On a Computer Security Incident Response System

Motoyuki OHMORI[1,a)] Masayuki HIGASHINO[1,b)] Shinichi KAWANARI[1,c)]
Satoshi FUJIO[1,d)] Kiyoyuki NAKASHIMA[1,e)] Naoki MIYATA[1,f)] Shinichi MOTOMURA[1,g)]

Abstract: Computer security has been gathering more attentions, and responses against security incidents have been getting more important. We are now facing difficulties to quickly and properly respond against incidents. For examples, we may be unable to immediately locate a suspicious host when an Intrusion Detection System (IDS), Intrusion Prevention System (IPS) or external organization detects the anomaly, and we cannot immediately isolate the suspicious host. We cannot also properly share information about the security incident, and it is getting difficult to find out what is going on, who is in charge of the incident. In order to solve these issues, we are now trying to establish a computer security incident response system using a bug tracking system. This paper discusses these issues and possible solutions.

1. Introduction

Computer security has been gathering more attentions. A response against a security incident has been then getting more important in order to mitigate damage of the incident. This mitigation requires a quick and/or proper response against the incident. A quickness is more important because of following reasons. It take a day or few days after an infection until a malware begins a malicious behavior such as compromise of confidential information or an attack to other hosts in many cases XXXCitationXXX. One can then avoid or reduce a severe damage if a suspicious host is isolated from a network in few days. To this end, it is recently becoming more common that an institute organizes a Computer Security Incident Response Team (CSIRT). A CSIRT is a cross organizational team that is in charge of the first response against a security incident. A CSIRT recognizes a security incident suspicion reported from:

- a member himself/herself in a organization,
- a CSIRT itself, or
- an external organization.

Among them, this paper focuses on a security incident detected by an external organization such as Security Operation Centers (SOCs) of Japan SOC (JSOC) [1] operated by LAC Co., Ltd, so-called NII-SOC operated by National Institute of Informatics (NII), government organizations or others. When an external organization reports a security incident suspicion, IP addresses of a suspicious host and its corresponding host are given. If an IP address and MAC address of a host are registered into a database, a CSIRT can then *locate* the suspicious host; which switch accommodates the suspicious host on which port. All organization, however, do not have and maintain such database, and we, Tottori University, indeed do not. It may be difficult to establish such database from scratch. Even if such database is maintained, information of all hosts may not be up-to-date, e.g., some hosts are not registered or registered to have previously assigned IP address. An administrator may be then unable to locate a host that is assigned the given IP address from a database. In this case, an administrator can manually locate a host as follows:

(1) locate a router that has a *directly connected* route for an given IP address,
(2) identify a VLAN for the given IP address at the router,
(3) resolve a MAC address for the given IP address at the router,
(4) identify a port on which the MAC address is seen in a MAC address forwarding table,
(5) discover a neighboring switch on the port,
(6) repeat from (4) to (5), and
(7) finally locate an edge switch and a port accommodating the MAC address.

This may, however, take time and cause a misoperation, and they should be improved from a point of a view of a security incident response.

Another issue on a security incident response is how to isolate a suspicious host from a network. For example, an administrator can plug off a network cable or shut down a port. These method may, however, isolate other hosts that are accommodated to the same port via a another switch that a end user installs and an administrator cannot handle. In addition, these method cannot be effect when a suspicious host is a mobile host such as note PC, smart phone or tablet and frequently moves around.

Sharing information is also an important issue on a security incident response. On a security incident, a traditional communication such as phone tends to be often used because those who may

[1] Tottori University, Koyama-minami, Tottori Japan, 680–8550 Japan
[a)] ohmori@tottori-u.ac.jp
[b)] higashino@tottori-u.ac.jp
[c)] kawanari@tottori-u.ac.jp
[d)] s-fujio@tottori-u.ac.jp
[e)] nakashima@tottori-u.ac.jp
[f)] miyata@tottori-u.ac.jp
[g)] motomura@tottori-u.ac.jp

concern seem to consider a traditional communication is faster. A traditional communication, however, tends to be a one-to-one communication, and information is limited to those two persons and cannot be shared with others, especially actual working-level operators. To make matters worse, information is wrongly propagated one after another, and the original information is lost. In addition, no record is rarely left because persons involved are in a hurry.

