マルチメディア通信と分散処理　106－27
コンピュータセキュリティ　16－27
（２００２．２．１５）

ファイアウォール検証に適した高速スキャンツールの研究

種茂文之

ＮＴＴ情報流通プラットフォーム研究所
〒239-0847 神奈川県横須賀市光の丘1-1
tanemo@isl.ntt.co.jp

概要

インターネット接続時のセキュリティを確保するにあたっては、外部からのアクセスをフィルタリングするファイアウォールが正しく設定されていることが重要であり、これを検証する必要が生じている。このファイアウォールの設定検証のために既存のセキュリティスキャンツールを利用した場合、ＵＤＰポートに対するフィルタリング状況が正確に判別できない、検証処理時間に非常に時間がかかる等の問題が発生する。これらの問題を解決するため、サイトの外／内に設置する２つのエージェントが対向で試験パケットを交換することにより、ファイアウォールに対する検証を正確かつ高速に行うスキャン方式の検討を行い、本方式を実装する高速スキャンツールの開発を進めている。本稿では、この高速スキャンツールの概要について述べ、ネットワーク環境での実験結果により処理速度の向上等の効果を示す。

Implementation of High-Speed Scanning Tool adapted to Validate Firewall's Filtering Rules

Fumiyuki Tanemo

NTT Information Sharing Platform Laboratories
1-1 Hikarinooka Yokosuka-Shi Kanagawa 239-0847 Japan

Abstract

To protect a user's network from security threads, firewalls must be set up in secure condition. Therefore, some method to validate firewalls, especially to examine their filtering rules, is needed. In case of using existing scanning tool for this examination, following problems occur: (1) a result of scanning UDP ports is anything but valid, (2) time of scanning TCP/UDP ports is enormous. On that account, we propose new scanning method that consists of two agents: transmitter agent and receiver agent. This paper surveys our high-speed scanning tool, and shows result of examination in which we measure performance of our tool in a network environment.

1. はじめに

インターネット常時接続サービスの普及等に伴い、ますます多くのユーザが自サイトを外部接続させるようになってきているが、それらのサイトのセキュリティは現状では十分確保されているとはいえない[1]。セキュリティに不備があるサイトは、現在頻繁に発生している不正アクセスやDoS攻撃に対して被害を受けたり、あるいは、これらの攻撃の踏み台として使われたりする。サイトのセキュリティ防御のためにはファイアウォールの利用が有効であり、多くのサイトが、ファイアウォールにより外部からの攻撃を遮断しようとしている。

ファイアウォールは外部からのアクセスについて予め定めた一定のルール（アクセスルール）に基づきパケットをフィルタリングすることで、サイトのセキュリティ確保を行うが、本当に外部からの攻撃に対して防御壁としての役割を果たすためには、ファイアウォールがセキュリティ的に正しく設置され、アクセスルールが正しく設定されていることが必須である。セキュリティ的に間違った設定は、多様な理由で発生し、例えば、管理者の設定ミスや知識不足により、あるいは管理者の意図と実際のファイアウォール動作とのミスマッチ等の問題により起こり得る。そのためファイアウォールがセキュリティ的に問題なく動作し、適切なアクセスフィルタリングを行っていることを検証することが重要になる。

本稿では、ファイアウォールが適切に設定されているかを検証可能なセキュリティ監査ツールの提案を行う。一般に、このようなセキュリティ監査のためにはネットワークスキャンと呼ばれる手法が使われ、ツールも多く存在するが、それらの既存の技術を、この問題に適用した場合の限界についても触れる。

本稿では、まず2章でセキュリティ監査の手法であるネットワークスキャン技術について説明する。次に、我々の提案する高速スキャン手法について3章で説明し、それを実装したスキャンツールと、ネットワーク上で本ツールを動作させ処理速度等の測定を行った試験の概要と結果を4章で紹介する。最後に、まとめと今後の課題について述べることとする。

2. 既存のスキャン技術

2.1. セキュリティ監査とスキャン技術

一般に、サイトのセキュリティレベルを検証する作業はセキュリティ監査と呼ばれる。特にインターネットから試験的にアクセスし、サイトのセキュリティ問題を洗い出すインターネットセキュリティ監査において、スキャン技術と呼ばれる監査手法の検討が進んでいる。このスキャン技術を利用したツールは多く存在するが、一般に、ホストやサービスの存在を確認するネットワークスキャン（nmap等）と、サーバやネットワーク機器等でセキュリティホールがあるかどうか確認する脆弱性スキャン(ISS Internet Scanner™, CyberCop Scanner™等)の2種類に分けられる[2]。ファイアウォールのアクセスルール検証も、インターネットセキュリティ監査の一つとしてなされるべきものであり、技術的には上記のネットワークスキャンの領域に属している。

