Bloom Filterによるキャッシュポリューション攻撃の検知

Detecting Cache Pollution Attack with Bloom Filter

芦原　大和　　上山 憲昭
Takakazu Ashihara　　Noriaki Kamiyama
福岡大学　工学部 電子情報工学科
Faculty of Engineering, Fukuoka University

1. はじめに　Web 閲覧や動画配信サービスを快適に行うため、ネットワークの様々な場所にキャッシュサーバを設置しユーザの近くに存在するキャッシュサーバからデータを配信するキャッシュ配信が広く行われている．しかし悪意を持ったユーザが意図的に低人気のコンテンツに多数の要求を行うことでキャッシュの効果を低下させる Cache pollution attack (CPA) の問題が指摘されている [2]．CPA 発生時にはできるだけ迅速に攻撃ホストを特定し，キャッシュを保護することが重要である．そこで筆者らは，少ないメモリ量とメモリアクセス回数で高精度に攻撃ホストを特定する，ホスト ID とコンテンツ ID の組をキーとする Bloom Filter (BF) を用いた CPA ホストの検知法を提案した [1]．本稿では [1] を拡張し，正常ホストの誤検知を抑制するため BF による検知回数に閾値を設けること，また継続的な CPA ホスト検知を可能とするため BF を 2 つ用意して交互に運用することを提案する．そして計算機シミュレーションによる性能評価を行い，提案方式を用いることで正常ホストの誤検知を防ぎつつ，CPA ホストを高精度に検出しキャッシュヒット率の低下を回避可能なことを示す．

2. 提案方式　筆者らが [1] で提案した方式では 1 つの bitmap (BM) のみを用いるため，BM のビットセットを反復していくにつれ，やがて BM のほとんどの位置が 1 で埋まり，全てのホストが CPA ホストとして検知されるため永続的な運用ができない．また，正常なユーザも同一コンテンツを複数回，視聴する可能性があること，さらに BF の偽陽性判定により，CPA ホストとして誤検知する可能性がある．

BF を永続的に使用するため，BF 用に 2 つの BM を用意し，これらを切り替えながら運用することを提案する．各 BM に挿入するキーの総数を BF への入力回数上限 N 以下に抑えることから，BM への挿入キー数が N に達した時点で，もう一方の BM に運用を切り替える．ただし一方の BF から他方の BF に切り替えた時点で，新規に稼働を始める BF の BM は全てのビットが 0 の状態から始めると，CPA ホストの検知が困難である．そこで運用開始の閾値パラメタ α を導入し，運用中の BM の挿入キー数が αN に達した時点で，もう一方の BM へのキーの挿入も並行して行う．そのため $\alpha < 0.5$ のとき，3 個以上の BM が必要になるので，$0.5 \leq \alpha \leq 1.0$ の範囲で α を設定する．その時に検出に使用している BM を MBM と定義する．以下の手順で 2 つの BM を運用する．初めに 2 つの BM を初期化 (全ビットをゼロに設定) した状態で運用を開始する．MBM の挿入キー数が αN に達した時点で他方の BM の更新も並行して開始する．MBM の挿入キー数が N に達した時点で MBM を初期化して，他方の BM に運用を切り替える．この切り替えの手順を繰り返す．

正常ユーザの誤検知を減らすため，BF での検知回数に BF 検出閾値 Y を設け，Y 回検知されたホストを CPA として検知する．そのため BF 検知された各ホストの検知回数を記録するためのテーブル (BFDC: BF detection count) を用意し，ホスト u からの配信要求時には，まず BFDC の u に対する BF 検知回数 D_u をチェックし，$D_u < Y$ の場合は BF 検索を行う．BF によって検知されなかった場合は，BM を更新し通常通りキャッシュを行う．BF によって検知された場合は，D_u のカウンタをインクリメントする．$D_u = Y$ の場合は，CPA ホストとして検知してキャッシュしない．

検知を回避するため周回的に要求コンテンツを変える Smart 型の CPA[1] の場合，同一の CPA ホストが同一のコンテンツを要求するまでに C 回の要求が発生する．一方で提案方式の各 BM は N 回のキー入力まで受け付けられるため，提案方式で BF による CPA ホスト検出を行うには，N が満たすべき下限値 N_{min}(検出リミット) が存在する．全正常ユーザの毎秒要求数を R_n，全 CPA ユーザの毎秒要求数を R_c，CPA ホスト数を N_c とすると，1 つの BM に必要なメモリ量の下限値 M_{min} は次式で与えられる．

$$M_{min} = -\frac{N_c\,(R_n + R_c)\,(C + I)\ \log\ p}{R_c(\ \log\ 2)^2} \tag{1}$$

