



Advances in OpenSolaris Network Administration

Solaris Networking

Sun Microsystems, Inc.

September 2007



Overview

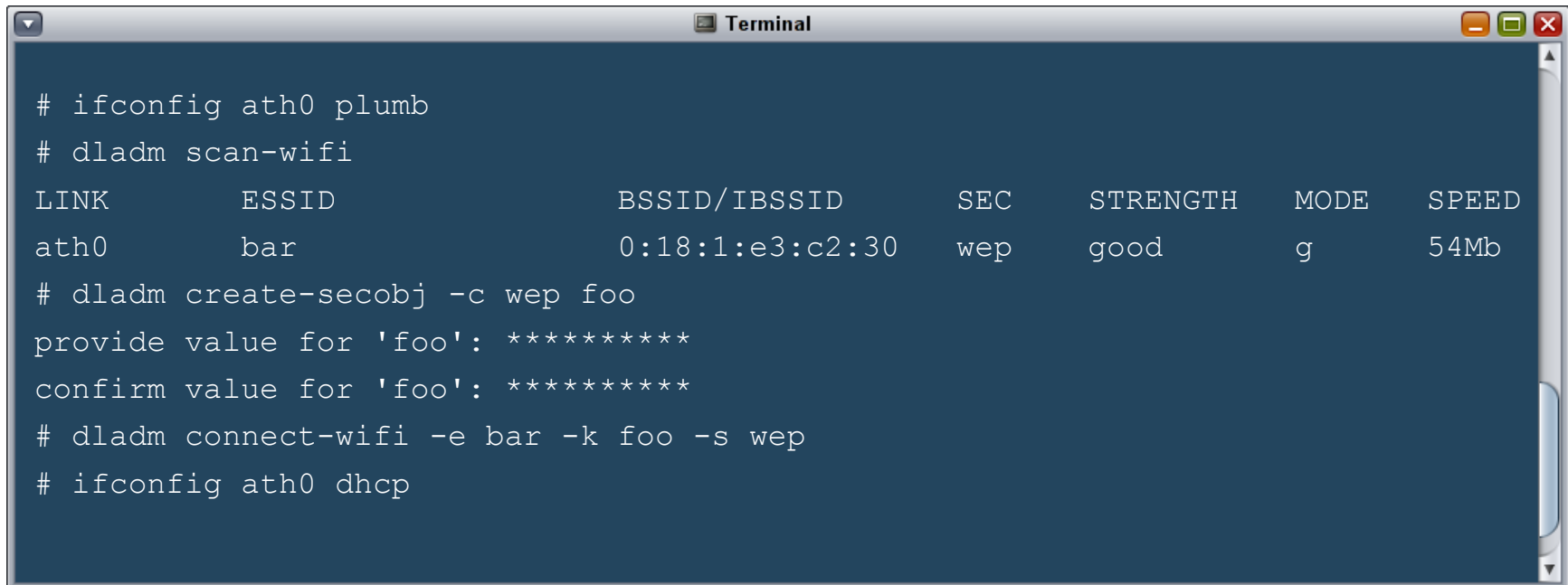
- Make OpenSolaris a more compelling platform for developers, administrators, and users.
- Reduce barriers to Solaris adoption by:
 - > Making network configuration easier (Network Auto-Magic project)
 - > Providing a uniform set of features on all network interfaces (project Clearview)
 - > Simplifying NIC configuration and tuning (project Brussels)
 - > Integrating virtualization & resource management into the network interface (project Crossbow)

