

POLITECNICO DI TORINO

Master's Degree in ICT for Smart Societies



Master's Degree Thesis

**Innovative Monitoring Systems and New Protocols for Wireless
Networks and Wireless Sensor Networks**

Candidate

Samaneh Sadat Ghoreishi

Supervisor

Prof. Daniele Trincherò

June 2022

To Sam

You have always been there for me

Table of Contents

Pages

Chapter 1.....	6
1 Introduction	6
1.1 Open Systems Interconnect (OSI) Model.....	7
1.2 Common Network Devices	8
1.2.1 Routers	8
1.2.2 Switches	9
1.2.3 Firewalls.....	9
1.2.4 Servers	9
1.3 How Data Passes Through a Network	10
1.4 Why Monitor a Network?	12
1.5 The Five Functions of Network Monitoring Systems.....	12
Chapter 2.....	18
2 Current Network	18
2.1 Thesis Objective.....	18
2.2 How Does a Network Monitoring Tool Monitor the Network?	19
2.2.1 Simple Network Management Protocol (SNMP).....	19
2.2.2 Windows Management Instrumentation (WMI)	20
2.2.3 SSH(Secure Shell)	21
2.2.4 Internet Control Message Protocol(ICMP).....	21
2.2.5 Why are network monitoring tools important?.....	22
Chapter 3.....	23
3 Network Monitoring System.....	23
3.1 What is a monitoring system?.....	23
3.2 Network Monitoring Softwares	24
3.2.1 The Dude	24
3.2.2 PRTG	25
3.2.3 SolarWind.....	26
3.2.4 Zabbix.....	27

3.2.5 Result.....	27
3.3 what is zabbix	27
3.3.1 Features	28
3.3.2 Characteristics.....	28
Chapter 4.....	33
4 Thesis Organization	33
4.1 Methodology.....	35
4.2 Convert.....	40
4.3 Mapping.....	45
4.4 Telegram	50
Chapter 5.....	54
5 Result.....	54
5.1 Ivrea city.....	54
5.2 Full Transform.....	56
Chapter 6.....	57
6 Conclusion.....	57
Bibliography	58

List of Figures

Pages

<i>Figure 1- Common Network Devices</i>	10
<i>Figure 2-Simple Network Components</i>	11
<i>Figure 3-The Dude Dashboard</i>	25
<i>Figure 4- PRTG Dashoard</i>	26
<i>Figure 5- SolarWind Dashboard</i>	26
<i>Figure 6-Zabbix Workflow</i>	32
<i>Figure 7-ZABBIX Database Configuration</i>	35
<i>Figure 8-ZABBIX Discovery Phase</i>	37
<i>Figure 9-ZABBIX Discovery Rule</i>	38
<i>Figure 10-ZABBIX Action Operation</i>	38
<i>Figure 11-ZABBIX Selction of Templates</i>	40
<i>Figure 12-SNMP Failure Devices</i>	44
<i>Figure 13-Network Map</i>	46
<i>Figure 14-The Dude Map in Ivrea City</i>	47
<i>Figure 15-GRAPHVIZ Layout</i>	49
<i>Figure 16-Final Map in ZABBIX Format</i>	50
<i>Figure 17-Telegram Bot Father</i>	51
<i>Figure 18-Telegram Media Type</i>	52
<i>Figure 19-Detailed Telegram Message in Media Type</i>	52
<i>Figure 20-ZABBIX Operation For Telegram Alert</i>	53
<i>Figure 21-Ivrea City Devices Added to ZABBIX by Discovery</i>	55
<i>Figure 22-Latest Data in Casa Blotto-Ivrea Device</i>	55
<i>Figure 23-Graph of Network Traffic in Casa-Blotto Device in Ivrea City</i>	55
<i>Figure 24- Map Creation in Ivrea City</i>	56

ACRONYMS

OSI	Open System Interconnection
NMS	Network Monitoring System
PDA	Personal Digital Assistant
CPU	Central Processing Unit
HTTP	Hypertext Transfer Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
FTP	File Transfer Protocol
DNS	Domain Name System
LAN	Local Network Area
SNMP	Simple Network Management Protocol
SLA	Service Level Agreement
IOT	Internet of Things
AWS	Amazon Web Services
VPN	Virtual Private Network
OS	Operating System
SQL	Structured Query Language
PRTG	Paessler Router Traffic Grapher
WMI	Windows Management Instrumentation
SSH	Secure Shell
ICMP	Internet Control Message Protocol
MIB	Management Information Base

OID	Object Identifiers
GUI	Graphical User Interface
IBM	International Business Machines Corporation
GPL	General Public License
JMX	Java Management Extensions
PHP	Hypertext Preprocessor
LTS	Long Term Support
IPMI	Intelligent Platform Management Interface
XLS	Microsoft Excel Spreadsheet
DOT	Graph Description Language
API	Application Programming Interface
GRAPHVIZ	Graph Visualization
JSON	Java Script Object Notation
XML	Extensible Markup Language
CSV	Comma Separated Values

SUMMARY

Computer networks have been used in everyday life for a long time, but recently, the majority of the enormous IoT projects need network appliances to have a properly integrated system for both administrators and clients as they need to keep their network simple and practical with installation, configuration, and communication with other networks.

In this field of technology, the most important factor is the quality alongside quantity at the same time to have a simpler, easier, accessible network with a one-size-fits-all solution to all network monitoring and management problems.

In this thesis, the focus is on monitoring systems in iXem laboratory research projects at the Polytechnic of Turin.

There are approximately 2000 routers and switches used to create a wide network topology in iXem project, with different capabilities and brands, at the beginning of the thesis, they were all connected to a proprietary monitoring system but as the project developed during the years, the number of connected devices on a given network has almost always constantly grown together with the variety of each data transmitted. media streams, application data, backups, and database queries.

Replication tends to saturate bandwidth just as much as they eat up storage space. also, there is the need to manage different physical mediums (fiber, cable, radio, and so on) and to provide high performance and availability, both on the connection and on the application level with more availability,

performance, power, flexibility, and adaptability, To meet specific needs, The reliability and stability of the network itself have become an important factor that cannot be ignored, so monitoring system emerges.

The fundamental aim of the thesis is improvement in the capability of the exciting network in the future. A cloud virtual experiment has been run in a smaller portion of the network and then applied on the entire project, using Zabbix.

To check Zabbix performance in analysis and prediction and the most important factors which is ease of deployment and use, we are using automated steps along the way to scale down the average time of each part. Hence it can work on other parts of the network in the Piedmont region and convert the entire network to Zabbix using different scripts in each section.

The main step at the beginning is to select the accurate version of the operating system to run Zabbix for the project and configure it on the AWS cloud and activate the way to access it from both monitoring systems contemporary with the help of two OpenVPN.

The layout of the scheme is to activate the most effective and operational protocols in each device, to automate the Zabbix implementation and have an efficient troubleshoot. This relies on a suitable SNMP protocol and all devices must be aligned with the same SNMP version, Python libraries, and modules, in particular, parallel SSH and Paramiko were the best automation solutions, and all devices were added to Zabbix by discovery method placed by the group of routers and switches and start to monitor all the substantial components in the real-time to set alerts situated for main critical factors.

The second major key has been to build a suitable interactive map to visualize devices on a network, their inter-relationships, and the transport layers providing network services. Since there are more than 2000 hosts in the network, there is a database that can export all the devices and useful information including ID, IP address, and parent's ID. The network is in layer 2 and creating a table based on neighbors was not possible since with regular protocols, all the devices in the network reply to each ping. Therefore, it has been needed to create a script to check the parents in the network and discover the neighbors and routes. Since the main format in export and import file are not the same, an open-source visualization software named Graphviz as a middle agent to take descriptions of graphs in a simple text language called Dotfile, and make diagrams in useful formats, Thus, I created another python script to get the correct format for Zabbix to create the map.

In the last step the alerting system has been created as Telegram BOT since Zabbix has all the media types and permission work alongside with Telegram token and as the final result, compares all the capabilities in the new monitoring system. In conclusion, I moved all the networks to a new, open-source, free, easy to use and real-time accurate monitoring system.

This migration, including the addition of all the devices and the creation of the related map, can be made in less than a day.

The new system helps network managers to evaluate potential solutions, solve problems and predict the incoming issues in a shorter amount of time compared to the past.

Chapter 1

1 Introduction

With the ongoing development of the Internet and computer technology, data has a fierce growth trend. If a network admin can't see all the devices and connections on the network, it can't guarantee that its network performance is up to snuff. Any network monitoring solution worth its salt will contain full network visibility features, such as automatic network device detection and dynamic network maps. If network traffic isn't routed properly, essential network data will take longer to reach its destination than it should. This is a critical problem to control if a business or project relies on delivering resources to customers and clients through a network. Many network monitoring solutions are designed to alert to any misconfigured routing protocols and help to remediate them through automatic updates, a device may be malfunctioning and the health of devices to ensure that no switch, router, or endpoint isn't working correctly is another reason to monitor the network, moreover, establish an intelligent network alerting system and Understand the future network growth are two other important factors in network performance. Based on the above reasons, this thesis proposes a real-time monitoring and prediction platform for users based on Zabbix. [1]

1.1 Open Systems Interconnect (OSI) Model

Understanding of basic networking begins with the Open Systems Interconnect model and specifically the layers and protocols that have the main impact on monitoring the systems.

The OSI model standardizes the key functions of a network using networking protocols. This authorizes individual device types from different vendors to communicate with each other over a network in several ways.

In the OSI model, network communications are grouped into seven logical layers. Two devices communicate using OSI standardized protocols at each layer. [2]

Layer	Function
Layer 7: Application	Supports communications for end-user processes and applications and presentation of data for user-facing software.
Layer 6: Presentation	Transform incoming and outgoing data from one presentation format to the form that application accepts (Data encryption, text compression).
Layer 5: Session	Opening, closing and managing communication sessions between end-user applications and their processes.
Layer 4: Transport	Oversees end-to-end delivery and also is responsible for flow control, segmentation and reassembly of packets.
Layer 3: Network	Routes and transfers variable length data packets between two nodes on a network using an IP address.

Layer 2: Data Link	Provides a functional and procedural connection between two connected nodes by correcting errors that may occur in the physical layer.
Layer 1: Physical	Transmits a bit stream over physical media and defines the relationship between a device and a transmission medium.

Table 1- OSI Model

The Datalink (2), Network (3), and Application (7) layers are the most common used for monitoring in the network. Network monitoring systems use these layers to discover the devices on the network and the way they are connected, to generate network topology maps and to monitor the network.

1.2 Common Network Devices

There is component used to connect a computer to other devices named network devices. Devices used to setup a Local Area Network (LAN).

Figure (1) shows the most common network devices and their relationship.

1.2.1 Routers

We can use routers for creating connections to the internet, networks in order to find the best route. Routers are located in the third layer. For example, create connections from a private network to the public network. A router acts as an agent, selecting the best path for the data traveling. Routers connect users to the internet.

