M_020

サービス可用性を考慮した動的防御のためのACL精緻化方式

An ACL Refinement Method for Balancing Availability and Security

森田 陽一郎†，中江 政行†，小川 隆一†
Yoichiro MORITA, Masayuki NAKAE, Ryuichi OGAWA

1. はじめに

現在，EC サービスや Weblog サービスなど不特定多数のコンシューマ向けのインターネットサービスが多数提供されている．そうしたサービスの提供者にとって，サービス不能 (DoS) 攻撃が大きな脅威である．

この脅威への対策技術として，ファイアウォール (FW)・侵入検知システム (IDS) 連携に基づく動的防御方式が知られている．しかし，サービス可用性を考慮していないため，過剰防衛になりやすいという問題があった．

本稿ではこの問題に対して，サービス重要度・攻撃深刻度・パケット流量に基づく自動的なアクセス遮断・回復により，可用性を極力維持しながら，攻撃被害を抑制する FW のアクセス制御リスト (ACL) 精緻化方式を提案する．本方式の試作と実験により，重要サービスほど遮断する時間を短くできることを確認した．

2. 従来の動的防御方式と課題

近年，ネットワークを流れるパケットの内容に基づいて FW の ACL を動的に生成・更新する動的防御方式が提案・製品化されている．代表例として，FW・IDS 連携方式を挙げることができる．これは，IDS がネットワークを通過するパケットを検査し，不正アクセスを構成するパケットを発見すると，FW にアラートを送信して，当該不正アクセスを即時遮断させる方式である．

しかし，FW・IDS 連携方式では，DoS 攻撃などに対する動的防御について，以下のような課題があった．まず，サービスに必要な正規パケットがアラートに記述された情報と同じ特徴を持つ場合，不正パケットと一緒に正規パケットも遮断してしまい，サービスの可用性を損なうことがある．また，即時遮断するほどではないが数が多くなれば遮断しなければならないような不正パケットに対しては，自動対処することが出来ず，管理者自身が帯域制限や遮断などの対応を行う必要がある．さらに，遮断後の回復についても，管理者自身がアラートやパケットの時系列の分布の変化などを追って判断する必要がある．このような作業は非常に煩雑であるため，判断を誤ってサービスの可用性を損なったり，攻撃の遮断や，やむをえず遮断したサービスの回復が遅れる要因となる．

これらの問題を解決するためには，管理者の人手に依らず，サービスの可用性を考慮しながら動的防御を実現する手法が必要である．

3. ACL 精緻化方式

3.1. アプローチ

提案する ACL 精緻化方式の基本アーキテクチャを図1に示す．ACL 精緻化とは，サービス重要度を制約とした攻撃対策立案 (ACL 生成) を行うものである．

具体的には，個々のサービスへのアクセスや攻撃で観測されるパケットに対し，「必要度」と呼ぶ値を割り当てる．必要度は，パケットを送信するクライアントや宛先となるサービスの重要度と，攻撃の深刻さを包括した概念であり，0〜1 の連続値で表される．最も不要なパケット (深刻な攻撃) は必要度を 0，最も必要なパケット (重要サービスのアクセス) は必要度を 1 とする．必要度の値は，IDS アラートなどの危険度情報，サービスの設定・管理システムの情報や QoS ルールなどに基づき，管理者が設定しておく．

パケットのアクセス可否は，許可/拒否の二者択一ではなく，0〜1 の連続値とし、この値を「許可度」と呼ぶ．連続値のため，パケットの履歴に基づいた，アクセスを許可すべき度合い (アクセス許可の優先順位) の情報を保持できる．許可度は，パケット列と必要度から，3.3 節に述べる ACL 精緻化アルゴリズムに従って算出される．

その後，許可度を，所定の閾値を用いて 0 または 1 に 2 値化した上で，0 を拒否，1 を許可として 当該パケットの ACL を生成し，FW 設定を更新することで，動的防御を実現する．

本方式により，正規パケットが，アラートに記述された情報に該当してしまう場合でも，許可度が一定値を下回らない限り遮断されないため，普段 許可度が高いアクセス (重要なサービス) ほど遮断タイミングが遅くなる．数が多くなれば遮断しなければならない不正パケットに対しても，許可度が高いうちは許可され，許可度が低下すると自動的に遮断される．遮断後の回復についても，正規パケットの必要度が許可度に累積されるので，重要なサービスほど早期に自動回復する．

つまり 本方式は，重要なサービスほど遮断期間を短くできるという点で，サービスの可用性を考慮した動的防御方式といえる．

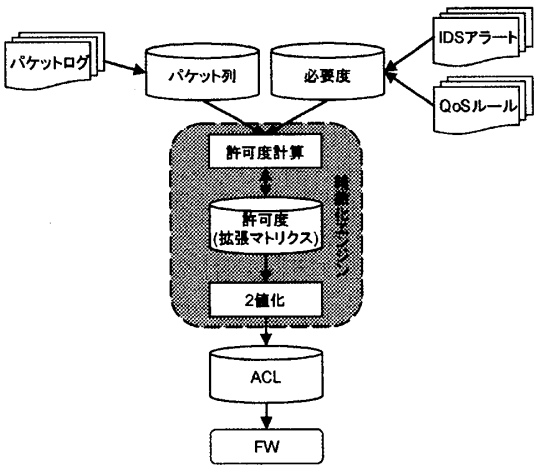

図1: ACL 精緻化方式の基本アーキテクチャ

†日本電気(株) インターネットシステム研究所，
Internet Systems Research Laboratories, NEC Corp.

