特集2

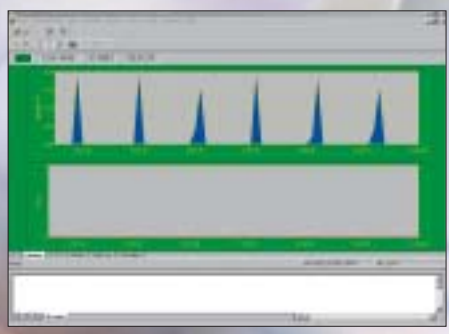
KEYWORD ●セキュリティ●IDS●評価

IDSの検知能力を検証する

20Mビット/秒の負荷で検知率が低下
導入時にはチューニングが不可欠

Webサイトなどへのアタックを自動検出するIDS（侵入検知システム）。セキュリティ強化策として，多くの企業が導入するようになった。ただ，せっかくのIDSも，その特性を踏まえて，きちんと導入・設定しないと，誤検知や取りこぼしが頻発するなど，効果が半減してしまう。にもかかわらず，IDSの動作特性はあまりよく知られていない。そこで，国内で販売されているいくつかのIDS製品を対象に，検知率の評価を実施した。

（三井物産 GTI PROJECT CENTER　伊藤 良孝，平井 伸幸／検証協力：コムワース）



社内ネットワークへの不正侵入や攻撃を監視する「侵入検知システム」（IDS）。すでに，多くの企業や組織に導入が進んでいる。しかし，IDSが攻撃・侵入を検知する動作の特性や，ネットワーク・トラフィックの負荷によってIDSが受けるパフォーマンス上の影響など，必ずしもよく知られていない点がある。

IDSが侵入や攻撃を自動検知するとは言っても，万能なわけではない。トラフィックのパターンなどによって，実際には発生していない侵入・攻撃行為を誤検知するフォールス・ポジティブ状態や，侵入・攻撃行為を検知し損なうフォールス・ネガティブ状態に陥ることがある。

ネットワーク監視型のIDSは，ほとんどの製品が，ネットワーク上を流れるデータ・パケットを監視・解析し，「シグニチャ」と呼ぶ攻撃手法データベースを参照して侵入や攻撃を自動検知する。ところが，現実のネットワークで

◆シグニチャ
IDSに搭載されている攻撃手法に関する情報。攻撃に使われるパケットに含まれる文字列など，各種の攻撃手法の特徴をまとめたデータベース。IDSは，監視しているパケットとシグニチャを付き合わせて，攻撃かどうかを判断する。

◆データの消滅
ネットワーク上では，機器やネットワークそのものの障害や混雑状況によって，パケットが消失してしまうことがある。また，パケットごとに異なる経路を通る場合など，パケットの到着順序が入れ替わる，一部のパケットだけが大幅に遅れて到達するといった現象も起こりうる。

は，データの消滅，重複，到着順序の逆転，遅延といった現象が起こりうる。こうした状況で，IDSはどこまで正確に攻撃を検知できるかは重要である。

クラッカがターゲットを攻略する場合には，データの送出順序を変えるなどして同様の状況を疑似的に作り出し，IDSの監視を逃れようとする可能性がある。この際，IDSがフォールス・ネガティブの状態に陥ってしまうのではIDSを導入した意味がない。反対にIDSによる誤検知が頻発すると，今度はIDSが実質的に機能しなくなり，運用面で支障をきたす。

バックグラウンドのトラフィックの種類や量も，すべてのトラフィックを監視するネットワーク型IDSの侵入検知率と無関係ではない。IDSはネットワーク上を流れるすべてのパケットを監視して，攻撃ではないかどうかを判断する。当然，トラフィック量が増えればIDSの処理負荷が高まる。その際，解析の処理がトラフィックに追いつかなければ，攻撃を検知できなくなる可能性は否定できない（図1）。

そこで，三井物産GTI PROJECT CENTERでは，いくつかのベンダーのIDS製品を対象に評価を実施した。IDSの動作特性がネットワーク・トラフィックの負荷によってどのように変化するのかを測定（定量検証）。さらに，さまざまなパターンで前述の状態を疑似的に作り出し，誤検知や検知漏れに陥る可能性がどの程度あるのかを検証した（定性検証）。

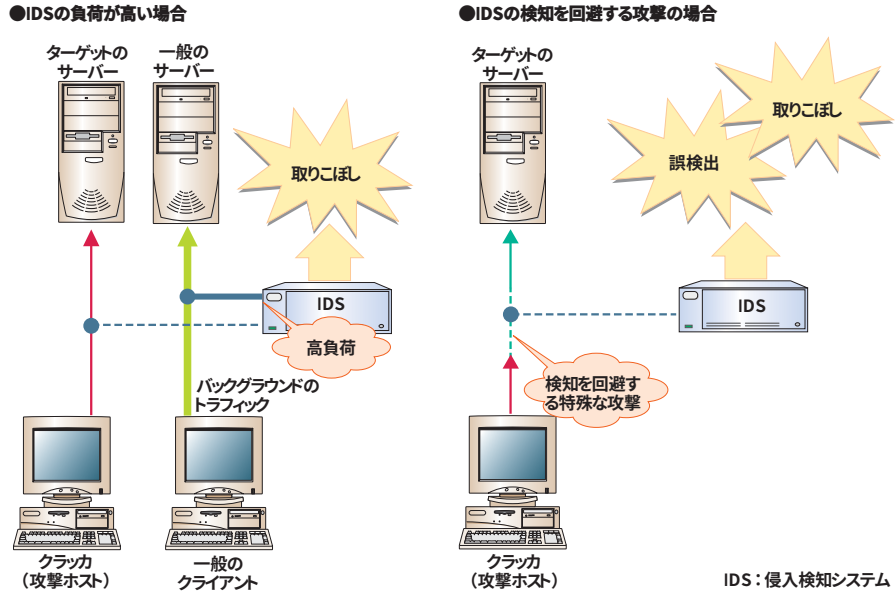


図1　IDSも設定や使い方次第で誤検出や取りこぼしが発生する
特にトラフィックが多く負荷が高まる場合，IDSによる検知を回避するような特殊な攻撃があった場合に発生しやすい。

