

## **LVS (Linux Virtual Server) Nedir?**

### **Hakan Velioglu**

İstanbul Teknik Üniversitesi, Bilgi İşlem Daire Başkanlığı, Maslak / İstanbul  
veliogluh@itu.edu.tr

**Özet:** İnternetin ucuzlaması ve erişimin kolaylaşması neticesinde internet üzerinden sunulan hizmetlere olan talep ve beklentilerde aynı oranda artmıştır. Ayrıca verilen hizmetlerin kesintiye uğramaması hemen hemen tüm sunucu servislerinin gereği haline gelmiştir. Bu gerekliliğin karşılanması ise sistem tasarımlarında kümeli yapıların kullanımını mecbur kılmıştır.

Sunucu donanımlarının kapasiteleri yeterince gelişmiş durumdadır. Fakat hizmetlerin tek bir sunucu üzerinden verilmesi yerine kümeli bir yapı kurularak birden fazla daha küçük kapsamlı sunucu ile karşılanmasının faydaları daha çoktur. En önemli faydası yedeklilik sağlamasıdır. Böylece olası bir donanım arızasında hizmetin devamı sağlanabilir. Diğer bir önemli faydası ise maliyet yönünde olmaktadır. Çok kapsamlı bir sunucunun fiyatı donanım gücü ile aynı oranda değil, daha yüksek bir oranla artmaktadır.

Bu bildiride LVS (Linux Virtual Server) yazılımı kullanılarak bir kümeli sistemin nasıl tasarlanacağı ve tasarımda nelere dikkat edilmesi gerektiği anlatılacaktır. Teknik kısımlar için ise referanslar bölümünden faydalanılabilir.

### **Giriş**

Kümeli yapı aynı hizmeti vermek için birden fazla sunucunun kurulması olarak açıklanabilir. Kullanım amacına uygun olarak farklı kümeleme yapıları kullanılabilir. Örneğin yüksek işlem gücü gerektiren bilimsel denklemlerin çözülmesi durumunda GRID yapıları oluşturularak birden fazla bilgisayarın işlem gücü birleştirilebilir. Yada verilen bir internet hizmetinin (örn: mail, web, dns vb...) kesintisiz ve hızlı bir şekilde çalışması amacı ile oluşturulmuş kümeli yapılar olabilir. Bu yazıda bizim kapsamımız hizmet kalitesini arttırmak yönündeki kümeli yapılar olacaktır.

Hizmet kalitesini arttırıcı yöndeki kümeli yapılar genel kapsam olarak iki türdür.

Yüksek erişimli kümeli mimari (Aktif-Pasif yapı) sadece servisin donanım arızalarından bağımsız olarak sürekli erişilebilir olması amacını benimsemektedir. Bu yapıda bir küme elemanı (yani sunucu) hizmet verirken, diğer

küme elemanları yedekte beklerler. Hizmet veren ve bekleyen küme elemanları birbirlerini kalp atışı (heart beat) sinyalleri ile ağ üzerinden sürekli takip ederler. Eğer hizmet veren küme elemanında bir sorun olursa kalp atışı sinyali kesilir ve yedekte bekleyen küme elemanı tüm kaynakları kendi üstüne alarak hizmet görevini kendi üstüne alır.

Yük paylaşımli kümeli mimari (Aktif-Aktif yapısında ise küme elemanlarının tümü, servise ait gelen istekleri cevaplayabilirler. Yine kalp atışı sinyalleri ile birbirlerini takip ederek servis dışı kalan küme elemanının yükü de diğer küme elemanlarına dağıtılabilir. Böylece güçzüs donanımlar bile kullanılarak hem yedeklilik hem de hizmet yükünün paylaşılması sağlanabilir.

Aktif-Aktif yapıların kurulmasındaki en genel sınırlama ortak kaynakların kullanımınıdır. Bu kaynaklardan ilki ortak disk alanıdır. Aynı hizmete ait sunucular bir şekilde aynı verileri barındırmalıdır. Bu noktada farklı çözümler bulunmaktadır.

Web hizmeti gibi disk üstündeki verilerin anlık değişmediği durumlarda RSYNC yazılımı ile diskler belirli aralıklarda senkronize edilebilir ya da Linux/UNIX sunucularında NFS veya Windows sunucularında SMB üzerinden ağ paylaşımlı diskler kullanılabilir. SAN (Storage Area Network) olan ortamlarda GFS, Lustre, Coda vb. dosya sistemleri de kullanılabilir. Bu dosya sistemlerinin ortak yanı aynı dosya sistemine aynı anda birden fazla sunucudan erişim imkanı sunmalarıdır.

