

## **NFS**

Ağ Dosya Sistemi(Network File System), 1984 yılında Sun Microsystems tarafından geliştirilen dağıtık dosya sistemidir. Kullanıcılara, ağ üzerinden, istemci bilgisayardaki dosyalara yerel bilgisayardaymış gibi erişim sağlamaya izin veren bir protokoldür. Benzer bütün protokoller gibi, Açık Ağ Hesaplama Uzak Yordam Çağırımı(Open Network Computing Remote Procedure Call ONC RPC) üzerine kuruludur.

### **Versiyonlar ve Çeşitleri**

#### **NFSv2**

Protokolün ikinci versiyonu(Mart 1989'da RFC 1094[1] tanımlandı), sadece User Datagram Protocol(UDP) üzerinde çalışıyordu. Tasarımcıları, sunucu tarafını durumsuz(stateless) çekirdek protokolün dışında kitleme(locking) ile uygulamıştır. Sanal Dosya Sistemi(Virtual File System) arayüzün modüler uygulamaya izin vermesiyle birlikte basit bir protocol olmuştur. Şubat 1986'dan itibaren, uygulamalar Eunice kullanılarak DOS, VAX/VMS gibi işletim sistemlerinde denenmiştir[2].

#### **NFSv3**

NFS üçüncü versiyonu, ikinci versiyondan yakın bir zaman sonra çıkarılmıştır.[3] Genel motivasyon ikinci versiyondaki yazma performansında yaşanan problemleri azaltmaktır. Haziran 1992'den itibaren ikinci versiyondaki çoğu problem pratikte çözülmüştür ve geriye sadece 2 GB'lık boyut kısıtlaması problemi kalmıştır. Bu problem, Ultrix'in yeni 64-bitlik RISC işlemcilerini sunmasıyla şiddetli bir probleme dönüşmüştür. Üçüncü versiyonun tanıtıldığı dönemde, taşıma-katmanında TCP protokolü yaygınlaşmaya başlamıştır. Bazı satıcılar, ikinci versiyona TCP desteği eklemesine rağmen Sun Microsystems bu desteği üçüncü versiyonda eklemiştir. TCP'yi kullanmak, NFS'i Wide Area Network(WAN) üzerinde kullanmayı daha mümkün kılmıştır ve UDP tarafından konulan 8 KB'lık okuma ve yazma limitlerini ortadan kaldırmıştır.

Üçüncü versiyonda eklenen bazı özellikler:

- \* 64-bitlik dosya boyutlarına ve 2 GB'dan büyük dosyalar ile çalışma desteklenmiştir.
- \* Sunucuda yazma performansını arttırmak için asenkron yazmalar desteklenmiştir.

#### **NFSv4**

NFS dördüncü versiyonu[4], Andrew File System(AFS) ve Server Message Block(SMB or CIFS) etkilenilmiş, performans artışları, güçlü güvenlik desteği ve durumlu(stateful) protokol tanıtılmıştır. Dördüncü versiyon, Internet Engineering Task Force(IETF) ve Sun Microsystems tarafından birlikte geliştirilmiş ilk versiyondur.

NFS 4.1 versiyonu[5] kümelenmiş sunucuların(clustered server) avantajlarını kullanarak çalışma ve paralel sistemlerde, çoklu sunucular üzerinde dağıtılmış dosyalara erişim imkanı sunmayı hedeflemiştir.

NFS 4.2 versiyonu[6] Kasım 2016'da yayınlanmış olup aşağıdaki özellikler eklenmiştir:

- \* Sunucu-taraflı çoğaltma ve kopyalama
- \* Uygulama girdi/çıkışı
- \* Seyrek dosyalar(Sparse files)

- \* Bellek rezervi
- \* Uygulama veri bloğu(ADB)

## Diğer Eklentiler

WebNFS, versiyon 2 ve versiyon 3'e uzantı/ekleme olarak yayımlanmış ve NFS'in Web tarayıcıları ile güvenlik duvarları(firewalls) aracılığıyla daha kolay entegre olmasına imkan sunmuştur. Sun Microsystems 2007 yılında istemci-terafli WebNFS uygulamasını açık kaynak olarak sunmuştur[7].

NFS ile ilişkili çeşitli protokoller:

- \* Bayt-aralıklı danışman Network Lock Manager(NLM) protokolü
- \* NFS kullanıcılarına, NFS sunucularında ne kadar veri saklama kotası kaldığını gösteren, uzaktan kota-rapor protokolü(RQUOTAD) (Örneğin; Google Drive)
- \* Taşıma olarak uzaktan direkt bellek erişiminin(remote direct memory access RDMA) adaptasyonu
- \* Güvenilir NFS[8]

## Platformlar

NFS, genel olarak Unix Unix benzeri işletim sistemleri ile birlikte kullanılır. Ayrıca MS-DOS[9], Microsoft Windows[10], Novell NetWare[11] ve IBM AS/400[12] gibi işletim sistemleri ile de kullanılabilir.

