Vol.2012-CSEC-59 No.9
2012/12/14

TCPによるIP Spoofing Defense

内海 哲史[1,a)]　Salahuddin Muhammad Salim Zabir[2,b)]　加藤 靖[1,c)]

概要：2012 年 10 月 PC 遠隔操作事件が発覚し，それに伴う冤罪も明らかになった．今回のような犯罪の場合でも，犯行者のホストの真の IP アドレスが分かれば，真犯人を特定できる可能性が高い．本稿では，（1）コネクション開始時，TCP のように 3 ウェイ・ハンドシェーキングを必ず行うこと，（2）OS のカーネルは（1）をチェックすること，（3）受け取ったデータパケットに対して必ず確認応答を返すこと，（4）OS のカーネルは（3）をチェックすること，（5）身に覚えのない確認応答を受け取ったホストは警告パケットを必ず返すこと，を提案し，IP レベルでのなりすましのできない安全なインターネットの実現を目指す．

1. はじめに

2012 年 10 月 PC 遠隔操作事件が発覚し，それに伴う冤罪も明らかになった．今回のような犯罪の場合でも，犯行者のホストの真の IP アドレスが分かれば，真犯人を特定できる可能性が高い．本稿では，（1）コネクション開始時，TCP のように 3 ウェイ・ハンドシェーキングを必ず行うこと，（2）OS のカーネルは（1）をチェックすること，（3）受け取ったデータパケットに対して，必ず確認応答を返すこと，（4）OS のカーネルは（3）をチェックすること，（5）身に覚えのない確認応答を受け取ったホストは，警告パケットを必ず返すこと，を提案し，IP レベルでのなりすましのできない安全なインターネットの実現を目指す．以下の章は，次のような構成である．まず，２章で関連研究について述べる． 3 章では，本研究における提案を述べる． 4 章では，提案の IP Spoofing Defense の有効性について検証する． 5 章で，提案の IP Spoofing Defense について議論する．最後に，6 章で本稿をまとめる．

2. 関連研究

今日のインターネットでは，なお IP Spoofing について深刻な問題がある．たくさんの IP Spoofing に対する対策が提案されているが，どれもなりすまし問題を根絶するに至っていない．本章では，これまで提案されている IP Spoofing に対する対策の概要をまとめる．

1 鶴岡工業高等専門学校
2 フランステレコム株式会社
a) u-satoshi@tsuruoka-nct.ac.jp
b) szabir@ieee.org
c) president@tsuruoka-nct.ac.jp

2.1 IP Spoofing Defense の分類

IP Spoofing Defense は，本質的に以下の 3 つのカテゴリーに分けることができる [1].

- ホストに基づく対策 : この対策は，エンドホストに実装され，エンドホストがなりすましパケットを認識できるようにすることが目的である．そのいくつかの対策はルータでも展開できるが，ホストに基づく対策はエンドホストを念頭に設計されており，特別なルータの機能を必要としない．一般的に，これらの対策はネットワークインフラの変更を必要とせず，普及が最も容易である．すべての送信元ホストが正直でない限り，送信元ホストに基づく対策は IP Spoofing を防ぐことができないので，本稿では特に，宛先のエンドホストに展開されるホストに基づく対策のみを紹介する．
- ルータに基づく対策 : この対策は，コアルータ，エッジルータまたは，その両方に実装される．この対策は，一般的に展開の面でより高いハードルがあるが，エンドホストになりすましパケットが到達するのを防ぐことができるので，非常に効果的である．ルータに基づく対策は，ほとんどの場合，エンドホストに適用できない．悪意のあるパケットがどこから来たのかトレースするなど，ルータにリアクティブ型の機能を持たせることもできる．
- ルータとホストによる対策 : この対策は，主に，パケットにマークすることによって実現される．ルータは，そのネットワークを通過するパケットをマークする．パケットがエンドホストに到達したとき，そのパケットの送信元アドレスフィールドに関わらず，エンドホストはそのマーキングを用いて，そのパケットがどこ

