DOAG 2010

SCAN, GPnP, GNS und Co.

Erfahrungen mit den neuesten RAC Features

Dr. Günter Unbescheid
Database Consult GmbH
Jachenau

Database Consult GmbH

- Gegründet 1996
- Kompetenzen im Umfeld von ORACLE-basierten Systemen
- Tätigkeitsbereiche
  - Tuning, Installation, Konfiguration, Systemanalysen
  - Security, Identity Management
  - Expertisen/Gutachten
  - Support, Troubleshooting, DBA-Aufgaben
  - Datenbankdesign, Datenmodellierung und –design
  - Maßgeschneiderte Workshops
  - www.database-consult.de

SCAN, GPnP, GNS & Co.

Agenda

- Allgemeine Einführung
- Planung und Basiskonfiguration
- Grid Infrastucture – Installation und Test
- RDBMS
- Ausblick

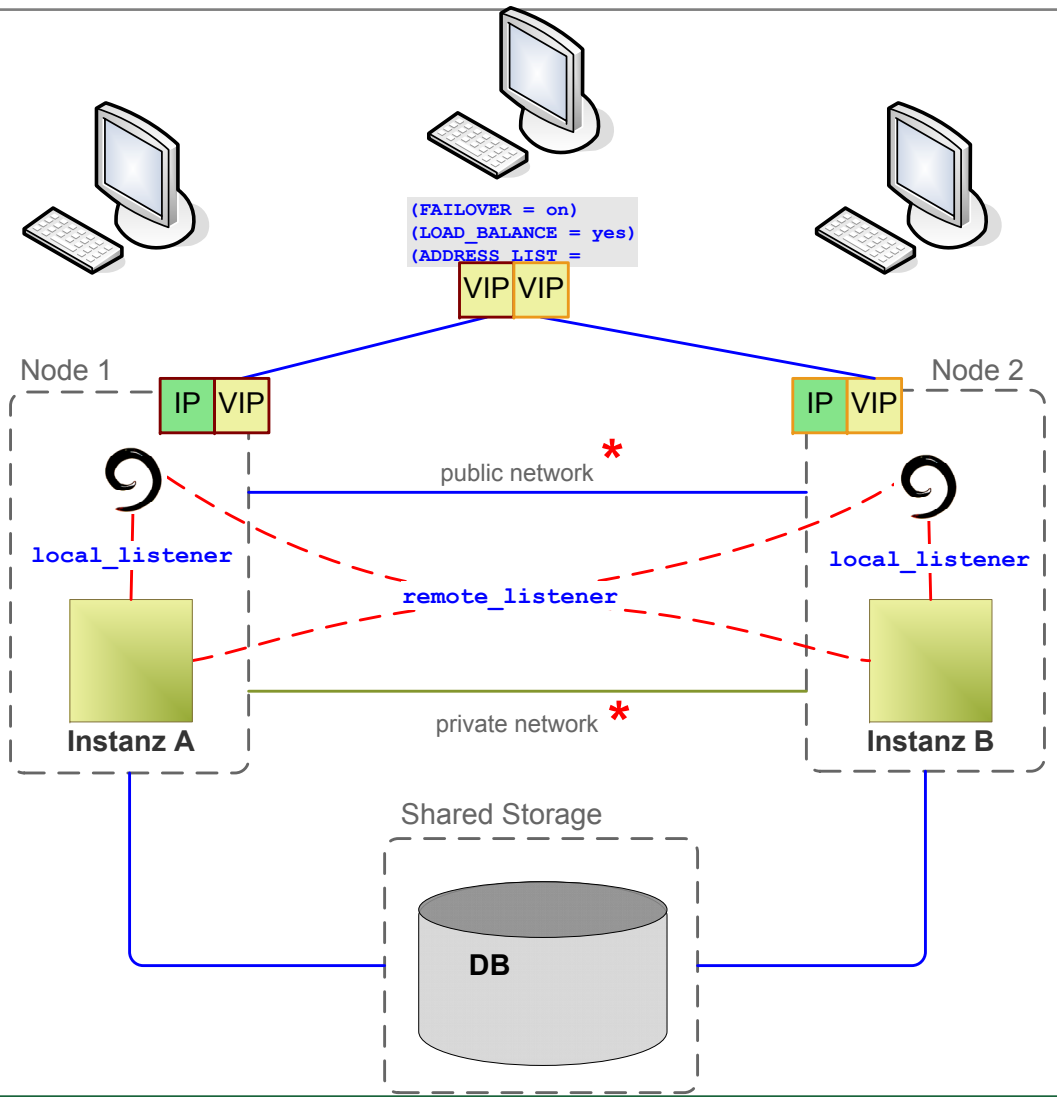
Tests und Beispiele unter:
Oracle Linux Server release 5.6 (x86_64 - 2.6.18-238.el5xen)
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production

