Including Security Monitoring in Cloud Service Level Agreements

Amir Teshome
Email: amir-teshome.wonjiga@inria.fr
Inria, IRISA

Louis Rilling
Email: louis.rilling@irisa.fr
DGA

Christine Morin
Email: christine.morin@inria.fr
Inria, IRISA

1. Introduction

One of the risks of moving to a public cloud is losing full control of the information system infrastructure. The service provider will be in charge of monitoring the physical infrastructure and providing the required service to clients. This pushes clients to have trust on providers. Service providers give assurance on some aspects of the service but, as of today, security monitoring is not one of them. In our work, we aim to allow providers to provide customers with guarantees on security monitoring of their outsourced information system.

We focus our work on security monitoring in clouds. *Security Monitoring* is the collection, analysis, and escalation of indications and warnings to detect and respond to intrusions. By monitoring a system it is possible to detect suspicious behaviors and take action before severe damage. Intrusion Detection Systems (IDS) and logs from firewalls are often used for this purpose.

A Service Level Agreement (SLA) is a contract between clients and service providers. SLAs describe the provided service, the rights and obligations of both parties and state penalties for when the specified terms are not respected. Hence, SLAs help providers to build more trust.

To include security monitoring terms into an SLA the following tasks are required, (i) a way for providers/clients to specify their security monitoring parameters/requirements, (ii) mechanisms to enforce these requirements in a cloud infrastructure and (iii) a verification method to check if the requirements are respected at any given time.

In the rest of this paper, Section 2 presents related work, Section 3 describes challenges to be addressed, Section 4 presents our approach, and Section 5 concludes the paper.

2. Related works

There exists some works on both creating security monitoring devices for a cloud [6] and defining languages and frameworks for SLA description [3]. The domain specific language proposed in [6] describes the detection algorithms of the IDS rather than Service-Level Objectives (SLO), for example a set of rules that can be negotiated before figuring in an SLA. In other words, the language is too low-level to describe SLOs.

To our knowledge, there have been no attempts to include security monitoring terms in SLAs. The difficulty is that SLA terms related to security monitoring devices need to be verifiable in the cloud setup.

Regarding IDS verification metrics, Stefan Axelsson [1] and Gu *et al* [2] presented a theoretical approach to measure IDSs and showed that metrics that don't include base rates (defined as the ratio between the number of attack network packets and the total number of network packets) do not truly describe the ability of an IDS to be practically usable. This problem is known as the *Base-Rate Fallacy* [1]. The latter one proposed a single unified metrics called Intrusion Detection Capability (C_{ID}).

To verify an IDS, Probst *et al.* [5] describe a method in two phases: an analysis of network access control followed by the IDS evaluation based on the set of services running in the virtual infrastructure. Before this work, Massicotte *et al.* [4] used a virtual infrastructure to generate traffic traces and used the traces to evaluate IDSs in traditional servers (non-cloud environment). Both approaches measure the efficiency of an IDS, the former in a given virtualized infrastructure and the latter as a generic product, but neither of them take the *base rate* into account.

3. Challenges

Including security monitoring terms in SLAs raises a number of technical challenges, which include:

1) The malleability of virtualized infrastructures: by its nature the cloud is very dynamic. Creation, deletion and migration of VMs is frequent. Security monitoring terms must anticipate such changes.
2) There is no standard to express precise security monitoring properties independently from the actual devices used.
3) To our knowledge, there is no method to automatically configure a set of security monitoring devices according to an abstract policy.
4) There is a lack of method to evaluate security monitoring setups specifically in clouds.

In this paper we present preliminary work to address two of the challenges (the first and the last ones). The remaining ones are left as a future work.

4. Proposed Approach

A design requirement is to make the security monitoring process - definition, enforcement and verification of SLA terms - automatic. Indeed, manual management of security properties in a cloud is tedious and error prone.

The baseline of our approach is first to find verification mechanisms for security monitoring setups. This will give insights in the expected efficiency of different strategies to setup security monitoring. From these insights, we should be able to propose heuristics for automatically computing security monitoring setups out of SLA terms.

Our first study focuses on IDSs. First, we found measurable parameters for a network IDS, and verification mechanisms for these parameters. *Intrusion Detection Capability* is a single unified metric, which aggregates a base rate in its formula in addition to other traditionally used parameters like detection and precision rate. Since the exact value of the base rate can't be known in a production environment, we used a range of statistically proposed values.

