

High Available RKE2 Kubernetes Cluster için OpenStack Altyapısında Network ve Sanal Makine Yapılandırılması

OpenStack üzerinde kuracağımız Kubernetes Cluster'i için kullanacağımız yapı 3 Master, 3 Worker ve Master HA yapısını uygulayacağımız Bastion LoadBalancer VM'lerinden oluşmaktadır.

Bu kısımda ilk önemli olan yer Network yapılandırılmasıdır. Bu noktada kullanacağımız Kubernetes Cluster Dağıtıcısı olan RKE2'dir. Network tarafında yapacağımız işlemler öncesi RKE2 dokümantasyonu üzerinden ağ yapılandırılmasına dair gereksinim şu şekilde belirtilmiştir:

Inbound Network Rules

Port	Protocol	Source	Destination	Description
6443	TCP	All RKE2 nodes	RKE2 server nodes	Kubernetes API
9345	TCP	All RKE2 nodes	RKE2 server nodes	RKE2 supervisor API
10250	TCP	All RKE2 nodes	All RKE2 nodes	kubelet metrics
2379	TCP	RKE2 server nodes	RKE2 server nodes	etcd client port
2380	TCP	RKE2 server nodes	RKE2 server nodes	etcd peer port
2381	TCP	RKE2 server nodes	RKE2 server nodes	etcd metrics port
30000-32767	TCP	All RKE2 nodes	All RKE2 nodes	NodePort port range

CNI Specific Inbound Network Rules

[Canal](#) [Cilium](#) [Calico](#) [Flannel](#)

Port	Protocol	Source	Destination	Description
8472	UDP	All RKE2 nodes	All RKE2 nodes	Canal CNI with VXLAN
9099	TCP	All RKE2 nodes	All RKE2 nodes	Canal CNI health checks
51820	UDP	All RKE2 nodes	All RKE2 nodes	Canal CNI with WireGuard IPv4
51821	UDP	All RKE2 nodes	All RKE2 nodes	Canal CNI with WireGuard IPv6/dual-stack

Container Network Interface (CNI) kısmında birden fazla seçeneğimiz vardır. Bu rehberde Canal olarak devam edeceğiz. Bir sonraki kısımda OpenStack üzerinden cluster network ayarlanması üzerinden devam edebiliriz.

OpenStack üzerinden Network Yapılandırılması

Yapılandırmanın tamamı arayüz üzerinden yapılmıştır aşağıdaki şekilde kendi yapı ve mimarinize uygun şekilde isimlendirmede değişikliğe gidebilirsiniz. İlk önce Create Network

kısından örnek datacenter üzerine kurulu olacak kubernetes clusteri ifade etmek için dc01 olarak gidilmiştir. Network ismi yerinde **dc01-private-net** yazarak devam edebiliriz.

Create Network

Network Subnet Subnet Details

Network Name

dc01-private-net

Enable Admin State ?

Shared

Create Subnet

Availability Zone Hints ?

MTU ?

- +

CANCEL « BACK NEXT »

Oluşturulacak node'ların paylaşacağı ip subnet için **private-subnet** diyerek devam edebiliriz. Yapılandıracağımız kubernetes cluster yapısında /24 olarak kullanacağız. Biz **10.20.0.0/24** olarak devam edebiliriz. Gateway IP kısmını boş olarak geçebilirsiniz default olarak ip'yi altyapı otomatik olarak tahsis edecektir.

Kubernetes service cidr ve pod cidr ile çakışmamasına dikkat ediniz.

Create Network ×

Network **Subnet** Subnet Details

Subnet Name
private-subnet

Network Address ⓘ
10.20.0.0/24

IP Version
IPV4

Gateway IP ⓘ
10.20.0.1

Disable Gateway

Creates a subnet associated with the network. You need to enter a valid "Network Address" and "Gateway IP". If you did not enter the "Gateway IP", the first value of a network will be assigned by default. If you do not want gateway please check the "Disable Gateway" checkbox. Advanced configuration is available by clicking on the "Subnet Details" tab.