Allgemeine Einführung

Begriffe

- SCAN – Single Client Access Name
  - eindeutiger Name für ein Oracle-Cluster
  - nutzbar für den Verbindungsauf von Client-Seite (HOST) – ersetzt dort VIP
  - nutzbar für alle DBs und Services des Clusters, unabhängig von den aktuell beteiligten Knoten
- GNS – Grid Naming Service
  - stellt VIP-Adressen und zugeordnete Namen des public network
  - kann SCAN-Konfiguration automatisieren im Zusammenspiel mit DHCP
  - erfordert eigene Domain- oder Subdomain-Konfiguration im DNS
- GPnP – Zusammenspiel auf SCAN und GNS

RAC "traditionell"



* Redundanz!
☐ statische IPs

VIP

- Virtual IP Address – verwaltet von Clusterware
- gültig im "public network" des Clusters
- Werden RAC-Nodes zugeordnet
- zur Umgehung von TCP timeouts bei *node failures*
  - ggf. 2 Minuten – abhängig von der Plattform
- VIPs "schwenken" auf überlebende Nodes (**`ifconfig`**), dadurch unverzüglicher Response an den Client
- Lokale Listener sind stets gebunden an lokale VIP!
  - kein "schwenkender" Endpoint (**`netstat -an`**)

VIP

- RAC-Nodes: **dbrac01** und **dbrac02 (gestoppt)**
- IP-Adressen:
  - public (NIC) 192.168.45.**86** und 192.168.45.**87**
  - VIP 192.168.45.**192** und 192.168.45.**187**

