Democratic and Popular Republic of Algeria
Abou Bakr Belkaid University - Tlemcen
Faculty of Science
Department of Computer Science

Option: *Network and Distributed Systems (NDS)*

Distributed IT Infrastructure Monitoring System

Use Case Elk Stack, Elastalert Vs Zabbix

Presented by :

- **MOUSSAOUI Sara**
- **MEZIANE Meryem**

Presented on June 30, 2020, in front of the jury composed of :

- **Mr BAMBRIK Ilyas** *(President)*
- **Mrs MALTI Djawida** *(Examiner)*
- **Mr MATALLAH Houcine** *(Supervisor)*

Academic Year : 2019-2020

Acknowledgement

*First of all, we thank **Allah The Almighty** for granting us the courage to complete this work.*

This work would not have been realised without the contribution of many people ; each with his special way. We would like to express our appreciation especially to the following:

*Thanks to the hosting **Company Kloufi** for their efforts to train a generation of developers, data scientists, cloud architect and giving us the opportunity to work with the modern technologies and prove ourselves in our country.*

*Thanks to our supervisor **Mr MATALLAH Houcine** for his psychological support , huge efforts and for the invaluable help he provide us with, which enable us to complete this graduation project in the best possible conditions.*

*Thanks to our co-supervisor **Mr KHERFI Foued** for his help, support and valuable knowledge, as well as for his complete availability and generosity in terms of training and supervision. He has never ceased to encourage us all along this project.*

*Finally, special thanks to the honourable jury members consisting of : **Mr BAMBRIK Ilyas** and **Mrs MALTI Djawida** for accepting to examine our work and engaging with their propositions.*

Dedications

To Allah The Almighty

Forgiving us his blessings and strength.

To My Parents

The reason for which I became who I am today.

Dear Mom, thanks for always being there for me, for your pieces of advice and patience... I am so grateful for the delicious meals you made for us.

And Dad, thank you for encouraging me and supporting me morally and financially, and for driving me everywhere.

Your persistent prayers made this road easier.

To my Sisters

I am really grateful to both of you.

You have been my soulmates.

To my Family and Friends

Thanks for listening to my complaints and letting me vent as much as I need … I have learned a lot from you.

To every hardworking, respectful Teacher

Finally, to Ourselves!!!

Meziane Meryem

Dedications

First, thanks to Allah the Almighty for the guidance , strength , power of mind and for giving us a healthy life to complete this work

To my Parents

My parents... my world , my source of inspiration. Words are not enough to express my love and gratitude.

To my mother, for everything you did for me. Your support and sacrifice helped me a great deal to complete my university career . I feel so blessed to have you in my life. Thank you for teaching me how to become a strong and successful women in this life.

To my father, for all the efforts you made for me . Due to your trust , faith in me and your help , I achieved a lot in my life , I am very grateful for being your daughter.
Thank you for everything.

To my Brothers

For you help , for every moment you made me smile. You are definitely very special to me , you are my positive energy.

To my Family and Friends

To my dear grandparents for your prayers and your mental support.

To my aunts, my uncles from "Moussaoui" and "Hadj Aek" families.

To my best friends who helped and supported me, I am so blessed to have you in my life.

To Every Person

who has taught me, guided me or even helped me morally to complete this work.

Moussaoui Sara

Abstract

Modern companies need different tools and mechanisms to monitor their IT infrastructure and send in real-time notifications and alerts to managers in case of errors or technical problems. Our graduation project, which is part of graduation internship thesis for a "Master's Degree in Networks and Distributed Systems" at the University of Tlemcen, proposes a solution to avoid these infrastructure malfunctions that could have a negative impact on the company image. The work carried out consists of setting up a tool for monitoring the IT infrastructure based on free open-source software in terms of use cases between ELK stack with Elastalert and Zabbix.

Keywords: ELK Stack , Elastalert , Zabbix , Monitoring , Logs , Alerting , Elasticsearch , kibana

Résumé

Les entreprises modernes ont besoin de différents outils et mécanismes pour superviser leurs infrastructures informatiques et envoyer des notifications et alertes en temps réel aux responsables en cas d'erreurs ou de problèmes techniques. Notre PFE, inscrit dans le cadre d'un mémoire de stage de fin d'études pour l'obtention du «Master en réseaux et systèmes distribués» à l'Université de Tlemcen, propose une solution pour éviter ces dysfonctionnements de l'infrastructure qui pourrait avoir un impact négatif sur l'image de marque de l'entreprise. Le travail effectué consiste à mettre en place un outil de surveillance de l'infrastructure informatique basé sur des logiciels libres en terme de cas d'utilisation entre ELK stack avec Elastalert et Zabbix

Mots clés : ELK Stack , Elastalert , Zabbix , Surveillance , Journaux , Alerte, Elasticsearch , kibana

ملخص

تحتاج الشركات الحديثة إلى أدوات وآليات مختلفة لمراقبة البنى التحتية لتكنولوجيا المعلومات الخاصة بها وإرسال إشعارات وتنبيهات إلى المسؤولين في حالة حدوث أخطاء أو مشاكل تقنية. هذا التقرير المسجل كجزء من أطروحة التدريب للحصول على "الماجستير في الشبكات والأنظمة الموزعة" في جامعة تلمسان ، يقدم حلاً لتجنب هذه الأعطال في البنية التحتية التي يمكن أن يكون لها تأثير سلبي على صورة العلامة التجارية للشركة. يتكون العمل المنجز من إنشاء أداة لمراقبة البنية التحتية لتكنولوجيا المعلومات على أساس البرمجيات الحرة و المجانية بين حالتي الاستخدام ELK stack مع Elastalert و Zabbix

الكلمات المفتاحية: ELK Stack، Elastalert، Zabbix، Monitoring، Logs، , Alerting , Elasticsearch , kibana

Table of contents

General Introduction

Nowadays, we talk about medium-sized and big companies when the network depends on their size Their IT equipment consists of hundreds of interconnected pieces (servers, scanners, routers, switches, ...). This perspective generates a serious problem not only for intervention and maintenance but also an unexpected breakdown management.

All types of business rely on their network that should work properly at all times. Indeed, a network is the place where the important business data is stored and where the mission-critical applications are running . When an issue arises and any component of the network is down it could have major consequences on business performance. For this reason, it is absolutely essential to invest in an IT infrastructure monitoring solution today. To achieve this, it is necessary to keep an eye on these devices in terms of availability, health and performance. A proper functioning of the IT infrastructure, therefore, requires the implementation of a supervision tool. This security policy allows monitoring the IT infrastructure, detecting possible intrusions and alerting the proper functioning and malfunctioning of the system.

The malfunction of a system is declared by messages on the supervision console, sending emails or SMS to the system administrator or the whole team which is the main important option in monitoring tool to directly find the solution. There are two types of monitoring tools for an IT infrastructure: paid and free monitoring tools based on open-source software.

The main interest of this project realized in Kloufi , our hosting company, is to find a free IT infrastructure monitoring solution based on alerting that could be integrated with the old monitoring solution to make alerting in real-time by install ,configure it and do the monitoring.

This report is structured in four chapters. In the first part of Chapter A, we will present a global view of IT infrastructure, describe its component, talk about the solid IT architecture and we will list the consequences when an IT infrastructure is not monitored properly. In the next part, we will present and define the concept of monitoring, describe its approaches, then we will present monitoring tools architecture and what are the criteria to choose a tool, and finally we will list the most popular monitorings tools.

In chapter B, we will give a general description of the company, we will also talk about the IT environment and conditions . Then, we will specify the main problem and propose a solution, Finally we will list the project objectives.

Through chapter C, we will test and make a deep study about the two proposed solutions by giving in details the installation procedures and configuration , and monitor the whole infrastructure

Finally, in chapter D, we will compare the two solutions by providing the advantages and disadvantages of each, to end with a choice of the suitable solution for the company. This work is concluded with a general conclusion , then a list of references , appendices and figures.

Chapter A: IT Infrastructure Monitoring

A-I Introduction

The term IT infrastructure is defined in ITIL as a combined set of hardware, software, networks, facilities, etc. in this chapter we will give a general idea about IT infrastructure, the major components, we will explain also what makes the IT architect solid and the consequence if there is not a monitoring tool in IT department. In the other hand, we will talk about monitoring and its approaches and we will move to the architecture of monitoring tools and list the most popular solutions of monitoring

A-II IT Infrastructure in General

A-II-1 IT Infrastructure Definition

A company's IT infrastructure is, in a way, the skeleton that allows all its organs to function properly without problems. The IT infrastructure includes all of a company's hardware and software equipment as presented in Figure 1. All of these elements, connected to each other. It is also referred to as a computer system, computer architecture or computer network. This equipment requires well-managed installation and maintenance to ensure that the company has good services [1].

A-II-2 Components of IT Infrastructure

IT infrastructure is composed of these major components:

A-II-2-1 Software and Applications

Is a critical aspect of IT infrastructure monitoring. Software applications deployed on servers may be used by members of IT organization or by customers of the business. In either case, applications represent a potential attack vector for a malicious actor and a powerful source of operational and business intelligence.

A-II-2-2 Access and Devices

The things there are connected to the network.

A-II-2-3 Servers and Storage

Things that provide services for devices to consume, like Servers, Storage Security, Cloud Storage, Virtualization, Hyperconvergence.

A-II-2-4 Professional and Managed Services

The services that purchase to deliver IT projects or ensure they’re always working like Service desk, Risk assessments, Software/Hardware audits, Engineer visits, Network upgrades, Change management, Installations and decommissions.

A-II-2-5 Networking and Telecommunications Platforms

As a network increases in size and importance, so does the need to ensure the network runs and keeps running effectively. The network infrastructure is part of global IT infrastructure, it contains devices which are Switches, Routers, Firewalls, Wireless access points, Physical cabling . The entire network infrastructure is interconnected [2].

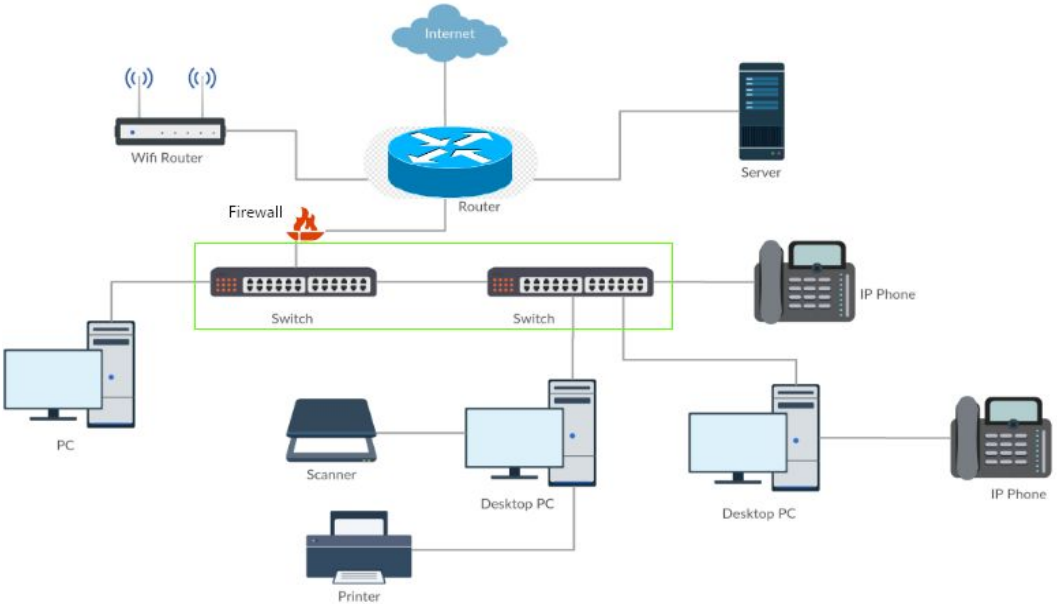


Figure A-1: IT Infrastructure Architecture

A-II-3 Solid and Efficient IT Architecture

The objective of every organization is to build a solid architecture in terms of availability, responsiveness and performance. Here are a few important elements to be taken into account.

A-II-3-1 Activity Continuity

Ensure that the network and data are accessible at all times. Have a disaster recovery plan in place so that even if a problem occurs should be detected early and resolve it, so users are not adversely affected and business continues seamlessly.

So this is the main goal of our study, trying to find the best solution to make the IT infrastructure monitored in real-time so we get a notification before IT infrastructure or servers are down.

A-II-3-2 Application Performance

Load time or system response speed are crucial elements that have an impact on the use of applications, they can crash for users if they are not sufficient. The overall performance of the company depends on the performance of the applications.

A-II-3-3 Flexibility of Infrastructure Evolution

It is important to anticipate future changes so that the IT infrastructure can remain robust for users. The growth of a company induces the development of the computer park, creates new needs in terms of equipment, storage capacity, speed of response, etc.

A-II-3-4 Storage Capacity

Ensuring that there is sufficient storage capacity for all company data and that this capacity increases as the business grows so the company need to monitor it and get notified before the storage run out.

A-II-3-5 System Security

The infrastructure must be built in such a way as to avoid security threats, which could lead to loss of data, for example. Applications and physical equipment must be protected to avoid such threats and ensure continuity of service under optimal conditions.

In our project or the monitoring tool we will choose is based on the activity continuity , flexibility and storage capacity .

A-II-4 Consequences when an IT Infrastructure isn't Monitored Properly

We noticed that the most popular problem in all business categories is they wait till when servers are down or web site get 404 error then they start searching the problem and try to resolve it , this strategy outputs many Consequences, here the most important of them [3].

A-II-4-1 Increased Downtime

Without this kind of automatic solution, we are left to wait until the problem is manually detected, which could take hours or even days to diagnose properly. In addition, an IT infrastructure monitoring solution provides remote access to the server as opposed to the IT professional needing.

A-II-4-2 Problems Needs to be Found Manually

When a problem arises, without a monitoring solution we are left to find a highly experienced IT professional that can do some digging to find the root cause of the problem.

A-II-4-3 Security Threats

Without an IT infrastructure monitoring solution, we wouldn't know it until after security was breached and the network became the victim of a Malware or Spyware attack that was able to get through the security system.

A-II-4-4 Excess IT Costs

It may cost a lot of money in the long run if we don't have a monitoring solution in place and are left with frequent network or application downtime which means that work isn't getting done which in turn results in less profit.

A-III- IT Infrastructure Monitoring & Tools

A-III-1 Introduction

IT Infrastructure Monitoring measures the real-time availability of all IT resources from a single console common to all IT teams, providing them with greater proactivity. Infrastructure Monitoring provides a global view of the status of all components whether they are located in DataCenter, at the hosting provider or at a Cloud Provider ….

A-III-2 Monitoring Definition

Monitoring is a continuous and systematic collection, analysis, correlation, and evaluation of data related to the state and behaviour of monitored entities. Monitoring works with data of a monitored entity. An entity is part of the infrastructure that is monitored like application, database, router, operating system, machine, or hard disc.

The kind of data collected depends on the use-case and what we need to monitor for example operating system events, firewall logs, application errors, web server response time and memory usage. The collected data typically contain some sort of timestamp. With timestamp, the data can be ordered into time series that can give more insight. Collecting the data adds no value. They need to be analyzed and then acted on the basis of results [4].

A-III-3 Monitoring Approaches

There are two popular monitoring approaches, Agent-based and Agentless. Recently new methods are being introduced that encompasses Agent-based and Agentless advantages into one hybrid approach or collection of monitoring metrics through data streams [5].

A-III-3-1 Agent-based Approach

An agent-based approach is a platform-dependent and requires additional software on monitored systems. It provides in-depth monitoring data as agents are domain-specific and are designed to collect every possible metric. On the other side, this introduces limitation in scalability as the solution cannot be easily deployed in an organization that uses multiple platforms, systems and applications. The overall architecture of the agent-based approach in a typical application, database and web server environment.

A-III-3-2 Agentless Approach

An agentless approach utilizes systems built-in monitoring technologies and protocols such as Windows Management Instrumentation (WMI) and widely available Simple Network Management Protocol (SNMP). It is a lightweight solution as it doesn't require additional software to be installed as well it is much easier to deploy in a distributed environment.

A-III-3-3 Hybrid Approach

A hybrid approach provides a new way of collecting data as it combines the benefits of both agent-based and agentless approaches. To meet all monitoring requirements it allows choosing between traditional monitoring approaches as well as gives an interface to integrate with custom monitoring scripts and agents. This enables full flexibility and scalability in a distributed environment.

A-III-3-4 Data Streams Approach

With the evolution of distributed systems to Heterogeneous and Cloud environments, a new approach was developed to measure availability and performance from an application and service perspective. This approach focuses on business transaction execution and validates end-to-end path from the user initiating the task to the connected systems in the infrastructure.

A-III-4 IT Infrastructure Monitoring Tool Architecture

General-purpose infrastructure monitoring tools typically utilize a client-server model by installing an agent in every system to be monitored. Figure 1 shows this general architecture. Monitoring agents measure the metric values from monitored components and send them to the monitoring server. The server stores the collected metrics into database analyses them and sends alerts. It is the same architect for the monitoring system architect we will work on later.

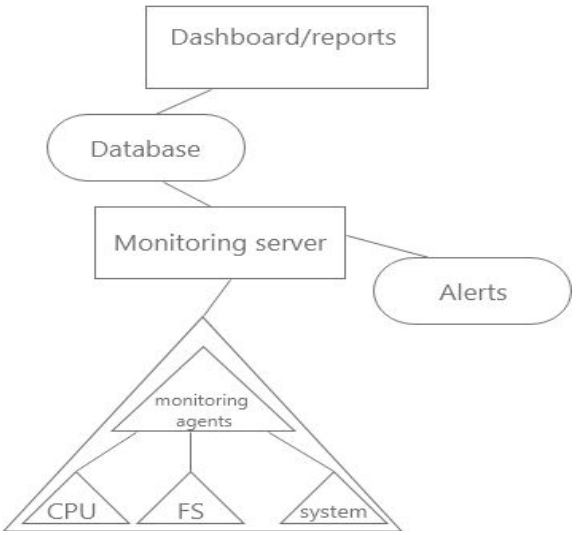


Figure A-2: General Monitoring Tools Architecture

A-III-5 Choose The Right Network Monitoring Solution

An important part of any system administrator's job is to monitor the whole system for performance, traffic usage, CPU and memory, availability, etc.

A high-performance system is a basic 'must-have' for a functioning IT infrastructure in any company. Of course, every company has different requirements for a monitoring solution, and as the market offers numerous different tools and solutions, careful selection of a suitable solution is a must. We will see in the next chapter **kloufi's** requirements and witch tool suits them but first to know the requirement we should answer the following question below:

- What should the monitoring system be able to accomplish?
- Are there already concrete plans for expansion that should be considered in planning? Do upgrade options make the solution future proof?
- Should comprehensive monitoring be performed over the entire network, or should only specific areas be monitored?
- Which protocols and technologies support the solution regarding bandwidth and availability monitoring?
- Is centralised monitoring of distributed locations possible? What data is collected by the solution? How are these evaluated?
- Is there a long-term data archive that would provide the foundation for trend analysis?
- How does the solution alert personnel in case of an emergency?
- Is the solution's structure user-friendly and can it be operated intuitively?
- Does the manufacturer provide sufficient support and how about the budget?

