

Open Source, Incremental Backup for Windows, Step By Step

Tom Scott

BarCampLondon2, 17/2/07

Tools

- Cygwin, a Linux emulator
 - rsync, a sync/copy tool
- Linux file management commands
 - NTFS formatted drive

Screenshots Only Today

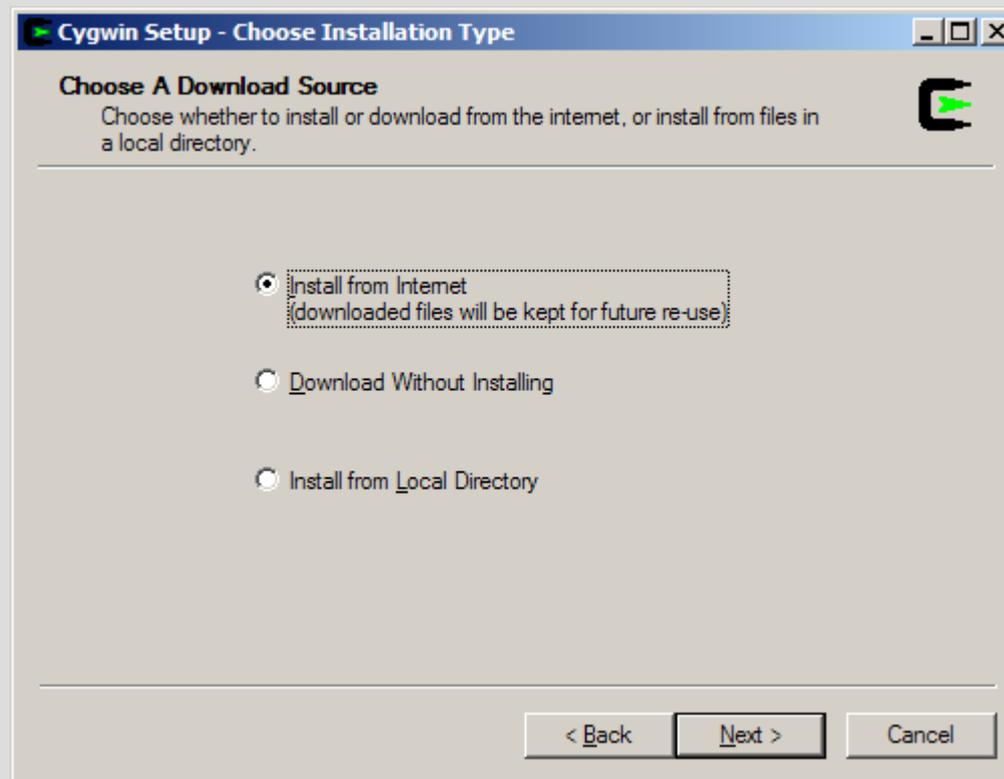
- Backup drives and backpacks do not mix...

Step 1: Install Cygwin

- Cygwin ports Windows tools to Linux
- Basically a Linux emulator for Win32/64
- <http://cygwin.com>

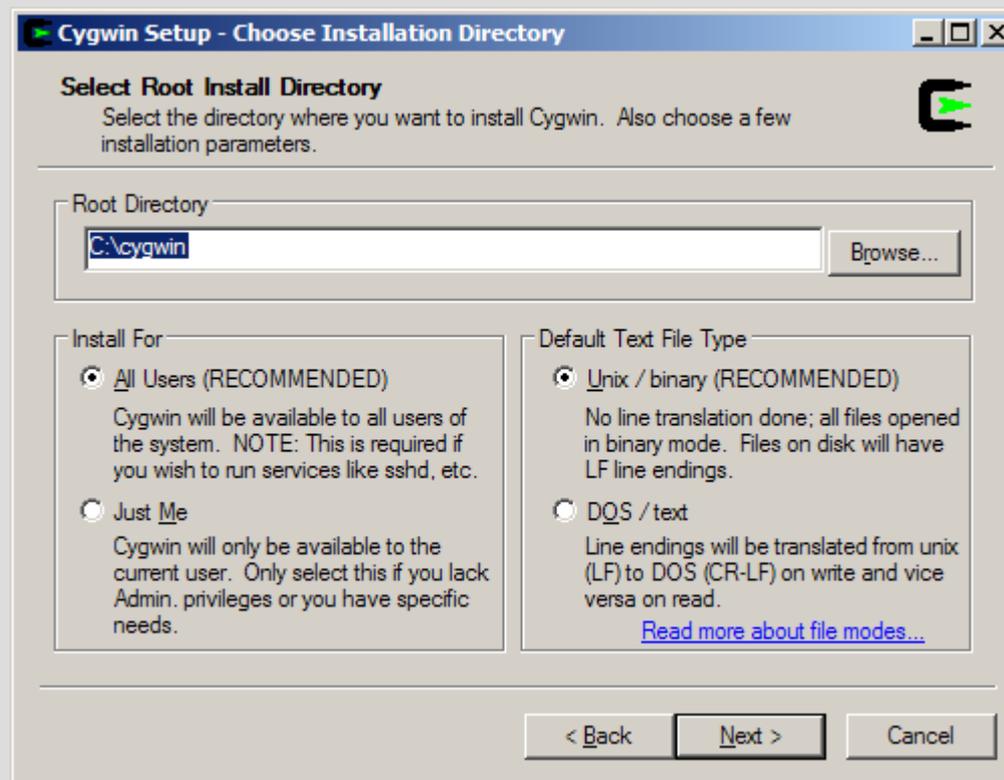
Step 1: Install Cygwin

- Download and run setup.exe
- Pick "Install from Internet"