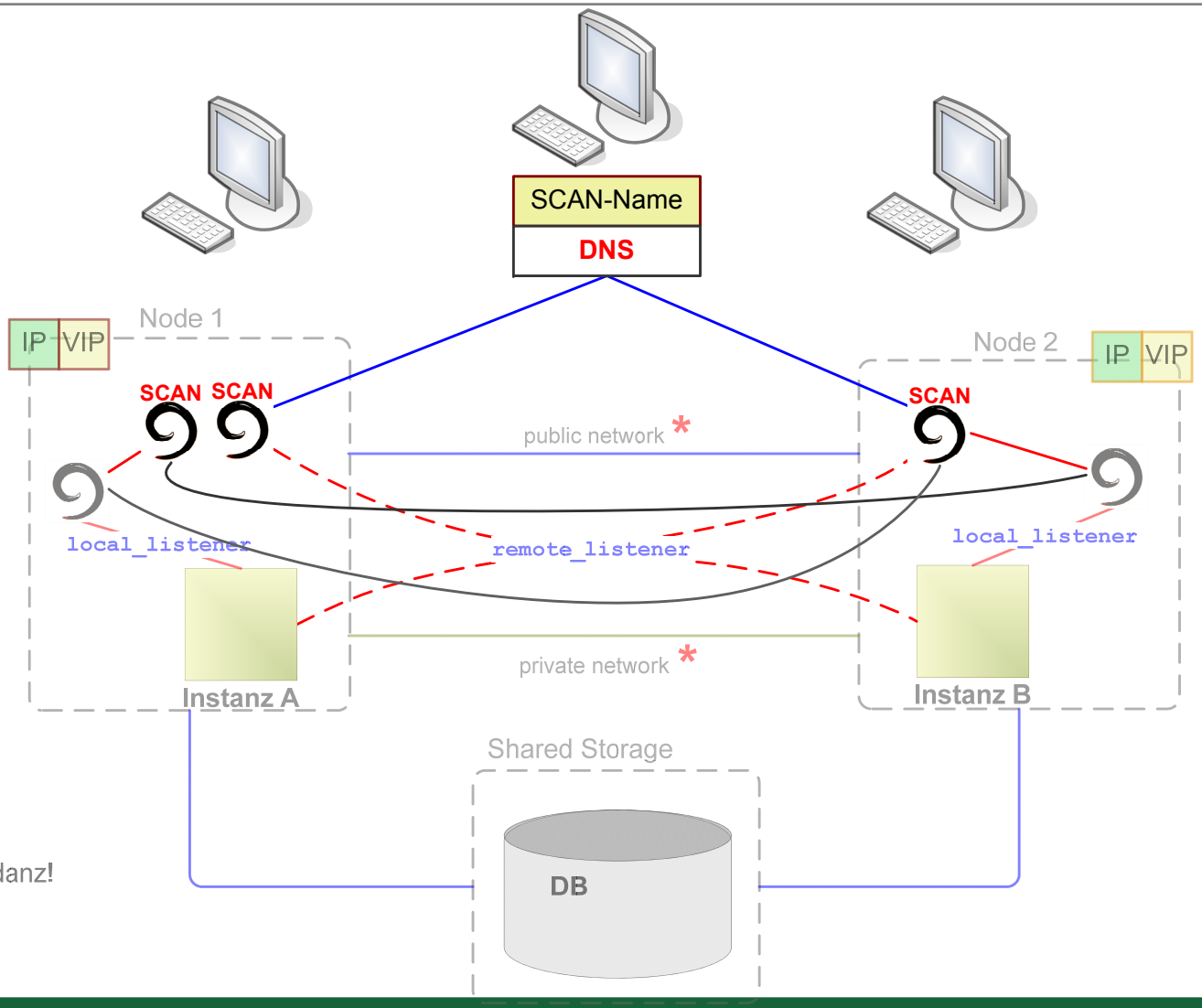
```
[grid@dbrac01 ~]$ olsnodes -i
dbrac01 192.168.45.192
dbrac02 192.168.45.187
[root@dbrac01 ~]# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:16:3E:23:06:B2
          inet addr:192.168.45.86 ...
eth0:4    Link encap:Ethernet  HWaddr 00:16:3E:23:06:B2
          inet addr:192.168.45.187 ...  (verschwindet nach Start 02)
eth0:6    Link encap:Ethernet  HWaddr 00:16:3E:23:06:B2
          inet addr:192.168.45.192
[root@dbrac01 ~]# netstat -an | grep 1521 | grep LISTEN
tcp        0      0 192.168.45.192:1521         0.0.0.0:*          LISTEN
tcp        0      0 192.168.45.86:1521          0.0.0.0:*          LISTEN
```


SCAN-Erweiterungen 11gR2

- Zusätzlich: SCAN Listener
- Client konfiguriert SCAN-Name
- DNS löst diesen (mit Hilfe von GNS) auf und übergibt an SCAN-Listener
- (Drei) verteilte SCAN-Listener (SL)
- SL kennen DB-Services und Lastprofile
- SL leiten Request an DB-Listener weiter
  - implizites Load Balancing

```
11203 =
 (DESCRIPTION =
 (ADDRESS_LIST =
 (ADDRESS = (PROTOCOL = TCP)(HOST = clu01-scan.grid.dbc.de)(PORT = 1521)))
    (CONNECT_DATA =
      (SERVICE_NAME = demo.dbc.de)
      (SERVER = DEDICATED)))
```


SCAN 11gR2



* Redundanz!