Server Message Block(SMB) ve NetWare Core Protocol(NCP) Microsoft Windows kullanan sistemlerde NFS'e kıyasla daha çok kullanılır. Apple Filling Protocol(AFP), Apple Macintosh kullanan sistemlerde NFS'e kıyasla daha çok kullanılır.

## Genel Uygulama

Unix-tarzı sistem kullanıldığı varsayılarak, bir sunucu makinenin bir başka bilgisayarda(NFS sunucu) bulunan veriye erişmek istediği senaryosu düşünüldüğünde:

1. Sunucu NFS'e yardımcı deamon işlemleri, **nfsd**'i çalıştırarak, veriyi istemcilere görünür kılmak için uygular/başlatır,
2. Sunucu yöneticisi nelerin görünür olması gerektiğine karar verir ve /etc/exports içerisinde, istenilen klasörlerin parametreleri ve isimleri kaydedilerek **exportfs** komutu ile hazırlanır.
3. Sunucu güvenlik-yöneticisi, istemcilerin tanındığından ve doğrulandığından emin olur.
4. Sunucu ağ yapılandırması güvenlik duvarı üzerinden uygun istemciler ile görüşülebildiğinden emin olur.
5. İstemci makine, **mount** komutunu kullanarak paylaşılan dosyaya erişmek için istekte bulunur(İstemci, sunucuya NFS'in hangi portu kullandığını sorar ve sonrasında NFS sunucuya bağlanır).
6. Bu adımlara kadar hata alınmazsa istemci makine, izin verilen dosyalara ağ üzerinden erişim hakkı elde eder ve kullanılan versiyona göre UDP veya TCP üzerinden haberleşme sağlanır.

## Programın Çalıştırılması

Bu bölümde demoda sunulan programların çalıştırılması ve ilgili kurulumların nasıl yapılacağı Linux işletim sistemi üzerinde anlatılmıştır.

## Önkoşullar

### MPICH2'nin kurulumu

MPICH2, Argonne National Laboratory tarafından geliştirilen MPI'nin sıklıkla kullanılan uygulamalarından biridir. MPICH2'nin son versiyonu <http://www.mpich.org/> adresinden indirilebilir. Dosya indirildikten sonra:

**tar -xzf mpich2-1.4.tar.gz** komutu ile çıkarılır ve **cd mpich2-1.4** komutu ile dosyanın içine gidilir. Bundan sonra **./configure** komutu ile indirme sırasındaki ayarlamalar yapılabilmektedir. **./configure --disable-fortran** komutu ile fortran kütüphaneleri hariç gerekli kurulumlar yapılır. Yapılandırma işlemi bittiğinde MPICH2 kurulabilir durumdadır ve **make; sudo make install** komutu ile indirme işlemi başlatılır. İndirme işlemi başarılı olarak tamamlanıp tamamlanmadığını kontrol etmek için **mpiexec --version** komutu çalıştırılır. Bu komut ile hangi versiyonun yüklendiği tespit edilir ve indirmenin durumu kontrol edilir.

### Adım 1. hosts dosyasının ayarlanması

Bilgisayarlar arasında iletişim kurulurken sürekli IP adreslerini akılda tutmamak için, ağ içerisinde iletişim kurmak istediğimiz bilgisayarları birbirine tanıtmamız gerekir. *hosts* dosyası cihaz tarafından host isimlerini IP adresleriyle eşleyen bir dosyadır. Bu dosyanın içerisine iletişim kurmak istediğimiz cihazların host isimlerini ve IP adreslerini giriyoruz. **sudo gedit /etc/hosts** komutu ile *hosts* dosyası açılır ve içerisine sisteme dahil etmek istediğimiz bilgisayarların host isimleri ve IP adresleri yazılır.

### Adım 2. Yeni kullanıcının oluşturulması

Mevcut kullanıcı hesapları ile de bilgisayar kümesi üzerinde işlem yapılabilir ancak yapılandırma işlemlerini daha basit kılmak için her bir bilgisayarda ayrı ayrı **aynı isimde** kullanıcı açılır. Her bir bilgisayar üzerinde **sudo adduser <username>** komutu ile yeni kullanıcı açılabilir.

### Adım 3. SSH ayarlanması

Bilgisayarlar ağ üzerinde SSH ile iletişim kurarlar ve NFS ile de veri paylaşımı yaparlar. Bu yüzden ilk adım olarak SSH kurulur. **sudo apt-get install openssh-server** komutu ile SSH indirilir ve 2. adımda oluşturulan kullanıcıya **su - <username>** komutu ile giriş yapılır. Artık SSH server kurulu olduğu için **ssh <username>@<hostname>** ile diğer bilgisayarlara giriş yapılabilir. Bu komuttan sonra bizden bir şifre istenir. Giriş işleminin daha kolay olması adına, anahtarlar(keys) üretip istemci bilgisayarların **authorized\_keys** dosyasına kopyalamamız gerekir.