3. 性能評価　提案方式の有効性を計算機シミュレーションにより評価する．キャッシュ置換方式は LRU を想定し，コンテンツ数 $M = 10,000$ に対しキャッシュサイズは 500 とする．正常ホスト数 $N = 10,000$ で 6,000 秒間シミュレーションを行う．$N_c = 10$ 個の CPA 攻撃ホストが存在し，各々はコンテンツの人気順位を把握していることを想定し，2,500 秒から 5,800 秒の期間，攻撃ホストは CPA 対象コンテンツ $C = 1,000$ 個に対して要求を行う．最も低人気の C 個のコンテンツに対しランダムに要求を行う Rand 型，攻撃対象コンテンツ数/CPA ホスト の間隔で等間隔に各 CPA ホストの要求開始点を決め，人気の昇順にコンテンツを要求する Smart 型を CPA 種別として定義する．BF の運用開始パラメタ α を 0.5 に設定する．BF の各 BM のメモリサイズは M_{min} より大きな 300KB に，BF の偽陽性の発生確率を $p = 0.01$ で BM を設計する．BF 検出閾値を $Y = 10$ とする．正常ホストはパラメタ $\theta = 0.7$ の Zipf 分布に従いランダムに選択したコンテンツを要求する．全 CPA ホストの総要求発生レートの，全正常ホストの総要求発生レートに対する比率を $\gamma = 1.0$ に設定する．結果は全て 10 回の施行における平均値で行う．

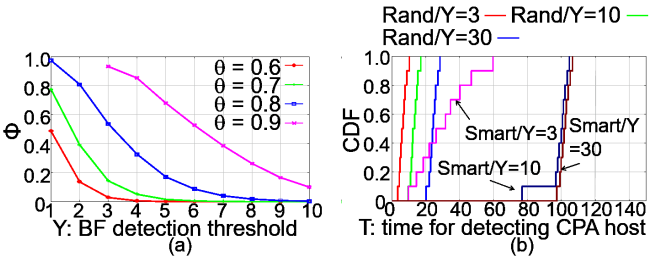

図 1:（a）巻き添え率（b）検出時間の累積分布

BF は偽陽性により，過去に要求がないにもかかわらず要求があったと判断される False positive の生じる可能性がある．正常ホストが提案方式によって誤検出される割合を巻き添え率 Φ と定義する．巻き添えは BF の False positive に加え，同一ホストが偶然，同一コンテンツを要求した場合にも発生する．

図 1(a) に Zipf パラメタ θ の 4 つの値に対し，BF 検出閾値 Y を変化させたときの Φ をプロットする．θ に増加に伴い，要求対象コンテンツの偏りが増加するため Φ は増加する．メモリサイズを増加させることで，より長時間，同一の BM を CPA 検出に使用できるため CPA の検出能力は向上するが，一方で Φ が増加する．しかし BF 検出閾値 Y を増加させることで，Φ を抑えることが可能である．

図 1（b）に，CPA が開始されてから各 CPA ホストの検出に要した時間の累積分布を BF 検出閾値の 3 つの値に対し示す．Smart 型 CPA は Y が増えても検出時間はほとんど増加しない．これは Smart 型では C のコンテンツを周回的に要求するため，2 回の同一コンテンツの要求が始まると連続して BF の検知がなされるためである．そのため Y を増加させても CPA ホストの検知に要する時間の増加を避けることができる．

謝辞本研究成果は JSPS 科研費 18K11283 の助成を受けたものである．ここに記して謝意を表す．

参考文献

[1] 芦原大和, 上山憲昭, ”MEC キャッシュにおけるキャッシュポリューション攻撃の検知”, 信学会 NS 研究会，NS2019-108, 2019 年 10 月．
[2] S. Paul, A. Seetharam, A. Mukherjee, and M. K. Naskar, Investigating the Impact of Cache Pollution Attacks in Heterogeneous Cellular Networks, IEEE ICNP 2017.