

# Solaris 10 Zonen

Release 6.0: 14.05.2011/bk

Die Informationen in diesem Dokument werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Bei der Zusammenstellung von Texten wurde mit grösster Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Der Autor kann für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen. Für Verbesserungsvorschläge und Hinweise auf Fehler ist der Autor dankbar.

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Die gewerbliche Nutzung ist nicht zulässig.

Fast alle Hardware- und Softwarebezeichnungen und weitere Stichworte und sonstige Angaben, die in diesem Dokument verwendet werden, sind als eingetragene Marken geschützt. Da es nicht möglich ist, in allen Fällen zeitnah zu ermitteln, ob ein Markenschutz besteht, wird das ©-Zeichen in diesem Buch nicht verwendet.

Hinweise, Tipps, Anregungen, Kritik, Verbesserungsvorschläge bitte an den  
Autor: [Helmut.Birk@edv-birk.de](mailto:Helmut.Birk@edv-birk.de) Homepage: <http://www.edv-birk.de>

## Modul 1 Zonen

- **Überblick**

[1-1 Allgemeines](#)

[1-2 Zonen konfigurieren und administrieren \(Grundlagen\)](#)

[1-3 Zonen konfigurieren und administrieren \(Erweiterungen\)](#)

[1-4 Resource-Pools konfigurieren und administrieren](#)

[1-5 Zonen konfigurieren mit Resource-Pool und FSS \(Übungsbeispiel\)](#)

[1-6 Resource Management](#)

[1-7 Neue Zonen-Features mit update 4 / Rel 807 bzw. update 6 / Rel 1008](#)

[1-8 BrandZ \( Branded Zonen \)](#)

### 1-1 Allgemeines

1 globale Zone (Name=global, Zoneid=0)

n nicht-globale Zonen (Name ist frei wählbar, Zonenid wird dynamisch vergeben beim booten)

- Infrastruktur der nicht-globalen Zone wird von der globalen Zone verwaltet
- Filesysteme, Prozessoren, Netzwerke können von der globalen Zone mitgenutzt werden.
- die Filesysteme /dev /lib /platform /sbin /usr werden readonly mit dem Typ lofs eingehangen.

#### **Hinweis : Patche**

Globale Zone :

Werden Patche in der globalen Zone eingespielt, so werden sie normalerweise auch in den vorhandenen Zonen eingespielt. Abweichend dazu kann beim patchadd die Option -G (patch nur in global-Zone einspielen) oder die Variable SUNW\_PKG\_ALLZONES (Package in allen Zonen) in einem pkginfo-File gesetzt sein.

Die entsprechenden Wechselwirkungen sind in der Manualseite zu patchadd dargestellt.

Nicht-globale Zonen :

Hier gilt nur, ob die Variable SUNW\_PKG\_ALLZONES (Package in allen Zonen) in einem pkginfo-File gesetzt

ist. Siehe auch hierzu die Manualseite von patchadd.

Aufstellung von Kommandos, Hintergrundprozessen(Daemonen), Dateien und Direktories:

Kommandos :

```
/usr/bin/zonename  
/usr/sbin/zonecfg  
/usr/sbin/zoneadm  
/usr/sbin/zlogin  
/usr/sbin/pooladm  
/usr/sbin/poolcfg  
/usr/sbin/poolbind
```

Daemonen :

```
Kernel-Thread zsched  
/usr/lib/zones/zoneadmd  
/usr/lib/pool/poold
```

Dateien und Direktories :

```
/etc/zones/index  
/etc/zones/*.xml  
/etc/pooladm.conf  
Modulanfang
```

## **1-2 Zonen konfigurieren und administrieren (Grundlagen)**

Zonen Status :

```
undefined <-> configured ( delete / create )  
configured <-> installed ( uninstall / install )  
installed <-> ready -> running (halt / ready)  
running -> installed ( reboot / shutdown)
```

Anlegen einer Zone mit minimaler Konfiguration (nur zonepath)

```
# zonecfg -z zone-a  
zonecfg:zone-a> create  
zonecfg:zone-a> set zonepath=/zones/zone-a  
zonecfg:zone-a> commit  
zonecfg:zone-a> exit
```

```
# zoneadm -z zone-a install  
# zoneadm list -iv  
ID NAME STATUS PATH  
0 global running /  
- zone-a installed /zones/zone-a
```

```
# zoneadm -z zone-a boot  
# zoneadm list -iv  
ID NAME STATUS PATH  
0 global running /  
1 zone-a running /zones/zone-a  
# df -aZ
```

