Design of Tracking and Capturing Method for Abnormal Behavior in Marketing System Based on Sandbox Technology

Yunzhao Li

State Grid Xinjiang Electric Power CO., LTD., 830001 Urumqi, China

Abstract. In this paper, a tracking and capturing method for abnormal behavior in marketing system is designed. The main steps of the method include: (1) Trace generation based on function injection. This method selects NOP, HLT and other instructions of untrusted processes to fill the memory area, and injects the test. The function and control process execute the function to obtain Trace; (2) Trace & capture-based sandbox interception system function analysis method, first establish a finite state automaton model describing the instruction address translation of the untrusted process call system function, according to the state of the automaton. The conversion determines whether the system function calling process has interception behavior, and further identifies the system function of the sandbox interception according to the position of the state machine conversion instruction in the automatic machine. This paper designs and implements the prototype system, called TC-analyser, and solves the key problems of the function injection timing selection, Trace intermediate representation and other implementation processes. Finally, choose Chromium and Adobe Reader to test the TC-analyser's interception recognition capability and compare it with Hooks hark. The experimental results show that the TC-analyser has good interception and recognition capabilities.

1 Introduction

With the extensive development of various marketing activities, various types of attacks against marketing systems are becoming more frequent. Common behaviors include the use of lottery credits or coupons for bulk piracy, the use of lottery point's logic loopholes for profit, and the use of account bank piracy Points, scalpers, etc. These attacks greatly reduce the expected benefits of marketing activities and threaten business development.

In the security protection of the marketing system, sandbox is suspicious program (malicious program or undisclosed vulnerabilities using) the important security mechanism, is composed of trusted and untrusted part. Sandbox untrusted part is dealing with untrusted data into/thread or untrusted programs, sandbox credible parts to untrusted service [1] and intercept the untrustworthy marketing system function called [2] run-time monitoring and verification to ensure the safety of the system. Sandbox of monitoring and verification process is complex and error prone, attackers often use its defects to realize sandbox escape, such as CVE-2011-1353 [3], CVE-2013-0641 [4] and CVE-2013-3186 [5].

The researchers analysed the realization mechanism of different sandbox, and on this basis, using the fuzzy measurement technology [6,7] test sandbox defects, for example, some researchers studied different implementation of the sandbox [8-11] and service test method [12-13]. Existing sandbox at test sandbox to provide service, but the lack of a sandbox to intercept and validation unreliable part of the call marketing system function test, and the process is safety defects. Compared to the services provided by the sandbox intercepted and change control flow to obtain unreliable part of the marketing system function of resource access behavior and verification, therefore, the sandbox mechanism of monitoring and validation of test need to first identify the sandbox intercept marketing system function set, which identify the sandbox intercept.

This paper designs a sandbox to tracking and capturing abnormal behaviour in marketing system. This method firstly in the process of untrusted injection and perform the marketing system function to get used to analyze Trace, secondly, introduces the address space of finite state automaton, the sandbox intercepted by automata state recognition behavior, further analysis to achieve state transition directive information for identifying marketing system function; Again, the prototype system is constructed TC-analyser, in the process of implementation, such as injection timing, Trace the function conversion implementation issues; Finally, choose the mainstream sandbox to verify the accuracy and practicability of the method in this paper, the experimental results showed that compared with the existing capture identification methods, this method has the same sandbox intercept ability to recognize, at the same time, more automatic, more efficient.

2 Based on the method of determining trace injection function

Based on function method of determining Trace injection in credibility in the process of memory testing marketing system function, and control of the injection process execution function to generate Trace for sandbox intercept recognition (figure 1). In order to better describe the method of determining Trace injection based on function, and several important concepts are given here first:

Definition 1 address space: according to different sources of executable code in the memory, the instruction of address range is marked as the corresponding address space, the paper address space including unreliable process address space (PC), the system address space (OS) and the sandbox address space (SC) of three parts.

Definition 2 Trace: refers to the process of unreliable in the process of marketing system function called sandbox, not credible to the instruction information process execution.

