Traceroutの発生間隔に基づくCrossfire Attack 攻撃検知方式

1. 研究背景

■Crossfire Attack (CFA): 特定のサーバではなく，複数のターゲットサーバ(TS)を含むターゲットエリア(TA)に至るネットワークのリンクを高負荷にすることで，TSにパケットを到達不可にする攻撃
■攻撃対象リンクの選定のため攻撃に先立ち，攻撃に用いる大量のボットからTSに対し大量のtracerouteが発生

- traceroute：実行したノードから指定したノードまでの経路（経由するルータ）のリストを得ることが可能

2. 研究の目的

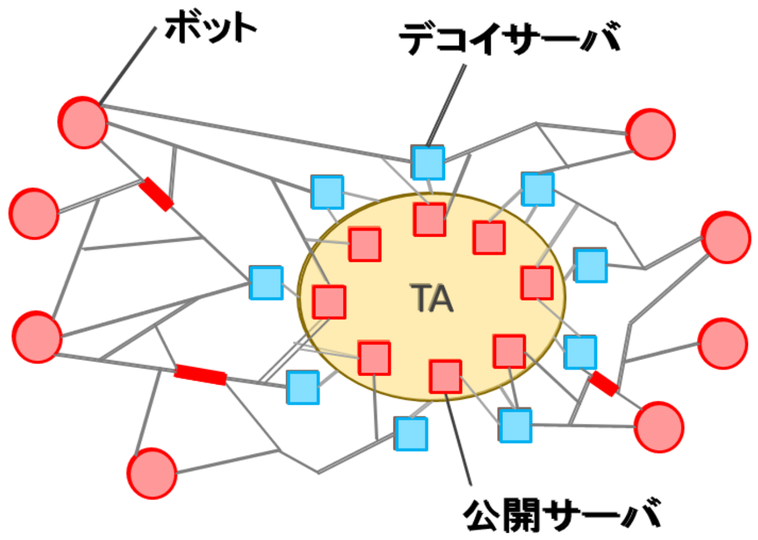
■攻撃に先立ち発生するtracerouteの発生間隔に基づく，CFAの検知・防御方式を提案

■発ホスト・着サーバの両方に対し，tracerouteの発生間隔を考慮 ⇒正常ホストの誤検知を抑制しつつ，ボットを検知

3. 提案防御方式

■CFA-攻撃者の流れ

1. 多数のボットからTA内の多数の公開サーバに対しtracerouteを行うことで，TA内のサーバ宛てのフローの多くが経由する少数のリンク(攻撃対象リンク)を発見
2. 多数のボットからTAの周囲に存在する多数の公開サーバ(デコイサーバ)に対しtracerouteを行い，攻撃対象リンクをフローが経由するボット・デコイサーバ組を選定
3. 選定したボット・デコイサーバ組間に，少量の正常トラヒックを発生 ⇒ 攻撃対象リンクを過負荷



■攻撃に先立ち，ボット・デコイサーバ組の選定のため短い時間内に連続して大量のボット・TS間にtracerouteが発生
⇒閾値時間（Ts)内に連続してtracerouteを行ったホストをボットとして検知しブラックリスト(BL)化
■正常なホストもTs以内に連続してtraceroute を行う可能性があり，ボットとして誤検知される可能性
■TSはボットから短い時間に連続してtracerouteを受ける
⇒閾値時間(Td)内に連続してtracerouteを受けたサーバをターゲットサーバリスト(TSL)化
⇒閾値時間（Ts)内に連続して，TSLに含まれるサーバに対してtracerouteを行ったホストをボットとして検知しブラックリスト(BL)化

■ traceroute間隔の検査方法：
任意のルータが発もしくは着IPアドレスをキーとするハッシュテーブルを用いてtraceroute発生時刻を記録

4. 検知閾値の最適設計

■提案方式においては，2つの検知閾値Ts，Tdをどのように設定するかが課題
■正常ホストの誤検知確率の許容最大上限を設定し，それを満足する発ホスト閾値Tsと着ホスト閾値Tdを設定
⇒閾値に対する正常ホストの誤検知確率FPRを簡易な式で導出

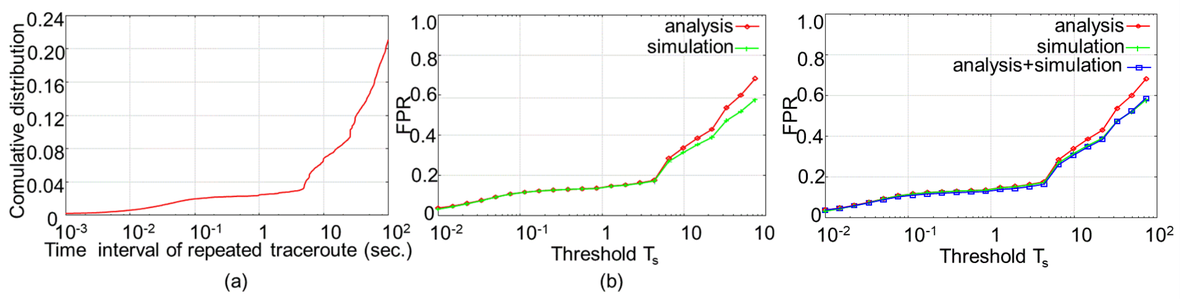
■サーバがTSLに入る確率の導出

- 正常ホストのtraceroute発生間隔と，全正常ホストのtraceroute発生間隔は実測の分布を付与

■正常ホストの誤検知確率の導出

- 測定開始時点において，既にTSLが作成されていることを想定
- 正常ホストの検知確率P_n： $p_n = \sum_{k=1}^{\infty} P_n(k)h(k)$
  - D(Ts): 1組の連続する2回のtracerouteによって，ある正常ホストが検知される確率
  - $P_n(q)$: 測定期間中にq回のtracerouteを行った正常ホストが検知される確率: $P_n(q) = 1 - \left\{1 - D(T_s)\right\}^{q-1}$
  - h(k): ある正常ホストが測定期間中にk回のtracerouteを行う確率

5. 性能評価



■閾値T_sを変化させた際のFPRをプロット

- 解析値とシミュレーション値を比較
- 解析値とシミュレーション値は約7秒より小さい場合はおおよそ一致するが，約7秒より大きいとFPRの誤差が大
- FPRの増加傾向は正常ホストのtraceroute発生間隔と似た傾向

■導出式においてポアソン分布に従うと仮定した値に関して，シミュレーションによって得られた値を用いたグラフをプロット ⇒ おおよそシミュレーション値と一致

6. まとめと今後の予定

■CFAは攻撃に先立ち短い時間内に多数のボットからTS宛てにtracerouteを発生させる特徴あり
■2つの閾値を用いたCFA検知方式を提案
■最適閾値を設定するために，簡易的な式でのFPR導出
■提案したFPR解析式は，Tsが約7秒より小さい場合には，高精度にFPRを推定
■正常ホストのtraceroute発生間隔の仮定に問題
■今後は，攻撃者の挙動やTSの選定方法を変更し，FPRの解析式を改善する予定