Planung und Basiskonfiguration

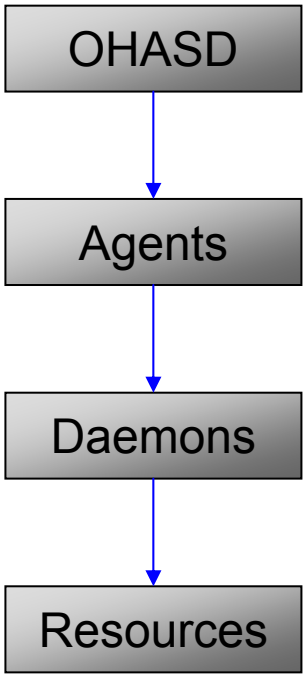
Exaktes und schrittweises Vorgehen mit allen möglichen Zwischenprüfungen erleichtern die Arbeit und erhöhen die Erfolgsquote

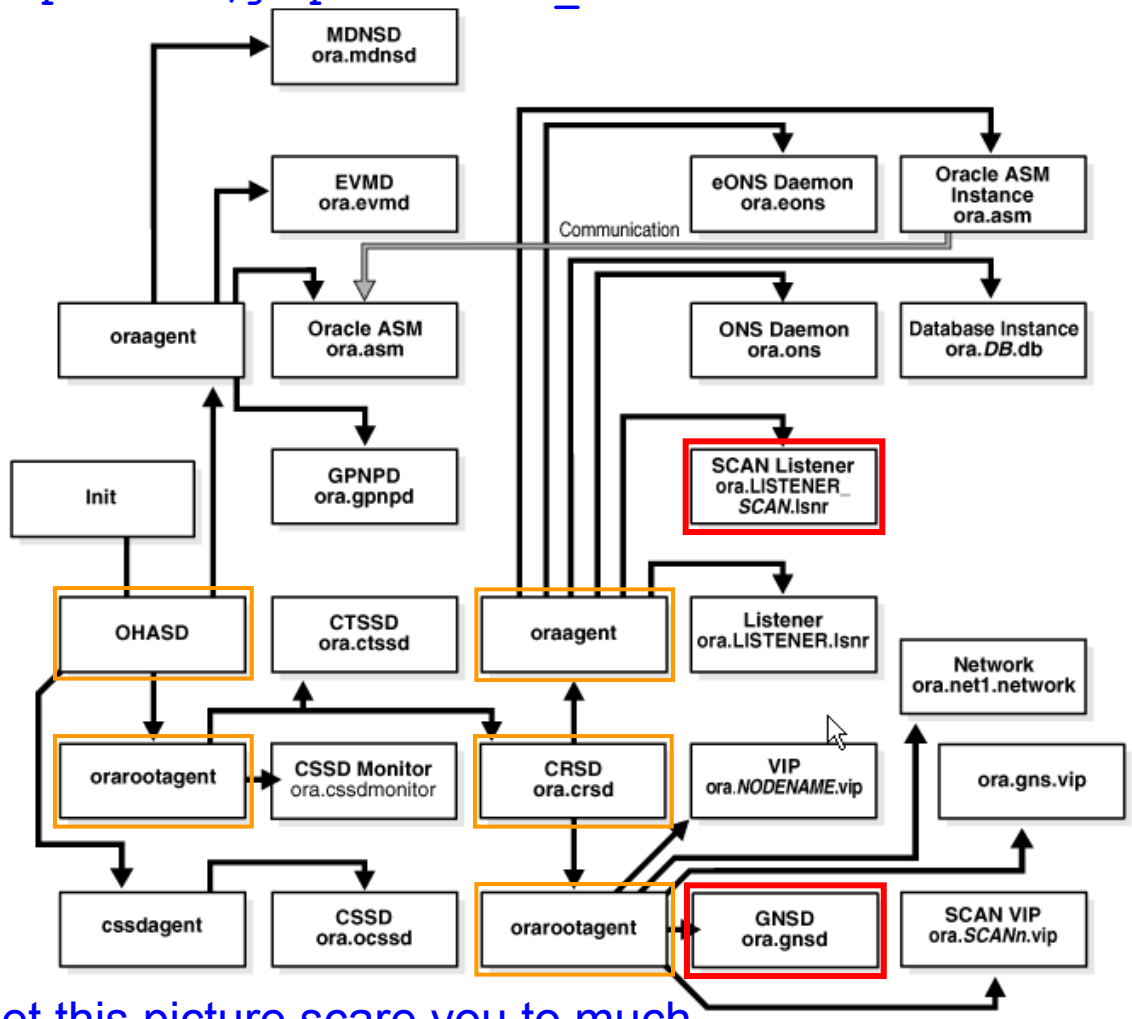
Software Komponenten

- SCAN/GNS Komponenten in "Grid Infrastructure" (GI)
  - zahlreiche Erweiterungen gegenüber älterer Clusterware
  - enthält CRS und ASM, Listener/SCAN Listener
  - muss als erstes auf einem Knoten installiert/konfiguriert werden
  - separates Grid-Home Verzeichnis auf jedem Node
  - OCR und Voting Disks auch in ASM möglich
- GNS-Daemon (root)
  - implementiert Grid Naming Service auf Basis von Zeroconf (zero configuration – DNS-freies Namen-IP Mapping)
  - läuft nur auf einem Cluster-Knoten, Wechsel bei stop des Kn.
- GPNPD-Daemon (grid)
  - koordiniert GPnP-Profile auf allen Cluster-Knoten

Clusterware Stack

```
crsctl stat res -p -init |grep -e "START_DEPENDENCIES" -e "^NAME"
```





Don't let this picture scare you to much ...

IP-Adressen (GNS)

Art	Type	Hosts-File	DNS	DHCP	Bem.
Node IP	public	☑	☑	☒	statisch
Interconnect	private	☑	kann	☒	Eigenes Subnetz
Node VIPs	virtual	☒	☒	☑	
GNS VIP	virtual	☒	☑	☒	GNS Anker public net