Bu işlem için **ssh-keygen -t dsa** komutu ile anahtar üretilir ve **ssh-copy-id <hostname>** ile diğer bilgisayarın **authorized\_keys** dosyasına üretilen anahtar kopyalanır. Kopyalama işlemi her bir istemci bilgisayar ve sunucu bilgisayar(localhost) için tekrarlanır. Bu adımlar tamamlandıktan sonra **openssh-server** kurulmuş olur ve istemci bilgisayarlar ile güvenli bir şekilde iletişim sağlanır. Her bir bilgisayarın bilinen host'lara(**known\_hosts**) eklenebilmesi için bir kerelik **ssh** ile bağlanması gerekir. Bu adımdan sonra şifresiz **ssh**'in aktif edilmesi için **eval `ssh-agent`** ve **ssh-add ~/.ssh/id\_dsa** komutları çalıştırılır. Belirtilen adımların hepsini tamamladıktan sonra **ssh <hostname>** ile şifre kullanmadan güvenli bir şekilde istenilen bilgisayara bağlanılabilir.

### Adım 4. NFS ayarlanması

Master ve client bilgisayarlar arasında veri paylaşımını sağlamak adına NFS üzerinden bir klasör oluşturmamız gerekir.

Sunucu üzerinde NFS kurmak için **sudo apt-get install nfs-kernel-server** komutu çalıştırılır. Hala `<username>` kullanıcısında olduğu varsayılarak **mkdir <filename>** komutuyla paylaşılması istenen klasör `<filename>` adıyla oluşturulur. `<filename>` klasörünü dışa aktarmak için `/etc/exports` dosyasının sonuna `/home/<username>/<filename> *(rw,sync,no_root_squash,no_subtree_check)` eklenir. Burada:

**rw:** Klasör için hem yazma hem de okuma işlemine izin verildiğini,

**sync:** Yapılan değişikliklerin sadece işlendikten sonra paylaşılan klasöre uygulandığını,

**no\_subtree\_check:** Eğer paylaşılan klasör çok daha büyük bir dosya sisteminin alt klasörü ise NFS, paylaşılan klasörün üzerindeki bütün klasörleri de tarar. Bu ekleme, tarama işleminin yapılmasını engellemek istediğimizi belirtir.

**no\_root\_squash:** Root hesapların klasöre bağlanmasına izin verdiğimizizi

belirtir. Satır eklendikten sonra **exportfs -a** komutu çalıştırılır ve **sudo service nfs-kernel-server restart** komutu ile NFS sunucusu tekrardan başlatılır.

İstemci üzerinde NFS kurmak için ise `sudo apt-get install nfs-common` komutu çalıştırılır ve `<filename>` ile aynı bir dosya ismi kullanılarak `mkdir <filename>` komutu çalıştırılır. Paylaşılan klasörün bağlanması için **sudo mount -t nfs <master\_name>:/home/<username>/<filename> ~/<filename>** komutu çalıştırılır. Bu bağlamayı kalıcı yapmak için `/etc/fstab` dosyasına `<master_name>:/home/<username>/<filename> /home/<username>/<filename> nfs yazılır ve kaydedilir.`

## Adım 5. MPI programlarının çalıştırılması

Demoda anlatılan programların çalıştırılması için `mpirun -np <process_sayısı> --hosts <hostname1>,<hostname2>,<hostname3> <calistirilabilir_program>` şablonunda bir kod çalıştırılır. Bizim örneğimizde **mpirun -np 10 --hosts hakan,ibrahim,alpbey ./sum\_demo** komutu çalıştırılarak toplama örneği çalıştırılır. Burada işlemi 10 process'e bölmek istediğimizi, *hakan*, *ibrahim*, *alpbey* bilgisayarlarını işlem için kullanmak istediğimizi ve *sum\_demo* programını çalıştırmak istediğimizi belirtmiş oluyoruz.

## Referanslar

- [1] <https://tools.ietf.org/html/rfc1094>
- [2] Russel Sandberg. "The Sun Network Filesystem: Design, Implementation and Experience" *Technical Report*. Sun Microsystems.
- [3] <https://tools.ietf.org/html/rfc1813>
- [4] <https://tools.ietf.org/html/rfc3010>
- [5] <https://tools.ietf.org/html/rfc5661>
- [6] <https://tools.ietf.org/html/rfc7862>
- [7] <https://yanfs.dev.java.net/>
- [8] Glover, Fred. "A Specification of Trusted NFS (TNFS) Protocol Extensions"
- [9] ["Other Software by SUN Microsystems". www.computinghistory.org.uk](http://www.computinghistory.org.uk). The Centre for Computing History. Erişildi 30 Aralık 2018.
- [10] ["Introduction to Microsoft Windows Services for UNIX 3.5". technet.microsoft.com](http://technet.microsoft.com). Microsoft. Erişildi 30 Aralık 2018.
- [11] ["NFS Gateway for NetWare 6.5". www.novell.com](http://www.novell.com). Novell. Erişildi 30 Aralık 2018.
- [12] ["OS/400 Network File System Support" \(PDF\). publib.boulder.ibm.com](http://publib.boulder.ibm.com). IBM. Erişildi 30 Aralık 2018.