サイレントワーム検知のためのアノマリコネクションツリーメソッド

川口信隆[†] 重野寛[†] 上田真太郎[†]
塩澤秀和[††] 岡田謙一[†]

本論文では，イントラネットやLAN内におけるサイレントワームの検知手法を提案する．既存の検知手法の多くは，ワームに感染したホストが起こすアドレススキャンなどの異常なネットワーク活動を検知する．このため，あらかじめ脆弱性を持つホストのリストを利用して，静かに感染活動を行うワームを検知することは難しい．本論文ではこのようなワームを検知するために，アノマリコネクションツリーメソッド（Anomaly Connection Tree Method，ACTM）を提案する．ACTMは多くのワームの感染活動に見られる2つの特徴を検知に利用する．1つ目は，感染コネクションをエッジ，ホストをノードとするツリーが構築されることである．2つ目は，ワームが次の感染先ホストを選択するとき，自身が感染しているホストがどのホストと頻繁に通信を行うかということを考慮しないことである．シミュレーションにより，ACTMがワームの感染活動の初期段階（感染ホスト数が全ホスト数の数パーセント程度）で検知を行えることを示す．

ACTM: Anomaly Connection Tree Method for Detection of Silent Worms

Nobutaka Kawaguchi,[†] Hiroshi Shigeno,[†] Shintaro Ueda,[†]
Hidekazu Shiozawa[††] and Ken'ichi Okada[†]

In this paper we propose a novel worm detection method that can detect Silent worms in intranet and local area network. Most existing detection methods use aggressive activities of worms as a clue for detection and are ineffective against worms that propagate silently using a list of vulnerable hosts. To detect such worms, we propose Anomaly Connection Tree Method (ACTM). ACTM uses two features present to most worms to detect worms. First is that the worms's propagation behaviour is expressed as tree-like structures. Second is that the worm's selection of infection targets does not consider which hosts its infected host communicates to frequently. Through the simulation results, we have shown that ACTM can detect the worms in an early stage.

1. はじめに

Staniford らは近年フラッシュワームという新しい形態のワームが現れる可能性について言及している[7),8)]．フラッシュワームは脆弱性を持つホストのアドレスリストを持ち，このリストを利用して高速かつ効率的な感染活動をするワームである．フラッシュワームは無差別的なアドレススキャンを行わない点で，Blaster，CodeRedといったこれまでのワームと大きく異なる．我々はこのフラッシュワームのようにアドレスリストを持ち，かつ個々の感染ホストが攻撃するホスト数を控え，異常なネットワーク活動を抑制するワームをサイレントワームと定義する．このワームは，他のワームに比べて少ないアクティビティで効率的な感染活動を行う．アドレススキャンにともなうトラフィックの異常度[2)]を検知に利用する既存手法では，サイレントワームの感染活動を早期に検知するのは難しい．

そこで本論文では，イントラネットやLAN内における未知のサイレントワームの感染活動を効率的に検知する手法として，アノマリコネクションツリーメソッド（Anomaly Connection Tree Method，以下ACTM）[1)]を提案する．ACTMは，通常のネットワーク活動では発生頻度が低いコネクションからなるツリーを検出することで，ワームの存在を検知する．なお，このようなコネクションのことをAC（Anomaly Connection）と呼ぶ．ACTMは，サイレントワームを含む多くのワームの感染活動が示す2つの特徴を利用する．1つ目は，ワームの感染活動は感染ホストをノードするツリー構造をとることである．2つ目は

† 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University
†† 玉川大学工学部
Faculty of Science and Technology, Tamagawa University

ワームが攻撃ホストを選択するとき，自身が感染しているホストがどのホストと頻繁に通信を行うかということを考慮しないことである．

以下，2 章ではサイレントワームと既存のワーム検知手法について論じる．3 章では ACTM を提案し手法の詳細について述べる．4 章ではシミュレーション実験を通じて ACTM の評価と考察を行う．そして，第 5 章を本論文のまとめとする．

2. サイレントワームと既存のワーム検知手法

2.1 サイレントワーム

Staniford らが提起したフラッシュワーム[7),8)]は，あらかじめ何らかの手段で，脆弱性を持つホストのアドレスリストであるヒットリストを作成し，このリストをもとに効率的かつ高速な感染活動を行うワームである．ヒットリストはワーム間で共有され，感染施行先ホストを決定する際に用いられる．アドレススキャンによる感染活動と異なり，ヒットリストを用いた感染活動は高い確率で成功する．

我々はフラッシュワーム同様にヒットリストを持ち，かつ個々の感染ホストの他ホストへの感染試行回数を数回程度に抑えることで，ネットワークベースの IDS[3)]による検知を困難とするワームをサイレントワームと定義する．本論文ではその中でも特に，サーバホストだけではなくクライアントホストも有するネットワークサービスの脆弱性を利用することで，1 つのイントラネットや LAN 中の全脆弱ホストに感染しようとするサイレントワームを対象とする．代表的な脆弱性には，Blaster や Sasser が利用した Microsoft Windows の RPC サービスの脆弱性などがある．

2.2 既存のワーム検知手法

既存のアノマリ型のネットワークベースの検知手法の多くは，ワームが脆弱ホストの発見ために行う無差別的なアドレススキャンにより，短時間に大量のコネクション確立要求が発生することを検知に利用する．具体的には，一定時間内に確立するコネクション数[4)]や無効な送信先アドレス数[5),6)]，トラヒック量[18),19)]などを，平常時のトラフィックプロファイルと比較するなどして検知を行う．しかしサイレントワームに感染したホストはアドレススキャンなどの上記のネットワーク活動を示さないため，これらの手法では検知が難しい．

Gu ら[14)]は，アドレススキャンを受けたホストが，それから一定時間以内にアドレススキャン活動を開始した場合，そのホストがワームに感染している可能性が高いと判断することでワームの感染活動を検知する手法を提案している．しかしこの手法は，前述の手法と同様にアドレススキャンに依存するため，サイレントワームの検知には効果的ではない．また Xie らは，多数のホストがワームに感染した後で，通信ログをもとに感染源となったホストを高い精度で特定するトレースバック手法を提案している[16),17)]．この手法はランダムウォークを使って通信ログをたどることで感染源ホストを特定する．本論文で提案する ACTM もネットワーク内で発生するコネクションをたどるが，Xie らとは異なり，感染活動の早期にワームを検知することを目的としている．