SCAN VIPs	virtual	☒	☒	☑	

IP-Adressen (ohne GNS)

Art	Type	Hosts-File	DNS	DHCP	Bem.
Node IP	public	☑	☑	☒	statisch
Interconnect	private	☑	kann	☒	Eigenes Subnetz
Node VIPs	virtual	☑	☑	☒	
GNS VIP	virtual				entfällt
SCAN VIPs	virtual	☒	☑	☒	

Logische Namenskonventionen

- Nach Möglichkeit, Konventionen einhalten
- Knotennamen – direkt z.B. **dbrac01.dbc.de** ...
- GNS- Subdomain – abgeleitet z.B. **grid.dbc.de**
- Clustername – direkt z.B. **clu01**
- Interconnect – abgeleitet **<nodename>-priv**
- VIP-Namen – abgeleitet **<nodename>-vip**
- SCAN-Name – abgeleitet
  <clustername>-scan.grid.dbc.de

Vorbereitung Knoten


```
/etc/hosts
127.0.0.1               localhost.localdomain localhost
192.168.45.86           dbrac01.dbc.de dbrac01
192.168.45.87           dbrac02.dbc.de dbrac02
192.168.1.11            dbrac01-priv.dbc.de dbrac01-priv
192.168.1.12            dbrac02-priv.dbc.de dbrac02-priv
# DNS und DHCP
192.168.45.70           rac0.dbc.de rac0

[root@dbrac01 ~]# cat /etc/resolv.conf
search dbc.de grid.dbc.de
nameserver 192.168.45.70
options attempts: 2
options timeout: 1

/etc/nsswitch.conf (Ausschnitt)
#hosts:     db files nisplus nis dns
hosts:      dns files
```


