Treemap と Edge Bundling を利用したダークネットデータの可視化システムの提案

高柳涼[†1]　　岡田義広[†2]

概要：インターネットの普及に伴い，マルウェアによる被害の増加が世界的な問題になっている．攻撃者はコンピュータの脆弱性と呼ばれるシステムソフトウェア等の不備を利用し，コンピュータにマルウェアを感染させ悪意のある行為を行う．特にゼロデイ攻撃と呼ばれる，脆弱性が報告されてから対策が周知されるまでに受ける攻撃が問題となっており，攻撃発見のためにネットワークトラフィックを監視することは非常に重要である．本研究では Treemap と Edge Bundling を利用しダークネットのトラフィックを可視化することで，攻撃と考えられるトラフィックの発見を補助するシステムの提案を行う．マクロ視点からミクロ視点へドリルダウンして可視化をすることで，膨大なデータの可視化解析を効率的に行うことが可能となる．

キーワード:Treemap，Edge Bundling，ダークネット

Development of Visualization System with Edge Bundling and Treemap for Darknet

RYO TAKAYANAGI[†1]　YOSHIHIRO OKADA[†2]

Abstract: With the spread of the Internet, the increase of damage caused by malware has become a worldwide problem. Malicious persons attack computer systems by focusing on their some vulnerabilities called security holes in order to install malware. Especially, a zero-day attack is a serious problem that tries to attack through a certain security hole before the treatment against it. So, the visualization of network traffic is very important in order to find out such attacks. In this paper, the authors propose a visualization system of darknet data using Treemap and Edge Bundling for analysis of such attacks. The proposed system enables us to visually analyze a vast of darknet data at from macro level to micro level.

Keywords: Visualization, Network Data, Darknet, Treemap, and Edge Bundling

1. はじめに

近年，デジタル技術の発展により，インターネットは私たちの生活に広く普及しており我々の生活に欠かせないものとなっている．様々な情報をインターネット上でやり取りすることにより生活の利便性は向上しているが，その反面，情報の流出，不正アクセス，遠隔操作等のサイバー犯罪の危険にさらされることも多くなっている．また，犯罪行為自体も複雑化，巧妙化しており年々被害は増大の傾向にある．2013 年のノートンの統計調査[*1]によると，世界で 3 億 7800 万人，国内では 400 万人が被害を受けており，金額にして世界で 1,130 億ドル，日本だけで 10 億ドルの被害を受けているといわれている．

攻撃者たちは様々な手法で我々の使用しているコンピュータへの侵入を試みており，近年では標的にメールを送りつけて添付ファイルのウイルスに感染させる攻撃や，メール本文の URL にアクセスさせることでウイルスに感染させる攻撃がある．また，パスワードの使い回しやよく使われる単語をパスワードに設定していることによる不正アクセスなど，使用者のセキュリティへの軽視や設定不備を狙った攻撃がある．さらに，Web サービスに登録している情報が漏れることによる不正アクセス，Web サイトを書き換えマルウェア配布サイトへ誘導する水飲み場攻撃など，使用者がセキュリティに気を付けているだけでは対処できない攻撃もある．特にセキュリティ対策ソフトを入れるだけでは対策できない，ゼロデイ攻撃と呼ばれるシステムの脆弱性を狙った攻撃は問題になっている．特定のサービスに未知の脆弱性が発見され，その脆弱性に対する修正パッチがリリースされたとしても，セキュリティ意識が低いユーザのまだパッチが適用されていないコンピュータをスキャンし不正を働こうとするものである．会員登録を行うサービスや社員の管理部門などのサーバに脆弱性があり，それらが攻撃される場合は，前述した不正アクセスなどの 2 次被害を引き起こす．そのため，ネットワークトラフィックを監視し，攻撃者の活動を把握することは非常に重要であり、様々な可視化システムが開発されている．[8][9][10]

†1 九州大学,福岡県福岡市西区元岡 744 番地
Kyushu University, Motooka 744, Nishi-Ku, Fukuoka, 819-0395, JAPAN
†2 九州先端科学技術研究所，福岡県福岡市早良区百道浜 2 丁目 1 番 22 号 福岡 SRP センタービル 7 階
Institute of Systems, Information Technologies and Nanotechnologies (ISIT), Fukuoka SRP Center Building 7F, Momochihama 2-1-22, Sawara-ku, Fukuoka, 814-0001, JAPAN

