# 2009 年度上期未踏 IT 人材発掘・育成事業　採択案件評価書

## 1. 担当PM

首藤　一幸 PM(東京工業大学　大学院情報理工学研究科
数理・計算科学専攻　准教授)

## 2. 採択者氏名

チーフクリエータ:　鈴木　友博(東京大学大学院　情報理工学系研究科)
コクリエータ　　:　なし

## 3. プロジェクト管理組織

株式会社メルコホールディングス

## 4. 委託金支払額

2,747,431 円

## 5. テーマ名

仮想ネットワークを利用した攻撃者監視システムの開発

## 6. 関連Webサイト

なし

## 7. テーマ概要

複数の仮想的なホストを内部に持つ攻撃監視システム(ハニーポット)を実装する。システムの内部に仮想的なホストを作り出し攻撃をそこへリダイレクトさせることでシステム外部のホストへ危害を与えることなくハニーポット内での攻撃者の行動を観察

することができるのが特長である。

また、これらのホスト全体でログを得られるという利点を活かし、従来の各ホスト上のローカルな情報(攻撃者がどんなコマンドを打ったか)に加えて、ホスト間にまたがったグローバルな侵入者の情報(どのホストに侵入したか)を組み合わせた侵入者やマルウェアの情報の提供をグラフィカルに行う。具体的には、システム内のホストでの侵入者の行動記録を特定の時間とホストを指定することで「再生」して見られるようなインターフェイスを作る。

既存のハニーポットシステムでは攻撃者の観察と同時に外部のホストを安全に保つのは難しい。また、ハニーポット内から外部へ向かう接続をリダイレクトする研究はいくつかあるが、そのシステム自体のセキュリティの問題や何台もホストを用意したときの使いやすさ、そして利用者へ侵入者の情報を提供する方法が十分とは言えず、今回のプロジェクトではこれらを解消するシステムを構築する。

## 8. 採択理由

サーバ、PC 等へのネットワーク越しの攻撃者をハニーポットというおとりマシンの上で泳がせる、という防衛・解析手法が盛んに研究されている。提案者は、マシン1台ではなくて、マシン「群」を見せて、そのマシン群のネットワーク上で泳がせることを提案している。提案テーマでは、泳がせるためのソフトウェアシステムと、攻撃者の(おとり)ネットワーク上での挙動を理解するための可視化ツールを開発する。

大学院での研究テーマと兼ねるものの、このテーマを考え付いたのは当人だという。アイディアの有用性を示して、現実性がなくもないということさえ示すことができれば、大学院での学業、研究としては成立する。査読をパスできる論文が書けるだろう。ただ、それだけだとすると、あえて未踏で取り組むことの意義は薄い。

一方で、このテーマで実用ソフトウェアを開発するのは大変すぎる。手作業での侵入を試みる攻撃者を騙し切ることのできるハニーポットを開発するには、極めて膨大かつ地道な作業が必要となる。とはいえコンセプトの提示だけで満足して欲しくはない。現実的な到達点は、既存の攻撃ツールをいくつか想定して、それらを騙し切ることのできるところまで作ってデモし、現実性を示す、というあたりだろうか。

オーディションである PM が指摘したように、攻撃者、攻撃ツールの挙動をいかに可視化するかという点に注力する、という方向もいいだろう。むしろ、研究と実地の狭間のどのあたりに注力していくか、提案者自身に悩んでもらい、その中で自身の方向をつかんでいってもらいたい。

セキュリティやソフトウェア工学を含めた技術への意識の高さ、また、インターンや技術者の集まりに出て行く意欲などにも期待する。

## ９. 開発目標

このプロジェクトで開発するソフトウェアは、「仮想ホスト」、「攻撃のリダイレクション」、「UI 部」の3つに分けられる。

「仮想ホスト」は攻撃者に侵入されるサンドボックスのような環境を用意する。異なった仮想ホスト中にあるプロセスはあたかもそれぞれが異なったマシン上で実行されているかのような振る舞いをする。ここでプロセスとは攻撃者のコントロールするシェルプログラムやマルウェアが含まれる。

「攻撃のリダイレクション」では仮想ホストの中から外のコンピュータへ向けられた接続を別の仮想ホストへ振り向ける。これにより攻撃者による仮想ホストを踏み台にした別のコンピュータへの侵入は全て他の仮想ホストへ振り向けられ、一見成功したように見える攻撃でも実際には別の仮想ホストに侵入していることになる。

