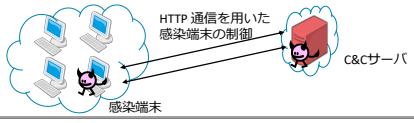
時系列アクセスパターンのテンプレートを用いたマルウェア感染端末検知

外口 大凱*, 大下 裕一*, 芝原 俊樹†, 千葉 大紀†,
秋山 満昭†, 村田 正幸*

* 大阪大学 大学院情報科学研究科
† NTT セキュアプラットフォーム研究所

研究背景

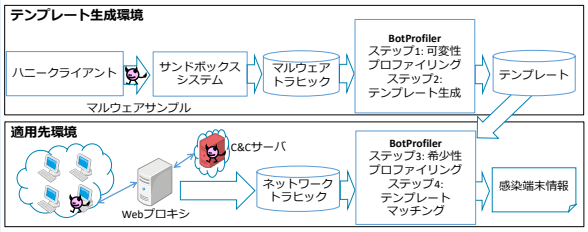
- **マルウェア感染端末検知は重要な課題**
  - 攻撃者は C&C サーバ経由で感染端末を制御
  - 感染が拡大することで攻撃者は大規模な攻撃が可能
- **2 種類の感染端末検知手法が存在**
  - ホストベース: 端末上でマルウェアが動作していないか検出するプログラムを実行
    - 端末が攻撃者によって制御されている場合，プログラム自体が停止
  - ネットワークベース: 通信を検査することで感染端末を検知
- **HTTP 通信に着目した感染端末検知**
  - C&C サーバは HTTP 通信を用いて感染端末を制御
  - C&C サーバが行う HTTP 通信をもとにした研究が多く存在



HTTP 通信を用いた感染端末検知

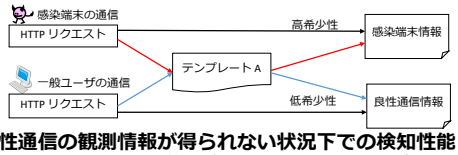
- **テンプレート**
  - マルウェアが送出する HTTP リクエストをもとに生成
    - ➢正規表現等を用いることで攻撃規則をテンプレートに反映
- **テンプレートを用いた従来手法**

Path: /images/<str;5>.<str;3>
Query: ?<str;2>=<int;5>
UserAgent: Opera/<int;1>.<int;2> (Windows <str;2> <int;1>.<int;1>)
テンプレートの例



従来手法の検知手法の課題

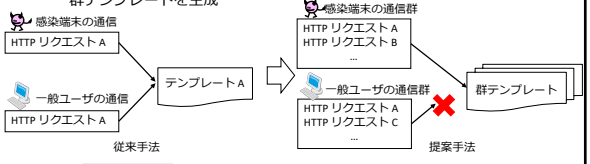
- **正規化による良性通信の誤検知**
  - 適用先環境の観測情報を用いた誤検知の抑制が実施
  - 観測情報を事前に収集し，各要素の頻出度 (希少性) を算出
    - 通信の各要素の希少性が高い ≒ 適用先ネットワーク内での出現が稀
      - ➢感染端末の可能性大
  - テンプレートの類似度が高く，希少性が高い HTTP リクエストを感染端末として検知



- **良性通信の観測情報が得られない状況下での検知性能**
  - 十分な良性通信の観測情報が得られない状況下では，多くの良性通信の誤検知が発生
  - 誤検知を避けるよう閾値を設定した場合，感染端末の検知漏れが発生

研究目的

- **良性通信が十分に得られない状況下において高い検知率と低い誤検知率を両立できる手法を提案**
  - マルウェアが複数の HTTP リクエストを発生させていることに着目
    - マルウェアが発生させている HTTP リクエスト群を束ね，群テンプレートを生成

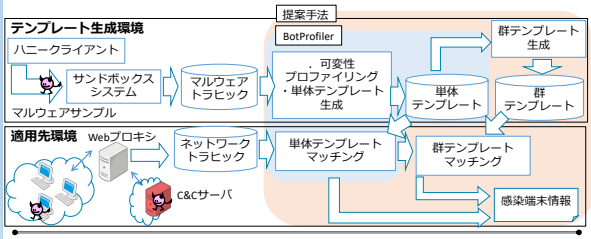


研究の目的

群テンプレートによる，良性通信の観測状況が得られない状況下でのマルウェア感染端末の高い検知率と良性通信の低誤検知率の両立

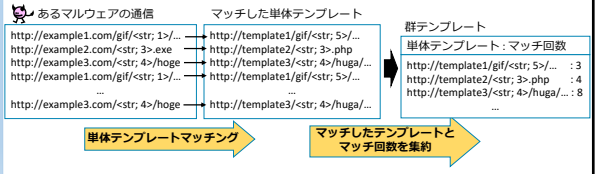
提案手法の概要

- **テンプレート生成環境**
  - 生成された単体テンプレートを用いて群テンプレートを生成
- **適用先環境**
  - 適用先で発生する HTTP リクエストを時間分割し，HTTP リクエスト群を生成
  - 各 HTTP リクエスト群に対しマッチングを実施



