptographic Module Validation Program による FIPS140-2 コンピューター・セキュリティー標準で検証されたいくつかのオープン・ソース・プログラムのうちの 1 つです。

7.2.6. Linux* カーネル

Linux* カーネルは、GNU General Public License バージョン 2（GPLv2）でリリースされており、世界中の協力者によって開発されています。物議を醸した一部のバイナリー・ラージ・オブジェクト（BLOB）には所有権ライセンスがあります。

7.3. コンパイラー

AES 暗号にライブラリーを使用しないアプリケーションでは、近く利用可能になるコンパイラー（表 2）を使用したインテル® AES-NI のサポートの追加を検討する必要があります。業界の主要なコンパイラーは、組込み関数またはアセンブリーを使用したインテル® AES-NI のプログラミングをサポートしています。

インテル® AES-NI を使用したソフトウェアの初期開発を可能にするエミュレーターが用意されています。エミュレーターは、こちらから入手してください。http://software.intel.com/en-us/articles/pre-release-license-agreement-for-intel-software-development-emulator-accept-end-user-license-agreement-and-download/（英語）

8. 結論

インテル® AES-NI は、インテル® Xeon® プロセッサー 5600 番台を搭載したサーバーで初めて利用可能になる新しいプロセッサー命令セットを提供します。これらの命令により、高速でセキュリティー保護されたデータの暗号化と復号が可能になります。AES は主要なブロック暗号であり、さまざまなプロトコルで導入されているため、この新しい命令は幅広いアプリケーションで価値を発揮します。

このアーキテクチャーは、AES のハードウェア・サポートを実現する 7 つの命令で構成されています。4 つの命令が AES の暗号化と復号をサポートし、他の 2 つの命令が AES のキー拡張をサポートします。そして、7 つ目の新しい命令 CLMUL が、AES の GCM モードを高速化し、ECC、汎用 CRC、およびデータ重複除外を支援します。これらの命令により、純粋なソフトウェア実装と比較してパフォーマンスが大幅に向上します。

AES の命令には、AES の 3 つの標準的なキー長、すべての標準処理モード、さらにいくつかの非標準の機能または将来の機能をサポートする柔軟性が備わっています。

パフォーマンスの向上に加えて、AES 命令は重要なセキュリティー上の利点ももたらします。命令は、データに依存しないタイミングで実行され、ルックアップ・テーブルを使用しないため、AES のテーブルベースのソフトウェア実装を脅かす主要なタイミングおよびキャッシュベースの攻撃の排除に役立ちます。さらに、これらの命令により AES が実装しやすくなり、コードサイズが縮小されます。このため、セキュリティー上の欠陥を気付かずに残してしまうリスクを軽減できます。

ソフトウェア・ベンダーは、OS の暗号化サービスおよびサードパーティーの最適化されたライブラリーを使用して、インテル® AES-NI に最適化されたルーチンを、静的または動的な方法で容易かつ効率的に統合できます。インテル® AES-NI をアプリケーションに実装することで、ソフトウェア・ベンダーは、最新のインテル・プラットフォーム（インテル® Xeon® プロセッサー 5600 番台以降）のエンドユーザーに AES 暗号化のすべての利点を提供することになりますが、命令ベースの加速化により、パフォーマンスの問題はほとんど発生しません。

このホワイトペーパーでは、セキュリティー保護された商取引、基幹業務アプリケーション、ストレージ、フルディスク暗号化、アプリケーションレベルの暗号化（電子メール、データベースなど）、およびセキュリティー保護された仮想マシンの移行での使用事例におけるインテル® AES-NI 実装の利点について説明しています。暗号化 / 復号ではプロセッサーが集中的に使用されるため、AES アルゴリズムのサブステップを加速化し、アプリケーションをより高速かつ安全に実行するためには、インテル® AES-NI が不可欠です。パフォーマンスにおいては、新しいインテル® Xeon® プロセッサーでは、前世代のプロセッサーで暗号化を使用しない場合と比較して、暗号化を使用しても Web サーバーを最大 23% 高速に実行できることが明らかになっています。これらのパフォーマンスの向上により、パフォーマンス上の理由から以前は実現不能であった場所で暗号化が利用できるようになり、データセンターの情報資産の保護が強化されます。