CANCEL « BACK NEXT »

Burada sahip olduğumuz altyapıdan ötürü DHCP açık olarakta devam edebilirsiniz. Ama Kubernetes fixed ip'ler önerildiği ve ip değişikliğinde yaşanacak platform dengesizlik problemi için DHCP'yi deaktif yapabilirsiniz.

Bu dokümantasyonda DHCP aktif olarak devam edeceğiz.

Create Network

Network Subnet Subnet Details

Enable DHCP

Specify additional attributes for the subnet.

Allocation Pools ?

DNS Name Servers ?

Host Routes ?

CANCEL « BACK CREATE

Router'in Oluşturulması

Kubernetes RKE2 kurulumu gerçekleştirebilmek için bazı paketlerin güncelliği ve argümanlara göre dinamik indirip ona göre paketlerin çekilebilmesi için cluster'imizin internete erişimi gerekmektedir bunun için bir router tanımlaması yapalım. İsimlendirme olarak **dc01-router** olarak devam edebiliriz.

Create Router ✕

Router Name
dc01-router

Enable Admin State ?

External Network
EXT_NET ▼

Enable SNAT

Availability Zone Hints ?

CANCEL CREATE ROUTER

Description:

Creates a router with specified parameters.

Enable SNAT will only have an effect if an external network is set.

Subnet kısmında önceki başlıkta oluşturduğumuz subnet'i seçerek opsiyonel olarak ip'sini verebiliriz.

Add Interface ×

Subnet *

DC01-PRIVATE-NET: 10.20.0.0/24 (PRIV... ▾

IP Address (optional) ?

10.20.0.1

Description:

You can connect a specified subnet to the router.

If you don't specify an IP address here, the gateway's IP address of the selected subnet will be used as the IP address of the newly created interface of the router.

If the gateway's IP address is in use, you must use a different address which belongs to the selected subnet.

Security Group ile Cluster Internal Konfigürasyonu

Bu kısımda iki farklı security group tanımlaması ile devam edeceğiz bunlardan ilki RKE2 cluster'imizin master ve worker node'lar üzerindeki RKE2 Gereksinimleri kısmından istenilen port ve protokollere izin vererek devam edeceğiz. İlk önce node-to-node yapısının anlaşılabilir olması için **dc01-internal-sec** yazarak Security Group işlemine başlayalım.

Create Security Group ✕

Name ^{*}

dc01-internal-sec

Description

Description

Security groups are sets of IP filter rules that are applied to network interfaces of a VM. After the security group is created, you can add rules to the security group.

CREATE SECURITY GROUP

İlk önce oluşturacağımız LoadBalancer tarafından master ve worker node'lara bağlanabilmek için 10.20.0.0/24 altından SSH kuralı ekleyerek devam edelim.

Add Rule ✕

Rule ^{*}

SSH ▾

Description [?]

internal-ssh

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Node'lar arasındaki bağlantının test edilmesi ve bazı noktalardaki network debug işlemi için ping kullanılabilir bundan dolayı ICMP kuralı ekleyerek 10.20.0.0/24 cidr kısmını seçelim.

Add Rule ✕

Rule ^{*}

ALL ICMP ▾

Description [?]

icmp-debug

Direction

INGRESS ▾

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Cluster kurulumu sonrası kullanacağımız ingress-nginx node'larda hostPort olarak 80 portunu HTTP için açmaktadır bunun için HTTP kuralını da ekleyelim.

Add Rule

✕

Rule *

HTTP

Description ?

ingress-nginx

Remote * ?

CIDR

CIDR ?

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

CANCEL ADD

Ingress-nginx ayrıca HTTPS için 443 portunu da açmaktadır bunun için kural kısmından HTTPS seçerek private subnetimizi de ekleyerek devam edelim.

Add Rule

×

Rule *

HTTPS

Description ?

ingress-nginx

Remote ?

CIDR

CIDR ?

