2004－CSEC－27　(11)
2004／12／20

行動制限型ハニーポットの改良方法の提案・実装・運用

小　泉　芳[1]　　小池英樹[2]　　安村　通晃[3]

慶應義塾大学　大学院政策・メディア研究科[1]　電気通信大学　大学院情報システム学研究科[2]　慶應義塾大学　環境情報学部[3]

本研究では不正侵入者の行動観察を目的とした行動制限型ハニーポットにおける幾つかの改良方法を提案する．改良のコンセプトはラッパー方式によるコマンド制御を行うことで、ハニーポットであるとより気付かれ難く、また設置・運用をより効率的に行うものである．主な改良機能として特殊 OS の偽装（ダミー OS)、偽の対話インターフェイス（ダミープロンプト)、コマンドに対する偽の返答（ダミーメッセージ）などである．本論文では改良システムの設計・実装について説明し、また運用を通じて得られた侵入者の行動に関する知見を述べる．

A Proposal and Development of Honeypot with Command Wrapper for Activity Restricted

Kanba KOIZUMI[1] , Hideki KOIKE[2]
and Michiaki YASUMURA[3]

Graduate School of Medea and Governance, Keio University[1] Graduate School of Information Systems, University of Electro-Communications[2] Faculty of Environmental Information, Keio University[3]

In this paper, we propose and develop the honeypot with the activity restricted. Proposal system is for the purpose of intruder monitoring and consist of command wrapper. Command wrapper has data capture and data control ability. It's basically output dummy message and dummy prompt that pretend UNIX like OS. We explain the detail of our system, and knowledge by monitoring intruders.

1. はじめに

侵入者の行動観察用システムとしておとりシステム (以下、ハニーポット）が注目されている．ハニーポットはわざと脆弱なサービスを立ち上げることで侵入させ、侵入者の行動を記録するシステムである．ハニーポットの中には、シェル（コマンドベースの対話用インターフェイス）へのアクセスを許す高対話型ハニーポットやシェルの機能を制限する中対話型ハニーポットがある．本研究では、行動制限型である中対話型ハニーポットを主な対象として議論する．

中対話型ハニーポットはシェルの機能制限などにより、高対話型ハニーポットが持つリスクの高さの緩和や運用の負担を軽減させる利点がある．しかし問題点として、シェルの機能制限により高対話型ハニーポットに比べ得られる情報が少ないこと、通常システムとハニーポットの識別が侵入者に容易であること、ハニーポットの設置の困難さがある．

本研究では既知の脆弱性を利用する侵入者の流行（利用ツール、コマンドなど）分析を主な目的として行動制限型ハニーポットに対する改良方法を提案する．改良のコンセプトはラッパー方式によるコマンド制御機能を追加することで、ハニーポットであるとより気付かれ難く、また設置をより効率的に行うものである．特に侵入者を欺くための幾つかの改良方法を追加することで、情報制御 (偽装) をする．改良型ハニーポットは侵入者にダミープロンプトを提供し対話させ、入力に対して制限コマンドにはダミーメッセージを返すシステムである．また特定の OS を模倣するのではなく、UNIX ベースの特殊ケース (ダミー OS) として振舞い、侵入者が利用すると予想される限られたコマンドのみに対応させる．

本論文では提案に基づくプロトタイプの実装・運用を行った．また運用中の 20 回の不正侵入の内容報告（運用知見）および考察を行う．

以下の構成は２章で先行研究、３章で設計と実装、４章で運用報告、５章で考察である．

表 1　対話レベルごとのハニーポットの特徴

対話レベル	利点	欠点
高対話 OS レベル	侵入者の行動を全て観察 侵入者に気付かれにくい 得られる情報が多い	リスクが高い 設置が困難 運用コスト大
中対話 コマンド制限	侵入者の行動を一部観察 運用コスト少 リスク少	得られる情報少 設置がやや困難 識別が容易

2. 先行研究

2.1 ハニーポットの分類

ハニーポットは利用目的から研究用と商用に分けられる [5]. 研究用とは侵入者から学び, 侵入行為そのものを研究の対象とする. また商用とは主に侵入者をおとりに誘導し正規サーバーなどを防護するためのものである. 本研究の対象は研究用ハニーポットであり、侵入者の行動観察が目的である. 研究用ハニーポットの主な目標としては、未知の脆弱性を狙った攻撃（0-Day Attack）の発見をすること、取得データから教育・啓蒙をすること、流行分析をすることが挙げられる. 主な分析項目としては戦術（スキル）、ツール、動機などがある.

