BotProfiler++: Detecting Malware-Infected Hosts using Templates of Time-Series HTTP Request Patterns

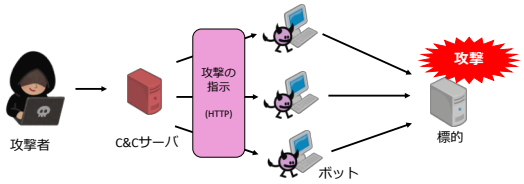
外口 大凱

大阪大学 大学院情報科学研究科
村田研究室

マルウェア感染端末による脅威

・**マルウェア感染端末 (ボット) を使った攻撃が発生**
- 攻撃者はコマンド & コントロールサーバ (C&C サーバ) 経由でボットを制御
- 感染が拡大することで攻撃者は大規模な攻撃が可能

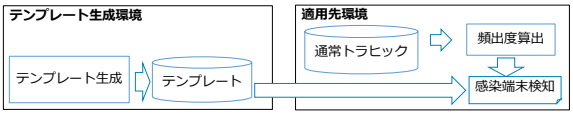
サイバー攻撃を軽減するためにはボットの検知が重要



マルウェア感染端末による攻撃の例

1

BotProfiler[1]: HTTP 通信に着目した感染端末検知手法



・**テンプレートを生成**
- マルウェアが用いる HTTP リクエストの特徴を抽出

Template #1
Path: /logos_<str;1>.gif
Query: <hex;6>=<int;7>
UserAgent: Opera/<int;1>.<int;2> (Windows NT <int;1> .<int;1>; <str;1>; en)

テンプレートの例

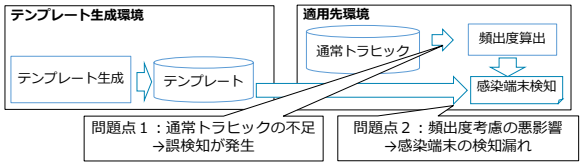
・**顧客である適用先の観測情報を収集し HTTP リクエストの頻出度を算出**

・**各 HTTP リクエストに対しテンプレートの類似度と頻出度をもとにマルウェア感染端末を検知**

[1] D. Chiba, T. Yagi, M. Akiyama, K. Aoki, T. Hariu, and S. Goto, "Botprofiler: Detecting malware-infected hosts by profiling variability of malicious infrastructure," IEICE Transactions on Communications, vol. 99, no. 5, pp. 1012–1023, 2016.

2

BotProfiler の問題点



・**問題点1: 十分な観測情報が得られていない状況下での検知性能の劣化**
- BotProfiler 適用直後といった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テンプレートの類似度の閾値を調整した場合感染端末の検知漏れが発生

・**問題点2: 頻出度を考慮することによる感染端末の検知漏れ**
- テンプレートの類似度が高い場合であっても頻出度が高いマルウェアの通信は検知されない
  - ➢マルウェアの通信においても頻出度が高い，一般的に見られる通信が発生

3

研究目的と提案手法

・**研究目的**

適用先の観測情報が得られない状況下においても高い検知率と低誤検知率を両立できるマルウェア感染端末検知手法

・**提案手法**

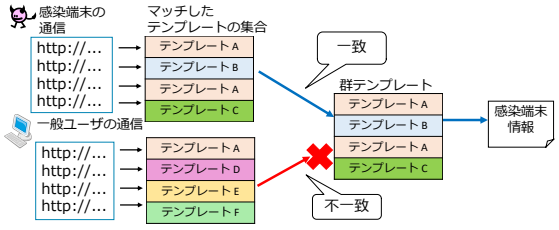
マルウェアが送出する複数の HTTP リクエストを束ねたテンプレートマッチングの実施

- 多くのマルウェアが複数の HTTP リクエストを生成
- マルウェアの通信とマッチしたテンプレートの組み合わせからなる「群テンプレート」を生成・利用する手法を提案

