Monthly Research

Behavioral-based malware clustering

FFRI, Inc

http://www.ffri.jp

Ver2.00.01

Agenda

1. Background and purpose
2. Overview of clustering
3. An experiment
4. The Result
5. Considerations
6. Conclusions
7. Future Works

1.Background and purpose

- In recent year the number of malware has been extremely increased
- It is hardly possible to analyze all malware manually
  - **req-1) Need to determine malware to analyze preferentially (novel unique malware, etc.)**
  - **req-2) Need to make analysis more efficiently (referring similar malware information, code diffing, etc.)**
- In development of malware detection engines, it is difficult to use all the collected malware for prototyping and testing
  - **req-3) Need to group malware and select representative samples**
- Evaluating behavioral-based clustering as a sort of methods to solve the problem

2.Overview of clustering

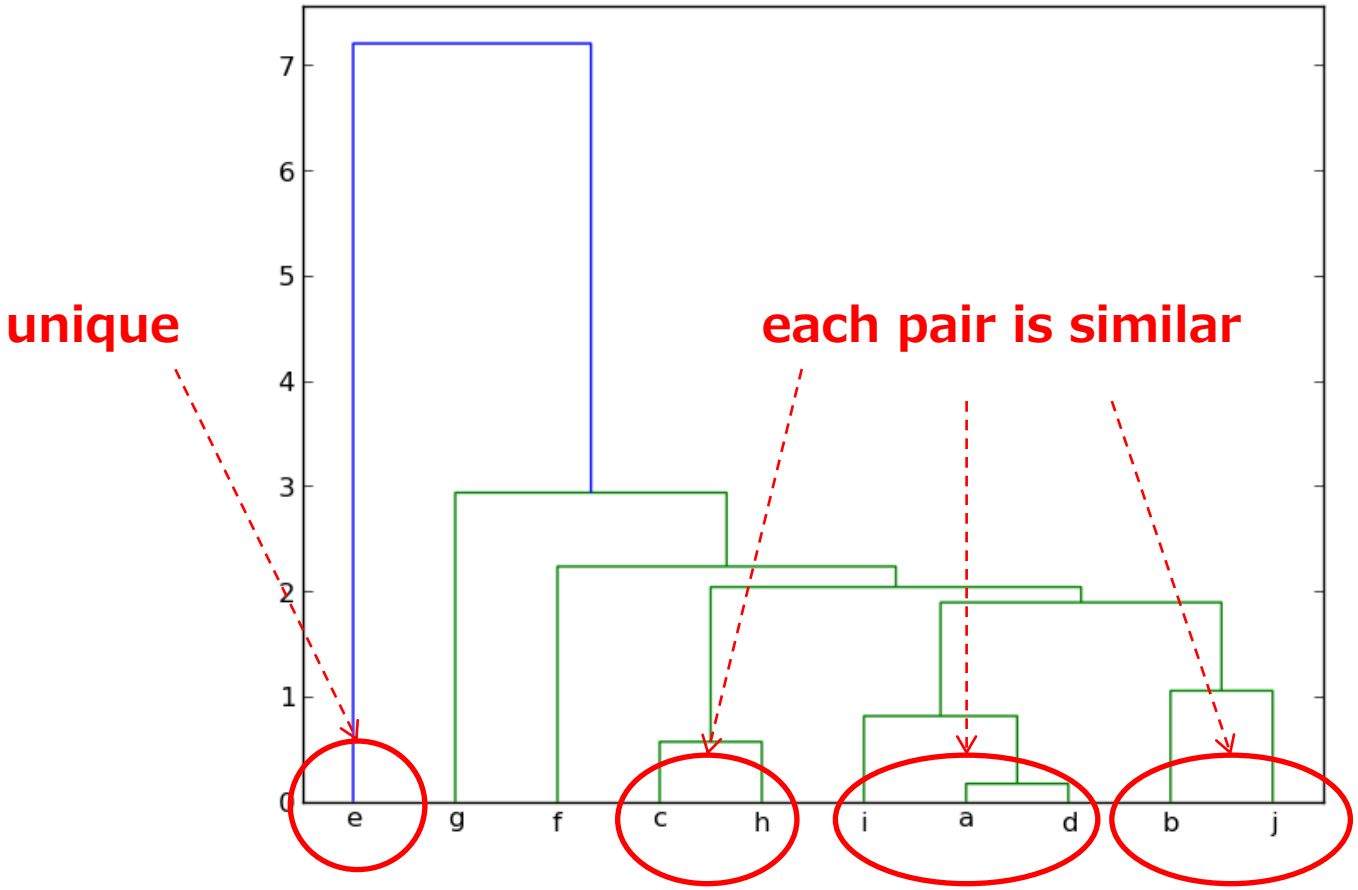
- Dividing data into some clusters (groups) based on “features”
  - “features” must be selected manually

- Mainly there 2 types of clusterings
  - Hierarchical clustering
    - Considering each data as a cluster and merging them as a tree based on similarities or distances
    - The result is shown as dendrogram(tree)
    - eg. single linkage, complete linkage, average linkage, ward, etc.