1.2.2 Switches

Switches are located in the second layer and connect hardware devices including computers, printers, servers and other devices to the private network using packet switching to receive and forward data in the Data link layer of the OSI model.

1.2.3 Firewalls

Firewalls are responsible for network protection in hardware or software. Provide a control on all traffic in order to filter them or block the unauthorized or unwanted access on the data to gain a secure connection between the private network on devices and hosts and public network for access to the internet and beyond.

1.2.4 Servers

Networks deliver resources, data, services, or programs to users. A server is a piece of hardware or software that provides functionality for other programs, devices and applications. Servers take requests from users and respond accordingly. For example, the store all the data and it shares them when requested or provides accessibility for the same host or others. [3]

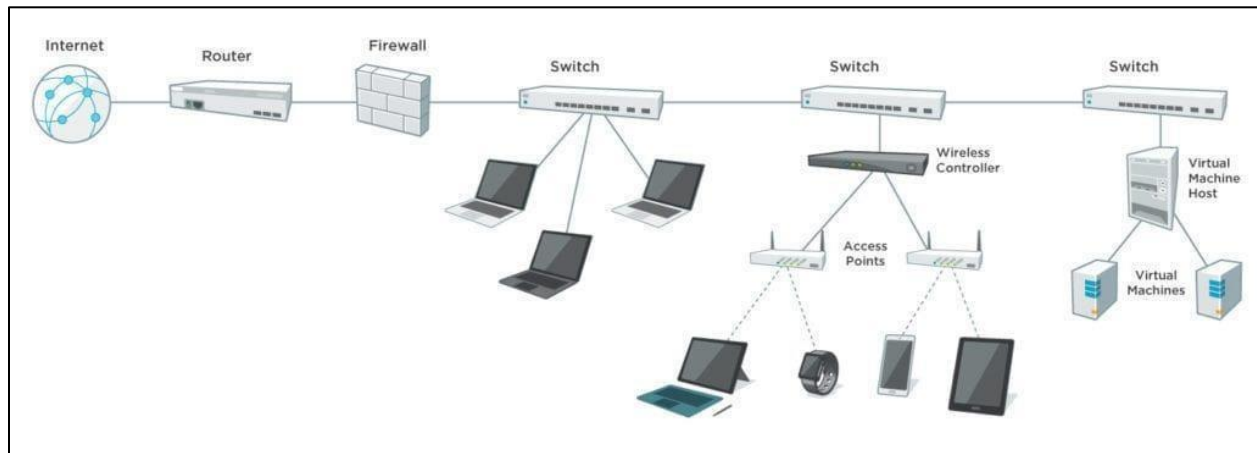


Figure 1- Common Network Devices

1.3 How Data Passes Through a Network

Most private networks are connected to the worldwide internet in different ways. For example, the internet connects remote users to central offices to access routine assignments and tasks. It connects customers and clients to related websites whenever they want.

Private networks are connected to the internet using specified routes that are determined by routers. Information is sent over the internet in the form of chunks named packets. data packets travel from one machine to another until they reach their destinations. All packets have the information in terms of IP form of source, and destination so they can travel across the internet. Each packet can carry a maximum of 1,500 bytes. Data assemblies arrive at the destination and the host fits them together like a puzzle to receive the complete data based on some extra parts named header and footer with all needed information about where this data came from and what is the final destination. Routes act like traffic officers in intersections and when they receive a packet that belongs to them they pass it to their destination in another private network.

In most recent networks, data packets must first pass through a firewall. The goal is to control the outgoing and ongoing traffic to protect private networks and elevates the security and it can be done by filtering traffic between the internet and private network. When an incoming data packet is flagged by firewall rules, it is blocked from the private network.

Data packets passed through the firewall are received by a switch in layer two on the private network. Switches are the bridges between laptops, servers, printers, and other devices to the private network in each host. A network interface card is responsible for the connection between all these devices and switches. Network Interface Card or NIC for short always has a unique Media Access Control or MAC address. Switches transmit data between devices in private networks using these MAC addresses that are impossible to change. [3]

In figure (2) it is possible to see a very simple network from the Internet to home devices and a server as a database in between and also the connection for wireless devices.

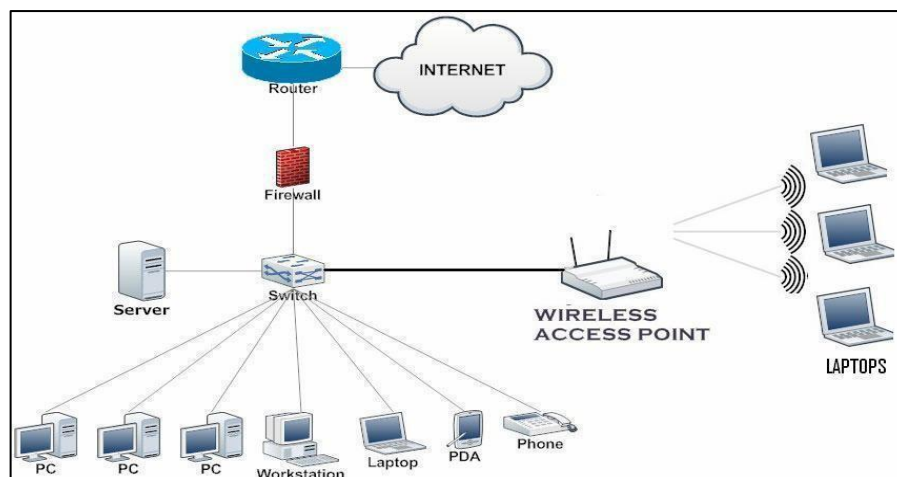


Figure 2-Simple Network Components

1.4 Why Monitor a Network?

The network is the main branch of the IT infrastructure river. When networks fail, the ongoing route of information required by applications and business operations stop and it is not just about damage or wreck in one part, sometimes a lack of quality can bring down the main parts of the system.

Network monitoring It's a key concept in a network manager's structure for troubleshooting network matters anytime and anywhere. Networks have some dynamic issues and then different dynamic topics to take care of, Admins continually grow the network by adding different users and devices and with them come various technologies and applications in each separate network. If network cannot adjust with all the changes in any level and cannot inform admins for better performance, the whole system fall apart but in case with a reliable monitoring system all the proceedings can detect in time and act in case of any issue or failure and predict the further problems and help to have integrated healthy network ready to work and grow and manage all the inside and outside communications and work with all kind of resources and when network face a problem, it can helps admin to find the cause and the beginning point and identify the weak point of the network before they occur even with alternating and periodic issues that is hard to evoke and diagnose. [3]

1.5 The Five Functions of Network Monitoring Systems

Network Monitoring Systems provide five basic functions:

1. Discover
2. Map
3. Monitor
4. Alert
5. Report
- 6.

NMSs differ in the capabilities they deliver for each of these functions.

Discover: Find the Devices on Network

The beginning step in network monitoring is discovery. For monitoring a network whether it is big or not, knowing all the devices and the way they are connected is the key point. discover all of the devices on the network – the routers, switches, firewalls, servers, printers and more.

NMSs include a library of monitoring templates, to define the way to monitor each different device. Device roles are type and vendor specific. For example, what monitor on a Mikrotik Router will differ from what monitor on a Ubiquiti product.

When a network monitoring system completes the discovery procedure, it automatically allocates an appropriate device role to each discovered device to continue.

Network Monitoring Systems differ in their discovery capabilities. All NMSs discover devices on the network. However, not all will discover how devices are connected to the network as an advantage for better performance in the discovery stage. For example, a NMS may have

identified a router on the network but it won't know what switch or router board it is connected to.

Apart from that when it comes to port connection between devices, a network monitoring tool, with Layer 2/3 discovery will discover the port-to-port connectivity between devices on the network and automatically specify the order of how they are connected. For efficient and executable it is important to know all the devices which are connected and how the complete plan connects to each other one by one.

The main reason for having information about the structure of connection between devices is, in a network when on device apart that the position fails, it can impact all the neighbors and further their performance and it can act like a domino. For instance, when a router stops communicating with the network, all switches and devices from that router branch fail to communicate and it is a big issue over a network that is functional for a business with clients and customers.

Map: Visualize the Network

Visualization in a network system is the most precious fault detection tool that an admin can go along with. The ability to visualize a network is the most valuable diagnostic tool for network admins and can save hours, and even days troubleshooting network problems.

Network monitoring systems develop network maps in order to help and prepare network admins to confront issues and also have a proper visualization of their network and all its devices. Network maps arrange a clear and well-ordered characterization of all the added maps in a network from the basic connection to represent all the latest status and information.

In most of the network monitoring systems to create a network map, there is a need for mapping tools and challenges with a major amount of manual processing to be able to visualize the map. Some tools merely provide a

drawing tool and rely on the Network Admin's knowledge to map out the network. Other tools can automatically discover the network based on databases and the information that gives them the specific father and children of each device in order to provide a map of neighbors and visualize a complete map of all the devices in the network and their connectivity.

Monitor: Keep an Eye on Your Network

A network monitoring software provides a complete product and ready for immediate use device roles that is useful for monitoring. Network admins can adjust or improve device roles or create new ones from the basis. NMSs represent network admins to an enormous selection of monitors based on their subject.

As a starting point, network admins are willing to monitor the "big 5" for each device on the network. Ping availability and latency, and CPU, memory, disk and interface utilization are the most critical information for any system admin to be aware of.

Most network monitoring tools include monitors for other hardware components such as the fans and power supplies in a switch, and even monitor the temperature in a wiring closet. Also network services like HTTP, TCP/IP and FTP or DNS can be monitored by request in NMSs.

Alert: Get Notified When Devices Go Down

Network Monitoring Systems are crucial aspects for Network Admins when network issues happen, allowing for faster troubleshooting. They are designed to provide alerts via email, text, Telegram and logging for the best performance in the network.

Some functions in alerting enabled admins to face the issues before they happen and effect on the network and its users and put impression on the applications and the business and basically disable the network from

normal and regular performance, these thresholds are needed to define in the NMS base on the capability of the network and admins responds. For instance, the NMS has been configured to the point that when the CPU utilization on a router exceeds 80%, it starts a concern alerting to admins before the impact. This allows the network admin to dynamically investigate and acknowledge before the router fails altogether at the time.

In network system monitoring, there are some performance metrics like CPU, memory and interface utilization that can vary during the day and change the parameters in different situations. In some moments they can exceed the threshold or maximum for an amount of time for example a few seconds or minutes during the up times in the network but they will come back to normal performance soon. Network admins in these periods don't want to alert or notify for such minor issues, here they can use alerting configurations to avoid the problem and use the time elements. For example, if CPU utilization exceeds 80% for more than 5 minutes, then the system can send an alert by telegram bot to admin to check the network related metrics.

At the end, network admins usually spend an amount of time for personal life like everybody else and they can configure their NMS in that blackout periods to act completely automate and suspend the alerts when it is needed, a good example can be related to save cost energy in evenings and nights and shut down devices like printers when there no use. Alerts can be stopped and be suspended from NMS to admins in those times by configuration in related fields and save a big amount of cost and effort and energy.

