

# User management--

a generalized management script example

David Morgan

© David Morgan 2005

## Adding users – actions/mechanics

- add record to /etc/passwd
- add record to /etc/shadow
- add record to /etc/group for user's default group
- add user to pre-existing groups
- create user home directory /home/<username>
- copy default startup files to home directory
- set permissions on new files and directories
- set ownership on new files and directories
- set system password
- set other passwords (e.g., Samba)
- customize user info with, e.g., usermod or chage
- setup mail home/aliases
- set disk quotas

© David Morgan 2005

## Process of adding users

- surprisingly extensive, isn't it!?
- varies among sites (local policies differ)
- no utility does it all
  - some do it partially (useradd, passwd)

© David Morgan 2005

## Steps performed by useradd

- ✓ ● add record to /etc/passwd
- ✓ ● add record to /etc/shadow
- ✓ ● add record to /etc/group for user's default group
  - add user to pre-existing groups
- ✓ ● create user home directory /home/<username>
- ✓ ● copy default startup files to home directory
- ✓ ● set permissions on new files and directories
- ✓ ● set ownership on new files and directories
  - set system password
  - set other passwords (e.g., Samba)
  - customize user info with, e.g., usermod or chage
  - setup mail home/aliases
  - set disk quotas

© David Morgan 2005

## Steps performed by passwd

- add record to /etc/passwd
- add record to /etc/shadow
- add record to /etc/group for user's default group
- add user to pre-existing groups
- create user home directory /home/<username>
- copy default startup files to home directory
- set permissions on new files and directories
- set ownership on new files and directories
- ✓ ● set system password
- set other passwords (e.g., Samba)
- customize user info with, e.g., usermod or chage
- setup mail home/aliases
- set disk quotas

© David Morgan 2005

## A common approach -- adding users in 2 steps

- run useradd
- then set password with passwd

© David Morgan 2005

## Other approaches

- manual - perform individual steps separately
- hybrid - some with utilities, others manually
- automated - all by script(s) you write

“Although the `useradd` and `userdel` commands are convenient, they are usually not sufficient to implement all of a site’s local policies. Don’t hesitate to write your own `adduser` and `rmuser` scripts; most larger sites do. ... You homebrew scripts can call the standard utilities to accomplish part of their work.”

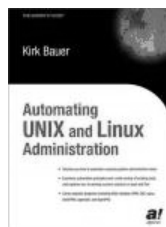
Linux Administration Handbook Nemeth, Snyder, and Hein

© David Morgan 2005

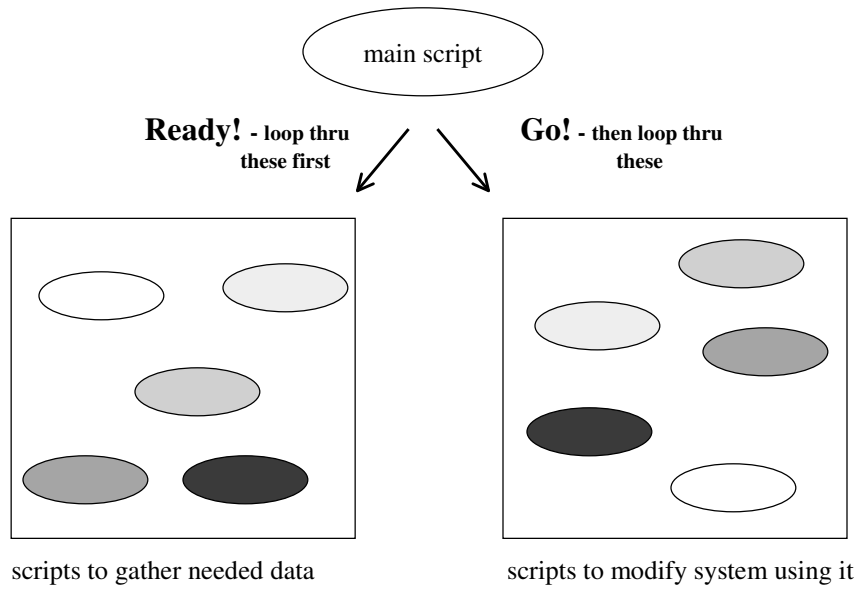
## Where credit is due

Following approach and scripts are from:

