Honeypot:Argos,Nepenthsの性能評価

奥野 将嘉†
早稲田大学大学院 理工学研究科 情報ネットワーク専攻†

1 はじめに

近年，様々なハニーポットがあるが，ハニーポットの性能を比較している論文は見当たらない．各ハニーポットの長所・短所を把握することにより，新規ハニーポットを構築しやすくなり，また，性能比較項目を示すことにより，新規にハニーポットを構築した際にも，既存ハニーポットとの評価項目となり得る．本発表では，現在世の中で広く利用されているArogs[1],Nepenthes[2]という2つのハニーポットに着目し，情報収集能力・検知能力・偽装能力・運用性能という4つの比較項目を設け，性能比較を行った結果について述べる．

2 Argos

Argosは，0-day Attackを検知することを目的として作られた次世代ハニーポットであり，仮想マシンである（Qemu）上で動作し，実際のOSをハニーポットとしてエミュレートするHigh Interaction型ハニーポットである．

Argosにおける攻撃検出は，ネットワークから来たデータをすべて汚染データとし，その汚染データが書き込まれたメモリやレジスタも汚染扱いとし，汚染データが制御フローを変更させると警告を出す，という動的テイント解析が用いられている．これには，次に実行する命令の位置を記憶しているレジスタであるEIPレジスタに着目し，ネットワークから送られてきたデータをプログラムが実行しないかをチェックするという方法がとられている．

3 Nepenthes

Nepenthesは，マルウェア収集のためのLow Interaction型ハニーポットである．既知の脆弱性とShellcodeの1部をエミュレートすることによりマルウェアを収集する．

Nepenthesはモジュール構造となっていて，アーキテクチャの概要としては，攻撃検知フェーズ，マルウェア収集フェーズ，マルウェア解析フェーズとなっている．攻撃検知フェーズでは既知の脆弱性をエミュレートし，攻撃者からの攻撃パケットに応答する．攻撃者からのShellcodeをエミュレートしている既知のShellcodeとパターンマッチングし，攻撃かどうか判断している．

4 性能比較

性能比較項目を**4.1**～**4.4**で提案するとともにその性能比較項目に基づき2つのハニーポットの性能比較を行った．

4.1 情報収集能力

情報収集能力とは，そのハニーポットがどれくらい攻撃者の情報を収集できるか，ということである．

4.1.1 対応脆弱数

本実験において，ArgosはWindows2000をエミュレートしているため，対応脆弱性数としてはWindows2000の既知の脆弱性数10とした． Nepenthesは21種類の脆弱性をエミュレートしている．しかし，元々サーバのサービスが使っているポートをbindできなくなるという問題がある．

4.1.2 攻撃コード収集

Argosは実装上の問題により攻撃コードのデータは理論的に汚されている可能性がある．これはArgosが攻撃を検知する前に，攻撃コードを書き換えられる可能性があるからだ．一方Nepenthesはこの日202回攻撃コードを収集した．

4.1.3 マルウェア収集効率

Argosはマルウェアを収集しないため，以下でNepenthesのみを評価する．

NepenthesはShellcodeModulesでマルウェアのダウンロード先URLを380回（186種類）取得し，DownloadModulesでマルウェアを実際に258回取得した．このうち同一のマルウェアは44種類であった．

4.2 検知能力

検知能力とは攻撃者からの攻撃を攻撃とみなすことができるのか，ということである．

Performance comparisons of honeypots: Argos and Nepenthes.
†Masayoshi OKUNO
†Graduate School of Science and Engineering,Waseda University

4.2.1 既知の攻撃・未知の攻撃

Argos はこの日 61 回，Nepenthes は 380 回攻撃を検知した．攻撃を検知した回数は Nepenthes の方が遥かに多いと言える．次に Argos の EIP レジスタに使われた ReturnAddress を検証した結果，Argos が検知した 61 回の攻撃のうち正確に検知した回数は 60 回，誤検知が 1 回あった．また，False Negative が何回あったか，というのは不明だが，False Positive は 0 回だった．

4.2.2 未知の脆弱性

Argos は理論上未知の脆弱性をつく攻撃も検知できるが，本実験では未知の脆弱性をつく攻撃がなかったため，実際は分かりかねる．一方 Nepenthes は既知の脆弱性をエミュレートしているため，未知の脆弱性には対応していない．

4.3 偽装能力

偽装能力とはいかにハニーポットだとバレないか，ということである．Argos は OS をそのままエミュレートしているので調査対象ではないと言えるため，脆弱性をエミュレートしている Nepenthes を中心に調べた．

4.3.1 脆弱性の再現

MS04-011 の脆弱性を利用し，Resposnse Value, Response Time で検証した．
特定のプロトコルでリクエストを送りレスポンスを見ると，Nepenthes は正しいレスポンスを返していない．これはソースコードにおいて適当なランダム値でレスポンスパケットを生成しているからである．したがって，レスポンスの値をちゃんとチェックすると接続先が Nepenthes であることがわかる．また，ターゲットを WindowsXPSP2 と Nepenthes として，Response Time を計ってみると，Nepenthes は WinXPSP2 よりもレスポンスタイムが 10 倍～80 倍大きいことがわかった．Response Time から見分けるのは難しいが，特定のポートだけ Response Time が異常に遅いということに着目すると見分けることができる可能性がある．

4.3.2 サイドエフェクト

サイドエフェクトとは，修正パッチを適用することで発生する，プログラムの挙動の特徴のことである．Linux 上で Nepenthes を動作させ，あらゆる脆弱性をエミュレートさせておき，Eeye セキュリティスキャナ Retina を利用して Nepenthes をスキャンしサイドエフェクトの振る舞いを観測した．結果として，Retina は HostOS である Lunux と判定し，Lunux の脆弱性を検出した．HostOS と各サービスの振る舞いの矛盾からハニーポットだとばれる可能性があると言える．

4.4 運用性能

運用能力とはハニーポットを実際に運用していく上での使いやすさはどうか，安全性は保たれているか，ということである．

4.4.1 安全性・パフォーマンス

Argos は実装上の問題から実行中にフリーズしてしまうことがあるため，安全性やパフォーマンスに問題がある．Nepenthes も本実験中にセグメンテーション違反で 1 度だけ止まったがパフォーマンスは Argos に比べてよかったと言える．しかし，Nepenthes は未知の脆弱性には対応していないため，未知の脆弱性をつく攻撃を受けたときに乗っ取られてしまう危険性はある．

5 考察

高対話型ハニーポットである Argos は理論上，未知・既知の攻撃，未知の脆弱性をつく攻撃にも耐えられ，ハニーポットともばれにくい．しかし，Nepenthes と比べるとパフォーマンスが悪く，欠点も多々あった．
一方，低対話型ハニーポットである Nepenthes は情報収集能力，検知能力に長けていたが，偽装性能はないに等しいと言える．
今後として，さらに多くの既存ハニーポットについて性能比較を行い，それぞれのハニーポットの特徴について把握していきたい．また，そこから得られたデータを新規ハニーポットの構築に役立て，新規ハニーポットを構築した際，既存ハニーポットとの比較評価をする上で役立てていきたい．

参考文献

[1] Argos: an Emulator for Fingerprinting Zero-Day Attacks
Georgios Portokalidis Asia Slowinska Herbert Bos
EuroSys2006 Leuven,Belgium -April 18-21, 2006

[2] nepenthes -first collection - http://nepenthes. mwcollect. org/