Vol.2020-CSEC-90 No.10
Vol.2020-SPT-38 No.10
2020/7/20

詐欺メール対策の試作

今橋泰則 [1]

概要：いわゆる詐欺メールは URL やマルウェアの添付など有害なデータが添付されていないのでアプライアンスでは検知できない．本稿ではメールのメタ情報を利用してこのようなメールの検知方法を探った経緯を紹介する．

キーワード：詐欺メール，類似ドメイン，ゲシュタルトパターンマッチング，メールのやり取り

How to Hunt the BEC (Business Email Compromise)

YASUNORI IMAHASHI[†1]

Abstract: [**]

Keywords: BEC (Business Email Compromise), Pattern matching, White List, mail sequence
.

1. 背景

不正なメールの中心はいわゆる「バラマキ」型と呼ばれ，不特定多数に送り付け，URL の記載や実行形式ファイルの添付などの特徴があり専用の装置での除去が容易になったことや利用者の意識向上により被害は減少傾向にあるようだ．

代わって我々が注目したのはこれらの特徴を伴わない不正なメールである．いわゆるビジネスメール詐欺(BEC:)メールである．実態はメールを使った「オレオレ詐欺」で被害も報道されている．

本稿ではあるモデルを想定しその対策を実施した過程を報告し参加者との情報共有を目的としている．

2. 現状整理と攻撃モデルの想定

社内での観測を元に 2 つのエッセンスでメールを分類することで現状を整理した．図 1 メールの分類で X 軸，Y 軸をエッセンスとして分類した．エッセンスの説明である．

- ✓ 攻撃者のドメインの年齢，図中では X 軸で Age と記載．取得してからの時間である．若いと攻撃元になりやすい．取得即攻撃．その年齢は 100 日以下とした．これは分布から十分な年齢と判断した．
- ✓ 攻撃者のドメイン名は取引相手のそれと類似している．詳細は後述する．
- ✓ 攻撃者の表示アドレス(Header)と実アドレス(Envelope)は同じである． 簡単に判明できるアドレスの差異を生じさせるようなことはしない．むしろ第 1 象限(Age<100 で HeaderFrom≠EnvelopeFrom)のバラマキ型で良く観測されている．第 2 象限の分類では Age>100 としているが，クラウドサービスからの送信が殆どで，表示アドレスをわかりやすくするために実アドレスであるサービス元のドメインと異なることを確認している．第 3 象限は 2 つのエッセンスの両方が良好という状態であるので正常と分類した．

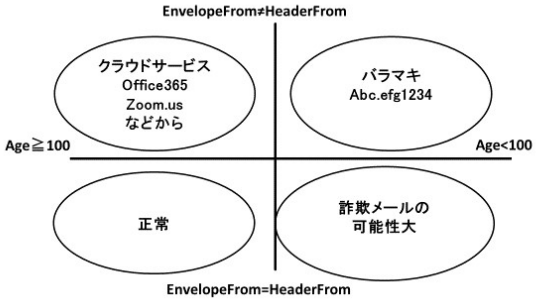

図 1 メールの分類

対象は第 4 象限であることは決まった．次にそこから不正メールを抽出するためのフィルタとしてモデルを想定する．攻撃者は何らかの方法でメールのやり取りの情報を得て，「図 2 攻撃モデル」の上から下へ X と Y のやり取りの間に入って詐欺へと導く．

ここではメールの盗聴方法は問わない．結果として図の下の状態をモデルとして想定し★の区間で検出する方法を試みた．

1 NEC ソリューションイノベータ

ⓒ2020 Information Processing Society of Japan

Vol.2020-CSEC-90 No.10
Vol.2020-SPT-38 No.10
2020/7/20

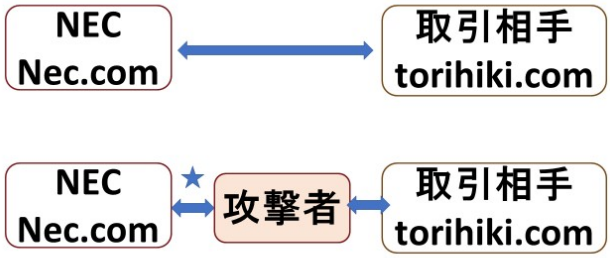

図 2 攻撃モデル

3. 類似ドメインの抽出

類似とは何か(A)と何か(B)が似ている，と言い換えることができ，本モデルで(B}は着信したリアルタイムのメールの送信元となり，(A}をどうするかを考えた．

これはモデルから自然に導かれる．つまり過去にやり取りした送信元となる．送信元をメールアドレスとするとデータが冗長となるのでドメインとして丸めた．

過去の送信先ドメインのリストとリアルタイムで受信した送信元ドメインを比較し類似度を計測した．

類似度は「ゲシュタルトパターンマッチング」という人

```
import difflib
str1 = u"バナナ"
str2 = u"ハバナ"
s = difflib.SequenceMatcher(None, str1, str2).ratio()

print str1, "<~>", str2
print "match ratio:", s, "¥n"

# >> バナナ <~> ハバナ
# >> match ratio: 0.833333333333
```

