渡り歩き検出方法の検討

竹尾 大輔† 渡邊 晃‡

†‡名城大学大学院理工学研究科 〒468-8502 愛知県名古屋市天白区塩釜口 1-501
E-mail: †m0432028@ccmailg.meijo-u.ac.jp, ‡wtnbakr@ccmfs.meijo-u.ac.jp

あらまし 近年，ネットワーク上の不正アクセスが増加傾向にある．クラッカーが不正アクセスを行う場合，多段の踏み台ホストを経由する“渡り歩き”（Island Hop）を行っているケースが多い．本研究では，踏み台ホストにおいて，リモートログインパケットの受信と TCP コネクションパケットの送信を短時間の間に検出することにより渡り歩きを検出する方式の提案を行う．提案方式を実装し，実験による有効性の確認と性能の評価結果を示す.

キーワード 渡り歩き，不正アクセス，侵入検知，ネットワークセキュリティ

Researches on Island Hop Detection Method

Daisuke TAKEO† and Akira WATANABE‡

†‡ Graduate School of Science and Technology, Meijo University
1-501 Shiogamaguchi, Tenpaku-ku, Nagoya-shi, Aichi, 468-8502 Japan
E-mail: †m0432028@ccmailg.meijo-u.ac.jp, ‡wtnbakr@ccmfs.meijo-u.ac.jp

Abstract In recent years, illegal access on network is increasing. When crackers access the target host illegally, they usually hop through many hosts witch we call it "Island Hop". In this paper, we propose the Island Hop detection method by detecting reception of remote login packets and transmission of TCP connection packets in a short time. We implement the proposed method, confirm the validity and show the evaluation results of the performance by experiment.

Keywords Island Hop, Illegal Access, Intrusion Detection, Network Security

1. はじめに

近年，ネットワーク上の不正アクセスが増加傾向にあり[1]，内外からの犯罪の脅威にさらされている．不正アクセスや攻撃に対して，ファイアウォールなどのセキュリティ機器の導入や，ホストの要塞化によるセキュリティ対策が取られているが，不正アクセスを完全に防ぐことは非常に困難である．

ネットワーク上の不正アクセスを発見するシステムとして，IDS（Intrusion Detection System；侵入検知システム）がある．IDS により，ネットワークを流れる通信やホストに対するアクションを監視し，不正なアクセスを検知することができる．システムに対する攻撃を監視・記録して得られた記録を分析することにより，存在する脅威の程度を確認し，セキュリティポリシやシステムの設定を見直すことができる.また，ユーザに監視システムの存在を知らしめて不正行為を抑制することもできる．

しかしながら，このようなセキュリティ対策技術を導入してもクラッカーは様々な不正アクセスを試みる．一般にクラッカーが不正アクセスを行う場合，多段の踏み台ホストを経由する“渡り歩き”（Island Hop）を行っているケースが多く見られる．渡り歩きは主にリモートログインなどの TCP サービスが用いられる場合が多い．個々のリモートログインが正常なものである場合，IDS ではそれ自体を不正と見なすことは難しい．また，管理者が管理目的で渡り歩きを行っているのか，クラッカーが不正な渡り歩きを行っているのか，判断材料が無いためにこれらを区別することも難しい．この問題を解決するには，渡り歩き自体を検出することと，渡り歩きが正常であるか不正であるかを判断することが求められる．

従来の渡り歩き検出方法として，Telnet のデータ一致検出方式が挙げられる[2],[3]．この方法は，Telnet による渡り歩きでは踏み台ホスト上でデータが同一の送受信パケットが発生していることに着目したものである．しかし，Telnet による渡り歩きのみを検出対象としているため，検出できる渡り歩きのパターンに制限がある．そこで本研究では，より幅広い渡り歩き検出が可能な，コネクション検出方式の検討を行う．これは，クラッカーがリモートログインしている踏み台ホストから別のホストへアクセスするときに TCP コネクションが確立されることに着目したものであり，アクセスの種類に依らない渡り歩き検出が可能となる．

以降，2 章で従来の渡り歩き検出方法を説明し，3 章で提案方式の概要を説明する．4 章では実装について述べ，5 章で評価を行う．そして最後に 6 章でまとめる．

2. 従来の渡り歩き検出方法

本章では，2.1 節で本研究で想定する渡り歩きモデルを定義し，2.2 節では従来のデータ一致検出方式について説明する．

2.1. 渡り歩きモデル

本研究で想定する渡り歩きモデルのネットワーク構成を**図 1** に示す．渡り歩きモデルの構成要素として，Attacker, Foot Hold, Target がある．Attacker（攻撃者ホスト）は渡り歩きを行うホストである．Foot Hold（踏み台ホスト）は Attacker がリモートログインするホストである．Target（ターゲットホスト）は Attacker が Foot Hold を介して最終的にアクセスするホストである．各ホストはネットワークによって接続されており，Attacker はクライアント型のホスト，Foot Hold と Target はサーバ型のホストを想定している．このとき Attacker が Foot Hold を介して Target にアクセスすることを，渡り歩きと定義する．

2.2. データ一致検出方式の概要

データ一致検出方式では，Telnet による渡り歩きが検出できる．**図 2** は，Attacker が Foot Hold に Telnet でログインし，さらに Foot Hold を介して Target にログインした場合の例を示している．Target へのログイン完了後は，Attacker と Target 間のデータ交換を Foot Hold が仲介する．即ち，Attacker から，送信元が Attacker，宛先が

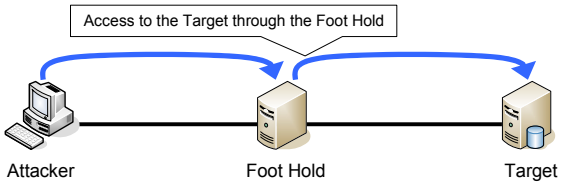


図 1　渡り歩きモデル

Fig.1　Island Hop model.

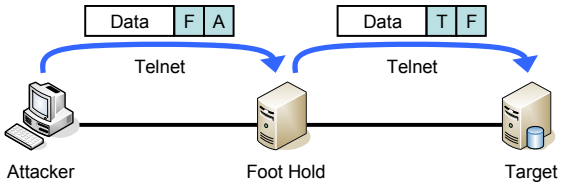


図 2　データ一致検出方式

Fig.2　Data agreement detection method.

