RL-005

Honeypot 設置に伴う近隣 IP アドレスに対する攻撃傾向の分析
Analysis of the Effect of Honeypots on neighbor IP Addresses

下田 晃弘† 森 達哉‡ 後藤 滋樹†
Akihiro Shimoda Tatsuya Mori Shigeki Goto

1 はじめに

インターネット上のホスト数が急速に増大している[1]. それと同時にネットワークを介して送信されるマルウェア, botnet, その他の悪意のある通信が増加しており, インターネット・セキュリティの分野で解決すべき重大な課題となっている. 世界の Malware が経済に与える損失は 2006 年時点で 130 億ドル[2]に上っており, ネットワーク管理者は悪意のある通信を早期に検出し攻撃傾向の傾向を分析, 早急に対策を練る必要がある.

インターネットを介在した攻撃を検出する方法の 1 つとして応答を返さない受信専用のセンサーをネットワーク上の随所に配置してパケットログを採取・分析する脅威検出システムが各所で運用されている[3][4][5]. 広域に設置されたセンサーによる監視は不特定多数に送信される悪意のある通信を検出するための有用な手段であるが, 複数のセンサーを分散配置するための運用コストが大きいという問題がある. またセンサーは非インタラクティブかつ受動的な監視システムであるため, 詳細な攻撃通信の分析ができないという問題がある.

広域センサーシステムの運用コストを低減するため, 我々の先行研究[6]では, 組織のサブネット内部で未使用の IP アドレスをゲートウェイ上で検出し, 仮想センサーとして運用する方式を提案した. この方法は 1 組織が持つサブネットの大きさに応じて広範囲のアドレス空間をセンサーとし利用することができるため, 多くのデータを収集し, スキャンパターンの分析等に応用することができる.

上記の方式とは異なり, 攻撃の発信元とインタラクティブな通信を行うことで悪意のある通信を詳細に分析する Honeypot が提案されている[7][8]. 脆弱性のある OS を利用する High-interaction Honeypot は, ウイルスや botnet に意図的に感染することで, 未知の攻撃やバイナリを検出することが可能である. ただしマシン(実機)を用いるために機材の調達コストが高く, セキュリティを確保した上で運用することが難しい. また, 脆弱性のあるサービスをエミュレートする Low-interaction Honeypot は, ログやバイナリの収集を主な目的としている. このためセキュリティの確保が容易であるが, 脆弱性モジュールが実装されていない未知の攻撃に対応できない問題がある.

以上の状況を踏まえて, 著者らは広範囲のネットワーク空間をカバーし, かつ Honeypot と同等の詳細なログ取得を可能とする DarkPots を提案している[9]. DarkPots の主要なアイディアは Honeypot と仮想センサーを組み合わせることである. すなわち DarkPots はゲートウェイ上で未使用の IP アドレスを検出して, 次に検出した未使用 IP アドレスに対して送信される悪意のあるパケットをハニーポットによって構成される解析サーバに転送する. ここで転送先の解析サーバは複数指定することができる. この機能を利用して, 同時に異なる測定方式やサービスが混在する解析サーバを並列動作させることにより, 互いの結果を比較することや, 検出結果を補完することができる. さらにランダムなアドレス空間を動的に使うことができるため, ハニーポットの存在自体を隠蔽する効果が期待できる利点がある.

DarkPots は柔軟な未使用アドレスの利用を可能とするアーキテクチャであるが, ハニーポットの最適な配置方法については未検討であった. 本論文はこの点に着目し, Honeypot の存在が, 近隣の IP アドレスで観測される不正なパケットに与える影響を検証する. このような検証により, 空きアドレス空間に対する効果的なハニーポットの配置や動的な再配置の手法を確立することを狙いとする.

本論文では, 未使用 IP アドレス空間に対しセンサーと Honeypot を集中あるいは分散させて配置する. それぞれの配置について採取できるログや発信元の攻撃傾向の差異について分析を行う.

本論文の構成は以下の通りである. はじめに 2 章で関連研究について述べ, 3 章では DarkPots の概要を示す. つぎに 4 章では実験方法および, 実験方法を示し, 5 章では実験結果と考察について述べる. 最後に 6 章で本論文の結論を述べる.