Ellis ら[15)]は，ワームの感染活動により構築されるツリーのサイズや深さを検知に利用できると述べているが，具体的な検知アルゴリズムやその評価については述べられていない．また，McDaniel ら[25)]は，通常のネットワーク活動では発生しないコネクションをスイッチなどでブロックすることでワームの感染活動を抑制する手法を提案している．ACTM は，個々のコネクションの異常度に着目するが，さらにそのうえで，異常なコネクションにより構築されるツリー構造を独自の方法を用いて効率的に検出する点が，これらの手法とは異なる．

その他の検知手法としては，パケットペイロード解析型がある[10),11)]．しかしこれらの手法ではパケットが暗号化されている場合などは解析が困難であり，演算負荷も大きい．同様に，同一ペイロードを持つパケット数をカウントすることでワームを検知する手法[12),13)]もペイロードを暗号化するワームやポリモフィックワームに対して効果的ではない．

3. アノマリコネクションツリーメソッド

本章では，イントラネットワーク内で発生したサイレントワームを検知するための手法としてアノマリコネクションツリーメソッド（ACTM）を提案する．

3.1 サイレントワームのモデル

サイレントワームは何らかの手段により感染対象ネットワーク内の 1 台のホストに侵入し，そのホストを起点として感染活動を開始するものとする．本論文では，サイレントワームを以下のようにモデル化する．

(1) ワームは感染対象のイントラネットワークや LAN の完全なヒットリストを持ち，感染ターゲットをリストからランダムに決定する．ただし，すでに感染しているホストへの再感染は行わないものとする．

(2) ワームはサーバホストだけではなくクライアントホストも有するネットワークサービスの脆弱

性を利用することで，サーバ，クライアントホストの区別なく感染する．

(3) 多くの他のワームと異なり，個々のサイレントワームの他ホストへの感染試行回数はそのライフタイムを通じて数回程度とする．

(4) 感染に用いられる通信はユニキャスト通信とする．

(1)は，フラッシュワームなどのヒットリストワームのモデル[8),9)]に従った．検知手法の性能の向上などにより，スキャン型ワームの効率性は著しく低下すると考えられる．よって，今後，あらかじめ何らかの手段で攻撃対象ネットワークのヒットリストを取得するワームが出現する可能性が高い[21),22)]．ワームがヒットリスト中から感染試行対象ホストを選択する方法は様々に考えられるが，本論文では最も想定しやすく基本的な方法である「ヒットリスト中からランダムにホストを選択する」方法をモデル化した．より高機能なワームが，より効率的でIDSによる検知を逃れるような選択方法を用いる可能性はあるが，このようなワームは本論文の対象外とする．また，サイレントワームは再感染を行わないとモデル化しているが，これは効率的な感染手段をとるワームを想定しているからである．このようなワームは現時点では一般的ではないが，フラッシュワームなどのヒットリストワームに関する研究[8),9)]ではモデル化されており，十分現実的になってきている．ワームが再感染を行うとすると，このことを利用した検知アルゴリズムを考えることができるが，本論文ではより効率的なワームを考えるため，この点を検知の際に考慮しない．

(2)については，「通常，サーバホストは他のホストにコネクションを確立することが少ない」という前提を用いずにワームを検知するためである．この前提を利用した手法[15)]は，Blasterなどクライアントホストも提供するネットワークサービスを利用するワームに対して非効率的である．

(3)については，本論文では，ホストが開くコネクション数を元に検知を行う手法[4)]では検知できないようなワームを対象とするため導入した．

(4)については，現在の多くのワームがユニキャスト通信で1台ずつ感染試行を行うため，サイレントワームも同様とした．

3.2 コネクションモデル

本論文では，2つのホスト間の通信を抽象化してコネクションと呼ぶことにする．このようなコネクションは実際にはTCPコネクションである場合や，UDPによる一連のパケットのやりとりである場合もある．ACTMはコネクションを何らかの手段で検出できることを前提する．コネクションは送信元，送信先アドレス，ポート番号などで識別できるものとする．ACTMは，接続元ホスト，接続先ホストがともに検知対象ネットワーク内に存在するコネクションをIC（Internal Connection）と呼ぶ．ICの中で，正規のネットワーク活動により発生するICをLC（Legistimate Connection），ワームの感染活動により発生するICをWC（Worm Connection）と呼ぶ．

次に，ICのうち，ある閾値を基準として発生頻度が高いICをNC（Normal Connection），発生頻度が低いICをAC（Anomaly Connection）とする．つまりACは，ネットワーク活動下における通信頻度が一定数以下の相手ホストとのICと定義される．NC，ACの分類はあくまで過去のコネクション発生の履歴から求めた頻度によるため，LC，WCとは異なる概念である．

ACTMは個々のICを，接続元ホスト，接続先ホストで分類する．1つのホストを共有するIC，たとえばホストAからBへのICと，ホストAからCへのICは区別される．また同じホスト間のコネクションであっても向きが異なれば，異なるコネクションとして区別される．

3.3 検知アルゴリズム

3.3.1 アルゴリズムの概要

ACTMは，ICからACを抽出し，ACをエッジ，ホストをノードとするツリー構造を検出することで，ネットワーク中のワームの存在を検知する．ACTMは以下に述べるワーム感染活動の2つの特徴を検知に利用する．

(1) ワームは自身を再帰的に感染ホストにコピーし，感染ホストはさらに他のホストに再感染を行う．よってワームの感染活動は，感染ホストをノード，WCをエッジとしたツリーとして表現できる．

(2) ワームは，自身が感染しているホストがどのホストと頻繁に通信するかを行うかを考慮せずに，感染活動を行う．

一方，一般に，ワームに感染していないホストは，ネットワーク中の全ホストの一部のホストとのみ頻繁に通信する[23),24)]．たとえば，あるホストからの通信のうち80%の宛先ホストは全ホストの20%のホストに集中するなど，通信頻度に偏りがある．

