MindWorks UK

TTP
TECHNOLOGY
TRANSFER
PARTNERS
Novell

Making directories reappear elsewhere (file system worm holes)

Joe R. Doupnik

MindworksUK and Univ of Oxford

jrd@netlab1.oucs.ox.ac.uk

Mind Works IT Training & Event Specialists

JRD Networking Research Institute

What we are discussing

How to make subdirectories, or just individual files, appear at other places in the user level file tree -

- **without copying**
- **without symbolic links or hard links**
- **without CIFS Junctions**
- **without Shadow Volumes/DST**
- **without revealing names of superior directories just to get to the interesting area**
- **without complexity or side effects**
- **without system loading**

MindWorks UK

What we are discussing

For both NSS and POSIX file systems (source and destination)

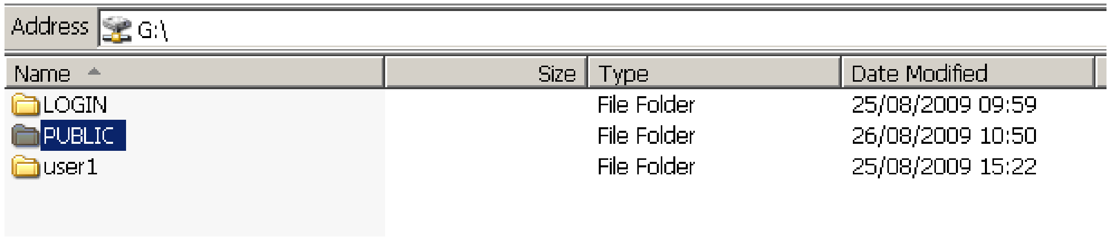
For all methods of access, users and applications alike

By the way, it is free, simple and already present

MindWorks UK

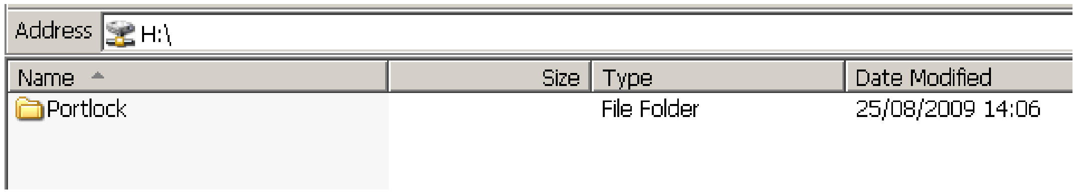
Our starting point, normal things

Volume SYS: (NCP/POSIX), starting view, user1's home dir is present


Address G:\

Name	Size	Type	Date Modified
LOGIN		File Folder	25/08/2009 09:59
PUBLIC		File Folder	26/08/2009 10:50
user1		File Folder	25/08/2009 15:22

Volume NSSVOL: (NSS), user1 has access to \Portlock\NWDSK\DOS but not to files above DOS