A-III-6 IT Infrastructure Monitoring Tools

IT infrastructure Monitoring is important for availability, troubleshooting, and to save time & money. It saves time and money that could be required for the investigation in case of any issues. This technology will give the visibility and the admin will be able to plan for the changes accordingly, below are the most popular IT infrastructure Monitoring tools that are available in the market:

A-III-6-1 ManageEngine OpManager

OpManager provides comprehensive network analysis capabilities and allows global network monitoring. It analyses the performance of a network in real-time and monitors it with proactive fault detection. Also, it guarantees business continuity by maintaining high availability. It empowers network admins to simultaneously perform multiple operations such as Network performance monitoring, Bandwidth analysis, Firewall management, Storage Monitoring, IP Address Management (IPAM) and Switch Port management (SPM) [6].

A-III-6-2 PRTG

PRTG is network-monitoring software that can run on a Windows machine within a network. It can collect statistics from designated hosts such as routers, servers, switches and other important devices or applications. Its benefit is that it can detect problems before they deteriorate into faults and by alerting these problems to a network administrator, many costly service outages can be avoided. Furthermore, RPTG is free for small businesses tracking less than 25 devices [7].

Nagios

A-III-6-3 Nagios

Nagios is probably the most popular network monitoring tool partly because it is the oldest. It monitors specified hosts and services, alerting when systems are down and when they return to normal operation. It is free software under the GPL license.

It is a modular program that consists of three parts: the application engine that schedules the monitoring tasks, the web interface which gives an overview of the information system and possible anomalies, and the probes (called plugins), a hundred or so mini-programs that can be completed according to each person's needs to supervise each service or resource available on all the computers or network elements of the IS [8].

ZABBIX

A-III-6-4 Zabbix

Zabbix is a free (open source) application for the supervision of systems and networks in IT infrastructure, developed in C. The web interface is developed in PHP and JavaScript. The server and proxy versions which are exclusively on Unix, Zabbix is multi-platform and is available under operating systems such as Windows, Linux, Solaris, etc.

Zabbix can supervise and check the status of a multitude of network services, or systems, while monitoring at the hardware level many types of equipment present in an IT infrastructure, such as a router, a printer, an IP phone, etc. Thanks to the use of the SNMP protocol. Zabbix also supports the IPMI protocol [9].

A-III-6-5 NetCrunch

NetCrunch (not free software) is a multi-platform network monitoring software. It combines agent-free monitoring of Windows, Linux, Mac OS X, BSD, NetWare, SNMP. It unifies fault management by collecting events and creating alerts from SNMP sources, Windows event logs and Syslog servers. It automatically detects TCP/IP nodes and represents the physical/logical topology of the network in dynamic maps and views, customized or not.

NetCrunch 6 has more than 65 built-in network services monitors (HTTP/S, POP3, FTP, SMTP, DNS) and offers user experience-based monitoring. It monitors performance counter values in real-time, on graphs [10].

A-III-6-6 SolarWinds

SolarWinds Network Performance Monitor (NPM) is an advanced network monitoring software that scales and expands with the needs of the network. Key features include multi-vendor network monitoring, network insights for deeper visibility, intelligent maps, NetPath and PerfStack for easy troubleshooting, smarter scalability for large environments, advanced alerting, and much more, which is not free [11].

A-III-6-7 Spiceworks

Spiceworks is Monitoring solution that offers real-time updates on servers, switches, and any IP device. It is designed for companies that monitor less than 25 devices. Thousands of companies use Spiceworks to monitor critical infrastructure, including switches, servers, and IP devices [12].

A-III-6-8 ELK Stack

The ELK Stack is the world's most popular log management platform it is open source. In contrast, Splunk — the historical leader in the space — self-reports 15,000 customers in total. the ELK Stack was a collection of three open-source products — Elasticsearch, Logstash, and Kibana all developed, managed and maintained by Elastic. The introduction and subsequent addition of Beats turned the stack into a four-legged project and led to a renaming of the stack as the Elastic Stack [13].

A-IV- Conclusion

In this chapter, we saw what is IT infrastructure, its main components like network, software applications and others. After we talked about monitoring in computer science some approaches, and why we should monitor our infrastructure until it became so important for a network engineer to know this notion.

At the end we talked about multiple IT monitoring software and solutions, there are multiple great software but each one has its own advantage and disadvantage, two tools we choose to work with will be the subject of chapter C, but first let talk about our hosting company

Chapitre B: Work Environment

B-I Introduction

In this chapter, we will give a small presentation about kloufi startup and a brief description, will talk about professional benefits and also our work environment, in the other hand, we will present the major problem we want to solve and give the proposed solution and the objective of this project.

B-II Company Description

Kloufi is a search engine that looks for public data on different Algerian advertising and E-shops websites and displays them in milliseconds. The user can get all ads related to real estate, cars, electronics and even job offers in real-time. Users as (agencies, businessmen, and people) usually have the trouble of navigating to different sites so Kloufi came to help reach results that meet the needs by determining what the user is looking for by using its various filters [14].

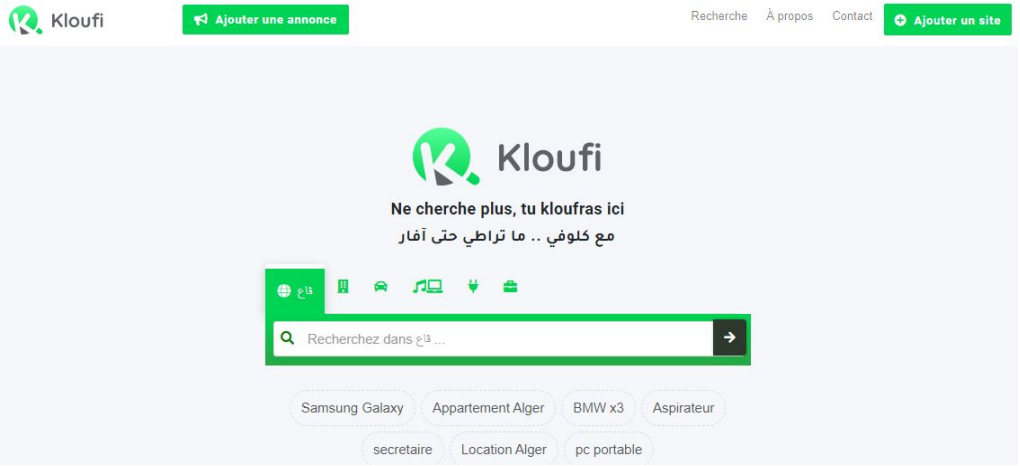


Figure B-1: kloufi Search Engine

B-III Sites Benefits

Kloufi will provide an opportunity for various web sites and new startup to emerge, and obtain a number of visits from interested parties, it will also help high-quality advertising to achieve sales, encouraging others to improve the way they place their ads and thus will be filled with Algerians sites with quality ads.

B-IV Professionals Benefit

Kloufi's target is to organize and arrange public information and set up a mechanism that enables professionals and business owners to filter data so that they can get good deals without wasting time and effort in browsing ads that do not suit their needs.

With just a few clicks on the Kloufi website, we will be able for example to exclude ads that do not contain images, or that do not contain price! we also can even determine the budget in dinars

or in other currencies with the appropriate rate, we can find many other filters that can be added before clicking on search, to get by the end accurate results from different Algerians websites.

B-V Kloufi Specialization

Kloufi is a search engine with specific terms of reference, unlike major search sites such as Google or Yahoo. it contains five search terms: Cars, Real estate, Electronics, Home Appliances and Job offers. The objective of the kloufi is to train a generation of developers, data scientists, cloud architects capable of accompanying technological development. Another goal is to give enthusiasts the opportunity to express their talents through an innovative and rewarding project.

B-VI Proposed Solution & Our Objectives

To satisfy the needs of our host company, we thought of setting up an IT infrastructure monitoring tool based on free software. This solution must be integrated with Elk stack to send alerts in real-time in case of infrastructure failure, this solution also will enable the administrator to intervene more quickly in case of breakdowns, to manage his network in the desired way and to offer the Kloufi a significant financial gain. After working on this we need to compare this monitoring solution with another performance and popular free solution which is Zabbix, to see if it really suits the company or not.

B-VI-1 Methodology Used

B-VI-1-1 Comparative Methodology

The comparative method consists of comparing two objects(companies, softwares , hardwares …) that have the same characteristics and similar profile. The comparative method, also known as the scale method. So for us, we will compare two systems that has the same goal.

B-VII IT Environment

The system administrator gave us remote access to the system we used as remote access client Remote Desktop Protocol “RDP”.

B-VII-1 Remote Desktop client

We used a Microsoft Remote Desktop client to connect to a remote session and from almost anywhere using any device. We connected to our session and have access to all of the apps, files, and network resources.

We installed the application from the Microsoft docs [15] and access to the session by entering the URL of the session with port :

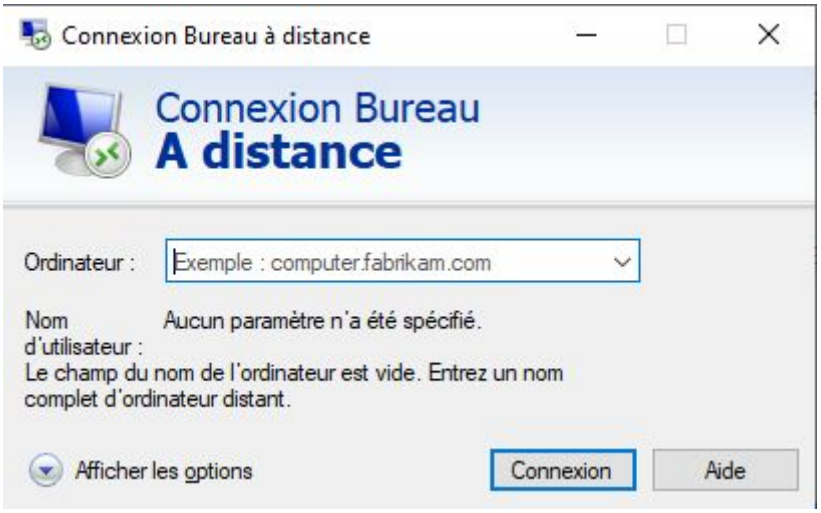


Figure B-2: RDP Dashboard

B-VII-2 Software Environment

- **For Elk Stack and Elastalert Work :**

Under a computer: Windows 10 64bits, here what we worked with :

B-VII-2-1 CentOS 7

CentOS is based on the open-source code of RHEL (Red Hat Enterprise Linux). It provides an enterprise-level operating system free of charge. The first version of CentOS was released in 2004 and was named CentOS 2. In fact, it was named so because it was based on RHEL 2.0. The latest version is CentOS 7. CentOS is probably better for companies that want a more stable and secure Linux distribution [16]**.**

B-VII-2-2 SuperPutty

SuperPutty [17] is a Windows application used primarily as a window manager for the PuTTY SSH Client. It allows to embed PuTTY terminal instances inside of windows form providing a better-tabbed interface when multiple connections are used. Additionally, SuperPutty has support for using pscp.exe to transfer files to and from a remote host. Local terminal sessions can be started with MinTTY. SuperPutty does not do any ssh or terminal management itself since PuTTY does an excellent job of this.

B-VII-2-2-a License

Licensed under a liberal MIT/X11 license, which allows this program and source code to be used in both commercial and non-commercial applications. The complete text can be found in the License.txt file included with the download.

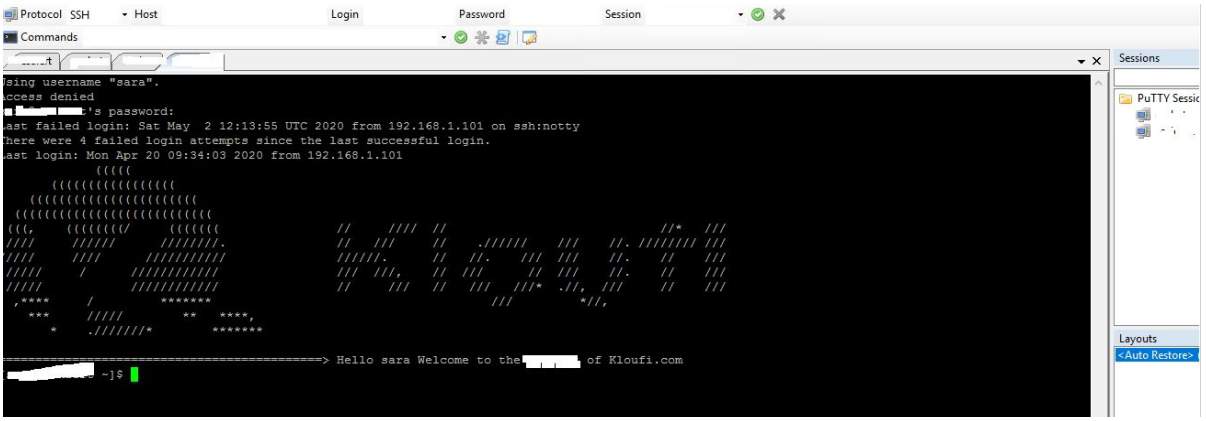


Figure B-3: Super-putty Dashboard

- **For Zabbix Work**

When we wanted to test Zabbix, the technical team of kloufi company needed to make some tests, so we couldn't work with remote machine cause was inaccessible, we use

Computer: Windows 8.1 , 64 bits we needed to use a virtual machine which is:

B-VII-2-3 Oracle VM VirtualBox 04

VirtualBox is a powerful x86 and virtualization product for enterprise as well as home use. Not only is VirtualBox an extremely feature-rich, high-performance product for enterprise customers [18]. We used under it Ubuntu Desktop 16.04 LTS iso with 2 GB system memory.

B-VIII Problem

In Kloufi IT department, The system administrators use as a solution to monitor and manage logs for the whole IT infrastructure: Elasticsearch Logstash, Kibana, and Beats for shipping the logs (which is called the ELK Stack). It is a reliable and secure suite that allows them to collect any data format from any source and then query, analyze and visualize the data in real-time, and as we know in any monitoring solution we need to send alerts to the IT employees if there is any problem.

Elk stack has this option witch called X-Pack it is an Elastic Stack extension that provides security, alerting, monitoring, reporting, machine learning, and many other capabilities. By default, when we install Elasticsearch, X-Pack is installed.

So if we want to try all of the X-Pack features, we can start a 30-day trial. At the end of the trial period, we need to purchase a subscription to keep using the full functionality of the X-Pack components. But it’s so expensive for a small company like Kloufi, so it needs a free solution for alerting that can be integrated with ELK Stack. In parallel there is a concurrent free monitoring solution which is Zabbix, so we cannot choose what is good for Kloufi just when we test them, because of that, we will compare them to decide what is the best solution for the company.

B-IX Company Requirements

Free monitoring solution which can monitor the metrics (CPU, memory usage, files system, process) with some conditions which are presented below to get notified, the availability

and log files of the servers below, the activity continuity is important point and flexibility, a modern dashboard to visual data, real-time alerting, easy implementation, performance tool for big data and easy to work with it. The Global system we are working on is :

Elasticsearch: Elasticsearch is a server using Lucene for indexing and searching data. It provides a distributed and multi-entity search engine through a REST interface.

HAProxy: Is free and open-source software that provides a load balancer and a high availability proxy server for TCP and HTTP applications that distribute requests across multiple servers.

PFSense: Is an open-source router/firewall based on the FreeBSD operating system.

MySQL: Is a relational database management system. It is distributed under a dual GPL and proprietary license.

Kibana: Is a data visualization plugin for Elasticsearch published under the free Apache version 2 license. It provides visualization functions on content indexed in an Elasticsearch cluster.

PostgreSQL: Is a free and open-source relational database management system with an emphasis on scalability and SQL compliance.

Nginx: Is an open-source webserver software and reverse proxy written by Igor Sysoev.

B-IX-1 Conditions

- Monitor CPU usage and send email alert at 90%.
- Monitor Memory usage and send email alert at 90%.
- Monitor Disk space usage (File system) and send email alert at 80%.
- Monitor system process and send an email alert if stops running.

B-X Conclusion

In this chapter we saw all the details about our hosting company kloufi, we also saw the IT environment, We touched upon the core issue and what are the proposed solutions. Before we choose the right solution which suits the company, we will test the proposed solutions after the whole installation and configuration and monitor with them our system, all that we will see in the next chapter.

Chapter C: Distributed IT Infrastructure Monitoring System

C-I Introduction

This is the most important chapter. First, we are going to discuss the global view about log management by defining the most popular tool ELK stack for monitoring and with what it works. Also, we are going to see the global architecture of our system and provide details about the two proposed solutions through installing and configuring them properly so we can make our monitoring of all the servers and systems. We are going to start our research work with ELastalert then we will move to Zabbix.

C-II Log Management

It is the act of dealing with large volumes of computer-generated log messages. Nowadays, almost all monitoring software use log management because they work with logs, so it is crucial to understand the mechanism of logs and log management. Log file in computing is a file that records either events that occur in an operating system or other software runs or messages between different users of communication software [19]. And Log management is divided into consecutive parts [4].

C-II-1 Log Generation

The first thing is to know the log entry: from where we will collect data(events, messages) why we should keep this events and time lag between log generation, there are multiple formats of logs the most commune is text so it will be readable.

C-II-2 Log Collection

We may have multiple log entry with multiple messages structure. Thus, they are collected and transported to one log; this task is often made by a software called log collector.

C-II-3 Log Transformation

Here we will change the representation of the log if necessary by normalizing it or adding/removing data from it, of course, we can always use software to make the task easier.

C-II-4 Log Storage

This is an important part of log management; how to store and where centralized or distributed, and the main problem is in the size of these documents that can get to terabytes or

petabytes. People nowadays store logs in the NoSQL database because it is Schemaless and may contain a big amount of data.

C-II-5 Log Analysis

This is the core part which will help us for monitoring. Here, we have to find the most important and valuable events. Some approaches can be used here like Complex Event Processing which will look for events that will happen if a series of events happened before. So we will analyse our logs to see if a server shutdown for example.

C-III ELK Stack

Elastic Stack, or ELK Stack, is a representative of an open-source monitoring solution.The term ELK Stack was coined as an acronym of Elasticsearch, Logstash, and Kibana. Later, Beats was introduced and the stack was renamed to Elastic Stack. It is widely used as a platform for Log management. Coverage of Elastic Stack monitoring capabilities on Log management model is illustrated in **Appendix A**.

Elastic Stack is a popular solution which has numerous third-party plugins and extensions. There is also the "official" extension, X-Pack with added security, alerting, and reporting capabilities to name but a few. However, X-Pack is not an open-source solution, there are multiple licenses of Elastic Stack: Basic, Gold, Platinum, and Enterprise. The Basic license is for free and the rest are paid licenses. X-Pack is available under Basic license, but only with a small subset of features. Paid licenses contain premium support. Elastic Stack is also offered on the cloud as a SaaS solution. For our case, Kloufi company has the basic licence of ELastic stack [4].

C-III-1 Elasticsearch Definition

Elasticsearch [13] is the distributed search and analytics engine in the heart of the Elastic Stack. It is where the indexing, search, and analysis magic happen. Elasticsearch provides real-time search and analytics for all types of data. Whether we have a structured or unstructured text, numerical data or geospatial data, Elasticsearch could efficiently store and index it in a way that supports fast searches.

Elasticsearch is categorised as a NoSQL database. Elasticsearch stores data in an unstructured way and up until now we could not query the data using SQL. The new Elasticsearch SQL project will allow the use of SQL statements to interact with the data.

