対策一覧

対策の内容	参照文書	対策表との対比		頻度	影響度
		方法	有効度		
最も普及したアクセスコントロールメカニズムは、単純なユーザ名／パスワードです。ユーザは、利用しようとしているホストに、ユーザ名と再利用可能なパスワードを入力します。このシステムは、単純な待ち伏せ攻撃に対して脆弱です。ここで、攻撃者は、回線外でパスワードを盗聴し、新しいセッションを開始し、そのパスワードを入力します。この脅威は、TLSやIPSECのような暗号化されたコネクション上にそのプロトコルを置くことによって緩和できます。防護されていない（平文）ユーザ名／パスワードシステムは、IETF標準において許容されていません。	RFC3552	ユーザ名／パスワード	△	高	しばしば、それゆえ、このトラフィックを読むことができる攻撃者は、パスワードを捕捉し、それをリプレイすることができます。換言すれば、攻撃者は、サーバーに対してコネクションを開始し、クライアントのふりをして、捕捉されたパスワードを使ってログインすることができます。
ユーザ名／パスワードよりも高いセキュリティを要求するシステムは、しばしば、ワンタイムパスワードスキームかチャレンジレスポンスのいずれかを採用します。ワンタイムパスワードスキームにおいて、ユーザには、パスワードのリストが提供され、これは、順番に毎回1つずつ使わなければならないものです。（しばしば、これらのパスワードは、何らかの秘密鍵から生成されるので、ユーザは、単純に、順番に次のパスワードを計算できます。）SecureIDやDESGoldは、このスキームの流派です。	RFC3552	ユーザ名／ワンタイム	△	高	ユーザ名／パスワードよりも高いセキュリティを要求するシステムは、しばしば、ワンタイムパスワード[OTP]スキームかチャレンジレスポンスのいずれかを採用します。
企業のように比較的大きな機関によって通常とられているパスワード盗聴に対する予防措置は、OTP（ワンタイムパスワード）システムを使用することです。	RFC2504				
両種のスキーム（ワンタイムパスワードスキームかチャレンジレスポンススキーム）は、リプレイ攻撃に対する防護を提供しますが、しばしば、「オフライン鍵検索攻撃」(待ち伏せ攻撃の 1 形態)に対して脆弱なままです。: 既述のように、しばしば、ワンタイムパスワードやレスポンスは、共有された秘密から計算されます。攻撃者が使われている関数を知っている場合、彼は、正しい出力を作り出すものを発見するまで、すべての共有された秘密の候補を単に試すことができます。共有された秘密がパスワードであり、「辞書攻撃」をしかけることができる場合、これは容易になります。（「単なる乱雑な文字列ではなく、通常の単語（もしくは文字列）のリストを試すこと」を意味します。）これらのシステムは、しばしば、積極的な攻撃に対しても脆弱です。通信セキュリティがセッション全体について提供されない限り、攻撃者は、単に、認証が行われるまで待って、コネクションをハイジャックすることができます。	RFC3552				
ユーザ名／パスワードよりも高いセキュリティを要求するシステムは、しばしば、ワンタイムパスワードスキームかチャレンジレスポンスのいずれかを採用します。チャレンジレスポンスのスキームにおいて、ホストとユーザは、何らかの秘密を共有します。（これは、しばしば、パスワードとして現れます。）ユーザを認証するために、ホストは、ユーザに（乱雑に生成された）チャレンジを提供します。ユーザは、チャレンジとその秘密に基づくいくつかの関数を計算し、それをホストに提供し、ホストはそれを検証します。しばしば、この計算は、DESGoldカードのような携帯デバイスで処理されます。	RFC3552	ユーザ名／チャレンジ	△	高	ユーザ名／パスワードよりも高いセキュリティを要求するシステムは、しばしば、ワンタイムパスワード[OTP]スキームかチャレンジレスポンスのいずれかを採用します。
両種のスキーム（ワンタイムパスワードスキームかチャレンジレスポンススキーム）は、リプレイ攻撃に対する防護を提供しますが、しばしば、「オフライン鍵検索攻撃」(待ち伏せ攻撃の 1 形態)に対して脆弱なままです。: 既述のように、しばしば、ワンタイムパスワードやレスポンスは、共有された秘密から計算されます。攻撃者が使われている関数を知っている場合、彼は、正しい出力を作り出すものを発見するまで、すべての共有された秘密の候補を単に試すことができます。共有された秘密がパスワードであり、「辞書攻撃」をしかけることができる場合、これは容易になります。（「単なる乱雑な文字列ではなく、通常の単語（もしくは文字列）のリストを試すこと」を意味します。）これらのシステムは、しばしば、積極的な攻撃に対しても脆弱です。通信セキュリティがセッション全体について提供されない限り、攻撃者は、単に、認証が行われるまで待って、コネクションをハイジャックすることができます。	RFC3552				
数多くの鍵の問題を解決するためのひとつのアプローチは、認証する主体間を仲介するオンラインの「信用できる第三者（trustedthirdparty）」を使うことです。（一般的にKDC（KeyDistributionCenter）と呼ばれる）信用できる第三者は、共通鍵またはパスワードをシステム中の各主体と共有します。各主体は、まず、KDCと連絡を取ります。KDCは、ランダムに生成されて両者の鍵で暗号化された共通鍵を含むチケットを各主体に提供します。正しいペアのみが共通鍵を復号できるので、そのチケットを信用できる協定を確立するために使うことができます。今日に至るまで最も普及したKDCシステムは、[KERBEROS]です	RFC3552	鍵配布	◎	高	特に、自動化された鍵管理テクニックと関連づけられたプロトコルは、ピアが生きていることを確認し、再生（replay）攻撃から護り、短期セッション鍵の源泉を認証し、プロトコル状態情報を短期セッション鍵と関連づけ、「フレッシュな短期セッション鍵が生成されていること」を確認します。
自動化された鍵管理は、セッション鍵を確立するために使われる必要があります。	RFC4107				
