Monthly Research

Consideration and evaluation of using fuzzy hashing

FFRI, Inc

http://www.ffri.jp

Ver2.00.01

Agenda

- Background and purpose
- Basis of fuzzy hashing
- An experiment
- The result
- Consideration

Background and purpose

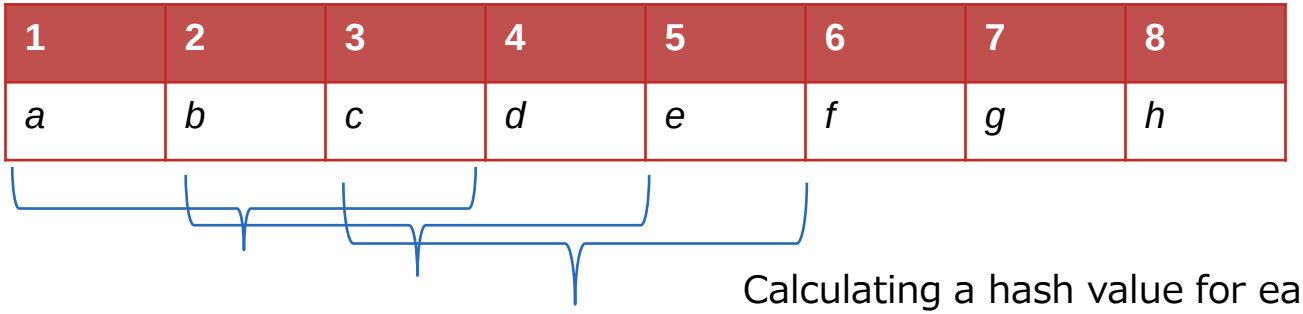
- ‘fuzzy hashing’ was introduced in 2006 by Jesse Kornblum
  - http://dfrws.org/2006/proceedings/12-Kornblum.pdf
- In malware analysis fuzzy hashing algorithms such as ssdeep are being introduced in recent years
- (IMHO) However, we don’t consider the effective usage of them enough
- In this slides, we evaluate an effectiveness of classification of malware similarity by fuzzy hashing

Basis of Fuzzy hashing(1/4)

- In general, cryptographic hashing like MD5 is popular and it has the attributes as follows: (cf. http://en.wikipedia.org/wiki/Cryptographic_hash_function)
  - it is easy to compute the hash value for any given message
  - it is infeasible to generate a message that has a given hash
  - it is infeasible to modify a message without changing the hash
  - it is infeasible to find two different messages with the same hash
- Cryptographic hashing is often used for identify the same files
- On the other hand, it is unsuitable to identify similar files because digests are completely different even if 1 bit is altered in the other file
- In DFIR, this need exists and fuzzy hashing was developed to solve this problem

Basis of Fuzzy hashing(2/4)

- Fuzzy hashing = Context Triggered Piecewise Hashing(CTPH) = Piecewise hashing + Rolling hashing
- Piecewise Hashing
  - Dividing message into N-block and calculating hash value of each blocks
- Rolling Hash
  - A method to calculate hash value of sub-message(position 1-3, 2-4…) fast
  - In general, if calculated values are the same, it is a high probability that original messages are identical



Calculating a hash value for each N chars(N=3)

Basis of Fuzzy hashing(3/4)

- Fuzzy hashing(CTPH)
  - When rolling hash generates a specific value at any position, it calculates cryptographic hash value of the partial message from the beginning to the position
  - Generate a hash value by concatenating all (partial) hashes

