社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006－CSEC－35 (12)
2006／12／8

定点観測による不正アクセス分析システム

榊原裕之　北澤繁樹　大野一広　藤井誠司

三菱電機株式会社　情報技術総合研究所　〒247-8501　鎌倉市大船 5-1-1

あらまし　近年ワーム，Dos 等のネットワーク経由の不正アクセスが増加しており社会的な問題となっている．本稿では IDS Alert ログ等のネットワークの定点観測データの変化を早期に検知する Anomaly 方式の不正アクセス分析システムについて論ずる．本システムでは，定点観測で得られる時系列データをスライディングウィンドウによりパターン化し，不正アクセスを受けていない正常状態におけるパターンに類似しないパターンを観測した場合に異常と判定する．パターンの比較を効率化するため，パターンに対して主成分分析により特徴量を計算し，正常状態の特徴量に対する最新のパターンの特徴量の乖離を調べ類似を判定する．

An Intrusion Detection System by fixed-point observation of network security data

Hiroyuki Sakakibara　Shigeki Kitazawa　Kazuhiro Ohno　Seiji Fujii

MITSUBISHI ELECTRIC CORPORATION, INFORMATION TECHNOLOGY R&D CENTER

Abstract Today, unlawful network accesses such as Internet Worm and DoS are increasing and have caused a social problem. As a counter measures for these threats, we propose an anomaly-based IDS analyzing and finding a change of time-series data obtained by fixed-point observation such as IDS Alert log. Time-series data is converted to multiple patterns by a fixed-size sliding window. When the latest pattern is not similar to patterns of normal status, it is determined as anomaly status. For efficient pattern comparison the proposed system converts patterns to principal component scores in low dimensions. When principal component score of the latest pattern deviates from scores of normal patterns the system judges the latest pattern is anomaly status.

1．　はじめに

近年 DoS 攻撃，ワーム等のネットワーク（以下 N/W）経由の不正アクセスが増加しており社会的な問題となっている．防御策の一つとして Network-based Intrusion Detection System (NIDS)[1]がある．N/W データを受動的に監視するため導入による N/W への影響を低く抑えられ，N/W 単位に 1 台で監視可能なため普及している方式である．検知方式は大別して，シグネチャ方式とデータの挙動が通常とは異なることを検知する Anomaly 方式に分けられる[2]．シグネチャ方式ではシグネチャが対応す

る個々の攻撃の検知には優れているが攻撃の内容が分析されないとシグネチャが用意できない欠点がある．一方で Anomaly 方式はデータの傾向の変化を分析することで通常状態と異なる状態を検知するため未知の不正アクセスへの防御が期待される[3][4]．

筆者らは，不正アクセス分析システムとして Signature-based NIDS(以下 S-NIDS) Alert ログなどのネットワークの定点観測データに対し主成分分析を適用した Anomaly 方式の NIDS を開発しており，本稿ではその概要と異常検知の方式について論ずる．

2． 不正アクセス分析システムの概要

2．1 目標

当不正アクセス分析システム（以下，当システム）は，監視対象の N/W において定点観測による N/W 監視データを分析し不正アクセスを早期に検知すること，及び検知した不正アクセスに対する対策の指標を示すことを目標とする．