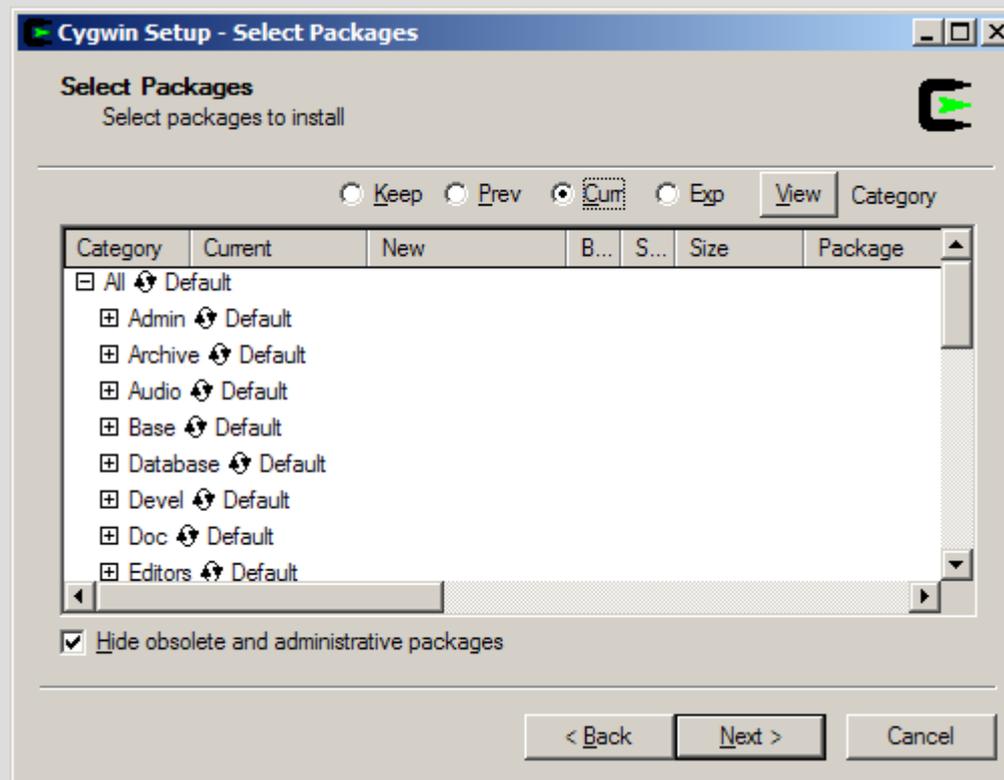
Step 1: Install Cygwin

- Leave options as default
- Pick local mirror (mirror.ac.uk)



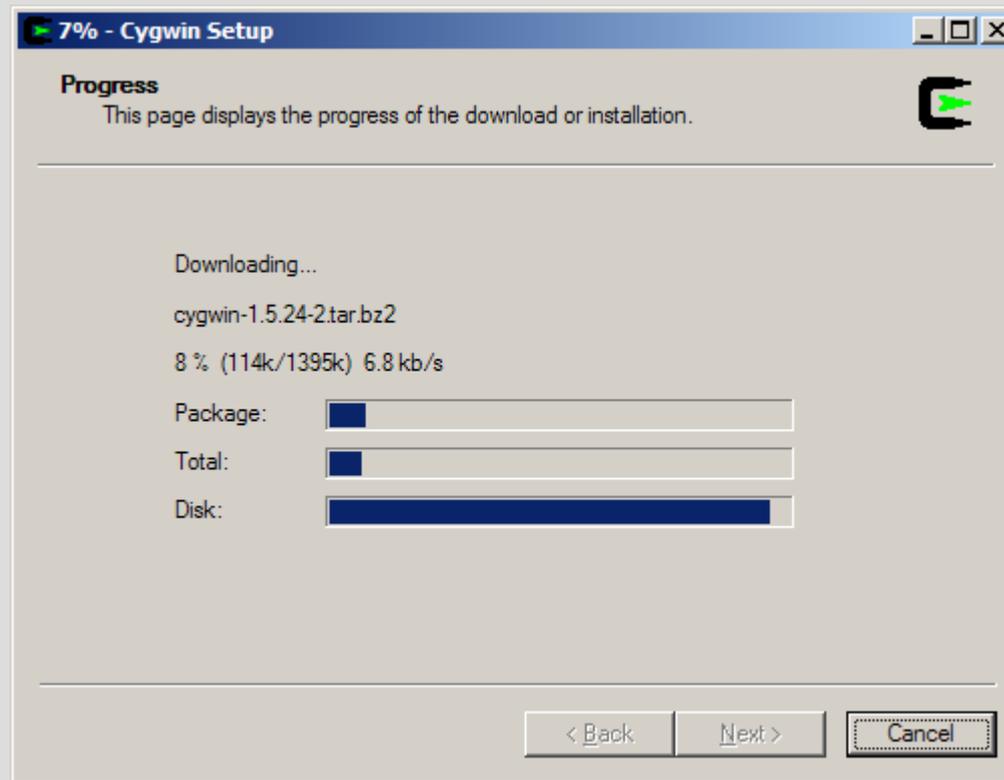
Step 1: Install Cygwin

- After a few 'Next's, choose packages
- Use the default packages except:
- Under "Net", toggle Rsync to 2.6.3-1



Step 1: Install Cygwin

- Wait for the download and install



Step 2: Check External Drive

- I'm assuming you're using an external drive to back up to
- It's possible to use this to back up over a LAN or the net, but it's trickier

Step 2: Check External Drive

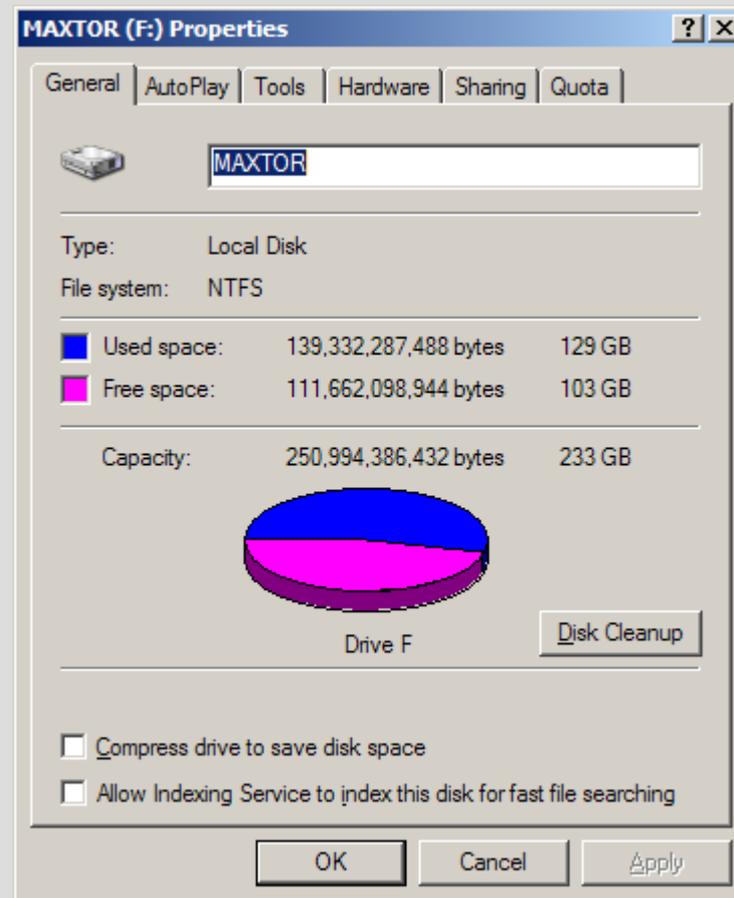
- The external drive needs to be formatted as NTFS, not FAT32
- ...because NTFS understands hardlinks
- even if Windows doesn't

Hardlinks?

- In Windows, and on FAT32, one file has one directory entry
- In Linux, and on NTFS, one file can have many directory entries
- This'll come in handy later

Is your drive NTFS?