© 2012 Information Processing Society of Japan

を通ってきたのかをトレースすることができる．

ホストに基づく対策とルータに基づく対策の大きな違いは，End-to-End についての議論に関係する [2]．ホストに基づく対策は，明らかに End-to-End の原則に従っているが，ルータに基づく対策はそうではない．このことは，ホストに基づく対策は，より容易に実現できることを意味しており，ルータの複雑化を防ぐことができる．

本稿における提案の IP Spoofing Defense は，本質的には，ホストに基づく対策であるため，ホストに基づく対策についての関連研究を以下にまとめる．

2.2 ホストに基づく IP Spoofing Defense

ホストに基づく IP Spoofing Defense は，アクティブ型とパッシブ型に分類できる．アクティブ型は，エンドホストにプロービングや，プロアクティブな動作を要求する．一方，パッシブ型は，パケットの送信元をプローブすることなく，ローカルに集めることのできる情報のみに頼る．

2.2.1 アクティブ型

アクティブ型のホストに基づく対策には，暗号化による対策，アクティブプロービングによる対策，IP Puzzles が含まれる．

IPsec のような暗号化による対策 [3] では，ハンドシェイキングのとき，二つのホストの間で秘密鍵をセットアップする．さらに，一方のホストが受け取ったメッセージがもう一方のホストに送ったということを保証するために，二つのホストの間の通信は，暗号化される．アタッカー（攻撃者）は，ハンドシェイキングの完了に必要な返信を受け取らないので、コネクションを生成するパケットのなりすましに成功しない．同様に，アタッカーは秘密鍵を知らないので，すでに存在するコネクションのパケットになりすますことも失敗する．IPsec は，多くの場面で有効であり，なりすましを防ぎ，信頼性や完全性を保証するが，すべての場面で利用することは非現実的である．なぜなら，すべてのホストに IPsec によって通信することを要求することはできないからである．さらに，暗号化によって計算にかかるコストのため，ホストは多くのコネクションを管理できない．また，計算コストのため，ルータはすべてのコネクションの IPsec トンネルを作ることができない．

アクティブ型のプロービングによる対策にはいくつかのプローブがあり，アクティブ・オペレーティングシステム・フィンガープリンティングや IP 識別フィールドプローブ，TCP-specific プローブも含まれる．これらのプローブによる対策は、IP Spoofing Defense のために設計されたわけではないが，IP Spoofing Defense のためにも使うことができる．

NMAP [4] のようなツールを用いることによって，そのホストで動いている OS を知るためにそのホストをプローブすることは，なりすましパケットを検出するために役立つ．これらのツールは，注意深く作られたパケットをエンドホストに送信し，その応答を観測する．ほとんどの OS は一般的に標準的なプロトコルの仕様を実装しているが，実装の違いは測定できる．この実装の違いによって，そのホストで動いている OS をフィンガープリント（指紋採取）できる．ホストがアクティブに偽装された送信元ホストのフィンガープリントができ，そこで動いている OS が X であると分かり，一方，実際に受け取ったパケットによってパッシブ・フィンガープリント [5]， [6]， [7] することによって，OS が Y であることが分かれば，実際に受け取ったパケットはなりすましであると言える．しかし，フィンガープリントには，大きなオーバヘッドがかかる．ホストは，疑わしいパケット群を処理する前にその送信元を検証する必要があるとき，大量のプローブを送信し，疑わしいパケットをバッファしなければならない．さらに，アタッカーが，偽装されたアドレスのホストと同じ OS を使っているとき，フィンガープリントはなりすましパケットを識別できない．

一方，IP パケットの識別子フィールドを調べることによって，なりすましパケットを識別することもできる．疑わしいパケットを受け取ったあと，ホストは偽装された送信元にパケットを送り，その返信の識別子フィールドを観測できる．異なる IP スタックは，IP パケットの識別子フィールドに異なった値をセットする．例えば，あるホストはランダムな識別番号を選び，他のホストはすべてのパケットについて単純にインクリメントした識別番号を選ぶ．送信元のホストが単純にすべてのパケットについて識別番号をインクリメントすると仮定すると，疑わしいパケットがなりすましパケットでないならば，プローブに対する返信に付いた識別番号は，疑わしいパケットの識別番号に近い値になるはずである．そうでなければ，その疑わしいパケットはなりすましパケットである．しかし，送信元ホストがもっと複雑な方法で識別フィールドを設定するとき，疑わしいパケットがなりすましパケットかどうかを判断するのは極めて困難であるか，不可能である．