ACTMはパケットキャプチャリングなどにより，ネットワーク内で発生するICを観測する．次に，観測したICをACまたはNCに分類する．そして，ACに

分類された複数の IC を連結することで，アノマリコネクションツリー（AC ツリー）を検出する．

ACTM はワームが発生していない一定期間に観測した LC のうち発生頻度が閾値以上である LC を NC と判断する．そして，NC 以外の IC を AC と判断する．前述のとおりワームに感染していないホストの LC の相手先ホストは一部ホストに集中するため，LC の多くは NC に分類される．一方で，ワームの特徴 (2) で述べたとおり，ワームは自身の感染ホストの通信先ホストの特性を考慮せずに感染活動を行う．このため，WC の多くは AC に分類される．

図 1 に，ホスト A-I 間で複数の AC，NC を確立している様子を示す．ここでは {A,B,D,E,H}，{C,F,G,I} の 2 つの AC ツリーが検出される．LC が AC である確率は WC が AC である確率よりも相対的に低い．このため，通常のネットワーク活動時にツリー状になった n 個の AC が検出される確率は，ワームが発生した場合にツリー状になった n 個の AC が検出される確率と比べて相対的に低くなる．そこで ACTM は閾値を超えるサイズの AC ツリーが検出された場合，ワームがツリー中に存在すると判断する．なお本論文ではツリーサイズを，ツリーに含まれるホスト数とする．たとえば，図 1 の 2 つの AC ツリーのサイズは 5 と 4 である．

ここで，WC が NC として検出される確率も一定量ある．このため，ワーム発生に起因する AC ツリーは，1 つではなく複数のツリーに分割された形で検出される．このような場合，ある 1 つのツリーから一定距離内に平均的なツリーサイズよりも大きい AC ツリーが複数検出されるという傾向がある．そこで ACTM は複数の距離が近い AC ツリーを集約することで，VAC ツリー（Virtual AC ツリー）という仮想的な AC ツリーを検出する．そして AC ツリー同様，閾値を超える VAC ツリーが検出された場合も AC ツリーの場合と同様に，ワームが VAC ツリー中に存在すると判断する．なお，AC ツリー間の距離の定義などは 3.3.4 項で述べる．

ACTM は (1) 学習フェイズ，(2) 検知フェイズの 2 つのフェイズを持つ．学習フェイズではネットワークを観測し，検知に用いられる AC ツリー，VAC ツリーの閾値を決定する．またそのために，AC と判断されるべき IC のリストである AC リストを生成する．このフェイズではワームは存在していないものとする．検知フェイズではリアルタイムに AC，VAC ツリーを検出していき，閾値を超えるツリーが検出された場合，ワームがネットワーク中に存在すると判断する．なお，

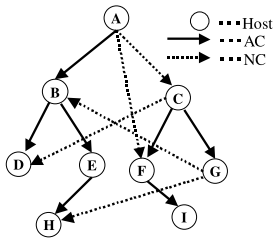


図 1 AC ツリーの例
Fig. 1 Example of AC Tree.

接続先ホスト	IC数
A	90
B	70
C	13
D	10
E	5
F	5
G	4
H	3
I	0
J	0

2 Fhosts (=10*FR)
CR=0.8 (=160/200)
合計10ホスト 合計 200IC

図 2 ホスト X の CList（FR=0.2）
Fig. 2 CList for host X (FR=0.2).

本論文では学習フェイズではワームは発生していないことを前提とし，学習フェイズ以後にネットワークやホストのアクセスパターン，使用用途が変化しない範囲での ACTM の検知性能について論じる．また，学習フェイズは 1 度だけ実行され，検知フェイズ中で再学習は行わないものとする．

以下，AC リスト生成アルゴリズム，AC，VAC ツリー検出アルゴリズム，学習，検知フェイズについて説明する．

3.3.2 AC リスト作成アルゴリズム

ここでは学習フェイズで実行する AC リスト作成アルゴリズムについて述べる．学習フェイズでは一定期間ネットワークを観測し IC ログを収集する．そして個々のホストごとに，ネットワーク中のその他のホストを接続先とする IC 数の頻度分布のリストを作成する．このリストを CList と呼ぶ．CList にはネットワーク中の全ホストが登録される．ネットワーク中の全ホストは，ネットワーク管理者からの情報などによりあらかじめ備わっていることを前提する．ACTM はネットワークベース IDS であり，SNMP などによりホストのリストを取得できると考えられるため，この前提条件は現実的といえる．接続先ホストは，IC 数が多い順に並べられる．**図 2** に，あるホスト X の CList の例を示す．ここで，CList の上位 FR（$0 \leq FR \leq 1$）のホストを Fhost と呼ぶ．そして，Fhost との IC を NC，Fhost 以外のホストとの IC を AC とする．ま

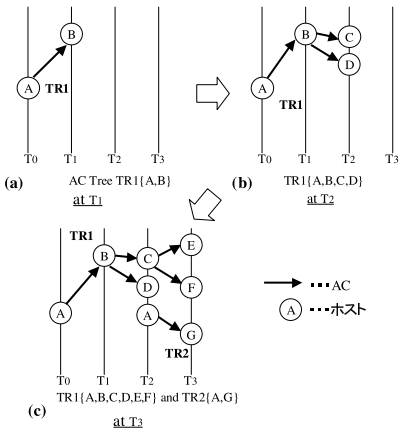


図 3 AC ツリーの構築
Fig. 3 AC Tree construction.