Foot Hold の Telnet パケットが送信されると，このパケットを Foot Hold が受信し，Foot Hold から送信元が Foot Hold，宛先が Target，データは受信パケットと同一の Telnet パケットが，Target へと送信される．データ一致検出方式では，IP アドレスは異なるが，データは同じである送受信パケットが Foot Hold でほぼ同時に発生していることに着目し，Foot Hold において送受信パケットを監視することで渡り歩きを検出する．

しかし，データ一致検出方式には以下のような 3 つの課題がある．

① 渡り歩きが検出できるのは，Attacker・Foot Hold 間で TCP コネクションが確立された後である．これは不正な渡り歩きを検出する場合に，不正なコネクションが確立されてからでないと検出できないことを意味している．

② Telnet パケットのデータを比較しているため，SSH などの暗号化されたリモートログインを用いた渡り歩きの検出ができない．

③ Foot Hold から Target へのアクセスが FTP などのリモートログイン以外の方式を用いていると渡り歩きを検出できない．

3. 提案方式

データ一致検出方式では，連鎖状に Telnet によるリモートログインを行う渡り歩きを想定したが，Attacker から（Target の直前の）Foot Hold まではリモートログインでアクセスし，そこか

ら Target へはリモートログインだけでなく FTP などのサービスを利用することも考えられる．提案方式では，Attacker が Foot Hold へとリモートログインしており，そこから Target へとリモートログインを含めた様々な TCP サービスでアクセスするという状況を想定し，これを Foot Hold 上で検出するものである．この方式をコネクション検出方式と呼ぶことにする．

図３にコネクション検出方式を示す．既に Attacker と Foot Hold の間でリモートログインのコネクションが確立している．リモートログインとしては，SSH，Telnet，rlogin などがありうる．ここで，Foot Hold がリモートログインの通信パケットを受信した後の極短い間に Foot Hold から Target に新たな TCP コネクション確立要求を送信する場合，Attacker が Foot Hold に TCP サービスを起動するコマンドを送信した可能性があり，渡り歩きの可能性があると言える．

この原理に従い，コネクション検出方式に必要な 2 つの処理を個別に説明する．

(1) パケット受信処理の監視

Foot Hold が受信する TCP パケットのうち，Telnet や SSH, rlogin などのリモートログインの通信パケットを検出する．検出した時刻をリモートログイン受信記録として保存していく（**図４**）．

検出対象としては，TCP パケットの PSH フラグを用いる．なぜなら，リモートログインの通信パケットはローカルホストで入力された 1 文字分のデータであり，受信したらすぐにアプリケーションに渡す必要があるために，PSH フラグがセットされているからである．ここでリモートログインの SYN パケットを受信したとしても，それは Attacker から Foot Hold へのリモートログイン自体のコネクション確立要求であるので，この SYN パケット受信直後に渡り歩きに至ることはない．しかしリモートログインのコネクションを管理する目的で SYN，FIN パケットを監視することは有用である．

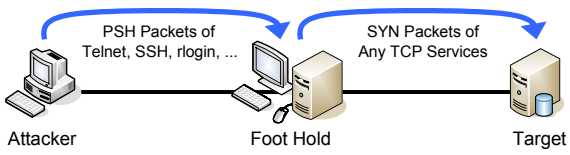


図３　コネクション検出方式

Fig.3　Connection detection method.

コネクション検出方式ではリモートログインの通信パケットのデータ内容を参照しないため，監視対象を Telnet 以外にも広げることができ，データ一致検出方式の課題②が解決できる．

(2) パケット送信処理の監視

Foot Hold が送信する TCP パケットのうち，あらゆる TCP サービスの SYN フラグを検出する．パケット送信時にリモートログイン受信記録を参照し，現在時刻と比較し，一定時間内に SYN フラグの立っているパケットの送信が行われると，渡り歩きと判断する（**図５**）．

検出対象が SYN パケットのみでよい理由は，Attacker からのコマンドを受けて Foot Hold が Target に対して TCP サービスを起動する場合，必ず初めにコネクション確立要求を送信するので，これを検出すればよいからである．宛先 IP アドレスが Attacker 自身である送信パケットは対象としない．Attacker が Foot Hold を介して自分自身にアクセスしたとしても，それは渡り歩

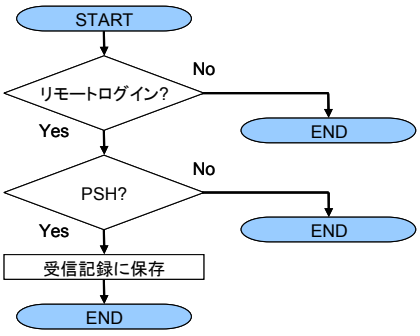


図４　パケット受信処理監視フローチャート

Fig.4　Flow chart of packet receive process.

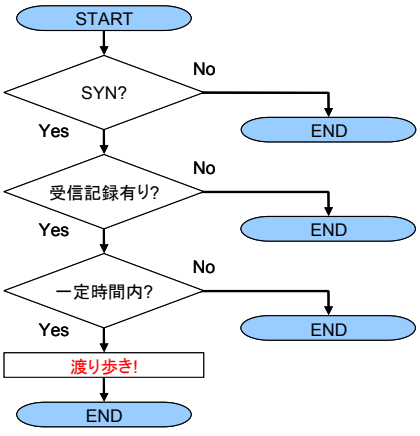


図５　パケット送信処理監視フローチャート

Fig.5　Flow chart of packet send process.

きではないからである．

SYN パケットを検出することで，データ一致検出方式の課題①が解決できる．そしてあらゆる TCP サービスを検出対象とすることで，データ一致検出方式の課題③が解決できる．

図 6 にコネクション検出方式の処理の流れを示す．まず Attacker が Foot Hold へリモートログインするためにコネクションの確立を行う．このときやり取りされるパケットは監視対象ではない．その後のリモートログインパケットの監視を行いつつ，Foot Hold から新たな TCP コネクションが確立されようとするのを監視する．リモートログイン通信パケットの受信とコネクション確立要求の送信との間の時間が十分短ければ，この状況は“渡り歩き”である．

4. 実装

本章ではコネクション検出方式の実装方法について述べる．

コネクション検出方式を実現するためには，全送受信パケットの監視を行う必要がある．送受信パケットの監視は libpcap などのパケットキャプチャライブラリ，あるいは FreeBSD の BPF（BSD Packet Filter）というパケットキャプチャ機能を使用してキャプチャすることで可能であるが，これらで得られるパケットデータはオリジナルパケットのコピーであり，パケットの取りこぼしが発生する可能性がある．本方式は OS のカーネルに組み込むことで実現することにする．これにより送受信パケットを完全にリアルタイムに監視でき，パケットを取りこぼすこともなく，パケットの送受信が発生した時刻の正確な差を計測することが可能となる．また本方式はホスト上に実装するため，ホスト型 IDS の形式を取っているが，カーネル内部で直接パケットを扱えるため，渡り歩き検出時にパケットを破棄するなどの操作も可能である．

図 7 に実装箇所を示す．実装する OS は，オープンソースであり，IP 層の処理を含めたネットワークコードに関する資料が多い FreeBSD を選択した．送受信パケット監視処理をそれぞれモジュール化し，適当な箇所で呼び出されるようにする．送受信パケットの監視は IP 層で行う．本方式はパケットの TCP ヘッダ（宛先ポート番号フィールド，コントロールフラグフィールド）を参照するため，トランスポート層でパケットを監視することも考えられるが，Attacker や Target の特定のために送信元 IP アドレスと宛先

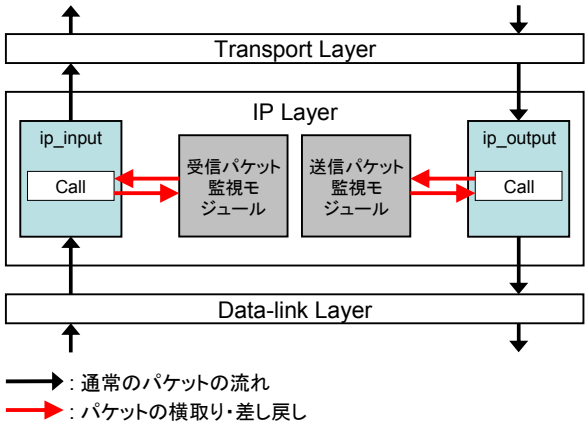


図 7　送受信パケットの監視位置

Fig.7　Observation points of send and receive packets.