Report: Deliver on SLAs with Real-time and Historical Reporting

There is a non-stop ongoing life cycle of design, analysis and redesign of a network by admins that is always in rolling.

To keep up this life cycle, NMSs systems issue immediate and historical monitoring data. This information enables Network admins:

- To validate that network designs are delivering the desired and preference results
- To reveal trends that could impact the capability of the network to transfer the performance demanded by users, applications and systems
- To isolate and fix performance problems to provide comprehensive troubleshooting solutions
- And to provide evidence based on the SLA commitments

NMSs deliver monitoring information in web pages called dashboards. Dashboards provide an instant snapshot of an organization's key performance indicator. For example, a top 10 CPU utilization view or a Top 10 Memory utilization view.

Network Admins inspect dashboards for quick summary to assess the functionality of the entire network. And then move to detailed data by focusing on particular devices and monitor the isolated problems in performance.

It is possible in most of the NMSs to customize them. Network admins can create dashboards for different levels of their internal clients, their managers, line of business owners, Help Desk, and counterparts managing systems and applications. [3]

Chapter 2

2 Current Network

2.1 Thesis Objective

The main idea of the thesis is the project iXem labs based on IOT technology in the Piedmont area; all the communication devices are already connected to The Dude software for monitoring and related mapping, alert and reporting.

There are Ivrea, Casale, Asti and Sorin cities in the network range and about 2000 devices including; MikroTik routers, MikroTik switches and Ubiquiti devices, all connected.

The main subject for transform is the city of Ivrea in the dude with Mikrotiks, Ubiquiti and switches all connected together.

There are some backwards related to using the dude for monitoring the non Mikrotik devices so the main goal of this project is connect all the networks in the city to Zabbix and compare the result and get a more dynamic map and alerting system based on Zabbix and by the result and execute and perform all automation procedure on the rest of the network.

The experimental project runs in Ubuntu as an operating system on AWS cloud and executes on a windows OS by connection through SSH protocol

in a free terminal emulator named Putty and both side Open VPN for easy access.

2.2 How Does a Network Monitoring Tool Monitor the Network?

Some NMSs support scripting languages like Powershell in order to create custom monitors for Windows Servers, and SQL queries and also to create custom monitors for databases but in any case scenario, it is possible by using certain network protocols.

Network Monitoring Systems poll network devices and servers for performance data using standard protocols such as:

- SNMP
- WMI
- SSH
- ICMP

The two most widely used monitoring protocols are SNMP and WMI. They provide Network Admins with thousands of monitors to assess the health of their networks and the devices on them. In this project the main and fundamental protocols are SNMP, SSH and ICMP but not the WMI protocol since there is no windows base device in the network.

2.2.1 Simple Network Management Protocol (SNMP)

SNMP is an internet standard protocol for collecting information from almost any network attached managed devices, including: Routers,

Switches, Wireless LAN Controllers, Wireless Access Points, Servers, Printers and more.

In the application layer, SNMP asks questions about something, basically talking to the network to find out about related information, especially in order to express an object's doubts about whether it works or to check its validity or accuracy by querying "Objects". An object is something that an NMS collects information about. For example, CPU utilization or bandwidth are an SNMP object.

SNMP works by sending messages that are called protocol data units maintained in a Management Information Base, or MIB. The objects in a MIB are catalogued using an standardized numerating system. Each object has its own, unique Object Identifier, or OID. These components gather information to bring back to the network requester.

As a leading network monitoring software choice, SNMP can give a wide variety of data about network performance, and it works with different versions based on device type and vendors so can update in a way that enhances the overall quality of network monitoring. [4]

2.2.2 Windows Management Instrumentation (WMI)

WMI from Microsoft is a suite of tools and extensions within the windows driver model, a software for consolidating the management of devices and applications in a network from windows computing systems.

This protocol offers a uniform way to creates an operating system interface that receives information from devices running a WMI agent and execute scripts that can check continuously and WMI gathers details about the operating system, hardware or software data, the status and properties of remote or local systems, configuration and security information, and process and services information. It then provides all of this information

along to the network management software, which monitors network health, performance, and availability.

Microsoft assembles WMI in order to work with SNMP and other protocols. Even though it is a proprietary protocol for Windows-based systems and applications. [5]

2.2.3 SSH(Secure Shell)

Secure Shell (SSH) is common TCP network protocol for operating network services securely over an unsecured network. the physical location of the systems in both ends in a SSH connection is not a matter because typical applications include remote command-line, login, and remote command execution and in common, all the network devices can be secured with SSH.

SSH provides a secure channel over an unsecured network by using a client– server model, connecting an SSH client that always initialize the setup of connection with an SSH server listen for incoming connection requests, most of the time on port 22 on the host system and respond to them. The way that SSH works is the use of two related keys, as a public key and private key, therefore, together create a pair key to use in a secure access. In addition to providing strong encryption, SSH is widely used by network administrators to manage systems and applications. In this project, SSH is one of the main protocols in use, on both sides of network monitoring systems for accessing them with secure credentials. [6]

2.2.4 Internet Control Message Protocol(ICMP)

ICMP is a network level protocol. It relays messages from the receiver to the sender about the data that was supposed to arrive. It sends control

messages such as destination network unreachable, source route failed, and source quench. The primary purpose is error reporting in case of if the data did not get to its intended destination when two devices connect over the internet and one of them is a sending device. It uses a data packet structure with an 8-byte header and variable-size data section.

Ping is a utility which uses ICMP messages to report back information on network connectivity and the speed of data relay between a host and a destination computer. [7]

2.2.5 Why are network monitoring tools important?

Businesses depend on networks, whether they're on-site or remote. Network monitoring tools are fundamental to:

- Ensure detect the problem before the users of business get noticed and affect them.
- Provide insights on not just failure and fault issues but network slowness and help to detect and troubleshoot any performance degradation.
- survey the performance at the device and provide historical data and establish a baseline and interface levels using performance metrics.
- Provide decision-making data that can be used in many different ways. [8]

Chapter 3

3 Network Monitoring System

3.1 What is a monitoring system?

Modern networks are very complicated in detail. With this complexity the possibility of problems and errors concerning the network are increasingly higher. Moreover, networks are often the essential parts of any business, and so when they do not have a full functionality and always suffer from multiple problems the business is the main victim and never can grow or improve.

Network monitoring is a critical and one of the most important aspects of the IT process that can track network elements and units and provides troubleshooting, performance, detect deficiency and traffic monitoring. It involves monitoring fundamental network issues, providing weakness detection, and health monitoring of variety of network components from the device level in host to the protocol field and interface levels, as network grow, it provides visualization for administrators, as it knows all the backbones of network improve the security and track trends without spending hours to discover and deep dig for a fault in any moment.

There are original and basic tools that monitoring systems use to trace network processes. These tools are principal for various tasks including monitoring the traffic, bandwidth and other metrics. Further, for managing the resources on both side in on-site and remote a network needs on demand health and performance check to detect problems. IT admins have a beneficial performance using network monitoring to decrease the meantime of repair and solve the errors in real time with an instant alerting from structures like tables, charts, reports and graphs gathered by monitoring system tools. [9]

3.2 Network Monitoring Softwares

To decide to choose a capable and suitable network monitoring among the all softwares that can be found at the time, there were conditions and situations that could help the comparison for the selected city, thus for other cities in the existing network for network management but first there is a quick review on exciting NMS (The Dude) to check the vulnerabilities and insufficiency and shortage, therefore, the three top choices based on the current network needs were analyzed.

3.2.1 The Dude

The dude is a network monitoring tool that is managed by MikroTik which can improve the management in MikroTik devices. The environment is designed closely to MikroTik routers and it can automatically scan all devices within specified subnets and can discover the type of device. The installation and usage is easy and it can support most of the protocols in network monitoring. This monitoring system currently is using in the iXem but as imagine, there are some issues and downwards trends; the significant amount of devices in iXem are ubiquiti brands and they cannot be monitor properly by The Dude also the real time statistic in current NMS

is not accurate, it's not lightweight and in terms of history and alerting has problems. [10]

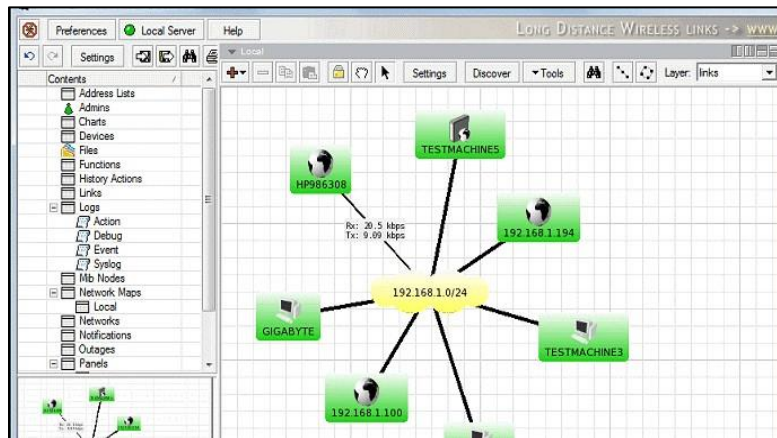


Figure 3-The Dude Dashboard

3.2.2 PRTG

PRTG is a software that is known for its advanced infrastructure management capabilities. It has a very particular feature that can monitor devices in the data-center with a mobile app based on a QR code but it is a very comprehensive platform with many features and moving parts that require a significant amount of time to learn and another disadvantage is in the pricing, it has a free edition up to 100 sensors but in unlimited time. [11]

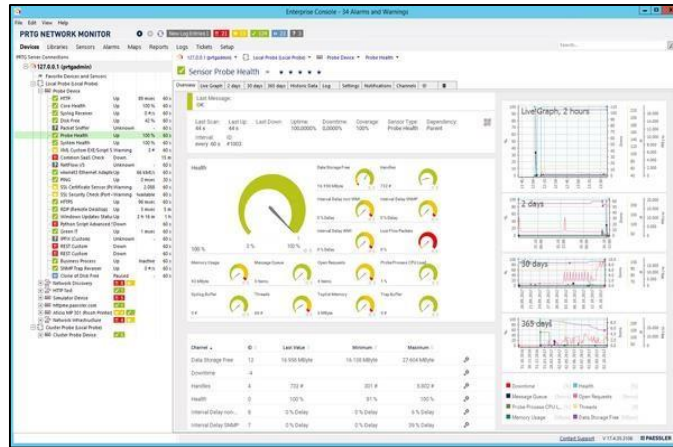


Figure 4- PRTG Dashboard

3.2.3 SolarWind

Solarwind is easy to set up and has the automatic discovery tool and it can deploy within an hour. Its interface is highly customizable and has a simple approach. based on this feature, it has web-based performance dashboards, charts and views. The product is accessible by separate modules that should be paid and one-time license including first year maintenance but it is a feature-rich tool designed for sysadmins and its OS is windows server. [12]