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

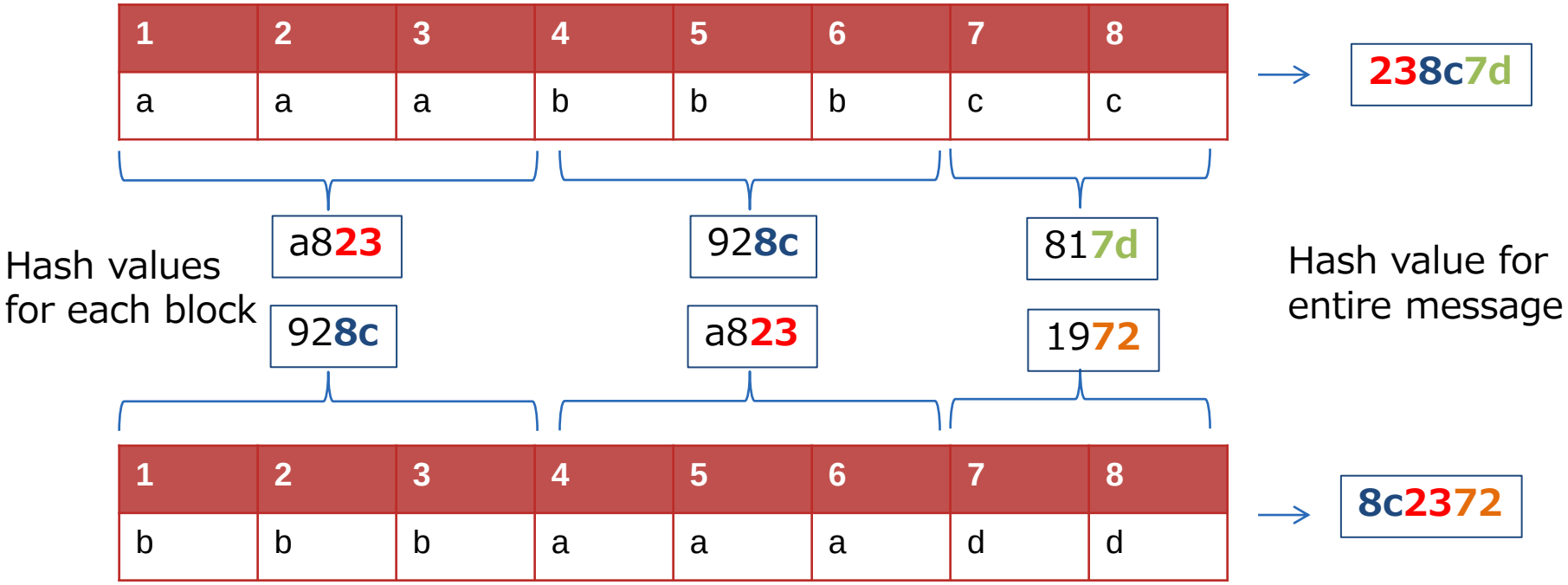
①calculating rolling hash, and the result is the specific value (trigger)

②calculating a hash value of this block by cryptographic hash

③calculating rolling hash value from position 4(trying to determine a next block)

Basis of Fuzzy hashing(4/4)

- Fuzzy hashing(CTPH)
  - With almost identical messages, it would calculate a hash value of identical partial message stochastically
  - We can identify partial matches between similar messages



An experiment(1/2)

- In general, usage of fuzzy hashing is proposed as follows:
  - Determining similar files(i.e. almost identical but MD5s aren't matched)
  - Matching partial data in files

- This time, we evaluate determining similar malware by fuzzy hashing

- We make it clear "how effective it is in actually" and "what we should consider if we applying it" for the above

An experiment(2/2)

- Preparation
  - Preparing 2,036 unique malware files in MD5s collected by ourselves
  - Calculating(fuzzy) hash values by ssdeep and similarity of all of each files
    - nCr: $_{2,036}C_2$ = 2,071,630 combinations
- Determining similar malware
  - Extracting all the pairs whose similarities are 50%-100%
  - Determining if the detection name of files in a pair is matched for each similarity threshold

similarity of each malware files(%)

