Máster en Ingeniería Informática

Administración de Sistemas y Seguridad

Security in cloud environments

Saidani Khaled Mohamed Elamine

Hamada Bouhacida

GRANADA 2021

Index:

01. Introduction

Today, the presence of the internet or any other type of network in our lives is indisputable. These networks allow us to communicate between distant places with a negligible delay.

It is because of this technology that a new computing model known as Cloud Computing emerges. However, this technology predates the use of the internet and the idea of distributed computing arose in the 1960s when **John McCarthy** proposed the idea of shared system computing to sell the resources of a computer as if it were one more service. However, due to the limitations of technology in terms of networks, the project was put on hold.

It was in the 90s when this idea reappeared, since the internet of the time already had enough bandwidth to support cloud computing. From then on, more and more companies began to develop this technology, including Compaq Computer or Amazon, which, in view of the amount of idle resources on their servers, launched Amazon Web Services.

That is when other companies jumped on the wave of Cloud Computing, such as Google or Apple. Therefore, we can define Cloud Computing as a computing paradigm in which computing services are offered through the network. These services are very diverse and can cover several areas, for example email services or storage services such as Gmail or Dropbox.

Its main characteristics are extracted from this definition:

- The infrastructure that supports the service is completely transparent to the user without the need for them to worry about maintenance or any other aspect related to it.

- The resources or services offered are accessible from any computer with an internet connection, at any time and from anywhere.

- The resources offered are scalable and elastic. This means that the functionalities offered to the client can increase or decrease depending on their needs.

However, this type of computational model is not exempt from errors and therefore some of the main drawbacks of this model are presented:

- By contracting this type of service, you lose control of the data, fully trusting that the contracted service will keep all the data in the safest possible way.

- Resources are limited, so the user can start with what may appear to be sufficient resources at first. However, as your business or resource needs increase the budget for these types of services may not.

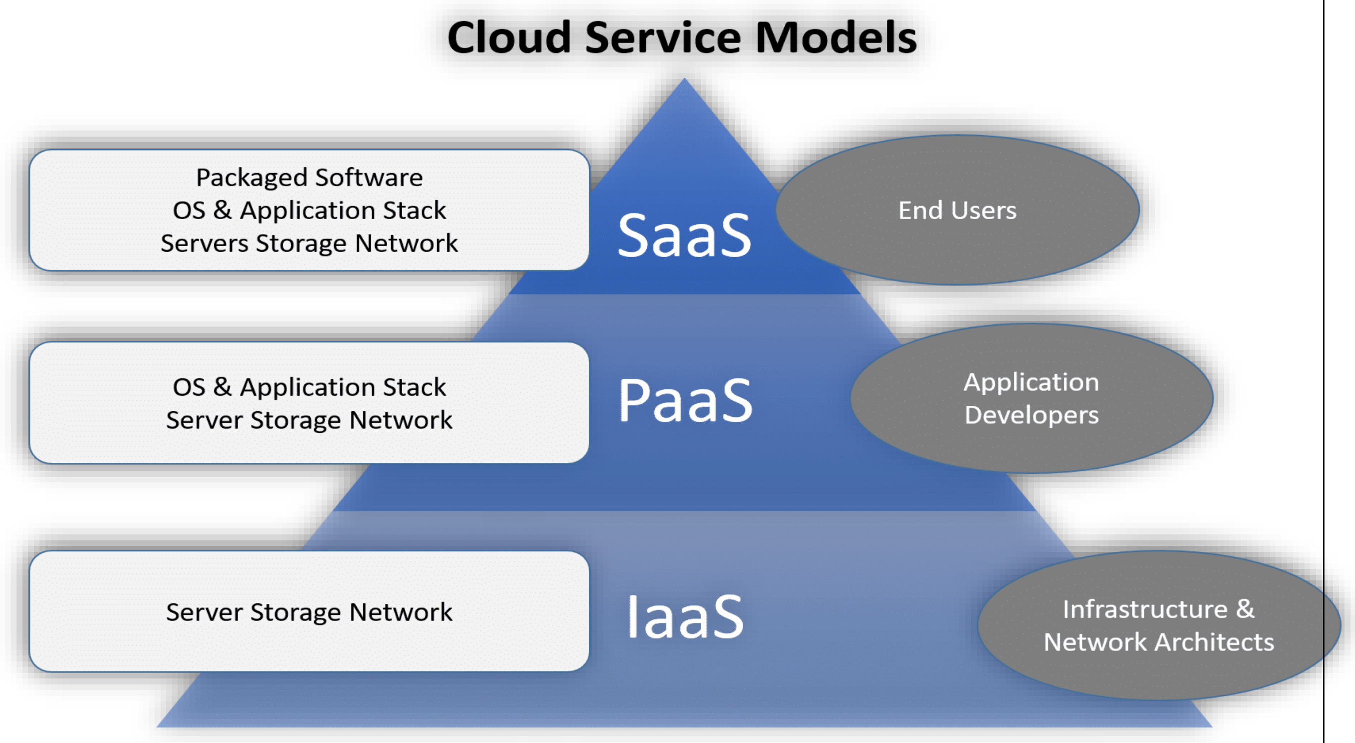
- Another clear problem with this type of technology is that to use a cloud service, you accept the conditions that the provider imposes.

- In a similar way to the advantage mentioned above, for which the service is accessible from anywhere, there is the inconvenience that restricts the use of the application in the event that we do not have an internet connection of sufficient quality.

In the event that these conditions change, the only option for the user is to accept them without further ado or to stop using said service, which is not always possible due to a possible dependency caused by using it for a long time.

02. Types of cloud environments

As we have commented previously, Cloud Computing covers a wide range of functionalities. Depending on the type of service offered, we refer to one environment or another. However, the main characteristics, advantages and limitations will be common among all of them, these being the ones discussed in the previous section. Knowing this, below we will explain the different types of associated services.



01-Cloud Service Models

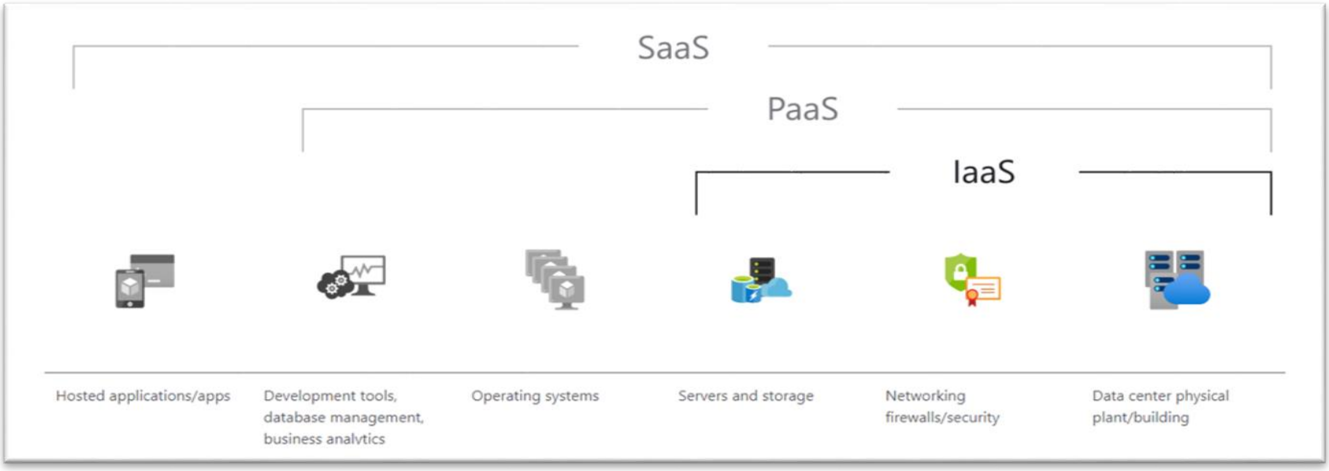
We start with the so-called On-Premise, Which indicates the ability of the company to have access to its own local infrastructure. This way you do not lose as much control over your data and machine settings. However, it can lead to an increase in costs compared to the use of solutions in the public cloud.