2.2 研究用ハニーポットの問題点

侵入者の行動観察用ハニーポットはその対話レベルにより高対話型と中対話型に大別される. 表 1 に高対話と中対話の利点・欠点の一覧を示す.

(1) 高対話型ハニーポット

高対話型ハニーポットの代表として Honeynet Project[3][4] がある. Honeynet は基本的におとりであるホストの OS(シェル) を侵入者に直接操作可能にしている. 高対話型ハニーポットの機能はデータキャプチャとデータコントロールに分けて考えられる. データキャプチャとしては通常侵入者におとりであると気付かれることなく監視をし、侵入者の全部の行動 (ありのままの姿) の記録をする. 侵入者に気付かれることなく行動記録を取得するために、カーネルレベルでのキーストローク記録ツールやシェル (bash) の改造版などが利用されている.

データコントロールとしては、外部への攻撃（踏み台）など加害者になりうるというというリスクがあるため、ネットワークへの接続にはある程度の制限をする. これにはパケットの回数制限や既知の攻撃を無効化するツールなどが利用されている. しかしこれらの対策は絶対に安全である保証はないため、管理者はハニーポットを常時監視しなければならなく、運用時のコストが高い.

高対話型ハニーポットの最大の目標は未知の脆弱性を利用した攻撃を発見することであるが、世界規模で活動する Honeynet Project においても未知の攻撃の発見は数件程度である. これよりハニーポットは未知の脆弱性の発見という最大の目標よりも、侵入者の流行分析（利用ツール、コマンドなど）という主旨が強くなっている. このような状況から従来の高対話型ハニーポットはレベルの低い侵入者のためにコストの高いシステムを構築・運用しているといえる.

(2) 中対話型ハニーポット

中対話型ハニーポットは高対話型の問題である踏み台などのリスクを緩和させるために、侵入者の行動制限をすることが主な特徴である. 侵入者の全ての行動を観察するのではなく、制限された行動の観察（記録）をする. 中対話型ハニーポットのデータコントロールは jail, chroot, rbash (restricted bash) などの環境を制限するコマンドの機能を利用して実装されている. しかし、単なるコマンド制限ではハニーポットであることの識別が容易であり、侵入者に気付かれ易い. 特に制限コマンドの返答内容により監視している（ハニーポットである）ことが露呈しやすく、なんらかの制限をしていることが侵入者に容易に識別可能である. 中対話型ハニーポットでは、安全性を強調する反面、侵入者にハニーポットであると気付かれ、ありのままの姿を観察できなくなり得られる情報が少なくなる傾向がある.

その他の問題として、カーネルレベルでのセキュリティホールを利用すれば管理者権限を奪取することは可能であるためリスクは依然残る. また jail や chroot などは、UNIX 固有の機能であり、Windows などでは利用できないなどの汎用性の課題がある.

2.3 改善の方針

改善の方針は行動制限方法として、ラッパー方式を用いることで入出力の制御を行い、コマンド制限や監視をしていることを侵入者に気付き難くすることである. またラッパー方式により既存のシステムを変更せずにラッパーのみを変更することで設置を可能にする. これらはハニーポットの識別を困難にし、システム構築のコストを下げるためである.

3. 設計と実装

本章では提案システムの設計方針と実装の詳細について述べる.

3.1 コマンドラッパーの設計方針

設計の概要は、既存システムを修正するのではなく、コマンドラッパーを用いて入出力を制御することである. 以下コマンドラッパーの設計の詳細を述べる.

3.1.1 コマンドラッパーの機能要件

コマンドラッパーの機能要件はデータキャプチャとデータコントロールに分けて考えられる.

- データキャプチャ
  侵入者の行動記録（キーストローク）をとる. 例え暗号化通信 (ssh) を用いても、複号後にコマンドを記録し入力コマンドは全て記録する.