自動化された鍵管理テクニックと関連づけられたプロトコルは、ピアが生きていることを確認し、再生（replay）攻撃から護り、短期セッション鍵の源泉を認証し、プロトコル状態情報を短期セッション鍵と関連づけ、「フレッシュな短期セッション鍵が生成されていること」を確認します。さらに、自動化された鍵管理プロトコルは、暗号アルゴリズムについての交渉メカニズムを含めることによって、相互運用可能性を向上することができます。これらの可変な機能は、マニュアル鍵管理で達成することが不可能、もしくは、極めて面倒です。	RFC4107				
Kerberos は、分散ネットワークセキュリティシステムであり、セキュアでないネットワークに認証機能を提供します。アプリケーションの要求に従って、インテグリティと暗号化の機能も提供することができます。	RFC2196				
4.3 自動鍵配送 IP セキュリティを広く展開し利用するには、インターネット標準規模の鍵管理プロトコルが必要となる。	RFC1825				
変造した電子メールを送ることは、非常に簡単で、電話で（虚偽の）身元をつくろうことは難しいことではありません。暗号技術、例えば PGP （ Pretty Good Privacy ）もしくは PEM(Privacy Enhanced Mail)は、電子メールをセキュアにする有効なやり方を提供することができます。正しい機器をもつことによって、電話のコミュニケーションもセキュアにすることができます。しかし、そのような機構を使う前に、双方の主体が「正しい」基盤を必要とします。つまり、事前の準備です。最も重要な準備は、セキュアなコミュニケーションで使用される暗号鍵の真正性を確認することです。: ＊公開鍵(PGP や PEM のような技術のためのもの): これらはインターネット上のどこからでもアクセスできるので、公開鍵は 使用する前に認証されなければなりません。（ユーザが他の人の鍵に署名 する）PGP は「信頼の蜘蛛の巣」に依存する一方、(CA 局がユーザの鍵に署名する）PEM は、階層構造に依存します。 ＊秘密鍵(DES や PGP/ コンベンショナル暗号化のような技術のためのもの): これらは送り手と受け手の双方が知っていなければならないので、秘密鍵はコミュニケーションの前にセキュアなチャネルを通じて交換されなければなりません。	RFC2350				
1.2. 環境要件 English Kerberos が正常に機能するためには、いくつかの環境要件があります。 ＊Kerberos は、「サービス妨害」攻撃には対処できません。これらのプロトコルには、侵入者によって、アプリケーションが正規の認証ステップを実行するのを妨害できる余地があります。そのような攻撃の検出と対処は、管理者とユーザーが行うのが最も適しています (これらの攻撃は、特別な障害ではなく「正常な」システム障害モードのように見えることがあります)。 ＊プリンシパルは、それらの秘密鍵を秘密にしておかなければなりません。プリンシパルの鍵を盗んだ侵入者は、そのプリンシパルを装ったり、合法的なプリンシパルに対するサーバーになりすましま す。 ＊Kerberos は、「パスワード推測」攻撃には対処できません。ユーザーが簡単なパスワードを使用している場合、攻撃者は解読を繰り返してオフライン ディクショナリ攻撃し、ディクショナリから連続したエントリを取り出し、ユーザーのパスワードから取り出した鍵によって暗号化されているメッセージを取得します。 ＊ネットワーク上の各ホストのクロックは、他のホストの時刻と「柔軟に（ゆるく）同期」しているはずです。この同期は、アプリケーション サーバーがリプレイ検出を実行するときに記録の必要性を削減するために使用されます。「柔軟さ」（ゆるさ）の度合いはサーバーごとに設定できます。ネットワーク上でクロックが同期されている場合、クロック同期プロトコル自体もネットワーク攻撃者から守る必要があります。 ＊プリンシパル識別子は、短期間では再利用されません。一般的なモードのアクセス制御は、アクセス制御リスト((ACL)を使用して特定のプリンシパルに許可を与えます。削除したプリンシパルの古い ACL エントリが残っている状態でプリンシパル識別子を再利用すると、新しいプリンシパルは古い ACL エントリで指定されている権利を引き継ぎます。プリンシパル識別子を再利用しなければ、不注意なアクセスの危険性はありません。	RFC2350				
良いシーケンス番号は、暗号技術による認証の代わりになるものではありません。いいところ、それらは一時しのぎの手段です。	RFC1948				
「チャレンジレスポンス」タイプのシステムは、ユーザが生成したパスワードの代わりに乱雑に生成された共有鍵を使うことによって、辞書攻撃に対してセキュアにできます。鍵が十分に大きい場合、鍵検索攻撃は非現実的になります。このアプローチは、ユーザによって記憶されたり打鍵されるときよりも、鍵が終端に設定されるときに最もうまくいきます。なぜなら、ユーザには、十分に長い鍵を覚えるのに問題があるからです。	RFC3552	共有鍵	◎	高	パスワードに基づくシステムのように、共有鍵システムは、管理問題をわずらいます。通信主体の各ペアは、自らが合意した鍵をもたなければなりません。これは、そこにたくさんの鍵がある状況をもたらします。
場合によっては、（ワンタイムパッドによる共通鍵暗号化、もしくは、米国の AES（Advanced Encryption Standard）[AES] のようなアルゴリズムの利用のように、）秘匿性を確保して、かつ／あるいは、認証と共に通信することを望む主体は、同一の秘密鍵を知らなければなりません。他の場合として、共通鍵もしくは「公開鍵」暗号技術的テクニックが使われるとき、鍵は、ペアとなります。そのペアのひとつの鍵は、プライベートなものであり、ひとりによって、秘密に保たなければなりません。他方は、公開するものであり、世界中に公開することができます。プライベート鍵をその公開鍵から判定することは、計算量的に非現実的であり、公開鍵の知識は、攻撃者 [ASYMMETRIC] に有用となりません。一般的な参考文献 [SCHNEIER, FERGUSON, KAUFMAN] を参照。	RFC4086				
良いシーケンス番号は、暗号技術による認証の代わりになるものではありません。いいところ、それらは一時しのぎの手段です。	RFC1948				

対策一覧

対策の内容	参照文書	対策表との対比		頻度	影響度
		方法	有効度		
