

15,000 Packages Can't Be Wrong!

Systems Administration Using Debian Packages

Andrew McMillan

andrew@catalyst.net.nz

Once upon a time...

Once upon a time...

- I had accounts on about a dozen systems.



Once upon a time...

- I had accounts on about a dozen systems.
- I needed to update my `.bashrc`



Once upon a time...

- I had accounts on about a dozen systems.
- I needed to update my `.bashrc`

```
for C in santayana socrates  
plato boole leibniz  
wittgenstien popper mill; do  
scp this.bashrc $C:.bashrc;  
done<Enter>
```



Granular physics: A bridge to sandpile stability
Peter Schiffer

Some Simple Problems

Some Simple Problems

His name was “Wittgenstein”.



Some Simple Problems

His name was “Wittgenstein”.

Two slides back there were a dozen systems, and I've already forgotten five of them!



Deeper Problems

- It doesn't scale.

Deeper Problems

- It doesn't scale.
- It doesn't cope nicely with more complexly firewalled systems.

Deeper Problems

- It doesn't scale.
- It doesn't cope nicely with more complexly firewalled systems.
- So I wrote another shell script!

Deeper Problems

- It doesn't scale.
- It doesn't cope nicely with more complexly firewalled systems.
- So I wrote another shell script!

Clearly, this was getting silly.

Thinks!

- The common factor here was that these were all Debian systems.
- And I was a Debian developer.

Thinks!

- The common factor here was that these were all Debian systems.
- And I was a Debian developer.
- Or should those be the other way around?

Other Thinks!

- There are other, more traditional approaches:
 - LDAP authentication
 - VPN access
 - FAI
 - cfengine

Other Thinks!

- There are other, more traditional approaches:
 - LDAP authentication
 - VPN access
 - FAI
 - cfengine
 - \$\$\$\$\$\$

Other Thinks!

- There are other, more traditional approaches:
 - LDAP authentication
 - VPN access
 - FAI
 - cfengine
 - \$\$\$\$\$\$
- These tend to be centrally organised, which wouldn't work well for our environment.

Problems with Central Control

- Client Agreement
 - Clients who are paying for security don't want their networks connected to other clients.
 - Different clients, including on their LAN and/or on the internet.

Problems with Central Control

- Client Agreement
 - Clients who are paying for security don't want their networks connected to other clients.
 - Different clients, including on their LAN and/or on the internet.
- Nor Should They!

Problems with Central Control

- Routing Difficulties
 - Machines on disparate, heavily firewalled and segmented networks.
 - Different geographical locations.
 - VPNs don't help

Problems with Central Control

- Routing Difficulties
 - Machines on disparate, heavily firewalled and segmented networks.
 - Different geographical locations.
 - VPNs don't help
- Multiplying Complexity

Problems with Central Control

- Reliability
 - Need access when everything else is fucked
 - Or how can we fix those problems?

Problems with Central Control

- Reliability
 - Need access when everything else is fucked
 - Or how can we fix those problems?
- Basic Access

Simplification

Simplification

- Is the problem the standard?

Simplification

- Is the problem the standard?
- Or the way the standard scales?

Simplification

Simplification

- Can we make the standard scale better?
 - Use basic functionality
 - Lowest common denominator
 - Use the standard better...

Simplification

- Can we make the standard scale better?
 - Use basic functionality
 - Lowest common denominator
 - Use it better
- Management is built into the Operating System

Simplification

- Can we make the standard scale better?
 - Use basic functionality
 - Lowest common denominator
 - Use it better
- Management is built into the Operating System

Simplification

Thousands of people, all over the world,
manage to maintain simple stuff like this
in their spare time!

Simplification

Thousands of people, all over the world,
manage to maintain simple stuff like this
in their spare time!

For FUN!!!

Using the Packaging System

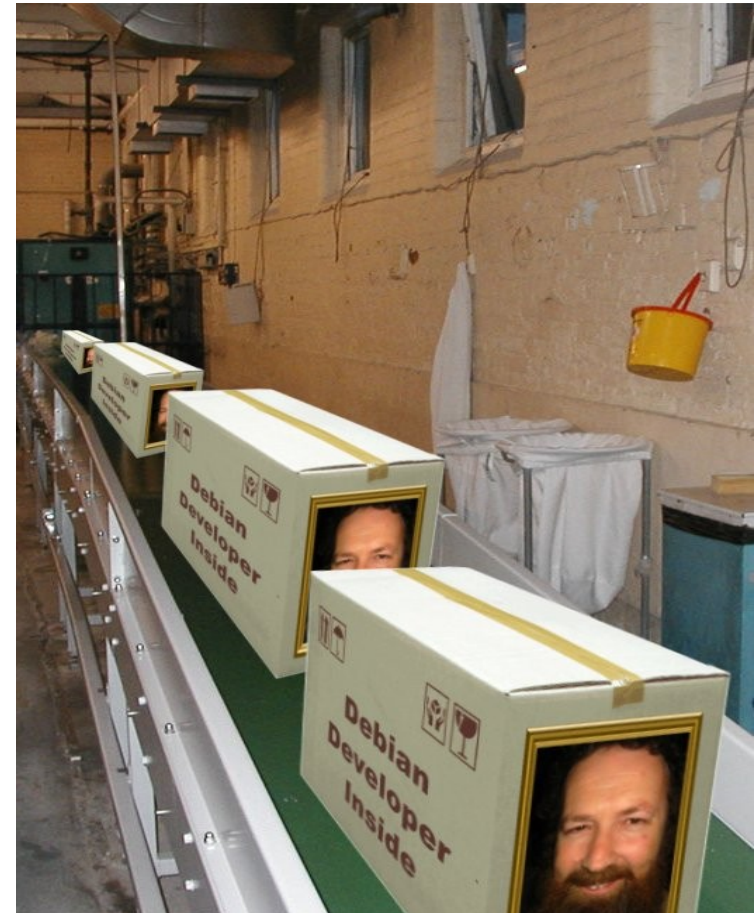
Uses for Debian Packages

- User Management
 - Create account, disable account, remove account



Uses for Debian Packages

- User Management
 - Create account, disable account, remove account
- Group Management
 - Sudo access, group membership