TCP-specific プローブ [8], [9] は，なりすましパケットから守るもうひとつの賢い方法である．アタッカーは、TCP のシーケンス番号を予測できることもあるので，単に TCP ハンドシェイクを要求することだけでは，なりすましの TCP パケットを防ぐのに十分でない．多くの OS はランダムシーケンス番号セレクションを用いるが，それらの OS が用いる擬似乱数ジェネレータは，十分ランダムではない．TCP-specific プローブでは，巧みに TCP 確認応答を生成する．なりすましパケットの送信元は確認応答を見ることができないので，宛先ホストは，TCP ウィンドウ・サイズの変更を促し，パケットの再送を促す確認応答を送信し，疑わしい送信元が正しく応答しているかを観測することができる．もし，疑わしい送信元がウィンドウ・サイ

© 2012 Information Processing Society of Japan

ズを変更せず，また，パケットを再送しなかった場合，その宛先ホストはそのパケットの送信元は偽装されていると判断する．

いくつかのサーバは，偽装された送信元アドレスにコネクションを開くのを防ぐため，SYN クッキーズ [10] を用いている．SYN クッキーズを用いる主な理由は，TCP ハンドシェイクをステートレスにすることによって，SYN フラッドの影響を緩和することである．サーバが SYN クッキーズを用いるとき，TCP の 3 ウェイ・ハンドシェイクが完了するまで，サーバはコネクションにリソースを割り当てない．最初に，サーバは，クライアントの初期 SYN パケットとタイムスタンプ，クライアントの MSS といった TCP ヘッダの情報のハッシュを含んだ，特別にエンコードされた初期シーケンス番号（クッキー）を付けて，SYN/ACK パケットを送信する．そして，サーバがクライアントの返信を受け取ったとき，サーバはそのシーケンス番号を確認し，クライアントのシーケンス番号がクッキーの番号＋１であったときのみ，コネクションを確立する．クッキーはサーバの秘密鍵を含んだハッシュを用いるので，アタッカーは正しいクッキーの値を推測することができない．しかし，パフォーマンスの問題や，TCP オプションを無視しなければならないことから，ホストの SYN キューがいっぱいになるまで，OS は一般的に SYN クッキーズを起動しない．少ないトラヒックでなりすますアタッカーは，SYN クッキーズの起動を避けることができる．管理者は強制的にすべてのコネクションに対して SYN クッキーズを有効にできるが，その副作用を認識しておかなければならない．

IP Puzzles [12] は，なりすましに対抗するもうひとつの方法である．サーバは IP Puzzle をクライアントに送信し，クライアントは計算タスクを実行することで，その問題を解く必要がある．サーバがクライアントから問題の答えを受け取ったときのみ，サーバはクライアントに接続を許す．IP Puzzles の主な目的は，悪意のあるホストが大量のパケットを送るのを禁止することにあるが，その別な効果として，アタッカーがなりすましパケットを送信するのを防ぐこともできる．アタッカーではなく，指定された送信元に IP Puzzle は送られるので，アタッカーは問題の答えを送ることができず，アタッカーがなりすますことを防げる．

2.2.2 パッシブ型

パッシブ型のホストに基づく対策は，入力トラヒックを受動的に観測することによってパケットがなりすまされているか，そうでないかを判断する．

Hop-Count Filtering （HCF） [13] [14] は，到着パケットのホップカウントを観測する． HCF は平常時のホップカウントを測定し，IP アドレスとホップカウントをマッピングする．もし，アタッカーがなりすましパケットをそのホストへ送ってきたとき，パケットのホップカウントはそのマップにより予想されるホップカウントと一致しない．ホップカウントはルートの変更によって変化するので，すべてのパケットに対するフィルタリングはホールスポジティブである．そのホールスポジティブを最小化するため，HCF は，ホップカウントが一致しないパケットの量があるしきい値を超えたときのみ，フィルタリングを開始する．このしきい値は，誤って正しいパケットをフィルタリングすることを防止するが，しきい値に達しない少ない量のなりすましパケットに対して効果的でない．さらに，インターネットにおけるホップカウントの範囲は狭いので，約 10%のなりすましパケットのホップカウントは正しいホップカウントと一致してしまうことが考えられる [13]．

