

# Процедура мониторинга сетевых сегментов тракта передачи данных ИС

## Постановка задач

Произошел инцидент первого приоритета в АИС Портал Поставщиков №2318370 в СА ITSM – “Сбой в работе портала [zakupki.mos.ru](https://zakupki.mos.ru)”. Описание инцидента приложено в файле [2024\\_02\\_14\\_Отчёт\\_о\\_критическом\\_инциденте\\_2318370\\_.dot+](#)

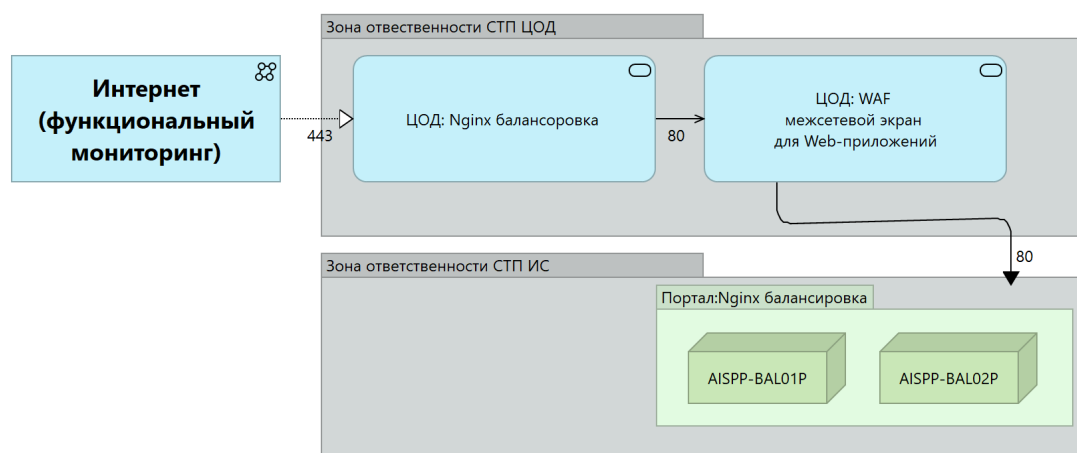
Более подробная диагностика [zakupki.mos.ru\\_1602.har](https://zakupki.mos.ru_1602.har)

Файлы со статическим контентом html страницы имеют скорость доставки на порядок меньшую чем все остальные данные. Обработка данных мониторинга приложения не зафиксировала существенного снижения скорости для статики между балансировщиком приложения и кэшем статики. Значит снижение скорости происходит на участке между балансировщиком приложения и балансировщиком на внешнем интерфейсе [zakupki.mos.ru](https://zakupki.mos.ru).

В результате анализа последовательности действий СТП со стороны ЦОД, кластера и подрядчика

было выяснено что служба поддержки не имеет методики анализа, позволяющей отделить задержки возникающие на этапе прохождения запроса от балансировщика ЦОД до балансировщика приложения.

Это не позволяет оперативно и однозначно определить ответственного, что приводит к дополнительным временным затратам. Архитектура используемого решения с зонами ответственности приведена ниже.



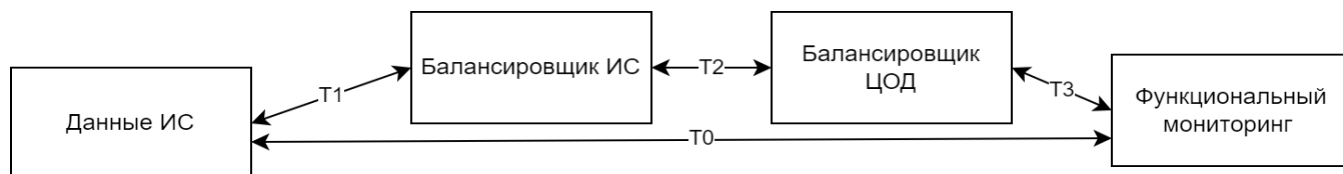
По этому необходимо найти способ обнаружения подобных проблем и их локализации, используя доступные средства диагностики для СТП. Эта проблема является общей для всех систем кластера, использующих внешние балансировщики [mos.ru](https://mos.ru). Поэтому рационально найти общее универсальное решение.

В заключении к постановке к задачи надо отметить, что использование кластера `kubernetes` будет частным случаем рассмотренной выше конфигурации, т.к. в качестве балансировщика приложений будет выступать `ingress` пространства имен выделенных приложения ИС на кластере `rancher kubernetes`, предоставляемом как услуга ЦОД.

## Общая схема решения

Для решения задачи необходимо получить численные оценки прохождения запроса от клиента через зоны ответственности СТП участвующих в обеспечении тракта обработки данных для клиента ИС. Чтобы обеспечить проактивность диагностики клиентом должен выступать функциональный мониторинг.