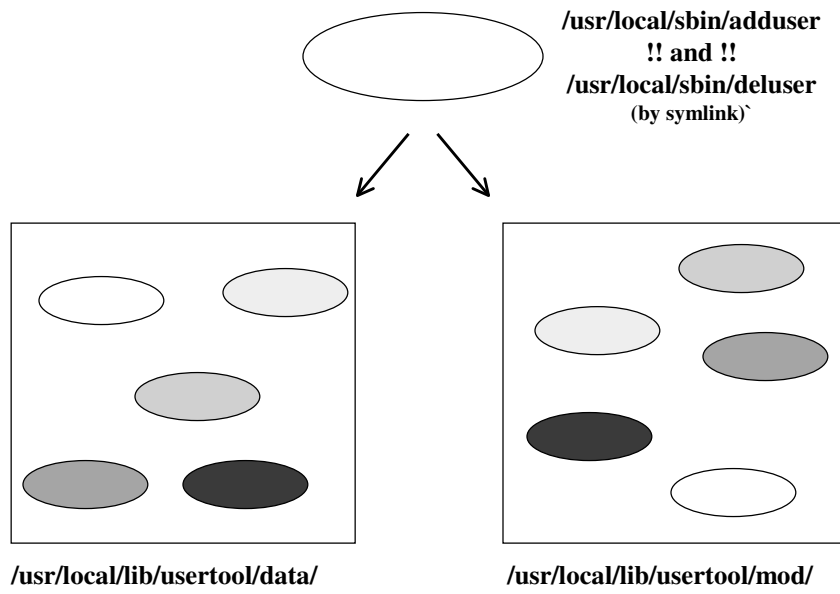
Automating Unix and Linux Administration,  
Kirk Bauer, Apress, 2003