インテル® AES-NI の詳細については、http://www.intel.co.jp/ を参照してください。

[1] Privacy Rights Clearinghouse、「Chronology of Data Breaches」。http://www.privacyrights.org/data-breach/（英語）

[2] Charles J. Kolodgy、「IDC Encryption Usage Survey」、IDC #213646、Volume:1、2008 年 8 月。

[3] Charles J. Kolodgy、「IDC Encryption Usage Survey」、IDC #213646、Volume:1、2008 年 8 月。

[4] 「States Move to Mandate Encryption of Sensitive Personal Data」、The Last Watchdog on Internet Security、2009 年 3 月 2 日。http://lastwatchdog.com/states-moving-mandate-encryption-sensitive-personal/（英語）

[5] Federal Information Processing Standards Publication 197。http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf（英語）

[6] Shay Gueron、「Advanced Encryption Standard (AES) Instructions Set - Rev 3」、インテル・ホワイトペーパー、2009 年 6 月。http://software.intel.com/en-us/articles/advanced-encryption-standard-aes-instructions-set/（英語）

[7] Google の CEO、Eric Schmidt 氏への 6 ページにわたる手紙には、コンピューター・サイエンス、情報セキュリティー、およびプライバシー法の分野に携わる 38 人の研究者と学者が署名しています。彼らは協働で、Google が顧客と取り交わした重要なプライバシーに関する約束を履行し、Google メール、Google ドキュメント、Google カレンダーで業界標準の伝送暗号化テクノロジー（HTTPS）を有効にして、ユーザーの通信を窃盗や覗き見から保護するように Google に求めています。http://files.cloudprivacy.net/google-letter-final.pdf（英語）

[8] Paul Needham、「Oracle Advanced Security Data Sheet」、2007 年 6 月。http://www.docstoc.com/docs/2659717/Oracle-Advanced-Security/（英語）

[9] Sung Hsueh、Database Encryption in SQL Server 2008 Enterprise Edition、Microsoft SQL Server の技術情報に関する記事、2008 年 2 月。http://msdn.microsoft.com/en-us/library/cc278098.aspx（英語）

[10] 「The Year of the Mega Data Breach」、Forbes、2009 年 11 月 24 日。

[11] Charles J. Kolodgy、「IDC Encryption Usage Survey」、IDC #213646、Volume:1、2008 年 8 月。

[12] Sung Hsueh、Database Encryption in SQL Server 2008 Enterprise Edition、Microsoft SQL Server の技術情報に関する記事、2008 年 2 月。http://msdn.microsoft.com/en-us/library/cc278098.aspx（英語）

[13] Oracle ホワイトペーパー、「Oracle Database 11g: Cost-Effective Solutions for Security and Compliance」、2009 年 6 月。

[14] http://en.wikipedia.org/wiki/Cipher_Block_Chaining#Cipher-block_chaining_.28CBC.29（英語）

性能に関するテストや評価は、特定のコンピューター・システム、コンポーネント、またはそれらを組み合わせて行ったものであり、このテストによるインテル製品の性能の概算の値を表しているものです。システム・ハードウェア、ソフトウェアの設計、構成などの違いにより、実際の性能は掲載された性能テストや評価とは異なる場合があります。システムやコンポーネントの購入を検討される場合は、ほかの情報も参考にして、パフォーマンスを総合的に評価することをお勧めします。インテル製品の性能評価についてさらに詳しい情報をお知りになりたい場合は、「インテル・パフォーマンス：ベンチマークの限界」を参照してください。

インテル株式会社
〒 100-0005 東京都千代田区丸の内 3-1-1
http://www.intel.co.jp/

2010 年 11 月

323587-001JA
JPN/1011/1K/SE/MKTG/NY