Network Auto-Magic

Automating Network Configuration

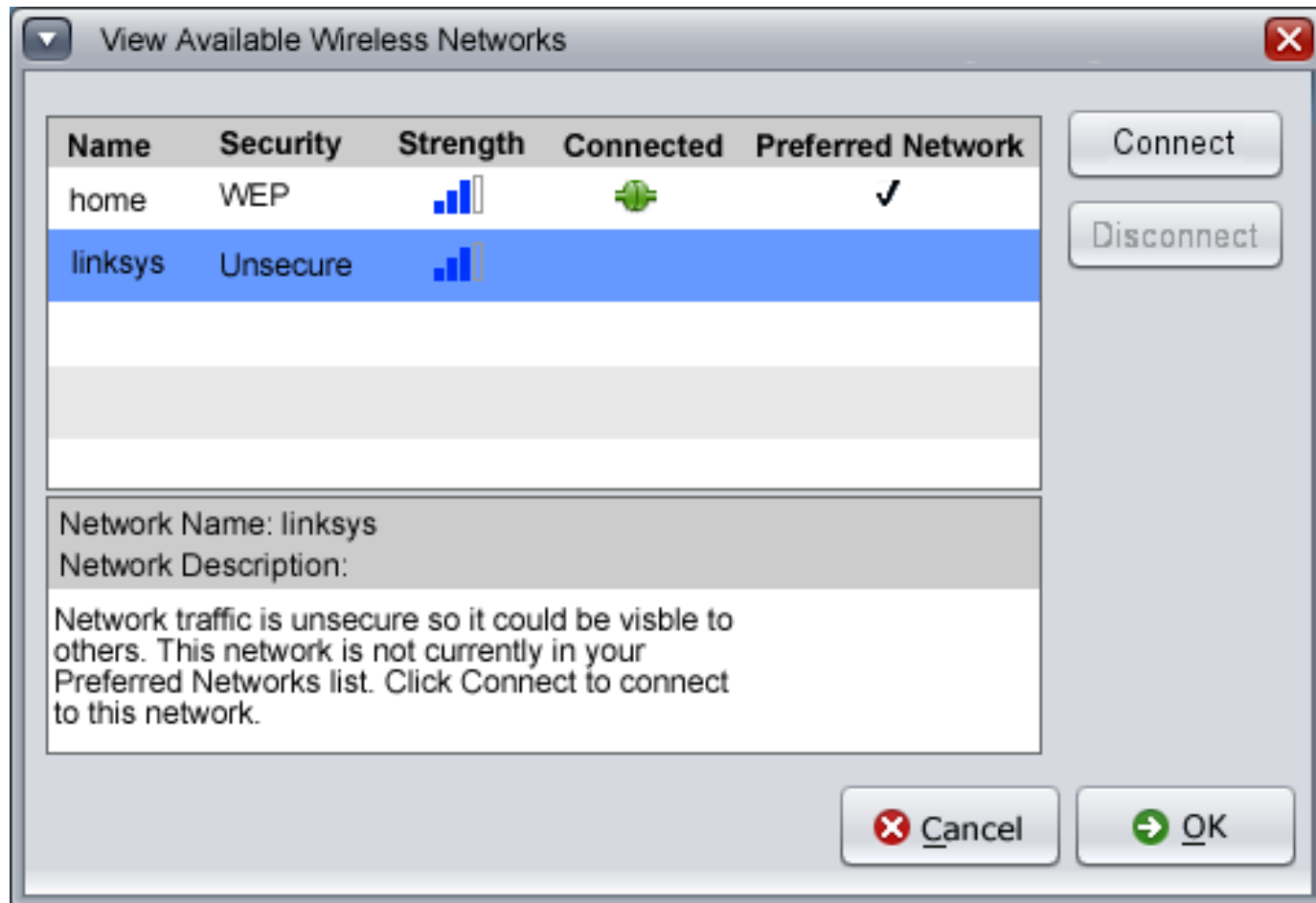
Background

- It has long been painful to configure networking on Solaris. Mobility and security makes it harder:



```
# ifconfig ath0 plumb
# dladm scan-wifi
LINK          ESSID          BSSID/IBSSID    SEC    STRENGTH    MODE    SPEED
ath0          bar              0:18:1:e3:c2:30  wep    good        g        54Mb
# dladm create-secobj -c wep foo
provide value for 'foo': *****
confirm value for 'foo': *****
# dladm connect-wifi -e bar -k foo -s wep
# ifconfig ath0 dhcp
```

Why Can't This “Just Work”?



Obviously... But why stop?

- During their day Solaris users encounter many different environments.
 - > Home
 - > Coffee Shop
 - > Work
- And from each they might use...
 - > VPNs
 - > Varying security products
 - > Varying name services
- Why can't they just work also?

NWAM

- Network Auto-Magic is an OpenSolaris project to simplify and automate network configuration
 - > Basic principle: network configuration just works
 - > Networking should be easy to use from the moment Solaris is installed
 - > System can automatically configure itself for networks as they become available
 - > User has the choice to override default system behavior and set preferences

When Configuration is Needed

- Two areas of configuration:
 - > Devices and interfaces: the Network
 - > Services and their properties: the Environment
- Can mix-and-match: a single Environment can be applied over different underlying Networks
- The Network area can include dependencies: when a new link becomes available...
 - > create a tunnel
 - > run an arbitrary script

What Users See By Default

- System automatically chooses an interface and uses DHCP to configure IP
- Wired is preferred over wireless
- DHCP requests are done in parallel so that delays are minimized
- If the nwam service is enabled, then /etc/hostname.<intf> files are ignored

What You Can Do

- Create Environments for the different places you go:
 - > Home
 - > Coffee Shop
 - > Work
- After doing some surfing at home with the Home Environment enabled, you decide to get some work done, and enable your VPN
- The tunnel is detected, triggering a switch to the Work Environment