*1 http://www.symantec.com/content/ja/jp/about/presskits/2013_Norton_Report.pdf

本研究では，ダークネットのトラフィックデータを可視化することで，攻撃者が行う感染活動やコンピュータシステムの脆弱性を狙ったスキャニングなどの活動を可視化するシステムを提案する．空間充填手法である Treemap [3][4][5]とグラフ描画におけるエッジ集約手法である Edge Bundling [6][7]を組み合わせてマクロな視点からミクロな視点でダークネットデータの可視化が行えるシステムの開発を行った．

2. ダークネット

本研究では NICT 提供のダークネットトラフィックデータと，国際連携によるサイバー攻撃予知・即応技術の研究開発プロジェクト（PRACTICE：Proactive Response Against Cyber-attacks Through International Collaborative Exchange）の国際連携のデータ[11][12]を対象として可視化システムの開発を行った．ダークネットとはインターネット上で到達可能な IP アドレスのうち，ホストが割り振られておらず未使用のアドレス空間のことを指す．IPv4 のアドレス空間は全てにホストが割り振られているわけではなく未使用の領域が存在する．未割当領域であるため，通常ダークネットにはパケットが送信されることはめったにないが，実際には相当量のパケットが観測されている[1][2]．ダークネットにパケットが送信される原因として以下のものが挙げられる．

1. マルウェアによる自動感染活動
2. 攻撃者やマルウェアによる脆弱性調査のスキャン
3. ボットネットの活動
4. 送信元を偽装したパケットへの応答

これらの要因により，多くのパケットがダークネットに到達している．1,2,3 に関してはターゲットを自動で設定することがほとんどであるためダークネットにパケットが届く．したがってダークネットのトラフィックを分析することで，現在流行しているマルウェア，周知されていない脆弱性への攻撃やスキャン，ボットネットの活動などをとらえることが可能である．

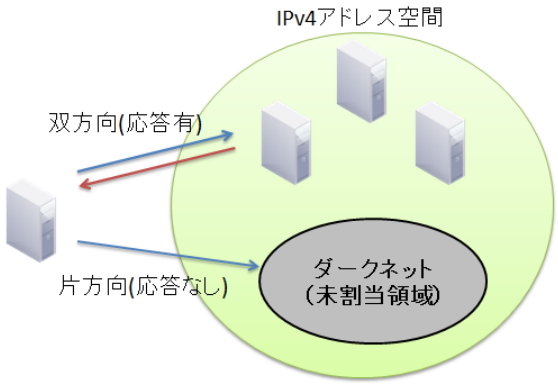

図 2.1　ダークネットイメージ図

また，ダークネットは実際のホストと違って応答はしないため，データにはマルウェアの検体そのものが含まれていることはない．

3. Treemap

本章では，B. Shneiderman によって考案された階層構造の可視化手法である Treemap [3]について述べる．階層構造の視覚化とその特徴は以下のようにまとめることができる．

- 表形式
  - 詳細な内容情報を提供できる
  - 構造的情報の可視化には向かない
- 箇条書き形式
  - 内容・構造情報をはっきりと提供できる
  - 大規模階層ではぎざぎざな構造になり分かり難い
- 木構造形式
  - ディプレイスペースの利用効率が十分ではない
  - 各々のノードは単純なテキストラベルしかもたないため内容情報が不足する

Treemap は与えられた階層構造に従い，与えられた長方形領域内部を入れ子で配置するため空間充填手法と呼ばれている．以降では，Treemap の詳しいアルゴリズムについて説明する．

3.1 Treemap アルゴリズム

Treemap アルゴリズムは階層的な情報を 2 次元平面上の領域を入れ子状に分割することによって可視化する手法の一つである．階層的な構造とは木構造に基づく構造である．たとえばファイルシステムにおけるディレクトリ構造，会社などにおける組織構造がそれに該当する．

