

1. GİRİŞ .....	1
2. NAGIOS NEDİR? .....	1
3. NAGIOS PLUGİN NEDİR? .....	2
4. NAGIOS KURULUMU .....	3
4.1 NAGIOS Kaynak Koddan Kurulum .....	3
4.2 NAGIOS FreeBSD Port Ağacından Kurulum .....	4
5. KURULUMUNDA YAŞANABİLECEK PROBLEMLER VE ÇÖZÜMLERİ .....	5
6. NAGIOS KURULUM DOSYALARI .....	6
6. NAGIOS YAPILANDIRMA DOSYALARI .....	6
6.1 NAGIOS yapılandırma dosyaları .....	6
6.1.1 Ana Yapılandırma Dosyası .....	6
6.1.2 Kaynak Dosyası .....	6
6.1.3 Nesne Tanımlama Dosyası .....	7
6.1.4 Cgi Yapılandırma Dosyası .....	7
7. ÖRNEK NAGIOS YAPILANDIRMASI .....	8
8. NAGIOS ÇALIŞTIRILMASI .....	21
6.1 Ön Planda Çalıştırmak Ve Hata Ayıklama .....	21
6.2 Arka Planda Çalıştırmak .....	21
6.3 Daemon Olarak Çalıştırmak .....	22
9. NAGIOS WEB ARABİRİMİNİ KULLANMAK .....	22
10. NAGIOS'DA GÜVENLİK .....	24
11. OLASI HATALAR VE ÇÖZÜMLERİ .....	24
12. KAYNAKLAR: .....	26

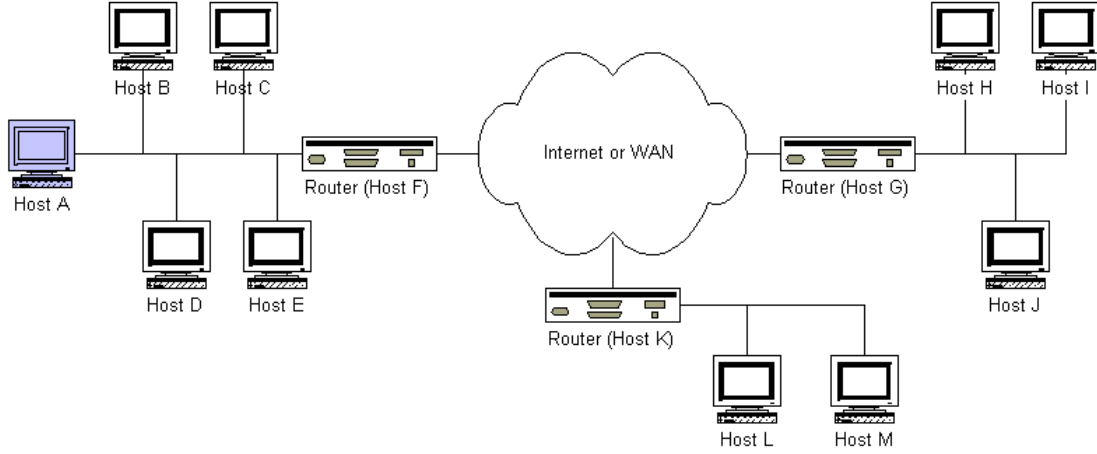
## 1. GİRİŞ

Bu dökümanda Linux/BSD sistemler üzerinde NAGIOS kurulum ve yapılandırması anlatılacaktır. Ayrıca NAGIOS'u web arabiriminden izleyebilmek için Apache kullanılacaktır. Apache kurulumu için <http://www.enderunix.org/docs/apache.pdf> bağındaki dokümana bakılabilir. Kaynak koddan kurulum esnasında NAGIOS için gerekli paketlerin */usr/local/src/NAGIOS* altına temin edildiği varsayılmıştır.

## 2. NAGIOS NEDİR?

NAGIOS sistem ve ağ denetleme uygulamasıdır. Belirtilen makine üzerinde çalışan servislerin durumunu kontrol eder ve uygulamalarda bir problem olduğunda belirtilen yöntemle kullanıcıya bildirir. Aynı segmentte yada değil NAGIOS ile belirtilen makine yada servisler kontrol edilebilir. Aşağıda buna bir örnek verilmiştir.

**Example Network Layout**  
Last Modified 5/31/1999



[http://NAGIOS.sourceforge.net/docs/2\\_0/networkreachability.html](http://NAGIOS.sourceforge.net/docs/2_0/networkreachability.html)

Burada NAGIOS A makinesinde çalışmaktadır ve ister aynı segmentte ister farklı segmentte bulunan makine yada servislerin durumunu izleyebilmektedir.

NAGIOS GPL (General Public License) lisansı ile , version 2 FSF (Free Software Foundation) lisansı ile dağıtılmaktadır .

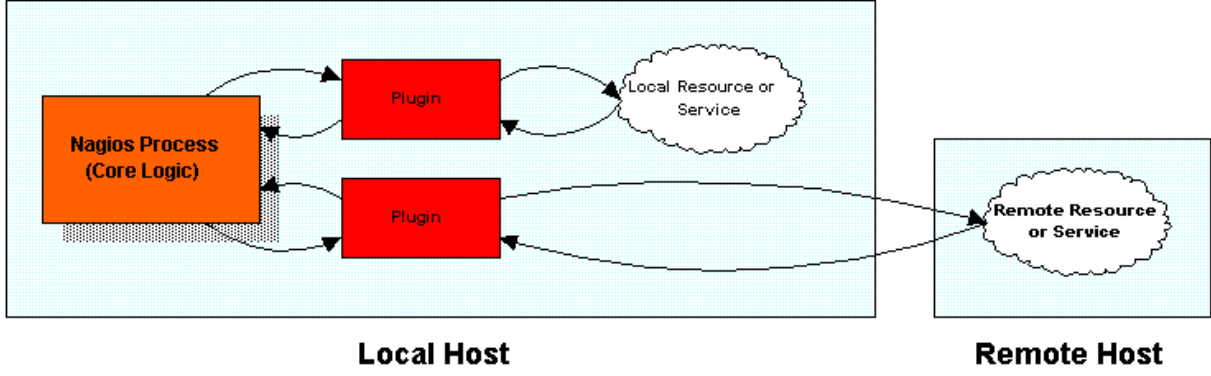
NAGIOS ile yapılabilecekler ;

- Network servisleri denetleneme (SMTP, POP3, HTTP, NNTP, PING, vs)
- Makine kaynakları denetleneme (disk kullanımı vs )
- Otomatik log döndürme (log rotation)
- Belirtilen servisler yada makineler üzerinde belirtilen durumlarda kullanıcıya çeşitli yöntemlerle uyarı verme
- Mail , SMS , Telefon , ICQ vb yöntemlerle uyarı verebilme
- Web arayüzünden makinelerin , servislerin , logların ayrıntılı takibi

olarak sıralanabilir.

### 3. NAGIOS PLUGİN NEDİR?

NAGIOS esnekliğini pluginlere borçludur. Pluginler perl , kabuk vb yazılmış komut satırından çalıştırılabilir belirtilen makine ve servislerin durumunu kontrol eder. NAGIOS pluginlerden gelen cevaba göre servislerin yada makinelerin durumlarını bildirir. Aşağıda NAGIOS'un pluginlerle birlikte kullanımı şematize edilmiştir.



[http://NAGIOS.sourceforge.net/docs/2\\_0/plugintheory.html](http://NAGIOS.sourceforge.net/docs/2_0/plugintheory.html)

Pluginler network üzerindeki makine yada servislerin durumunu kontrol eder ve NAGIOS'a bildirir. NAGIOS gelen cevaba göre bildirimlerini yada gerekli yapılandırmaya göre işlemlerini yapar.

Linux sistemlerde kaynak koddan yada rpm paketleri temin edilip kurulmalıdır. FreeBSD için port ağacından kurulumda NAGIOS-plugin de yüklenmektedir.

<http://sourceforge.net/projects/NAGIOSplug/> bağından temin edilebilir.

## 4. NAGIOS KURULUMU

NAGIOS esas olarak Linux sistemler üzerinde çalışacak şekilde dizayn edilmiştir ancak Unix sistemler üzerinde de port edilmiştir.

### 4.1 NAGIOS Kaynak Koddan Kurulum

NAGIOS kurulumunu gerçekleştirebilmek için (NAGIOS için gerekli kullanıcı ve grubu oluşturabilmek , apache ayar dosyasında gerekli değişiklikleri yapabilmek) root kullanıcı haklarına sahip olunması gerekmektedir. NAGIOS kurulumuna geçmeden önce gerekli izin oluşturulmalıdır. Burada şu anki son sürüm olan NAGIOS-2.0b5 versiyonu kullanılmıştır.

NAGIOS <http://www.NAGIOS.org/download> bağından temin edilebilir.

```
# mkdir -p /usr/local/src/NAGIOS
# cd /usr/local/src/NAGIOS
# wget http://mesh.dl.sourceforge.net/sourceforge/NAGIOS/NAGIOS-2.0b5.tar.gz
```

tar -xzf NAGIOS-versiyon-numarası.tar.gz komutu ile paket açılır.

```
# tar xzf NAGIOS-20b5.tar.gz
```

komutunun verilmesinin ardından bulunan dizinde *NAGIOS-versiyon-numarası* adında bir klasör oluşacaktır.

NAGIOS'un çalışması için gerekli kullanıcı ve grubunun oluşturulması için aşağıdaki komutlar verilir. Eğer NAGIOS port ağacından kurulacaksa kurulum esnasından gerekli kullanıcı ve grup oluşturulur.

## Linux

```
# adduser NAGIOS
# addgroup NAGIOS
```

## FreeBSD

```
# pw groupadd NAGIOS
# pw useradd NAGIOS -g NAGIOS
```

NAGIOS'un kurulumun yapılacağı dizin oluşturulmalıdır

```
# mkdir /usr/local/NAGIOS
```

Oluşturulan dizin için gerekli izinleri vermek için.

```
# chown NAGIOS:NAGIOS /usr/local/NAGIOS
```

Aşağıda NAGIOS'un derlenmesi ve derleme seçenekleri anlatılmıştır.

```
# cd /usr/local/src/NAGIOS
# ./configure --prefix=/usr/local/NAGIOS
--with-cgiurl=/NAGIOS/cgi-bin --with-htmurl=/NAGIOS/ --with-NAGIOS-user=NAGIOS
--with-NAGIOS-grp=NAGIOS
```

```
--prefix=/usr/local/NAGIOS      : NAGIOSun kurulacağı dizin belirtilir
--with-cgiurl=/NAGIOS/cgi-bin   : NAGIOS için cgi-bin dizini
--with-htmurl=/NAGIOS/         : NAGIOS için html dizini
--with-NAGIOS-user=NAGIOS      : NAGIOS'un çalışacağı kullanıcı
--with-NAGIOS-grp=NAGIOS       : NAGIOS'un çalışacağı grup
```

```
# make all
# make install
# make install-init
```

## 4.2 NAGIOS FreeBSD Port Ağacından Kurulum

NAGIOS'u FreeBSD'de port ağacından kurmadan önce port ağacı güncellenmelidir.

```
# pkg_add -rv cvsup-without-gui
# cp /usr/share/examples/cvsup/ports-supfile /etc/
# vi /etc/ports-supfile
```

```
*default host=CHANGE_THIS.FreeBSD.org olan satırın  
*default host=cvsup.tr.freebsd.org şeklinde değiştirilmesi gerekir.
```

```
# cvsup -g -L 2 /etc/ports-supfile
```

NAGIOSu port ağacında `/usr/ports/net-mgmt/NAGIOS` bulunur. NAGIOS'u port ağacından kurmak için aşağıdaki komutlar verilir. NAGIOS için gerekli paketler (plugin vs) indirilip sisteme kurulacaktır.

```
# cd /usr/ports/net-mgmt/NAGIOS  
# make install clean
```

NAGIOS için gerekli kullanıcı ve grubun oluşturulması için gelen sorulara “y” yada “<Enter>” ile izin verilir.

```
You need a group "NAGIOS".  
Would you like me to create it [y]?
```

```
You need a group "NAGIOS".  
Would you like me to create it [y]?
```

NAGIOS'un açılışta başlayabilmesi için `/etc/rc.conf` dosyasında `NAGIOS_enable="YES"` satırının olması gerekir.

```
# echo NAGIOS_enable="YES" >> /etc/c.conf
```

## 5. KURULUMUNDA YAŞANABİLECEK PROBLEMLER VE ÇÖZÜMLERİ

NAGIOS `snprintf()`, `strncat()`, and `strncpy()` fonksiyonlarına ihtiyaç duyar. Eğer kurulum yapılacak makinede bu C fonksiyonları mevcut değilse NAGIOS kurulumu başarısızlıkla sonuçlanacaktır.

NAGIOS `gd` kütüphanesi yüklü olmasına rağmen derlenmiyorsa `gd` kütüphanesini sistemden kaldırılır `-disable-shared` seçeneğiyle derlenir ve tekrar sisteme kurulur. Yada `libgd.so.2.0.0`, `libgd.so` ve `libgd.so.2` sistemden silinir.