2．2 機能構成

図 1 に本システムの機能構成を示す．

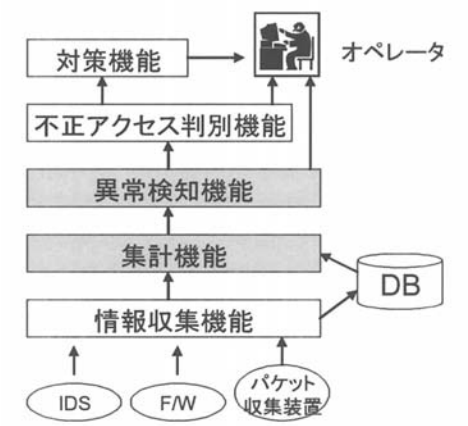

図 1 機能構成

・ 情報収集機能

監視対象の N/W における，S-NIDS のログ等

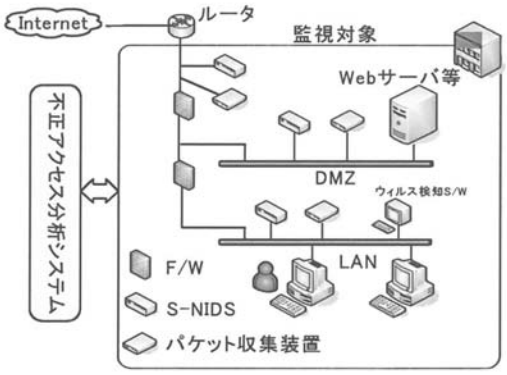

図 2 監視対象

の定点観測による N/W 監視データを取り込む．S-NIDS 等のログの情報が分析に不足している場合や機器自体が設置されていない箇所には，スニッフィング形式のパケット収集装置を設置し N/W 監視データを収集する（図 2）．これらの機器による N/W 監視データを定期的に収集し DB に格納する．

・ 集計機能

情報収集機能で集められた N/W 監視データから検知に必要なデータを集計する．

・ 異常検知機能

集計機能により集計されたデータをもとに N/W 監視データの通常と異なる状態（異常状態）を検知する．

・ 不正アクセス判別機能

異常検知機能において異常状態が検知された場合，その原因を判定する．

・ 対策機能

不正アクセス判別機能により不正アクセスが確定された場合，特定ポートへのアクセスの制限等，対策の指針を出力する．

3．検知方式

異常検知機能で採用する検知アルゴリズムとして当システムでは主成分分析を利用する．主成分分析(Principal Component Analysis 以下 PCA)は多数の変量から少数の変量を合成

し，合成した変量においてデータを評価する分析手法である[5]．PCA を N/W 監視データの分析に適用する方法として，N/W 全体の通信の傾向の変化を捉える目的で TCP ヘッダ情報に適用する方式[6][7]と，1 つの時系列データをスライディングウィンドウで分割し含まれるパターンの変化を検知することで時系列データの傾向の変化を検知する方式がある(本稿では SW-PCA と呼ぶことにする) [8].

当システムにおいては，ワームの拡散，DoS，を検知することを目的とし，これらの不正アクセスが悪用する特定のポートにおける N/W 監視データの時系列の変化を早期に検知するために SW-PCA を採用した．

3．1　SW-PCA による検知方法

・　パターンの比較による検知

当検知方法は，DoS/アウトブレイク型ワームの拡散時に不正アクセス数の増加が観測されることに着目し，一定時間に含まれるデータを切り出した場合(以下本稿ではパターン)，“不正アクセスが無い状態と有る状態で得られる

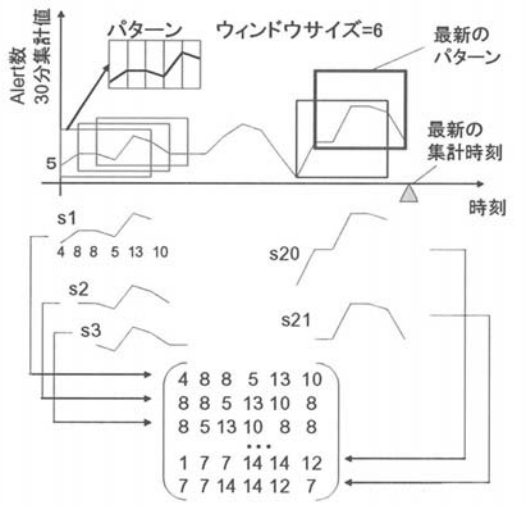

図　3　時系列データに含まれるパターン

パターンに差異がある”という仮定に基づいている．例えば，図 3は特定の TCP ポートへのアクセス数(Alert 数)を 30 分毎に集計した時系列データに対して，ウィンドウサイズを 6，ずらしを 1 としてデータを切り出した例である．ウィンドウで切り出された S1, S2, S3 は不正アクセスが無い時のパターンの例であり，S20, S21 は不正アクセスが有る時のパターンの例である．不正アクセスが無い時のパターンが予め学習パターンとして与えられていれば，後に得られるパターンと類似性を比較することにより，学習パターンに属するパターンか否かを調べることができる．属さない場合に異常なパターンとして検知する．

