CHAPTER 6: EGA TECHNIQUE FOR CLOUD WORM RESPONSE

6.1 Introduction

Since twenty years ago, experts working in incident response field were taught that it is better to shutdown an entire system in case of a breach. Today, with cloud computing surrounding us, this task is impossible and there is a need for new approaches. For example, in a virtual machine, memory can store important data for investigators, such as network connections and running processes in case of a power failure.

Given these realities, it is necessary to handle incident response and reporting with great care. Incident responders and forensic investigators must now rely more and more on live collection and analysis of system RAM. Live response analysis has been used for the last six to eight years only in important high-end cases, under the command of highly qualified computer forensics specialists, but modern reality forces this approach to almost all data cases.

The current tools that are used in forensic analysis work in a simple way: you upload a remote agent to the target system, wait for it, and gather snapshots of data from memory, and send it back to the investigator's computers, where a human or a software program reviews them.

Furthermore, forensic investigators need to avoid copying the entire disk images. Somehow, in today's cloud environments, data can easily reach large sizes and it is not always technically possible or cost effective to do it. Even more, judges, agents and investigators were classically taught that only a perfect data copy is a "forensically sound" copy that can be used. This must change because there is no need to copy an entire disk image in order to retrieve only a few files representing evidence. In order to accomplish this, the forensic investigators will need also a way to view and analyse "live remote" data.

6.2 Security Metrics

Security metrics help to understand what metrics are by drawing a distinction between metrics and measurements. Measurements provide single point in time views of specific and discrete factors, while metrics are resultant from the comparison of a predetermined reference point. Metrics are generated from analysis, and are either objective or subjective human interpretations of data, while measurements are objective raw data. Good metrics must have SMART capabilities, i.e. specific, measurable, attainable, repeatable, and time-dependent (Payne, 2006).

Truly useful metrics indicate the degree to which security goals, such as data confidentiality, are being met, and they derive the actions taken to improve an organisation's overall security program. Distinguishing metrics are meaningful primarily to those with direct responsibility for security management from those that speak directly to executive management interests. Issues are critical to the development of an effective security metrics program.

Metrics can be an effective tool for security managers to discern the effectiveness of various components of their security programs, the security of a specific system, product or process, and the ability of staff or departments within an organisation to address security issues for which they are responsible. Metrics can also help identify the level of risk in not taking a given action, and in that way, provide guidance in prioritising the corrective actions. Additionally, they may be used to raise the level of security awareness within the organisation. Finally, with knowledge gained through metrics, security managers can better answer hard questions from their executives and others, such as:

- Are the systems more secure today than they were before?
- How could it be compared to others in this regard?
- Are the current computing environments secured enough?

Many research works focused on security metrics to measure threats and attack power. These researches are dedicated to various parameters like: rogue network (Stone-Gross *et al.*, 2009), number of zero-day attacks (Fossi, 2011), attack rate (Zhan *et al.*, 2013; Zhan *et al.*, 2015), exploit kits (Ablon *et al.*, 2014), cyber security posture, and

sweep time including malicious network (Zhang *et al.*, 2014) and packer structural complexity (Ugarte-Pedrero *et al.*, 2015). This work defined security metric by measuring risk level, followed by the capacity of damage by worm in cloud.

Many in the security industry agreed that the number of successful security attacks an organisation has experienced is not necessarily an indication of how secure that organisation is. Luck plays a major role, and how does one measure luck? So, a security manager needs to look beyond the organisation's security incident record for determining the indicators of security strength. In fact, there are further complications they need to keep in mind in their search for meaningful metrics.

6.2.1 Cloud Worm Weight and Severity Measuring

Weight is measured based on the security level of five main features namely: infection, activation, payload, propagation, and operating algorithm. Based on CIA (Confidentiality, Integrity and Availability), weight value is defined. Weight and severity values are justified by the CIA, and based on the weight value, severity value is also defined.

6.2.2 CIA Constraints

Confidentiality is roughly equivalent to privacy. Measures undertaken to ensure confidentiality are designed to prevent sensitive information from reaching the wrong people, while making sure that the right people can in fact get it: Access must be restricted to those authorised to view the data in question. It is common, as well, for data to be categorised according to the amount and type of damage that could be done should it fall into the unintended hands. More or less stringent measures can then be implemented according to those categories.

