Monthly Research

Fuzzy hashingの利用に関する検討及び評価

株式会社ＦＦＲＩ
http://www.ffri.jp

Ver2.00.01

Agenda

- 背景と目的
- Fuzzy hashingの概要
- 実験概要
- 実験結果
- 考察

背景と目的

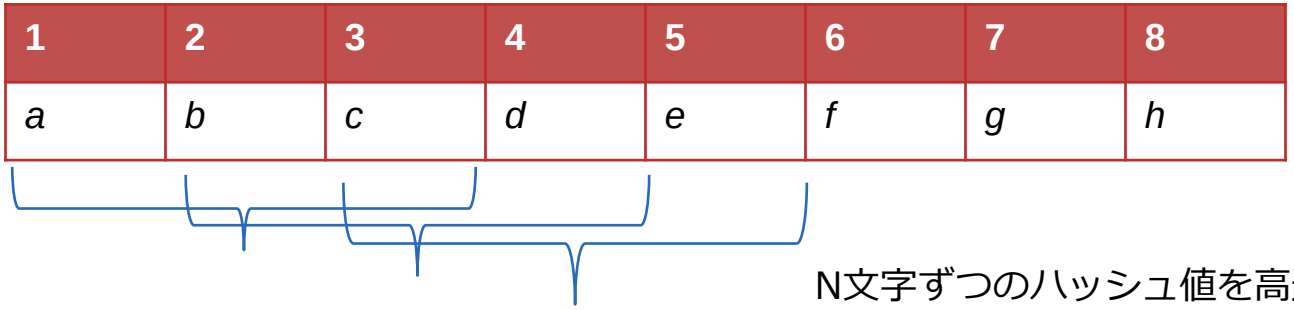
- 2006年にJesse KornblumによってFuzzy hashingが導入された
  - http://dfrws.org/2006/proceedings/12-Kornblum.pdf
- マルウェア解析においては、ssdeep等のfuzzy hashingが利用され始めている
- しかし、これらの効果的な利用法については現時点ではあまり検討されていない
- 本書では、Fuzzy hashingによる類似マルウェアの分類の有効性について評価を実施

Fuzzy hashingの概要(1/4)

- 一般的には、MD5, SHA1等の暗号学的ハッシュ関数が有名であり、これらは以下の性質を備える(http://ja.wikipedia.org/wiki/暗号学的ハッシュ関数)
  - 与えられたメッセージから対応するハッシュ値を容易に計算できる
  - ハッシュ値から元のメッセージを得ることが事実上不可能である
  - ハッシュ値を変えずにメッセージを改ざんすることが事実上不可能である
  - 同じハッシュ値を持つ2つのメッセージを求めることが事実上不可能である
- 暗号学的ハッシュ関数は上記の性質からファイルの同定によく利用されている
- 一方で1ビットでも変更された場合、ハッシュ値が全く別の値となるため「多少違うファイル」を特定する用途には向かない
- デジタルフォレンジックにおいてはこうしたニーズが存在し、これを解決するために開発されたのがFuzzy hashingである

Fuzzy hashingの概要(2/4)

- Fuzzy hashing = Context Triggered Piecewise Hashing(CTPH) = Piecewise hash + Rolling hash
- Piecewise Hashing
  - メッセージをN個に分割し、各ブロックのハッシュ値を計算
- Rolling Hash
  - メッセージ中のサブメッセージ(1文字目から3文字目、2文字目から4文字目等)のハッシュ値を高速に計算する手法
  - 一般的に計算の結果、求められたハッシュ値が同一であった場合、元のサブメッセージも同一である確率が高い



N文字ずつのハッシュ値を高速に計算(N=3)

Fuzzy hashingの概要(3/4)

- Fuzzy hashing(CTPH)
  - Rolling hashの結果が特定の値になった場合(トリガー)、起点からそこまでのメッセージをブロックと各ブロックのハッシュ値（の一部）を連結見做して暗号学的なハッシュ値を求め、

