ACCESUL PE ROUTER

1. **Acces prin consola (router-ul are configuratie factory-default)**

Folosim unul din emulatoarele de terminal:

Windows: PuTTY (ver > 0.60 beta), Terra Term [Pro], HyperTerminal, SecureCRT (comercial)
Linux: minicom

Configuratie (default):
Speed: 9600 bps
Data bits: 8
Stop bits: 1
Parity bits: none
Flow control: none

Nota 1: pentru HyperTerminal se poate da click pe butonul "Restore Defaults"
Nota 2: pentru PuTTY configuratia de mai sus este cea default pentru comunicare seriala
Nota 3: pentru emulatoarele USB-COM se va afla din 'Device Manager' portul serial de comunicatii

2. **Acces prin telnet (exemplu propice unui laborator/situatie de testare, NU in productie)**

```
Router> enable
Router# configure terminal
Router (config)# line vty 0 4                ← putem configura mai multe linii vty decat primele 5
Router (config-line)# no login               ← nu se vor cere credentiale la logare
Router (config-line)# privilege level 15     ← ne vom loga direct in PRIVILEGE EXEC Mode
```

3. **Acces pe router prin HTTP(S)**

```
Router> enable
Router# configure terminal
Router (config)# ip http server                  ← porneste serverul de HTTP (TCP/80)
Router (config)# ip http secure-server           ← porneste serverul de HTTPS (TCP/443)
Router (config)# ip http authentication local    ← autentificarea foloseste 'baza de date' locala
Router (config)# username xyz privilege 15 secret abc
```

Nota1: unele IOS-uri au configurat serverul de HTTP(S) by-default. Se poate verifica cu **Router# sh tcp brief all** sau cu **Router# show ip http server** si **Router# show ip http secure-server**

4. **Acces pe router prin SSH**
   a. **Varianta 1 (fara configurare *hostname* si *domain* DNS)**

```
Router> enable
Router# configure terminal
Router (config)# crypto key generate rsa usage-keys label my_SSH_key modulus 1024
```

Router (config)# ip ssh rsa keypair-name *my_SSH_key*
Router (config)# ip ssh version 2 ← se configureaza versiunea 2 (def: compatibility mode)
Router (config)# ip ssh authentication-retries 2 ← numarul de autentificari esuate (def: 3)
Router (config)# ip ssh time-out 10 ← secunde de asteptare a raspunsului clientului (def: 120)
Router (config)# username *xzy* privilege 15 secret *abc*
Router (config)# line vty 0 4
Router (config)# transport input ssh ← se permite accesul doar prin SSH
Router (config)# login local ← autentificarea utilizatorului se face cu user & pass

b. **Varianta 2 (cu configurare *hostname* si domain *DNS*)**

Router> enable
Router# configure terminal
Router (config)# hostname R1
R1 (config)# ip domain-name infoacademy.net
R1 (config)# crypto key generate rsa
R1 (config)# ip ssh version 2
R1 (config)# ip ssh authentication-retries 2
R1 (config)# ip ssh time-out 10
R1 (config)# username *xzy* privilege 15 secret *abc*
R1 (config)# line vty 0 4
R1 (config)# transport input ssh
R1 (config)# login local

Nota 1: client SSH in Windows: PuTTY, SecureCRT, OpenSSH (functionaliate oferita tip client-server)
Nota 2: numarul de biti pentru crearea cheilor asimetrice este, ideal, mai mare de 1024 (default 512)
Nota 3: **Router# sh line vty <nr>** arata ce protocoale sunt permise pentru managementul router-ului
Nota 4: **Router# sh crypto key mypubkey rsa** reda propriile chei publice RSA
Nota 5: Pentru functionalitatea de server de SSH este nevoie de o imagine software ce contine **k9** (3DES) si o versiune de IOS > 12.3(4)T
Nota 6: Router-ul poate fi client SSH: IOS > 12.3(7)T; atat versiunea SSH 1 cat si 2; imagini cu **k8** sau **k9**
Nota 7: *login banner* este suportat in vers 2 SSH, nu in versiunea 1
Nota 8: Pentru a sterge o pereche de chei RSA: **Router(config)# crypto key zeroize rsa [<nume>]** – automat se va opri serverul SSH daca stergem perechea de chei utilizata (numele se incheie cu *.server*)

Comenzi suplimentare SSH:
Router# show ip ssh
Router# show ssh
Router# disconnect ssh [<nr conexiune>]
Router# debug ip ssh