A misuse pattern for Denial-of-Service in federated Inter-Clouds

Oscar Encina[†], Eduardo B. Fernandez* and Raúl Monge[†]
[†]Department of Computer Science,
Universidad Técnica Federico Santa María, Valparaíso, Chile
*Department of Computer and Electrical Engineering and Computer Science,
Florida Atlantic University, Boca Raton, FL 33431, USA
[†]oencina@inf.utfsm.cl, *ed@cse.fau.edu, [†]rmonge@inf.utfsm.cl

Abstract

We have proposed previously a new type of pattern, the misuse pattern. A misuse pattern describes how a misuse is performed from the point of view of the attacker, what system units it uses and how, provides ways of stopping the attack by enumerating possible security patterns that can be applied for this purpose, and provides forensic information. A catalog of misuse patterns is needed to let designers evaluate their designs with respect to possible threats. *Inter-Cloud* systems are growing in popularity, and some of them are federated *Inter-Cloud*s, which allow their *Service Providers* to share resources when needed. We present here a misuse pattern for a generic *Denial-of-Service* attack for federated *Inter-Cloud* systems. A *Denial-of-Service* misuse of this kind tries to disrupt the availability of the *Inter-Cloud* system by making many resource requests or by interrupting the monitoring of compliance agreements between *Consumers* and *Service Providers*.

Keywords: misuse patterns, Inter-Clouds, security patterns, Denial-of-Service, federated Inter-Clouds

Introduction

The vision of computing as a utility, where customers pay only for what they use exists since the 60s [Buy09]; however, this vision has come true just a few years ago with the advent of *Cloud Computing*. This technology provides clients with various computing services without the need to acquire technological infrastructure and let them pay only for the amount of services they use, freeing them from expensive technology purchases and maintenance. All these problems are delegated to the supplier, who worries about the continuous delivery of services. *Cloud Computing* caused an increase in demand for such services, because the customer is offered an apparently infinite amount of resources at a low price with a completely outsourced management service. However, this seemingly endless source of computing resources is tied to the *Service Provider (SP)* size, and there are only a few providers that actually can provide huge amounts of resources. *SPs* must ensure that there are sufficient resources available to them in case that the demand increases unexpectedly. As an answer to the above problem, federated *Cloud Computing* emerged, better known under the name of *federated Inter-Cloud* [Gro12, Ber10, Buy10], and that refers to the agreement between *SPs* to share resources in order to increase their computational resources and provide a larger variety of services.

Misuse patterns describe, from the point of view of the attacker, how a type of attack is performed (what units it uses and how), considers the ways of stopping the attack by enumerating possible security patterns that can be applied for this purpose, and describes how to trace the attack once it has happened by appropriate collection and observation of forensics data. It also describes precisely the context where the attack may occur. A catalog of misuse patterns is needed to let designers evaluate their designs with respect to possible threats.

In this paper we present a misuse pattern for a generic *Denial-of-Service* attack for federated *Inter-Cloud* systems. A *Denial-of-Service* misuse of this kind tries to disrupt the availability of the *Inter-Cloud* system by making many resource requests or by interrupting the monitoring of compliance agreements between *Consumers* and *Service Providers*. We have previously developed a *federated Inter-Cloud* pattern [Enc14], embracing most relevant *Inter-Cloud* proposals [Ber10, Buy10, Kec12, Gro12]. That pattern will be used in this work.

Denial-of-Service in federated Inter-Cloud

Intent

A DoS attack may activate the request of many resources, which can exhaust the resources of federation thus denying legitimate users the use of these resources; or create a flood of messages for disrupting the monitoring of compliance agreements between *Consumers* and *Service Providers*.

Context

In an *Inter-Cloud* system, resources are shared by multiple *Service Providers* who belong to a Federation. A *Service Provider* must belong in advance to the Federation to request resources through the Internet. The requested resource can be at the *SaaS*, *PaaS*, or *IaaS* level. The *Inter-Cloud* is organized using a centralized topology. A *Cloud Exchange* is the component where every request and assignment are bound. Requests are not made directly to the *Cloud Exchange,* but first pass through a broker. The Federation has insufficient (or even none) regulation(s) for accepting new *Service Providers or users*.