DNS-Server

Einstieg: /etc/named.conf (Ausschnitt)

```
options {
 directory "/var/named";
....
zone "dbc.de." {
        type master;
        file "master.dbc.de";
        allow-update { none; };
        notify no;
};
```


DNS-Server

Details `/var/named/master.dbc.de` **(Ausschnitt)**
$ORIGIN überschreibt Standard-Domänenname (@)

```
$TTL    86400
@       IN      SOA     rac0.dbc.de.    root.localhost. (
                        20110321        ; serial
                        28800           ; refresh
                        14400           ; retry
                        3600000         ; expire
                        86400 )         ; minimum
@               IN      NS      rac0.dbc.de.
localhost       IN      A       127.0.0.1
dbrac01         IN      A       192.168.45.86
dbrac02         IN      A       192.168.45.87
rac0            IN      A       192.168.45.70

$ORIGIN grid.dbc.de.
@       IN NS   clu01-gns.grid.dbc.de.
clu01-gns.grid.dbc.de.  IN A     192.168.45.180
```


DHCP-Beispiel


```
[root@rac0]# cat /etc/dhcpd.conf
ddns-update-style interim;
ignore client-updates;
subnet 192.168.45.0 netmask 255.255.255.0 {
        option routers                  192.168.45.254;
        option subnet-mask              255.255.255.0;
        option ip-forwarding            off;
        option domain-name              "dbc.de";
        option domain-name-servers      192.168.45.70;
        option ntp-servers              64.99.80.30;
        range                           192.168.45.181 192.168.45.196;
        default-lease-time              21600;
        max-lease-time                  43200;
}
```

Services starten/neu starten:

```
service named start
service dhcpd start
```


Tests vor der Grid Installation

- Konfiguration
  - gleiche MTU, NIC Namen (Ausnahmen private ab 11gR2)
- Connectivity (Namen/IPs), DNS Auflösung
  - Systematische `ping` mit MTU-Größe (MOS 1054902.1)
- ggf Cluster Verify einsetzen – vor Installation auf Medium

```
netstat -in
ifconfig -a
ping -s 1500 -c 2 -I 192.168.45.86 dbrac01
ping -s 1500 -c 2 -I 192.168.1.11 dbrac02-priv
-- Traceroute in 1 hop
traceroute -s 192.168.1.11 -r -F 192.168.1.12 1472
nslookup dbrac01.dbc.de
nslookup 192.168.45.86
./runcluvfy.sh stage -post hwos -n dbrac01,dbrac02
./runcluvfy.sh stage -pre crsinst -n dbrac01,dbrac02
```


Grid Infrastructure

Installationshinweise

- Vorab separate Prüfung empfohlen (cluvfy)
  - ssh kann von Installer konfiguriert werden, dann ggf. Fehler in cluvfy
- Installation in eigenem Home wie dokumentiert
- Eigener Benutzer (grid) empfohlen
  - wegen ASM Gruppenüberlappung mit oracle, identisch auf allen Kn.
- Namen und Einstellungen wie dargestellt im Dialog oder über response-Datei

```
id grid
uid=54322(grid) gid=54321(oinstall)
groups=54322(dba),54323(asmadmin),54325(asmdba),54326(asmop
er),54321(oinstall)

id oracle
uid=54321(oracle) gid=54321(oinstall)
groups=54322(dba),54324(oper),54325(asmdba),54321(oinstall)
```


Überprüfung


```
[grid@dbrac01 grid]$ crsctl check cluster -all
**************************************************************
dbrac01:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
**************************************************************
dbrac02:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
**************************************************************
```


