# Отчет по домашнему заданию № 11 – Мониторинг и алертинг

В результате выполнения ДЗ №11 был организован мониторинг сервиса диалогов.

В данном задании тренируются навыки:

* эксплутация prometheus;
* эксплутация grafana;
* эксплутация zabbix.

В процессе выполнения ДЗ был создан файл docker-compose, который запускал необходимые для мониторинга и работы приложения сервисы:

* postgres – для сохранения данных Zabbix
* zabbix-server – собственно сервер Zabbix
* zabbix-web – web-интерфейс к серверу Zabbix
* zabbix-agent – агент сбора инфраструктурных данных Zabbix
* prometheus – сервис приложения Prometheus
* grafana – сервис приложения Grafana
* postgres\_db – база данных для приложения SocialNet
* redis – БД кэшей приложения SocialNet

Была проведена соответствующая настройка сервисов и создан дашборд в приложении Grafana, который объединял данные из Zabbix и Prometheus. Для настройки приложения Prometheus был создан файл Prometheus.yml с необходимыми для сбора данных джобами (jobs)

Для мониторинга Zabbix были выбраны инфраструктурные метрики:

* CPU load
* Memory usage
* Disk usage

Для мониторинга сервиса диалогов из Prometheus был выбран метод RED, где в качестве метрик были:

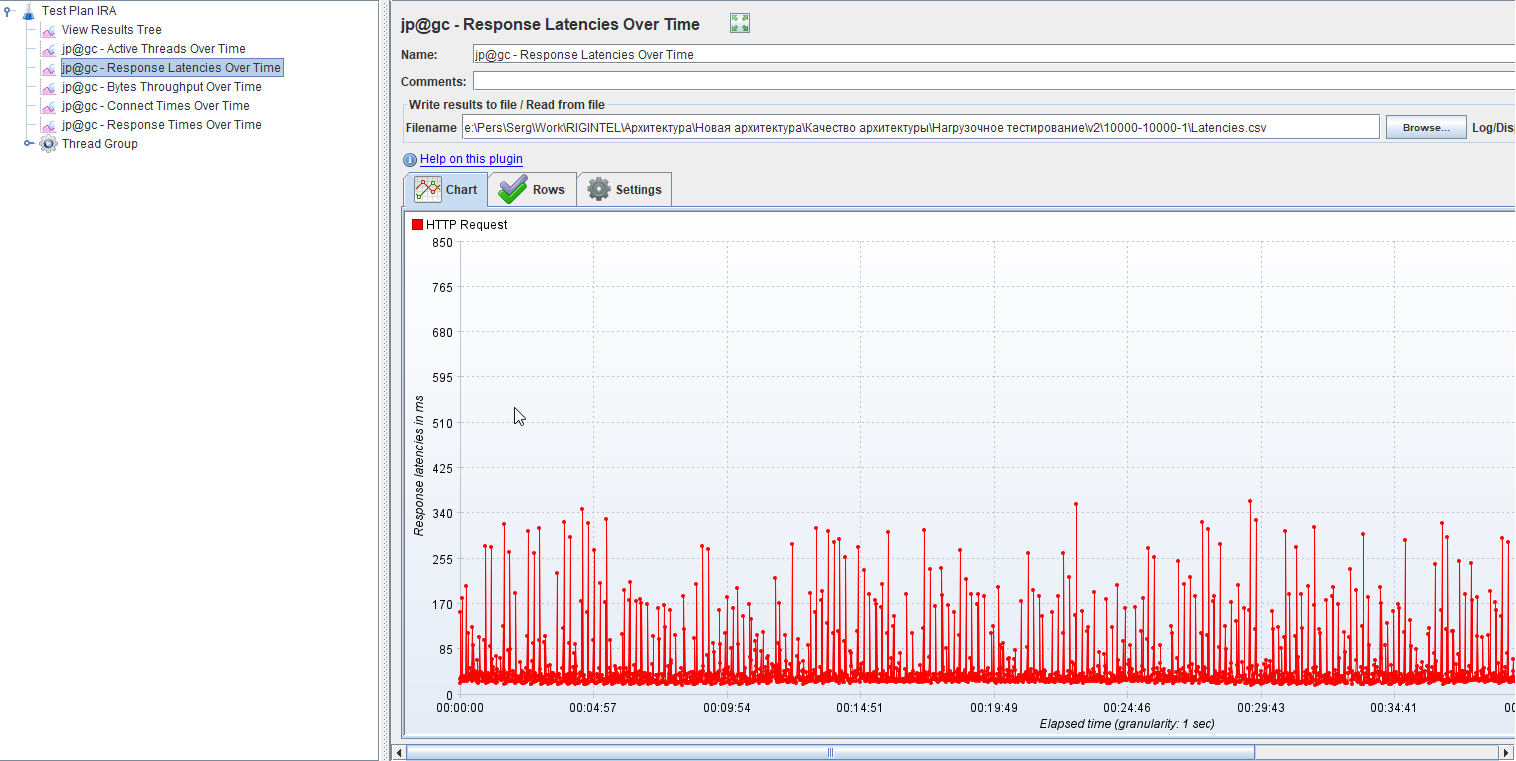
* Requests per Second (RPS)
* Errors per Second
* 95th Percentile Latency

Также было переделано приложение SocialNet с добавлением возможности мониторинга Prometheus с помощью библиотеки prometheus\_client

Для создания нагрузки использовалось приложение Jmeter, которое имитировало запись диалогов, поэтому появлялись данные по метрикам.

Ниже показаны скриншоты приложений Grafana и Jmeter на небольшой нагрузке. Можно отметить, что огибающие графиков Latence в двух приложениях коррелируют между собой. Также можно видеть, что другие метрики тоже собираются и отображаются.





Другие скриншоты с нагрузкой побольше:

