Simple OSPF configuration

Prosta konfiguracja dynamicznego protokołu rutingu OSPF

- W ramach zajęć zostanie utworzona sieć składająca się z czterech maszyn wirtualnych
- Studenci nauczą się, w jaki sposób uruchomić kilka maszyn wirtualnych jednocześnie z wykorzystaniem oprogramowania **`netkit`**
- Maszyny zostaną wyposażone w podstawową konfigurację, jeszcze przed ich uruchomieniem

Simple OSPF configuration

Prosta konfiguracja dynamicznego protokołu rutingu OSPF

- W ramach zajęć należy:
  - Skonfigurować protokół OSPF działający w jednym obszarze (ang. ***area***)
  - Zrozumieć przeznaczenie interwałów ***hello-interval*** oraz ***dead-interval*** protokołu OSPF
  - Zrozumieć funkcje rutera wyróżnionego (ang. ***designated router***, **DR**)
  - Zrozumieć działanie parametrów protokołu OSPF: ***router-id***, ***priority***
  - Zrozumieć działanie protokołu OSFP w przypadku sieci rozgłoszeniowych (ang. *broadcast*) oraz typu punkt-punkt (ang. *point to point*)
  - Przechwycić i przeanalizować wymianę komunikatów protokołu OSPF

Simple OSPF configuration

Prosta konfiguracja dynamicznego protokołu rutingu OSPF

- **Głównym celem zajęć jest**
  - zapoznanie się z działaniem dynamicznego protokołu rutingu OSPF opartego na algorytmie stanu łącza (ang. *link state*)
  - nabycie umiejętności konfigurowania i analizowania protokołu w celu zapewnienia osiągalności urządzeń sieciowych oraz diagnozowania przyczyn trudności w prawidłowym doborze trasy

Preparing a netkit lab

Authors: G. Di Battista, M. Patrignani, M. Pizzonia, M. Rimondini

Modified and extended for the purpose of the IP Networks lab

Simple OSPF configuration

Authors: K. Kosek-Szott, P. Pacyna, S. Szott

Introduction

preparing a lab

- a netkit lab is a set of preconfigured virtual machines (VMs) that can be started and halted together
- we will learn how to set up a standard netkit lab that can be launched by using the **`lcommands`**
  - instructions based on http://wiki.netkit.org/netkit-labs/netkit_introduction/netkit-introduction.pdf
- we will create a **simple OSPF** configuration

netkit labs using `lcommands`

- a standard netkit lab is a directory tree containing
  - a **`lab.conf`** file describing the network topology
  - a set of subdirectories that contain the configuration settings for each virtual machine
  - **`.startup`** file that describe actions performed by virtual machines when they are started
- **Download tar with a sample netkit lab**
- **List the files in the lab directory**

`lab.conf`

- this file describes
  - the settings of the VMs that make up a lab
  - the topology of the network that interconnects the VMs of the lab
- list of **`machine[arg]=value`** assignments
  - **`machine`** is the name of the vm (e.g., **`pc1`**)
  - **`arg`** is an integer number (say ***`i`***)
  - **`value`**is the name of the collision domain to which interface ***`eth i`*** should be attached

- **Note**: a collision domain represents a multi-access network such as, for example, computers connected to an Ethernet switch