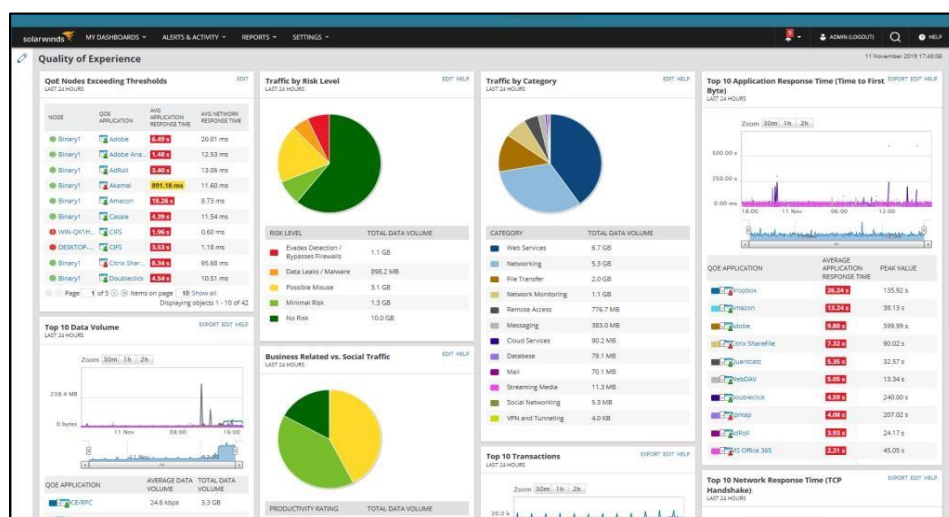


Figure 5- SolarWind Dashboard

3.2.4 Zabbix

Zabbix is an open source monitoring tool and easy-to-use and pleasing Web GUI that is fully configurable and is frequently used for monitoring network hardware. One of the particular features is forecasting future behavior to predict trends in data. Since its open source, it has an active user community among the world and very powerful documentation, moreover, in the terms of pricing it is free. [13]

3.2.5 Result

As a result Zabbix selected for the network monitoring system in terms of pricing, user friendly, scalability, powerful Web GUI, Ubuntu based OS and wide documentation.

3.3 what is zabbix

zabbix is an open source program created by Alexei Vladishev and one of the most powerful programs for monitoring networks and applications. This software is frequently used to monitor and detect the status of servers and network hardware. ZABBIX can collect almost all network information and it can detect new devices and configuration changes immediately. The amount of devices that this system can monitor is a wide range and this ability helps the developing networks to have confidence in using it. Zabbix can be installed on Linux, Unix and Windows systems.

Zabbix supports MySQL, PostgreSQL, SQLite, Oracle and IBM DB2 for data storage. Server-side programming uses C language and user-side programming uses PHP language. ZABBIX monitoring software is also released under the GPL V2 license, so it is completely free for commercial and non-commercial use. [14]

3.3.1 Features

- . the variety of data collection methods
- . Multiple operating systems supported by Zabbix
- . Variety of warning methods
- . Intelligently identify equipment or servers under the network
- . Distributed design to cover very large scales
- . Ability to create large clusters to cover heavy information traffic
- . Ability to expand and customize the system
- . Convenient scalability

3.3.2 Characteristics

◀ SNMP Protocol Support

Zabbix supports the SNMP protocol found on most network equipment such as switches, routers and servers. Which can play a vital role in network management by providing information about the network, CPU, memory, port status and more.

As in this project, SNMP is used as the main network protocol to cover and monitor the whole system.

◀ Virtual Infrastructure Monitoring

Using Zabbix, network admin can automatically identify and monitor a variety of virtualization systems such as VMware vSphere, VMware vCenter, HyperV through Low Level Discovery.

◀ Customization

Network admins in Zabbix will be able to expand or customize their monitoring system, using python, perl, shell, php programming language or any other programming language

◀ Database and Web Monitoring

Zabbix will be able to monitor all types of SQL Server, MySQL, PostgreSQL, Oracle, etc. databases to identify Query Slow and other things that decrease database performance. Zabbix allows admins to monitor all the pages of the website individually and also enter the pages that need to be logged in by entering the Username and Password and analyze the circumstances in the page.

◀ Monitoring Java Servers

After version 2, Zabbix has added a new feature written in Java called Zabbix .Java Gateway that allows admins to monitor Java-based software using JMX Java Management Extensions. It sends the appropriate JMX Counter to the Zabbix Java Gateway, and the Zabbix Java Gateway receives the answers without the need for any other Java-based software and sends them to the server, which is very important when it's needed.

◀ Hardware Monitoring

Zabbix monitoring allows the hardware that has IPMI capability to be connected directly without the need for any interface and to receive the required information such as temperature, fan speed, hard drive status, etc. and if Problems such as rising temperatures happens, a device execute IPMI commands in the network platform to turn on or off network equipment.

IPMI stands for Intelligent Platform Management Interface.

◀ **Monitoring Environmental Conditions**

Using Zabbix, network admins can use sensors that are used to control humidity and temperature, as well as ambient pressure, by monitoring environmental conditions.

◀ **Monitoring without the Need for a Software Agent**

In some cases, it may not be possible to use Agent on some systems and equipment. For this purpose, Zabbix offers a feature called Agent Less, i.e. without the need for Agent.

Zabbix consists of several important components, which are:

❖ **Server**

This component is the main and central component of Zabbix, which includes a report on the availability of agents, integrated information and statistics. It is also a repository of all configurations, charts and operations stored on the data. In general, it can be said that the core is the center of Zabbix and all the important and main configurations of Zabbix are done by this section.

❖ **Database Storage**

All configuration information as well as information collected by Zabbix server is stored in the database, which can be Mysql, Sql Server, Oracle, etc.

❖ **Web Interface**

To access Zabbix, you can use it anywhere and on any platform through the web interface, which usually runs on the server where Zabbix is running.

❖ **Proxy**

Zabbix Proxy can receive information from the systems to be monitored as a representative and then send that information to the Zabbix server, so it is very suitable for environments where we can not access those systems directly.

❖ **Agent**

This section sends the collected information such as system programs and resources, etc. to monitor the mentioned system for Zabbix server.

❖ **Data Flow**

This section describes the workflow in Zabbix. [13]

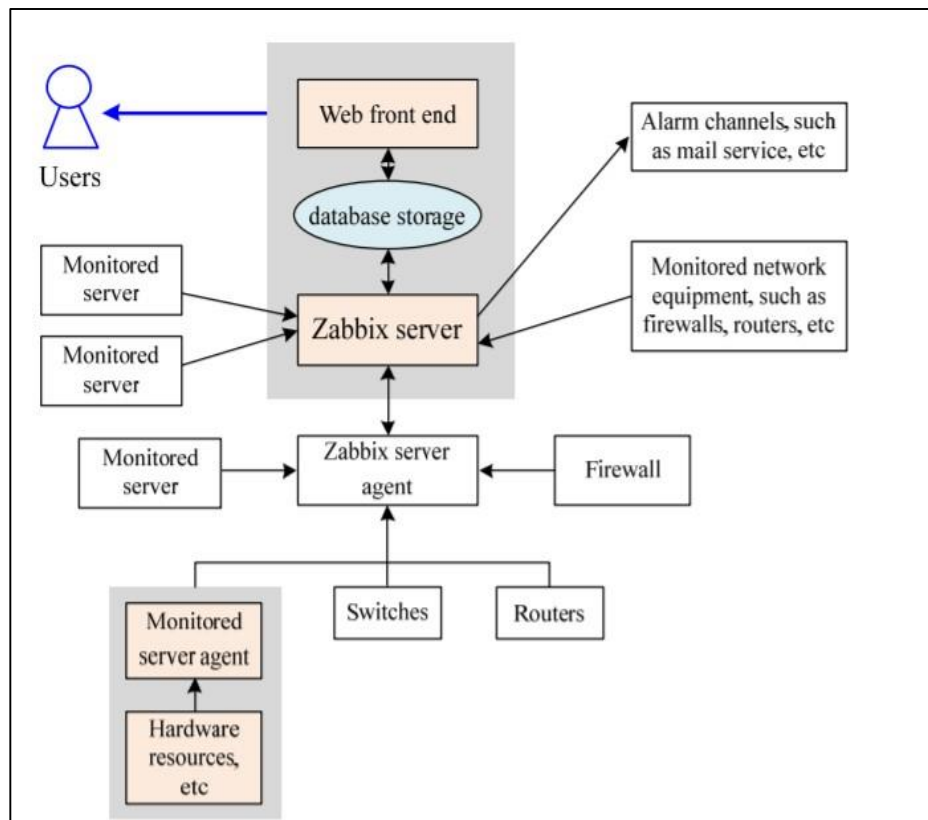


Figure 6-Zabbix Workflow

Chapter 4

4 Thesis Organization

For installing Zabbix, I choose Zabbix 5.0 LTS (long term support)

Because during the time and project it is more stable and the package used Ubuntu as operational distribution in version 20.4. installing Zabbix from the package has some benefits compared to downloading the latest source code and compiling it in this project because it is usually easy to update and upgrade and all the dependencies are automatically sorted out.

a. Install Zabbix repository

```
#wget  
https://repo.zabbix.com/zabbix/5.4/ubuntu/pool/main/z/zabbixrelease/zab  
bix-release_5.4-1+ubuntu20.04_all.deb  
# dpkg -i zabbix-release_5.4-1+ubuntu20.04_all.deb  
# apt update
```

b. Install Zabbix Server, Frontend ,agent

```
# apt install zabbix-server-pgsql zabbix-frontend-php php7.4-pgsql  
zabbixapache-conf zabbix-sql-scripts zabbix-agent
```

For installing the database in server side; Edit the file in

/etc/zabbix/zabbix_server.conf

And the only change is to put the DBpassword equal to zabbix.

The configuration on the agent side is quite easy; basically, we need to write the IP address of our Zabbix server.

c. Configure PHP for Zabbix frontend

It is very important during the process in the front-end to edit the time zone.

```
# php_value date.timezone Europe/Roma
```

d. Start Zabbix server and agent processes

Start Zabbix server and agent processes and make it start at system boot.

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

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

After finishing the main installation by entering the IP the next step is configure database connection.

ZABBIX

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type: PostgreSQL

Database host: localhost

Database port: 0 (0 - use default port)

Database name: zabbix

Database schema:

Store credentials in: Plain text (selected) | HashiCorp Vault

User: zabbix

Password:

Database TLS encryption: ☒

Back | **Next step**

Figure 7-ZABBIX Database Configuration

There is some reason for defining PostgreSQL as database type here. PostgreSQL is an Object Relational Database Management System and can support modern applications like JSON, XML etc. then the server name is localhost and the port is 10051.