- Check 'File System' in its Properties window

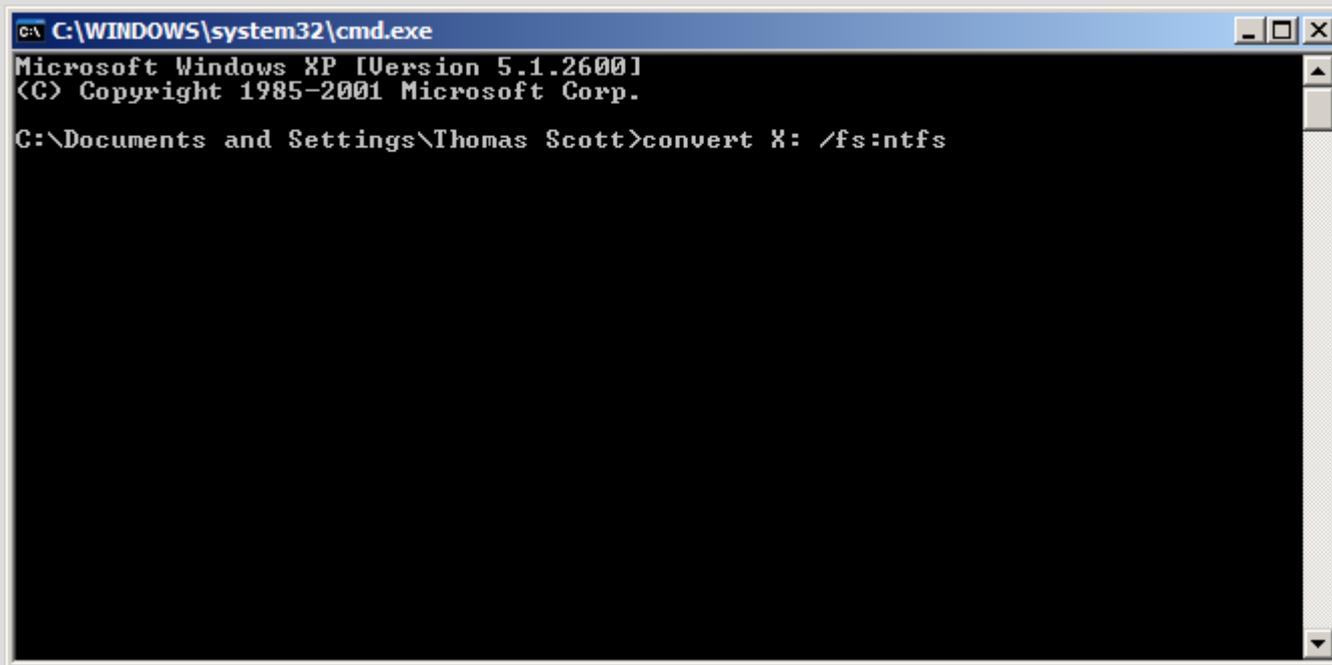


Convert to NTFS

- Non-destructive and low risk
- Not reversible

Convert to NTFS

- Command Prompt (Start > Run > cmd)
- **convert X: /fs:ntfs**
- Follow instructions



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

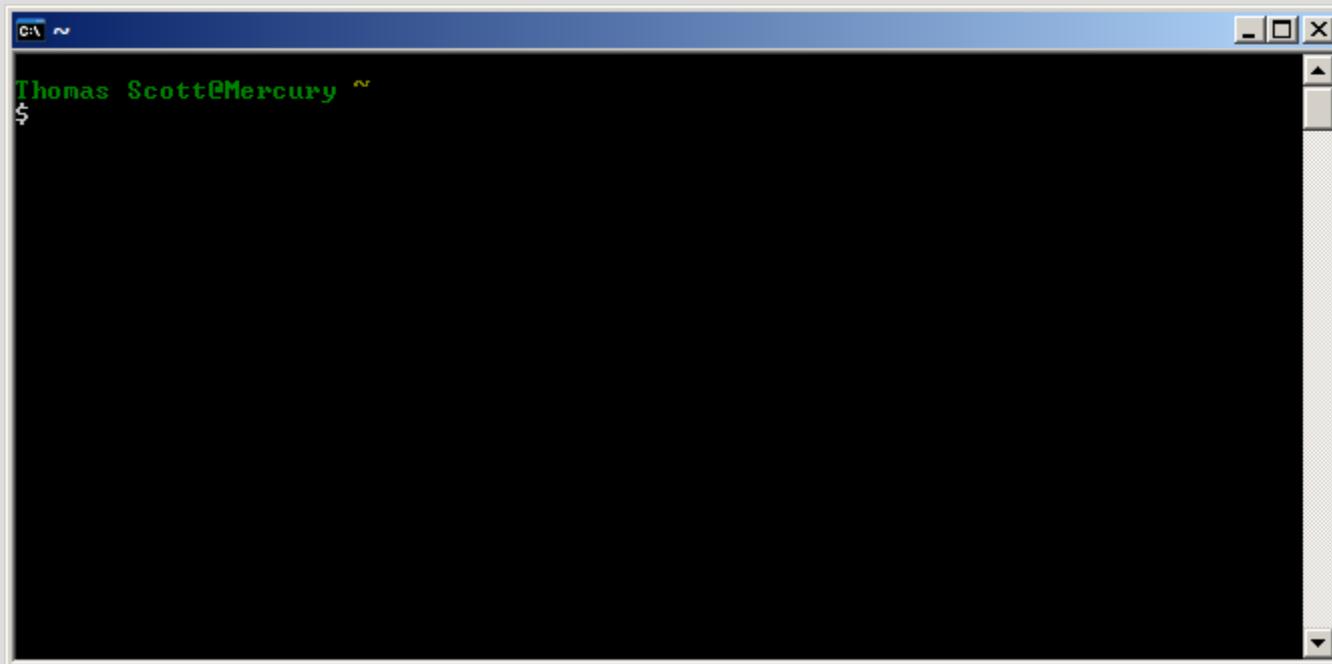
C:\Documents and Settings\Thomas Scott>convert X: /fs:ntfs
```

Quick Recap

- Download and install Cygwin
- Convert your backup drive to NTFS

Step 3: Test Rsync

- Start Cygwin
- Either through the Start Menu or
C:\cygwin\cygwin.bat
- You have a bash shell!