実際には，ユーザーの環境にそのまま適用できる結果とは言い切れない。ただ，IDSの動作概要を知っておくことは，IDSの設計，導入，運用において有効だろう。

主対象はネット型のソフト製品

評価対象とした製品は表1の通り。米イントルージョンの「SecureNet Pro」と米NFRセキュリティの「NFR」，米エンテラシス・ネットワークスの「Dragon Sensor」と，中心はソフトウエア単体で提供される製品である。これに，オープン・ソースの「Snort」を加えた。これらのソフトについては，基本的にGTI側で用意したハードウエア環境で稼働させた。

GTIが用意したハードウエアは，Pentium III（1GHz）のシングル・プロセッサと512Mバイトのメモリーを搭載した薄型サーバー・マシン（提供はヒューレット・パッカード・ソリューションデリバリ）。OSは，独自OSを同こんしているNFRを除いてWindows 2000またはLinux。LANカードはインテル製である。一部ベンダーの製品については，アプライアンス製品，あるいはベンダー側で用意したハードウエアにインストール済みの製品を使用し，参考として評価した。具体的には，イントルージョンのアプライアンス「SecureNet 7145C」と米SRIインタナショナルのアプライアンス「EMERALD」，米インターネット・セキュリティ・システムズ（ISS）の「Real Secure7.0」である（表1）。

◆phf
Webサーバー向けの攻撃としてよく知られた手法の1つ。phfというのは，一部のWebサーバーに標準的に添付されているCGIスクリプト。このスクリプトが稼働しているWebサーバーに，コマンドを含んだURLを送信すると，そのコマンドを実行してしまうというセキュリティ・ホールがある。

◆CGI
Common Gateway Interfaceの略。共通ゲートウエイ・インタフェース。Webサーバーと他のアプリケーションを連携させるための標準的なインタフェース。

◆ミラー・ポート
LANスイッチに装備されるポートの一種。特定のLANセグメント（またはVLAN）に流れるデータ・パケットをすべて複製して送出するための専用ポート。

評価内容は大きく分けて2通り。1つは，バックグラウンドに別のトラフィックを流した状態で検知率を測定する定量評価。もう1つは，さまざまなパターンの攻撃トラフィックに対する検知/非検知の特性を調べた定性評価である。定性評価としては，攻撃パケットを分割（フラグメント化）して，送出順序を変えるなど，主に検知回避（エバージョン）を意識した評価を実施した（評価環境や攻撃トラフィックの詳細はpp.90-91の別掲記事を参照）。

攻撃には，Webサーバーに付属している「phf」というCGIサンプル・スクリプトのぜい弱性を利用した古典的な手法を採用した。phfには，コマンド文を含んだURLを送ると，サーバー側でそのコマンドを実行してしまうというセキュリティ・ホールがあり，悪用されると，/etc/passwdなどのセキュリティ上重要な情報を盗用される危険がある。

攻撃の基本パターンは，IDSが侵入・攻撃として検知するテスト・ストリング（ここではパスワード検査コマンドの「CRACK」）を含んだ単一パケットの送信である（図2）。次に，テスト・ストリングをTCP，IPそれぞれのレベルで分割して送信した場合に，IDSがテスト・ストリングを正しく再構築し攻撃を検知できるかどうかを検証した。

評価に先立って，ベンダー各社には，定性評価と定量評価の概要を伝え，各社のエンジニア（または各ベンダーが指定したシステム・インテグレータのエンジニア）に，最適なチューニングを施してもらうこととした。ただし，今回の評価に際しては，各社とも定性評価の検知率を意識したためか，すべてのシグニチャを利用したフルオプションに近い設定になっていた。IDSとしては一番負荷が重い状態だが，IDSを導入しようとするユーザーが陥りやすい，「現実によくある設定」の1つと言える。これからチューニングしようとしているIDSの初期状態を再現しているとも考えられる。

20M前後で検知率が低下

まず，定量評価では，バックグラウンド・トラフィックの量に応じてIDSの検知性能がどう変化するかを調べた。攻撃とは無関係なトラフィックが大量に流れる中から，どれだけ正確に攻撃パケットだけを検出できるかである。

バックグラウンドのトラフィックは，Webサーバーとブラウザの間のHTTP（ハイパーテキスト転送プロトコル）トラフィックをシミュレートしたもので，0～60Mビット/秒の範囲で変化させた。60Mビット/秒を上限にしたのは，トラフィックが60Mビット/秒を超えた時点で，IDSに接続するLANスイッチのミラー・ポートにパケット損失が発生したためである。

表1　評価対象のIDS製品
上段の4種類が，今回の検証の中心になったソフトウエアによるIDSで，GTI PROJECT CENTERが用意したハードウエア上で稼働させたもの。下段の3種類は，ハードウエアまでベンダーが用意した製品。このうちSecureNetとEMERALDはアプライアンス。

製品名	メーカー名	国内販売元（今回の製品提供元）	検証時に有効になっていたシグニチャ数	備考
SecureNet Pro	米イントルージョン	エス・シー・コムテクス	検出文字列537，ほかに複数のルール・セット（数は不明）	日本ではアプライアンスのみの提供。性能，シグニチャ数を絞り込んだ低価格製品も提供中
NFR	米NFRセキュリティ	エヌ・シー・エル・コミュニケーションズ	不明	OS内蔵型のソフト。現在は後継機種となるアプライアンスのみ提供。さらに，2002年8月に最新版の製品が登場
Dragon Sensor	米エンテラシス・ネットワークス	エンテラシス・ネットワークス	1672	通常は，1200程度に絞り込んだシグニチャ・セットを推奨。ソフトのほかアプライアンスも提供。2002年8月に後継バージョンが登場した
Snort	（オープン・ソース）	－	977	－
RealSecure	米インターネット・セキュリティ・システムズ	インターネット セキュリティ システムズ	約1200	ソフトウエア製品だが，ISSが用意したハードウエア上で稼働させた。Pentium 4（1.8GHz）×1，メモリー512Mバイト
SecureNet 7145C	米イントルージョン	エス・シー・コムテクス	検出文字列537，ほかに複数のルール・セット（数は不明）	アプライアンス。ハードウエア・スペックは，Pentium III（1.26GHz）×2，メモリー512Mバイト
EMERALD	米SRIインタナショナル	沖電気工業	ルール数109，ほかにパラメータとして検出文字列632	アプライアンス。評価時に使用した機種のCPUはPentium III（1.13GHz）×2