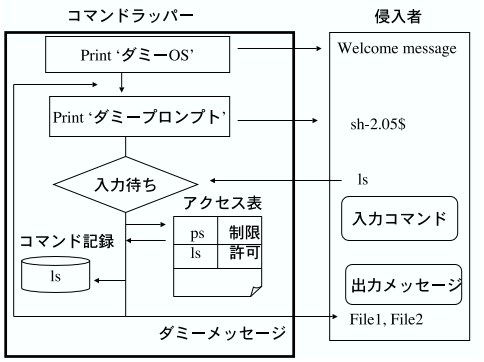

図 1　コマンドラッパーの動作フロー

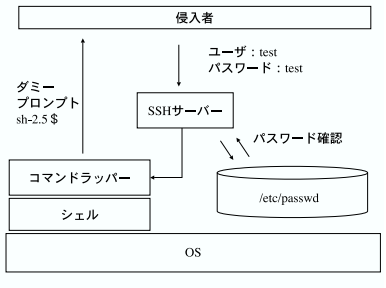

図 2　ラッパーへのエントリーポイント
ssh からアカウントの不正利用する場合

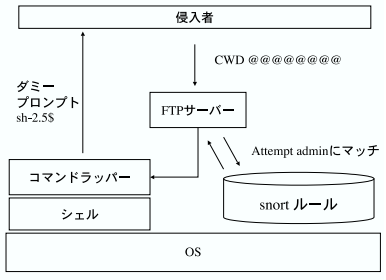

図 3　ラッパーへエントリーポイント
ftp から権限奪取の場合

- **データコントロール**
  侵入者との対話インターフェイスであるダミープロンプトを提供し、侵入者の入力コマンドに対して、ダミーメッセージ（予め用意した定型データやエラーメッセージ）を返答する. また侵入者に一般的な Linux **のディストリビューションではない** Unix **ベースな特殊なシステム（ダミー** OS**）と思わせるようにする. ダミーメッセージでは、例えば** ps **コマンドによるプロセス表示ではコマンドラッパーなどのハニーポットに関わるプロセスを表示せず（情報を削除し）返答し、監視をしていることを侵入者に気付き難くする.**

中対話型ハニーポットで目指しているのは完全なデータコントロールであり、許可コマンド以外は基本的にネットワークを利用させず、リスクを完全に無くす方針である. 従ってデータコントロールに重きを置き、捕捉データが多少減る可能性は否定できないが、それは設計の方針である.

3.1.2　コマンドラッパーのアルゴリズム

以下にコマンドラッパーのアルゴリズムを説明する. また動作フローを図 1 **に示す.**

(1) **ウェルカムメッセージ（ダミー** OS**）の出力. 存在しないディストリビューションを出力する.** Welcome Anzen Linux 2.4

(2) **コマンドプロンプトと呼ばれる入力を促す文字（ダミープロンプト:** sh-2.05$**）を侵入者に送り、コマンド入力を待ち受ける.**

(3) **コマンド入力に対して. 許可コマンドであれば実行する. 拒否コマンドであれば、実際に実行せずに、ダミーメッセージを返答する. またキーストロークをファイルに記録する**

(4) **コマンドが** exit **などの終了を意図したものであれば接続を終了する. それ以外であれば** (2) **に戻る.**

3.1.3　コマンドラッパーへのエントリーポイント

コマンドラッパーを呼び出す（侵入者と対話させる）にはアカウントの不正利用と既知の攻撃方法の２つを用いる.

- **アカウントの不正利用**
  ssh 用のログインシェルとして、コマンドラッパーを呼び出す. 脆弱なパスワード test **を持つユーザ** test **を追加し、**/etc/passwd **ファイルを修正してログインシェルのパスをコマンドラッパーにする. 図** 2 **に概要を示す.**
- **アプリケーションのセキュリティホールの利用**
  snort **のルールにおいて** ftp **への管理者権限奪取** (attempted admin) **行為にマッチした場合にコマンドラッパーを呼び出す. 実装においては** snort **のルールの正規表現である** pcre **を利用した. ftp 本体は低対話型ハニーポットを利用し、ルールにマッチした段階でコマンドラッパーを呼び出すことで制御を切り替え、以降はコマンドラッパーと侵入者との対話モードとなる. これより侵入者に攻撃が成功してシェルを奪取したと思わせる. 図** 3 **に概要を示す.**

3.1.4　コマンドラッパーの偽装機能

コマンドラッパーは従来の中対話型ハニーポットに対して、侵入者への情報制御をすることが主な改良である. 情報制御は侵入者を欺くことが目的である. 主な偽装機能として、特殊 OS **の偽装 (ダミー** OS**）, アプリケーションレベルでの対話インターフェイス（ダミープロンプト）, 制限コマンドに対する偽の返答（ダ**

ミーメッセージ）がある.

