TREND MICRO

Securing Your Web World

File-Fraction Reputation Based On Digest of High Granularity

Yixian Ethan Chen
TrendMicro

- Keywords:

Reputation

Digest

Reputation

- Reputation based approaches in anti-malware:
  - focus on **characteristics**
  - usually **not depend on content**

Such as

- prevalence
- download source
- age

Reputation

In the contrast:

- Content based technologies:
  - use “fingerprint” or “signatures”

Reputation

Why do people use reputation-based approach?

Reputation

- To solve problems like:
  - sample which occurs **only once**
  - sample which **only available at certain endpoint at certain time**
  - ever growing **volume** of malware samples

Reputation

- Pros
  - no need to get sample in advance
  - less chance to be fooled by content manipulation

Reputation

- Cons
  - know less about the malware
  - limited choice of action

Block download	V
Policy control / auditing	V
Clean up system	X
Disinfect files	X

- The other keyword….

Digest

Digest

- compact form to present the data

- e.g.
  - Cryptographic hash: md5, sha1 → **Identity**
  - Nilsimsa, ssdeep → **Similarity**

Digest

- Problems:
  - how to index fuzzy hash?
    - high dimension
    - edit distance
  - global similarity / local similarity

Digest

- Given
  - a collection of samples
  - a sample (which want to find similar sample)

Cost quite some computing to search in a collection of tens of millions of files!

- … we proposed this idea

File-Fraction Reputation

File-Fraction Reputation

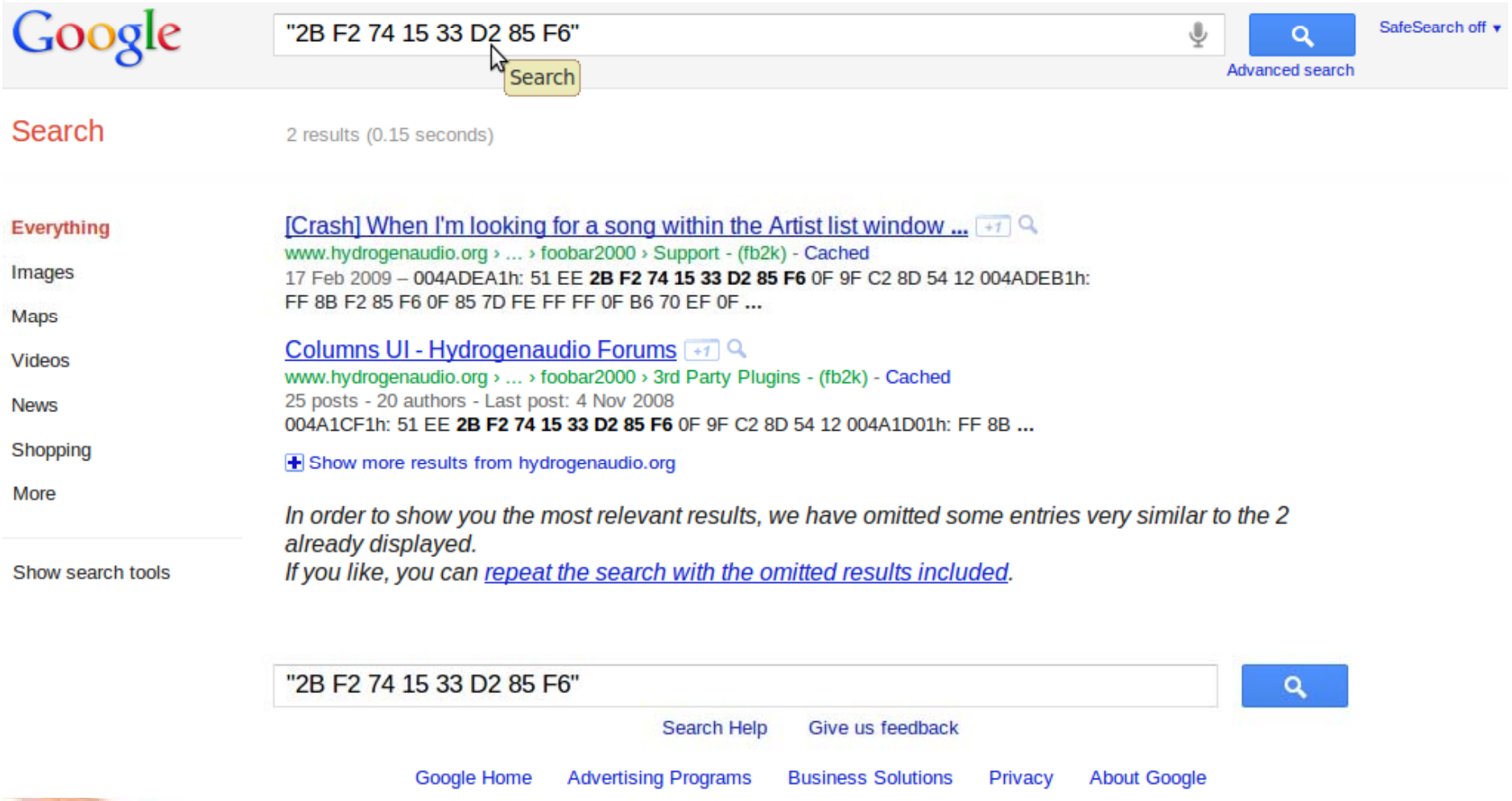
- An attempt to connect
  - reputation based
  - content based
- Efficient way to
  - find N-nearest
  - discover cluster