ファイルやシステムの階層構造を可視化する手法として，Windows Explorer や Mac Finder などのような表示方法が一

般的に用いられる．ほとんどのインターフェイスはこれらの手法を踏襲しており広く利用されている．しかし，これらのツールでは大きな階層構造の概観を表示することができないという欠点がある．階層構造が深い場合には，ノードが横に広がりすべての階層を同時に見ることができない．またある階層に多くのファイルやディレクトリが存在する場合には，ノードが縦に広がりすべての階層を見るためにはスクロールが必要になる．上記の問題を解決するための手法として空間充填手法と呼ばれる手法がある．Treemap アルゴリズムは，階層的な情報を可視化する空間充填手法の中で代表的なものの一つである．Treemap は，あらかじめ設定された長方形領域内に，木構造のリーフノードをノードの重みが面積に対応するように密に埋めるアルゴリズムである．その際，階層構造を維持する．すなわち，中間ノードに対応する長方形領域について，その子ノードの重みに対応する面積となるように長方形領域を分割し，各子ノードを配置していく．各ノードはその親ノードの領域内の入れ子として表現されるため，階層構造が維持される．ノードの重みとしてある属性値を割り当てると，領域の大きさとして視覚的にその属性の値を把握できるようになる．加えて，RGB などの色情報として別の属性値を与えることでも視覚的にその属性の値を把握できるようになる．以下では，各ノードに割り当てられる重みを”size”と表現する．Treemap のノード配置アルゴリズムがいくつか提案されている．ここでは，Slice and Dice [3], Strip [4], Squarified [5] について紹介する．

3.1.1 Slice and Dice Treemap

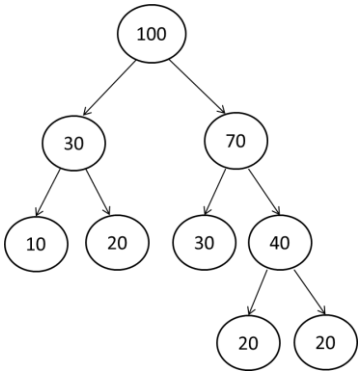

図 3.1　木ノード

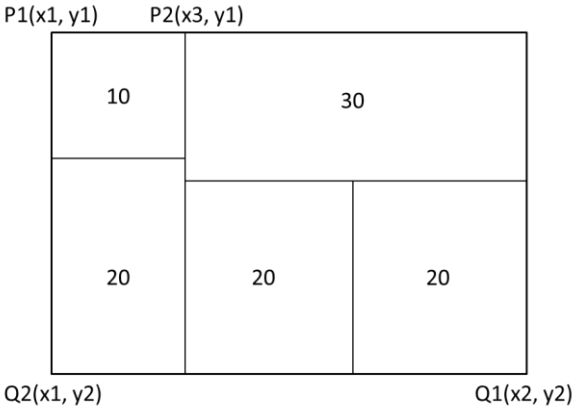

図 3.2　Slice and Dice Treemap

最初に，図 3.1 に示される木と図 3.2 に示される P1(x1, y1), Q1(x2, y2)で決まる長方形領域が与えられるとする．Slice and Dice Treemap では，まず，ルートノードが長方形領域 P1，Q1 全体にマッピングされる．さらにルートノードの 2 つの子ノードは，長方形領域 P2, Q2 内にマッピングされる．その際，長方形領域を X 軸方向に 2 つに分割しそれぞれの領域に 2 つの子ノードをマッピングする．その最初の分割線は以下のように引く．

$$x_3 = x_1 + \frac{size\ of\ childnode}{size\ of\ rootnode} \times (x_2 - x_1)$$

以下，このアルゴリズムは長方形 P2, Q2 を 90° 回転させて再帰的に分割・マッピングを行う．また，その長方形領域の面積は，そのノードの size に厳密に比例している．Slice and Dice Treemap によるマッピングは視覚的に分かり易いのだが，マッピングされた長方形の多くのアスペクト比が高くなってしまうという欠点を持つ．

3.1.2 Strip Treemap

Strip Treemap は，長方形領 R と順序付きのノードのリスト L を入力とする．長方形領域 R にはリストされているノードの親ノードがマッピングされており，この領域内にリストの子ノードをマッピングする．このアルゴリズムは，まず，入力された長方形領域 R に strip と呼ばれる水平方向の長さが 0 で垂直方向の長さが長方形領域 R と等しい（もしくは垂直方向の長さが 0 で，水平方向の長さが長方形領域 R と等しい）長方形領域を挿入する．このとき挿入された strip を current strip と呼ぶ．次に，入力されたリストの順に current strip にノードを加えていく．この際，1 つのノードを加える毎に，current strip の現在の size の総計と親ノードの size から current strip の水平方向の長さを計算し直す．また，current strip は挿入された各ノードがその size となるように水平方向に分割される．current strip 内の全ノードのアスペクト比の平均がノードを加えたことによって上がるならば，そのノードを current strip から除き，入力のノードリストに戻し，新たな strip を current strip として長