Beim ersten einloggen mit der Console, werden über das sysid-Tool die fehlenden Systemeinstellungen erfragt.

```
# zlogin -C zone-a
```

Normales einloggen in die Zone

```
# zlogin zone-a
```

Das herunterfahren der Zone wird ohne nachfragen durchgeführt.

```
# zoneadm -z zone-a reboot
```

```
# zoneadm -z zone-a halt
```

Zone loeschen

```
# zoneadm -z zone-a uninstall
```

```
# zoneadm -z zone-a delete
```

[Modulanfang](#)

### 1-3 Zonen konfigurieren und administrieren (Erweiterungen)

```
zonecfg -z <zonenname>
```

```
zonecfg -z <zonenname> <subcommand>
```

```
zonecfg -z <zonenname> -f <command-file>
```

```
zonecfg help
```

```
# zonecfg -z zone-a info
```

```
# zonecfg -z zone-a export
```

#### Beispiele: Nachträgliches einrichten oder ändern von Zonen-Ressourcen

Netzwerk zusätzlich einrichten :

```
# zonecfg -z zone-a
```

```
zonecfg:zone-a> add net
```

```
zonecfg:zone-a> set physical=hme0
```

```
zonecfg:zone-a> set address=10.148.6.201
```

```
zonecfg:zone-a> verify
```

```
zonecfg:zone-a> commit
```

```
zonecfg:zone-a> exit
```

Filesysteme für die nicht-globale Zone zusätzlich einrichten :

```
# zonecfg -z zone-a
```

```
zonecfg:zone-a> add fs
```

```
zonecfg:zone-a:fs> set dir=/export/applikation
```

```
zonecfg:zone-a:fs> set special=/dev/dsk/c2t0d0s7
```

```
zonecfg:zone-a:fs> set raw=/dev/rdisk/c2t0d0s7
```

```
zonecfg:zone-a:fs> set type=ufs
```

```
zonecfg:zone-a:fs> add options logging
```

```
zonecfg:zone-a:fs> end
```

```
zonecfg:zone-a> verify
```

```
zonecfg:zone-a> commit  
zonecfg:zone-a> exit
```

Verzeichnisse aus der globalen Zone in die nicht-globale Zone vererben :

```
# zonecfg -z zone-a  
zonecfg:zone-a> add inherit-pkg-dir  
zonecfg:zone-a:inherit-pkg-dir> set dir=/opt/software  
zonecfg:zone-a:inherit-pkg-dir> end  
zonecfg:zone-a> verify  
zonecfg:zone-a> commit  
zonecfg:zone-a> exit
```

Resource-Controls einrichten und Werte eintragen :

```
# zonecfg -z zone-a  
zonecfg:zone-a> add rctl  
zonecfg:zone-a:rctl> set name=zone.cpu-shares  
zonecfg:zone-a:rctl> add value (priv=privileged,limit=20,action=none)  
zonecfg:zone-a:rctl> end  
zonecfg:zone-a> verify  
zonecfg:zone-a> commit  
zonecfg:zone-a> exit
```

Attribute erzeugen und Werte eintragen :

```
# zonecfg -z zone-a  
zonecfg:zone-a> add attr  
zonecfg:zone-a:attr> set name=comment  
zonecfg:zone-a:attr> set type=string  
zonecfg:zone-a:attr> set value="Name der Zone ist zone-a"  
zonecfg:zone-a:attr> end  
zonecfg:zone-a> verify  
zonecfg:zone-a> commit  
zonecfg:zone-a> exit
```

Resource Pool eintragen :

```
# zonecfg -z zone-a  
zonecfg:zone-a> set pool=pool_zonen  
zonecfg:zone-a> verify  
zonecfg:zone-a> commit  
zonecfg:zone-a> exit
```

Änderungen werden durch das Kommando zonecfg eingetragen in  
/etc/zones/<zone>.xml

**Hinweis : Wirksam werden diese Änderungen nur nach dem reboot der Zone**

```
# cat /etc/zones/zone-a.xml  
# zoneadm -z zone-a reboot
```

[Modulanfang](#)