2 関連研究

非対話型のセンサーを用いてインターネット上の攻撃, 脅威を検出する様々な既存の方法が存在する. [15][16][17]は, 地理的に離れたネットワーク上に受信専用のセンサーを分散配置して不正なパケットを収集. ログを 1 箇所に集めて分析して, Web ページ等を介してネットワークユーザに警告を促す活動を行っている. [19][20]は, ルーティングの変更により, 特定のネットワークサブネットに対するパケットをログ収集サーバに転送することにより, 広範囲の IP アドレス空間をモニタリングするシステムを構築している. [21]はセンサーから得られるログに加えて, 一般のユーザから提出されるファイアウォールのログ情報など, 多く情報源を組み合わせることで, 攻撃の局所性に影響されにくい広範囲な観測を可能にしている.

非対話型のセンサーに共通する欠点として, TCP による攻撃では backscatter などの例外を除き, 一般に syn パケットのみしか観測されない. そのため, セッションを確立した後の通信シーケンスがまったく観測できず, 攻撃の詳細な情報が得られない点が挙げられる.

一方で対話型のハニーポットを用いて, 攻撃の詳細な情報を得る取り組みも各方面で行われている.

Potemkin[12] はゲートウェイ上で特定の未使用の prefix 単位でルーティングを変更して, 仮想サーバ上の Honeypot に転送する仕組みを採用している. この仕組みで数万単位のアドレス空間を Honeypot に割り当てることを実現している. その一方で, prefix 単位ではなく個々の IP アドレスを取捨選択して割り当てることができないため, 使用/未使

† 早稲田大学
‡ NTT サービスインテグレーション基盤研究所

用の IP アドレスが混在する環境では，大きな prefix 単位でアドレスを確保することが難しいという欠点がある．

SGNET[22] は分散配置された各センサーに対する TCP 接続要求を，ゲートウェイに介してハニーポットに転送することで，測定点を地理的に分散配置したハニーポットによる分析を可能としている．その一方で，各センサーに割り当てられる IP アドレスは高々1 つであり，センサーの展開には多くのハードウェアリソースを必要とする．

nicter[23] はセンサーのセキュリティ・ログを用いたマクロな分析と，Malware の実行ファイルを解析して得られるミクロな分析を連携されることで，より精度の高い Malware の測定とインシデント・レポートの生成を実現している．

Sourcefilre RNA[13]は，ネットワークの中継点に Probe を設けて，当該サブネット範囲に含まれる IP アドレスの使用/未使用の有無を動的に判定する．ただし上記は IP アドレスのみではなく，MAC アドレス，OS の判別機能も備わっており，Probe をローカルサブネット上に局所的に複数設置して，ログを収集する方式を採用する．DarkPots のように，ゲートウェイのみに Probe を設置して検出した IP アドレスを他の目的に応用することは想定していない．

一方，未使用 IP アドレスの割り当てに関する類似研究として[14]は Honeypot に割り当てる IP アドレスの分散化による利点を示している．監視用のアドレス空間を規模の小さいサブネット単位分散して確保すると，同一サイズのアドレス空間単一のサブネットで確保する場合と比較して botnet による感染を 4～100 倍早く検知可能としている．本論文で提案する DarkPots は，Honeypot に隣接する IP アドレスに対するスキャンの傾向変化に注目しているため，攻撃検知の効率化を検証する[14]とは目的が異なっている

Greynets[24]は，企業や大学など使用/未使用のグローバル IP アドレスが混在する環境において，未使用の IP アドレス空間を寄せ集めて，従来のサブネット単位における未使用アドレス空間(darknets)を拡張した Greynets の概念を提案している．次章で述べる DarkPots [9]の Vacancy Checker が検出する IP アドレス群は一種の Greynets と見なすことができる．しかし，本論文における未使用の IP アドレスを検出する手法は[6]に基づいており，[24]とは異なる．また，現状の Greynets では DarkPots における Forwarder, Analyzer のように，他のセンサーやハニーポットと連携する仕組みについては考慮されていない．

3. DarkPots の概要

本論文の実験は[9] で提案する DarkPots のアーテキテクチャを利用しており，本章では要素技術として DarkPots の概要を説明する．図 1 に DarkPots のアーキテクチャを示す．DarkPots は，Vacancy Checker, Forwarder, Analyzer の 3 つのコンポーネントで構成される．Vacancy Checker はゲートウェイの前後のトラフィックをモニターし，通信中の IP アドレスを分析し，ゲートウェイ内部のサブネットの中で未使用となっている IP アドレスを検出する．Forwarder は Vacancy Checker によって検出された未使用 IP アドレスのリストを，4 章で述べる方式によって分割し複数の解析サーバ(Analyzer)にパケットを振り分ける．Analyzer はセンサーもしくは Honeypot サーバとして機能し，攻撃元に対してレスポンスを返す．次に，各コンポーネントの機能を述べる．

