L-025

機能特性に基づくマルウェアの可視化とその分類
Malware visualization based on the behavior and its classification

浦辻 和也† 松重 雄大‡ 甲斐 博† 森井 昌克‡
Kazuya Uratsuji Takahiro Matsushige Hiroshi Kai Masakatu Morii

1. Introduction

Due to rapid growth of the Internet, a great variety of malware has spread all over the Internet, and damages from malware are yearly increasing. Malware is a generic term of codes or software developed with malicious intent, e.g. computer virus, Trojan horse, warm and so on. To protect against malware attacks, their behavior analysis is important. There are many researches about the method to analyze the malware behavior. Major security vendors have provided analysis results for malware behavior using their automated system. However, if malware have complicated behavior, then the analysis results are also complicated. We are required to have enough technical knowledge and plenty of time to understand the analysis results.

The number of types of malware grows even larger every day, but most malware are created as variants of original malware. There are not so many malware created as completely original one. Therefore, classification of malware families benefits to understand characteristics of malware.

However, we do not have a clear classification criteria about malware. Since malware generally have complicated structures, the analysis could produce somewhat different results due to different criteria.

In this paper, we propose a malware 3D model based on the malware functions defined by the properties or objectives of malware, for example, modifying files and connecting networks and so on. Suppose a malware as a set of the malware functions, the 3D model is presented as a combination of various 3D models representing the malware functions. We refer the Security Response provided by Symantec Cooperation to construct the malware model. We will show that we can understand difference or similarity among the malware variants from the visualization directly through some experiments.

2. Functional Classification of Malware

Malware are classified roughly into several categories, for example, computer virus, warm, Trojan horse, bot and so on, depend on their functionalities. But, we do not have a clear commonly defined criteria for detailed classification of malware after the rough classification results.

The benefits of the detailed classification are easily to confirm malware activities and the differences between variants of malware. Therefore, there are a lot of researches for the detailed classification of malware, for example, see [3], [4].

As a scheme about the detailed classification, the naming rules of malware defined by the security vendors might be utilized. But, the naming rules are depend on the security vendors. For example, a worm-type malware called W32/Bugbear@mm by the Symantec Corporation is named as I-Worm.Tanatos.b by the Kaspersky Corporation. Thus, we do not consider about the malware names in this paper.

The detailed classification has been studied using the results of the static analysis and the dynamic analysis [3] [4], but different criteria give different classification results.

A malware often has a specific behavior such as infections, subversive activities, and network accesses. In the infection activities, a malware investigates the version of the operating system, software we are running, the directory structures and so on, and then rewrites or adds files on the computer system. In the subversive activities, a malware investigates the environment of the computer system, and rewrites files and data. In the network access activities, a malware also investigate the environment. Thus functions to obtain various information such as files, registries, DLLs, and so on are important for malware.

Therefore, in this paper, we define malware functions, and classify the malware functions into some function groups. If we can get functions of a malware from a malware analysis result, then we have a detailed classification according to the combination of the functions and function groups.

In this paper, we define the following four function groups.

[Information Collection Activities]

We classify the functions to collect files, software versions and passwords into this group. We defined three functions in this group: “Investigate specific files”, “Collect user information”, and “Collect strings in specific places”.

[Infection Activities]

We classify the functions to enable the spread of malware, the creation of botnets into this group. We define three functions in this group: “Create files, registries, and DLLs”, “Rewrite files, registries, and DLLs”, and “Delete files, registries, and DLLs”.

[Subversive Activities]

We classify the functions to perform destruction of data and systems, creation of a backdoor into this group. We define two functions in this group: “Create backdoors”, and “Destruction of systems”.

[Outside Activities]

We classify the functions to perform accessing to specific computers in a network, sending e-mails, and entering to a specific server into this group. We define five functions in this group: “Access to sites”, “Send e-mails”, “Login to servers”, “Download files”, and “Share networks”.

† 愛媛大学 Ehime University
‡ 神戸大学 Kobe University

We explain a method to obtain functions from the analysis results of malware in the next section, and we try to classify these functions into the function groups.