C-III-1-1 Elasticsearch As a Log Search Tool

Elasticsearch provides a horizontally scalable search and supports multithreading. Search indexes can be divided into segments that could have several replicas. Several segments can be placed on each node which acts as a coordinator for delegating operations to the correct segment,

rebalancing and routing are performed automatically. Related data is often stored in the same index which consists of one or more primary segments and possibly multiple replicas.

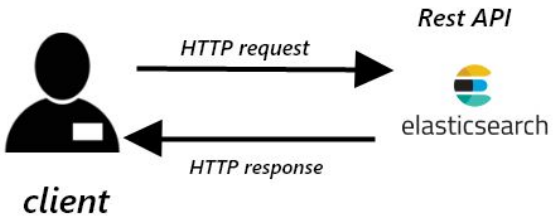


Figure C-1: Querying Elasticsearch

C-III-1-2 Elasticsearch Clusters and Nodes

When we start an instance of Elasticsearch, we are starting a node and we have a cluster with single node. We start another instance of Elasticsearch that has the same cluster name with the first instance so we have a cluster with two nodes. We can start more instances of Elasticsearch to form a cluster with the number of nodes we want. In our case, we had one cluster with six nodes. Each node in the cluster recognises the others within the cluster. They communicate with each other directly through the native Elasticsearch language over TCP. This is known as Fully Connected Mesh topology.

Each node in the cluster plays one or more roles. It can be a master node which is responsible for creating, deleting indices, adding or removing the nodes from the cluster. Data node is for holding the data in the shards and performing data related operations such as create, read, update, delete, search... Client node is for routing the cluster-related requests to the master node.Each role has its own purpose.

As a cluster grows , Elasticsearch automatically migrates shards to rebalance the cluster. There are two types of shards: primaries and replicas. Each document in an index belongs to one primary shard. A replica shard is a copy of a primary shard. It provides redundant copies of your data to protect them against hardware failure and increase capacity to serve reading requests like searching or retrieving a document.

C-III-2 Logstash

“Logstash is an open-source, server-side data processing pipeline that ingests data from a multitude of sources simultaneously, transforms it, and then sends it to your favourite stash.” — quote by Elasticsearch Co.

In Logstash, we define pipelines in which we specify the data sources from which we want to collect our data, followed by filters if necessary, and finally, a data sink to which we can send our data [20].

C-III-3 Kibana

Kibana is a window into the Elastic Stack. It enables visual exploration and real-time analysis of data in Elasticsearch. It allows developers to create different types of dashboards that could be checked everyday to monitor the metrics already defined [20].

C-III-3-1 Configuration

The configuration .yml for the Kibana service could be found in kibana.yml → kibana.It basically contains similar configurations to the other services: where to fetch the metrics so they can be represented in visualizations , where to write logs and time interval to fetch metrics, etc.

Once Kibana is up and running , the first thing we have to do is to create an index pattern for one or more indexes from Elasticsearch. In our case, the company has the whole installation of ELK stack as monitoring solution and it has already created its index in kibana . Consequently, we can visualise any metrics we want.

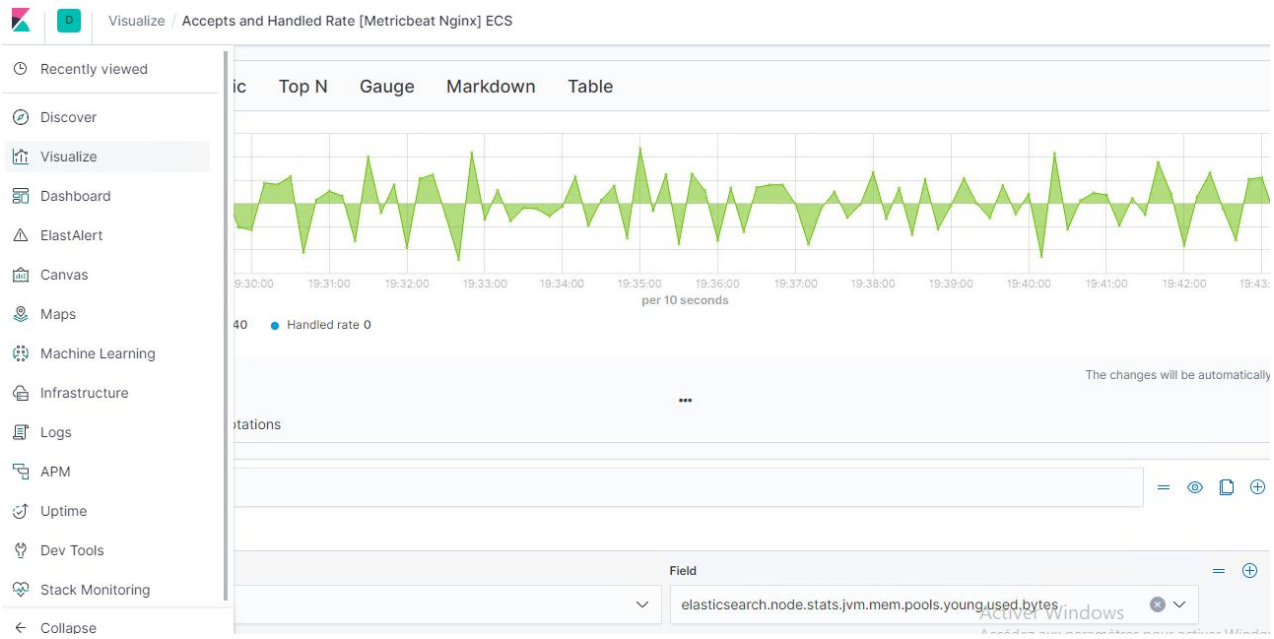


Figure C-2: Kibana Visualisation Dashboard “Nginx Metrics”

C-III-4 Beats: Collect, Parse and Ship

Beats are open source data shippers that could be installed as agents on servers to send operational data to Elasticsearch. Elastic provides Beats for capturing.

Audit data	Auditbeat
Log files	Filebeat
Cloud data	Functionbeat
Availability of services	Heartbeat

Systemd journals	Journalbeat
Metrics, availability of servers	Metricbeat
Network traffic	Packetbeat
Windows event logs	Winlogbeat

Table C-1: Beats Types

Beats sends data directly to Elasticsearch or via Logstash where data could be further processed and enhanced before being visualized in Kibana. In our case, we used Filebeat and Metricbeat that were already installed in the company servers, in order to satisfy the company objectives which are monitoring the metrics and Log files and make sure of the availability of servers. This picture is an overview of our monitoring system architecture before adding the alerting tool.

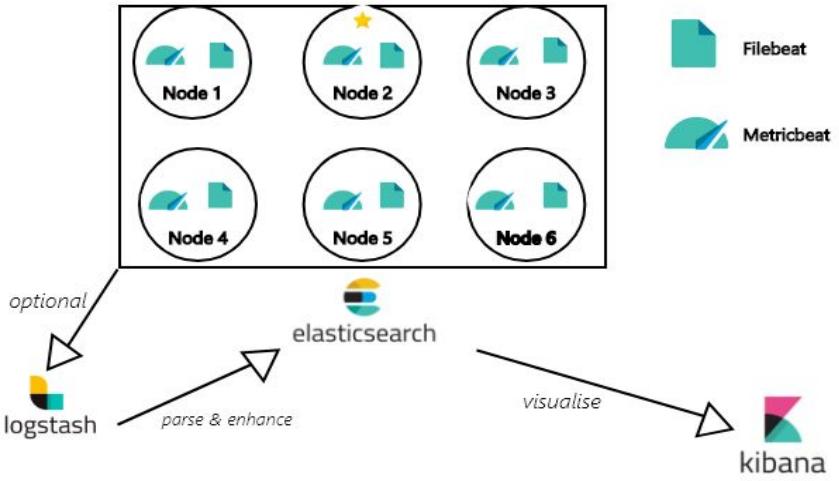


Figure C-3: Elk Monitoring System Architecture

C-III-5 Alerting

As it has been said before, some features like monitoring are for free but the Alerting is paid in X-Pack. That is why, we have searched about alerting solutions that could be integrated with Elasticsearch. We found that the only solution is Elastalert .

C-IV ELastalert

C-IV-1 Definition

ElastAlert is a framework written in Python for real-time alerting on anomalies, spikes, or other patterns of interest from data in ElasticSearch. It combines two types of components: rule types and alerts. It was built by the engineering team at Yelp.

ElastAlert periodically queries ElasticSearch and data are passed to the rule type. The rule type determines when a match is found and a match activates one or more alerts. This process is configured by a set of rules and each rule defines a query, a rule type , and a set of alerts [21].

There are multiple rule types and multiple alert sinks to which warnings could be sent (Email, Slack, etc),we will see them later.

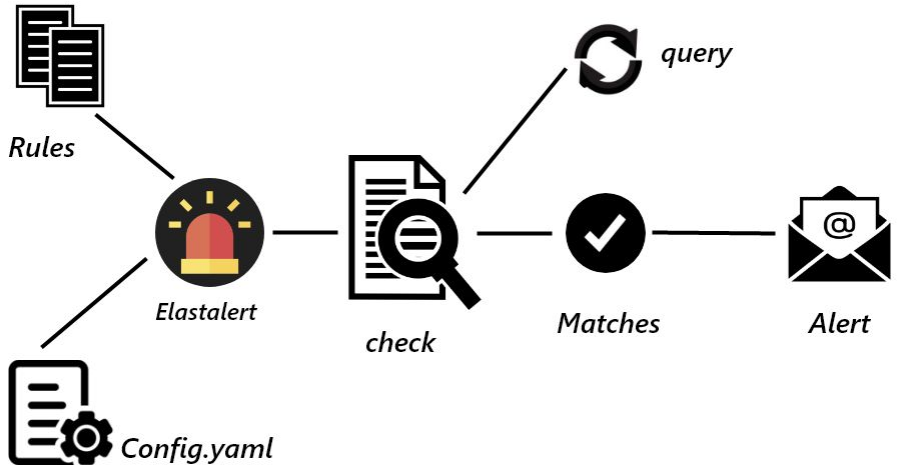


Figure C-4: Elastalert Process

C-IV-2 Elastalert Installation

C-IV-2-1 Requirements

Before we start the installation, we install the requirements which are listed in the requirement text in the repository of Elastalert, then we install Python 3.6 and pip.

- Install Python 3.6.4 on CentOS 7 from a Repository:

```
sudo yum install -y https://centos7.iuscommunity.org/ius-release.rpm.
sudo yum update
```

- Download and install Python:

```
sudo yum install -y python36u python36u-libs python36u-devel python36u-pip
```

Once these commands are executed, simply check if the correct version of Python has been installed by executing the following commands: `python3.6 -V`

C-IV-2-2 Installation

- After that we install pip

```
sudo yum install python3-pip  /  sudo pip3 install --upgrade pip
```

- Installation of the latest version of elastalert using pip

```
$ pip install elastalert
```

- Finally, we install the modules

```
$ pip install "setuptools>=11.3"
$ python setup.py install
```

Now elastalert is available in our machine, here are all the files we found :

```
[sara@ESAlert elastalert]$ ls
Dockerfile-test  build               dist                elastalert.egg-info  requirements.txt      supervisord.conf.example
LICENSE          changelog.md        docker-compose.yml  example_rules        setup.cfg             tests
Makefile         config.yaml         docs                pytest.ini           setup.py              tox.ini
README.md        config.yaml.example elastalert          requirements-dev.txt smtp_auth_file.yaml
```

Figure C-5: Elastalert Files

C-IV-3 Configuration

The first thing to do after the installation is the creation of Elasticsearch index. ElastAlert saves information and metadata about its queries and alerts back to Elasticsearch. This is useful for auditing, debugging and it allows ElastAlert to restart and resume exactly where it is left off. It is not required for ElastAlert to run but it is highly recommended, so we need to create an index for ElastAlert by running:

```
$ elastalert-create-index
New index name Default elastalert_status)
Name of existing index to copy (Default None)
New index elastalert_status created
Done!
```

Then, we move to the global configuration of Elastalert. We created a copy of config.yaml.example, we renamed it config.yaml , here our global config.yaml file:

```
# This is the folder that contains the rule yaml files
# Any .yaml file will be loaded as a rule
rules_folder: example_rules

run_every:
  minutes: 1

buffer_time:
  minutes: 15

es_host: elastic2
es_port: 9200

writeback_index: elastalert_status
writeback_alias: elastalert_alerts

alert_time_limit:
  days: 2
```

Figure C-6: Elastalert Configuration

In the above file, we defined the instance where Elastalert applies its rules (an Elasticsearch instance), we defined :

- *Rules_folder*: Where ElastAlert loads rule configuration files from.
- *Run_every*: How often elastalert queries Elasticsearch.
- *Buffer_time*: Elastalert buffers results from the most recent period,it is the size of the query window, stretching backwards from the time each query runs

- *Es_host* and *es_port*: Elasticsearch hostname for metadata writeback and its port
- *writeback_index*: The index on es_host which is used for metadata storage, ElastAlert creates three different types of documents in the writeback index . More about it is available in **Appendix B**.

After this configuration, we create our first rule. However, in order to test it ,configuration should be added to allow Elastalert to send alerts when there are matches.

C-IV-4 Test Alerting with Microsoft Teams

Microsoft teams are the communication tools for Kloufi employees. The full description of how we added and configured it is available in **Appendix C**.

This was just an option that we tested during the setup of the SMTP server because IT department needs to send alerts in Gmail. Currently, Elastalert supports Command Email, JIRA, OpsGenie, SNS, HipChat, Slack, Telegram, Google Chat, Debug, Stomp, TheHive. Now we will explain how to configure SMTP.

C-IV-5 Configuration of SMTP

First of all, we need to create a smtp_auth_file.yaml file which contains the user and the server's password, then we add this to the config.yaml file :

```
smtp_host: "smtp.gmail.com"
smtp_port: 465
smtp_ssl: true
from_addr: "xxxx@xxxx.com"
smtp_auth_file: "smtp_auth_file.yaml"
```

Now we need to create our first rule. As it was stated before, there are multiple existed rules which power ElastAlert. Each rule defines a query to perform, parameters on what triggers a match, and a list of alerts to fire for each match.

C-IV-6 Creation of Rules

Based on the company objectives, we need to create rules. Actually, there were just four available rule types, but after the last version they developed eleven rule types which are available now on Elastalert :

Rule type	Description
Frequency	Matches where there are X events in Y time.
Spike	Matches when the rate of events increases or decreases.
Flatline	Matches when there are less than X events in Y time.

Any	Matches on any event matching a given filter.
Change	Matches when a field has two different values within some time.
Blacklist /Whitelist	Matches when a certain field matches a blacklist/whitelist.
New Term	Matches when a new value appears in a field that has never been seen before
Cardinality	Matches when the total number of unique values for a certain field within a time frame is higher or lower than a threshold.
Metric Aggregation	Matches when the value of a metric within the calculation window is higher or lower than a threshold.
Spike Aggregation	Matches when the value of a metric within the calculation window is spike_height times larger or smaller than during the previous periods of time.
Percentage Match	Matches when the percentage of a document in the matching bucket within a calculation window is higher or lower than a threshold.

Table C-2: Rules Type Description

There is a strong point in Elastalert which enables to create a new rule type that meets the administrator's needs. Because Elastalert is an open source, there are several rule types created by the community . Also, we can create Alerters which are subclasses of Alerter found in Elastalert/alerts.py. They are given matches and perform some action based on that. Alerter needs to implement two members of functions:

- **Alert(self, match):** ElastAlert call this function to send an alert, matches is a list of dictionary objects with information about the match.
- **Get_info(self):** This function is called to get information about the alert to save back to Elasticsearch.

To create a rule, there is some configurations that are required or optional, depending on the rule type chosen, but specify the es_host, es_port, index, type and alert are required settings, one of the optional is "name" ,the name of the rule which must be unique across all rules, if it is not added, it will take by default the file name.

"/etc/elastalert/examples_rules": in this file we have defined the different rules that will trigger a warning in Kloufi IT department Gmail.

C-IV-6-1 Writing Filter

The filters used in rules are part of the Elasticsearch query DSL which is Domain Specific Language based on JSON to define queries.

C-IV-6-1-a Common Filter Types: There are many types of filters, here are the most useful for us and for the community.

- **Query_string:** It is used for partial or full matches to multiple fields, it is structured as follows:

```
filter:
- query:
    query_string:
      query: "name: A"
- query:
    query_string:
      query: "field: value"
```

- **Term:** Allows for exact field matches and for easy combination of multiple term filters:

```
filter:
- terms:
    field: ["value1", "value2"] # value1 OR value2
```

- **Range:** Allows to make range on fields, it is the most useful:

```
filter:
- range:
    Fields:
      from: 0
      to: 3
```

C-IV-6-2 Monitor Metrics Using Metricbeat

Metricbeat uses modules to collect metrics. Each module defines the basic logic for collecting data from a specific service such as Redis or MySQL, Elasticsearch....
A module consists of Metricsets that fetch and structure the data. We only need to enable the modules that we want to run. If we keep the default configuration without enabling additional modules, Metricbeat would collect these system metrics : *core , CPU ,Diskio ,entropy ,filesystem ,Fsstat ,load ,memory ,network ,network_summary , process ,process_summary ,raid ,service [beta], socket ,socket_summary , uptime , users [beta]*. Here are some of the rules created.

C-IV-6-2-a System Rules

Process Rule:

```
es_host: Elastic2
es_port: 9200
name: system.process.summary.stopped
type: frequency
index: metricbeat-*
is_enable: true
num_events: 2
timeframe:
  minutes: 1
filter:
- range:
    system.process.summary.running:
      to: 0
alert:
- "email"
email:
```

File System Rule:

```
es_host: Elastic1
es_port: 9200
name: Range Memory
type: frequency
index: metricbeat-*
num_events: 50
timeframe:
    minutes: 1
filter:
- range:
    system.filesystem.used.pct:
      from: 80
      to: 90
alert:
- "email"
email:
```

Memory Usage :

```
es_host: Elastic2
es_port: 9200
name: Range Memory
type: frequency
index: metricbeat-*
num_events: 50
timeframe:
    minutes: 1
filter:
- range:
    system.memory.actual.used.pct:
      from: 80
      to: 90
alert:
- "email"
email:
```

Cpu Total:

```
es_host: Elastic2
es_port: 9200
name: system.cpu.total.pct
type: frequency
index: metricbeat-*
is_enable: true
num_events: 50
timeframe:
  minutes: 1
filter:
- range:
    system.cpu.total.pct:
      from: 80
      to: 90
alert:
- "email"
email:
```

Note: We add the attribute *is_enabled* whether to disable or enable a rule .Disabling a rule means to stop alerting and querying (eventually).

Network Rule:

```
name: system.network.in.error
type: frequency
index: metricbeat-*
is_enable: true
num_events: 50
timeframe:
  minutes: 5
filter:
- term:
    system.network.name: "eth0"
- range:
   system.network.in.errors:
     to: 1
alert:
- "email"
email:
```

Socket Summary (TCP listening):

```
name: tcp_listening
type: any
index: metricbeat-*
is_enabled: true
filter:
- range:
    system.socket.summary.tcp.all.listening:
       from: 0
       to: 10
alert:
- "email"
email:
```

C-IV-6-2-b Servers Rules

Here are some of the rules when we used metricbeat-* as index to monitor our servers

HaProxy Stat:

```
es_host: Elastic2
es_port: 9200
name: haproxy_stat
type: frequency
index: metricbeat-*
is_enable: true
num_events: 50
timeframe:
  minutes: 1
filter:
- term:
    haproxy.stat.status: "DOWN"
alert:
- "email"
email:
```

Nginx Client Requests:

```
name: client request nginx
is_enabled: true
type: any
index: metricbeat-*
filter:
- query:
    query_string:
      query: "module.name: nginx"
- range:
    nginx.stubstatus.requests:
        to: 20"
alert:
- "email"
email:
```

Note : There are two ways to monitor the availability: with Heartbeat or Metricbeat. Heartbeat is a lightweight daemon that is installed on a remote server to check periodically services status and determine their availability. Unlike Metricbeat which only informs if servers are up or down, Heartbeat tells if services are reachable. That is why, we used only Metricbeat.