「UI 部(Viewer)」は honeypot 内部で何が行われているかを管理者に提供するインターフェイスである。UI 部は honeypot が実行されているコンピュータとは別のコンピュータで実行し、honeypot のコンピュータから送られてくるメッセージを処理し、仮想ホストを表すノードや攻撃を表す情報を表示する。

## １０. 進捗概要

予定していたすべての機能、ソフトウェアを実装した。

１台のホスト上に数千から万におよぶ多数の仮想ホストを用意するために活用できる手法には目当てのものがあった。しかし実装は、クリエータ自身があるべき構造を考え直して新たに行った。

## 11. 成果

図1に、開発した honeypot システム全体の概要を示す。

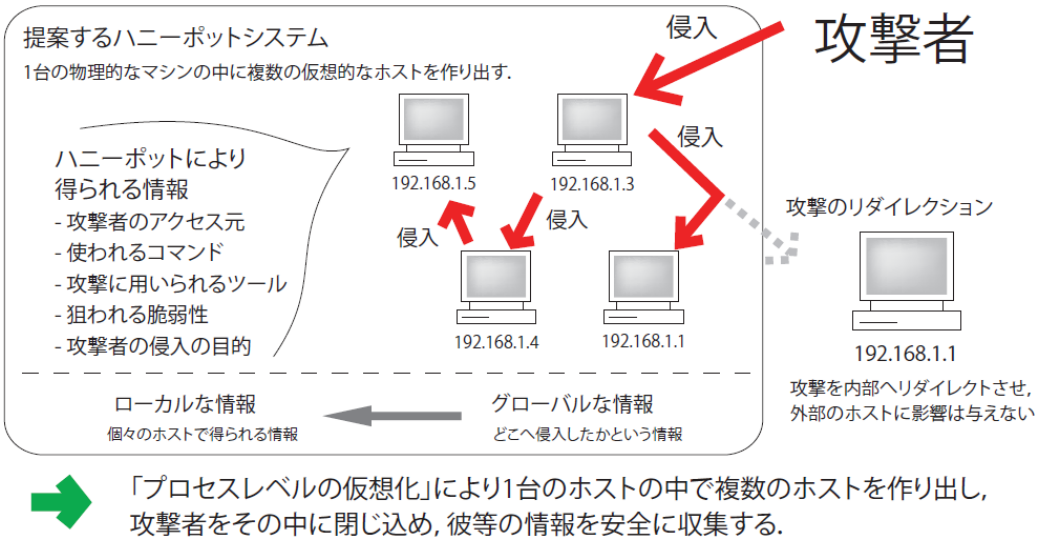


図1:システム全体の概要

■ 仮想ホスト

仮想ホスト機能は、攻撃者はマルウェアのプロセスに対して、1台のホストを多数のホストに見せかける機能である。通常の PC 上で数千以上の仮想ホストを動作させることができる。

今回の honeypot 内の仮想ホスト上で動いているプロセスは実際はすべて単一の OS 上で動いているプロセスである。プロセスが発行するシステムコールの動作をそのプロセスが属している仮想ホストを元に変えることによってカーネル内部の動作を変化させる。プロセスがどの仮想ホストで動作しているかを識別するために、その情報をプロセスを表すカーネル内部の構造体(task struct 構造体) に付け加え、その情報に基づいてプロセスとカーネルの通信であるシステムコールのふるまいを変更している。このために task struct 構造体に与えた変更はプロセスがどの仮想ホストに属しているかという情報だけで済み、32 ビットの値をもつ構造体のメンバを付け加えるだけで充分であった。

■ 攻撃のリダイレクション

攻撃のリダイレクションは、仮想ホストが外のコンピュータへ攻撃を行った場合にその接続を別の仮想ホストに振り向ける機能である。攻撃のリダイレクトは、具体的には、プロセスがネットワーク接続のためのシステムコール(sys connect)を発行した際に呼び出されるフックをカーネル内部に挿入することにより、このシステムコールの

宛先を書き換える。

本提案の honeypot ではこのシステムコールの宛先である IP アドレスを自身の実ホストにし、ポート番号を特定の daemon が待ち受けている番号に書き換えることにより、実際には単一 OS 内の複数プロセスであるにも関わらず、複数のホストがネットワークを通じて通信をしているかのように見せている。

■ UI 部(Viewer)

UI 部では仮想ホストを表すノードの表示とグラフの表示を実装した。

UI 部は honeypot が動いている時にリアルタイムでどのホストで何が行われているか(どのような端末入力があるか)を表示することができる。リアルタイムで honeypot を監視している場合をリアルタイムモードという。