4.1 Methodology

After configuration on Ubuntu side and check the availability of web browser GUI, it is time for configuration of vpn for this specific project as a special solution. Because this project was done remotely there were two open vpn on each side for connection to the dude and zabbix server.

In Ubuntu for connecting to other networks on the internet, one of the options was configure an openvpn so all the networks can connect to each other.

Before explaining the zabbix side of the project, in zabbix each single, specific box or appliance in the network is a host. Host groups are very useful as they make it easy to navigate Zabbix's interface, separating hosts into categories and allowing admins to organize and manage a huge amount of appliances without having to deal with impossibly long lists of hostnames. The same host can be part of different host groups. [15]

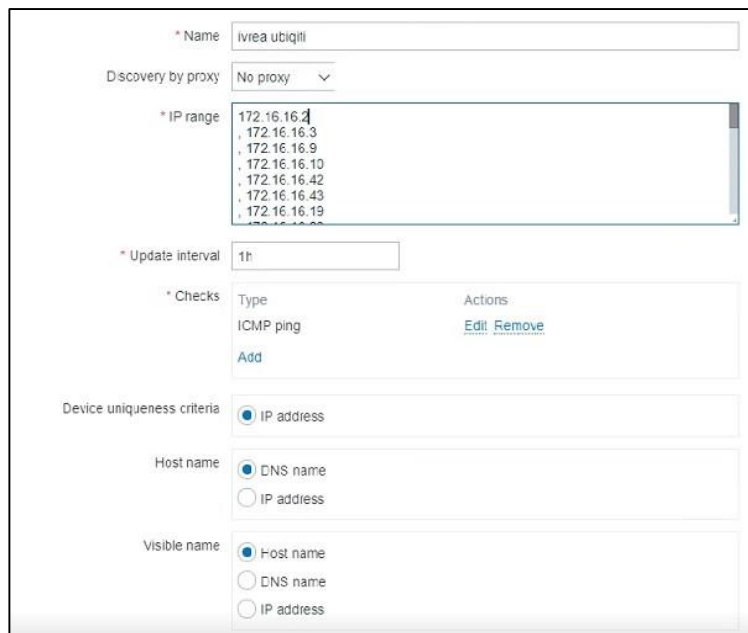
Base on zabbix interfaces each host or host group can notify by SNMP, ICMP, TCP or external checks to can navigate the host so base on a simple ICMP echo request, an SNMP query, an SNMP trap, netflow logging, or a custom script can connect to the host and monitor it.

Starting to add routers to zabbix by using fping in Ubuntu so it can get a list from IPs to check the availability in both Mikrotik and ubiquities and limit the IP addresses for discovery part in Zabbix base on ICMP echo request and put them in the right host group in zabbix and create a Discovery rule so it can provide a way to automatically create items, triggers, and graphs for different entities on a device base on their group host (mikrotik, ubiquiti or...)

In the discovery rule, the first thing to notice is SNMP version that is in this test is version1 for all ubiquities products and version two for all Mikrotiks, the difference is version1 doesn't offer real security for the monitoring data that crosses the network between an appliance and the monitoring server.

The second important information is finding the right OIDs to monitor, OID uniquely identifies managed objects in a MIB hierarchy. This can be depicted as a tree, the levels of which are assigned by different organizations. Top level MIB object IDs (OIDs) belong to different standard organizations, OIDs are sent and received by SNMP agents and servers as

dotted sequences of numbers. Just like IP addresses, this is convenient for machine-to-machine communication, but hard to read for humans. Right OID can be found in vendor's documentation on the device or using SNMPwalk utility or simply like here in item part of host by SNMP agent.



The screenshot shows the ZABBIX Discovery Phase configuration form. It includes the following fields and options:

- Name:** A text input field containing "ivrea ubiqiti".
- Discovery by proxy:** A dropdown menu set to "No proxy".
- IP range:** A text input field containing "172.16.16.2", with a list of IP addresses (172.16.16.3, 172.16.16.9, 172.16.16.10, 172.16.16.42, 172.16.16.43, 172.16.16.19) displayed below it.
- Update interval:** A text input field containing "1h".
- Checks:** A table with columns "Type" and "Actions". It contains one row for "ICMP ping" with "Edit" and "Remove" links. An "Add" button is at the bottom.
- Device uniqueness criteria:** A radio button selection with "IP address" selected.
- Host name:** A radio button selection with "DNS name" selected.
- Visible name:** A radio button selection with "Host name" selected.

Figure 8-ZABBIX Discovery Phase

In the second phase, to access the discovery actions section in the web UI, head to Configuration | Actions and then select Discovery from the Event source drop-down menu that is a part for define actions base on discovery rules, there are two part(action and operations),The action section lets define conditions based on the event's reported host IP address, service status and reported value, discovery rules, and a few others and in condition part it can define any new condition base on provide filtering intelligence and also choose the right host group in discovery rules to be equal with logical operators.

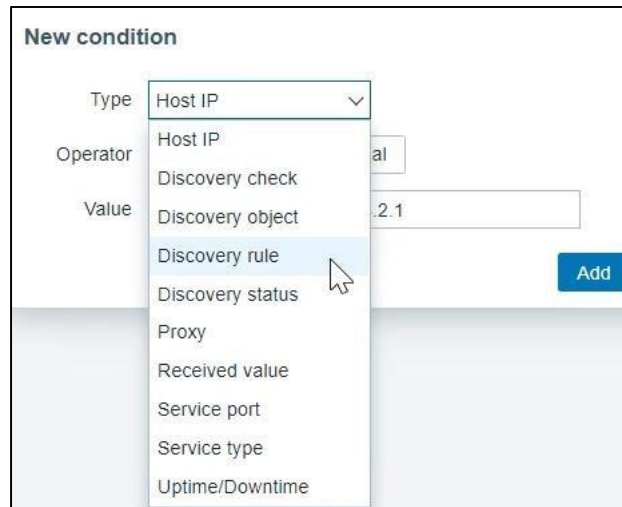


Figure 9-ZABBIX Discovery Rule

An action operations section provides the action's core functionality. Here,

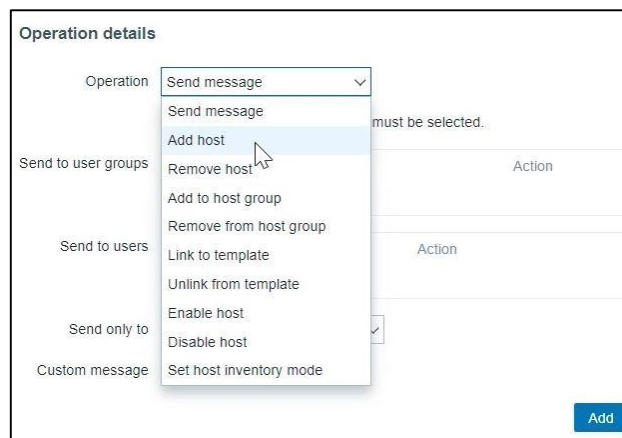


Figure 10-ZABBIX Action Operation

it should add a host but also it can send a message or any operation connected to the host and add the related template from the host group. (here, the template is network devices)

After defining the actions, one more step to do is provide templates for hosts. A template is a set of entities that can be conveniently applied to multiple hosts.

The entities may be:

- ◀ Items
- ◀ Triggers
- ◀ Graphs
- ◀ Applications
- ◀ Dashboards
- ◀ Low-level discovery rules
- ◀ Web scenarios

As many hosts in real life are identical or fairly similar so it naturally follows that the set of entities (items, triggers, graphs, ...) you have created for one host, may be useful for many. Of course, you could copy them to each new host, but that would be a lot of manual work. Instead, with templates you can copy them to one template and then apply the template to as many hosts as needed.

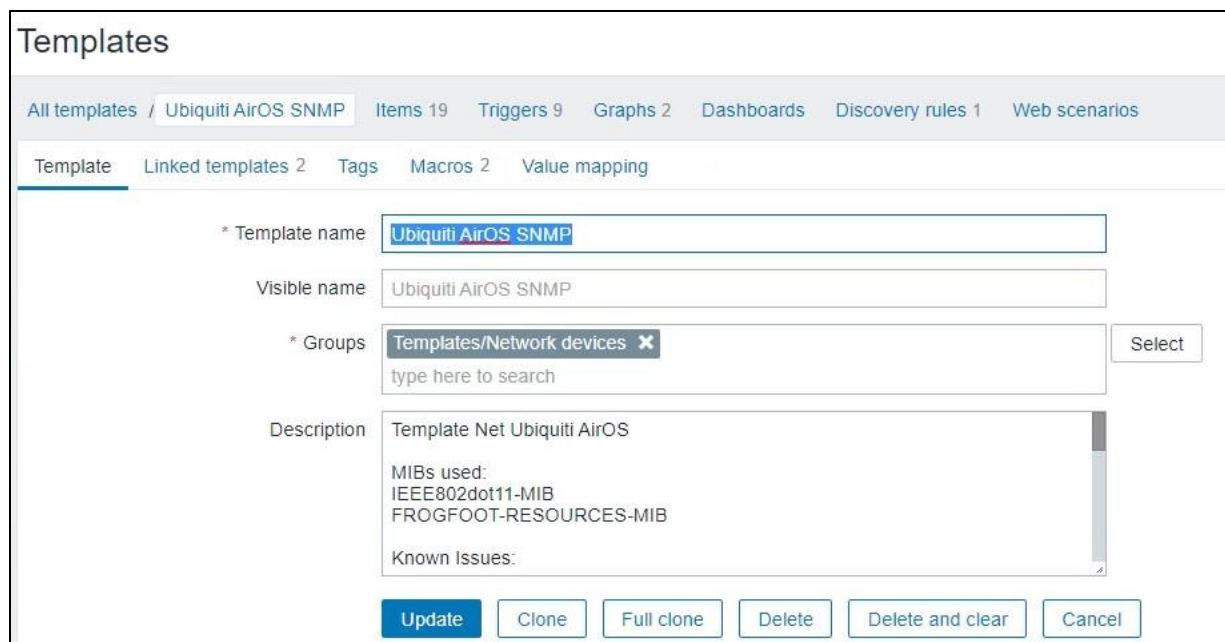
When a template is linked to a host, all entities (items, triggers, graphs,...) of the template are added to the host. Templates are assigned to each individual host directly (and not to a host group).

Templates are often used to group entities for particular services or applications (like Apache, MySQL, PostgreSQL, Postfix...) and then applied to hosts running those services.

Another benefit of using templates is when something has to be changed for all the hosts. Changing something on the template level once will propagate the change to all the linked hosts.

Thus, the use of templates is an excellent way of reducing one's workload and streamlining the Zabbix configuration. [13] [15]

There is a unique template for both ubiquiti and mikrotik in zabbix.