MySQL Availability Rule:

```
name: MySQL availability
type: any
index: metricbeat*-
filter:
- term:
   module.name: mysql
- query_string:
      query: "type:metricsets"
-query
  - query_string:
      query: "mysql.status.delayed.errors:1"
alert:
- "email"
email:
```

HTTP Availability Response:

```
es_host: Elastic2
es_port: 9200

name: http_response_code_status
is_enabled: true
type: any
index: metricbeat*-
filter:
- term:
   module.name: HTTP
- query_string:
      query: "type:metricsets"
- query_string:
      query: "http.response.code: 200"
alert:
- "email"
email:
```

C-IV-6-3 Elasticsearch and Kibana Rules Using .monitoring Indexes

The company IT team made an update for the whole infrastructure during our internship and they activated the basic version of Elk Stack X-pack monitoring feature which is available in this licence but only with a small subset of features such as monitoring Elasticsearch and Kibana.

Elasticsearch provides internal statistics based on X-Pack [22] which comes for free. Those statistics are written in Elasticsearch index, per default [.monitoring-es-7-]YYYY.MM.DD. By default, the X-Pack monitoring component collects the monitoring data at intervals of 10 seconds and saves the data to the indexes starting with .monitoring-* on our Elasticsearch and Kibana instance.

Currently, the .monitoring-es-7-* and .monitoring-kibana-7-* indexes are used to store the monitoring data. The instance rolls over to a new index each day. The .monitoring-es-7-* index stores information about the cluster status, cluster statistics, node statistics, and index statistics which consume a large amount of disk space. We wanted to test these indexes with Elastalert and here are some of these rules :

Elasticsearch CPU Load Average Rule:

```
name: elasticsearch_metric
is_enabled: true
type: any
index: .monitoring-es-*
filter:
- - query:
    query_string:
      query: "node_stats.node_id:**** AND
      node_stats.os.cpu.load_average.5m: 9 "
alert:
- "email"
email:
```

Kibana Statue Red Rule:

```
es_host: elastic2
es_port: 9200
name: kibana_status
is_enable: true
index: .monitoring-kibana-*
type: any
filter:
- query:
    query_string:
      query: "source_node.name: ELastic2 AND
      source.kibana_stats.kibana.status: red"
alert:
- "email"
email:
```

Cluster status :

```
namme: elasticsearch_status
is_enabled: true
index: .monitoring-es-*
type: any
filter:
- query:
    query_string:
      query: "cluster_state.nodes_hash: **** AND cluster_state..status: red"
alert:
- "email"
email:
```

C-IV-6-4 Monitor Log Files Using Filebeat

To monitor log file, we used Filebeat which were installed in servers. Filebeat, as the name implies, ships log files , written in Go and based on the Lumberjack protocol. Filebeat was designed to have a low memory footprint, handle large bulks of data, support encryption, and deal efficiently with back pressure. Here is is an example of its configuration :

```
filebeat.inputs:
- type: log
  #Change value to true to activate the input configuration
  enabled: false
  paths:
    - "/var/log/postgresql/*"
    - "/var/log/nginx/*"
    - "/var/log/mysql/*"...
```

We specified a list of inputs in the Filebeat.inputs section of the filebeat.yml. Inputs specify how Filebeat locates and processes input data. There are multiple input types, only the logs were needed for our project then the outputs, here is an example of yaml file:

```
output.elasticsearch:
  hosts: ["xx.xx.xx.xx"]
  protocol: https
  path: /elasticsearch
```

Filebeat modules [23] are ready-made configurations for common log types, such as Apache, Nginx and MySql logs… They can be used to simplify the process of configuring Filebeat, parsing the data and analyzing it in Kibana with ready-made dashboards. They contain standard filesets, such as access logs or error logs. The modules are disabled by default so we enabled them in yaml configuration file.

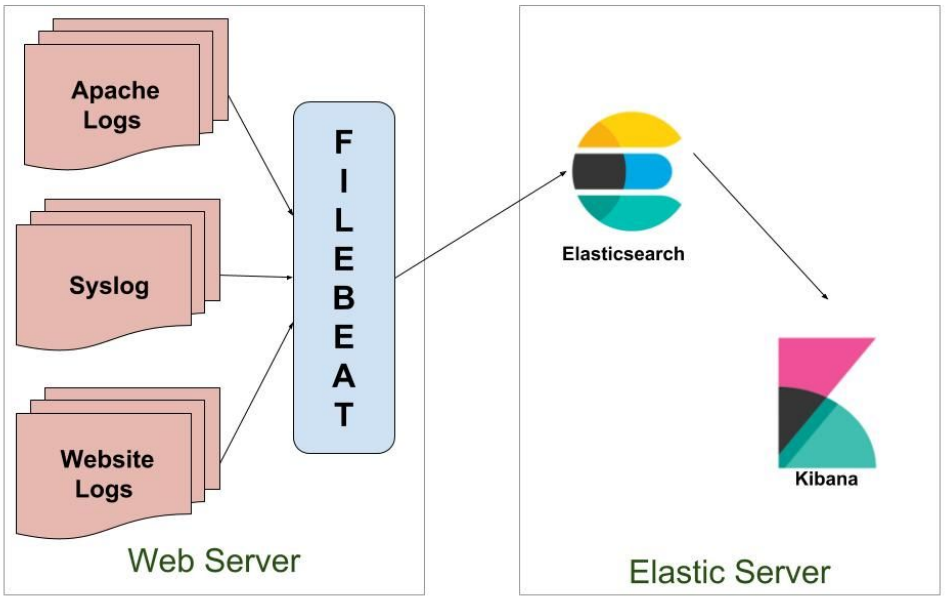


Figure C-7: Filebeats Process

Rules

Monitor all log files are available in Kibana, but we need to send alerts when error log appears, few rules are presented below. Since these rules are similar, we have just modified the module according to the server we want.

Postgresql error log rule :

```
es_host: Elastic2
es_port: 9200

name: rule_postgresql
is_enabled: true
index: filebeat-*
type: any
filter:
- term:
    fileset.module: "PostgreSQL"
- term:
    fileset.name: "log"
- query_string:
     query: "fileset.log : error"
alert:
- "email"
email:
```

haproxy log error rule:

```
es_host: Elastic2
es_port: 9200

name: rule_haproxy
is_enabled: true
index: filebeat-*
type: any
filter:
- term:
    fileset.module: "HAProxy"
- term:
    fileset.name: "log"
- query:
    query_string:
      query: "fileset.log : error"
alert:
- "email"
email:
```

Note: Except Pfsense provides rich information about the state of the firewall, its services, traffic flowing through the firewall, and log data. If we want to receive pfsense logs in elasticsearch, we need to go to pfsense dashboard in remote logging option , check '*Send log messages to remote syslog server*', and enter our ELK servers IP address and custom port, and check 'Everything ' to receive all kind of logs.

C-IV-7 Testing Rules

Once we have created our rules, we wanted to test them by The following command line

```
$ elastalert-test-rule /etc/elastalert/example_rules/elastic_cpu.yaml
```

Running the elastalert-test-rule tool will test if our config file has been successfully loaded and run in debug mode. Here is the results :

```
[sara@ESAlert elastalert]$ elastalert-test-rule --config /etc/elastalert/config.yaml  /etc/elastalert/example_rules/system.network.out.error.yaml
INFO:elastalert:Note: In debug mode, alerts will be logged to console but NOT actually sent.
          To send them but remain verbose, use --verbose instead.
Didn't get any results.
INFO:elastalert:Note: In debug mode, alerts will be logged to console but NOT actually sent.
              To send them but remain verbose, use --verbose instead.
1 rules loaded
INFO:apscheduler.scheduler:Adding job tentatively -- it will be properly scheduled when the scheduler starts
INFO:elastalert:Queried rule system.network.in.error from 2020-06-12 17:15 UTC to 2020-06-12 17:20 UTC: 151 / 151 hits
INFO:elastalert:Alert for system.network.in.error at 2020-06-12T17:17:06.463Z:
INFO:elastalert:system.network.in.error

At least 50 events occurred between 2020-06-12 17:12 UTC and 2020-06-12 17:17 UTC
```

Figure C-8: Running Elastalert in Debug Mode

Elastalert-test-rule its a script that facilitate various aspects of testing. its tasks are to :

- Check that the configuration file is successfully loaded.
- Check that the Elasticsearch filter parses.
- Show the number of hits that match the filter.
- Show the available terms in one of the results.
- Save documents returned to a JSON file.
- Ran ElastAlert using either a JSON file or actual results from Elasticsearch.
- Print out debug alerts or trigger real alerts.
- Show what metadata documents would be written to Elastalert_status

C-IV-8 Run Elastalert as a Daemon

After validating our rules, we wanted to run Elastalert as a daemon. Two ways were found out:

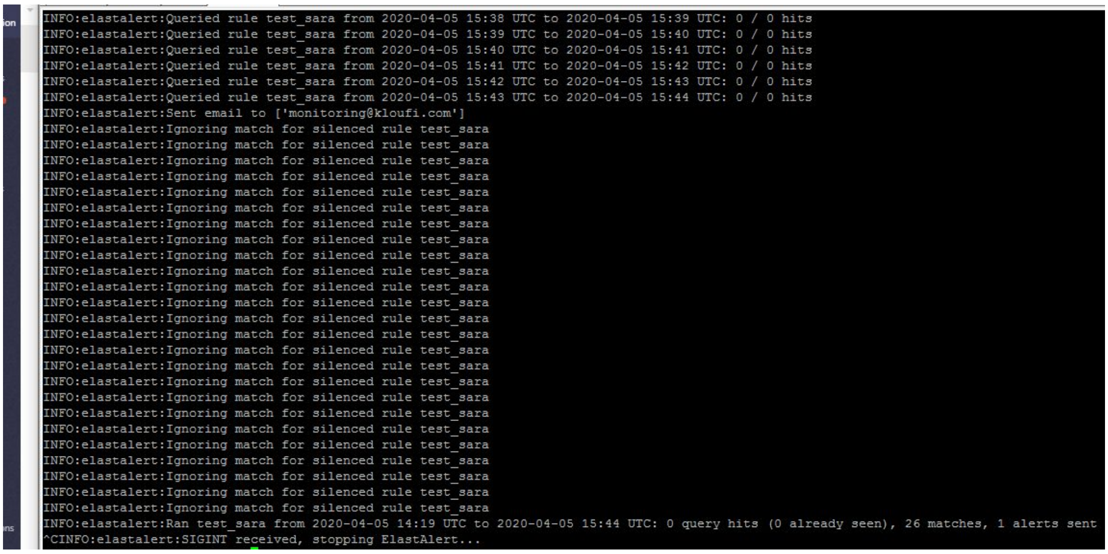
```
$ python -m elastalert.elastalert --verbose --rule example_frequency.yaml
```

or

```
$ elastalert --verbose --rule …
```

For our case, we used the second command which is more useful. Here are the results when we run our first rule which was without filter just frequency rule:

```
elastalert --verbose --config /etc/elastalert/config.yaml --rule /etc/elastalert/example_rules/test.yaml
```



```
INFO:elastalert:Queried rule test_sara from 2020-04-05 15:38 UTC to 2020-04-05 15:39 UTC: 0 / 0 hits
INFO:elastalert:Queried rule test_sara from 2020-04-05 15:39 UTC to 2020-04-05 15:40 UTC: 0 / 0 hits
INFO:elastalert:Queried rule test_sara from 2020-04-05 15:40 UTC to 2020-04-05 15:41 UTC: 0 / 0 hits
INFO:elastalert:Queried rule test_sara from 2020-04-05 15:41 UTC to 2020-04-05 15:42 UTC: 0 / 0 hits
INFO:elastalert:Queried rule test_sara from 2020-04-05 15:42 UTC to 2020-04-05 15:43 UTC: 0 / 0 hits
INFO:elastalert:Queried rule test_sara from 2020-04-05 15:43 UTC to 2020-04-05 15:44 UTC: 0 / 0 hits
INFO:elastalert:Sent email to ['monitoring@kloufi.com']
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ignoring match for silenced rule test_sara
INFO:elastalert:Ran test_sara from 2020-04-05 14:19 UTC to 2020-04-05 15:44 UTC: 0 query hits (0 already seen), 26 matches, 1 alerts sent
^CINFO:elastalert:SIGINT received, stopping ElastAlert...
```

Figure C-9: Running Elastalert in Daemon

It has been noticed that when matches appeared, Elastalert queries send alert in realtime. In this rule, there were 26 matches. Here are the first alerts when they have been sent and arrived to our monitoring gmail account :

Objet	★	De	Date	Taille
ElastAlert: system.cpu.total.pct		[redacted].com	Mer 15:10	3 ko
ElastAlert: system.cpu.total.pct		[redacted].com	Mer 14:54	3 ko
ElastAlert: CPU_usage		[redacted].com	Mer 14:09	3 ko
ElastAlert: test_sara		[redacted].com	2020-04-05 17:44	3 ko

Figure C-10: Elastalert Emails Notifications

C-IV-9 Elastalert Kibana Plugin

The developer of Yalp has recently creates Kibana Plugin for Elastalert. This plugin provides a way to create, test and edit ElastAlert rules within Kibana. This project needs to be developed and It needs to run ElastAlert as server to make use of this plugin for this reasons it has not been installed and as developers, we prefer command line. The dashboard is shown in the figure below:

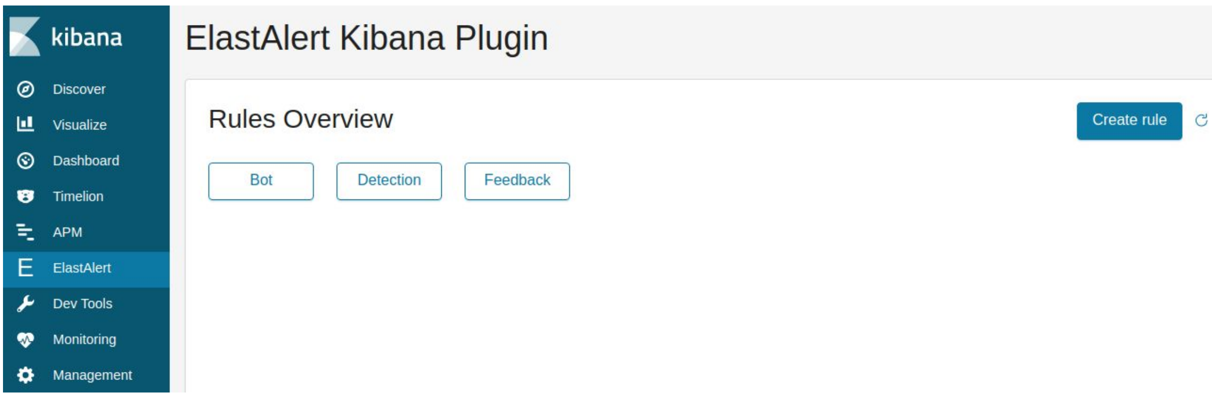


Figure C-11: Elastalert Kibana Plugin Dashboard

C-IV-10 Summary

Elastalert is a performance framework for alerting. Some issues has been faced in the installation and utilisation due to the lack of documentation but we received all the rest of alerts on time, and that what really matters.

C-V Zabbix

C-V-1 Definition

Zabbix is a free distributed open-source all-in-one network monitoring solution. It provides many ways to monitor different aspects of the IT infrastructure. It can be characterised as a semi-distributed monitoring system with centralised management. While many installations have a single central database, it is possible to use distributed monitoring with nodes and proxies. Most installations are using Zabbix agents [24].

C-V-2 Zabbix Features and Architecture

If we look at a simplified network from the Zabbix perspective, placing a Zabbix server at the centre, the communication of the various monitoring aspects matters. The following figure shows us the architecture of Zabbix and how the components are connected. Zabbix has [24]:

- A centralised and easy to use web interface.
- A server that runs on most UNIX operating systems.
- Native agents for most UNIX-like operating systems and Microsoft Windows.
- The ability to directly monitor SNMP (SNMPv1, SNMPv2c, SNMPv3) and IPMI devices.
- The ability to directly monitor Java applications using JMX.
- The ability to directly monitor vCenter or vSphere instances using the VMware API.
- Built-in graphing and other visualisation capabilities.
- Low-Level Discovery (LLD) and the ability to generate items, graphs, and triggers (among others) in an automated way.

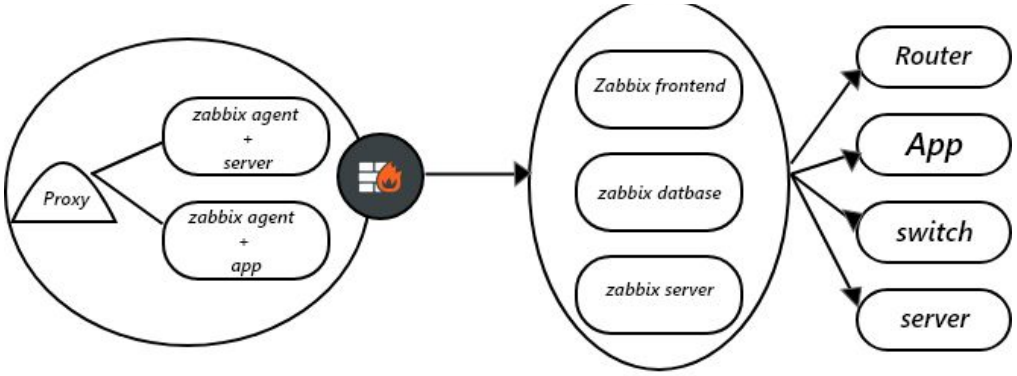


Figure C-12: Zabbix Architecture

C-V-2-1 Zabbix Server

It monitors directly multiple devices, but a remote location is separated by a firewall, so it is easier to gather data through a Zabbix proxy.

C-V-2-2 Zabbix Proxy and Zabbix Agents

It is a software that is deployed on the target machine which needs to be monitored. It is responsible for reporting the gathered data from the target to Zabbix Server.

C-V-2-3 Zabbix Database

Our central object supports several backends like Zabbix server which is written with C and the web frontend in PHP. When running each component on a separate machine, both the Zabbix server and the Zabbix web frontend need access to the Zabbix database. The Zabbix web frontend needs access to the Zabbix server to display the server status and for some additional functionality [24].

C-V-3 Gathering Data Concept

In Zabbix, there is the concept of gathering data and this is made by items that can collect a particular pieces of data like logs, metrics, system performance, downtime.., they are the main entities of Zabbix because without data nothing could be monitored and each item should be created in a host.

C-V-4 Grafana Dashboard

The user interface of Zabbix is so boring . That is why, there is a new plugin to be installed with Zabbix which is Grafana plugin allowing to visualize monitoring data from Zabbix and create modern dashboards for analysing metrics and real-time monitoring. The picture of Grafana is illustrated in **Appendix D**. It has not been installed because we are just in the testing period.