(1) Vacancy Checker

Vacancy Checker は，監視対象の回線を流れるトラフィックを分析して，未使用の IP アドレスを検出する．検出アルゴリズムは [6] に基づく．以下，Vacancy Checker の概略について述べる．

未使用の IP アドレスには以下の 2 つのケースがある．

(a) ゲートウェイのファイアウォールによって双方向の通信がブロックされている IP アドレス
(b) ファイアウォールは通過するが，LAN 側のいかなるホストにも割り当てられていない未使用の IP アドレス

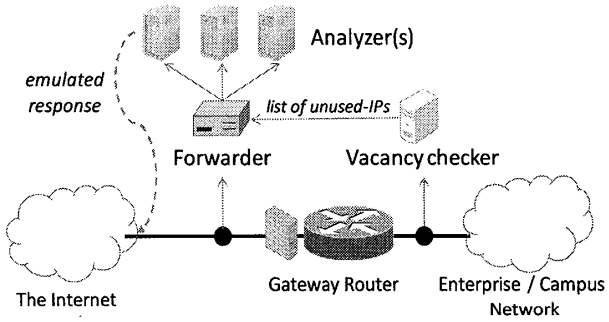

図 1 DarkPots アーキテクチャ

(a)のケースでは，WAN/LAN 側からの一切のパケットがブロックされるため，Analyzer が代理で応答をしてもネットワークの運用に影響を与えない．(b)のケースでは，IP アドレスがホストに未割当てである，あるいは端末の電源が OFF か，LAN 内部のみで常に使用される場合を想定している．Vacancy Checker は[6]と同様のアルゴリズムを用いて，一定期間以上，WAN 側への通信が観測されない非アクティブな IP アドレスを検出する．一旦，非アクティブと認識された IP アドレスでも，そのアドレスを送信元とする LAN から WAN 方向のパケットを検出した場合には，直ちに当該アドレスを未使用 IP アドレスのリストから除外する．

(2) Forwarder

図 2 に Forwarder の実装を示す．Forwarder は，Vacancy Checker から転送された未使用 IP アドレスリストを，任意のポリシーによって分割する．分割数は転送すべき Analyzer の台数と等しい．実装上はリストの分割数や IP アドレスの割当数に制限はないが，Forwarder の処理性能に応じて制限を設ける必要がある．分割ポリシーは自由に定めることができるため，例えば，各 Analyzer に割り当てる IP アドレス数を可変にする，もしくはリストからランダムに抽出した IP アドレスを割り当てるなどの柔軟なポリシーを設定可能である．

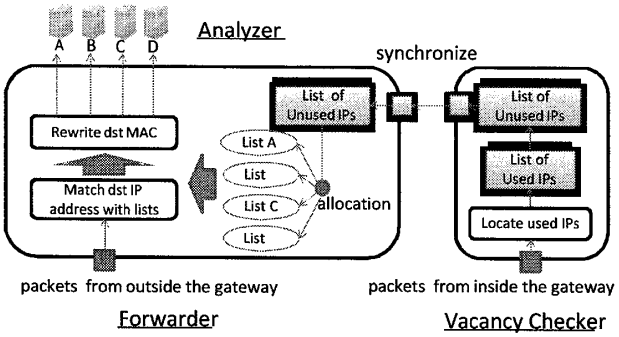

図 2 Forwarder の実装

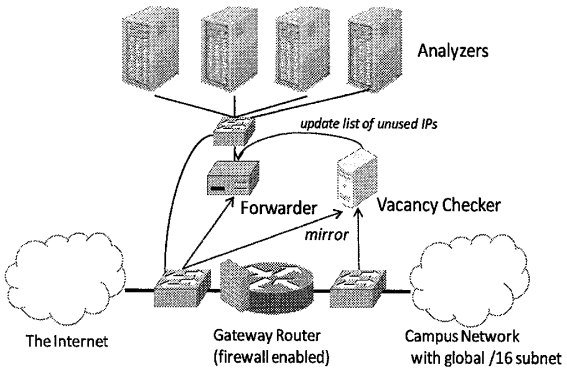

図 3　実験環境における DarkPots の実装

本論文では Forwarder は一般的に入手可能な Linux サーバ上で実装している．またリストの処理や転送処理を行うためのプログラムは C 言語と libpcap ライブラリを組み合わせで実装している．Forwarder はネットワークからキャプチャしたパケットの宛先 IP アドレスを分割したリストと照合して，リストに対応する Analyzer に対して転送する．パケットの転送は，宛先 MAC アドレスを Analyzer の受信インターフェイスの MAC アドレスに書き換えることで実現する．転送性能を測定したところ，分割数 6，合計 IP 割当数 15360 個でドロップ率 0.05%以下で転送できることを確認した．

