M-084

サービスを秘匿するTCPコネクション確立方式の設計
Design of a TCP Connection Scheme for hiding services

梅澤 健太郎 † 高橋 俊成 † 鬼頭 利之 †
UMESAWA kentaro TAKAHASHI Toshinari KITO Toshiyuki

1. はじめに

現在, サーバ計算機において公開されているサービスに対して無差別かつ広範囲に行われる攻撃が問題となっている. この攻撃（以降, **単純攻撃**と呼ぶ）は, サービスの有無を確かめるためのポートスキャンと, ソフトウェア脆弱性に対する攻撃のいずれかもしくは両方によるものである. ソフトウェア脆弱性（以降, 単に**脆弱性**と呼ぶ）に対する攻撃は, ソフトウェアのバグへの攻撃であり, 攻撃対象の計算機の特権を得たりすることを可能とする. この攻撃は, 多くのサーバ／クライアントシステムが安全性の拠り所としているセキュリティアプリケーション (SSL (Secure Sockets Layer)[1], SSH (Secure SHell)[2]) に対しても有効であること [3, 4] から大きな問題となっており, また, 未知の脆弱性に対する攻撃も起こりえるなどその潜在的な危険性は計り知れない.

我々はこの問題への対策の一つとして TAP (TCP layer Application Protector) を提案している [5]. この技術は, サービスの存在を秘匿することで攻撃対象となることすら回避し, 仮に攻撃対象となった場合でも攻撃者には接続をさせないというものである. このような特徴を活かすことでリモートアクセス／メンテナンス用のサーバアプリケーションなどの安全性が高められ, 副次的効果として, アプリケーション（ファームウェア）のアップデート頻度を下げることができるため保守コストを下げることも可能である. 本論文では, TAP の技術概要を説明すると共に, 実用性の検討や考察を行う.

2. 単純攻撃の分析と対策

2.1 単純攻撃の分析

単純攻撃は, TCP[6, 7] の仕様に準拠したサーバにおいて TCP コネクションの確立は誰もが可能であるということに起因する問題である. つまり, 一般的な TCP プロトコルにおいてはクライアントがサーバに送信する SYN(SYNchronize) パケット, サーバがそれに応答しクライアントへ送信する SYN/ACK(ACKnowledgement) パケット, それに応答してクライアントがサーバへ送信する ACK パケット, からなるスリーウェイハンドシェイクによって誰でもコネクションが確立できることが原因である.

2.2 対策方針と既存の問題点

単純攻撃を防御するためにはコネクション確立要求である SYN パケットを受信した際に, それに対する応答を制御するのが一つの方法である. このような制御処理は, IP アドレス, ポート番号を指定するパケットフィルタリング技術として, ファイアウォール等で一般に用いられているが, サービスによっては端末 IP が動的に変化

† （株）東芝 研究開発センター, TOSHIBA Corporate Research & Development Center

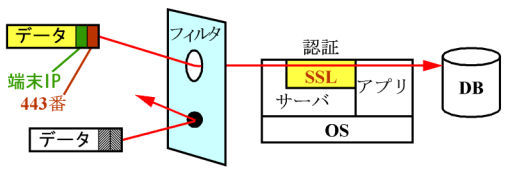

図 1: パケットフィルタリングの限界と現状の様子

するなど, パケットフィルタリングの適用には限界がある. この場合には, SSL や SSH の完備性を拠り所としてセキュリティが構築されているのが現状である (図 1).

3. TAP 技術概要

TAP は TCP コネクションの確立を制御し, サービスを秘匿する技術であり, 単純攻撃を目的としたコネクションの確立要求からサービスを防御する. TCP コネクションの確立制御は, 複数の SYN パケットに格納された認証情報を利用して正規のコネクション確立要求を識別することで行う. 通常の SYN パケットを利用することで既存のネットワークインフラとの高い親和性を実現し, さらに SYN に格納した認証情報を用いて制御することで IP アドレスの動的変化にも対応する. 以下で TAP の接続処理の概要を説明する:

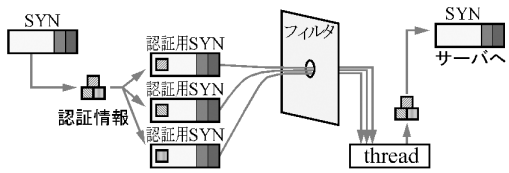

図 2: TAP 処理概要

Step.1 SYN パケット送信：
TAP クライアント (TAP のクライアント機能が実装されたプログラム) は TAP サーバ (TAP のサーバ機能が実装されたプログラム) と共有した認証情報を複数の SYN パケットに格納して送信する. この際, 認証情報として接続毎に異なる値を生成することでリプレイ攻撃を防ぐ. これらの生成された認証情報は, 論文 [5] で検討した既存のネットワーク制御技術との整合性を考慮し, TCP ヘッダのシーケンス番号に埋め込んでおり, その他にリプレイ防止のための時間情報や複数の SYN から認証情報を再構成するための情報も埋め込んでいる.

Step.2 SYN パケット検証：
TAP サーバは TAP クライアントから複数の SYN パケットを受信し, そこに格納された認証情報を予め TAP クライアントと共有した情報を利用して検証処理を行う. ここで複数の SYN パケットを送信してきたクライアン

トの同一性は, 送信元IP アドレスや他フィールド値などを利用して判断する.

