

Wie benutzt der NetWorker Remote Procedure Calls (RPC) ?

Der NetWorker - wie jede andere Client/Server (Backup) Software - benutzt immer diese zwei grundlegenden Funktionen des Betriebssystems:

- **Die eindeutige Namensauflösung**

In jedem Fall müssen alle Hostnamen einer NetWorker Data Zone von allen Seiten zur gleichen IP Adresse aufgelöst werden.

- **Remote Procedure Calls (RPC)**

Da der NetWorker Server Befehle auf den NetWorker Clients/Storage Nodes starten muß, ist diese Forderung ebenfalls unabdingbar.

Beide Aufgaben/Dienste arbeiten normalerweise im Hintergrund unsichtbar für den Anwender. Das macht auch Sinn, denn normalerweise soll er ja nicht sehen, wie ein Programm intern arbeitet. Andere Software kann sich ihrer bei Bedarf jedoch bedienen, z.B., wenn die Sicherung von einem Anwendungsprogramm aus gestartet werden soll. In diesem Tip beschreibe ich, wie dies prinzipiell möglich ist.

Zum Beispiel ist die Steuerung einer Jukebox mit dem Befehl (`nsrjb`) nur an dem NetWorker Rechner möglich, an dem der Robot angeschlossen ist. Das bedeutet aber auch, daß sich der Operator hierzu an dem Rechner anmelden muß, damit der diesen Befehl aufrufen kann.

Allerdings ist dies von der Admin GUI auch ohne ein Remote Login möglich. Wenn nun aber stimmt, daß die GUI nichts anderes macht, als einen Befehl auszuführen, muß sie sich quasi im Hintergrund anmelden, ohne daß der Operator dies explizit machen muß.

Um ein Remote Login überhaupt zu unterstützen, brauchen Sie natürlich ein Programm, daß die gesendeten Befehle empfängt und ausführt - genau dies macht ein sog. 'Listener'. Er muß natürlich auf auch jedem NetWorker Rechner laufen - logischerweise bleibt dann nur der NetWorker Client Service `nsrexecd (.exe)` übrig, der ja in der Windows Service Liste als *NetWorker Remote Exec Service* eingetragen wird. Und natürlich ist ein 'Listener' immer aktiv, deshalb wird der Dienst mit dem Betriebssystem automatisch gestartet.

Um also einen NetWorker Befehl mittels RPC zu starten, müssen Sie nur:

- Den richtigen Usernamen verwenden, der kein Paßwort benötigt *root* bzw. *system*
- Den RPC Befehl mit all seinen Parametern zu übermitteln
- Den Befehl mit dem NetWorker Befehl `nsrexec` am richtigen NetWorker Rechner starten



Vergessen Sie nicht, daß Sie die Datei der zugelassenen NetWorker Rechner (`/nsr/res/servers`) entsprechend anpassen müssen - sie muß die Namen aller NetWorker Rechner enthalten, die berechtigt sein sollen, auf dem lokalen System Befehle auszuführen.

Denken Sie auch daran, daß nach der Änderung dieser Datei der Service `nsrexecd` neu gestartet werden muß, damit die Änderungen auch wirksam werden.