・　パターンの比較方法

パターンはベクトルとして表現可能であるが，あるパターンが学習パターンに属するか否かの判断について学習パターンの重心からの距離で判断する方法がある．パターンの分布が一様とは限らないことを考慮すればマハラノビス距離(Maharanobis Distance:MD)による判定が適している[5]．しかし，ベクトルの次元が増えるにつれ計算量が多くなる課題がある．

そこで，パターンに PCA を適用しパターンの特徴をより小さい次元で表す特徴量(主成分得点)に変換し，その上でこの特徴量同士の類似性の比較に MD を適用することとした．

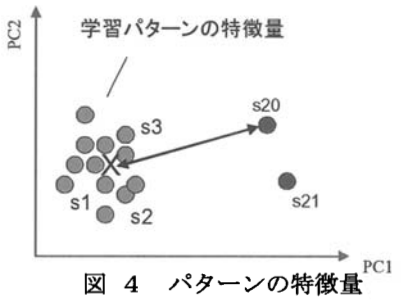

図　4　パターンの特徴量

図 3に示すように，切り出されたパターンは行列化できる．この行列に PCA を適用すると図 4の様に各パターンに対応する特徴量(PC1, PC2)が得られ，学習パターンに含まれる

パターンの特徴量は群を成すが，不正アクセス発生時のパターンの特徴量は群から離れる．学習パターンの特徴量の重心から比較対象のパターンの特徴量の離れ具合を MD で調べることで学習パターンに属するか否かを判断する．

3．2　PCA と SVD

本システムでは PCA の計算方法として SVD(Singular Value Decomposition)を用いている．SVD は任意の行列 X(m×n)を，直行行列 U(m×r)，対角行列Λ(r×r)，直行行列 V(r×n)に分解する演算である．図 3の例で示されるパターンを行に並べた行列を X とした場合，SVD の結果の U に各パターンの特徴量(PC)が算出される（図 5）．なおΛの第 i 番目の対角成分はλi は PCi が元のデータを表現する割合を示す．通常λi の和が 70～80%を満たすように PC を小さい方から選択する．

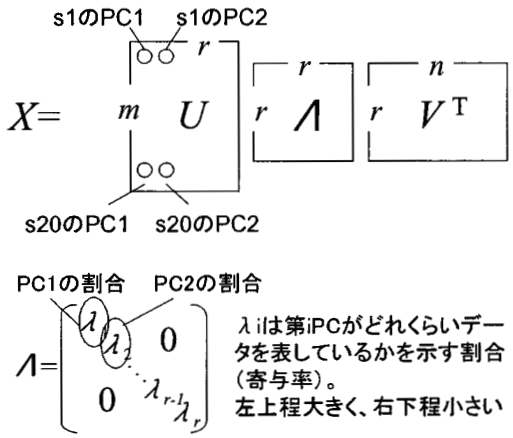

図 5　PCA と SVD の関係

4．検知処理の流れ

検知処理の全体の流れについて記述する．

①　学習パターンの準備

検知開始時点の学習パターンは，分析対象データにおいて，検知開始時点から遡り不正アクセスによる影響を受けていない h（分）を対象にスライディングウィンドウで切り出す．集計時間（分）を c，ウィンドウのサイズを s（分）とすると，切り出される学習パターン数は，(h－s)/c＋1　となる．切り出した学習パターンから，行数=(h－s)/c＋1，列数=s/c　の行列 X_lern を生成する（図6）．