Integrity involves maintaining the consistency, accuracy, and trustworthiness of data over its entire life cycle. Data must not be changed in transit, and steps must be taken to ensure that data cannot be altered by unauthorised people (for example, in a breach of confidentiality). These measures include file permissions and user access controls. Version control maybe used to prevent erroneous changes or accidental deletion by

Table 6.2: Ranking based on potential loss.

Asset	**Loss of Confidentiality**	**Loss of Integrity**	**Loss of Availability**
Customer database	High	High	Medium
Internal documents	Medium	Medium	Low
Advertising literature	Low	Medium	Low
Classified records	High	High	Medium

6.2.3 Features Threat Score

In this research, cloud worms are classified into infection, activation, payload, operating algorithm and propagation. In addition, this research defined the features of these five classes. These classifications are presented in Table 6.3 above.

Each of the features in Table 6.4 is given a threat score value adapted from the works done by Dini *et al.*, (2016) and Ross (2012) based on the impacts of infection, activation, payload, operating algorithm, and propagation as shown in Table 6.4 below.

Table 6.3: Impact Level and Threat Score.

Likelihood of Impact	Capabilities	Threat Score
Low	The capabilities of risks sources to perform a threat are low or do not exist.	0.4
Moderate	The capabilities of risks' sources to carry out a threat are moderate.	0.7
High	The capabilities of risks sources to carry out a threat are real and high.	1

6.2.4 Computation of Cloud Worms Risk and Level

In this phase, each cloud worm risk's scores and levels are calculated. The first step is to calculate each worm's risk score toward infection, activation, payload, operating

algorithm, and propagation based on Formula (1), Formula (2), Formula (3), Formula (4), and Formula (5) shown below:

$$\text{RSI} = \frac{\sum_{i=1}^{F_{fI}} TSI_i}{F_{fI}} \quad (1)$$

$$\text{RSA} = \frac{\sum_{i=1}^{F_{fA}} TSA_i}{F_{fA}} \quad (2)$$

$$\text{RSP}_y = \frac{\sum_{i=1}^{F_{fP_y}} TSP_{y_i}}{F_{fP_y}} \quad (3)$$

$$\text{RSO} = \frac{\sum_{i=1}^{F_{fO}} TSO_i}{F_{fO}} \quad (4)$$

$$\text{RSP}_r = \frac{\sum_{i=1}^{F_{fP_r}} TSP_{r_i}}{F_{fP_r}} \quad (5)$$

Where: RSI = Risk score for Infection.

RSA = Risk score for Activation.

RSP_y = Risk score for Payload.

RSO = Risk score for Operating Algorithm.

RSP_r = Risk score for Propagation.

TSI = Threat score for Infection.

TSA = Threat score for Activation.

TSP_y = Threat score for Payload.

TSO = Threat score for Operating Algorithm.

TSP_r = Threat score for Propagation.

F_{fI} = Frequency of Infection feature.

F_{fA} = Frequency of Activation feature.

F_{fP_y} = Frequency of Payload feature.

F_{fO} = Frequency of Operating Algorithm feature.

F_{fP_r} = Frequency of Propagation feature.

Further, the Total Risk Score (TRS) of a cloud worm can be calculated based on Formula (6) below:

$$TRS = [(RSI \times Infection\_Load) + (RSA \times Activation\_Load) + (RSP_y \times Payload\_Load) + (RSO \times Operating\ Algorithm\_Load) + (RSP_r \times Propagation\_Load)] \times 100 \quad (6)$$

The worm's Total Risk Score (TRS) is normalised based on infection, activation, payload, operating algorithm, and propagation. In order to calculate the Load for each feature of Infection_Load, Activation_Load, Payload_Load, Operating Algorithm_Load, and Propagation_Load, Formula (7) as follows is employed:

$$\text{The Load for each feature} = \frac{\text{Total Risk Score for each feature}}{\text{Total Risk Score for all features}} \quad (7)$$

Based on Formula (7), the load of infection is equal to 0.388, the load of activation is equal to 0.124, the load of payload is equal to 0.279, the load of operating algorithm is equal to 0.155, and the load of propagation is equal to 0.051.

