

AWS Availability zone: EU-WEST-1A



Hostname: compute1
IP: 10.0.0.178
Flavour: t2.large
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: notty2, node.js, docker



Hostname: compute2
IP: 10.0.0.92
Flavour: t2.large
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: notty2, node.js, docker



Hostname: compute3
IP: 10.0.0.58
Flavour: m4.10xlarge
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: notty2, node.js, docker



Hostname: compute4
IP: 10.0.0.191
Flavour: m4.10xlarge
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: notty2, node.js, docker



Hostname: compute5
IP: 10.0.0.164
Flavour: m4.10xlarge
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: notty2, node.js, docker



Hostname: balancer
IP: 10.0.0.34
52.18.162.225 (AWS DNAT)
DNS: cloudba.ru
Flavour: t2.micro
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: haproxy



Hostname: frontend
IP: 10.0.0.24
Flavour: t2.micro
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: nginx (FE), notty(BE), node.js



Hostname: mongodb
IP: 10.0.0.52
Flavour: t2.micro
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: mongodb



Hostname: monitoring
IP: 10.0.0.22
Flavour: t2.micro
Security groups: cloudba
OS: Ubuntu Server 18.04
Roles/SW: prometheus + grafana

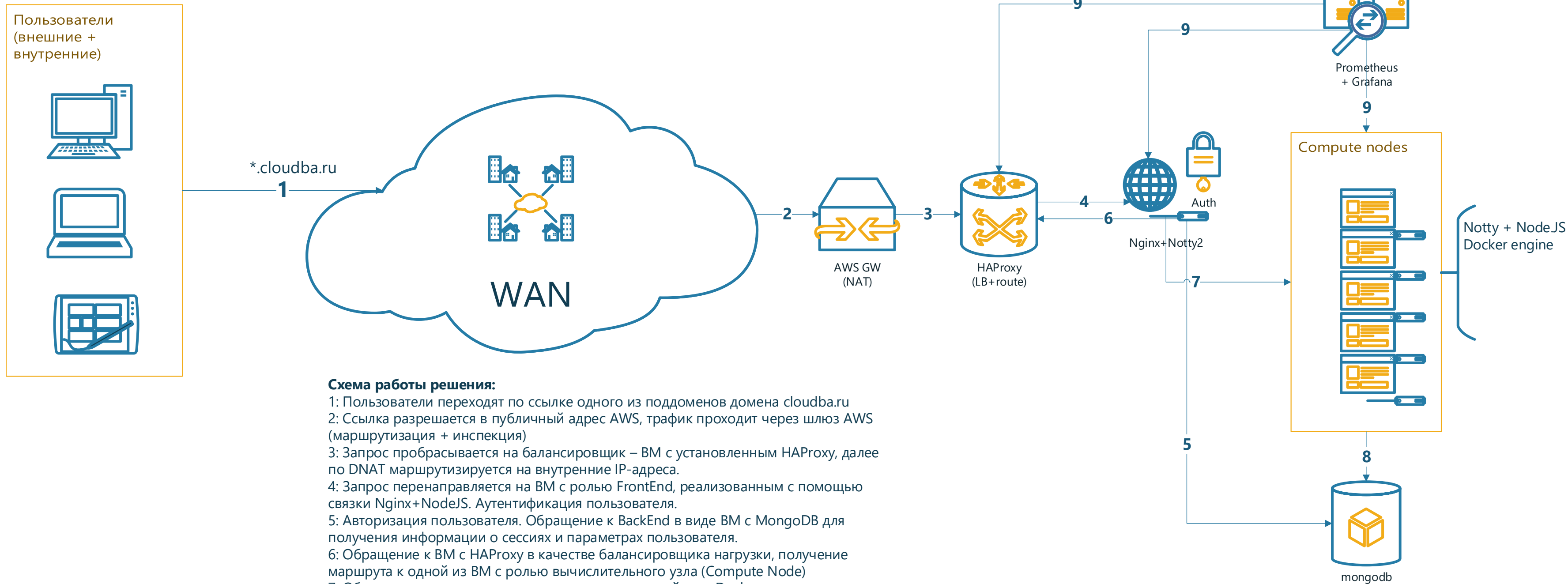


Схема работы решения:

- 1: Пользователи переходят по ссылке одного из поддоменов домена cloudba.ru
- 2: Ссылка разрешается в публичный адрес AWS, трафик проходит через шлюз AWS (маршрутизация + инспекция)
- 3: Запрос пробрасывается на балансировщик – VM с установленным HAProxy, далее по DNAT маршрутизируется на внутренние IP-адреса.
- 4: Запрос перенаправляется на VM с ролью FrontEnd, реализованным с помощью связки Nginx+NodeJS. Аутентификация пользователя.
- 5: Авторизация пользователя. Обращение к BackEnd в виде VM с MongoDB для получения информации о сессиях и параметрах пользователя.
- 6: Обращение к VM с HAProxy в качестве балансировщика нагрузки, получение маршрута к одной из VM с ролью вычислительного узла (Compute Node)
- 7: Обращение к вычислительному узлу, запуск контейнера Docker со сценарием квеста
- 8: Обращение к MongoDB для сохранения состояния сессии, статистики и телеметрии пользовательской сессии
- 9: Мониторинг всех узлов при помощи Prometheus и Grafana