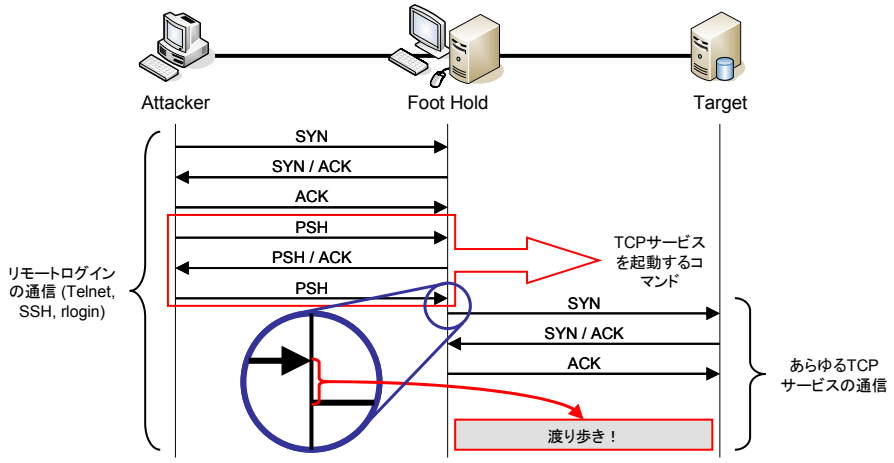


図 6　コネクション検出方式の処理の流れ

Fig.6　Process flow of connection detection method.

IP アドレスも監視する必要があり，IP 層を選んだ．

図 7 において，ip_input は，下位層から渡された受信パケットを検査し，上位層へ渡す処理である．受信パケットの監視は上位層へ渡す前に行う．ip_output は，上位層から渡された送信パケットを完成させ，下位層へ渡す処理である．送信パケットの監視は下位層へ渡す前に行う．送受信パケット監視モジュールは，パケットを横取りして参照したあとに差し戻すだけであるので，IP 層本来の処理を変更する必要はない．

5. 評価

本章では，コネクション検出方式の実装および動作実験を行い，その有効性確認と性能計測を行う．そしてデータ一致検出方式とコネクション検出方式を比較して評価と考察を行う．

5.1. 評価環境

評価環境は，渡り歩きモデルに General（第三者ホスト）を加えたネットワーク環境を用いた（**図 8**）．各ホストは 100BASE-TX の LAN によって接続されており，Foot Hold に渡り歩き検出機能を実装した．Foot Hold では SSH，Telnet，rlogin，FTP サービスを，Target では Telnet サービスを起動させた．

5.2. 性能測定結果

渡り歩きモデルに従って渡り歩きを行い，Foot Hold で検出できるかどうかを実験し，その性能を測定した．Attacker から Foot Hold へのリモートログインには SSH，Telnet，rlogin を，Foot Hold から Target への TCP 通信には Telnet を用いたが，いずれのリモートログインにおいても渡り歩きを検出できることを確認した．

渡り歩きが検出できるまでの時間（PSH を受信してから SYN を送信するまでの時間）は**表 1** のとおりである．表 1 において Telnet 直接起動とは，接続先の IP アドレスをパラメータとして Telnet を起動した場合であり，Telnet 事前起動とは，Telnet 自体を起動した後に接続先を Target として接続した場合である．

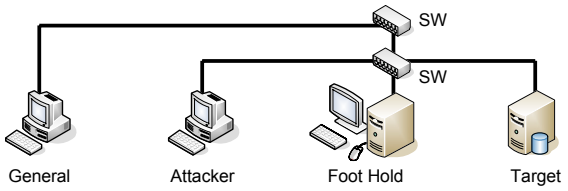


図 8　評価環境

Fig.8　Valuation environment.

リモートログインによって若干異なるが，Telnet 直接起動では 13～15 ミリ秒程度（13500～15000 マイクロ秒），Telnet 事前起動では 6～8 ミリ秒程度（6500～8000 マイクロ秒）の間に SYN パケットが送信されていることがわかる．

5.3. 他処理への影響

渡り歩き検出機能を実装した場合に，他の処理へ与える影響がどの程度あるかを実験した．Foot Hold に検出機能を実装した時と未実装の時において，General から Foot Hold へ，500MB の FTP ファイルのダウンロードとアップロードを行ったときの転送時間を計測した（**表 2**）．

表 2 からわかるように，コネクション検出方式による渡り歩き検出機能を実装した場合でも，ネットワーク処理能力の低下はほとんど発生しなかった．

5.4. 検出方式の比較

表 3 に，データ一致検出方式とコネクション検出方式との，機能面からの比較を示す．

データ一致検出方式では Telnet のリモートログインによる渡り歩きしか検出できないが，コネクション検出方式では，Attacker から Foot Hold への通信はリモートログイン，Foot Hold から Target への通信は全ての TCP サービスである渡り歩きが検出できる．

データ一致検出方式では Telnet のコネクションを確立しなければ渡り歩きを検出できないが，コネクション検出方式ではコネクションを確立

表 1　渡り歩き検出までの時間

Table.1　Time to Island Hop detection.

	Telnet直接起動	Telnet事前起動
SSH -> Telnet	13693.20(usec)	7228.05(usec)
Telnet -> Telnet	14985.65(usec)	7952.05(usec)
rlogin -> Telnet	15018.75(usec)	6553.55(usec)

※ 20回試行の平均

表 2　500MB のファイルの FTP 転送時間

Table.2　FTP transmission time of 500MB file.

	実装時	未実装時
ダウンロード	45.1515(sec)	45.1520(sec)
アップロード	45.9155(sec)	45.9865(sec)

※ 20回試行の平均

表３　検出方式の比較

Table.3　Comparison of the detection methods.

		データー致検出方式（旧提案方式）	コネクション検出方式（新提案方式）
渡り歩き検出		可能	可能
	対象となるA-F間の通信	Telnetのみ	Telnet, SSH, rloginなど全てのリモートログイン
	対象となるF-T間の通信	Telnetのみ	全てのTCPサービス
検出できるタイミング		コネクション確立後	コネクション確立前
リアルタイム性		高い	高い

(注)
A-F間：Attacker・Foot Hold間
F-T間：Foot Hold・Target間

する前に検出できる．

両方式共に OS のカーネル内部で監視するため，検出のリアルタイム性は高い．

5.5.　今後の課題

コネクション検出方式の採用により，渡り歩き検出対象の幅を広げることができた．しかし，送受信パケット発生時刻の時間差のみを手がかりとしているため，大量の送受信パケットが発生すると Attacker の特定は困難となる可能性がある．よって，渡り歩き検出機能で Attacker を特定せず，疑わしい Attacker のリストを管理者に報告するなど，最終的な判断は人間に任せる必要があると考えられる．

不正な渡り歩きを検出するためには，本方式に加えて渡り歩きの正常・不正の判断をしなければならない．これに関しては，事前に正常・不正パターンを定義した IP アドレスリストなどを用意することで，渡り歩きの正常・不正の判断ができると考えられる．

リモートログインの PSH パケットを受信してから一定時間の間に SYN パケットが送信されるまでの監視時間を適切な値に設定する必要がある．コマンドを受理してから実際にコネクションを確立するまでの時間の統計を取るなどが必要になると思われる．また，この時間はホストのマシンスペックに左右されることを考慮しなければならない．

6.　まとめ

本研究では，踏み台ホストからの新たな TCP コネクション確立要求の送信を監視することにより渡り歩きを検出する，コネクション検出方式について検討した．コネクション検出方式を実装して動作確認を行い，汎用的に渡り歩きが検出できることを示した．コネクション検出方式を適用することによる他のネットワーク処理への影響はほとんどなく，有効な方法であることを示した．今後の課題としては，Attacker の特定，不正な渡り歩きの検出，適切な監視時間の算出などを行う必要がある．

文　献

[1] 2003 CSI/FBI Computer Crime and Security Survey，Computer Security Institute. http://i.cmpnet.com/gocsi/db_area/pdfs/fbi/FBI2003.pdf

[2] 竹尾大輔，渡邊晃，“GSCIP を構成する渡り歩き検出機能の仕組みの検討”，第 66 回情報処理学会全国大会, 分冊 3, no.5V-2, pp.481-482, March 2004.