Then, the Total Risk Score (TRS) for all features = Total Risk Score for Infection + Total Risk Score for Activation + Total Risk Score for Payload+ Total Risk Score for Operating Algorithm + Total Risk Score for Propagation. Hence, Formula (6) can be rewritten as Formula (8) below:

$$TRS = [(RSI \times 0.388) + (RSA \times 0.124) + (RSP_y \times 0.279) + (RSO \times 0.155) + (RSP_r \times 0.051)] \times 100 \quad (8)$$

6.2.5 Rules for defining weight

During the process, CIA also undertook to perform the calculation as presented in section 6.3.2. Rules are defined to get the value of weight considering the threat level of five features; for example, if infection involves any activities, which can initiate attack using root privilege, then the weight is high. On the other hand, if infection involves any activities, which can initiate attack using other users' resources except root privilege, then the weight is medium. Similarly, if infection does not involve any activities, which can initiate attack using root privilege and using other users' resources, then the weight is low. All rules generated in similar way and are presented as follows.

Rules for defining weight

1. If infection involves any activities, which can initiate attack using root privilege, then the weight is high.
2. If infection involves any activities, which can initiate attack using other users' resources except root privilege, then the weight is medium.
3. If infection does not involve any combination featuring in rules 1 and 2, then the weight is low.
4. If infection involves any activities related to visualisation, then the weight is high.
5. If infection involves any activities related to communication, then the weight is low.
6. If infection involves any activities related to application, then the weight is low.
7. If infection involves any combination featuring in rules 4 and 5, then the weight is medium.
8. If infection involves any combination featuring in rules 4, 5 and 6, then the weight is medium.
9. If infection involves any combination featuring in rules 1 and 7, then the weight is high.

10. If infection involves any combination featuring in rules 1 and 8, then the weight is high.
11. If infection involves any combination featuring in rules 1 and 6, then the weight is medium.
12. If infection involves any combination featuring in rules 2 and 7, then the weight is medium.
13. If infection involves any combination featuring in rules 2 and 8, then the weight is medium.
14. If infection involves any combination featuring in rules 2 and 6, then the weight is low.
15. If activation involves a human trigger, then the weight is medium.
16. If activation involves a scheduled process, then the weight is medium.
17. If activation involves a self activation, then the weight is high.
18. If activation involves any combination featuring in rules 15 and 16, then the weight is medium.
19. If activation involves any combination featuring in rules 15, 16 and 17, then the weight is medium.
20. If activation involves any combination featuring in rules 15 and 17, then the weight is high.
21. If activation involves any combination featuring in rules 16 and 17, then the weight is high.
22. If payload involves any activities related to backdoor, then the weight is high.
23. If payload involves any activities, which launched denial of service attack, then the weight is high.
24. If payload involves any activities, which destructed any kind of system resources, then the weight is high.
25. If payload involves any activities, which steal the system related information, then the weight is high.
26. If payload involves any activities related to phishing, then the weight is medium.
27. If payload involves any malicious activities related to registry alternation, then the weight is medium.

28. If payload involves any combination featuring in rules 22, 23, 24, 25, 26 and 27, then the weight is high.
29. If payload involves any one combination featuring in rules 22, 23, 24 and 25 with any one combination featuring in rules 26 and 27, then the weight is medium.
30. If payload involves any two or more combination featuring in rules 22, 23, 24 and 25 with any one combination featuring in rules 26 and 27, then the weight is high.
31. If payload involves any two or more combination featuring in rules 22, 23, 24 and 25 with any combination featuring in rules 26 and 27, then the weight is medium.
32. If operating algorithm involves any activities related to stealth, then the weight is high.
33. If operating algorithm involves any activities related to polymorphic, then the weight is high.
34. If operating algorithm involves any activities related to anti anti-virus, then the weight is high.
35. If operating algorithm does not involve any activities, then the weight is low.
36. If operating algorithm involves stealth, polymorphic and anti anti-virus activities, then the weight is high.
37. If operating algorithm involves any one combination featuring in rules 32, 33 and 34, then the weight is high.
38. If operating algorithm involves any two combinations featuring in rules 32, 33 and 34, then the weight is high.
39. If propagation involves random scanning activities, then the weight is high.
40. If propagation does not involve random scanning activities, then the weight is low.