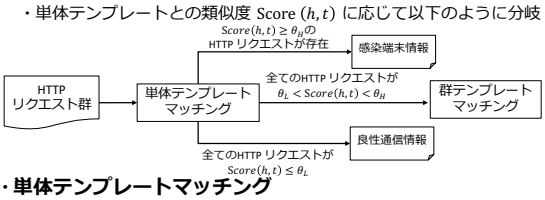
群テンプレートの生成

- **各マルウェアごとに群テンプレートを生成**
  - HTTP リクエストを各マルウェアごとに集約し，HTTP リクエストと単体テンプレートのマッチングを実施
  - マルウェアごとにマッチした単体テンプレートとマッチ回数を集約し，「群テンプレート」として生成
    - ➢複数通信を行う感染端末の挙動を確認



提案する感染端末検知手法

- **検知手順**
  - HTTP リクエスト群中の各 HTTP リクエストに対し単体テンプレートマッチングを実施
  - 単体テンプレートとの類似度 Score (h,t) に応じて以下のように分岐



- **単体テンプレートマッチング**
  - HTTP リクエスト群中の HTTP リクエスト h と各テンプレート t の類似度をスコア Score (h,t) により算出
  - HTTP リクエストとテンプレートの各要素 (URL パス，URL クエリ，ユーザエージェント) の類似度が高ければ Score (h,t) も高くなるよう設計

群テンプレートマッチング

- **群テンプレートとの類似度を利用したスコア計算**
  - HTTP リクエスト群 D と 群テンプレート T の類似度を群スコア S(D, T) により算出
    - 群テンプレート内の単体テンプレートのうち，HTTP リクエスト群にてマッチした単体テンプレートの割合が高くなれば S(D, T) も高くなるよう設計
- **$S(\mathrm{D},\mathrm{T})$ をもとに感染端末を検知**
  - S(D, T) ≥ θ_G のとき当該 HTTP リクエスト群を悪性と検知

$$\mathrm{S}(\mathrm{D},\mathrm{T}) = 1 - \frac{1}{|U_T|}\sum_{i \in U_T} s(d_i, t_i)$$

U_T: T において 1 回以上マッチしたURLの集合

$$s(d_i,t_i) = \begin{cases} \alpha & d_i = 0 \\ \beta(t_i - d_i)/t_i & 0 < d_i \leq t_i \\ 0 & d_i > t_i \end{cases}$$

d_i, t_i : D, T それぞれにおいて i 番目の単体テンプレートにマッチした HTTP リクエストの個数
α, β: パラメータ ($\alpha >> \beta$)

評価手順

- **サンプル収集**
  - 悪性データ: マルウェアサンプルをサンドボックス上で実行して収集
  - 良性データ: 実際に大学内で日常的なネットワーク利用の際に生じるトラヒックを収集

ラベル	教師		テスト	
	期間	HTTP リクエスト数	期間	HTTP リクエスト数
悪性	2017/8/1 – 2017/12/31	656,714	2018/1/1 – 2018/3/31	442,532
良性	-	-	2018/3/1 – 2018/3/31	293,120

- **HTTP リクエスト群の生成**
  - ある HTTP リクエストから時間 s 秒以内に送信されている HTTP リクエストを 1 つの HTTP リクエスト群として生成
- **検知指標**
  - 検知率 (TPR: True Positive Rate)
    - テストデータの全感染端末中の検出された感染端末の割合
  - 誤検知率 (FPR: False Positive Rate)
    - 生成された HTTP リクエスト群における誤って検出された HTTP リクエストの割合

提案手法と従来手法の比較

- **従来手法に比べて検知性能が向上**
  - θ_H が大きくなると検知率は低下するが，誤検知率は減少
  - θ_L が大きくなると検知率は低下するが，誤検知率は減少
- **θ_L, θ_H の制御により検知性能が向上**
  - θ_H を高い値に設定し，$\theta_L < \theta_H$ とすることで低い誤検知率を維持しつつ高い検知率を達成

	θ_H		0.87	0.90	0.93
TPR	BotProfiler without RP		87.42%	30.48%	29.28%
	Proposed	θ_L			
		0.30	**95.23%**	94.35%	92.94%
		0.78	87.87%	86.12%	84.69%
		0.87	-	84.23%	82.77%
FPR	BotProfiler without RP		3.02%	2.79%	0.42%
	Proposed	θ_L			
		0.30	4.06%	3.90%	2.03%
		0.78	3.18%	3.02%	1.14%
		0.87	-	2.75%	**0.88%**

BotProfiler without RP: BotProfiler で希少性を用いない手法
Proposed:提案手法

$\theta_G = 0.85$ のときの提案手法と従来手法の比較

まとめ

- **研究背景**
  - マルウェア感染端末検知は重要な課題
  - HTTP 通信に着目した研究が多く実施
- **従来手法の課題と研究目的**
  - 良性通信が十分に得られない状況下では検知率と誤検知率の両立が困難
  - マルウェアが複数の HTTP リクエストを送出することに着目し，群テンプレートを生成することを提案
- **提案手法**
  - HTTP リクエスト群に対し単体テンプレートマッチングを実施
  - 類似度が中程度となった HTTP リクエスト群に対し，群テンプレートマッチングを実施
- **評価結果**
  - 閾値を制御することで従来手法より検知性能が向上