Bandbreiten-Management in virtualisierten Netzwerken mit Solaris 11 Zonen

Martin Muschkiet
as-systeme
Stuttgart

Schlüsselworte
Solaris11, Virtualisierung, Zonen, Netzwerk, Bandbreitenmanagement, exclusive-ip, flow

Einleitung
Solaris Zonen sind eine bewährte Technologie zur OS-Virtualisierung. Mit Solaris 11 sind die Zonen noch erwachsener geworden.

Jede nicht-globale Zone verwendet jetzt per default einen eigenen Netzwerkstack. Möglich macht dies die Netzwerkvirtualisierung, mit der sich auch komplexe Netzwerktopologien abbilden lassen. Ein leicht verständliches Bandbreitenmanagement, ganz ohne komplexe Konfigurationsdateien, komplettiert die Netzwerkvirtualisierung. Netzwerkverkehr lässt sich an Hand von Attributen wie z.B. der Portnummer in *Flows* kategorisieren, diesen Flows können maximale Bandbreiten zugewiesen werden. Das Ergebnis kann mit dem „flowstat"-Kommando verfolgt werden.

Dieser Vortrag stellt Netzwerkvirtualisierung vor, erläutert wie Zonen virtualisierte Interfaces nutzen können und illustriert das Einrichten und Überwachen von Flows.

Solaris 11 Netzwerkkonfiguration: Profile und neue persistente Kommandos
Solaris 11 verfügt über einem neuen Netzwerkstack, der unter dem Namen Crossbow entwickelt wurde.

Die wichtigsten Features sind Performaneoptimierung, Netzwerkvirtualisierung, Bandbreitenmanagement, erweiterte Statistik/Accounting Funktionalität, sowie vereinfachte Konfiguration.

Die Art und Weise wie bei Solaris 11 Netzwerkinterfaces konfiguriert werden unterscheidet sich grundsätzlich von Solaris 10. Klassische Kommandos wie **ndd** und **ifconfig** sind zwar noch verfügbar, aber eigentlich entbehrlich.

Solaris 11 konfiguriert Interfaces per Profil. Mit dem Kommando **netadm list** kann das aktuelle Profil angezeigt werden.

```
# netadm list
TYPE        PROFILE     STATE
ncp         Automatic   disabled
ncp         start_state online
...(Ausgabe gekürzt)
```

Alternativ lässt sich die entsprechende Property im network/physical:default-Service anzeigen.

```
# svcprop -p netcfg/active_ncp network/physical:default
start_state
```

NWAM

Bei der Installation vom Live-Medium (CD/USB), oder der Wahl von "Automatically" in der Text-basierten Installation verwendet Solaris NWAM. Dieses Network Auto Magic eignet sich für nomadische Systeme und kann auf Ereignisse im Netzwerk (z.B. ein bestimmtes WLAN ist nicht mehr verfügbar) reagieren und eine Umkonfiguration durchführen. NWAM selbst wird per GUI "**nwam-manager-properties**" oder per CLI "**netadm**" und "**netcfg**" administriert.

Manuelle Konfiguration

Wird die Flexibilität von NWAM nicht benötigt, so kann für eine statische, manuelle Netzwerkkonfiguration das Profil "DefaultFixed" verwendet werden. Die Netzwerkinterfaces werden dann ausschließlich per **dladm** (OSI-Layer2) und **ipadm** (OSI_layer3) konfiguriert. Diese Kommandos sind persistent; Konfigurationsdateien müssen nicht editiert werden.

Das aktive Profil kann mit dem **netadm** Kommando gewechselt werden:

```
# netadm enable -p ncp DefaultFixed
Enabling ncp 'DefaultFixed'
```

Hinweis: Der Wechsel des Profils gilt für alle Netzwerkinterfaces. Nicht möglich ist z.B. 2 Interfaces per NWAM und 1 Interface manuell zu konfigurieren.

Der Service **network/physical:nwam**, mit dem man NWAM bei *OpenSolaris* und *Solaris 11 Express* aktiviert hat, ist ohne Funktion.

Im Folgenden wird davon ausgegangen, dass NWAM *nicht* aktiviert ist.