3. Extract the malware behavior

We use the Security Response [6] provided by the Symantec Corporation as the analysis results. We read the Security Response of a malware from the top of the pages to the bottom, and extract the type of malware, the functions, and the number of functions as malware characteristics.

Table 1. Function groups

Function Groups	Functions	Keywords	Color of Function Models	Shape of Function Models
Information Collection Activities	Investigate specific files	search	Red	Ellipsoid body
	Collect user information	collect	Green	
	Collect strings in specific planes	password	Blue	
Infection Activities	Create files, registries, and DLLs	create, copy,	Red	Cone
	Rewrite files, registries, and DLLs	modify, add	Green	
	Delete files, registries, and DLLs	delete	Blue	
Subversive Activities	Create backdoors	back, door	Red	Bowl
	Destruction of systems	end	Green	
Outside Activities	Access to sites	connect	Red	Horn
	Send e-mails	mail	Green	
	Login to severs	server	Blue	
	Download files	download	Yellow	
	Share networks	network	Cyan	

The malware type is extracted by a string which is descripted in the “Type” of the Security Response. There are several types of the “Type”, which are Worm, Trojan, Virus, and so on. Most popular malware are worms and Trojan houses in the Security Response. Therefore, in this paper, we handle about the two types of malware as shown in the Table 2.

Table 2. Malware types

Type	Character strings in the “Type”	Shape of the model
WORM	Worm	solid sphere
TROJAN	Trojan, Trojan Horse	cuboid

The malware behavior is described in “Technical Details” of the Security Response. It describe how to infect about the malware and what actions the malware performs after the infection. It also enumerates objects to be infected (Hereafter, they are called “function objects”).

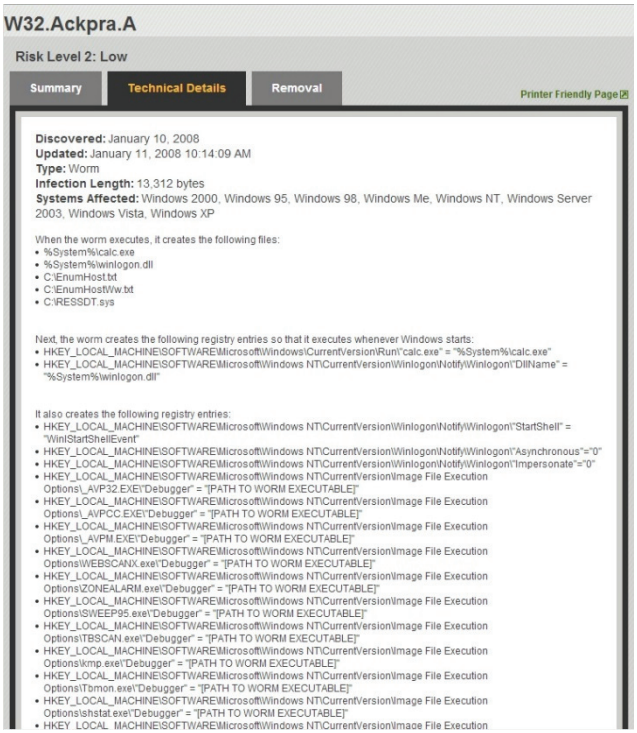

Figure 1. Security Response about W32.Ackpra.A.

As an example, Figure 1 shows the Technical Details of the Security Response about a malware, W32.Ackpra.A.

For example, “Type” of this malware is “Worm” from the third line in Figure 1. In addition, malware functions are shown from the seventh line in Figure 1. We can see that the malware creates files when it is executed. From lines 8-12, we can see five files which are actually to be created (which are called as function objects in this paper).

The function is explained by a sentence in English. In this paper, we extract which function is provided by the sentence using Keywords shown in the Table 1. Further, we count the number of the function objects which are followed by an English description about a malware function. We use them as characteristics of the 3D model of the malware.

These processes are performed automatically using Perl script. We ignore descriptions in the Technical Details which cannot be

classified using the Keywords in the Table 1, but we may classify them into functions if some appropriate keywords can be registered.

As stated above, we can extract type, functions, and function objects of a malware from the Security Response. (Note that the number of function objects is presented as the number of the actual number plus one for convenience of our system. If a function has no function objects, the number of function objects is unity.)

