APRS su Linux con Xastir Installazione dai sorgenti

L'installazione di Xastir

Per installare Xastir non è richiesto essere un guru di Linux, anche se una conoscenza minima della piattaforma è necessaria. Questo perché dobbiamo "costruirci" a mano sia il programma sia le librerie che lo stesso sfrutta per il suo funzionamento.

Ok senza preoccuparci più di tanto, iniziamo l'opera. Diamo per scontato che abbiate installato una copia completa e funzionante di Ubuntu 10.04. In realtà vanno bene anche altre distribuzioni o altre versioni, ma su distribuzioni non Debian based, potreste non trovare il comando apt-get. In tal caso provate a usare YUM o fate riferimento alla documentazione della vostra distribuzione.

NOTA: le immagini fanno riferimento alla versione inglese di Ubuntu, ma è tranquillamente possibile portare a termine l'installazione di Xastir anche sulla versione italiana del sistema, senza nessuna modifica.

NOTA: per portare a termine l'installazione è necessario avere una connessione ad Internet, per scaricare i sorgenti e i componenti necessari.

Per prima cosa verifichiamo che la nostra installazione sia aggiornata. Per farlo possiamo procedere o tramite interfaccia grafica (menu **System, Adminstration** quindi scegliere **Update Manager**) o tramite

terminale.

Visto che per l'installazione lavoreremo sempre con il terminale... facciamo che iniziare a usarlo ;-). Quindi dal menù **Application** scegliamo la voce **Accessories** quindi **Terminal**. Visto che il terminale è probabilmente l'applicazione più utile, facciamo che aggiungerlo al menù superiore (sempre dal menù Application scegliamo la voce Accessories quindi tasto destro sopra Terminal e scegliamo **Add this launcher to the panel** o **Aggiungi questa icona di avvio al pannello**).



All'apertura del terminale ci spostiamo su di esso e diamo il comando **sudo apt-get update**. Analizziamo brevemente il suo significato:

- sudo: senza entrare nei dettagli, nei sistemi operativi Unix c'è un utente particolare, detto super user e contraddistinto dall'avere un UID (User ID) uguale a 0 e nome utente root, che ha totale accesso al sistema senza nessuna restrizione, cioè ne è l' amministratore. Quando l'amministratore di sistema ha la necessità di svolgere mansioni di amministrazione apre un terminale e avvia una sessione come utente root, oppure se si trova già in un terminale come utente normale usa il comando su per diventare utente root. Ubuntu usa un sistema diverso per compiere operazioni amministrative, basato sull'utilizzo del comando sudo. Con sudo l'utente non ha la necessità di ricordare per sempre la password di un account diverso dal suo (root). Inoltre, costringendo all'esecuzione controllata dei singoli comandi, si evitano eventuali disattenzioni nell'utilizzo degli stessi, chiedendo la password nel caso si stiano per lanciare comandi amministrativi e mantenendo un log completo dei comandi eseguiti, permettendo l'analisi di ciò che è avvenuto nel sistema.
- apt-get update: Il comando apt-get è lo strumento a riga di comando usato per interagire con l'APT (Advanced Packaging Tool) di Debian (distribuzione storica su cui si basa Ubuntu) per eseguire l'installazione di nuovi pacchetti software, l'aggiornamento del software esistente, l'aggiornamento dell'indice dell'elenco di pacchetti e anche l'aggiornamento del l'intero sistema operativo. In particolare, l'opzione update fa si che venga aggiornato l'elenco dei software installati sulla macchina (presenti negli archivi elencati in /etc/apt/sources.list).

NOTA: questo comando ci chiederà di inserire la password del proprio utente che deve far parte del gruppo "sudoers" del sistema.

Eseguito l'update, proseguiamo con il comando **sudo apt-get upgrade**. L'opzione upgrade confronta la versione dei singoli pacchetti installati nel sistema con quella presente nei repository e se necessario propone l'aggiornamento (un po' come fa Windows Update, tanto per capirci). Se il sistema è aggiornato, ci verrà restituita una riga del tipo 0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.