Another type of cloud solution is known as **SaaS (Software as a Service)**. In this case, the client contracts the use of software that is provided as a service through the network. Typical examples of this type of solution are email servers like Gmail, storage systems like Google Drive or Dropbox, music playback systems like Spotify, etc.

On the other hand, if what we want is to completely outsource the use of servers, saving us the maintenance they entail but controlling their infrastructure, we would be talking about **IaaS (Infrastructure as a Service)**. In this type of service, the client hires the hardware components that he deems appropriate (CPU cycles, hard disk capacity, memory, etc.). This means that in the event that the user wants to vary the resources according to their use, they can do so easily. Thus, elasticity is provided to the infrastructure.

Finally, we will talk about another of the most common solutions provided by Cloud Computing. This is **PaaS (Platform as a Service)**, platform as a service for its acronym in English. In this case, as its name suggests, it provides a platform that includes a development environment that can include databases, font managers, collaborative work software, etc. As you might suppose, application developers who want to obtain an environment to work on without having to worry about its administration usually hire this type of service.

However, as already mentioned, not only do these types of cloud solutions exist, but there are also a wide variety of “as a service” elements, such as **IDaaS (Identity as a service)**, **AIaaS (Artificial intelligence as a service), IoTaaS (Internet of things as service**) and a long etcetera.


SaaS
PaaS
IaaS
Hosted applications/apps
Development tools, database management, business analytics
Operating systems
Servers and storage
Networking firewalls/security
Data center physical plant/building

02- Structure of the cloud services

03. Evolution of cloud environments

Although it may seem that the cloud is a very recent environment, as discussed above, it comes from the 1960s, when access to a computer was reserved for universities and large companies.

They implemented access to said computers through time-sharing systems. In them, access to computer resources was done through queues, in which programs were added and their use of CPU time was limited.

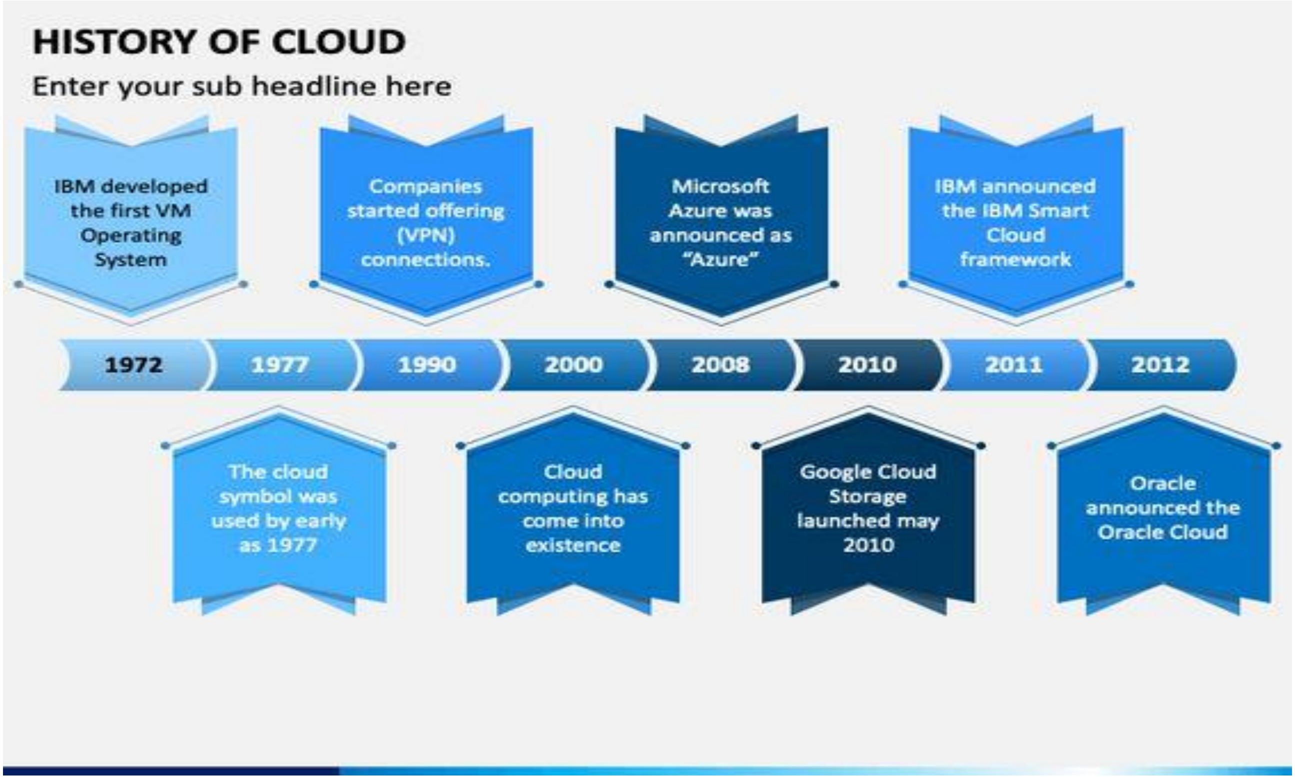
As technology advanced, this way of working was changed and it is that to face the problems derived from previous systems, the creation of networks of computers connected to each other was proposed, which sold their use as a more resource for a company. However, technology did not allow such infrastructure at that time. Over time, computer networks were created that were connected to each other, giving rise to what was known as ARPANET, this being the forerunner of the Internet.

To this all this was added the emergence of virtualization technology, making it possible in the 70s to run more than one operating system in isolation on the same machine. The orchestrator of this technology was IBM who launched the VMOS (Virtual Machine Operating

System). This concept evolved along with the Internet until virtual private network offers began to appear as an economically affordable service, now leading to Cloud Computing in the 1990s.Later, at the beginning of the century, AWS emerged, launching Elastic Compute Cloud (EC2) shortly after allowing companies and individuals to rent virtual machines through which they could use their own programs and applications.

At the same time, Google also launched its Google Docs service, which made it possible to save, edit and transfer documents in the cloud. Shortly after, IBM, Google, and several universities collaborated to develop a server farm in which to jointly investigate. It also coincided in the same year, in 2007, that Netflix launched its streaming service using the cloud to transmit movies and audiovisual content to its customers.

After all this, the cloud has not stopped growing until today. In which it has acquired an incalculable strength and value.



HISTORY OF CLOUD

Enter your sub headline here

IBM developed the first VM Operating System

Companies started offering (VPN) connections.

Microsoft Azure was announced as "Azure"

IBM announced the IBM Smart Cloud framework

1972

1977

1990

2000

2008

2010

2011

2012

The cloud symbol was used by early as 1977

Cloud computing has come into existence

Google Cloud Storage launched may 2010

Oracle announced the Oracle Cloud

03-History of cloud

04. Importance of cloud security:

As we have seen, the cloud is a ubiquitous tool in our lives. In it, various types of essential data are stored for many entities. So we can determine that by increasing the use of this paradigm, the danger posed by any threat to it has also increased. In addition, the fact is that the data present in the cloud is constantly in danger: data corruption, destruction of the physical medium where it is stored, loss of connection, etc.

For, one of the most important requirements of this model is **security**. In addition, it is that, both the contracting companies and the service providers have the task of keeping all these data and applications safe.