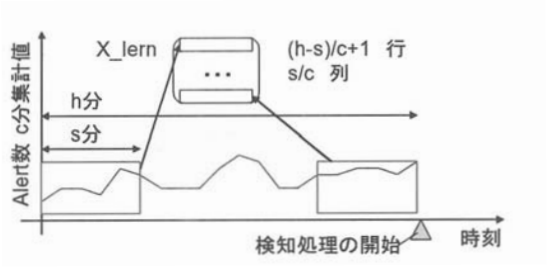

図 6　学習パターンの準備

②　データ集計

分析対象のデータを定期的に集計し分析用の行列を生成する処理である．集計時間 c が経過した後，スライディングウィンドウでパターンを切り出し，x_new とおく．これを X_lern の最後の行に追加し X とおく．

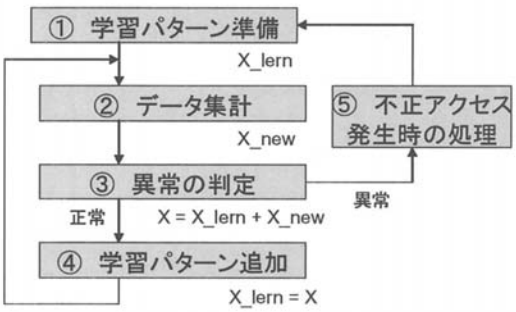

図 7　検知処理の流れ

③　異常の判定

Step1　行列 X に対して PCA を実施し X の各行（各パターン）に対応する特徴量を得る．なお，X の最後の行に対応する特徴量は X_new の特徴量である．

Step2　X_lern の特徴量の重心を求める．

Step3　X の各行の特徴量について重心からの MD を算出する．

Step4　X_lern の各特徴量の MD について，最大値を MD_max とする．X_new の特徴量の MD を

計算し　**x_new の特徴量の MD　＞　MD_max×β** の場合に異常と判定する．ここで，βは係数である．異常と判定された場合は不正アクセス発生時の処理に移行する．

④　学習パターン追加

異常と判定されなかった場合は，X を学習パターン X_lern に置き直す．つまり，異常と判定されなかったパターンは全て学習パターンに加えていく．

⑤　不正アクセス発生時の処理

図 1 における不正アクセス判別機能・対策機能の処理を実施する．

③で異常と判定されない限り，②③④を繰り返す（図 7）．

なお，異常と判定されない状態が継続すると学習パターンが増え続け処理時間に影響を及ぼす．従って，例えば，検知処理を開始してからｈ分経過した場合は，最新のｈ分のデータのみ X_lern に残し，古いｈ分のデータは削除するように学習パターンを更新する．

５．シュミレーションデータによる検知実験

当システムにおける検知機能についてワームのシュミレーションデータを利用し検知実験を行った．

５．１　実験環境

以下の構成の PC をハブで接続し検知実験を行った．なお，DB-PC には予めシュミレーションデータを全て格納してから実施した．

・　DB-PC

CPU:Pentium4 2.4 GHz，メモリ：1GB，OS:Windows Server 2003 std edition，S/W：Oracle 10g

・　分析 PC

CPU：Pentium4 2.4 GHz メモリ 1GB，OS：Windows XP Pro SP2，S/W：データ集計ライブラリ，PCA/SVD/MD ライブラリ，分析ツール

５．２　実験条件

・　正常状態の N/W データ

1 分当たりの集計値が，平均29.5，分散8^2 の正規乱数（ $N(29.5, 8^2)$ ）に従うものと仮定して生成した．

・　ワームのシュミレーションデータ

[9]を元にトラフィックに対するワームによる影響を表すシュミレーションデータを生成した．下記の式で全感染ホスト数の時間変化が表現される．

$$I_t=(1+\alpha)I_{t-1}-(\alpha/N)I^2_{t-1}$$

$$Z_t = m/2^{32} \cdot \eta I_{t-1} \quad , \quad \alpha=\eta N/2^{32}$$

N：ワームが拡散に利用する脆弱性を持つインターネット上の全ホスト数，I_t：時刻 t における全感染ホスト数 I_t ，　η ：1 感染ホストが単位時間当たりに送出するスキャンパケット数