- **特殊 OS の偽装（ダミー OS）**
  従来の中対話型ハニーポットが侵入者に識別が容易であることの理由として、汎用的な OS を前提としたからと考えられる. そこで改良方法として侵入者に不馴れな特殊な OS を偽装することで識別を困難にする.
- **偽の対話インターフェイス（ダミープロンプト）**
  侵入者にとって、ハニーポットへのアクセスはプロンプトを通じて行われる. 改良方法として本物のシェルは提供しないが、あたかも本物のシェルと対話していると侵入者に思わせるために、ダミープロンプト（$や#などの特殊文字）を侵入者に提供する.
- **コマンドに対する偽の返答（ダミーメッセージ）**
  従来の中対話型は制御コマンドへの扱いが粗雑であるため、侵入者にハニーポットの識別を容易にしていた. 改良方法としてはコマンドに対して不信感を緩和させるダミーメッセージを返答して、ハニーポットの識別を困難にする.

3.2 実　　装

次に実装環境について述べる.

3.2.1 実 装 環 境

実装環境は OS **に** RedHad Linux 7.3 **を用い,サービスの偽造は** ssh **サーバーおよび低対話型ハニーポット** single honeypot[6] **の** ftp **エミュレート機能を利用した. コマンドラッパーは** perl **言語を用いて実装した.**

3.2.2 入力コマンドとダミーメッセージ

ここでは入力コマンドと対応するダミーメッセージについて実装内容を述べる. 出力メッセージは事前にファイル形式で保存しておく. 例：ps ＞ ps.txt

コマンドラッパーは制限コマンドであれば、ダミーメッセージを返答する. ダミーメッセージは perror **関数などで利用されるシステムエラーが主体である. 以下に制限コマンドとダミーメッセージを示す.**

- **返答データを出力するコマンド**:
  ps, id, uname, netstat
  例：print ps.txt
  ただし、ハニーポットの識別に影響する情報 (コマンドラッパーに関するプロセス、OS **バージョンなど) は表示しない.**
- **権限なしエラーを出力するコマンド：**
  cat, rm, reboot, shutdown, mkdir, mv, more
  例：print “Permmision deny”
  ファイルの中身などは一切表示しない.
- **ネットワーク系コマンド**:
  ftp, ssh, nc, telnet, lynx
  ・ホスト名がドメイン名の場合
  print “Name resolution failture”
  ・ホスト名が IP **アドレスの場合**
  print “Network host unreachable”
- **特殊コマンド：** passwd
  コマンドと同様の文字を出力する.
- **返答なしコマンド**
  （カレントディレクトリの変更、環境設定、所有権の変更など）
  cd, set, unset, chmod, chown
- **終了コマンド**: exit, quit
  返答せずコマンドラッパーを終了する.
- **ダウンロードファイルの実行**: ./file
  print ”Segmentation fault”
  実行エラーメッセージを出力する.
- **その他コマンドなし：**
  print “Command not found”

またラッパーは許可コマンドであれば実際にコマンドを発行する. 以下に許可コマンドを示す.

- ls
  ファイルシステム（ファイルやディレクトリ）表示コマンドは実際に実行する. 侵入者に気付かれ難くするためである.
- wget
  ツールのダウンロードコマンドは実際に実行する. ツールを取得するためである.

改良ハニーポットの特徴はわずか２つの許可コマンドの発行を行うことで侵入者を欺くことである. 従来のコマンド制限型ハニーポットでは、この２つのコマンドだけでは直ちに不信感を抱かせるが、ダミーメッセージなどにより許可コマンドの補完をしている.

4. 運 用 報 告

ここでは実装システムの運用結果について述べる.

4.1 運 用 環 境

設置環境は 5 **つの異なるネットワークにハニーポットを構築した. 運用は** 2004 **年** 8 **月** 30 **から** 10 **月** 30 **までの** 2**ヶ月間行った. 各ホストとも** ssh **サーバーおよび偽装** ftp **を構築し、その他のサービスは利用しない. 運用中に** 20 **回の侵入があった. 全て** ssh **から脆弱なパスワードを用いたアカウントの不正利用であった.**

4.2 運 用 知 見

本システムの運用経験からの知見を述べる.

