

Utilitaires réseau de base

Vous avez donc finalement réussi à établir votre connexion réseau, maintenant quel est le programme ? Comment vérifier si cela fonctionne ? Comment savoir si vos paramètres sont corrects ? Et que pouvez-vous faire maintenant que l'installation est finie ? Ce chapitre parle de tout cela.

Outils de diagnostic réseau

Slackware Linux propose de très nombreux utilitaires réseaux pour le diagnostic et le dépannage de problèmes de connexion réseau. La plupart de ces outils s'utilisent en ligne de commande, vous pouvez donc les exécuter depuis un terminal virtuel ou dans une fenêtre de terminal depuis votre bureau graphique. Une minorité dispose d'interface graphique, mais nous n'allons nous concentrer que sur les outils en ligne de commande pour le moment.

ping

ping(8) est un outil pratique pour déterminer si un ordinateur est fonctionnel que ce soit sur votre réseau ou sur Internet. Vous pouvez le comparer à un espèce de sonar pour ordinateur. En l'utilisant vous envoyez un "ping" et attendez un écho pour savoir si l'autre ordinateur ou équipement réseau est à l'écoute. Par défaut **ping** recherche l'ordinateur distant une fois par seconde de manière infinie, mais vous pouvez facilement modifier les intervalles entre les tests et le nombre total de tests, consultez la page de manuel. Vous pouvez tuer ce programme à n'importe quel moment avec `CTRL+C`. Lorsque **ping** se termine il affiche un résumé des activités. **ping** est très pratique pour déterminer si un ordinateur sur votre réseau ou sur Internet est disponible mais certains systèmes bloquent les paquets envoyés par **ping** donc parfois un système peut fonctionner normalement mais ne pas envoyer de réponse.

```
darkstar:~# ping -c 3 www.slackware.com
64 bytes from slackware.com (64.57.102.34): icmp_seq=1 ttl=47 time=87.1 ms
64 bytes from slackware.com (64.57.102.34): icmp_seq=2 ttl=47 time=86.2 ms
64 bytes from slackware.com (64.57.102.34): icmp_seq=3 ttl=47 time=86.7 ms

--- slackware.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 86.282/86.718/87.127/0.345 ms
```

traceroute

traceroute(8) est un outil pratique pour déterminer quelle route empruntent vos paquets pour atteindre un autre ordinateur. Il est principalement utilisé pour savoir quels ordinateurs sont "proches" ou "loin" de vous. La distance n'est pas vraiment géographique, du fait que votre Fournisseur d'Accès Internet peut router le trafic de votre ordinateur de manière étrange. **traceroute** vous montre chaque routeur entre votre ordinateur et la machine à laquelle vous souhaitez vous connecter. Malheureusement, de nombreux fournisseurs, pare-feux et routeurs bloqueront **traceroute** vous n'obtiendrez donc pas une vue complète en l'utilisant. Il reste toutefois un outil

pratique pour le dépannage réseau.

```
darkstar:~# traceroute www.slackware.com
traceroute to slackware.com (64.57.102.34), 30 hops max, 46 byte
packets
 1 gw.ctsmacon.com (192.168.1.254)1.468 ms2.045 ms1.387 ms
 2 10.0.0.1 (10.0.0.1)7.642 ms8.019 ms6.006 ms
 3 68.1.8.49 (68.1.8.49)10.446 ms9.739 ms7.003 ms
 4 68.1.8.69 (68.1.8.69)11.564 ms6.235 ms7.971 ms
 5 dalsbbrj01-ae0.r2.dl.cox.net (68.1.0.142)43.859 ms43.287 ms 44.125 ms
 6 dpr1-ge-2-0-0.dallasequinix.savvis.net (204.70.204.146)41.927 ms 58.247
ms44.989 ms
 7 cr2-tengige0-7-5-0.dallas.savvis.net (204.70.196.29)42.577 ms 46.110
ms43.977 ms
 8 cr1-pos-0-3-3-0.losangeles.savvis.net (204.70.194.53)78.070 ms 76.735
ms76.145 ms
 9 bpr1-ge-3-0-0.LosAngeles.savvis.net (204.70.192.222)77.533 ms 108.328
ms120.096 ms
10 wiltel-communications-group-inc.LosAngeles.savvis.net
(208.173.55.186)79.607 ms76.847 ms75.998 ms
11 tg9-4.cr01.lsancarc.integra.net (209.63.113.57)84.789 ms85.436 ms85.575
ms
12 tg13-1.cr01.sntdcabl.integra.net (209.63.113.106)87.608 ms 84.278
ms86.922 ms
13 tg13-4.cr02.sntdcabl.integra.net (209.63.113.134)87.284 ms 85.924
ms86.102 ms
14 tg13-1.cr02.rcrdcauu.integra.net (209.63.114.169)85.578 ms 85.285
ms84.148 ms
15 209.63.99.166 (209.63.99.166)84.515 ms85.424 ms85.956 ms
16 208.186.199.158 (208.186.199.158)86.557 ms85.822 ms86.072 ms
17 sac-main.cwo.com (209.210.78.20)88.105 ms87.467 ms87.526 ms
18 slackware.com (64.57.102.34)85.682 ms86.322 ms85.594 ms
```

telnet

Dans les temps anciens, **telnet**(1) était la meilleure chose au monde après le pain tranché. De manière basique, **telnet** ouvre une connexion réseau non chiffrée entre deux ordinateurs et laisse le contrôle de la session à l'utilisateur plutôt qu'à une autre application. En utilisant **telnet** les gens peuvent se connecter à des shells sur d'autres ordinateurs et exécuter des commandes comme s'ils étaient physiquement connectés. Comme tout circule en clair, il est maintenant déconseillé ; toutefois **telnet** est toujours utilisé de cette manière par de nombreux équipements.

