

Sisteme de Operare: Tema 3

Termen de predare: Vineri, 17 Mai 2013, ora 23:00

Descriere

Tema constă în realizarea unui set de task-uri asupra unui sistem de operare Linux, gata instalat într-o mașină virtuală.

Pentru predarea temei sunt necesare:

- conexiunea la internet

Termen de predare:

- Timpul de lucru efectiv este de 10 de zile
- Uploadarea codului: 17 mai 2013, ora 23:00
- Uploadarea se face pe formularul de la adresa următoare:
https://docs.google.com/forms/d/1_kU1bq1CiDv8mzSYNrVNT6FauTKNkQwW12vBVmZJmy8/viewform

Deadline-ul temei este hard. **NU** se acceptă întârzieri după 17 mai, ora 23:55.

Atenție! Important!

Pentru a putea rula `uso submit` cu succes `uso check` **NU** trebuie să se blocheze. **NU** oferim suport pentru eșecul de a trimite tema dacă `uso check` se blochează. Verificați comenzile și scripturile folosite pentru a nu bloca verificarea!

Atenție!

Chiar și după ce ați predat codul, **NU** ștergeți arhiva cu mașina virtuală până nu vă aflați nota! Dacă vor exista probleme în corectarea temei, fără arhivă **NU** se pot face contestații sau rezolva eventualele probleme apărute.

Testare

Testarea temei se va face **DOAR** autentificat ca utilizator `student`, folosind programul instalat `uso` (ca `root` prin intermediul `sudo`). Programul `uso` implementează testele pentru toate task-urile din această temă. Acesta poate fi executat în orice moment pentru a verifica dacă au fost rezolvate corect task-urile.

Versiunea programului `check` este afișată pe prima linie a output-ului.

Versiunea curentă este: **v7**. Codename: **Summer is coming**.

În cazul în care testele vor fi actualizate, modificările ajung în mod automat pe mașinile virtuale. Comanda `uso update` va descărca ultima versiune a programului `uso` folosit pentru testarea temei. Pentru a actualiza manual checker-ul, puteți apela `uso update`.

Testarea finală a temei se realizează prin intermediul utilitarului automat de verificare a temelor. Tot ce trebuie voi să faceți este să rulați `uso submit` și să încărcați codul în formularul descris în pagina anterioară.

Pentru a ușura testarea, recomandăm:

- Setările mașinii virtuale **NU** ar trebui modificate (nici măcar o actualizare a VMWare Tools). Dacă totuși considerați că este necesară o modificare întrebați întâi pe forum-ul temei dacă este acceptată.
- Mașina virtuală să pornească iar sistemul de operare să se încarce complet **fără interacțiune din exterior**
- Parola de root să rămână cea implicită ("student")

ATENȚIE: Mașina virtuală vine implicit cu aceste condiții îndeplinite și nici un task nu va afecta vreuna din ele. Vă rugăm să **NU** alterați vreuna din aceste componente deoarece **riscați să primiți 0 puncte pe temă**.

Notare

Nota pe întreaga temă este dată de punctajul acumulat pe toate task-urile, punctaj afișat de către programul `check` la execuția fără nici un parametru.

Se pot lua maxim **100** de puncte pe întreaga temă.

Nu este obligatorie rezolvarea tuturor task-urilor. Task-urile pot fi rezolvate în orice ordine, mai puțin în situația în care un task depinde de rezolvarea unui alt task.

Task 1 (20p)

Configurări de rețea

Configurează conexiunea la rețea folosind setări **permanente** aplicate pe **eth1** astfel:

- Adresa IP să fie 10.x.y.200 (unde x = \$n_nume și y = \$n_prenume)
- Masca de rețea să fie 255.255.255.128 (/25)
- Gateway-ul să aibă alocată prima adresă utilizabilă din rețeaua respectivă
- Adresa MAC să fie AA:BB:CC:DD:x:y, setată **înainte** de pornirea interfeței

– **Hint:** folosiți directiva 'pre-up' în fișierul **interfaces**

– **Hint:** cifrele adresei MAC sunt cifre în sistem de numerație **hexazecimal**

- folosind un "one-liner" adăugat în fișierul de mai sus, crează un fișier de jurnal în `/var/log/retea.log`, care, la deschiderea sau închiderea interfeței, să scrie (prin adăugare la fișier) evenimentul în format "UP/DOWN [ZZ-LL-AA/OO:MM:SS]". Exemplu:

– UP [01-12-12/12:00:00]

– DOWN [10-12-12/18:58:34]

– **Hint:** folosiți directivele **up** și **down**

Task 2 (20p)

Configurări de rețea II

Obs: Acest task depinde de rezolvarea task-ului anterior.

