

Дано:

1. Кластер из 6 веб-серверов (на базе задачи HTTP в HCL Domino на AstraLinux, если важно). Это бекэнд.
2. HAProxy, который распределяет веб-запросы между серверами. Это фронтэнд.
3. Формат лога HAProxy в виде:

log-format

```
'{"status":"%ST","bytes_read":"%B","bytes_uploaded":"%U","method":"%HM","handshake_time":"%Th","request_idle_time":"%Ti","request_time":"%TR","response_time":"%Tr","client_ip":"%ci","server_name":"%s","http_request":"%r","http_referer":"%[capture.req.hdr(1)]"}
```

Что необходимо:

1. Настроить журнал логов (либо в haproxy, либо в любом другом стеке, пробовали ELK, сами не разобрались) в удобном для администраторов и техподдержки виде:
 - a. каждая строка записи в haproxy.log должна быть разложена на составляющие атрибуты:
 - i. server_name - это имя бекэнда?
 - ii. date_time
 - iii. timestamp - в чём отличие от предыдущего пункта?
 - iv. client_ip
 - v. UserLotusName - может содержаться в http_request
 - vi. http_request
 - vii. request_uri - в чём отличие от предыдущего пункта?
 - viii. status
 - ix. bytes_read
 - x. bytes_uploaded
 - xi. request_time (время получения запроса (ответа?) от первого байта) - что конкретно замеряется?
 - xii. request_idle_time - что конкретно замеряется?
 - xiii. response_time - что конкретно замеряется?

про <https://www.haproxy.com/blog/haproxy-log-customization/> знаем, но всё же нужны пояснения
 - b. возможность формирования выборок (язык запросов) по отдельным атрибутам. пример типовых запросов:
 - i. найти все запросы от конкретного пользователя (по ip или username). например, в случае, если пользователь жалуется на тормоза
 - ii. найти запросы, выполняющиеся больше определённого времени. например, нам гвоорят, что какие-то действия начинают тормозить у всех, надо понять, какой именно запрос “тормозит”
 - iii. отобрать все ошибки за последнее время (минуту/полчаса/час). например, пользователь гвоорит, что у него постоянно “вылезают ошибки”. нам надо понять, что за ошибки, при каких действиях
 - iv. и т.п.
2. Настроить средство анализа логов (есть zabbix, предпочтительно использовать его или если нет вариантов, готовы рассмотреть предложения) с возможностью строить графики и диаграммы для наглядного отслеживания состояния веб-части:
 - a. количество текущих веб-пользователей
 - b. распределение веб-пользователей по узлам бекэнда
 - c. группировка возникающих ошибок по бэкэндам

- d. вывод времени выполнения конкретных “тяжёлых” запросов в виде графиков по узлам бекэнда
 - e. и т.п.
3. мы должны понимать, как это работает и как это сопровождать:
- a. по каждому пункту должна быть мини-инструкция, как это настраивается, со ссылками на документацию, минимальными пояснениями
 - b. должен быть созвон под запись с рассказом и ответами на наши вопросы