[3] 竹尾大輔，渡邊晃，“FPN における渡り歩きの検出方法の検討”，情報処理学会研究報告 2004-CSEC-26，pp.275-280，July 2004.

[4] Snort．http://www.snort.org/

[5] 日本 Snort ユーザ会．http://www.snort.gr.jp/

[6] J. Postel，J.K. Reynolds：Telnet Protocol Specification，RFC854，http://www.ietf.org/rfc/rfc0854.txt，May 1983.

[7] Tatu Ylonen，Chris Lonvick，“SSH Protocol Architecture”，Internet-Drafts，http://www.ietf.org/internet-drafts/draft-ietf-secsh-architecture-16.txt，June 2004.

[8] Tatu Ylonen，Tero Kivinen，Timo Rinne，Sami Lehtinen，“SSH Connection Protocol”，Internet-Drafts，http://www.ietf.org/internet-drafts/draft-ietf-secsh-connect-19.txt, June 2004.

[9] 大塚丈司，白石善明，森井晶克，“センター管理型不正アクセス検知システムの提案”，情報処理学会研究報告 2002-CSEC-18，pp.39-44，July 2002.

[10] 岡本忠士，白石善明，大家隆弘，森井昌克，“なりすましに対する不正侵入検知システム (IDS-M)”，情報処理学会研究報告 1999-ICS-6，pp.39-46，July 1999.

[11] Constantine Manikopoulos，Symeon Papavassiliou，“Network intrusion and fault detection: A statistical anomaly approach”，IEEE Communications Magazine，vol.40，no.10，pp.76-82，Oct. 2002.

渡り歩き検出方法の検討

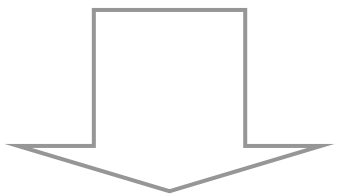
- Researches on Island Hop Detection Method -

名城大学大学院理工学研究科

竹尾大輔　　渡邊　晃

背景

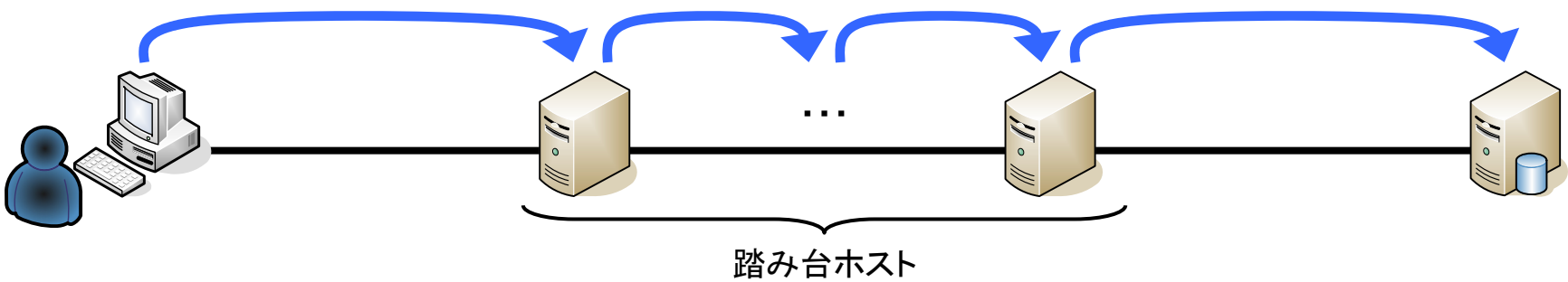
- 増加するネットワーク上の不正アクセス
  - FW導入，ホスト要塞化による対策
  - 完全には防げない
- IDSでネットワークを監視
  - 不正アクセスの発見，抑制



クラッカーは様々な不正アクセスを試みる

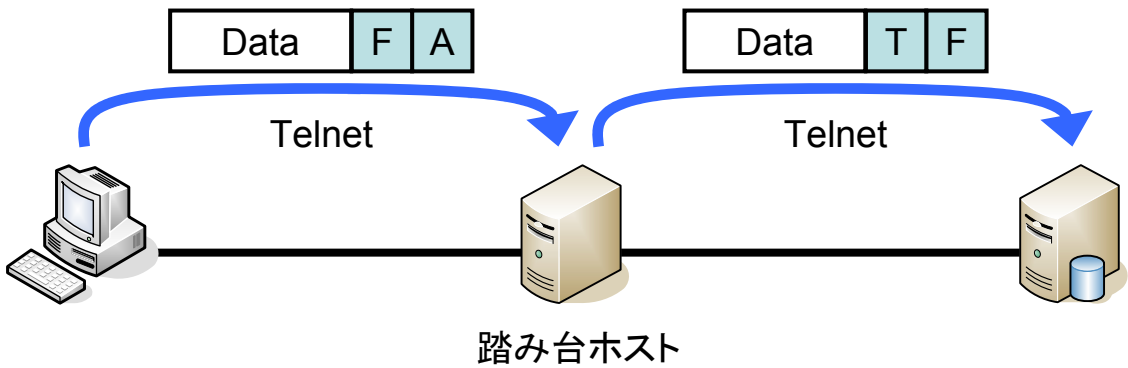
背景

- クラッカーは渡り歩きを行う
  - 多段の踏み台ホストを経由
  - 個々の通信の正常・不正の判断
    - 正常な通信を不正と見なすことは難しい
  - 渡り歩きの正常・不正の判断
    - 管理者による正常な渡り歩きか？クラッカーによる不正な渡り歩きか？
    - 判断材料が無い



既存技術：データ一致検出方式

- Telnetによる渡り歩きを検出
  - IPアドレスは異なるがデータは同一の送受信パケットが踏み台ホストでほぼ同時に発生している
  - 踏み台ホストで送受信パケットを監視