More uses for Debian packages

- Standardisation
 - We do it this way, here, matey..., Metapackages depend / conflict
 - Apache logfile analysis.
- Configuration Management
 - Hostinfo database

And, hang on, we could...

- Application Management
 - NZ Electoral Roll
 - tab.co.nz
 - mahara.org
 - .nz registry

Catalyst User Packages

- Install account skeleton, including:
 - .bashrc, .bash_profile, .ssh/authorized_keys, ...
- Create account on 'install' (postinst)
 - Generate random password, encrypt to owner using GPG public key, optionally e-mail to them.
- Disable account on 'delete'
- Remove account on 'purge'

User Package Structure

- Debian Package
 - “ar” archive with three files:
 - debian-binary
 - control.tar.gz
 - data.tar.gz
- Approach also applies to RPM, ebuild or other packaging formats (e.g. Solaris)
- Most useful in conjunction with the repository management on Linux systems

User Package Control

- Debian control files
 - ./conffiles – list of configuration files
 - ./control – Debian control information

User Package Control

- Debian control files
 - ./conffiles – list of configuration files
 - ./control – Debian control information
- Configuration Information
 - ./config – configuration script
 - ./templates – configuration questions

User Package Control

- Debian control files
 - ./conffiles – list of configuration files
 - ./control – Debian control information
- Configuration Information
 - ./config – configuration script
 - ./templates – configuration questions
- Scripts
 - ./postinst – post installation script
 - ./prerm – pre removal script
 - ./postrm – post removal script

User Package Data

- /etc/sysadmins/catalyst-andrewmcmillan
 - preferred_fullname
 - preferred_gid
 - preferred_names
 - preferred_uid
 - notification_email
 - notifying.gpg
 - skel/.bash_profile
 - skel/.bash_logout
 - skel/.bashrc
 - skel/.inputrc
 - skel/.vimrc
 - skel/.ssh

User Package: postinst

If pre-existing installation

if cfg_update_from_skel

update from skel

else

if (can find existing username && cfg_overwrite_user)

take over existing user

update from skel

else

make new user from skel

save installation details (uid,gid,username)

notify

User Package: prerm

If “remove” && validly installed user

```
passwd -l "${USERNAME}" || true
```

```
chown root:root /home/${USERNAME} || true
```

```
chmod 700 /home/${USERNAME} || true
```

User Package: prerm

If “remove” && validly installed user

```
passwd -l "${USERNAME}" || true
```

```
chown root:root /home/${USERNAME} || true
```

```
chmod 700 /home/${USERNAME} || true
```

- Could probably be more bulletproof

User Package: prerm

If “remove” && validly installed user

```
passwd -l "${USERNAME}" || true
```

```
chown root:root /home/${USERNAME} || true
```

```
chmod 700 /home/${USERNAME} || true
```

- Could probably be more bulletproof
- “passwd -l” (lock account) may be GNU specific(?)

User Package: prerm

If “remove” && validly installed user

```
passwd -l "${USERNAME}" || true
```

```
chown root:root /home/${USERNAME} || true
```

```
chmod 700 /home/${USERNAME} || true
```

- Could probably be more bulletproof
- “passwd -l” (lock account) may be GNU specific(?)
- Not necessarily getting user's home directory correct

User Package: postrm

If “purge”

if cfg_remove_on_purge

remove user home directory

remove user account

User Package: postrm

If “purge”

if cfg_remove_on_purge

remove user home directory

remove user account

- Could probably be more bulletproof too...
- Also not necessarily getting user's home directory correct

Security Considerations

- You have to trust the person making the package.

Security Considerations

- You have to trust the person making the package.
- And everyone with control over what goes in there.
 - Separation of responsibilities.
 - Auditing.
 - Clients have to have that level of trust in us already.

Security Considerations

- You have to trust the person making the package.
- And everyone with control over what goes in there.
 - Separation of responsibilities.
 - Auditing.
 - Clients have to have that level of trust in us already.
- And everyone who develops for Debian!

Hostinfo System

- hostinfo-client package
 - Daily e-mail of configuration information

Hostinfo System

- hostinfo-client package
 - Daily e-mail of configuration information
 - Installed on every machine
- Centralised host database
 - Searching
 - Links
 - Quick Reference
 - Particularly useful for package management questions

Hostinfo Client Script

- Gets information from system
 - Installed packages
 - Hardware: memory, disk
 - OS type and version
 - Disk Partitioning
 - Users & Groups
 - Debian, Red Hat, OpenBSD and Solaris
 - Only packaged for Debian

hostinfo-client Package

- Can be depended on
- Some sanity checks during installation
- Runs hostinfo client script daily
- Sends to central database
- Staggered cronjob
 - We don't want to hit our mailserver all at once

Thank You

Further Reading:

The Debian Packaging Manual