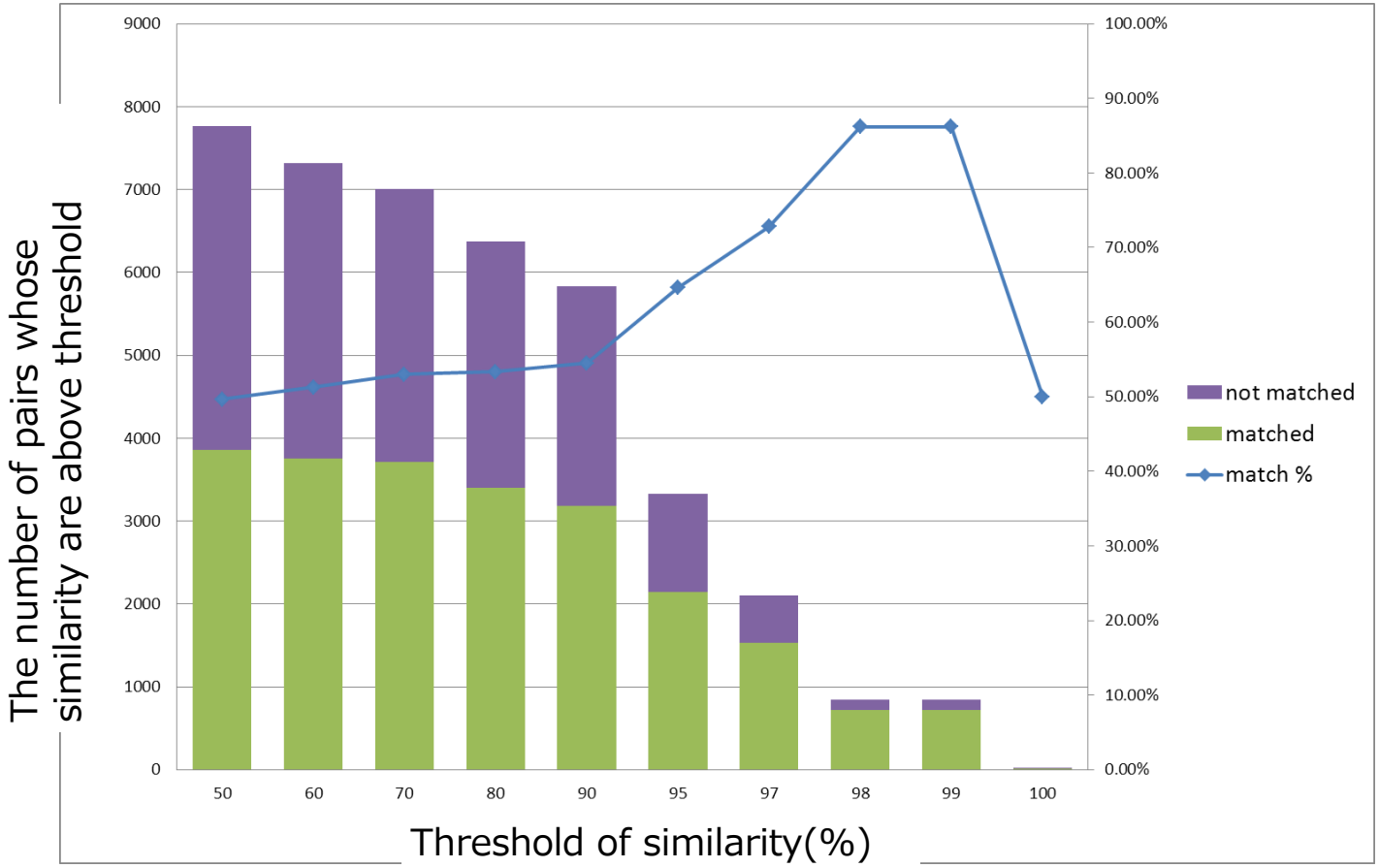
	A	B	C	…
A		90	82	54
B			76	62
C				46
…				