Überprüfung


```
[grid@dbrac01 grid]$ srvctl status scan
SCAN VIP scan1 is enabled
SCAN VIP scan1 is running on node dbrac02
SCAN VIP scan2 is enabled
SCAN VIP scan2 is running on node dbrac01
SCAN VIP scan3 is enabled
SCAN VIP scan3 is running on node dbrac01

[grid@dbrac01 grid]$ srvctl config scan
SCAN name: clu01-scan.grid.dbc.de, Network:
1/192.168.45.0/255.255.255.0/eth0
SCAN VIP name: scan1, IP: /clu01-scan.grid.dbc.de/192.168.45.191
SCAN VIP name: scan2, IP: /clu01-scan.grid.dbc.de/192.168.45.186
SCAN VIP name: scan3, IP: /clu01-scan.grid.dbc.de/192.168.45.185

[grid@dbrac01 grid]$ srvctl config scan_listener
SCAN Listener LISTENER_SCAN1 exists. Port: TCP:1521
SCAN Listener LISTENER_SCAN2 exists. Port: TCP:1521
SCAN Listener LISTENER_SCAN3 exists. Port: TCP:1521
```


Überprüfung


```
-- Auf wechselnde Reihenfolge achten
[grid@dbrac01 grid]$ nslookup clu01-scan.grid.dbc.de
Server:         192.168.45.70
Address:        192.168.45.70#53

Non-authoritative answer:
Name:   clu01-scan.grid.dbc.de
Address: 192.168.45.186
Name:   clu01-scan.grid.dbc.de
Address: 192.168.45.191
Name:   clu01-scan.grid.dbc.de
Address: 192.168.45.185

[grid@dbrac01 grid]$ srvctl status gns
GNS is running on node dbrac01.
GNS is enabled on node dbrac01.
[grid@dbrac01 ~]$ srvctl status gns -n dbrac02
GNS is not running on node dbrac02.
GNS is enabled on node dbrac02.
```


Überprüfung


```
[grid@dbrac01 admin]$ ping -c 2 dbrac02-vip
PING dbrac02-vip.grid.dbc.de (192.168.45.187) 56(84) bytes of data.
64 bytes from 192.168.45.187: icmp_seq=1 ttl=64 time=2.34 ms
64 bytes from 192.168.45.187: icmp_seq=2 ttl=64 time=0.299 ms

--- dbrac02-vip.grid.dbc.de ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.299/1.324/2.349/1.025 ms
```

DHCPD Kontrolle über Datei `/var/lib/dhcpd/dhcpd.leases`

```
[grid@dbrac01 admin]$ srvctl config vip -n dbrac02
VIP exists:
/192.168.45.187/192.168.45.187/192.168.45.0/255.255.255.0/eth0, hosting
node dbrac02
[grid@dbrac01 admin]$ srvctl config vip -i dbrac02-vip
VIP exists:
/192.168.45.187/192.168.45.187/192.168.45.0/255.255.255.0/eth0, hosting
node dbrac02
```


RDBMS

Software und Datenbank

- Getrennter Aufbau empfohlen:
  - RDBMS-Software (Benutzer **oracle**)
  - Anlegen der Datenbank interaktiv oder per Response-Datei, ggf. bei ASM vorher zusätzliche Diskgruppen anlegen
- Bei funktionsfähiger Grid Infrastructure in der Regel unspektakulär

```
-- tnsnames.ora
11203 =
 (DESCRIPTION =
  (ADDRESS_LIST =
  (ADDRESS = (PROTOCOL = TCP)(HOST = clu01-scan.grid.dbc.de)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = demo.dbc.de)
      (SERVER = DEDICATED)
    )
  )
```


