機械学習によるネットワーク型IDSのfalse positive削減手法の提案

宮地 玲奈[1], 小宅 宏明[1], 川口 信隆[1], 重野 寛[1], 岡田 謙一[1]

概 要

近年，セキュリティ侵害の増加に伴い，常にネットワークを通過するパケットを監視できるネットワークIDSへの関心が高まってきている。しかしネットワークIDSは誤検知，特に実際には攻撃でない事象を誤って攻撃と認識するfalse positiveが多発することが知られている. 本稿では機械学習によってfalse positiveのパターンを学習することでIDSのログに含まれるfalse positiveを検出する手法を提案した。

A Proposal for Technique to Reduce False Positive of Network IDS with Machine Learning

Reina Miyaji [1], Hiroaki Ohya[1], Nobutaka Kawaguchi[1], Hiroshi Shigeno[1], Kenichi Okada[1]

Recently,network-based IDS,which always observes the packets flowing in the networks, has become the focus of the public attention with increasing security incident.However,network-based IDS frequently mistakes attacks.Especially, IDS generates many false positives,that are bogus alerts caused by mistakes normal events with attacks.In this paper,we proposed a technique to detect false positive with machine learning.

1 はじめに

インターネットの普及の弊害として, コネクティビティの提供されている計算機へのセキュリティ侵害が急増している. IDSは外部のネットワークからのセキュリティ侵害の防御として組織のネットワーク内からのセキュリティ侵害への対応や, ネットワークのセキュリティを監視するための有効な手段として注目を集めている. 一方でIDSは誤検知の高さが問題となっている. 特に攻撃でない事象を誤って攻撃と判断するfalse positiveが多発する事が知られており, 管理者は実際の攻撃なのか否かを判断しなければならず, これは大きな負担となっている. 本稿では現在最も一般的に利用されているsignatureマッチングを用いたネットワーク型IDSにおいて, 定常的に発生するfalse positiveのパターンを機械学習することによって, 警告 (alert) の中からfalse positiveと思われるものを検出する手法を提案する.

[1] 慶應義塾大学理工学研究科
Faculty of Science and Technology,Keio University

2 IDSによる誤検知

IDSは攻撃の可能性があると判断したとき，たとえ本当に攻撃があった場合ではなくても警報を発することがある．この様に実際には攻撃ではない行為を不正として検出する誤検出のことをfalse positiveという．これに対し，実際の攻撃を攻撃として検出しない誤検出のことをfalse negativeという．IDSの感度を増したり (例えば，検出すべきパターンを記述したsignatureの種類をセキュリティの重要度の低いものまで増やすことなど)，抽象的なsignature を用いたりすることにより，false positiveの割合は増す．false negativeの方がfalse positiveよりも被害が大きいため，一般にIDSの設定はfalse positiveを優先して行なわれる傾向にある．

false positiveはネットワーク構成や監視対象にあわせたIDSの初期設定の問題, パフォーマンスとのトレードオフによるsignatureの表記の非柔軟性によるカスタマイズの限界, 不完全なsignature等が原因で発生する.

false positive が発生することについては,false positive に紛れて本当の攻撃 (true positive) を見逃す恐れがある, 監視すべきログの情報量が増え，管理者の負担が増す, といった点で問題がある. また,false positive でなくとも，重要度の低い alert(ICMP destination unreach など) が多発することで同様の問題が生じる．

3 提案

false positive は 2 で述べたような事が原因となって発生する. これらのうち，IDS の初期設定の問題とカスタマイズの限界に関しては，パケットの評価が不充分なことが false positive 発生の原因である．IDS が発する alert の中から全ての false positive を特定することは不可能である．しかし，パケットの評価か不充分なために，同一の原因による似たような false positive が多数発生する場合に限り，false positive を特定することが可能である．本提案では，false positive の原因となったパケットを収集し，特徴を抽出する．次に上記の特徴を元に，IDS が発する alert の中から類似したものを選び出し，管理者に対して false positive の疑いを指摘する．提案する手法は次の 2 つのフェーズから構成される．

構築フェーズ false positive のパターンを発見し，システムに学習させる

1. false positive を収集
2. false positive をグループ化
3. グループ毎にパターンを抽出
4. ニューラルネットワークによりパターンを学習

運用フェーズ 構築したシステムを用いて IDS のログから false positive を検出する

1. 学習したニューラルネットワークを用いてパターンを識別
2. false positive を検出

3.1 構築フェーズ

構築フェーズでは，IDS のログから false positive のパターンを発見し，ニューラルネットワークに学習させるフェーズである．以下，構築フェーズについて詳細に説明する．