4

提案手法の概要

・**マルウェアが送出する HTTP リクエストの組み合わせをテンプレート化（群テンプレート）**



・**感染端末が行う通信全体の特徴を捉えることが可能**
- 誤検知を抑えつつ，高い検知率を期待

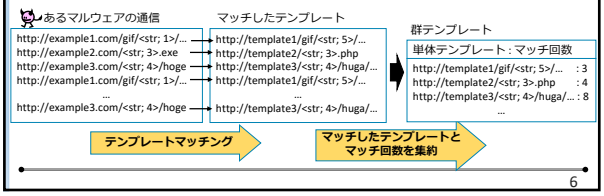
5

群テンプレートの生成

・マルウェアが送出するHTTP リクエストをもとに生成

手順

1. HTTP リクエストのテンプレートを生成
2. 各マルウェアについて，HTTP リクエストとテンプレートを照合
3. 各マルウェアが送出する HTTP リクエストとマッチしたテンプレートとマッチ回数を集計し，群テンプレートとする



検知の流れ

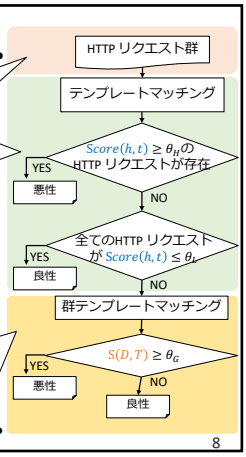
HTTP リクエスト群の生成
・適用先で流れる HTTP リクエストを一定時間ごとに分割

HTTP リクエスト群中の各HTTP リクエストで感染有無を判断
・テンプレートとの類似度 $Score(h,t)$ をもとに感染の有無を判断

群テンプレートマッチングによる感染端末検知
・HTTP リクエスト群 D と群テンプレート T の類似度を $S(D,T)$ をもとに検知

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

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



評価環境

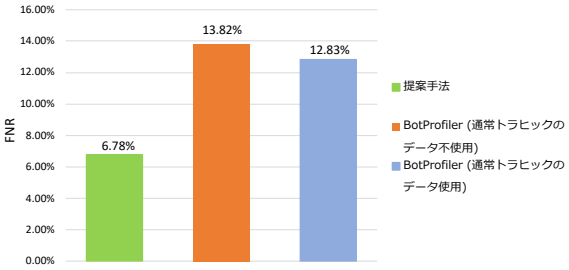
・**データセット**
- 悪性データ: マルウェアサンプルをサンドボックス 上で実行して収集
  - 教師データはテンプレート，群テンプレートの生成に使用
  - サンドボックス: 攻撃が外部に影響しないよう設けられた環境
- 良性データ: 実際に大学内で日常的なネットワーク利用の際に生じるトラヒックを収集
  - 教師データは BotProfiler（比較対象） の頻出度算出にのみ使用

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	2017/12/1 – 2017/12/31	291,343	2018/1/1 – 2018/3/31	876,778

・**検知指標**
- False Negative Rate: FNR
  - テストデータの全感染端末のうち検出されなかった感染端末の割合
- False Positive Rate: FPR
  - 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

・**提案手法で最小 FNR を達成**
- 群テンプレートにより，従来検知できなかった感染端末を検知



各手法のパラメータに対し FPR が 3% 以下を達成できるように調整

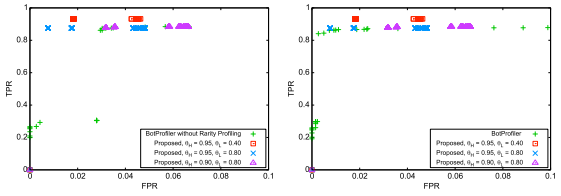
まとめと今後の課題

・**まとめ**
- 良性通信の観測情報が得られない状況下においても，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を両立する手法を提案
- 多くのマルウェアが複数通信を行っていることに着目し，複数の HTTP リクエストを束ねた群テンプレートを生成
- 誤検知率が 3% 以下となるようパラメータを調整したとき，提案手法にて検知率を最大 93.22% まで達成

・**今後の課題**
- 良性通信 / 悪性通信の混在環境における提案手法の適用
- パラメータの適切な設定

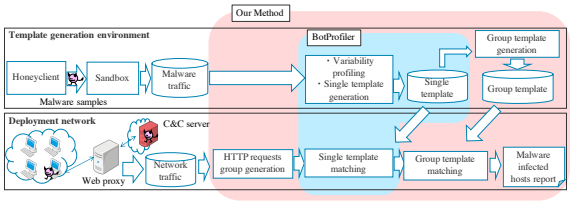
Appendix: 検知率と誤検知率の関係

・**提案手法と BotProfiler の TPR, FPR を比較**
- 通常トラヒックを用いない BotProfiler より高い検知率と低誤検知率を実現
- 通常トラヒックを用いた BotProfiler に比べ，高い検知率を実現



Appendix: 提案手法の概要

- **テンプレート生成環境**
  - テンプレート，群テンプレートを生成
- **適用先環境**
  - HTTP リクエスト群の生成
  - テンプレートマッチングによる感染の有無の判断
  - 群テンプレートマッチングによる感染端末の検知



14