Ok adesso possiamo veramente partire con l'installazione di Xastir.

Installazione dei "prerequisiti"

Per prima cosa installiamo gli strumenti necessari alla compilazione di un programma dal suo codice sorgente. Per farlo utilizziamo il comando **sudo apt-get install build-essential**. L'opzione install... installa quello che viene specificato di seguito nel nostro sistema.



Il sistema si connette ad Internet, verifica se il package è disponibile e ci risponde con qualcosa di simile a quanto riportato sopra, in cui ci dice che verranno installati 8 pacchetti nuovi per un totale di circa 25 MB di spazio usato.

Accettiamo con Y e proseguiamo installando tutte le librerie necessarie per il funzionamento di Xastir: cvs, autoconf, automake, xorg-dev, imagemagick, gv, libxp-dev, lesstif2-dev e libcurl3-dev. Per farlo possiamo procedere eseguendo singolarmente le operazioni di apt-get install o "concatenando" i comandi in modo da eseguire una "singola" istruzione sudo apt-get install cvs autoconf automake xorg-dev imagemagick gv libxp-dev lesstif2-dev libcurl3-dev.

Anche in questo caso accettiamo l'installazione dei nuovi pacchetti e attendiamo che gli stessi vengano scaricati e installati nel sistema.

Proseguiamo con l'installazione di una serie di componenti "opzionali" ma che ci verranno utili in seguito, con il comando sudo apt-get install gpsman gpsmanshp libpcre3-dev libdb4.8dev python-dev libax25-dev shapelib libshp-dev festival festival-dev libmagickcore-dev (ad esempio festival ci consentirà di far "parlare" Xastir).

Scarichiamo i sorgenti

A questo punto è necessario scaricare i sorgenti di Xastir. Per farlo utilizzeremo il CVS (Concurrent Versioning System), un sistema particolarmente diffuso nel mondo dell'open-source per la gestione di progetti software. Per prima cosa è necessario preparare l'ambiente per agganciarci al CVS. Verifichiamo se esiste già sul nostro sistema una configurazione con il **comando ls -l ~/.cvsrc**. Prima di impazzire per trovare la tilde (~) sulla nostra tastiera, sappiate che se la stessa è in italiano... non c'è! Ma ~/ significa semplicemente la cartella home dell'utente che state utilizzando, ad esempio nel mio caso /home/iz1mhn. Quindi il comando si può riscrivere come **ls -l /home/iz1mhn/.cvrsc**.



In entrambi i casi se la configurazione non è presente ci verrà restituito un messaggio di errore che potremo tranquillamente ignorare!

Se invece ci viene restituito un file, controlliamo il suo contenuto e eventualmente facciamone una copia di backup prima di modificarlo.

In tutti i casi procediamo con la creazione (o la modifica) del file con il comando **nano** ~/.cvsrc che aprirà l'editor nano già sul file corretto. Inseriamo al suo interno i seguenti parametri (uno per riga!): cvs -z3, update -P -d, status -v ed infine diff -u, quindi usciamo con CRTL + X e confermiamo il salvataggio del file con Y accettando il nome che ci viene proposto.



Proseguiamo creando una sottocartella della nostra home directory che chiamiamo src con il comando **mkdir** ~/src. Ci sposiamo al suo interno (cd ~/src) quindi ci validiamo sul server cvs di Xastir con il commando

cvs -d:pserver:anonymous@xastir.cvs.sourceforge.net:/cvsroot/xastir login. Ci verrà chiesto di inserire una password, diamo semplicemente Invio e procediamo scaricando Xastir con il comando

cvs -d:pserver:anonymous@xastir.cvs.sourceforge.net:/cvsroot/xastir co xastir.