Aktif-Aktif yapıların diğer kurulum sorunu ise ortak ağ adresi kullanmaları ve hizmete gelen yükün küme elemanlarına dengeli bir şekilde dağılmasını sağlamaktır. LVS bu noktadaki ihtiyaçların giderilmesinde kullanılmaktadır.

## LVS

LVS Aktif-Aktif yapılarıdaki yükün küme elemanları üzerine belirli algoritmalar kapsamında paylaştırılmasını sağlamak amacıyla geliştirilmiş bir projedir. Genel kapsamda bir tabir arandığında, transparan bir yönlendirici yazılımı olarak da tanımlanabilir.

LVS yazılımı Linux çekirdeği için yazılmış bir çekirdek modülünden ve bu modülün yapılandırmasını yapan ipvsadm isminde kullanıcı seviyesindeki bir komuttan oluşur. Modülün amacı kendi IP adresine gelen bir paketi yük dağıtma algoritmalarından seçilene bağlı olarak önceden tanımlanmış küme elemanlarından uygun olana yönlendirmektir.

## Yük Dağıtma Algoritmaları

**Round Robin (rr)** yük dağıtma algoritmasında LVS sunucusu kendisine gelen isteği küme elemanlarına sırası ile dağıtır. Örneğin birinci küme elemanına bir oturum yönlendirdikten sonra diğer oturum ikinci küme elemanına olacak şekilde devam eder. Küme elemanlarına eşit yük dağıtmak amacıyla kullanılabilir.

**Least Connection (lc)** yük dağıtma algoritması çalışma şekli açısından Round Robin algoritması ile aynıdır. Fakat bu işlemi farklı bir yazılım algoritması ile yapar. Küme elemanlarına yapılan aktif bağlantı sayılarını tutarak en az bağlantıya sahip küme elemanına gelen yeni isteği yönlendirir.

**Weighted Round Robin (wrr)** ve **Weighted Least Connection (wlc)** algoritmaları ile amaç hizmet yükünü küme elemanlarına eşitlik ilkesini bozarak dağıtmaktır. Verilen bir ağırlık katsayısı ile küme elemanlarına gidecek bağlantı dağılımları farklı kılınabilir. Örneğin ağırlık katsayısı 2 olan A küme elemanı ile ağırlık katsayısı 1 olan B küme elemanı olan bir kümede A makinesi 2 istek cevapladığı durumda B makinesine 1 istek yönlendirilir. Yani ağırlık sayısı yüksek olan daha fazla isteği cevaplar.

Küme içerisindeki sunucuların donanım yapılandırmaları eş seviyede olmaması durumunda bu ağırlıklandırma yöntemi kullanılabilir. Ayrıca Piranha yazılımı wlc algoritması kullanılması durumunda ağırlık katsayılarını sunucu yüklerine bakarak dinamik olarak ayarlamaktadır.

**Locality-Based Least-Connection (lbc)** algoritması gelen istekleri hedef adreslerine bakarak dağıtır. Daha çok yönlendirici (Proxy) sunucularını kümelemede kullanılır. Aynı hedefe giden istekler aynı küme elemanına yollanmaya çalışır. Eğer küme elemanına gelen aktif istek sayısı ağırlık katsayısını aşmış ise diğer küme elemanı da aynı hedef için kullanılır.

Lbc algoritmasının amacı önbellekli yönlendirici (caching Proxy servers) sunucu kümesi içerisinde, gelen bir isteğin önbellek tarafından karşılanma sıklığını aynı hedefe ait istekleri aynı sunucuya yönlendirerek önbellekten cevap verme oranını (cache hit rate) arttırmaktır.

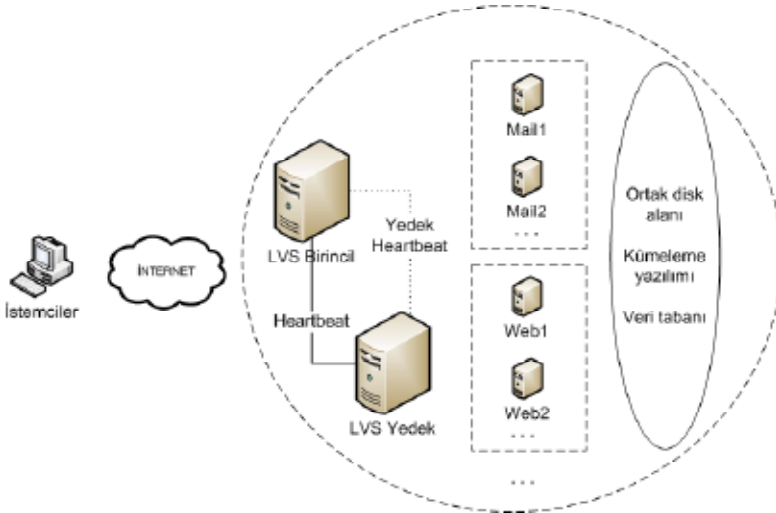
**Locality-Based Least-Connection Scheduling with Replication Scheduling (lbcrr)** algoritması da önbellekli yönlendirici sunucula-