Die folgende Seite zeigt ein Beispiel mit einer aktuellen NetWorker Installationen (Start einer automatischen Sicherung vom Client `2008r2` am NetWorker Server `nw761`) - die grundlegende Beschreibung ab Seite 3 habe ich einem älteren, englischen Dokument von mir entnommen. Wie Sie sehen, hat sich fast nichts geändert.

```
C:\>set RUSER=root
```

```
C:\>set RCMD=savegrp -v -G Default
```

```
C:\>nsrexec -v -c nw761
```

```
32451 1318776635 2 0 0 3700 4020 0 nw761 savegrp 16 %s:%-s level=%s 4 0 6 2008r2 1 2 34 0 7 D:\TEST 0 4 full
32451 1318776635 2 0 0 3700 4020 0 nw761 savegrp 16 %s:%-s level=%s 4 0 5 nw761 1 2 35 0 7 D:\TEST 0 4 full
7236 1318776636 2 0 0 3700 4020 0 nw761 savegrp 36 Group will not limit job parallelism 0
32494 1318776636 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 6 2008r2 1 2 38 0 5 probe
0 1318776636 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 67 savefs -s nw761 -c 2008r2 -g Default -p -l full -R -v -F "D:\TEST"
32494 1318776636 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 5 nw761 1 2 39 0 5 probe
0 1318776636 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 66 savefs -s nw761 -c nw761 -g Default -p -l full -R -v -F "D:\TEST"
0 1318776643 2 0 0 3700 4020 0 nw761 savegrp 8 %s:%-s 3 0 6 2008r2 1 2 30 0 7 D:\TEST
0 1318776643 2 0 0 3700 4020 0 nw761 savegrp 37 level=%s, dn=%d, mx=%d, vers=%s, p=%d 5 0 4 full 1 1 0 1 1 0 5 pools 1 1 4
77567 1318776643 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 15 * //
2008r2:Probe 23 32 E:\nsr\temp\sg\Default\sso.000002
7341 1318776643 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 6 2008r2 0 5 probe 0 9 succeeded
32494 1318776643 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 6 2008r2 1 2 38 0 7 D:\TEST
0 1318776643 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 77 save -s nw761 -g Default -LL -m 2008r2 -l full -W 78 -N "D:\TEST" "D:\TEST"
0 1318776648 2 0 0 3700 4020 0 nw761 savegrp 8 %s:%-s 3 0 5 nw761 1 2 28 0 7 D:\TEST
0 1318776648 2 0 0 3700 4020 0 nw761 savegrp 37 level=%s, dn=%d, mx=%d, vers=%s, p=%d 5 0 4 full 1 1 0 1 1 0 8 ssbrowse 1 2 12
77567 1318776648 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 14 * //
nw761:Probe 23 32 E:\nsr\temp\sg\Default\sso.000003
7341 1318776648 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 5 nw761 0 5 probe 0 9 succeeded
32494 1318776648 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 5 nw761 1 2 39 0 7 D:\TEST
0 1318776648 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 76 save -s nw761 -g Default -LL -m nw761 -l full -W 78 -N "D:\TEST" "D:\TEST"
4690 1318776686 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 40 Default waiting for 2 job(s) to complete
77567 1318776772 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 17 * //
2008r2:D:\TEST 23 32 E:\nsr\temp\sg\Default\sso.000004
7341 1318776772 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 6 2008r2 0 7 D:\TEST 0 9 succeeded
32494 1318776772 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 6 2008r2 1 2 38 0 5 index
0 1318776777 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 154 save -s nw761 -S -g Default -LL -f - -m nw761 -V -l full -LL -W 78 //
-N index:db7cd5c6-00000004-4e97be39-4e9aed94-00050c00-e28f8f29 "E:\nsr\index\2008r2"
77567 1318776803 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 15 * //
2008r2:index 23 32 E:\nsr\temp\sg\Default\sso.000006
7341 1318776803 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 6 2008r2 0 5 index 0 9 succeeded
4690 1318776833 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 40 Default waiting for 1 job(s) to complete
77567 1318776889 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 16 * //
nw761:D:\TEST 23 32 E:\nsr\temp\sg\Default\sso.000005
7341 1318776889 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 5 nw761 0 7 D:\TEST 0 9 succeeded
32494 1318776889 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 5 nw761 1 2 39 0 5 index
0 1318776889 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 153 save -s nw761 -S -g Default -LL -f - -m nw761 -V -l full -LL -W 78 //
-N index:59da0b0d-00000004-4e97be38-4e97be37-00010c00-e28f8f29 "E:\nsr\index\nw761"
77567 1318776895 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 14 * //
nw761:index 23 32 E:\nsr\temp\sg\Default\sso.000007
7341 1318776895 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 5 nw761 0 5 index 0 9 succeeded
32494 1318776895 2 0 0 3700 4020 0 nw761 savegrp 15 %s:%-s started 3 12 5 nw761 1 2 39 0 9 bootstrap
0 1318776895 2 0 0 3700 4020 0 nw761 savegrp 2 %s 1 0 108 save -s nw761 -S -g Default -LL -f - -m nw761 -V -l 0 -LL -W 78 //
-N bootstrap -x "E:\nsr\res" "E:\nsr\mm"
77567 1318776901 2 0 0 3700 4020 0 nw761 savegrp 49 %s %s See the file %s for output of save command. 3 26 7 Default 0 18 * //
nw761:bootstrap 23 32 E:\nsr\temp\sg\Default\sso.000008
7341 1318776901 2 0 0 3700 4020 0 nw761 savegrp 9 %s:%s %s. 3 0 5 nw761 0 9 bootstrap 0 9 succeeded
7241 1318776907 2 0 0 3700 4020 0 nw761 savegrp 28 nsrim run recently, skipping 0
```

```
C:\>
```

1. UNIX > UNIX

Starten einer Sicherungsgruppe von einem NetWorker Client aus:

```
# RUSER=root; export RUSER
# RCMD="savegrp -G Default"; export RCMD
# nsrexec -c dx2kserver
#
```

Das ist bereits alles. Leider ist das Programm `savegrp` nicht sehr mitteilend. Andere Beispiele sind:

Unmounting eines Mediums von einem Laufwerk:

```
# RUSER=root; export RUSER
# RCMD="nsrmm -u -f E:\BU_DEV1"; export RCMD
# nsrexec -c dx2kserver
Unmounted device: E:\BU_DEV1
```

