

NETWORK MANAGEMENT SYSTEM

Done By: Sophil Sthapit

This report contains steps on how I created a Network Management System
during my internship at **Connect Technology Pvt Ltd.**

Contents

1. Introduction
 - a. Purpose of the Report
 - b. Scope of the Report
 - c. Requirements for the Project
2. Network Design
 - a. Packet Tracer Planning
 - b. Devices in the Network
 - c. IP Addressing Scheme
3. Configuration Steps
 - a. Initial Setup
 - b. Router-to-Router Configuration
 - c. Switch Configuration
 - d. SNMP Configuration
 - e. Network Management System (NMS) Setup
 - i. Setting up Zabbix Server in Raspberry Pi
 - ii. Setting up Zabbix Agent in Host devices (Ubuntu and Windows PC)
 - f. Alert and Notification

Introduction

Purpose of Report

The purpose of this report is to document the steps involved in setting up a working Network Management System. The steps will include designing, configuring and testing a simulated network using Cisco Packet Tracer. After proper testing and evaluation, the project will be implemented using real hardware for practical deployment.

Scope of the Report

- a. Designing the network topology
- b. Configuring devices and routing protocols
- c. Setting up SNMP for network monitoring
- d. Testing connectivity and performance in Cisco Packet Tracer
- e. Transitioning the configuration to real hardware

Requirements for the project

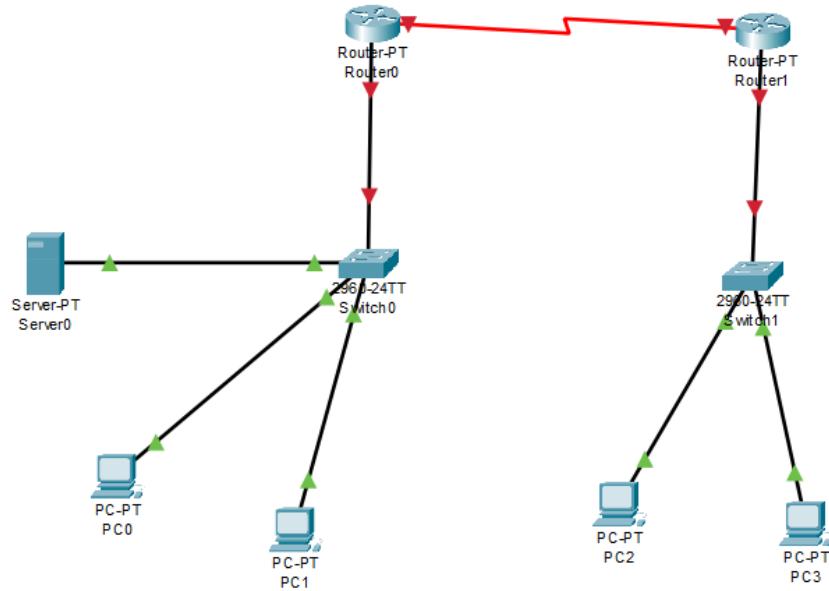
- a. Hardware Requirements
 - i. Server Machine – 1 PC or Raspberry Pi
 - ii. Network Devices –
 - 1. 2 Routers (TP-Link Router with SNMP enabled)
 - 2. 2 Switch (TP-Link TL-SG108E)
 - 3. 4 PCs (Can be done using VMs) (Laptop, office PC, 2 VMs)
 - iii. Internet Connection – Stable LAN/ Wi-Fi network
 - 1. Connect Router0's WAN port for external access
 - 2. Ethernet or Serial Cable
- b. Software Requirements
 - i. Operating System – Ubuntu Server for stable NMS
 - ii. Monitoring Tools – Zabbix or Nagios or PRTG
 - iii. Network Scanning – Nmap, Netdiscover

- iv. Traffic Analysis – Wireshark, Suricata
 - v. SNMP Tools – Snmpd, Snmpwalk
 - vi. Visualization – Grafana + InfluxDB
 - vii. Automation – Python Scripts (eg. Paramiko, psutil, requests)
-
- c. NMS Requirements
 - i. Components
 - i. Databases (eg. MySQL)
 - ii. Monitoring Software (e.g Zabbix, Nagios, OpenNMS)
 - iii. Web interface/ dashboard for visualization

Network Design

Cisco Packet Tracer Planning

Network Topology:



Devices in the Network:

Routers – The routers are connected to 2 network each, network 1 with the server and network 2 without the server. They are also connected to each other via wired serial communication for connectivity.