間の目からみて「正しい感じ」にマッチする1ようなアルゴリズで Python の DIFFLIB というライブラリで実装されており，試用したところ良好だったのでこれを用いた．

4. 観測結果

下記は観測結果の例である．(マスク済み)

嫌疑ドメイン	類似度	比較対象ドメイン
sanyuu-ggggg.co.jp	0.97	sanyu-ggggg.co.jp
jjjjjj-syste.co.jp	0.97	jjjjjj-system.co.jp
itelefonica.com.bb	0.97	telefonica.com.bb
yotsuichemicals.co	0.97	yotsuichemicals.com
okisoukou.xx	0.96	okisokou.xx
taiyo-ge.yy.xx	0.96	taiyo-e.yy.xx

類似度が高い順に条件にかかった「嫌疑ドメイン」のメール群を「攻撃モデル」に合致するか確認した．確認方法は次の通りである．

・「嫌疑ドメイン」と「比較対象ドメイン」の両方から同じ相手に送信している．

・「嫌疑ドメイン」と「比較対象ドメイン」の両方の件名が類似している．RE:などや主たる単語が同じである．

「表 1 やり取りの例」に「合致」した例を示す(本表もマスク済みである)．まず件名中のキーワードとして「ガンバ」が共通している．そして受信者も同じであるが，送信者のドメインが 1 通目と類似しているが違っている．

日時	送信者	受信者	件名
6/4 11:39	Yamase @jjjjjj-system.co.jp	xxxxx.daijiro @abc.com	【着工前見積書】ガンバ
6/8 10:13	Yutani @jjjjjj-syste.co.jp	xxxxx.daijiro @abc.com	RE: ガンバレイアウト変更工事のご相談

表 1 やり取りの例

手順は説明のために前後させた．確認する前に「嫌疑ドメイン」を篩にかけた．Google 検索でアドレスが得られない，などレピュテーションとして不審であるかないかを切り分けた．

DIFFLIB による類似で即嫌疑とはしなかった．通常のメールの割合は 52%であった．

残りの 48%に対し「嫌疑ドメイン」と「比較対象ドメイン」を含む送受信が「攻撃モデル」に合致しているかを前述の方法で確認した．33%でそのような流れが観測された．全嫌疑ドメインの 16%(48%の 33%)が「攻撃モデル」に合致した(「図 3 観測結果の分類」でまとめた)．

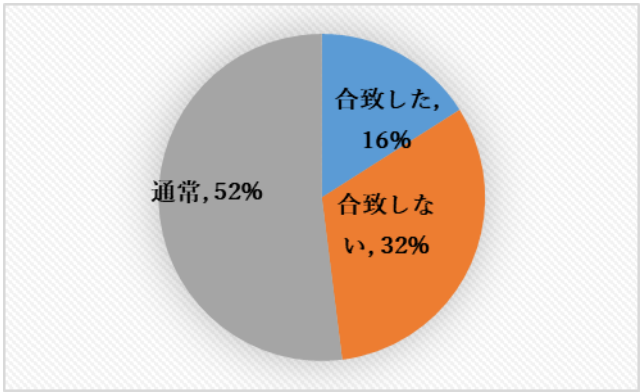

図 3 観測結果の分類

「合致した」16%は真に攻撃の可能性もあり件数が絞り込まれてきたのでこの段階で利用者にヒアリングを行った．75%が相手の送信元ドメインが TYPO(記述ミス)だった．残りの 25%は確認中である．

ⓒ2020 Information Processing Society of Japan

Vol.2020-CSEC-90 No.10
Vol.2020-SPT-38 No.10
2020/7/20

5. 現時点でのまとめと課題

本稿投稿時点での真の「詐欺メール」は検出できていない．しかしこの状況はそもそもの出現頻度は極稀れで数ヶ月で結果を求めるべきではなかった．従って引き続き観測は続ける．

やり取りが「合致」したほとんどが送信元の TYPO だったが，発生原因を究明していない．今後は自動排除も必要だろう．

また改善点も見えてきた．攻撃モデルは同じだか，絞り込みをアドレスのドメインだけではなく，件名に変えてみるなども思いついた．更に経験に頼るのだけではなく，機械学習など統計手法も試みたい．教師データがあまりに少ないので，まずはメールメタ情報の要素を絞るためのクラスター分けし不審なクラスターに分けられるか，など実施したい．

参考文献

1 Copyright 2001-2017, Python Software Foundation.
python3.5 ドキュメント » Python 標準ライブラリ » 6. テキスト処理サービス » 6.3. difflib − 差分の計算を助ける

ⓒ2020 Information Processing Society of Japan