単純なアプローチは、[TLS]もしくは[S/MIME]のように、すべてのユーザが何らかのプロトコル固有のやり方で認証するのに使う証明書[PKIX]をもつようにすることです。証明書は、エンティティの身元をその公開鍵に結合する署名されたクレデンシャルです。証明書の署名者は認証局(CA)であり、この証明書自体は、何らかの上位CAによって署名される可能性があります。このシステムが働くようにするために、ひとつ、もしくは複数のCAにおける信頼は、オフラインで確立されなければなりません。このようなCAは、「トラステッドルート」もしくは「ルートCA」と呼ばれます。クライアント／サーバーシステムにおけるこのアプローチに対する主要な障害は、「クライアントが証明書をもっていることを要求すること」であり、これは採用の問題である可能性があります。	RFC2510	証明書	◎	高	このシステムが働くようにするために、ひとつ、もしくは複数のCAにおける信頼は、オフラインで確立されなければなりません。このようなCAは、「トラステッドルート」もしくは「ルートCA」と呼ばれます。クライアント／サーバーシステムにおけるこのアプローチに対する主要な障害は、「クライアントが証明書をもっていることを要求すること」であり、これは採用の問題である可能性があります。
ここに提示しているプロトコルは下記の PKI 管理の要件を満たしています。 1. PKI 管理は、ISO 9594-8 標準およびそれに関連する修正(証明書拡張領域)に準拠しなければならない。 2. PKI 管理は、このシリーズの他の部分にも準拠しなければならない。 3. 他の鍵ペアに何も影響を与えず、任意の鍵ペアの定期的な更新を可能にしなければならない。 4. PKI 管理プロトコルにおける秘密性の使用は、規制問題を緩和するため最小限に留められなければならない。 5. PKI 管理プロトコルは様々な業界標準暗号アルゴリズム(特に RSA、DSA、MD5、SHA-1 を含む)を使用できるようにしなければなりません。 -- すなわち、任意の CA、RA または EE(エンドエンティティ)は、原則として、自分の鍵ペアに適するようなアルゴリズムを何でも使用できることです。 6. PKI 管理プロトコルは対象エンドエンディティ、RA、もしくは CA による鍵ペアの生成を排除してはいけません。鍵生成は他のところで行うこともありますが、PKI管理という目的においては、鍵がエンドエンティティ、RA、または CA のいずれかで最初に現れたときに鍵生成と見なすことができます。 7. PKI 管理プロトコルは対象となるエンド・エンティティや、RA もしくは CA による証明書の公開に対応しなければなりません。実装方式および環境の違いによって、上記のアプローチのいずれかが選択されることがあります。 8. PKI 管理プロトコルは認証済みのエンドエンティティによる証明書の失効要求に応え、証明書失効リスト (CRL) の生成に対応しなければなりません。実施するにあたって、Dos 攻撃 (サービス妨害攻撃)を容易化しないような方式で行わなければなりません。 9. PKI 管理プロトコルは、mail、http、TCP/IP、および ftp を含めた様々な伝送方式が使用できるようにしなければなりません。 10. 認証(証明書)作成の最終権限は、CA にあります; RA またはエンドエンティティ装置は、CA 発行の証明書に要求されたものが含まれているかどうかを推測できません。CA は、運営ポリシーに従って、証明書フィールドの値を変更したり、もしくは拡張領域を追加、削除、変更したりすることができます。すなわち、たとえ実際に発行される証明書と申請の証明書と異なっても、全ての PKI エンティティ(エンドエンティティ、RA 及び CA) は、その申請に対する応答が可能でなければなりません (例えば、CA は有効期間を申請よりも短くすることもあります)。ポリシーの規定に従い、申請元のエンティティが新規生成された証明書を確認し、かつ(典型的な場合は、PKIConfirm メッセージを使用して)受け入れるまでは、CA が証明書の公開もしくは配布を実施してはいけないことに注意してください。 11. 危殆化していない CA 鍵ペアから次の CA 鍵ペアへの切換(CA 鍵更新)を巧みに、計画的に対応しなければなりません。(CA 鍵が失効した場合は、当該CAドメイン内にあるすべてのエンティティに対して再初期化(全ての証明書の再発行)を実施しなければならないことに注意してください) 。(CA鍵更新の結果として)新しい CA 公開鍵を含んだ PSE をもったエンドエンティティは、古い公開鍵で証明書を検証できなければなりません。また、古い CA 鍵ペアを直接に信頼するエンドエンティティは新しい CA 秘密鍵で署名した証明書も検証できなければなりません。(古い CA 公開鍵がエンドエンティティの暗号装置に「ハードウェア的に組み込まれている」状況を想定した場合) 12. 実装方式や環境によって、RA の機能は CA 自身で実現することもあります。通信先が RA なのか CA なのかを問わず、EE が同一のプロトコル(ただし、勿論、同じ鍵ではない！)を使用するように、プロトコルを設計しなければなりません。 13. EE が指定の公開鍵の値を含んだ証明書を申請する場合に、当該 EE が相応の秘密鍵の所有を示す用意が必要です。認証要求の種別によって、実現方法もいろいろあります。PKIX-CMP (すなわち、証明書管理プロトコル) メッセージを対象に定義したイン・バンド方式の詳細については、2.3 節の「秘密鍵所持の証明(Proof of Possession of Private Key)」を参照してください。	RFC2510				
良いシーケンス番号は、暗号技術による認証の代わりになるものではありません。いいところ、それらは一時しのぎの手段です。	RFC1948				