また、記録された情報をもとに honeypot 内部で行われたことを再生する機能を持つ。再生機能を用いて動作させている場合をリプレイモードという。

図2に、仮想ホストへの端末入力を表示している様を示す。図3、4に、仮想ホスト間の通信を可視化している様、図5に仮想ホスト上のアクティビティ(通信量等)を可視化している様、図6に複数仮想ホストのアクティビティをまとめて可視化している様を示す。

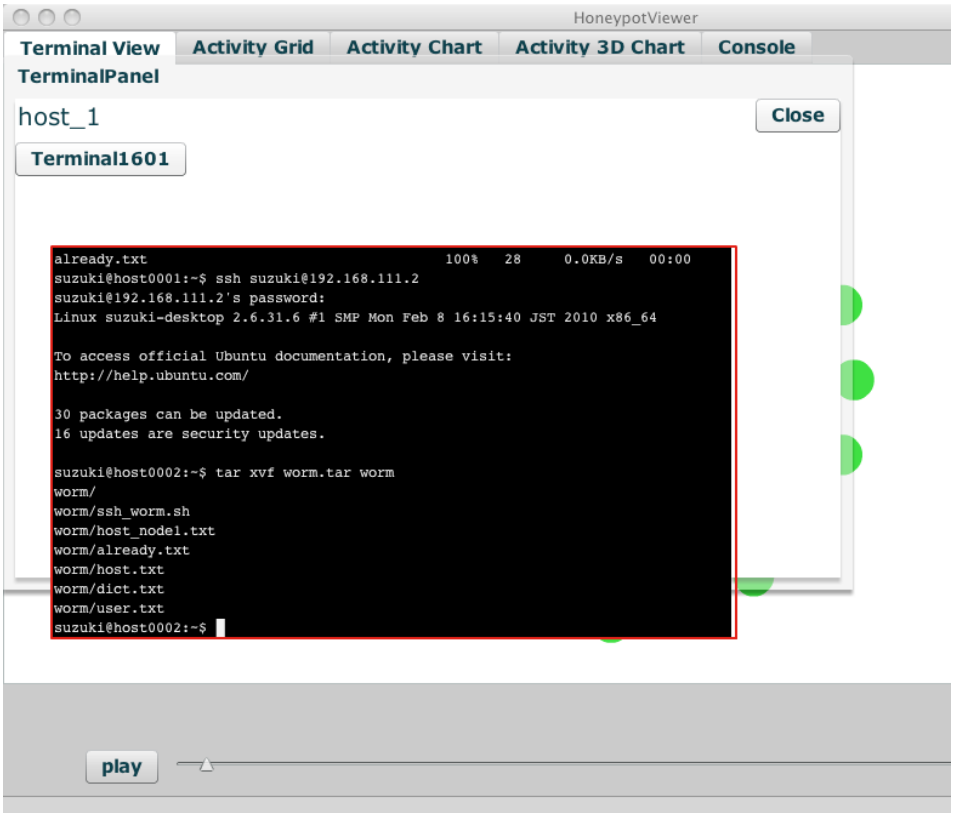


図2:仮想ホストへの端末入力の表示

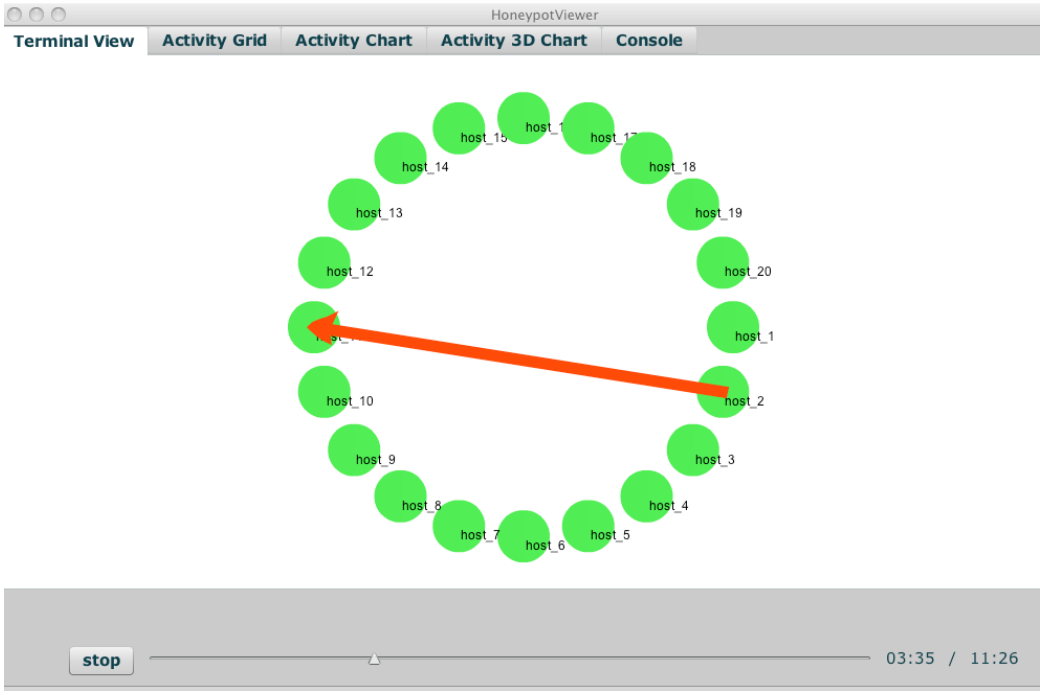


図3：仮想ホスト間の通信を可視化している様

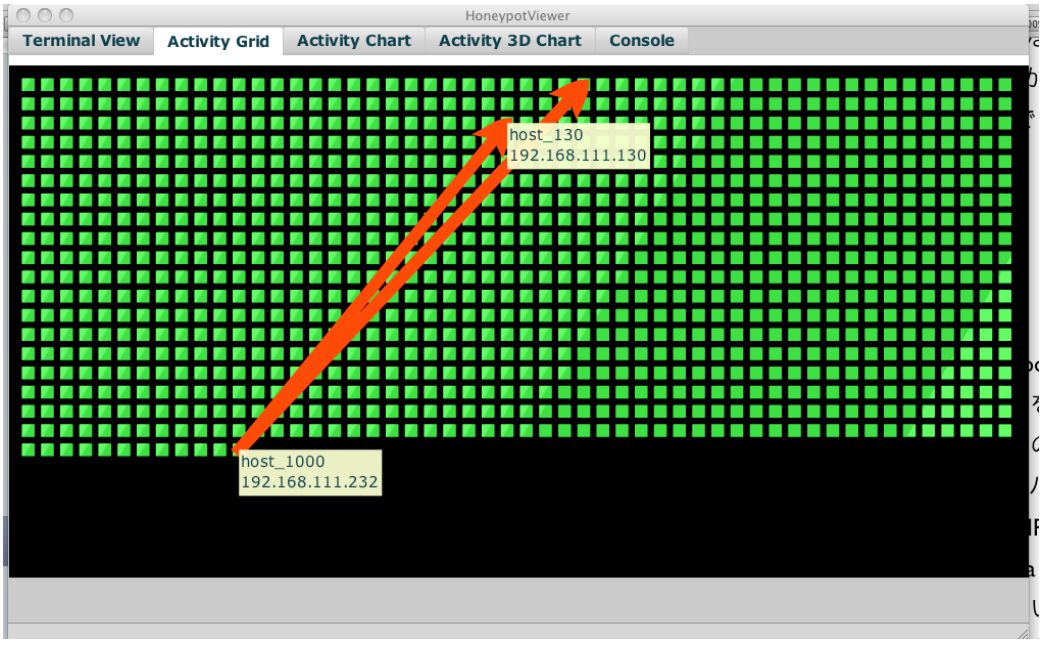