3. 提案

本章では，TCP の持つ確認応答機能をベースにした IP Spoofing Defense について提案する．ここで，**IP Spoofing Defense とは，送信元 IP アドレスを偽装してデータパケットが送られてきたとき，そのことを宛先ホストが認識できること**とする．その目的を実現するため，すべてのインターネットアプリケーションとすべてのインターネットにおけるホスト，すべての中継ノードは，以下の【提案１】～【提案８】の機能を実装し，実行することを提案する．

【提案１】データパケットを送信するアプリケーション（ホスト A）は，コネクション開始時，TCP のように必ず SYN パケットを，データパケットを受信するホスト（ホスト B）に送信する（図 1）．

【提案２】SYN パケットを受け取ったアプリケーション（ホスト B）は，TCP のように必ず SYN/ACK パケットを SYN パケットの送信元 IP アドレス（ホスト A）に返す（図 2）．

【提案３】SYN/ACK パケットを受け取ったアプリケーション（ホスト A）は，TCP のように必ず確認応答を SYN/ACK パケットの送信元 IP アドレス（ホスト B）に返す（図 3）．

【提案４】データパケットを受け取るアプリケーション（ホスト B）は，データパケットを受け取った後，TCP のように必ずデータパケットの送信元 IP アドレス（ホスト A）に確認応答を返す（図 4）．

【提案５】SYN パケットを受け取ったアプリケーションを実行するホスト（ホスト B）の OS のカーネルは，そのデータパケットの送信元 IP アドレスに対して，必ず SYN/ACK パケットを返していることをチェックする（図 5）．

【提案６】SYN/ACK パケットを送信するアプリケーションを実行するホスト（ホスト B）の OS のカーネルは，その SYN/ACK パケットの送信元 IP アドレス（ホスト A）から，必ず確認応答を受け取っていることをチェックする（図 6）．

【提案７】データパケットを受け取るアプリケーションを実行するホスト（ホスト B）の OS のカーネルは，そのデー

© 2012 Information Processing Society of Japan

タパケットの送信元 IP アドレス（ホスト A）に対して，必ず確認応答を返していることをチェックする（図 7）．
【提案８】インターネットにおけるホスト（ホスト C）は，送信した覚えのない SYN パケットに対する SYN/ACK パケット，または，送信した覚えのない SYN/ACK パケットやデータパケットに対する確認応答を受け取ったとき，その SYN/ACK パケットまたは確認応答の送信元ホスト（ホスト B）に対して，警告パケットを送信する（図 8，図 9）．
【提案９】インターネットにおける中継ノード（ノード D）は，パケットの宛先 IP アドレスに対応するホストが存在しないとき，そのパケットの送信元ホスト（ホスト B）に ICMP Destination Unreachable パケットを必ず返す（図 10）．

ここで，本提案で，【提案５】，【提案６】，【提案７】のように，OS のカーネルが SYN/ACK パケットまたは確認応答の返信をチェックしているのは，UDP ソケットなどによって，SYN/ACK パケットまたは確認応答を返さない機能の利用を防ぐためである．