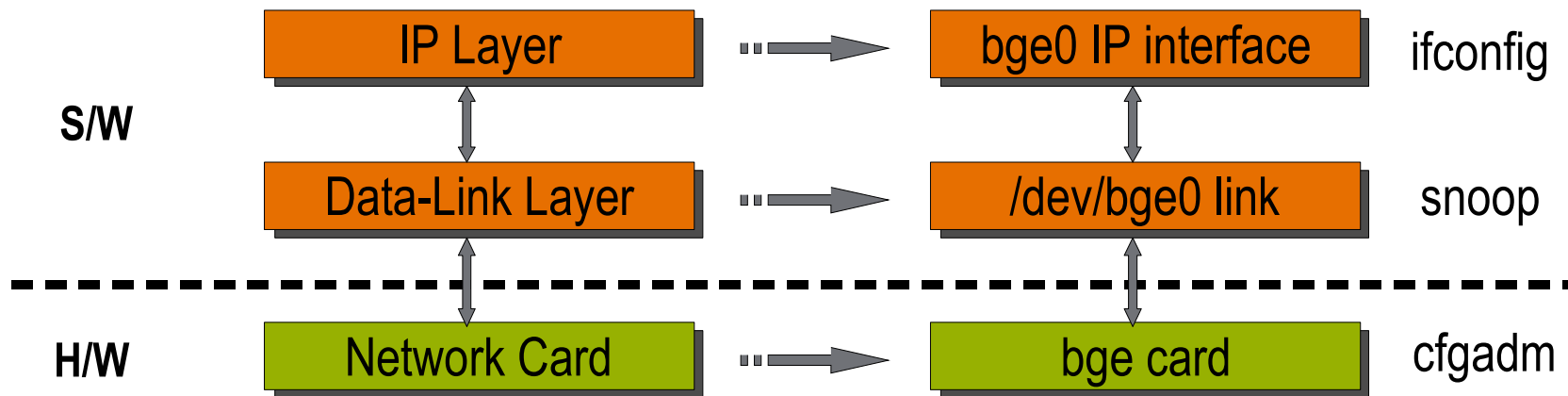
NWAM: More Information

- NWAM OpenSolaris Home
 - > <http://opensolaris.org/os/project/nwam/>
- Mailing List
 - > nwam-discuss@opensolaris.org

Project Clearview

Unified Set of Network Interface Features

What is a Network Interface?



Project Clearview

- Unify, simplify, and enhance the features provided by Solaris networking interfaces
 - > “Network interfaces” as in ce, bge, tun, ...
- Goals:
 - > Unify network interface feature set
 - > Simplify network interface administration
 - > Enhance observability of network interfaces
 - > Increase interoperability between networking features
 - > Improve third-party network application capture

Network Interfaces: Complaints

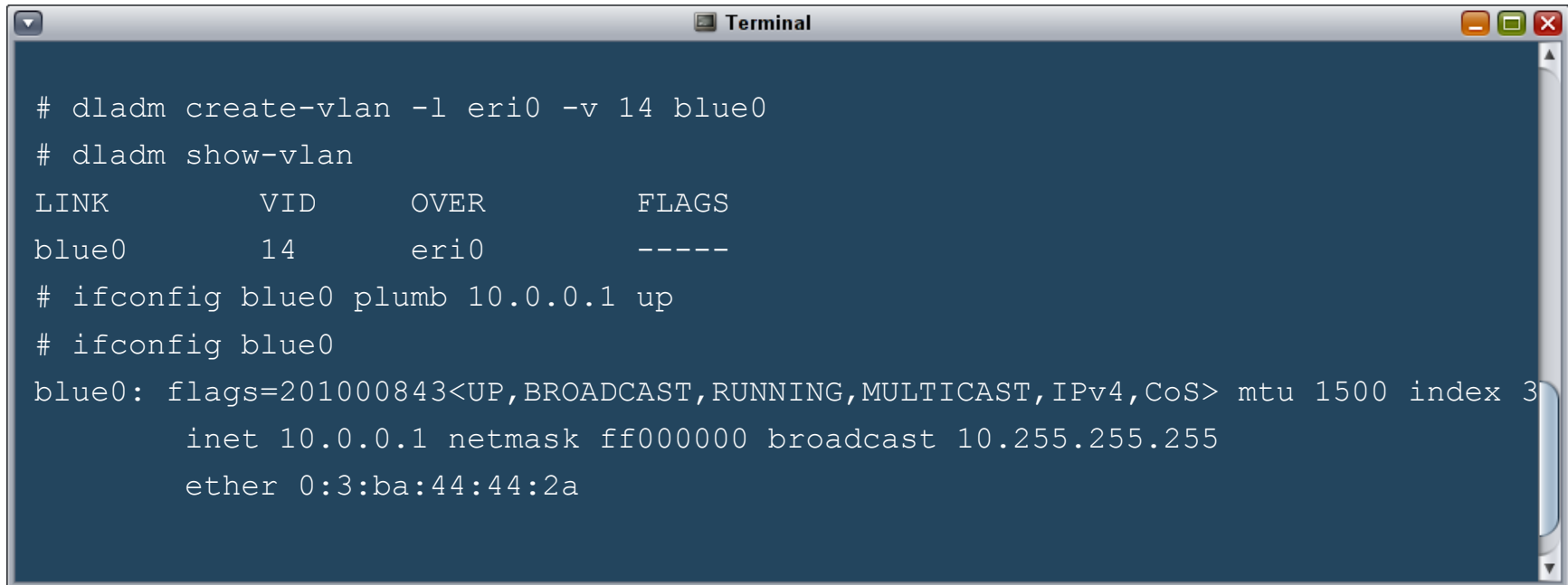
- 802.1q VLAN's work with an arbitrary subset of Ethernet networking interfaces.
- 802.3ad Link Aggregation support is even worse:
 - > Some links are aggregated with `d1adm(1M)`
 - > Others are aggregated with the unbundled `nettr(1M)`
 - > Many cannot be aggregated at all!
- Packets cannot be seen on all network interfaces
 - > Cannot see traffic for loopback, tunnels, or IPMP groups
- Network configuration is chipset-dependent
 - > e.g., upgrading `hme` to `bge` means changing `ipfilter` rules