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

CANCEL ADD

RKE2 cluster yapımızın sağlıklı çalışabilmesi için **etcd** bileşeninin iletişimini sağlamamız gerekiyor:

- Client Port (2379): **Kubernetes API Server'ın veri tabanına (etcd) yazabilmesi ve sorgu atabilmesi için bu yolu açık tutuyoruz.**
- Peer Port (2380): **Birden fazla server (master) node kullandığımızda, node'ların birbirlerinden haberdar olması ve veriyi kendi aralarında eşitlemesi (High Availability) için bu portu kullanıyoruz.**
- Health Metrics (2381): **Sistemimizin durumunu izlemek ve metrikleri toplamak için bu port üzerinden haberleşiyoruz.**

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

etcd-ports

Direction

INGRESS ▾

Open Port ^{*}

PORT RANGE ▾

From Port ^{*} [?]

2379

To Port ^{*} [?]

2381

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

CANCEL ADD

OpenStack üzerinde **6443 TCP** portunu, Kubernetes'in ana yönetim merkezi olan **API Server**'a erişim sağlamak için açıyoruz. Bu sayede hem `kubectl` komutlarımızla cluster'ı dışarıdan yönetebiliyor hem de worker node'larımızın master node'lar ile iletişim kurup cluster'a dahil olmasını sağlıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

kubernetes-api

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

6443

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

CANCEL ADD

OpenStack üzerinde **9345 TCP** portunu, RKE2'ye özel **Supervisor API** servisi için açıyoruz; bu port sayesinde yeni eklenen node'ların cluster'a güvenli bir şekilde kayıt olmasını (registration) sağlıyor ve node'lar arasındaki TLS sertifikalarının dağıtımını ile güncellenmesini yönetiyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

rke2-supervisor-api

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

9345

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

OpenStack üzerinde **10250 TCP** portunu, her node üzerinde çalışan **Kubelet** servisinin güvenli iletişimi için açıyoruz; bu sayede master node üzerindeki API Server'ın worker node'lara bağlanarak pod loglarını çekmesini, komut çalıştırmasını (`kubectl exec`) ve node üzerindeki kaynakları yönetmesini sağlıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

kubelet-port

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

10250

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

OpenStack üzerinde **30000-32767 TCP/UDP** port aralığını **NodePort** servisleri için açıyoruz; bu sayede Kubernetes üzerinde koşan uygulamalarımızı dış dünyaya açabiliyor ve gelen trafiğin herhangi bir node üzerinden doğrudan ilgili pod'lara yönlendirilmesini sağlıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

node-port-range

Direction

INGRESS ▾

Open Port ^{*}

PORT RANGE ▾

From Port ^{*} [?]

30000

To Port ^{*} [?]

32767

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

CANCEL ADD

OpenStack üzerinde **8472 UDP** portunu, Kubernetes ağ trafiğini yöneten **Canal CNI** eklentisinin **VXLAN** katmanı için açıyoruz; bu sayede farklı node'lar üzerinde çalışan pod'ların birbirleriyle sanki aynı fiziksel ağdaymış gibi güvenli bir tünel üzerinden haberleşmesini sağlıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM UDP RULE ▾

Description [?]

canal-cni with vxlan

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

8472

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

OpenStack üzerinde **9099 TCP** portunu, RKE2 içerisinde ağ trafiğini yöneten **Canal (Calico + Flannel)** bileşenin sağlık kontrollerini (health checks) yapabilmesi için açıyoruz; bu sayede sistemin her bir node üzerindeki ağ sürücüsünün düzgün çalışıp çalışmadığını denetlemesine ve olası ağ hatalarını raporlamasına olanak tanıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

canal-cni health check

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

9099

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

OpenStack üzerinde **51820 UDP** portunu, RKE2 içerisinde ağ trafiğini şifrelemek için kullanılan **WireGuard** protokolü için açıyoruz; bu sayede node'lar arasındaki tüm pod trafiğinin (CNI seviyesinde) yüksek performanslı ve güvenli bir tünel üzerinden şifrelenmiş olarak iletilmesini sağlıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM UDP RULE ▾