# Architecture



# Default names/locations

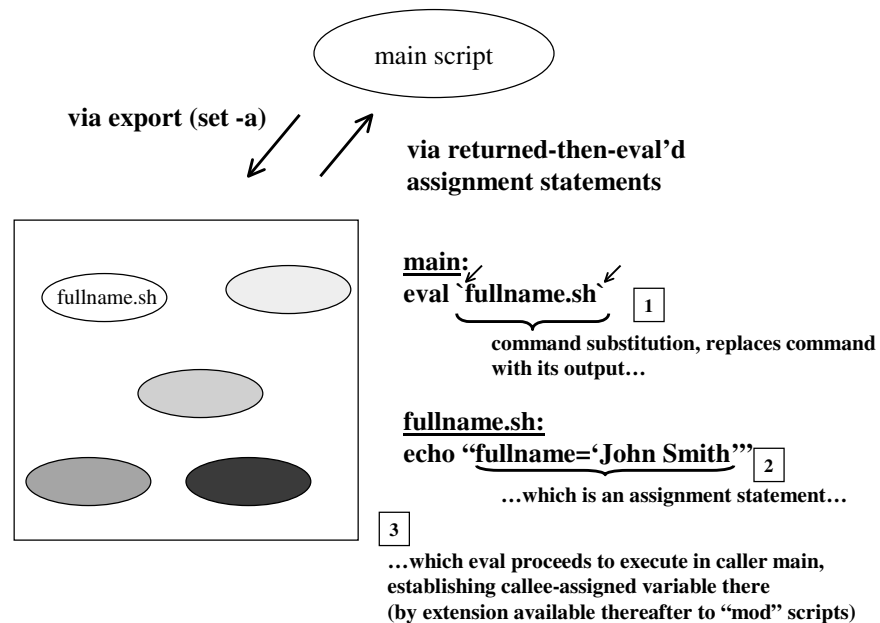


# Features of note

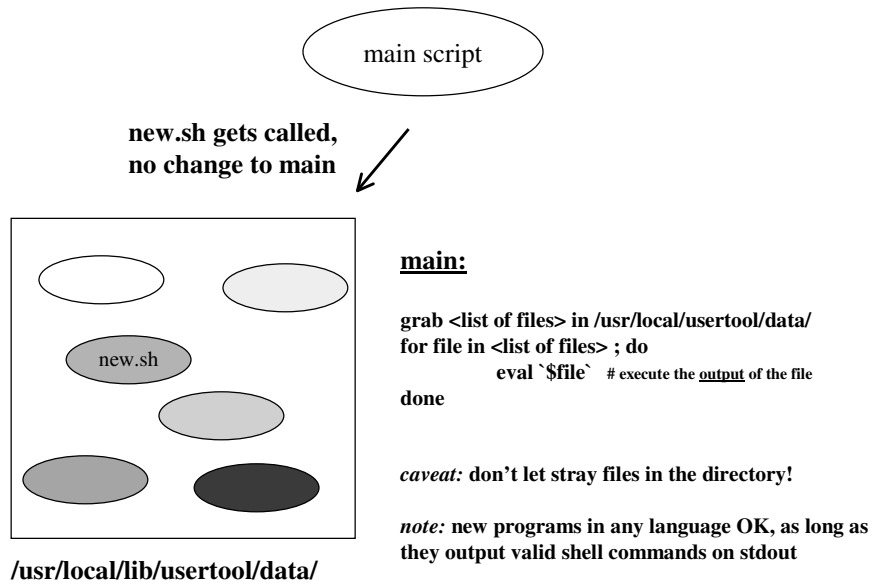
- 2-way caller < > callee variable communication
  - shells can't pass variables back to callers, normally
- extensibility - loops capture any/all scripts provided
  - no code changes, mere placement plugs new callees in

© David Morgan 2005

# Communication of variables



# Auto-extensible by drop-in



## Example data flows

fullname.sh → basic.sh → /etc/passwd  
shell.sh

extragroups.sh → extragroups.sh → /etc/group

homedir.sh → homedir.sh → /home or ...

e.g., fullname.sh and shell.sh in the data directory solicit from user his full name and choice of shell, pass results to basic.sh in the mod directory, which from that and other info composes a standard user record and inscribes it into the "official user database" /etc/passwd

Typical usage screenshot

```

root@EMACH1:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help

[root@EMACH1 ~]# /usr/local/sbin/adduser lew
Enter mail alias [leave blank when done]: ← from aliases.sh

Select one or more extra groups:
1) dev
2) cvs
3) web
Enter group ('q' to quit): q ← from extragroups.sh

Enter full name for lew: Lew Smith ← from fullname.sh

Allocated UID/GID: 500 ← from getuid.pl ← from homedir.sh

1) /home (12029 MB avail) 3) server2:/export/home (?? MB avail)
2) /export/home (?? MB avail) 4) server3:/export/home (?? MB avail)
#? 1
Select a shell: ← from shell.sh
1) /bin/bash 3) /bin/ksh 5) /bin/tcsh 7) /sbin/nologin
2) /bin/csh 4) /bin/sh 6) /bin/zsh
#? 1

Adding mail alias for lew: Lew.Smith ← from aliases.sh
Creating entry in /etc/passwd... ← from basic.sh
Creating entry in /etc/group... }
Creating entry in /etc/shadow... } ← from extragroups.sh
Adding lew to group public ← from extragroups.sh

Setting account password... ← from zpasswd.sh & embedded /etc/passwd
Changing password for user lew.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.

Action adduser for lew has been completed ← sign-off from main script adduser.sh

[root@EMACH1 ~]#

```

“data” scripts’ output  
(in /usr/local/lib/usertool/data/)

“mod” scripts’ output  
(in /usr/local/lib/usertool/mod/)

## deluser alternative functionality

- “adduser” callable by alternative name “deluser”
- it checks by which name it was called
- undoes most (not all) of its “adduser” actions when called as “deluser” instead
- undoing remainder can be implemented as a custom add-on script you supply

# Custom add-ons

- write additional programs in any language
- “implementation-by-placement”
  - data-gatherers
    - must print valid shell commands to stdout, any screen messages to stderr
    - drop into /usr/local/lib/usertool/data/
  - system-modifiers
    - no output restrictions
    - drop into /usr/local/lib/usertool/mod/
  - auto-called on next run
  - avoid stray files in script directories
- execution order is alphabetical within directory, name accordingly

© David Morgan 2005