(3) Analyzer

Analyzer は Forwarder から転送されたパケットを受信してログを取得する．Analyzer をセンサーサーバとして機能させる場合は Iptables[10]を用いて，受信専用のサーバとして設定する．また Analyzer を Honeypot サーバとして実装する場合には Honeypot サービスからの応答を外部に送信できるように iptables を設定する．ここで WAN 側のホストと未使用 IP アドレス間で通信を確立するために，Analyzer はパケットの送信時に，送信元 IP アドレスを，受信時の宛先 IP アドレス(=未使用 IP アドレス) に偽装して送信する．この処理を実現するため，各 Analyzer は Forwarder から分割済み未使用 IP アドレスのリストをリアルタイムで受信して Analyzer のサブインターフェイスの IP アドレスとして設定する(i.e. eth0:1, eth0:2, ..., eth0:n)．加えて，サービスの bind 先アドレスを任意 (i.e. 0.0.0.0, any)に指定することで，サービスに変更を加えることなく，未使用 IP アドレスからレスポンスを返すのに相当する処理を実現した．

4　実験環境と測定条件

図 3 に実験で使用したネットワークと DarkPots の実装を示す実験は早稲田大学のネットワーク管理者の協力のもとで，大学のサブネット全体を集約するゲートウェイ周辺で実施した．ネットワークのサブネットの大きさは/16 であり回線帯域は 10Gbps である．日中のトラフィックは平均 300～400Mbps 前後である．実験目的において一時的に 1Gbps を超えるトラフィックが流れる場合もある．

Vacancy Checker はゲートウェイの WAN 側，LAN 側の両方に Probe を設けており，Firewall を通過するパケットを検査することで Firewall の適用状況を調べている．それと同時に，3-(1)節で述べた方法により，学内における未使用 IP アドレスを検出する．実験に当たっては，未使用 IP アドレスの誤検知があると，既存の通信に影響を与えるリスクがある．誤検知を避けるために，検出した IP アドレスと，大学側の IP アドレス割当表を照合して，確実に利用されていないことを保証している．実験においては/16 のサブネットから，29811 個の未使用 IP アドレスを検出した．

実験は 2010 年 4 月 19 日～20 日に行われ，その間にゲートウェイの WAN 側に到達した全 7,214,589,101 パケットのうち，59,077,703 パケットを Forwarder に転送している．本実験では Analyzer を 4 種類，計 6 台稼働させている．6 台のそれぞれに本章で述べるポリシーにより，未使用 IP アドレスから各 2048 個の IP アドレスを抽出して各 Analyzer に割り当てている．

本実験では Honeypot サービスとして Nepenthes[10]を利用している．本ツールは主要な OS の脆弱性をエミュレートしており，MS04-012 (TCP/445, RPC-DCOM)MS02-039 (1434/TCP, SQL Server) 等の通信接続要求に対して，あたかも実 OS を稼働させているかのような応答を再現できる．Nepenthes は Low-interaction Honeypot であり，本ツールは未対応の脆弱性に対しては適切な応答を返せない．しかし，本実験は攻撃の詳細を分析するのが主目的ではなく，IP アドレス割当方法の違いによる攻撃傾向の差違を分析するのが主な目的である．そこで，実装の簡易さやセキュリティの確保のしやすさを優先して Nepenthes を採用した．

本実験と DarkPots[9]における実験構成の違いは，Analyzer の台数，種類，および各 Analyzer に割り当てる未使用 IP アドレスの確保の仕方である．本論文は攻撃のスキャンパターンの分析に主眼を置いており Honeypot が応答パケットを返すか否かによって，スキャンパターンに差が生じるかを検証する．そこで，実験では 4 種類の Analyzer (A, B, C, D)を用意して，それぞれに対して以下のようなサーバ設定と未使用 IP アドレスのパターンで DarkPots を構成した．

Analyzer A: センサーによる測定 (非対話型，非混在配置)