Verbindungsaufbau

- Client kontaktiert DNS zur Auflösung des SCAN-Namens
- C erhält IPs der SCAN-Listener und kontaktiert einen (*random*-Prinzip) – zusätzlich Port und Servicename
  - Clients <= 11.2 beachten nur die erste gelieferte Adresse, Verifizierung per Net-Tracefile
- SCAN-Listener leitet die Anfrage per VIP an einen DB Listener weiter
  - bestimmt per load profile
- Listener startet einen Serverprozeß (falls "dedicated")

listener.ora


```
[grid@dbrac01 ~]$ cat $ORACLE_HOME/network/admin/listener.ora

LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER))))
# line added by Agent
LISTENER_SCAN3=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN3))))
# line added by Agent
LISTENER_SCAN2=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN2))))
# line added by Agent
LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN1))))
# line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1=ON        # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2=ON        # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3=ON        # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON              # line added by Agent

[grid@dbrac01 ~]$ cat $ORACLE_HOME/network/admin/endpoints_listener.ora
LISTENER_DBRAC01=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.45.192)
(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.45.86)(PORT=1521)(IP=FIRST))))
# line added by Agent

-- init.ora
local_listener
(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.45.192)(PORT=1521))))

remote_listener  clu01-scan.grid.dbc.de:1521
```


Listener-Status (Ausschnitte)

```
[grid@dbrac02 ~]$ olsnodes -i
dbrac01 192.168.45.192
dbrac02 192.168.45.187
[grid@dbrac02 ~]$ lsnrctl stat listener
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.45.87)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.45.187)(PORT=1521)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM2", status READY, has 1 handler(s) for this service...
Service "demo.dbc.de" has 1 instance(s).
  Instance "demo2", status READY, has 1 handler(s) for this service...
[grid@dbrac02 ~]$ srvctl status scan
SCAN VIP scan1 is enabled
SCAN VIP scan1 is running on node dbrac02
...
[grid@dbrac02 ~]$ srvctl config scan
SCAN name: clu01-scan.grid.dbc.de, Network:
1/192.168.45.0/255.255.255.0/eth0
SCAN VIP name: scan1, IP: /clu01-scan.grid.dbc.de/192.168.45.191
SCAN VIP name: scan2, IP: /clu01-scan.grid.dbc.de/192.168.45.186
SCAN VIP name: scan3, IP: /clu01-scan.grid.dbc.de/192.168.45.185
```


Listener-Status (Ausschnitte)


```
lsnrctl status listener_scan1
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER_SCAN1)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.45.191)(PORT=1521)))

Services Summary...
Service "demo.dbc.de" has 2 instance(s).
  Instance "demo1", status READY, has 1 handler(s) for this service...
  Instance "demo2", status READY, has 1 handler(s) for this service...
```


Ausblick

Kompatibilitäten

Oracle Client Version	Oracle Database Version	Comment
Oracle Database 11g Release 2	Oracle Database 11g Release 2	No change required.
Oracle Database 11g Release 2	**Pre-** Oracle Database 11g Release 2	Add the SCAN VIPs as hosts to the REMOTE_LISTENER parameter.
Pre- Oracle Database 11g Release 2	Oracle Database 11g Release 2	Change the client TNSNAMES.ora to include the SCAN VIPs (* see below). **IF** the database was upgraded using the DBUA from a pre-11g Rel. 2 database, the DBUA will configure the REMOTE_LISTENER parameter to point to the node-VIPs as well as the SCAN.
Pre- Oracle Database 11g Release 2	**Pre-** Oracle Database 11g Release 2	If you want to make use of SCAN (recommended): add the SCAN VIPs as hosts to the REMOTE_LISTENER parameter. AND Change the client TNSNAMES.ora to include the SCAN VIPs (* see below). Otherwise, no change required.

Table 1: Oracle Client and Oracle Database Version Compatibility for SCAN

Quelle:Oracle

Fazit

- SCAN
  - nützliche Vereinfachung für Client Connects vor allem für homogene Client Landschaft
  - bei älteren Versionen zusätzlicher Aufwand
- GNS
  - "ungewohnte" Domänen-Arbeit (?)
  - nur sinnvoll für Node-Skalierung
  - Automatisiert nur einen Aslpekt der Node-Ergänzung
  - alternative Strategien denkbar

Danke für‘s Zuhören

www.database-consult.de