方形領域 R に挿入し入力ノードを追加する．すべてのノードが加えられるとアルゴリズムは終了する．以下にアルゴリズムを示す．

1. 新しい空の strip をつくる．この strip を current strip と呼ぶ．
2. 入力されたノードリストから，次のノードを current strip に加え，strip 内のノードの size の総計が親のノードに対して占める割合によって strip の高さを再計算する．そのとき strip 内の全ノードの分割幅も再計算し，それぞれのノードを strip 内にマッピングする．
3. current strip 内の全ノードのアスペクト比の平均がノードを加えたことによって上がるならば，そのノードを current strip から除き，入力のノードリストに戻して，1 から繰り返す．strip からノードを除いたとき，その strip は除いたノードを加える前の状態に戻る．
4. すべてのノードが加えられているならば終了する．そうでないならステップ 2 から繰り返す．
5. 以下，このアルゴリズムはマッピングされた各ノードに対して 1,2,3,4 を再帰的に繰り返す．

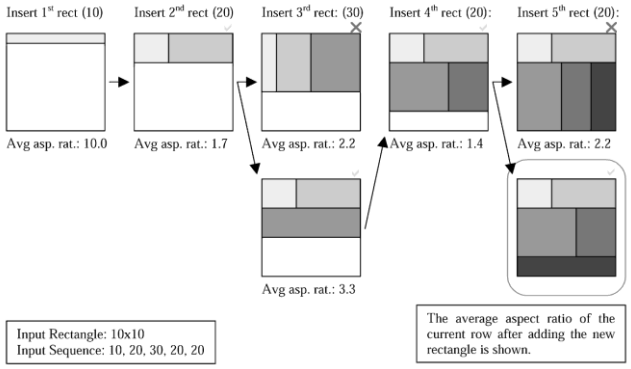

図 3.3　Strip Treemap のマッピング例 [4]

3.1.3 Squarified Treemap

Squarified Treemap は，長方形領域 R と順序付きのノードのリスト L を入力とする．以下にアルゴリズムを示す．

1. 入力で与えられる長方形領域 R の辺の短い方に沿ってノードを配置する．
2. 1 と同じ辺に沿ってノードを追加する．この際，各々のノードの size は追加したノードを含めた長方形領域 R に対する比を再計算しマッピングし直される．
3. 2 でノードを追加した際に，以前の各ノードの状態よりもアスペクト比が悪くなるならば追加しない．もしアスペクト比が悪くならない場合は 2 を再帰的に続ける．
4. 長方形領域 R で空いている領域を新たな長方形領域 R として 1,2,3 を再帰的に繰り返す．すべてのノードが加えられるとアルゴリズムは終了する．

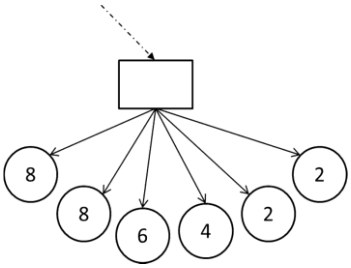

図 3.4　List の分割

Squarified Treemap は重みの大きい順に順序を入れ替えた配置であり，レイアウト上でのデータの順序が失われるため特定のノードを探すことが困難である．また多少のデータ変更の場合でもレイアウトが劇的に変化する可能性があるため個々のノードの追跡や選択も困難であり，表示の明滅も引き起こす．

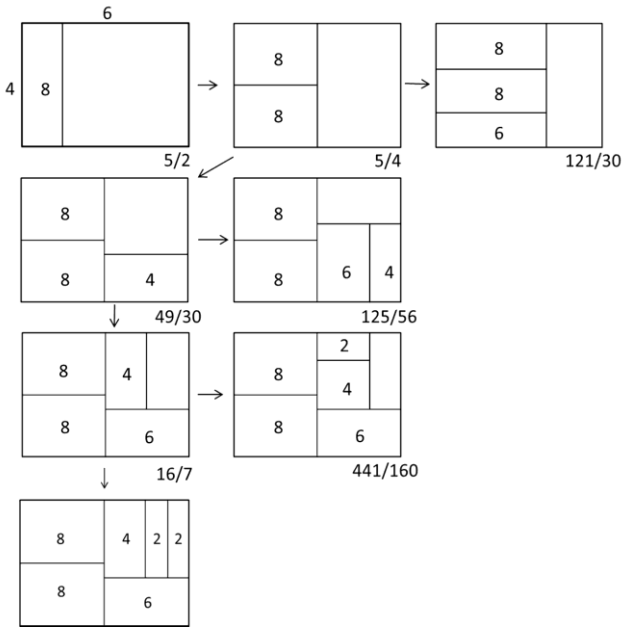

図 3.5　Squarified Treemap のマッピング例 [5]