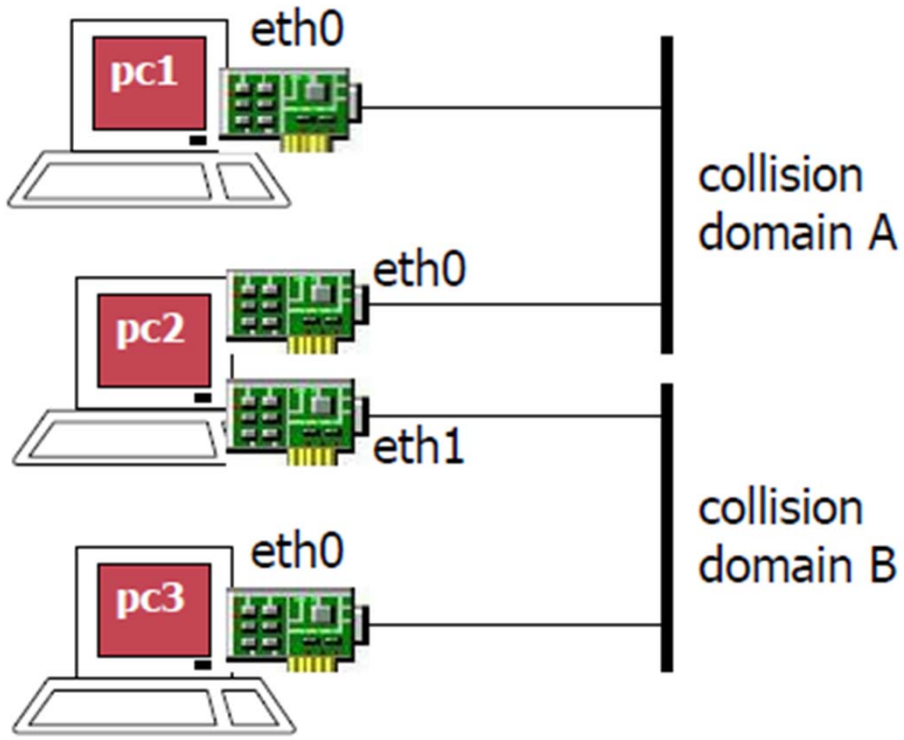
lab.conf

- example of **lab.conf**

```
pc1[0]=A

pc2[0]=A
pc2[1]=B


pc3[0]=B
```



lab subdirectories

- netkit starts a VM for each subdirectory, with the same name of the subdirectory itself

- contents of subdirectory **`vm`** are copied into the root (**`/`**) of **`vm`**'s filesystem
  - for example, **`vm/foo/file.txt`** is copied to **`/foo/file.txt`** inside virtual machine **`vm`**
  - this only happens the 1st time the **`vm`** is started; in order to force the copying you have to remove the **`vm`** filesystem (**`.disk`** file)

startup files

- shell scripts that tell VMs what to do when starting up or shutting down
- they are executed inside VMs
- upon startup, a vm named **vm_name** runs
  - **vm_name.startup**
- a **.startup** file can be used, for example, to configure network interfaces and/or start network services
- sample of **vm_name.startup**

```
ifconfig eth0 10.0.0.1 up
/etc/init.d/zebra start
```

launching/stopping a lab

- **launching the lab**
  - go to the lab directory (**`cd lab_directory`**)
  - in the terminal enter the **`lcommand`**
- where **`lcommand`** is one of the following
  - **`lstart`**, to start the lab
  - **`lhalt`**, to gracefully shut down the VMs of a lab
  - **`lcrash`**, to suddenly crash the VMs of a lab

removing temporary files

- a running lab creates some temporary files inside both the current directory and the lab directory
- to get rid of them all, use **`lclean`** after the lab has been halted/crashed
  - notice: **`lclean`** also removes virtual machine filesystems (**`.disk`** files); do not use it if you are going to launch your lab again using the same (previously configured) filesystems

ltest

- makes it easier to check that labs work properly
- **`ltest`** starts a lab and dumps information about each virtual machine **`vm`**
  - the output goes into **`_test/results/vm.default`**
  - after configuring your lab check the contents of **`*.default`** files