Aujourd'hui **telnet** est plus utile en tant qu'outil de diagnostic réseau. Comme il laisse le contrôle de la session à l'utilisateur il peut être utilisé pour une grande variété de besoins. Dès lors que vous connaissez des commandes ASCII à envoyer à l'ordinateur distant vous pouvez effectuer beaucoup de choses, comme lire des pages web ou vérifier votre courrier électronique. Indiquez simplement à **telnet** quel port utiliser et vous êtes paré.

```
darkstar:~$ telnet www.slackware.com 80
```

```
Trying 64.57.102.34...
Connected to www.slackware.com.
Escape character is '^]'.
HEAD / HTTP/1.1
Host: www.slackware.com

HTTP/1.1 200 OK
Date: Thu, 04 Feb 2010 18:01:35 GMT
Server: Apache/1.3.27 (Unix) PHP/4.3.1
Last-Modified: Fri, 28 Aug 2009 01:30:27 GMT
ETag: "61dc2-5374-4a973333"
Accept-Ranges: bytes
Content-Length: 21364
Content-Type: text/html
```

ssh

Comme nous l'avons mentionné **telnet** peut être un outil de diagnostic utile mais n'utilisant pas de chiffrement cela pose des problèmes en terme de sécurité pour les accès shell. Heureusement, il existe un protocole de shell sécurisé (*secure shell protocol* - SSH). Quasiment toutes les distributions Linux, UNIX et BSD utilisent OpenSSH actuellement, ou **ssh**(1) pour faire simple. C'est l'un des outils réseau le plus utilisé de nos jours et il utilise les techniques cryptographiques les plus évoluées. **ssh** dispose de nombreuses fonctionnalités, options de configuration et bidouilles ingénieuses, suffisamment pour remplir un livre sur le sujet, nous ne verrons donc ici que les bases. Lancez simplement **ssh** avec votre nom d'utilisateur et d'hôte et vous pourrez vous connecter rapidement et de manière sécurisée. Si c'est la première fois que vous vous connectez à cet ordinateur, **ssh** vous demandera de confirmer votre souhait et effectuera une copie locale de la clé cryptographique utilisée. Si cette clé vient à changer plus tard, **ssh** vous avertira et refusera la connexion car il est possible que quelqu'un essaye de détourner la connexion utilisant ce qui est connu comme l'attaque de l'homme du milieu (*man-in-the-middle attack*).

```
darkstar:~# ssh alan@slackware.com
alan@slackware.com's password: secret
alan@slackware.com:~$
```

Le nom d'utilisateur et le nom d'hôte sont de la même forme que celle utilisée pour les adresses de courrier électronique. Si vous n'indiquez pas de nom d'utilisateur **ssh** utilisera votre identifiant utilisateur actuel pour établir la connexion.

tcpdump

Jusqu'à maintenant les outils que nous avons vu étaient destinés à établir des connexions vers d'autres ordinateurs, mais maintenant nous allons regarder le trafic lui-même. **tcpdump**(1) (qui doit être lancé en tant que root) nous permet de voir tout ou partie du trafic réseau émanant ou reçu par notre ordinateur. **tcpdump** affiche les paquets de données bruts de manière différente tout en laissant les en-tête réseau intacts. Ne soyez pas inquiétés si vous ne comprenez pas tout ce qui est affiché, **tcpdump** est un outil pour les administrateurs système et réseau professionnels. Par défaut, il observe la première carte réseau qu'il trouve, mais si vous avez plusieurs interfaces utilisez

simplement l'option *-i* pour préciser laquelle vous intéresse. Vous pouvez aussi limiter la quantité de données affichées en utilisant des expressions et modifier la façon dont elles sont affichées, mais cela est mieux expliqué dans la page de manuel et autres références.

```
darkstar:~# tcpdump -i wlan0
tcpdump: verbose output suppressed, use -v or -vv for full protocol
decode
listening on wlan0, link-type EN10MB (Ethernet), capture size 96 bytes
13:22:28.221985 IP gw.ctsmacon.com.microsoft-ds > 192.168.1.198.59387:
Flags [P.], ack 838190560, win 3079, options [nop,nop,TS val 1382697489
ecr 339048583], length 164WARNING: Short packet. Try increasing the
snap length by 140
SMB PACKET: SMBtrans2 (REPLY)

13:22:28.222392 IP 192.168.1.198.59387 > gw.ctsmacon.com.microsoft-ds:
Flags [P.], ack 164, win 775, options [nop,nop,TS val 339048667 ecr
1382697489], length 134WARNING: Short packet. Try increasing the snap
length by 110
SMB PACKET: SMBtrans2 (REQUEST)
```

nmap

Supposons que vous avez besoin de savoir quels services réseaux sont actifs sur une machine, ou sur plusieurs machines, ou que vous souhaitez déterminer si plusieurs machines sont actives. Vous pouvez envoyer des **ping** sur chacune d'elles, lancer **telnet** pour chaque port qui vous intéresse et noter tous les résultats mais cela serait très ennuyeux et fastidieux. La meilleure alternative serait d'utiliser un scanner de port, et **nmap** est l'outil adapté pour ce travail.

nmap est capable de scanner les ports TCP et UDP, déterminer le système d'exploitation d'un équipement réseau, sonder chacun des services trouvés pour déterminer sa nature et beaucoup, beaucoup plus. La voie la plus simple d'utiliser **nmap** est peut-être d'envoyer un "ping" vers plusieurs ordinateurs en même temps. Vous pouvez utiliser des notations d'adresses réseau (CIDR) ou indiquer une plage d'adresses et **nmap** scannera chacune d'entre elles et retournera les résultats à la fin. Vous pouvez aussi utiliser des noms d'hôtes si vous voulez.

