九州大学学術情報リポジトリ
Kyushu University Institutional Repository

OpenFlow を用いたハニーポットの動的配置手法

山内, 一将
九州大学 | 九州先端科学技術研究所

川本, 淳平
九州大学 | 九州先端科学技術研究所

松本, 晋一
九州先端科学技術研究所

堀, 良彰
佐賀大学全学教育機構 | 九州先端科学技術研究所

他

https://hdl.handle.net/2324/1662101

出版情報：電気・情報関係学会九州支部連合大会. 平成26年度(第67回), pp.11-1P-07-, 2014-09-18
バージョン：
権利関係：

OpenFlow を用いたハニーポットの動的配置手法

山内一将 *,***　川本淳平 *,***　松本晋一 *,***　堀良彰 **,***　櫻井幸一 *,***
(*九州大学院システム情報科学府情報学専攻　**佐賀大学全学教育機構　***公益財団法人九州先端科学技術研究所)

1　はじめに

社会の情報化が進むにつれて，ネットワークを流れる情報量は増大している．サイバー犯罪者は金銭を得る目的で，個人情報の搾取や改ざん，破壊活動などを行なう．また，サイバー攻撃は日々巧妙化しており，大量の情報の中でこれらの悪性通信を取り出すことが困難になっている．ハニーポットは攻撃者をおびきよせるための設定を意図的に行ったシステムであり，これを利用して悪性通信の監視を行なう．しかし，サイバー攻撃の標的となる PC は世界各地に点在しているため，ハニーポット用に静的に割り当てられる IP アドレス空間では攻撃者に関する情報も限定される．. そこで，本研究では OpenFlow を用いてハニーポットへを動的に配置することで効率的に悪性通信を集約することで，ユーザが安全にサービスを利用できることを目的としている．

2　SDN(Software Defined Network)

SDN[1] はネットワークをソフトウェアレベルで制御するためのアーキテクチャ，概念である．図 1 では従来のスイッチと SDN でのスイッチの比較を示している．ルータ，ファイヤーウォール，IDS などの従来のスイッチでは各スイッチ毎にパケット転送ポリシーを決定するためのソフトウェアが組み込まれている．これに対し，SDN ではコントロールプレーンとデータプレーンは分離しており，ネットワーク知性とステータスは論理的にコントロールプレーンに集約されている．故に，ハードウェア機器に依存することなくトラフィックの増減によってリソース配置や機能追加を柔軟に行うことを可能にする．この SDN を実現する技術の一つとして，OpenFlow[2] が注目されている．OpenFlow はコントロールプレーンとデータプレーン間のプロトコルのことである．従来のスイッチの制御を行なうための API は各ベンダ毎に用意されており，これらはユーザに公開されていなかったが，OpenFlow では API が標準化されているためユーザ独自に開発することが可能となる．

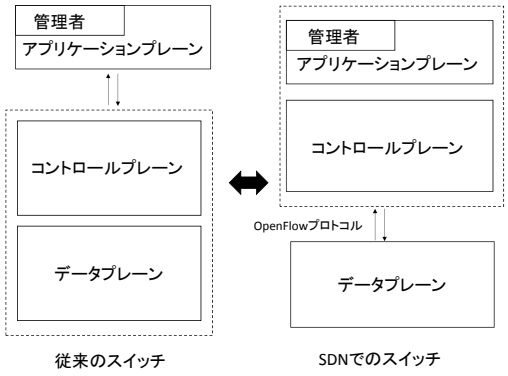

図 1: SDN の概念

3　関連研究

3.1　ハニーポット

ハニーポットとは，意図的に脆弱性を持たせることで，攻撃者に故意に不正アクセスをさせ，その手法の分析や重要なシステムへの侵入を防ぐための囮のコンピュータである．ハニーポットは攻撃者と対話するので攻撃手法を分析するために有効な手段であるが，攻撃者にハニーポットであることを検知されないように慎重に運用する必要がある．先行研究として，ネットワークの状態をリアルタイムで把握するために未使用の IP アドレスが割り当てられたハニーポットでネットワークの観測を行なう手法がある [3]．文献 [3] では，IP アドレスの動的割り当て機構である DHCP プロトコルと連携してハニーポットへの IP アドレス割り当てを行っている．しかし，ハニーポットまたはホストへ割り当てる IP アドレスの優先度を動的に設定できない．例えば，ハニーポットで悪性通信が観測された場合，これら以外の IP アドレスをホストに優先的に割り当てる必要がある．このような優先度の設定を元に動的な IP アドレス割り当てを行なうことで，ユーザが安全にサービスを利用することを可能にする．

