# Scale for project [roger-skyline-1](https://projects.intra.42.fr/projects/roger-skyline-1)

# You should correct 1 student in this team

#### **Git repository**

### **Introduction**

Nous vous demandons pour le bon déroulement de cette évaluation de

respecter les règles suivantes :

- Restez courtois, polis, respectueux et constructifs en toutes

situations lors de cet échange. Le lien de confiance entre la

communauté 42 et vous en dépend.

- Mettez en évidence auprès de la personne (ou du groupe) notée les

dysfonctionnements éventuels du travail rendu, et prenez le temps

d'en discuter et d'en débattre.

- Acceptez qu'il puisse y avoir parfois des différences

d’interprétation sur les demandes du sujet ou l’étendue des

fonctionnalités. Restez ouvert d'esprit face à la vision de

l'autre (a-t-il ou elle raison ou tort ?), et notez le plus

honnêtement possible. La pédagogie de 42 n'a de sens que si

la peer-évaluation est faite sérieusement.

### **Guidelines**

- Vous ne devez évaluer que ce qui se trouve sur le depôt GiT de rendu

de l'étudiant(e) ou du groupe.

- Prenez soin de vérifier que le depôt GiT est bien celui

correspondant a l'étudiant(e) ou au groupe, et au projet.

- Verifiez méticuleusement qu'aucun alias malicieux n'a été utilisé

pour vous induire en erreur et vous faire évaluer autre chose que

le contenu du dépot officiel.

- Tout script sensé faciliter l'évaluation fourni par l'un des deux

partis doit être rigoureusement vérifié par l'autre parti pour

éviter des mauvaises surprises.

- Si l'étudiant(e) correcteur/correctrice n'a pas encore fait ce

projet, il est obligatoire pour cet(te) etudiant(e) de lire le

sujet en entier avant de commencer cette soutenance.

- Utilisez les flags disponibles sur ce barème pour signaler un

rendu vide, non fonctionnel, une faute de norme, un cas de triche,

etc. Dans ce cas, l'évaluation est terminée et la note finale est

0 (ou -42 dans le cas special de la triche). Toutefois, hors cas

de triche, vous etes encouragés a continuer d\'échanger autour du

travail éffectué (ou non éffectué justement) pour identifier les

problemes ayant entrainé cette situation et les éviter pour le

prochain rendu.

## **Attachments**

#### [**Sujet**](https://cdn.intra.42.fr/pdf/pdf/750/roger-skyline-1.fr.pdf)

## **Sections**

### **Suivre Slash16 Partout Dans le Monde**

Il faut nous suivre dans le monde entier !

#### **L etudiant a suivi Slash16 sur Linkedin, Facebook et Twitter**

L etudiant a suivi Slash16 sur Linkedin, Facebook et Twitter

### **Partie 2 - Network**

Evaluation de la partie 2 du sujet - Network

#### **Récupérez la liste des interfaces réseau de la machine sans afficher aucun détail**

Verifiez que le fichier de reponse contient la commande qui

liste les noms des interfaces presentes sur la machine et aucune

autre information. Par exemple :

$>`cat 01`

lo0 gif0 stf0 en0 en1 en2 en3 p2p0 awdl0 bridge0

$>

#### **Identifiez l’adresse IP de l’interface Ethernet**

Verifiez que le fichier de reponse contient la commande qui

identifie et affiche les caracteristiques de l interface

ethernet. Par exemple :

$>`cat 02`

en0: flags=XXXX

mtu 1500

options=10b

ether 00:00:00:00:00:00

inet 42.42.42.42 netmask 0xffffff00 broadcast 42.42.42.255

nd6 options=1

media: autoselect (1000baseT )

status: active

$>

#### **Identifiez l’adresse MAC de la carte Wi-Fi**

Verifiez que le fichier de reponse contient la commande qui

identifie et affiche l'adresse mac de l'interface wifi. Par

exemple :

$>`cat 03`

xxn: flags=XXXX

ether 00:00:00:00:00:00

$>

#### **Identifiez la gateway par défaut dans la table de routage**

Verifiez que le fichier de reponse contient la commande qui

identifie et affiche la gateway par defaut dans la table de

routage. Par exemple :

$>sh 04

default 42.42.42.42 UGSc 19 16 en0

$>

#### **Identifiez l’IP du serveur DNS qui répond sur le domaine suivant - slash16.org**

Verifiez que le fichier de reponse contient la commande qui

identifie et affiche l IP du serveur DNS. Par exemple :

$>`cat 05`

Server:10.51.1.42

Address:10.51.1.42

Non-authoritative answer:

Name:slash16.org

Address: 195.154.52.157

Name:slash16.org

Address: 195.154.52.158

$>

#### **Récupérez le path complet du fichier dans lequel est écrit l’adresse IP du serveur DNS que vous utilisez**

Verifiez que le fichier de reponse contient le path du fichier

des serveurs DNS utilises.