Рассмотрим типовую схему прохождения запроса.



Необходимо получить нижеперечисленные временные интервалы обработки запросов.

1. T0 - время прохождения запроса для клиента, фиксируется в системе функционального мониторинга.
2. T1 - время прохождения запроса в зоне ответственности СТП продукта.
3. T2 - время прохождения запроса в зоне ответственности СТП сетевого сегмента ЦОД(PTAF + etc).
4. T3 - время прохождения запроса в зоне ответственности провайдера клиента.
5. T0=T1+T2+T3.

## Описание предлагаемой реализации

В настоящий момент в качестве балансировщика приложения и в качестве балансировщика ЦОД используется NGINX.

В документации на этот продукт в разделе

[https://nginx.org/en/docs/http/nginx\\_http\\_core\\_module.html#var\\_request\\_time](https://nginx.org/en/docs/http/nginx_http_core_module.html#var_request_time)

указано, что в журнале доступа описанном в

[https://nginx.org/en/docs/http/nginx\\_http\\_log\\_module.html#access\\_log](https://nginx.org/en/docs/http/nginx_http_log_module.html#access_log)

есть параметры

`$request_time` request processing time in seconds with a milliseconds resolution (1.3.9, 1.2.6); time elapsed since the first bytes were read from the client

`$request_id` unique request identifier generated from 16 random bytes, in hexadecimal (1.11.0)

Тогда

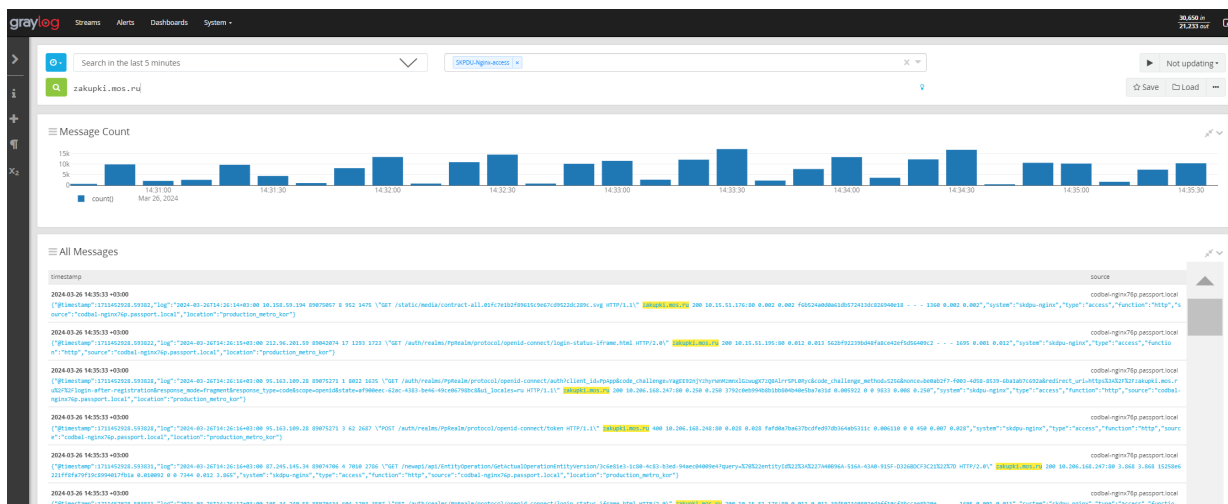
`$request_time` балансировщика приложений = T1;

`$request_time` - `$request_time` балансировщика приложений = T2

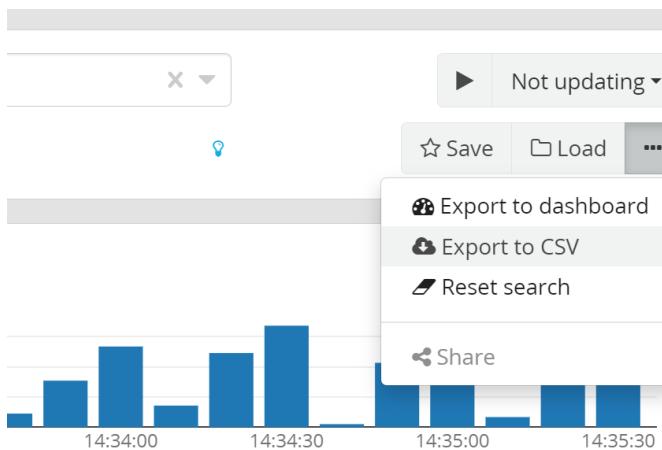
`$request_time` функционального мониторинга = T0

Значение `$request_time` можно получить по ссылке (учетная запись должна иметь соответствующий доступ)

<http://graylog.passport.local>



сделаем запрос на выгрузку



### Export search results as CSV

Please right click the download link below and choose "Save Link As..." to download the CSV file.