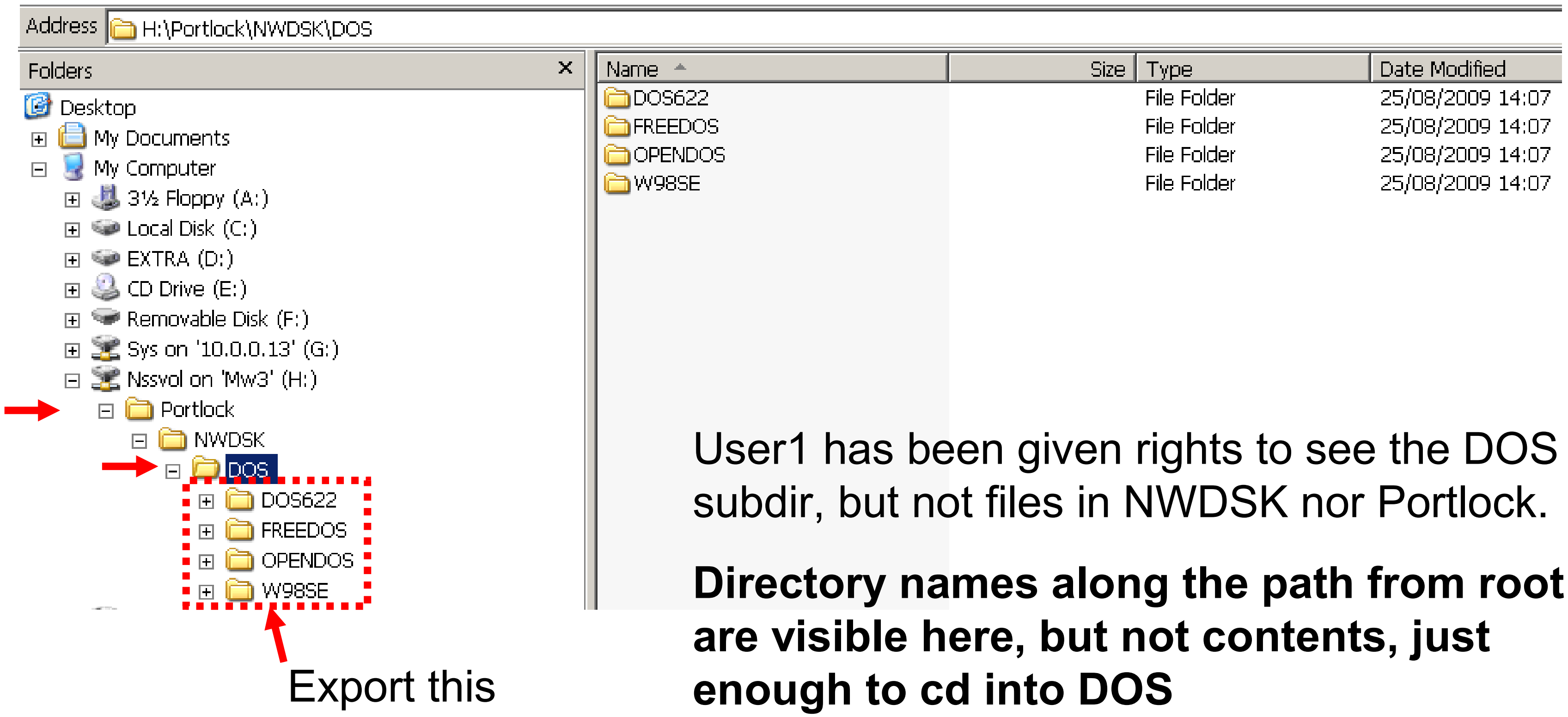

Address H:\

Name	Size	Type	Date Modified
Portlock		File Folder	25/08/2009 14:06

MindWorks UK

User1 looks at an NSS volume

Choose subdir **DOS** of **NSSVOL:\Portlock\NWDISK\DOS** as source



Address: H:\Portlock\NWDSK\DOS

Name	Size	Type	Date Modified
DOS622		File Folder	25/08/2009 14:07
FREEDOS		File Folder	25/08/2009 14:07
OPENDOS		File Folder	25/08/2009 14:07
W98SE		File Folder	25/08/2009 14:07

Export this

User1 has been given rights to see the DOS subdir, but not files in NWDSK nor Portlock.

Directory names along the path from root are visible here, but not contents, just enough to cd into DOS

The plan: create an illusion

Make contents of “DOS” appear in two locations on volume SYS: (which happens to be an XFS file system in my case). Two places are just for fun.

Destinations are

subdir “buried” under SYS:PUBLIC

subdir “toplevel” at SYS: root level

I manually create both directories (script shown next), then do *mount -o bind* to duplicate

MindWorks UK

How we do it

We can remount part of the file hierarchy somewhere else

```
mount --bind  olddir  newdir
```

or

```
mount -o bind olddir  newdir
```

Afterward, the same content is accessible in **two places**

(paraphrased from the *mount* man page)

©MindworksUK 2009

How we do it: magic is explained

Mount **man page excerpt --**

Since Linux 2.4.0 it is possible to remount part of the file hierarchy somewhere else.
The call is

```
    mount --bind olddir newdir
```

After this call the same content is accessible in **two places**. One can also remount a **single file** (on a single file).
This call attaches only (part of) a single file system, not possible submounts.