2 つのホスト間のトラフィックのためのIPsec プロトコル(具体的には AH と ESP)は、トランスポートセキュリティを提供できます。IPsec プロトコルは、様々な粒度のユーザ認証をサポートします。例えば、「IP サブネット」、「IP アドレス」、「FQDN(Fully Qualified Domain Name)」、個々のユーザ("Mailbox name")が含まれます。これらの様々なレベルの識別は、IPsec の本質的な部分であるアクセスコントロール機能に対する入力として採用されています。しかし、IPsec 実装によっては、すべての身元タイプはサポートしていない可能性があります。特に、セキュリティゲートウェイは、「ユーザ to ユーザ」認証を提供しない可能性、または、アプリケーションに認証情報を提供するメカニズムをもつ可能性があります。	RFC3552	IPSec	◎	中	IPsec実装によっては、すべての身元タイプはサポートしていない可能性があります。特に、セキュリティゲートウェイは、「ユーザtoユーザ」認証を提供しない可能性、または、アプリケーションに認証情報を提供するメカニズムをもつ可能性があります。 現在、IPsecの主要な用途は、VPNアプリケーション用であり、特にリモートネットワークアクセス用です。セキュリティ管理者とアプリケーション開発者の間の極めて密接な調整無しには、VPNの利用は、個々のアプリケーションにセキュリティサービスを提供するのに適していません。なぜなら、このようなアプリケーションには、「どのセキュリティサービスが実際に提供されていたか」を判定することが困難であるからです。
次に、TLS は、IPsec が無い IP 層の攻撃の影響を受けます。典型的には、これらの攻撃は、何らかのサービス妨害またはコネクション切断の形態をとります。例えば、攻撃者は、SSL コネクションを切断するために、TCP RST を偽装する可能性があります。TLS は、「切断攻撃」を検知するメカニズムをもっていますが、これらは、被害者に攻撃されていることを単に知らせるだけで、このような攻撃に直面したとき、コネクション存続性を提供しません。逆に、IPsec が使われている場合、このような偽装された RST を、TCP コネクションに影響を与えずに棄却できます。偽装された RST もしくは、TCP コネクション上の他のこのような攻撃が懸念される場合、AH/ESP もしくは TCP MD5 オプション [TCPMD5] が選好される選択肢です。	RFC3552				
IPsec は、IPv4 および IPv6 に対して、相互接続可能で高品質な暗号化ベースのセキュリティを提供するように設計されている。提供されるセキュリティサービスには、アクセス制御、コネクションレスインテグリティ、データ生成元認証、リプレイに対する保護(部分的なシーケンスインテグリティの形式)、守秘性(暗号)、そして限定されたトラフィックフロー守秘性が含まれる。これらのサービスは IP 層において提供され、IP とその上位層プロトコルの保護機能を提供する。	RFC2401				
これらのサービスの目標は、認証ヘッダ(AH)と IP 暗号ペイロード(ESP)の 2 つのトラフィックセキュリティプロトコルの利用、および暗号鍵管理手法とそのプロトコルの利用によって達成される。使用される IPsec プロトコルのセット、およびプロトコルの使用法は、ユーザ、アプリケーション、そしてサイト/組織のセキュリティ要件とシステム要件によって決定される。	RFC2401				
IPsec では、システムに要求されるセキュリティプロトコルを選択させ、サービスに利用するアルゴリズムを決定させ、要求されたサービスの提供に必要な暗号鍵を配備させることによって、IP 層におけるセキュリティサービスを提供する。IPsec は、ホスト間、セキュリティゲートウェイ間、およびセキュリティゲートウェイとホストの間の 1 つ以上の「経路」の保護に利用できる(「セキュリティゲートウェイ」は、IPsec プロトコルを実装する中間システムを表すものとして、IPsec 文書全体で用いられる。例えば、IPsec を実装するルータやファイアウォールはセキュリティゲートウェイとなる)。	RFC2401				
IPsec が提供可能なセキュリティサービスには、アクセス制御、コネクションレスインテグリティ、データ生成元認証、リプレイパケットの拒否(部分的なシーケンスインテグリティの形式)、守秘性(暗号)、そして限定されたトラフィックフロー守秘性が含まれる。これらのサービスは IP 層において提供されるため、上位層プロトコル、すなわち、TCP、UDP、ICMP、BGP などでも利用することが可能である。	RFC2401				
3.1. What IPsec Does IPsec creates a boundary between unprotected and protected interfaces, for a host or a network (see Figure 1 below). Traffic traversing the boundary is subject to the access controls specified by the user or administrator responsible for the IPsec configuration. These controls indicate whether packets cross the boundary unimpeded, are afforded security services via AH or ESP, or are discarded. IPsec security services are offered at the IP layer through selection of appropriate security protocols, cryptographic algorithms, and cryptographic keys. IPsec can be used to protect one or more "paths" (a) between a pair of hosts, (b) between a pair of security gateways, or (c) between a security gateway and a host. A compliant host implementation MUST support (a) and (c) and a compliant security gateway must support all three of these forms of connectivity, since under certain circumstances a security gateway acts as a host.	RFC4301				