This is why one of the biggest concerns when hiring a cloud service for your company is how secure your data and applications will be. Robust solutions are needed that meet the essential requirements in this environment. The latest security tools are also required while implementing advanced security protocols to eliminate or minimize the impact of potential failures or potential attacks. The reasons are many and some of them are listed below:

- The number of **security breaches** always increases over time. These gaps can lead to a service drop by a company, which obviously leads to a loss of money for the company, in which case the only thing that can be done is to implement a security strategy in the cloud that is as strong as possible.

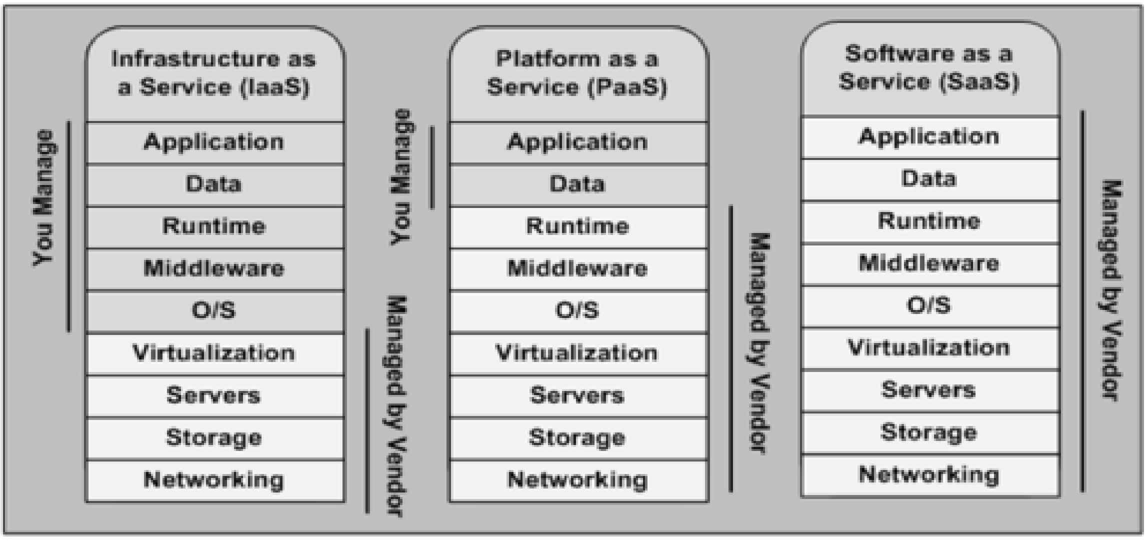
- **Security** solutions always differ from one company to another and that is that even if companies place a dedicated team to monitor security, there will always be companies that are an easier target than others. This means that you always have to try to be on top of **cloud security**.

- Businesses should completely eliminate the possibility of **data loss**. These are stored on remote servers and can be lost for a variety of reasons. Therefore, it is necessary to secure said data from the cloud, ensuring its redundancy and performing backups. Moreover, a disaster recovery strategy should be developed so that even in the event of an accident at the main data storage site, it is possible to recover said data.

As a summary of all this, it can only be said that for a company or individual to deliver their data or applications to a third party, there must be mutual trust that both parties will implement the necessary measures to maximize the security of the environment Cloud. This is something that can only be achieved by being at the top of **cloud security** by preventing failures and attacks on these systems.

05. Security flaws

Once some of the security requirements for cloud environments have been detailed, here are the most common risks that can occur when we do not implement any of the necessary security measures. Depending on the type of cloud model, we can see in the following figure those responsible for the different security flaws.



04-Responsible for security flaws

As we can see, one of the main threats is **unauthorized access** to both servers and data. In the first case, if they access the machine managed by the provider, the responsibility lies with the provider, but the services hosted and the data used by them are the responsibility of the developer. In both cases, the importance of implementing reliable authentication measures to verify the connection of both users and applications to the machine and the services hosted on it is noteworthy.

If the accesses are made through the network, there is an architecture called zero trust, which is based on the principle of not trusting any entity, not even those that are identified in the system. Therefore, with this structure, all entities must authenticate to access each resource. Likewise, it allows you to limit their access and block operations to both identified and non-identified users.

Although we are more aware of cloud security today, many companies still include unreliable security measures that can be circumvented by attackers or malware to gain access to servers and data. In the first case, attackers can use various methods such as brute force, hacking of employee accounts, phishing and even social engineering. That is why it is extremely important to provide staff with training on the different techniques they can use to gain control of their accounts, as well as stress the importance of establishing strong passwords and enabling the double authentication factor, if applicable. is available. In the case of malware, attackers can attempt to upload malicious programs to servers as a cloud service. This procedure is known as malware injection, with which they can run their malicious applications to perform various operations such as data theft, espionage, among others.

06. Examples of famous cloud attacks:

6.1. OneDrive

Microsoft's cloud storage system is a clear target for attackers. In 2019, the company warned of a false email that bore its signature in which they urged the user to access their account to review some of their files. Accessing the attached link presented a false website where they could log in with different accounts, such as Google, Outlook, Yahoo, among others. Its purpose was to steal the credentials to continue sending this type of message through phishing or to carry out attacks on the user's contacts.

6.2. Dropbox:

One of the most popular attacks in cloud environments is the theft of credentials and then commercialize them on the black market. Among the many real cases that have occurred, Dropbox was the most notorious in 2012, due to the massive amount of compromised data that reached five gigabytes. This translates to more than 68 million users whose credentials were exposed on the dark web.

The origin was not revealed until four years later when the company had to admit that both the emails and their keys were stolen. Apparently, the trigger arose when a Dropbox employee used the same password that he had for **LinkedIn** when this platform suffered a similar attack in which many accounts were compromised, including this person's.

This allowed attackers to access the user database to collect their credentials .Dropbox tried to fix it by forcing all users to enter a new password and they reconsidered their situation in relation to security measures to protect data.

6.3. Tesla

One of the expert companies in cloud security **RedLock** published a report in 2018 in which it reported a security flaw in one of the consoles of the famous Kubernetes services orchestrator.

This system was used by the Tesla Company to manage its data cloud on Amazon, whose access was not protected by a password, which meant that anyone could infiltrate its administration panel. Thus, the attackers entered their cloud from which they could access confidential information, in addition to introducing programs to perform crypto mining. This attack is known as crypto jacking, whose objective is to use the resources in the cloud to inject crypto mining programs with which to earn large sums of money.

A novelty is that they used a series of obfuscation techniques to hide their activity. The first of these was to install crypto mining programs on a single server, which they configured a proxy to mask the real IP address with which it communicated with the attackers. They also kept CPU usage low to avoid creating too much traffic on the network and encrypted these communications with SSL to go unnoticed by monitoring tools. They got this encryption service through the company **Cloudflare**, which offers it for free to help maintain privacy on websites and security tools.

6.4. DynDNS

When we connect to an Internet address, we actually use a **DNS** server that transports us to it. This is the basis of Internet browsing. One of the companies that is dedicated to offering this service is DynDNS, which in 2016 suffered a **DDos** attack, consisting of collapsing the target servers sending more requests than they can handle. This attack caused part of the network in the US and Europe to crash for several hours. Among the companies that were affected were **Twitter**, **Pinterest**, **Reddit**, **GitHub**, **Spotify** or **PayPal**.

This type of attack is not new and therefore companies are usually prepared by having a large bandwidth so that the chances of collapse are minimal.

However, in this attack both the number and the origin of the requests surprised. **Dyn** estimates that 100,000 devices were used which together sent 1.2 **Tbps** to various company servers. The moral of this attack is that no matter how prepared a company is, it is always vulnerable to attacks of this type. As cloud environments advance, you must also invest in security to try to minimize the associated risks.

7. Security tools:

In this section, we will discuss some of the most popular tools, both open source and paid, for protecting cloud systems.