3.1.1 false positive の収集

このフェーズでは，false positive と思われる alert の収集を行なう．alert はパケットの送信元 IP アドレス, ポート番号パケットの送信先 IP アドレス, ポート番号,IDS が判断した攻撃のタイプ, パケットのペイロードの情報が含まれている必要がある. また, サンプルとなる false positive の収集には，以下の 3 通りの方法が考えられる．

- 攻撃がない状態で IDS を運用し，false positive のデータを収集する
- 攻撃が起こり得る状態で IDS を運用し，alert の中から false positive と思われるものを管理者が抽出する
- 上記の 2 つを併用する

3.1.2 false positive をグループ化

パターン抽出の前にクラスター分析を行なう理由は，同一の (もしくは類似した) 原因によって発生したと思われる false positive のみを選び出すためである．2 つの alert に同じ signature が付けられているからと言って，それらが同じ原因によって発生したものであるとは言い難い．

そこで，収集した false positive に対して signature ごとにクラスター分析を行ない，ペイロードの類似したものをグループ化する．この操作は，後のフェーズで学習を行なう際にノイズとなる入力を除去することを目的とする．

クラスター分析を行なうにあたって，alert を幾つのクラスタにまとめるかあらかじめ決まっていないので，階層的クラスター分析を行なう．まず，signature ごとにグループ化された alert に対して，似通ったもの同士をさらにグループ化して 1 のような樹状図 (デンドログラム) を作成する．

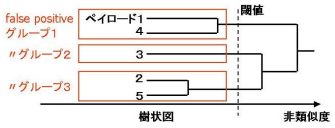

図 1: 樹状図

距離 (非類似度) の判定基準として，パケットペイロードの最小編集距離を用いる．次に，類似度に関してある閾値を決定し，その閾値未満の距離にある要素をクラスターとしてまとめる．最終的に，要素数の少ないクラスターは除外する．

最小編集距離とは2つの文字列がある場合に，一方の文字列に編集を加えて他方の文字列に一致させるために必要な編集操作の回数の最小値を表す．例えば，”AAAABB”(文字列 1) を編集して”AACBB”(文字列 2) にするためには (1) 左端の A を削除する,(2) 左端の A を削除する,(3)3 文字目に C を挿入するという 3 回の操作を行なえばよいので，この場合の最小編集距離は 3 になる．

3.1.3 パターン抽出

false positive のグループ毎に共通して出現するパターンを抽出する．

まず，false positive グループの中から，2 つのペイロードを選ぶ．その 2 つの間で固定長の部分文字列を比較しながら，共通して現われるパターンを探す．共通のパターンを発見したら，パターンごとに出現頻度も数える．この作業を部分文字列の長さを変えながら全てのペイロードについて行なう．

次に，グループ全体で共通して出現するパターンを頻度順に並べて，表を作成する．

3.1.4 パターンの学習

false positive グループ 1 つに対して 1 つのニューラルネットワークを用意し，抽出したパターンを用いてニューラルネットワークに教師あり学習を行なう (図 2 参照)．ニューラルネットワークは，入力層，中間層，出力層の 3 層から構成され，各層のノード数は，入力層 24，中間層 8，出力層 1 である．学習アルゴリズムとしてはバックプロパゲーション (逆誤差伝搬法) を用いる．

ニューラルネットワークの入力信号は，ペイロードごとに以下の値を求め，入力として用いる．

- 抽出したパターンのうち，出現頻度の高い 20 個の出現頻度
- 送信側の IP アドレスとポート番号
- 受信側の IP アドレスとポート番号

ニューラルネットワークの出力信号は，この alert がこのニューラルネットワークが対応する false positive グループに似ているかどうかを表す値で，0 以上 1 以下の実数とする．

教師信号は，実際に入力した alert が false positive グループに属しているかどうかの値で，実際に false positive グループに属している場合には 1, そうでない場合には 0 とする．

パターン学習では，以下の作業を誤差が一定以下になるまで繰り返す．

- ニューラルネットワークに入力信号を入力し，出力信号を得る
- 教師信号と出力信号を比較し，ノードの重み付けを修正する

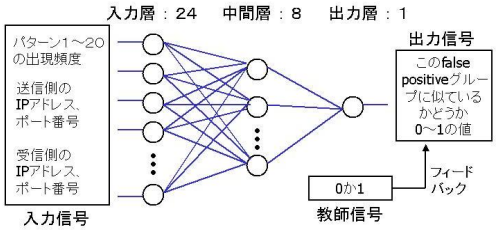

図 2: パターン抽出

3.2 運用フェーズ

運用フェーズでは，構築したシステムを用いて IDS のログから false positive を検出する．

3.2.1 パターンの識別

3.1.4 においてニューラルネットワークの学習について述べた．この段階では，学習済みのニューラルネットワークに実際のデータを入力して逆解析を行ない，入力したデータがどの false positive グループに最も似ているかを調べる．