rını kümelemede kullanılır. Lblc algoritmasından farklı olan yanı ise bir siteye giden bağlantıların sayısına bakmak yerine bağlantıların artım hızına bakarak diğer küme elemanlarına dağıtmasıdır. Böylece küme kapasitesini sonuna kadar zorlamadan çok istek alan bir sitenin daha hızlı önbelleklenmesini sağlamaktır.

**Destination Hash (dh)** algoritması da önbellekli yönlendirici sunucularının kümelenmesinde kullanılır. Elle hazırlanan bir adres listesi kullanılarak hangi siteleri hedef alan bağlantıların hangi küme elemanlarına gideceği ayarlanabilir.

Bu yöntem yedekli bir yapı kullanılabilecekse pek uygun olmamaktadır.

**Source Hash (sh)** tekniğinde gelen isteklerin, kaynak adreslerine bakılarak elle belirtilen bir yönlendirme tablosu üzerinden küme elemanlarına dağıtımı sağlanır. Bu yöntemde küme elemanlarına verilen ağırlık katsayısı o küme elemanının alabileceği bağlantı sayısının yarısını ifade etmektedir. Yani ağırlık katsayısı ile küme elemanına gidecek olan aktif bağlantılar sınırlandırılmaktadır. Sınırın aşılması durumunda ise istekler kaybedilir.



Şekil 1: LVS ile kurulmuş kümeli yapı şeması

### LVS Kullanım Yöntemleri ve Yedekleme

LVS basit anlamda OSI referans modelinin 4. katmanına kadar gelen paketleri açarak gelen oturumları küme elemanlarına dağıtmaktadır. Bu durumda LVS sunucusunun bozulması tüm küme yapısını bozan noktayı oluşturmaktadır. Bu sebeple LVS hizmeti veren makinenin (pasif) bir yedeğinin kurulması önemle tavsiye edilir.

LVS'in kendisi açısından Aktif-Pasif mimaride bir yedeği sağlamak için LVS üzerine "heartbeat" yazılımı yüklenebilir. Bu yazılım açık kaynak kodlu olup iki veya daha fazla Linux makinenin

kalp atışı sinyallerini birbirlerine yollamasını ve takip etmesini sağlar. Buna ek olarak "Idirectord" veya "mon" yazılımları da yüklenerek kalp atışı sinyallerinin kesilmesi durumunda yapılacak işlemleri yöneten yazılımlar yüklenebilir.

Bu şekilde bir kurulum yerine kendi bünyesinde bu yedekli yapı ve daha fazlasını barındıran projeler kullanılabilir. Bu kapsamda başı çeken açık kaynak kodlu projeler: Piranha, UltraMonkey ve KeepAlived çalışmalarıdır. Bu 3 projenin da yedekli olma ve LVS kullanma kısımları dışında bazı kullanım ve işleyiş farkları da bulunmaktadır.

Örneğin Piranha yazılımı Red Hat dağıtımlar ile birlikte gelmektedir. Yapılandırmalarının hepsi kendi içerisinde bulunan ufak bir web sayfası üzerinden yapılabilmektedir. Ayrıca hizmet veren küme elemanlarını protokol seviyesinde kontrol edebilmektedir. Örneğin bir web servisi veren küme elemanına http protokolü dahilinde bir istekte bulunup verdiği cevabın doğruluğuna bakarak küme elemanının çalışıp çalışmadığına veya doğru çalışmadığına karar verebilmektedir. Ayrıca sistem yöneticisinin yazacağı kendi betikleri ile de kişisel kontroller yapıları kurmasına olanak sağlamaktadır.

Piranha içerisinde bulunan diğer bir güzel özellik ise rrp ve rrp2 protokolleri üzerinden küme elemanlarının yük durumunu öğrenerek wlc algoritmasının kullanılması durumunda sunucu ağırlık katsayılarını dinamik olarak değiştirebilmesidir. Böylece sunucuların yüküne bağlı olarak gelen isteklerin dinamik olarak dengeli yapıda dağıtılması sağlanmış olur.