Network Interfaces: More Complaints

- Only some data links are administered with `dladm`
 - > Some – such as IP tunnels – are buried in `ifconfig`
 - > Many cannot be directly administered at all.
- Solaris IPMP – a key part of many high-availability networking deployments – often cannot be used because its odd network interface model breaks:
 - > Dynamic routing daemons
 - > IPsec IKE daemons
 - > IPv6 autoconfiguration
 - > DHCP clients
 - > ... and **countless** third-party applications

Use VLANs on all Ethernet Links

- If it's Ethernet, you can create a VLAN over it!



```
# dladm create-vlan -l eri0 -v 14 blue0
# dladm show-vlan
LINK          VID      OVER      FLAGS
blue0         14      eri0      -----
# ifconfig blue0 plumb 10.0.0.1 up
# ifconfig blue0
blue0: flags=201000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,CoS> mtu 1500 index 3
      inet 10.0.0.1 netmask ff000000 broadcast 10.255.255.255
      ether 0:3:ba:44:44:2a
```

802.3ad Link Aggregations on any set of Ethernet Links

- If it's Ethernet, you can aggregate!

```

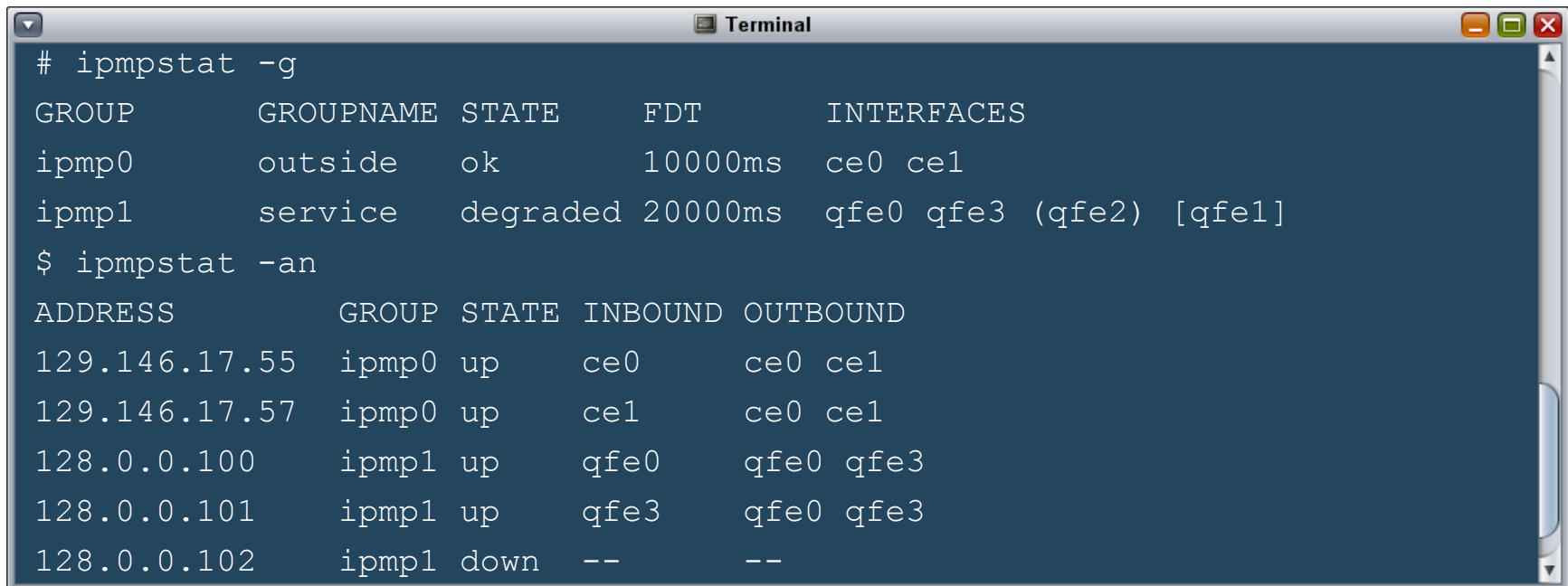
# dladm create-aggr -l bge0 -l ce0 customer3
# dladm show-link customer3
LINK          CLASS    MTU      STATE    OVER
customer3     aggr      1500     unknown  bge0 ce0
# dladm show-aggr
LINK          POLICY  ADDRPOLICY          LACPACTIVITY  LACPTIMER  FLAGS
customer3     L4      auto              off           short      -----
# ifconfig customer3 plumb
  