センサーとして機能させ，受信専用機としてログ収集のみを行う．未使用 IP アドレスリストより，2048 個の連続した IP アドレスを割り当てて，センサーをネットワーク空間上に展開する．さらに，攻撃の局所性による影響を評価するために，A1, A2, A3 の 3 台の Analyzer を設置して，互いに重複しないアドレス空間の 2048 個の IP アドレスを割り振る．

Analyzer B: Honeypot による測定 (対話型，非混在配置)

Honeypot サーバとして Nepenthes を動作させる．Analyzer A と同じく，他のセンサーと重複しない 2048 個の連続した IP アドレスを割り当てる．

次に，同一のサブネット空間上で，センサーと Honeypot を 1 つの IP アドレスごとに交互に配置して，Honeypot に隣接する IP アドレスに対する攻撃を分析する．以下の Analyzer C および D は，いずれも同じサブネットから IP アドレスを割り当てているが，重複しないように，C と D を 1 個の IP アドレスごとに交互に割り当てる．

Analyzer C: センサーによる測定 (非対話型，混在配置)
Analyzer A と同様にセンサーとして動作させる．2048 個の IP アドレスを割り当てる．

Analyzer D: Honeypot による測定 (対話型，混在配置)
Analyzer B と同様，Honeypot として Nepenthes を動作させる．2048 個の IP アドレスを割り当てる．

以上に挙げた，A1, A2, A3, B, C, D の各 Analyzer に対する IP アドレスの割り当て状況を図 4 に示す．また，IP アドレスの分散具合による影響を出来る限り排除するため，各 Analyzer がカバーするアドレス空間の範囲が±20%に収まるように調整をした．各 Analyzer のアドレス空間の大きさを，割り当て先端 IP と末尾 IP の差を IP アドレスごとにカウントしたものを距離とし，表 1 に示す．(e.g. x.x.10.0 から x.x.11.255 までの距離は 512 と計算)

表 1 Analyzer に割り当てるアドレス空間の先端，終端間の距離

	A1	A2	A3	B	C	D
distance	6532	5853	5371	7280	6828	6828

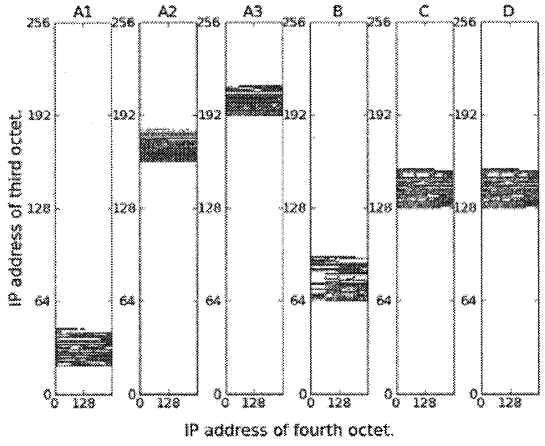

図 4 各 Analyzer に対して割り当てる未使用 IP アドレスの分布

5 実験結果

本章では，3 章で挙げた 6 台の Analyzer を，2010 年 4 月 19 日～20 日の期間， 同時並行に動作させた結果、得られたログデータと考察を述べる．

まず，測定全体を通じて観測された Inbound 方向のパケット総数，およびユニークな IP アドレス総数を表 2 に示す．比較対象は A1, A2, A3 (対話型)，C(対話型,非連続割当)である．

表 2 センサーにおける合計/ユニーク パケット観測個数

	A1	A2	A3	C
total	1,117,136	1,186,518	1,111,460	2,478,770
uniq	217,681	217,718	212,920	197,695

非対話型のセンサー同士を比較した場合には，非混在配置の A1, A2, A3 がおおよそ同じパケット数であるのに対して，混在配置の C のパケット観測数は，非混在配置の場合の 2 倍以上に上る．一方のユニーク IP 数は減少している傾向が観測されている．すなわち，対話，非対話型が混在する環境下に配置されたセンサーでは，混在しない場合と比較して，各ユニーク IP アドレスごとの平均観測パケット数が 2 倍以上に増加するという結果が得られた．

次に，表 2 と同一の測定条件とセンサーにおいて観測された tcp の syn, syn/ack, および udp パケットの観測個数を表 3 に示す．

表 3 Protocol/Flag 別パケット観測個数 (Inbound)