Remember a trace for access to the marketing system function T (A). $T(A)=\{i_1,i_2,...i_k...,i_{m-1},i_m\}$ is the execution of instructions set, $i_k =< no, op, saddr, taddr >$ is the first k instructions in the Trace information, including instruction operation code (op), the host address (saddr), instruction execution order (no) and the next address (taddr) to carry out the instructions.

Definition 3 can be injected into the address area (ms): can be used to inject untrusted process area of memory function, ms including address (addr) and size (size) two attributes. In Windows, the program USES the NOP, HLT instructions filling program code to ensure that alignment, in order not to damage not credible process itself the function of the code, we choose the NOP, HLT instruction filling area for ms.

Definition 4 marketing system function (s): s function is used to test the system, including the function parameter information (params) and the required memory size (size) two attributes.

$$\begin{cases} ms_i.size - DJMP.size - Max(Inst.size) \geq 0 \cdots\cdots 1 \\ \sum_{i=1}^{k} ms_i.size - s.size \geq 0\ ,\ \min(k) \quad \cdots\cdots 2 \end{cases} \tag{1}$$

Trace generation algorithm based on function into the main steps of as shown in figure 2, including injection, function implementation and Trace for three steps. Injection in function, if there is a single memory is greater than the injection function required memory space of ms, the direct injection on the ms address s_i, otherwise, select multiple injection s_i, ms function, after the completion of injection control suspect the function and obtain the Trace process execution.

When choosing multiple ms injection, in order to reduce the instruction execution of untrusted process influence on Trace injection function, at the same time ensure injection function coherent execution and improve the efficiency of Trace formation, this article USES the formula 1 choice function into related ms, due to the use of jump instruction to connect multiple ms, so a formula 1 child type (1) to ensure that selection of ms at least able to accommodate a direct jump instructions, an injection of function, among them, because the injection function of instruction length is unknown, in order to guarantee the universality of Trace generation algorithm, fruit type (1) the instruction length (Inst. Size) choose the sandbox maximum length of instruction system, for example, a 16-bit system's Max (Inst. Size) = 7, 32 bit system's Max (Inst. Size) = 14; Child type (1) (2) in the type selection, on the basis of choice as little as possible of ms to inject the marketing system function of the test.

Algorithm: based on the function of injection Trace generation algorithm

Input: $MS=\{ms_1,ms_2,ms_3...ms_n \mid n>0 \,\&\, n\in N\}$

The test set of functions

$$S=\{s_1,s_2,s_3...s_m \mid m>0 \,\&\, m\in N\}$$

Output: S **Trace** elements in collection, **T**;

(1) Initialization: Untrust Process p, $T=\phi$

(2) While $i \leq S.size$:

(3) Select s_i

(4) If exist $ms_j.size \geq s.length$:

(5) Inject s_i in ms_j;

(6) Else use formula1 to select multi-ms: inject s_i in multi-ms

(7) execute s_i and get $trace_{si}$

(8) T=T+$trace_{si}$

Return T

In recognition of the sandbox in 32-bit Windows system S whether intercepted CreateFileW, for example, assume that S not credible process there is no single memory space than the CreateFileW required memory space, need to select more than one ms CreateFileW injection and its parameters. Due to the 32-bit system directly jump instructions 5 bytes and maximum length 14 bytes, namely the DJMP. Size = 5, Max (Inst. Size) = 14, therefore, formula 1 screening of ms is not less than 19 bytes. This paper use the function USES of the greedy strategy, that is, as much as possible in the screening of ms injection CreateFileW related instructions, assuming that the size of the screening of ms is 19 bytes, so injection results as shown in figure 1, among them, mss, ms1 and ms2 respectively, and injected CreateFileW related 5, 2, and 1 instruction.