## 6. NAGIOS KURULUM DOSYALARI

*/usr/local/bin/NAGIOS*: NAGIOS için gerekli ana program binaryleri

*/usr/local/share/NAGIOS*: Uyarı mesajlarının bulunduğu dizin

*/usr/local/etc/NAGIOS*: Konfigürasyon dosyalarının bulunduğu dizin

*/usr/local/libexec/NAGIOS*: NAGIOS için gerekli pluginlerin bulunduğu dizin

*/usr/local/www/NAGIOS*: Web arayüzü için gerekli html ve cgi dosyalarının bulunduğu dizin

*/var/spool/NAGIOS*: Nagis loglarının tutulduğu dizin

*/var/spool/NAGIOS/archives*: Dönderilen NAGIOS loglarının arışvlendiği dizin

## 6. NAGIOS YAPILANDIRMA DOSYALARI

NAGIOS konfigürasyonu tek bir dosyada yapılabildiği (*minimal.cfg* yada *NAGIOS.cfg* dosyasında belirtilen bir isimde dosya) gibi bütün tanımlamalar için ayrı ayrı dosyalar da kullanılabilir(*services.cfg* , *hosts.cfg* vs )

### 6.1 NAGIOS yapılandırma dosyaları

- Ana Yapılandırma Dosyası (Main Configuration File)
- Kaynak Dosyası (Resources Configuration File)
- Nesne Tanımlama Dosyası (Object Definition File)
- CGI Yapılandırma Dosyası (CGI Configuration File)

#### 6.1.1 Ana Yapılandırma Dosyası

NAGIOS ana yapılandırma dosyası (genellikle */usr/local/etc/NAGIOS.cfg*) olan *NAGIOS.cfg* , NAGIOS'un nasıl çalışacağı hakkında direktiflerin bulunduğu dosyadır. Yapılandırma dosyalarının (kaynak dosyası, nesne tanımlama dosyası vs)yeri *NAGIOS.cfg* dosyasında belirtilir. NAGIOS'da yapılandırma dosyaları tek bir dosyada tanımlanacağı gibi ayrı ayrı dosyalarda da tanımlama yapılabilir. Bu dosya yada dosyaların yeri *NAGIOS.cfg* dosyasında belirtilmelidir. Burada tanımlamalar ayrı ayrı dosyalarda yapılacaktır. Basit bir uygulama için tek bir yapılandırma dosyası düşünülebilir ancak büyük bir uygulama için tanımlamalar fazla miktarda olacağı için ayrı ayrı tanımlama dosyaları kullanmak daha mantıklı olacaktır.

#### 6.1.2 Kaynak Dosyası

NAGIOS'u güçlü kılan özelliklerinden bir tanesi de komut tanımlamalarında (command definition)makro kullanımındır .Kaynak dosyası (*resources file*) kullanıcı tarafından tanımlanan macroların bulunduğu dosyadır. Macro tanımlarından başka veritabanı bağlantı ayarları (NAGIOS kurulumuna göre değişir) gibi bilgilerde kaynak dosyasında (resource file) tanımlanır. Ana yapılandırma dosyasında birden fazla kaynak dosyası (resource file) tanımlaması yapılabilir.

### 6.1.3 Nesne Tanımlama Dosyası

Nesne tanımlama dosyası (Object Definition File) hosts , services , command vs gibi tanımlamaların yapıldığı dosyadır. *NAGIOS.cfg* dosyasında tanımlamaya göre tam yolu belirtilen dosyadır.

```
..  
cfg_file=/usr/local/etc/NAGIOS/contactgroups.cfg  
cfg_file=/usr/local/etc/NAGIOS/contacts.cfg  
.. vs
```

### 6.1.4 Cgi Yapılandırma Dosyası

CGI tanımlamalarının yapıldığı dosyadır

Bütün bu yapılandırma dosyaları için örnekler */NAGIOS\_tam\_yolu/NAGIOS* (genellikle */usr/local/etc/NAGIOS*) dizini altında **dosya\_adi.cfg-sample** şeklinde bulunur. Burada bu dosyalar üzerinde değişiklikler yapılarak tanımlamalar yapılmıştır. Öncelikle bu dosyaların **dosya\_adi.cfg** şekline getirilmesi gerekmektedir. Bu işlem her dosya için teker teker yapılabileceği gibi aşağıdaki ufak betiklede yapılabilir.

```
for i in *cfg-sample; do mv $i  
`echo $i | sed -e s/cfg-sample/cfg/`; done;
```

Yapılandırmalar senaryo üzerinde gerçekleştirilecektir. Önce senaryo verilecek daha sonra bu senaryo için NAGIOS yapılandırma dosyalarında yapılması gereken değişiklikler ve etkileri anlatılacaktır. Birinci olarak NAGIOS ile izlenmek istenen makineler ve çalışan servisler hakkında bilgi verilecektir.

İp adresi:192.168.0.1  
Hostname:www.denememakinesi.com.tr  
Çalışan servisler: ftp , http , pop

İp adresi : 192.168.0.2  
Hostmame:www.denememakinesi2.com.tr  
Çalışan servisler:dns , ssh

İp adresi : 192.168.0.3  
Hostmame:www.denememakinesi3.com.tr  
Çalışan servisler: ftp

İp adresi : 192.168.0.4  
Hostname:www.denememakinesi4.com.tr  
Çalışan servisler: http , pop , smtp ,ssh

İp adresi : 192.168.0.5  
Hostname:www.denememakinesi5.com.tr  
Çalışan servisler: http

İlk 3 makine bsd ,4. makine Linux ve son makinede solaris olarak gruplandırılacak (hostgroup). bsd grubuna dahil olanlar için bsd iletişim grubu , linux grubuna dahil olanlar için Linux iletişim grubu son olarak da solaris grubuna dahil olanlar içinde solaris iletişim grubu oluşturulacaktır.

Bu yapı üzerinde NAGIOS'un çalışması için gereken yapılandırmalar için gerekli bölümler anlatılacaktır. Öncelikle NAGIOS'un ana yapılandırma dosyasında nasıl bir konfigürasyon yapılacağına karar verilmelidir. Yapılandırma tek bir dosyada yapılacağı gibi (servisler , hostlar vs tek bir dosyada tanımlanabilir) ayrı ayrı dosyalarda da tanımlamalar yapılabilir. İzlenecek makine yada servis sayısı çok fazla ise tek bir dosyadan yapılandırma yapmak NAGIOS yönetimini zorlaştıracaktır. Bunun için tanımlamaları ayrı ayrı dosyalarda yapmak daha mantıklıdır.

