仮想ホストを用いた攻撃パターンの収集と分析

2000MT002 安藤 純一　　2000MT036 壁屋 喜規
指導教員　後藤 邦夫

1　はじめに

現在、インターネットが普及しており、企業だけでなく、個人でも常時接続が当然のこととなっている。それと同時に、知識の有無に関わらず、クラッキングツールを使用することで、容易に不正アクセスを行うことができる。また、最近ではワームが自動的に他のホストを無差別に攻撃し、繁殖していくこともある。以上のことから、不正アクセスに対するセキュリティが重要視される。セキュリティを向上させる方法として最近少しづつ注目されつつある HoneyPot について研究し、その可能性を調べる。

HoneyPot とは、セキュリティの不十分な状態でホストを設置し、攻撃者から攻撃を受け、その行動をログとして記録するものである。名が表すとおり、はちみつ (Honey) が入ったつぼ (Pot) という意味を持っており、悪意のある攻撃者をおびきよせる「甘いわな」となるものである。

現在では、セキュリティホール (脆弱性) が発見されてすぐにその脆弱性をついたクラッキングツールが出回ったり、セキュリティホール発見の発表よりも先にクラッキングツールが出回ってしまうこともある。つまり、将来的にクラッキングの高速化に対策が追いつかなくなる可能性もあるということである。そこで、われわれは、HoneyPot を用いることにより、素早く攻撃方法を解析することが必要だと考える。

本研究では、基本的な IP や ICMP により、OS を模倣できる。honeyd というオープンソースのハニーポットを使用して仮想ホストを構築し、そこに偽サービスを組込み、脆弱性を模倣する。この仮想ホストをネットワークに接続して、不正アクセスを行う者に攻撃をさせる。初回の攻撃を成功させたように見せかけて、次の攻撃をさせる。その攻撃パターンをデータとして記録・収集し、攻撃パターンを分析することを目標とする。分析したデータにより攻撃者がどのような攻撃をしてくるかということを予測することができ、それに対する対策を立てることができると考えられる。

安藤純一は、ログ解析を担当し、壁屋喜規は、アプリケーションサービスの模倣を担当した。

2　実験システムの構成

この節では、HoneyPot がどのようなものかを説明し、HoneyPot の一種である honeyd の利点やしくみについて説明する。また、honeyd に追加する偽のアプリケーションサービスのしくみを説明し、どのようなアプリケーションサービスがあるのかを述べる。

2.1　HoneyPot について

HoneyPot には、2 つの種類がある。

一つは、本物の OS を使用し、本物のサービスとやりとりする機会を攻撃者に与える。このことにより、攻撃者の行動を詳しく記録することができる。この種の HoneyPot では、本物の OS を使用するので、攻撃の踏み台にならないようにセキュリティ対策をしっかりしなくてはいけない。

もう一つは、仮想 OS と IP アドレスを用意して、そこに攻撃者が、アクセスしてきた場合、その行動を監視するものである。この種の HoneyPot では、限られたサービスのみを許可する。また、本物の OS を使用しないので、セキュリティリスクを最小限にすることができる。

2.2　honeyd について

本研究では、HoneyPot の機能を持っている honeyd というデーモンを利用して、不正アクセスの手法の収集を行う [1]。

HoneyPot には、ManTrap や SPECTER といったものがあるが、以下の利点を満たすのは honeyd しかないので、本研究では、honeyd を使用する。

1. オープンソースである。
2. 仮想 OS を用いるので、セキュリティ面においてリスクが小さい。
3. 提供するサービスの定義が容易である。

2.3　honeyd のしくみ

honeyd は、仮想ホストを作成するデーモンプロセスであり、複数の仮想ホストを作成することができる。honeyd によって作成される仮想ホストの OS は、Linux や Windows など何にでも設定できる。この OS は実在する OS ではなく、仮想の OS である。また、Honeyd は、それぞれの仮想 OS に IP アドレスを持たせることができる。honeyd のしくみを**図 1** に示す。

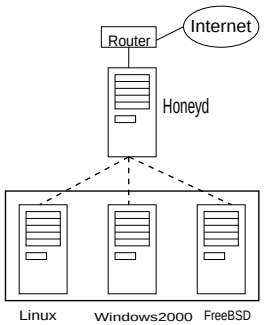

図 1: honeyd のしくみ

本研究では、honeyd によって作成される仮想ホストを

三つにし、それぞれ OS は Windows2000Server(IIS5.0)、WindowsNT4.0(IIS4.0)、Linux に設定した。Windows 系を模倣する理由としては、Windows 系に多くのセキュリティホール (脆弱性) が指摘されており、攻撃者からの攻撃を受けやすいと考えたこととセキュリティホール (脆弱性) が数多く発見されている IIS を模倣できるためである。Linux を模倣する理由は、Windows 系 (本研究では Windows2000Server、WindowsNT4.0) と比較するためである。攻撃される回数や、攻撃パターンがどのように違うのかを調べるためである。