The screenshot shows the Zabbix web interface for editing a template. The title is 'Templates'. Below it is a breadcrumb trail: 'All templates / Ubiquiti AirOS SNMP'. To the right of the breadcrumb are counts for various items: 'Items 19', 'Triggers 9', 'Graphs 2', 'Dashboards', 'Discovery rules 1', and 'Web scenarios'. Below this is a sub-menu with 'Template' (selected), 'Linked templates 2', 'Tags', 'Macros 2', and 'Value mapping'. The main form contains the following fields:

- * Template name: A text input field containing 'Ubiquiti AirOS SNMP'.
- Visible name: A text input field containing 'Ubiquiti AirOS SNMP'.
- * Groups: A dropdown menu showing 'Templates/Network devices' with a close button (X). Below it is a search box with the placeholder text 'type here to search'. To the right of the dropdown is a 'Select' button.
- Description: A text area containing 'Template Net Ubiquiti AirOS'. Below this, it lists 'MIBs used: IEEE802dot11-MIB' and 'FROGFOOT-RESOURCES-MIB'. Below that is a section for 'Known Issues:'.

At the bottom of the form are several buttons: 'Update' (highlighted in blue), 'Clone', 'Full clone', 'Delete', 'Delete and clear', and 'Cancel'.

Figure 11-ZABBIX Selction of Templates

4.2 Convert

For most common Linux-based applications and devices, enabling the SNMP background service is an essential step to configuring the host for monitoring. look at the network and figure out the components that want to monitor, the kind of data that want to collect, and the conditions which to be notified about problems and state changes, using SNMP protocol is the best solution.

For activation SNMP on both mikrotik and ubiquiti routers at once, one way is writing a script for each of them then activating them based on that.

For ubiquiti; first start to use the paramiko library in python and ssh client to write a python script for executing multiple SSH commands.

SSH is the method typically used to access a remote machine and run commands, retrieve files or upload files and the python paramiko model gives an abstraction of the SSHv2 protocol with both the client side and server side functionality. As a client, you can authenticate yourself using a password or key and as a server you can decide which users are allowed access and the channels you allow. [16]

In ubiquiti routers for activating SNMP and its configuration in one session it is need to change the status in SNMP field from disable to enable for all routers then put community field on public, change all locations to Italy and also change RO community (read-only community)on public base on code in each session all the hosts can connect with IP and username and change SNMP configuration.

Another way is the use of parallel-SSH in Ubuntu based on python again.

First creating a host file that hosts will be listed in one line so all IPs in Ivrea city go to a IvreaIPs file then with parallel-SSH all commands execute on every host in one session so instead of running commands individually, create a single file that contain all of the commands that should run on remote server.

The main command is

```
Parallel-SSH -i -A -l USERNAME -h LIST_OF_IPS -I < COMMANDS_FILE  
| tee A_FILE
```

Where;

-i is --inline

Display standard output and standard error as each host completes.

-A is --askpass

Prompt for a password and pass it to ssh. The password may be used for either to unlock a key or for password authentication. The password is transferred in a fairly secure manner (e.g., it will not show up in argument

lists). However, be aware that a root user on your system could potentially intercept the password.

-l is --user

Use the given username as the default for any host entries that don't specifically specify a user.

-I is --send-input

Read input and send to each ssh process. Since ssh allows a command script to be sent on standard input, the -I option may be used in lieu of the command argument.

And username here is Admin and commands file is contain of below commands;

```
sed -i 's/snmp.status=disabled/snmp.status=enabled/g' /tmp/system.cfg
sed -i '/snmp.status=enabled/a snmp.community=public' /tmp/system.cfg sed -i
'/snmp.status=enabled/a snmp.location=italy' /tmp/system.cfg sed -i
'/snmp.status=enabled/a snmp.contact=callAdmin' /tmp/system.cfg sed -i
'/snmp.status=enabled/a snmp.rocommunity=public' /tmp/system.cfg

sed -i 's/snmp.community=.*snmp.community=public/g' /tmp/system.cfg sed -i
's/snmp.location=.*snmp.location=italy/g' /tmp/system.cfg sed -i
's/snmp.contact=.*snmp.contact=italy/g' /tmp/system.cfg
sed -i 's/snmp.rocommunity=.*snmp.rocommunity=public/g'
/tmp/system.cfg ubntconf cfmtd -w -p /etc/
```

and the last part of the command gets the result and puts it in the specific file.

probably there are some failures so by searching the word failure in the fail routers they can be found and troubleshooteed.

In mikrotik routers to activate SNMP services, there is a graphical and easier way to enable SNMP in terminal, SNMP set enabled on yes and in SNMP public where is in version two the mechanism is the device and server exchange a string for authentication and here community in all Mikrotiks sets on public.

SNMP set enabled=yes

SNMP community set public addresses= "SERVER_IP"

Here the IP is Zabbix IP so all the routers can only answer to Zabbix.

In this case scenario, first for each city export a CSV file of just mikrotik devices by filtering and in Ubuntu, create a file in the name of that city with mikrotik sign in name. then using the same technique as ubiquiti based on parallel SSH to activate the range at once by creating a file with enable commands and putting them in a file.

Parallel-SSH -i -A -l USERNAME -h LIST_OF_IPS -I < COMMANDS_FILE l tee A_FILE

Usually there are some failures among the routers.


```

samane@DESKTOP-00UF8FI:~$ cat casaleastimoks.xls2 | grep FAILURE
[1] 21:29:00 [FAILURE] 172.16.8.138 Exited with error code 255
[32] 21:29:02 [FAILURE] 172.16.8.132 Exited with error code 255
[62] 21:29:05 [FAILURE] 172.16.8.247 Exited with error code 255
[63] 21:29:05 [FAILURE] 185.168.99.1 Exited with error code 255
[92] 21:29:07 [FAILURE] 172.16.9.186 Exited with error code 255
[95] 21:29:08 [FAILURE] 172.16.8.83 Exited with error code 255
[124] 21:29:10 [FAILURE] 172.16.8.142 Exited with error code 255
[201] 21:29:19 [FAILURE] 172.16.8.155 Exited with error code 255
[217] 21:29:22 [FAILURE] 172.16.8.170 Exited with error code 255
[218] 21:29:22 [FAILURE] 172.16.8.133 Exited with error code 255
[223] 21:29:23 [FAILURE] 172.16.25.116 Exited with error code 127
[252] 21:29:27 [FAILURE] 172.16.9.17 Exited with error code 255
[263] 21:29:29 [FAILURE] 172.16.8.224 Exited with error code 255
[272] 21:29:30 [FAILURE] 172.16.9.7 Exited with error code 255
[275] 21:29:31 [FAILURE] 172.16.9.82 Exited with error code 255
[281] 21:29:31 [FAILURE] 172.16.9.77 Exited with error code 255
[288] 21:29:32 [FAILURE] 172.16.9.27 Exited with error code 255
[299] 21:29:33 [FAILURE] 172.16.25.140 Exited with error code 255
[305] 21:29:34 [FAILURE] 172.16.25.139 Exited with error code 255
[358] 21:29:39 [FAILURE] 172.16.25.217 Exited with error code 255
[362] 21:29:39 [FAILURE] 172.16.25.215 Exited with error code 255
[363] 21:29:39 [FAILURE] 172.16.9.71 Exited with error code 255
[378] 21:29:40 [FAILURE] 172.16.25.193 Exited with error code 255
[382] 21:29:41 [FAILURE] 172.16.27.100 Exited with error code 127
[383] 21:29:41 [FAILURE] 172.16.27.101 Exited with error code 127
[387] 21:29:41 [FAILURE] 172.16.27.136 Exited with error code 255
[389] 21:29:41 [FAILURE] 172.16.27.102 Exited with error code 127

```

Figure 12-SNMP Failure Devices

The reason to drop all the data in a XLS file is that the failures can be filter by grep command and at the same time by using the command;

Fping -4 -F LIST_OF_IP_CITY

It can be seen and by comparing these two, the ones that are unreachable in the second command don't exist and others in the failure list can be checked one by one.

The first reason for failure is some devices are switched and they cannot be reached by SSH, second reason is in some devices, the port 22 (SSH port) is not active and it was inactive from the first configuration. Therefore, it needs to be active via the services section in the router manually, another reason is some devices have different passwords and finally the last reason is some ubiquiti devices are accidentally in the Mikrotik category and their way for SNMP activation is different as mentioned before.

In SNMP community public in the part of addresses if I enter the IP in range of routers IP when the VPN is connected it cannot accept IP from other sources.

Zabbix has templates that can be read from SNMP for Mikrotik and each one has an OID and it knows what is the transmitter OID in each router and base on that can track the device and start monitoring procedure.

Zabbix server is the central process of Zabbix software which is calculates triggers and sends notifications to admin. In this work, with progress through automated monitoring system builds, Zabbix server tends to triggers alerts and problems therefore I manually change the configuration inside the server configuration file in some parameters based on Zabbix recommendation :

1. Cache Size
2. History Index Cache Size
3. Trend Cache Size
4. Value Cache Size
5. Log Slow Queries

4.3 Mapping

Network mapping is the operation of visualizing all the hosts in the existing network, their connection and structure with useful information about functionality of the particular device and the network. Configure and generate the network map by automation during the process is the main key of real-time development in this project.

Creating the map, especially when there are hundreds to thousands of devices are in the network to monitor and also for a dynamic network it

increases during the time and different devices with different brands that use in a network have their unique information can be a big challenge for any network admin and also system monitoring.

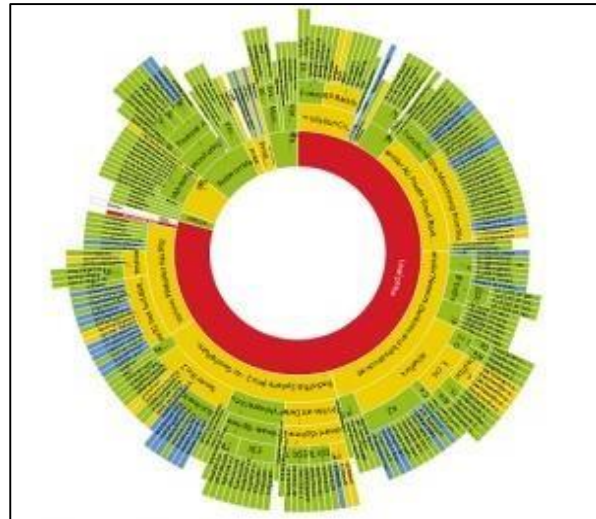


Figure 13-Network Map

The first step to create a proper map is to add all the devices and also their useful information and their criteria, then, move the items around to get a striking disposition but in this project there was already an enormous map consisting of all the devices in the city of Ivrea in the dude monitoring system.