た，全 IC 数に対する NC 数の比率を，CR（Fhost's Connection Rate）とする．ワームはヒットリスト中からランダムに感染試行先ホストを選ぶため，ある WC が AC と判断される確率は 1.0−FR となる．一方，LC が AC と判断される確率は 1.0−CR となる．なお，ホストごとに CList の内容は違うため全ホストで同一の FR 値を用いたとしても，CR の値は各ホストによって異なる．

図 2 では，全接続先ホスト数が 10 台，FR が 0.2 の場合を示している．ホスト X の Fhost はホスト A，B となる．よってホスト X と，ホスト A，B との通信は NC，それ以外のホストとの IC は AC となる．また，CR は 0.8（$= (90 + 70)/200$）となる．よって WC が AC である確率は 0.8，LC が AC である確率は 0.2 となる．

最後に ACTM は，個々の接続元ホストと，AC と見なされる IC の接続先ホストの対応付けを AC リストとしてまとめる．ある IC が AC であるか NC であるかを判断する際には AC リストを用いる．

3.3.3 AC ツリー検出アルゴリズム

ACTM は新しく検出した AC をすでに検出した AC ツリーに連結することで，より大きな AC ツリーを探索していく．**図 3** に AC が連結され，より大きなツリーが検出される様子を示す．この図では AC は接続元ホストを始点，接続先ホストを終点とした矢印で表現される．検出時間は終点が示す時間となる．以下，ホスト X が通信開始ホスト（始点）ホスト Y が通信先ホスト（終点）で時刻 Z に検出された AC を $AC^{Z}_{X,Y}$ と表記する．図 3 (a) の AC は $AC^{T_1}_{A,B}$ と表記する．

ある AC X を検知した場合，AC X の始点が過去 T_{limit} 以内に検出した AC Y の終点または始点と一致するとき，ACTM は AC X を AC Y に連結する．T_{limit} を接続限界時間と呼ぶ．AC を複数のツリーに AC に接続できる場合は，最も大きいツリーに属する AC を選択するものとする．

図 3 に，$T_2 - T_1 \leq T_{limit} < T_3 - T_1$ であるときの AC ツリーの検出の様子について示す．図 3 (b) では，時刻 T_2 に検出された $AC^{T_2}_{B,C}$，$AC^{T_2}_{B,D}$ はともに，時刻 T_1 に検出された $AC^{T_1}_{A,B}$ に連結される．これは，検知間隔時間 $T_2 - T_1$ が T_{limit} 以内であるためである．これによりツリー TR1 のサイズは 2 から 4 へと増える．一方，条件を満たす連結相手がない場合，新しく検出された AC は新しい AC ツリーの最初の AC となる．たとえば図 3 (C) では，時刻 T_3 に検出された $AC^{T_3}_{A,G}$ は既存のツリー TR1 の $AC^{T_1}_{A,B}$ とは連結されず，新しいツリー TR2 として認識される．

アルゴリズムに接続限界時間を導入するのは，長期間をかけて大きくなる AC ツリーの検知を制限するためである．接続限界時間により，ワームが存在していないときに必要以上に大きい AC ツリーが検出されることを制限することで，ワームの侵入により急速に成長する AC ツリーと，通常のネットワーク活動により比較的ゆっくりと成長するツリーの区別が可能であると考えられる．

3.3.4 VAC ツリー検出アルゴリズム

前述のとおり，WC が AC として検出される確率 1.0−FR であり，NC として検出される確率は FR である．したがって，FR が十分小さいときは WC が AC として検出される確率は NC として検出される確率よりも高い．しかし，WC であっても確率 FR で NC と見なされる．このため，ワームの感染活動により成長していく AC ツリーが途中で NC と見なされた WC によって分断される．結果的に，**図 4** に示すように，大規模な 1 つの AC ツリーが検出されるのではなく，NC によって隔てられた複数の AC ツリーが検出される．図 4 では 2 つの NC によって，TR1 から 2 つのツリー TR2，TR3 が分離され，3 つのツリーが検出される．NC によって元々のツリー（図 4 では TR1）の成長が妨げられるため，ツリーサイズが閾値を超えるのに時間がかかり，検知が遅れることになる．

ここで，NC によって分離された 2 つの AC ツリー（TR1 と TR2，TR1 と TR3）の距離を，ツリー間を最短で連結する NC のパスの長さと定義する．ワームの感染活動にともない発生する AC ツリーが NC によって 2 つに分離される場合，ツリー間の距離が d となる確率 X は $X = FR^d \times (1.0 - FR)$ となる．FR=0.2

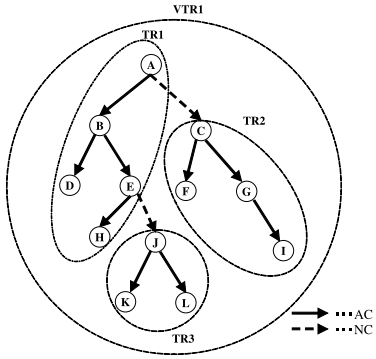


図 4 $V_d = 1$, $V_n = 2$ のときの VAC ツリー VTR1
Fig. 4 VAC Tree with $V_d = 1, V_n = 2$.

の場合，$d \geq 2$ になる確率は 0.04 となる．よってワームにより発生した AC ツリーは，NC によって分離されたとしても，比較的近い距離内（$d < 2$）に密集する確率は高いといえる．よって，ある AC ツリーから一定距離内に存在する AC ツリーを集約し 1 つのツリーとして扱うことで，NC により分離されたツリー構造を高い確率で復元して検出できる．そこで ACTM は，任意の AC ツリーをセンターツリーと定め，センターツリーから NC によって分離され，かつセンターツリーの近傍にある AC ツリーを集約して，VAC ツリーとして検出する．そしてこの VAC ツリーのサイズが閾値を超えた場合に，ワームがネットワーク中に存在すると判断する．図 4 では，TR1 をセンターツリー，TR2，TR3 を近傍ツリーとする VAC ツリー VTR1 が検出される．

このアルゴリズムはすべての AC ツリーに対して適用される．つまり，AC ツリーが N 個あれば，各ツリーをセンターツリーとした VAC ツリーが計 N 個検出され，ツリーサイズがチェックされる．

ここで，VAC ツリーに集約する近傍ツリーを決定するためのパラメータとして V_d，V_n を定義する．V_d は，VAC ツリーに集約される近傍ツリーのセンターツリー（図 4 では TR1）からの最大距離（図 4 では 1），V_n は近傍ツリー数の上限（図 4 では 2）を示す．センターツリー TR1 から距離 V_d 以内に S_d 個の近傍ツリー NTR_i，（$1 \leq i \leq S_d$，ただし $sizeof(NTR_i) \geq sizeof(NTR_{i+1})$）が存在する場合，VAC ツリー VTR1 のサイズ $sizeof(VTR1)$ は

$$sizeof(VTR) = sizeof(TR1) + \sum_{i=1}^{V_n} sizeof(NTR_i) \quad (1)$$

となる．ただし $S_d < V_n$ の場合は，$V_n = S_d$ とする．

この式から分かるようにサイズが大きい NTR_i を優先して VAC ツリーに加える．これは，ワームの感染活動に起因する AC ツリーを優先して集約するためである．サイズが小さい AC ツリーはワームの感染活動の有無にかかわらず多数検出されるため，これらを集約してもワーム検知に役立てることはできない．