6.2.6 Rules for defining severity

Severity scheming is based on two or more weight values of five main features. For example, if the weight for the infection is low and the payload is medium, then the

severity is low. In a similar way, if the weight for the infection is low and the payload is high, then the severity is high. Moreover, if the weight of the infection is medium and the payload is medium, then the severity is medium. All rules generated for the severity calculation is presented below.

Rules for defining severity

1. If the weight for the infection is low and the payload is medium, then the severity is low.
2. If the weight for the infection is low and the payload is high, then the severity is high.
3. If the weight for the infection is medium and the payload is medium, then the severity is medium.
4. If the weight for the infection is high and the payload is high, then the severity is high.
5. If the weight for the infection is high and the payload is medium, then the severity is high.
6. If the weight for the infection, activation, payload, operating algorithm and propagation is high then the severity is high.
7. If the weight for the infection, activation, payload and propagation is high and the weight of operating algorithm is low then the severity is high.
8. If the weight for the infection, activation, operating algorithm and propagation is high and the weight of payload is medium then the severity is high.
9. If the weight for the infection is high, activation is high, payload is medium, operating algorithm is high and propagation is low then the severity is high.
10. If the weight for the infection is high, activation is medium, payload is high, operating algorithm is high and propagation is high then the severity is high.
11. If the weight for the infection is high, activation is medium, payload is high, operating algorithm is high and propagation is low then the severity is high.
12. If the weight for the infection is high, activation is medium, payload is high, operating algorithm is low and propagation is high then the severity is high.

13. If the weight for the infection is high, activation is medium, payload is medium, operating algorithm is high and propagation is high then the severity is high.
14. If the weight for the infection is high, activation is medium, payload is medium, operating algorithm is high and propagation is low then the severity is high.
15. If the weight for the infection is low, activation is high, payload is high, operating algorithm is high and propagation is high then the severity is high.
16. If the weight for the infection is low, activation is high, payload is high, operating algorithm is high and propagation is low then the severity is high.
17. If the weight for the infection is low, activation is high, payload is medium, operating algorithm is high and propagation is low then the severity is low.
18. If the weight for the infection is low, activation is medium, payload is high, operating algorithm is high and propagation is high then the severity is high.
19. If the weight for the infection is low, activation is medium, payload is medium, operating algorithm is high and propagation is high then the severity is low.
20. If the weight for the infection is low, activation is medium, payload is medium, operating algorithm is high and propagation is low then the severity is low.
21. If the weight for the infection is medium, activation is high, payload is high, operating algorithm is high and propagation is high then the severity is high.
22. If the weight for the infection is medium, activation is high, payload is high, operating algorithm is high and propagation is low then the severity is high.
23. If the weight for the infection is medium, activation is high, payload is high, operating algorithm is low and propagation is high then the severity is high.
24. If the weight for the infection is medium, activation is high, payload is medium, operating algorithm is high and propagation is high then the severity is medium.
25. If the weight for the infection is medium, activation is high, payload is medium, operating algorithm is high and propagation is low then the severity is medium.
26. If the weight for the infection is medium, activation is medium, payload is high, operating algorithm is high and propagation is high then the severity is high.

27. If the weight for the infection is medium, activation is medium, payload is high, operating algorithm is high and propagation is low then the severity is high.
28. If the weight for the infection is medium, activation is medium, payload is high, operating algorithm is low and propagation is high then the severity is high.
29. If the weight for the infection is medium, activation is medium, payload is medium, operating algorithm is high and propagation is high then the severity is medium.
30. If the weight for the infection is medium, activation is medium, payload is medium, operating algorithm is high and propagation is low then the severity is medium.

Following the concept presented in Tables 6.1 and 6.2, CIA and risk level are generated for all features. The result of the calculation is presented in Table 6.4 based on the definition of features threat score at section 6.2.3.