Description [?]

canal-cni wireguard ipv4 port

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

51820

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

OpenStack üzerinde **51821 UDP** portunu, RKE2'de **WireGuard** protokolü ile ağ şifrelemesi (IPv6 desteği dahil) yaparken kullanılan ikinci veri tüneli için açıyoruz; bu sayede node'lar arasındaki güvenli haberleşmenin kesintisiz devam etmesini ve şifrelenmiş trafiğin alternatif yollar üzerinden de akabilmesini sağlıyoruz.

Add Rule ✕

Rule ^{*}

CUSTOM UDP RULE ▾

Description [?]

canal-cni wireguard ipv6/dualstack port

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

51821

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

10.20.0.0/24

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

CANCEL ADD

Security Group ile LoadBalancer Konfigürasyonu

Dışardan clusterimize bir geçiş kapısı ve HA fonksiyonallitesini sağlayabilmek için aşağıdaki şekilde security group yapılandırılmasına ihtiyacımız vardır. **dc01-sec** diyerek security group oluşturmaya başlayabiliriz.

Add Rule ✕

Rule ^{*}

ALL ICMP ▾

Description [?]

icmp-debug

Direction

INGRESS ▾

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

193.140.85.182/32

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Kendi public ip miz ile loadbalancer sanal makinemize erişmek için SSH iznini de tanımlayalım.

Add Rule ✕

Rule ^{*}

SSH ▾

Description [?]

remote-connect

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

193.140.85.182/32

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

HTTP yönlendirilmesi bizim web üzerinden kubernetes cluster'a yaptığımız deploy'ları inceleyebilmek ve kullanabilmek adına önemlidir. Bunun için sırasıyla HTTP ve HTTPS kurallarını kendi public ip mizi belirterek ekleyelim.

Add Rule



Rule *

HTTP

Description ?

nginx-access

Remote * ?

CIDR

CIDR ?

193.140.85.182/32

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

CANCEL

ADD

Add Rule ✕

Rule ^{*}

HTTPS ▾

Description [?]

nginx-access

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

193.140.85.182/32

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

İleride Cluster yönetim için bir Lens veya Portainer gibi bir tool kullanacaksak onun içinde kubernetes api portuna da izin verelim.

Add Rule ✕

Rule ^{*}

CUSTOM TCP RULE ▾

Description [?]

remote host to kubernetes api access port

Direction

INGRESS ▾

Open Port ^{*}

PORT ▾

Port ^{*} [?]

6443

Remote ^{*} [?]

CIDR ▾

CIDR ^{*} [?]

193.140.85.182

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

LoadBalancer Sanal Makinesinin Oluşturulması

LoadBalancer sanal makinesini oluşturma aşamasına adını kullanacağımız kubernetes node'larıyla bağlantılı olduğunu belirtmek için adın dc01-bastion-lb diyerek kuruluma devam edebiliriz.

Launch Instance



Details

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Source *

Project Name

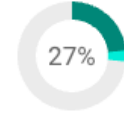
alperen_egitim

Total Instances
(30 Max)

Flavor *

Instance Name *

dc01-bastion-lb



Networks *

Description

7 Current Usage
1 Added
22 Remaining

Network Ports

Security Groups

Availability Zone

AZ1

Key Pair

Count *

1

Configuration

Server Groups

Scheduler Hints

Metadata

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Image seçiminde ise Ubuntu 22-04 üzerinden ilerleyeceğiz. Hem load balancer hem de kubernetes node'ları bu şekilde oluşturulacak.

Launch Instance



Details

Instance source is the template used to create an instance. You can use an image, a snapshot of an instance (image snapshot), a volume or a volume snapshot (if enabled). You can also choose to use persistent storage by creating a new volume.

