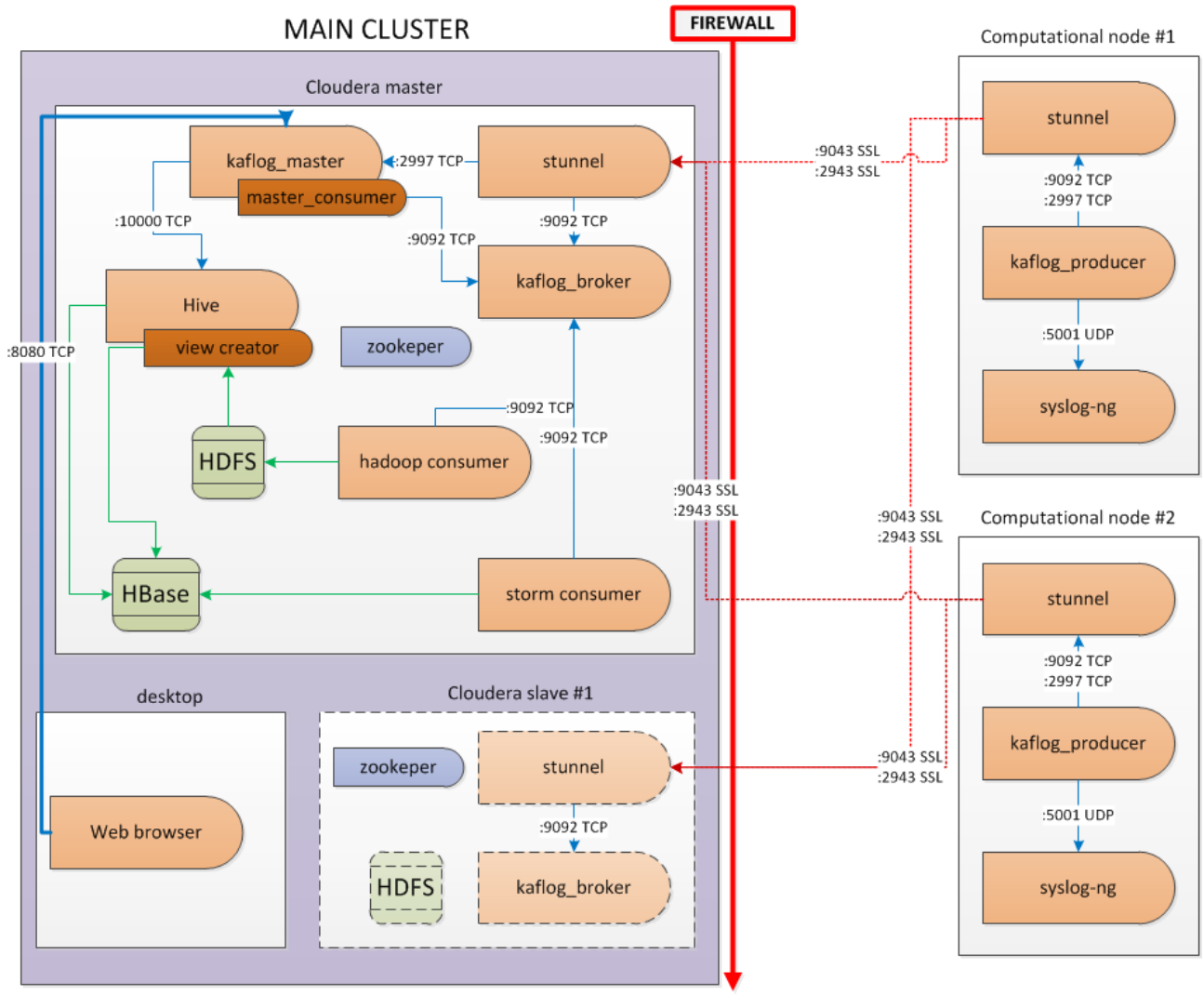
Kaflog – architektura

W tym dokumencie opisana jest architektura systemu i zadania poszczególnych modułów.