CloudGuard Dome9

It is a multifunction tool designed to secure cloud systems by analyzing the configuration in search of weaknesses, making suggestions based on the best practices for these environments, both in legal and security aspects. It also provides protection against credential theft and data loss. To do this, it uses contextual intelligence, based on the collection of information about the threats detected, their objectives, origins, and environments they attack, so that tools with greater prevention capacity are developed. It is a payment tool that can be integrated with providers such as AWS, Azure and Google Cloud.

Okta

It is a tool, also for payment, oriented to the protection of networks through the monitoring of the authentications in any device, the access control of both the users and the employees. It provides a general panel to manage their privileges and implement particular authentication policies on each device. For this, it includes a configurable login system so that it can be unique or use a double authentication factor.

Zeek

Also known as Bro, it is software aimed at preventing and detecting intrusions through network monitoring. Unlike other similar systems, Zeek collects information about the activity and data associated with the nature of the network to adapt to the context of the same. To do this, it provides scripts in an interpretable language that can be modified to customize their behavior to the particular communication scheme. It also provides forensic services to analyze what has happened prior to an attack.

Panther

It is one of the most powerful open source tools used for the security and monitoring of cloud systems. It can be integrated through the AWS CloudFormation service. Its main characteristics are:

-Threat detection using deterministic rules that help reduce false positives.

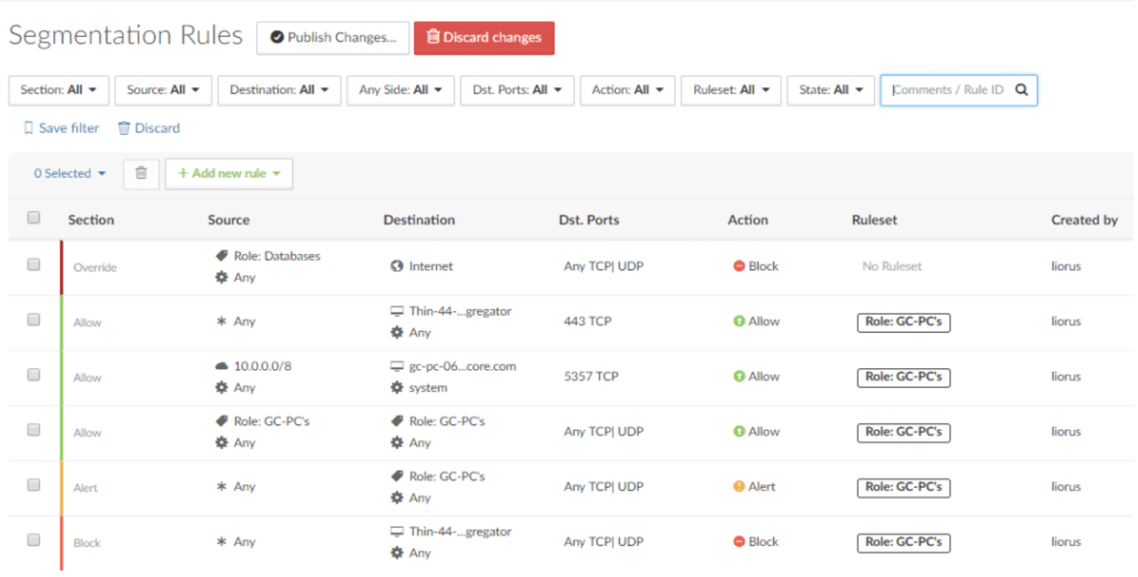
-Analysis of system logs to identify unauthorized access and suspicious behavior according to its own threat database.

- Automatic correction of vulnerable configurations in the cloud.

08. Infection Monkey

Out of all the existing tools, we have decided to try Infection Monkey. It is an open source software developed by the Guardicore cloud security company, aimed at testing a cloud system carried out various attacks to verify its resistance. It can be applied to both public and private clouds.

Its origin lies in an investigation carried out by the company in 2017 on micro-segmentation applied to networks. With this methodology, a network can be divided into several subnets dedicated to a single entity, such as processes, applications, users, among others. In this way, only the strictly necessary permissions are enabled for them to carry out their tasks. However, this technique requires that the company have enough information about its network to be able to divide it, which is often an extremely complicated task to be carried out by one person. Therefore, this company developed a tool called Guardicore Centra Platform, capable of obtaining information about its infrastructure through a series of software agents and records of its information flow. It then automatically contextualizes all the data collected in order to generate a representative and detailed map of the infrastructure. It also helps define micro-segmentation policies through a simple interface based on the information collected and the vulnerability analysis it performs, suggesting measures to improve network fragmentation. This tool is valid for hybrid environments, as well as for virtual machines, containers and instances in providers such as Amazon Web Services, Microsoft Azure and Google Cloud.



Segmentation Rules

□	Section	Source	Destination	Dst. Ports	Action	Ruleset	Created by
□	Override	Role: Databases Any	Internet	Any TCP| UDP	Block	No Ruleset	liorus
□	Allow	Any	Thin-44-...gregator Any	443 TCP	Allow	Role: GC-PC's	liorus
□	Allow	10.0.0.0/8 Any	gc-pc-06...core.com system	5357 TCP	Allow	Role: GC-PC's	liorus
□	Allow	Role: GC-PC's Any	Role: GC-PC's Any	Any TCP| UDP	Allow	Role: GC-PC's	liorus
□	Alert	Any	Role: GC-PC's Any	Any TCP| UDP	Alert	Role: GC-PC's	liorus
□	Block	Any	Thin-44-...gregator Any	Any TCP| UDP	Block	Role: GC-PC's	liorus

After the success of the previous tool, the Guardicore company decided to develop Infection Monkey, a new software with which to **test the infrastructure** of a system regardless of its components and security measures. Unlike the previous one, they decided that this would be free code, so that it could be modified according to particular needs. Essentially, it was devised as a tool to test various penetration strategies. However, the latest updates have also included the ability **to review the configuration of a network**, in order to look for security parameter errors in newly created networks.

It is multiplatform since it supports operating systems such as Windows and Linux, as well as those of cloud systems such as OpenStack. It is programmed in Python and can be configured to work with isolated computers or sets of systems. It should be noted that **the attacks perpetrated by the tool are not simulated**, but are carried out, although it then erases all traces of them to leave the machine in the same state it was in. This means that if, for example, you try to access through a user whose credentials are not secure, and the tool creates a new user on the machine and then executes the attack.

Among the various characteristics it has, three in particular have caught our attention. First, Infection Monkey is designed so that its evaluations do **not affect the performance of the system**, so that they can be performed without slowing down the performance of your tasks.

Likewise, at the end of the tests, **it generates various reports with different levels of detail**, ranging from an overview of the results to a detailed record of the operations and their results. In this way, security experts can use this information to gain insight into which system components have security problems. As a third relevant quality, we highlight that for each security incident **it suggests at least one solution** to resolve it. In this way, it identifies the problems but also helps to solve them by proposing some alternatives.

Among the different attacks that it can carry out, a set of them is based on replicating some of the most popular **exploits**, such as the following.

- **Sambacry**. It allows executing code by accessing in write mode one of the shared resources located on a Linux server. To do this, the tool uses brute force to test the connections to these resources in order to infiltrate a system.

- Shellshock. Allows attackers to execute code by appending it to the end of environment variable values. In this case, if the tool is successful, it collects information about the architecture of the machine, downloads its own file called Monkey dropper and executes.

- **ElasticGroovy**. With it, you can execute system commands using the Java classes of the **ElasticSearch** search engine. The tool examines the default ports that servers are serving queries on and tries to perform the same operations as above.

- Struts2. This vulnerability is associated with certain versions of the framework of the same name, which allows an attacker to insert commands into the system payload to perform malicious actions. Infection Monkey checks if it can exploit this vulnerability in the system and if so infects the machine.