この評価からは，バックグラウンド・トラフィックによる負荷を徐々に増やしていくと，ほとんどのIDS製品で，検知率が下降する傾向が見られた。しかも，検知率が低下し始めるトラフィック負荷は，どの製品も20Mビット/秒前後とほぼ共通している(図3)。また，攻撃パケットを1つのデータとして送信した場合，攻撃パケットを分割して送信した場合のどのパターンでも，同様に検知率が低下した。

IDSの検知能力については，多くのベンダーがギガビット対応をうたっており，数十Mビット/秒はもちろん，数百Mビット/秒のトラフィックにも対応できるという印象が強い。評価したIDSが，どれもフルシグニチャの設定で，本来必要なチューニングが施されていないことを勘案しても，興味深い結果である。

最大の原因はトラフィックの質

IDSのパフォーマンスがこんなにも早い段階で低下した原因としては，バックグラウンド・トラフィックの質による影響が考えられる。IDSのパフォーマンスについては，ほとんどのベンダーが社内評価を経て数値を公表している。しかし，これらの負荷テストでは，レイヤー2レベルのパケットを生成するトラフィック・ジェネレータを使って負荷をかけている場合が多い。

ところが，これらのトラフィックは，アプリケーション・レイヤーから見ると互いに関連性がない無意味なデータになる。これに対して，今回の評価で利用した負荷トラフィックは，HTTPの要求/応答。IDSはシグニチャ参照のために，アプリケーション・レイヤーのパケット・フィールド値やペイロードの中身をすべて検査する必要に迫られる。この処理負荷が，パフォーマンスに影響を与えたようだ。

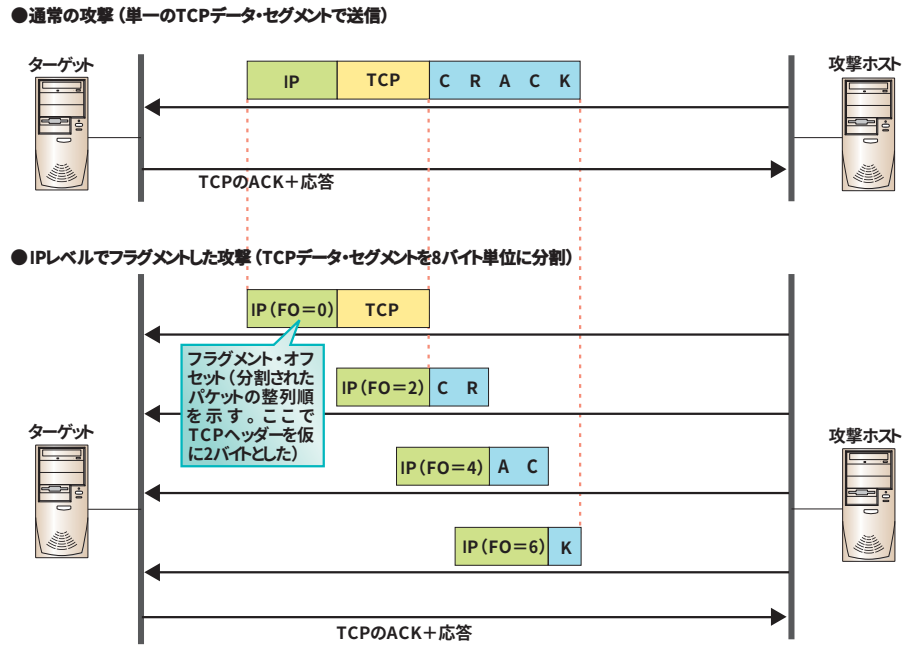


図2　基本テストの内容

基本テストとして，攻撃パケットを1つのTCPデータ・セグメントとして送出した場合と，複数のデータ・セグメントに分割して送出した場合について，IDSが正確に検知するかどうかをチェックした。データ・パケットの分割としては，TCPレベルでのセグメント化と，IPレベルでのフラグメントの両方についてチェックした。

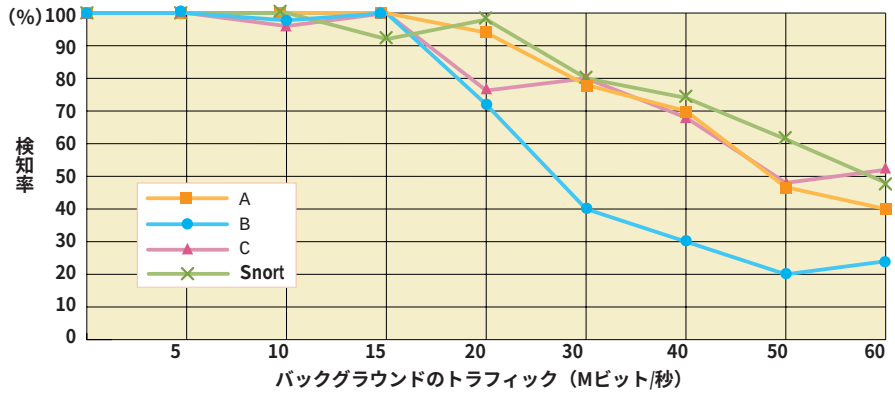


図3　20Mビット/秒程度のトラフィックが発生すると検知率が低下

定量評価の結果。テスト・ストリング（CRACK）を1つのパケットに格納して送信した場合の検知率の推移。

◆DMZ
DeMilitalized Zoneの略。非武装地帯。ファイアウォールを設置する場合に，Webサーバーなど外部公開用のサーバーを設置するセグメント。Webサーバーなどへのアクセス要求については，基本的に制限なく受け付けるため，「非武装」と呼ばれる。ただし，ファイアウォールを経由するため，ある程度のフィルタリングは可能。