なお，ある NC，NC1 が 2 つの AC ツリー TR1，TR2 を連結するには，3.3.3 項の AC ツリー検出アルゴリズムでの AC の連結条件と同様，TR1，TR2 中の AC1，AC2 が NC1 と始点または終点を共有し，接続間隔時間が接続限界時間以内である必要がある．TR1 と TR2 の距離が 2 以上の場合は複数の連続した NC によって連結されるが，このときの AC と NC，または NC どうしの連結条件も同様である．

3.3.5 学習フェイズと検知フェイズ

学習フェイズでは一定期間ネットワークを観測し，IC ログを生成する．そしてこのログをもとに AC リストを生成する．最後に，AC リストを用いて IC ログから，AC ツリーと VAC ツリーを検出する．そして最も大きい AC，VAC ツリーのサイズを閾値 TH_{AC}，TH_{VAC} とする．なおこのフェイズ中ではワームは存在せず，検出される IC はすべて LC であるとする．

検知フェイズでは学習フェイズで生成された AC リストを利用し，リアルタイムに AC，VAC ツリーを検出していく．そして TH_{AC}，TH_{VAC} を超える AC ツリーか VAC ツリーが検出された場合，ワームがツリー中に存在すると判断する．

4. 評　　価

本章では，ACTM の有効性を示すための計算機シミュレーションによる評価について述べる．

4.1 シミュレーション条件

本シミュレーションでは，ネットワークとしてイントラネットを想定し，ネットワーク内の全ホストが脆弱ホストであるとする．各ホストはファイアウォールなどに妨げられずに他のどのホストとも自由に通信を行えるとする．時間は TU を単位時間として用いる．通常のネットワーク活動では指数分布に従って平均 10 TU ごとに，各ホストは他のホストへ LC を確立するものとする．検知フェイズでは 1 つのサイレントワームが何らかの手段でネットワーク内の 1 台のホストに感染し，そこから全ホストをターゲットとした感染活動を行うものとする．ワームは感染コネクションとして TCP を用いるものとする．**表 1** にシミュレーションパラメータを示す．本シミュレーションでは断りがない限り，FR を 0.2 に設定し，全ホストで CR

表 1 シミュレーションパラメータ
Table 1 Simulation parameters.

ホスト数	1,000 台
FR	0.2
CR	0.8
通常コネクションインターバル	平均 10 TU の指数分布
感染試行回数	最大で 2 回
感染インターバル	可変 (1-20 TU)
V_d	1
V_n	2

は 0.8 になるものとした．また 1 つのワーム感染ホストが行う感染試行回数は 2 とする．現状のほとんどのワームがほぼ無制限に感染活動を行うことを考えると，この値は小さく，検知側にとってはより厳しい条件である．感染インターバルは連続する 2 つの WC の開始時間の間隔の平均値を意味する．またあるホストにワームが感染してから第 1 回目の WC の確立を行うまでの時間間隔も同様とする．実際の感染インターバルはこの値の ±20% の範囲の乱数値をとるものとした．

学習フェイズではワームが存在しない状態でシミュレーションを 10000 TU 時間実行し，この時検出される AC ツリーと VAC ツリーの最大値を TH_{AC}，TH_{VAC} とした．検知フェイズではシミュレーション実行 1000 TU 時間後にサイレントワームを 1 台のホストに感染させる．ワームはすぐに感染活動を開始する．そして，閾値を超える AC，VAC ツリーが生成されたときのワーム感染ホスト数である検知時感染ホスト数を測定する．結果はシミュレーション 20 回の平均値である．

この条件の下で以下の 5 項目について評価する．

(1) 検知時感染ホスト数の比較
(2) 接続限界時間の検知時感染ホスト数への影響
(3) 閾値の false positive rate，false negative rate への影響
(4) 閾値の検知時感染ホスト数への影響
(5) 通信の偏りの検知時感染ホスト数への影響

比較対象手法としては，ウイルススロットル[4] と AC カウント手法の 2 つを用いる．

ウイルススロットル[4] は，新しい IC が確立されるたびに接続開始パケット（TCP SYN パケットなど）を，送信元ホストのキューにプッシュする．その一方で，一定時間ごとにキューからパケットをポップする．大量のパケットによりキューが溢れた場合，ワームが存在すると判断する．なお，この評価実験ではウイルススロットルは，接続開始パケットから IC が AC か NC であるかを判定し，AC の接続開始パケットのみをキューにプッシュするものとした．これによりキューサイズは小さくなるため，ウイルススロットルの検知速度は向上する．

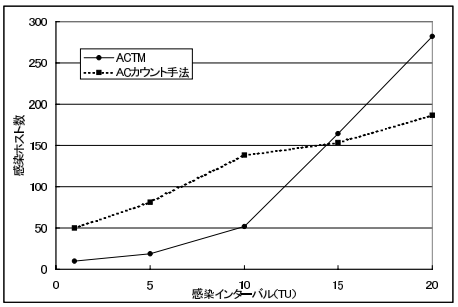


図 5 検知時感染ホスト数の比較
Fig. 5 Comparison of number of infected hosts.