4. 3D models of malware

We define 3D models for the type and the functions which are shown in the previous section, and create a 3D visualization model of malware.

We define a 3D model of each type of malware (Hereafter, it is called as a type model). Figure 2 shows that the type models defined in this paper. WORM is expressed by a solid sphere, and TROJAN is expressed as a cuboid.

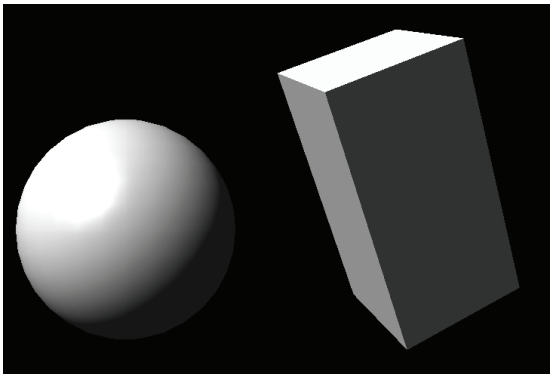

Figure 2. The type models（left : WORM, right : TROJAN）

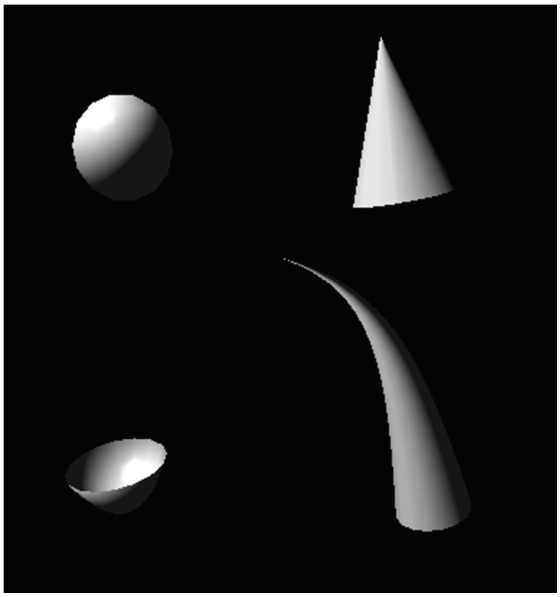

Figure 3. The function models (upper-left (ellipsoid body): Information Collection Activities, upper-right (cone): Infection Activities, bottom-left (bowl): Subversive Activities, bottom-right (horn): Outside Activities

Further, we define a 3D model of each function (Hereafter, it is called as a function model). We use a scheme to define the function models. That is, all functions belonging to a function group are expressed in a same shape (see Table 1 and Figure 3), and each function is distinguished with different colors (see Table 1).

Our 3D model is composed by a type model and one or more function models. The function models are arranged on the surface of a type model, because we need to know characteristics of the malware by the number of function models. The arrangement order of function models is important, since the function models are defined by functions of malware. Therefore, the function models are arranged from the top of the type model to the bottom, in a spiral form in the extracting order from the Security Response. We put the function models on the surface as many as the number of the function objects.

5. Experiments

In this paper, we create a software system which automatically obtain a 3D model of a malware from the Security Response.