- **Weblogic**. It is a security flaw in Oracle **weblogic** servers, specialized in the development of business Java applications, which allows infecting machines by sending malicious packets. To test it, Infection Monkey configures a new server to send malicious data to various Oracle machines using certain commands to force its response. In case it happens, the tool spreads the infection.

- Theft of credentials. If it is a Windows system, the tool uses a customized version of the **Mimikatz** framework, popularly known for its ability to extract keys, pins, hashes, among others. If it is a Linux system, it generates a multitude of SSH keys and then infiltrates the system.

- **Hadoop**. It is a framework that allows the execution of remote tasks in distributed systems. If the tool finds such a server, it creates a task to spread the infection to all nodes in the infrastructure.

● Brute force intrusion. Infection Monkey uses the credentials stolen in previous steps to enter systems through the following protocols:

- SSH. If it manages to enter the system through SSH it collects Infrastructure and kernel information on the machine. Then download your Monkey dropper file via SFTP and run it.

- SMB (Server Message Block). It is a network protocol that allows sharing resources, such as files or external devices such as printers, between multiple Windows machines. If the tool manages to sneak in, it creates a new server to run an instance of it on the machine.

- WMI (Windows Management Instrumentation). It is an administration server to manage distributed systems remotely. When the tool infiltrates, it copies itself to other computers and spreads the infection.

In addition to the various exploits that it can apply, Infection Monkey also includes analysis tools to prevent some of the most common attacks.

Credential analysis. It applies different techniques such as analyzing the caches to obtain the keys or their hash, as well as generating passwords similar to the domain name or username to detect insecure keys. An example of this is using admin / admin as the credentials for an administrator account. To do this, it uses a customized version of the Mimikatz framework.

- Network segmentation. If the tool detects that the network is fragmented, it tries to infect the systems of a subnet and then tries to spread the infection to the rest through the channels that communicate them.

- Tunneling. Analyze segmentation rules to check the strength of communications between subnets. If the systems do not verify the origin of the messages received, they can attend to requests from malicious machines with which they establish communication to start the attacks.

While this tool has a wide range of vulnerabilities and attacks, it also has some limitations:

- **Privilege escalation**. It is one of the most common intrusion techniques in networks, whose objective is to infiltrate a system using a user with few permissions. The attack can then be carried out vertically, in which the attacker also grants himself root privileges through a series of kernel-level commands. While in horizontal scaling you must impersonate another user to be able to benefit from his permissions. Despite being so popular, this vulnerability is not included in Infection Monkey.

- **Defense evasion techniques**. It groups together a set of methods by which attackers can perform their malicious exploits without being detected by security measures. For this they can classify malicious processes as safe, hide their activity, disable security tools, among others. At the moment, the tool does not have any of the evasion techniques to test on the chosen network.

- **Cyber-collection.** It refers to a type of attack whose objective is to spy on entities through the inclusion of malware to collect and filter sensitive information. Although the tool is capable of gathering information about the architecture of the system, it does not include any spying technique with which to analyze the information stored in it.

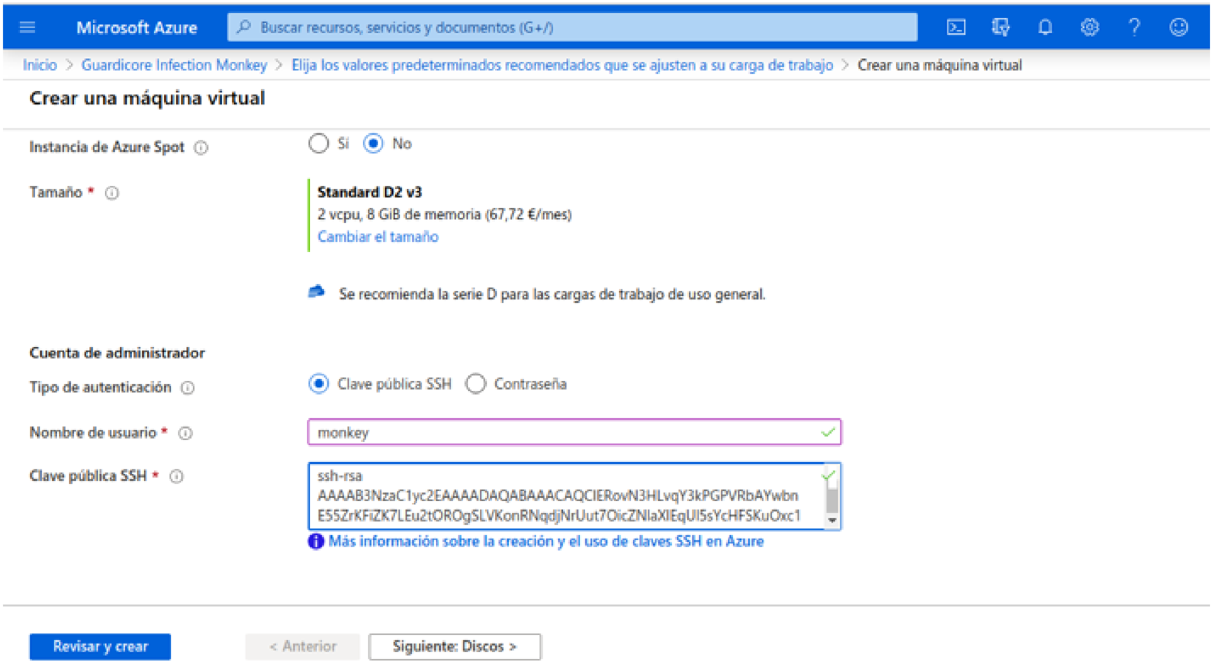
- Data leakage. Today it is one of the most important security problems faced by companies. For this, a wide variety of techniques and malware can be used in order to carry out unauthorized data transmissions to attackers' systems .However, Monkey Infect has not yet included the option to carry out any of the data theft strategies.

8.2. First practical application of the tool

Once we have detailed the qualities of Infection Monkey, we are going to test it experimentally. To do this, we first register on your platform to receive the download address. In addition to some personal data, we must specify the environment in which we want to install it. In our case, we are going to apply it on Azure, since we have a virtual machine in which some micro services deployed for the Cloud Computing subject of the previous semester are stored.

Once we obtain the link of the download by mail, when choosing Azure as a cloud provider, it redirects us directly to the creation of a new virtual machine with the Infection Monkey image, in which we can decide its characteristics, as can be seen in the next two figures. In our we have opted for the basic configuration.

Below in the figure we can see the specification of some fields such as the assigned resource group, the name of the machine, the region in which it is located and the base image that is that of the tool itself. To then be able to connect to it we generate a pair of SSH keys, specifying the public one as can be seen in the figure. For this we have followed the steps of the official documentation for the Ubuntu operating system.


Microsoft Azure
Buscar recursos, servicios y documentos (G+/)
Inicio > Guardicore Infection Monkey > Elija los valores predeterminados recomendados que se ajusten a su carga de trabajo > Crear una máquina virtual

Crear una máquina virtual

Instancia de Azure Spot ○ Sí ● No

Tamaño * Standard D2 v3
2 vcpu, 8 GiB de memoria (67,72 €/mes)
Cambiar el tamaño

Se recomienda la serie D para las cargas de trabajo de uso general.