Configurează *eth1* ca sistem de failsafe (<http://en.wikipedia.org/wiki/Fail-safe>) în caz că *eth0* cade. Considerăm momentul în care *eth0* devine **down** sau **up** simulat prin intermediul comenzilor `ifdown eth0` respectiv `ifup eth0`.

Interfața *eth0* va fi configurată permanent, dinamic și să pornească o dată cu sistemul (este deja implicit).

NU este necesară nici o configurație permanentă pentru *eth1*. Toate configurațiile (inclusiv pt. *eth1*) vor fi efectuate în secțiunea interfeței *eth0*.

Va trebui ca atunci când *eth0* devine down, *eth1* să devină up, iar când *eth0* revine up, *eth1* să devină automat down.

Asigură-te că la rularea checker-ului *eth0* este **up** iar *eth1* este **down**.

- Pentru mai multe detalii consultă pagina de manual *man interfaces*, directivele **up** și **down**.
- Acest task are mai multe rezolvări posibile. Se consideră validă orice rezolvare *persistentă*, adică o rezolvare ce este validată de către checker inclusiv imediat după repornirea sistemului. .

Task 3 (20p)

DNS

Pentru a evita căutările DNS spre domeniile des folosite, asociați local

- google.com cu 74.125.79.105
- cs.curs.pub.ro cu 141.85.241.51

Task 4 (20p)

Restricționare site-uri

Instalează pachetul `elinks`. `elinks` este un browser în mod text ce poate fi folosit din linia de comandă.

Ai observat că petreci foarte mult timp citind știri pe `yahoo.com`. Realizează configurările necesare astfel încât să blochezi accesul la acest site (inclusiv pentru **toate** subdomeniile sale) din browser-ul `elinks`. După realizarea configurărilor, nu va trebui să poți accesa nici `ro.yahoo.com` sau `fr.yahoo.com`, de exemplu.

Restricționarea accesului se referă doar la site-ul `yahoo.com` și la traficul HTTP. De exemplu, trebuie să poți accesa orice alt site în afara celui menționat. În plus, trebuie să poți primi în continuare pachete de tip ICMP folosind `ping`, inclusiv pentru `yahoo.com`. Pentru a primi punctaj pe acest task, va trebui să ai conexiune la internet.

Pentru rezolvarea task-ului este indicat să citiți următoarele articole: [Proxy Server](#), [Web Access with Squid](#) și [Elinks using Proxy](#). Se recomandă folosirea directivei `dstdom_regex` în fișierul de configurare pentru `squid`. **Atenție:** pentru ca setările să meargă pentru orice utilizator, editați fișierul global de configurare `elinks!`

Task 5 (20p)

Configurare apache2

Prima dată instalează server-ul web `apache2`. Configurează server-ul să asculte numai pe portul 8080 și document root să fie `/home/student/public_html`.

Pentru amuzament, poți vedea efectele schimbărilor, adăugând în noul document root un fișier `index.html`, cu textul "Hello, World!" pe care să îl accesezi folosind URL: <http://localhost:8080/index.html>, din mașina virtuală.

- Fișierele pe care trebuie să le modifizi sunt `ports.conf` și `sites-available/default` din `/etc/apache2/`.
- Folosește comanda `service apache2 reload` pentru a cere server-ului `apache2` să citească noile modificări.
- Asigură-te că există directorul `/home/student/public_html`.

- Nu testa corectitudinea exercițiului folosind `elinks`, deoarece l-ai configurat să folosească ca server proxy `squid`, iar acesta face caching la pagini.
- Hint: <http://tareq.wedevs.com/2012/04/changing-apache-document-root-in-ubuntu/>

Lista schimbărilor