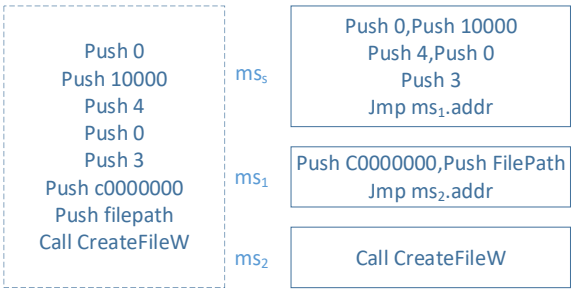

Figure 1. Injection CreateFileW beckoned.

Define 5 address space finite state automaton $M =< Q, q_0, \xi, q_0, \delta >$, Q is the set state of address space, including q_0,q_2,q_1 three elements, among them, the q_0

state said instruction in PC, q_1 state said instruction in OS, q_2 state said instructions in SC; ξ is the input table M, as a result of M state transitions are implemented by an instruction, therefore, ξ including in the system to provide all the instructions, and we need according to the analysis of the intercept, the instruction is divided into nine categories, and provide the address space transform semantics, as shown in table 1. δ is the state transition function of M, and, $\delta \subset \{Q \times \xi \to Q\}$ as shown in figure 2.

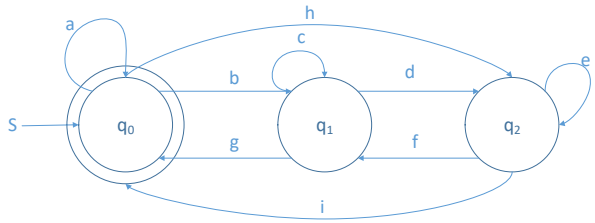

Figure 2. State transition function δ.

Table 1. The type of elements ξ.

Instruction type	Saddr	Taddr	Instruction type	Saddr	Taddr
a	PC	PC	f	SC	OS
b	PC	OS	g	OS	PC
c	OS	OS	h	PC	SC
d	OS	SC	i	SC	PC
e	SC	SC			

Use a two-column format, and set the spacing between the columns at 8 mm. Insert “MATEC Web of Conferences” in even headers and the short form title of the conference in odd headers, except the first one. Please check with the organiser the exact short title of the conference. Do not add any page numbers. The articles will be quoted as follows:

2.1 Identification of behaviours

In Figure 3, it is a no-credible process call of the marketing system function, which could in the schematic diagram of the trace address space transformation, among them, the (a) function has not been a sandbox intercepted, belongs only to the instructions of the Trace process address space and system address space; (b), (c) the sandbox function in the system address space to implement system of intercepting the address space of transformation; (d), (e) represent the untrusted sandbox process address space to implement Capture of address space transformation.

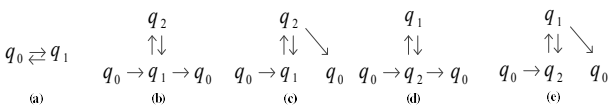

Figure 3. Trace address space transformation.

By comparing the five different situation, is the sign of marketing system function, can be found as a result, we by judging whether included in the Trace can identify whether there is intercepted.

2.2 Recognition

Sandbox intercepted don't access to the marketing system function can make not credible process flow of control is transferred to the sandbox address space, namely $q_0 \xrightarrow{h} q_2$ or $q_1 \xrightarrow{d} q_2$ is involved in the analysis of Trace class d, h class instruction can judge the existence of a sandbox intercept behavior, but if only through the class d, h class instruction to determine the marketing system function, the sandbox intercept will appear. To 6 as an example, the function is used to test A marketing system function, the function is the function B A call in the process of implementation of marketing system functions, (a) there is no intercept marketing system function, (b), (c) is the condition of the intercept, but intercept (b) function is A, (c) intercepted function is B. Therefore, we use the class d, h class instruction to determine the location of marketing system function.

$$\begin{cases} m.no - c_i.no \geq 0 & 0 < i \leq k \\ r_i.no - m.no \geq 0 & 0 < i \leq k \\ m.no - c_i.no \leq \min(m.no - c_j.no), 0 < i, j \leq k \end{cases} \quad (2)$$

The method adopts the instruction address to identify the relationship between the sandbox intercepted marketing system function, any normal execution of a function call at the instruction level of existence call instruction c and the corresponding return instructions r, and the two meet, so, execute the instructions for the function between the corresponding instruction, therefore, through the class d, h class instruction of < c, r > scope to identify sandbox Capture marketing system function. We use the formula to determine the marketing system function, the sandbox intercept 2 first, before using two formulas to choose meet instruction > < c, r, limit the scope of further selection contains d, h class instruction the minimum < c, r > of identified as a sandbox intercepting marketing system function, among them, we use m on behalf of the class d, h class instruction, k said Trace number contains < c, r >.