Cuenta de administrador

Tipo de autenticación ● Clave pública SSH ○ Contraseña

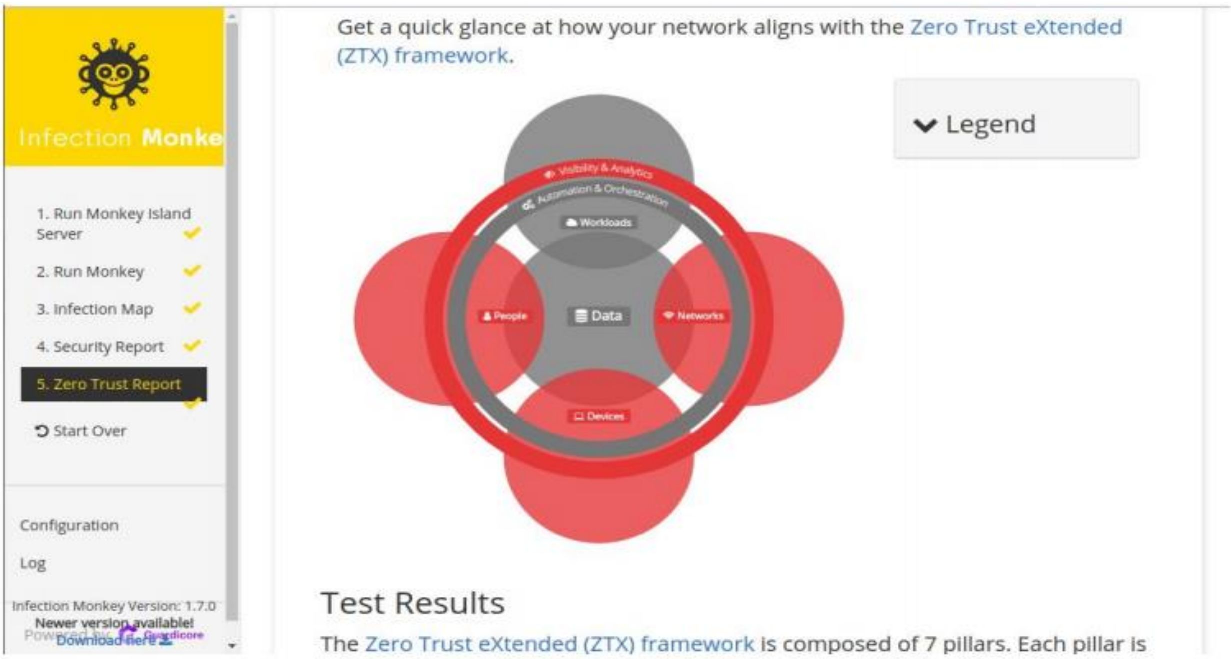
Nombre de usuario * monkey

Clave pública SSH *

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAACAQCIERovN3HLvqY3kPGPVRbAYwbn
E55ZrKFiZK7LEu2tOROgSLVKonRNqdjNrUut7OicZNlaXlEqUI5sYcHFSKuOxc1
```

Más información sobre la creación y el uso de claves SSH en Azure

Once created, we have followed this guide to run the tool. First, we access it through the address https: // <public IP of the MV>: 5000. A general panel appears in which, as a previous step, the purpose of the same is explained. Then with Run Monkey you can decide between running the tool inside the same machine where it is installed, simulating the propagation of an attack within the network, or choosing another system on which to simulate an intrusion. In both cases, specific values can be specified for the previously explained attacks and vulnerabilities.

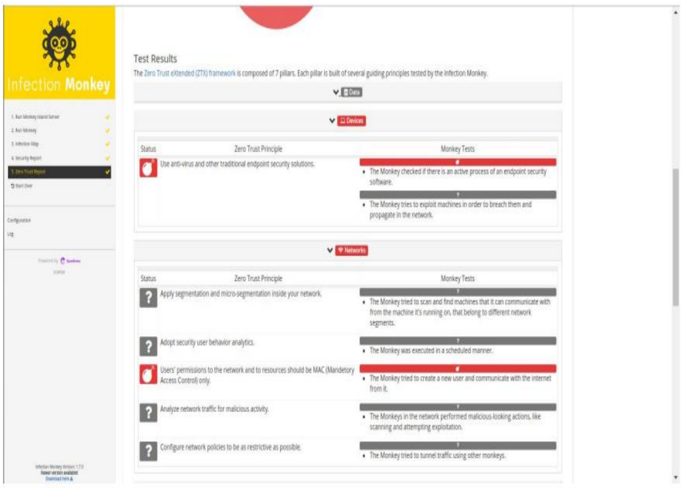


Get a quick glance at how your network aligns with the Zero Trust eXtended (ZTX) framework.

Test Results

The Zero Trust eXtended (ZTX) framework is composed of 7 pillars. Each pillar is

In the image, you can see the analyzed entities. Those in gray color are those that could not be tested due, in our case, to the fact that we do not have tasks running on the machine or data transfers and / or use. Below are each of the incidents detected according to the category to which it belongs, as can be seen in the following two figures.

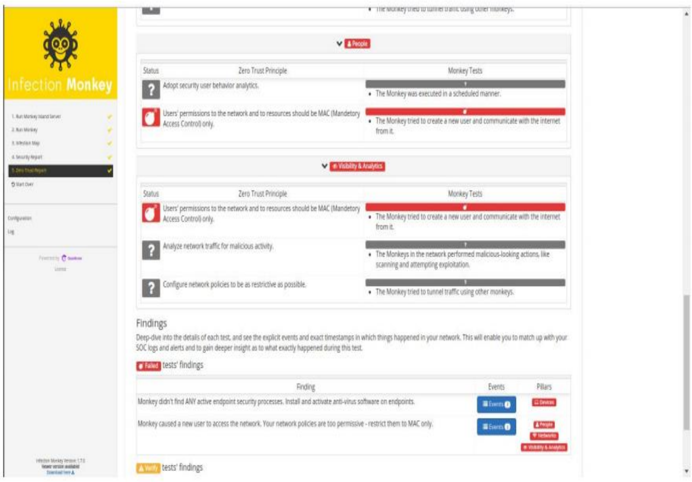


Test Results

The Zero Trust eXtended (ZTX) framework is composed of 7 pillars. Each pillar is built of several guiding principles tested by the Infection Monkey.

Status	Zero Trust Principle	Monkey Tests
●	Use anti-virus and other traditional endpoint security solutions.	The Monkey checked if there is an active process of an endpoint security software. The Monkey tries to exploit machines in order to breach them and propagate in the network.

Status	Zero Trust Principle	Monkey Tests
?	Apply segmentation and micro-segmentation inside your network.	The Monkey tried to scan and find machines that it can communicate with from the machine it's running on, that belong to different network segments.
?	Adopt security user behavior analytics.	The Monkey was executed in a scheduled manner.
●	Users' permissions to the network and to resources should be MAC (Mandetory Access Control) only.	The Monkey tried to create a new user and communicate with the internet from it.
?	Analyze network traffic for malicious activity.	The Monkeys in the network performed malicious-looking actions, like scanning and attempting exploitation.
?	Configure network policies to be as restrictive as possible.	The Monkey tried to tunnel traffic using other monkeys.



- The Monkey tried to tunnel traffic using other monkeys.

Status	Zero Trust Principle	Monkey Tests
?	Adopt security user behavior analytics.	The Monkey was executed in a scheduled manner.
●	Users' permissions to the network and to resources should be MAC (Mandetory Access Control) only.	The Monkey tried to create a new user and communicate with the internet from it.

Status	Zero Trust Principle	Monkey Tests
●	Users' permissions to the network and to resources should be MAC (Mandetory Access Control) only.	The Monkey tried to create a new user and communicate with the internet from it.