もちろん，パフォーマンスが上がらない理由として，ハードウエアのスペック不足も挙げられる。ただ，各ベンダーが推奨するシステム環境や，アプライアンスのスペックを見ても，大抵はPentium III（1GHz）で2CPU構成という程度。たとえば今回の1CPUの結果を2倍したとしても，100Mビット/秒以下で検知率が低下し始めると予想できる。

フラグメントも性能に影響

攻撃パケットを分割して送信した場合は，検知率の低下はより顕著に表れた。バックグラウンド・トラフィックが20Mビット/秒より少ない状況で検知率が低下し始めたのである（図4）。これは前述したパケットの中身を検査する処理の前に，パケットの組み立て処理が発生するため，その負荷が加わったことが原因と見られる。

パケット組み立てでは，フラグメント・パケットを一時的にバッファ・メモリーに記憶させる。このため，バッファ容量によって，パフォーマンスに影響が出る。多くのパケットを記憶させるために，バッファ・メモリーの容量を大きくとると，IDS全体としてのメモリー容量を圧迫し，パフォーマンス低下を招きかねない。逆に小さくとると，パケットの受信量に組み立て処理が追いつかず，検知漏れにつながる可能性がある。

チューニングは不可欠

ただし，今回の結果からIDSを役に立たない仕組みと考えるのは早計である。今回の評価環境は，必ずしも実際のユーザーのネットワーク環境と同じわけではないからだ。

バックグラウンド・トラフィックとして，HTTPがコンスタントに数十Mビット/秒流れるケースはそう多くない。特に，DMZなど，攻撃を受けやすい特定のネットワーク・セグメントを監視する場合は，余分なトラフィックはルーターやファイアウォールでフィルタリングされるため，IDSの負担は少なくなる。

IDSは，チューニングによって性能ががらりと変わる点も見過ごせない。ほとんどのIDS製品は，数の大小はあるが，シグニチャを持つ。たとえばDragon Sensorは1600以上のシグニチャを誇る。NFRのようにシグニチャを持たない製品もあるが，独自の攻撃情報データを持ち，これを参照する処理は必ず実行する。

この際，すべてのトラフィックに対してシグニチャ全体を検索するとなると，処理が重過ぎる。そこで通常は，製品導入時に，利用するシグニチャを絞り込む。こうすることで，検索対象を減らし，パフォーマンスを向上させる。これがIDSの基本的なチューニングである。

シグニチャの絞り込み方は，IDSを導入する環境に依存する。例えばDMZにIDSを設置する場合，攻撃対象になりうるサーバーや，サーバーOS，アプリケーションの種類を特定できる。特定のサーバーに対するトラフィックだけを監視するなら，MACアドレス/IPアドレスをベースにIDS上でトラフィックをフィルタリングし，負荷を軽減できる。対象がLinuxベースのWebサーバーだけなら，Windows関連のシグニチャやHTTP以外のアプリケーションにかかわるシグニチャは使わなく

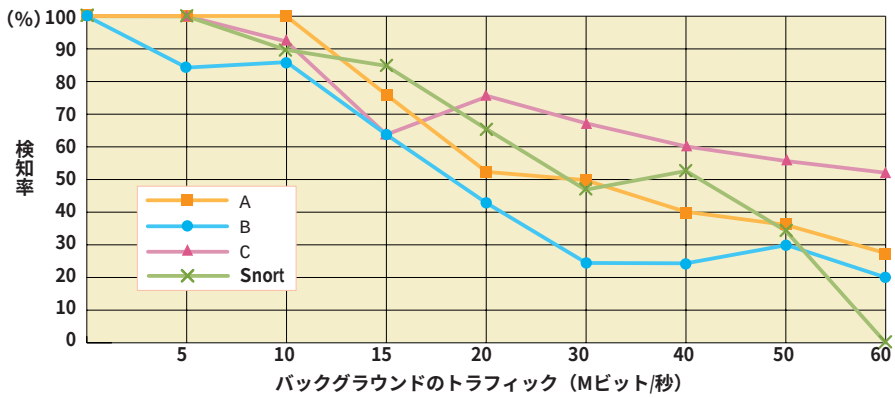


図4　パケットを分割して送信すると性能低下はより顕著に
テスト・ストリング（CRACK）を含む攻撃パケットを，IPレベルで複数に分割して送信した場合の定量評価の結果。1パケットとして送信した場合（図3）と比べると，検知率の低下傾向はバックグラウンド・トラフィックが10Mビット/秒と，より低い領域で表れた。

◆管理コンソール
ここでは，IDSの管理ツールのこと。実際にトラフィックを監視するセンサーから，アラートなどを受け取って，管理者に通知する。

◆センサー
IDSの心臓部。ネットワークを流れるトラフィックを基本的にすべて監視し，パケットの内容やトラフィック・パターンから，侵入・攻撃が発生していないかどうかをリアルタイムに調査する。

て済む。エンテラシスのように，パフォーマンスを向上させられるようにあらかじめ絞り込んだシグニチャ・セットを用意し，推奨しているベンダーもある。

アプライアンスはチューニング済み

チューニングの重要性は，今回参考として検証したアプライアンス製品（またはベンダーが用意したハードウエアを使った製品）の性能からもうかがえる（図5）。

アプライアンスは，マルチCPU構成であるだけでなく，OSやLANカードのドライバまで最適化してある。アプライアンス製品のEMERALDは，検知率低下の原因になったパケット組み立てなどの処理部分を，最新バージョンで強化してあるという。

こうした強化の影響で，アプライアンス製品は，ソフトウエア製品に比べて高い負荷状態でも検知率は良好だった。1製品だけ40Mビット/秒以上で検知率が低下したが，これは☞管理コンソールを☞センサーと同一マシン上で稼働させるなど，望ましくない使い方をしてしまったことが原因と見られる。

このほか，パフォーマンスが高い理由として，攻撃検知のアルゴリズムの影響もあるようだ。一口にIDSと言っても，実際には実装されている攻撃検知の仕組みは製品によってかなり違いがある。たとえばISSのRealSecure 7.0は，検知アルゴリズム自体が従来バージョンとはまったく異なり，同社が買収した米ネットワーク・アイスのアルゴリズムを採用している。検知の高速性を最大の特徴とするアルゴリズムで，ISSは今回の検証ではその効果が表れたと見ている。