「本書は、ネットワーク中におけるトランスポートヘッダーの変更によってもらされる潜在的な危険を考慮しません。我々は、『IPsec が使われているとき、そのトランスポートヘッダーは、トンネルモードとトランスポートモード [ESP, AH] の両方において防護されること』を指摘します。」	RFC3360				
IPsec [RFC2401],[RFC2402],[RFC2406],[RFC2407],[RFC2411] は、一般的な、IP 層の暗号化と認証のプロトコルです。そうであるので、これは、TCP と UDP の両方を含むすべての上位層を防護します。その通常の防護の粒度は、「ホスト to ホスト」、「ホスト to ゲートウェイ」および「ゲートウェイ to ゲートウェイ」です。この仕様は、ユーザ単位の粒度の防護を許容しますが、これは、比較的まれです。このようであるので、IPsec は、現在、ホスト単位の粒度が粗すぎるとき、不適切です。	RFC3631				
IPsec は、IP 層に導入されるので、ネットワーキングのコードにまで入り込む可能性があります。これを実装することは、一般に、新しいハードウェアか、あるいは、新しいプロトコルスタックのいずれかを要求します。他方、これは、アプリケーションにとっては、相当に透過的です。IPsec 上で動作するアプリケーションは、それらのプロトコルをまったく変更することなく、向上したセキュリティを得ることができます。しかし、少なくとも、IPsec がより広く配備されるまでは、大部分のアプリケーションは、「自身のセキュリティメカニズムを規定する代わりに、IPsec の上で動作するもの」と想定しては、いけません。大部分の最近の OS(オペレーティングシステム)は、利用可能な IPsec をもっています。大部分のルーターは、少なくとも、コントロールパスについては、もっていません。TLS を使うアプリケーションは、アプリケーション固有の認証の利点を生かすようにする可能性が高いです。	RFC3631				
IPsec についての鍵管理は、証明書か、「共有された秘密」のいずれかを使うことができます。明白な理由によって、証明書が選好されます。しかし、それらは、システム管理者に、より多くの頭痛をもたらす可能性があります。	RFC3631				

対策一覧

対策の内容	参照文書	対策表との対比		頻度	影響度
		方法	有効度		
3.1.1. ESP Encryption and Authentication Algorithms These tables list encryption and authentication algorithms for the IPsec Encapsulating Security Payload protocol. Requirement Encryption Algorithm (notes) MUST NULL (1) MUST- TripleDES-CBC [RFC2451] SHOULD+ AES-CBC with 128-bit keys [RFC3602] SHOULD AES-CTR [RFC3686] SHOULD NOT DES-CBC [RFC2405] (3) Requirement Authentication Algorithm (notes) MUST HMAC-SHA1-96 [RFC2404] MUST NULL (1) SHOULD+ AES-XCBC-MAC-96 [RFC3566] MAY HMAC-MD5-96 [RFC2403] (2)	RFC4305	ESP	◎	中	暗号の基づいたシステムのセキュリティは、選ばれた暗号のアルゴリズムの強さ、およびそれらのアルゴリズムと共に使用されるキーの強さの両方に依存します。そのセキュリティは、さらに総合体系のセキュリティを回避する非暗号の方法がないことを保証するためにシステムによって使用されるプロトコルのエンジニアリングおよび管理に依存します。
ESP can be used to provide confidentiality, data origin authentication, connectionless integrity, an anti-replay service (a form of partial sequence integrity), and (limited) traffic flow confidentiality. The set of services provided depends on options selected at the time of Security Association (SA) establishment and on the location of the implementation in a network topology.	RFC4303				
このメモでは、IPSEC 暗号ペイロードの改訂版 [ESP] および IPSEC 認証ヘッダの改訂版 [AH] での認証の仕組みとして、SHA-1 アルゴリズム [FIPS-180-1] と組み合わせた HMAC アルゴリズム [RFC-2104] の使用法について説明する。HMAC-SHA-1 は、データ生成元認証とインテグリティ保護を提供する。	RFC2404				
暗号ハッシュ関数を使用してメッセージ認証を行なう仕組みである HMAC について記述する。HMAC は、MD5 や SHA-1 などの反復暗号ハッシュ関数を秘密の共有鍵と組み合わせて使用する。HMAC の暗号としての強度は、使用しているハッシュ関数のプロパティに依存する。					
この文書の執筆時点では、特定の暗号アルゴリズムとともに HMAC-SHA-1-96 アルゴリズムを使用することを妨げる問題は知られていない。					
暗号ハッシュ関数を使用してメッセージ認証を行なう仕組みである HMAC について記述する。HMAC は、MD5 や SHA-1 などの反復暗号ハッシュ関数を秘密の共有鍵と組み合わせて使用する。HMAC の暗号としての強度は、使用しているハッシュ関数のプロパティに依存する。	RFC2104				