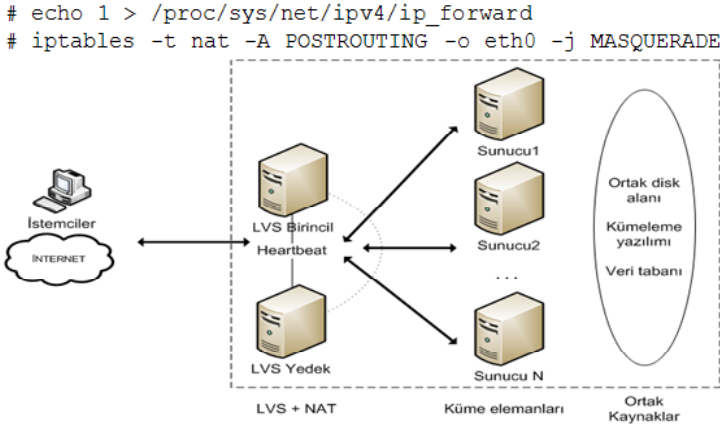
## Yönlendirme Seçenekleri

Yönlendirme seçenekleri LVS tarafından değil Linux tarafından sağlanan özellikler ile gelmektedir.

**NAT** (Network Address Translation) yönlendirme de LVS olarak ayarlanan sunucu aynı zamanda küme elemanları için NAT ağ geçit sunucusu olarak ayarlanmalıdır.

NAT yönlendirme kullanıldığında tüm küme elemanlarının trafiği LVS hizmeti veren sunucu üzerinden geçeceğinden dolayı LVS açısından çok yük getirir. Diğer yandan küme elemanları dış ağdan izole edildikleri için sunucuların korunması sağlanmış olur.

NAT yönlendirme yapılması durumunda küme elemanları üzerinde NAT yapısına uygun ağ ayarları dışında ek bir yapılandırmaya ihtiyaç duyulmaz. Bu sebeple herhangi bir işletim sistemi veya cihaz rahatlıkla kullanılabilir.



**Şekil 2:** NAT Yönlendirme tekniğinin çalışma şeması ve Linux sunucuda NAT yönlendirmesini açmak için gerekli komut

**Tunneling** (tünelleme) tekniğinde LVS kurulu olan sunucuya küme elemanlarının VPN (Virtual Private Network) benzeri bir tünel üzerinden bağlı olmasıdır. Bu kullanım NAT yönlendirmesi gibi tüm trafiğin LVS üstünden geçmesini sağladığı gibi aynı zamanda tünel üzerinde şifreleme yapılması söz konusu oldu-

ğunda NAT yönlendirmeden daha da fazla ek yük getirmektedir. Bu sebeple bu yöntem pek tercih edilmemektedir.

**Direct Routing (DR)** yöntemi ise LVS sunucusuna en az yük getiren ve dolayısı ile en çok tavsiye edilen yöntemdir.

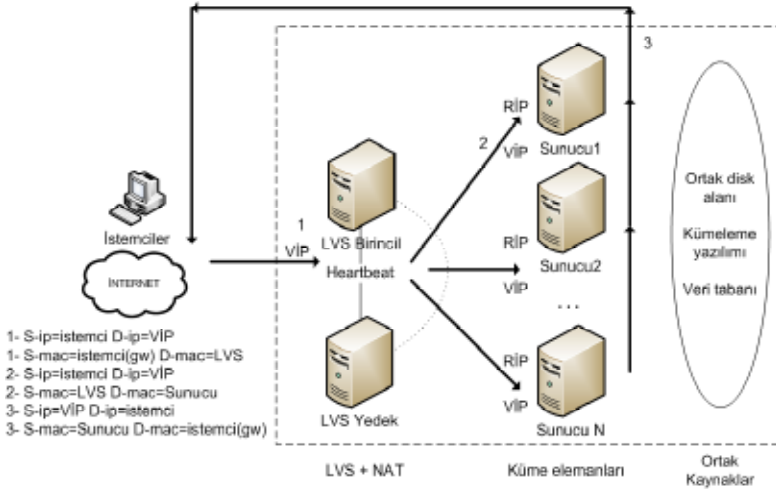
DR yönteminde LVS sunucusu istekleri aralık küme elemanlarına yönlendirir fakat küme elemanları cevapları LVS üstünden döndürmek yerine direkt istemciye yollarlar. Yöntemde verilecek hizmete ilişkin bir IP adresi belirlenir (VIP). Bu IP adresi hem LVS hizmeti verecek sunucuya hem de küme elemanlarına tanımlanmalıdır.