  - Non hierarchical clustering
    - Dividing data into some groups based on its similarity
    - Hard clustering(data belong to a cluster) and Soft clustering (data might belong to some clusters)
    - eg. k-means, mixture model, NMF

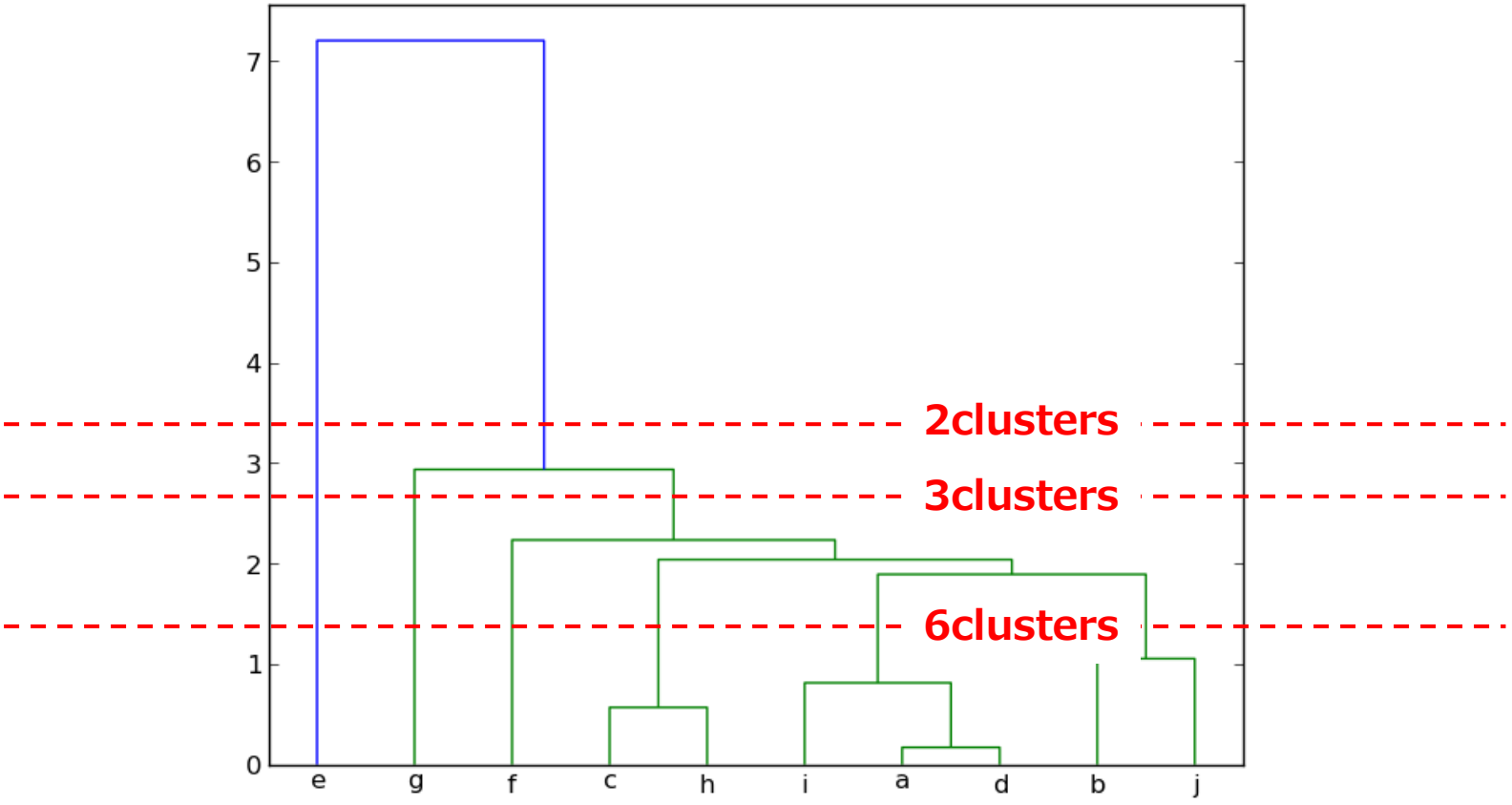
2.Overview of clustering / Hierarchical clustering