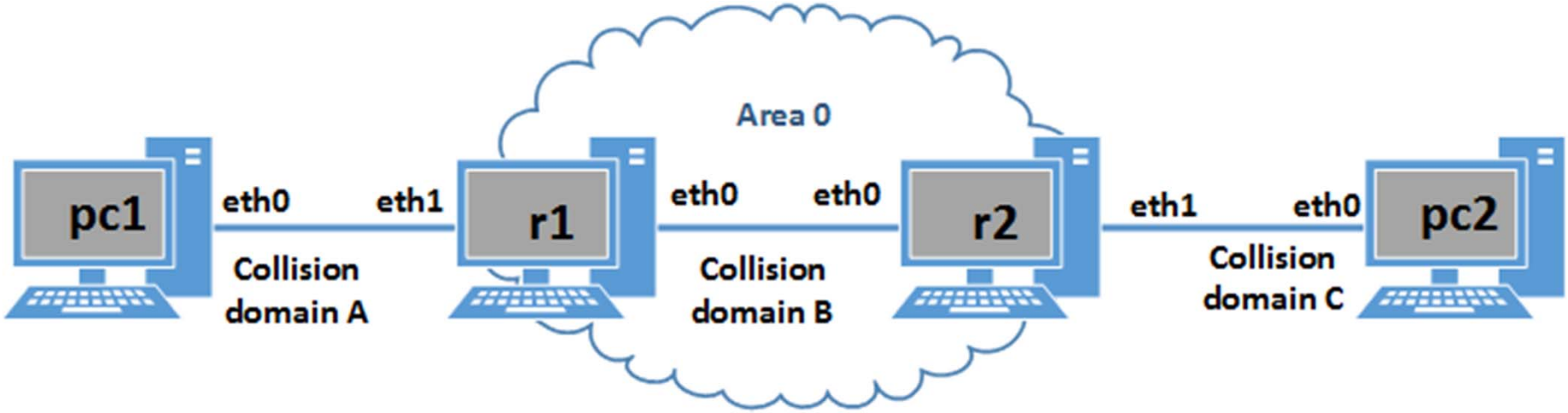
copyright notice

- All the pages/slides in this presentation, including but not limited to, images, photos, animations, videos, sounds, music, and text (hereby referred to as “material”) are protected by copyright.
- This material, with the exception of some multimedia elements licensed by other organizations, is property of the authors and/or organizations appearing in the first slide.
- This material, or its parts, can be reproduced and used for didactical purposes within universities and schools, provided that this happens for non-profit purposes.
- Information contained in this material cannot be used within network design projects or other products of any kind.
- Any other use is prohibited, unless explicitly authorized by the authors on the basis of an explicit agreement.
- The authors assume no responsibility about this material and provide this material “as is”, with no implicit or explicit warranty about the correctness and completeness of its contents, which may be subject to changes.
- This copyright notice must always be redistributed together with the material, or its portions.

simple OSPF lab
(DR election)

assign IP addresses

- Assign IP addresses to ALL interfaces on routers and PCs



Please note that the address of the subnet in Area 0 should include the LAB-ID assigned to you by the lab instructor, e.g., in the following way: `<LAB-ID>.0.0.0/24`

update `lab.conf`

- inside the lab directory update the **`lab.conf`** file and put entries for **`pc1`**, **`pc2`**, **`r1`**, and **`r2`**

- exemplary entry

```
pc1[0]="A"
```

create `*.startup` files

- Create **`startup`** files for all VMs
  - in your lab directory you should have:
    `r1.startup`, **`r2.startup`**,
    `pc1.startup`, **`pc2.startup`**