4. Edge Bundling

Treemap などの空間充填手法はノードの重みとなる 1 つの属性を可視化する際に便利であるが，多次元データの可視化は得意ではない．そこで，提案手法では，ネットワークパケットの送信元 IP アドレス，宛先 IP アドレス，宛先ポート番号、これら 3 つの属性を用いてダークネットのパケットデータを可視化する．送信元 IP アドレスと宛先 IP アドレスを別の Treemap 上に配置し，宛先ポート番号のノードを中間に配置し，それぞれをエッジで結ぶことで一つ

のパケットデータを表すことにした．そして，Edge Bundling 手法を用いてエッジを束ねることによりグラフの視覚的煩雑さの軽減を行う．Treemap は IP アドレスのオクテットによって階層構造を持つため特定のネットワークセグメントにあるコンピュータの協調的活動の可視化や，脆弱性のある宛先ポート番号を指定した攻撃などを可視化することが可能となる．

4.1 Edge Bundling アルゴリズム

Edge Bundling [6]とは，木構造を持ったデータの２つの葉ノード間の関係を表すエッジを集約して，グラフ表現の視覚的煩雑さを軽減する手法である．木構造を持ったグラフ G についてP$_{start}$ からP$_{end}$ へエッジを引く場合，まず経路探索を行い，P$_{start}$からP$_{end}$ への経路を求める．その際，訪れた親ノードを全て記録する．次にP$_{start}$ からP$_{end}$ への経路上の点を制御点にして 3 次 B-スプライン曲線を求める．

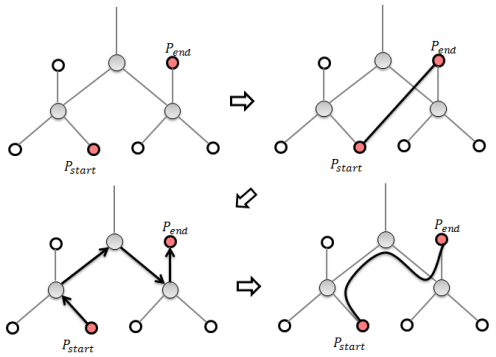

図 4.1　Edge Bundling アルゴリズム

Edge Bundling を用いることで，葉ノード同士のエッジは自身の親ノードに引っ張られるようになり，単純に葉ノード間にエッジを引くより視覚的に明瞭なグラフを得ることができる．

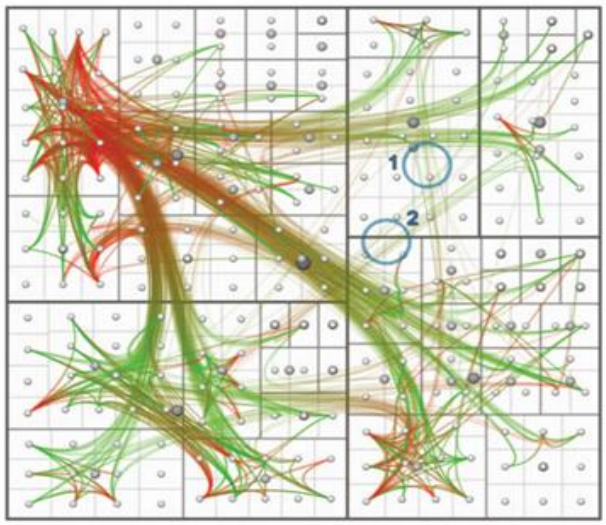

図 4.2　Edge Bundling 例 [6]

4.2 B-spline 曲線

B-spline 曲線とは，与えられた複数の制御点からなる滑らかな曲線である．制御点の数は(B-spline 曲線の次数+1)以上であれば幾つでもよい．また曲線の制御にノット t と呼ばれるパラメータを用いて柔軟な操作を加えることができるため，Bezier 曲線よりも局所的な操作を行うことが可能である．ここでノット列を以下のように与える．

$$T = [t_0, , t_1, \cdots, t_{m-1}]$$

B は basis(基底) の略であり，n 次の B-spline 基底関数を以下のように定義する．ただし$t_{j+n} - t_j, t_{j+n+1} - t_{j+1}$ が 0 のときはその項を 0 とおく．

$$N_{j,0}(t) = \begin{cases} 1 & for\ t_j \le t < t_{j+1} \\ 0 & otherwise \end{cases}$$

$$N_{j,n}(t) = \frac{t - t_j}{t_{j+n} - t_j} N_{j,n-1}(t) + \frac{t_{j+n+1} - t}{t_{j+n+1} - t_{j+1}} N_{j+1,n-1}(t)$$