4. 検証

３章で提案した TCP に基づく IP Spoofing Defense により，宛先ホストが，いかにして，送信元 IP アドレスを偽装してデータパケットが送られてきたことを認識するかについて，検証する．ここで，**「正規のホスト」とは，３章で提案した機能をすべて実装，実行しているホストのこと**を言う．データパケットを送るホストをホスト A，データパケットを受け取り，その確認応答を返すホストをホスト B，その確認応答を受け取るホストをホスト C とする．
【検証１】送信元 IP アドレスが存在する正規のホストに偽装されたとき
【提案２】，【提案５】によって，SYN パケットを受け取るアプリケーションを実行するホスト B は，その SYN パケットに対して，SYN/ACK パケットを返すことが保証される．ホスト B は，真の送信元であるホスト A ではなく，偽装されたホストであるホスト C に SYN/ACK パケットを返す．ホスト C は，【提案９】によって，警告パケットをホスト B に返すことが保証され，ホスト B は，そのデータパケットの送信元 IP アドレスが偽装されていたことを認識できる．また，【提案４】，【提案７】によって，データパケットを受け取るアプリケーションを実行するホスト B は，そのデータパケットに対して，確認応答を返すことが保証される．ホスト B は，真の送信元であるホスト A ではなく，偽装されたホストであるホスト C に確認応答を返す．ホスト C は，【提案９】によって，警告パケットをホスト B に返すことが保証され，ホスト B は，そのデータパケットの送信元 IP アドレスが偽装されていたことを認識できる（図 11，図 12）．
【検証２】送信元 IP アドレスが存在しないホストに偽装されたとき
【提案２】，【提案５】によって，SYN パケットを受け取るアプリケーションを実行するホスト B は，その SYN パケットに対して，SYN/ACK パケットを返すことが保証される．ホスト B は，真の送信元であるホスト A ではなく，偽装されたホストである存在しないホスト C に SYN/ACK パケットを返す．【提案９】により，SYN/ACK パケットを中継するノードは，ICMP Destination Unreachable パケットがホスト B に送信されることが保証される．ICMP Destination Unreachable パケットを受け取ることにより，ホスト B はその SYN パケットの送信元 IP アドレスが詐称されていたことを認識できる．【提案４】，【提案７】によって，データパケットを受け取るアプリケーションを実行するホスト B は，そのデータパケットに対して，確認応答を返すことが保証される．ホスト B は，真の送信元であるホスト A ではなく，偽装されたホストである存在しないホスト C に確認応答を返す．【提案９】により，確認応答を中継するノード D は，ICMP Destination Unreachable パケットをホスト B に送信することが保証される．ICMP Destination Unreachable パケットを受け取ることにより，ホスト B はそのデータパケットの送信元 IP アドレスが偽装されていたことを認識できる（図 13）．
【検証３】送信元 IP アドレスが存在する正規でないホストに偽装されたとき
【提案２】，【提案５】によって，SYN パケットを受け取るアプリケーションを実行するホスト B は，その SYN パケットに対して，SYN/ACK パケットを返すことが保証される．ホスト B は，真の送信元であるホスト A ではなく，偽装されたホストである正規でないホスト C に SYN/ACK パケットを返す．また，【提案４】，【提案７】によって，データパケットを受け取るアプリケーションを実行するホスト B は，そのデータパケットに対して，確認応答を返すことが保証される．ホスト B は，真の送信元であるホスト A ではなく，偽装されたホストである正規でないホスト C に確認応答を返す．どちらの場合も，ホスト C は，正規でないため，【提案８】を実装しておらず，警告パケットが送信されないので，ホスト B は送信元 IP アドレスを偽装して SYN パケットまたはデータパケットが送られてきたことを認識できない（図 14）が，ホスト B があるホスト（ここではホスト A）に乗っ取られた時など，なにか問題が発覚してから，ホスト B のアクセスログを検証することにより，正規でないホスト C が割り出せる．この場合，ホスト A とホスト C の実行者は関係を持っている可能性が高く，ホスト C を割り出せることは，意味のあることであると言える．

5. 議論

本稿では，インターネットアプリケーションが，TCP

© 2012 Information Processing Society of Japan

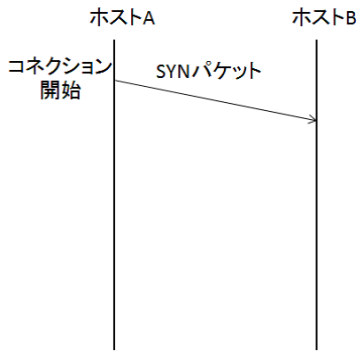

図 1 SYN パケットの送信.