An example of dendrogram (alphabets on x-axis is corresponding to each datum)



2. Overview of clustering / Hierarchical clustering

Capable of considering a result as N clusters depending on each depth



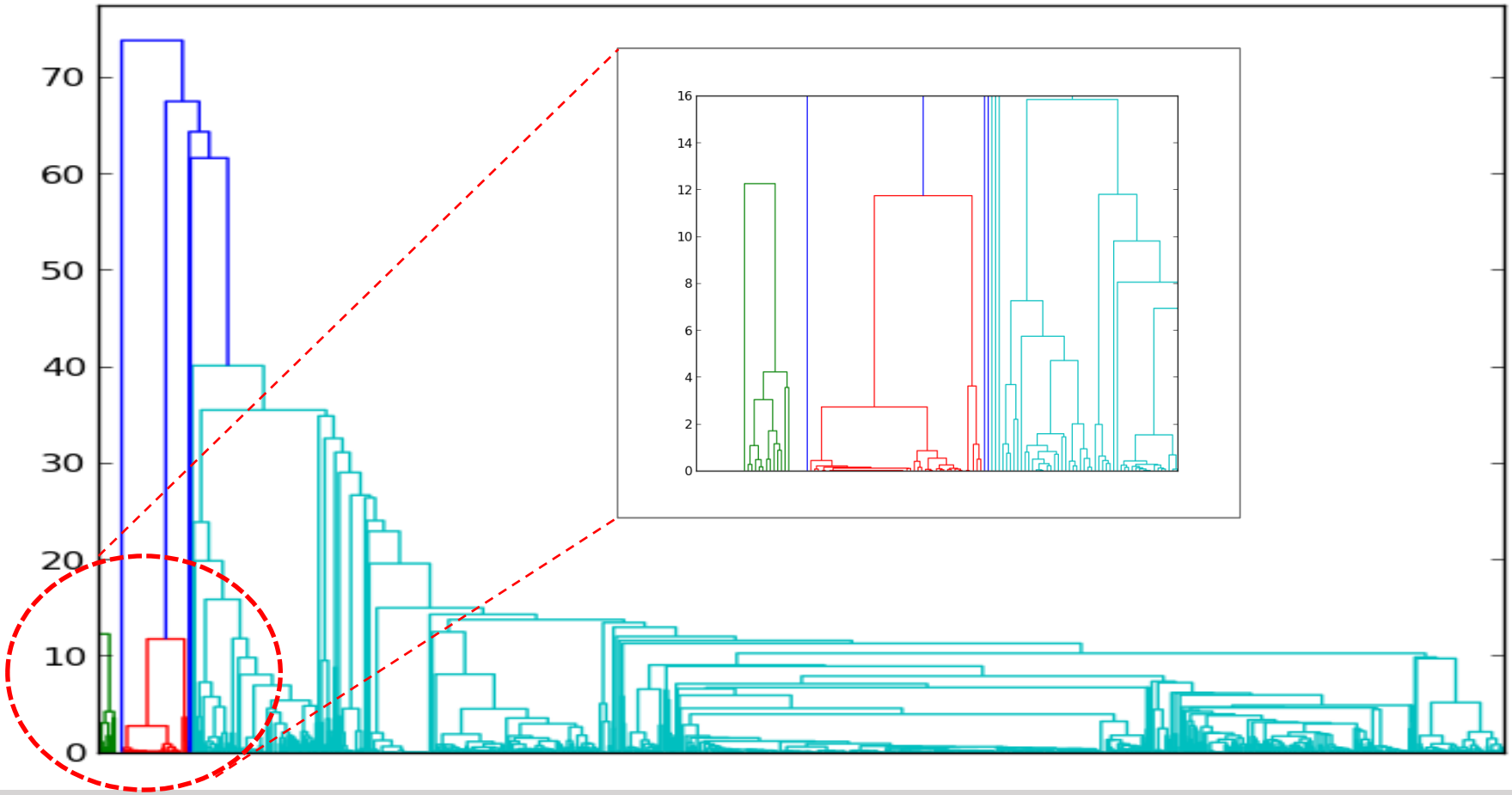
3.An experiment

- Applying ward's method(hierarchical clustering)
  - Most non hierarchical clusterings have to be specified a number of clusters
    - Determining the best cluster size is also problem
  - Using 3-gram of API-calls(no args) with weighting by tf-idf as features
- Datasets
  - Sampling 1,000 malware randomly from our collections
    - Confirming the detection rate using VirusTotal based on hash values
    - Most of vendors detect around 80% of them
- Software
  - Extracting each malware's API-calls using Cuckoo Sandbox 0.6
  - Using Scipy(and Matplotlib) for clustering

tf-idf: http://ja.wikipedia.org/wiki/Tf-idf
Scipy: http://www.scipy.org/

4.The result

- Mainly it was divided into 3 clusters (green, red, light blue)