Afin d'envoyer un "ping" vers les hôtes vous devez utiliser l'option *-sP*. La commande suivante ordonne à **nmap** de faire un "ping" sur www.slackware.fr et les 16 adresses IP comprises entre 72.168.24.0 et 72.168.24.15.

```
darkstar:~# nmap -sP www.slackware.com 72.168.24.0/28
```

Pour lancer une analyse de ports, **nmap** propose de nombreuses options. Lorsqu'il est exécuté sans option **nmap** effectue une analyse de ports TCP standard sur tous les hôtes indiqués. Il existe aussi des options pour rendre **nmap** plus ou moins agressif et renvoyer des réponses plus rapidement ou tromper les systèmes de détection d'intrusion. Pour une présentation complète, vous devriez vous référer à la page de manuel assez complète. Les trois commandes suivantes effectuent un scan normal, un SYN scan et un scan "Christmas tree" (*sapin de Noël*).

```
darkstar:~# nmap www.example.com
```

```
darkstar:~# nmap -sS www.example.com
darkstar:~# nmap -sX www.example.com
```



Faites attention ! De nombreux fournisseurs d'accès Internet n'apprécient pas les scans de ports et peuvent prendre des mesures pour vous empêcher de le faire. **nmap** et autres programmes similaires sont mieux employés sur vos propres systèmes à des fins de sécurité et de maintenance, plutôt que pour rechercher ce qui est connecté à Internet.

host

Bien souvent les problèmes réseau proviennent d'une déficience du DNS (*Domain Name Service*) qui associe les noms de domaines aux adresses IP. Un moyen simple pour vérifier les correspondances DNS est d'utiliser la commande **host(1)**. Lorsqu'elle est exécutée votre ordinateur effectuera quelques correspondances DNS et retournera le résultat.

```
darkstar:~# host www.slackware.com
www.slackware.com is an alias for slackware.com.
slackware.com has address 64.57.102.34
slackware.com mail is handled by 1 mail-mx.cwo.com.
```

dig

Les requêtes DNS plus complexes peuvent être effectuées manuellement avec l'outil **dig(1)**. **dig** est "le plus méchant chien de la fourrière" lorsque l'on a besoin de dépanner des problèmes de DNS. Avec cet outil vous pouvez virtuellement effectuer n'importe quelle requête DNS pour des enregistrements A, CNAME, MX, SP, TXT et bien plus. Il y a bien trop d'options et de types de requêtes pour toutes les présenter ici, mais la page de manuel présente les options les plus courantes.

```
darkstar:~# dig @207.69.188.185 www.slackware.com a

; <<>> DiG 9.4.3-P4 <<>> @207.69.188.185 www.slackware.com a
; (1 server found)
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 57965
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;www.slackware.com.      IN      A

;; ANSWER SECTION:
www.slackware.com.     86400   IN      CNAME   slackware.com.
slackware.com.        86400   IN      A       64.57.102.34

;; AUTHORITY SECTION:
```

```
slackware.com.      86400      IN         NS         ns2.cwo.com.  
slackware.com.      86400      IN         NS         ns1.cwo.com.  
  
;; Query time: 348 msec  
;; SERVER: 207.69.188.185#53(207.69.188.185)  
;; WHEN: Sat Jul 3 16:25:10 2010  
;; MSG SIZE rcvd: 105
```

Jetons un œil sur les options de la ligne de commande utilisée ci-dessus. L'option `@207.69.188.185` indique à **dig** quel serveur DNS interroger. Si rien n'est indiqué, **dig** utilisera simplement n'importe quel serveur listé dans `/etc/resolv.conf`. L'option `a` à la fin de la commande est le type d'enregistrement DNS recherché. Dans ce cas nous recherchons un enregistrement de type "A" qui doit correspondre à une adresse IPv4.

finger

finger(1) n'est pas tout à fait un outil de diagnostic réseau mais plutôt un outil de diagnostic d'utilisateur de réseau. En utilisant **finger** vous pouvez récupérer des informations utiles sur les utilisateurs des serveurs exécutant le daemon **fingerd**(8). Aujourd'hui très peu de serveurs proposent encore **fingerd**, mais pour ceux qui le font cela peut-être un outil utile pour suivre les activités de vos amis et collègues.

```
darkstar:~# finger alan@cardinal.lizella.net  
[cardinal.lizella.net]  
Login: alan                      Name: Alan Hicks  
Directory: /home/alan             Shell: /bin/bash  
Office: 478 808 9919, 478 935 8133  
On since Wed Apr 13 17:43 (UTC) on pts/9 from  
75-150-12-113-atlanta.hfc.comcastbusiness.net  
    32 minutes 24 seconds idle  
    (messages off)  
On since Wed Apr 13 17:45 (UTC) on pts/10 from :pts/9:S.0  
    48 minutes 56 seconds idle  
Mail forwarded to alan@lizella.net  
No mail.  
No Plan.
```