まず，ニューラルネットワークの数は 3.1.2 で分類した false positive グループと同数だけ用意されているものとし，全て 3.1.4 に記述した操作によって学習を済ませているものとする．入出力信号の形式は 3.1.4 で用いたものと同じものを用い，教師信号は使用しない．

各ニューラルネットワークにそれぞれ入力信号を入力し，出力の値が最も 1 に近いニューラルネットワークを選択する．選択したニューラルネットワークが対応する false positive グループが，入力した alert に最も近い false positive グループであると判断する．

3.2.2 false positive の検出

この段階では，alert が実際にその false positive グループに類似しているかどうかを調べるとともに，その alert が false positive であるかどうかを判別する．

3.2.1 における処理の結果，最も似ているとされた false positive グループがある．false positive グループの中から任意に 3 つのペイロードを選択し，各々と alert のペイロードとの間で，最小編集距離を求める．

全ての最小編集距離の値が false positive グループを決定する際に用いた閾値の値よりも小さければ，この alert は false positive グループに属すると判断する．逆に，一つでも閾値を上まわる最小編集距離が得られた場合には，false positive グループに属さないと判断する．

false positive グループに屬すると判断した場合には，alert は false positive であると考え，その false positive グループの番号をログに追記する．

4 実装

提案システムを, 実装言語に C++, コンパイラに g++ 3.2,OS に Linux 2.4.20 を用いて実装し, プロトタイプを作成した．実装にあたって，IDS にはフリーの IDS である Snort を利用した．また，IDS のログはログ処理の際の利便性を考慮し，データベースに出力するようにした．DBMS には PostgreSQL を用いた．ニューラルネットワークのアルゴリズム部分には Annie[1] を用いた．アルゴリズムとしてバックプロパゲーションを利用した．

5 評価方法

プロトタイプを用いて，提案手法の評価を行なった．

5.1 評価用データ

評価用のデータとして，MIT の LINCOLN 研究所が作成した IDS 評価用のデータ [2] を使用した．このデータは同研究所が DARPA(高等研究計画局)[4] の支援によって 1998 年から作成しているもので，IDS の性能を評価するための統一的なデータとして広く利用されている．

データには以下の内容が含まれている．

- LAN の外部, 内部のトラフィック
- カーネルのシステムコール
- Windows NT のイベントログ
- 各種ログや設定ファイル
- 全ファイルのリスト
- 行なった攻撃
- 攻撃を識別するための情報
- ネットワーク構成図

5.2 評価用データの利用方法

本研究で利用したのは 1999 年に作れたもの [3] で，3 週間分のデータから構成される．1 週目と 3

週目は攻撃がなかったことが明記されている．2週目には攻撃が含まれており，トラフィックやログ中におけるどの部分が攻撃であったかが書かれている．

評価実験では，LAN の外側のトラフィックデータをを次のように使用する．

- 1 週目 false positive パターンの学習
- 2 週目 false negative 発生率の評価
- 3 週目 false positive 発生率の評価

第 2 週のデータは，本システムの検出精度の測定に利用する．本稿で提案する手法は IDS の検出精度を向上させるものではないが，実際に行なわれた攻撃に対して，本システムが誤って false positive と判断してしまう危険性がある．そのような危険性がどの程度存在するのかを測定する．

第 3 週のデータについては，本システムの目的，すなわち”false positive の検出” がどの程度達成されているかを評価するために利用する．

5.2.1 false positive パターンの学習

1 週目のトラフィックデータはニューラルネットワークの学習に利用する．トラフィックデータを IDS に読み込ませると alert が発生する．このデータには攻撃は含まれていないので，このデータを読み込んで発生した alert は，全て false positive もしくは重要度の低い alert と考えることができる．この操作によって発生した alert を入力とする．

5.2.2 false negative 発生率の評価方法

2 週目のトラフィックデータには攻撃が含まれているので false negative 発生率の評価に利用する．snort にトラフィックデータを読み込ませた場合と,snort にトラフィックデータを読み込ませてから提案システムが誤って false positive と判断した攻撃を取り除いた場合の 2 通りを実行し発生した false negative の数を比較する．評価用のデータには，5.1 に記述した通り，攻撃の種類と攻撃の時刻が含まれている．IDS にトラフィックデータを読み込ませた場合に検出された攻撃と，実際にトラフィックに含まれていた攻撃とを比較することで，IDS がどの程度実際に行なわれた攻撃を検出することができたかを調べることができる．