80%+
・A-B
・A-C

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

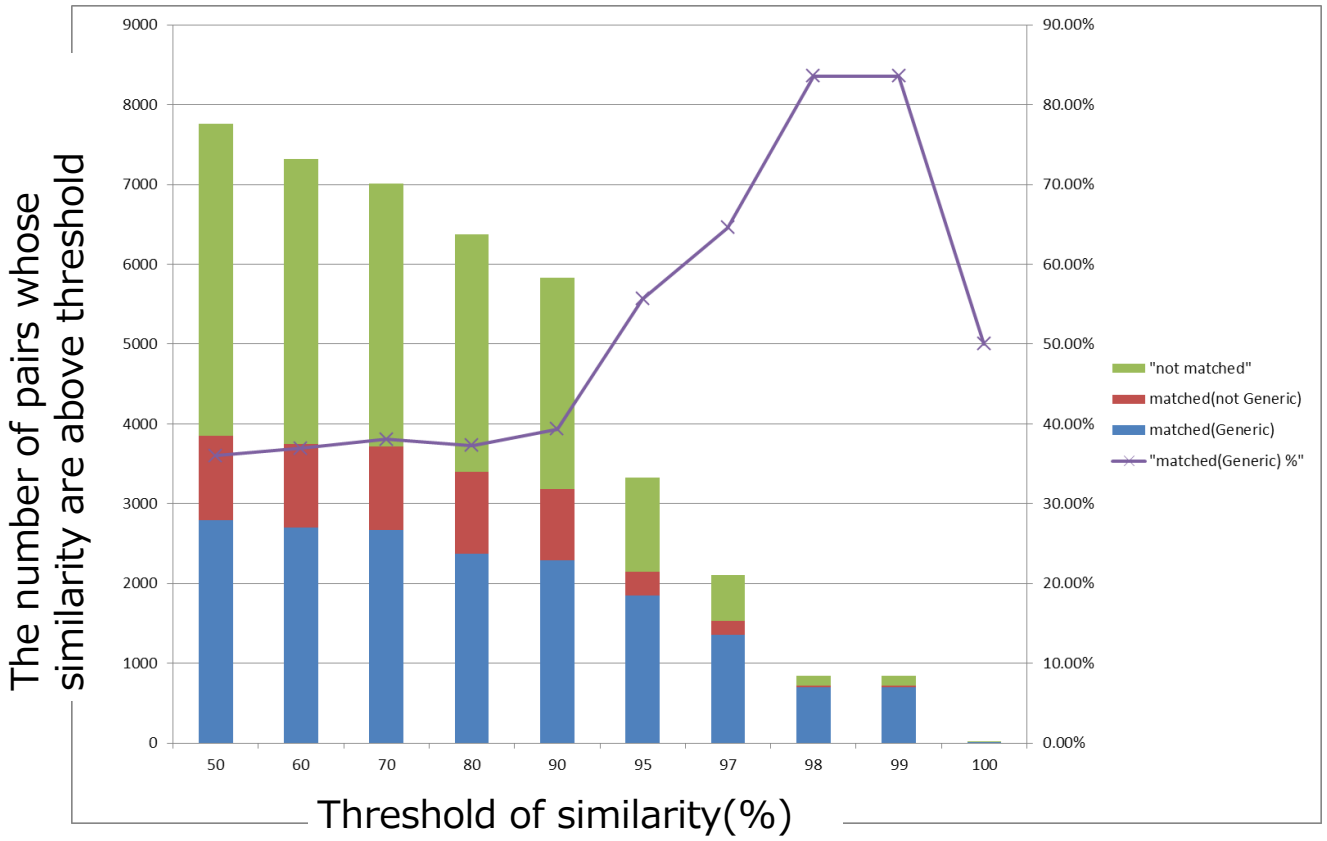
The result

- The higher the threshold is, the higher matching rate of detection name we get
  - Up to the threshold of 90% it keeps around 50-60% matching rate



The rate detected by the name “Generic”?

- Dividing “matched” pairs into a group who has “generic” in its name and the others
- “matched(Generic)%” shows the same trend with the matching rate above
  -> The higher the threshold is, the more malware are detected as “generic”



Consideration

- A meaning of the result depends on if the AVV uses fuzzy hashing for generic detection
  - If they use fuzzy hashing for generic detection
    - The result is natural
  - If not
    - By using fuzzy hashing, we may obtain a similar result to the generic detection

- If we use fuzzy hashing for generic detection, 90%+ similarity might be required with known malware (fuzzy) hash values

Contact Information

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