The entire file hierarchy <u>including submounts</u> is attached to a second place using

```
    mount --rbind olddir newdir
```

Note that the file system <u>mount options will remain the same</u> as those on the original mount point, and <u>cannot be changed</u> by passing the -o option along with --bind/--rbind.

mount -o option source destination   is better syntax

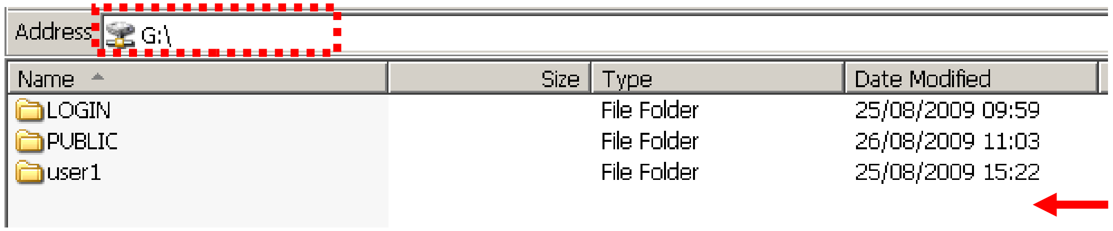
Experimental setup

```
#!/bin/sh

mkdir /usr/novell/sys/toplevel
mkdir /usr/novell/sys/PUBLIC/buried


# syntax: mount -o options  source  destination

mount -o bind /home/NSSVOL/Portlock/NWDSK/DOS \
   /usr/novell/sys/toplevel

mount -o bind /home/NSSVOL/Portlock/NWDSK/DOS \
   /usr/novell/sys/PUBLIC/buried
```