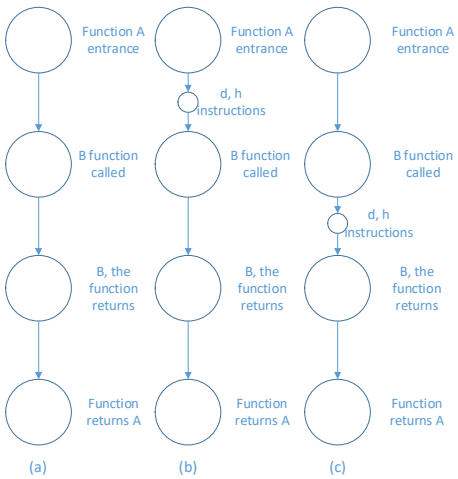

Figure 4. Marketing system function identification of false positiv.

3 Efficiency analysis

We will be deployed in a dual core 3.4 GHz TC-analyser I3 CPU processor and 4 gb of RAM Linux machine, with 512 MB of memory 32-bit Windows XP Sp3 as test system. Because the 32-bit operating systems, direct jump instruction takes up space for 5 bits, biggest instruction length for 14 bits, therefore, in this experiment, DJMP in formula 1. The size, Max (Inst. Size) values of 5 and 14 respectively.

Trace is the basis of analysis of the intercept, as a result, the efficiency of the method analysis focused on marketing system function Trace the formation of scale, Trace the time needed for generating and Trace the efficiency. We are on the basis of Chromium default interceptor, through modify and recompile the Chromium sandbox source to realize the capture of different function.

Do not break general, this article choose Windows Xp two core library NTDLL and KERNEL32 exported functions as a test set, among them, the NTDLL and KERNEL32 deduced respectively 1317 and 955 systems function.

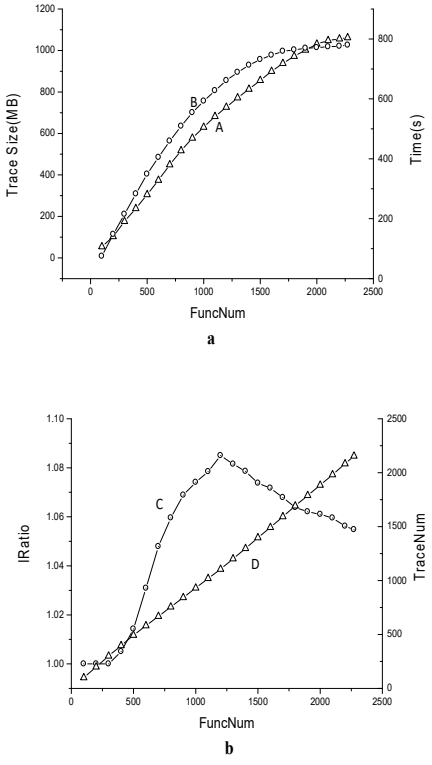

Figure 5. TC-analyser efficiency analysis.

Figure 5 is the result of the analysis efficiency curve, among them, the (a) of the two curves are Trace generation time, the relationship between Trace scale and the sandbox intercepted number, A curve representing the scale of Trace ,curve B on behalf of the Trace generation time, increase in the number of the sandbox Capture marketing system function, to generate the scale and timing of the Trace will increase, but due to marketing system function call each other between the relationship, with the increase of the sandbox Capture function, a Trace of TC-analyser can identify multiple intercepted marketing system function, therefore, the two curves are increase in the number of Capture function, time cost and the scale of the Trace of growth slowing down.