4.2.1 一連の行動記録

提案システムのデータキャプチャ能力はキーストロークが主体である. 以下取得したキーストロークの一例 (表 2**) から侵入者の行動分析を行う. 以下１回目の侵入時における侵入者の一連の行動の抜粋記録について説明する.**

(1) **コマンド履歴を初期化. 返答なし.**
(2) **ログインユーザの確認.** root **と** test **がログインしているダミーメッセージを返答.**
(3) **バージョンの確認.** Anzen Linux **というダミー**

表 2　侵入者のキーストローク

1. unset HISTFILE
2. w
3. uname
4. cd /tmp
5. wget *****.ro/loginx.zip (*は伏せ字)
6. passwd
7. chmod +x loginx.zip
8. ./loginx.zip

OS を返答.

(4) カレントディレクトリを/tmp へ移動.

(5) wget でダウンロードコマンドを実行.
ルーマニアのサイトへツールの取得をする. ラッパーは wget は許可コマンドであるため実際にコマンドを発行.

(6) パスワードの変更.
ラッパーは古いパスワードと新しいパスワードの入力を促すダミーメッセージを返答し侵入者の入力したパスワードを記録した.

(7) ダウンロードツールに実行権限を付与. ダミーメッセージはなくダミープロンプトのみを返す.

(8) ダウンロードツールの実行.
コマンド実行は制限しているのでダミーメッセージ（Segmentation fault）を返答する.

一連の行動から、侵入者はログイン後コマンド履歴の初期化やログインユーザの確認し、ツールのダウンロードを試みている. その後パスワードの変更をし、ダウンロードツールに実行権限を与えて、ツールの実行を試みている. ただし、ツールの実行はラッパーで許可していないためダミーメッセージが返答される. この後は所有権の再設定や別ツールのダウンロードなどを試みており、結局なにもできずに立ち去っていった.

4.2.2　侵入者の手口

管理者権限の奪取

侵入者がダウンロードしたツールについてオフライン解析を行った結果、ローカルホストでユーザから管理者権限を奪取するツールであることが解った. これは、2003 年に発見された Linux の do_brk 関数に存在するカーネルセキュリティホールを狙うものであり、Linux のカーネルバージョンが 2.4 以前であればユーザから管理者に昇格可能である. このセキュリティホールは侵入者たちの常とう手段のようで取得した大半のツールは同じセキュリティホールを狙うものであった.

秘密のディレクトリ

侵入者の特徴として、ファイルをダウンロードするさい、管理者に気が付かれない場所を好む. このような秘密のディレクトリの作成において、功名であるものを紹介する.

- スペース文字
  mkdir " "
- ドット（.）文字
  mkdir ...

このようなディレクトリはファイルの詳細表示においても識別が困難であり、侵入者側も管理者を欺くために工夫をしていることが解った.

破壊的行動

中対話型ハニーポットの特徴として、ハニーポットであることが侵入者に識別され易いことがある. しかし、識別後にどのような行動に変化するかも重要な調査項目である. 以下は、特に破壊的な行動を行う場合の手口を説明する.

- 全部のプロセスの終了:
  killall -9 *
- 全部のファイルを削除:
  rm /*

4.2.3　侵入者の識別

ハニーポットにおける重要な課題として、侵入者の行動分析（プロファイリング）がある. 特に、今回の侵入は全て ssh の脆弱なパスワードを狙った手口であるが、いったい侵入者は何人存在し、どのような繋がりを持っているかなどは未知である. ここでは、プロファイリングの試みとして、得られたキースロークから、侵入者の特徴抽出を試みる.

パスワード

パスワードは通常個人個人で異なるため、侵入者の識別に利用できる. ただし、同じ侵入者が異なるハニーポットにおいて同じパスワードを利用するのか、毎回変更するかは未知である. 今回得られた記録からは侵入者は同じパスワードを流用している模様である.

特殊文字の利用

文献 [2] において、クリフォードは侵入者のオプションにより侵入者の計算機の環境などの特徴を見分けている. 侵入者の識別に利用可能な例を以下に挙げる.

- コマンドのオプション
  exit -0 とオプションをつける.
- ワイルドカード*の使い方
  /etc/redhat-release を見るために、cat /etc/*rel* とワイルドカードを用いる.

これらの特殊文字の利用方法の特徴は、侵入者ごとに異なり侵入者のプロファイリング作成に役立つ重要な指紋といえる. これらの情報をデータベースとして記録することで、将来不正侵入を行う者を識別するための重要な情報になると考えられる.

5. 考　　察

ここでは改良システムの特徴と課題について述べる. 始めに運用経験から従来の中対話型ハニーポットとの比較をし、改良システムの特徴や課題について考察する.