5.Considerations

a. Did the clustering work well?
(clustering data based on functions and behaviors)

b. Is it useful to determine novel unique malware? (req-1)

c. Does it boost manual malware analysis? (req-2)

d. Is it helpful to sample worthful data? (req-3)

5.Considerations / a. Did the clustering work well?

- Selecting 3 pairs which are similar in the deepest level of clustering

- Comparing both malware's functions and behaviors for each pairs
  - P-1(MD5 and detection name)
    - aac95e967b1ce621bd2b1a5854d0294d (HEUR:Trojan.Win32.Generic)
    - 69fcc9c0dca876307d97a64683936bad (Unknown)
  - P-2
    - 5dfca9602289f20f13902c4ed3710fb2 (HEUR:Trojan.Win32.Generic)
    - 90c4af98638d7d9418f2e29f55ec6c9f (HEUR:Trojan.Win32.Generic)
  - P-3
    - 9f267ae8fb419f2071795803216a3455 (Trojan.Win32.Jorik.Buterat.nwr)
    - dadcb4ab9827f66ba5bd350d78b902cc (Backdoor.Win32.Buterat.zqy)

P-1

- Result
  - Both malware might be belong to the same family, or they might be generated by the same tool with different configurations
- Common points
  - Generating a 23148 bytes data file with same MD5 under C:¥Windows¥Registration
  - Accessed registry keys and created mutexes are identical
  - Encoding method for accessing file is the same
  - Containing ASCII strings in PE are mostly common
  - Registering itself to the same 2 ASPEs(Auto-Start Extensibility Point)
- Difference
  - A dropped executable's MD5 hash are different
  - Detection statuses are much different('Unknown' is mostly undetectable)

P-1 / Accessed registry keys (completely matched)


```
 "keys": [
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥COM3",
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Classes",
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Classes¥¥CLSID",
            "CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}",
            "CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥TreatAs",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥InprocServer32",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥InprocServerX86",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥LocalServer32",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥InprocHandler32",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥InprocHandlerX86",
            "¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}¥¥LocalServer",
            "HKEY_CLASSES_ROOT¥¥CLSID¥¥{304CE942-6E39-40D8-943A-B913C40C9CD4}",
            "HKEY_CLASSES_ROOT¥¥CLSID¥¥{304CE942-6E39-40D8-943A-
B913C40C9CD4}¥¥TreatAs",
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥Rpc¥¥SecurityService",

"HKEY_LOCAL_MACHINE¥¥SYSTEM¥¥CurrentControlSet¥¥Services¥¥SharedAccess¥¥Parameters¥¥
FirewallPolicy¥¥StandardProfile",
            "HKEY_LOCAL_MACHINE¥¥software¥¥microsoft¥¥windows nt¥¥currentversion¥¥winlogon",
            "HKEY_CURRENT_USER¥¥software¥¥microsoft¥¥windows¥¥currentversion¥¥run"
        ],
```

P-2

- Result
  - Both malware might be belong to the same family, or they might be generated by the same tool with different configurations
- Common points
  - The number of dropped files and MD5 and size for 4 out of the 6 files
  - Accessed registry keys and files and created mutexes are identical
  - Confirming registry settings for audio related keys such as aux, mixer
  - Changing error reporting settings on Windows
- Difference
  - 2 out of the 6 files are different

P-2 / Accessed registry keys (completely matched)

```
 "keys": [
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥Windows NT¥¥CurrentVersion¥¥IMM",
            "HKEY_CURRENT_USER¥¥SOFTWARE¥¥Microsoft¥¥CTF",
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥CTF¥¥SystemShared",
...
            "Drivers¥¥wave",
            "Drivers¥¥wave¥¥wdmaud.drv",
              "Drivers¥¥midi",
            "Drivers¥¥midi¥¥wdmaud.drv",
            "Drivers¥¥aux",
            "Drivers¥¥aux¥¥wdmaud.drv",
            "Drivers¥¥mixer",
            "Drivers¥¥mixer¥¥wdmaud.drv",
...
"HKEY_LOCAL_MACHINE¥¥Software¥¥Policies¥¥Microsoft¥¥PCHealth¥¥ErrorReporting",
            "HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥PCHealth¥¥ErrorReporting",

"HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥PCHealth¥¥ErrorReporting¥¥ExclusionList",

"HKEY_LOCAL_MACHINE¥¥Software¥¥Microsoft¥¥PCHealth¥¥ErrorReporting¥¥InclusionList",
            "HKEY_LOCAL_MACHINE¥¥System¥¥Setup"
        ],
```