Remounten eines Mediums in einem Laufwerk:

```
# RUSER=root; export RUSER
# RCMD="nsrmm -m -f E:\BU_DEV1"; export RCMD
# nsrexec -c dx2kserver
file disk dx2kserver.001 mounted on E:\BU_DEV1, write enabled
#
```

Probleme lassen sich in der Regel auf fehlerhafte Befehlssyntax bzw. Gegebenheiten zurückführen. Auch hierzu ein Beispiel:

```
# RUSER=system; export RUSER
# RCMD="savegrp -s dx2kserver -G Default"; export RCMD
# nsrexec -c dx2kserver
Connection refused
#
```

Das Problem: `savegrp` kann ja nur auf dem NetWorker Server ausgeführt werden - deshalb der Parameter `-s server_name` ! - Allerdings gibt es auch eine andere Methode (siehe S. 2).



Natürlich muß dies auch Plattform-übergreifend möglich sein. Dies bespreche ich auf den nächsten Seiten.

2. UNIX > Windows

Starten einer Sicherungsgruppe von einem NetWorker Client aus:

```
# RUSER=system; export RUSER
# RCMD="savegrp -G Default"; export RCMD
# echo $RUSER
root
# echo $RCMD
savegrp -G Default
# nsrexec -c dx2kserver
#
```

Wie sie an diesem Befehl erkennen können, gibt es keinen Unterschied bzgl. des Usernamens. Allerdings zeigten weitere Tests, daß der auch User *root* funktionierte. (Siehe nächste Seite):

Hier die entsprechenden Beispiele der letzten Seite:

Unmounting eines Mediums von einem Laufwerk:

```
# RUSER=system; export RUSER
# RCMD="nsrmm -u -f E:\BU_DEV1"; export RCMD
# nsrexec -c dx2kserver
Unmounted device: E:\BU_DEV1
#
```

Remounten eines Mediums in einem Laufwerk:

```
# RUSER=system; export RUSER
# RCMD="nsrmm -m -f E:\BU_DEV1"; export RCMD
# nsrexec -c dx2kserver
file disk dx2kserver.001 mounted on E:\BU_DEV1, write enabled
#
```

Nur an den Device-Namen können Sie erkennen, daß es sich um einen Windows Rechner handelt.

3. Windows > UNIX

Starten einer Sicherungsgruppe von einem NetWorker Client aus:

```
C:\>SET RUSER=root

C:\>echo %RUSER%
root

C:\>SET RCMD=savegrp -G Default

C:\>echo %RCMD%
savegrp -G Default

C:\>nsrexec -c ultra

C:\>
```

```

C:\>nsrexec -c ultra
ultra:/etc                               level=incr
01/17/04 09:30:50 savegrp: Run up to 4 clients in parallel
01/17/04 09:30:50 savegrp: ultra:probe                                     started
saveefs -s ultra -c ultra -g Default -p -l full -R -v -F /etc
01/17/04 09:30:54 savegrp: ultra:probe succeeded.
rcmd localhost, user root: `saveefs -s ultra -c ultra -g Default -p -l full -R -v -F /etc'
nsrexec: authtype nsrexec
type: NSR client description;
pools supported: Yes;
migration supported: Yes;
browse time supported: Yes;
multiple balanced streams supported: Yes;
remote user: root;
arch: sparc;
client OS type: Solaris;
CPU type: sun4u;
CPUs: 1;
IP address: 195.214.75.200, 195.214.70.200;
kernel arch: sun4u;
machine type: workstation;
MB used: 3362;
NetWorker version: 7.0.Build.322;
OS: SunOS 5.6;
version: 7.0.Build.322;
save set: path=/etc, arg=/etc, level=full, diskno=0, max_sessions=1, stype=save ;
parallelism: 8
ultra:/etc                               level=incr, dn=0, mx=1, vers=ssbrowse, p=4
01/17/04 09:30:54 savegrp: ultra:/etc                                     started
save -s ultra -g Default -LL -m ultra -t 1074327968 -l incr -W 78 -N /etc /etc
01/17/04 09:30:55 savegrp: ultra:/etc succeeded.
.....
* ultra:bootstrap /export/home/nsr/res/nsrdb/09/
* ultra:bootstrap /export/home/nsr/res/nsrdb/
* ultra:bootstrap /export/home/nsr/res/
* ultra:bootstrap /export/home/nsr/mm/
* ultra:bootstrap
  ultra: bootstrap                level=full,      45 KB 00:00:01      91 files
* ultra:bootstrap completed savetime=1074328257
01/17/04 09:30:59 savegrp: nsrim run recently, skipping

C:\>

```

4. Windows > Windows

Starten einer Sicherungsgruppe von einem NetWorker Client aus:

```
C:\>SET RUSER=root

C:\>echo %RUSER%
root

C:\>SET RCMD=savegrp -G Default

C:\>echo %RCMD%
savegrp -G Default

C:\>nsrexec -c dx2kserver

C:\>
```

Noch einmal - dies ist bereits alles. Leider ist das Programm `savegrp` nicht sehr mitteilend. Die Ausgabe finden Sie auf der nächsten Seite.



