분산처리시스템에서 공유에 대한 고찰

김규석*
*한국과학기술정보연구원
e-mail:gskim@kisti.re.kr

Study on the shared in a distributed processing system

Gyu-Seok Kim*
*Dept of Information System Operation, KISTI

요 약

인프라로써의 전산자원은 사용자간에 공동 활용될 때 비용 효율을 얻을 수 있다. 이것은 web, was, dbms 중심의 기존 정보시스템 뿐 만 아니라 빅 데이터 등 분산처리가 요구되는 분야에서도 마찬가지이다. 분산처리 시스템에서 보안상 안전하게 전산 리소스를 공유하기 위한 요소를 살펴보고 사용자별로 격리된 클러스터를 제공하는 분산처리 공유 플랫폼을 구축하여 확인해 보았다.

1. 서론

각종 응용(Application)과 다수의 사용자를 수용해야하는 데이터센터는 어떻게 효과적인 시스템을 구축할 수 있을까? 최근의 IT 트랜드는 “소프트웨어 정의 데이터센터(SDDC)의 구축”으로 집중되고 있다. 특별하지 않은(commodity) 서버와 스위치 장비를 가지고 소프트웨어를 이용하여 네트워킹과 스토리지 서비스를 구성하고 추상화된 제어와 관리를 제공하는 SDDC는 Openstack과 같은 오픈소스 커뮤니티에 힘입어 도약을 거듭하고 있다.

데이터센터 운영자는 시스템 리소스를 여럿이 최대한 공유하면서도 안전하게 격리되는(Multi-Tenant) 시스템을 구현하기를 희망한다. 어떤 요구사항이 만족해야 멀티테넌시를 구현했다고 할 수 있을까? 이를 만족하는 솔루션은 무엇이고 어떻게 구현 되는가?

2. 분산처리 플랫폼으로써의 Openstack

Openstack은 아파치 라이센스를 가지고 IaaS 서비스를 제공하는 클라우드 OS다. 이것은 이용자로부터 요구되는 시스템 리소스를 프로비저닝 해주는 전형적인 IaaS이지만, 부하분산(LBaaS)과 분산파일시스템으로 구성된 점은 분산처리 플랫폼으로써의 기본 특성을 갖고 있다고 할 수 있다. 이러한 기본 플랫폼 위에 Kubernetes 또는 Mesos 등의 클러스터를 배포하는 환경을 구현한다면 분산컴퓨팅과 데이터센터 내 자원의 공유가 가능한 “분산처리 공유 플랫폼”의 위상을 갖출 수 있게 될 것이다.

2.1 Kubernetes 클러스터

Kubernetes는 컨테이너로 구성된 응용 프로그램의 자동배포, 확장 및 관리를 위한 오픈 소스 시스템이며, 다음과 같은 제공 기능에서 분산과 공유의 특성을 확인 할 수 있다.[1]

- -가용성을 유지하는 가운데 자원 요구사항과 다른 제약조건에 따라 자동 배치되며
- -간단한 명령으로, 또는 CPU 사용량에 따라 자동으로 응용(Application)을 확장 또는 축소할 수 있고
- -장애 때는 자동 재시작, 노드가 죽었을 때는 재배치 등 자가치료 기능이 있으며,
- -컨테이너 마다 전용 IP를 주고, 그룹에게는 단독 도메인 이름을 주어 그들 간에 부하분산이 가능하며
- -이미지 재구성 없이 정보보호와 응용구성을 배포 및 갱신이 가능하다.

2.2 Mesos 클러스터

Mesos는 분산 시스템 커널이다. Mesos는 추상화의 레벨만 다를 뿐 linux 커널과 똑같은 원리로 만들어 졌다. Mesos 커널은 모든 머신에서 실행되면서 응용프로그램(Spark, Elasticsearch 등)들에게 데이터센터 전체를 대상으로(혹은 클러스터를 대상으로) 자원 관리와 스케쥴링을 해준다. 이에 따라, CPU, memory, disk, ports, module 등 일반 자원에 대한 최고 수준의 격리를 지원하며 1만개 노드로의 선형적인 확장이 가능하다.

3. 안전한 공유를 위한 요건

분산처리 플랫폼을 여럿이 공유하려면 다음 5가지 사항이 만족되어야 한다.[2]

① 멀티테넌시 서비스를 이용하는 모든 사용자들은 시스템의 자원을 최대한으로 공유해서 사용해야 한다. ② 서비스를 이용하는 사용자들 사이에 데이터와 성능에 대

해서 간섭이 없어야 한다. ③ 싱글테넌트 시스템과 비슷한 수준의 보안을 지원한다. ④ 다른 사용자의 실행 로직을 실행할 수 없어야 한다. ⑤ 개개인의 사용자는 싱글테넌트 모델처럼 시스템의 모든 설정을 할 수 없다. 하지만 자신만의 환경을 이용하는 것과 같이 시스템을 설정해서 사용할 수 있어야 한다.