```
C:\ ~
Thomas Scott@Mercury ~
$
```

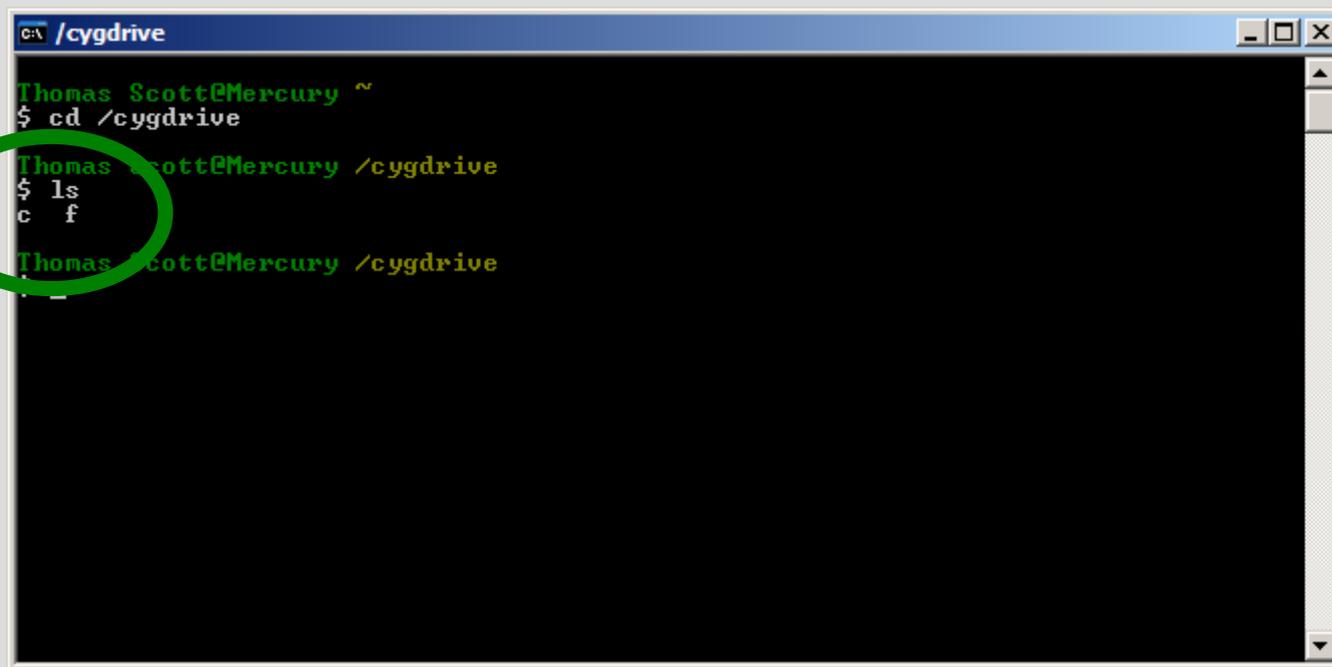
The image shows a screenshot of a Cygwin terminal window. The window title bar indicates the current directory is C:\ and the user is logged in as 'Thomas Scott' on a machine named 'Mercury'. The terminal prompt is '\$', indicating a bash shell is active.

Step 3: Test Rsync

- Don't panic.
- Cygwin gives you access to your Windows drives through a special directory, /cygdrive

Step 3: Test Rsync

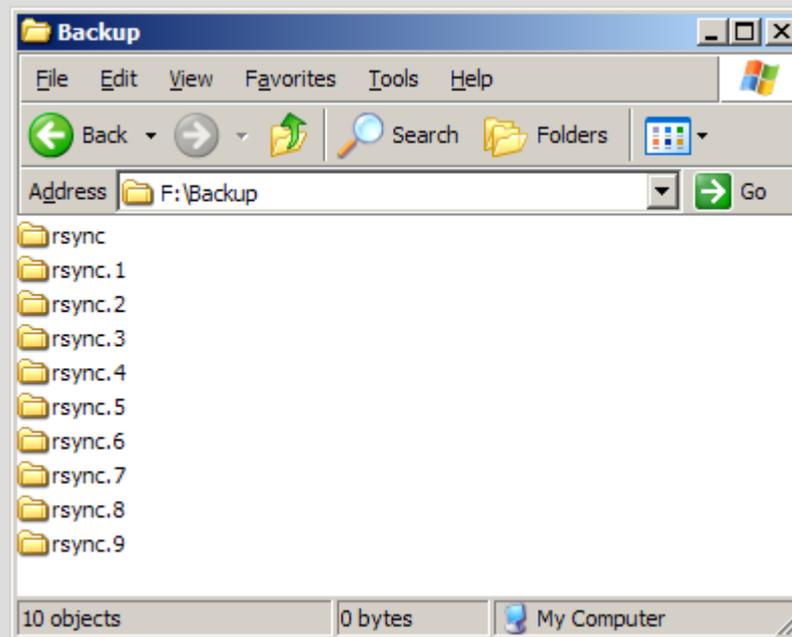
- Let's check that we can see both drives
- `cd /cygdrive`
- `ls`



```
C:\ /cygdrive
Thomas Scott@Mercury ~
$ cd /cygdrive
Thomas Scott@Mercury /cygdrive
$ ls
c f
Thomas Scott@Mercury /cygdrive
```

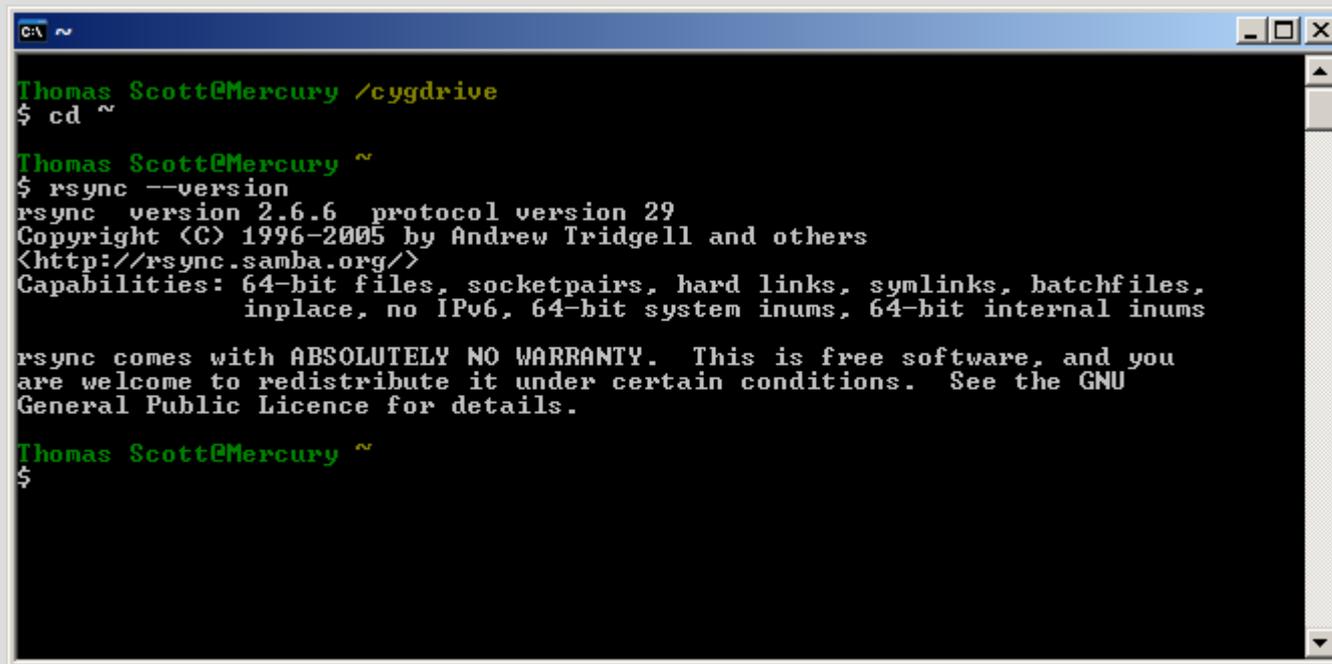
Step 3: Test Rsync

- Let's make our backup folder
- Go to your external drive, create a folder called Backup, and inside that, create folders called rsync and rsync.1 through rsync.9



Step 3: Test Rsync

- Back to your bash shell
- Go back to your "home" directory with `cd ~`
- Test rsync with `rsync --version`

A terminal window with a blue title bar containing the text 'C:\ ~'. The terminal background is black with green text for the prompt and white text for the command and output. The prompt is 'Thomas Scott@Mercury /cygdrive'. The user enters '\$ cd ~'. The prompt changes to 'Thomas Scott@Mercury ~'. The user enters '\$ rsync --version'. The output is: 'rsync version 2.6.6 protocol version 29', 'Copyright (C) 1996-2005 by Andrew Tridgell and others', '<http://rsync.samba.org/>', 'Capabilities: 64-bit files, socketpairs, hard links, symlinks, batchfiles, inplace, no IPv6, 64-bit system inums, 64-bit internal inums', 'rsync comes with ABSOLUTELY NO WARRANTY. This is free software, and you are welcome to redistribute it under certain conditions. See the GNU General Public Licence for details.', and the prompt returns to 'Thomas Scott@Mercury ~' followed by '\$'.