Problem

To perform some types of misuse it is necessary to have an account in one or more *Service Providers* who belong to the Federation. How can the attacker deny access to others consumers? The attacker could request many resources from the *Cloud Exchange* (maybe a high amount of one specific service or many of many kinds), exhausting the resources available to the other consumers. Furthermore, it could generate a flood of messages for disrupting the monitoring of agreements compliance between *Consumers* and *Service Providers*, making the entire system unable to operate fulfilling the *agreements*. Finally, it is possible that one attacker performs many requests from one or multiple *Service Providers* in order to disrupt the use of services from the rest of the consumers. The *Inter-Cloud* has already some kind of defense due the fact that it is not possible to make requests directly to the *Cloud Exchange*; all requests must pass before through a broker.

The solution is affected by the following **forces**:

- Objectives – Its objectives may be vandalism, political action, or monetary gain.
- Duration – Usually, the longer the service will be unusable, the better for the objectives of the Attacker.
- Untraceability – Since the attack compromises several components and consumers, it would be better if no one knows who is the responsible one.
- Obfuscation – Since the attack compromises several shared resources, all of them would be affected; even if the attacker objective was just one service. If he blocks them all, it could cause consumers and providers confusion (even panic) and will obfuscate its real intentions.

The attack can be performed by taking advantage of the following vulnerabilities:

- Any consumer can open an account in a *Service Provider*.
- Any consumer can request what he needs to its *Service Provider*.
- The *Service Provider* consults automatically the broker *to see* if it does not have enough amount or the kind of requested resource. The broker will redirect the request to the *Cloud Exchange*.
- The attacker should be able to receive unlimited resources if needed.
- Most providers put no control over the number of requests that can be done in a certain amount of time.
- *Service Providers* access *the Inter-Cloud* only through one broker.
- The broker is responsible of the compliance of the agreements of their associated *Service Providers*.

Solution

A *Denial-of-Service* (*DoS*) attack could flood the *Inter-Cloud* in order to disrupt the monitoring of compliance agreements between *Consumers* and *Service Providers* with the objective of making impossible the accurate control of the subscribed services, thus making the *Inter-Cloud* be a non-regulated system. Also it could perform many requests to the *Federation* in order to significantly reduce the availability of the resources (resource exhaustion) with the objective of leaving the rest of the *Consumers* without resources to use.

Structure

Figure 1 shows a class diagram of a **Cloud Federation**. A **Service Provider** processes requests from **Consumers** through a **Portal**. The **Service Provider** can be assembled in different ways such as an aggregation of clouds, a set of peers, etc. The composite pattern defines a hierarchical structure for the **Service Providers** that can be used for special services or types of services (more secure or premium, for example). The **Service Provider** sends requests to a **Cloud Broker**. The **Cloud Broker** implemented internally or externally redirects the requests to the **Cloud Exchange** component. The **Cloud Exchange** consults its **Catalog** (a.k.a. Information Repository) looking for the best match for the request. The **Catalog** lists several kinds of services. The **Cloud Exchange** is aware of the conditions of the entire system through status information coming from **Cloud_Brokers**. The **Catalog** is updated using the information sent by the **Cloud Brokers**.