$N_{j,n}(t)$に対して，$q_0, \cdots, q_j \in \boldsymbol{R}$ を制御点とするとき，n 次の B-spline 曲線の定義は以下である．

$$P(t) = \sum_{i=0}^{m-n-2} N_{i,n}(t) q_i$$

Edge Bundling 手法ではノットを開一様(Open Uniform)に設定する．開一様ノットとはノット列の両端で n+1 個のノットが重複しているものであり，このとき B-spline 曲線の端点と制御点q_0, q_jは一致する．

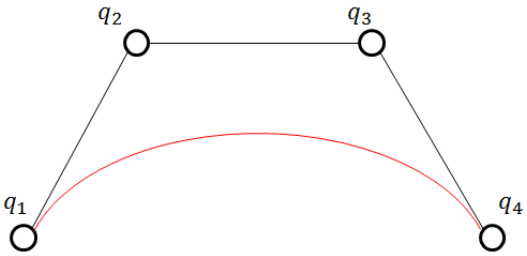

図 4.3　開一様ノットでの B-spline 曲線の例

5. 可視化システムと可視化例

本節では，開発した可視化システムの概要と可視化例を述べる．以下では，開発したシステムを PacketVisualization と呼ぶこととする．図 5.1 に示すのが PacketVisualization のスクリーンショットである．

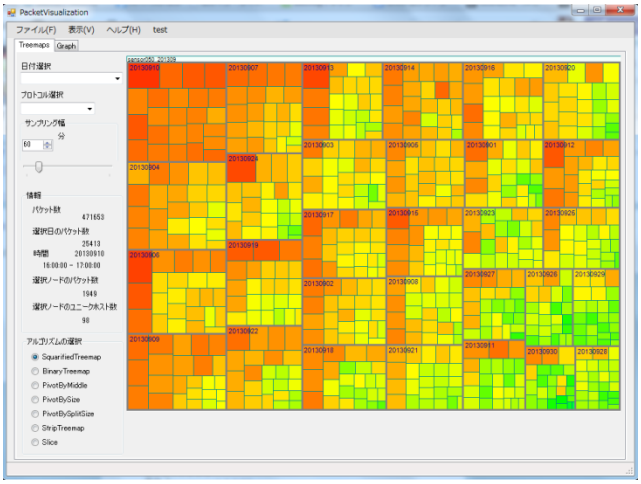

図 5.1　PacketVisualization のスクリーンショット

5.1　システム構成

PacketVisualization の開発言語は C#であり Windows ソフトウェアである.NET Framework4.0 以上の環境での動作を前提としている．読み込むデータは，WireShark の CUI ツール TShark を用いて, ネットワークキャプチャデータをテキストファイルに変換したものを利用している．ダークネットデータには様々な情報が含まれているが，本システムでは以下のものを用いる．

1　Time：通信が行われた時間,
2　Protocol：使用されたプロトコル,
3　Source IP：送信元 IP アドレス,
4　Destination IP：宛先 IP アドレス,
5　Destination Port：宛先ポート番号

統計量を用いたマクロ視点での可視化解析では 1~3，パケット単位のミクロ視点での可視化解析ではすべての情報を使用した．

5.2　マクロ視点での可視化解析

PacketVisualization は 1 か月分のトラフィックデータを読み込んで Treemap に表示する．階層構造はファイルのディレクトリ構造に従って 1 月→1 日→60 分の構造を持つ．操作としては，葉ノードのサンプリング幅の変更や，使用する Treemap アルゴリズムの変更を行うことができる．ノードのサイズは葉ノードに含まれる TCP のパケット数になっており，色は葉ノードに含まれるパケットの Source IP のユニークホスト数についての情報量によって求められる．情報量が少ないとき, そのノードは特定の Source IP から大量のアクセスを受けていると判断することができる．またマウス・クリックにより選択ノードの詳細表示，詳細解析画面への出力を行うことができる．