本研究の主題

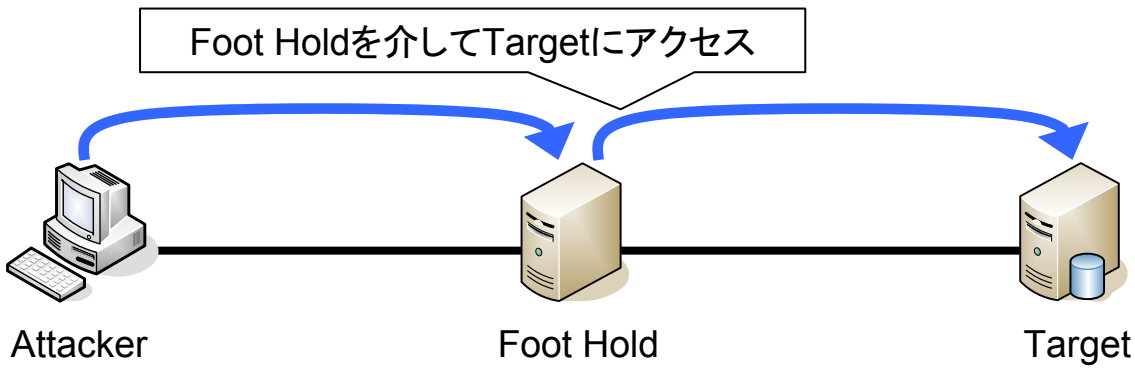
- データー致検出方式は検出対象が限定されている

踏み台ホストからTCPコネクションが確立されることに着目

- アクセスの種類に依らない，より幅広い渡り歩き検出が可能な“コネクション検出方式”を検討
  - 提案方式の性能評価実験

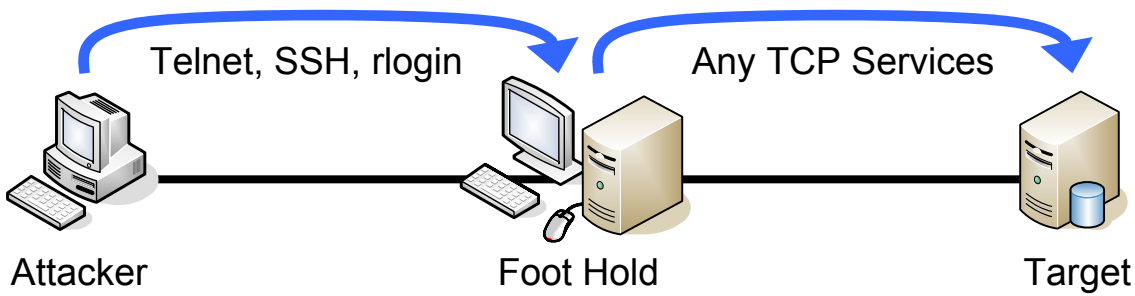
渡り歩きモデル

- 渡り歩きモデルの構成要素
  - Attacker（攻撃者ホスト）　：クライアント型
  - Foot Hold（踏み台ホスト）　：サーバ型
  - Target（ターゲットホスト）　：サーバ型

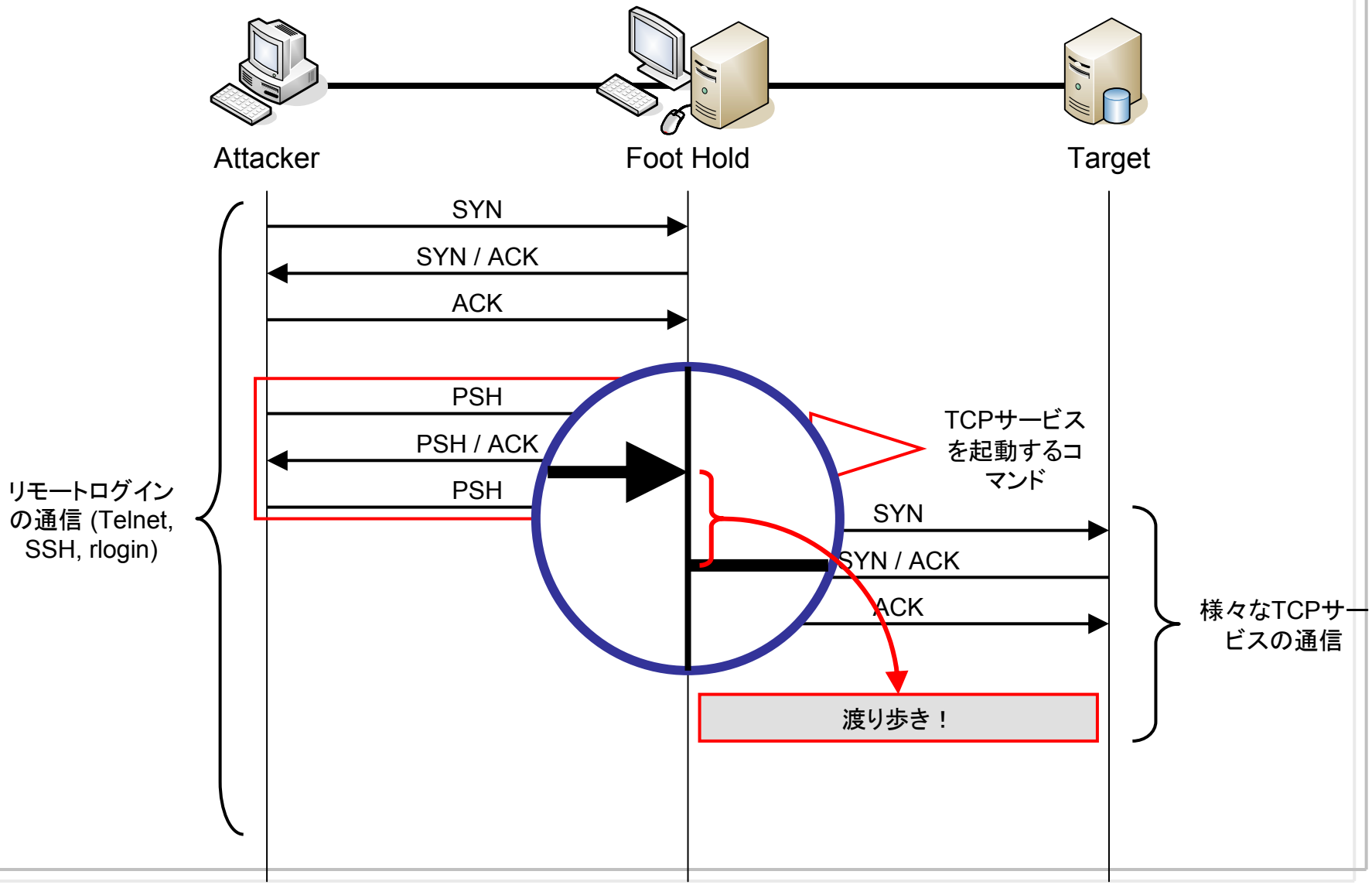


コネクション検出方式

- AttackerからFoot Holdへの通信
  - リモートログイン（SSH, Telnet, rlogin）
- Foot HoldからTargetへの通信
  - 様々なTCPサービス
- AttackerがFoot Holdにリモートログインし，様々なTCPサービスでTargetへアクセス
  - Foot Hold上で検出