C-V-5 Notifications & Automatic Actions Process

Zabbix provides a complete workflow: sending notifications, allowing acknowledgement of information received, escalation of information to other people, and the ability to take actions. The following diagram presents the process of notification in Zabbix .

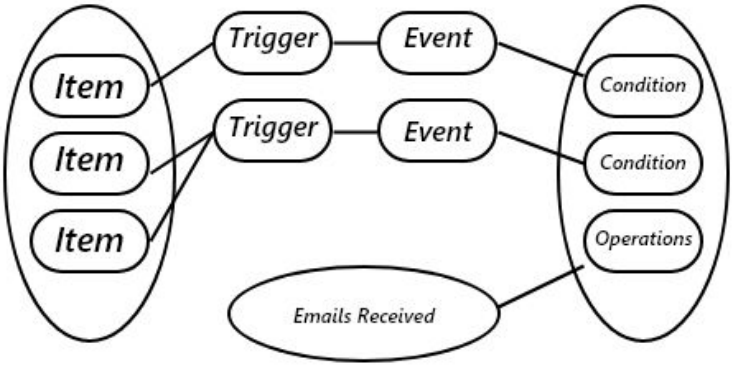


Figure C-13: The Process of Notification in Zabbix Diagram

Notifications could be scripted and the notification content is completely customizable depending on the context so the main point is that each contact can be notified for specific levels using specific media which are: e-mail, SMS, Jabber, Ez texting or using a custom alert script, at specified days and times.

C-V-5-1 Host

Is a logical entity that groups items. Definition of what a host is can be freely adapted to a specific environment and situation. In Zabbix, the host could be a network switch, a physical server, a virtual machine, or a website.

C-V-5-2 Items

Zabbix uses the principle of an active or passive item to retrieve particular values, description is available in **Appendix E** on a supervised host. These retrieved values could be for example SNMP or IPMI queries, return codes, various program results, or data on a system. It is from an item, that Zabbix would be able to trigger events and then alerts through the other elements such as the trigger and the action. A list of the main parameters (or attributes) of an item is presented in **Appendix F**.

C-V-5-3 Triggers

Triggers have the role of generating (triggering) events in reaction to a certain value or data raised by an item. Events in Zabbix are not alerts strictly speaking. In fact, they are elements indicating the status of an item depending on the chosen conditions in the form of a statue type *WARNING, UNKNOWN* or *OK*. For alerts, Zabbix uses action elements to generate notifications or any other operation from the status of a trigger (event).

These two notions can be summarised as follows:

- For triggers (from an item) : trigger = event = status
- For actions (from a trigger) : action = alert = notification

List of the main parameters defining a trigger is presented in **Appendix G.**

Change of trigger status is the most frequent and most important source of events. There are other types:

- **Internal events**: when trigger go to an unknown state or when items become unsupported.
- **Discovery events**: when hosts or services are detected.
- **Auto-registration events**: when active agents are auto-registered by the server.

C-V-5-4 Action

If we want any result from these events we need to create action. As it is presented in the figure, action has two main components: ***condition*** that mainly defines which data event should match, and ***operation*** is the result of this condition that could be any type of notification previously mentioned. The main parameters defining action are presented in **Appendix H**.

C-V-6 Templates

In monitoring, we usually monitor the same thing (CPU, FS..). So entities (items, triggers, graphs, applications, screens, low-level discovery rules, web scenarios) would be the same. It helps to accelerate the deployment of monitoring tasks on a host, also it makes the application of mass changes to monitoring tasks easier, simply by linking Templates directly to individual hosts and it creates automatically all entities that belong to this Template. There is another concept called "Nested templates" which implies the creation of a template that encompasses one or more other templates. Thus, if we create host from this template, all entities would be inherited [26].

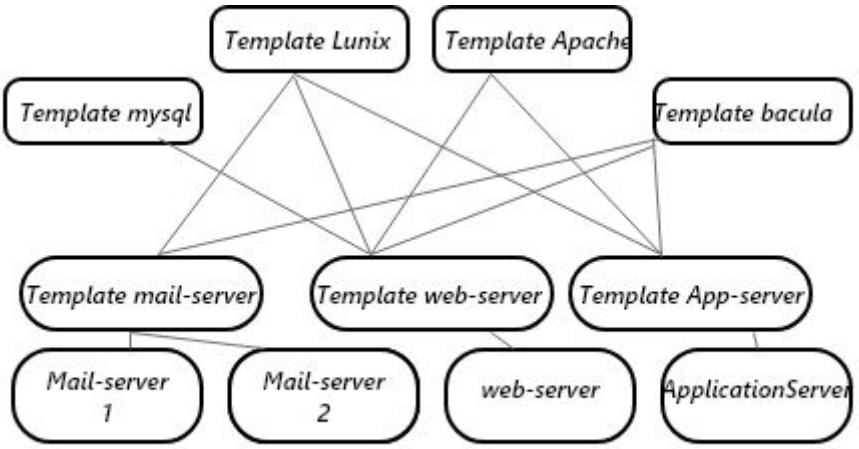


Figure C-14: Concept of Zabbix Templates

C-V-7 Zabbix Installation

We install Zabbix 4.0 in ubuntu 18.04 LTS system 64 bit

C-V-7-1 Install Some Prerequisites

Apache: 1.3.12 or later

MySQL: 5.0.3 or 8.0.x / MariaDB: Latest

PHP: 5.4.0 or later

C-V-7-2 Installation

C-V-7-2-a Configure Zabbix Repository: Because Zabbix is not included in Ubuntu repositories

- `wget https://repo.zabbix.com/zabbix/4.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_4.0-3+bionic_all.deb`
- `sudo dpkg -i zabbix-release_4.0-3+bionic_all.deb`

C-V-7-2-b Install Zabbix Server,Agent & Frontend :

- `sudo apt update`
- `sudo apt -y install zabbix-server-mysql zabbix-frontend-php zabbix-agent`

C-V-7-2-c Timezone: We Edited the Zabbix configuration file to update it with our timezone.

- `sudo nano /etc/zabbix/apache.conf`
- `... php_value date.timezone Europe/Paris ...`

C-V-7-2-d Create Database: Login to the server and create a database for our Zabbix installation

- `sudo mysql -u root -p`
- `MySql create database zabbixdb character set utf8 collate utf8_bin;`
- `MySql grant all privileges on zabbix.* to zabbixuser@localhost identified by 'password';`
- `MySql quit;`

Once we create a database for Zabbix installation, we import the initial schema and data into it.

- `cd /usr/share/doc/zabbix-server-mysql`
- `zcat create.sql.gz | mysql -u zabbixuser -p zabbix`

C-V-7-2-e Update Zabbix Configuration: Now edit the zabbix_server.conf file to set the database details created in step d .

- `sudo nano /etc/zabbix/zabbix_server.conf`
- `DBHost=localhost`
- `DBName=zabbixbd`
- `DBUser=zabbixuser`
- `DBPassword=password`

now we restart all the services :

- `sudo systemctl restart zabbix-server zabbix-agent apache2`

C-V-7-2-f Setup Zabbix via Web Installer :

In a web browser, we go to https:\\server_IP/zabbix/ , in our case server ip is localhost.

We keep pressing next and filling the required field, in the end, we will obtain Zabbix frontend successfully installed. Now we have the login page by default username is *Admin* and password is *zabbix*. Here is the main dashboard of Zabbix:

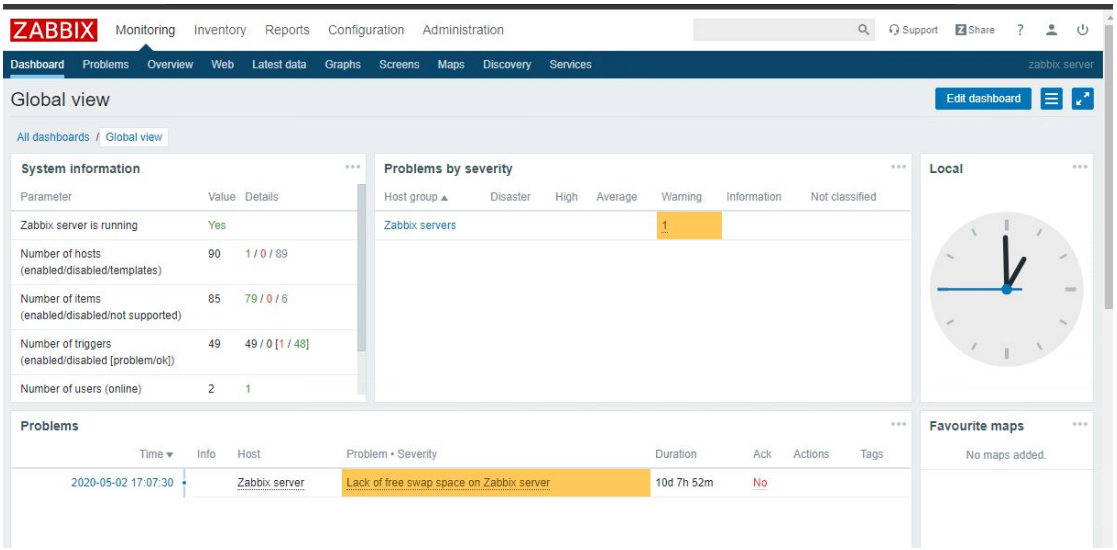


Figure C-15: Zabbix Dashboard

As we are basically working with Zabbix frontend , here is an analysis of its main menu [24] :

- **Monitoring:** This category contains most of the monitoring-related pages including data, problems, and graphs .
- **Inventory:** It enables to see the gathered data.
- **Reports:** Enables to see reports and visualise items.

- **Configuration:** Setting up everything related to the monitoring of systems, parameters, notification sending.
- **Administration:** This section allows to set up more of the Zabbix internals, including authentication methods, users, permissions, and global Zabbix configuration.

C-V-7 Installation and Configuration of Zabbix Agent

Zabbix agent collects various types of data like CPU, logs, filesystem and many others from the machine and sends them to the Zabbix Server. We must install the Zabbix agent in all the servers that we want to monitor MySQL, Pfsense, HaProxy… However, the remote machine was not available so we decided to monitor :

- Ubuntu system which is located in VMware.
- Mysql server which is located in the windows operating system.
- Windows system

We start by installing Zabbix agent in Windows as it has been already installed in Ubuntu. For this, we use MSI package or archive package as we did:

We download windows agent from:

- `https://www.zabbix.com/download_agents?version=4.0+LTS&release=4.0.21&os=Windows&os_version=Any&hardware=amd64&encryption=OpenSSL&packaging=Archive`

We create directory C:\zabbix\ , copy after unzipping windows agent to this new folder.

We edit zabbix_agentd.conf:

- `server = IP of zabbix server`
- `Hostname = FQDN of windows system`

Now we install the Zabbix agent by running this command :

- `C:\zabbix\zabbix_agent.exe -c C:\zabbix\zabbix_agentd.conf --install`

Next, we need to start the Zabbix agent, so there are two ways:

- `From cmd by running C:\zabbix\zabbix_agent.exe --start`
- `Open services in windows and click right on Zabbix Agent and click on start.`

Since all windows have an active firewall, we must open a port for Zabbix agent:

- `we go to Control manual -> System and security -> Windows firewall and click on Allow app through windows firewall and we add zabbix_agentd.exe to the list`

C-V-8 Monitoring IT Infrastructure With Zabbix

First, we have to create hosts we just need to create one host for each machine(server, router, etc) we want to monitor for our case we create one for Windows machine and the other for Ubuntu VMware but since it's where Zabbix server is installed so it's created by default with the name of Zabbix server

C-V-8-1 Creating hosts

There are multiple ways to create hosts, and here is the basic way:

In configuration section -> hots-> create host

*Clone or fully clone existing entity***:** Cloning is creating all the existing templates in cloning host, and full clone is to create everything existing in cloned host to the new one. Note that clone/full clone will copy template by their default configuration [25].

C-V-8-1-a Creating Windows Host

Now we create a Windows host and link to it Template OS Windows.

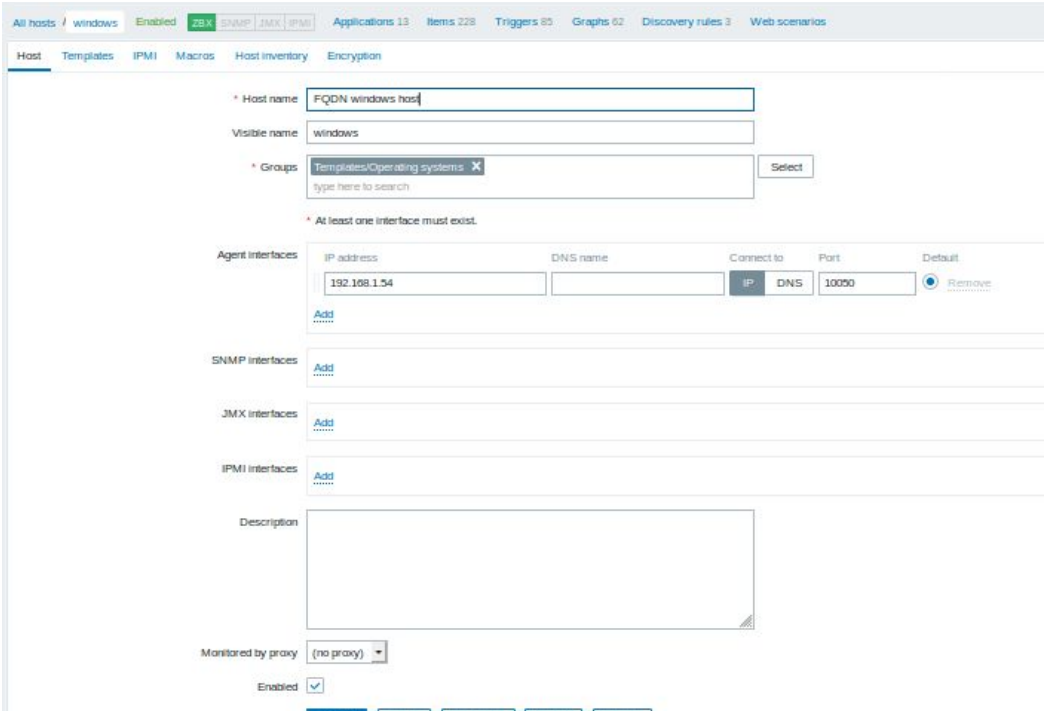


Figure C-16: Host Configuration

Notes:

- The host's name must be unique and in this case, it is the FQDN of Windows. Also, we have to marquee the checkbox enabled to make it active to be monitored.

In tab Template, we selected Template OS Windows to monitor windows host, it will create automatically multiple items like items collect CPU, memory, disk space, available and Discovery rules which are rules that detect automatically some changes in network or in the computer and create entities based on that.

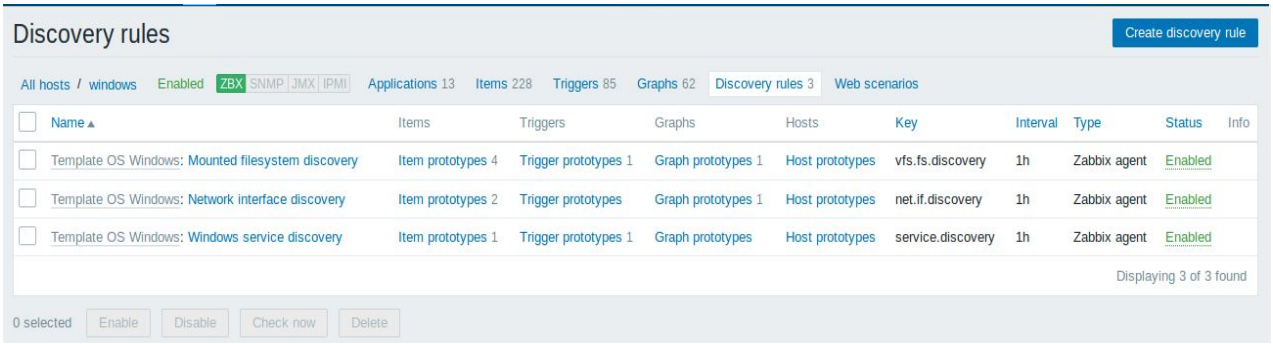


Figure C-17: Windows Discovery Rules

- If we want to secure our communication we go to tab Encryption and add it, we can use Transport Layer Security or pre-shared key-based or others, note that we have to link specific libraries to support this, in our case we didn't use it.

C-V-8-1-b Creating MySQL Host

Since MySQL is installed in windows and as we already had host there. Just in windows host, we link another template DB MySQL, it automatically creates multiple items such as statue item, version and uptime. But we have to configure certain things in Windows first, First, create a user for Zabbix in MySQL and give him all privileges:

- `CREATE USER 'zabbix'@'localhost' IDENTIFIED WITH mysql_native_password;`
- `GRANT PROCESS ON *.* TO 'zabbix'@'localhost' IDENTIFIED BY 'password'`

Next we created a configuration file C:\zabbix\zabbix_db_authontification.conf with this login information:

- `[client]`
- `user = zabbix`
- `password = password`
- `host = localhost`

Now we install some Unix commands (grep, cut, wc) that we download from the web, we add the path of this binary and MySQL binary to environment variable "PATH" under both "User variables" and "System variables".

Now we create another file Zabbix_agentd.userparams.conf to collect data with other details after edit zabbix_agentd.conf by including this file:

- `Include=C:\path\to\zabbix_agentd.userparams.conf`

Restart Zabbix Agent, and add Template DB MySQL to Windows host.

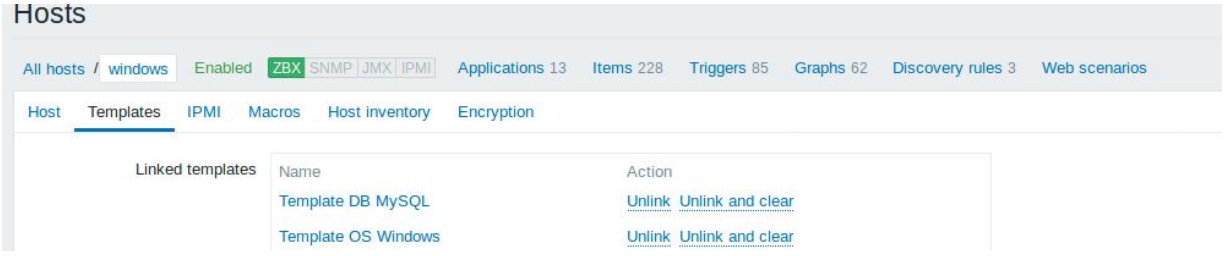


Figure C-18: Windows Host Templates

C-V-8-2 Creating Items

Once hosts have been created, it is time to look at the concept of data gathering by adding some Zabbix-items.To create an item we go to :

- Configuration->Hosts
- Click on item in the row of the chosen host
- Click on create item in the upper right corner of the screen

So Templates created many items, they check our system conditions. The item type that was used is "*Zabbix agent*". There is a lot of other item types like"*SNMP agent*", It is used on devices such

as switch/router/printer in which it would be impractical to set an operating system and install Zabbix agent.