5.3　ミクロ視点での可視化解析

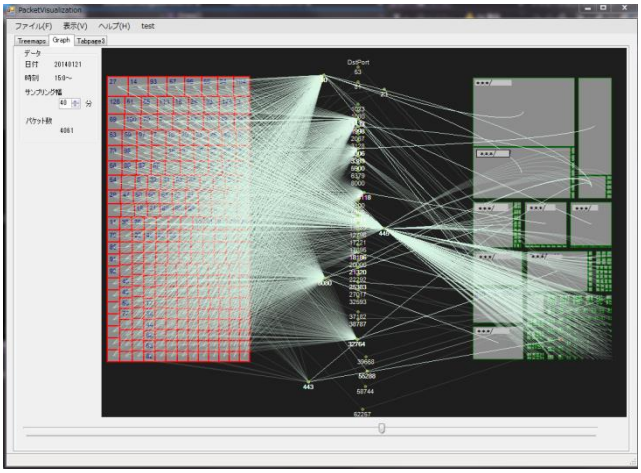

図 5.2 詳細解析画面のスクリーンショット

図 5.2 に示すように、前述したマクロ視点での可視化画面で選択したノードを 2 つの Treemap と Edge Bundling を用いて、パケット単位で詳細に可視化できる．図 5.2 の可視化の詳細を以下で述べる．

- 図 5.2 左側 Treemap
  - 各ノードは，Destination IP を意味する．そのサイズはパケット数で重み付けされており，番号は Destination IP の第 4 オクテットそのものである．実装的には階層構造化されているが，センサの範囲がサブネットマスク/24 のため, 第 4 オクテットのみ異なるデータが入っている．
- 図 5.2 中央ノード群
  - Destination Port を表すノード群であり, 数値はその番号である．当該時間区間のパケット群に現れていない Destination Port のノードは表示されない．
- 図 5.2 右側 Treemap
  - 各ノードは，Source IP を意味する．そのサイズは Destination IP の Treemap と同様にパケット数で重み付けされており，数値は Source IP アドレスそのものである．ただし，ノードの大きさによって表示できる文字数が制限されるため, 第 3 オクテットが含まれる階層の IP アドレスのみを表示している．こちらも Source IP アドレスの各オクテットに階層構造化されており, 黒(太)⇒黒+緑⇒緑⇒赤の順に第 1 オクテットから第 4 オクテットの順で階層構造を可視化している．

エッジは，Destination IP と Source IP の Treemap について，それぞれの第 4 オクテットに対応するノードの中心と，Destination Port のノードを通過する B-spline 曲線である．

制御点に上位オクテットのノードの中心を用いる．データの都合上，Destination IP の Treemap(図 5.2 左側)に関して第 4 オクテットのみ異なるものとなっており多少煩雑になっている．Destination Port のノードは自由に動かすことができユーザの要求に従った可視化結果を得ることができる．また，こちらの画面でもサンプリング幅や測定時間の変更は可能であり，詳細解析に出力した近傍を簡単に見ることができる．

5.4 可視化例

本節では国際連携ダークネットデータを実際に可視化した結果とそこから読み取れる結果について説明する．

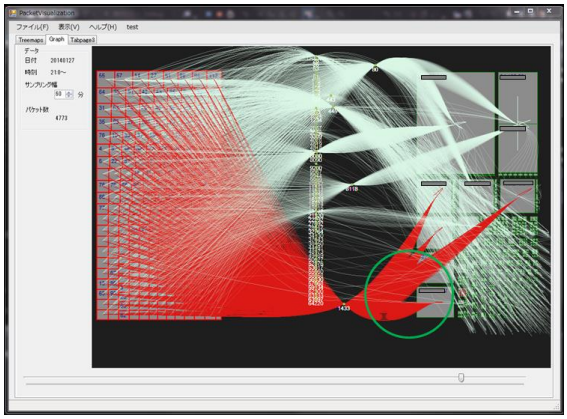

図 5.3　1433 ポートへの攻撃**(Sensor053 2014/1/27)**

この図 5.3 は，Sensor053 と呼ばれるダークネットの 2014 年 1 月 27 日 21 時から 1 時間分のパケットデータを可視化したものである．パケット数は約 5,000 である．スプライン曲線であるため, ある Source IP が特定のポートを使用してスキャンしている場合，図のようにエッジに幅が現れるためすぐにスキャン様の攻撃と判別できる. また 1433 へ向かうエッジに関しては複数の Source IP がスキャンをかけている様子が見て取れる．1433 ポートへのスキャンは Microsoft SQL Server が稼働しているコンピュータの探索行為と考えられる．