## 1-4 Resource-Pools konfigurieren und administrieren

Freigeben (enable) der Pool Facility, danach ist der /usr/lib/pool/poold gestartet  
Sperrern (disable) der Pool Facility, danach ist der /usr/lib/pool/poold gestoppt.

```
# pooladm -e  
# pooladm -d
```

Aktuelle (dynamische) Pool Konfiguration des Kernels ansehen  
# pooladm

Pool Konfiguration ansehen von default (/etc/pooladm.conf) oder beliebiger Datei hier (/etc/poolzonen.conf)

```
# poolcfg -c info  
# poolcfg -c info /etc/poolzonen.conf
```

Aktuelle Pool Konfiguration speichern (-s) oder laden (-c)  
default ist die Datei /etc/pooladm.conf

```
# pooladm -s [datei]  
# pooladm -c [datei]
```

Wiederherstellen der default Konfiguration  
# pooladm -x

### Beispiele: Einrichten, Konfigurieren, ändern und löschen von Pools (kein Kernelupdate mit -d)

Neuen Pool mit Prozessorset erstellen und 2 Prozessoren zuordnen.

Die Konfiguration wird in der Defaultdatei /etc/pooladm.conf abgelegt (sonst muss die Datei angegeben werden)

Soll die Konfiguration für den Kernel geladen werden, so muss pooladm -c ausgeführt werden.

```
# poolcfg -c 'create pool pool_zonen'  
# poolcfg -c 'create pset pset_zonen (uint pset.min=1; uint pset.max = 3)'  
# poolcfg -c 'associate pool pool_zonen (pset pset_zonen)'  
# poolcfg -c 'transfer to pset pset_zonen (cpu 11; cpu 15)'
```

Namen und Werte ändern, Prozessor-Sets und Pools löschen

```
# poolcfg -c 'rename pset pset_zonen to small_pset'  
# poolcfg -c 'modify pset pset_small (uint pset.max = 2)'  
# poolcfg -c 'destroy pset pset_small'  
# poolcfg -c 'destroy pool pool_zonen'
```

Wenn der Pool mit dem 'Fair share scheduler' betreut werden soll, muss dies in den Pool eingetragen werden.

Ansonsten werden die normalen Klassen verwendet (TS,IA.FX)

```
# poolcfg -c 'modify pool pool_zonen (string pool.scheduler="FSS")'
```

Die Zuweisung eines Pools zu einer Zone wird mit dem zonecfg Kommando gemacht.

Entweder beim anlegen der Zone oder später.

Bei nachträglicher Zuweisung muss die Zone neu gestartet werden oder mit dem Kommando poolbind temporär zugewiesen werden.

**Hinweis : max. 1 Pool pro nicht-globaler Zone**

[Modulanfang](#)

## 1-5 Zonen konfigurieren mit Resource-Pool und FSS (Übungsbeispiel)

Es werden 2 Zonen angelegt mit:

Netzwerk, zusätzlicher Disk-Partition, vererbtes Directory aus der global-Zone und zone.cpu-shares.

Für die Zonen wird ein eigener Resource Pool mit Prozessorset angelegt.