```
C:\ ~
Thomas Scott@Mercury /cygdrive
$ cd ~

Thomas Scott@Mercury ~
$ rsync --version
rsync version 2.6.6 protocol version 29
Copyright (C) 1996-2005 by Andrew Tridgell and others
<http://rsync.samba.org/>
Capabilities: 64-bit files, socketpairs, hard links, symlinks, batchfiles,
               inplace, no IPv6, 64-bit system inums, 64-bit internal inums

rsync comes with ABSOLUTELY NO WARRANTY. This is free software, and you
are welcome to redistribute it under certain conditions. See the GNU
General Public Licence for details.

Thomas Scott@Mercury ~
$
```

Step 4: Your First Backup

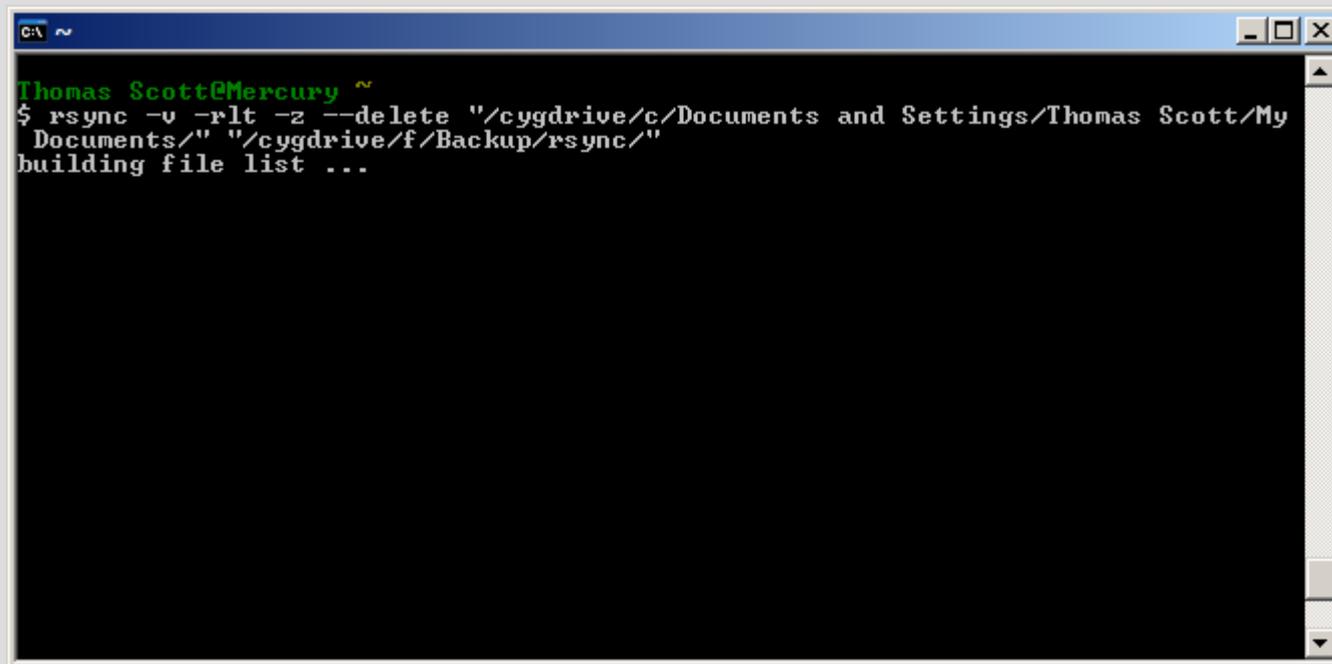
- Here comes the tough bit
- `rsync -v -rlt -z --delete
"/cygdrive/c/Documents and
Settings/[username]/My Documents/"
/cygdrive/x/Backup/rsync/`
- All on one line
- Remember the trailing slashes and quotes
- Case sensitive
- Tab-complete helps!

Wait. What does that mean?

- `rsync -v -rlt -z --delete`
`"/cygdrive/c/Documents and Settings/[username]/My Documents/"`
`/cygdrive/x/Backup/rsync/`
- **rsync**: starts rsync
- `-v` is *verbose*, it tells you what's happening
- `-rlt` is *recursive, keep symlinks, keep times*
- `-z` *compresses* for speed
- `--delete` removes *destination* files that are no longer in the *source*
- the rest specifies where to back up

Don't panic!

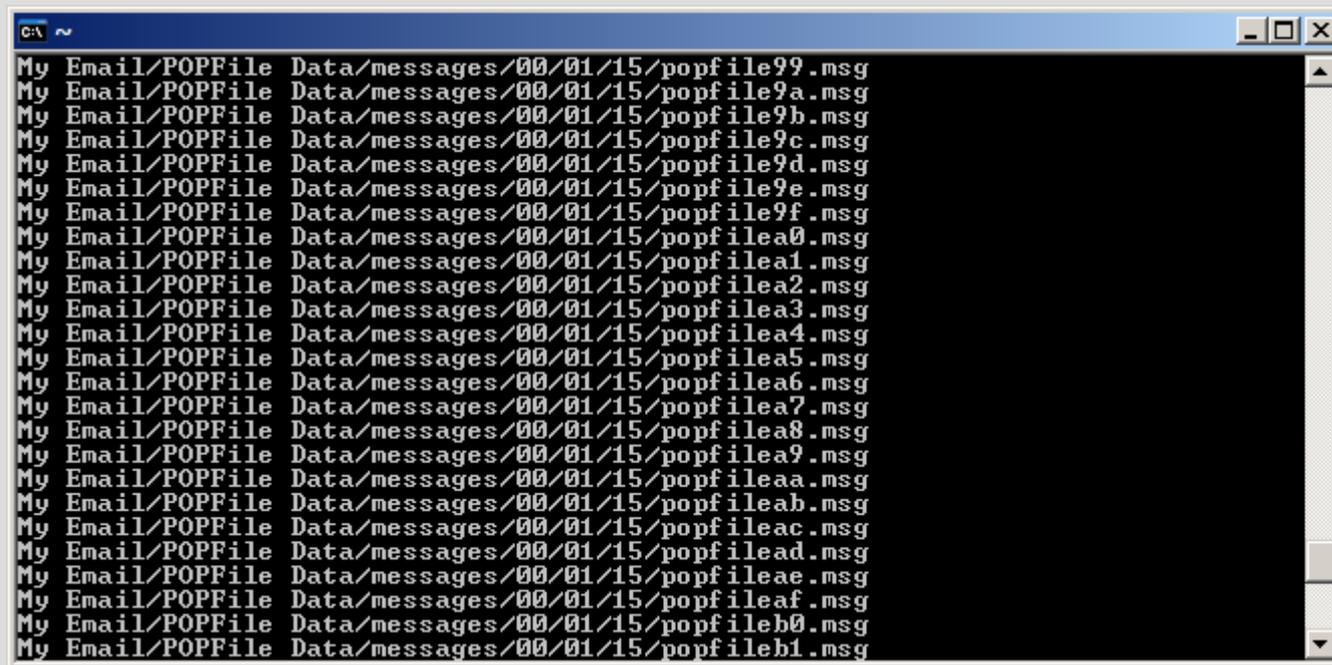
- It will say "building file list"...

A screenshot of a terminal window with a blue title bar. The window title is "C:\ ~". The terminal text shows a user prompt "Thomas Scott@Mercury ~" followed by an rsync command: "\$ rsync -v -rlt -z --delete "/cygdrive/c/Documents and Settings/Thomas Scott/My Documents/" "/cygdrive/f/Backup/rsync/" building file list ...". The text is displayed in a monospaced font on a black background.