C-V-8-2-a Metrics Item

The most important attribute is **item key** which is the technical name that describes our data, like "system.cpu.util[,system]". It collects data and gives a percentage to system CPU utilisation. There are two types of item key: flexible which is the one that accepts an argument, so at the end of the key we add [parameters separated with comma] like system.cpu.util[0,user,avrg5] the percentage of using CPU 0 by the user for 5 min average. The second type is non-flexible item key which does not accept an argument. Items Windows would be created in host Windows whereas Ubuntu in host Zabbix server and because we want this item in percentage we use the following type of information "Numeric(float)" and unit "%".

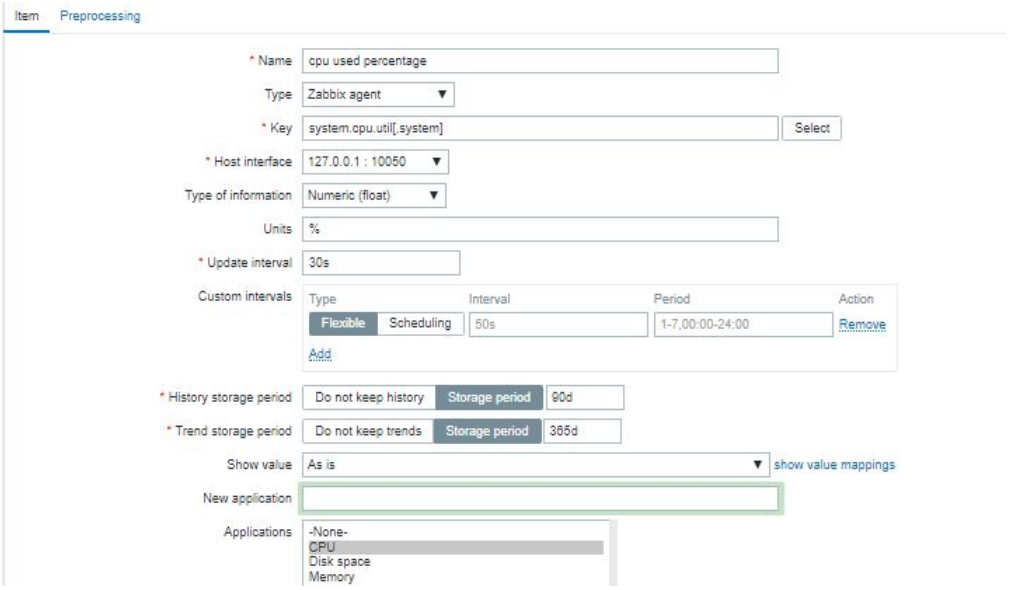


Figure C-19: Windows CPU Used Percentage Item

For the other items here are the item keys we used:

- In Zabbix server host for Ubuntu :
  - Disk space percentage: vfs.fs.size[/,pfree]
- In Windows host for Windows :
  - Disk space pourcentage: vfs.fs.size[C:,pfree]
- This was created in the both host:
  - CPU used pourcentage : system.cpu.util[,user,avg(5)]
  - Memory available pourcentage:VM.memory.size[pavailable]

C-V-8-2-b MySQL Item

This two we created them in Windows host for gathering data about MySQL server status and the avg number of query per second

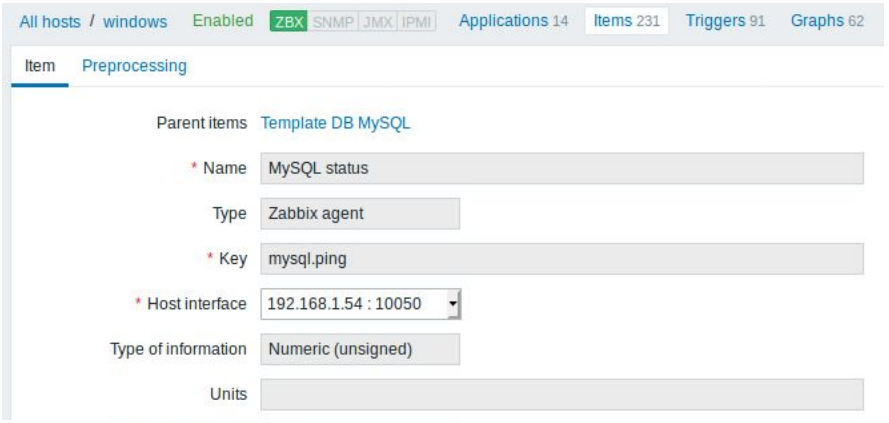


Figure C-20:Item MySQL Status

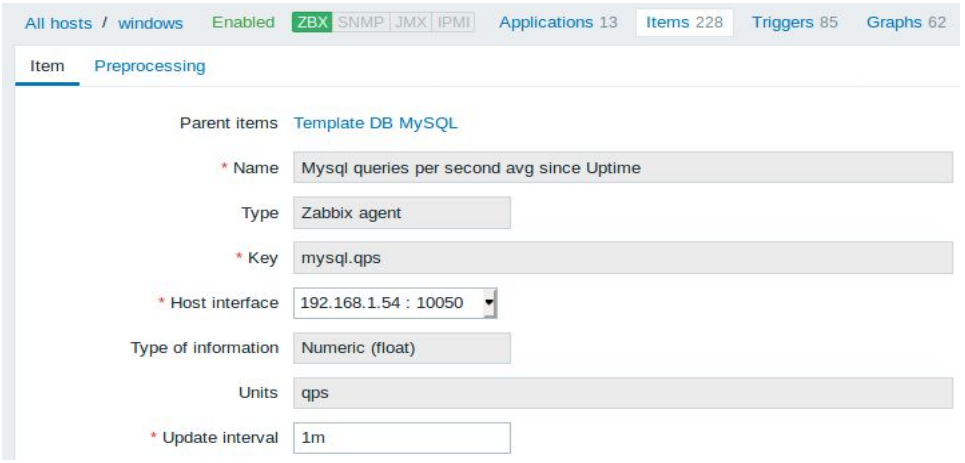


Figure C-21: Item Number of MySQL Queries

Good practices: we created these two items in MySQL template so if we wanna monitoring another Mysql server it will be easy and automatically created.

C-V-8-2-c Log files Item

For Log files we used zabbix Agent active mode,we edited both Zabbix Agent configuration files in windows and Ubuntu by adding: ServerActive= (zabbix_server_IP).

- **Windows log file**

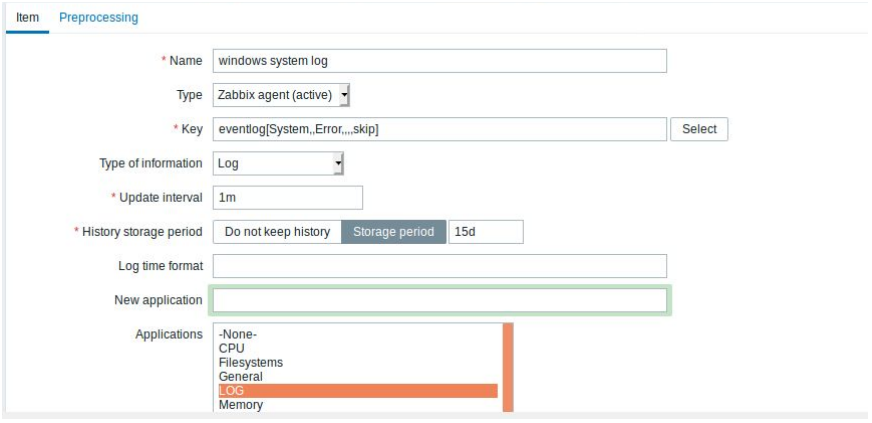


Figure C-22: Item Windows Log File Configuration

- **Ubuntu Log File**

First, we check log file permissions using terminal command -la, if Zabbix does not have the permission to read that file, we permit to it do so.

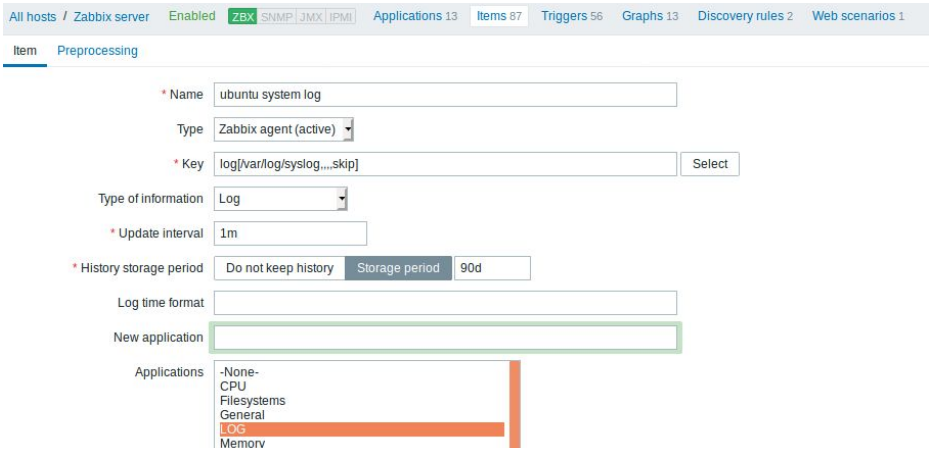


Figure C-23: Item Ubuntu Log File Configuration

C-V-8-3 Visualizing Data

Now we can see our data if these items work properly. In the menu, we go to monitoring->latest data, we choose our Hello host. There are two ways to visualise our data: by graph and map.

-Graphs: Graphs allow to grasp the data flow at a glance, correlate problems, discover when something started or make a presentation of when something might turn into a problem [27]. Zabbix provides users with:

- ➢ **simple graphs**: They are provided for the visualisation of data gathered by items and No configuration effort is required on the user part to view simple graphs, in Monitoring section → Latest data and click on the Graph link.
- ➢ **complex customised graphs:** If we want to compare the data of several items like incoming outgoing traffic, we can create our custom graphs in Configuration->hosts click on graph in the row next to the desired host, in the Graphs screen we click on create graph.
- ➢ There is a third quick way to visualise our data by **ad-hoc graphs** which allows us with no configuration to quickly see a comparison graphs by checking the box of the desired data in Latest data and click in apply [27].

-Network Maps: When we want to look to our network and an overview of our infrastructure we should create Zabbix maps of networks and of anything the company need to see.

-Screens: Screens in Zabbix is a data visualisation method where we will englobe various data from various sources in one screen for a quick overview, it's mainly a table we choose what

elements ‘simple graph, custom graphs, maps, plain text information...etc) to display in each cell. we chose the first way which is graphs because that what we need for now :

C-V-8-3-a CPU, Memory, Disk space percentage complex graph: In the figure below we created a graph that bring together CPU, Memory,Disk space percentage.For the parameters: *name* must be unique and Graph type there are several *type* Normal means values displayed as lines, Stacked means filled areas displayed, Pie graph and Exploded which is portions displayed as “cut out” of the pie. *Items* parameter here we can add all the item we want to see, note that we can add items of one or several hosts or single template.

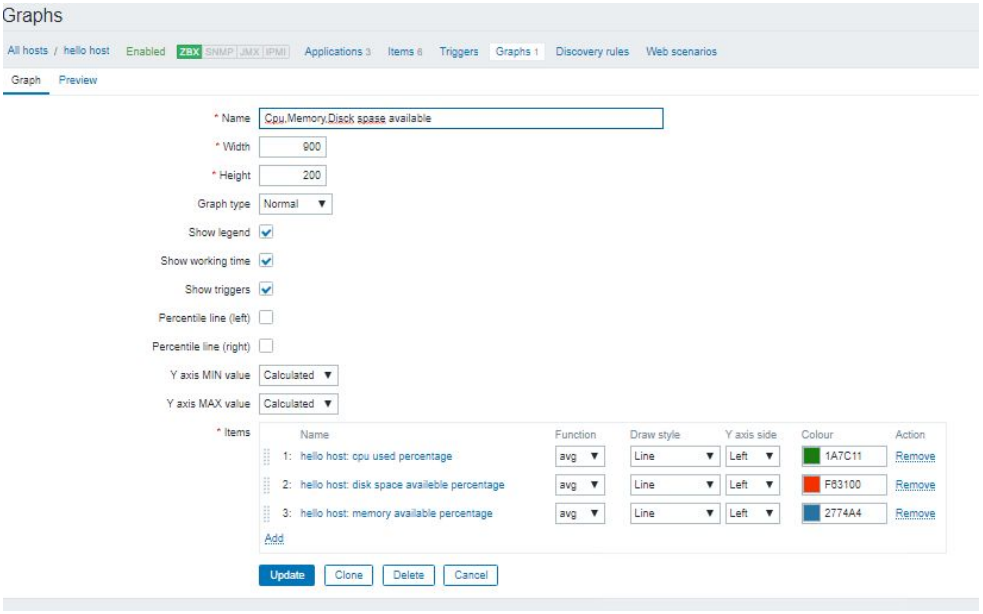


Figure C-24: Graphs Configuration

C-V-8-4 Web Monitoring

In Zabbix, we can monitor a website easily just by creating a web scenario. This latter consists of multiple HTTP request “steps” and it automatically creates items:

- Download speed for scenario.
- Failed step of scenario.
- Last error message of scenario.

And the same one for each step, to create web scenario :

- Configuration ->Hosts.
- Click on Web scenarios
- Click on create scenarios in the upper right corner of the screen

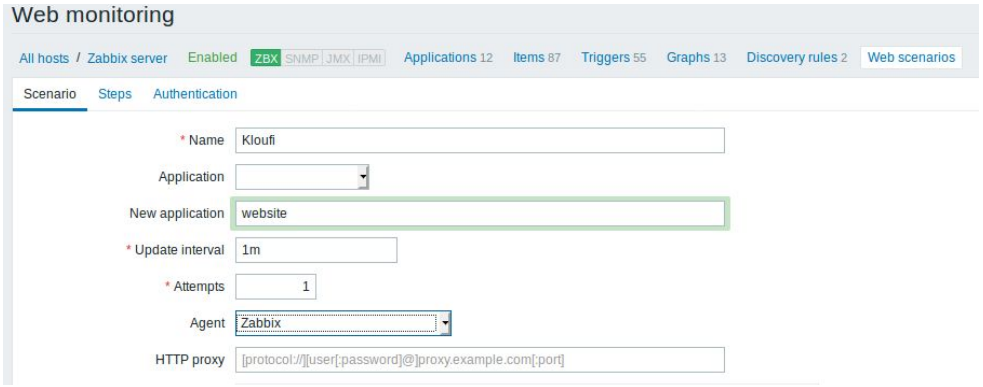


Figure C-25: Web Monitoring Configurations

In Steps tab which is next to scenario tab, we add all the URL we want to monitor and require statue which is 200.

C-V-8-5 Creating Triggers

We gathered data, now we define our conditions based on our objectives :

- Configuration->Hosts
- Click on trigger in the row of the chosen host
- Click on create trigger in the upper right corner of the screen

Like items, thanks to the template most of the required triggers are already created.

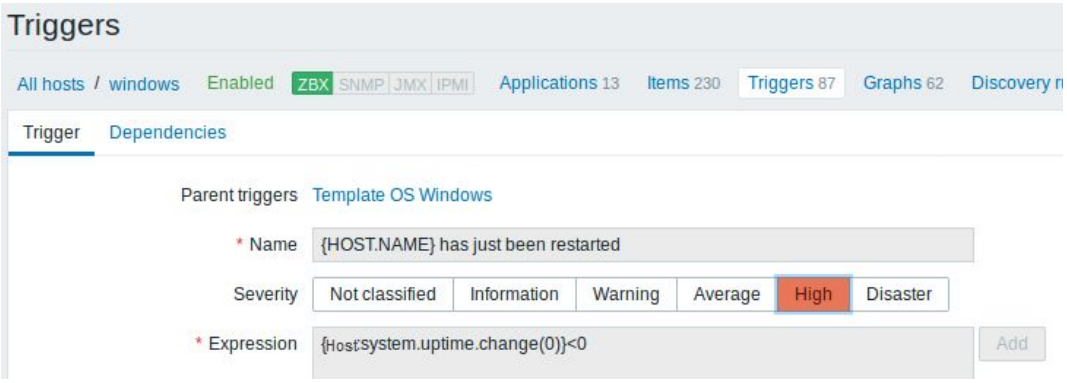


Figure C-26: Trigger Windows Status

This trigger was created by Windows template -as it is shown in parent triggers-, it expression is true once we refresh Zabbix agent in Windows.

Note: The most important parameter in trigger is *expression* to make tests, the basic form is `{<host name>:<item key>.<function>(<parameter>)}<operator><constant>`, the first two parts are clear for item key, we must have an item that use it ,and for function there are multiple ones for period avg(),min(),max() and others for comparing between data last(),diff(),change() and others. These factions are used to define conditions on data, likely Zabbix made interface for this task so we just click on add in expression clone and choose the right parameters.

Figure C-27: Trigger Condition

The configuration of triggers is the same, we need just to change the Expression based on our need , so here are the expression we used :

- Availability:
  - MySQL: {**mysql.ping.last(0)}=0
  - Kloufi website: `{Zabbix server.web.test.error[Kloufi.strln(0)}>0 and {Zabbix server:web.test.fail{Kloufi}.last()}>0;` Here we created failed connection trigger with a useful problem description in the trigger name.
- Windows and Ubuntu metrics:

  Here we replace ** by the host name, for ubuntu triggers we replace it with zabbix server and for windows with windows FQDN:
  - CPU utilisation: {**. system.cpu.util[, system,avg5].last(0)}>=80
  - Memory: {**. vm.memory.size[pavailable].avg(5)}<20
  - Windows Disk space: {**.vfs.fs.size[C:,pfree].last(0)}<20
  - Ubuntu Disk space: {**.vfs.fs.size[/:,pfree].last(0)}<20
- Log File :
  - Ubuntu System Log Error:{**:log[/var/log/syslog,,,skip].iregexp(error)}=1
  - Windows System Log Error:{**:eventlog[System,"Error",,,Skip].change()}=1

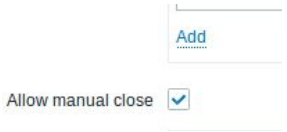
C-V-8-6 Events Configuration

There are several types of events generated automatically and they are automatically closed but sometimes we need to know the problem in order to be aware.

C-V-8-6-a Manually Closing Trigger

Trigger events are closed automatically when trigger status goes from "Problem" to "OK", but if we want to close them manually, we change the configuration of triggers :

- In trigger configuration we allow manual close in trigger windows disk space



- And when the problem is solved we go to Monitoring -> Problems and choose our event and click No on Ack row.

- So we need to check the box *close problem* to have the option of closing just this event or all events generated from the same trigger.

This event is not closed directly but through task manager, we use this for complicated problems that we need to know their reason.

C-V-8-6-b Event Correlation

Event correlation allows to correlate separate problems reported by one trigger. In the normal way, events of one trigger are all closed if one of the trigger go to "OK" statue . Sometimes, like in log file trigger, there are multiple applications or services so we need to close each event separately to understand the reason of the problem. In this trigger, we use *the tag* so that events of same tag will close together. In the tab below, we decided to create trigger for Ubuntu services, in problem event generation mode we choose "Multiple" and in OK event closes "All problems if tag values match".

Figure C-28:Trigger Services Configuration

C-V-8-7 Notifications

We created items to collect our data and triggers to make a condition on them. Now we want to receive notification to see if something goes wrong, notifications are sent to the destination we want in the appropriate time.