	A1	A2	A3	C
tcp/syn	865,031	875,024	871,705	2,248,087
tcp/syn ack	66,897	74,856	73,523	69,245
udp	124,238	105,567	95,567	151,156

inbound の tcp/syn においては，表 2 と同じく，非交互配置のセンサーA1,A2,A3 が数%の値に収まっているのに対して，ハニーポットとの交互配置の下にあるセンサーC では A の平均の 2 倍以上のパケットを観測している．A1, A2, A3 の間で偏りが小さいことを考えると，C のアドレス空間に対する局所的な攻撃が直接的な原因である可能性は小さく，C と同じアドレス空間で動作している対話型の Honeypot D の影響が想定される．これを説明する 1 つの仮説として，対話型のハニーポットはその周囲の IP アドレスに対する攻撃頻度を増加させていることが考えられる．

上記を検証するため，各 Analyzer に割り当てた未使用 IP アドレス 2048 個がスキャンされる速度を分析した． 非対話型のセンサーA1, A2, A3, C において，各センサーに割り当てた IP アドレスに対するスキャン数の時系列の累積個数を図 5 に示す．ここでのスキャンの定義は，測定開始時からセンサーに対して不正なパケット(TCP or UDP)が少なくとも 1 つ送り付けられた場合としている．ICMP は ping スキャン等で高速に網羅される可能性があるため測定対象から除いている．実験の結果，混在配置のセンサーC は，非混在配置の A1, 2, 3 と比較して未使用 IP アドレスがスキャンによって網羅される速度が速く，センサーA が平均 3721sec で全アドレスを網羅するのに対して，センサーC は 2783sec で網羅されている．すなわち，Honeypot を運用しているサブネットにおいては，周辺の Honeypot 以外の IP アドレスも早期にスキャンされることが確認できる．この現象を説明する理由の 1 つとしては，botnet や Worm のスキャンアルゴリズムが，通信が確立する IP アドレスの周辺も感染できる可能性が高いホストが存在すると見なし，近隣の IP アドレスに対して集中して不正なパケットを送りつけていることが推測される．

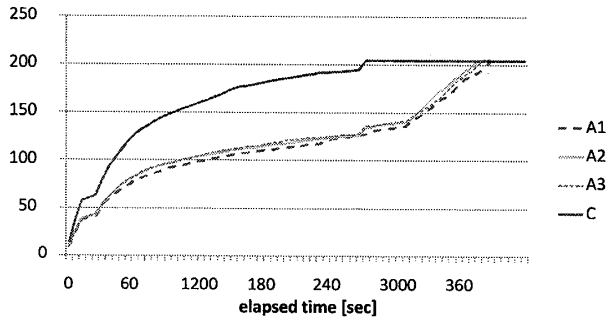

図 5 未使用 IP アドレスに対するスキャン速度の比較

次に，攻撃元 IP アドレスの分散傾向に関する分析を行う．攻撃元が分散または集中しているかを確認するために，攻撃元のネットワークを/24 の単位に集約して考える．攻撃元 IP アドレスの所属する実際のサブネットは受信側では推測が困難であり，/24 という値は正確ではない．しかし，本実験では攻撃元 IP アドレスが/24 の範囲で近接していると仮定して 1 つのサブネットの集約することで，攻撃元の IP アドレスの分散傾向を把握することを目的とする．本分析では，ユニークな IP アドレスの出現数の累積値を X，/24 で集約した場合の IP アドレスの出現数の累積値を Y としたとき，Y/X を求めることで，IP アドレスが/24 に集中している割合を求める．測定開始時から経過した時間に対する Y/X の変化を図 6 に示す．

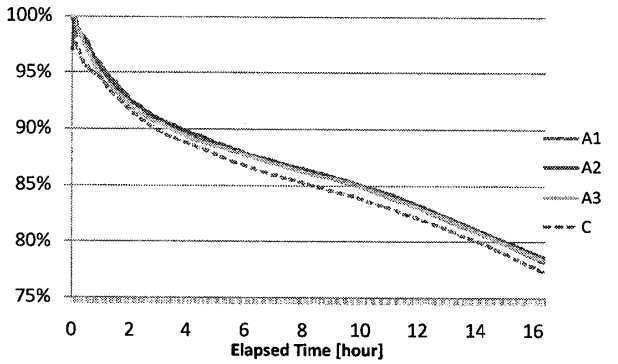

図 6 攻撃元 IP アドレスの/24 単位での集中の傾向

図 6 は，パーセンテージの値が高いほど，攻撃送信元の IP アドレスが分散していることを表す．全センサーに共通する特徴として，時間経過につれて，ある/24 のサブネット内に，複数の攻撃元 IP アドレスが出現する割合が高くなる傾向が見られる．混在/非混在の違いに着目した場合，非混在配置のセンサーA1, A2, A3 に対して，混在配置のセンサーC のパーセンテージの減少が僅かに大きいことが確認できる．すなわち，センサーC においては，攻撃元の IP アドレスが/24 の範囲内に複数出現する割合が高いことを意味する．この現象を説明する 1 つの推測として，攻撃元の botnet や malware の一部には，感染活動の試行をして応答が返された場合に，近隣に配置された botnet, worm と情報を共有し，連携している可能性が考えられる．