2.4 honeyd の設置について

honeyd を設置するにあたり、honeyd は、PC 一台で複数の仮想 OS を作成できるので、PC は、一台用意する。パケットキャプチャ用ホスト (モニタリング用のホスト) では、tcpdump を用いて、通信のやりとりを全て記録する。このホストは、監視をしていることを外部に知られないようにするために、IP アドレスをつけず、外部から見えないようにする。

2.5 偽アプリケーションのしくみ

すべてのパケットは、honeyd のパケット配送機 (Packet Dispatcher) によって、それぞれの偽サービスに渡される。

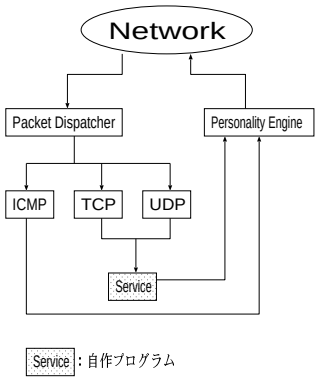

図 2: アプリケーションのしくみ

ICMP での通信には、honeyd が直接応答するが、TCP、UDP の通信がきた場合、偽のサービスを起動させる。これにより、色々なサービスを模倣することができる。そして、全ての返事は Personality Engine(OS を模倣する機構) を通る。

2.6 Web サービスへの攻撃の対応

偽アプリケーションを模倣する例として、ここで Web サービスの模倣を説明する [2]。研究室のサーバへの不正アクセスを調べたところ、IIS に対しての攻撃で C のディレクトリを表示させる要求が非常に多かった。研究室のサーバは Apache なので、この要求は拒否している。したがって、攻撃者はそれ以上攻撃をしてこない。このことにより、現在のところ研究室のサーバは安全であるが、攻撃者が次にどのような攻撃を仕掛けてくるのかが分からない。攻撃者が不正アクセスをしてきて何をしようとしているのかを理解していないと対策の立てようがない。

本研究では、Web サーバへの攻撃において、あえて攻撃を受け、攻撃者が望んでいるデータを返すようにしている。これは、攻撃者が望むデータを手に入れた後、次に何をしてくるのかをログにとり、データとして収集するためである。これにより、攻撃者がしようとしていることを推測することができ、それに対するセキュリティ対策をとることも可能になる。

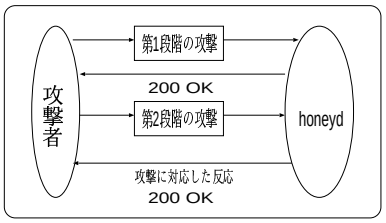

図 3: Web サーバの対応

また、攻撃者にどこまで攻撃をさせるかが問題となる。本研究では、**図 3** で示すように、少なくとも第 2 段階まで攻撃させる。攻撃者が初回の攻撃をしてきたとき、応答コード「200 OK」を返す。200 は、通信が成功したことを意味する。これにより、攻撃者が次に行う攻撃を調べる。

2.7 Web サービスへの攻撃の具体例

Web サービスへの攻撃の具体例として、NIMDA の攻撃の例を以下に挙げる [3]。

1. honeyd(WindowsNT4.0) の Web サービスに対して以下のような不正な要求 (コマンド) が送りつけられてくる。
   ●第一段階の攻撃
   GET /scripts/root.exe?/c+dir HTTP/1.0
   この不正な要求 (コマンド) は、C のディレクトリを表示させる命令である。
2. これに対して、honeyd(WindowsNT4.0) は、「200 OK」という応答コードと C のディレクトリを表示させる。
3. 次に攻撃者は、以下のような不正な要求 (コマンド) を送りつけてくる。
   ●第二段階の攻撃
   GET /scripts/root.exe?/c+tftp%20-i%20133.96.88.57%20GET %20cool.dll%20httpodbc.dll HTTP/1.0
   この不正な要求 (コマンド) は、TFTP を使用して Web サーバに httpodbc.dll として、NIMDA 自身をコピーしている。
4. これに対して honeyd(WindowsNT4.0) は、「200 OK」の応答コードを返す。

3　アクセスログの分析 (結果)

2003 年 11 月 19 日午後 6 時に、学内の実験用のネットワークに honeyd を設置した。honeyd によって作成された仮想ホスト (WindowsNT4.0、Windows2000Server、Linux) にそれぞれ IP アドレスを持たせた。WindowsNT4.0 は 133.29.8.106、Linux は 133.29.8.107、Windows2000Server は 133.29.8.108 である。それぞれの仮想ホストで起動させるサービスは、3 アプリケーションサービスの模倣で述べた Web サービス、SMTP、telnet のログイン・プログラムである。これらの仮想ホスト、アプリケーションサービスを用いて実験を行った。2003 年 11 月 19 日午後 6 時以後のハニーポットへのアクセスログを分析する。