L'esecuzione del download può essere lunga, quindi armiamoci di pazienza e attendiamo che la stessa finisca. Dovesse capitare che vada in time out, rilanciammo semplicemente il comando che ripartirà da dove si era precedentemente interrotto.



Terminato il download ci spostiamo nella cartella xastir (cd xastir) e tramite il comando ./bootstrap.sh (attenzione al punto!) generiamo il makefile e il file di configurazione.

proj.4 e geotiff

Queste librerie sono necessarie per il supporto a due formati (USGS Topos e Aerial Photos), ma più in generale sono usate per l'accesso a sistemi GISS. L'installazione non è necessaria per il corretto

funzionamento di Xastir. Valutate se pensate di averne bisogno o se preferite risparmiare cinque minuti di tempo.

Sfortunatamente non possiamo usare un semplice apt-get install (a causa di problemi delle versioni presenti sui repository) ma abbiamo due strade da scegliere: usare un repository non ufficiale ma affidabile (UbuntuGIS) o installarci il tutto a mano compilandoci dai sorgenti anche le librerie.

Installazione via UbuntuGIS

Partiamo dal primo metodo, forse leggermente più semplice. Innanzitutto è necessario aggiungere i due repository all'interno dell'elenco degli archivi che il nostro sistema consulta per le installazioni. Per farlo aggiungeremo due righe in coda al file /et/apt/sources.list. il sistema più veloce è ridirigere il comando echo, con i comandi

```
echo "deb http://ppa.launchpad.net/ubuntugis/ubuntugis-unstable/ubuntu
lucid main" >> /etc/apt/sources.list
```

```
echo "deb-src http://ppa.launchpad.net/ubuntugis/ubuntugis-
unstable/ubuntu lucid main" >> /etc/apt/sources.list
```

Terminata la modifica all'elenco degli archivi aggiorniamo l'elenco stesso con un **apt-get update**. Ci verrà restituito un errore, dato che al repository ppa.launchpad.net manca la chiave pubblica, lo ignoriamo e installiamo le due librerie con **apt-get install libproj-dev libgeotiff-dev**.

Visto che useremo diversi comandi da super utente, invece che preciderli singolarmente con il sudo, diamo all'inizio il comando **sudo** –i e al termine il comando **exit**.

In pratica i comandi che daremo sono i seguenti:

sudo -i

echo "deb http://ppa.launchpad.net/ubuntugis/ubuntugis-unstable/ubuntu lucid main" >> /etc/apt/sources.list

echo "deb-src http://ppa.launchpad.net/ubuntugis/ubuntugisunstable/ubuntu lucid main" >> /etc/apt/sources.list

```
apt-get update
```

apt-get install libproj-dev libgeotiff-dev

exit

Installazione dai sorgenti

È un metodo forse leggermente più complesso ma forse leggermente più stabile e affidabile: partendo dai sorgenti avremo sempre un'installazione pulita e "sicura"!

Iniziamo con proj.4; per prima cosa ci spostiamo nella cartella /src quindi ci scarichiamo con wget la libreria dal sito download.osgeo.org. Il tutto con questi tre comandi:

cd ~/src

```
wget http://download.osgeo.org/proj/proj-4.7.0.tar.gz
```

wget http://download.osgeo.org/proj/proj-datumgrid-1.4.tar.gz

Procediamo estraendo I due file con tar, ma prima di estrare il secondo ci posizioniamo nella cartella corretta

tar zxvf proj-4.7.0.tar.gz
cd proj-4.7.0/nad
tar xzf ../../proj-datumgrid-1.4.tar.gz
cd ..
Compiliamo e installiamo la libreria
./configure
make
sudo make install
sudo ldconfig

Ora proseguiamo con la seconda libreria, libgeotiff, in modo analogo: scarichiamo il sorgente, lo scompattiamo, compiliamo e installiamo.