?	Analyze network traffic for malicious activity.	The Monkeys in the network performed malicious-looking actions, like scanning and attempting exploitation.
?	Configure network policies to be as restrictive as possible.	The Monkey tried to tunnel traffic using other monkeys.

Findings

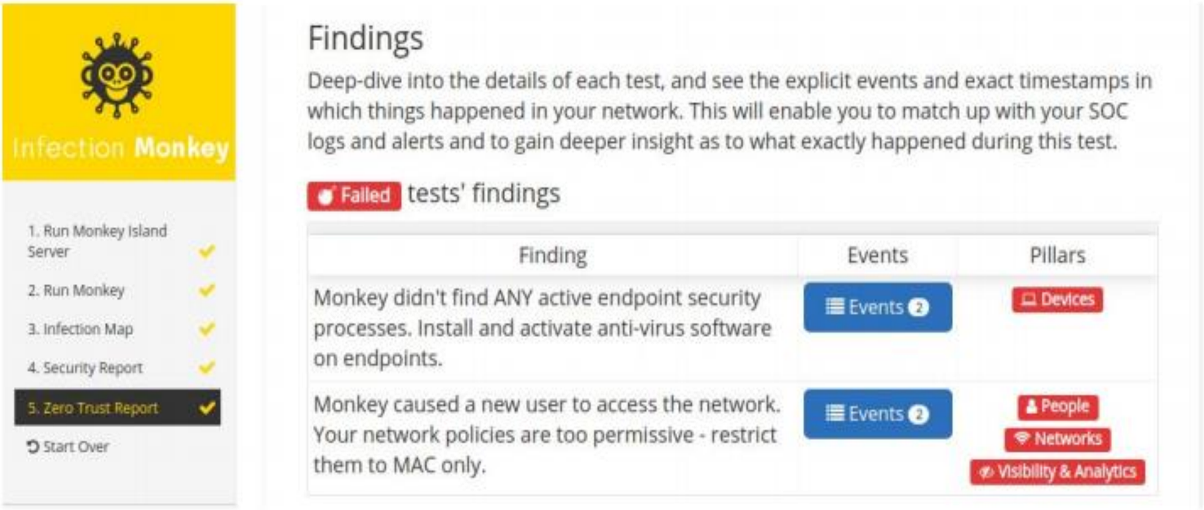
Deep-dive into the details of each test, and see the explicit events and exact timestamps in which things happened in your network. This will enable you to match up with your SOC logs and alerts and to gain deeper insight as to what exactly happened during this test.

Failed tests' findings

Finding	Events	Pillars
Monkey didn't find ANY active endpoint security processes. Install and activate anti-virus software on endpoints.	Events	Devices
Monkey caused a new user to access the network. Your network policies are too permissive - restrict them to MAC only.	Events	People Networks Visibility & Analytics

Verify tests' findings

Among the different flaws it has found is the absence of antivirus or firewalls and the possibility of creating a new user with which to connect to the Internet. Likewise, it makes a record of the operations carried out and their respective results, as we explained above, in such a way that it generates a kind of detailed logs of how it has managed to carry out the attacks. Likewise, it suggests some solutions such as those shown in the figure, which in our case advise us to install some security software in addition to restricting the network access policy for users.



Findings

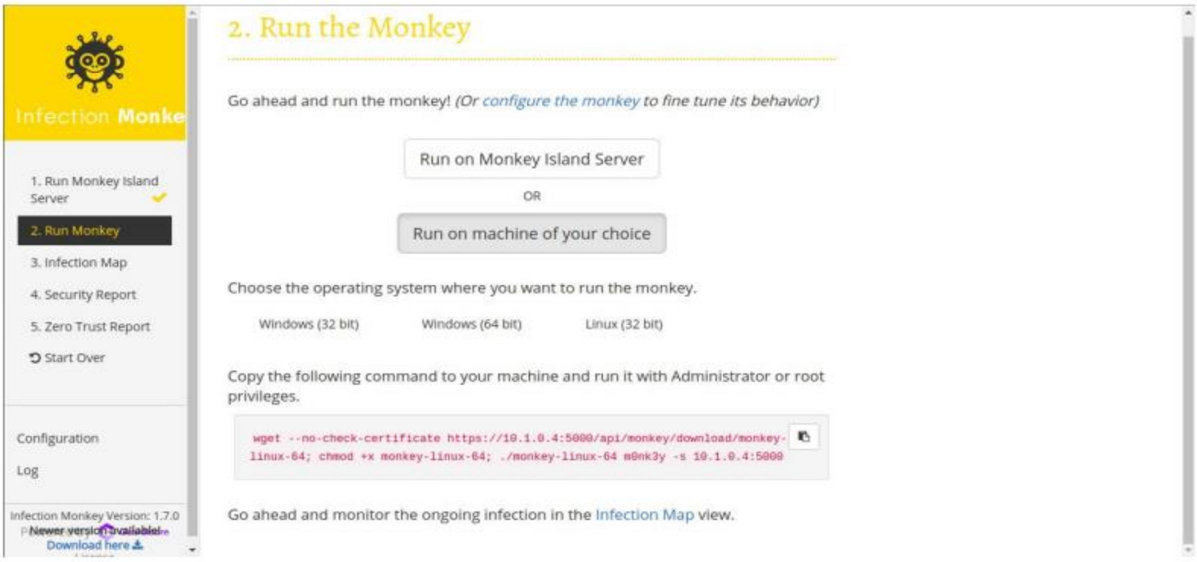
Deep-dive into the details of each test, and see the explicit events and exact timestamps in which things happened in your network. This will enable you to match up with your SOC logs and alerts and to gain deeper insight as to what exactly happened during this test.

Failed tests' findings

Finding	Events	Pillars
Monkey didn't find ANY active endpoint security processes. Install and activate anti-virus software on endpoints.	Events 2	Devices
Monkey caused a new user to access the network. Your network policies are too permissive - restrict them to MAC only.	Events 2	People Networks Visibility & Analytics

08.3. Second practical application of the tool:

We are going to run the tool in our own virtual machine that contains some **microservices**. While in the previous case the tool had access to the system to be tested since that is where it is installed, in this case it is not and therefore there are some additional steps. These include the configuration prior to downloading an instance of the tool to analyze the machine. In our case, we have included some values, such as the user of the machine to try to steal its credentials, in addition to the ports on which the two deployed **microservices** listen (8000 and 8001). Next, we access our machine to be tested, using SSH, to obtain a copy of the tool configured to execute it. This procedure



2. Run the Monkey

Go ahead and run the monkey! *(Or configure the monkey to fine tune its behavior)*

Choose the operating system where you want to run the monkey.

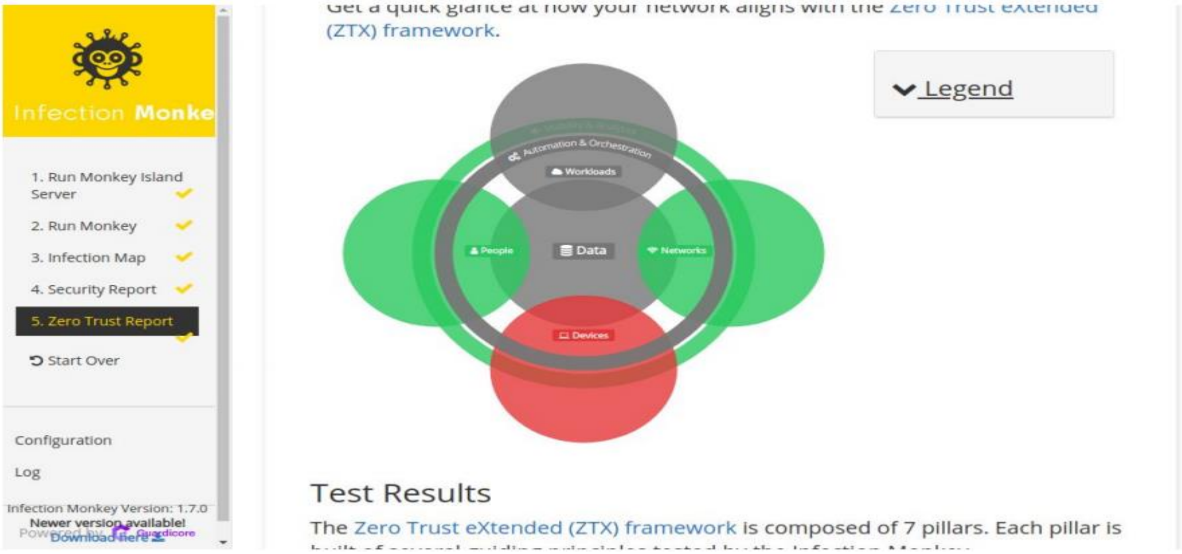
Copy the following command to your machine and run it with Administrator or root privileges.