C-V-8-7-a Create Media Type

In order to send a notification in Zabbix, we create a media and action. First, we create media which is for our case is Gmail. We have to enable our Gmail account to receive connections from external programs, then we install SMTP in Ubuntu system by this command

- `sudo apt-get install ssmtp`

Edit the ssmtp.conf file:

- `root=*****@gmail.com`
- `mailhub=smtp.gmail.com:465`
- `FromLineOverride=YES`
- `AuthUser=Gmail address`
- `AuthPass=Gmail password`
- `UseTLS=YES`

Now, we go to Administration→Media types and click in create media type, we choose Email type and fill our SMTP parameters. Then , we need to add media to an Zabbix user.

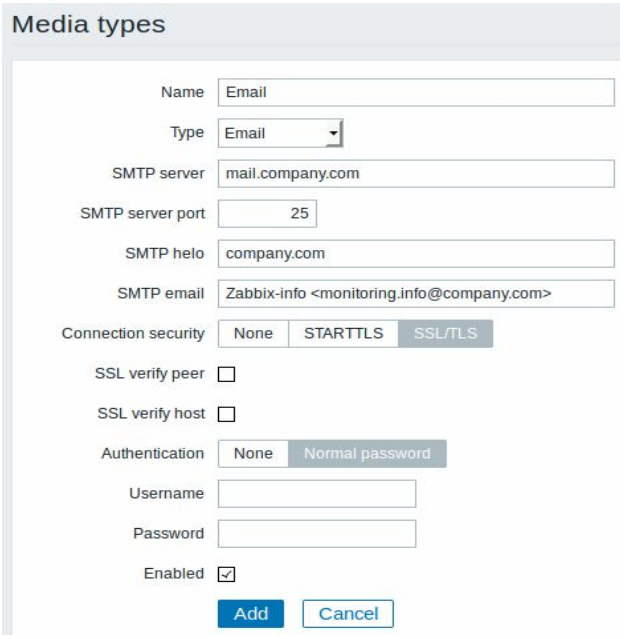


Figure C-29: Media Types Configuration

C-V-8-7-b Creating Actions

The last step is action. Without action there is no notification in Zabbix. Since actions depend on event status, it is possible to create one action for multiple triggers and specify triggers name in the report. First, we create action in configuration -> Actions

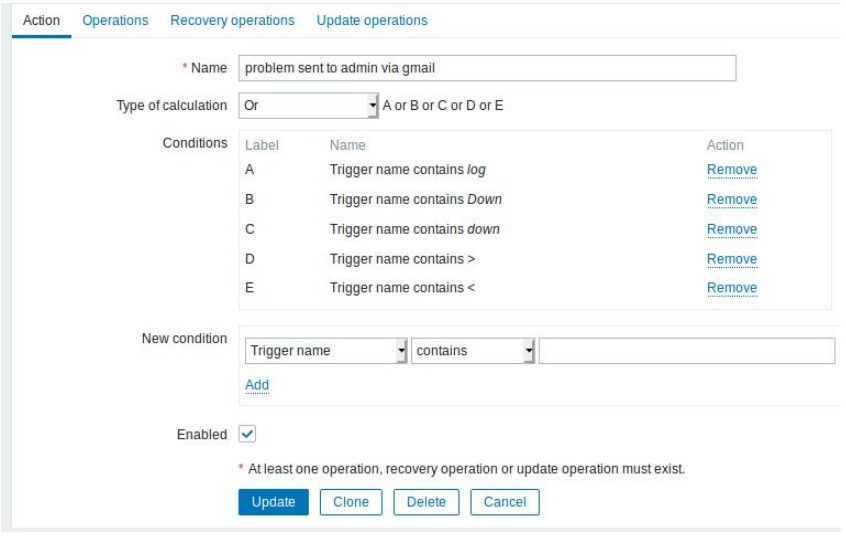


Figure C-30: Action Configuration

After that, in operation tab we add a message which is the core of email, Then we add media in operation by specifying the name of gmail account.

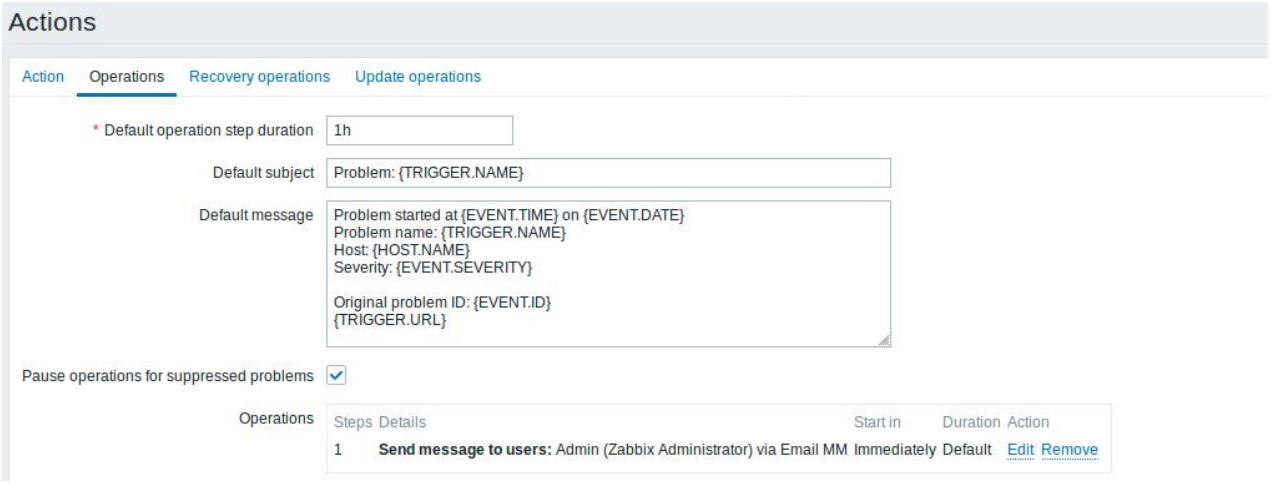


Figure C-31: Operations Configurations

In Recovery tab we add the recovery operation, it would be realized if the problem is solved.

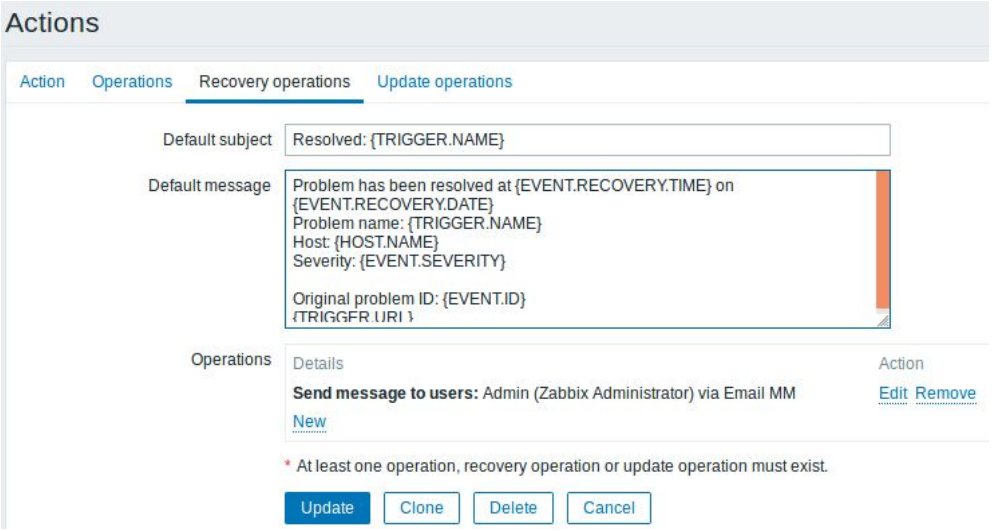


Figure C-32: Recovery Operations Configurations

We check the statue of our messages in monitoring->Problems, in the action column we can see if any email was sent and why. Here are the problem and resolved emails

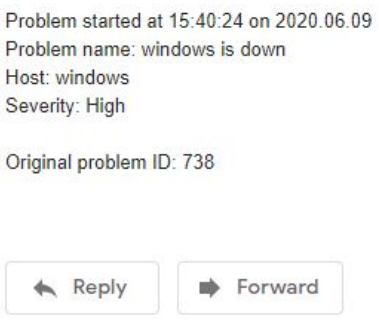


Figure C-33: Email Problem

Figure C-34: Email Resolved

C-V-9 Summary

Zabbix agent is a very powerful open-source software that allow to collect metric from multiple devices, systems, applications. It may also detect problems and troubleshooting in real-time. However, this does not deny the fact that it is somehow bothering and it is hard to configure all his entities. We also made a research about templates and the result was that all the templates for all the company servers, applications(MySQL, Pfsense, Nginx...) are predefined .

VI- Conclusion

The first part of this chapter shows a general idea about log management. Then, we moved to the suite Elk Stack and its main components with Beats. After that, we have installed and tested the alternative of X-pack Elastalert and the second solution Zabbix according to the company needs.Both Zabbix and Elk Stack with Elastalert are great tools that have been used in big companies. In the following chapter, we will compare these tools based on all this work and choose which solution will suit the company better.

Chapter D: Evaluation & Decision

D-I Introduction

In this last chapter, we will analyse the results of each tool and extract the advantages and disadvantages of each tool, to compare between the two proposed solutions after based on our objectives , to end with deciding what is the best solution for the company based on its needs.

D-II Evaluate Elk Stack

The company has already Elastic stack installed as we know, we need now to evaluate it

D-II-1 Advantages

- Open Source.
- Fast and stable.
- Secure and reliable suite.
- Scalable, Flexible and dynamic.
- Modern dashboard and has performance features. In addition, the ability to create personal dashboards and dev tools.
- Features such as Machine Learning, Security, and Reporting.
- A large-scale and real-time, analysis and research.
- Many modules of programs ready to use (Analyzer, full-text search...).
- Easy implementation due to Java, JSON and REST-API.
- High availability and Resilience (no loss of data).
- Optimization of the number of queries performed to retrieve data.
- Cache simplified: Using the server cache (Nginx).
- Flexibility of beats, the best thing about Beats is the Libbeat environment, which makes it easy to create custom beats for any type of data.
- The number of beats is growing rapidly and configuration is easy with the predefined modules.

D-II-2 Disadvantages

- Not all features are free.
- When using docker may have a temporary shutdown and a long restart after a rebooting.
- Elasticsearch works only with JSON and is therefore not directly compatible with CSV and XML data formats(a pre-phase of transformation to JSON is required).
- Elasticsearch requires more disk space since it creates indexes.
- The first problem face the IT admin when deploying in production is memory management issues, need to make a good strategy.
- The more data the cluster contains, the more difficult it will be to solve the problem, as it is sometimes necessary to re-index large amounts of data.
- Elasticsearch is more efficient by using structured data.
- Logstash-indexer may hang at times.

D-II-3 Criticize and Analyze the Results

When we worked with the Elastic Stack, we noticed that it is a performance system, all the needs of IT administrator are available in this pack: super-fast searching and analysis, visualisation of data and monitor it easily. Specially elasticsearch, its has a real-time analysis and research, Many decisions about how best to distribute the company data across indexes and partitions to optimize the maximum will depend on the details of the use cases, and it can sometimes be difficult to determine how best to apply the guidance available.

The other advantage which is the most attractive thing which gives the power to the suite Elk that it works with Beats where the diversity and the flexibility. Actually we didn't see any inconvenience just the need of more disk space when index grows and the problem of X-Pack features, we needed to purchase a subscription to keep using the most important function which is alerting, but we found a solution for this which is Elastalert, we will evaluate later.

D-III Evaluate Elastalert

Elastalert is a discovery for us, when we worked and tested it, we extracted its advantages and disadvantages.

D-III-1 Advantages

- It's completely free and Open Source.
- Elastalert uses the same query language used by Elasticsearch, Query DSL.
- Reliable cause it saves its state to elasticsearch and we noticed If Elasticsearch is unresponsive, ElastAlert waits until it recovers before continuing.
- Real-time alerting and get notified in the communication tool which IT admin prefers.
- Highly Modular and Easy to Set Up & Configure.
- Integration and work with all Elasticsearch versions.
- ElastAlert truly provides real-world alerting rule types like spikes, frequency, etc.
- The ability to create a personal rule type.
- The ability to write Enhancements which are modules that let us modify a match before an alert is sent.
- Ability to use any rule type to define alerts using any query on Elasticsearch.
- Supports many common notification endpoints - email, Jira and Slack for example.
- Prevent multiple alerts and Ability to add custom text to an alert.
- A recent new project which is Elastalert kibana plugin.
- The ability to run it as service and Help Automate Threat Hunting [28].

D-III-2 Disadvantages

- Lack of documentation and Lack of feedback from the community.
- Still need development and it needs time to understand its process and how it works in the background and need some query optimization
- In Recent changes as of Elastalert 0.2.0, it requires Python 3.6, Python 2 will no longer be supported.

- ElastAlert does not currently support stateful alerts or resolve events.
- Many issues in installation especially which concern python, and there are no solutions.

D-III-3 Criticize and Analyze the Results

The perfect alternative to X-Pack Alerting. ElastAlert has saved us actually. It started off with us watching the alerts closely and looping in team members. ElastAlert handles a lot by using A different kind of rule types catches any service that fails suddenly. What about the disk space on the database, ElastAlert sends Warning and Critical alerts when available disk space falls below the threshold. ElastAlert has helped us bring down our failure rate and has been a guardian for the company services.

All of that is good but the perfect point is real-time alerting, despite some installation issues and the lack of documentation, we needed a long time period to understand its process if someone doesn't understand command line it will be a complex framework.

D-IV Evaluate Zabbix

D-IV-1 Advantages

- All in one place and Strong documentation.
- Zabbix does monitor any kind of IT solution for us like servers, networks, services, virtual machines, databases or websites.
- The trigger-framework is pretty mighty and can act on a lot of metrics.
- Super performance tool for network monitoring.
- The ability to choose active and/or passive mode of the Zabbix agents .
- The automatic discovery of servers and network devices.
- Allows us to create our own graphics.
- The software architecture is divided into components (proxy, server, etc.) in order to facilitate the distributed monitoring.

D-IV-2 Disadvantages

- The associated scalability limits
- Server software for Linux, Solaris, HP-UX, AIX, Free BSD, Open BSD, OS X
- Busy Zabbix environment, and hard to comprehend.
- Zabbix has a steep learning curve and doesn't have a very intuitive and user-friendly interface.
- Setting up items, triggers, hosts, classes, etc is tedious, and not very obvious.
- Weakness in reporting, can not export reports in general formats.
- Weak in availability reports.
- Framework with minimal tools out of the (free) box.
- Templates needs tuning to meet specific situations.
- Lack of database partitioning and/or better housekeeping routines is a significant issue.
- The export of templates has improved substantially over past versions.

- Each Zabbix version has a life cycle so after a couple of years, some versions are unsupported so always we need to upgrade it.
- Templates still lack completeness (and entails lots of difficulties from linked templates, regex values, and the like that might not be carried along).

D-IV-3 Criticize And Analyze The Results

Zabbix is a super performance tool for network monitoring, we noticed many good points in like the templates which are ready to use, the automatic reports, but the big problem which was really complicated for us is the manual work, isn't user friendly and take a lot of time. Also when we have tried to monitor Ubuntu and Windows 10 OS using SNMP we have noticed that it takes more time to identify services that are running in the OS Sometimes it takes more than 1 hour, and this reduced the performance. In the other hand, the disk memory depends on the number of hosts

D-V Comparaison Based on our Objectifs

D-V-1 ELk + ELastalert

The first thing is the flexibility of beats, which we can specify the data we want, so install the agent, configure it with the activation of the modules depends on the server we have which are ready to use, then we receive logs in real-time they don't need time to identify the systems like Zabbix agent.

The performance of elasticsearch is not compared to Zabbix, there is no loss of data due to the many numbers of copy (shards) it's really performance in search and analysis and this why kloufi prefer it, designed for Big Data. we can check the data in the beautiful kibana and create a personal dashboard, Most Beats agents and their modules come with predefined Kibana dashboards.We agree that kibana needs disc space but it deserves. Elastalert the alternative of X-pack, actually it doesn't take a long time to create alerting rules like Zabbix, it's pretty easy and the best part is that we could create filter with fields we want and it is a strong point so monitoring CPU, memory usage, Availability was pretty simple due the predefined rule types. Also, we have the possibility to create our own and personal rule type depends on our needs. In addition , we received a real-time alerts, on the other hand we found a lack of documentation compared with the strong documentation of Zabbix.

D-V-2 Zabbix

Because there are already predefined templates for any operating systems, servers the company need to monitor, it was easy in first to monitor CPU, memory usage …, but we noticed that it's really bored when we started working, it's totally manual so it needs to create triggers, actions, hosts, every time and all of that take a long time, but there is another good point is the reporting despite it's need some evaluation because sometimes it doesn't work.

Zabbix database is from the big weaknesses, because Zabbix support just the structured data , in addition to that it needs its own database. In the other hand for a remote location, it needs to install another thing which is proxy, it can be a good/bad point in term of disc space. Zabbix agent takes more time to identify services that are running in the OS.

About alerting it's in real time, and we could see if an alert has been well arrived or not in Zabbix dashboard which is a strong point, in Elastalert we can see and verify it using the command line when there's match and when alert sent, it will inform us.

In Zabbix, disk memory depends on the number of hosts we used and parameters are being monitored. So it's not a good point if we look at the future of kloufi, in parallel elasticsearch support Big data. In addition, we got a survey result from stackshare[29] and here what the community said, we can see how just kibana has more followers and why they chose it vs Zabbix .

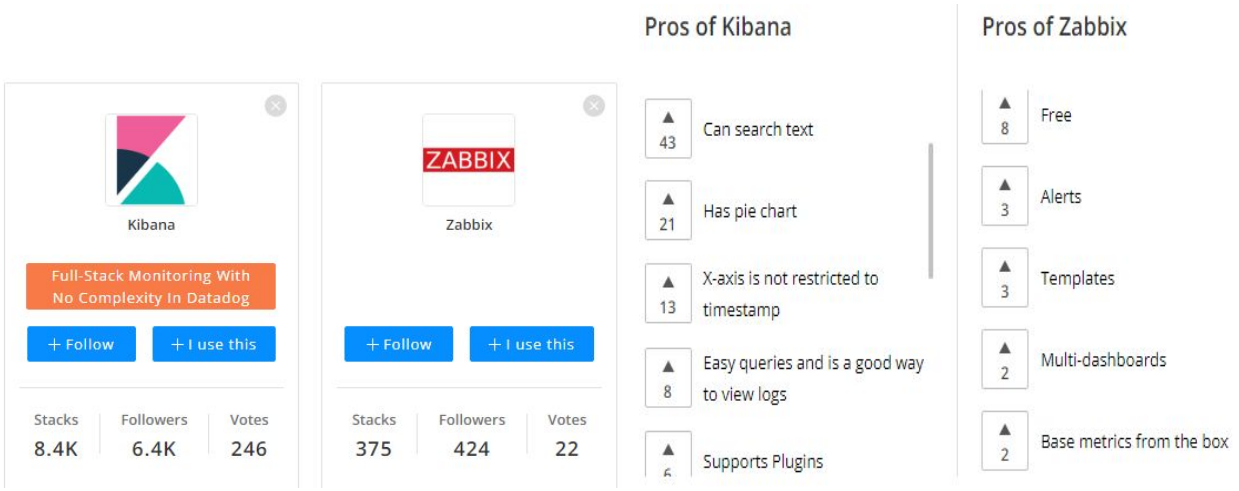


Figure D-1 : kibana vs Zabbix Survey

D-VI Best Solution for the Company

Alerting and analytics go together like cookies and milk. These two functionalities should be available in a monitoring solution, after the testing period of the two proposed solutions which are elk stack with a deep study of the alerting framework Elastalert and Zabbix

Based on the company needs and objectives, the results of each tool, the comparison made before and the pros/cons we extracted, without forgetting what the company has before as monitoring solution, we have decided to keep the suites Elastic stack and integrate with them the framework Elastalert to complete the functionality of performance monitoring tool.