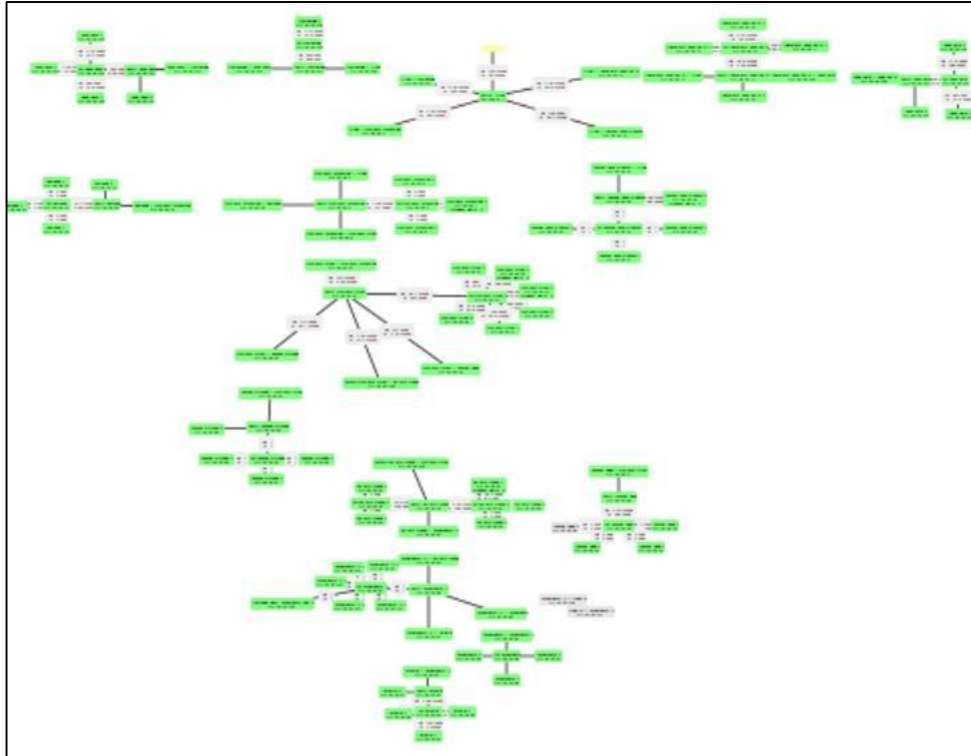


Figure 14-The Dude Map in Ivrea City

There was a list of all devices and information from a database (PHPmy admin) in this city that exports in both xml and csv format. The information in the file included the id of each device, the name, the IP address and finally the father ID. One way was using a Script of converting output files(XML) from Dude (mikrotik) to Zabbix Monitoring software based on the python language.

The request information is putting in XML file in input folder and change filename to name of your XML Result: when procedure is finished, find zabbix file export in OUTPUT folder. As a result, all the devices exported from the dude has a correct replica with information that can be imported in a right format to zabbix but unfortunately, for the map there was no way to build a map with that size and information so I use NetworkX, a python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex network structures.

For using this library, the first outcome that needed for an exported file is;

- ◀ IP address
- ◀ System name
- ◀ SysObjectID
- ◀ Neighbors

There was a fundamental problem concerning creating the first csv file to continue as the input for the first code. For design and creation, I needed one more row of information related to each device's neighbors and not just father but also all the children and it could be impossible to build manually. therefore, due to devices order in the database as a tree, it could be possible to provide a file with each host neighbors inside based on a python code so each device father is known and the first father in each city is the internet but fathers are similar between children and it is possible that several routers or switches have the same father but eventually reach to all neighbors IP addresses for every single device was possible, also there was some problems related to devices which were in the border of two cities or was the bridge of two cities that solved. Then, after creation of the proper file with data, what I can do is write some Python lines that can read this file, identify all the required information, and write in the output a DOT file. The reason to use a middle visualization format is between different vendors and monitoring systems, there is a need for a standard language and format so it can quickly check the difference between versions and it can be easy to maintain and normalize in a common language.

Graphviz DOT file is an easy to read, maintain and update that can store in different formats from Graphviz and basically is a description language, which is an open source graph visualization software and the filename extension is dot. It has several main graph layout programs and packages. For installing the package of language the easiest way is using Ubuntu distribution of linux to use apt package manager.

sudo apt install Graphviz

And then write a file based on information from devices in the dude. The DOT language is a language made to represent objects connected between each other. For instance, for connection between three routers in Ivrea City;

graph {

Casa Bodo 1[hostname="172.16.16.124"] Casa

Bodo 2 [hostname="172.16.16.125"] router

[label="RB Casa Bodo" zbximage="router"]

}

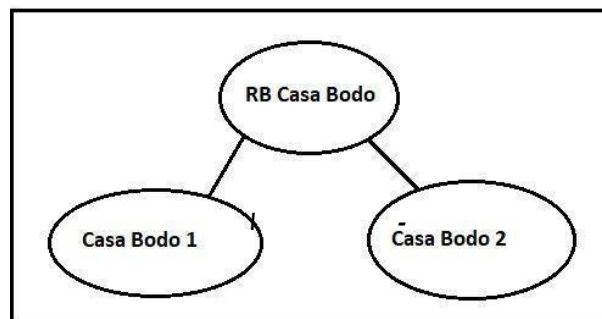


Figure 15-GRAPHVIZ Layout

After export the file with the require information that is needed to create the complete map with the help of networkx library in python to get the desire output, the next step is drafting zabbix map from DOT file in several step;

1. Read out the DOT file from the last step.
2. Create the topology of the given network using Graphviz.
3. Obtain all the coordinates from our topology generated.
4. Use pyzabbix to connect to our Zabbix server.
5. Generate our topology in a fully automated way.

And then with the use of python and networkx library and also import the zabbixAPI from pyzabbix, it can define a relative and generated map in zabbix. [17] [15]

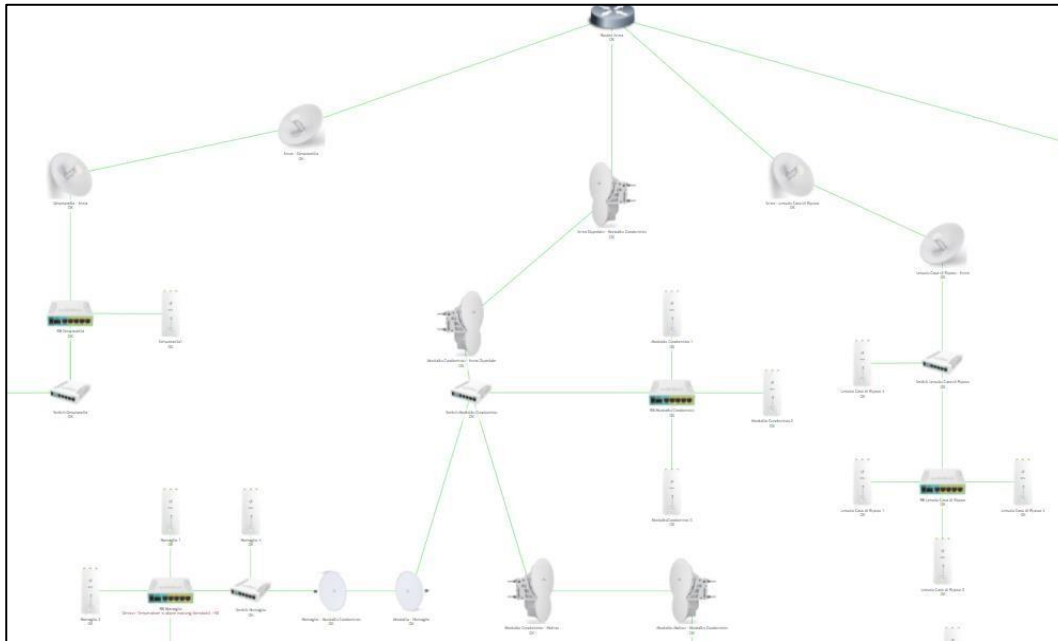


Figure 16-Final Map in ZABBIX Format

4.4 Telegram

Zabbix provides a complete workflow: sending notifications, allowing acknowledgement of information received, escalation of information to other people, and ability to take actions.

There are different methods for delivery from email, SMS, custom alert script and most importantly webhook, Furthermore, notifications can be scripted. Notification content is completely customizable depending on the context. Each contact can be notified for specified levels using specified media at specified days and times.

In this project the main focus for alert is on Telegram via webhook, a cloudbased instant messaging and voice over IP service., to receive notifications on telegram first need to create a new Telegram bot in BotFather.

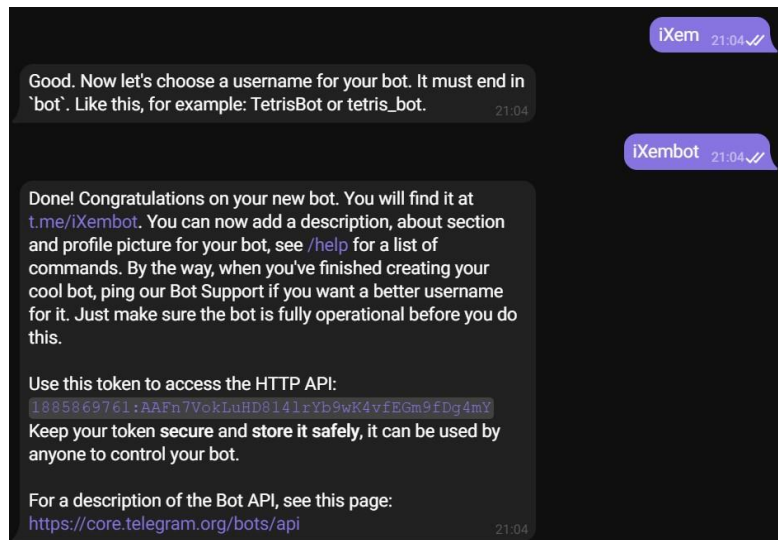


Figure 17-Telegram Bot Father

Check the availability of Bot by using the HTTP API in the web browser in api.telegram.org;

```
{"ok":true,"result":[]}
```

```
{"ok":true,"result":[{"update_id":467979619,
"message":{"message_id":4,"from":{"id":109585074,"is_bot":false,"first_name":"S@mi
Sgh","username":"samansgh7","language_code":"en"},"chat":{"id":109585074,
"first_name":"S@mi
Sgh","username":"samansgh7","type":"private"},"date":1631882658,"text":"Te st"}}
```

Use the “id” in Administration, Media types and add id as the receiver and put the token as the same token in BotFather and test the Telegram Bot;

Name	Value	Action
Message	{ALERT.MESSAGE}	Remove
ParseMode		Remove
Subject	{ALERT.SUBJECT}	Remove
To	109585074	Remove
Token	1885869761:AAF7VokLuHD814I	Remove

[Add](#)

Figure 18-Telegram Media Type

Then in Administration>User, in media type, the type of media, the receiver, the time and the severity can be selected;

Type: Telegram

* Send to: 109585074

* When active: 1-7,00:00-24:00

Use if severity:

- ☐ Not classified
- ☒ Information
- ☒ Warning
- ☒ Average
- ☒ High
- ☐ Disaster

Enabled: ☒

[Add](#) [Cancel](#)

Figure 19-Detailed Telegram Message in Media Type

And in Configuration → Action → Trigger Action → Report problems to Zabbix Administrator, An operation with steps, duration and receivers and custom message can be created; [18]

Operation details

Operation

Send message

Steps

1

-

10

(0 - infinitely)

Step duration

1

(0 - use action default)

* At least one user or user group must be selected.