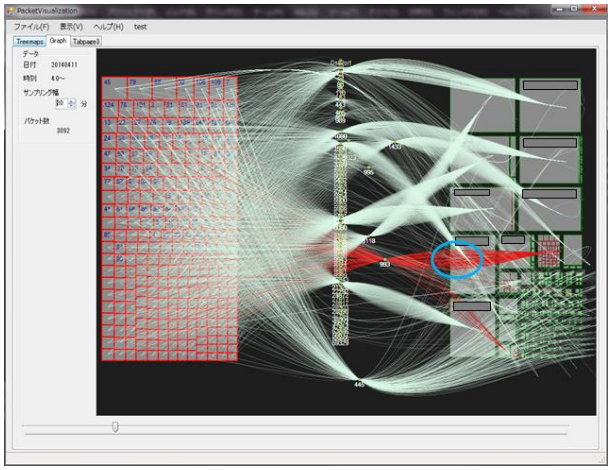

図 5.4 993 ポートへの攻撃(Sensor053 2014/4/11)

この図 5.4 は，Sensor053，2014 年 4 月 11 日 4 時から 30 分間のパケットデータを可視化したものである．パケット数は約 3,000 である．エッジの束の内部が他と比べて少ないことから，総当たりでスキャンしているのではなく，ランダムまたは Destination IP アドレスにある程度の間隔を持たせての攻撃であると思われる. また 993 は IMAPS(Internet Message Access Protocol over TLS/SSL)に使用されるポートであり，OpenSSL の脆弱性(Heartbleed：ハートブリード)を狙った攻撃だと推定される．

6. まとめと今後の課題

6.1 まとめ

本研究では，ダークネットトラフィックを監視するための Treemap と Edge Bundling を組み合わせた可視化システムの提案を行った．送信元 IP と宛先 IP に別の Treemap を用い，攻撃の内容に関連の強い宛先ポート番号を中継ノード群として，各パケットをエッジ表現することにより，3 つの属性の関係が直感的に理解し易くなったばかりでなく，パケット間の関連性も同時に把握可能となった．

6.2 今後の課題

本システムの解析機能は限定的であり，特に，詳細解析画面での情報の表示機能については，さらに検討する必要がある．選択したエッジ束についての統計量を Treemap のノードの色を用いて把握できるようにするなどの改善を検討している．また，使用していない次元があるため，それを制御点にして次元を増やすことで，さらに良い可視化結果が得られないかの調査を進めていく予定である．

謝辞

本研究は総務省による「国際連携によるサイバー攻撃の予知技術の研究開発」の支援を受けたものである．また，使用したデータは, 独立行政法人・情報通信研究機構(NICT)からダークネット観測データの提供を受けたものである．

参考文献

[1] Biddle, P., England, P., Peinado, M., & Willman, B., The darknet and the future of content distribution, ACM Workshop on Digital Rights Management, Vol. 6, p. 54, 2002.
[2] Bailey, M., Cooke, E., Jahanian, F., Myrick, A., & Sinha, S., Practical darknet measurement, Information Sciences and Systems(CISS), pp. 1496-1501, 2006 40th Annual Conference on IEEE
[3] Shneiderman, Ben., Tree visualization with tree-maps: 2-d space-filling approach, ACM Transactions on graphics (TOG), 11(1), pp.92-99, 1992.
[4] Bederson, Benjamin B., Ben Shneiderman, and Martin Wattenberg., Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies, ACM Transactions on graphics (TOG), 12(4), pp.833-854, 2002.
[5] Bruls, Mark, Kees Huizing, and Jarke J. Van Wijk., Squarified treemaps, Springer Vienna, 2000.
[6] Holten, Danny, Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data, IEEE Transactions on

Visualization and Computer Graphics, pp.741-748, 2006.
[7] Holten, Danny, and Jarke J. Van Wijk., Force-Directed Edge Bundling for Graph Visualization, Computer Graphics Forum, (Blackwell Publishing Ltd), 28(3), 2009.
[8] Taylor, T., Paterson, D., Glanfield, J., Gates, C., Brooks, S., & McHugh, J., Flovis: Flow visualization system, Conference for Homeland Security, Cybersecurity Applications & Technology, IEEE, pp. 186-198, 2009.
[9] Fischer, F., Mansmann, F., Keim, D. A., Pietzko, S., & Waldvogel, M., Large-scale network monitoring for visual analysis of attacks, Springer Berlin Heidelberg, 2008.
[10] Ball, Robert, Glenn A. Fink, and Chris North., Home-centric visualization of network traffic for security administration, Proc. of ACM workshop on Visualization and data mining for computer security. ACM, pp. 55-64, 2004.
[11] 政府における情報セキュリティ政策の取り組みについて, http://www.soumu.go.jp/main_content/000274855.pdf
[12] サイバー攻撃情報の類似性・局所性・時系列性解析技術の研究開発, http://itslab.inf.kyushu-u.ac.jp/cyber/jp/index.html