最後に，対話型の Nepenthes とアドレス空間の配置場所による攻撃の局所性の影響を確認するため，Honeypot B, D のそれぞれで測定されるログを比較した．各 Honeypot で観測された各種ログの集計結果を表 4 に示す．

表 4 Honeypot B, D におけるパケット観測個数の比較

	Analyzer B	Analyzer D
inbound total	13,874,750	14,045,916
inbound syn total	1,139,168	1,132,119
inbound syn 445/tcp	838,452	891,640
inbound syn 139/tcp	51,880	61,204
inbound syn 135/tcp	38,171	31,267
inbound syn 1433/tcp	16,085	18,592
outbound total	12,235,060	13,028,093
outbound syn/ack	988,891	1,012,750

対話型の Analyzer B, D はまったく同一の設定を行っている．表 1 で示したようにアドレス空間の大きさもほぼ同じであり，図 4 に示すようにアドレス空間の割り当てる場所が異なるのみである．Analyzer B, D の間において，若干の違いは見られるものの，数値の傾向は似ており，アドレス空間が異なっていても，Nepenthes の実装が測定結果に影響している可能性は低いと言える．

本論文の実験における共通の問題点として，各 Analyzer には，それぞれ異なるアドレス空間を割り当てているため，Honeypot 設置の有無に依存しないような，局所的な攻撃の影響を受けている可能性を否定できない．しかしながら，早稲田大学の/16 のサブネットにおいて，非混在配置の A1, A2, A3 のセンサーは，互いに異なる空間に配置されているにも関わらず，同様の傾向が見られ，混在配置の C のセンサーでは A とは異なる結果が得られていることから，C に対して送信されるパケットには，攻撃の局所性以外の影響を受けていると推定される．本章の実験においては，C のセンサーが D の Honeypot の影響を受けている可能性を示すために，センサーに対する Inbound 方向のパケット数，未使用 IP アドレスに対するスキャン速度，および，攻撃元 IP アドレスの分散具合から，botnet, worm の攻撃傾向に変化が見られることを確認した．

6 まとめ

本論文では，未使用 IP アドレスの柔軟な割り当てを可能とする DarkPots アーキテクチャを活用し，Honeypot 設置の有無による攻撃傾向に変化の観測と分析を行った．実験は，センサー，Honeypot のそれぞれを，IP アドレス空間上に連続的に配置する場合と，同一のサブネット上で交互に配置した場合それぞれで比較することで，Honeypot の設置が近隣の IP アドレスに与える影響を調べた．実験は，未使用 IP アドレスの空間上に，センサーのみ，ハニーポットのみ，センサー/ハニーポット混在するケースを想定した 4 種類の Analyzer を設置することで行った．また，/16 よりもサイズの小さい特定のサブネットに対する局所的な攻撃による測定結果への影響を考慮し，学内の/16 サブネット内に同一条件の 2048 個を 1 単位とするセンサーを複数箇所に設置した．

その結果，Honeypot 近辺のセンサーは，その他のセンサーに比べてより多くの syn パケットを受信することを確認した．測定開始直後においては，Honeypot 近隣の IP アドレスに対して，その他の場所に設置したセンサーに比べて早い速度でスキャンされることを確認した．また，攻撃元 IP アドレスの/24 単位での分散・集中の度合いを調べたところ，Honeypot 近隣に対する攻撃は，他と比較して，攻撃元が/24 単位に集中している割合が高いという結果が得られた．

以上により，Malware や botnet の中には攻撃対象が意味のある応答を返した場合に，その近隣に対しても攻撃頻度を増す場合があることを示した．この結果から，botnet は応答の有無によって，近隣に対するスキャンの傾向を変えるアルゴリズムを有している可能性があることを示した．botnet バイナリの解析による攻撃先の探索アルゴリズムの検証による本結果の裏付けは今後の課題である．