5.1　従来の中対話ハニーポットとの比較

表 3 に改良システムと chroot や jail などで構築し

表 3　中対話型ハニーポットの機能比較

目的・機能	従来型	改良型
プロンプト	シェルが提供	ラッパーが提供 ダミープロンプト
対応 OS	UNIX 系 chroot に依存	UNIX 系 (Windows も可)
コマンドの 返答制限	制限なし	制限あり ダミーメッセージ
識別性	低	中
侵入方法	未知・既知の攻撃	既知の攻撃
設置の容易さ	やや困難 (システムの変更)	容易 (ラッパーの設置のみ)

た従来システムとの機能比較を示す.

5.1.1　データキャプチャ

既知の脆弱性

改良システムは既知の脆弱性を利用した侵入者を対象とした中対話型ハニーポットである. 侵入方法はアカウントの不正利用や既知の攻撃方法に限られる. これは大多数の攻撃は既知の脆弱性を狙って行われることから、未知の脆弱性を狙う攻撃は改良システムは対象としない.

得られる情報

得られる情報は実際のツールの実行やネットワークに接続できないなど制限があるものの、侵入者の動機、ツールの取得、スキルなどを判定するには十分であると考えられる. 改良システムではダミーメッセージの返答などで制限コマンドを補完しており、従来の中対話型ハニーポットの問題点である識別容易性は緩和されると考えている.

5.1.2　データコントロール

識　別　性

改良システムはダミーメッセージにより、従来型より侵入者の識別は困難であると考えられる. 実際運用を通じて、侵入後ツールの実行までは気付かれることなく偽装可能であることを確認している.

設置の容易性

改良システムは基本的にはラッパーを呼び出すエントリーポイントの設定が必要なだけであり設置は容易である. 従来方式とは異なり、システムそのものを変更する必要はない. コマンドラッパー方式の利点として、OS に依存しないことである. 従来型は chroot などの UNIX 固有の機能に依存するため他の OS では実現が難しいが、改良方法は他の OS の偽装も可能である.

5.2　課　　題

5.2.1　対 話 制 限

高対話型ハニーポットを使用することで得られるが、提案システムでは得られない情報としては、ダウンロードツール（root kit, backdoor など）の実行内容、チャットの内容などである. 例えばダウンロードツールは実行させないため、ありのままの姿の記録などはできず得られる情報に限りがある.

5.2.2　識　別　性

改良システムは特殊な OS を模倣しているため侵入者が不信感を抱く可能性が高い. しかし、今後組み込み OS などによる情報家電などの普及により、従来のパソコン以外にもネットワークに接続する特殊機器は増加することが予想されるため、識別性容易性は緩和されると思われる.

5.2.3　拡 張 機 能

実装したコマンドラッパーはシェルとしてはまだ不完全であり、パイプやシグナルなどの機能は対応していない. 今後は通常のシェルに求められている機能を補う必要がある.

6. おわりに

本論文では中対話型ハニーポットの改良方法について述べた. 改良システムはラッパー方式によりダミープロンプトを侵入者に提供し、コマンド制限やダミーメッセージによる返答など情報制御を行い侵入者を欺く. 本論文では提案に基づくプロトタイプシステムの実装および実環境での運用について述べた. 改良システムは ls と wget というわずか 2 つのコマンドのみが許可されているが、ダウンロードツールの取得など攻撃者の手口を解析するための基礎情報を取得可能であることを確認した. また運用知見として侵入者の活動内容（攻撃の手口など）の説明や侵入者のプロファイリングの試みについて記述した.

謝　　辞

本研究を行うにあたり、東京海上研究所石井威望氏にアドバイスを頂いたことに感謝致します.

参　考　文　献

1) 澁谷芳洋, 小池英樹, 高田哲司, 安村通晃, 石井威望、”高対話型ハニーポットの運用経験に関する考察”, 情報処理学会論文誌 2004 Vol.45 No.8 pp.1921-pp.1930.
2) クリフォードストール、カッコーはコンピュータに卵を産む（上下）, 草思社.
3) Honeynet Project, http://www.hoenynet.org.
4) Honeynet Whitepapers (Japanese Translation), http://www.vogue.is.uec.ac.jp/secteam/honeynetpapers/.
5) Lance Spizner, ハニーポット, 慶應大学出版会.
6) Single honeypot, http://sourceforge.net/projects/single-honeypot/.
7) The Honeynet Project, Know Your Enemy, Addison Wesley.
8) Snort, Open source intrusion detection system, http://www.snort.org.