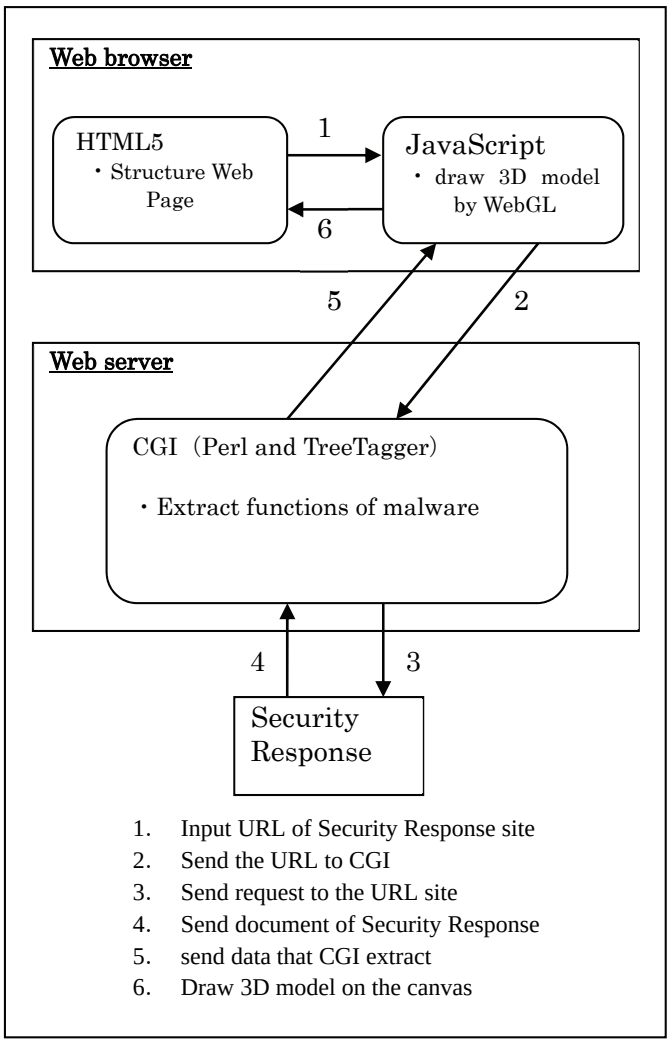

Figure 4. Visualization system of malware

When we input a URL of a malware's Security Response, the system extract a type, functions, and the number of function objects. Then, it displays a 3D model of the malware on the web browser using the extracted data.

A sentence appeared in the Technical Details in divided into a list of words using a morphological analyzer TreeTagger. A word are replaced with the base form of the word. Then, we confirm whether the keywords in Table 1 exist in the list of words or not.

As shown in the Figure 4, the process is done by using a CGI script (written in Perl language) from the web server program. The type, functions, and function objects are sent to the browser by Ajax. The 3D models of malware are displayed in the browser using JavaScript and WebGL [7] in <canvas> element of HTML5.

A Security Response is written by HTML. We can find the description of the malware type in the following part of the Security Response.

<dt class="unit mrgnRgtXSM">
Type:
</dt>
<dd>“TYPE”</dd>

The “TYPE” shows a string, and it decide the type of malware. The malware functions can be found between <h2>TECHNICAL DETAILS</h2> and <h3>Recommendations</h3>. If the malware has no description about <h3>Recommendations</h3>, we suppose the malware functions can be found between <h2>TECHNICAL DETAILS</h2> to the end of the file.

We extract the functions and the number of the function objects in the following steps.

[Step 1] We divide a sentence into a list of words using the TreeTagger, and replace each word with a base form.

[Step 2] If some keywords appear in the sentence, the number of each keyword is counted. Most appealing keyword is used to classify the sentence to the function. If we have no keyword in the sentence, then we go to step 3.

[Step 3] The sentence does not describe about a function. We suppose it present a function object.

We can extract functions and function objects from Step 1 to Step 3. If there is no function before a function object, the function object is ignored. It is a future work to process a document more precisely.

In this paper, we experimented to visualize the following malware, and confirm the effectiveness of this technique.

[W32.Ackpra.A]

This is a worm that spreads by copying itself to all network shares and removable disks. It also downloads malicious files.

[W32.BeagelCX@mm]

This is a mass-mailing worm that sends out copies of other malware to mail addresses which are included in downloaded files using its own SMTP engine. It also opens a backdoor.

[W32.BeagleDA@mm]

This is a variant of W32.BeagelCX@mm. There are modifications about website addresses to be downloaded and malware to be sent.

Table 3. W32.Ackpra.A

Function Group	Function	The Number of Function Objects
Infection Activities	Create files, registries, and DLLs	6
Infection Activities	Create files, registries, and DLLs	3
Infection Activities	Create files, registries, and DLLs	125
Infection Activities	Rewrite files, registries, and DLLs	2
Infection Activities	Delete files, registries, and DLLs	8
Infection Activities	Create files, registries, and DLLs	7
Outside Activities	Download files	4
Information Collection Activities	Investigates specific files	4
Information Collection Activities	Investigates specific files	9
Infection Activities	Rewrite files, registries, and DLLs	2
Outside Activities	Download files	9
Infection Activities	Create files, registries, and DLLs	2
Infection Activities	Create files, registries, and DLLs	2
Information Collection Activities	Collect strings in specific planes	14
Information Collection Activities	Collect strings in specific planes	15