Table 6.4: Ranking based on potential loss

Classification	Sub-Classification	Sub-sub Classification	CIA	Level	Threat Score
Infection	Authorized User (Insider)	Root privilege	Confidentiality and Integrity and Availability	H	1
		Other users resources	Confidentiality and Integrity	M	0.7
	User using virtualisation	Hypervisor	Confidentiality and Integrity and Availability	H	1
		VM image sharing	Confidentiality and Integrity and Availability	H	1
		VM Migration	Confidentiality and Integrity and Availability	H	1
		VM Rollback	Confidentiality and Integrity and Availability	H	1
		VM Isolation	Confidentiality and Integrity and Availability	H	1
	Communication		Confidentiality and Availability	L	0.4
	Application		Confidentiality and Availability	L	0.4

Activation	Human Trigger	Confidentiality, Integrity and Availability	M	0.7
	Scheduled Process	Confidentiality, Integrity and Availability	M	0.7
	Self Activation	Confidentiality and Integrity and Availability	H	1
Payload	Backdoor	Confidentiality and Integrity and Availability	H	1
	Denial of Services (DoS)	Confidentiality and Integrity and Availability	H	1
	Destructive	Confidentiality and Integrity and Availability	H	1
	Steal Information	Confidentiality and Integrity and Availability	H	1
	Phishing	Confidentiality, Integrity and Availability	M	0.7
	Registry Shuffling	Confidentiality, Integrity and Availability	M	0.7
Operating Algorithm	Stealth	Confidentiality and Integrity and Availability	H	1
	Polymorphic	Confidentiality and Integrity and Availability	H	1
	Anti anti-virus	Confidentiality and Integrity and Availability	H	1
Propagation	Scanning	Confidentiality and Integrity and Availability	H	1

The results obtained in the table 6.4 show the threat level of each worm features or classification. For example, in relation to "Root privilege" located under "infection," it is known that worm has the ability to sniff password and gain access to legal users' account. Through this process, it is able to infiltrate the root cloud system causing serious damages to the cloud system. In this situation, based on the CIA rule, the severity of this incident level is high.

On the other hand, in relation to other worm classifications such as "Human Trigger" located on the "activation" classification, worm is caused by user's action through running an application or inserting CD and memory card. It can also be caused by clicking on an unknown link that can lead to cloud been infected by malicious worm. In this situation, based on the CIA rule, the damage level is medium.

Additionally, another example is related to communication under "infection" classification; it requires cloud users to communicate with cloud system for file

sharing. In this situation, if users do not communicate, no suspicious incident takes place. Therefore, based on the CIA rule, the severity is low. Likewise, all results obtained for other worm features related to the threat level follows the same CIA rule.

In Table 6.3, the CIA level of Infection through communication and application is low. It means that the communication and application attacker can get the interface to initiate attack which reduces confidentiality and integrity. However, attackers are unable to attack cloud system unless they have authorised permission to the cloud system.

6.3 New Cloud Worm Response Algorithm

For this research, a new algorithm for response was proposed. It is a new algorithm based on the suggestion that was initially proposed after the detection process. No other work is carried out on worm response in cloud environment. Hence, the proposed algorithm is claimed as a new algorithm in cloud worm response domain. The threat level of worm was studied based on the rule of Confidentiality, Integrity and Availability (CIA) which was also derived from a researcher known as Swanson (2001) and Gregg (2005).

Many researchers used the CIA rules to define security metrics (Kim, 2013; Pant & Khairnar, 2014; Elmrabit, 2015). Table 6.1 shows the rules for CIA. The CIA shows and explains the level of damage that may be performed by malicious cloud worm. It helps by assisting against prevention systems from serious damages. It breaks the threat level into three major categories which are: "High," "Medium" and "low." Through these threat level rules, the proposed algorithm for this research was developed which is presented and explained below.

This research suggested isolating the infected host by turning it off, following the pseudo code initiated once the infected host is identified.

```
1 Input: Infected_Host_id      Output: Action
2       If infected_Host_id=infected then
3       |       checkThreatLevel()
4       |       |       If tL= "Low"
5       |       |       |       Action = Ignore()
6       |       |       elseif tL = "Medium"
7       |       |       |       Action = Monitor()
8       |       |       elseif tL = "High"
9       |       |       |       Action = Shutdown()
10      |       |       else
21      return
```

Figure 6.1: Pseudo code 5 (Response)

Figure 6.2 presents how the system will respond once a worm is detected. If the threat level is low, then the system will ignore it. In the case of medium threat level, the system will be under monitor. However, if high threat level is found, the system will shut down immediately. Due to the immediate shutdown, service interruption will happen until an assigned task is initiated to a new virtual host. It is suggested that service interruption for some time is good enough from dragging the whole cloud system under threat by worm attack. The proposed algorithm works on the intended security metrics which define the threat level.