3.1　日別アクセス数

実験で得たアクセスログを日ごとに区別して、日ごとの比較をした。日ごとのアクセス数をグラフにして**図 4** に示す。

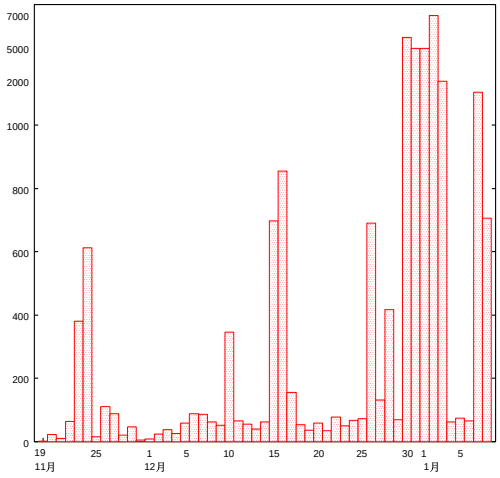

図 4: 日別アクセス数

HoneyPot は、DNS に登録しておらず、世間にはまったく知られていないホストである。われわれは、実験をするなかで、クラッキングをしてくる攻撃者がどのようにして HoneyPot の存在を知り、どれほどの期間で HoneyPot の存在を知るかを調べようと考えた。実験を開始して、HoneyPot をネットワークに接続してから、徐々に HoneyPot へのアクセス数は増加している。しかし、HoneyPot に対してコマンドによる攻撃やポートスキャンなどのアクセスはあったが、これが「人間」によるものであるのか「ワーム」などの機械的なツールによるものなのか判断することは難しい。例えば、12 月 10 日は、一日で 700 回ちかくのアクセスがある。700 回のうちの 9 割は一つのホストからのアクセスである。一度にこれだけ多くのアクセスを行ってくるが、これを「人間」がツールを使用してアクセスをしてくるのか、「ワーム」が一度に何度もアクセスしてくるのか区別することは難しい。

しかし、「ワーム」による攻撃で一度に 700 回ほどのアクセスをしてくることは考えにくい。そこで、攻撃者が「人間」であると仮定するならば、攻撃者は、HoneyPot の存在を知り、集中的に HoneyPot へ攻撃をしてきたのではないかと推測される。12 月 10 日の攻撃は、80 番ポートへポートスキャンをした後に Windows 系の仮想ホストに対して、1433 番ポートに攻撃を何度も行っている。つまり、攻撃者は、HoneyPot に対してポートスキャンを行い、攻撃対象のポートが開いているかを確認して、攻撃を行ったと推測される。

以上のことから、もし 12 月 10 日に攻撃してきた相手が「人間」であるならば、HoneyPot は、ネットワークに接続して 3 週間ほどで存在を知られたことになる。

3.2　国別アクセス数

実験で得たアクセスログから通信先の IP アドレスを抜き出し、その IP アドレスがどこの国のものであるかを調べた。honeyd の仮想ホストに対してアクセスが多かった国は、US(アメリカ)、NL(オランダ)、AU(オーストラリア) である。これらの国の他にも仮想ホストにアクセスをしてきた国はあったが、この三ヶ国に比べるとアクセス数は少なかった。国別アクセス数をグラフ化したものを**図 5** に示す。

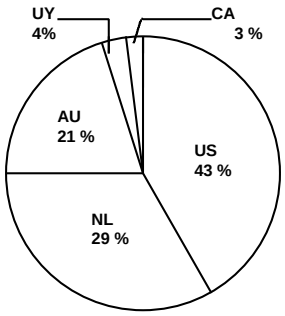

図 5: 国別アクセス数

3.3　時間帯別アクセス数

実験で得たアクセスログから時間帯別にアクセス数をカウントしたものをグラフ化した。

グラフを見て分かったことを以下に示す。

1. 攻撃が多いときと少ないときの差はあるが、深夜から昼にかけて、アクセスが集中している。
2. 昼から夜にかけてアクセス数が少ない。とくに、15 時と 20 時がアクセス数が少ない。
3. 深夜 2 時が、最もアクセス数が多く、10000 回ちかくのアクセス数がある。2 時は、複数のホストから攻撃を受けているが、一度で 2700 回もアクセスしてきたホストもあった。この他のホストも一つのホストあたり 800 回前後のアクセスをしてきた。