```

Give Interfaces Meaningful Names

- System configuration containing interface names no longer tied to specific system or hardware
- Assign meaningful names to
 - > physical data-link interfaces
 - > `dladm rename-link bge0 admin3`
 - > VLANs
 - > Link Aggregations
 - > IP tunnels
 - > Crossbow VNICs
 - > IPMP interfaces

Improved IPMP Administration

- Represent an IPMP group as a network interface
 - > Improves interoperability with other networking features such as dynamic routing and DHCP
- New `ipmpstat` command:



```
# ipmpstat -g
GROUP      GROUPNAME  STATE      FDT        INTERFACES
ipmp0      outside    ok         10000ms    ce0 ce1
ipmp1      service    degraded   20000ms    qfe0 qfe3 (qfe2) [qfe1]
$ ipmpstat -an
ADDRESS      GROUP  STATE  INBOUND  OUTBOUND
129.146.17.55 ipmp0  up     ce0      ce0 ce1
129.146.17.57 ipmp0  up     ce1      ce0 ce1
128.0.0.100    ipmp1  up     qfe0     qfe0 qfe3
128.0.0.101    ipmp1  up     qfe3     qfe0 qfe3
128.0.0.102    ipmp1  down   --       --
```

Observe Packets Over any Interface

- Clearview allows observability over interfaces previously not possible
- Loopback
 - > snoop -d lo0
- IP tunnel
 - > snoop -d vpn3
- IPMP group interface
 - > snoop -I ipmp2

Observe Packets Between Zones

- Problems with zone networking observability today:
 - > Cannot observe packets from a zone to another host
 - > Cannot observe packets from a zone to another zone
 - > Cannot observe packets flowing within a zone
- Clearview enables such observability using traditional network observability tools such as snoop, wireshark, etc.

Project Clearview: More Information

- OpenSolaris Clearview Project
 - > <http://opensolaris.org/os/projects/clearview>
 - > Overview; design documents; links to design discussion
- Mailing List
 - > clearview-discuss@opensolaris.org

Brussels Project

Simple NIC Configuration and Tuning

Brussels Project

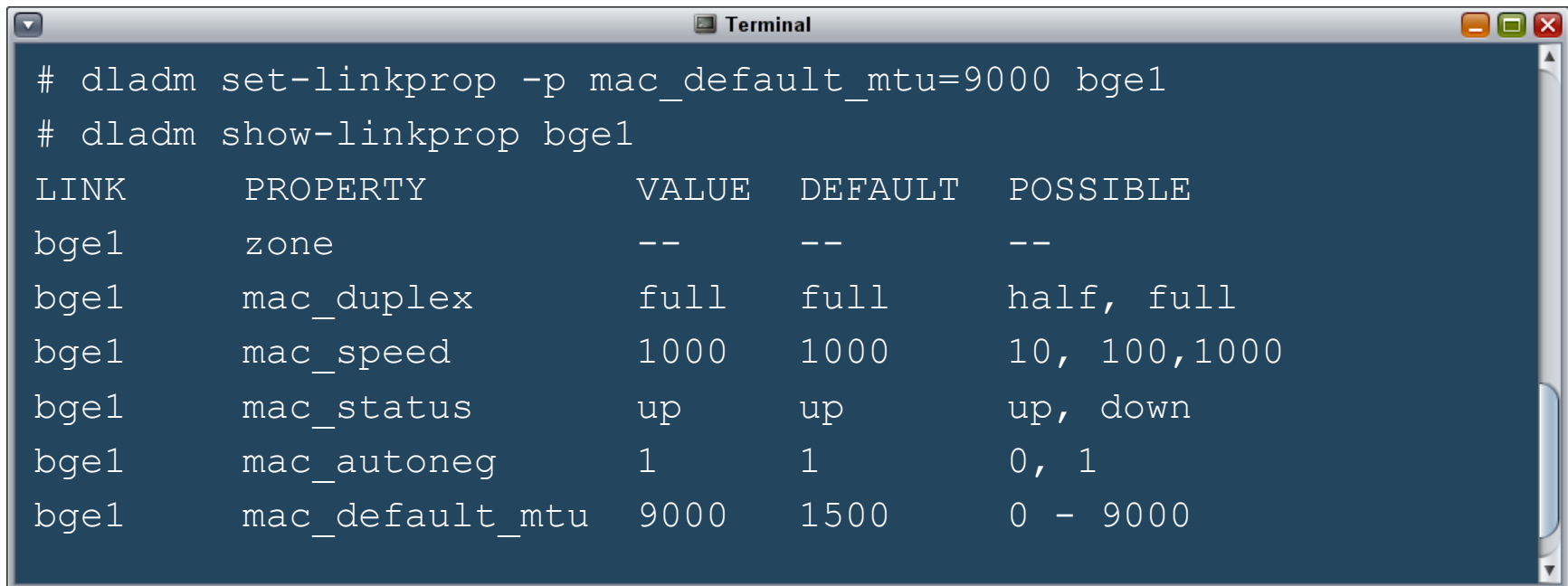
- NIC configuration and tuning is a mess:
 - > `/kernel/drv/*.conf`
 - > `ndd(1M)`
 - > SPARC OBP
 - > `kstat(1M)`
- Methods of configuration for common features are different between drivers; confusing to administrators