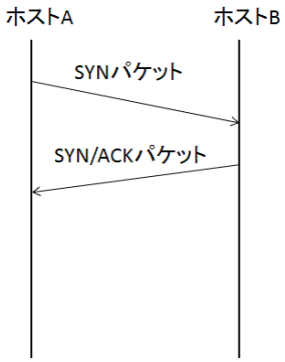

図 2 SYN/ACK パケットの送信.

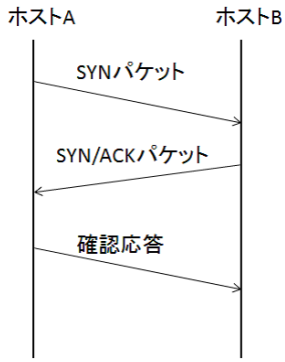

図 3 SYN/ACK に対する確認応答の送信.

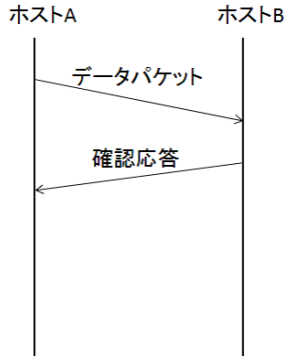

図 4 確認応答の送信.

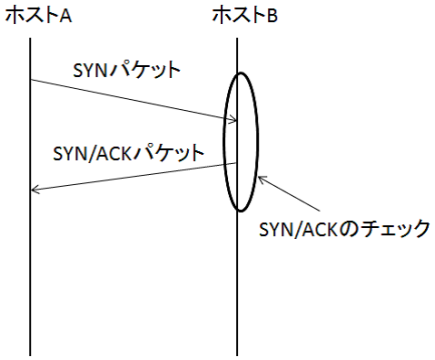

図 5 SYN/ACK パケット送信のチェック.

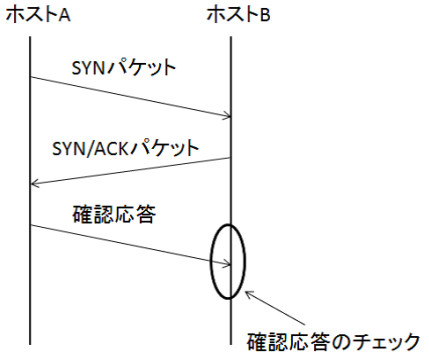

図 6 確認応答送信のチェック (1).

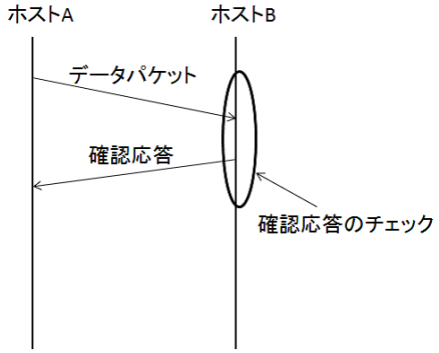

図 7 確認応答送信のチェック (2).

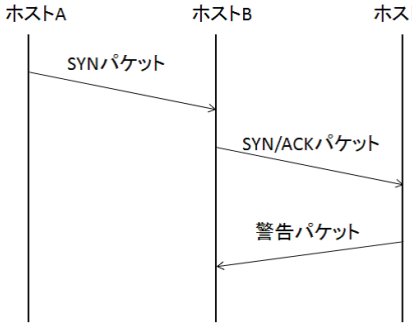

図 8 警告パケットの送信 (1).

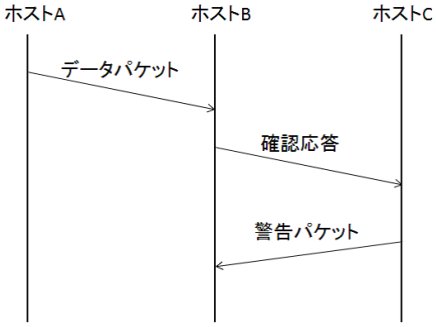

図 9 警告パケットの送信 (2).