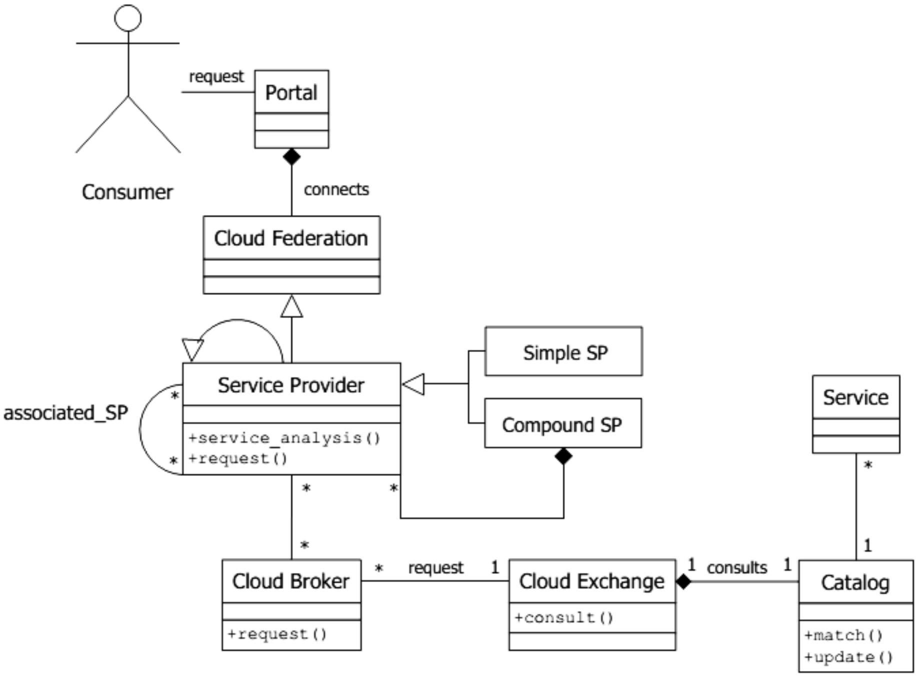

Figure 1: Class diagram of a *Cloud Federation*

Dynamics

We show here two misuse use cases representing the previous solution, both use cases make use of the *federated Inter-Cloud* pattern [Enc14]:

- *UC1:* Disrupt the monitoring of compliance agreements (actor: Attacker)
  Summary: An **Attacker** floods with messages the **Cloud Broker**, making very difficult to accurately monitor the services between the **Service Provider** and the **Consumer**.
  Actor**: Attacker**
  Precondition: The **Cloud Broker** monitors the information and compliance agreements compliance of the **Service Providers** associated to him. The address of the **Cloud Broker** is public and reachable for anybody.
  Description:
  a) The **Attacker** floods of messages the **Cloud Broker**.
  b) The **Cloud Broker** tries to monitor the communication between **Service Provider 1** and **Service Provider 2**.

Alternate flows:

- o The **Cloud Broker** sends to the **Cloud Exchange** the monitoring information, but this information is not accurate.

Post condition: The **Cloud Broker** failed to monitor the compliance agreements.

- *UC2:* Resource exhaustion (actor: Attacker) (Figure 2)

  Summary: An **Attacker** performs many requests to its **Service Provider** in order to slow down the **Cloud Exchange**, thus delaying the requests from other **Consumers** in the Federation.

  Actor**: Attacker**

  Precondition: The **Attacker** must have an account in one or more **Service Providers** that belong to the Federation.

  Description:

  a) The **Attacker** performs multiple independent requests to its **Service Provider**

  b) The **Service Provider** redirects the requests to the **Cloud Exchange**

  c) The **Cloud Exchange** processes the requests

  d) The requests take up most resources and requests from other users are denied.

  Alternate flow:

  - o The requests are accepted and the resources are assigned to the **Attacker**.

  Post condition: **Cloud Exchange** or **Catalog** take too long to respond other consumers' requests, or even might not be able to answer them due to resource exhaustion.