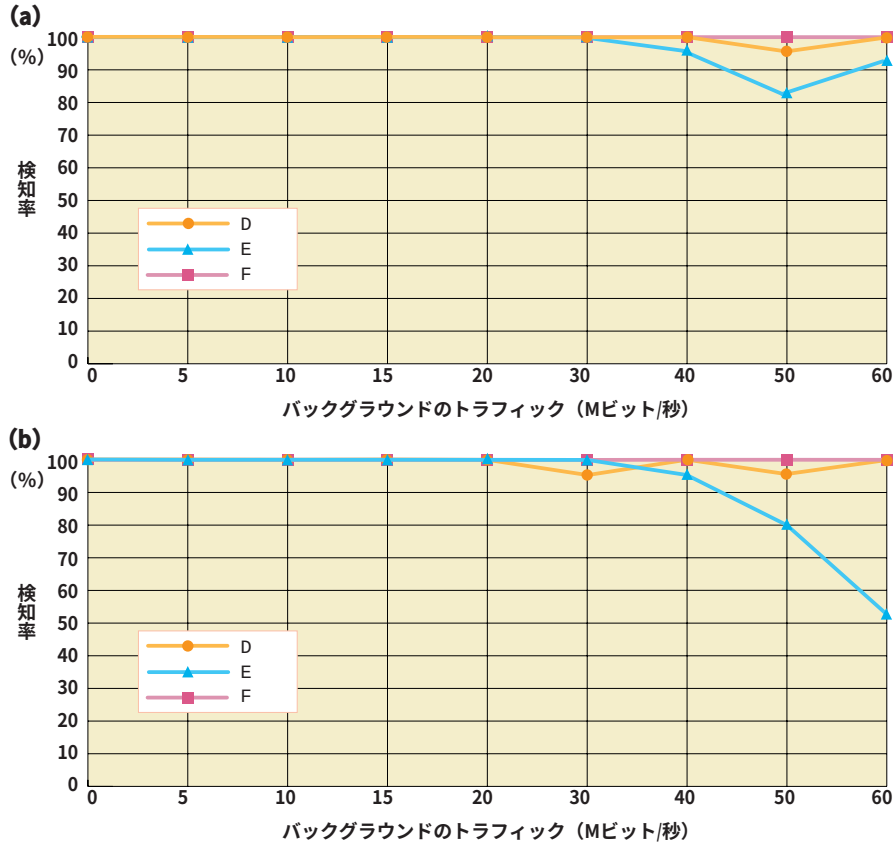


図5　メーカーが推奨する環境では性能は向上
メーカーが推奨する条件に近いハードウエア環境で稼働させた「RealSecure7.0」，アプライアンス製品の「EMERALD」，「Secure Net 7145C」は高い性能を示した。このうち1つの製品については管理コンソールをセンサーと同一マシン上で稼働させるなど，推奨されない条件があったため，バックグラウンドのトラフィックが40Mビット/秒を超えると検知率が低下してしまった。

エバージョンには各社対応済み

次に，定性評価の結果を見てみよう。前述したように，定性評価はIDSの検知を故意に回避するようなエバージョン攻撃を意識した，さまざまなパターンの攻撃に対するIDSの振る舞いの検証である。

ここで言うさまざまなパターンとは，攻撃の種類ではなく，攻撃パケットを分割したり分割したパケットの送出順序を入れ替えたりといったパターンを指す。こうしたパターンの違いにより，誤検知や検知漏れが発生しないかどうかをチェックした。

定性評価では，定量評価でも実施し

◆3ウエイ・ハンドシェーク
TCPのコネクションを確立する場合の動作。クライアントからサーバーに接続する場合，クライアントが接続要求を送信。サーバーが応答を返し，再度クライアントがサーバーに応答を返す。この3つのステップを経てはじめてコネクション確立の動作を完了する。

た基本パターンの測定に加え，TCP，IPレベルで分割したパケットの送出順序を故意に入れ替える，まったく同じパケットを重複して送信する，パケット順を示す識別子が同じで内容が異なる複数のデータ・パケットを送信する（フォワード・オーバーラップ）といった攻撃パターンを試みた。さらに，TCPセッション確立の「3ウエイ・ハンドシェークを悪用したエバージョン攻撃が可能かどうかをチェックした。

基本パターンの攻撃については，どのIDS製品もほぼ結果は同じ。攻撃内容によっては，直接その攻撃を識別できないまでも，「Non-Printable Character」，「BAD ACCESS」というように警告を出すものもあったが，検知したという点は同じである。