P-3

- Result
  - Both malware might be belong to the same family, or they might be generated by the same tool with different configurations.
  - Same attacker might generate both malware in different period because of appearance of a common C&C FQDN

- Common points
  - The number of dropped files and MD5 and size for 4 out of the 6 files
  - Accessed registry keys and files and created mutexes are identical
  - Changing the same IE and explorer settings
  - C&C FQDN is the same (sharing one identical C&C FQDN where each malware has 3 C&C FQDN)
  - Process tree in execution is the same

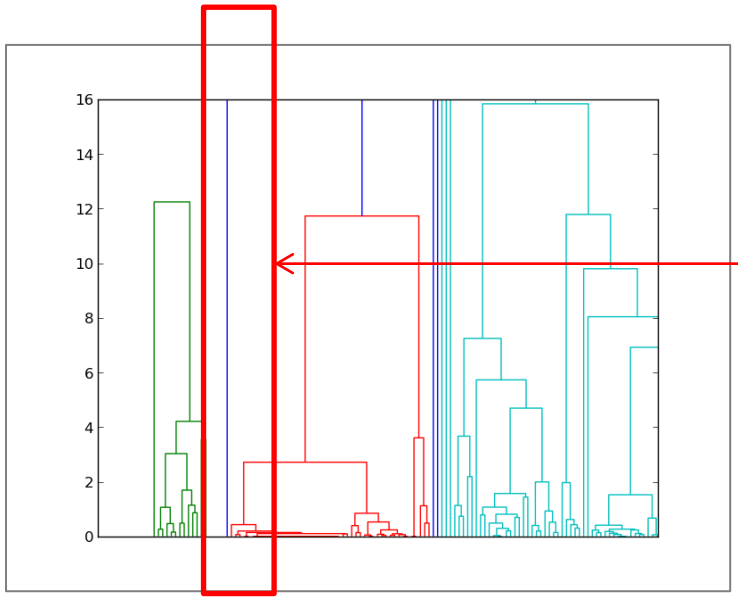
- Difference
  - 2 C&C FQDNs out of the 3 are different

P-3 / Process tree


```
{
        "pid": 404,
        "name": "9F267AE8FB419F2071795803216A3455.bin",
        "children": [
          {
            "pid": 388,
            "name": "9F267AE8FB419F2071795803216A3455.bin",
            "children": [
              {
                "pid": 1832,
                "name": "taskhost.exe",
                "children": [
                  {
                    "pid": 1828,
                    "name": "taskhost.exe",
                    "children": []
                  }
                ]
              }
            ]
          }
        ]
      }
```

5.Considerations / Is it useful to determine novel unique malware?

- Following malware is likely to novel unique ones if we assume that clustering works well based on the result above
  - Undetectable by current anti-virus software
  - As a result of clustering, it is represented by unique or sparse tree



If this malware is undetectable,
It is possibly new malware

5.Considerations / Does it boost manual malware analysis?

- If unknown malware are making up a group in a cluster, their functions and behaviors also should be similar (It would be helpful to analyze them)
- Malware detected by heuristic engines are also able to be classified more accurately
  - We confirmed the same "HEUR"-prefixed malware is classified to another family

5.Considerations / Is it helpful to sample worthful data?

- We can consider dividing data into arbitrary sized clusters and sampling data from each clusters

- Especially if trend of each clusters are different, it is useful
  - Stratified sampling
    - http://en.wikipedia.org/wiki/Stratified_sampling

6.Conclusions

- Need to consider schemes to use malware as assets since increasing of malware
- Therefore, we evaluated clustering with ward's method
- In limited evaluation, we confirmed it works well
- We can solve the requests using this result
  - Determining malware to analyze preferentially
  - Making analysis more efficiently
  - Valuable malware sampling

7.Future works

- Comparing other features and methods
- Considering methods to compare malware behaviors and functions
  - MAEC(Malware Attribute Enumeration and Characterization) might be useful
  - http://maec.mitre.org/
- More user-friendly UI/IF for a result of clustering
- Considering automation and systematization

Contact Information

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