HMAC [RFC2104] は、選好される shared-secret 認証テクニックです。両者が同一の秘密鍵を知っている場合、HMAC は、あらゆる任意のメッセージを認証するために使えます。これは、乱雑なチャレンジを含み、これは、「HMAC は、古いセッションのリプレイを予防するために採用できること」を意味します。	RFC3631				
ESP は守秘性、データ生成元認証、コネクションレスインテグリティ、リプレイ防止サービス（部分的なシーケンスインテグリティの形式）、そして限定されたトラヒックフロー 守秘性を提供するために使用される。提供されるサービスは、セキュリティアソシエーションの確立時に選択されたオプションとその実装の配置に依存する。守秘性は、他のどのサービスとも独立して選択してもよい。ただし、（ESP 自身、または別に AH を使用することによって提供される）インテグリティや認証を伴わないで守秘性を使用した場合、そのトラヒックは守秘性サービスを弱めることになるある形態の積極的攻撃を受けやすくなる可能性がある（[Bel96] を参照のこと）。データ生成元認証とコネクションレスインテグリティは連携しているサービスであり（以降、まとめて「認証」と呼ぶ）、（オプションの）守秘性と組み合わせてオプションとして提供される。リプレイ防止サービスは、データ生成元認証が選択される場合にのみ選択され、これは完全に受信側の判断で選択される。（デフォルトでは、送信側でリプレイ防止に使用されるシーケンス番号をインクリメントすることが要求されるが、このサービスは受信側がシーケンス番号をチェックする場合のみ有効となる）。トラヒックフロー守秘性のためにはトンネルモードを選択する必要があり、これは、トラヒックを集約することによって実際の送信元と宛先を隠すことが可能な、セキュリティゲートウェイに実装するのが最も効果的である。ここで、守秘性と認証はいずれもオプションではあるが、少なくともこのうち 1 つは選択されなければならない（MUST）ことに注意すること。	RFC2406				
暗号ペイロード（Encapsulating Security Payload：ESP）[RFC-1827] は、ペイロードデータを暗号化することによって、IP データグラムに機密性を提供するものである。 この仕様では US Data Encryption Standard（DES）アルゴリズムの暗号ブロック連鎖（Cipher Block Chaining：CBC）モード [FIPS-46, FIPS-46-1, FIPS-74, FIPS-81] の変形を用いた ESP の利用法について記述している。トリプル DES（3DES）として知られるこの変形は、平文のそれぞれのブロックを 3 回に渡って処理し、それぞれの回では異なる鍵が使用される [Tuchman79]。	RFC1851				
通信する組織の間で共有される秘密の 3DES 鍵は、実際には 168 ビットの長さである。この鍵には DES アルゴリズムによって使用される 3 つの独立した 56 ビット分が含まれる。この 3 つのそれぞれの 56 ビットの副鍵に、パリティビットとして使用するバイト毎の最下位ビット（least significant bit）を足して、64 ビット（8 バイト）として格納される。	RFC1851				
暗号ペイロード（ESP）[Kent98] は、保護すべきペイロードデータを暗号化することによって、IP データグラムに機密性を提供する。この仕様では、CBC モード暗号アルゴリズムの ESP での使用法について説明する。	RFC2451				
IPsec 上で動作する SMTP コネクションは、送信者と最初のホップとなる SMTP ゲートウェイ間、あるいは、あらゆる接続された SMTP ゲートウェイ間のメッセージについて守秘性を提供することができます。すなわち、これは、SMTP コネクションのためにチャネルセキュリティを提供します。メッセージが直接、クライアントから受信者のゲートウェイに行く状況において、（受信者は、ゲートウェイを信用しなければなりませんが）これは、実質的なセキュリティを提供する可能性があります。リプレイ攻撃に対する防護は提供されています。なぜなら、データ自体は防護されており、パケットはリプレイできないからです。	RFC3552				
AES の選考は、以下のいくつかの特性を基本として行われた。 + セキュリティ + 機密扱いではないこと + 一般に公開されていること + 世界中で特許権使用料が無料で利用できること + 最低 128 ビットのブロックサイズを扱えること + 最低、128 ビット、192 ビット、256 ビットの鍵長を扱えること + スマートカードを含め、様々なソフトウェアおよびハードウェアにおける計算効率とメモリ要件 + 実装の柔軟性、単純性、そして、容易性 AES は、政府指定の暗号となるだろう。AES は、最低でも次世紀までは、政府の取扱注意（機密扱いなし）情報を保護するのに十分であると予測される。また、それは、ビジネスや金融機関にも広く採用されると予測される。 IETF IPsec ワーキンググループは、将来的には AES を IPsec ESP のデフォルト暗号として採用し、仕様に適合した IPsec 実装に含まれる MUST のステータスにするつもりである。	RFC3602				
3.2. Authentication Header The implementation conformance requirements for security algorithms for AH are given below. See Section 2 for definitions of the values in the "Requirement" column. As you would suspect, all of these algorithms are authentication algorithms. Requirement Algorithm (notes) MUST HMAC-SHA1-96 [RFC2404] SHOULD+ AES-XCBC-MAC-96 [RFC3566] MAY HMAC-MD5-96 [RFC2403] (1)	RFC4305	メッセージ認証	◎	中	コネクション認証について HMAC を使うことの残念な欠点は、「その秘密は、両者によってクリアに知られていなければならないこと」であり、鍵が長期間使われるとき、この秘密は、望まれないものとなります。