Netzwerkvirtualisierung

Die Server Virtualisierung ist im Mainstream angekommen. Sie ist häufig das Mittel der Wahl um Infrastruktur zu konsolidieren und gleichzeitig Ressourcen an einzelne Nutzer fein granular zu delegieren.

Solaris bietet ab der Version 10 die Möglichkeit der OS-Virtualisierung durch *Zonen*.

Solaris 11 erlaubt nun auch die Netzwerkvirtualisierung. Die Vorteile virtualisierter Netzwerke umfassen u.a. größere Flexibilität bei der Planung der Netzwerktopologie, verbesserte Bandbreite, geringere Latenz, die Fähigkeit Netzwerkverkehr zu priorisieren, um die angestrebten Durchsatz zu erzielen, sowie geringere Anschaffungs- und Betriebskosten.

Mit der Netzwerkvirtualisierung lassen sich virtuelle Netzwerk Interface Controller (vnics), virtuelle Switches und Bridges erstellen. Die Netzwerkstacks der einzelnen Interfaces sind isoliert, separiert und damit sicher.

VNICs

Die Funktionalität, die ein VNIC bereit stellt ist identisch mit der eines physischen Datalinks. Es hat eine eigene MAC-Adresse (in der Regel automatisch generiert), verwendet separate kstat counter, kann aggregiert oder in eine IPMP Gruppe konfiguriert werden. Auch die Administration ist weitestgehend identisch.

Virtueller Switch

Genauso wie ein physisches Interface braucht ein virtuelles Interface Konnektivität. Um dies zu erreichen wird automatisch ein impliziter virtueller Switch angelegt. Er ähnelt einem einfachen Ethernet Switch und stellt die Verbindung von NIC und VNIC her.

Zum Anlegen eines VNIC ist eine Basis (-l Option) erforderlich. Dies kann zum einen ein physischer Link sein. Der VNIC nutzt genau dieses NIC. Die Bandbreite lässt sich einfach mit einer Obergrenze

(Cap) limitieren (maxbw=).

```
# dladm create-vnic -l net0 vnic1
# dladm create-vnic -l net0 -p maxbw=10 vnic2
# dladm show-vnic
LINK    OVER    SPEED    MACADDRESS        MACADDRTYPE    VID
vnic1   net     1000     2:8:20:2a:d3:72   random         0
vnic2   net0    10       2:8:20:c5:7a:75   random         0
```

Sollen hingegen virtuelle Interfaces nur untereinander kommunizieren und ist keine Netzwerkverbindung nach außen gewünscht, so läßt sich ein VNIC auch auf Basis eines Etherstub erstellen. Ein Etherstub (engl. stub = 'lokaler Anknüpfungspunkt', 'Stichleitung') ist ein Pseudo Ethernet NIC, der unabhängig von der Hardware ist. Er verhindert, dass Daten physische Interfaces durchlaufen.

```
# dladm create-etherstub stub0
# dladm create-vnic -l stub0 vnic3
# dladm create-vnic -l stub0 vnic4
```

Soll der Netzwerkverkehr kontrolliert weitergeleitet bzw. limitiert werden, kann zwischen dem VNIC und einem physischen Interface Routing und/oder eine Firewall eingerichtet werden.

Zonen und virtuelle Netzwerkinterfaces

Damit Zonen Zugang zum Netzwerk erhalten gibt es zwei Modelle: *shared-ip* und *exclusive*.

Shared-ip bedeutet, dass ein Interface von der globalen und der nicht-globalen Zone gemeinsam genutzt wird. Dies führt jedoch zu zahlreichen Einschränkungen auf Seiten der nicht-globalen Zone, durch den gemeinsam genutzten IP-Stack.

Exclusive bedeutet weniger Einschränkungen, erfordert aber eine dezidierte Ressource für die lokale Zone. Bei Solaris 10 musste dies ein physisches Interface sein; Solaris 11 erlaubt darüber hinaus die Verwendung von einem VNIC für die exclusive Zone.