- Configure the previously chosen IP addresses for each VM

- Exemplary entry

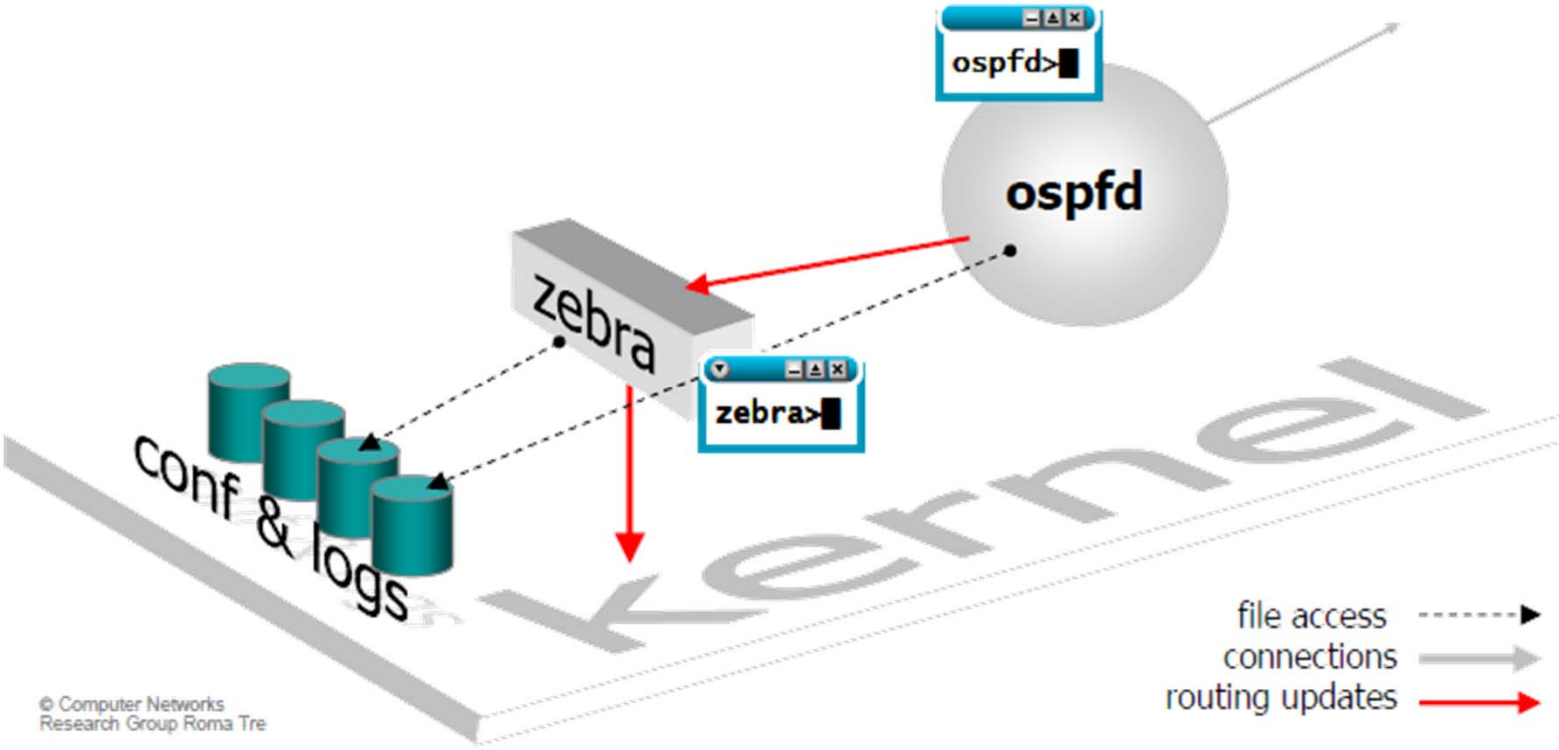
```
ifconfig eth0 10.10.10.1 up
```

create subdirectories for **`r1`**, **`r2`**, **`pc1`**, and **`pc2`**

- duplicate the **`router`** and **`pc`** subdirectories in your lab directory
  - as a result you should have two **`pc`** subdirectories and two **`router`** subdirectories
- change the names of your subdirectories to **`r1`**, **`r2`**, **`pc1`**, and **`pc2`**

modify **daemons** files in **r1** and **r2** subdirectories

- enable **zebra** and **ospfd** routing daemons for **r1** and **r2** by editing their **daemons files**



start the lab

- start the configured laboratory using the **`lstart`** command in the lab directory
- if you did not configure **`zebra`** to be started automatically do it now
  - on **`r1`** and **`r2`** issue the following command

```
/etc/init.d/zebra start
```

OSPF configuration
dummy0 interface

- on each virtual router, connect to the **zebra** daemon and configure the special loopback address called **dummy0**

- **Hints**
  - **telnet** to **zebra** daemon, **configure terminal**, **interface dummy0**
  - enter the **no shutdown** command
  - enter a special address which is different than the IP adderess configured on the **eth** interface
  - write changes to the **zebra** configuration file

Question 1: Why should we enable a special loopback address for OSPF?