Source

Select Boot Source

Create New Volume

Image

YES

NO

Flavor *

Networks *

Allocated

Displaying 1 item

Name	Updated	Size	Format	Visibility
> ULAKBIM-Ubuntu-22.04-jammy	8/7/25 6:46 AM	645.93 MB	QCOW2	Public

Network Ports

Security Groups

Key Pair

Displaying 1 item

Configuration

Available 23

Select one

Server Groups

ubuntu

Scheduler Hints

Displaying 3 items

Name	Updated	Size	Format	Visibility
> ULAKBIM-Ubuntu-18.04-bionic	8/7/25 6:45 AM	386.88 MB	QCOW2	Public
> ULAKBIM-Ubuntu-20.04-focal	8/7/25 6:46 AM	617.97 MB	QCOW2	Public
> ULAKBIM-Ubuntu-24.04-noble	8/7/25 6:46 AM	587.12 MB	QCOW2	Public

Metadata

Displaying 3 items

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Load balancer sanal makinemiz için 4 CPU 8 RAM yeterli olacaktır. **fkm.c4m8.d20** seçerek kurulumu devam edelim.

Launch Instance



Details

Source

Flavor

Networks *

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

Allocated

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public	
fkm.c4m8.d2 > 0	4	8 GB	20 GB	20 GB	0 GB	Yes	↓

Available 38

Select one

🔍 c4m8



Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
------	-------	-----	------------	-----------	----------------	--------

× CANCEL

← BACK

NEXT →

🔧 LAUNCH INSTANCE

Oluşturmuş olduğumuz dc01-private-net'i burada seçerek buradan ip almasını ve yönlendirilmesini sağlayalım.

Launch Instance



Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Networks provide the communication channels for instances in the cloud.

Allocated ¹

Select networks from those listed below.

	Network	Subnets Associated	Shared	Admin State	Status	
I# 1	> dc01-private-net	private-subnet	No	Up	Active	↓

Available ²

Select at least one network

Click here for filters or full text search. ×

	Network	Subnets Associated	Shared	Admin State	Status	
>	ext_net	subnet_ext	Yes	Up	Active	↑
>	rke2-private-net	rke2-private-subnet	No	Up	Active	↑

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Burada önceden oluşturulmuş olan ip'ler üzerinden gidebiliriz fakat biz DHCP Enable yaptığımız için Allocated kısmını boş bırakarak devam edelim.

Launch Instance



Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Ports provide extra communication channels to your instances. You can select ports instead of networks or a mix of both.

Allocated ¹

Select ports from those listed below.

Displaying 1 item

Name	IP	Admin State	Status	
> dc01-bastion-lb-ip	10.20.0.14 on subnet private-subnet	Up	Down	↓

Displaying 1 item

Available ⁶

Select one or more ports

Click here for filters or full text search. ×

Displaying 6 items

Name	IP	Admin State	Status	
> dc01-worker-1-ip	10.20.0.15 on subnet private-subnet	Up	Down	↑
> dc01-worker-3-ip	10.20.0.17 on subnet private-subnet	Up	Down	↑
> dc01-master-2-ip	10.20.0.12 on subnet private-subnet	Up	Down	↑
> dc01-master-1-ip	10.20.0.11 on subnet private-subnet	Up	Down	↑
> dc01-master-3-ip	10.20.0.13 on subnet private-subnet	Up	Down	↑
> dc01-worker-2-ip	10.20.0.16 on subnet private-subnet	Up	Down	↑

Displaying 6 items

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Load balancer hem dış taraftan erişeceğimiz ve kubernetes cluster'imiz arasındaki iletişime sahip olacağı için hem **dc01-sec** hem de **dc01-internal-sec** security group'larını atayalım.

Launch Instance



Details

Select the security groups to launch the instance in.