```
C:\ ~
Thomas Scott@Mercury ~
$ rsync -v -rlt -z --delete "/cygdrive/c/Documents and Settings/Thomas Scott/My
Documents/" "/cygdrive/f/Backup/rsync/"
building file list ...
```

Don't panic!

- After a few minutes, it'll start backing up.



```
C:\>
My Email/POPFile Data/messages/00/01/15/popfile99.msg
My Email/POPFile Data/messages/00/01/15/popfile9a.msg
My Email/POPFile Data/messages/00/01/15/popfile9b.msg
My Email/POPFile Data/messages/00/01/15/popfile9c.msg
My Email/POPFile Data/messages/00/01/15/popfile9d.msg
My Email/POPFile Data/messages/00/01/15/popfile9e.msg
My Email/POPFile Data/messages/00/01/15/popfile9f.msg
My Email/POPFile Data/messages/00/01/15/popfilea0.msg
My Email/POPFile Data/messages/00/01/15/popfilea1.msg
My Email/POPFile Data/messages/00/01/15/popfilea2.msg
My Email/POPFile Data/messages/00/01/15/popfilea3.msg
My Email/POPFile Data/messages/00/01/15/popfilea4.msg
My Email/POPFile Data/messages/00/01/15/popfilea5.msg
My Email/POPFile Data/messages/00/01/15/popfilea6.msg
My Email/POPFile Data/messages/00/01/15/popfilea7.msg
My Email/POPFile Data/messages/00/01/15/popfilea8.msg
My Email/POPFile Data/messages/00/01/15/popfilea9.msg
My Email/POPFile Data/messages/00/01/15/popfileaa.msg
My Email/POPFile Data/messages/00/01/15/popfileab.msg
My Email/POPFile Data/messages/00/01/15/popfileac.msg
My Email/POPFile Data/messages/00/01/15/popfilead.msg
My Email/POPFile Data/messages/00/01/15/popfileae.msg
My Email/POPFile Data/messages/00/01/15/popfileaf.msg
My Email/POPFile Data/messages/00/01/15/popfileb0.msg
My Email/POPFile Data/messages/00/01/15/popfileb1.msg
```

Step 5: Getting Incremental

- When it's done, it'll tell you how much data it's transferred.
- Right now, that'll be a lot...
- But try running the same command again!
(press the up arrow to repeat commands)

Step 5: Getting Incremental

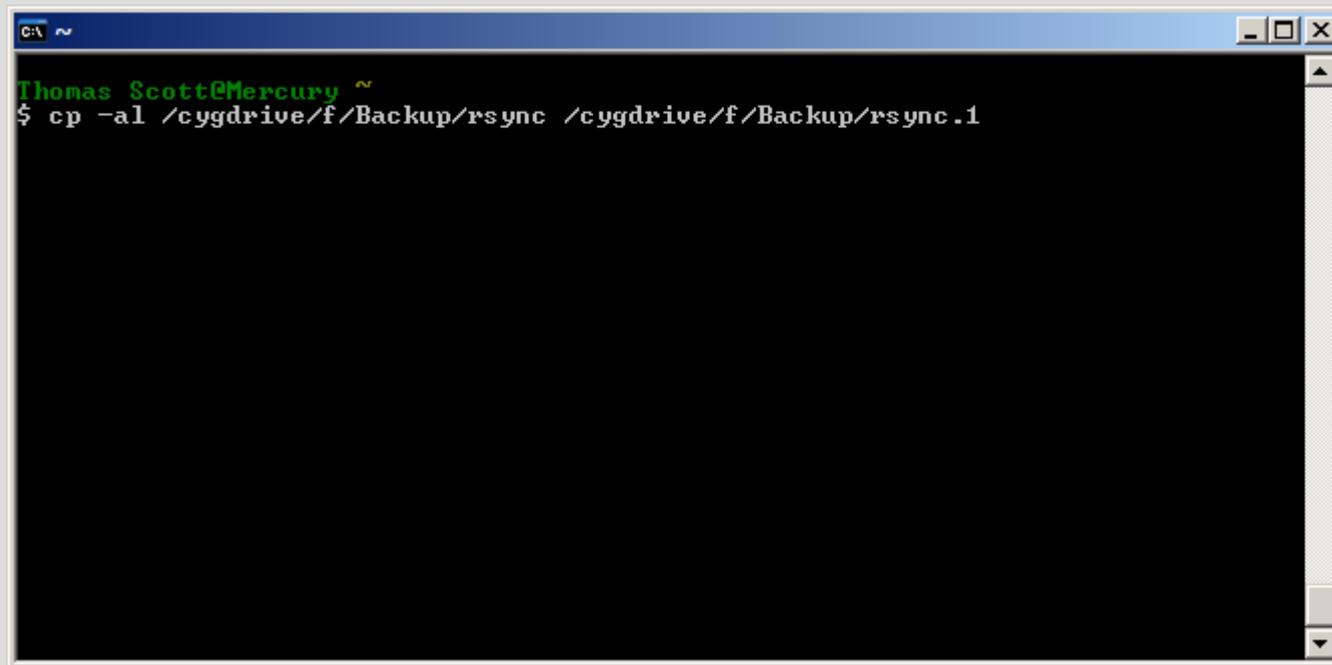
- Rsync only transfers files that have changed, so the second time, transfer is *fast*.
- But that's not an incremental backup...

Step 5: Getting Incremental

- Remember hardlinks?
- `cp` is the bash copy command
- `cp -a1` makes an archive copy of a folder *using hardlinks*.
- So while it looks like there's two copies, the second one is actually filled with *pointers* to the same file on disk.

Step 5: Getting Incremental

- `cp -al /cygdrive/x/Backup/rsync /cygdrive/x/Backup/rsync.1`
- ...and wait

A screenshot of a terminal window with a blue title bar. The title bar contains the text 'C:\ ~' and standard window control buttons (minimize, maximize, close). The terminal content shows a green prompt 'Thomas Scott@Mercury ~' followed by a command '\$ cp -al /cygdrive/f/Backup/rsync /cygdrive/f/Backup/rsync.1'.