2.2. ネットワークスキャン手法

既存のネットワークスキャン手法では、図1に示すように、検査端末から試験パケットを送信し、その応答状況によりホストの存在やポート上でのサービスの有無を調査する。ファイアウォールは主に、宛先のIPアドレスとTCP/UDPのポートによりアクセス制御をするものであるため、この検証には任意のIPアドレス、ポートへのアクセスを試み、その返答からフィルタリング状況を判断するネットワークスキャン手法(ポートスキャン手法)が利用できると考えられる。

ファイアウォール上のアクセスルールの完全な試験のためには、TCPとUDPの全ポート(1~65535)を試験する必要があるが、ポートスキャンでTCPとUDPとでは、そのプロトコル特性の違いにより、ポートが公開されているかどうかの判断基準が異なったものとなる。

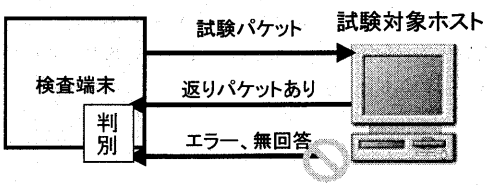

図1　既存スキャンツールの動作

TCP ポートスキャン: ポートへのアクセス時に、必ず返答パケットが戻る。特にサービスが割り当てられたポートへのアクセスではACKフラグ付きのTCPパケットが戻ってくる。このため、正常な返答パケットが戻ってきたことをもって、ポートが公開されているとみなす。

UDP ポートスキャン: TCPと異なり、アクセスしても返答があることがプロトコル的に保証されない。検査対象となるホストの多くが、存在しないポートへのアクセスについてICMPのエラーメッセージを返すため、この場合のみポートが「公開されていない」とみなす。多くのツールで、何度かの試行で返答パケットがない場合は、そのポートは公開されていると判断される。

2.3. 既存のスキャン技術の問題点

上記のようなポートスキャン技術を、ファイアウォール上のアクセスルールの検証に利用する場合、一般に図2のように行うこととなる。

この流れを簡単に説明する。

(1) 検査端末から、対象のIPアドレスとポートに対して試験用のパケットを送付する。

(2) 試験パケットがファイアウォールの通過を試みる。アクセスルールで許可されている場合には通過が成功し、対象ホストにパケットが到達する。通過が拒否された場合には、一般にパケットが廃棄される[1]。

(3) 対象ホストに試験パケットが到達したら、
(TCP) 何らかの返答パケットを戻す
(UDP) UDPやICMPエラーで返答やエラーを戻すか、何も戻さない

(4) 検査端末が、試験パケットに対する返答状況を確認し、上記のポートスキャンの判定基準に従いポートの公開状況を判定する。タイムアウト時間内に返答がない場合、TCPの場合にはポートが閉じられているとみなし、UCPの場合にはポートが開いているとみなす。

このように既存のスキャン技術を、ファイアウォールのアクセスルールの検証に利用した場合、次の問題が発生する。

---

[1] ファイアウォールの製品や設定によっては、ICMPエラーパケットを生成して検査端末に返答することもあるが、全遮断パケットに対して確実にエラーパケットを返すわけではない。

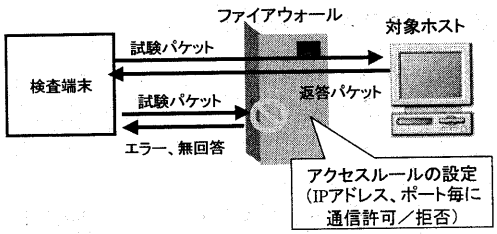

図2 既存ツールによるFWの検証

(1) UDPスキャンの不正確性
UDPポートの調査で、返答がない場合、検査端末はポートの開閉状況を判断する手段がない。なぜなら検査端末から見て、ファイアウォールが試験パケットを廃棄しているのか、サービスが返答しないだけなのか分からないからである。このため多くのツールで、ファイアウォールを跨いだ試験では、現状を正確に反映しない「全ポートが公開されている」という結果が返る。