Source

▼ Allocated ²

Displaying 2 items

Flavor

Name

Description

> dc01-internal-sec



Networks

> dc01-sec



Network Ports

Displaying 2 items

Security Groups

▼ Available ³

Select one or more

Key Pair

🔍 Click here for filters or full text search. ×

Configuration

Displaying 3 items

Name

Description

> rke2-sec



Server Groups

Scheduler Hints

> default

Default security group



Metadata

> rke2-internal-sec



Displaying 3 items

× CANCEL

← BACK

NEXT →

🔥 LAUNCH INSTANCE

Burada ise eğer cluster'de kullanmak üzere kendi lokal bilgisayarımızdan oluşturduğumuz bir key-pair varsa onu seçerek devam edelim. Eğer yoksa kısa bir şekilde oluşturmak için. Bir terminal açıp şu komutları yazınız:

```
ssh-keygen -t rsa -b 2048
# Onaylamalarla devam edelim ardından
cat ~/.ssh/id_rsa.pub
```

Komutun çıktısını Import Key Pair tarafını seçerek key type SSH Key olacak şekilde public key kısmına terminal çıktımızda olan public key'i yapıştırarak oluşturabilirsiniz.

Launch Instance



Details

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

Source

+ CREATE KEY PAIR

IMPORT KEY PAIR

Flavor

Allocated

Displaying 1 item

Networks

Name	Type	Fingerprint
------	------	-------------

Network Ports

> alperen-macbook	ssh	da:c4:31:8d:44:a6:c7:c3:1e:c3:92:0f:ad:41:2c:5e	↓
-------------------	-----	---	---

Displaying 1 item

Security Groups

Key Pair

Available ¹

Select one

Click here for filters or full text search.

Configuration

Displaying 1 item

Server Groups

Name	Type	Fingerprint
------	------	-------------

Scheduler Hints

> bastion-ssh	ssh	5c:08:a8:79:a7:09:aa:fc:24:92:c2:9e:ee:45:5a:a6	↑
---------------	-----	---	---

Displaying 1 item

Metadata

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Launch Instance diyerek load balancer sanal makinemizi oluşturalım. Ardından sanal makinelerin olduğu menüden load balancer sanal makinemizin yanındaki actions kısmından Associate Floating IP verelim.

Allocate Floating IP ×

Pool *

EXT_NET

Description

dc01-rke2-access

DNS Domain

DNS Name

Description:

Allocate a floating IP from a given floating IP pool.

Project Quotas

Floating IP 1 of 5 Used

Kubernetes Cluster Node'lerinin Oluşturulması

Cluster'ın yüksek erişilebilirliğini (High Availability) sağlamak amacıyla, node'larımızı OpenStack üzerindeki farklı Availability Zone'lara (AZ) dağıtarak oluşturuyoruz. Bu sayede bir bölgede (AZ) oluşabilecek donanım veya enerji kesintisinden tüm cluster'ın etkilenmesini engelliyoruz.

- **Master (Control Plane) Node'lar:** Toplam 3 adet master node; her biri sırasıyla **AZ1, AZ2 ve AZ3** bölgelerinde yaratılacak. Bu yapı, etcd veri tutarlılığı ve yönetim katmanının kesintisizliği için kritiktir.
- **Worker Node'lar:** Toplam 3 adet worker node; iş yüklerinin dengeli dağılması için yine sırasıyla **AZ1, AZ2 ve AZ3** bölgelerine dağıtılacak.

Aşağıda belirtilen tüm adımlar her bir node için tekrarlanacaktır.

Sanal Makine Adı	Rol	Availability Zone
dc01-master-1	Control Plane / etcd	AZ1
dc01-master-2	Control Plane / etcd	AZ2
dc01-master-3	Control Plane / etcd	AZ3
dc01-worker-1	Worker	AZ1

Sanal Makine Adı	Rol	Availability Zone
dc01-worker-2	Worker	AZ2
dc01-worker-3	Worker	AZ3

Launch Instance



Details

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Source *

Flavor *

Networks *

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Project Name

alperen_egitim

Instance Name *

dc01-master-1

Description

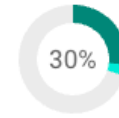
Availability Zone

AZ1

Count *

1

Total Instances
(30 Max)



8 Current Usage
1 Added
21 Remaining

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Sanal makine imajımız için de Ubuntu 22.04 üzerinden seçerek devam edebiliriz.