3.2　OpenFlow

OpenFlow は SDN を実現させる仮想化技術の一つであり，ネットワークをソフトウェアレベルで管理することにより物理的な位置に縛られないネットワーク管理を可能にする．OpenFlow スイッチ (以下，スイッチ) では観測されるパケットをフローレベルで制御する．スイッチには少なくとも一つのフローテーブルがあり，これらは OpenFlow コントローラ (以下，コントローラ) によって管理される．コントローラはスイッチとセキュアチャネルで接続し，Openflow プロトコルを用いてスイッチの制御を行っている．具体的には，スイッチはフローテーブルと呼ばれるパケットの処理方法をルール化したものがあり，これに従ってパケット処理を行なう．フローテーブルは L1 層から L4 層までの幅広いパケットの転送ルールを定義できる．また，コントローラはフローテーブルの更新や削除を自由に行なうことができる．OpenFlow をセキュリティに応用する手法が先行研究としてある [4]．文献 [4] では送信元 IP アドレスと TCP コネクションのエラー数を元に，パケットを正規の宛先ホストへ転送するかハニーポットへリダイレクトさせるかを決定する．この場合，攻撃者に気づかれずにハニーポットは攻撃通信を観測することができる．しかし，送信元 IP アドレスと TCP コネクションエラーのみの情報では攻撃の判定が困難である．

4　提案手法

本研究では悪性通信を効率的に収集するためにハニーポットの配置を動的に決定する手法に関して述べる．ハニーポットの配置を動的に決定するために OpenFlow を用いる．図 2，図 3，図 4 で示すネットワークの各要素の役割，設定方法について説明する．

ハニーポットでは，未使用の IP アドレスを監視する．例えば，ボットネットで感染したボット群がスキャン攻撃などで宛先の IP アドレスを無作為に指定して通信するもの

をハニーポットで検知したと仮定する．このとき，ハニーポットが正規システムを持たない IP アドレスのふりをすることで相手のボットの特定が出来るだけでなく，ボットの背後に隠れた同じボットネットに属する他のボットや C&C サーバに関する情報まで取り出す事が可能となる．

スイッチでは，フローテーブルに従ってパケット処理を行なう．スイッチのフローエントリは以下の様に記述できる．

$if(ip\_dst == host\_ip)\ output\ host\_port;$
$else\ output\ honey\_port;$

コントローラは制御している各ネットワークでの IP アドレスの割り当て範囲とその中での使用状況を把握している．使用状況を把握するために Nmap を利用して各ネットワークで割り当てられている IP アドレスの範囲全てに ping パケットを送り，リプライが返ってきたものに関して，IP アドレスは使用されているものとする．

悪性通信を効率的に集約するために，具体的なハニーポットの配置方法に関して図 2 に示す．本研究では 1 台のハニーポットで組織内に割り振られた IP アドレス区間の中の未使用 IP アドレスを監視する．コントローラでは IP アドレスの使用状況をリスト化したものを作成する．図 2 では 133.5.17.B，133.5.17.C，133.5.17.D が使用されている IP アドレスとなり，これ以外のトラフィックに関してハニーポットへリダイレクトさせる．