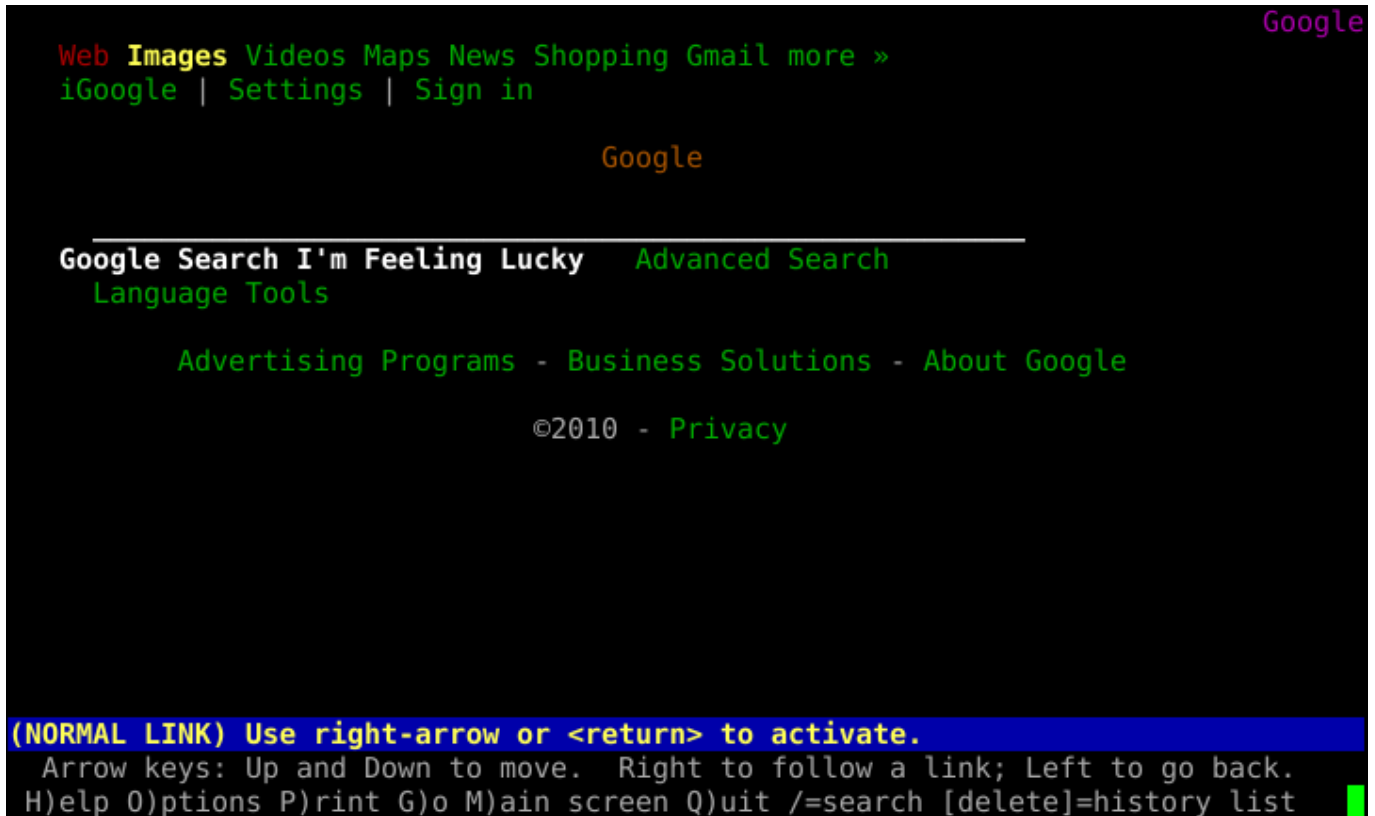
Navigateurs web

Slackware propose un grand choix de navigateurs web. Si vous utilisez un bureau graphique, vous trouverez **Firefox**, **Seamonkey** et d'autres qui vous seront sans doute familiers, mais comment faire depuis un accès en console ? Heureusement, quelques navigateurs web utiles sont aussi présents.

lynx

Le plus ancien navigateur web orienté texte fourni par Slackware est définitivement **lynx**(1), un