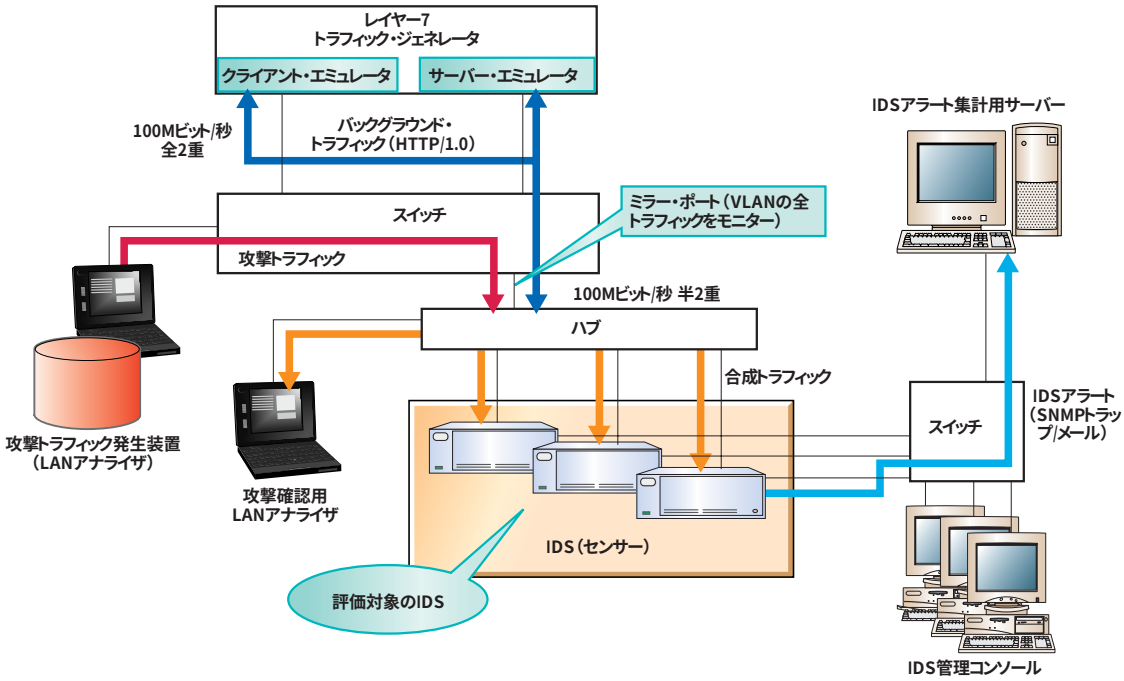
製品によって一部動作が異なる

製品によって違いが出た項目もあった。たとえばフォワード・オーバーラップへの対応である。IPレベルで分割した場合のフォワード・オーバーラッ

検証用トラフィックと評価環境

今回，テストベッドはコムワースの社内に構築した。テスト環境の概要は図Aの通り。負荷を発生させるトラフィック・ジェネレータと，攻撃トラフィックを再生・送信するシステムをLANスイッチで接続。すべてのトラフィックを，IDSを接続するミラー・ポートに送出した。ミラー・ポートにはハブを介して，評価対象のIDSを並列に接続した。また，送信された攻撃トラフィックが途中で消失していないことを確認するために，確認用のLANアナライザも接続した。検知アラートは，別のネットワーク・セグメントに接続した集計システムに，SNMPトラップまたはアラート・メールとして送信する構成である。

実施した検証項目は，トーマス・H・プタセク氏らの論文（Insertion，Evasion，and Denial of Service：Eluding Network Intrusion Detection，Thomas H.Ptacek，Timothy N.Newsham，Secure Networks，Inc. January，1998）をもとにした。これに，米インターネット・セキュリティ・システムズのロバート・グラハム氏



図A　テストベッドの概要
LANアナライザにいったん攻撃トラフィックを記録し，攻撃を再現した。バックグラウンドのトラフィックなしでIDSの純粋な検知率を測定する定性評価と，バックグラウンド・トラフィックがある場合の検知率を測定する定量評価を実施した。定量評価では，トラフィック発生装置でHTTPのやり取りをシミュレートし，バックグラウンド・トラフィックとした。

◆fragrouter
ダグ・ソング氏が作成した攻撃デモ・プログラム。攻撃用のデータ・パケットを細かく分割し，送出順序を入れ替えたり，一部のデータを重複させたりといった操作が可能。改良版のfragrouteもある。

プについてはDragonとRealSecureだけが，TCPレベルについてはNFRとDragonだけが正確に攻撃を検知した。

フォワード・オーバーラップは，パケットを重複させる場合と似たパケット操作である（p.92の図6）。IPパケットの順序（厳密にはデータグラムを組み立てたときのデータの位置）を示すフラグメント・オフセット値を調整し，バッファ・メモリー上で変則的なパケット組み立て処理を発生させる。

たとえば，「CRACK」の文字列のうち，「CR」を含んだIPパケットのフラグメント・オフセットが2，「AC」を含んだIPパケットでは4，「K」を含んだIPパケットでは6になっているとしよう。ここで「AC」の前に，「N」という文字を含んだパケットを，フラグメント・オフセットを5として送信する。こうすると，パケット組み立て処理に使用されるIDSのバッファ・メモリー上では，先に送ったCRとNの間に空白領域ができる。

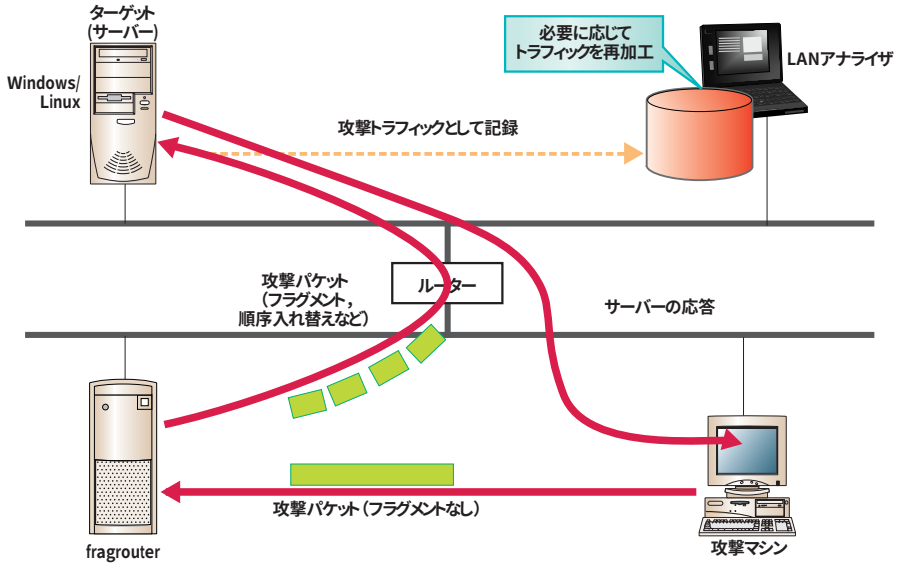
次に，通常通りに「AC」を含んだパケットを送る。当然，フラグメント・オフセットが5になる部分は，パケットが重なる。この際，IDSによって，バッファ上の空白領域ごと「AC」を上書きするか，空白領域を埋めるだけにするかといった動作が異なる。