This paper proposes and discusses a computer security incident response system that tries to deal with issues described above. The proposed system comprises of three sub-system, host locating system, host isolating system and incident tracking system.

The contributions of this paper can be summarized as follows:

- a suspicious host can be located within 10 seconds even though a database of all hosts in a network is not maintained,
- a suspicious host can be isolate in 10 min. right after an external organization reports a security incident suspicion,
- only a suspicious host can be isolated even though other hosts is on the same port which the suspicious host is connected to, and
- a tracking system can help CSIRT in a organization to share correct information about a security incident and reduce operations.

The rest of this paper is organized as follows. Section 4 proposes a computer security incident response system that comprises three sub-systems. Section 2 and 3 defines assumptions and requirements of a computer security incident response system, respectively. Section 5 presents our first prototype implementation. Section 6 disscusses various issues to be considered on a computer security incident response system. Section 7 refers to related works. Section 8 finally concludes this paper.

2. Assumptions

This section describes assumptions of information given by an external organization on an security incident suspicion and a network configuration in an organization. The proposed system assumes that:

- an IP address of a suspicious host is given when an external organization report a security incident suspicion,
- a Virtual Routing Forwarding (VRF) or a routing domain is given or no IP address is duplicately assigned to different hosts in a organization network,
- Link Layer Discovery Protocol (LLDP) [2] or similar protocol such as Cisco Discovery Protocol (CDP) is enabled on all links where two neighboring switches directly connect in an organization, and
- a department may install a its own router, and may be able to be operated only by an administrator in its department, not by an organization-wide administrator.

3. Requirements

This section defines requirements of a computer security incident response system.

3.1 Requirements for Host Locating System

The host locating system is in charge of *locating* a suspicious host. When a security incident suspicion is reported, the host locating system must:

(1) need no pre-defined host database that holds information about all hosts in an organization network,
(2) require an IP address of one of routers or L3 switches in an organization network,
(3) require an IP address of the suspicious host,
(4) require a Route Distinguisher (RD) or name of VRF if and only if necessary,
(5) produce *location information* of the suspicious host for the host isolating system, and
(6) identify a responsible person, e.g., a user, of the suspicious host.

3.2 Requirements for Host Isolating System

The host isolating system is in charge of immediately isolating a suspicious host from a network in an organization. The host isolating system must:

(1) minimize the number of isolated hosts when the suspicious host is isolated,
(2) avoid wrongly isolating hosts or other nodes,
(3) require *location information* of the suspicious host that the host locating system produces,
(4) isolate the suspicious host from a network,
(5) notify an administrator that the suspicious host is isolated,
(6) revert the isolation, and
(7) support a dry run mode in which the suspicious host is not actually isolated.

3.3 Requirements for Incident Tracking System

The incident tracking system is in charge of sharing information among staffs involved, recording actions that staffs involved take and observed phenomenon, and make an incident trackable. The incident tracking system must be able to:

(1) share information among staffs involved in a security incident response,
(2) issue a ticket for an incident,
(3) differentiate *open* and *closed* issues.
(4) merge the similar incidents into one ticket,
(5) group members in an organization by department,
(6) register staffs contact information in advance,
(7) notify staffs involved of updates of an incident,
(8) upload a file for an incident,
(9) automatically produce a final report of an incident, and
(10) automatically produce a summary of incidents during specified duration.

4. Computer Security Incident Response System

This section proposes a computer security incident response system. The system comprises of three sub-system, host locating system, host isolating system and incident tracking system.