navigateur web pratique quoi que parfois limité. **lynx** ne gère pas les *frames*, javascript ou les images, c'est un navigateur en mode texte uniquement. La navigation s'effectue avec les touches fléchées du clavier et, de manière optionnelle, à la souris. Bien qu'il manque de fonctions par rapport à d'autres navigateurs, **lynx** est l'un des plus rapides que vous puissiez utiliser pour récupérer des informations. Par exemple, l'option `-dump` envoie une page web formatée directement vers la console, qui peut être renvoyée via un *pipe* à d'autres programmes.



```
Google
Web Images Videos Maps News Shopping Gmail more »
iGoogle | Settings | Sign in

Google

Google Search I'm Feeling Lucky Advanced Search
Language Tools

Advertising Programs - Business Solutions - About Google

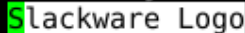
©2010 - Privacy

(NORMAL LINK) Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

links

Une alternative plus riche en fonctionnalités est le populaire **links**(1), un navigateur web orienté console qui supporte les *frames* et affiche les tableaux mieux que **lynx**. Comme son prédécesseur, **links** utilise les touches fléchées pour la navigation et l'utilisation de la souris est possible. Au contraire de **lynx**, il propose aussi un menu agréable (cliquez simplement en haut d'une ligne avec votre souris pour le faire apparaître) et d'une manière générale propose un meilleur rendu des pages web.

The Slackware Linux Project (p1 of 15)

The Slackware Linux Project 

<p>News</p> <p>Security Advisories</p> <p>FAQ</p> <p>Book</p> <p>General Info</p> <p>Get Slack</p> <p>Install Help</p> <p>Configuration</p> <p>Packages</p> <p>ChangeLogs</p>	<p>Slackware 13.0 is released!</p> <p>After one of the most intensive periods of development in Slackware's history, the long awaited stable release of Slackware 13.0 is ready. This release brings with it many major changes since Slackware 12.2, including a completely reworked collection of X packages (a configuration file for X is no longer needed in most cases), major upgrades to the desktop environments (KDE version 4.2.4 and Xfce version 4.6.1), a new .txz package format with much better compression, and other upgrades all around -- to the development system, network services, libraries, and major applications like Firefox and Thunderbird. We think you'll agree</p>	<p>2009-08-27</p>
---	--	--------------------------

<http://www.slackware.com/index.html>

wget

À l'opposé des autres navigateurs web que nous avons déjà vu, **wget**(1) n'est pas interactif. Au lieu d'afficher le contenu HTTP, **wget** le télécharge. Ce qui permet la "navigation" en dehors du navigateur web. Au contraire des modes de rendu des autres navigateurs, **wget** ne formate pas ce qu'il télécharge, il copie le contenu de façon identique à ce qui se trouve sur le serveur web conservant toutes les balises et le contenu binaire en place. Il supporte aussi plusieurs options de récursivité pour créer un miroir d'un contenu en ligne vers votre ordinateur local. **wget** ne gère pas uniquement HTTP, il supporte aussi FTP et plusieurs autres protocoles.

```
darkstar:~# wget
ftp://ftp.osuosl.org/pub/slackware/slackware-current/ChangeLog.txt
--2010-05-01 13:51:19--
ftp://ftp.osuosl.org/pub/slackware/slackware-current/ChangeLog.txt
=> `ChangeLog.txt'
Resolving ftp.osuosl.org... 64.50.236.52
Connecting to ftp.osuosl.org|64.50.236.52|:21... connected.
Logging in as anonymous ... Logged in!
==> SYST ... done.      ==> PWD ... done.
==> TYPE I ... done.    ==> CWD /pub/slackware/slackware-current ... done.
==> SIZE ChangeLog.txt ... 75306
==> PASV ... done.     ==> RETR ChangeLog.txt ... done.
Length: 75306 (74K)

100%[=====>] 75,306      110K/s   in 0.7s

2010-05-01 13:51:22 (110 KB/s) - `ChangeLog.txt' saved [75306]
```


Clients de courrier électronique

Slackware propose aussi un large choix de clients de courrier électronique. Si vous utilisez un environnement de bureau graphique, vous trouverez **Thunderbird**, **Kmail**, **sylpheed** et d'autres. Comme pour les navigateurs web, il existe également des applications qui fonctionnent depuis l'invite de commande. Une fois que vous avez commencé à utiliser un client de courrier électronique en mode console, il se peut que vous ne vouliez plus utiliser autre chose, les possibilités de configuration et de versatilité peuvent être addictives.

pine

pine est l'un des plus anciens client de courrier électronique en ligne de commande en activité et reste le plus convivial. **pine** a été créé à l'université de Washington et porte une licence et un marque commerciale avec lesquels il est difficile de travailler. Heureusement en 2005 l'université a produit une ré-écriture sans marque et avec une licence plus libre et **alpine**(1), clone de pine fournit avec Slackware, apparut.

Pour commencer à utiliser **alpine**, tapez simplement **pine** en ligne de commande. Son usage est très simple car tout se fait via un système de menus et l'aide pour les commandes se trouve au bas de l'écran. Voyez par vous-même :

```
ALPINE 2.02(1266) MAIN MENU      Folder: INBOX   No Messages

?  HELP                Get help using Alpine
C  COMPOSE MESSAGE     Compose and send a message
I  MESSAGE INDEX       View messages in current folder
L  FOLDER LIST         Select a folder to view
A  ADDRESS BOOK        Update address book
S  SETUP               Configure Alpine Options
Q  QUIT                Leave the Alpine program

      Copyright 2006-2008 University of Washington
      Copyright 2009-2010 Re-Alpine Project
      [Folder "INBOX" opened with 0 messages]