NAGIOS kurulumu ile ön tanımlı gelen örnek dosyalar üzerinde önemli görülen değişiklikler yapılarak anlatılacaktır. Diğer değerler seçimlik olup isteğe göre değiştirilebilir. NAGIOS için ana yapılandırma dosyası *NAGIOS.cfg* dir. Öncelikle bu dosyada gerekli yapılandırma dosyalarının yerleri belirtilmelidir. Yapılandırma dosyalarında kullanılan ” # “işareti açıklama satırını ifade etmektedir NAGIOS tarafından dikkate alınmaz

## 7. ÖRNEK NAGIOS YAPILANDIRMASI

### NAGIOS.CFG

```
cfg_file=/usr/local/etc/NAGIOS/checkcommands.cfg
cfg_file=/usr/local/etc/NAGIOS/misccommands.cfg
cfg_file=/usr/local/etc/NAGIOS/contactgroups.cfg
cfg_file=/usr/local/etc/NAGIOS/contacts.cfg
cfg_file=/usr/local/etc/NAGIOS/hostgroups.cfg
cfg_file=/usr/local/etc/NAGIOS/hosts.cfg
cfg_file=/usr/local/etc/NAGIOS/services.cfg
cfg_file=/usr/local/etc/NAGIOS/timeperiods.cfg
resource_file=/usr/local/etc/NAGIOS/resource.cfg

log_file=/var/spool/NAGIOS/NAGIOS.log
NAGIOS_user=NAGIOS
NAGIOS_group=NAGIOS
```

NAGIOS yapılandırması için gerekli yapılandırma dosyalarının yerleri , NAGIOS loglarının tutulacağı dosya ve NAGIOS'un çalışması için gerekli kullanıcı/grup tanımlanmıştır.



## CHECKCOMMANDS.CFG

```
define command{
    command_name    check_smtp
    command_line    $USER1$/check_smtp -H $HOSTADDRESS$
}

define command{
    command_name    check_http
    command_line    $USER1$/check_http -H $HOSTADDRESS$
}

define command{
    command_name    check_pop
    command_line    $USER1$/check_pop -H $HOSTADDRESS$
}

define command{
    command_name    check_dns
    command_line    $USER1$/check_dns -H www.yahoo.com -s $HOSTADDRESS$
}

define command{
    command_name    check_ssh
    command_line    $USER1$/check_ssh -H $HOSTADDRESS$
}

define command{
    command_name    check_ftp
    command_line    $USER1$/check_ftp -H $HOSTADDRESS$
}

define command{
    command_name    check-host-alive
    command_line    $USER1$/check_ping -H $HOSTADDRESS$ -w 3000.0,80% -c 5000.0,100% -p 1
}
```

**check-host-alive** ile belirtilen adresdeki makinenin durumunu kontrol edilir. **command\_name** ile belirtilen isim */usr/local/libexec/NAGIOS* dizini altında belirtilen dosyaya referans verir. Her komutun ayrı ayrı değişkeni olabilir. Burada **-H -w** gibi değişkenler almıştır. Gerekli bilgi **-h** yada **-help** parametresiyle öğrenilebilir. Örneğin **check\_ssh** komutunun alacağı parametreleri öğrenmek için

```

#/usr/local/libexec/NAGIOS/check_ssh -help
...
Usage: check_ssh [-46] [-t <timeout>] [-r <remote version>] [-p <port>]
<host>

Options:
-h, --help
    Print detailed help screen
-V, --version
    Print version information
-H, --hostname=ADDRESS
    Host name or IP Address
-p, --port=INTEGER
    Port number (default: 22)
-4, --use-ipv4
    Use IPv4 connection
-6, --use-ipv6
    Use IPv6 connection
-t, --timeout=INTEGER
    Seconds before connection times out (default: 10)
-r, --remote-version=STRING
    Warn if string doesn't match expected server version (ex:
OpenSSH_3.9p1)
-v, --verbose
    Show details for command-line debugging (NAGIOS may truncate output)
...
#

```

## MISCCOMMANDS.CFG

```

define command{
    command_name    notify-by-email
    command_line    /usr/bin/printf "%b" "***** NAGIOS
*****\n\nNotification Type: $NOTIFICATIONTYPE$\n\nService:
$SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\nState:
$SERVICESTATE$\n\nDate/Time: $LONGDATETIME$\n\nAdditional
Info:\n\n$SERVICEOUTPUT$" | /usr/bin/mail -s "*** $NOTIFICATIONTYPE$ alert -
$HOSTALIAS$/ $SERVICEDESC$ is $SERVICESTATE$ *" $CONTACTEMAIL$
}

define command{
    command_name    host-notify-by-email
    command_line    /usr/bin/printf "%b" "***** NAGIOS
*****\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState:
$HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time:
$LONGDATETIME$\n" | /usr/bin/mail -s "Host $HOSTSTATE$ alert for
$HOSTNAME$!" $CONTACTEMAIL$
}

```

**NOT:** command\_name ile başlayıp \$CONTACTEMAIL\$ ile biten satır tek bir satırdır.

*misccommands.cfg* dosyasında bildirimlerin nasıl yapılacağı tanımlanmıştır. Belirtilen servislerde yada makinelerde belirtilen durumlarda kullanıcıya gönderilecek mail içeriğinde belirtilmektedir.

## CONTACTS.CFG

```
define contact{
    contact_name          linux
    alias                 System Admin
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email                linux@mailadresisi.com.tr
}

define contact{
    contact_name          bsd
    alias                 System Admin
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email                bsd@mailadresisi.com.tr
}

define contact{
    contact_name          solaris
    alias                 System Admin
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email                solaris@mailadresisi.com.tr
}
```

*contacts.cfg* dosyasında belirtilen durumlarda mail yolu ile bildirim yapılacak kişiler tanımlanmaktadır. Burada dikkat edilmesi gereken husus **contact\_name** parametresine verilecek olan değer *contactgroups.cfg* dosyasında iletişim grupları oluşturulurken kullanılacak olmasıdır. NAGIOS yapılandırma dosyaları kapsama mantığına göre düzenlenir. İletişim için oluşturulan *contacts.cfg* dosyasındaki değer iletişim grupları oluşturmak için *contactgroups.cfg* dosyasında tanımlanır ve bu değerde istenilen servisi tanımlamak için diğer yapılandırma dosyalarında tanımlanır. Bu şekilde değerler birbirine aktarılır.