Aynı IP adresinin ortak bir ağ içerisinde iki makinede bulunması sorun çıkarmaktadır. Bu sorunu aşmak için farklı platformlarda farklı yöntemler uygulanmaktadır. Linux küme elemanları için arptables veya iptables yazılımlarından faydalanılabilmektedirken Windows küme elemanlarının kullanılması durumunda "Loop Back" ağ aygıtı tanımlanması gerekir.

Arptables yazılımı ile Linux işletim sistemlerindeki ARP (Address Resolution Protocol) kayıt sorguları üzerinde işlem yapılabilir. Böylece küme elemanlarına yazılacak kurallar ile küme elemanlarının kendilerine eklenen VIP adresine ilişkin ARP bilgileri yollamaları engellenebilir.

Küme elemanlarında kullanılacak örnek arptables komutları:

```
# arptables -A IN -d <VIP> -j DROP  
  
# arptables -A OUT -d <VIP> -j mangle  
--mangle-ip-s <RIP>
```



Şekil 3: Direct Routing(DR) yönlendirme tekniğinin çalışma şeması.

Arptables yönteminin kötü yanı arptables servisinin bozulması yada hata yapması durumunda küme yapısının bundan etkilenmesi ve düzgün hizmet verememesidir.

Iptables yazılımının kullanılması daha kolay ve güvenilirdir. Bu yöntemde küme elemanlarına VIP adresi tanımlanmaz. Iptables ile NAT tablosuna ait REDIRECT kuralı yazılarak hedef\_IP = VIP adresi hedef olarak gelen paketlerin hedef\_IP = RIP (RIP : Küme elemanlarının gerçek IP adresi) olarak dönüştürülmesini

sağlar. Servis tarafından işlenip dönülen cevap paketleri ise takip edilerek paketin içindeki kaynak\_IP = RIP bilgisi tekrar kaynak\_IP=VIP olarak iptables tarafından değiştirilir. Kısacası iptables sisteme tanımlı olmayan VIP adresi hedef olarak küme elemanına gelen isteklerin küme elemanı tarafından kendisine gelen normal bir paketmiş gibi işlenebilmesini sağlar.

Küme elemanları tarafından kullanılacak örnek iptables komutu:

```
# iptables -t nat -A PREROUTING -p  
<tcp_veya_udp> -d <VIP> --dport <ser-  
vis_portu> -j REDIRECT
```

Bu yöntemde iptables servisinin bozulması sadece o küme elemanını etkileyeceği ve diğer küme elemanlarının çalışmasını engelleme-yeceği için arptables kullanımından çok daha güvenilirdir.

Windows küme elemanlarının kullanılması durumunda ise iptables veya arptables benzeri yazılımların kullanımı mümkün değildir. Bunların yerine, VIP adresi sisteme ek olarak eklenen ikinci bir ağ kartına verilir. Bu ağ kartı eklenirken donanım modeli olarak “Loop Back Adapter” seçilmelidir. Böylece fiziksel ağ kartına gelen ARP isteklerine cevap verilmeyip bu ağ kartına VIP hedef olarak yollanan istekler düzgün olarak işlenebilecektir.

## Sonuç

Kümelimimariler günümüz sistemlerinin vazgeçilmez birer parçası halindedir. Bu parçada gelen ağ trafiğini küme elemanları üzerine dengeli bir şekilde dağıtabilmek önemli bir sorundur. LVS yazılımı veya LVS yazılımını kullanan diğer yazılımlar ve Linux işletim sisteminin yetenekleri ile birlikte bu soruna uygulaması kolay ve maliyeti düşük çözümler bulunabilmektedir.

## Referanslar

- [1] Cluster (computing) – Wikipedia, the free encyclopedia [http://en.wikipedia.org/wiki/Computer\\_cluster](http://en.wikipedia.org/wiki/Computer_cluster) Erişim Tarihi:21/02/2009
- [2] Bilgisayar kümesi – Vikipedi [http://tr.wikipedia.org/wiki/Bilgisayar\\_K%C3%BCmeleri](http://tr.wikipedia.org/wiki/Bilgisayar_K%C3%BCmeleri) Erişim Tarihi: 10/02/2009
- [3] The Linux Virtual Server Project – Linux Server Cluster for Load Balancing <http://www.linuxvirtualserver.org/> Erişim Tarihi: 28/01/2009
- [4] Red Hat Enterprise Linux 5 Virtual Server Administration [http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/pdf/Virtual\\_Server\\_Administration.pdf](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/pdf/Virtual_Server_Administration.pdf)