Bitte beachten Sie, daß der NetWorker diesem Mechanismus auch intern für seine eigenen Zwecke verwendet - manchmal kann der Anwender dies sogar erkennen. Zum Beispiel, wenn er eine RAW Sicherung an einem Remote Client durchführt:

```
C:\>savegrp -v -G RAW_Backups
creinfeld2:\\.\S: level=full
01/25/04 00:48:55 savegrp: Run up to 4 clients in parallel
01/25/04 00:48:55 savegrp: creinfeld2:probe started
savefs -s dx2kserver -c creinfeld2 -g RAW_Backups -p -l full -R //
-v -F \\.\S:
01/25/04 00:49:18 savegrp: creinfeld2:probe succeeded.
rcmd creinfeld2, user root: `savefs -s dx2kserver -c creinfeld2 //
-g RAW_Backups -p -l full -R -v -F "\\.\S:"
nsrexec: authtype nsrexec
type: NSR client description;
```

Und natürlich sind gute RPC Programme so geschrieben, daß nicht die Ausführung aller Programm möglich ist - andern falls könnte man ungewohnt sogar das System beschädigen. Auch hierzu ein Beispiel:

```
C:\>SET RUSER=root

C:\>SET RCMD=reboot

C:\>nsrexec -c ultra
Invalid command.
01/15/04 23:59:20 nsrexec: Invalid command.
Permission denied

C:\>
```

```

C:\>nsrexec -c dx2kserver
dx2kserver:D:\TEST                level=incr
01/16/04 00:09:55 savegrp: Run up to 4 clients in parallel
01/16/04 00:09:55 savegrp: dx2kserver:probe                               started
savefs -s dx2kserver -c dx2kserver -g Default -p -l full -R -v -F D:\TEST
01/16/04 00:10:02 savegrp: dx2kserver:probe succeeded.
dx2kserver:D:\TEST                level=incr, dn=0, mx=1, vers=ssbrowse, p=4
01/16/04 00:10:02 savegrp: dx2kserver:D:\TEST                            started
save -s dx2kserver -g Default -LL -m dx2kserver -t 1074203977 -l incr -W 78 -N D:\TEST D:\TEST
01/16/04 00:10:11 savegrp: dx2kserver:D:\TEST succeeded.
01/16/04 00:10:15 savegrp: dx2kserver:index                               started
save -s dx2kserver -S -g Default -LL -f - -m dx2kserver -V -t 1074202452 -l 9 -LL -W 78 //
-N index:d9c5b3c5-00000004-400706b2-400706b1-00010000-c3d64b9f E:\Program Files\nsr\index\dx2kserver
01/16/04 00:10:24 savegrp: dx2kserver:index succeeded.
01/16/04 00:10:29 savegrp: dx2kserver:bootstrap                          started
save -s dx2kserver -S -g Default -LL -f - -m dx2kserver -V -l 0 -LL -W 78 -N bootstrap //
E:\Program Files\nsr\res E:\Program Files\nsr\mm
01/16/04 00:10:38 savegrp: dx2kserver:bootstrap succeeded.
rcmd localhost, user root: `savefs -s dx2kserver -c dx2kserver -g Default -p -l full -R -v -F "D:\\TEST" '
nsrexec: authtype nsrexec
type: NSR client description;
pools supported: Yes;
browse time supported: Yes;
multiple balanced streams supported: Yes;
remote user: SYSTEM;
groups: Administrators, Everyone, Authenticated Users;
.....
* dx2kserver:bootstrap E:\Program Files\nsr\res\nsrdb\09\4f007805af060740c3d64b9f
* dx2kserver:bootstrap E:\Program Files\nsr\res\nsrdb\09\
* dx2kserver:bootstrap E:\Program Files\nsr\res\nsrdb\
* dx2kserver:bootstrap E:\Program Files\nsr\res\nsrla.res
* dx2kserver:bootstrap E:\Program Files\nsr\res\servers
* dx2kserver:bootstrap E:\Program Files\nsr\res\
* dx2kserver:bootstrap nsrmmdbasm -s E:\Program Files\nsr\mm\mmvolume6\
* dx2kserver:bootstrap E:\Program Files\nsr\mm\
* dx2kserver:bootstrap
dx2kserver: bootstrap                level=full,      69 KB 00:00:06      91 files
* dx2kserver:bootstrap completed savetime=1074208230

C:\>

```