Folglich ist exclusive jetzt auch Standard bei Solaris 11 Zonen. Per default wird eine *anet* (=automatic net) Ressource verwendet. Diese sorgt dafür, dass beim Boot der Zone ein temporärer VNIC angelegt wird. Zahlreiche Properties des VNIC können direkt in der Zonenkonfiguration festgelegt werden. Im folgenden ein Beispiel einer konfigurierten *anet* Ressource:

```
global# zonecfg -z amazone info anet
anet:
linkname: net0                          #Name des Interface in der zone1
lower-link: net2                        #Physisches Interface der globalen Zone
allowed-address: 192.168.56.12/24       #Nur diese Adr. darf verwendet werden
configure-allowed-address: true         #Adr. wird beim Boot der Zone konfiguriert
link-protection: "mac-nospoof, ip-nospoof"  #Pakete müssen die konfig. MAC&IP-Adr. haben
mac-address: random                     #Beim 1. Boot wird MAC-Adr zufällig gewählt
vlan-id not specified
priority: low                           #Relative Priorität
mtu not specified
maxbw: 300m                             #Obergrenze der verfügbaren Bandbreite
```

Wird die Zone installiert und gebootet, so wird das automatisch angelegte Interface von **dladm** angezeigt, genauso wie persistent angelegte VNICs (vnic2):

```
global# dladm show-link
LINK            CLASS     MTU  STATE    OVER
net0            phys      1500 up       --
net1            phys      1500 up       --
zone1/net1      phys      1500 up       --          #zone1 nutzt net1 exclusiv
stub0           etherstub 9000 unknown  --
vnic2           vnic      9000 up       stub0       #zone2 nutzt vnic2
zone2/vnic2     vnic      9000 up       stub0
amazone/net0    vnic      1500 up       net0        #amazone nutzt anet
```

Das **ipadm** Kommando hingegen zeigt nur Informationen der aktuellen Zone. Per zlogin lässt sich die IP-Adresse der amazone darstellen.

```
global# ipadm show-addr
ADDROBJ           TYPE     STATE       ADDR
lo0/v4            static   ok          127.0.0.1/8
net0/v4           static   ok          192.168.0.112/24
lo0/v6            static   ok          ::1/128
global# zlogin amazone ipadm show-addr
ADDROBJ           TYPE     STATE       ADDR
lo0/v4            static   ok          127.0.0.1/8
net0/v4           static   ok          192.168.0.118/24
lo0/v6            static   ok          ::1/1
```

Ressourcenmanagement

Netzwerkresssourcemanagement ermöglicht es die Verteilung der vorhandenen Ressourcen zu steuern, um vereinbarte Anforderungsprofile zu erfüllen. Dies können zum einen CPU-Ressourcen sein, die den NICs zugeordnet werden. Zum anderen lässt sich die Bandbreite von physischen und virtuellen Datalinks beschränken. Das in Solaris 11 stark erweiterte dladm Kommando kann die folgenden Eigenschaften setzen:

tx-rings/rx-rings **txrings=<Anzahl der Ringe>**

10GB Interfaces verfügen in der Regel über diskrete TX/RX Rings. Diese Hardwareressurcen können an VNICs vergeben werden. Auf einem physischem Interface können so -wie auf einer breiten Straße- mehrere Fahrspuren "Lanes" parallel nebeneinander eingerichtet werden. Der Netzwerkverkehr von der Applikation bis zum physischen Interface ist separiert.

Bandbreite **maxbw=<Wert in Mbit>**

Eine Obergrenze (Cap) wird für den NIC/VNIC eingerichtet, "overcommitment" ist möglich

Priorität **priority={low|medium|high}**

Die relative Priorität verglichen mit anderen Nutzern des NICs. Bei *low* werden Pakete bei hoher Auslastung eher gedroppt.

Bindung des NIC an Ressource Pool **pool=<poolname>**

Die CPUs, die die Karte bedienen stammen aus einem Ressourcepool. Bei der Zuordnung des VNICs an eine Zone wird automatisch der CPU-Pool der Zone verwendet.

Bindung des NIC an **CPUs cpus=<cpu-id>**

Werden keine CPU-Pools verwendet, lässt sich steuern, welche CPUs den NIC bedienen.
Hier beispielhaft das Setzen von zwei Properties.

```
# dladm set-linkprop -p maxbw=200,cpus=0,1 vnic0
# dladm show-linkprop -p maxbw,cpus vnic0
LINK     PROPERTY PERM    VALUE     DEFAULT   POSSIBLE
vnic0    maxbw    rw      200       --        --
vnic0    cpus     rw      0-1       --        --
```

Flows

Ein Flow charakterisiert eine bestimmte Art von Netzwerkverkehr. Zum Beispiel alle Pakete, die als Ziel den lokalen Port 80 haben. Oder ausgehende Pakete, die ein bestimmtes System als Ziel haben. Flows lassen sich mit Obergrenzen in der Bandbreite limitieren.