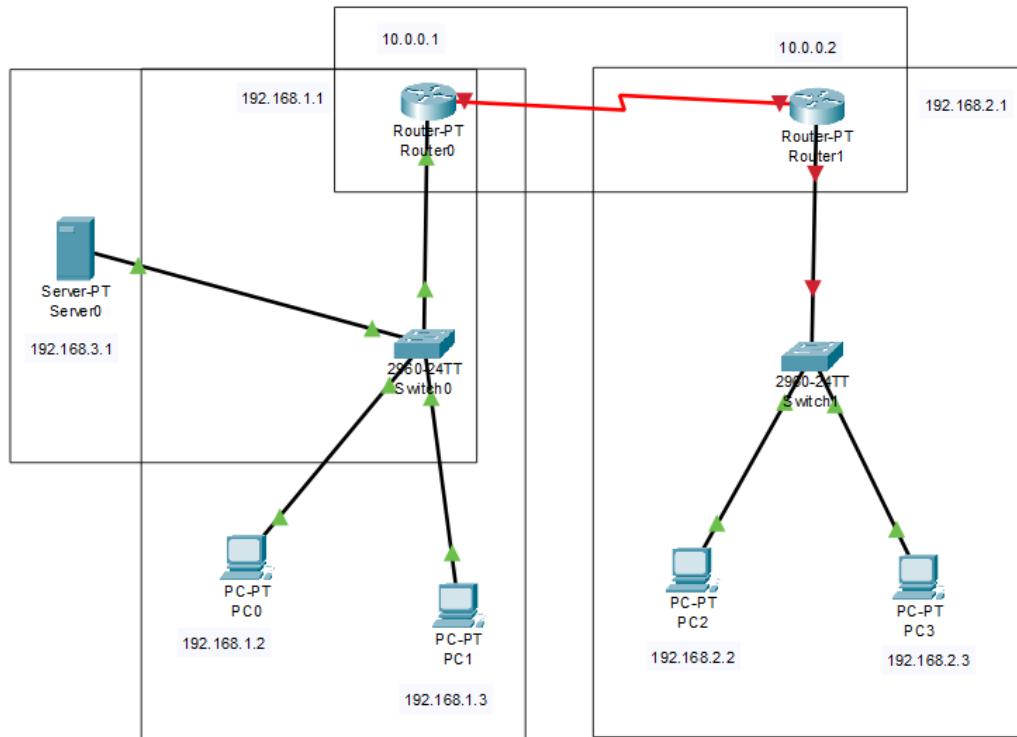
Switch – There are two switches for each network that connects to all the devices for the networks. They connect the devices to the routers.

Server – This is used to host the NMS; this device is used to run a monitoring software for the network.

PCs – These devices are employee devices within the network.

IP Addressing Scheme:

- Router-0: 192.168.1.1/24
- Router-1: 192.168.2.1/24
- PCs:
 - PC0: 192.168.1.2
 - PC1: 192.168.1.3
 - PC2: 192.168.2.2
 - PC3: 192.168.2.3
- NMS Server: 192.168.3.1



Configuration Steps

Initial Setup

Basic Device Configuration, Assigning static IP addresses to routers, PCs and the NMS server

Router-0 to switch (network 1):

Commands:

```
> Enable  
> Configure terminal  
> Interface FastEthernet0/0  
> Ip address 192.168.1.1 255.255.255.0  
> No shutdown
```

Router-1 to switch (network 2):

Commands:

```
> Enable  
> Configure terminal  
> Interface FastEthernet0/0  
> Ip address 192.168.2.1 255.255.255.0  
> No shutdown
```

Router-to-router Configuration

Router-0 Commands:

```
> Enable  
> Configure terminal  
> Interface Serial2/0  
> Ip address 10.0.0.1 255.255.255.252  
> no shutdown
```

Router-1 Commands:

```
> Enable  
> configure terminal  
> Interface Serial2/0
```

```
> Ip address 10.0.0.2 255.255.255.252  
> no shutdown
```

Routing Configuration:

Router-0:

```
> ip route 192.168.2.0 255.255.255.0 10.0.0.2
```

Router 1:

```
> ip route 192.168.1.0 255.255.255.0 10.0.0.1
```

```

Processor board ID PT0123 (0123)
PT2005 processor: part number 0, mask 01
Bridging software.
X.25 software, Version 3.0.0.
4 FastEthernet/IEEE 802.3 interface(s)
2 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)

--- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]:
Press RETURN to get started!

Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface Serial2/0
Router(config-if)#ip address 10.0.0.2 255.255.255.252
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