```
C:\ ~
Thomas Scott@Mercury ~
$ cp -al /cygdrive/f/Backup/rsync /cygdrive/f/Backup/rsync.1
```

Step 5: Getting Incremental

- So we've got rsync and rsync.1, both filled with the *same* files.
- But here's the cool bit:
rsync unlinks before overwriting.

What?

- If you issue that long rsync command again (press up until it appears), it'll update the "rsync" folder...
- ...but **before** it overwrites a file, it'll remove that folder's link to it
- which means rsync.1 will keep the original version!

So...

- Not only will it only transfer files that have changed
- It'll only require storage space for files that have changed!
- Now we just extend this to the folders rsync.2 through rsync.9
- And you have nine "snapshot" backups, *all* apparently complete, distinct copies!

Step 6: Putting it all together

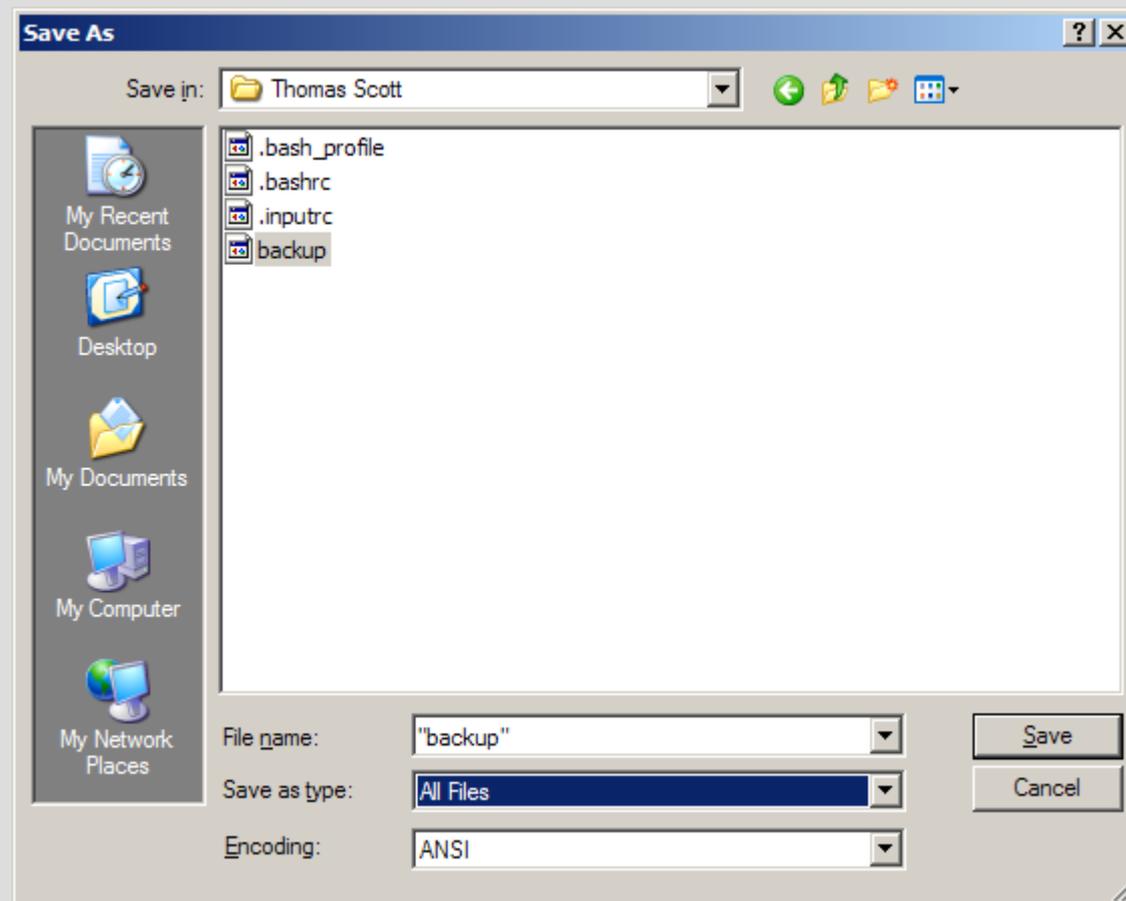
- Open up Notepad and copy and paste in...