Engpässe durch Applikationen, die nur gelegentlich laufen, dann aber andere Applikationen beeinträchtigen würden, lassen sich so eindämmen. Auch eine äußerst einfache Firewall, die einen bestimmten Netzwerkverkehr auf Null setzt, lässt sich so ohne komplexes Regelwerk realisieren.

Des weiteren kann der Datenverkehr, der den konfigurierten Flows entspricht gemessen werden.

Zum Charakterisieren des Flows werden Attribute verwendet:

IP-Adresse **local_ip=<IP-Adr>| remote_ip=<IP-Adr>**

Protokoll **transport={tcp|udp|sctp|icmp|icmpv6}**

Port **local_port=<Port-nr> | remote_port=<Port-Nr>**

Die Kombination von Attributen ist nur eingeschränkt möglich, Flowattribute können nachträglich nicht geändert werden. Die Bandbreite ist derzeit der einzige Wert (Property), der für den Flow konfiguriert werden kann.

Beispiel zum Anlegen zweier konkurrierender Flows auf dem gleichen vnic:

```
# flowadm add-flow -l vnic0 -a transport=tcp,local_port=80 httpflow
# flowadm add-flow -l vnic0 -a transport=tcp,local_port=2049 nfsflow
# flowadm show-flow
FLOW     LINK     IPADDR   PROTO    LPORT    RPORT    DSFLD
httpflow vnic0    --       tcp      80       --       --
nfsflow  vnic0    --       tcp      2049     --       --
# flowadm set-flowprop -p maxbw=200 httpflow
# flowadm set-flowprop -p maxbw=500 nfsflow
# flowadm show-flowprop
FLOW     PROPERTY VALUE    DEFAULT  POSSIBLE
httpflow maxbw    200      --       --
nfsflow  maxbw    500      --       --
# flowstat
FLOW     IPKTS    RBYTES   IDROPS   OPKTS    OBYTES   ODROPS
httpflow 1,53K    134,71K  0        828      105,27K  0
nfsflow  0        0        0        0        0        0
```

Neben der Flow-Statistik lässt sich der Datenverkehr auch per Interface überwachen.
dlstat zeigt Informationen über ein- und ausgehende Pakete an

```
# dlstat
LINK IPKTS RBYTES OPKTS OBYTES
net0 0 0 87 7,37K
net1 4,30K 378,17K 2,50K 367,30K
```

Wenn man das Accounting für Netzwerkverkehr aktiviert, lassen sich auch zurückliegende Ereignisse darstellen.

```
# acctadm -e extended -f /var/log/net.log net
# acctadm net
Net accounting: active
Net accounting file: /var/log/net.log
Tracked net resources: extended
Untracked net resources: none
```

In der Vergangenheit liegende (-h Option >history<) Angaben zum Flow nfsflow

```
# flowstat -h -s 04/16/2012,13:31:00 -e 04/16/2012,13:32:00 -f /var/log/net.log
FLOW START END RBYTES OBYTES BANDWIDTH
nfsflow 13:30:47 13:31:07 22218 162016 0.073 Mbps
nfsflow 13:31:07 13:31:27 4200 5496 0.003 Mbps
nfsflow 13:31:27 13:31:47 6912 6684 0.005 Mbps
```

Fazit

Solaris 11 komplettiert mit der Netzwerkvirtualisierung die OS-Virtualisierung mit Zonen. Leicht verständliche, gut strukturierte Befehle helfen virtuelle Interfaces und Ressourcemanagement aufzusetzen.

Kontaktadresse:
Martin Muschkiet
AS-SYSTEME GmbH
Stuttgarter Engineering Park
Wankelstrasse 1
D-70563 Stuttgart

Phone: +49 711 90146 0
Fax:Fax: +49 711 90146 42
E-Mail: info@as-systeme.de
Internet: www.as-systeme.de