Send to user groups

User group

Zabbix administrators

Add

Action

[Remove](#)

Send to users

User

Admin (Zabbix Administrator)

Add

Action

[Remove](#)

Send only to

Telegram

▼

Custom message

☐

Update

Figure 20-ZABBIX Operation For Telegram Alert

Chapter 5

5 Result

5.1 Ivrea city

As mentioned before, Ivrea city is the first subject in this project, it has 120 hosts containing sixteen router-boards, ten switches and the rest are ubiquiti with activated SNMP and added to Zabbix with their specific names. As all the groups of devices added to Zabbix by discovery and sorted in host and start to check the critical issues by SNMP protocol.

There was a concerning problem after adding all the devices to Zabbix due to lack of memory that constantly stopped Zabbix and need to reboot but after worked on all the process and confirm the ability for automation, memory was upgraded to 8 Giga byte for the rest cities in the network to add.

5.2 Full Transform

Casa Blotto - Casa Borio	172.16.16.128:10050	ZBX SNMP	Enabled
Casa Blotto - Ivrea	172.16.16.72:10050	ZBX SNMP	Enabled
CasaBlotto1	172.16.16.75:10050	ZBX SNMP	Enabled
CasaBlotto2	172.16.16.76:10050	ZBX SNMP	Enabled
CasaBlotto3	172.16.16.77:10050	ZBX SNMP	Enabled
CasaBlotto4	172.16.16.78:10050	ZBX SNMP	Enabled
Casa Bodo - Genzianella	172.16.16.121:10050	ZBX SNMP	Enabled

Figure 21-Ivrea City Devices Added to ZABBIX by Discovery

<input type="checkbox"/> Casa Blotto - Ivrea	CPU utilization ?	10/30/2021 11:30:3...	3 %	-1 %
<input type="checkbox"/> Casa Blotto - Ivrea	Free memory ?	10/30/2021 11:30:3...	23.89 MB	+72 KB
<input type="checkbox"/> Casa Blotto - Ivrea	ICMP loss	10/30/2021 11:30:2...	0 %	
<input type="checkbox"/> Casa Blotto - Ivrea	ICMP ping	10/30/2021 11:30:2...	Up (1)	
<input type="checkbox"/> Casa Blotto - Ivrea	ICMP response time	10/30/2021 11:30:2...	20.57ms	-17.9ms
<input type="checkbox"/> Casa Blotto - Ivrea	Interface airview1: Bits received ?	10/30/2021 11:28:2...	0 bps	
<input type="checkbox"/> Casa Blotto - Ivrea	Interface airview1: Bits sent ?	10/30/2021 11:28:2...	0 bps	
<input type="checkbox"/> Casa Blotto - Ivrea	Interface airview1: Inbound packets discarded ?	10/30/2021 11:28:2...	0	

Figure 22-Latest Data in Casa Blotto-Ivrea Device

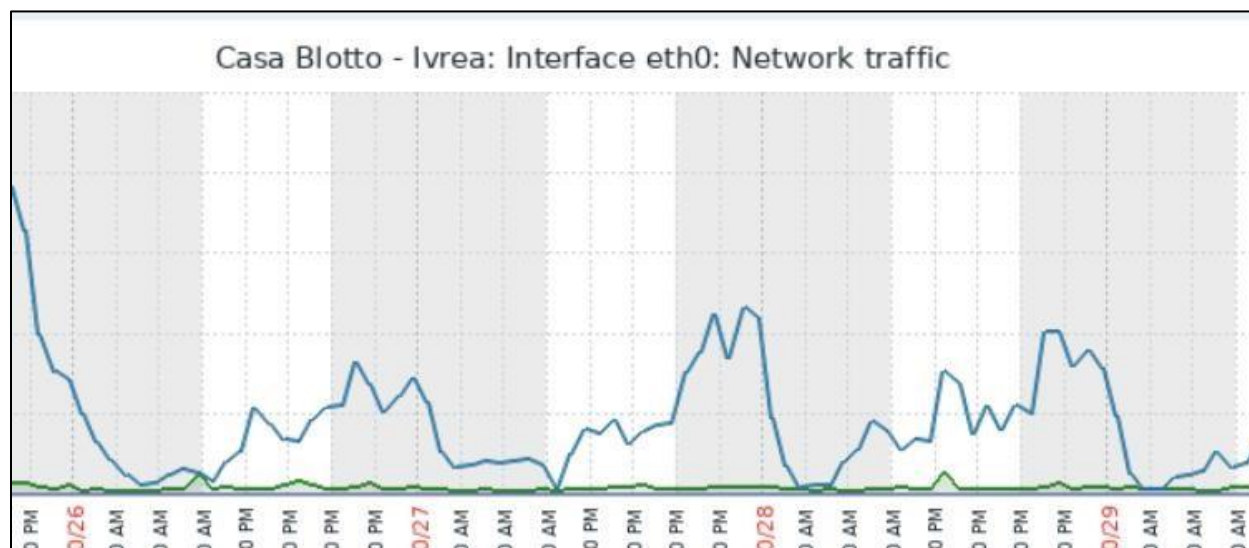


Figure 23-Graph of Network Traffic in Casa-Blotto Device in Ivrea City

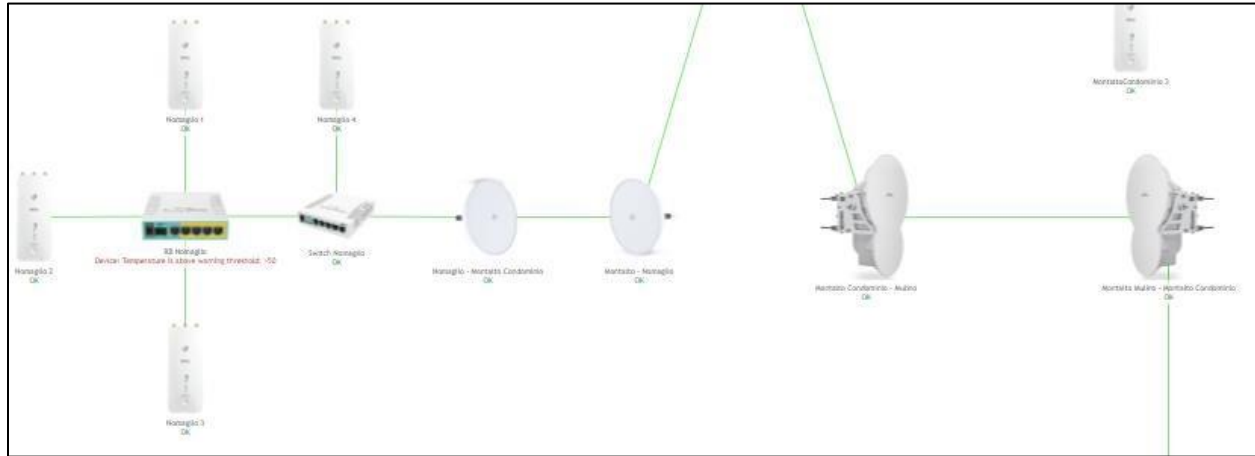


Figure 24- Map Creation in Ivrea City

Rest of the network was include 2000 devices from different areas and two of the were combine in The Dude map.

For the full transform the main part was to first activate all the required protocols in automation way and fix all the concerning error, then add all the devices to host section by discovery and create the separated maps.

In Figure21 , it can be seen the several added devices with their IP and active SNMP in host section. Casa Blotto was chosen for more details in figure (22) as a single host to monitor. The information includes CPU utilization, free memory, ICMP or ping or loss at the time, respond time of ICMP, the state of interfaces, the traffic passing in real time and the graphs that is possible to observe in figure (23) in just one interface in the same host.

Also all the devices added to the specific maps base on the city network category and it was possible to upload the exact photo of device based on the vendor and model into the map section to show the clear and good-looking visualization of the network. Figure (24)

Chapter 6

6 Conclusion

After all configuration as a result, zabbix has real-time statistics compared to the dude so in real time it is possible to monitor millions of metrics collected from tens of thousands of servers, virtual machines and network devices. There is a substantial difference in terms of network tools in Zabbix and the dude because there are no network tools in The Dude also uptime monitoring is very important concerning availability of service provided by iXem project and identifying weak places of IT infrastructure, there is a availability of website monitoring in Zabbix. At the end two big difference in term of management of the project infrastructure is log management in every situation and also tracking the errors when it happens so in many non-identical ways.

In many ways, when there is a better financial situation or bigger network as a business, as discussed before Zabbix can improve quickly but certainly with The Dude there is no recommended option to deploy.

It can be said network monitoring is taken to the next level through Zabbix as a network monitoring system in this project and further developments.

Bibliography

- [1] "dade2.net," [Online]. Available: <https://dade2.net/kb/how-to-know-what-the-zabbix-server-version/>.
- [2] "docs.microsoft.com," [Online]. Available: <https://docs.microsoft.com/en-us/windows/win32/wmisdk/wmi-start-page>.
- [3] "en.wikipedia.org," [Online]. Available: https://en.wikipedia.org/wiki/Simple_Network_Management_Protocol.
- [4] A. D. V. S. K. Lee, Zabbix Network Monitoring, Packt Publishing Ltd., 2015.
- [5] "Zabbix.com," [Online].
- [6] "whatsupgold.com," [Online]. Available: <https://www.whatsupgold.com/what-is-network-monitoring>.
- [7] "techtarget.com," [Online]. Available: <https://www.techtarget.com/searchnetworking/definition/OSI>.
- [8] "solarwinds.com," [Online]. Available: <https://www.solarwinds.com/>.
- [9] "site24x7.com," [Online]. Available: <https://www.site24x7.com/network-monitoring.html>.
- [1] "prtg.com," [Online]. Available: https://www.paessler.com/prtg?gclid=Cj0KCQjw_fiLBhDOARIsAF4khR16kq6vcby0IzK5YwYmEAetaASa8sflidLxefEyyaeLEDP2kRxa954aAh-oEALw_wcB.
- [1] "obkio.com," [Online]. Available: <https://obkio.com/blog/top-7-reasons-why-you-should-monitor-1-network-performance/>.
- [1] "mikrotik.com," [Online]. Available: <https://mikrotik.com/thedude>.
- 2]
- [1] "medium.com," [Online]. Available: <https://medium.com/@keagileageek/paramiko-how-to-ssh-3-and-file-transfers-with-python-75766179de73>.
- [1] "iopscience.iop.org," [Online]. Available: <https://iopscience.iop.org/article/10.1088/1755-4131/512/1/012155/pdf>.
- 4]
- [1] "graphviz.org," [Online]. Available: <https://graphviz.org/>.
- 5]

[1 "extrashop.com," [Online]. Available: <https://www.extrahop.com/resources/protocols/icmp/>.
6]