5.2.3 false positive 発生率の評価方法

3 週目のトラフィックデータには攻撃が含まれていないので false positive 発生率の評価に利用する．snort にトラフィックデータを読み込ませた場合と, snort にトラフィックデータを読み込ませてから提案システムによって検出された false positive を取り除いた場合の 2 通りを実行し発生した false positive の数を比較する．

false positive の数え方　3 週目のトラフィックデータには攻撃は含まれていないので，3 週目のトラフィックデータを読み込ませた結果検出された alert は，全て false positive であると考えられる．

5.3 評価条件

対象としない攻撃タイプ　TCP プロトコルに違反した奇形パケットなど，ヘッダの内容をもとに比較的容易に検出可能な攻撃については，false positive が発生しにくく，また Firewall によって除去されてしまうことが多いため，本提案の対象から外すことにする．また，syn flood のように統計的な処理によって判断される攻撃についても，個々のパケットの比較によって false positive であるかどうかを判断することが不可能なため，提案の対象から除外する．

6 評価結果

評価結果の全体を，表 1 に示す．(fn:false negative,fp:false positive)

表 1: 評価結果

	fn(第 2 週)	fp(第 3 週)
オリジナル IDS	73	24122
本提案	73	15792

6.1 学習結果

第 1 週のトラフィックデータを用いてシステムの学習を行なったところ，21657 個，36 種類の alert が発生した．発生した false positive をシステムの入力として，学習を行なった．

6.2 false negative の検出精度

第 2 週のトラフィックデータを用いて，検出精度に対する評価を行なった．

本稿で提案する手法は IDS の検出精度を向上させるものではないが，実際に行なわれた攻撃に対して，本システムが誤って false positive と判断してしまう危険性，すなわち false negative を発生させる危険性が存在する．この実験では，実際に行なわれた攻撃のうち，IDS が検出できたものに対して，本システムが誤って false positive と判断した回数を測定した．

トラフィックデータには，4592 個，168 種類の攻撃が含まれていることが攻撃に関する記録からわかっている．このうち，5.3 で述べた攻撃を除外すると，3557 個，132 種類の攻撃が評価の対象となった．

Snort[?] を用いて侵入検知を行なったところ，3557 個の攻撃のうち，3484 個の攻撃を検出し，73 個の false negative が発生した．

Snort が検出した 3484 個の alert を本システムに入力したところ，false positive として検出したものは一つもなかった．

この実験では，本システムは false negative を全く発生させなかった．

6.3 false positive の検出精度

第 3 週のトラフィックデータを用いて，本稿の目的である，”false positive の検出”がどの程度達成できたかを測定した．

トラフィックデータを読み込ませたことろ，24122 個，29 種類の false positive が発生した．IDS の出力を本システムに入力したところ，8330 個 (35%)，8 種類の false positive を検出した．

7 考察

評価実験の結果を見る限り，false negative の数を増加させることなく，false positive の一部を検出することができた. 検出精度に関しては全ての false positive のうちの 35%程度であり，学習を行なった false positive に限定すれば 52%の検出精度であった．管理者がログを閲覧する作業を考えれば，見る必要のない情報が 1/3 減少したということであり，管理者のログ監視作業を支援するという，本提案の目的は達せられていると考えられる．

8 まとめ

本研究では signature マッチングを用いたネットワーク型 IDS において alert の中から false positive と思われるものを検出し, 管理者のログ監視作業を支援するシステムを提案した. 提案システムを実装し, 評価を行った結果, 手を加えない状態の IDS と比較して false positive の発生数をおよそ 35%削減する事ができ, 提案の有効性が確認された.

参考文献

[1] Asim Shankar, ”Annie”, http://annie.sourceforge.net/, Jan 2003

[2] Lincorn Lab, MIT, http://www.ll.mit.edu/IST/ideval/, Jan 2003

[3] Lincorn Lab, MIT, ”1999 DARPA Intrusion Detection Evaluation Data Set”, http://www.ll.mit.edu/IST/ideval/data/1999/1999_data_index.html, Jan 2003

[4] ”The Defense Advanced Research Projects Agency (DARPA)”, http://www.darpa.mil/, Jan 2003

[5] tcpdump,http://www.tcpdump.org/, Jan 2003