? HELP                P PrevCmd          R RelNotes
O OTHER CMDS         > [ListFoldrs]   N NextCmd          K KBlock
```

Avant de configurer un quelconque client de courrier électronique vous devez vérifier la documentation de votre serveur de courrier pour récupérer toutes les informations pertinents concernant les protocoles et moyens de sécurisation que votre service de courrier utilise. Cela vous aidera à configurer **pine** correctement. Par défaut, **pine** cherchera à relever des messages sur un serveur de courrier fonctionnant sur votre ordinateur. À moins que vous n'exécutiez un tel service de courrier (beaucoup de gens le font) ce n'est probablement pas ce que vous souhaitez. Heureusement configurer **pine** est un processus assez clair. Entrez dans le menu [S]etup et choisissez l'option [C]onfiguration. Vous aurez le choix d'entrer votre nom, le chemin de votre boîte de courrier, le serveur SMTP et de nombreuses autres options.

mutt

Certaines personnes n'aiment pas **pine**. Certaines personnes veulent plus de contrôle. Certaines personnes veulent un client de courrier électronique supportant des extensions et un comportement qui ne soit pas illogique. Ces personnes utilisent **mutt**(1). **mutt** n'est pas aussi convivial que **pine** mais compense par sa richesse. Vous ne trouverez pas de référence de commandes au bas de l'écran, **mutt**(1). **mutt** utilise chaque centimètre carré pour le traitement du courrier. Son support de fonctionnalités est large. Vous pouvez configurer **mutt** avec un fichier `.muttrc` dans votre répertoire personnel. Avec toutes les différentes options de configuration possibles, il y a même une page de manuel pour cela, `muttrc(5)`. Vous voudrez sûrement prendre le temps de la lire.

```
q:Quit d:Del u:Undel s:Save m:Mail r:Reply g:Group ?:Help
5424 Aug 19 slakmagik (3.9K) Re: [Slackbuilds-users] rc2 slac
5425 Aug 19 LukenShiro (3.0K) Re: [Slackbuilds-users] eric ide
5426 Aug 19 B Watson (3.8K) Re: [Slackbuilds-users] eric ide
5427 Aug 19 Marcin Herda (7.2K) Re: [Slackbuilds-users] i3 updat
5429 Aug 19 Binh Nguyen ( 26K) [Slackbuilds-users] My SlackBuil
5430 Aug 20 rudsonalves ( 77K) Re: [Slackbuilds-users] eric ide
5432 Aug 21 Nicolas Kovacs (3.3K) Re: [Slackbuilds-users] murrine-
5433 N Aug 21 Mr. B-o-B (3.3K) [Slackbuilds-users] unixODBC bus
5434 N Aug 21 Mikko Varri (2.9K) [Slackbuilds-users] REQUIRES inf
5435 N Aug 21 xgizzmo@slackbu (3.3K) Re: [Slackbuilds-users] REQUIRES
5436 N Aug 21 Mikko Varri (4.0K) Re: [Slackbuilds-users] REQUIRES
5437 N Aug 22 Christoph Willi (4.2K) Re: [Slackbuilds-users] REQUIRES
5438 N Aug 21 Robby Workman (5.2K) Re: [Slackbuilds-users] REQUIRES
5439 N Aug 21 King Beowulf (5.6K) [Slackbuilds-users] 13.37 nvidi
-Mutt: =slackbuilds-users [Msgs:5442 New:4 29M]--(date/date)--(100%)--
```

Utiliser **mutt** est quelque chose d'unique du fait de sa nature de *Mail User Agent (MUA)*, ce qui signifie que son but avoué est de lire et trier du courrier électronique. Ce n'était pas son travail initialement, bien que certaines fonctions supplémentaires comme récupérer du courrier via POP3 et même le basique transfert de messages par SMTP se soient faufilees dans le programme.

Comme c'est souvent le cas pour les applications robustes en mode console, les options de configuration foisonnent et il n'y a pas de "bonne" ou "mauvaise" façon d'utiliser **mutt** tant que cela correspond à ce que vous souhaitez. Une chose à garder à l'esprit si vous souhaitez utiliser **mutt** pour la gestion du courrier est que les possibilités d'envoi et de réception de courrier sont très limitées. **mutt** se focalise uniquement sur le tri, la lecture et la rédaction de messages en plus des autres rôles traditionnels d'un *Mail User Agent*. Cela correspond à la philosophie UNIX d'utiliser de petits outils qui font correctement leur travail et qui peuvent être combinés (ou "chaînés") avec d'autres outils pour achever une tâche quelconque. Une fois cela compris vous aurez au minimum besoin de mettre en place un outil supplémentaire pour recevoir votre courrier.

Les commandes pour naviguer dans **mutt** sont personnalisables mais celles par défaut peuvent être listées avec la touche `?`.

mailx

Tout cela est bien et bon mais si vous voulez juste utiliser un client de courrier électronique sans

menu ? Heureusement **mailx** est là pour vous sauver.

mailx est basé sur l'application Mail de Berkeley apparue avec la commande **mail** de la version 1 de l'UNIX d'AT&T. Il peut être utilisé de manière interactive ou non.

mailx consulte le courrier depuis la file d'attente de courrier (*mail spool*) de l'ordinateur et affiche les champs habituels pour l'expéditeur, le sujet, le statut et la taille sous forme de liste, laissant l'utilisateur devant une invite de commande. En fait, cela vous semblera familier si vous avez déjà pris le temps de vérifier votre courrier juste après avoir installé Slackware et lu le message de bienvenue de Pat Volkerding.

```
darkstar:~# mailx
Heirloom mailx version 12.4 7/29/08.Type ? for help.
"/var/spool/mail/root": 2 messages 2 new
>N1 To rootThu Mar 10 23:33 52/1902Register with the Linux counter project
  N2 To rootThu Mar 10 23:35321/15417 Welcome to Linux (Slackware 14.0)!
?;
```

Pour lire un message, entrez le numéro correspondant à l'invite de commande. Cela affichera le message **more*** et vous pourrez utiliser la touche **ENTRÉE** pour voir la page suivante. Une fois à la fin du message appuyez sur la touche **q** pour retourner à la liste initiale ou sur **ENTRÉE** pour passer au message suivant.

Pour connaître la liste des commandes disponibles, appuyez sur la touche **?** à l'invite de commande. Avec les commandes affichées vous pourrez lire les en-têtes des messages en attente, répondre, supprimer, sauvegarder et tout autre commande nécessaire à la gestion des messages.

mailx est beaucoup plus utile lorsqu'il est utilisé dans des scripts. Pour toutes les options disponibles avec **mailx**, consultez la page de manuel. Le moyen le plus simple d'envoyer un message à quelqu'un ne demande que la commande elle-même et l'adresse du destinataire.