(2) 検査処理時間の増大
ファイアウォールにより試験パケットが廃棄されるため、多くのポートについて、対象ホストから返答がなく、各試行でタイムアウト処理を行う。かつ、サービスによっては返答に時間がかかるものがあるため、ポートスキャンツールは十分長くタイムアウト値をとっている(UDPの場合1試験パケットにつき1～3秒程度)ことが多い。このため、多くのポートに対して試験をした場合(1～65535の全ポートの試験等)、非常に大きい検査処理時間がかかることになる。nmapのような高速型のスキャンツールで1万ポートの試験において、TCPで1時間程度、UDPで3時間以上かかるのが実状である。

これらの問題に加えて、既存のスキャン技術を利用した場合、そもそもファイアウォールで通過が制御されていないのか、ホストにサービスが存在しないだけなのか確実に判断できないという面もある。ネットワークセキュリティの確保のためには、アクセスが正しく制御されるかどうかを検証するのが重要であるが、既存のスキャン技術では、このような検証が十分できないといえる。このため、ファイアウォールのアクセスルール検証に適した新しいネットワークスキャン技術が必要となるのである。

3. 高速スキャン技術

前記の問題を解消するために、我々は、サイトの外／内に置かれた2つのエージェントにより対向で試験パケットを交換する新しいスキャン技術を検討した（図3）。

ここで、検査エージェントはサイト外部に置かれ、従来のスキャン技術における検査装置に相当し、試験対象となるサイト内部ホスト（IPアドレス）に対して一連の試験パケットを発行する。一方、監視エージェントは、検査対象となるホスト（あるいはその近辺）に設置される装置であり、検査エージェントから届く試験パケットの監視を行っている。これにより、試験パケットがファイアウォールを通過する状況を、試験対象のホストOSやアプリケーションの動作とは関係なく、監視エージェントが判断するため、ファイアウォールのアクセスルールのみを対象とした試験ができ、またUDPの場合でも正確な結果が得られる。

さらに本技術の特徴は、監視エージェントが検査エージェントからの試験パケットを捕捉すると、直ちに検査エージェントに返答パケットを送るようになっている点である。対象ホストに試験パケットが届き次第、返答があることより、次のような性能上の効果がある。

(1) 返答パケットが届いたら検査エージェントは該当パケットに対する処理を完了し、直ちに次の試験パケットの発行処理に入れる。つまり、返答パケットがある場合において、試験処理を高速化できる。

(2) 試験パケットがファイアウォールを通過したら必ず直ちに返答があることが保証されているため、1試験パケット処理におけるタイムアウト値を十分小さくできる。これは、返答パケットがなくタイムアウトが発生する場合においても、試験処理を高速化する。

すなわち、ファイアウォールのアクセスルールがどのように設定されていようとも、本スキャン技術による検査は従来スキャン技術を用いた場合と比べて、かなり高速に検査処理をすることが可能である。実際にツールを作成して試験した場合、従来のスキャンツールと比較して2桁の処理速度の向上がみられるが、これについては次章で述べる。これにより、既存スキャン技術を利用した場合の、検査処理時間の増大の問題を解消できるのである。

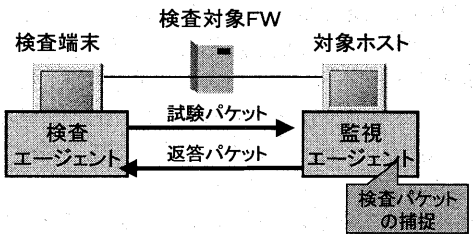

図3　高速スキャン方式の概要

4. ファイアウォール検証試験

前章で示した高速スキャン技術の性能上の効果を確認するため、実際にスキャンツールを実装し、ネットワーク環境でファイアウォール検証試験を行った（図4）。

4.1. スキャンツールについて

今回のスキャンツールの実装では、市販のホスト/OSに搭載されるプログラムとし、検査エージェントおよび監査エージェントの作成を行った。各エージェントの動作環境は共通で、ハードウェアとしてDOS/V互換PC（Intel Pentium IIIプロセッサ800MHz, メモリ512MB, 10/100Mbps IF）を、OSはRedHat Linux 6.2Jでプログラム言語はPerl5を用いた。本ツールにおいて特に監視エージェントは、任意のTCP/IPパケット(生パケット)の取得が必要である。このため、Linux OSにEthernetレベルでパケットを取得するlibpcapライブラリを導入し、また、Perlには、libpcapから生パケットを取得するRawIPモジュールを導入して利用した。