Table 4. W32.Beagle.CX@mm

Function Group	Function	The Number of Function Objects
Infection Activities	Create files, registries, and DLLs	1
Infection Activities	Rewrite files, registries, and DLLs	6
Infection Activities	Create files, registries, and DLLs	8
Infection Activities	Delete files, registries, and DLLs	40
Infection Activities	Delete files, registries, and DLLs	3
Subversive Activities	Create backdoors	1
Outside Activities	Login to severs	1
Outside Activities	Send e-mails	3
Outside Activities	Download files	15
Infection Activities	Create files, registries, and DLLs	130
Information Collection Activities	Collect strings in specific planes	1

Information Collection Activities	Collect strings in specific planes	121
Infection Activities	Create files, registries, and DLLs	1
Infection Activities	Create files, registries, and DLLs	46
Subversive Activities	Destruction of systems	3

Table 5. W32.Beagle.DA@mm

Function Group	Function	The Number of Function Objects
Infection Activities	Create files, registries, and DLLs	1
Infection Activities	Rewrite files, registries, and DLLs	5
Infection Activities	Create files, registries, and DLLs	8
Infection Activities	Delete files, registries, and DLLs	41
Infection Activities	Delete files, registries, and DLLs	2
Subversive Activities	Create backdoors	1
Outside Activities	Access to sites	1
Outside Activities	Send e-mails	3
Outside Activities	Download files	14
Infection Activities	Create files, registries, and DLLs	131
Information Collection Activities	Collect strings in specific planes	1
Information Collection Activities	Collect strings in specific planes	121
Infection Activities	Create files, registries, and DLLs	1
Infection Activities	Create files, registries, and DLLs	46
Subversive Activities	Destruction of systems	3

We extract characteristics and visualization models by our method. The malwares, W32.Ackpra.A, W32.Beagle.CX@mm, and W32.Beagle.DA@mm are experimented.

The characteristics are shown in Table 3, Table 4, and Table 5, respectively. The 3D models are shown in Figure 5, Figure 6, and Figure 7.

We can visually confirm these malware type are worm, because each 3D model has a solid sphere with many function models on the surface.

We can confirm that the two malwares, W32.Beagle.CX@mm and W32.Beagle.DA@mm, are very close each other by the results of Table 3 and Table 4. However, we can see that W32.Ackpra.A is different from the others by Table 3.

Regarding about the 3D models, we can confirm that the two 3D models shown in Figure 6 and Figure 7 are very similar each other. On the other hand, the 3D model shown in Figure 5 is different from the others. The malwares can be successfully classified by our method in these experiments.

