Business Requirement Document

360º Monitoring project

Prepared by

- Abhay Singh Bajeta
- Omkar Sambhaji Damame
- Siddik Jahangir Attar
- Prasad Mohan Dhupkar
- Sunita Kumari Nayak
- Deepak Panigrahi

Contents

.

1 Introduction

This documentation is report for 360 degree monitoring project. This documentation intent to provide all specific requirements of our project like software requirements, industry requirements, methodology, approach etc.

1.1 Document Purpose

The sole purpose of the document of the document is to help reader to understand the flow of project and provide the overview of project and clearing all ties between the technical and business aspect of the project.

1.2 Product Scope

360 monitoring is a full fledged monitoring provided to the customer server, application, and network monitoring to make sure a transparent process where admin knows what, when and how of its systems and applications.
We as a team will be implementing this project, We will be implementing decentralised monitoring of server application and networking
Our objective will include following steps-
1. Project initiation- We initialize our project by researching and collecting all the information about the project and assign roles to tribe members.
2. Project planning- We decide all the software tools and technology that we will use in the project.
3. Project execution-execution includes installing and deployment of tools according to the decided flow of project.

FUNTIONALITY-

360-Degree Monitoring for Cloud Servers using Nagios-Core

- We are going to use separate EC2 instances just to monitor the the remote Hosts/Servers that we need to monitor.

- Also we are going to setup this as to decentralize this monitoring service in case one goes down.

  Traditionally AWS CloudWatch or services like that are very easy to implement on the infrastructure you have, the reason why we are going with Nagios-Core is that it is an Open Source monitoring tool. It charges nothing, totally free of cost monitoring for all of the instances/servers/appliances you have. Also, it gives accurate statistical information about the modules you have activated

NOTE: We at least need 4 and maximum 5 EC2 instances to implement this on at least of testing level and same amount of Elastic IP addresses to associate them with all the server so you won't have to reconfigure them all the time.

1.3 Total Monitoring Modules:

- PING
- SSH
- HTTP
- Packet Loss
- No. of Users Logged In
- Databases (MySQL in this case)

The next thing is, Hardware Monitoring :

- CPU Load Average
- RAM Usage
- Root Partitions
- Disk Usage
- Total Procs

Then there is Webserver/Application Webserver Monitoring

- Tomcat
- XAMPP
- Any Website
- Servers with particular Ports

GOALS-

- To protect our application from possible failure that would interrupt service availability.
- To demonstrate the use of open source tools efficiently.
- Overcoming the disadvantages of centralised monitoring.

.

BENIFITS-

- **Get complete visibility into your network health and performance:-**
  Server monitoring lets the IT administrator, "look under the hood" and closely examine the issues which could impact network performance. It can look at a variety of variables and show you, in real-time, how your server and workstations are functioning. In this way you can quickly identify any areas that are in need of attention and take appropriate action. Thus, server monitoring allows you to take a proactive approach to problem resolution and to fix any issues before they impact your network's performance

- **Frees up your IT resources:-**
  By taking over many of your server monitoring tasks, this solution allows you to reallocate your IT resources and time to other higher value tasks such as development, or infrastructure renewal. Not only does the software watch over the system, but it can even perform some remedial actions, such as a server reboot or automatically restarting a service.

- **Improve end-user experience:-**
  A fast and responsive application and software monitoring can help you find relative issue.

  - **Reduce downtime:-** Easy troubleshooting hence better availability.

  - **Boost up Innovation:-** Easy bug finds and fixing problems lead to more time for engineers to add new feature in the application.

- **Project Description**:-Through our continuous monitoring tool we had monitor servers, databases and web servers in a decentralized way.We have on configured our servers with two monitoring tool in case one host fails we still can monitor other remote systems.

  We deployed our servers as Linux instances on AWS through EC2 service

1.4 Definitions, Acronyms and Abbreviations

- **Open Source**-Open source is a term that originally referred to open source software (OSS). Open source software is code that is designed to be publicly accessible—anyone can see, modify, and distribute the code as they see fit.
- Open source software is developed in a decentralized and collaborative way, relying on peer review and community production. Open source software is often cheaper, more flexible, and has more longevity than its proprietary peers because it is developed by communities rather than a single author or company.
- **Cloud Computing-** The practice of using a network of remote servers hosted on the internet to store, manage, and process data, rather than a local server or a personal computer.

- **Server monitoring-** Server monitoring is the process of gaining visibility into the activity on your servers — whether physical or virtual. Servers are devices (or increasingly, applications) that store and process information that is provided to other devices, applications or users on-demand.

- **Network monitoring-** Network monitoring provides the information that network administrators need to determine, in real time, whether a network is running optimally.

- **AWS**- Amazon Web Services(cloud service provider)

- **EC2-** Elastic Compute Cloud(complete service of AWS that provides the facilities of creating instances and all compute services related to it.

- **Nagios-** Open source monitoring tool.

- **Zabbix-** Open source monitoring tool

1.5 Challenges Faced Throughout

Challenges

- Setting up mailing service for Nagios
- Setting up NRPE
- Could not assign Elastic IPs to all our 7 instances.(as AWS only allows 5)
- Had to face compatibility issues with OS

How we handled them

- Did it through trial and error and official documentation.
- Gone through various CLI logs.
- Had to contact to AWS support team to request quota increase.
- Tried on multiple OS to **troubleshoot.**

1.6 Flow of The Project

- Ubuntu 20.04 LTS CLI is required on EC2 instance.
- Assigned IAM roles for collaborative work.
- Assigned Elastic IP to all the instances.
- Installed Nagios on host and remote.
- Configured Plugins and NRPE on every instance.
- Edited config file of Nagios and NRPE to add monitoring according to our need.
- Deployed Webserver to monitor it.
- Initiated MySQL database to monitor it.
- Did Set up Postfix mailing service for Nagios; for alerting the Admin incase of any fault.
- Installed another Open Source monitoring tool Zabbix to make it decentralized monitoring.
- Configured Zabbix on remote and host.
- Interlinked Zabbix and Nagios in order to if one goes down, others will inform the Admin.

1.7   Schedule and Timeline:-

As per the schedule provided, we intend to come up with the working prototype by the end of 6 weeks. The following timeline will be followed by us to be on top of the project and get it delivered by 15th May 2022.