気をつけなければならないのは，フォワード・オーバーラップによる攻撃は，攻撃対象サーバーのOSの種別によって影響が異なる点である。古いバージョンのWindows NTやSolaris 2.6などでは，先に到着したデータが優先され，テキスト・ストリングは「CRANK」として組み立てられる。このため，攻撃としては有効ではない。これに対して，新しいWindows OSやLinuxでは後に到着したデータが優先されるようになっていて，攻撃は成功してしまう。これはIDSにとっては非常に難しい問題で，ターゲット・サーバーのOSの種別を認識しなければ，検知/非検知のどちらが正しい動作かが違ってしまう。

ちなみに，今回の評価では，ターゲット・サーバーのOSはWindows2000/Linuxであるため，攻撃は成功する。IDSに期待される動作は「検知」である。これに対し，攻撃を検知したIDS

が作成した「Sidestep」（http://www.robertgraham.com/tmp/sidestep.html）による検証，急激に増加しているWebサーバーに対する攻撃を考慮した検証項目を付け加えた全40種類の検証である。

基本的な検証方法は，IDSが存在するネットワーク上に，侵入・攻撃のための文字列を含んだトラフィックを，さまざまなパターンで繰り返し（62秒間隔で5～25回）送信し，IDSが検知するかどうかを調べた。攻撃トラフィックは，検証精度の向上と再現性を考慮し，あらかじめGTI PROJECT CENTER内で実行した攻撃をLANアナライザに記録し，テスト・ネットワーク上で再現した。また，攻撃パケットを分割して送信した場合などの検証項目については，「fragrouter-1.6」というツールを利用した（図B）。



図B　攻撃トラフィックの様子
通常の攻撃トラフィックのほかに，fragrouterというパケット操作専用の装置を使って，データ・パケットのフラグメント，パケット送出順序の入れ替え，ダミー・パケットの送出などを実現。さまざまなパターンでの検知率を調べた。

◆デシンクロナイゼーション
攻撃対象のサーバーと，IDSを異なる状態に陥れることで誤検知や検知漏れを起こさせる攻撃。たとえば，攻撃対象のサーバーに対しては有効ではない不正なTCPのコネクション確立/切断要求を発生させ，コネクションを確立したり切断したりするふりをして，IDSをだます方法。

◆クライング・ウルフ
IDSの誤検知を誘発させて，実質的にIDSにその役割を果たせなくする攻撃。

は，DragonとRealSecureだけ。SecureNet Proは，評価時には検知しなかったものの，ベンダーによれば対応済み。両方のパターンのデータ・パケットをそれぞれ2分の1の確率で組み立て，2回に1回は検知するような仕組みになっているという。

TCP ACKの識別にも違い

もう1つ，TCPレベルで分割したパケットに対する振る舞いでも違いが見られる。ターゲットのサーバーが攻撃ホストに対して，データを受信したことを示すACKパケットを送信したかどうかを確認したうえでデータ・ストリングを再構築できるかどうか，という検証である。

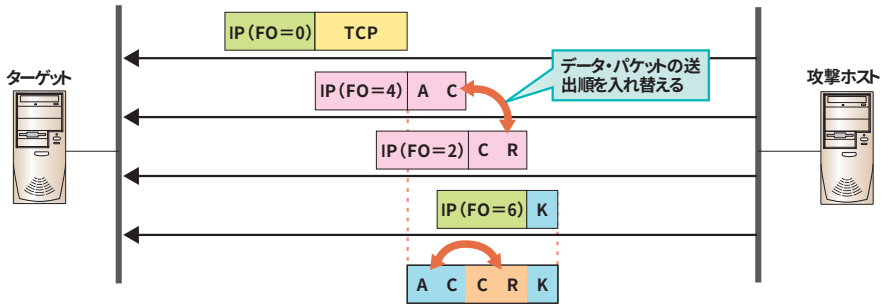
サーバーからのACKを確認するIDSは，侵入・攻撃行為がサーバー上で本当に有効な状態になっているかどうかを判断していることになる。それだけ，誤検知は少ないことが期待される。ただしその半面，何らかの原因でACKパケットが伝送路上で消滅してしまった場合は，IDSは検出漏れの状態に陥る可能性がある。

動作は，各IDS製品によってまちまち。Snort，NFR，EMERALDはすべての攻撃を検知せず。DragonとSecure Net Proはすべての攻撃のうち数回だけ検知。RealSecureだけがすべて検知した。

また，IDSの検知精度は，検知処理のトリガーによって変わる。TCPの3ウエイ・ハンドシェークが完了してから始めるか，クライアントからサーバー方向のコネクション（SYN）が成立した時点で始めるかである（図7）。後者の場合，誤検知や検知漏れに陥りやすい。不正なコネクション要求によってターゲットとIDSを異なる状態に導く「デシンクロナイゼーション」や，誤検知を頻発させて実質的にIDSを使い物にできなくする「クライング・ウルフ」といった攻撃が可能になるためである。

そこで検証では，このような状況でのIDSの動作を調べた。結果は，各IDSとも，デシンクロナイゼーションへの対応は良好だった。クライング・ウルフについては，完全に無視する（非検知の）製品と，特定時間間隔または特定回数ごとに攻撃を検知する製品に分かれた。後者は，有効でない攻撃でも「不正な行為が行われた事実」を表示するという設計思想に基づいていると考えられる。

（例1）データ・パケットの送出順序を変える



（例2）パケット組み立て時に同じ順番に相当するデータとして内容が異なるデータが重複するように送信する

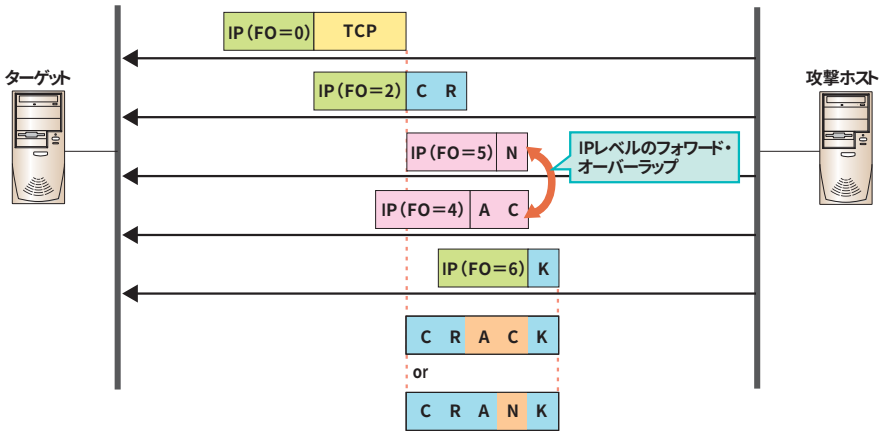


図6　フラグメントされたデータの送出順序を入れ替えてIDSの検知を回避
IDSをだますような攻撃方法もある。たとえばフラグメントされたデータ・パケットの送出順序を意図的に入れ替える，重複するデータ・パケットを送る，内容が異なるパケットを重複するパケットとして送る（フォワード・オーバーラップ）といった方法が考えられる。