File-Fraction Reputation

- The basic entity to assign reputation ?
  - a file
  - a set of files
  - **parts of a file**

- Parts of a file
  - code and data
  - payload
  - packer stub

File-Fraction Reputation



TREND MICRO

File-Fraction Reputation

- We want to ask and try to answer,

Given a file,
how to discover if it shares partial commonness with files already seen before?

File-Fraction

File-Fraction

- content of a malware can be regarded as series of bytes
  - raw file
  - memory dump
- let’s call them "string buffers"
- "string buffer" is factorized into parts, called "fractions"

File-Fraction

- File-Fraction algorithm is define with
  - A rolling hash
  - A hit condition

  and
  - A figerprinting hash function

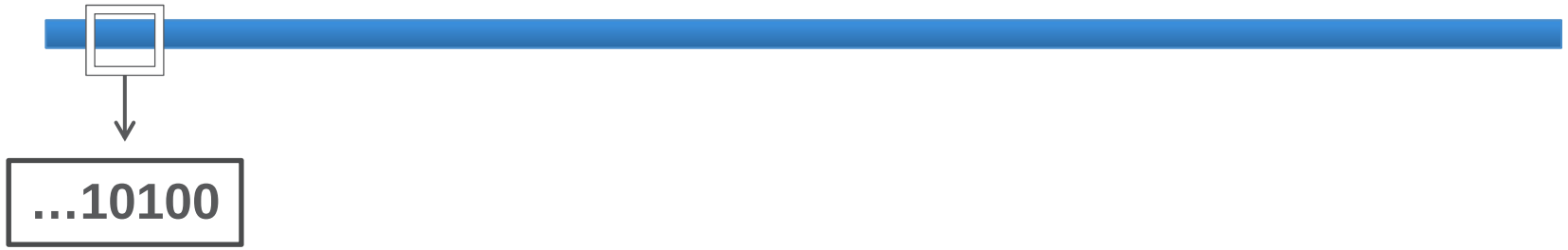
Decide how to factorize a string buffer into pieces

Define identity or equivalence class of each fraction

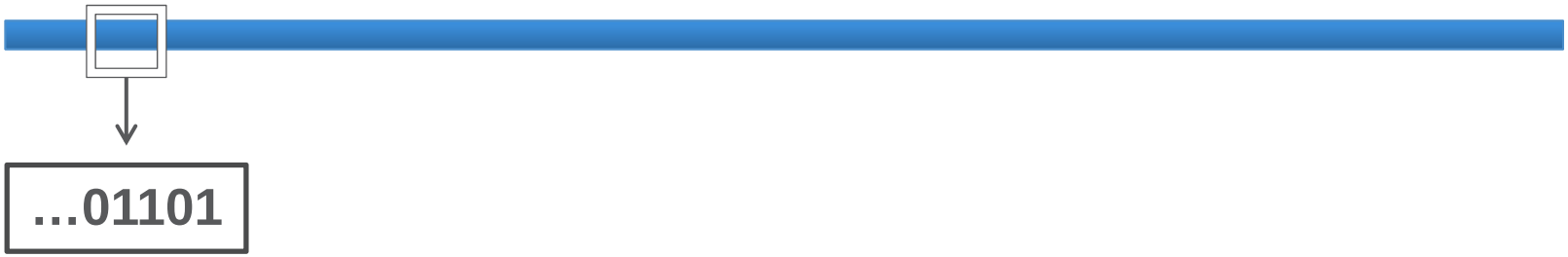
File-Fraction

```
file_fraction_digest (s, r, h, C)
initialize the rolling hash r and the fingerprinting
hash h
initialize output as an empty array
for addr from 0 to length(s)
     r.update( s[addr] )
     if r.hashvalue() in condition C
          output.push( h.hashvalue() )
          h.reset()
     h.update( s[addr] )to
return output
```