Brussels Solution

- All NIC configuration and tuning via `dladm(1M)` using “link properties”.
- Common properties in scope:
 - > Link MTU (including Jumbo Frame configuration)
 - > Link Speed
 - > Link Duplex
 - > Hardware Checksum Offload
 - > etc...

Example of Brussels Simplicity

- Increasing the MTU of the `bge1` interface to enable jumbo frames is done with a single `dladm(1M)` command:



```
# dladm set-linkprop -p mac_default_mtu=9000 bge1
# dladm show-linkprop bge1
```

LINK	PROPERTY	VALUE	DEFAULT	POSSIBLE
bge1	zone	--	--	--
bge1	mac_duplex	full	full	half, full
bge1	mac_speed	1000	1000	10, 100, 1000
bge1	mac_status	up	up	up, down
bge1	mac_autoneg	1	1	0, 1
bge1	mac_default_mtu	9000	1500	0 - 9000

Brussels: More Information

- Brussels OpenSolaris Home
 - > <http://opensolaris.org/os/project/brussels/>
- Mailing List
 - > brussels-dev@opensolaris.org

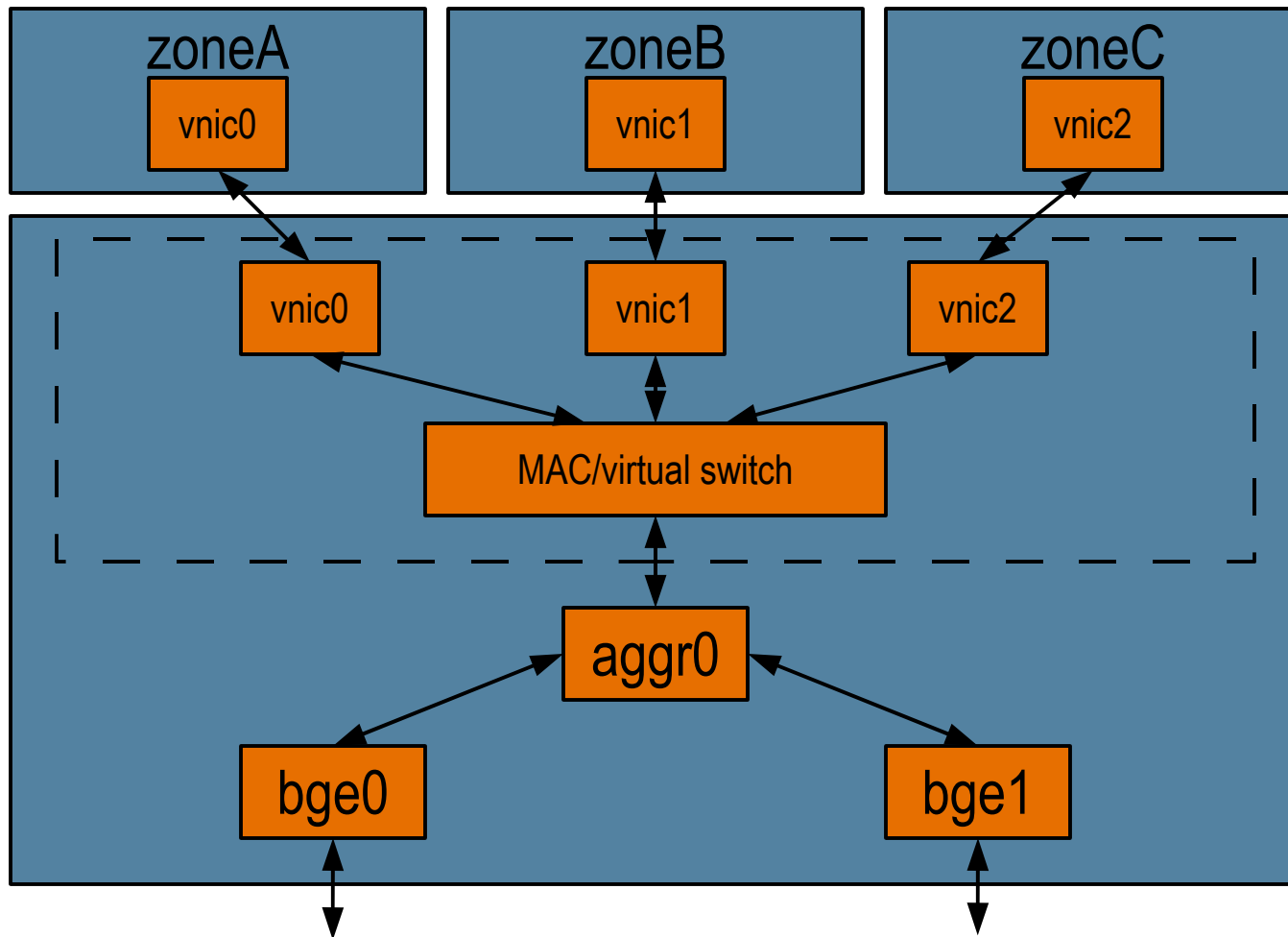
Project Crossbow

NIC Virtualization and Resource Management

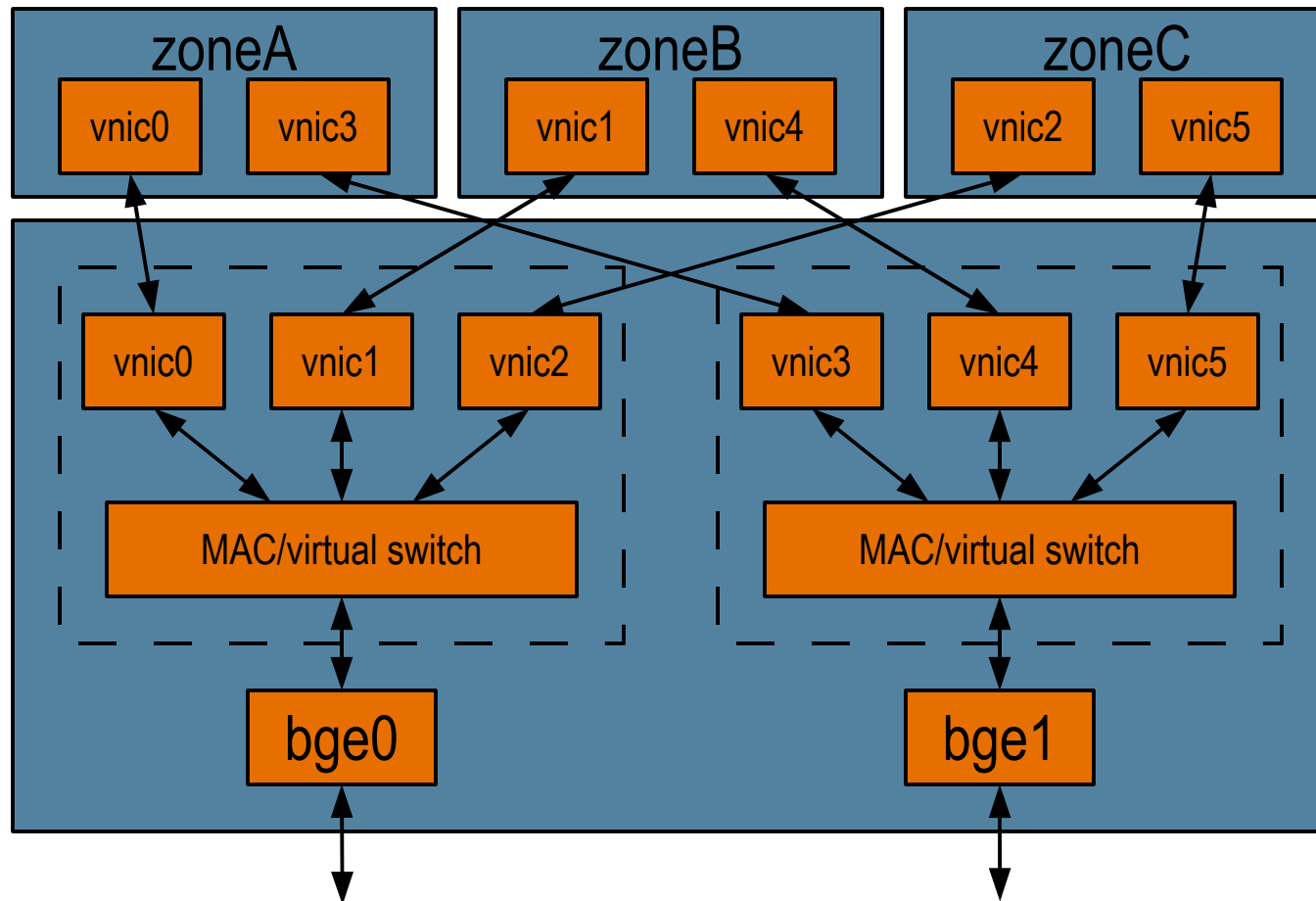
Crossbow Features

- NIC and network stack virtualization (VNICs)
- Resource partitioning, QoS/Diffserv
- Leverages hardware classification
- Better defense against DDOS attacks
- Real-time usage and history
- Allows VNICs to be plumbed by Solaris zones or virtual machines running under Solaris