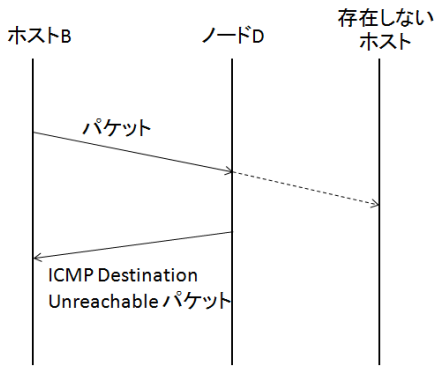

図 10 ICMP Destination Unreachable パケットの送信.

のように必ず確認応答を返すこととした．現在のインターネットアプリケーションでは，DNS が主に UDP を使っており，本提案を適用するのであれば，DNS も必ず TCP で通信するようにすべきである．また，ルーティングプロトコルとしては，UDP を使うプロトコル（RIP，RIP2）や IP の上で直接動くプロトコル（OSPF，EGP）があるが，これらも TCP 上に実装するか，OS のカーネルに実装し，侵入者によるハッキングを困難にする必要がある．

また，本稿では，偽装された送信元 IP アドレスに対する SYN/ACK パケットおよび確認応答について，警告パケットを送信することを提案している．一見，コネクション開始時の SYN/ACK パケットに対する警告パケットのみで，十分であると思われるかもしれないが，本提案では，あるタイミングでセッションハイジャック [15] が行われることも想定し，確認応答に対しても，警告パケットを送信することとした．

本提案を適用すれば，OS のカーネルが書き換えられない限り，データパケットの宛先ホストは，IP Spoofing を検

© 2012 Information Processing Society of Japan

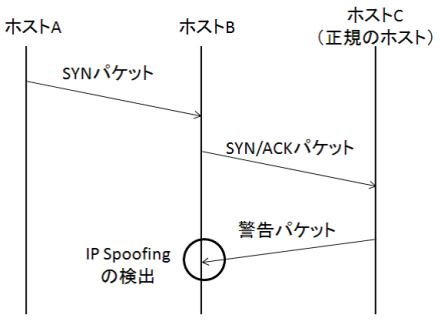

図 11 SYN/ACK パケットによる IP Spoofing の検出.

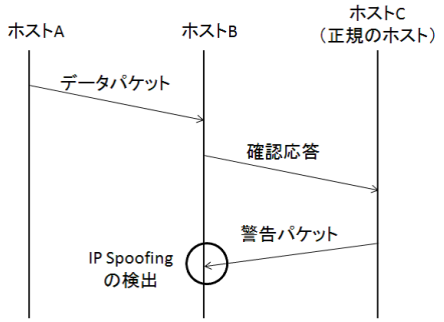

図 12 確認応答によるによる IP Spoofing の検出.

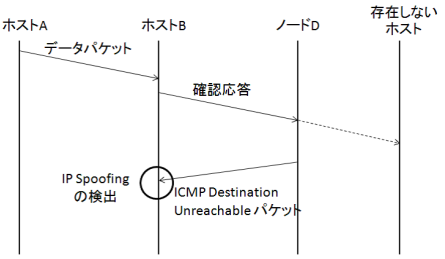

図 13 ICMP Destination Unreachable パケットによる IP Spoofing の検出.

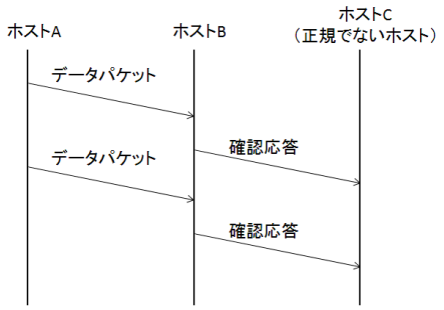

図 14 IP Spoofing が検出できない場合.