#### **Interrogez un serveur DNS externe sur le même nom de domaine (ex, google 8.8.8.8)**

Verifiez que le fichier de reponse contient la commande qui

utilise un autre serveur DNS pour resoudre le meme nom de

domaine. Par exemple :

$>`cat 07`

Server:8.8.8.8

Address:8.8.8.8

Non-authoritative answer:

Name:slash16.org

Address: 195.154.52.157

Name:slash16.org

Address: 195.154.52.158

$>

#### **Les IPs retournées sont-elles similaires ?**

Verifiez que la reponse de l etudiant dans le fichier est affirmative.

#### **Faites la même chose avec 42.fr, que remarquez-vous ?**

Verifiez que la reponse de l etudiant dans le fichier est que

les IPs sont différentes.

#### **Identifiez les différents appareils réseaux entre votre poste et le domaine slash16.org**

Verifiez que le fichier de reponse contient la commande qui

identifie et affiche les differents serveurs entre votre poste et slash16.org.

Par exemple :

$>`cat 10`

traceroute to slash16.org (195.154.52.158), 64 hops max, 52 byte packets

1 10.8.0.1 (10.8.0.1) 5.809 ms 6.087 ms 3.124 ms

2 10.42.1.254 (10.42.1.254) 6.005 ms 13.668 ms 7.037 ms

3 nat-1.42.fr (10.60.1.11) 7.530 ms 3.379 ms 9.966 ms

4 dc3.42.fr (62.210.35.1) 7.100 ms 7.587 ms 5.160 ms

5 195.154.1.174 (195.154.1.174) 57.350 ms 168.093 ms 8.906 ms

6 a9k2-45x-s44-2.dc3.poneytelecom.eu (195.154.1.106) 6.590 ms 3.910 ms 5.525 ms

7 195.154.1.179 (195.154.1.179) 4.077 ms 46.904 ms 3.883 ms

8 pub-1.slash16.org (195.154.52.158) 5.699 ms 6.034 ms 7.632 ms

$>

#### **Trouvez grâce au résultat de la commande précédente le nom et l’IP du matériel qui fait le lien entre vous et l’extérieur**

Verifiez que la reponse de l etudiant dans le fichier

est bien le serveur de NAT.

#### **Vérifiez que le serveur avec l’adresse IP 10.51.1.253 est touchable depuis votre poste**

Verifiez que le fichier de reponse contient la commande qui

affiche si l'IP est touchable ou non. Par exemple:

$>`cat 12`

64 bytes from 10.51.1.253: icmp\_seq=0 ttl=62 time=40.663 ms

64 bytes from 10.51.1.253: icmp\_seq=1 ttl=62 time=5.024 ms

$>

#### **Déterminez le type du serveur**

Verifiez que la reponse de l etudiant dans le fichier

est bien DHCP.

#### **Vérifiez grâce au Reverse DNS le nom du serveur correspondant à l’IP 10.51.1.81**

Verifiez que la reponse de l etudiant dans le fichier

est bien ldap-proxy.42.fr

#### **Quel est le fichier contenant les entrées locales DNS ?**

Verifiez que la reponse de l etudiant dans le fichier

est bien /etc/hosts.

#### **Faites pointer l’adresse suivante 62.210.207.183 sur l’adresse intra.42.fr**

Verifiez que la reponse de l etudiant dans le fichier

est bien '62.210.207.183 intra.42.fr'.

### **Partie 3 - System**

Evaluation de la partie 3 - System

#### **Dans quel fichier se trouve la version installée de votre Debian ?**

Verifiez que la reponse de l'etudiant dans le fichier

est bien /etc/debian\_version.

#### **Quelle commande permet de renommer votre système ?**

Verifiez que le fichier de reponse contient la commande qui

renomme le systeme. Par exemple:

$>`cat 02`

machine.old.name.local

$>

#### **Quel est le fichier à modifier pour rendre cela permanent ?**

Verifiez que la reponse de l etudiant dans le fichier

est bien /etc/hostname.