*contacts.cfg* → *contactgroups.cfg* şematize edilebilir. Aynı şekilde *hosts.cfg* dosyasında yapılan tanım *hostgroups.cfg* dosyasına referans olarak iletilir. *hosts.cfg* → *hostgroups.cfg*

## CONTACTGROUPS.CFG

```
define contactgroup{
    contactgroup_name    linux-admins
    alias                 Linux Mail Admin
    members               linux
}

define contactgroup{
    contactgroup_name    bsd-admins
    alias                 Bsd Mail Admin
    members               bsd
}

define contactgroup{
    contactgroup_name    solaris-admins
    alias                 Solaris Mail Admin
    members               solaris
}
```

**contact.cfg** dosyasında **contact\_name** parametresiyle verilen değerler burada gruplar oluşturulurken **members** değerine referans olarak verilir. Burada olduğu gibi *contact.cfg* dosyasında “**contact\_name solaris**” olarak tanımlanan değer *contactgroups.cfg* dosyasında **members** parametresine aynen aktarılmıştır.

## HOSTS.CFG

```
define host{
    name                generic-host
    notifications_enabled    1
    event_handler_enabled    1
    flap_detection_enabled    1
    failure_prediction_enabled    1
    process_perf_data        1
    retain_status_information    1
    retain_nonstatus_information    1
    register                0
}
```

```

define host{
    use                generic-host                ;
    host_name          freebsd
    alias              --
    address             192.168.0.1
    check_command      check-host-alive
    max_check_attempts 10
    notification_interval 10
    notification_period 24x7
    notification_options d,r
    contact_groups     bsd-admins
}

define host{
    use                generic-host                ;
    host_name          openbsd
    alias              --
    address             192.168.0.2
    check_command      check-host-alive
    max_check_attempts 10
    notification_interval 10
    notification_period 24x7
    notification_options d,r
    contact_groups     bsd-admins
}

define host{
    use                generic-host                ;
    host_name          linux
    alias              --
    address             192.168.0.3
    check_command      check-host-alive
    max_check_attempts 10
    notification_interval 10
    notification_period 24x7
    notification_options d,r
    contact_groups     linux-admins
}

define host{
    use                generic-host                ;
    host_name          solaris
    alias              --
    address             192.168.0.4
    check_command      check-host-alive
    max_check_attempts 10
    notification_interval 10
    notification_period 24x7
    notification_options d,r
    contact_groups     solaris-admins
}

```

```

define host{
    use                generic-host      ;
    host_name          netbsd
    alias              --
    address             192.168.0.5
    check_command       check-host-alive
    max_check_attempts 10
    notification_interval 10
    notification_period 24x7
    notification_options d,r
    contact_groups      bsd-admins
}

```

Başta tanımlanan **generic-host** bütün tanımlamalar için geçerlidir ve olması zorunludur. Aynı şekilde *services.cfg* dosyasında da **generic-service** tanımının yapılması gereklidir.

*hosts.cfg* dosyasında izelenmek istenen makineler tanımlanmaktadır. Burada dikkat edilmesi gereken bazı noktalar bulunmaktadır. **host\_name** parametresiyle verilen değer diğer yapılandırma dosyalarına referans verilmektedir. İstenilen bir isim verilebilir **alias** ile de daha açıklayıcı isimler verilebilir. **address** değeri ile de izlenecek makinenin ip adresi belirtilmiştir. **contact\_groups** değeri ile de iletilerin gönderileceği grup belirtilmiştir. Mailler bu gruba dahil kullanıcılara iletilecektir.

## HOSTGROUPS.CFG

```

define hostgroup{
    hostgroup_name    bsd
    alias             BSD
    members           freebsd , netbsd , openbsd
}

define hostgroup{
    hostgroup_name    linux
    alias             LINUX
    members           linux
}

define hostgroup{
    hostgroup_name    solaris
    alias             SOLARIS
    members           solaris
}

```

NAGIOS ile makine grupları oluşturulabilir. Web arabiriminden izlendiğinde oluşturulan gruplar izlenebilir.

## Service Overview For All Host Groups

BSD (bsd)				LINUX (linux)				SOLARIS (solaris)			
Host	Status	Services	Actions	Host	Status	Services	Actions	Host	Status	Services	Actions
freebsd	UP	3 OK		linux	UP	2 OK		solaris	UP	1 OK	
netbsd	UP	1 OK									
openbsd	UP	4 OK									

## CGI.CFG

```
main_config_file=/usr/local/etc/NAGIOS/NAGIOS.cfg
physical_html_path=/usr/local/share/NAGIOS
url_html_path=/NAGIOS
use_authentication=1
```

## RESOURCES.CFG

```
$USER1$=/usr/local/libexec/NAGIOS
```

## SERVICES.CFG

```
define service{
    name                generic-service ;
    active_checks_enabled 1 ;
    passive_checks_enabled 1 ;
    parallelize_check 1 ;
    obsess_over_service 1 ;
    check_freshness 0 ;
    notifications_enabled 1 ;
    event_handler_enabled 1 ;
    flap_detection_enabled 1 ;
    failure_prediction_enabled 1 ;
    process_perf_data 1 ;
    retain_status_information 1 ;
    retain_nonstatus_information 1 ;
    register 0 ;
}
```

```

define service{
    use                generic-service ;
    host_name          frebsd
    service_description POP
    is_volatile        0
    check_period        24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     linux-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_pop
}

define service{
    use                generic-service ;
    host_name          frebsd
    service_description HTTP
    is_volatile        0
    check_period        24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     bsd-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_http
}

define service{
    use                generic-service ;
    host_name          frebsd
    service_description FTP
    is_volatile        0
    check_period        24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     bsd-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_ftp
}

define service{
    use                generic-service ;
    host_name          openbsd
    service_description POP
    is_volatile        0
    check_period        24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     bsd-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_pop
}

```



```

define service{
    use                generic-service    ;
    host_name          openbsd
    service_description SMTP
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     bsd-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_smtp
}

define service{
    use                generic-service    ;
    host_name          openbsd
    service_description HTTP
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     bsd-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_http
}

define service{
    use                generic-service    ;
    host_name          openbsd
    service_description SSH
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     bsd-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_ssh
}

define service{
    use                generic-service    ;
    host_name          linux
    service_description DNS
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     linux-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_dns
}

```

```

define service{
    use                generic-service    ;
    host_name          linux
    service_description SSH
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     linux-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_ssh
}

define service{
    use                generic-service    ;
    host_name          solaris
    service_description HTTP
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     solaris-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_http
}

define service{
    use                generic-service    ;
    host_name          netbsd
    service_description FTP
    is_volatile        0
    check_period       24x7
    max_check_attempts 2
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     linux-admins
    notification_options w,u,c,r
    notification_interval 10
    notification_period 24x7
    check_command      check_ftp
}

```

**host\_name** ile verilen değer *hosts.cfg* dosyasında belirtilen değer olmalıdır. *hosts.cfg* dosyasında belirtilen değerler *services.cfg* dosyasına referans olarak kullanılır.