Scarichiamo i file

cd ~/src

```
wget http://download.osgeo.org/geotiff/libgeotiff/libgeotiff-1.2.5.tar.gz
```

Scompattiamo l'archivio

tar zxvf libgeotiff-1.2.5.tar.gz

cd libgeotiff-1.2.5/

Compiliamo e installiamo libgeotiff

./configure --with-ld-shared="gcc -shared"

make

sudo make install

sudo ldconfig

Ok anche questo passo è finito e possiamo passare oltre!

Compiliamo Xastir

Indipendentemente dal metodo scelto per l'installazione delle due librerie (proj.4 e geotiff) siamo alla fine: non ci resta che compilare Xastir per poterlo usare. Per farlo ci spostiamo nella cartella dove abbiamo scaricato i sorgenti (/home/iz1mhn/src/xastir) con il comando cd ~/src/xastir, creaiamo una cartella build (**mkdir build**) ci spostiamo in questa cartella (**cd build**) quindi lanciamo il comando ../configure (attenzione ai punti davanti al comando).



NOTA: se abbiamo scelto di installare le librerie dal repository UbuntuGIS, dobbiamo utilizzare il comando ../configure CPPFLAGS="-I/usr/include/geotiff" invece del ../configure!

Nel caso in cui abbiamo scelto di non installare le due librerie (proj.4 e geotiff) al termine del ../configure dovremo ottenere un output di questo tipo:

😡 📀 📀 iz1mhn@radio: ~/src/xastir/build	
File Edit View Terminal Help	
options and external libraries:	4
MINIMUM OPTIONS: ShapeLib (Vector maps) : yes	
RECOMMENDED OPTIONS: GraphicsMagick/ImageMagick (Raster maps) : yes (ImageMagick) pcre (Shapefile customization) : yes dbfawk (Shapefile customization) : yes rtree indexing (Shapefile speedups) : yes map caching (Raster map speedups) : yes internet map retrieval : yes (libcurl)	
FOR THE ADVENTUROUS: AX25 (Linux Kernel I/O Drivers) : yes libproj (USGS Topos & Aerial Photos) : no GeoTiff (USGS Topos & Aerial Photos) : no Festival (Text-to-speech) : yes GDAL/OGR (Obtuse map formats) : no GPSMan/gpsmanshp (GPS downloads) : yes	
xastir will be installed in /usr/local/bin. Type 'make' to build Xastir (Use 'gmake' instead on some systems). iz1mhn@radio:~/src/xastir/build\$]

Se invece abbiamo scelto di installare le due librerie (proj.4 e geotiff) al termine del ../configure Dovremo ottenere un output di questo tipo:

🕺 📀 iz1mhn@radio: ~/src/xastir/build	
File Edit View Terminal Help	
options and external libraries:	
MINIMUM OPTIONS: ShapeLib (Vector maps) : yes	
RECOMMENDED OPTIONS:	
GraphicsMagick/ImageMagick (Raster maps) : yes (ImageMagick) pcre (Shapefile customization) : yes dbfawk (Shapefile customization) : yes rtree indexing (Shapefile speedups) : yes map caching (Raster map speedups) : yes internet map retrieval : yes (libcurl)	
FOR THE ADVENTUROUS: AX25 (Linux Kernel I/O Drivers) : yes libproj (USGS Topos & Aerial Photos) : yes GeoTiff (USGS Topos & Aerial Photos) : yes Festival (Text-to-speech) : yes GDAL/OGR (Obtuse map formats) : no GPSMan/gpsmanshp (GPS downloads) : yes	
xastir will be installed in /usr/local/bin. Type 'make' to build Xastir (Use 'gmake' instead on some systems). iz1mhn@radio:~/src/xastir/build\$	(III) >

NOTA: per l'installazione fatta è corretto il no per le mappe GDAL/OGR.

Terminiamo l'installazione con i due comandi make e sudo make install.