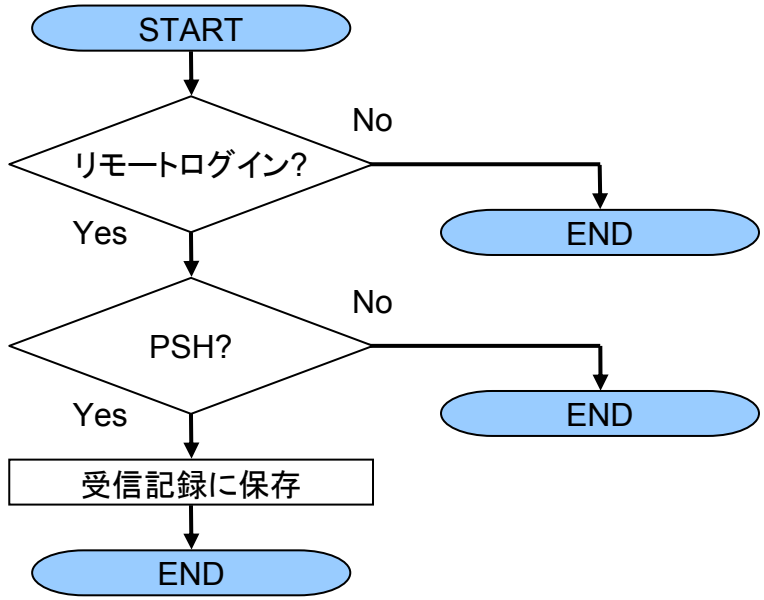


コネクション検出の流れ



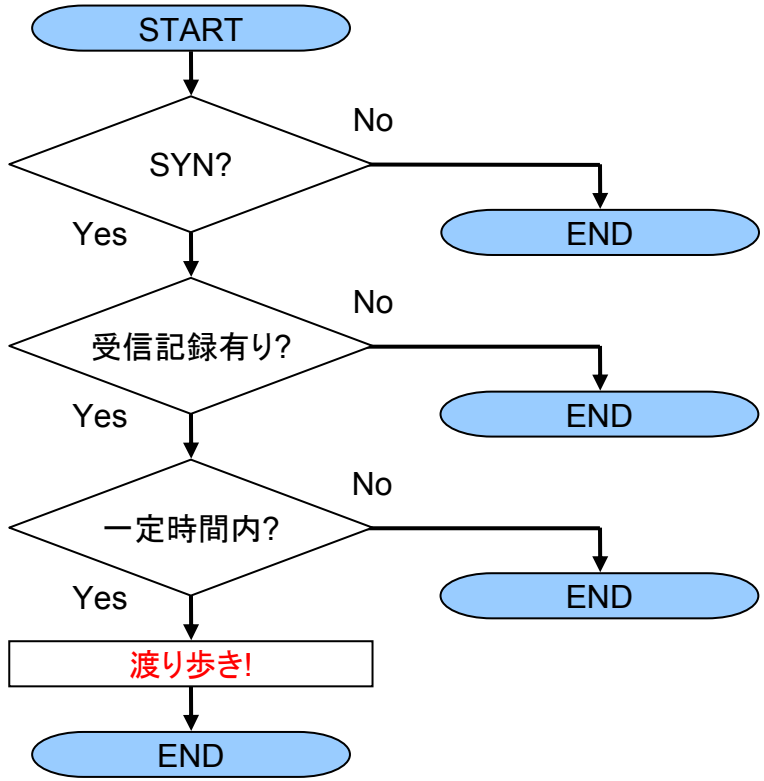
パケット受信処理の監視

- TCPパケットを監視，リモートログイン通信パケットを検出
- 検出対象はPSHパケットのみ
  - リモートログインの通信パケットはPSHフラグがセットされている
- 検出時刻をリモートログイン受信記録に保存
- リモートログインの種類に依らずに検出可能



パケット送信処理の監視

- TCPパケットを監視，TCPコネクション確立要求を検出
- 検出対象はSYNパケットのみ
- リモートログイン受信記録を参照，送信が一定時間内なら渡り歩き
- コネクション確立前に，様々なTCPサービスによる渡り歩きを検出できる

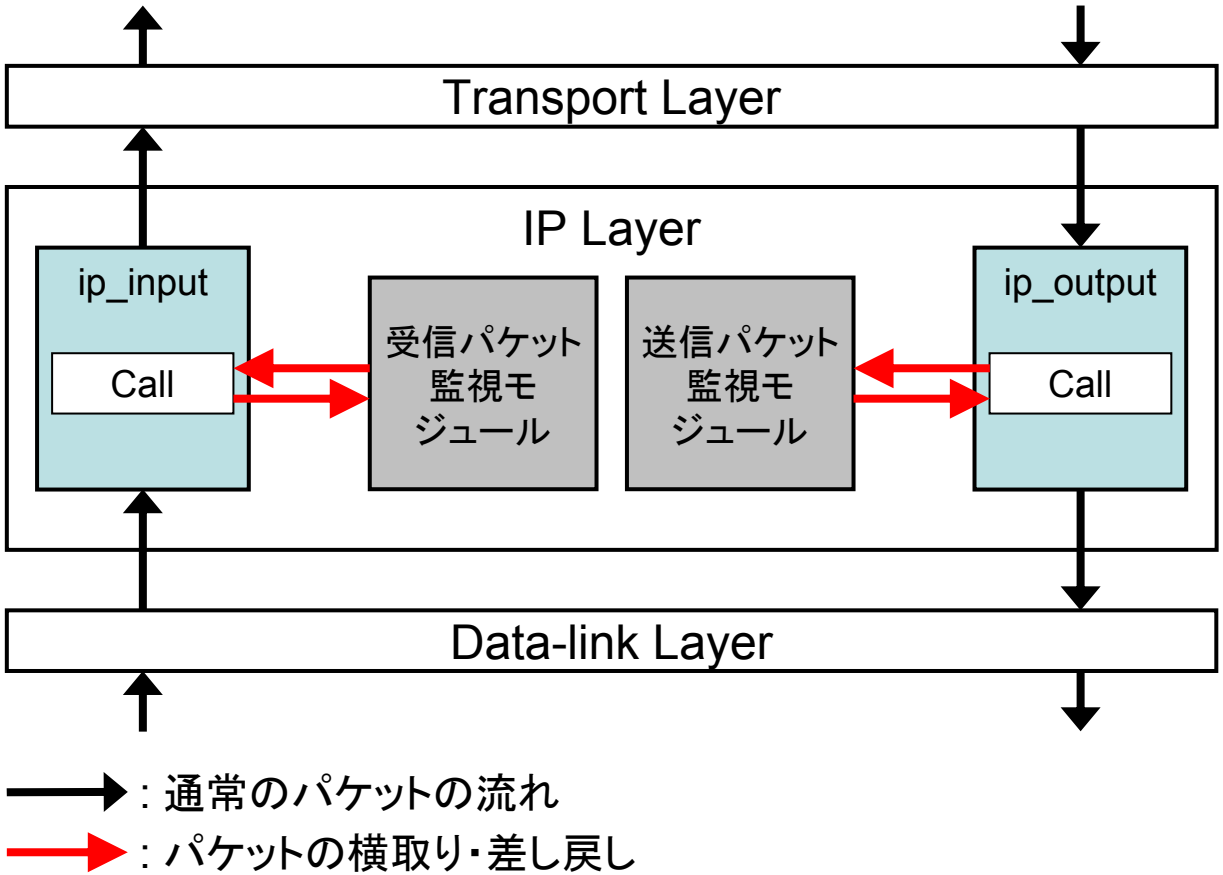


渡り歩き検出機能の実装

- リアルタイムに監視するために，OSのカーネルに実装する
  - イベントの取りこぼしが無い
  - 正確な検出時刻の比較が可能
  - 必要であればパケットの破棄も可能
- OSにFreeBSD 5.2.1Rを採用
  - ネットワークに関する資料が多い