図4：仮想ホスト間の通信を可視化している様

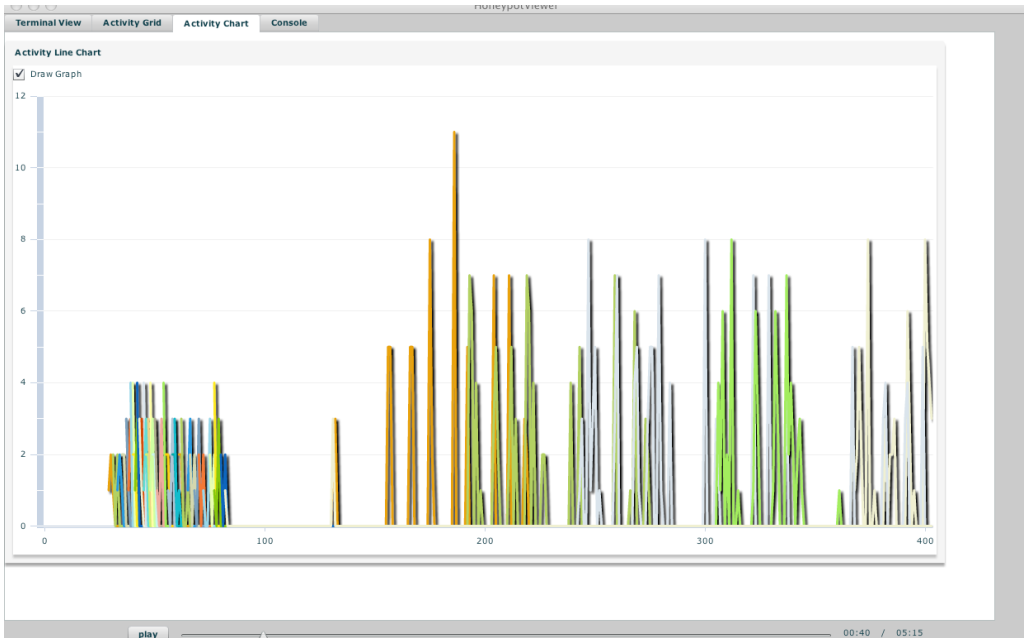


図5：仮想ホスト上のアクティビティを可視化している様

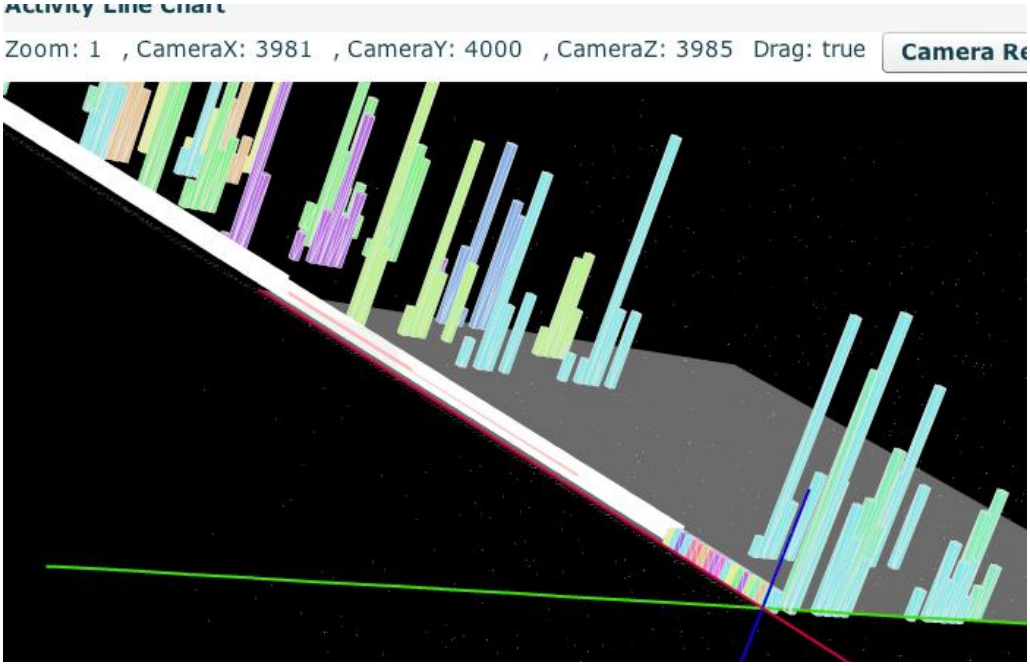


図6：複数の仮想ホスト上のアクティビティをまとめて可視化している様

## 12. プロジェクト評価

このプロジェクトもやはり、言うは易し、行なうは難し、である。1台のホスト上で数多くの仮想ホストを動作させる技法はクリエータの研究室で開発・提案されていたものであったが、クリエータ自身が拡張性や保守性を考えて開発し直した。動作確認を行なう方法も、現実のマルウェアを探すなど様々な試みの後、開発期間中については自身でサンプルのワームを開発することとなった。可視化も、思いつくのはたやすく、実装には大変な腕と労力を要する。

成果は、アカデミックな場での発表につながるだけでなく、セキュリティの実務家に対しても（対してこそ？）アピールするのではないかと考えている。ワームを誘導するという honeypot の通常用途だけでなく、ワームの挙動を調べるなど実験・研究を行なう上で有用な手法・技法であり、見る者に将来のセキュリティ研究プラットフォームを感じさせるのではないだろうか。そちらへのアピールも大変期待している。

## 13. 今後の課題

多くの仮想ホストを管理する手段、実地での妥当性・有用性の確認といった技術的な課題の他に、どういった可視化が有効かを調査・検討するといった人的要素まで関係する課題もあり、今後の展開には事欠かない。