1	2	3	4	5	6	7	8
a	a	a	b	b	b	c	c

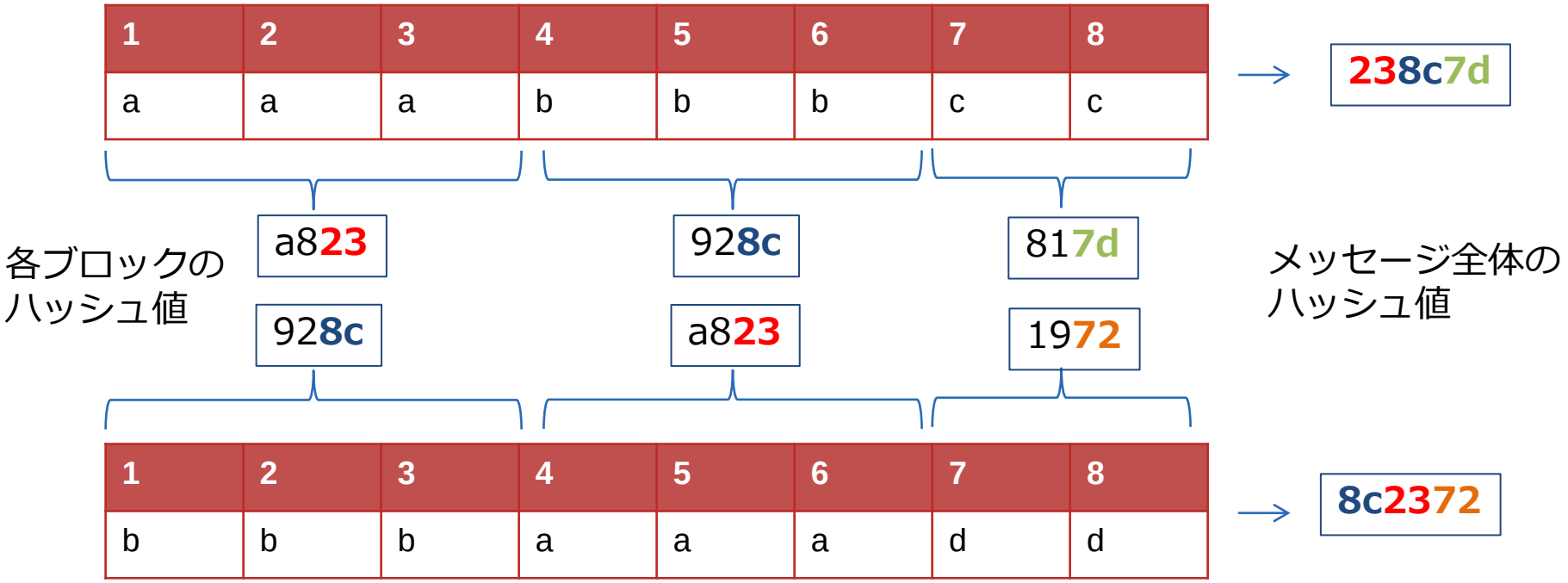
①Rolling Hashを計算、結果が特定の値に（トリガー）

②このブロックの暗号学的ハッシュ値を計算

③ポジション3からRolling hashを再計算（次ブロックの特定へ）

Fuzzy hashingの概要(4/4)

- Fuzzy hashing(CTPH)
  - 異なるメッセージ同士でも、確率的に部分的に同一のブロック毎のハッシュ値が計算される
  - ハッシュ値を比較することで部分的な一致を特定することが可能



実験概要(1/2)

- Fuzzy hashingの用途として主に下記の2つが提唱されている
  - 類似ファイルの特定
  - 部分的なデータのマッチング

- 今回は「マルウェア」を対象に下記の利用方法について検討する
  - 類似マルウェアの特定
    - Fuzzy hashingによる類似度に基づいてMD5等のハッシュ値が異なる“ほぼ同じ”マルウェアを特定する

- 上記について実際どの程度の効果があるのか、適用時にどういった点に注意すべきかを明らかにする

実験概要(2/2)

- 準備
  - 独自に収集したマルウェアを2,036件用意(MD5ハッシュ値上ユニーク)
  - 各ファイルのssdeepによるハッシュ値及びファイル間の類似度を算出
    - 2036件から2件を取り出す組み合わせ：2,071,630通り

- 類似マルウェアの特定
  - 上記のうち類似度が50%～100以上のペアをそれぞれ抽出
  - 各閾値においてペア双方の検出名がどの程度一致するかを確認、評価

検体同士の類似度(%)