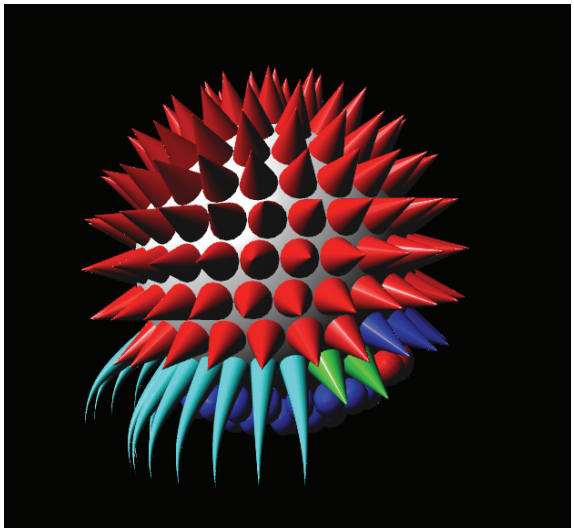

Figure 5. W32.Ackpra.A

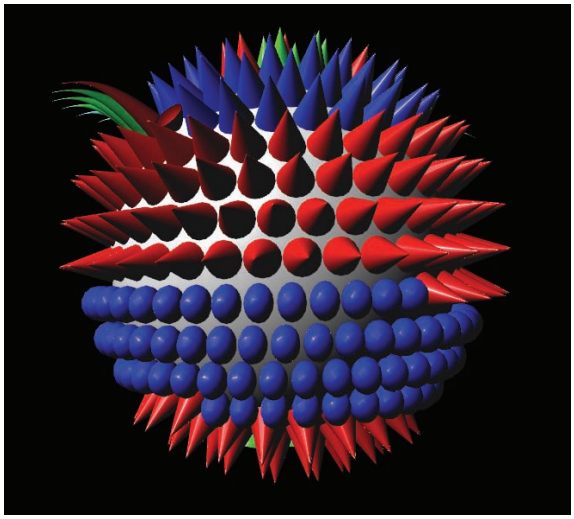

Figure 6. W32.Beagle.CX@mm

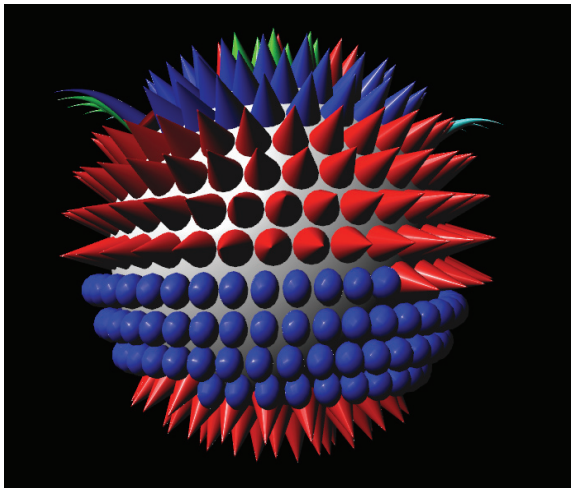

Figure 7. W32.Beagle.DA@mm

1. Conclusion

In this paper, we proposed a method to classify a malware using malware functions extracted from the Security Response. Then, we proposed a method to visualize a malware 3D model based on the malware functions.

The visualization method can produce a malware 3D model automatically. We created a software system which displays a 3D model of a malware on the browser using WebGL, when a Security Response URL is given as the input. In this paper, we showed the 3D model as a static image, but the 3D model can rotate the objects by any angles on the web browser.

We gave experiments to visualize three malwares using our system, and showed that it is possible to confirm similarities or differences among the malwares.

Malware is evolving every day, and the number of malware is increasing. Therefore, we need to change or add the definitions of functions as required according to the malware evolution.

References

[1] Kohei Nasaka, Takahiro Sakai, Takumi Yamamoto, Keisuke Takemori, Masakatsu Nishigaki, “A study on classification of malware based on automatic execution set-up”, IPSJ SIG Notes, Vol.2010-CSES-50, No.40, pp.1-5, 2010, in Japanese.

[2] Kazuki Iwamoto, Katsumi Wasaki, “Investigation on Results of Malware Classification based on the Static Analysis”, DICOM2010, Vol.2010, No.1, pp.477-491, 2010, in Japanese.

[3] Masashi Fujiwara, Masato Terada, Tetsuya Abe, Hiroaki Kikuchi, “Study for the classification of malware by infection activities”, IPSJ SIG Notes, Vol.2008, No.21, pp.177-182, 2008, in Japanese.

[4] Keiichi Horiai, Takafumi Imaizumi, Hidehiko Tanaka, “Proposal and Implementation of Automated Malware Variant Classification Based on Dynamic Behavior”, IPSJ Journal, Vol.50, No.4, pp.1321-1333, 2009, in Japanese.

[5] Soma Katsute, Kouki Yasumoto, Ryoichi Isawa, Masakatu Morii, Koji Nakao, “Study on Classification of Malware and it Applications”, IEICE technical report, ICSS, Vol.109, No.86, pp.103-108, 2009, in Japanese.

[6] Symantec Security Response, “http://www.symantec.com/security_response/” (June, 2014).

[7] Khronos Group, WebGL OpenGL ES 2.0 for the Web, “http://www.khronos.org/webgl/” (June, 2014).

[8] Akira Morikawa, Masakatsu Morii, “Visualization of the characteristics for malware behavior”, Kobe University master’s thesis, 2011 in Japanese.