The road map of the company is to have a big number of data, Zabbix cannot support and handle it with the same speed of Elk stack later despite it is the most popular and performance network monitoring tool and we agreed that we can keep it for this moments because it achieved what we want to monitor, but it will not be able to meet the needs of kloufi in the near future. In addition to that, the UI of Zabbix is not user friendly, it is no compared to kibana which is pretty cool and has performance features. We cannot forget how faster is elasticsearch with/without a big

number of data and how flexible are Beats so the ability to create custom beats for any type of data is available and will help kloufi company. All the servers we wanted to monitor ,their modules are available with their configuration of all metrics, log types… ELK has been the de-facto stack for logging and monitoring as the community said , ElastAlert builds on top of this and enables easy and flexible alerting. It has almost everything we need to build an alerting system on top of Elasticsearch.

Additionally, there are a few tools built around ElastAlert - that may remove the need for writing YAML configs also which are available in **Appendix I.**

So the combination of elk stack with elastalert give a good performance and strong monitoring system for the company.

D-VII Conclusion

This last chapter provided a complete analysis of the two suggested solutions, the advantages and disadvantages of the suite elastic , Elastalert and Zabbix. Then, we presented a comparison between Zabbix and elk stack with Elastalert. The chapter is finished by giving a decision of the best monitoring solution that suits the company.

General Conclusion

The last few years have shown the faster growing of data and the search for reliability and cost reduction in systems or software make the future of the IT world appear in the field of supervision, in which we talk about open source applications.

IT infrastructure distributed monitoring tool, offers clear visibility and simple to interpret. Indeed these tools allow us to check what is happening at any moment in the network and to react quickly in case of problems.

During this internship, we learned how to set up a monitoring solution by installing two tools which are Zabbix and Elastalert and well understanding and working with the most popular solution Elk stack. we learned how to compare and make a deep study to choose the best solution for the Kloufi company

This excellent experience led us to do a lot of research and to have a concrete vision on a very specific domain which is the IT infrastructure monitoring. In addition, this work allowed us to gain experience in Redhat system "CentOS7", Ubuntu and SuperPuTTY, through which we had the opportunity to apply our knowledge and to confront and apply the theoretical concept to the in the field of network and distributed system.

As perspectives, we propose the improvement of this work by:

- Automatic notification with a proposed solution for the problem.
- Use Machine Learning to grow the concept of monitoring.
- Use the hosted Elasticsearch Service on Elastic Cloud.

Bibliographical References

[1] Santis, D. M. (2019). L'infrastructure informatique, vecteur de l'essor de votre entreprise. Appvizer. Retrieved 14 may 2020 from https://www.appvizer.fr/magazine/services-informatiques/virtualisation/infrastructure-informatique

[2] Wilson, M. (2020). Best IT Infrastructure Monitoring Tools and Software. Retrieved 15 may 2020 from https://www.pcwdld.com/best-infrastructure-monitoring-tools-and-software

[3] Heroic staff. (2016).4 Consequences of Not Monitoring Your IT Infrastructure. Retrieved 20 may 2020 from https://blog.heroix.com/blog/not-monitoring-infrastructure

[4] Daubner, B. L. Effective computer infrastructure monitoring.

[5]Kufel, Łukasz. (2016). Tools for Distributed Systems Monitoring. Foundations of Computing and Decision Sciences. 41. 10.1515/fcds-2016-0014.

[6] ManageEngine (https://www.manageengine.com/fr/**).** Accessed : 28 may 2020

[7]Gilchrist, A. (2018). What is PRTG. Retrieved 15 jun 2020 from https://www.virtualhostedpbx.net/what-is-prtg/

[8] Nagios (https://www.nagios.org/) Accessed : 15 may 2020

[9] Monitoring-fr. (n.d). Zabbix. Retrieved 10 may 2020 from https://www.monitoring-fr.org/solutions/zabbix/

[10] AdRem Software, (2010). AdRem NetCrunch. *ZDNet.* Retrieved 12 may 2020 from https://www.zdnet.fr/telecharger/logiciel/adrem-netcrunch-39969764s.ht

[11] Solarwinds. (n.d). Network Performance Monitor (NPM). Retrieved 15 may 2020 from https://support.solarwinds.com/SuccessCenter/s/network-performance-monitor-npm.

[12] SpiceWorks. (n.d). Network Monitoring Solutions for IT Pros. Retrieved 16 may 2020 from https://www.spiceworks.com/it-articles/network-monitoring/.

[13]Berman, D. (2019). The Complete Guide to the ELK Stack. *Logz.* Retrieved 13 april 2020 from https://logz.io/learn/complete-guide-elk-stack/#intro .

[14] Kloufi (https://kloufi.com) Accessed : 09 june 2020

[15]Heidilohr. (n.d.). Remote Desktop clients. Retrieved 12 may 2020 from https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/clients/remote-desktop-clients

[16] CentOS (https://www.centos.org/) Accessed : 02 may 2020

[17] PuttyGen. (n.d). SuperPuTTY Guide – Download SuperPuTTY. Retrieved 02 may 2020 from https://www.puttygen.com/superputty .

[18] VirtualBox (https://www.virtualbox.org/) . Accessed: 02 may 2020

[19] Log file. (n.d.). In Wikipedia. Retrieved 29 may 2020 from https://en.m.wikipedia.org/wiki/Log_file

[20]Mourato, M. (2019). ELK Stack + Alerting: How to Monitor your Business and Infrastructure Data (Part One). *Medium.* Retrieved 05 may 2020 from https://medium.com/@manuelmourato25/elk-stack-alerting-how-to-monitor-zyour-business-and-infrastructure-data-part-one-a4a1c3427745.

[21] Dimitra Chatzichrysou, Infrastructure & Automation (IT-DB-IA). (August, 2019). Evaluate ElastAlert for IT-DB use cases.

[22] Sebastian, G. (n.d). Elasticsearch Monitoring based on X-Pack stats. Retrieved 01 june 2020, from https://grafana.com/grafana/dashboards/8642 .

[23]Modules. (n.d.). Retrieved 29 may 2020 from https://www.elastic.co/guide/en/beats/filebeat/current/filebeat-modules.html .

[24]Olups, R. (2010). Zabbix 1.8 network monitoring: monitor your networks hardware, servers, and web performance effectively and efficiently; Birmingham: Packt Publ.

[25]Zabbix Documentation 4.4,1 Hosts. (n.d.). Retrieved 22 May 2020, from https://www.zabbix.com/documentation/4.4/manual/config/hosts

[26]Zabbix Documentation 4.4,7 Templates. (n.d.). Retrieved 18 May 2020, from https://www.zabbix.com/documentation/4.4/manual/config/templates

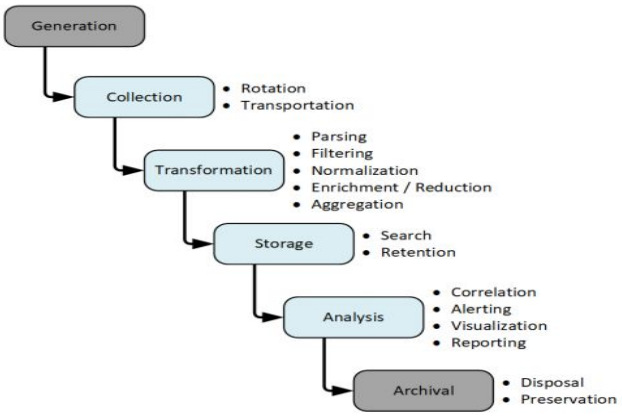
[27] Zabbix Documentation 4.4,6 Visualisation, 1 Graphs. (n.d.). Retrieved 24 May 2020, from https://www.zabbix.com/documentation/4.4/manual/config/visualisation/graphs

[28]Jp.(2017) . Using ElastAlert to Help Automate Threat Hunting. Retrieved 15 June 2020, from. https://jordanpotti.com/2017/12/22/using-elastalert-to-help-automate-threat-hunting/

[29] Stackshare (https://stackshare.io/stackups/kibana-vs-zabbix) . Accessed: 16 june 2020

Appendix A:

Coverage of Elastic Stack monitoring capabilities on log management model



Appendix B :

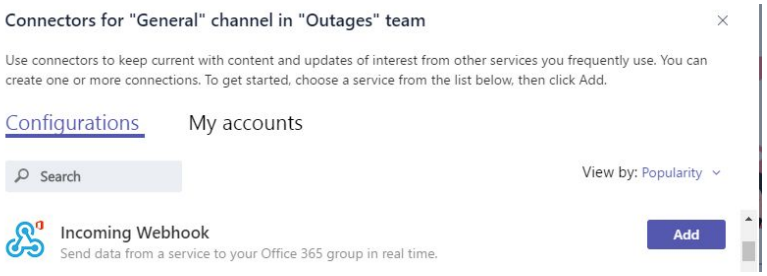
Three different types of documents in the writeback index are :

- **Elastalert_status:** is a log of the queries performed for a given rule and contains
- **Elastalert:** is a log of information about every alert triggered
- **Elastalert_error:** When an error occurs in ElastAlert, it is written to both Elasticsearch and to stderr

Appendix C:

Testing alerting with microsoft teams

In microsoft teams , by adding “Incoming Webhook” to the connectors in the channel we wanted like in the figure



we need to name the connector and assign an avatar, once done , we got a url, we saved it a copied it , because it is important to the next step.

The next step is to copy the following script

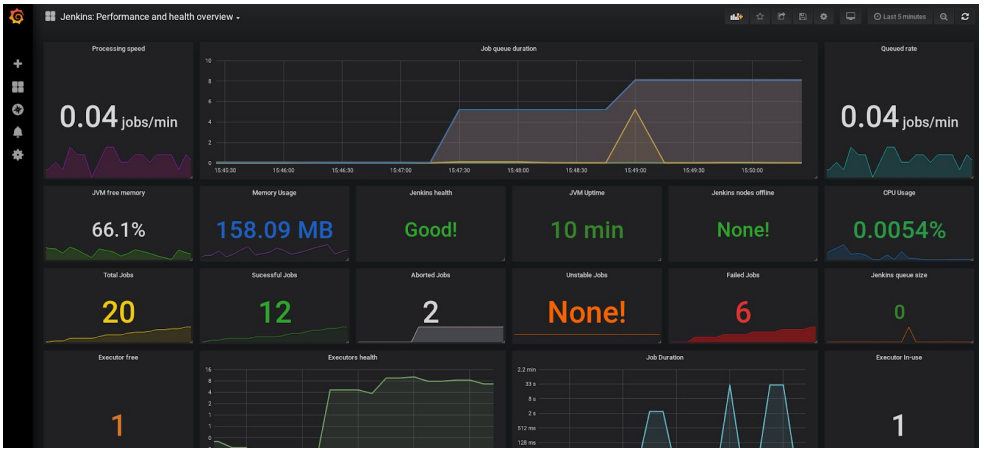
```
param (
  [string]$message = "This is a test of the alert system",
  [string]$title = "Test Message",
  [string]$status = "Cleared",
  [string]$monitor = "Testing - monitor",
  [string]$location = "Downtown",
  [string]$system = "labsys01"
)
$uri = 'https://outlook.office365.com/webhook/MISSING'; $body = ConvertTo-Json -Depth 4
@{
  title = $title
  text = ' '
  sections = @(@{ activityText = $message},
    @{facts = @(
        @{name = 'Status'
          value = $status
```

```
            },
            @{name = 'System Name'
              value = $system
            },
            @{ name = 'Monitor'
              value = $monitor
            },
            @{ name = 'Location'
                value = $location
            } ) }  ) }
  #Send the message to MS Teams
  Invoke-RestMethod -uri $uri -Method Post -body $body -ContentType 'application/json';
  Write-Output "INFO - Message has been sent.";
```

Then execute the script to see the default alert message , we created a simple rule with frequency rule type which is defined in github repository in the global documentation , just to test if it work or not, and it was successfully work , we received the alert in real-time .

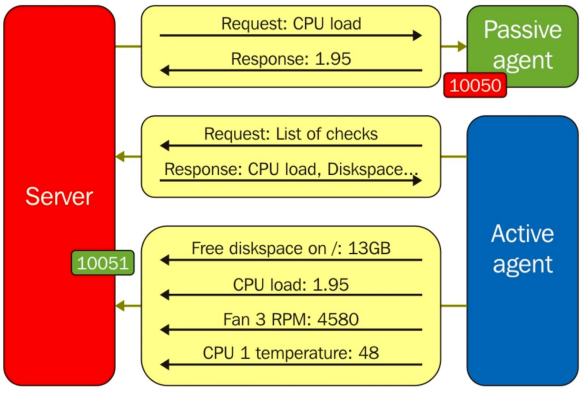
Appendix D:

grafana dashboard



Appendix E:

zabbix passive and active example and process



Appendix F:

List of the main parameters of an item

Host	Corresponds to the component to which the item is attached
Description	This first parameter corresponds to the name

Type	In this field, we have to select the type of item used such as Zabbix Agent or SNMP.
Key	The Key parameter is used to define the condition and/or the value that the item
Data type	Allows to specify the type of data expected
Type of information	is used to indicate the type of data to be recorded in the database for the value reported by the item.
Units	This field is used to inform the unit of measurement (for example bps, unixtime, ...etc) of the value reported by the item.
Use multiplier	you must specify in the Use field whether or not to use a multiplier, in particular in order to convert the values received by item
Custom multiplier	Depending on the previous choice.
Update interval	corresponds to the time in seconds between each new update of the item
Flexible intervals	Allows you to list exception periods with a time interval different from normal updates
Keep history	the length of time the history is kept in the database for this item must be indicated
Keep trends	This field is used to specify the number of days the trends are kept in the database for this item (used in particular for the creation of graphs).
Status	indicates whether the item is enabled, disabled or unsupported
Store value	is used to modify the value record of the item.
Show value	allows you to modify the display in the interface of the value reported by the item. By default it is the parameter As is which is required
Application	Allows you to link the item to one or more applications
Group	This optional parameter it is used to associate the item to a group.

costume intervals this is another important parameter , this define time when the item is checked, we have two methods first, Flexible intervals allow to redefine the default update interval for specific time periods here we have two additional parameters “interval” which is the duration of checks, and “period” the interval of time which this flexible intervals will be active. the period format should be like this d-d,hh:mm-hh:mm where d is the day written by number l.

Appendix G:
List of the main parameters defining a trigger

Name	Corresponds to the name of the trigger. The name can contain macros, usually it is similar to the item to which it is attached.
Expression	is used to fill in the logical expression used to calculate the state of the trigger.
Trigger depends on	This field is used to list possible dependencies with other triggers.
Dependency	Allows to add a dependency with a trigger

Event generation	The Normal option of the Event generation field allows to generate events normally, i.e. at each status change (PROBLEM/UNKNOW/OK).
Seversity	Indicates the severity (criticality) of the trigger.
Comments	is useful to specify some information about the trigger
URL	If this field is not empty, then this URL is used in the Status of Triggers screen.
Disabled	Allows to disable a trigger

Appendix H:
list of main components define an Action

Name	Name must be unique
Type of calculation	Choose how to calculate conditions, AND: all condition must be met, OR: one or multiple, and we can mix them
Conditions	List of conditions
New condition	Select conditions to add them to the list below
Enabled	To enable action

Appendix I:
Tools built around ElastAlert - that may remove the need for writing YAML configs

- ElastAlert API + Docker Image by bitsensor
- ElastAlert Kibana Plugin by bitsensor
- Praeco by ServerCentra

List of Figures

List of Tables

List of Abbreviations

API: Application Programming Interface

CSV: Comma-separated Values

DNS: Domain Name System

OS: Operating System

ELK: Elasticsearch, Logstash, Kibana

FQDN: Fully Qualified Domain Name

FTP: File Transfer Protocol

FS: File System

GPL: General Public License

ICMP: Internet Control Message Protocol

IP: Internet Protocol

IPMI: Intelligent Platform Management Interface

IT: Information Technology

LDD: Low-Level Discovery

LTS: Long-term Support

RDP: Remote Desktop Protocol

SNMP: Simple Network Management Protocol

SLL: Secure Socket layer

TCP: Transmission Control Protocol

HTTP: Hypertext Transfer Protocol

RHEL: Red Hat Enterprise Linux

REST: Representational state transfer

JMX: Java Management Extensions

VM : Virtual Machine

XML: Extensible Markup Language

Abstract

Modern companies, in case of errors or technical problems, need different tools and mechanisms to monitor their IT infrastructure and send real-time notifications and alerts to the administrators. Our graduation project , which is part of graduation internship thesis for a "Master's Degree in Networks and Distributed Systems" at the University of Tlemcen, proposes a solution to avoid these infrastructure malfunctions that could have a negative impact on the company image. The work carried out consists of setting up a tool for monitoring the IT infrastructure based on free open source software in terms of use cases between ELK stack with Elastalert and Zabbix.

Key words : ELK Stack , Elastalert , Zabbix , Monitoring , Logs , Alerting , Elasticsearch , kibana

Résumé

Les entreprises modernes ont besoin de différents outils et mécanismes pour superviser leurs infrastructures informatiques et envoyer des notifications et alertes en temps réel aux responsables en cas d'erreurs ou de problèmes techniques. Notre PFE, inscrit dans le cadre d'un mémoire de stage de fin d'études pour l'obtention du «Master en réseaux et systèmes distribués» à l'Université de Tlemcen, propose une solution pour éviter ces dysfonctionnements de l'infrastructure qui pourrait avoir un impact négatif sur l'image de marque de l'entreprise. Le travail effectué consiste à mettre en place un outil de surveillance de l'infrastructure informatique basé sur des logiciels libres en terme de cas d'utilisation entre ELK stack avec Elastalert et Zabbix

Mots clés : ELK Stack , Elastalert , Zabbix , Surveillance , Journaux , Alerte, Elasticsearch , kibana

ملخص

تحتاج الشركات الحديثة إلى أدوات وآليات مختلفة لمراقبة البنى التحتية لتكنولوجيا المعلومات الخاصة بها وإرسال إشعارات وتنبيهات إلى المسؤولين في حالة حدوث أخطاء أو مشاكل تقنية. هذا التقرير المسجل كجزء من أطروحة التدريب للحصول على "الماجستير في الشبكات والأنظمة الموزعة" في جامعة تلمسان ، يقدم حلاً لتجنب هذه الأعطال في البنية التحتية التي يمكن أن يكون لها تأثير سلبي على صورة العلامة التجارية للشركة. يتكون العمل المنجز من إنشاء أداة لمراقبة البنية التحتية لتكنولوجيا المعلومات على أساس البرمجيات الحرة و المجانية بين حالتي الاستخدام ELK stack مع Elastalert و Zabbix

الكلمات المفتاحية: ELK Stack، Elastalert، Zabbix، Monitoring، Logs، , Alerting , Elasticsearch , kibana