```

Top

Final Verification

Command:

```
> show ip interface brief
```

```

Router#show ip interface brief
Interface          IP-Address      OK? Method Status        Protocol
FastEthernet0/0    192.168.1.1    YES manual up          up
FastEthernet1/0    unassigned     YES unset administratively down down
Serial2/0          10.0.0.1       YES manual up          up
Serial3/0          unassigned     YES unset administratively down down
FastEthernet4/0    unassigned     YES unset administratively down down
FastEthernet5/0    unassigned     YES unset administratively down down

```

Router-0 interface brief

```
Router#show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    192.168.2.1    YES manual up       up
FastEthernet1/0    unassigned     YES unset administratively down down
Serial2/0          10.0.0.2      YES manual up       up
Serial3/0          unassigned     YES unset administratively down down
FastEthernet4/0    unassigned     YES unset administratively down down
FastEthernet5/0    unassigned     YES unset administratively down down
Router#
```

Router-1 interface brief

```
Router#ping 10.0.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/6/8 ms
```

Testing Connectivity from Router-0 to Router-1

```
Router#ping 10.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/7/10 ms
```

Testing Connectivity from Router-1 to Router-0

Setting up Zabbix Server in Raspberry Pi

Source: <https://www.zabbix.com/download>

Commands:

1. Choosing the platform

Zabbix Version	OS Distribution	OS Version	Zabbix Component	Database	Web Server
7.4	Debian	12 Bookworm (amd64, arm64)	Server, Frontend, Agent	MySQL	Apache

2. Install and configure Zabbix

Commands:

```
> wget https://repo.zabbix.com/zabbix/7.4/release/debian/pool/main/z/zabbix-release/zabbix-release_latest_7.4+debian12_all.deb  
> sudo dpkg -i zabbix-release_latest_7.4+debian12_all.deb  
> sudo apt update
```

3. Installing Zabbix server, frontend and agent

Command:

```
> Sudo apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent
```

4. Create initial database

Commands:

```
> mysql -u root -p
```

```
“password”
mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;
mysql> create user zabbix@localhost identified by 'password';
mysql> grant all privileges on zabbix.* to zabbix@localhost;
mysql> set global log_bin_trust_function_creators = 1;
mysql> quit
```

Run this command next:

```
> zcat /usr/share/zabbix/sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -uzabbix -p zabbix
```

5. Disable log_bin_trust_function_creators option after importing database schema

```
> mysql -uroot -p
“password”
mysql> set global log_bin_trust_function_creators = 0;
mysql> quit;
```

6. Configure the database for Zabbix server

Edit file /etc/zabbix/zabbix_server.conf

```
> DBPassword=password
```

7. Start Zabbix server and agent processes

```
> systemctl restart zabbix-server zabbix-agent apache2
> systemctl enable zabbix-server zabbix-agent apache2
```

Open: <http://192.168.11.141/zabbix>

After performing initial setup, we will be welcomed with this home page

The screenshot shows the Zabbix Global view dashboard. On the left, there is a sidebar with various navigation links: Dashboards, Monitoring, Services, Inventory, Reports, Data collection, Alerts, Users, Administration, Support, Integrations, Help, User settings, and Sign out. The main content area is titled "Global view". It features several widgets:

- Top hosts by CPU utilization:** A table showing host utilization over different time frames (1m avg, 5m avg, 15m avg) and processes.
- No Zabbix server Values per second:** A large red box indicating that no data is being received from the Zabbix server.
- System information:** A table with details about the Zabbix server, including version and number of hosts, templates, and items.
- Host availability:** A chart showing the count of hosts in different states: Available (0), Not available (0), Mixed (0), Unknown (2), and Total (2).
- Problems by severity:** A chart showing the count of problems by severity: Disaster (0), High (0), Average (0), Warning (0), Information (0), and Not classified (0).
- Current problems:** A table with columns for Time, Info, Host, Problem & Severity, Duration, Update, Actions, and Tags. It currently displays 0 rows.
- Geomap:** A map of Riga, Latvia, showing various monitoring points and geographical features.