3.2.アクセスマトリクス

一般に，FW は ACL に記述された複数のパラメータ(送信元 IP アドレス，送信元ポート番号，送信先 IP アドレス，送信先ポート番号，プロトコル)について，特定の値や範囲の指定を組み合わせることで，アクセス制御の対象となるパケットを特定する．また，特定されたパケットに対して，通過を許可するか拒否するかを判定する．つまり ACL は，上記パラメータに関する条件式から 2 値のアクセス可否を導出するための if-then ルールである．その論理構造は，アクセスマトリクスと呼ぶ多次元空間に投射できることが，松田ら [1] により示されている．アクセスマトリクスは，上記パラメータそれぞれに対応する軸をもつ多次元行列であり，各行列の要素に拒否を「0」，許可を「1」とした 2 値を割り当てることにより，ACL の論理構造を正確に表現できる．

ACL 精緻化方式では，このアクセスマトリクスの各要素に，前節で示した許可度を割り当てるように拡張した拡張マトリクスを用いる．

3.3.ACL 精緻化アルゴリズム

① **許可度の更新**: 新たなパケットが到達する度に，パケットのパラメータの組み合わせと，それに割り当てあてられた必要度を用いて，許可度を更新する．例えば，特定の HTTP サービスにある必要度 a が割り当てられており，そのサービスのパケットが到達した場合，パケットのパラメータは拡張マトリクス中のある 1 要素に対応するので，その要素に割り当てられていた許可度と，前述の必要度 a とから，以下の式を用いて新たな許可度を計算し，その要素の許可度を更新する．

$$A0 = C$$
$$Ai = (A_{i-1} * i + a(pi)) / (i + 1)$$

pi: パケット列中 i個目のパケット ($i \geq 1$)
$a(pi)$: pi の必要度
C: 事前定義された許可度の初期値
Ai: i個目までのパケットから算出される許可度

② **ACL 生成**: 許可度を更新した後，その値を，所定の閾値 (例えば 0.5) を用いて 2 値化し，1 であれば その要素に該当するパケットを許可する ACL を生成し，0 であれば拒否する ACL を生成する．

4. 評価実験

4.1.実験方法

本方式に基づいた動的防御動作を確認するため，シミュレーション実験を行った．

実験では，後述するシナリオにしたがって，パケット列を用意し 必要度を設定した．そして，3 章に示した ACL 精緻化アルゴリズムを実行する精緻化エンジンにパケット例を逐次入力して，ACL を生成・更新させた．また 比較のため，同じパケット列を用いた FW・IDS 連携方式のシミュレーションも行った．

【想定したシナリオ】

時刻 t0～t1: 重要サービス向け・通常サービス向けの正規アクセスが一定のトラフィックで連続する．

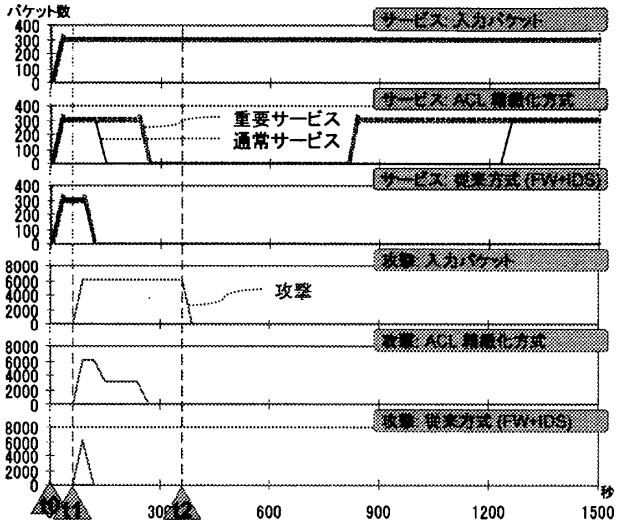

図 2: 通過パケット数の比較

時刻 t1～t2: 重要・通常サービス向けトラフィックの 10 倍のトラフィックをもつ DoS 攻撃が発生する．この間も，正規アクセスが継続する．

時刻 t2～: DoS 攻撃が終わり，時刻 t0～t1 と同様に．正規アクセスが一定のトラフィックで連続する．

4.2. 実験結果

図 2 に動作結果として時間経過 (x 軸) と通過パケット数 (y 軸) の関係のグラフを示す．上から 1～3 段目は正規パケットの通過数の比較，4～6 段目は攻撃パケットの通過数の比較である．

FW・IDS 連携方式では，時刻 t1 での攻撃発生の直後に，攻撃もサービスも即時遮断してしまうのに対して，ACL 精緻化方式では，重要なサービスほど長い時間維持されていることがわかる．つまり，複数のサービスが同じ深刻さを持つ攻撃を受けたとき，重要なサービスほど，サービス可用性の維持が優先される．また，攻撃パケットが終息し，アラートの発生が収まると，重要サービスから先に回復する．

他の特徴としては，パケットの数が多いほうが，許可度への影響が早く・大きくなるため，現状の計算式では，サービスよりも多量のパケットによる攻撃に対しては，遮断よりも回復のほうが，時間が長く必要であることが観測された．重要サービスの遮断時間をさらに短くするため，回復にかかる時間を短縮することが今後の課題である．

5. おわりに

サービスの可用性を考慮した動的防御を実現する ACL 精緻化方式の基本アルゴリズムについて述べた．また，本方式に基づく ACL 精緻化エンジンを試作し，DoS 攻撃を想定したシミュレーション実験を通じて，重要サービスほど遮断時間を短くできることを確認した．

参考文献

[1] 松田 勝志, “マトリックス分解によるパケットフィルタリングルールの分析 —不要ルールと冗長条件ルールの検出—,” 情報処理学会研究報告, v. 2005, n. 122, 2005, pp. 1-6.