AC カウント手法は一定時間（検知ウィンドウ）以内の過去に検出された AC の合計数を求め，この値が閾値を超えた場合ワームがネットワーク内に存在すると判定する．ACTM 同様，閾値は学習フェイズで検出された AC の合計数のうち最も大きいものを用いる．

4.2 シミュレーション結果

4.2.1 検知時感染ホスト数の比較

図 5 に，ACTM，AC カウント手法における検知時感染ホスト数を示す．ワームの感染インターバルが 14 TU より短い場合，ACTM は AC カウント手法よりも早い段階での検知を実現している．よってこの範囲内では，ワームの感染活動の特徴であるツリー構造の利用が検知時間の短縮に貢献しているといえる．一方で，感染インターバルが 14 TU を超えると AC カウント手法による検知のほうが早くなる．これは感染インターバルが長くなるにつれ，TH_{AC}，TH_{VAC} が大きくなるからである．このとき，NC が AC ツリーの成長を妨げる可能性は増え，VAC ツリーでも分断されたすべての WC に起因する AC ツリーを集約することが難しくなる．

通常，多くのワームのコネクション頻度は通常ネットワーク活動の数倍〜数十倍である．感染インターバルが通常コネクションインターバル以上であるワームはきわめて遅いワームといえる．このため通常コネクションインターバルの 1.4 倍程度までの感染インターバルにおいて，AC カウント手法よりも早期の検知を実現している ACTM の有効性は高い．特に感染インターバルが通常コネクションインターバル 10 TU より短い場合，ACTM は全ホストの 5%程度が感染した段階でワームを検知できる．

また，スケールの関係で図 5 には示されていないが，ウイルススロットルにおける検知時の感染ホスト数は

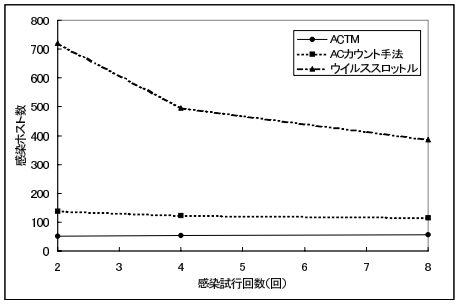


図 6 感染試行回数の影響（感染インターバル=10 TU）
Fig. 6 The effect of the number of infection trials with infection interval=10 TU.

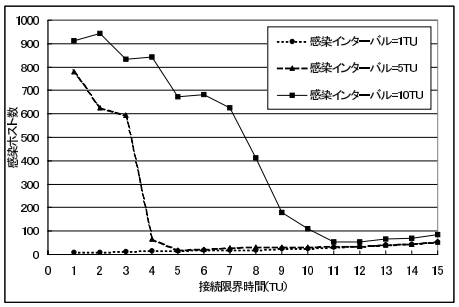


図 7 接続限界時間の影響
Fig. 7 The effect of the connection limit time.

感染インターバルが 1 TU で 519 台，10 TU で 720 台となり，ACTM に比べて大きい．これは，サイレントワームは従来のワームと異なり個々の感染ホストが数回程度しか感染活動を行わないため，キューが溢れるほどパケットが溜まる可能性が低いためである．

図 6 に，各感染ホストの感染試行回数を 4 回，8 回にしたときの検知時感染ホスト数の変化を示す．図からは判別しづらいが，ACTM は感染試行回数が 2 回から 8 回になった場合，感染台数は 1 割程度増加する．これは感染速度が速くなるため，ツリーから NC によって分離された AC に起因する感染ホスト数が多くなるからと考えられる．しかし，どちらの場合も，感染速度が上昇する前に検知しているため，感染ホスト数の差は小さい．感染試行回数が 8 回のときの AC カウント手法の感染ホスト数は 2 回のときと比べて 20 台程度減少している．ウイルススロットルは，個々の感染ホストの感染試行回数が多くなるほど検知速度が大きく向上する．

4.2.2 接続限界時間の感染ホスト数への影響と検知パラメータ

図 7 に接続限界時間の感染ホスト数への影響を示す．図から感染ホスト数を十分に小さくする接続限界時間は感染インターバルとほぼ同値であることが分かる．これは，接続限界時間が感染インターバル未満だと，2 つの連続した WC を連結することができなくなり，反対に接続限界時間が感染インターバルより大きくなると閾値が必要以上に大きくなりすぎるからである．

表 2 に閾値である TH_{AC}，TH_{VAC} と，VAC ツリーによる検知比率を示す．VAC ツリーによる検知比率はシミュレーション実行回数中，VAC ツリーによって検知された回数の割合を意味する．感染インターバルが長くなるほど検知比率は高くなる．これは，AC ツリーの閾値が大きくなるにつれて，閾値を超える 1

表 2 検知パラメータ
Table 2 Detection parameters.

感染インターバル (TU)	1 TU	5 TU	10 TU
TH_{AC}	7	11	21
TH_{VAC}	8	16	31
VAC ツリーによる検知比率	15%	35%	50%

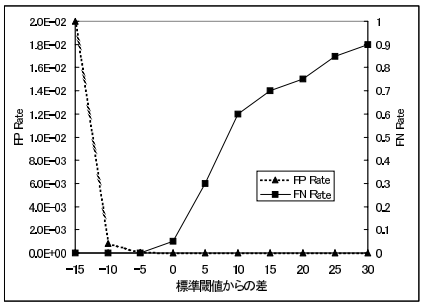


図 8 false positive rate と false negative rate
Fig. 8 False positive rate and false negative rate.

つの AC ツリーが検出される確率が低くなり，複数の密集した AC ツリーが検出されるからである．

4.2.3 false positive rate と false negative rate

図 8 に感染インターバルが 10 TU のワームに対する，閾値 TH_{AC}，TH_{VAC} の標準閾値からの差と false positive rate（FP Rate），false negative rate（FN Rate）の関係を示す．

ここで，標準閾値からの差とは，表 2 の，感染インターバルが 10 TU のワームに対する閾値（$TH_{AC} = 21$，$TH_{VAC} = 31$）からの TH_{AC}，TH_{VAC} の変化量を意味する．たとえば，差が− 10 である場合，閾値は $TH_{AC} = 11$，$TH_{VAC} = 21$ であり，差が+15 である場合，$TH_{AC} = 36$，$TH_{VAC} = 46$ である．また，FP Rate はワームが発生していない段階で，閾値を超えた VAC/AC ツリー数の全ツリー数に対する割合と定義し，FN Rate は「対象ネットワーク中の全ホストの x%が感染する前に検知を行えない確率」

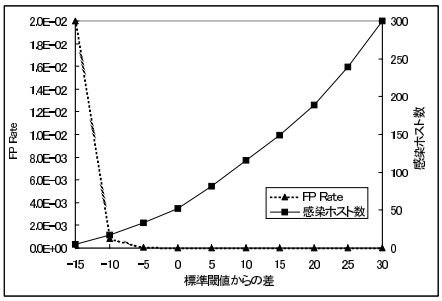


図 9 false positive rate と検知までの感染ホスト数
Fig. 9 False positive rate and number of infected hosts before detection.