各エージェントのプログラム動作の概要は次のようになっている。

検査エージェント： 試験対象ホスト宛てのUDP(あるいはTCP)の試験パケットを生成してネットワークに送信。その後、UDPパケットの受信待ち状態に入り、予め指定したタイムアウト値だけ、返答パケットを待つ。

監査エージェント： 常にパケットの受信待ち状態であり、試験対象ホスト宛ての試験パケットを捕捉したら、直ちにUDPの返答パケットを生成し、検査エージェント宛てに送信する。

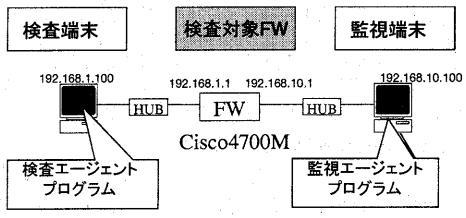

図4　試験環境

一般にネットワークスキャンの処理速度を決めるパラメータは、同時に送る試験パケットの数である並列性と、1試験パケットに対するタイムアウト値である。本プログラムでは、検査エージェントのパラメータとして、並列性とタイムアウト値を任意に指定できるようにし、これらの組み合わせで試験が実施できるようにした。特に並列性については、検査エージェントが子プロセスを複数立ち上げ、同時に試験することにより行った。以下試験を、並列性を決める**プロセス数**（1個～）と、**タイムアウト値**(0.01 秒単位)の組み合わせにより、複数行った結果について述べる。

4.2.　試験方法

図4に示すように、ファイアウォール装置を挟み、検査エージェントを搭載する検査端末と、監視エージェントを搭載する監視端末を設置して、試験を実施した。今回の試験は、ネットワークの帯域や遅延等に大きな影響を受けないようＬＡＮ環境にて行った。試験環境のスペックは以下の通りである。

ファイアウォール装置：Cisco4700M 10Mbps インタフェース（Cisco IOS の access-list 機能でパケットフィルタリングを実現）

ＬＡＮ環境：10Mbps ダムハブと 10baseT ケーブルにて端末ーファイアウォール間を接続

試験は、検査端末から 1～10,000 ポート宛ての1万個の試験パケットを生成・送付し、試験開始から結果が出るまでの時間を測定して行った。また、ファイアウォールでアクセスが、ほぼ遮断されている場合と、ほとんど開放されている場合では、タイムアウト発生の頻度の違いから処理時間に大きな差が出ると予想されるため、各々対応する2つのアクセスルールをファイアウォールに設定して試験を実施した。

許可ルール：　全アクセスを許可する。すなわち実質的にフィルタリングを実施しない。

拒否ルール：　1ポートへのアクセスのみ許可し、他のポートへのアクセスを全禁止する。

4.3.　試験結果

4.3.1.　許可ルールによる試験結果

図5は、ファイアウォールに許可ルールを設定した場合の UDP1 万ポートの試験結果である。許可ルールの場合、全ての試験パケットがファイアウォールを通過し、返答があるため、タイムアウトは発生せず、15～45 秒という比較的短時間で処理が可能である。

処理時間は、プロセス数やタイムアウト値に依存するが、それ程の違いはない。これは監視エージェントのパケットを処理する性能に限界があることによると考えられる。特にタイムアウト値を 0.03 秒未満といった小さな値に設定すると、監視エージェントからの返答パケットの遅れにより、全返答パケットが捕捉できず、正しい結果が得られない。このため、タイムアウト値は、検査の正確性を保証するためにも 0.04 秒以上が必要である。つまり、タイムアウト 0.04 秒以上で、同時パケット数 4 個程度における処理速度である 35 秒が最速値といえる。逆に、これ以上プロセス数やタイムアウト値を増やしても、あまり処理速度の変化はなく、特にプロセス数の増加による検査エージェントの負荷により、処理速度が若干低下する傾向がある。