#### **Quelle commande donne le temps depuis laquelle votre système est allumé ?**

Verifiez que le fichier de reponse contient la commande qui

donne le temps depuis laquelle votre système est allume. Par exemple:

$>`cat 04`

17:44 up 1 day, 6:45, 4 users, load averages: 1.33 1.42 1.40

$>

#### **Déterminez l’état du service SSH**

Verifiez que le fichier de reponse contient la commande qui

determine l etat du service SSH. Par exemple :

$>`cat 05`

openssh-daemon (pid 22405) is running...

$>

#### **Donnez la commande pour le redémarrer**

Verifiez que le fichier de reponse contient la commande qui

redemarre le service SSH. Par exemple :

$>`cat 06`

Stopping sshd: [ OK ]

Starting sshd: [ OK ]

$>

#### **Déterminez le PID du service SSHD**

Verifiez que le fichier de reponse contient la commande qui

determine le PID du service SSH. Par exemple :

$>`cat 07`

root 22405 0.0 0.0 66224 1184 ? Ss 17:46 0:00 /usr/sbin/sshd

$>

#### **Quel fichier contient les clés RSA autorisées à se connecter via SSH ?**

Verifiez que la reponse de l etudiant dans le fichier

est bien .ssh/authorized\_keys

#### **Quelle commande permet de savoir quelles personnes sont connectées sur le système ?**

Verifiez que le fichier de reponse contient la commande qui

permet de savoir qui est sur le systeme. Par exemple :

$>`cat 09`

skyline console Mar 23 10:59

skyline ttys000 Mar 24 17:04

$>

#### **Quelle commande permet de lister les tables de partitions des périphériques ?**

Verifiez que le fichier de reponse contient la commande qui

permet de lister les tables de partitions des peripheriques. Par exemple :

$>`cat 10`

Disk /dev/sdb: 2000.4 GB, 2000398934016 bytes

255 heads, 63 sectors/track, 243201 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk identifier: 0x00000000

$>

#### **Quelle commande permet d’afficher l’espace disponible par le système ?**

Verifiez que le fichier de reponse contient la commande qui

permet d afficher l espace disponible. Par exemple :

$>`cat 11`

Filesystem Size Used Avail Use% Mounted on

/dev/xvda2 7.8G 1.2G 6.3G 16% /

udev 10M 0 10M 0% /dev

tmpfs 200M 4.2M 196M 3% /run

tmpfs 500M 0 500M 0% /dev/shm

tmpfs 5.0M 0 5.0M 0% /run/lock

tmpfs 500M 0 500M 0% /sys/fs/cgroup

$>

#### **Déterminez la taille exacte de chaque dossier de /var**

Verifiez que le fichier de reponse contient la commande qui

permet d afficher la taille exacte de chaque dossier de /var.

Par exemple:

$>`cat 12`

4.0K /var/opt

864K /var/spool

1.3M /var/log

111M /var/lib

124M /var/cache

4.0K /var/local

8.0K /var/mail

1.1M /var/backups

4.0K /var/tmp

238M /var

$>

#### **Trouvez la commande qui permet de trouver les processus en cours d’exécution**

Verifiez que le fichier de reponse contient la commande qui

permet d afficher les processus en cours d execution en temps reel.

Par exemple:

$>`cat 13`

Tasks: 58 total, 1 running, 57 sleeping, 0 stopped, 0 zombie

%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st

KiB Mem: 1022952 total, 359176 used, 663776 free, 168200 buffers

KiB Swap: 0 total, 0 used, 0 free. 86924 cached Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND

1 root 20 0 28740 4760 3064 S 0.0 0.5 0:03.28 systemd

2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd

3 root 20 0 0 0 0 S 0.0 0.0 0:00.02 ksoftirqd/0

5 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0H

$>

#### **Lancez en background la commande tail -f /var/log/syslog**

Verifiez que le fichier de reponse contient la commande qui

permet de lancer en background la commande tail -f /var/log/syslog

Par exemple:

$>`cat 14`

[1] 2660

Mar 25 06:25:03 rsyslogd: [origin software="rsyslogd" swVersion="8.4.2" x-pid="330" x-info="http://www.rsyslog.com"] rsyslogd was HUPed

Mar 25 07:17:01 CRON[2601]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)

Mar 25 08:17:01 CRON[2656]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)

$>

#### **Trouvez la commande qui permet de tuer le processus de la commande en background**

Verifiez que le fichier de reponse contient la commande qui

permet de tuer le processus du tail -f /var/log/syslog.

Par exemple:

$>`cat 15`

[1]+ Terminated tail -f /var/log/syslog

$>

#### **Trouvez le service qui permet de lancer des tâches à horaires régulières**

Verifiez que la reponse de l etudiant dans le fichier

est bien cron.