4 Conclusion

Hook technology is the major technology of sandbox interception. Existed Hook detecting methods and tools always foucs on the existence of hooks, But sandbox-interception recognition not only need to determine whether there is hook, but need to find sandbox intercepted system functions. In this paper, we proposed a Function-Injecting based Sandbox interception recognition method. The method inject and execute the system function in the untrusted program, record the function trace, then, analysis trace in the address space finite state automata and identify sandbox intercepted system functions. Finally, traversing the function set to identify the sandbox interception. We implemented a prototype, TC-analyser, testing it with Chromium Sandbox and Adobe Reader Sandbox, the test results demonstrate the effectiveness and practicality of the method.

References

1. Bennet Yee, David Sehr, Gregory Dardyk, et al. Native Client: a sandbox for portable, untrusted x86 native code [J]. Commun. ACM, 2010 (1): 91-99.
2. Inside Adobe Reader Protect Mode [OL]. http://blogs.adobe.com/security, 2010.
3. CVE-2011-1353[OL], https://web.nvd.nist.gov, 2011.
4. CVE-2013-0641[OL], https://web.nvd.nist.gov, 2013.
5. CVE-2013-3186[OL], https://web.nvd.nist.gov, 2013.
6. Cui Bao-jiang, Liang Xiao-bing, Wang Yu. The study of binary program test techniques based on backtracking and leading for covering key code area [J]. Journal of Electronics& Information Technology, 2012, 34 (1): 108-114.
7. Ou Yang Yong-ji, Wei Qiang, Wang Qing-xian. Intelligent Fuzzing Based on Exception Distribution Steering [J]. Journal of Electronics & Information Technology, 2015, 37 (1): 143-149.
8. Sabanal, Paul, and Mark Vincent Yason. Playing in the Reader X Sandbox [OL], https://www.blackhat.com, 2011.
9. Mark Vincent Yason. Understanding the Attack Surface and Attack Resilience of Project Spartans New EdgeHtml Rendering Engine [OL], https://www.blackhat.com, 2015.
10. James Forshaw. Digging for Sandbox Escapes Finding sandbox breakouts in Internet Explorer [OL], https://www.blackhat.com, 2014.
11. Koh, Yong Chuan. Understanding the Microsoft Office 2013 Protected-View Sandbox [OL], https://recon.cx/2015/slides/ 2015.
12. Xiaoning Li, Haifei Li. Smart COM Fuzzing-Auditing IE Sandbox Bypass in COM Objects [OL], https://cansecwest.com/csw15archive.html, 2015.
13. Brain Gorenc, Jasiel Spelman. Thinking outside the sandbox Violating trust boundaries in uncommon

ways [OL], https://www.blackhat.com/html/bh-media-archives/bh-archives-2014.html, 2014.
14. Zhenhua Liu, Guilaume Lovet. Breeding Sandworms: How to fuzz your way out of Adobe Reader's Sandbox [OL], https://www.blackhat.com/html/bh-media-archives/bh-archives-2012.html, 2012.
15. Wang Z, Jiang X, Cui W, et al. Countering persistent kernel rootkits through systematic hook discovery[C], Recent Advances in Intrusion Detection. Springer Berlin Heidelberg, 2008: 21-38.
16. Yin H, Poosankam P, Hanna S, et al. HookScout: Proactive binary-centric hook detection [J], Detection of Intrusions and Malware, and Vulnerability Assessment. Springer Berlin Heidelberg. 2010: 1-20.
17. Butler, James, and Greg Hoglund. VICE–catch the hookers [OL]. https://www.blackhat.com/html/bh-media-archives/bh-archives-2014.html, 2014.
18. Rutkowska J. System virginity verifier [OL], http://www.cs.dartmouth.edu, 2015.
19. Hooks hark [OL]. http://www.gamedeception.net, 2010.
20. Bellard, Fabrice. QEMU, a Fast and Portable Dynamic Translator [C]. Proc. USENIX Annual Technical Conference, FREENIX Track. Marriott Anaheim, CA, 2005.