一方で，本論文では DarkPots アーキテクチャが，特定のサブネットに対する脆弱性への攻撃を目的としたパケットのアクセス傾向の分析に有効であることを示した．

本論文は，空きアドレス空間に対するハニーポットの配置方法を検討するための客観的なデータを示した．脆弱性のホストが多い環境において，近隣の IP アドレスにハニーポットを設置すると，セキュリティ上のリスクが増すことを示す理由となる．また，ハニーポットを運用・研究する立場においては，ハニーポットに割り当てる IP アドレスを連続的に確保すると，分散して配置するのに比べて，より多くのログが記録されることを示す理由となる．

7 参考文献

[1] C. Kreibich, C. Kanich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, S. Savage, “Spamcraft: An Inside Look At Spam Campaign Orchestration,” Bostron, USA, April, 2009.

[2] Computer Economics Inc, "2007 Malware Report: The Economic Impact of Viruses, Spyware, Adware, Botnets, and Other Malicious Code", June, 2007

[3] ”The darknet project”, http://www.cymru.com/Darknet.

[4] J. Ullrich, “Dshield”, http://www.dshield.org

[5] 戸田, 松本, 宮川, “ISDAS: Internet Scan Data Acquisition System”, CSS2004, 2004.

[6] 下田 晃弘, 後藤滋樹, “フローデータからの Dark IP 抽出による脅威観測法”, 電子情報通信学会論文誌 B, vol. 32, no. 1, pp. 163--173, 2009.

[7] N. Provos, “A virtual honeypot framework”, "Proceedings of the 13th USENIX Security Symposium", vol. 132, 2004.

[8] L. Spitzner, “The honeynet project: Trapping the hackers,” IEEE Security & Privacy Magazine, vol. 1, no. 2, pp. 15-23, 2003.

[9] A. Shimoda, T. Mori, and S. Goto, “Sensor in the Dark: Building Untraceable Large-scale Honeypots using Virtualization Technologies,” Proceedings of IEEE/IPSJ SAINT 2010, July 2010 (to appear)

[10] G. N. Purdy, *Linux iptables - kurz & gut.*, O’Reilly, August, 2004.

[11] P. Baecher, M. Koetter, T. Holz, M. Dornseif, and F. Freiling, “The nepenthes platform: An efficient approach to collect malware”, *Recent Advances in Intrusion Detection*, pp. 165--184.

[12] S. Pispas, N. Hadjichristidis, I. Potemkin, and A. Khokhlov, "Effect of architecture on the micellization properties of block copolymers: A2B miktoarm stars vs AB diblocks", Macromolecules, vol. 33, no. 5, pp. 1741--1746, 2000.

[13] Sourcefire RNA, http://www.sourcefire.com/.

[14] M. Rajab, F. Monrose, and A. Terzis, "On the effectiveness of distributed worm monitoring", Proceedings of the 14 th USENIX Security Symposium, pp. 225-237, 2005.

[15] “警察庁セキュリティポータルサイト@police”, www.cyberpolice.go.jp/

[16] Y. Toda, N. Matsumoto, Y. Miyagawa, “ISDAS: Internet Scan Data Acquisition System”, Joho Shori Gakkai Shinpojiumu Ronbunshu, vol. 2004, no. 11, pp. 3A--4, 2004.

[17] Masaki.Ishiguro, Hironobu Suzuki, Ichiro Murase, Hiroyuki Ohno, “Internet Threat Detection System Using Bayesian Estimations”, Proceedings of The 16th Annual Computer Security Incident Handling Conference, 2004.

[18] “インターネット定点観測システム MUSTAN”, http://mustan.ipa.go.jp/

[19] The IUCC/IDC Internet Telescope, http://www.ilan.net.il/research/telescope/

[20] The Darknet Project, http://www.team-cymru.org/Services/darknets.html

[21] SANS Instiute: Internet Storm Center http://isc.incidents.org/.

[22] C. Leita, M. Dacier, SGNET: Implementation insights, 2008.

[23] D. Inoue, M. Eto, K. Yoshioka, S. Baba, K. Suzuki, J. Nakazato, K. Ohtaka, K. Nakao, “nicter: An Incident Analysis System Toward Binding Network Monitoring with Malware Analysis”, WOMBAT Workshop on Information Security Threats Data Collection and Sharing, pp. 58--66, 2008.

[24] W. Harrop, G. Armitage, “Greynets: a definition and evaluation of sparsely populated darknets”, Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data, pp. 171--172, 2005.