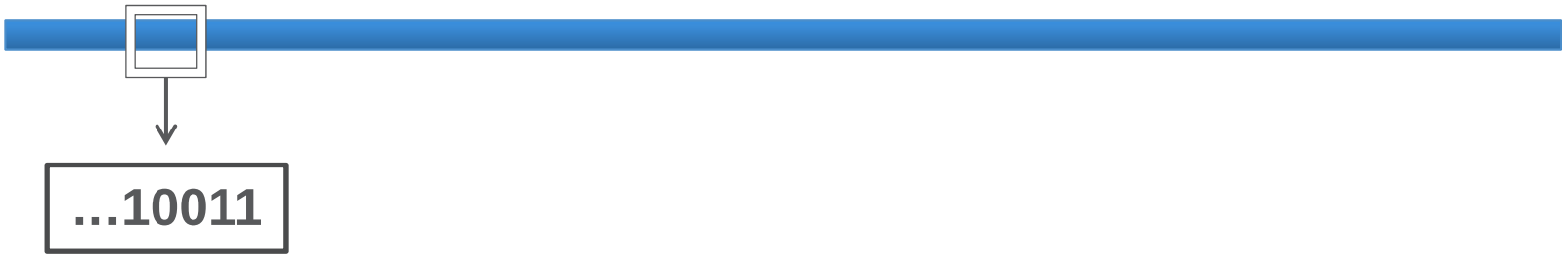
File-Fraction



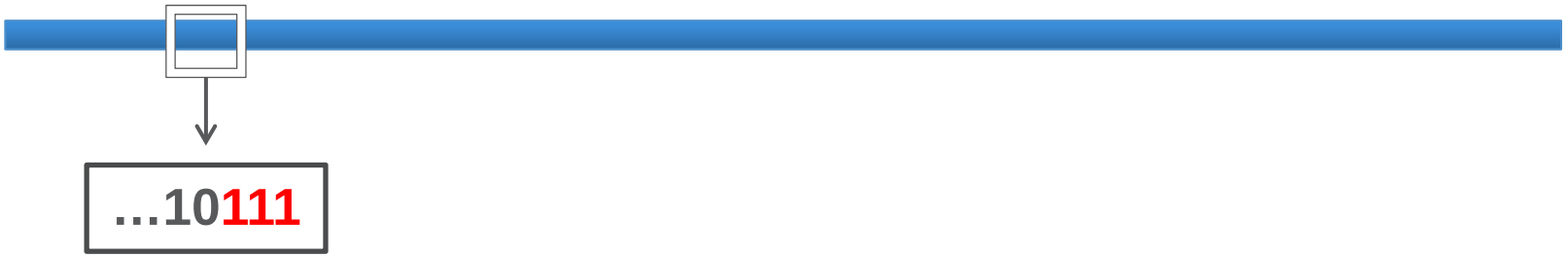
File-Fraction



File-Fraction

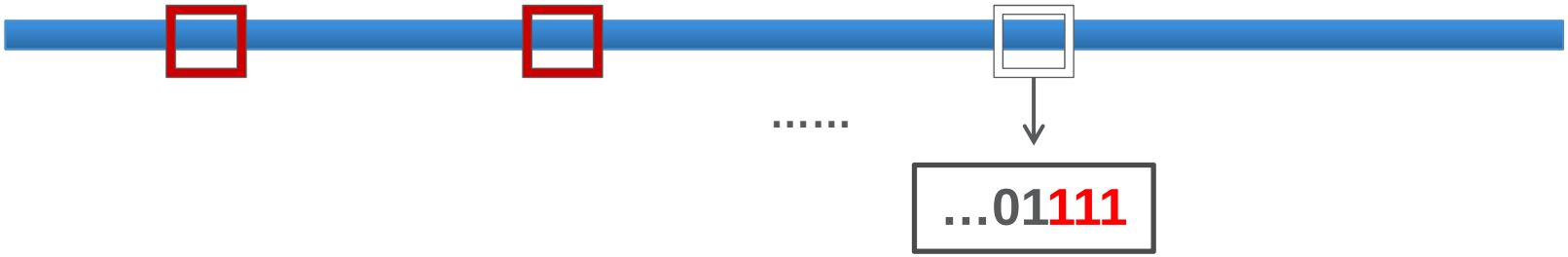


File-Fraction

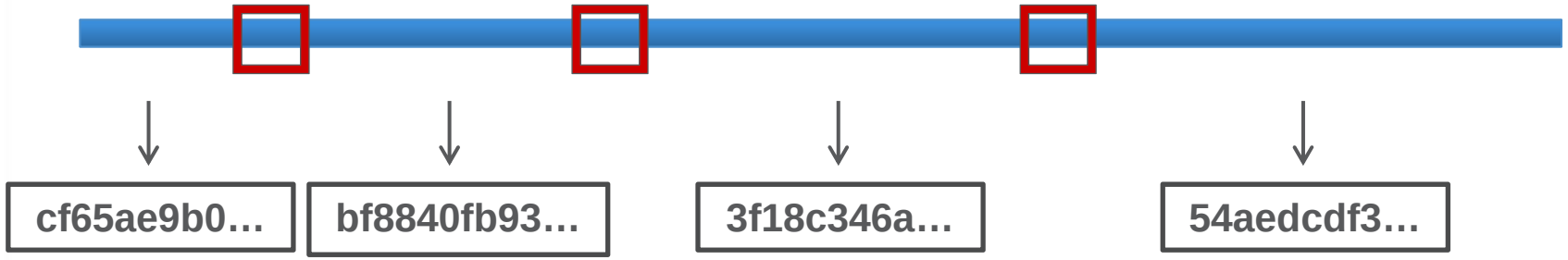


File-Fraction

File-Fraction

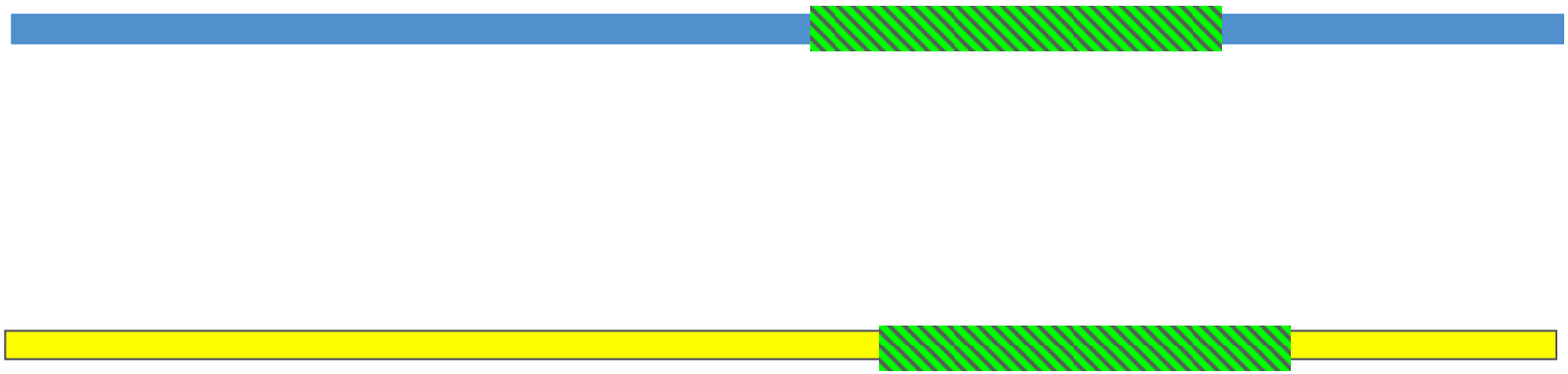


File-Fraction



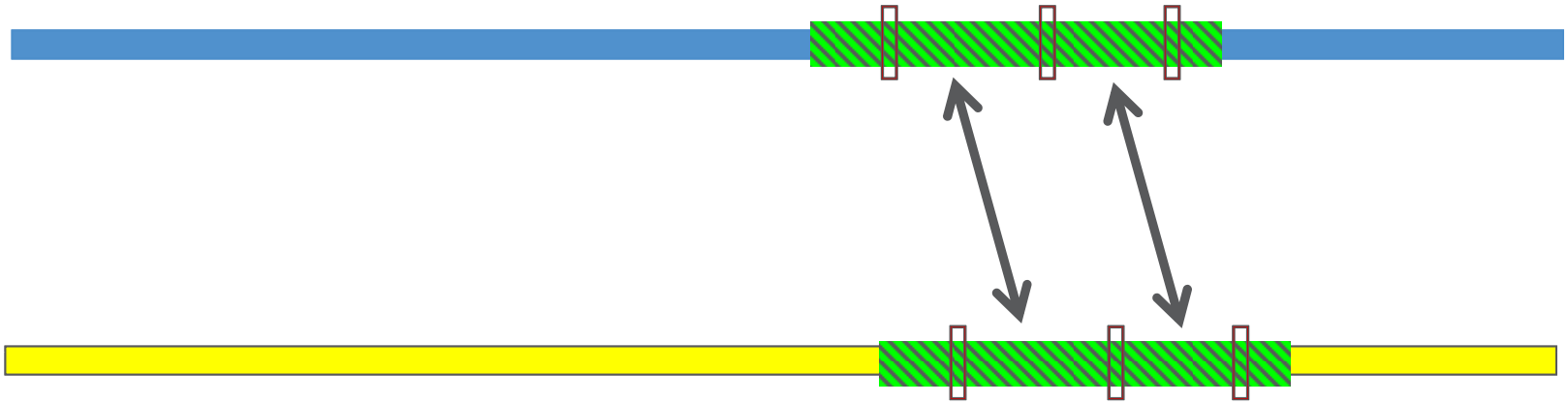
File-Fraction

- rolling hash