Come ultimo passo andiamo a modificare i permessi dell'eseguibile per consentirne il funzionamento con AX.25 (solo se ci serve: se non ci connettiamo a un TNC possiamo saltare quest'ultimo passo) con il comando sudo chmod u+s /usr/local/bin/xastir.

Ok l'installazione è terminata. Se non abbiamo commesso errori... possiamo iniziare a usare Xastir scrivendo semplicemente in un terminale **xastir**.

NOTA: per sicurezza è meglio fare un "giro" di aggiornamento con un bel **sudo** apt-get update e **sudo** apt-get upgrade, solo per essere sicuri che tutto il sistema sia perfetto!

Rapido riassunto dei comandi

Verifica che il sistema sia aggiornato sudo apt-get update

sudo apt-get upgrade

Installazione dei tool necessari alla compilazione sudo apt-get install build-essential

Installazione delle librerie minime per il funionamento di Xastir sudo apt-get install cvs autoconf automake xorg-dev imagemagick gv libxpdev lesstif2-dev libcurl3-dev

Installazione delle librerie per le funzionalità accessorie

sudo apt-get install gpsman gpsmanshp libpcre3-dev libdb4.8-dev pythondev libax25-dev shapelib libshp-dev festival festival-dev libmagickcoredev

Download di Xastir

Ceck dell'esistenza di un file cvsrc

ls -l ~/.cvsrc

Creazione con l'editor nano del file

nano ~/.cvsrc

Parametri da inserire

cvs -z3

update -P -d

status -v

```
diff -u
```

Chiudere nano e salvare con CRTL + X

Creazione della cartella per il download del codice

mkdir ~/src

cd ~/src

Download del codice con cvs

```
cvs -d:pserver:anonymous@xastir.cvs.sourceforge.net:/cvsroot/xastir login
```

cvs -d:pserver:anonymous@xastir.cvs.sourceforge.net:/cvsroot/xastir co xastir

Creazione del makefile

cd xastir

./bootstrap.sh

Installazione delle librerie grafiche aggiuntive

Medoto UbuntuGIS

sudo -i

```
echo "deb http://ppa.launchpad.net/ubuntugis/ubuntugis-unstable/ubuntu
lucid main" >> /etc/apt/sources.list
```

```
echo "deb-src http://ppa.launchpad.net/ubuntugis/ubuntugis-
unstable/ubuntu lucid main" >> /etc/apt/sources.list
```

```
IZ1MHN
```

```
apt-get update
apt-get install libproj-dev libgeotiff-dev
exit
Metodo dal codice sorgente
Download di proj.4
cd ~/src
wget http://download.osgeo.org/proj/proj-4.7.0.tar.gz
wget http://download.osgeo.org/proj/proj-datumgrid-1.4.tar.gz
Scompattare proj.4
tar zxvf proj-4.7.0.tar.gz
cd proj-4.7.0/nad
tar xzf ../../proj-datumgrid-1.4.tar.gz
cd ..
Compilare e installare proj.4
./configure
make
sudo make install
sudo ldconfig
Download di libgeotiff
cd ~/src
wget http://download.osgeo.org/geotiff/libgeotiff/libgeotiff-1.2.5.tar.gz
Scompattare
tar zxvf libgeotiff-1.2.5.tar.gz
cd libgeotiff-1.2.5/
Compilare e installare libgeotiff
./configure --with-ld-shared="gcc -shared"
make
sudo make install
sudo ldconfig
```

Installazione di Xastir Configurazione del codice cd ~/src/xastir mkdir build cd build Se abbiamo usato il metodo col codice sorgente ../configure Se abbiamo utilizzato il metodo UbuntuGIS ../configure CPPFLAGS="-I/usr/include/geotiff" Generazione del codice e installazione make sudo make install Cambio dei permessi per il funzionaaento con AX.25 sudo chmod u+s /usr/local/bin/xastir

E adesso?

E adesso possiamo tornare all'articolo e iniziare con la configurazione!