4.1 Host Locating System

The host locating system dynamically locates a suspicious host; the suspicious host is connected to which port on which

switch. The host locating system requires only an IP address of the suspicious host and RD or name of VRF if necessary, and do not requires pre-defined host database. This nature reduces a load of an administrator in an organization network to establish or periodically update a host database. This nature can then locates even a host that is not registered to such host database. The host locating system is given an IP address of one of routers of L3 switches and VRF in an organization network, and then locates a suspicious host as follows.

(1) connect to the router or L3 switch,
(2) look up a route for an IP address of the suspicious host and VRF,
(3) connect to the nexthop of the route if a route is not *directly connected*,
(4) repeat (2) and (3) until a *directly connected* rout is found, i.e., locate a router that has a *directly connected* route for an IP address of the suspicious host and VRF,
(5) identify a VLAN for the IP address at the router,
(6) locate a *directly connected* router for the IP address on the VRF,
(7) resolve a MAC address of the suspicious host from an Address Resolution Protocol (ARP) [3] table,
(8) identify a port on which the MAC address is seen in a MAC address forwarding table,
(9) discover a neighboring switch on the port,
(10)repeat from (8) to (9), and
(11)finally locate a port on a edge switch accommodating the MAC address.

One can see more detailed pseudo code in **Fig. 1**. As shown in Fig. 1, note that there is a special case where a departmental router is installed and routes are directed to the departmental router, i.e., an organization-wide administrator cannot operate the departmental router, and a MAC address of the actual suspicious host cannot be resolved. In this case, a MAC address of the departmental router should be resolved and the departmental router should be isolated. This allows an organization to flexibly design an organization network.

The host locating system also identifies a responsible person for the suspicious host. In many case, the responsible person may be a user of the suspicious host or an administrator of NAT/NAPT router or a departmental router. The host locating system then identifies a responsible person from logs of a various system that requires a login, e.g., a mail system or Shibboleth IdP/SP.

```
1  def locate_host(core_router, gia)
2          # resolve an internal IP address and VRF.
3          (ip, vrf) = resolve_local_ip_address(gip)
4
5          # find a directly connected router.
6          router = core_router
7          while router do
8                  route = router.lookup_route(ip, vrf)
9                  if route.is_directly_connected?
10                         break
11                 end
12                 # we cannot control user's or
13                 # departmental router.
14                 if not route.nexthop.is_ours?
15                         ip = route.nexthop.ip_address
16                         break
17                 end
18                 router = route.nexthop
19         end
20
21         vlan = route.vlan
22         mac = router.resolve_mac_address(ia, vlan)
23
24         # locate an edge switch and port.
25         sw = router
26         while sw do
27                 port = sw.mac_address_table(vlan, mac)
28                 neighbor = port.get_neighbor
29                 # we cannot control user's or
30                 # departmental router.
31                 if neighbor.nil?
32                         break
33                 end
34                 sw = neighbor
35         done
36         return sw, port
37 done
```

Fig. 1 A pseudo code to locate a suspicious host.

4.2 Host Isolating System

The host isolating system enables to immediately isolate a suspicious host from a network in an organization. The host isolating system has several ways to isolate the suspicious host as shown in **Table 1**. Each method has both good and bad points, which are discussed later.

The host isolating system isolates a suspicious host as follows.

(1) an IP address, a VRF, a VLAN, a MAC address, an edge switch and its port accommodating the suspicious host are given by the host locating system,
(2) connect to a router or edge switch,
(3) shut down a port or filter an IP or MAC address,

Table 1 Methods to isolate a host.