と定義する．x が小さいほど，早期の検知が可能である．本論文では x=10%とした．なお，本シミュレーション条件下では VAC/AC ツリーは，1 TU あたり，合わせて 20 程度発生する．

図 8 から，閾値が標準閾値以上になるとき，FP Rate は 5×10^{-6} 以下になり，誤検知は 10000 TU に 0–1 回程度の頻度で発生する．また，このとき，FN Rate は 0.05 となっている．標準閾値との差が大きくなるにつれ，FN Rate は上昇していく．多少の FP を許容しても確実に早期検知を実現したい場合は，閾値を低めに設定する．たとえば，閾値を標準閾値より 5 程度下げた場合，FP Rate は，図からは判別しにくいが，6.3×10^{-5} であり，800 TU に 1 回程度の割合で誤検知が発生する．FN Rate はほぼ 0 となる．なお，図 8 には示されないが，閾値が標準閾値より 100 程度大きくなると，全ホストが感染する前にワームを検知することがほぼ不可能となる．

4.2.4 false positive rate と検知までの感染ホスト数

図 9 に，標準閾値からの差と検知までの感染ホスト数について示す．図 8 と同様に，検知までの感染ホスト数は標準閾値からの差が大きくなるにつれて増加する．標準閾値からの差が −5 であるとき，感染ホスト数は 33 台と，標準閾値のときに比べて約 20 台程度減少する．よって，ある程度の FP Rate（$= 6.3 \times 10^{-5}$，800 TU に 1 回誤検知が発生）を許容することで，感染ホスト数を大きく減らせることが分かる．

4.2.5 通信の偏りの検知時感染ホスト数への影響

ACTM は，ホストが通常頻繁に通信を行うホスト数が少ない，すなわち通信ホストの偏りが大きいほど早期の検知が可能である．**表 3** に様々な通信の偏りのもとでの，検知時の感染ホスト数を示す．ここで通信の偏りを，「個々のホストの 80%のコネクションの相手先ホスト数で，全ホストの 80%にあたるホスト数を割った値」と定義し，この値を偏り値と呼ぶ．たとえば表 1 の場合のように，FR=0.2 のときに CR=0.8 となるような通信の偏りがある場合，偏り値は 4.0（$= (1000 \times 0.8)/(1000 \times 0.2)$）となる．表 3 より，偏り値が大きいほど早期の検知が可能になり，偏り値が小さいほど，検知までに時間がかかることが分かる．ここで感染インターバルが 10 TU のときの検知時感染ホスト数について考える．偏り値が 1.0，すなわち各ホストが多くの他ホストと均一に通信を行う場合，ほぼすべてのホストが検知前に感染してしまう．一方感染インターバルが 1 TU のときは，偏り値が 8.0 のときの 2 倍の 14 台が感染するが，全ホスト数に対する感染数は小さい．よって，偏り値が小さいという条件下であっても，ACTM は高速な感染活動を行うサイレントワームに対して効果的といえる．

表 3 通信ホストの偏りの感染ホスト数への影響
Table 3 Effect of FR on minimal infected hosts.

感染インターバル (TU)	1 TU	5 TU	10 TU
偏り値 8.0	7	14	29
偏り値 4.0	11	19	52
偏り値 2.0	13	62	133
偏り値 1.0	14	288	986

5. おわりに

本論文では，イントラネットワークや LAN 内でのサイレントワームの検知を行うために ACTM を提案した．ACTM は，(1) ワームの感染活動をツリー構造として表現できる，(2) ワームが通常のネットワーク活動での発生確率が低いコネクションを多数確立する，という 2 つの特徴を利用した検知を行う．評価実験を通じて，ACTM は感染インターバルが通常コネクションインターバルより短いワームを，全ホストの 5%以下が感染した段階で検知できることを示した．

今後の課題としては，学習フェイズ中にワームが存在する場合の対処方法があげられる．学習フェイズ中に発生した大きな AC/VAC ツリーを形成したホストを他の検知手法などを使用して調査し，ワームが発生していた場合には，ワームによるコネクションを学習用ログから取り除くなどして対処することが可能と考える．また学習フェイズ後にネットワークやホストの性質が変化することも考えられるため，定期的に再学習を行う必要がある．具体的な再学習方法については今後の課題である．

また，CList の状態をもとに各ホストごとに最適な FR，CR を求めるアルゴリズムや，最適な V_d，V_n を求めるアルゴリズムの確立を目指すつもりである．さ

らに，NC とソフトウェア・ダイバーシティの概念を用いてワームの感染活動を停滞させる研究[20] などと ACTM を組み合わせることで，より早い段階でワームを検知できる手法を検討してく予定である．

謝辞 本研究は文部科学省 21 世紀 COE プログラム，文部科学省科学研究費補助金（C）課題番号 1850063（2006 年），ASF（応用セキュリティフォーラム）の支援を受けて行われた．