```
darkstar:~$mailx bob@example.com
```

Après que la commande soit lancée, une invite de commandes apparaît pour la ligne du sujet, le corps du message et le caractère de fin (une ligne ne contenant qu'un unique point).

Cependant **mailx** peut être utilisé sans aucune intervention humaine. D'une manière générale il est bon de présumer que toute option que vous pouvez définir via l'invite de commande de **mailx** peut aussi être définie via un script ou en utilisant une commande non-interactive.

```
darkstar:~$ mailx -n -s "Test message" bob@example.com < ~/message.txt
```

Dans cet exemple, le contenu du fichier message.txt sera envoyé en tant que corps du message au destinataire indiqué. Aucune intervention de l'utilisateur n'est requise.

Que ce soit vers votre propre ordinateur (localhost) ou votre propre réseau, envoyer du courrier de cette manière est tout à fait possible. Mais via Internet quelques étapes supplémentaires sont souvent requises. Bien sûr, le plus notable est d'avoir un serveur SMTP capable de délivrer votre courrier électronique. Cela également peut être indiqué comme option à la commande **mail** :

```
darkstar:~$env MAILRC=/dev/null
```

```
from="bob@example.com (Bob Dobbs)"  
smtp=relay.example.com mail -n -s "Test message" connie@example.com <  
~/message.txt
```

Dans ce cas, la variable *MAILRC* est définie à vide pour ne pas tenir compte des réglages du système et leur serveur SMTP comme la ligne *FROM:* sont définies. Le reste de la commande demeure identique à l'utilisation de **mailx** sur l'ordinateur ou le réseau local.

Au-delà de cela, **mailx** est souvent vu comme un client de courrier électronique minimaliste, ce qui est assez vrai, mais lorsque vous avez besoin d'écrire un script avec des courriers de notification ou des messages de mises à jour importantes, il devient bien plus utile qu'une application interactive telle que **pine** ou **mutt**.

Clients FTP

Beaucoup de données sont stockées sur des serveurs FTP de par le monde. En fait, Slackware Linux fut initialement disponible publiquement via FTP et continue à être distribué de cette façon à l'heure actuelle. Beaucoup de logiciels libres peuvent être téléchargés sous forme de code source ou de binaires via FTP, et savoir comment récupérer ces fichiers peut s'avérer une connaissance utile.

ftp

Le client FTP le plus simple fournit avec Slackware se nomme simplement **ftp**(1) et peut servir pour les tâches basiques d'envoi et réception de données. **ftp** se connecte à un serveur FTP, demande un nom d'utilisateur et un mot de passe et vous permet de déposer ou récupérer des données vers ou depuis ce serveur. **ftp** est tombé en désuétude auprès des utilisateurs expérimentés du fait de son manque de fonctionnalités, mais reste un outil utile, et beaucoup de documentations que vous pourrez lire y font référence.

Une fois que la session FTP est initialisée, vous vous trouvez devant une invite de commande similaire à celle d'un shell. À partir de là vous pouvez changer de répertoire et lister leur contenu avec les commandes "*cd*" et "*ls*", tout comme en shell. De plus, vous pouvez utiliser la commande "*put*" pour envoyer un fichier vers le serveur ou "*get*" pour récupérer des données depuis le serveur. Si vous êtes connectés sur un serveur FTP public, vous pourrez utiliser l'identifiant "*anonymous*" et simplement entrer votre adresse de courrier électronique (pas forcément valide) comme mot de passe.

```
darkstar:~$ ftp ftp.osuosl.org  
Name (ftp.osuosl.org:alan): anonymous  
331 Please specify the password.  
Password: secret  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> cd pub/slackware/slackware-current/  
250 Directory successfully changed.  
ftp> get ChangeLog.txt
```

```
local: ChangeLog.txt remote: ChangeLog.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for ChangeLog.txt (33967 bytes).
226 File send OK.
33967 bytes received in 0.351 secs (94 Kbytes/sec)
ftp> bye
221 Goodbye.
```

ncftp

ncftp(1) (prononcez nick-f-t-p), est un héritier plus riche en fonctions de **ftp**, proposant la complétion de commandes et le transfert récursif de fichiers. Il se connecte automatiquement à un serveur en mode anonyme, à moins que vous n'indiquiez un nom d'utilisateur différent avec l'option **-u** en ligne de commande. L'avantage majeur par rapport à **ftp** est la possibilité d'envoyer ou recevoir plusieurs fichiers à la fois avec les commandes "**mput**" et "**mget**". Si vous utilisez l'option **-R** pour chacune d'entre elles, cela déposera ou récupérera les données depuis des répertoires.