Step.3 SYN パケット通過：

TAP サーバは TAP クライアントから受信した複数の SYN パケットの認証情報が正しい場合には, 受信した複数の SYN パケットの何れか一つ以上 (基本的には一つ) を通過させる. その後, その SYN パケットに対する SYN/ACK パケットの返信処理が行われる. 認証情報を格納した SYN パケットに対する応答処理として SYN/ACK パケットを返答するため, この SYN パケットを送信したクライアントは SYN/ACK を受信でき, TCP コネクションを確立できる.

4. 考察

4.1 TAP によって可能となること

TAP によりサービスが秘匿されることで攻撃対象となる機会が減少する. さらにウイルス・ワーム・スクリプトキディによる自動化された無差別な単純攻撃は, セッションの確立を防止することで防ぐことができる. また, SSL, SSH などと併せて用いた場合には, SSL, SSH のソフトウェア脆弱性の問題を防止したり, 処理量の多いこれらのプロトコルに対する不当なアクセスを制限し, サーバの処理負荷を軽減できる. これらのことはサーバ計算機の全体としての安全性を向上させ, SSL, SSH を利用したシステムのより安全な運用が可能となる.

4.2 TAP によっては防げないこと

TAP は TCP コネクションの確立を制御することで, システム全体の安全性を向上させることを目的とするプロトコルであり, TCP コネクションに相互認証や通信路の機密性・完全性を提供するものではない. つまり, クライアントとサーバの通信路上のノードに侵入したりしてサーバとクライアントの通信の間に入って行うセッションハイジャックや, サーバになりすましてクライアントを接続させる中間者攻撃を防ぐことを目的とはしていない. ただし, これらの攻撃は一般的に限定された環境の攻撃者のみが可能である. そして TAP の導入時にはサービスそのものが秘匿されているため, これらの攻撃の発生する可能性をより一層低下させることができる.

4.3 システム適用時の留意点

TAP をシステムに適用する際の留意点として, ロードバランサー [9] との関係がある. これは TAP クライアントと TAP サーバの間に L7 スイッチ/水平負荷分散装置が存在する場合に, 全てまたは一部の SYN パケットが TAP サーバに配送されないため, 認証処理がうまく動作しない可能性があるというものである. この場合ロードバランサーの設定変更などが必要となる.

4.4 性能評価

現仕様に準じた試作実装の性能評価を通じて, 本方式の実用性を検討する. 試作実装は, TAP サーバ (OS:linux, CPU:PenⅢ550MHz, メモリ:512MB) , TAP クライアント (OS:NetBSD, CPU:PenⅢ700MHz, メモリ:768MB) 上で動作している. 通常の TCP コネクション確立処理と比較して, 7.5ms 程度の遅延があるが, ネットワークの遅延やアプリケーションの処理時間と比較して実利用で意識される値ではないと考えられる. また, 現状でも 124 件/秒の TCP コネクションが確立可能であるがまだチューニングの余地は残されている.

4.5 関連技術

現在, port knocking[8] などと呼ばれる運用技術が提案されており, このような技術を利用することでも単純攻撃の防御が可能である. このような技術は, 認証情報を格納した複数の SYN パケットを受信した際に, 通信コネクションを確立するための SYN パケットを受け入れるようフィルタリングルールを変更するものである. それに対し TAP は, 認証情報を格納した SYN パケットに対する SYN/ACK パケットの返答処理を制御するものであり, TCP コネクションの確立処理と認証処理を一体化させることで安全性をさらに向上させている.

5. まとめ

本論文では, TCP コネクションの確立方法を制御することで, ソフトウェア脆弱性への耐性やポートスキャンからのサービス秘匿といった機能を実現する技術 TAP を示し, 実用性の検討も行った. 今後は, 計算機の処理能力と性能の相関やサービス不能攻撃へのシステムの耐性を評価することで, 様々な利用シーンにおける処理性能と安全性を検討する予定である.

参考文献

[1] A.O.Freier,P.Karlton,P.Kocher,“The SSL Protocol Version 3.0 ”, Internet Draft, Transport Layer Security Working Group, Nov 1996.

[2] T.Ylonen,“The SSH (Secure Shell) Remote Login Protocol ”, Internet Draft, Network Working Group, Nov 1995.

[3] CERT Advisory CA-2003-24 Buffer Management Vulnerability in OpenSSH, `http://www.cert.org/advisories/CA-2003-24.html`, Sep 2003.

[4] CERT Advisory CA-2003-26 Multiple Vulnerabilities in SSL/TLS Implementations , `http://www.cert.org/advisories/CA-2003-26.html` , Oct 2003.

[5] 梅澤 健太郎, 高橋 俊成, 鬼頭 利之, サービスを秘匿する TCP コネクション確立方法の提案, 暗号と情報セキュリティシンポジウム 2004, Jan 2004.

[6] M.Philip,「マスタリング TCP/IP 応用編」, オーム社, 1998.

[7] J.Postel,“TRANSMISSION CONTROL PROTOCOL ”,RFC793, Sep 1981.

[8] M.Krzywinski, “Port Knocking - Network Authentication Across Closed Ports”,SysAdmin Magazine, June 2003.

[9] Tony Bourke, “サーバ負荷分散技術 ”, オライリー・ジャパン, 2001.