4.1. Intrusion Detection Capability (C_{ID})

C_{ID} is a metric used to evaluate IDSs, which was introduced by Gu *et al* [2]. Let *'x'* be the random variable representing the IDS input packets where it can be either an attack or a legitimate packet and *'y'* representing the IDS output where it can be detected as an intrusive or non-intrusive packet by the IDS.

Note: ***base rate (B)*** *is p(x='is an attack packet')*

- Let *H(x)* indicates the entropy of *x* as defined in information theory and
- The mutual information *I(x;y)* which measures the amount of information shared between the two random variables, then
- Intrusion Detection Capability (C_{ID}) can be defined as:

$$C_{\mathrm{ID}} = \frac{I(x;y)}{H(x)}$$

Its value ranges in [0,1] and a higher value indicates a better IDS ability in accurately classifying the input packets.

4.2. Verification Mechanism

The verification mechanism runs attacks against a given configuration but without damaging the production environment. An example of the attack running environment is shown in Figure 1.

In a given infrastructure we add a target VM (shown in a green box) after an IDS to be verified. This VM exhibits the behavior of the other VMs monitored by the verified IDS. Multiple target VMs could also be added in a case where a single VM is unable to exhibit all the required behaviors. An attacker machine (virtual or physical) is also added. This machine could be located inside or outside the cloud. The attacker runs a set of representative attacks and the virtual

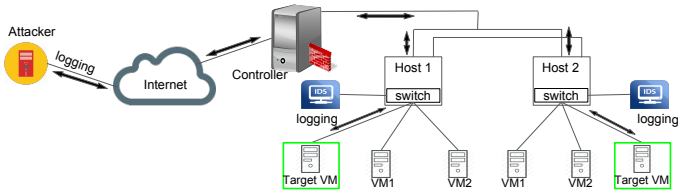

Figure 1. Attack Running Environment

switch is configured to redirect all the attack packets towards the target VM. Since the attack running mechanism uses the production infrastructure network resources, we must take care that the attacks have a reasonably low impact on those resources.

The rate of the occurrence of attack packets is determined by a given base rate. In this process all the outgoing packets from the attacker and the output of the IDS are logged. Using information from the attack packets we can differentiate true positives from false positives in the output of IDS. Using these values and the injected base rate we calculate the C_{ID}.

5. Conclusion and Future Work

In our work we chose the C_{ID} as a usable metric to describe the efficiency of an IDS because it takes the base rate into account. We also presented an evaluation method to measure C_{ID} of an IDS dynamically using attack injection.

The attack packets are redirected in order not to damage the production VMs. But there is a trade-off between the evaluation methodology and the performance of the production infrastructure. A care should be taken since the evaluation process uses the production network infrastructure (not a cloned or simulated one). In particular it needs caution for not creating unacceptable traffic load.

As future work we plan to extend this work to other monitoring probes (e.g firewalls). C_{ID} is used to describe efficiency of IDSs, other aspects of IDS could also be included into SLAs. Definition and enforcement of SLAs are also part of future work.

References

[1] S. Axelsson. The base-rate fallacy and its implications for the difficulty of intrusion detection. In *Proc. CCS*, 1999.

[2] G. Gu, P. Fogla, D. Dagon, W. Lee, and B. Skorić. Measuring intrusion detection capability: an information-theoretic approach. In *Proc. ASIACCS*, 2006.

[3] K. T. Kearney, F. Torelli, and C. Kotsokalis. SLA: An abstract syntax for Service Level Agreements. In *Proc. GRID*, 2010.

[4] F. Massicotte, F. Gagnon, Y. Labiche, L. Briand, and M. Couture. Automatic evaluation of intrusion detection systems. In *Proc. ACSAC*, 2006.

[5] T. Probst, E. Alata, M. Kaâniche, and V. Nicomette. Automated Evaluation of Network Intrusion Detection Systems in IaaS Clouds. In *Proc. EDCC*, 2015.

[6] D. Riquet, G. Grimaud, and M. Hauspie. DISCUS: A massively distributed IDS architecture using a DSL-based configuration. In *Proc. ISEEE*, 2014.