α ：感染レート，Z_t：時刻 t で観測されるトラフィックに対するワームによる影響

当式において[9]と同様に，η ＝357.9 ，N＝360000，$m=2^{20}$，$I_0=10$ と設定し CodeRed のシュミレーションデータを生成した．正常状態のデータに Z_t を足し合わせワーム発生時に観測点で観測されるトラフィックを表現した(図 8)．

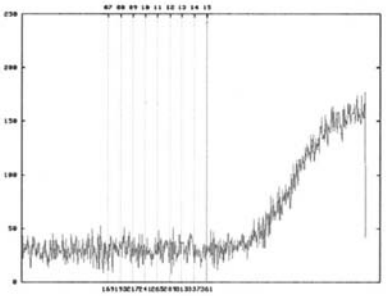

図 8　シュミレーションデータの例

・　パラメータ

(a)　集計時間 5 分，10 分，30 分，60 分

(b)　ウィンドウのサイズ　12，24，48

(c)　学習パターン　168 パターン

(d)　MDmax の係数β=1.0

として検知実験を行った．なお，βについては，正常状態のパターンから異なった時点で即異

常と判断するように1.0に設定した.
なお,検知した時点で処理を停止することとし図 7における処理の⑤は処理しない.

5．3　実験結果と考察

ワームのデータ挿入後,何分で検知したか表1に示す.当実験の結果では,集計時間に対してウィンドウサイズを変化させても検知のタイミングに大きな違いは出なかった.5分集計の場合の様にウィンドウサイズと検知時間に関連性がない結果もある.原因として,どの集計時間においても,ワーム感染後のデータの変動が正常時に比べて大きく変化していることが挙げられる.例えば,5分集計であれば,検知時のパケット数は正常時の約1.47倍,60分では約1.51倍となり,どのウィンドウサイズを適用したとしても検知時のウィンドウの最後のデータの変動が大きいためMDが閾値を超えたと考えられる.

今後は,正常データのパターン,異常データの立ち上がりや形状のパターンを増やし,検知特性を実験する予定である.

表 1　実験データによる検知結果(分)

集計時間(分) / ウィンドウ	5	10	30	60
12	80	90	90	120
24	85	90	120	120
48	65	90	120	120

5．4　処理時間

実験環境による集計 1 回における検知の処理時間はおよそ以下であった.

DBへのアクセス時間　約30秒
PCAの演算時間　約0.2秒
MDの演算時間　約0.3秒
合計 約30.5秒

PCA, MDの演算時間よりもDBへのアクセス時間が支配的であった.

6．おわりに

本稿では,SW-PCAを用いた時系列データのパターンの比較による不正アクセス検知システムについて論じた.現在は主にIDS Alertログを分析対象としているが,今後は高速なN/W環境での検知を実現するため,NetFlowに代表されるフローデータへの対応を行う予定である[10].フローデータはIDS Alertログと異なり正常な通信と異常な通信が混在しているため,その切り分けと効果的な検知方法の検討が課題である.

参考文献

[1] Snort, http://www.snort.org
[2] 不正侵入検知IDS入門,日吉龍,技術評論社
[3] TALOT2 http://www.ipa.go.jp/security/
[4] ISDAS http://www.jpcert.or.jp/isdas/
[5] 図解でわかる　多変量解析,涌井,日本実業出版社
[6] Labib and Vemuri,"Detecting and Visualizing Denial of Service And Network Probe Attacks Using Principal Component Analysis," SAR'04 the 3rd Conference on Security and Network Architectures
[7] 及川ら,"統計的クラスタリング手法によるネットワーク異常状態の検出",信学技報,CS2002-98,pp.83-88(2002)
[8]定点観測による不正アクセス分析システムの提案-ワーム攻撃による異常検出のためのネットワークログ分析手法,平井,鹿島,東 他,IPSJ68回全国大会予稿集
[9] C.C.Zou, L.Gao, W.Grong, and D.Towsley, ``Monitoring and Early Warning for Internet Worms'', In Proceedings of 10th ACM Conference on Computer and Communications Security (CCS'03), 2003.
[10] Y.Gong, 2004, Detecting Worms and Abnormal Activities with NetFlow Part 1, http://www.securityfocus.com/infocus/1796