Virtualized Networking



Virtualized Networking



Example VNIC Usage

- Creating VNICs is simple
- Done using `dladm(1M)`, as with other data-link interface administration

```

Terminal
# dladm create-vnic -l bge1 vnic1
# dladm create-vnic -l bge1 -m random -p maxbw=100M -p cpus=4,5,6 vnic2
# dladm show-vnic
LINK          OVER      MACTYPE    MACVALUE      BANDWIDTH     CPUS
vnic1         bge1      factory    0:1:2:3:4:5    -             -
vnic2         bge1      random     2:5:6:7:8:9    max=100M      4,5,6
# zonecfg -z zone1
zonecfg:zone1> set ip-type=exclusive
zonecfg:zone1> add net
zonecfg:zone1:net> setphysical=vnic1
zonecfg:zone1:net> end

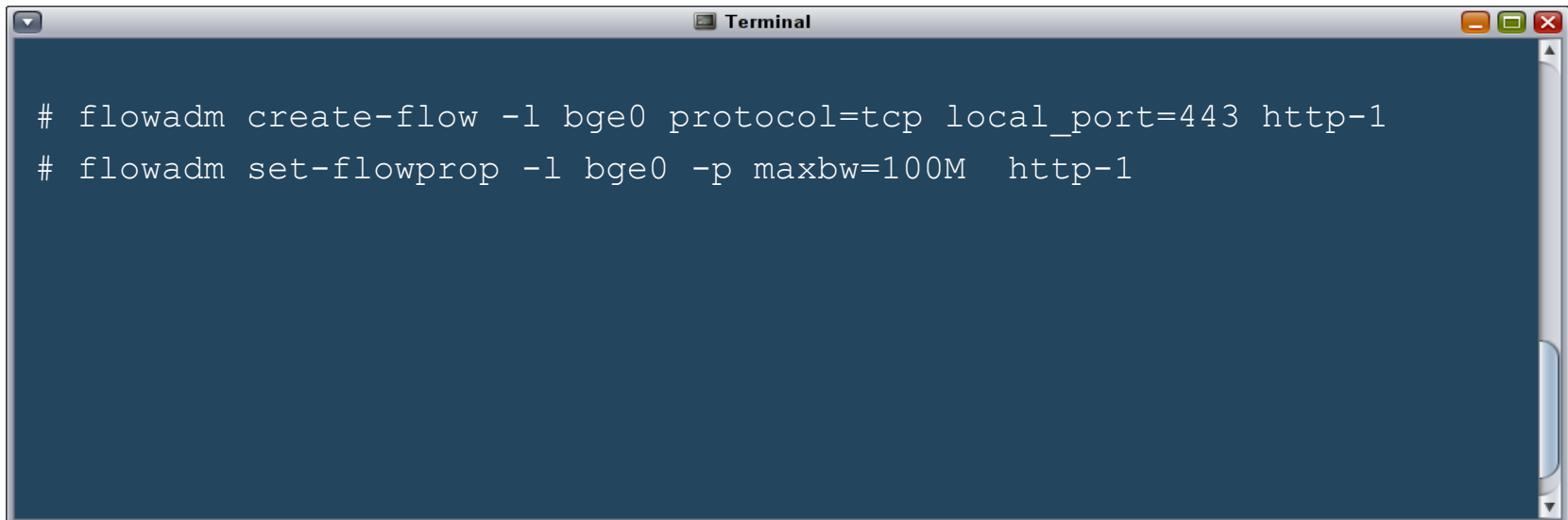
```

Bandwidth Partitioning & Accounting

- Bandwidth limits and priorities can be assigned to NICs, VNICs, protocols, or services
- Specified using `dladm(1M)` or `flowadm(1M)`
- Finer grain accounting comes for free
- Can track utilization of individual NICs and VNICs, services, and protocols
- The Solaris extended accounting framework (exacc) maintains per flow and NIC accounting

Example Flow Creation

- Flows are used to define packet classifications to which bandwidth limits and priorities may be applied
- Below, we simply create a bandwidth-limited HTTP flow for the bge0 interface:

A terminal window titled "Terminal" with a dark blue background and white text. It contains two lines of commands: "# flowadm create-flow -l bge0 protocol=tcp local_port=443 http-1" and "# flowadm set-flowprop -l bge0 -p maxbw=100M http-1".

```
# flowadm create-flow -l bge0 protocol=tcp local_port=443 http-1
# flowadm set-flowprop -l bge0 -p maxbw=100M http-1
```

Crossbow: More Information

- Crossbow OpenSolaris Home
 - > <http://www.opensolaris.org/os/project/crossbow/>
- Mailing List
 - > crossbow-discuss@opensolaris.org

Related OpenSolaris Networking Projects

- Quagga Routing Protocol Suite
 - > <http://www.opensolaris.org/os/project/quagga/>
- RBridge (IETF TRILL) Support
 - > <http://www.opensolaris.org/os/project/rbridges/>
- Virtual Network Machines
 - > <http://www.opensolaris.org/os/project/vnm/>
- OpenSolaris Networking Community
 - > <http://www.opensolaris.org/os/community/networking/>



Advances in Solaris Network Administration

networking-discuss@opensolaris.org