送受信パケットの監視位置

- 送受信パケットの監視はIP層で行う



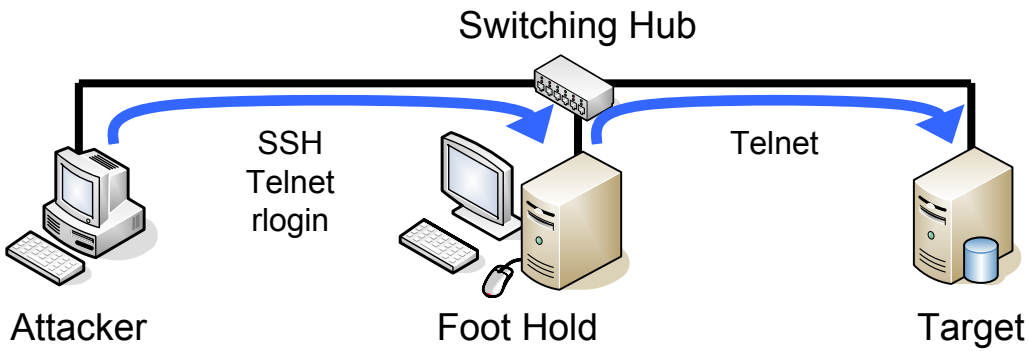
実験概要

- 実験項目
  - 検出機能の有効性とその性能の確認
  - 検出機能実装による他処理への影響の計測
  - 背景負荷付与時の渡り歩き検出時間の計測
- 評価環境
  - 100BASE-TXのLAN
  - Foot Holdに検出機能を実装

CPU	Pentium4 2.4GHz
メモリ	256MB
OS	FreeBSD 5.2.1R
起動サービス	SSH, Telnet, rlogin, FTP

性能の測定結果

- 検出機能の有効性とその性能の確認
  - 3種類のリモートログインによる渡り歩き
  - PSH受信とSYN送信の間の時間（渡り歩き検出時間）を計測

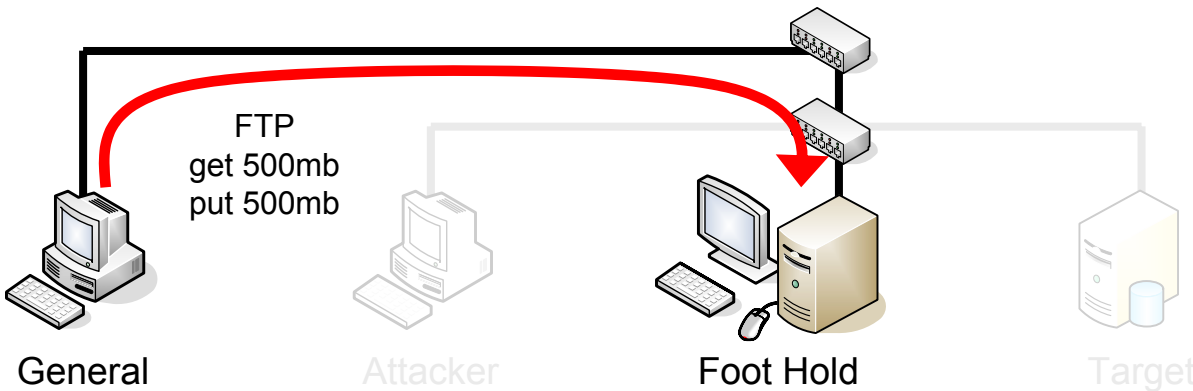


	Telnet直接起動	Telnet事前起動
SSH -> Telnet	13.69(msec)	7.23(msec)
Telnet -> Telnet	14.99(msec)	7.95(msec)
rlogin -> Telnet	15.02(msec)	6.55(msec)

※ 20回試行の平均

他処理への影響

- 検出機能の実装時/未実装時における，FTPを利用したダウン/アップロード時間測定
  - GeneralがFoot HoldにFTP接続
  - 500MBのファイルをダウン/アップロード

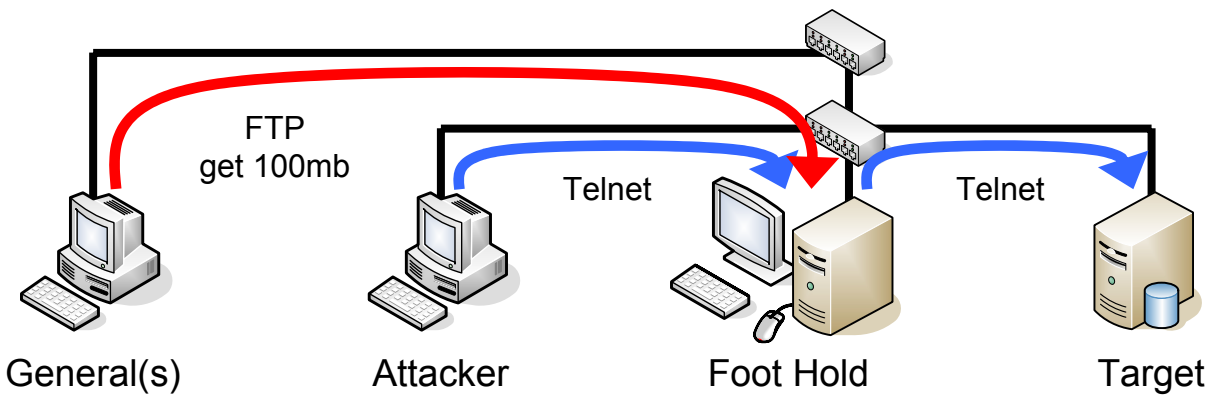


	実装時	未実装時
ダウンロード	45.15(sec)	45.15(sec)
アップロード	45.92(sec)	45.99(sec)