## TIMEPERIODS.CFG

```
define timeperiod{
    timeperiod_name 24x7
    alias            24 Hours A Day, 7 Days A Week
    sunday          00:00-24:00
    monday          00:00-24:00
    tuesday         00:00-24:00
    wednesday       00:00-24:00
    thursday        00:00-24:00
    friday          00:00-24:00
    saturday        00:00-24:00
}
```

Burada çalışma saatlerini temsil eden *timeperiods.cfg* dosyası belirtilmiştir.

```
define timeperiod{
    timeperiod_name workhours
    alias            "Normal" Working Hours
    monday          09:00-17:00
    tuesday         09:00-17:00
    wednesday       09:00-17:00
    thursday        09:00-17:00
    friday          09:00-17:00
}
```

Burada çalışma saatleri dışındaki saatleri temsil eden *timeperiods.cfg* dosyası belirtilmiştir.

```
define timeperiod{
    timeperiod_name nonworkhours
    alias            Non-Work Hours
    sunday          00:00-24:00
    monday          00:00-09:00,17:00-24:00
    tuesday         00:00-09:00,17:00-24:00
    wednesday       00:00-09:00,17:00-24:00
    thursday        00:00-09:00,17:00-24:00
    friday          00:00-09:00,17:00-24:00
    saturday        00:00-24:00
}
```

Burada ise çalışma saatleri dışındaki saatleri temsil eden *timeperiods.cfg* dosyası belirtilmiştir. İsteğe göre *timeperiods.cfg* dosyası düzenlenebilir.

## Host Status Details For All Host Groups

Host ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Status Information
treebsd	UP	12-02-2005 09:33:44	0d 7h 10m 27s	PING OK - Packet loss = 0%, RTA = 0.30 ms
linux	UP	12-02-2005 09:34:14	0d 7h 10m 0s	PING OK - Packet loss = 0%, RTA = 0.25 ms
netbsd	UP	12-02-2005 09:37:54	0d 7h 6m 22s	PING OK - Packet loss = 0%, RTA = 0.09 ms
openbsd	UP	12-02-2005 09:34:44	0d 7h 9m 33s	PING OK - Packet loss = 0%, RTA = 0.37 ms
solaris	UP	12-02-2005 09:37:04	0d 7h 7m 16s	PING OK - Packet loss = 0%, RTA = 0.33 ms

5 Matching Host Entries Displayed

Yapılandırma dosyaları ayarlandıktan sonra yapılandırma dosyalarının kontrolünü gerçekleştirmek için `/NAGIOS_tam_yol -v`  
`/NAGIOS_ana_yapilandirma_dosyasinin_tam_yolu` şeklinde bir komut ile bu işlem gerçekleştirilebilir.

```
# /usr/local/bin/NAGIOS -v /usr/local/etc/NAGIOS/NAGIOS.cfg
```

Burada NAGIOSun tam yolu yazılmalıdır kuruluma görede farklı bir dizinde olabilir. Bu komut çalıştırılmadan önce *“whereis”* komutuyla tam yolu öğrenilmelidir.

```
# whereis NAGIOS
NAGIOS: /usr/local/bin/NAGIOS
#
```

`/usr/local/etc/NAGIOS/NAGIOS.cfg` ilede NAGIOSu’un ana yapılandırma dosyasının tam yolu belirtilmelidir. Bu komutun çıktısı yapılandırma dosyalarında yapılmış hata ve sebepleri olacaktır. Bu doğrultuda yapılandırma dosyaları kontrol edilip hatalar giderilebilir. Herşey yolunda gittiye eğer buna benzer bir çıktı alınacaktır.

```
...
Total Warnings: 0
Total Errors: 0
```

Things look okay - No serious problems were detected during the pre-flight check

Bunun dışında NAGIOS.cfg dosyasında belirtilen log dosyasındanda hatalar takip edilebilir. Öntanımlı olarak bu dosyanın yeri `/var/spool/NAGIOS/NAGIOS.log` dur. *“NAGIOS.cfg”* dosyasında aşağıdaki şekilde tanımlanır.

```
log_file=/var/spool/NAGIOS/NAGIOS.log
```

```
# tail /var/spool/NAGIOS/NAGIOS.log
```

komutu ile NAGIOS tarafından üretilen loglar takip edilebilir.

Yapılandırma dosyalarının tanımlanmasından sonra NAGIOS başlatılabilir. NAGIOS çeşitli şekillerde başlatılabilir.

## 8. NAGIOS ÇALIŞTIRILMASI

### 6.1 Ön Planda Çalıştırmak Ve Hata Ayıklama

NAGIOS ön planda çalıştırıldığında yapılandırma dosyalarında bir hata varsa eğer bunları ekranda gösterir. Bu şekilde yapılandırma dosyalarında meydana gelecek hatalar için hata ayıklama yöntemi olarak da kullanılabilir. İstenirse çıktısı bir dosyaya yönlendirilebilir ve daha sonra bu dosyadan hatalar izlenebilir. Bu şekilde sorunsuz bir şekilde çalıştırıldığında aynı kabuktan başka işlem yapılamaz. Genellikle NAGIOS'u başlatmak için deil hata ayıklama için tercih edilir. Çalıştırmak için `/NAGIOS_tam_yolu` `/NAGIOS_ana_yapılandırma_dosyası`

```
# /usr/local/bin/NAGIOS /usr/local/etc/NAGIOS/NAGIOS.cfg
```

ekran çıktılarını bir dosyaya yönlendirmek içinde

```
# /usr/local/bin/NAGIOS /usr/local/etc/NAGIOS/NAGIOS.cfg > NAGIOS_log
```

komutu işletilebilir. Bulunulan dizinde “*NAGIOS\_log* “ adından bir dosya oluşacaktır. Bu dosyada varsa hata mesajları da bulunacaktır.

Durdurmak için CTRL + C tuş takımı kullanılabilir. Yada ps komutuyla çalışan NAGIOS süreçleri öldürülür.

```
# ps -aux | grep 'NAGIOS'
```

komutu ile NAGIOS pid'si öğrenilip “*kill*” komutuyla çalışan sürece son verilir.

```
# kill -9 NAGIOS_pid_no
```

### 6.2 Arka Planda Çalıştırmak

NAGIOS'u arka planda çalıştırmak için “&” parametresi kullanılabilir. Bu şekilde yeni kabuk açmadan aynı kabukla çalışılabilir.

```
# /usr/local/bin/NAGIOS /usr/local/etc/NAGIOS.cfg &
```

“*jobs*” komutuylada arka planda çalışan süreçler öğrenilebilir.

```
# jobs
[1]+  Running                  /usr/local/bin/NAGIOS
/usr/local/etc/NAGIOS/NAGIOS.cfg & (wd: /usr/local/libexec/NAGIOS)
#
```

## 6.3 Daemon Olarak Çalıştırmak

NAGIOS'u daemon olarak çalıştırmak için “-d” parametresi kullanılabilir.  
`/NAGIOS_tam_yolu /NAGIOS_ana_yapilandirma_dosyası_tam_yolu`

```
# /usr/local/bin/NAGIOS -d /usr/local/etc/NAGIOS.cfg
```

komutu ile NAGIOS daemon olarak çalıştırılır.