```
darkstar:~# ncftp ftp.osuosl.org
Logging in...
Login successful.
Logged in to ftp.osuosl.org.
ncftp / > cd pub/slackware/slackware-current
Directory successfully changed.
ncftp ..ware/slackware-current > mget -R isolinux
isolinux/README.TXT: 4.63 kB 16.77 kB/s
isolinux/README_SPLIT.TXT: 788.00 B5.43 kB/s
isolinux/f2.txt: 793.00 B5.68 kB/s
isolinux/initrd.img:13.75 MB 837.91 kB/s
isolinux/iso.sort:50.00 B354.50 B/s
isolinux/isolinux.bin:14.00 kB 33.99 kB/s
isolinux/isolinux.cfg: 487.00 B3.30 kB/s
isolinux/message.txt:760.00 B5.32 kB/s
isolinux/setpkg: 2.76 kB 19.11 kB/s
ncftp ..ware/slackware-current > bye
```

lftp

Le dernier client que nous allons voir est **lftp**(1). Comme **ncftp**, il supporte la complétion de commandes et les transferts récursifs, mais il dispose d'une licence plus sympathique. Au lieu d'utiliser "**mget**" et "**mput**", toutes les opérations récursives sont prises en charge par la commande "**mirror**". "**mirror**" dispose de plusieurs options, je vous conseille donc de consulter la page de manuel ou l'aide intégrée pour plus de détails.

```
darkstar:~# lftp ftp.osuosl.org
lftp ftp.osuosl.org:~> cd /pub/slackware/slackware-current
cd ok, cwd=/pub/slackware/slackware-current
lftp ftp.osuosl.org:/pub/slackware/slackware-current> mirror isolinux
Total: 2 directories, 16 files, 1 symlink
```

```
New: 16 files, 1 symlink
14636789 bytes transferred in 20 seconds (703.7K/s)
lftp ftp.osuosl.org:/pub/slackware/slackware-current> bye
```

rsync

Envie de voir quelque chose de sympa ? Vous êtes-vous déjà retrouvés à chercher quelques fichiers dans un grand répertoire, mais sans savoir exactement lesquels vous avez déjà et ceux qui vous manquent ? Vous pouvez tout re-télécharger mais la copie demandera beaucoup de travail. Vous pouvez chercher et prendre ce qu'il vous faut, en vérifiant manuellement, mais c'est fastidieux. Peut-être avez vous téléchargé un gros fichier comme une image ISO, mais quelque chose s'est mal passé pendant l'opération ? Ce ne serait pas très logique de tout récupérer à nouveau si seuls quelques bits ont été corrompus. Je vous présente donc **rsync**(1), un outil de copie rapide et versatile pour les fichiers locaux et distants.

rsync utilise une poignée de simples, mais efficaces techniques pour déterminer ce qui à besoin d'être changé. En vérifiant la taille d'un fichier et son horodatage, il peut déterminer si deux fichiers sont différents. Si quelque chose à changé, il peut déterminer quels octets sont différents, et simplement télécharger ce qui est utile plutôt que le fichier complet. C'est une vraie merveille de la technologie moderne.

Dans la forme la plus simple, **rsync** se connecte à un serveur rsync et récupère une liste de fichiers et répertoires, ainsi que leurs tailles, horodatages et autres informations. Il compare tout cela avec les (éventuels) fichiers locaux pour déterminer ce qu'il doit transférer. Seuls les fichiers qui diffèrent seront synchronisés. De plus, il peut découper de gros fichiers en petits morceaux et comparer ces morceaux en utilisant une fonction de condensat (*hash*) simple et rapide. Les morceaux qui correspondent ne sont pas transférés, ce qui réduit grandement la quantité de données à copier. **rsync** supporte également la compression, les sorties verbeuses, la suppression de fichiers, la gestion des permissions et de nombreuses autres options. Pour une liste complète vous aurez besoin de vous référer à la page de manuel, mais j'ai inclus un petit tableau des options les plus courantes.

Table 16.1. Options de rsync

-v	augmente la verbosité
-c	effectue une somme de contrôle (<i>checksum</i>) plutôt que de comparer la taille et l'horodatage des fichiers
-a	mode archive (équivalent à -rlptgoD)
-e	indique le shell distant à utiliser
-r	mode récursif
-u	mise à jour - n'envoie pas les fichiers plus récents par rapport au destinataire
-p	conserve les permissions
-n	non-effectif - lance une opération de test sans effectuer de changement
-z	active la compression - pratique pour les connexions réseau lentes

Du fait de sa richesse et de sa versatilité, **rsync** peut être lancé de nombreuses façons différentes. Dans les deux exemples suivants, le premier lance une connexion à un serveur rsync pour récupérer des informations et le second se connecte à un serveur via SSH pour chiffrer la transmission.

```
darkstar:~# rsync -avz rsync://ftp.osuosl.org/pub/slackware/slackware-  
current/ /src/slackware-current/  
darkstar:~# rsync -e ssh ftp.slackware.com:/home/alan/foo /tmp/foo
```

Navigation

Chapitre précédent : [Réseau sans fil](#)

Chapitre suivant : [Gestion de paquets](#)

Sources

- Source originale : <http://www.slackbook.org/beta>
- Publication initiale d'Alan Hicks, Chris Lumens, David Cantrell, Logan Johnson
- Traduction initiale de [escaflown](#)
- Traduction de [Ellendhel](#)

[slackbook](#), [networking](#), [mail clients](#), [web browsers](#), [rsync](#)

From:
<https://docs.slackware.com/> - **SlackDocs**

Permanent link:
https://docs.slackware.com/fr:slackbook:basic_networking_utilities

Last update: **2013/10/13 20:47 (UTC)**