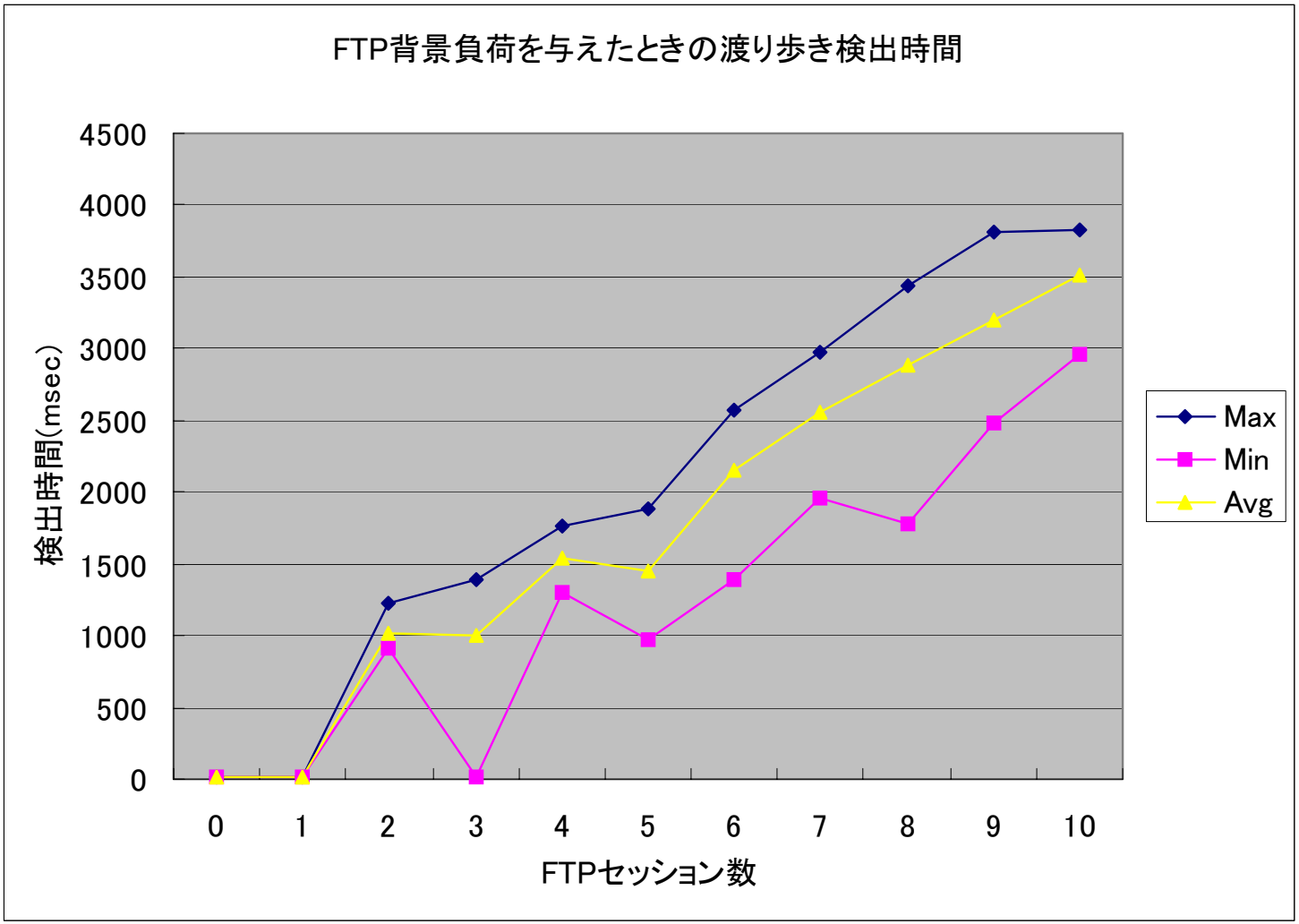
※ 20回試行の平均

背景負荷付与時の渡り歩き検出時間

- FTP接続数を0から10へ増加させていき，渡り歩き検出時間を計測
  - 連続してファイルをダウンロード
  - 複数のFTP接続は複数のGeneralから実行



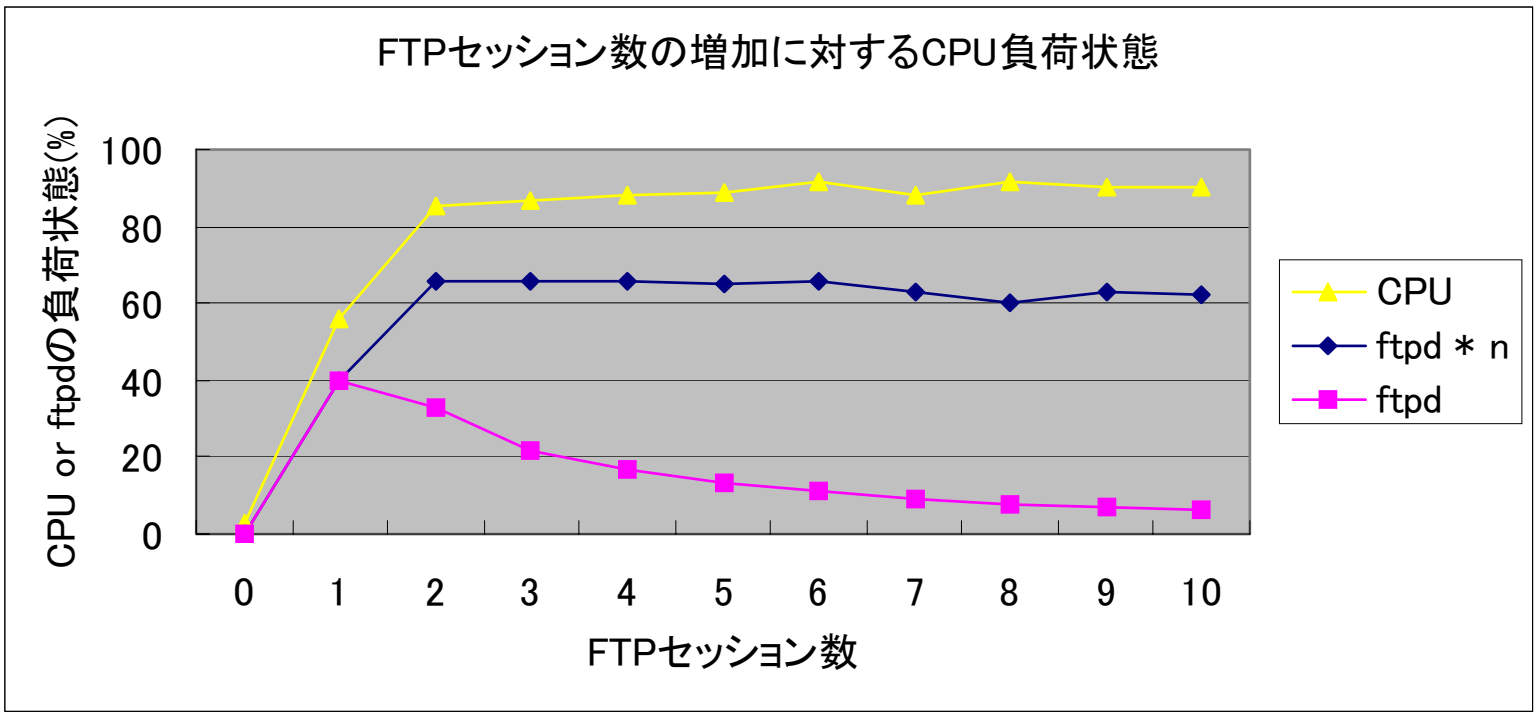
背景負荷付与時の渡り歩き検出時間



※ Avgは10回試行の平均

背景負荷付与時のCPU負荷状態

	0	1	2	3	4	5	6	7	8	9	10
CPU (%)	2.7	55.6	85.5	86.6	88	88.8	91.4	88.4	91.3	90	90.3
ftpd * n (%)	0	40.2	32.7	22	16.5	13	11	9	7.5	7	6.2
ftpd (%)	0	40.2	65.4	66	66	65	66	63	60	63	62



※ ftpdはプロセス数の合計

課題

- ネットワークトラヒックやCPU負荷状態により渡り歩き検出時間が変化
  - マシンスペックによっても変化する可能性あり
  - 渡り歩きと判断する監視時間の設定を微調整する必要がある
    - 監視時間が短いと
      - 高負荷状態で渡り歩きが検出できない
    - 監視時間が長いと
      - Attackerの候補が増加し，Attackerを特定しづらい
  - 対策案
    - 負荷状態に合わせて動的に監視時間を設定

まとめ

- 普遍的な渡り歩き検出方法を検討した
  - コネクション検出方式の有効性を確認した
  - 導入による他処理への影響はほとんど無い
  - ネットワークトラヒックやCPU負荷状態により検出時間が変化した
- 今後の課題
  - 適切な監視時間の設定
  - Attackerの特定
  - 不正な渡り歩きの検出

お わ り

Hokkaido CSS2004 Oct.20-22

Computer Security Symposium 2004