Add stream to filter messages:

None: click to add stream

Select fields to export:

timestamp request\_time

[Download](#)

Close

в результате запроса будет выгружен csv файл с фрагментом журнала доступа за 10 минут

Пример файла

```
"timestamp","request_time"
"2024-02-26T08:38:27.047Z","0.212",
"2024-02-26T08:38:27.059Z","0.021",
"2024-02-26T08:38:27.047Z","0.874",
```

Тогда чтобы получить T2 надо аналогичным способом получить `$request_time` балансировщика приложений, т.е. подключить балансировщик приложений к этой же системе анализа.

Но такой способ очень точен т.к. позволяет сопоставить длительность каждого запроса из `log` файла клиента, но требует либо участия эксперта либо специального нетривиального программирования.

В качестве автоматизированной оценки пригодной для автоматизированной обработки рационально использовать анализ журналов Nginx реализованный в шаблоне

[https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/app/nginx\\_http?at=ea07ba6650b981d6c7253594e3216b9dff91e092](https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/app/nginx_http?at=ea07ba6650b981d6c7253594e3216b9dff91e092)

Упрощенно реализованный там алгоритм можно представить в виде суммирования данных представленных в журнале представленном выше, но приведем его еще раз с пояснениями

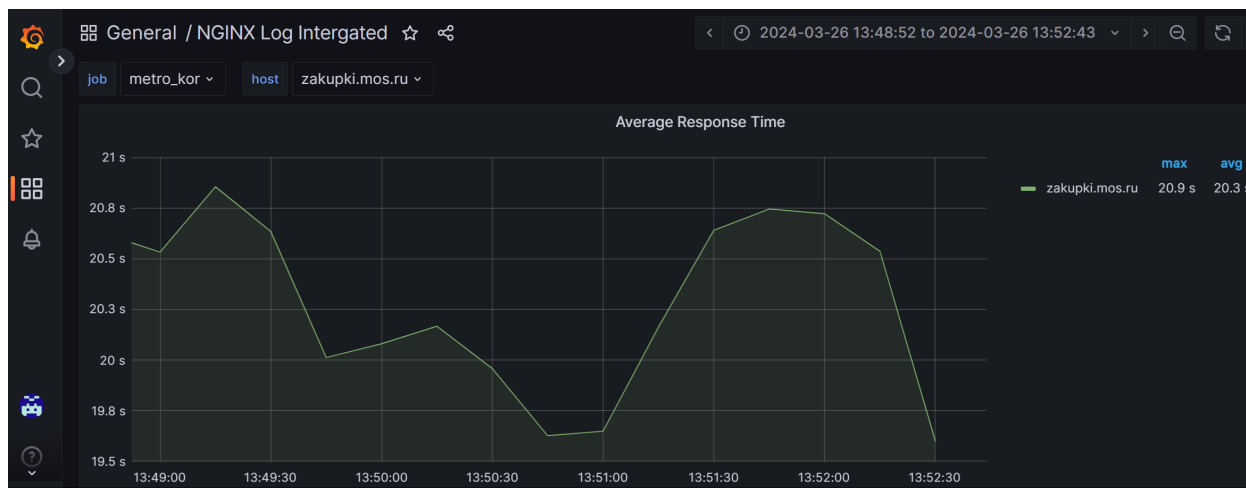
Пример файла

```
"timestamp","request_time"
"2024-02-26T08:38:27.047Z","0.212", -- время начало счета
"2024-02-26T08:38:27.059Z","0.021",
"2024-02-26T08:38:27.047Z","0.874", -- время окончания счета
```

`Request_time` суммируется по диапазону от начало счета до конца, которое составляет окно счета и скользит по журналу доступа выдавая временной ряд следующего состава -

время начало счета, сумма внутри окна по `request_time`. Запросы которые прерываются по ошибке не вносят искажения в статистический анализ т.к. они записываются в не в журнал `access.log`, а в журнал `error.log`.

Этот временной ряд визуализируется в графине ЦОД на сайте `mon-dc.mos.ru`



Если по двум балансировщикам это скользящее окно будет иметь одинаковые границы, то разница между этими графиками даст метрику которая позволит оценить время T2.

Таким образом можно выделить зону на которой происходит рост времени обработки запроса со стороны клиента и эскалировать инцидент в соответствующую СТП.

После реализации решения необходимо определить численные характеристики T2 путем анализа трафика, начиная с которых необходимо заводить инцидент и назначать на соответствующую СТП, с внесением изменений в регламент СТП.