```
wget --no-check-certificate https://10.1.0.4:5000/api/monkey/download/monkey-linux-64; chmod +x monkey-linux-64; ./monkey-linux-64 m0nk3y -s 10.1.0.4:5000
```

Go ahead and monitor the ongoing infection in the Infection Map view.

is detailed with the necessary commands when selecting another machine to test, as can be seen in the following figure.

The results of this second analysis can be seen in the following two figures. The first shows the graph of the entities tested, which, as we can see, do not have as many vulnerabilities as in the previous one. In this case, the security flaw detected corresponds to the lack of an antivirus or firewall. While the rest of the categories are represented in green, which means that the attacks carried out have not been successful and, therefore, no further incidence has been found. We do not include the suggestions, since for the one found the solution is the same as before: install a security tool.

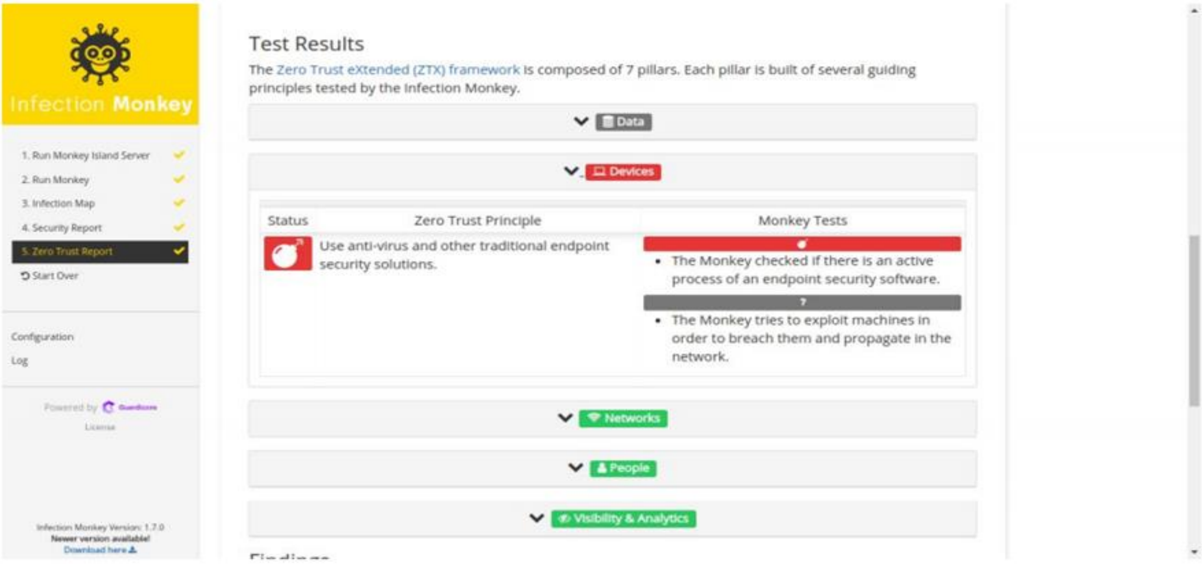


Get a quick glance at how your network aligns with the Zero Trust eXtended (ZTX) framework.

Test Results

The Zero Trust eXtended (ZTX) framework is composed of 7 pillars. Each pillar is

Graphical Infection Results Monkey on our machine in Azure.



Test Results

The Zero Trust eXtended (ZTX) framework is composed of 7 pillars. Each pillar is built of several guiding principles tested by the Infection Monkey.

Status	Zero Trust Principle	Monkey Tests
	Use anti-virus and other traditional endpoint security solutions.	- The Monkey checked if there is an active process of an endpoint security software. ? - The Monkey tries to exploit machines in order to breach them and propagate in the network.

Detected security flaws on our own machine in Azure.

9. Conclusions

As a conclusion to this work, it has been seen that security is an indispensable feature in cloud environments. All companies, applications, data or in short, any type of deployment in this environment must foresee the possible problems that this type of solution entails. These incidents can come from various sources and be of various kinds. And despite relying on this type of paradigm, it must be backed up with strong security measures.

In case of not implementing enough, the consequences can be critical, from loss of valuable information to the fall of relevant services on the network. That is why supplier companies must implement as many measures as possible, updating them as their systems evolve. For this, we can include techniques for collecting and analyzing detected threats as well as the infrastructure, to generate a knowledge base that allows improving the detection capacity of security tools.

Another important aspect to take into account is that users are not exempt from responsibility and that they should also be careful with their data, in the sense that, for example, if a user does not implement sufficiently secure passwords or does not save the Due care with your personal information, the security implemented by the company does not matter, since your data or applications may be compromised.

However, as seen, there are a large number of tools available to solve or detect many of the detailed problems. It will be the supplier's responsibility to implement these tools and to keep a proper control of all the aspects that they support. A clear example has been seen in the practical example that has been developed in this work. Infection Monkey is a powerful tool for cloud environments, providing us with valuable information regarding our environment. It will be in our power to solve the problems that said tool detects in addition to using other tools such as those also commented to obtain additional information on aspects not covered by the software used.

Bibliography:

Intro of cloud computing:

https://www.ecpi.edu/blog/a-brief-history-of-cloud-computing

https://aws.amazon.com/

https://www.centretechnologies.com/cloud

Types of Cloud Computing Structures:

https://www.uniprint.net/en/7-types-cloud-computing-structures/

Structure of the cloud services:

https://azure.microsoft.com/en-us/overview/what-is-iaas/

Evolution of cloud environments:

https://the-report.cloud/the-evolution-of-cloud-computing-wheres-it-going-next

https://in.pinterest.com/pin/420945896427863110/

Importance of cloud security:

https://www.box.com/fr-fr/resources/what-is-cloud-security#:~:text=Maintaining%20a%20strong%20cloud%20security,whole%20new%20way%20of%20working.

Security flaws:

https://cloudsec.tumblr.com/post/115833564074/cloud-security-reference-model

https://www.mcafee.com/enterprise/fr-fr/security-awareness/cloud/security-issues-in-cloud-computing.html

https://managedmethods.com/blog/security-issues-in-cloud-computing/

https://www.cwps.com/blog/cloud-computing-security-issues

famous cloud attacks:

https://blog.storagecraft.com/7-infamous-cloud-security-breaches/

https://redlock.io/blog/cryptojacking-tesla

https://www.itpro.co.uk/cloud-security/34663/cloud-storage-how-secure-are-dropbox-onedrive-google-drive-and-icloud

The Dyn DDoS Attack, Explained (telegeography.com)

https://cybersecurity.att.com/blogs/security-essentials/dynamic-dns-security-and-potential-threats

Software Tools :

https://zeek.org/

https://www.okta.com/fr/

https://www.mintsecurity.fi/en/cloudguard-dome9-en/

https://runpanther.io/

Infection Monkey:

https://www.guardicore.com/whats-new-guardicore-centra-r25/

https://www.guardicore.com/infectionmonkey/

Release the monkey! How Infection Monkey tests network security | InsiderPro (idginsiderpro.com)

https://www.youtube.com/watch?v=qy6RqCPLV8Y