#### **Trouvez la commande qui donne la liste des règles de pare-feu**

Verifiez que le fichier de reponse contient la commande qui

permet de lister les regles du pare-feu.

Par exemple:

$>`cat 17`

Chain INPUT (policy ACCEPT)

target prot opt source destination

Chain FORWARD (policy ACCEPT)

target prot opt source destination

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

$>

#### **Avec la commande précédente, autorisez uniquement les IPs provenant de 10.0.0.0/8 à se connecter sur votre système**

Verifiez que le fichier de reponse contient la commande qui

permet de d autoriser uniquement les IPs provenant de 10.0.0.0/8

à se connecter sur votre système.

Par exemple:

$>`cat 18`

$>

$> iptables -L

Chain INPUT (policy ACCEPT)

target prot opt source destination

ACCEPT all -- 10.0.0.0/8 anywhere

Chain FORWARD (policy ACCEPT)

target prot opt source destination

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

$>

#### **Avec la commande précédente, interdisez le reste**

Verifiez que le fichier de reponse contient la commande qui

permet de d interdire le reste à se connecter sur votre système.

Par exemple:

$>`cat 19`

$>

$> iptables -L

Chain INPUT (policy DROP)

target prot opt source destination

ACCEPT all -- 10.0.0.0/8 anywhere

Chain FORWARD (policy ACCEPT)

target prot opt source destination

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

$>

### **Partie 3 - Scripting**

Evalution de la partie 3 du sujet - Scripting

#### **Réaliser un script qui affiche seulement le login, le UID et le Path du fichier /etc/passwd**

Verifiez que le script petmet d afficher seulement le login, le UID

et le Path du fichier /etc/passwd. Par exemple:

$>sh 1

root:0:/bin/bash

daemon:1:/usr/sbin/nologin

bin:2:/usr/sbin/nologin

sys:3:/usr/sbin/nologin

sync:4:/bin/sync

games:5:/usr/sbin/nologin

man:6:/usr/sbin/nologin

lp:7:/usr/sbin/nologin

mail:8:/usr/sbin/nologin

news:9:/usr/sbin/nologin

uucp:10:/usr/sbin/nologin

proxy:13:/usr/sbin/nologin

www-data:33:/usr/sbin/nologin

backup:34:/usr/sbin/nologin

list:38:/usr/sbin/nologin

irc:39:/usr/sbin/nologin

gnats:41:/usr/sbin/nologin

nobody:65534:/usr/sbin/nologin

systemd-timesync:100:/bin/false

systemd-network:101:/bin/false

systemd-resolve:102:/bin/false

systemd-bus-proxy:103:/bin/false

sshd:104:/usr/sbin/nologin

Debian-exim:105:/bin/false

postfix:106:/bin/false

skyline:1000:/bin/bash

$>

#### **Réaliser un script qui met à jour l’ensemble des sources de package, puis de vos pa- ckages et qui log l’ensemble dans un fichier nommé /var/log/update\_script.log. Créez une tache plannifiée pour ce script une fois par semaine à 4h00 du matin.**

Verifiez que le script permet de mettre à jour l’ensemble des sources de package, des packages,

qui log dans le bon fichier et que celui-ci est en cron.

Par exemple:

$>sh script/2

$>cat /var/log/update\_script.log

Hit http://cloudfront.debian.net jessie-backports InRelease

Hit http://security.debian.org jessie/updates InRelease

Hit http://security.debian.org jessie/updates/main Sources

Hit http://security.debian.org jessie/updates/contrib Sources

Hit http://security.debian.org jessie/updates/non-free Sources

Hit http://security.debian.org jessie/updates/main amd64 Packages

Hit http://security.debian.org jessie/updates/contrib amd64 Packages

Hit http://security.debian.org jessie/updates/non-free amd64 Packages

Hit http://security.debian.org jessie/updates/contrib Translation-en

Ign http://httpredir.debian.org jessie InRelease

Hit http://security.debian.org jessie/updates/main Translation-en

Get:1 http://httpredir.debian.org jessie-updates InRelease [142 kB]

Hit http://security.debian.org jessie/updates/non-free Translation-en

Get:2 http://cloudfront.debian.net jessie-backports/main Sources/DiffIndex [27.8 kB]

Get:3 http://httpredir.debian.org jessie Release.gpg [2,373 B]

Get:4 http://cloudfront.debian.net jessie-backports/main amd64 Packages/DiffIndex [27.8 kB]

Get:5 http://cloudfront.debian.net jessie-backports/main Translation-en/DiffIndex [27.8 kB]