参 考 文 献

1) Kawaguchi, N., Azuma, Y., Ueda, S., Shigeno, H. and Okada, K.: ACTM: Anomaly Connection Tree Method to detect Silent Worms, *Proc. 20th IEEE International Conference on Advanced Information Networking and Applications*, Vol.1, pp.901–906 (2006).
2) 小畑直裕，川口信隆，塩澤秀和，重野 寛，岡田謙一：人間の行動を考慮したワーム感染シミュレーション，情報処理学会研究報告 No.2004-GN-053, pp.7–12 (2004).
3) 小宅宏明，宮地玲奈，川口信隆，重野 寛，岡田謙一：機械学習によるネットワーク IDS の false positive 削減手法，情報処理学会論文誌，Vol.45, No.8, pp.2105–2112 (2004).
4) Williamson, M. and Viruses, T.: Restricting Propagation to Defeat Malicious Mobile Code, Technical Report HPL-2002-172 (2002).
5) Jung, J., et al.: Fast Portscan Detection Using Sequential Hypothesis Testing, *Proc. IEEE Symposium on Security and Privacy* (May 2004).
6) Schechter, S.E., et al.: Fast Detection of Scanning Worm Infections, *Proc. RAID 2004*, pp.102–124 (2004).
7) Staniford, S., et al.: How to Own the Internet in Your Spare Time, *Proc. 11th USENIX Security Symposium* (2002).
8) Staniford, S., Moore, D., Paxson, V. and Weaver, N.: The Top Speed of Flash Worms, *Proc. WORM 2004*, pp.33–42 (2004).
9) Zou, C.C., Towsley, D. and Gong, W.: On the performance of Internet worm scanning strategies, *An International Journal on Performance Evaluation*, pp.700–723 (2006).
10) Kruegel, C., Kirda, E., Mutz, D., Robertson, W. and Vigna, G.: Polymorphic Worm Detection using Structural Information of Executables, *Proc. RAID 2005* (2005).
11) Newsome, J., Karp, B. and Song, D.: Polygraph: Automatically Generating Signatures for Polymorphic Worms, *Proc. IEEE Symposium on Security and Privacy*, pp.226–241 (2005).
12) Singh, S., Estan, C., Varghese, G. and Savage, S.: Automated worm fingerprinting, *Proc. ACM/USENIX Symposium on Operating System Design and Implementation* (2004).
13) Akritidis, P., Anagnostakis, K. and Markatos, E.P.: Efficient content-based detection of zero-day worms, *Proc. ICC 2005*, pp.837–843 (2005).
14) Gu, G., Sharif, M., Qin, X., Dagon, D., Lee, W. and Riley, G.: Worm detection, early warning and response based on local victim information, *Proc. IEEE 20th Annual Computer Security Application Conference*, pp.136–145 (2004).
15) Ellis, D., Aiken, J., Attwood, K. and Tenaglia, S.: A behavioral approach to worm detection, *Proc. WORM 2004*, pp.43–53 (2004).
16) Xie, Y., Sekar, V., Maltz, D., Reiter, M.K. and Zhang, H.: Worm origin identification using random moonwalks, *Proc. 2005 IEEE Symposium on Security and Privacy*, pp.242–256 (2005).
17) Sekar, V., Xie, Y., Maltz, D., Reiter, M. and Zhang, H.: Toward a Framework for Internet Forensic Analysis, *Proc. HotNets-III* (2004).
18) Zou, C.C., Gong, W., Towsley, D. and Giao, L.: The Monitoring and Early Detection of Internet Worms, *IEEE/ACM Trans. networking*, Vol.13, No.5 (2005).
19) Chen, X. and Heidemann, J.: Detecting, Early Worm Propagation through Packet Matching, Technical Report ISI-TR-2004-585, USC/Information Sciences Institute (2004).
20) O'Donnell, A.J. and Sethu, H.: On Achieving Software Diversity for Improved Network Security using Distributed Coloring Algorithms, *Proc. ACM CCS2004* (2004).
21) Kamra, A., Feng, H., Misra, V. and Keromytis, A.D.: The Effect of DNS Delays on Worm Propagation in an IPv6 Internet, *Proc. INFOCOM 2005* (2005).
22) Bellovin, S.M., Cheswick, B. and Keromytis, A.D.: Worm propagation strategies in an IPv6 Internet, *USENIX LOGIN*, Vol.31, No.1 (2005).
23) Aiello, W., Kalmanek, C., McDaniel, P., Sen, S., Spatscheck, O. and van der Merwe, K.: Analysis of Communities Of Interest in Data Networks, *Proc. Passive and Active Measurement Workshop 2005* (March 2005).
24) Pang, R., Allman, M., Bennett, M., Lee, J., Paxson, V. and Tierney, B.: A First Look at Modern Enterprise Traffic, *Proc. Internet Measurement Conference* (*IMC*) (2005).

25) McDaniel, P., et al.: Enterprise Security: A Community of Interest Based Approach, *Proc. NDSS 2006* (2006).

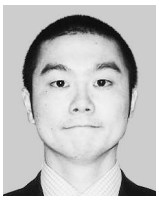
(平成 18 年 4 月 10 日受付)
(平成 18 年 11 月 2 日採録)

川口 信隆（学生会員）

2004 年慶應義塾大学大学院理工学研究科開放環境科学専攻前期博士課程修了．現在，同大学大学院理工学研究科開放システム科学専攻後期博士課程に在学中．ネットワークセキュリティの研究に従事．

重野 寛（正会員）

1990 年慶應義塾大学理工学部計測工学科卒業．1997 年同大学大学院理工学研究科博士課程修了．1998 年同大学理工学部情報工学科助手（有期）．現在，同大学理工学部情報工学科助教授．博士（工学）．無線 LAN の構成法と媒体アクセス制御方式，計算機ネットワークにおけるステーション移動サポート，モバイル・コンピューティング，遠隔教育システム等の研究に従事．電子通信学会，IEEE，ACM 各会員．



上田真太郎（学生会員）

2004 年慶應義塾大学大学院理工学研究科開放環境科学専攻前期博士課程修了．現在，同大学院理工学研究科開放システム科学専攻後期博士課程に在学中．ネットワークセキュリティの研究に従事．

塩澤 秀和（正会員）

1971 年生．1994 年慶應義塾大学理工学部計測工学科卒業．2000 年慶應義塾大学大学院理工学研究科計測工学専攻博士課程修了．博士（工学）．現在，玉川大学工学部知能情報システム学科講師．情報視覚化，グループウェア，ヒューマンインタフェース，ネットワークセキュリティ等に興味を持つ．電子情報通信学会，ACM，IEEE-CS 各会員．

岡田 謙一（フェロー）

慶應義塾大学理工学部情報工学科教授，工学博士．専門は，CSCW，グループウェア，コンピュータ・ヒューマン・インタラクション．情報処理学会誌編集主査，論文誌編集主査，GW 研究会主査等を歴任．現在，情報処理学会 GN 研究会運営委員，BCC 研究グループ幹事，日本 VR 学会仮想都市研究会副委員長．IEEE，ACM，電子情報通信学会，人工知能学会各会員．1995 年度情報処理学会論文賞，情報処理学会 40 周年記念論文賞，2000 年度情報処理学会論文賞受賞．2002 年情報処理学会フェロー．