**NOT: FreeBSD İşletim sisteminde başlangıç betiği çalıştırılarak NAGIOS başlatılabilir.**

```
# /usr/local/etc/rc.d/NAGIOS.sh start
```

## 9. NAGIOS WEB ARABİRİMİNİ KULLANMAK

NAGIOS web arabiriminden izlemek için Apache yada başka bir web server kurulu olması gerekmektedir. Bunun yanında ayrıca gd kütüphanesinin de kurulu olması gerekmektedir. Apache ve gd kurulumu için <http://www.enderunix.org/docs/apache.pdf> bağındaki dokümana bakılabilir. Burada Apache için gerekli değişiklikler anlatılacaktır.

```
ScriptAlias /NAGIOS/cgi-bin "/usr/local/galkan/cgi-bin/"
```

```
<Directory "/usr/local/galkan/cgi-bin">  
Options          ExecCGI  
AllowOverride    None  
Order            allow,deny  
Allow           from all  
AuthName        "NAGIOS Access"  
AuthType        Basic  
AuthUserFile    /usr/local/etc/NAGIOS/htpasswd.users  
Require         valid-user  
</Directory>
```

```
Alias /NAGIOS "/usr/local/www/data/NAGIOS"
```

```
<Directory "/usr/local/www/data/NAGIOS/share">  
Options          None  
AllowOverride    None  
Order            allow,deny  
Allow           from all  
AuthName        "NAGIOS Access"  
AuthType        Basic  
AuthUserFile    /usr/local/etc/NAGIOS/htpasswd.users  
Require         valid-user  
</Directory>
```

“**ScriptAlias /NAGIOS/cgi-bin**” *"/usr/local/galkan/cgi-bin/"* ile NAGIOS kurulumunda “**--with-cgiurl**” deęeri ile belirtilen dizine eriřim alias tanımlanmıřtır.

**Alias /NAGIOS** *"/usr/local/www/data/NAGIOS"* ilede “**--with-htmurl**” deęiřkeni ile belirtilen dizin ii alias tanımlanmıřtır. Yapılan deęiřikliklerin etkin olması iin apache yeniden bařlatılmalıdır. Apache'nin yeniden bařlatılmasıyla [http://makeine\\_adı/NAGIOS](http://makeine_adı/NAGIOS) řeklinde NAGIOS'a web arabiriminden eriřim yapılabilir.

NAGIOS arabirimine yetkisiz kiřilerin girmesi istenmeyen bir durum oluřturacaęı iin bu durumu engellemek iin “**use\_authentication**” deęeri kullanılır. “**use\_authentication**” deęeri *cgi.cfg* dosyasında tanımlanır. “**use\_authentication**” deęeri ile web arabirimini kullanmak iin kullanıcı adı yetkilendirmesinin yapılması saęlanır. Bu deęeri aktif hale getirmek deęeri “**1**” yapılır.

```
use_authentication=1
```

Apache'de ve NAGIOS'da yapılan deęiřikliklerin aktif olması iin bu iki servis yeniden bařlatılır.

Gerekli deęiřikliklerin yapılmasından sonra NAGIOS'a web arabiriminden eriřim yapmak istendięinde ekrana kullanıcı ve řifresini soran bir ekran ıkacaktır. Bu yapılan deęiřikliklerin doęru olduęunu gsterir. Web arabiriminden kullanıcı yetkilendirmesi yapabilmek iin **Apache** ile birlikte gelen “**htpasswd**” komutu kullanılır.

**htpasswd -c /usr/local/etc/NAGIOS/etc/htpasswd.users kullanıcı\_adı**

“**-c**” parametresi ilk defa kullanıcı oluřturuluyorsa verilmelidir. Daha sonra aılacak olan kullanıcılar iin komut ařaęıdaki řekilde kullanılmalıdır.

**htpasswd /usr/local/etc/NAGIOS/etc/htpasswd.users kullanıcı\_adı**

```
# htpasswd /usr/local/etc/NAGIOS/htpasswd.users galkan
New password: <sifre>
Re-type new password: <sifre>
Adding password for user galkan
#
```

**htpasswd** komutunun alıřtırılmasının ardından */usr/local/etc/NAGIOS* altında **htpasswd.users** adında kullanıcı adı ve řifrelerinin saklandıęı bir dosya oluřturulacaktır. Burada verilen rneęe gre gelen ekrana kullanıcı adı olarak “**galkan**” řifre olarakda “**sifre**” verilerek web arabirimine giriř yapılır. NAGIOS sadece **use\_authentication** deęerini kullanmamıř kendi iinde de istedięi kullanıcıya istedięi linkleri grmesini saęlamıřtır .Bu deęerlerin aktif ve belirtilen kullanıcıları belirtmesi lazımdır. Bu deęerler aktif hale getirilmemiřse eęer ařaęıdaki gibi bir hata mesajı alınacaktır.

*It appears as though you do not have permission to view information for any of the services you requested...*

*If you believe this is an error, check the HTTP server authentication requirements for accessing this CGI and check the authorization options in your CGI configuration file.*

Paremetre olarak kullanıcı adı alırlar. Genel yapısı **paremetre\_adi==<user1>** şeklindedir. Birden fazla kullanıcı adı tanımlamak için **<user1>** , **<user2>** şeklinde tanımlama yapılır. Örneğin galkan kullanıcısının bütün servisleri izleyebilmesi için şu şekilde tanımlama yapılır.

```
authorized_for_all_services=galkan
```

Kullanılan paremetreler aşağıda belirtilmiştir.Ayrıntıları için [http://NAGIOS.sourceforge.net/docs/2\\_0/configgi.html#authorized\\_for\\_system\\_information](http://NAGIOS.sourceforge.net/docs/2_0/configgi.html#authorized_for_system_information) başına başvurulabilir.

```
authorized_for_system_information=<user1>,<user2>,<user3>,...<usern>
authorized_for_system_commands=<user1>,<user2>,<user3>,...<usern>
authorized_for_configuration_information=<user1>,<user2>,<user3>,...<usern>
authorized_for_all_hosts=<user1>,<user2>,<user3>,...<usern>
authorized_for_all_host_commands=<user1>,<user2>,<user3>,...<usern>
authorized_for_all_services=<user1>,<user2>,<user3>,...<usern>
authorized_for_all_service_commands=<user1>,<user2>,<user3>,...<usern>
```

Değişikliklerin etkin hale getirilmesinden sonra gelen ekranda kullanıcı adı ve şifre yazılarak web arabirimine giriş yapılır. Eğer “**use\_authentication=1**” değeri “**0**” yapılmışsa [http://nakine\\_adi/NAGIOS](http://nakine_adi/NAGIOS) yazan herkes hiçbir yetkilendirme istemeden web arabirimini kullanabilir.