4.3.2.　拒否ルールによる試験結果

図6は、ファイアウォールに拒否ルールを設定した場合の UDP1 万ポートの試験結果である。ここでは、ほとんどの試験パケットでタイムアウトが発生するため、処理時間は予想の通り、タイムアウト値に比例し、プロセス数に反比例するものとなっている。許可ルールの場合と異なり、ファイアウォールの遮断により、監視エージェントに到達する試験パケットの数が少ないため、監視エージェントに負荷がかからず、これによる返答パケットの遅延は発生しなかった。今回の試験範囲では、プロセス 32 個のタイムアウト値 0.01 秒で、処理時間 13 秒という性能を上げている。上記の許可ルールの最適値であるプロセス数 4 個、タイムアウト値 0.04 秒でも、100 秒程度である。

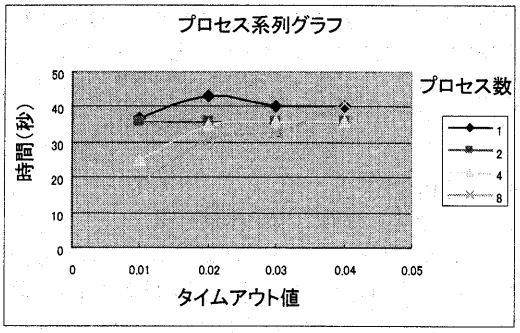

図5　許可ルールにおける試験結果

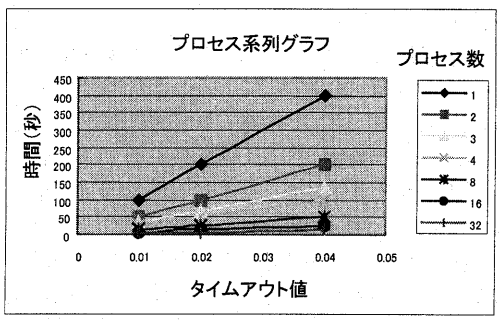

図6　拒否ルールにおける試験結果

4.4. 考察

拒否／許可の2つのアクセスルールにおいて、既存の代表的なスキャンツールであるnmapの処理時間と、本ツールでプロセス4個、タイムアウト値0.04秒で試験した測定結果の比較[2]を表1にまとめた。上述した通り、拒否ルールによる本ツールの最速値は13秒であるが、比較対象のnmapで性能関連のパラメータを自由に設定する機能がない[3]こともあり、許可ルールと同条件での試験結果で比較を行った。これにより、本スキャンツールが、従来のものと比べて2桁程度の速度向上の効果があることが分かる。拒否ルールに対する検証の方が、許可ルールに対するものより遅い（本ツールで3倍程度）が、この点は従来のツールと変わらない。

	本ツール	nmap
許可ルール	35”	2:47'09”
拒否ルール	100”	3:20'33”

表1　UDP スキャン1万ポートにおける、本ツールとnmapの処理速度の違い

5. まとめ

本稿では、ファイアウォール検証の重要性について述べ、ファイアウォールのフィルタリング状況の確認に利用可能なスキャン手法を提案した。また、本スキャン手法を実装するプログラムを作成し、ネットワーク環境で試験することにより、有効性の確認を行った。

現状のプログラムはLAN環境で十分な高速な試験が可能であり、これはサイトに出向いての現地監査時に効果を発揮すると考えるが、インターネット経由での試験などWANを想定した試験では、遅延の発生や帯域の狭さ、あるいはパケットロスにより、正確かつ高速な結果が得られないことが想定される。このため、WAN環境での試験を今後行い、適切なパラメータの選択やプログラムの修正により改善を図りたいと考えている。

本手法は、ファイアウォールを設置した後、その設定状況を確認するようなSI向けツールを想定して検討を進めたが、個人環境のセキュリティ状態を確認するためにも利用可能である。例えば、個人の端末とインターネット間のフィルタリング状況を確認し、端末に対してどのようなアクセスがあり得るかを検査するためにも使える。今後は、このようなセキュリティ診断サービスの高度化に向けて、提案した高速スキャン技術の適用を進めていく予定である。

[2] nmapは試験するポートの順番を毎回ランダムで変えるため、場合によっては、より速い結果が出ることがある。本結果は、数回の試行の平均値である。

[3] 現在のnmapは5段階のモードで試験速度を変えることができ、モード毎に固有のパラメータをもっている。各パラメータを自由に組み合わせることはできない。今回の試験は、3段階目のNormalでの試験とした。

参考文献

(1) JPCERT/CC　http://www.jpcert.or.jp/
(2) 武田圭史, 磯崎宏, “ネットワーク侵入検知” ソフトバンク, pp.136--170(2000)