AH provides authentication for as much of the IP header as possible, as well as for next level protocol data. However, some IP header fields may change in transit and the value of these fields, when the packet arrives at the receiver, may not be predictable by the sender. The values of such fields cannot be protected by AH. Thus, the protection provided to the IP header by AH is piecemeal.	RFC4302				
IP 認証ヘッダ（IP Authentication Header（AH））は、IP データグラムに対してコネクションレスインテグリティとデータ生成元認証（これ以降は、単に「認証」と呼ぶことにする）を提供し、さらにリプレイに対する保護を提供するために使用される。後者はオプションのサービスであり、セキュリティアソシエーションが確立される際に受信側によって選択される場合がある（デフォルトでは、リプレイ防止に使用されるシーケンス番号のインクリメントを送信側に要求するが、受信側がシーケンス番号をチェックする場合のみサービスが有効となる）。AH は上位層プロトコルに加え、IP ヘッダの可能な限り多くの部分の認証を提供する。しかしながら、一部の IP ヘッダフィールドは転送中に変化することがあり、パケットが受信側に到達した時のこのフィールドの値が送信側に予測できないものとなることがある。 このようなフィールドの値は AH によっては保護されない。従って AH が IP ヘッダに提供する保護は、ある程度断片的なものになる。	RFC2402				

対策一覧

対策の内容	参照文書	対策表との対比		頻度	影響度
		方法	有効度		
暗号ハッシュ関数を使用してメッセージ認証を行なう仕組みである HMAC について記述する。HMAC は、MD5 や SHA-1 などの反復暗号ハッシュ関数を秘密の共有鍵と組み合わせて使用する。HMAC の暗号としての強度は、使用しているハッシュ関数のプロパティに依存する。	RFC2104				
HMAC [RFC2104] は、選好される shared-secret 認証テクニックです。両者が同一の秘密鍵を知っている場合、HMAC は、あらゆる任意のメッセージを認証するために使えます。これは、乱雑なチャレンジを含み、これは、「HMAC は、古いセッションのリプレイを予防するために採用できること」を意味します。	RFC3631				
3.1 透明性 証拠を収集するために使用する手法は、透過的かつ再現可能である必要があります。あなたは、利用した手法を詳細に再現することを備える必要があり、それらの手法を独立の専門家によってテストされる必要があります。	RFC3227	否認防止	△	中	さらに、署名利用者は、署名者を騙して、署名しようとしているメッセージとは違うメッセージに署名させることを試みる可能性があります。
3.2 収集ステップ ＊証拠は、どこにあるか？どのシステムがインシデントに巻き込まれているか、また、どのシステムから証拠が収集されるかをリストする。 ＊何が関連し、また管理可能でありそうかを確立する。失敗が疑われるとき、不足しているのではなく、集めすぎている。 ＊各システムについて、関連する揮発性の順序を入手する。 ＊変更するための外部経路を削除する。 ＊揮発性の順序に従い、第 5 章で検討するルールで証拠を収集する。 ＊システムの時計のずれの程度を記録する。 ＊収集段階を通じて作業する際に、他のものが証拠である可能性を問う。 ＊各段階を文書化する。 ＊巻き込まれた人を忘れない。誰が居て何をしていたか、何を観察し、どのように反応したか、のノートをつける。 実施可能である場合、あなたは、チェックサムを生成し、収集された証拠に暗号技術的に署名することを検討する必要があります。それは、これにより強い証拠の連鎖を保全することが一層容易になるからです。この際に、証拠を変更してはなりません。	RFC3227				
4.1 カストディの連鎖 あなたは、「どのように証拠が発見されたか」、「どのように扱われたか」および「それについて起きたすべての事項」を明確に記述することができるはずです。 下記事項が、文書化される必要があります。 ＊どこで／いつ／誰によって、証拠が発見、収集されたか。 ＊どこで／いつ／誰によって、証拠が対処、検査されたか。 ＊誰が証拠のカストディとなり、その期間は。どのように、それは保存されたか。 ＊いつ、証拠のカストディを変えたか、いつ、どのように転送が行われたか。（送付番号等を含む。）	RFC3227				
4.2 どこに、どのようにアーカイブするか 可能な場合、（あまり使われていない保存メディアではなく）普通に利用されているメディアが、アーカイビングに利用される必要があります。 証拠へのアクセスは、厳格に制限される必要があり、明確に文書化される必要があります。認可されていないアクセスを検知することができる必要があります。	RFC3227				
認可は、認証された主体が特定の資源もしくはサービスにアクセスする権限を有するか否かを判定する過程です。密接に関連していますが、「認証と認可は、別個の 2 つのメカニズムであること」を認識することが重要です。おそらく、この密接な組み合わせに起因して、認証は、しばしば誤って認可を意味すると考えられています。認証は単に主体を識別し、認可は「人々が特定の行為をできる」か否かを定義します。	RFC3552	認証と認可	△	中	ユーザ名とパスワードのような単純な認証メカニズムを使うとき、認証と認可の区別は、直感的に理解できますが（すなわち、誰もがシステム管理者アカウントとユーザアカウントの相違を理解していますが）、より複雑な認証メカニズムについては、しばしば、その区別が無くなっています。
