直接対策

脅威の定義	直接的な対策を示すセキュリティ関連RFCとその記述
＜待ち伏せ攻撃（Passive Attack）：RFC1704＞ 認証システムに対する攻撃のひとつ。これは、ストリーム中に何らデータを注入しないが、代わりに、待ち伏せつつ他の主体間を送られる情報を監視できることに依拠する。この情報は、後で正規のセッションに見えるものの際に使われる可能性がある。	＜RFC3552＞ 「インターネットにおいて使われている多くのプロトコルは、それらが少なくとも待ち伏せ攻撃から防護されるようにするために、より強い認証メカニズムをもつ必要がある」と確信しています。また盗聴のような待ち伏せ攻撃に対する最低限の防護は、非開示パスワードシステムを使うことです。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。
＜積極的な攻撃（Active Attack）：RFC1704＞ データストリーム中に偽のパケットを注入することによって、あるいは、データ ストリームを運ぶパケットを変更することによって、不正に、データを変更したり、認証を得たり、あるいは、認可を得たりする試み。	＜RFC2828＞ $ keyed hash （鍵付ハッシュ） (I) 暗号技術的ハッシュ（例： [R1828]）。ここで、ハッシュ 結果への対応は、暗号技術的鍵である 2番目の入力パラメータによって多様となる。（checksum 参照。） (C) 入力データオブジェクトが変更された場合、新しいハッシュ結果は、その秘密鍵の知識無しには正しく計算できない。それゆえ、秘密鍵は、たとえ、そのデータについて積極的な攻撃の脅威があるときにもチェックサムとして使えるように、そのハッシュ結果を防護する。少なくとも 2 つの形態の鍵付ハッシュがある。： ＊鍵付暗号化アルゴリズムに基づく関数。（例： Data Authentication Code 参照。） ＊ハッシュ結果を対応づける前に、入力データオブジェクトパラメータと鍵パラメータを結合すること（例： 連鎖させること）によって拡張された鍵無しハッシュに基づく関数 。（例： HMAC 参照。）
＜再生攻撃（Replay Attack）：RFC1704＞ 以前に送信された正規のメッセージ（あるいは、メッセージの一部）を記録し、再生することによる認証システムに対する攻撃。（パスワード、あるいは、電子的に転送されるバイオメトリックデータのような）あらゆる一定の認証情報は、本物であるかのように見えるメッセージを偽造するために記録されて、後で使われる可能性がある。	＜RFC4107＞ 自動化された鍵管理とマニュアル鍵管理は、まったく異なる機能を提供します。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。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。
＜トポロジーの破壊：RFC3552＞ それゆえ、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。	＜RFC2196＞ 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。
＜同一リンクの判別：RFC3552＞ パス上の特殊ケースは、同一リンク上にいることです。状況によっては、ローカルネットワーク上のホストと、そうでないホストを区別することが望まれます。	＜RFC3552＞ 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。
＜否認防止 ：RFC3552＞ システムがデータインテグリティを提供するとき、受信者は、送信者の身元と 「彼は、送信者が送ろうとしたデータを受け取っていること」の両方に確信をもつことができます。しかし、彼は、必ずしもこの事実を第三者に実証することができるとは限りません。これを実現する機能は、「否認防止」と呼ばれています。	＜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。）

直接対策

＜サービス妨害攻撃：RFC3552＞ 問題のひとつは、「攻撃者は、しばしば被害者を迷惑させるために多くのサービス妨害攻撃から選択できること」であり、これらの攻撃の大部分が阻止できないので、普通の有識者は、しばしば、「可能性はあっても予防できない多くの他のサービス妨害攻撃があるとき、サービス妨害攻撃のうちの一種を防護する点はない」と想定します。	＜RFC2827＞ 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。