Poniżej zamieszczono diagram poglądowy:



Wyjaśnienie oznaczeń i zadania poszczególnych modułów:

1. **Computational Node –** Oznacza dowolną maszynę, z której logi chcielibyśmy zbierać z użyciem systemu Kaflog. Nazwa związana jest z sugerowanym zastosowaniem systemu – zbieranie logów z maszyn obliczeniowych, chociaż system może być równie przydatny dla innych

rodzajów maszyn.

1.1. **syslog-ng** – oprogramowanie dostępne na systemy UNIXowe, implementujące standard syslog i udostępniające dodatkowe funkcjonalności, takie jak publikacja logów na porcie UDP, co jest wykorzystywane w tym projekcie.

1.2. **kaflog_producer** – jeden z modułów systemu Kaflog. Jego zadaniami są:

a) Nasłuchiwanie na porcie UDP w celu przechwycenia logów pojawiających się w systemie.

b) Zbieranie statystyk odnośnie ilości przetwarzanych logów.

c) Publikowanie logów do **kaflog_broker'a**, po sformatowaniu ich.

d) Rejestracja w module **kaflog_master** i regularne przesyłanie statystyk, co jednocześnie funkcjonuje jako heartbeat i pozwala stwierdzić stan podłączenia producentów do **kaflog_master'a**. Jest to realizowane przy użyciu JMX.

1.3. **stunnel –** oprogramowanie pozwalające na opakowywanie zwykłych pakietów internetowych w warstwę SSL. Z racji, że Kafka nie ma wsparcia dla SSLa, takie rozwiązanie było konieczne w celu zapewnienia bezpieczeństwa na połączeniu **kaflog_producer** – **kaflog_broker**. Dodatkowo, osobnym kanałem SSL realizowana jest komunikacja przez JMX.

2. **Main Cluster –** główny, centralny klaster systemu. Jest chroniony przez dostępem z zewnątrz dzięki firewall'owi, który jest skonfigurowany w taki sposób, aby przepuszczać jedynie połączenia do broker'ów. Jest sercem systemu, w którym następuje właściwe przetwarzanie zebranych logów. Składa się z głównej maszyny, która jednocześnie służy za główna maszynę dla **Cloudery**. Zainstalowany jest kompletny zestaw narzędzi oferowanych w ramach **Cloudery,** m. in. Zookeper, Hadoop, Hive, Hbase, Oozie, Pig, Impala, Storm. Dodatkowe węzły mogą być dodawane w miarę potrzeby i służyć jako dodatkowe maszyny dla wyżej wymienionych narzędzi, DataNode'y dla Hadoop'a czy zapasowi brokerzy. W głównym klastrze działają między innymi:

2.1. **stunnel –** instancja po stronie klastra pozwala na „odpakowanie”

pakietów SSL i przekazanie ich do **kaflog_broker'a** czy do **kaflog_mastera**.

2.2. **kaflog_broker –** odpowiednio skonfigurowany broker Kafki. Gromadzi logi przesyłane z węzłów obliczeniowych. Możliwa jest dowolna ilość instancji, lecz należy wtedy dopasować liczbę partycji dla kanału, po którym przesyłane są logi. Zalecane jest po jednej instancji na maszynę.

2.3. **kaflog_master –** główna aplikacji monitorująca. Node'y obliczeniowe rejestrują się u niej przez JMX i utrzymują połączenie pingując co jakiś czas. Dodatkowo, jest konsumentem logów i pozwala przeglądać aktualnie pojawiające się w systemie logi. Ponadto, umożliwia generowanie raportów poprzez wyspecyfikowanie ram czasowych – wykonywane jest wtedy zapytanie do bazy danych przy użyciu Impali. Wystawia panel admina – GUI na porcie 8080.

2.4. **zookeper –** optymalnie po jednej instancji na węzeł klastra głównego, zarządza wszystkimi serwisami zarejestrowanymi w klastrze.

2.5. **hadoop_consumer –** job uruchamiany regularnie, powoduje import logów z brokerów kafki do HDFSa.

2.6. **hive_view_creator –** job uruchamiany regularnie, tworzy widoki w HBase ze wszystkich zebranych danych. Są one później odpytywane przy generowaniu raportu.

2.7. **storm_consumer –** konsumuje logi z broker'ów w realtime i inkrementacyjnie tworzy widoki, które są potem łączone z widokami w HBase w jedną całość.