Type	Method	Place	Description
authentication	authentication	authentication server	deauthenticate the host, and do not authenticate the host when connecting a network.
shutdown	physical port	edge	drop all traffics going through the port.
	VLAN (L2)	edge	drop traffics on the VLAN.
	VLAN (L2)	core	do not forward traffics on the VLAN.
	VLAN (L3)	core	do not forward traffics over the different broadcast domain.
filter	MAC address	edge	drop traffics to/from MAC address.
	MAC address	core	do not forward traffics to/from MAC address.
	IP address	edge	drop traffics to/from MAC address.
	IP address	core	do not forward traffic over the different broadcast domain.
	IP address	firewall	drop all traffics to/from the Internet.
	UDP/TCP port	edge	do not forward all traffic.
	UDP/TCP port	core	drop all traffics over the different broadcast domain.
	UDP/TCP port	firewall	drop all traffics to/from the Internet.

(4) quit if a port is not downstream,
(5) compute how to revert a shutdown or filter, and
(6) send an e-mail to an administrator, which includes executed commands and a reverting command.

The host isolating system reverts isolating the suspicious host when an administrator executes a reverting command that is indicated in the e-mail.

4.3 Incident Tracking System

The incident tracking system supports to share information among staffs involved in an incident. This incident tracking system also records actions that staffs involved take and observed phenomenon in order to make an incident trackable. The incident tracking system can then be built using an exiting Bug Tracking System (BTS) or Issue Tracking System (ITS) [4], [6], [7]. The incident tracking system, however, needs to assign a group of staffs involved to an incident, not a personal. This is very different from BTS or ITS. The incident tracking system should hold information as shown **Fig. 2**

5. Implementation

This section presents our first prototype implementation of a computer security incident response system. We have implemented a host locating and host isolating system as scripts written in a Ruby. Our implementation currently supports only two types of isolating methods: port shutdown at an edge switch and MAC address filtering at a core switch. The former is usually for a fixed host in a laboratory that never or rarely moves because this method can avoid a virus spreads into other hosts. On the other hand, the latter is mainly for a mobile host that usually belongs to a student. A student moves around in a campus with a host and, the host moves to connect to different networks and have different IP addresses. In this case, the former method cannot be applied because an incident may be detected for each network. We have used our host locating and isolating system, and it has appeared that out system can locate a suspicious host within 10 seconds after the IP address is given even though a database of all hosts in a network is not maintained. It has also appeared that a suspicious host can be isolate in 10 min. after an external organization reports a security incident suspicion.

We have then implemented incident tracking system using redmine [4]. We have currently just set up a normal redmine, and not modified yet. The current implementation lacks then many functions. The most important function that has not implemented yet is to group members in an organization by department and to send e-mail to them. We then must manually find contact persons and send e-mail. This task is very heavy for a staff when a security incident suspicion occurs because the staff should be in charge of many things. We will implement these lacking functions in the future.

6. Discussions

6.1 Confidentiality of Security Events

We are now planning to automatically isolate a suspicious host when we receive a report from an external organization such as JSOC and NII-SOC that reports an incident by a fixed-formatted e-mail. JSOC and NII-SOC, however, never send detailed information of a security event via an e-mail. They may consider that detailed information is confidential, and should not be sent via an e-mail. An administrator then needs to manually access to their portal sites in order to obtain detailed information. This manual operations results in longer time on an incident response. For example, an external organization usually gives only an global IP address of a suspicious host. In this case, we needs to resolve an internal private IP address when we employ NAPT. In case of authors' environment, all traffic logs of firewall are held, and its size per day ranges from about 10GB to 24GB. It then takes approximately 20 min. to resolve an internal private IP address and VRF. In addition, manual operations to see detailed information prevents us from implementing to automatically isolate a suspicious host.

It may be true that detailed information of a security event may be confidential. We, however, think that an IP address of a suspicious host should be reported in an e-mail for a quick response agains a security incident.

6.2 Host Isolation versus Forensics

A malware may stop a malicious behavior when an infected host is isolated from a network, and authors have already experienced such malwares in the wild. One may say that such isolation makes digital forensics difficult. We, however, cannot help but isolate a suspicious host as soon as possible in order to avoid or reduce possibilities of compromising confidential information.