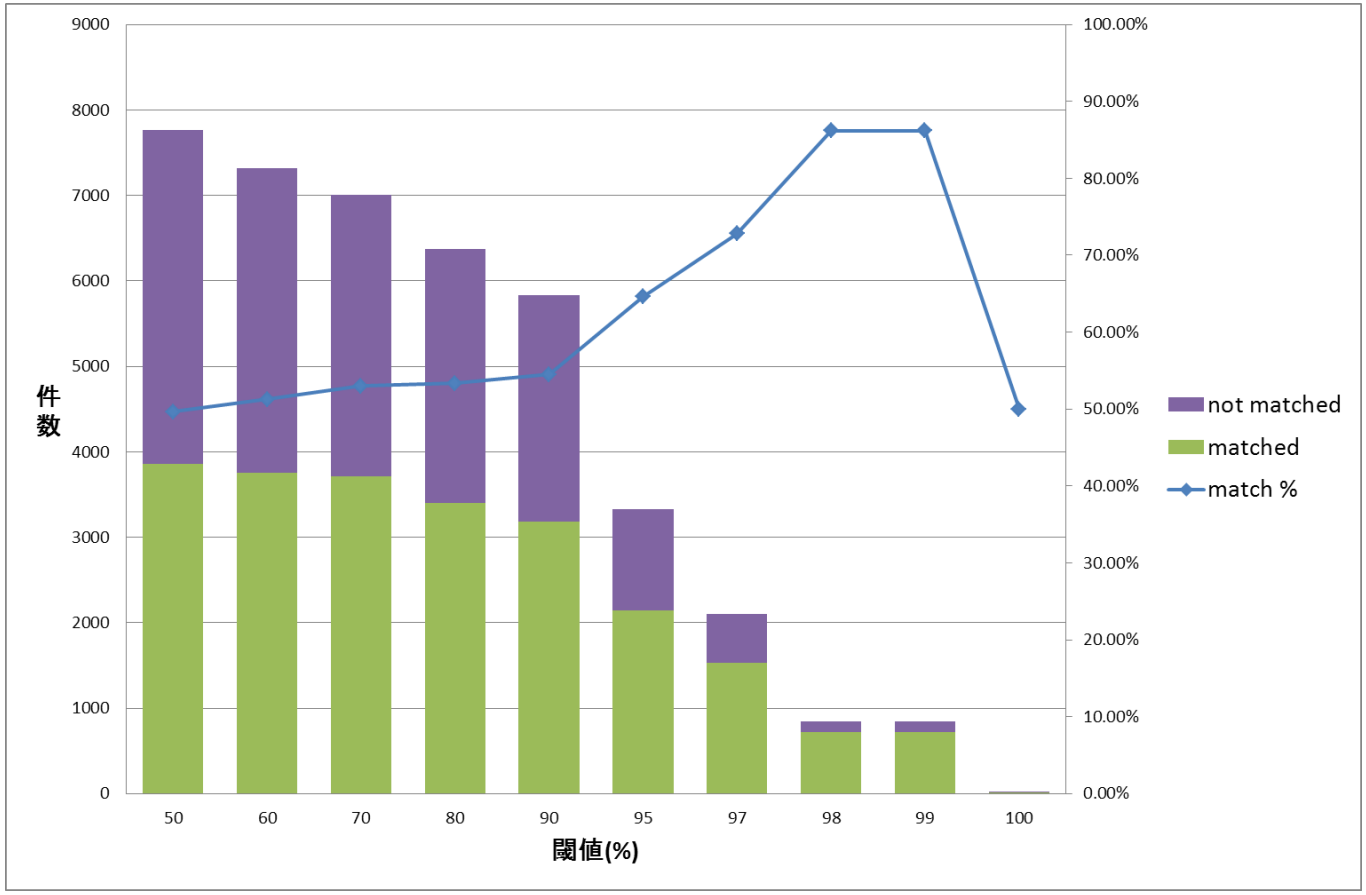
	A	B	C	…
A		90	82	54
B			76	62
C				46
…				

80%以上
・A-B
・A-C

80%以上
・(A)Trojan.XYZ – (B)Trojan.XYX ← **Matched**
・(A)Trojan.XYZ – (C)WORM.DEA ← **Not matched**