OSPF configuration

OSPF intervals

- on each virtual router, connect to **`ospfd`** and change the **`hello-interval`** to 5 and the **`dead-interval`** to 20 on the ***`eth0`*** interface

Question 2: What are these intervals used for? Why are we changing them to lower values?

OSPF configuration

- on each virtual **router**, while connected to **`ospfd`**
  - create an OSPF routing process (**`router ospf`** command)
    - configure the **`ospf router-id`**
      - set it to the value assigned to ***`eth0`***
    - define the IP addresses on which OSPF runs
      - **`network`** command

      HINT: don't forget to define the **`area ID`**
  - write changes to the **`ospfd`** configuration file using the **`write file`** command

OSPF routing information

- on each virtual router, check the running system information using different **`show ip ospf`** commands

Question 3: Which router was chosen as the Designated Router? Why? What is the status of the other router?

Reporting

- Please deliver the following items to the UPEL system using your account
  1. A photocopy or a screenshot showing the output of the following command
     - **`show ip ospf`** executed on router r2

remote destination availability

- try to **`ping pc1`** from **`pc2`** and vice versa
- if the devices cannot reach each other – fix the problem

Question 4: How did you fix the problem?

Reporting

- Please deliver the following items to the UPEL system using your account
  1. A photocopy or a screenshot showing the Wireshark output of the following command
     - **`ping`** of pc1 executed on pc2
     - **`ping`** of pc2 executed on pc1

OSPF: choosing the DR

- on the router that was chosen as DR change its **`router-id`** to the one assigned to the **`dummy0`** interface

Question 5: Which router was chosen as the Designated Router? Can you explain why? What is the status of the other router?

OSPF: choosing the DR cont'd

- on the router which is currently the DR change the **`priority`** to zero
  - **Hint**: go to the interface configuration

Question 6: What is the state of that router now? Can you explain why?

OSPF: choosing the DR cont'd

- change the priority values on both routers so that they are different than zero. Choose them so that the router with the lower **`router-id`** has the higher **`priority`** value
- **Hint:** to see the result on **`r1`** and **`r2`** go to the **`ospf`** configuration and issue the commands
  - **`no network`** `<IP address>` **`area 0`**
  - **`network`** `<IP address>` **`area 0`**

Question 7: Why do we need to disable and then enable ospf? What is more important when choosing the DR – **`router-id`** or **`priority`**?

Playing with network type

- Change the network type of one link to point-to-point
  Question 8: Is a DR selected? Can you explain why?

- Change the network type back to broadcast. Add a third virtual router to the environment, also connected to the same broadcast domain. Enable OSPF. Observe how the DR election algorithm works for three routers.
  Question 9: Check the **`neighbour count`** and the **`adjacent neighbour count`** on each router. Explain these numbers.

DR selection

- Create a new scenario with three routers connected in series.

Question 10: Can a router be a DR on one interface and not a DR on another one?

sniffing the OSPF protocol
on BMA networks

- use **`tcpdump`** to sniff on some interface
- capture the conversation between routers
- move the file from **`VM`** to **host machine**
- browse the file using **`wireshark`** on the host machine in the graphical mode

```
tcpdump -i eth0 proto ospf –w logfile.cap
tcpdump -r logfile.cap proto ospf
tcpdump -v -r logfile.cap proto ospf
```

Sniffing the OSPF protocol **on BMA networks** cont'd

Question 11: What addresses are used during the message exchange?

Question 12. What OSPF message types are used during the conversation?

Question 13. What information is conveyed inside **OSPF LS Update** packets?

Question 14. How often are packets exchanged? What events can trigger an **Update**?