OpenStack과 같은 가상 머신을 제공할 수 있는 클라우드 플랫폼을 이용하여 사용자별로 가상 클러스터를 제공하여 사용자에게 각자의 Hadoop 플랫폼을 지원하는 경우를 상정해보자. 클라우드 인증체계를 거쳐 논리적으로 분리된 네트워크위에 사용자별로 독립된 플랫폼을 주는 것이므로 데이터센터 운영자에게는 전체 자원 범위 내에서의 분배 부분을, 이용자 측면에서는 할당받은 클러스터에 대한 접근통제만 관리하면 될 것이다.

4. 플랫폼 구축

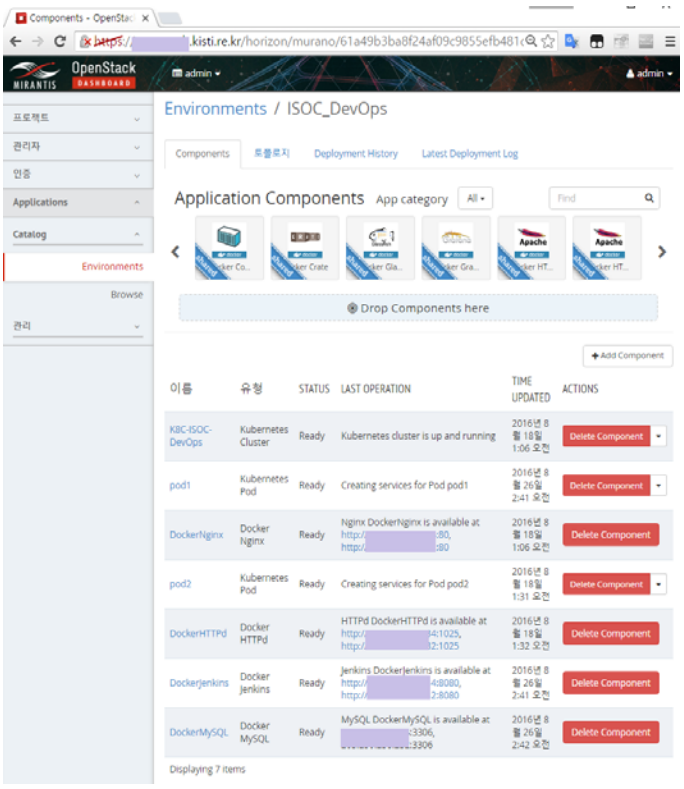
필자는 Mellanox sx1710 스위치 1대, HP DL370g7 10대, DL380g8 서버 4대 등의 물리적인 장비와 Mirantis Openstack V9(MOS9) 배포본을 가지고 파일럿 시스템을 구성하였다. Node 구성은 fuel master 1대, Control 3대, Computing 6대를, DL380서버로는 Computing & CEPH 역할로 4대를 구성했으며 DVR(High Availability using Distributed Virtual Routing) 시나리오[3]를 따라 설치하였다.

스위치 포트별로 VLAN Tagging 여부에 대한 설정에 오류가 없다면 MOS9의 설치는 완전 자동으로 진행된다. fuel master 설치 후 Environment를 생성 할 때 하이퍼바이져 유형은 QEMU-KVM을, Network은 Neutron with VLAN을, Storage 형태는 모두 Ceph을 선택하였고 추가 서비스로 Sahara와 Murano를 설치하였다.

이렇게 준비된 클라우드 플랫폼 위에 centos-7 Instance를 필요 용량과 대수만큼 생성한 후 수동 설치하는 방법으로 Mesos 클러스터는 만들어지며, 인증된 사용자들의 실행요청은 프레임워크를 통해 스케쥴링 된다. Kubernetes 클러스터는 Murano의 패키지 관리기능으로 Openstack App Catalog 커뮤니티(http://apps.openstack.org)에 있는 “Docker & Kubernetes Bundle”을 Import 하고, Application - Catalog - Environment - Crate Environment 메뉴를 통해 “Kubernetes Cluster”, “Kubernetes Pod” 및 “Docker Nginx” 등 컴포넌트를 모아 배포(Deploy)하면 (그림1)과 같이 Kubernetes Cluster가 만들어 지며, (그림 2) 네트워크 토폴로지와 같이 데이터 센터 내의 어떤 사용자와도 분리된/격리된 클러스터를 형성하게 된다.

5. 결론

급변하는 IT환경 속에서, 비용효율적인 컴퓨팅 플랫폼이 되려면 기능과 성능의 만족은 물론 투자가 보호되는 지속가능한 Topology를 찾는 것이 중요하다. 본 고 에서는 분산처리와 SDDC 등 최근의 IT트랜드를 바탕으로 멀



(그림 1) Kubernetes Cluster 구성 및 배포

(그림 2) 독립적인 클러스터

티 테넌시가 지원되는 공유 플랫폼의 한 형태를 구축하고 안전한 공유를 위한 구조를 확인해 보았다. 향후, 데이터센터 차원에서 자원의 적절한 배분은 어떻게 만족 될 수 있는지도 검토되면 좋겠다.

참고문헌

[1] http://kubernetes.io/

[2] 정진규 “대용량 분산처리 플랫폼 공유 모델 연구” 국가RnD연구보고서2015

[3] http://docs.openstack.org/mitaka/networking-guide/scenario-dvr-ovs.html