Launch Instance



Details

Source

Flavor *

Networks *

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Instance source is the template used to create an instance. You can use an image, a snapshot of an instance (image snapshot), a volume or a volume snapshot (if enabled). You can also choose to use persistent storage by creating a new volume.

Select Boot Source

Create New Volume

Image

YES

NO

Allocated

Displaying 1 item

Name	Updated	Size	Format	Visibility	
> ULAKBIM-Ubuntu-22.04-jammy	8/7/25 6:46 AM	645.93 MB	QCOW2	Public	↓

Displaying 1 item

Available 23

Select one

ubuntu

Displaying 3 items

Name	Updated	Size	Format	Visibility	
> ULAKBIM-Ubuntu-18.04-bionic	8/7/25 6:45 AM	386.88 MB	QCOW2	Public	↑
> ULAKBIM-Ubuntu-20.04-focal	8/7/25 6:46 AM	617.97 MB	QCOW2	Public	↑
> ULAKBIM-Ubuntu-24.04-noble	8/7/25 6:46 AM	587.12 MB	QCOW2	Public	↑

Displaying 3 items

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Cluster Node'larının RKE2'nin sistem gereksinimleri seviyesinde vermekte yeterli olabilir. Flavor olarak 8 CPU 8 RAM 100GB Root Disk olarak seçerek ilerlememizdeki sebepler şunlardır:

- **etcd Kararlılığı:** RKE2'nin kalbi olan etcd veri tabanının gecikme yaşamadan (latency) çalışması için yüksek işlemci gücü şart; 8 VCPU seçerek yönetim katmanında oluşabilecek darboğazların önüne geçiyoruz.
- **Servislerin Sürekliliği:** 8 GB RAM tercihiyle, hem işletim sisteminin hem de RKE2 bileşenlerinin (API Server, Kubelet vb.) aynı anda kararlı çalışmasını sağlıyor, bellek yetersizliği kaynaklı çökmeleri engelliyoruz.

- **Disk Güvenliđi:** 100 GB root disk alanı ile container logları ve imajları için geniř bir yer ayırıyoruz; böylece disk dolması sebebiyle node'un servis dıřı kalmasını (Node Pressure) engelliyoruz.
- **Ölçeklenebilir Yapı:** Bu donanım seviyesiyle, cluster üzerine eklenecek olan izleme araçları (Prometheus, Grafana vb.) ve diđer sistem servisleri için gerekli kaynak rezervini en baştan hazır tutuyoruz.

Launch Instance



Details

Source

Flavor

Networks *

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

Allocated

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public	
> fkm.c8m8.d10 0	8	8 GB	100 GB	100 GB	0 GB	Yes	↓

Available 38

Select one

🔍 d100

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public	
> fkm.c16m16.d1 00	16	16 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c4m16.d10 0	4	16 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c8m16.d10 0	8	16 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c16m32.d1 00	16	32 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c8m32.d10 0	8	32 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c32m32.d1 00	32	32 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c32m64.d1 00	32	64 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c16m64.d1 00	16	64 GB	100 GB	100 GB	0 GB	Yes	↑
> fkm.c32m128.d 100	32	128 GB	100 GB	100 GB	0 GB	Yes	↑

× CANCEL

← BACK

NEXT →

🔥 LAUNCH INSTANCE

Clusterimiz için oluşturduğumuz network'ü sanal makinelerimize verelim. **dc01-private-net**

Launch Instance



Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Networks provide the communication channels for instances in the cloud.

Allocated ¹

Select networks from those listed below.

	Network	Subnets Associated	Shared	Admin State	Status	
I# 1	> dc01-private-net	private-subnet	No	Up	Active	↓

Available ²

Select at least one network

Click here for filters or full text search. ×

	Network	Subnets Associated	Shared	Admin State	Status	
>	ext_net	subnet_ext	Yes	Up	Active	↑
>	rke2-private-net	rke2-private-subnet	No	Up	Active	↑

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Node'larımızın kendi aralarında güvenli bir şekilde konuşabilmesi için **dc01-internal-sec** 'i tanımlıyoruz. Bu grup, cluster içindeki servislerin birbirine erişimini garanti altına alıyor.