また，図 3 ではネットワークに接続していた Host1 が

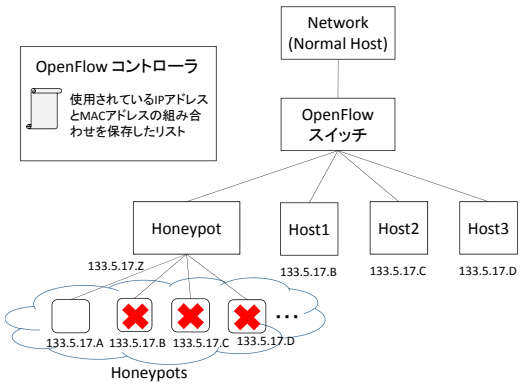

図 2: ハニーポットの配置方法

ネットワークを遮断した場合について示す．まずコントローラが Host1 がネットワークから遮断されたことを検知し，リストの更新を行なう．そして，Host 1 の IP アドレスはハニーポットでエミュレートされたサービスに割り当て外部ネットワークからは Host1 からサービスが継続されたように見せる．Host1 はネットワークを再接続する可能性があるため，攻撃を受けていない IP アドレスを優先的に割り振る必要がある．例えば，Host1 で使用していた 133.5.17.B という IP アドレスが，ハニーポットでの観測によって攻撃が検知された場合，Host1 がネットワークに再接続するときには図 4 のように他の未使用 IP アドレスである 133.5.17.A を割り当てることで攻撃を回避する．

5 まとめ

本研究では OpenFlow を用いて広域なネットワーク制御を行い，悪性通信を効率的に集約するためのハニーポットの動的配置手法に関して示した．今後の課題としては，ハニーポットでの悪性通信集約におけるトラフィックの負荷を考慮する必要があると考えている．

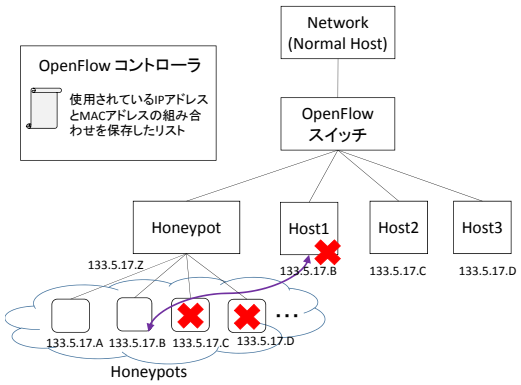

図 3: ハニーポットの配置方法（Host 1 がネットワークを遮断した場合）

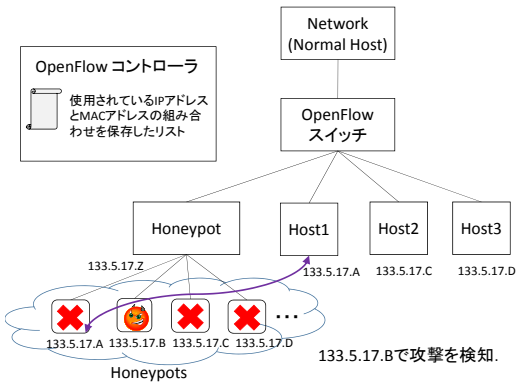

図 4: ハニーポットの配置方法（Host1 がネットワークに再接続した場合）

謝辞

本研究を進めるにあたり，株式会社 KDDI 研究所溝口誠一郎氏，松中隆志氏，窪田歩氏より助言をいただきました．ここに感謝いたします．

参考文献

[1] 高宮安仁，鈴木一哉，December 2012 “OpenFlow 実践入門” , 技術評論社

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner,”OpenFlow: enabling innovation in campus networks”,SIGCOMM Computer Communication Review, Volume 38, Number 2, pp.69-74, ACM, April 2008

[3] Seiichiro Mizoguchi, Yoshiaki Hori, Kouichi Sakurai, “ Monitoring Unused IP Address on Segments Managed by DHCP”, NCM ’08 Proceedings of the 2008 Fourth International Conference on Networked Computing and Advanced Information Management - Volume 01 Page 510-515

[4] Seugwon Shin, Phillip Porras, Vinod Yegneswaran, Martin Fong, Guofei Gu, Mabry Tyson, “ FRESCO: Modular Composable Security Services for Software-Defined Networks”, ISOC Network and Distributed System Security Symposium, February 2013