## 10. NAGIOS'DA GÜVENLİK

NAGIOS root kullanıcısının haklarıyla çalıştırılmamalıdır. Yapılandırma dosyasında “**NAGIOS\_user**” ve “**NAGIOS\_group**” paremetreleriyle belirtilen “**root**” kullanıcı haklarına sahip olmayana bir kullanıcı ile çalıştırılmalıdır. Eğer root haklarına sahip olması gereken bir plugin yada “**event\_handler**” parenteresi varsa bunlar “**sudo**” ile çalıştırılmalıdır.

harici\_komut (external\_command) paremetresi ön tanımlı kapalıdır. İhtiyaç olmadığı sürece bu değer kapalı şekilde tutulmalıdır.

## 11. OLASI HATALAR VE ÇÖZÜMLERİ

Öncelikle yapılandırma dosyasında bir tanımlama hatası yapıp yapılmadığı kontrol edilmelidir. Bu **/NAGIOS\_tam\_yolu -v**  
**/NAGIOS\_ana\_yapilandirma\_dosyası\_tam\_yolu** şeklinde öğrenilebilir.



```
# /usr/local/bin/NAGIOS -v /usr/local/etc/NAGIOS/NAGIOS.cfg
```

Yapılandırma dosyalarında bir hata yapılmışsa eğer hata mesajı buradan takip edilerek tekrar kontrol ederek çözülmelidir. Yada NAGIOS ön planda çalıştırılarak aynı şekilde hata mesajları izlenebilir.

Bunun yanında NAGIOS log dosyasında kontrol edilmelidir. Ön tanımlı olarak `/var/spool/NAGIOS/NAGIOS.log` dosyasında loglar tutulur.

```
# tail /var/spool/NAGIOS/NAGIOS.log
...
[1133446138] NAGIOS 2.0b5 starting... (PID=21690)
[1133446138] LOG VERSION: 2.0
[1133446138] Finished daemonizing... (New PID=21691)
# tail /var/spool/NAGIOS/NAGIOS.log
...
[1133446138] NAGIOS 2.0b5 starting... (PID=21690)
[1133446138] LOG VERSION: 2.0
[1133446138] Finished daemonizing... (New PID=21691)
#
```

Yapılandırma dosyalarında bir problem yoksa NAGIOS'un çalıştığından emin olunmalıdır. Bu "`ps`" komutuyla izlenebilir.

```
# ps -auwx | grep 'NAGIOS'
NAGIOS 21691 0.0 0.4 3220 1924 ?? Ss 2:08PM 0:00.01
/usr/local/bin/NAGIOS -d /usr/local/etc/NAGIOS/NAGIOS.cfg
#
```

NAGIOS'un çalıştığından emin olunmalıdır. Ayrıca NAGIOS'ın çalıştığı *kullanıcı/gurubun* gerekli *dizin/dosya* erişim hakları kontrol edilmelidir.

NAGIOS çalışıyor fakat belirtilen servis yada makinenin kontrolü yapılmıyorsa öncelikle pluginlerin çalışıp çalışmadığı kontrol edilmelidir. Pluginler `/usr/local/libexec/NAGIOS` dizini altında bulunur. Örneğin 192.168.1.1 makinesi üzerinde ssh (port 1981) servisini kontrol edilmek isteniyor. Yapılandırma dosyalarında bir hata yok NAGIOS çalışıyorsa

```
# cd /usr/local/libexec/NAGIOS
# ./check_ssh -help
```

komutu ile "`check_ssh`" pluginin aldığı parametreler öğrenilmelidir. Burada ssh servisinin 1981 portundan hizmet verdiği düşünülmüştür.

```
# ./check_ssh -H 192.168.1.1 -p 1981
SSH OK - OpenSSH_4.2p1 FreeBSD-20050903 (protocol 2.0)
#
```

pluginin çalışıp çalışmadığı kontrol edilerek hata giderilebilir. Aynı şekilde yapılandırma dosyasındaki tanımlamalarda hata yok NAGIOS ve pluginler çalışıyor fakat belirtilen şekilde uyarı alınamıyorsa. Örneğin sık kullanılan mail yolu ile bildirimler alınamıyorsa eğer

*hosts.cfg* ve *services.cfg* dosyasında ki “*contact\_groups*” ve “*notification\_options*” deęişkenler ve deęerleri kontrol edilmelidir. Aynı şekilde “*contacts.cfg*” ve “*contactgroups.cfg*” dosyasındaki deęişkenler ve deęerleride kontrol edilmelidir. Bu deęerler doęru fakat mail göndermiyorsa mail komutunun tam yolu öğrenilip *misccommands.cfg* dosyasındaki deęerle karşılaştırılmalıdır. Bunun yanında komut satırında mail gönderip göndermedięi kontrol edilmelidir. Eęer mail göndermiyorsa üzerinde çalıřan mta (Mail Transfer Agent) yapılandırması kontrol edilmelidir.

Bütün bunlar yapıldıktan gerekli servisler (NAGIOS , Apache ) yeniden başlatılıp deęişikliklerin etkin olması saęlanmalıdır.

Burada sunulan çözümler sadece yol göstermek amacıyla anlatılmıřtır kesinlikle tam bir çözümler için anlatılmamıřtır. Daha fazla bilgi için NAGIOS ana sayfasındaki ( <http://NAGIOS.com> ) dokümanlara , forumlara mail listelerinin arřivlerine de bakılmalıdır. NAGIOS yapılandırma seçenekleri çok fazla olan bir programdır. Burada bazı NAGIOS’u çalıştırıp gerekli bazı servislerin kontrolünün yapılması için yapılandırması yapılmıřtır.

## 12. KAYNAKLAR

[http://NAGIOS.sourceforge.net/docs/2\\_0/toc.html](http://NAGIOS.sourceforge.net/docs/2_0/toc.html)

<http://www.NAGIOSexchange.org/>

Gökhan ALKAN gokhan [at] enderunix [dot] org