Hit http://httpredir.debian.org jessie-updates/main Sources

Hit http://httpredir.debian.org jessie-updates/contrib Sources

Hit http://httpredir.debian.org jessie-updates/non-free Sources

Get:6 http://httpredir.debian.org jessie-updates/main amd64 Packages/DiffIndex [1,012 B]

Hit http://httpredir.debian.org jessie-updates/contrib amd64 Packages

Get:7 http://httpredir.debian.org jessie-updates/non-free amd64 Packages/DiffIndex [736 B]

Get:8 http://httpredir.debian.org jessie-updates/contrib Translation-en [14 B]

Get:9 http://httpredir.debian.org jessie-updates/main Translation-en/DiffIndex [736 B]

Get:10 http://httpredir.debian.org jessie-updates/non-free Translation-en/DiffIndex [736 B]

Get:11 http://httpredir.debian.org jessie Release [148 kB]

Hit http://httpredir.debian.org jessie/main Sources

Hit http://httpredir.debian.org jessie/contrib Sources

Hit http://httpredir.debian.org jessie/non-free Sources

Hit http://httpredir.debian.org jessie/main amd64 Packages

Hit http://httpredir.debian.org jessie/contrib amd64 Packages

Hit http://httpredir.debian.org jessie/non-free amd64 Packages

Get:12 http://httpredir.debian.org jessie/contrib Translation-en [38.5 kB]

Get:13 http://httpredir.debian.org jessie/main Translation-en [4,582 kB]

Get:14 http://httpredir.debian.org jessie/non-free Translation-en [72.5 kB]

Fetched 5,073 kB in 4s (1,143 kB/s)

Reading package lists...

Reading package lists...

Building dependency tree...

Reading state information...

The following packages will be upgraded:

git git-man

2 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

Need to get 4,530 kB of archives.

After this operation, 30.7 kB of additional disk space will be used.

Get:1 http://security.debian.org/ jessie/updates/main git-man all 1:2.1.4-2.1+deb8u2 [1,267 kB]

Get:2 http://security.debian.org/ jessie/updates/main git amd64 1:2.1.4-2.1+deb8u2 [3,262 kB]

Fetched 4,530 kB in 0s (14.5 MB/s)

(Reading database ... 41091 files and directories currently installed.)

Preparing to unpack .../git-man\_1%3a2.1.4-2.1+deb8u2\_all.deb ...

Unpacking git-man (1:2.1.4-2.1+deb8u2) over (1:2.1.4-2.1+deb8u1) ...

Preparing to unpack .../git\_1%3a2.1.4-2.1+deb8u2\_amd64.deb ...

Unpacking git (1:2.1.4-2.1+deb8u2) over (1:2.1.4-2.1+deb8u1) ...

Processing triggers for man-db (2.7.0.2-5) ...

Setting up git-man (1:2.1.4-2.1+deb8u2) ...

Setting up git (1:2.1.4-2.1+deb8u2) ...

$>

Pour verifier que ce script est en cron :

$>crontab -l

00 4 \* \* 1 /bin/sh /usr/local/bin/01.sh > /var/log/update\_script.log 2>&1

$>

#### **Réaliser un script qui affiche la liste des fichiers triés par taille présents dans le dossier passé en argument**

Verifiez que le script permet d afficher la liste des fichiers

triés par taille présents dans le dossier passé en argument.

Par exemple:

$>sh script/3 /var

248736 /var

131080 /var/cache

125012 /var/cache/apt

113992 /var/lib

97572 /var/lib/apt

97544 /var/lib/apt/lists

75024 /var/cache/apt/archives

33016 /var/cache/apt/archives/linux-image-3.16.0-4-amd64\_3.16.7-ckt20-1+deb8u4\_amd64.deb

33012 /var/lib/apt/lists/httpredir.debian.org\_debian\_dists\_jessie\_main\_binary-amd64\_Packages

31764 /var/lib/apt/lists/httpredir.debian.org\_debian\_dists\_jessie\_main\_source\_Sources

$>

#### **Réaliser un script qui permet de surveiller les modifications du fichier /etc/crontab et envoie un mail à root si celui-ci a été modifié. Créez une tache plannifiée pour script tous les jours à minuit.**

Verifiez que le script permet de surveiller les modifications du fichier /etc/crontab

et envoie un mail à root si celui-ci a été modifié.

Vous devez donc recevoir un mail montrant que le fichier a changer, soit en local avec la commande mail,

soit dans votre propre boite mail.

De plus, il faut verifier la crontab :

$>crontab -l

00 0 \* \* \* /bin/sh /usr/local/bin/04.sh 2>&1

$>

#### **Realiser un script qui affiche 42**

Bah 42 quoi :)