  - align the common parts



File-Fraction

- rolling hash
  - align the common parts



File-Fraction

- hit condition
  - control the granularity

- if hit condition is …10111 …00111 …01111
  - in average every 8 bytes will have a cut

- in practice we often choose the average cut size as 512 ~ 8192

File-Fraction

- fingerprinting hash function
  - as identifier number

  or
  - define equivalence class
- use crypto hash
- use fuzzy hash

File-Fraction Reputation

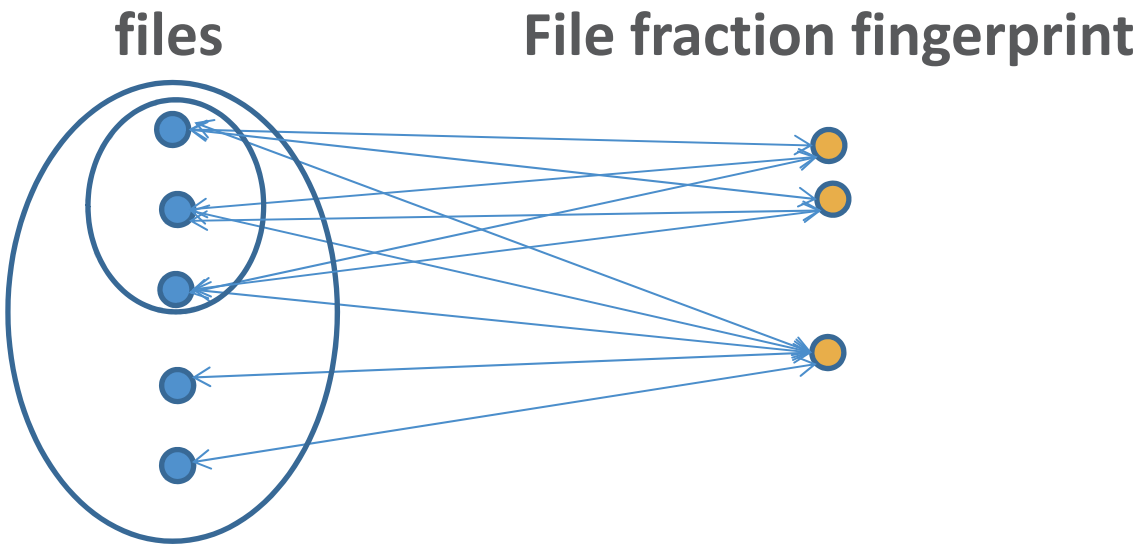
File-Fraction Reputation

- A common “file-fraction” in files may be part of:
  - Compiler stub
  - Packer stub
  - Static linked library

  or
  - Malicious code
  - Payload (to drop or to inject to other process)
  - Resources related to malicious code (graphics, text strings…)