*1 automatically generated

Table 2 An example of required fields of an issue on an incident tracking system.

Item	Value Type	Description
ID[*1]	integer	monotonically increasing number.
title	string	brief description of an incident.
created time[*1]	timestamp	created time.
updated time[*1]	timestamp	last updated time.
status	list	*open* or *closed*.
type	list	types of incidents: security, physical and so on.
host isolating	list	a suspicious host is isolated or not.
host locating	list	locating or already located.
network	list	types of networks: education, research and so on.
department	list	department that the network belongs to.
user type	list	staffs or students.
user ID	string	user ID of staffs or students.
confidentiality	list	a suspicious host contains confidential data or not.
encryption	list	confidential data is encrypted or not.
description	string	a description of an incident.
external IP address	IP address	an IP address of a corresponding host.
internal IP address	IP address	an IP address of a suspicious host.
MAC address	MAC address	a MAC address of a suspicious host
staffs	list	one of CSIRT members in charge.
department staffs	list	department staffs in charge.
JSOC ticket number	string	JSOC ticket number.
JSOC ticket status	string	*open*, *close*, and so on.
JSOC incident ID	string	multiple JSOC incident IDs associated with JSOC ticket number.
NII-SOC warning ID	string	an ID of a warning.
NII-SOC session ID	string	an ID of a suspicious communication.

6.3 Host Isolation versus Availability

This paper proposes the host isolating system. One may consider that the host isolating system reduces availability. The host isolating system may isolate all hosts in one laboratory when the laboratory employs NAPT. We, however, consider that locating and isolating a suspicious host take precedence over availability because an external organization report is enough accurate so far in authors' environment.

6.4 Host Location Management

This paper proposes the host locating system that requires no pre-defined host database. The proposed system, however, may be unable to locate a suspicious host that quickly or frequently moves. In order to solve this issue, we requires a host database that holds which host is connected to which port on which switch. We are now considering to build a host database using following methods:

- periodical MAC address table dump
- MAC address table change notification
- user authentication
- web authentication
- MAC address authentication
- IEEE802.1x authentication

6.5 Host Isolating Methods

This paper proposes several isolating methods as shown in Table 1. We have then implemented two methods: port shutdown at an edge switch and MAC address filtering at a core switch. From the viewpoint of a security, port shutdown at an edge may be better to confine a suspicious host in a narrow area. For example, a ransomware called *WannaCry* spreads all over the world on May 12th, 2017. *WannaCry* intrudes a computer via not an attachment of an e-mail or phishing but a SMBv1 vulnerability. *Wannacry* may then quickly spread inside an institute once a few computers in the institute are infected. Filtering a traffic from the Internet to an institute may not work for *WannaCry*. Port shutdown may be then suitable for a malware such as *WannaCry*. Port shutdown is, however, not suitable for a suspicious host that quickly or frequently moves. That is, each method has both good and bad points. These good and bad points can be summarize as as shown in **Table 3**.

6.6 Information Error Ratio

When we respond a security incident, information goes through many persons involved, e.g., laboratory staffs, department corresponding staff, CSIRT staffs, CIO, president. When the number of hops that information goes through increases, information errors may increase. We will try to clarify the its error ratio, and reduced error ratio by our proposed system.

6.7 Incident Tracking System

Request Tracker for Incident Response (RTIR) [5] is a famous incident tracking system written in Perl. There are also BTSs or ITSs such as redmine [4] written in Ruby, trac[6] written in Python, mantis[7] written in PHP and so on. We will try to find a best system for our purpose.

7. Related Works

Information Security Management System (ISMS) ISO/IEC-27001[8] briefly defines requirements of computer security incident responses. There are many security or network vendors such as TrendMicro, Paloalto, FireEye, Fortigate, Cisco, Alaxala and so on try to produce the best security solutions.