認可は、認証に依拠することが必要不可欠ですが、認証単独では認可を意味しません。むしろ、行為をするための認可をする前に、認可メカニズムは、その行為が許可されているか否かを判定するように作られねばなりません。	RFC3552				
守秘性（Confidentiality）： 情報にアクセスすることが認可されていない者が、たとえ、その情報の器（例： コンピュータのファイルやネットワークパケット）を見る可能性があっても、その情報を読めないようにする情報の防護。	RFC1704				
データインテグリティ（data integrity）サービス： 認可されていないデータの変更に対して防護するセキュリティサービス。意図的な変更（破壊を含む）とアクシデントによる変更（喪失を含む）の両方を含む。データへの変更が検知可能であることを確認することによる。	RFC3365				
ポリシー コントロール レベルは、2つの別個の機能である認証と認可から成ります。認証は、主張されたユーザの身元を検証する機能です。認証機能は、組織体 中の 1ユーザが他の組織体に認証されるうるように、インターネットをまたいで配布される必要があります。一旦、ユーザが認証されたら、次は、「そのユーザがそのローカル資源に対してアクセスすることが認可されているか」を判定する認可サービスの仕事です。認可が通った場合、ファイアウォール中のフィルタは、アクセスを許可するように更新することができます。	RFC1636				
BCP 38, RFC 2827 は、偽装されたアドレスでネットワークにアクセスするトラフィックを拒否することによって、分散型サービス妨害攻撃の影響を制限し、「トラフィックについて、その正しい発信元ネットワークを追跡可能であること」を確保し易くすることを意図しています。インターネットをこのような攻撃から防護することの副次的効果として、この解決策を実装しているネットワークは、また、この攻撃や、ネットワーク機器に対する偽装されたマネジメントアクセスのような他の攻撃からも自身を護ります。これが問題を生む可能性があるときがあります。（例：マルチホーミングによる場合）本書は、現在のイングレスフィルタリングの運用的メカニズムを記述し、イングレスフィルタリングに関する一般的な論点を吟味し、特に、マルチホーミングの影響について探求します。このメモは、RFC 2827 を更新します。	RFC3704	イングレスフィルタリング	△	中	ブラインド攻撃において、攻撃者は偽装されたIPアドレスを使うことができ、被害者が攻撃者のパケットをフィルタリングすることを極めて困難にしています。
RFC 2827 は、「ISP が、顧客ネットワークによって正規に使われていない発信元アドレスから彼らのネットワークに流入してくるトラフィックを棄却すること」によって、その顧客のトラフィックを警備すること」を推奨します。そのフィルタリングは、その発信元アドレスが、いわゆる「Martian アドレス（0.0.0.0/8, 10.0.0.0/8, 127.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16, 224.0.0.0/4, 240.0.0.0/4 中のあらゆるアドレスを含む予約されたアドレス [3]）」であるトラフィックを含みますが、これに限られません。	RFC3704				
本書において検討されるフィルタリング手法では、正当なプリフィックス（IP アドレス）からのフラッディング（洪水）攻撃に対しては全く何もしませんが、起点となったネットワークの中にいる攻撃者が、境界におけるフィルタリングルールに合わない偽ったソースアドレスを使用して、この種の攻撃を仕掛けることを防ぎます。攻撃者が、正規に通知されているプリフィックス（ IP アドレス）の範囲内にない、偽った発信元アドレスを使用することをはばむために、すべてのインターネット接続プロバイダーには、この文書に記述されたフィルタリングを実装することが強く薦められます。いいかえれば、ISP が、複数のダウンストリームネットワークの経路情報を持っている場合、これらの経路情報以外から来たトラフィックを防ぐために、厳格なトラフィックフィルタリングが使用される必要があります。	RFC2827				
この種のフィルタリングを実装することの利点には、他に、「発信者の本当の発信元を容易に追跡することができるようになること」があります。それは、攻撃者は、正規の、実在する到達可能な発信元アドレスを使用する必要があるからです。	RFC2827				
チェックサムは、たとえその侵入者が物理的なネットワークへの直接のアクセスができても、にせのパケットを受け取ることを防ぎます。シーケンス番号や、他のユニークな（一意の）識別子と併用することで、チェックサムは、「リプレイ（真似）」攻撃という、古い（当時は適切だった）ルーティング情報が侵入者、もしくは誤動作させられるルーターによって返送される攻撃も防ぐことができます。概ね完全なセキュリティは、シーケンス（通番）ないし固有な識別子とルーティング情報の完全な暗号化によって可能です。これは侵入者がネットワークのトポロジー（構成）を推定するのを防ぎます。暗号化の欠点は、情報を処理するのにかかるオーバーヘッド（負荷）です。	RFC2198	ﾄﾎﾟﾛｼﾞｰの破壊	－	中	攻撃がデータを受け取ることができることに依拠する場合、パス外のホストは、まず、自身をパス上におくために、トポロジーを壊さなければなりません。これは決して不可能ではありませんが、よくあるとも限りません。
このための標準的テクニックは、IP TTL の値 [IP] を検証することです。TTL は、各転送者によって、減算されなければならないので、プロトコルは、「TTL が 255 にセットすること」と、「すべての受信者が TTL を検証すること」を命令できます。次に、受信者は、「確認しているパケットは、同一のリンク上からのものである」と信じる根拠をもちます。トンネリングシステムがある状態でこのテクニックを使用するときは注意が必要です。そのようなシステムでは、TTLを減算せずにパケットを通過させる可能性があるからです。	RFC3552	同一リンクの判別	－	中	トンネリングシステムがある状態でこのテクニックを使用するときは注意が必要です。そのようなシステムでは、TTLを減算せずにパケットを通過させる可能性があるからです。