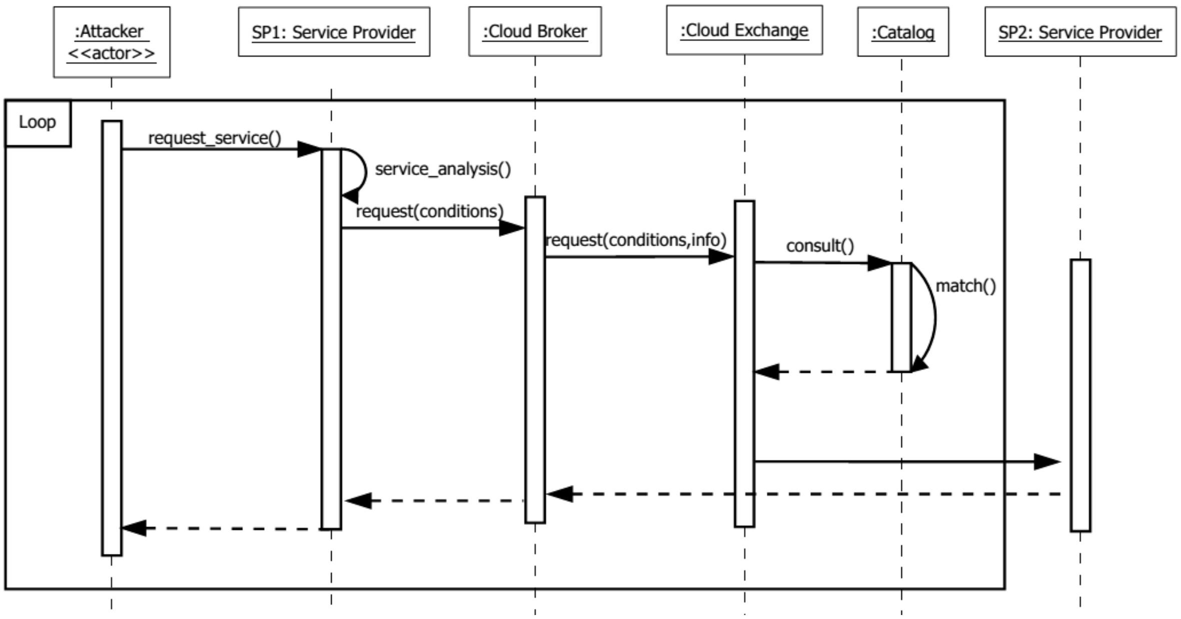

Figure 2: *Use Case 2,* Resource exhaustion

Known uses

Attacks in *federated Inter-Cloud* environments have not happened yet since the *Inter-Cloud* is very new and is still under development. However, we should keep in mind the possible vulnerabilities of the system in order to produce more secure designs.

Consequences

This misuse has the following advantages for the attacker:

- Objectives – Its vandalism objectives can be reached if the **Service Provider** allows multiple requests. If the **SP** provides an *API*, it is possible for the attacker to automate the attack with a script that can request resources for an arbitrary time. Its political objectives can be reached, for example, if he plans his attack the days before elections (the attacker can launch the attack when he considers it convenient). Its monetary objectives can be reached due the fact that the attacker can launch the attack whenever he wants (same as the previous objective); for example, shut down the federation the day before Christmas.
- Duration – The attacker can plan the duration of the attack and even keep attacking when the system is reestablished.
- Untraceable – Since anyone can get an account in a Service Provider or launch an attack from an anonymous place the attack might be untraceable.
- Obfuscation – Since every shared resource is affected, the real intention of the attacker can be obfuscated making it more difficult to stop or take some particular measure against the attack.

Possible sources of failure include:

- A limit on the amount of resources that can be requested by the consumer or **Service Provider** (through policies); for example, you cannot ask for more than half of the resources contributed to the federation.

  Passing through a broker and not directly to the **Cloud Exchange** can potentially be an advantage for the defender.

Countermeasures

Denial-of-Service in *federated Inter-Cloud* can be stopped by the following countermeasures:

- Have a policy about the amount of resources that can be requested.
- Filter requests- The **Cloud Broker** must analyze and filter all requests to determine if they are appropriate in type and quantity.
- Replication- Replication must be part of the design of the **Cloud Exchange** and **Catalog** component. Under a *DoS* attack, the components must stay always online, this also will avoid scaling problems and will improve the fault tolerance.