Der Scheduler wird dann auf FSS gesetzt

### Übung : 2 Zonen mit Resource-Pool, Prozessorset und FSS

```
# zonecfg -z zone-a
```

```
zonecfg:zone-a> create
```

```
zonecfg:zone-a> set zonepath=/zones/zone-a
```

```
zonecfg:zone-a> set autoboot=true
```

```
zonecfg:zone-a> set pool=pool_zonen
```

```
zonecfg:zone-a> add fs
```

```
zonecfg:zone-a:fs> set dir=/data
```

```
zonecfg:zone-a:fs> set special=/dev/dsk/c1t0d0s0
```

```
zonecfg:zone-a:fs> set raw=/dev/rdisk/c1t0d0s0
```

```
zonecfg:zone-a:fs> set type=ufs
```

```
zonecfg:zone-a:fs> add options logging
```

```
zonecfg:zone-a:fs> end
```

```
zonecfg:zone-a> add inherit-pkg-dir
```

```
zonecfg:zone-a:inherit-pkg-dir> set dir=/opt/oss
```

```
zonecfg:zone-a:inherit-pkg-dir> end
```

```
zonecfg:zone-a> add net
```

```
zonecfg:zone-a:net> set physical=hme0
```

```
zonecfg:zone-a:net> set address=192.168.20.122
```

```
zonecfg:zone-a:net> end
```

```
zonecfg:zone-a> add rctl
```

```
zonecfg:zone-a:rctl> set name=zone.cpu-shares
```

```
zonecfg:zone-a:rctl> add value (priv=privileged,limit=70,action=none)
```

```
zonecfg:zone-a:rctl> end
```

```
zonecfg:zone-a> add attr
```

```
zonecfg:zone-a:attr> set name=comment
```

```
zonecfg:zone-a:attr> set type=string
```

```
zonecfg:zone-a:attr> set value="Zone-a mit Apache Web-Server"
```

```
zonecfg:zone-a:attr> end
```

```
zonecfg:zone-a> commit
```

```
zonecfg:zone-a> exit
```

```

# zonecfg -z zone-b
zonecfg:zone-b> create
zonecfg:zone-b> set zonepath=/zones/zone-b
zonecfg:zone-b> set autoboot=true
zonecfg:zone-b> set pool=pool_zonen
zonecfg:zone-b> add fs
zonecfg:zone-b:fs> set dir=/data
zonecfg:zone-b:fs> set special=/dev/dsk/c1t1d0s0
zonecfg:zone-b:fs> set raw=/dev/rdisk/c1t1d0s0
zonecfg:zone-b:fs> set type=ufs
zonecfg:zone-b:fs> add options logging
zonecfg:zone-b:fs> end
zonecfg:zone-b> add inherit-pkg-dir
zonecfg:zone-b:inherit-pkg-dir> set dir=/opt/oss
zonecfg:zone-b:inherit-pkg-dir> end
zonecfg:zone-b> add net
zonecfg:zone-b:net> set physical=hme0
zonecfg:zone-b:net> set address=192.168.20.123
zonecfg:zone-b:net> end
zonecfg:zone-b> add rctl
zonecfg:zone-b:rctl> set name=zone.cpu-shares
zonecfg:zone-b:rctl> add value (priv=privileged,limit=30,action=none)
zonecfg:zone-b:rctl> end
zonecfg:zone-b> add attr
zonecfg:zone-b:attr> set name=comment
zonecfg:zone-b:attr> set type=string
zonecfg:zone-b:attr> set value="zone-b mit Datenbank"
zonecfg:zone-b:attr> end
zonecfg:zone-b> commit
zonecfg:zone-b> exit
# mkdir /opt/oss
# newfs /dev/rdisk/c1t0d0s0
# newfs /dev/rdisk/c1t1d0s0
# zoneadm -z zone-a install
# zoneadm -z zone-b install

```

Beim ersten boot der Zone wird das sysid-Tool durchlaufen, danach ist die Zone voll funktionsfähig

```

# zoneadm -z zone-a boot
# zoneadm -z zone-b boot
# zlogin -C -e ^ zone-a
# zlogin -C -e ^ zone-b

```

Freigeben (enable) der Pool Facility,  
anlegen der Konfigurationsdatei /etc/pooladm.conf mit pooladm -s

```

# pooladm -e
# pooladm -s

```

Anlegen eines Resource Pools und eines Prozessorsets



zuordnen eines Prozessors in den Prozessorset und mit dem Resource Pool verbinden  
freischalten der Fair-Share-Scheduling-Klasse für den Resource Pool  
angelegte Poolkonfiguration in den Kernel laden und die Zonen rebooten

```
# poolcfg -c 'create pool pool_zonen'  
# poolcfg -c 'create pset pset_zonen (uint pset.min=1; uint pset.max = 3)'  
# poolcfg -c 'associate pool pool_zonen (pset pset_zonen)'  
# poolcfg -c 'transfer to pset pset_zonen (cpu 11)'  
# poolcfg -c 'modify pool pool_zonen (string pool.scheduler="FSS")'  
# pooladm -c  
# zoneadm -z zone-a reboot  
# zoneadm -z zone-b reboot
```

[Modulanfang](#)

## 1-6 [Resource Management](#)

zone-a mit Apache Web-Server :

Einrichten eines Projectes und eines Users für den Webserver

```
zone-a# projadd -p 111 -G other -c "Web-Server" -K "rcap.max-rss=128MB"  
webserver  
zone-a# useradd -u 1000 -g other -d /export/home/webadmin -s /usr/bin/bash -m  
webadmin  
zone-a# passwd webadmin  
zone-a# projmod -U webadmin webserver
```