Setting Up Zabbix Agent in Ubuntu and Windows Desktops

UBUNTU DESKTOP

To setup zabbix agent in ubuntu desktop we have to install zabbix package first

Commands:

```
> wget https://repo.zabbix.com/zabbix/6.0/ubuntu/pool/main/z/zabbix-agent/zabbix-
agent_6.0.17-1+ubuntu20.04_amd64.deb
> sudo dpkg -i zabbix-agent_6.0.17-1+ubuntu20.04_amd64.deb
```

After downloading zabbix, we need to configure the file to specify zabbix server

```
> sudo nano /etc/zabbix/zabbix_agentd.conf
```

In the file search for the following:

```
> Server=192.168.11.141 #this is my zabbix server IP address
> ServerActive=192.168.11.141
> Hostname=Ubuntu-Desktop #your device name
> ListenPort=10050
```

After the configuration is done, start Zabbix Agent

```
> sudo systemctl start zabbix-agent
```

Check its status

```
> sudo systemctl status zabbix-agent
```

```

● zabbix-agent.service - Zabbix Agent
   Loaded: loaded (/usr/lib/systemd/system/zabbix-agent.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-10-13 16:27:29 +0545; 22h ago
     Process: 1243 ExecStart=/usr/sbin/zabbix_agentd -c $CONFFILE (code=exited, status=0/SUCCESS)
    Main PID: 1254 (zabbix_agentd)
      Tasks: 13 (limit: 18670)
     Memory: 9.6M (peak: 11.6M swap: 420.0K swap peak: 444.0K)
        CPU: 1min 54.049s
      CGroup: /system.slice/zabbix-agent.service
              └─1254 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf
                  ├─1257 "/usr/sbin/zabbix_agentd: collector [idle 1 sec]"
                  ├─1259 "/usr/sbin/zabbix_agentd: listener #1 [waiting for connection]"
                  ├─1260 "/usr/sbin/zabbix_agentd: listener #2 [waiting for connection]"
                  ├─1262 "/usr/sbin/zabbix_agentd: listener #3 [waiting for connection]"
                  ├─1263 "/usr/sbin/zabbix_agentd: listener #4 [waiting for connection]"
                  ├─1265 "/usr/sbin/zabbix_agentd: listener #5 [waiting for connection]"
                  ├─1266 "/usr/sbin/zabbix_agentd: listener #6 [waiting for connection]"
                  ├─1268 "/usr/sbin/zabbix_agentd: listener #7 [waiting for connection]"
                  ├─1269 "/usr/sbin/zabbix_agentd: listener #8 [waiting for connection]"
                  ├─1270 "/usr/sbin/zabbix_agentd: listener #9 [waiting for connection]"
                  ├─1271 "/usr/sbin/zabbix_agentd: listener #10 [waiting for connection]"
                  └─1272 "/usr/sbin/zabbix_agentd: active checks #1 [idle 1 sec]"

Oct 13 16:27:28 connect-ThinkPad-T490 systemd[1]: Starting zabbix-agent.service - Zabbix Agent...
Oct 13 16:27:29 connect-ThinkPad-T490 systemd[1]: Started zabbix-agent.service - Zabbix Agent.
~
~
~
```