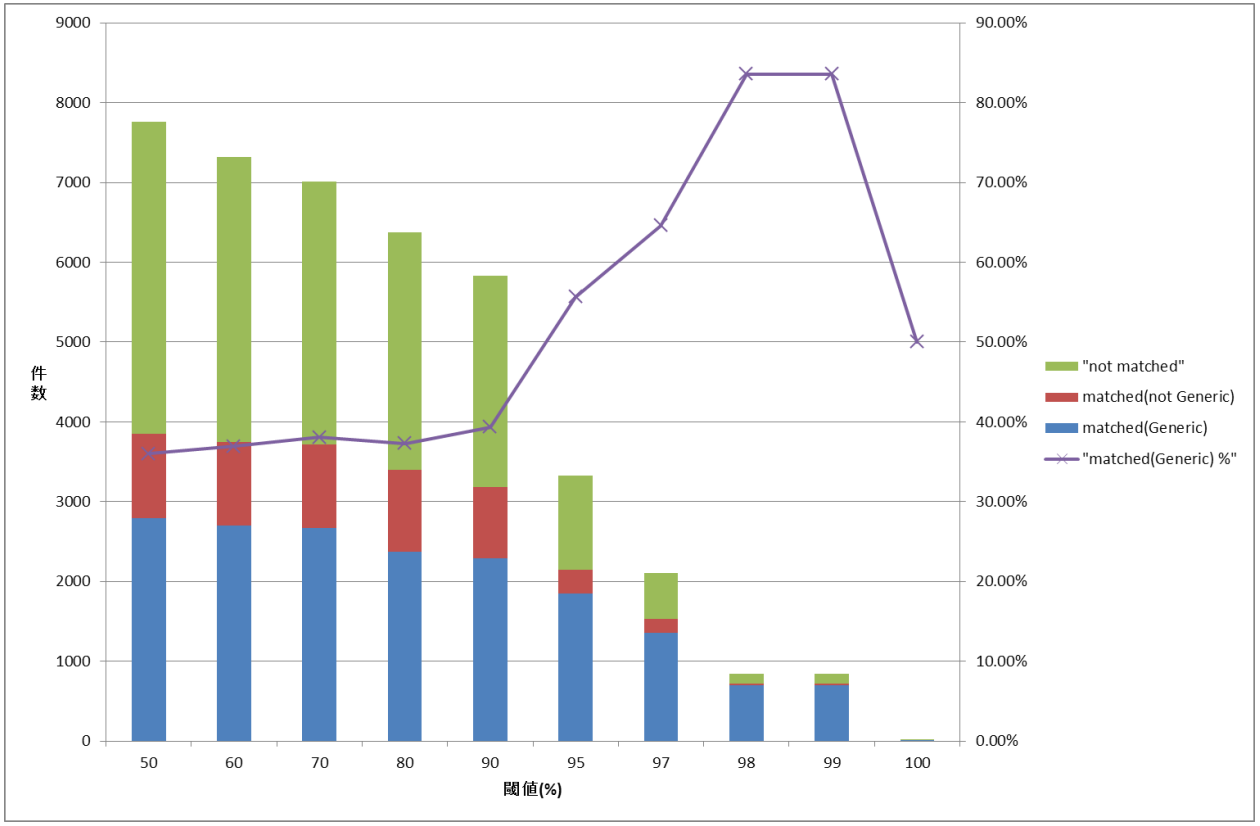
実験結果

- 閾値が高い程、検出名の一致率も上昇(100%時は件数自体が僅少のため例外)
  - 90%迄は50～60%の一致率で安定、90%以降は徐々に上昇



検出名に「Generic」を含む割合

- 前出グラフの”matched”について検出名に「Generic」を含むもの・含まないもので分離
- ジェネリック検知の上昇率は、前出の検体名の一致率と同じ傾向
  → 閾値を挙げるほどジェネリック検知によるマルウェアが増加



考察

- 今回検出名に使用したアンチウイルスベンダーがジェネリック検知にFuzzy hashingを利用しているか否かで結果の解釈が異なる
  - 利用している場合
    - 当然の結果
  - 利用していない場合
    - Fuzzy hashingにより当該ベンダーのジェネリック検知と同等の結果が得ることができるのではないか

- Fuzzy hashingをジェネリック検知に利用する場合、既知のハッシュ値に対して90%超の類似度を閾値するのが望ましいにより考えられる

Contact Information

- E-Mail: research-feedback@ffri.jp
- twitter: @FFRI_Research