NAGAI, Y. et al. investigated and reported differences between ISMSs in national universities in Japan[9]. They also presented their own incident management system using trac[6]. They then reported that their system could record information of only about a half of all security events because some of those events were

Table 3 Pros and cons of isolating methods.

Method	Place	Pros	Cons
auth.	auth. server	centeralized control.	all switch should be configured to authenticate a host.
physical port	edge	confine a malware into a restricted area.	all hosts connected to the port are isolated.
VLAN (L2)	edge	confine a malware into a restricted area.	all hosts on the same VLAN are isolated.
VLAN (L2)	core	centeralized control.	all hosts on the same VLAN still can communicates.
VLAN (L3)	core	centeralized control.	all hosts on the same VLAN still can communicates.
MAC address	edge	confine a malware into a restricted area. only a suspicious host is isolated.	a mobile node cannot be supported. some switches cannot support.
MAC address	core	centeralized control. support a mobile node.	a host can still communicate between downstream switches.
IP address	edge	confine a malware into a restricted area.	a mobile node cannot be supported.
IP address	core	centeralized control. support a mobile node.	a host can still communicate on the same VLAN. .
IP address	firewall	centeralized control. support a mobile node.	a host can still communicate all hosts in an organization.
UDP/TCP port	edge	confine a malware into a restricted area.	a mobile node cannot be supported.
UDP/TCP port	core	centeralized control. support a mobile node.	unknown malicious communications may succeed in.
UDP/TCP port	firewall	centeralized control. support a mobile node.	a host can still communicate all hosts in an organization.

reported or discussed in meetings and their data was never input to the system.

HASEGAWA, H. et al. proposes the supporting system against an incident caused by targeted attacks [10]. Their system automatically suggests 9 types of access filtering across VLANs to an administrator in accordance with a severity of an incident when a network configuration is pre-defined and given. They, however, assumes only filtering across VLANs, and do not consider the case where there is a router run by a department, not a information infrastructure department that is in charge of a management of a campus wide network. In addition, they do not consider a mobile host that moves around while our proposal do.

8. Concluding Remarks

This paper has proposed a computer security incident response system that automatically locates and isolate a suspicious host. Our first prototype implementation has shown that a suspicious host can be isolated within 10 min. right after an external organization reports a security incident suspicion. Before our system, we had spent more than one hour to do the same isolation. We will improve our system in the future.

References

[1] LAC Co., Ltd: Japan Security Operation Center(JSOC®) — Services and Products — LAC Co., Ltd., https://www.lac.co.jp/english/service/operation/jsoc.html (1995). Accessed: 2017/05/26.
[2] IEEE Std. 802.1ab-2004: *Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks: Station and Media Access Control Connectivity Discovery* (2004).
[3] Plummer, D.: Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware, RFC 826 (Standard) (1982). Updated by RFCs 5227, 5494.
[4] Lang, J. P.: Overview - Redmine, http://www.redmine.org/ (2006). Accessed: 2017/05/19.
[5] Best Practical Solutions, L.: RT for Incident Response, https://bestpractical.com/rtir/ (2002). Accessed: 2017/05/19.
[6] Software, E.: The Trac Project, https://trac.edgewall.org/ (2003). Accessed: 2017/05/19.
[7] MantisBT Team: Mantis Bug Tracker, https://www.mantisbt.org/ (2000). Accessed: 2017/05/19.
[8] ISO/IEC: Information Security Management Systems Requirements (2013). ISO/IEC27001:2013.
[9] NAGAI, Y., TADAMURA, K. and OGAWARA, K.: Considering Incident Management Systems in Some National Universities, *SIG Technical Reports*, Vol. 2014-IS-127, No. 7, pp. 1–7 (2014).
[10] Hasegawa, H., Yamaguchi, Y., Shimada, H. and Takakura, H.: A Countermeasure Support System against Incidents caused by Targeted Attacks, *Journal of Information Processing*, Vol. 57, No. 3, pp. 836–848 (2016).