After that go to your Zabbix server and add host

Fill in the details as per the device

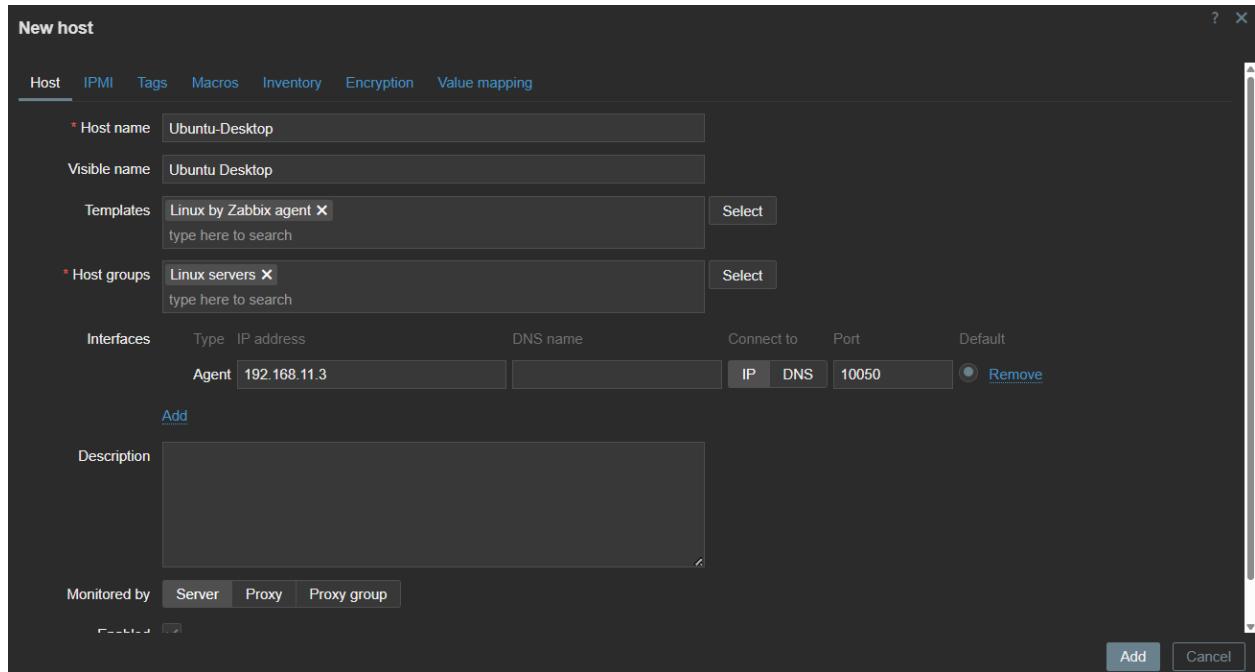
Hostname: Ubuntu-Desktop

Visible name: Ubuntu Desktop

Template: Operating system Linux by Zabbix Agent

Host groups: Linux Server

Interface: Agent>192.168.11.3 port:10050



WINDOWS DESKTOP

Download Zabbix-Agent setup from the source "https://www.zabbix.com/download_agents"

After downloading the installer, open and configure it.

Hostname: Windows-Desktop

Zabbix Server IP/DNS: 192.168.11.141

Port: 10050

Check its status

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> Get-Service -Name "Zabbix Agent"

Status    Name            DisplayName
-----   --          -----
Running  Zabbix Agent      Zabbix Agent

PS C:\Windows\system32>
```

After that go to your Zabbix server and add host

Fill in the details as per the device

Hostname: Windows-Desktop

Visible name: Windows Desktop

Template: Operating system Windows by Zabbix Agent

Host groups: Windows Server

Interface: Agent>192.168.11.2 port:10050

Host

- [Host](#) [IPMI](#) [Tags](#) [Macros](#) [Inventory](#) [Encryption](#) [Value mapping](#)

* Host name	Windows-Desktop				
Visible name	Windows Desktop				
Templates	Name	Action			
	Windows by Zabbix agent	Unlink Unlink and clear			
	type here to search	Select			
* Host groups	Windows Servers X	Select			
	type here to search				
Interfaces	Type IP address	DNS name	Connect to	Port	Default
	Agent 192.168.11.2		IP	DNS	10050
Add					
Description					
Monitored by	Server	Proxy	Proxy group		

[Update](#) [Clone](#) [Delete](#) [Cancel](#)

After adding both the devices to the server it will look like this:

ZABBIX

[Hosts](#)

Name	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status	Availability	Agent encryption	Info	Tags
NAS	Items 121	Triggers 15	Graphs 8	Discovery 3	Web	192.168.11.145.623			Enabled	ZBX	None		
Ubuntu Desktop	Items 43	Triggers 15	Graphs 8	Discovery 3	Web	192.168.11.3.10050			Enabled	ZBX	None		
Windows Desktop	Items 34	Triggers 13	Graphs 5	Discovery 4	Web	192.168.11.2.10050			Enabled	ZBX	None		
Zabbix server	Items 121	Triggers 68	Graphs 8	Discovery 6	Web	127.0.0.1:10050			Enabled	ZBX	None		

Displaying 4 of 4 found

0 selected [Enable](#) [Disable](#) [Export](#) [Mass update](#) [Delete](#)

Alerts

Users

Administration

! Zabbix server is not running: the information displayed may not be current.

Alert and notification

1. Create trigger

Parent triggers [Windows by Zabbix agent](#)

* Name

Event name

Operational data

Severity Not classified Information Warning Average High Disaster

* Expression

Expression constructor

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Allow manual close

URL

Description

Enabled

Based on the trigger set action. In this case we are setting an action for a Telegram message from a bot notifying us about the event.

Go to telegram and search for “BotFather” go to the chat and click on start/

Type command:

```
> /start
```

To start creating a new chat bot, give the chatbot its name and username. In this case:

- Name: zabbixbot
- Username: CTPLzabbix_bot

After that it will give us a API token which we will enter in our zabbix server. When you open the chat page of the bot, in the url you will see an ID, save the ID.

Go to Administration > Media types > Telegram and enter the following

Name: Telegram

Type: Webhook

Message: default

Subject: default

To: <the chatbot ID “for example: -12345”>

Token: <API token given by the chatbot>

Update it.

After that we have to set telegram as a media in our user account.

The screenshot shows the Zabbix 'Users' interface. On the left is a vertical toolbar with icons for Zabbix, search, user creation, user list, user details, user groups, user permissions, and settings. The main area has a dark header with the title 'Users'. Below the header is a navigation bar with tabs: 'User' (selected), 'Media 1' (highlighted in blue), and 'Permissions'. The 'Media' tab displays a table with one row. The table columns are: 'Media' (checkbox), 'Type' (dropdown menu), 'Send to' (text input field), 'When active' (time range input field), 'Use if severity' (color-coded severity legend), 'Status' (checkbox), and 'Action' (button group). The single row in the table has 'Media' checked, 'Type' set to 'Telegram', 'Send to' as a redacted URL, 'When active' as '1-7,00:00-24:00', 'Use if severity' with colors N (blue), I (orange), W (yellow), A (green), H (red), and D (purple), 'Status' checked, and 'Action' with 'Edit' and 'Remove' buttons. At the bottom of the table are 'Update', 'Delete', and 'Cancel' buttons.

Now we are going to set the trigger actions that will allow the server to communicate with the telegram bot, which will allow telegram bot to receive and forward notifications about the server events.

The screenshot shows the Zabbix Operations configuration interface. On the left, there's a main configuration panel with sections for 'Operations', 'Recovery operations', 'Update operations', and checkboxes for 'Pause operations for suppressed problems' and 'Notify about canceled escalations'. The 'Operations' section has a 'Default operation step duration' of '1h' and a list of steps: 'Send message to user groups: Zabbix administrators' and 'Send message to users: CTPLZadmin (Zabbix Admins)'. Below these are 'Add' and 'Details' buttons. On the right, a modal window titled 'Operation details' is open. It contains fields for 'Operation' (set to 'Send message'), 'Steps' (set to '1 - 1 (0 - infinitely)'), 'Step duration' (set to '0 (0 - use action default)'), and a note 'At least one user or user group must be selected.' Below this are sections for 'Send to user groups' (listing 'Zabbix administrators' with an 'Add' button) and 'Send to users' (listing 'User' with an 'Add' button). There are also dropdowns for 'Send only to' (set to '- All -') and a 'Custom message' checkbox. At the bottom of the modal are 'Update' and 'Cancel' buttons.

Testing the Alert and notification

1. I have registered a trigger where if the CPU utilization of one of the linux devices go over 90% for 5 minutes it will notify me through telegram.
2. I will force overload the CPU resource using command:

```
> sudo stress-ng --cpu 0 --timeout 6m
# This will use all CPU recourse of the device for 6 minutes
```

3. The alert message on telegram of when the event occurred, detected and reported:

C CTPL_zabbix_alerts
2 members

Add Members

Zabbixbot

Problem: Linux: High CPU utilization (over 90% for 5m)
Problem started at 16:24:28 on 2025.10.19
Problem name: Linux: High CPU utilization (over 90% for 5m)
Host: testbuntu
Severity: Warning
Operational data: Current utilization: 100 %
Original problem ID: 3367 16:24

Problem: Linux: High CPU utilization (over 90% for 5m)
Problem started at 16:24:28 on 2025.10.19
Problem name: Linux: High CPU utilization (over 90% for 5m)
Host: testbuntu
Severity: Warning
Operational data: Current utilization: 100 %
Original problem ID: 3367 16:24

Z

Resolved in 1m 0s: Linux: High CPU utilization (over 90% for 5m)
Problem has been resolved in 1m 0s at 16:25:28 on 2025.10.19
Problem name: Linux: High CPU utilization (over 90% for 5m)
Host: testbuntu
Severity: Warning
Original problem ID: 3367 16:25

Message