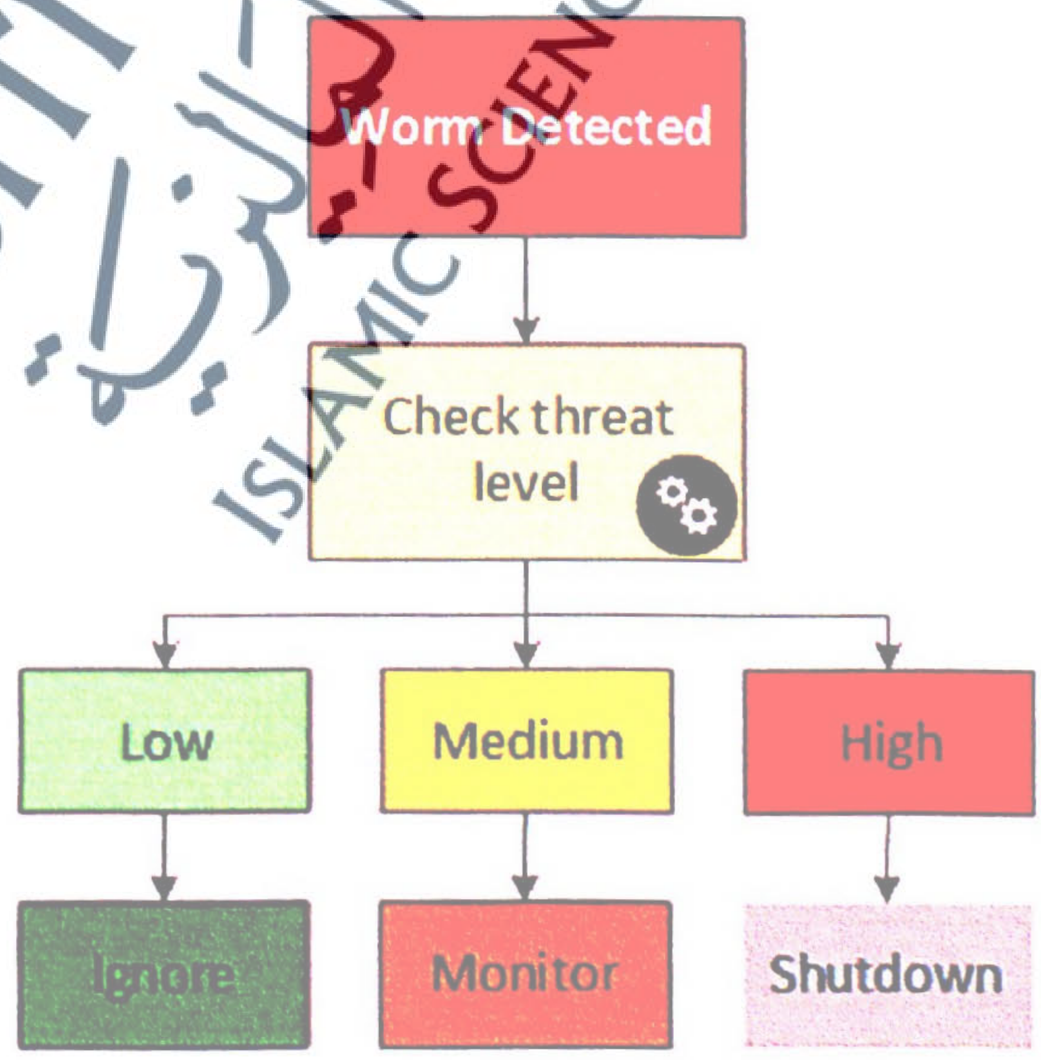

Figure 6.2: Cloud worm response upon detection based on threat level.

6.4 Summary

Based on the weight value, severity value also defined and explained the worm threat level as high, medium and low. The suggestion of EGA response algorithm is that the system shuts down immediately if high threat level is found. However, the system ignores it if the threat level is low. Furthermore, system monitoring should be performed in the case of medium threat level.