Launch Instance



Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Select the security groups to launch the instance in.

Allocated 1

Displaying 1 item

Name	Description
> dc01-internal-sec	

Displaying 1 item

Available 4

Select one or more

Click here for filters or full text search. ×

Displaying 4 items

Name	Description
> dc01-sec	
> rke2-sec	
> default	Default security group
> rke2-internal-sec	

Displaying 4 items

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

LoadBalancer VM'imizden yine ssh key pair oluşturarak bu VM'imiz üzerinden master ve worker'lara erişmesini sağlamamız gerekmektedir. Kubernetes node'larımızı bizzat dışarı açmayarak güvenliğini sağlamak amacıyla bu adım önemlidir.

Import Key Pair



Key Pairs are how you login to your instance after it is launched. Choose a key pair name you will recognize and paste your SSH public key into the space provided.

Key Pair Name *

dc01-bastion-ssh

Key Type *

SSH Key

Load Public Key from a file

Choose File

No file chosen

Public Key * (Modified)

Content size: 401 bytes of 16.00 KB

ssh-rsa

```
AAAAB3NzaC1yc2EAAAADAQABAAQBAQCyhXRYiRv2RoLVvaJ86E8G+G1dxDo0WUARpULs7W19+  
ab9J3z3Tt7T0Eo/kIK/t5ygRnU+FGhFA4pTVm0SoZjlq+YNkt/rpwIE+Jk1V2EKZCpqjHfZgZfoGQcaw  
xF4pnQHSXmAb3L2hlkSFTc176zoWy0xY3j+wMpG3L8DK36VPWrmLtIPB2jlbVfzlfMBa28zJ1laJlha  
Fnkf4T0D/AXaI9NCtTAu+rXBsCErQq/jl5R+DYI9b8WI7TUaR5vQc50V8fMdbR6FaJdE+YHJXW+8QF  
7ifp502g7UMM0rzFnTN4lc+M1katBxCtQL6hc/JSm0AwsaqoneBH1FdfcZljR7 root@dc01-bastion-  
lb
```

× CANCEL

↑ IMPORT KEY PAIR

Ardından bu ssh key pairimizi sanal makinemize ekleyerek devam edebiliriz.

Launch Instance



Details

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

Source

+ CREATE KEY PAIR

IMPORT KEY PAIR

Flavor

Allocated

Displaying 1 item

Networks

Name	Type	Fingerprint
------	------	-------------

Network Ports

> dc01-bastion-ssh	ssh	23:44:18:35:d6:46:88:46:2a:84:42:ef:1f:59:b6:54	↓
--------------------	-----	---	---

Displaying 1 item

Security Groups

Key Pair

Available ²

Select one

Click here for filters or full text search.

Displaying 2 items

Configuration

Name	Type	Fingerprint
------	------	-------------

Server Groups

Scheduler Hints

> alperen-macbook	ssh	da:c4:31:8d:44:a6:c7:c3:1e:c3:92:0f:ad:41:2c:5e	↑
-------------------	-----	---	---

Metadata

> bastion-ssh	ssh	5c:08:a8:79:a7:09:aa:fc:24:92:c2:9e:ee:45:5a:a6	↑
---------------	-----	---	---

Displaying 2 items

× CANCEL

← BACK

NEXT →

LAUNCH INSTANCE

Yukarıdaki bütün aşamaları tamamladıktan sonra Kubernetes Clusterimizi kurabilmek adına OpenStack altyapısının bütün komponentleriyle birlikte kurmuş olduk. Buradan sonraki kısımda LoadBalancer tarafında Haproxy kurulumu ve Node'larımız üzerine RKE2 HA kurulumunu gerçekleştireceğiz.