Eintragen des default Projects durch das Kommando usermod

ergibt folgende Zeile in /etc/user\_attr -> webadmin::::project=webserver

Danach ist kein Wechsel vom default project nach dem einloggen mehr nötig

```
zone-a# usermod -K "project=webserver" webadmin  
zone-a# su - webadmin  
zone-a% id -p  
uid=1000(webadmin) gid=1(other) projid=111(webserver)
```

Enablen des resource capping daemon (wird auch nach reboot gestartet)

und ansehen der Statistik

```
zone-a# rcapadm -E  
zone-a# rcapstat
```

### Beispiele:

Ändern des Projects über das erzeugen einer neuen Task mit geänderter Project-Zugehörigkeit

```
zone-a# su - webadmin  
zone-a% id -p  
uid=1000(webadmin) gid=1(other) projid=3(default)
```

```
zone-a% newtask -v -p webserver
zone-a% id -p
uid=1000(webadmin) gid=1(other) projid=111(webserver)
```

Das Kommando poolbind wird benutzt, um Zonen, Prozesse, Projekte und Tasks an einen Pools zu binden.

Außerdem kann über die Prozessnummer erfragt werden, an welchem Pool und Prozessorset der Prozess gebunden ist.

```
# poolbind -p pool_zonen -i pid 9966
# poolbind -q 9966
# poolbind -Q 9966
```

Abfragen oder setzen von Resource-Controls bei laufenden Prozessen, Tasks oder Projekten

```
prctl [-P] [-t [basic | privileged | system] ] [ -n name [-srx] [-v value] [-e | -d action]
[-p pid]] [-i idtype][id...]
```

```
# ps -o taskid -p $$
TASKID 55
```

```
# prctl -i task 55
```

#### Beispiele:

Abfragen von Ressourcen :

```
# prctl -t privileged -n zone.cpu-shares -i zone zone-a
# prctl -t privileged -n zone.cpu-shares -i zone global
# prctl -t privileged -n project.cpu-shares -i project webserver
```

Setzen von Ressourcen :

```
# prctl -n project.cpu-shares -v 30 -r -i project webserver
# prctl -n zone.cpu-shares -v 20 -r -i zone global
```

Abfragen oder ändern von 'Global States of System Resource Controls'

```
rctladm [-lu] [-e action] [-d action] [name...]
```

```
# rctladm -e syslog zone.cpu-shares
# rctladm -l
```

#### Hinweis : Siehe auch folgende man pages

```
# man user_attr
# man prof_attr
# man exec_attr
# man auth_attr
# man resource_controls
```

[Modulanfang](#)

[Dokumentanfang](#)

## 1-7 Neue Zonen-Features mit update 4 / Rel 807 bzw. update 6 Rel 1008

### Neuerungen update 4 / Rel 807

Festlegen von default-Bootargumenten:

```
# zonecfg -z zone1
```

..> set bootargs=-s *oder:*

..> set bootargs="-m multiuser=multi-user-server"

Ferner können Bootargumente auch temporär dem # zoneadm Kommando übergeben werden, z.B.:

# zoneadm -z zone1 boot -s

### Setzen von privileges, die nicht dem Standard der Zone entsprechen:

# zonecfg -z zone1

..> set limitpriv=default,sys\_time,dtrace\_proc,dtrace\_user

durch obige Einträge ist die Zone berechtigt, die Systemzeit zu ändern und dtrace Routinen auszuführen, was in unteren Releases nicht möglich war.

### Scheduling Class einrichten

# zonecfg -z zone1

..> set scheduling-class=FSS

Als Alternative zum persistenten Setzen von FSS über die Poolfacility. Dadurch kann der FSS für eine Zone gesetzt werden, ohne die Poolfacility zu konfigurieren.

Weiterhin kann so ein Prozessor mit 2 Scheduling Klassen fahren:

global = TS / IA

zone = FSS

( obwohl dies die nicht empfohlene Methode von Sun Microsystems ist ).

### Exclusive-IP Zone Network Address

# zonecfg -z zone1

..> set ip\_type=exclusive

..> add net

..net> set physical=ce0

Durch dieses Feature kann einer Zone eine NIC direkt und exklusiv zugeordnet werden ( der default "shared-ip" konfiguriert das Interface immer als virtuelle NIC, auch wenn nur die Zone diese Ressource benützt )

#### Zu beachten sind dabei 2 Punkte:

- eine IP darf hier mit zonecfg nicht vergeben werden !!