時間帯別アクセス数をグラフ化したものを**図 6** に示す。

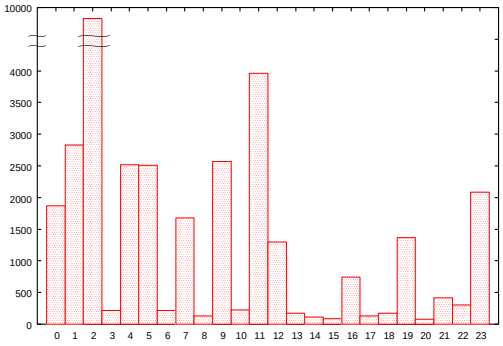

図 6: 時間帯別アクセス数

3.4 HoneyPot のポートに対するアクセス数

HoneyPot のポートに対するアクセス数を HoneyPot 設置から一ヶ月の結果と HoneyPot 設置から 50 日 (1 月 8 日まで) たった結果をポート別にして比較する。

1. 1433 番ポート：
   HoneyPot 設置から一ヶ月の時点では、HoneyPot への全アクセス数の 41%を占めていたが、50 日たった結果を見ると全アクセス数の 91%も占めている。1433 番ポートの占める割合が増加した理由としては、年末年始に非常に多くきたためである。1433 番ポートは、MSSQL で使用しているポートであり、1433 番ポートに対するアクセスは、MSSQL に対する攻撃だと考えられる。
2. 10 番ポート：
   HoneyPot 設置から一ヶ月の時点では、HoneyPot への全アクセス数の 30%を占めていたが、50 日たった結果を見ると全アクセス数の 3%しか占めていない。この理由としては、一ヶ月たったときから 50 日たつまでの間に 10 番ポートに対するアクセスはほとんど行われなかったためである。10 番ポートは、未使用のポートであり、アプリケーションサービスに対する攻撃ではなく、バックドアを作成するためにアクセスしてきていると考えられる。
3. 80 番ポート：
   HoneyPot 設置から一ヶ月の時点では、HoneyPot への全アクセス数の 12%を占めていたが、50 日たった結果を見ると全アクセス数の 2%しか占めていない。この理由としては、10 番ポートと同様で一ヶ月たったときから 50 日たつまでの間に 80 番ポートに対するアクセスはほとんど行われなかったためである。

HoneyPot のポートに対するアクセス数をグラフ化したものを**図 7** に示す。

4 おわりに

今回の実験では、80 番ポートや 25 番ポートに多くのアクセスを期待し、それに対する脆弱性も模倣した。しかし、実際には、予想された攻撃はほとんどなく、アクセスも決して多くはなかった。また、2003 年に猛威をふるった Windows RPC の脆弱性をついたポート 135 番ポートへの攻撃、netbios の脆弱性をついた攻撃、telnet への接続は大学側で遮断されているため観測できなかった。しかし、これにもかかわらず攻撃はありとあらゆるポートに対して行なわれた。人間が実際に攻撃を行ない、侵入を試みた形跡もなかったが、どのポートへの攻撃が存在するか、どのようにアクセスをしてくるのかがわかった。さらに、年末年始には非常に多くのアクセスを受けることが分かった。

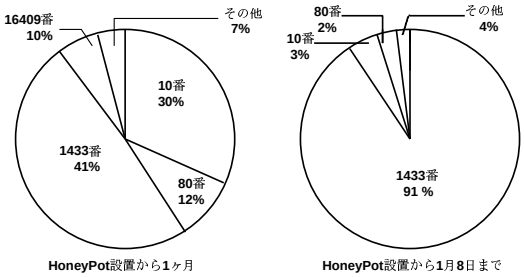

図 7: HoneyPot のポートへのアクセス数

設置時間が約一ヶ月と短く、攻撃ログの収集と分析だけにとどまってしまい、新しい攻撃を発見することはできなかった。しかし、HoneyPot を設置するだけで短時間でも安全に多くの攻撃を受けることが可能であることがわかった。アクセス全てを攻撃と見なすことができるため、ログ解析の時間の短縮や、いつ、どのポートに、どこのホストからの攻撃が行なわれたかすぐに確認できる。

今後の課題として、HoneyPot が新しい攻撃の発見やその対応がすぐにできるようになるとよいと考えられる。例えば、過去のアクセスを DB に登録し、HoneyPot に寄せあられた新たな攻撃と比較できれば、より新しい攻撃の発見や HoneyPot の拡張が簡単になると考えられる。

最後に、HoneyPot は攻撃を収集できる有用な手段のひとつであり、十分研究するに値する題材であると思う。そして、今後のセキュリティ対策の重要な分野のひとつであると考えられる。

参考文献

[1] Niels Provos：
Honeyd-Network Rhapsody foy you,
http://www.citi.umich.edu/
u/provos/honeyd/.
[2] Nessus, http://www.nessus.org/.
[3] CERT, http://www.cert.org/.