出できるか，アクセスログによって，IP Spoofing の実行者を特定できる可能性が高い．ただし，悪意のあるプログラムによって，OS のカーネルが書き換えられ，【提案１】～【提案９】のいずれかが実行されなくなれば，その限りではない．

また，アタッカーが，任意の確認応答を多数のホストに送信することによって，【提案８】が実行されていないホストをスキャンする可能性がある．その対策として，情報収集サーバを用意しておき，正規のホストが身に覚えのない確認応答を受け取ったとき，警告パケットを返信するだけでなく，情報収集サーバに，身に覚えのない確認応答に関する情報を送信することで，スキャンしているアタッカーを特定することが期待できる．すなわち，アタッカーが，警告パケットを送信しないホストを探すために，多数のホストに対して，任意の確認応答を送信したとする．正規のホストは，身に覚えのない確認応答を受け取ると，情報収集サーバにその確認応答の送信元などの情報を送る．たとえば，情報収集サーバによって，ひとつのホストから，多数ホストに対して，身に覚えのない確認応答が送信されていることが確認された場合，それらの確認応答の送信元のホストは極めて疑わしいと言える．

6. おわりに

本稿では，IP レベルでのなりすましのできない安全なインターネットの実現を目指し，（１）コネクション開始時，TCP のように 3 ウェイ・ハンドシェーキングを必ず行うこと，（２）OS のカーネルは（１）をチェックすること，（３）受け取ったデータパケットに対して，必ず確認応答を返すこと，（４）OS のカーネルは（３）をチェックすること，（５）身に覚えのない確認応答を受け取ったホストは，警告パケットを必ず返すこと，を提案した．

参考文献

[1] Ehrenkranz, T. and Li, J.: *On the State of IP Spoofing Defense,* ACM Transactions on Internet Technology (TOIT), vol. 9, issue 2 (2009).
[2] Saltzer, J.H., Reed, D.P., and Clark, D.D.: *End-to-End argument in System Design,* ACM Trans. Comput. Syst., vol. 2, no 4, pp. 277-288 (1984).
[3] Kent, S. and Seo, K.: *Security Architecture for the Internet Protocol,* RFC 4301.
[4] Fydor: *Remote OS detection,* http://nmap.org/book/osdetect.html (2006).
[5] Zalewski, M.: *Passive OS Fingerprinting Tool,* http://lcamtuf.coredump.cx/p0f.shtml (2006).
[6] Beverly, R: *A Robust Classifer for Passive TCP/IP fingerprinting,* In Proceedings of the Passive and Active Measurement Conference, pp.158-167 (2004).
[7] Taleck, G.: *Ambiguity Resolution via Passive OS Fingerprinting,* In Proceedings of the Symposium on Recent Advances in Intrusion Detection, pp. 192-175 (2003).
[8] Zalewski, M.: *Strange Attractors and TCP/IP Sequence Number Analysis,* http://lcamtuf.coredump.cx/oldtcp (2001).
[9] Zalewski, M.: *Strange Attractors and TCP/IP Sequence Number Analysis-One Year Later,* http://lcamtuf.coredump.cx/newtcp (2002).
[10] Bernstein, D.J.: *SYN cookies,* http://cr.yp.to/syncookies.html (1996).
[11] Aura, T. and Nikander, P.: *Stateless Connections,* In Proceedings of the International Conferene on Information and Communication Security, Lecture Notes in Computer Science, vol. 1334, Springer, pp.87-97(1997).
[12] Chang Feng, W. Kaiser, E.C., Chi Feng, W., and Luu,

© 2012 Information Processing Society of Japan

A.: *Design and Implementation of Network Puzzles,* In Proceedings of the Annual Joint Conference of the IEEE Computer and Communication Societies (InfoCom), pp. 2372-2382 (2005).

[13] Jin, C. Wang,H., and Shin, K.G.: *Hop-Count Filtering: An Effective Defense against Spoofed DDos Traffic,* In Proceedings of the Conference on Computer Communication Security, pp. 30-41 (2003).

[14] Wang, H., Jin, C., and Shin, K.G.: *Defense against Spoofed IP Traffic Using Hop-Count Filtering,* IEEE/ACM Trans. Netw., vol. 15, no. 1, pp.40-53 (2007).

[15] Harris, B. and Hunt, R.: *TCP/IP Security Threats and Attack Methods,* Computer Communications(Elsevier), vol. 22, pp. 885-897 (1999).

© 2012 Information Processing Society of Japan