- *IDS-Firewall- IDSs* and firewalls can ensure that packets with very large sequence numbers and garbage packets are discarded. Again, the *IDS* pattern is relevant, as well as the Firewall patterns [Sch06b].
- Use of Proxy and Stateful Firewalls [Sch06b], which can look inside the request packets and analyze their contents, as well as the headers, to decide if the information is appropriate or not. These can be implemented within the **Cloud Broker**.

Forensics

Where can we find evidence of this attack?

- *Cloud Broker* is the first component that should be analyzed, *IDS* and *Firewall* patterns can help in the forensic task.
- *Cloud Exchange* must receive request information from the *Cloud Broker*, every request made must be logged, also every time a *SP* joins the Federation the *Cloud Broker* must inform it to the *Cloud Exchange*.
- *Cloud Exchange* must log all the information about the assignment corresponding to a request.

Related Patterns

- *VoIP misuse patterns*- A *VoIP DoS attack* is presented in [Pel09] by overwhelming resources in order to disrupt *VoIP* operations, typically through a flood of messages.
  This attack is very similar to disruption of the monitoring of agreements compliance presented in this paper (at least the way the attack is performed, but not its direct objectives); that is an attack who interrupts the way the agreements compliance are measured or controlled (not just disrupt the normal operation of the services).
- *Abstract IDS pattern* – it allows monitoring of all traffic as it passes through a network, analysis the traffic to detect possible attacks, and trigger an appropriate response. The *Signature-Based IDS* pattern and the *Behavior-Based IDS pattern* are other two possible concrete versions
- *Firewall patterns like Packet Filter Firewall* and *Proxy-Based Firewall* [Sch06b].
- *Federated Inter-Cloud pattern* [Enc14] – solves the problem of resource exhaustion, or the lack of specific services that can affect at one Service Provider. The Service Providers form a federation in order to share capabilities. Under a *DoS* attack the resource exhaustion affect the entire Federation.

References

[Ber10] D. Bernstein and D. Vij, "Intercloud security considerations", In *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, IEEE, 2010, 537–544.

[Buy09] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility". In *Future Generation computer systems*, 2009, 25(6):599–616.

[Buy10] R. Buyya, R. Ranjan, and R. N. Calheiros, "Intercloud: Utility-oriented federation of Cloud computing environments for scaling of application services". In *algorithms and architectures for parallel processing*, Springer, 2010, 13–31.

[Enc14] O. Encina, E. B. Fernandez and R. Monge, "Towards Secure Inter-Cloud Architectures". Submitted for publication, 2014.

[Fer05d] E. B. Fernandez and A. Kumar, "A security pattern for rule-based intrusion detection", in *Proceedings of the Nordic Conference on Pattern Languages of Programs, Viking PLoP 2005*, Otaniemi, Finland, September 2005, 23–25.

[Gro12] N. Grozev and R. Buyya. Inter-cloud architectures and application brokering: taxonomy and survey. *Software: Practice and Experience*, 2012, 44: 369-390.

[Kec12] G. Kecskemeti, M. Maurer, I. Brandic, A. Kertesz, Z. Nemeth, and S. Dustdar, "Facilitating self-adaptable inter-cloud management", In *Parallel, Distributed and Network-Based Processing (PDP), 2012, 20th Euromicro International Conference,* IEEE, 2012, 575–582.

[Pel09] J. Pelaez, E. B. Fernandez, and M.M. Larrondo-Petrie. "Misuse patterns in VoIP", *Security and Communication Networks Journ*al. Wiley, vol. 2, No 2, 635-653, published online: 15 Apr 2009. Ref: http://onlinelibrary.wiley.com/doi/10.1002/sec.105/pdf

[Sch06b] M. Schumacher, E. B. Fernandez, D. Hybertson, F. Buschmann and P. Sommerlad. *Security Patterns: Integrating Security and Systems Engineering*. John Wiley & Sons, Inc., 2006.