◆TTL
Time To Liveの略。データ・パケットの有効期限（経由できるルーターの数）を示す識別子。ルーターを経由するたびに数字が1ずつ減る。同一のデータ・パケットがネットワーク上でループ状に流れ続けてしまうことを防ぐための仕組み。

導入時には入念な設計を

今回の検証は，IDSの機能や特性を知るためのほんの一部の検証を行ったに過ぎない。たとえば実環境では，トラフィック量は絶えず変化するし，さまざまな種類のプロトコルや大きさが異なるパケットが混在する。これは，IDSにとっては，さらに厳しい条件になる可能性がある。

IDSに対するエバージョン攻撃にしても，IPパケットの☞TTL（Time to Live）値を調整する方法やSYNパケットにわざとペイロードを付加する方法など，ほかにもさまざまな方法が考えられる。

このため，実際の導入にあたっては，今回の検証結果とは別に，IDSを導入・設定・運用する上で考慮しなければならない項目がいくつもある。

たとえばIDSをどういう目的で導入するかである。本当に有効（危険）な侵入・攻撃行為だけを監視したいのなら，前述したクライング・ウルフのような攻撃の可能性やIDSの負荷を減らすために，一部の攻撃は検知しない方が望ましい。しかし，なかには，自分のサイトにどんな侵入・攻撃が行われているのかすべて把握したいというユーザーがいるかもしれない。こうした場合，不審なパケットをすべて検知することを期待する可能性はある。

ファイアウォールやルーター，スイッチなど，IDS以外の装置との関連性も重要な問題である。たとえば不正なパラメータを持つパケットや，コネクションが成立していないパケットは，これらの装置で廃棄される可能性がある。これらの装置の動作特性や設定を考慮して，IDSを適切に配置しなければならない。

導入時にはIDSが設置されるネットワークの負荷を把握し，IDSの検知能力に十分に余裕を持たせるように配慮する必要がある。運用時にも，ネットワークの負荷がこのしきい値を超えていないかを常に監視し，IDSが検知漏れの状態に陥っていないかをチェックしながら運用する方がよい。IDSの能力は，こうした基本をおさえてはじめて，最大限に引き出せる。

IDSは検知方法や検知限界速度などの面で確実に進化している。しかし，すべての侵入・攻撃を監視できるわけではない。むしろ，何割かの侵入・攻撃行為は見逃してしまうことを前提に運用すべきである。

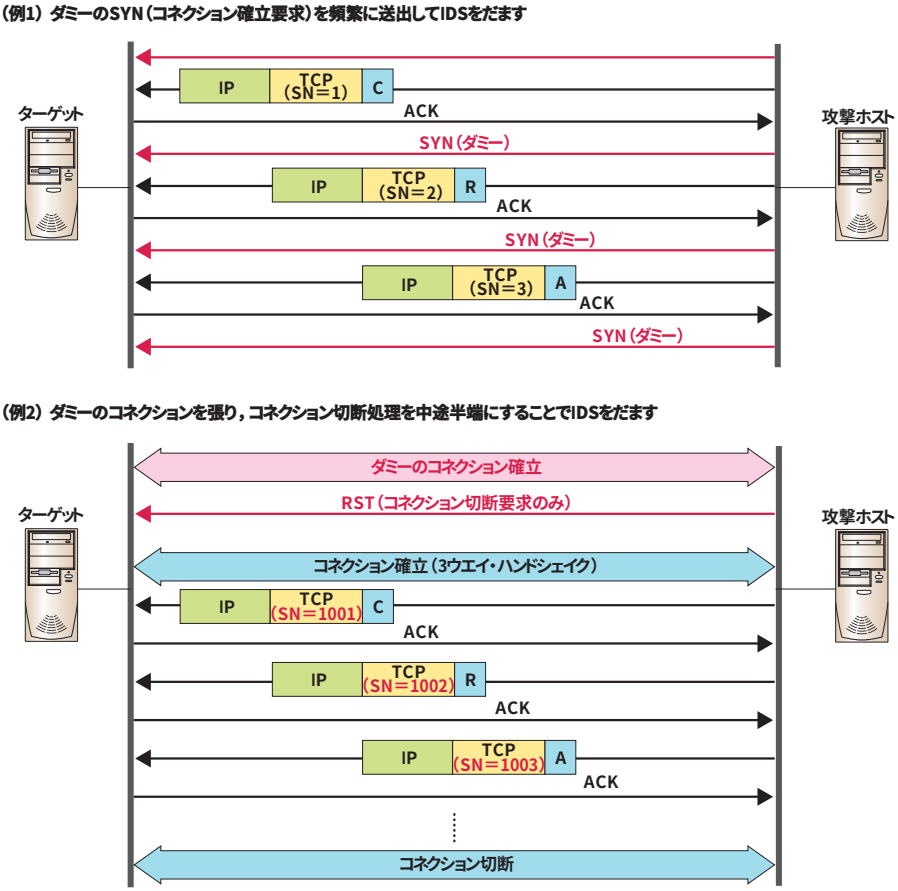


図7　TCPのコネクションを利用してIDSをだます方法もある
TCPコネクションの確立・切断の処理を悪用すると，IDSの検知を回避したり，誤検出を頻発させたりできる場合がある。