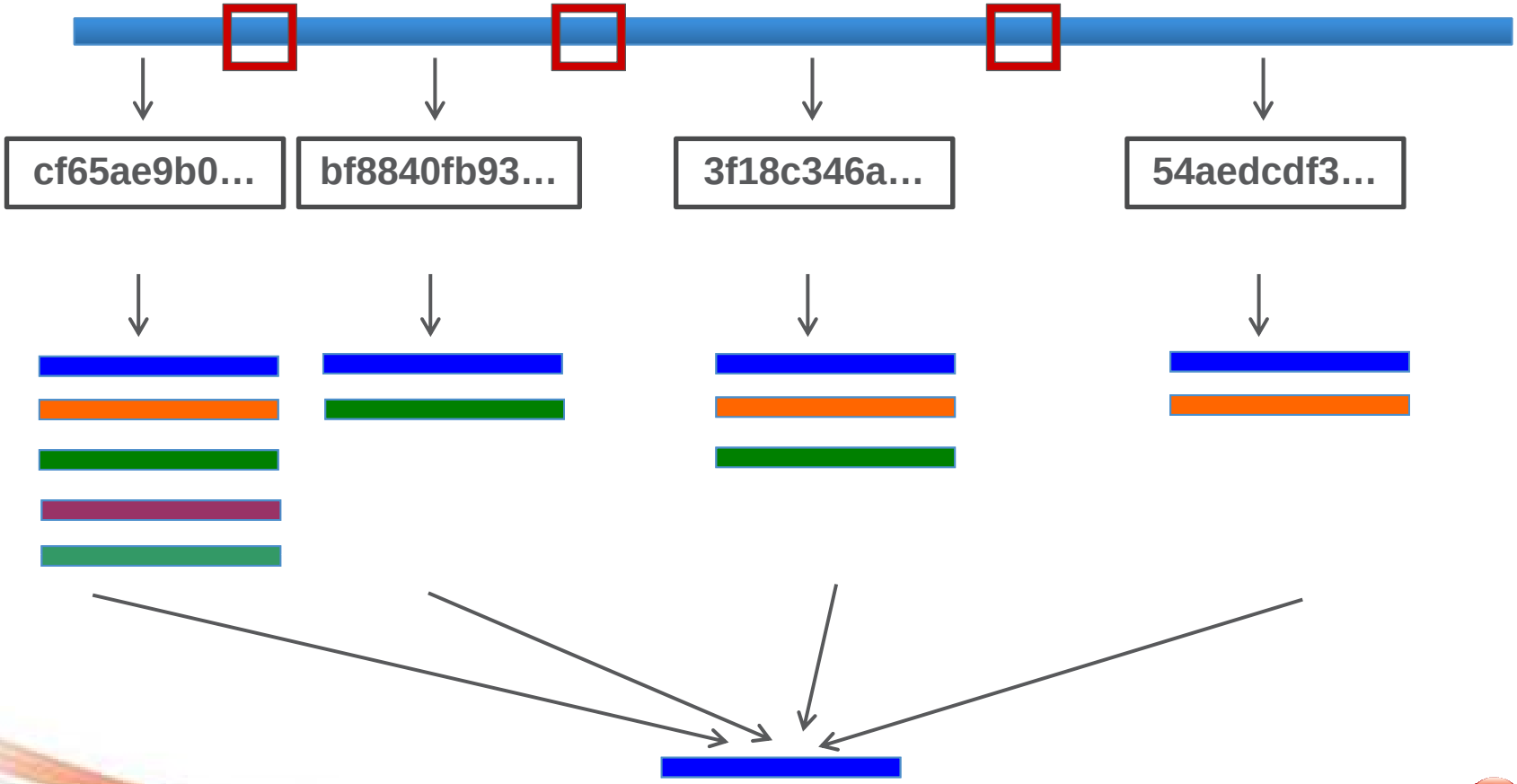
File-Fraction Reputation

- cluster analysis



File-Fraction Reputation

- lookup similar files



File-Fraction Reputation

- Pros compare to file-reputation
  - self-explanatory
  - better accuracy

File-Fraction Reputation

- Pros compare to other digest
  - query and retrieval
  - N-nearest problem
  - cluster discovery
  - sensitivity
  - local similarity

File-Fraction Reputation

- Cons
  - The size of the digest can be 1/100 ~ 1/1000 of the original string buffer.

    Traditional digest is constant size.

File-Fraction Reputation

- Cons
  - Traditional file reputation have advantage on a whole new malware;

    File-fraction reputation have advantage on a malware which has some relation to a old one.