```
rm -rf /cygdrive/x/Backup/rsync.9
mv /cygdrive/x/Backup/rsync.8 /cygdrive/x/Backup/rsync.9
mv /cygdrive/x/Backup/rsync.7 /cygdrive/x/Backup/rsync.8
mv /cygdrive/x/Backup/rsync.6 /cygdrive/x/Backup/rsync.7
mv /cygdrive/x/Backup/rsync.5 /cygdrive/x/Backup/rsync.6
mv /cygdrive/x/Backup/rsync.4 /cygdrive/x/Backup/rsync.5
mv /cygdrive/x/Backup/rsync.3 /cygdrive/x/Backup/rsync.4
mv /cygdrive/x/Backup/rsync.2 /cygdrive/x/Backup/rsync.3
mv /cygdrive/x/Backup/rsync.1 /cygdrive/x/Backup/rsync.2
cp -al /cygdrive/x/Backup/rsync /cygdrive/x/Backup/rsync.1
rsync -v -rlt -z --delete "/cygdrive/c/Documents and
  Settings/[username]/My Documents/"
  /cygdrive/x/Backup/rsync/
```

The rsync command should be all on one line; change your drive and username to suit.

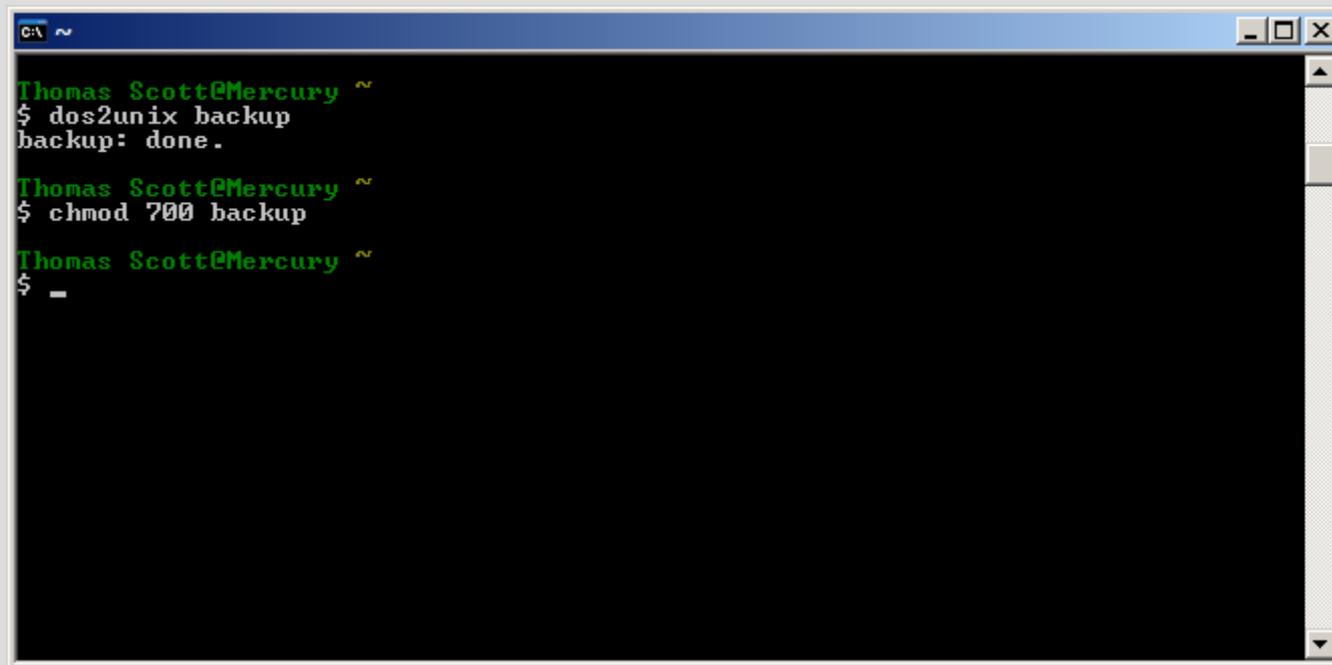
Step 6: Putting it all together

- Save as "C:\cygwin\home\[username]\backup"
- Use "All Files" and enclose backup in quotes



Step 6: Putting it all together

- Two last things to make it usable...
- Go back into your bash shell
- **dos2unix backup**
- **chmod 700 backup**



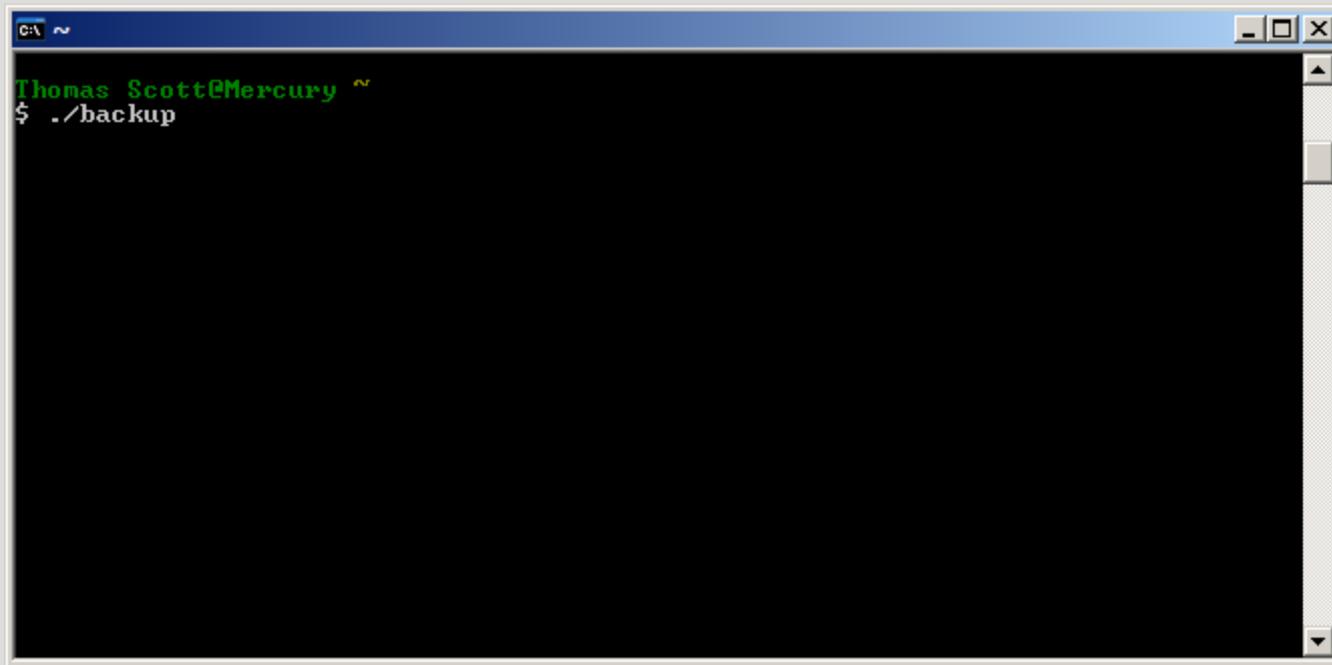
```
C:\ ~
Thomas Scott@Mercury ~
$ dos2unix backup
backup: done.

Thomas Scott@Mercury ~
$ chmod 700 backup

Thomas Scott@Mercury ~
$ _
```

Step 6: Putting it all together

- Then just enter `./backup`
- and wait...



```
C:\ ~
Thomas Scott@Mercury ~
$ ./backup
```

A screenshot of a terminal window. The window title bar shows 'C:\ ~' and standard window controls. The terminal content shows the prompt 'Thomas Scott@Mercury ~' followed by the command '\$./backup' entered on the next line. The rest of the terminal is empty.

Recap

- Install Cygwin with Rsync
- Convert drive to NTFS if needed
- Test rsync and create initial backup
- Create "backup" *shell script*
- Run Cygwin, then `./backup`, to back up
- ...and that's all there is to it

Original Linux Script Source:

- *Easy Automated Snapshot-Style Backups with Rsync*
Mike Rubel
http://www.mikerubel.org/computers/rsync_snapshots/

Download this presentation

- PDF of the presentation
- Rsync command
- and That Shell Script
- <http://www.thomasscott.net/barcamp2/>