Die IP-Adress-Konfiguration erfolgt analog der Start-Dateien der globalen Zone unter /etc/hostname.xxx

- Der Data Link des Interfaces muss GLDv3 unterstützen, damit sie als exclusive IP benützt werden kann.

# dladm show-link zeigt die Version dazu.

### Physical Memory Control

# zonecfg -z zone1

..>add capped-memory

..capped-memory> set physical=50m

..capped-memory> set swap=100m

..capped-memory> set locked=30m

Die neue capped-memory Ressource setzt Limits für das physical, swap und locked memory.

Alle drei Werte sind optional und können einzeln verwendet werden:

**physical** = kappt das physical memory mit Hilfe von rcapd  
**swap** = limitiert den swap .f. Zone ( zone.max-swap resource control )  
**locked** = limitiert den Wert des locked physical memorys  
f.d Zone( zone.max-locked-memory resource )

## Neuerungen update 6 / Rel 509

### Update on attach

**# zoneadm -z zone1 attach -u [ -F ]**

Bei dem Migrieren einer Zone von system1 nach system2, kann mit der Option -u Pakete und Patche auf den Stand von system2 gebracht werden. Ein downgrade ( system1 hat neuere Pakete/Patche als system2 ) ist nicht möglich. Durch die Option -F kann die Operation zwar erzwungen werden, ein berechenbarer Betrieb der Zone ist allerdings nicht mehr gewährleistet

### Setzen eines Default-Routers in einer IP-Shared – Zone

```
.> add net  
.net> set physical=hme0  
.net> set address=192.168.2.135  
.net> set defrouter=192.168.2.1  
.net> end
```

### ZFS Zonepath

ab diesem Release kann der Zonenpfad auf einem ZFS angelegt werden und bleibt dennoch upgrade-fähig.  
Zu beachten ist dabei, dass ZFS grundsätzlich nur über Solaris Live Upgrade upgegradet werden kann.

[Modulanfang](#)

[Dokumentanfang](#)

## 1-8 BrandZ ( Branded Zonen )

Ist die Möglichkeit, in einer Zone eine andere OS-Umgebung laufen zu lassen.

Z.Zt. gibt es 4 Varianten:

- **native** => das herkömmliche Zonenmodell => Sparse oder Whole Root Model

- **lx** => Container für Linux Applikationen

Unterstützung von 32 Bit Linux Distribution auf X86 und x64- Systeme

Zertifizierte Linux Distributionen:

- CentOS 3.x

- Red Hat Linux 3.x

- **solaris 8** => Solaris 8 Environment

- **solaris 9** => Solaris 9 Environment

komplette runtime Umgebung für Solaris 8 / 9 Umgebungen ab dem Release 807 / update 4

## Installation eines solaris 9 BrandZ auf dem Release 1008 / update 6

**Download der Solaris 9 Container Software** von <http://www.sun.com/download>

### Versionen:

Solaris 9 Containers 1.0 für Release 807 und 508  
Solaris 9 Containers 1.0.1 für Release 1008 und 509  
optional kann auch ein konfiguriertes Default Flash Solaris 9 Archiv heruntergeladen werden

### Installation der Software:

```
# pkgadd -d /path/to/media SUNWs9brandr  
# pkgadd -d /path/to/media SUNWs9branu  
# pkgadd -d /path/to/media SUNWs9brandk  
- wobei die ersten beiden Pakete i.d. Regel schon installiert sind
```

### Erstellen eines Flar Archives des Solaris 9 Systemes

```
sol9 # flarcreate -S -n sol9 /net/s10/export/sol9.flar
```

### Konfiguration der Solaris 9 Zone

```
# zonecfg -z sol9  
.> create -t SUNWsolaris9  
.> set zonepath=/zonen/sol9  
.> add net  
.net> set physical=hme0  
.net> set address=192.168.2.134  
.net> end  
... falls gewünscht weitere Properties hinzufügen  
.> verify  
.> commit  
.> exit
```

### Installation der Solaris 9 Zone

```
# zonadm -z sol9 install -p -a /export/sol9.flar  
-p => preserve sysidcfg / basale Kenndaten werden erhalten  
-u => sys-unconfig / revert to blank system
```

### Booten und Einloggen in die Solaris 9 Zone

```
# zlogin -C sol9  
# zoneadm -z sol9 boot  
=> Abfrage der basalen Kenndaten, sofern Option -u übergeben wurde
```