NSSVOL is, naturally, an NSS volume, /usr/novell/sys is POSIX

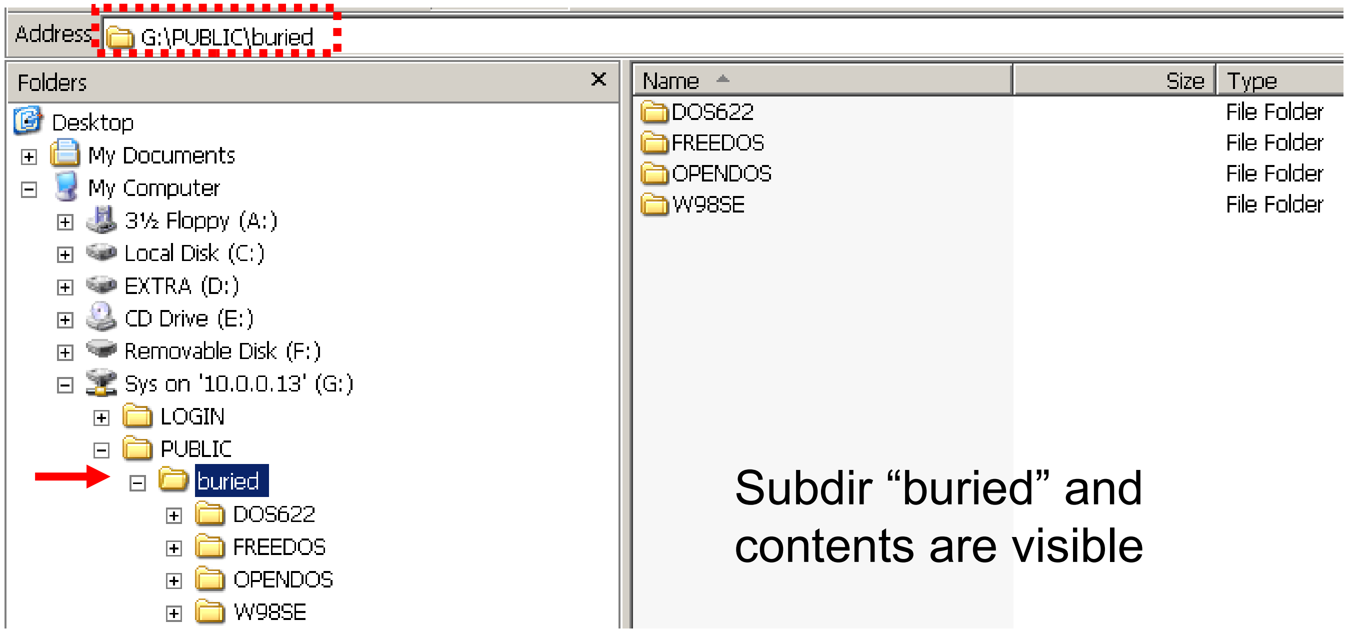
MindWorks UK

User1's view of these mounts


Address G:\

Name	Size	Type	Date Modified
LOGIN		File Folder	25/08/2009 09:59
PUBLIC		File Folder	26/08/2009 11:03
user1		File Folder	25/08/2009 15:22

"toplevel" is not visible!

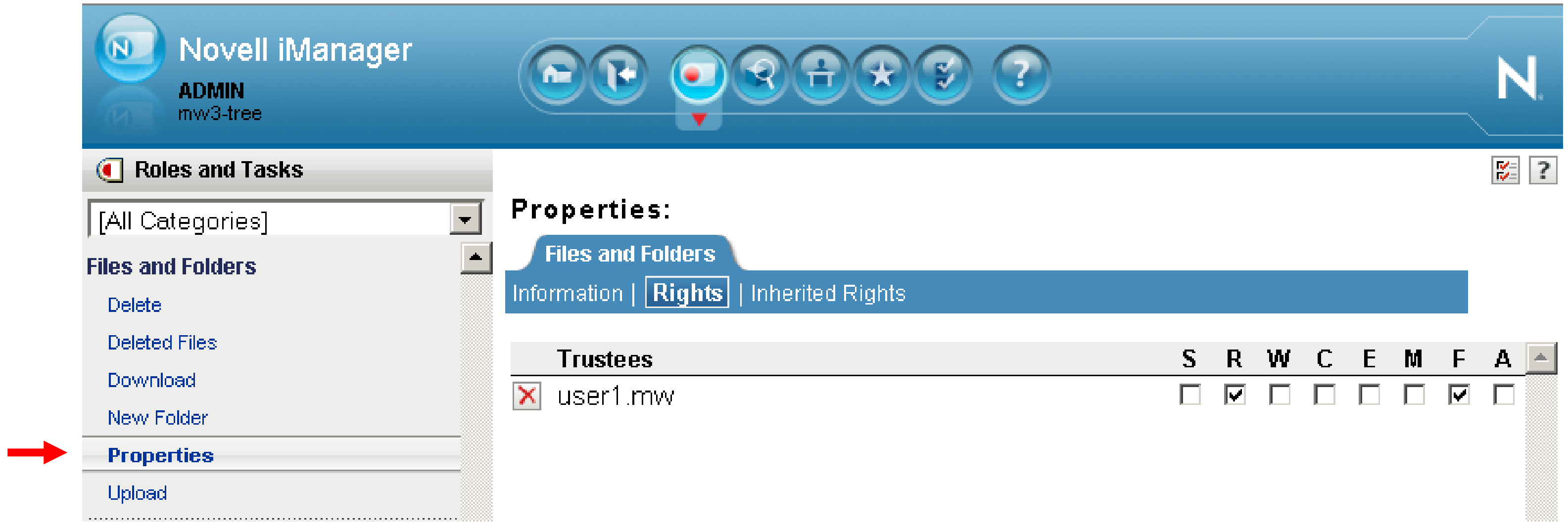

Address G:\PUBLIC\buried

```
Folders
Desktop
  + My Documents
  - My Computer
      + 3½ Floppy (A:)
      + Local Disk (C:)
      + EXTRA (D:)
      + CD Drive (E:)
      + Removable Disk (F:)
      - Sys on '10.0.0.13' (G:)
          + LOGIN
          - PUBLIC
              - buried
                  + DOS622
                  + FREEDOS
                  + OPENDOS
                  + W98SE
```

Name	Size	Type
DOS622		File Folder
FREEDOS		File Folder
OPENDOS		File Folder
W98SE		File Folder

Subdir "buried" and contents are visible

Toplevel needs trustee rights

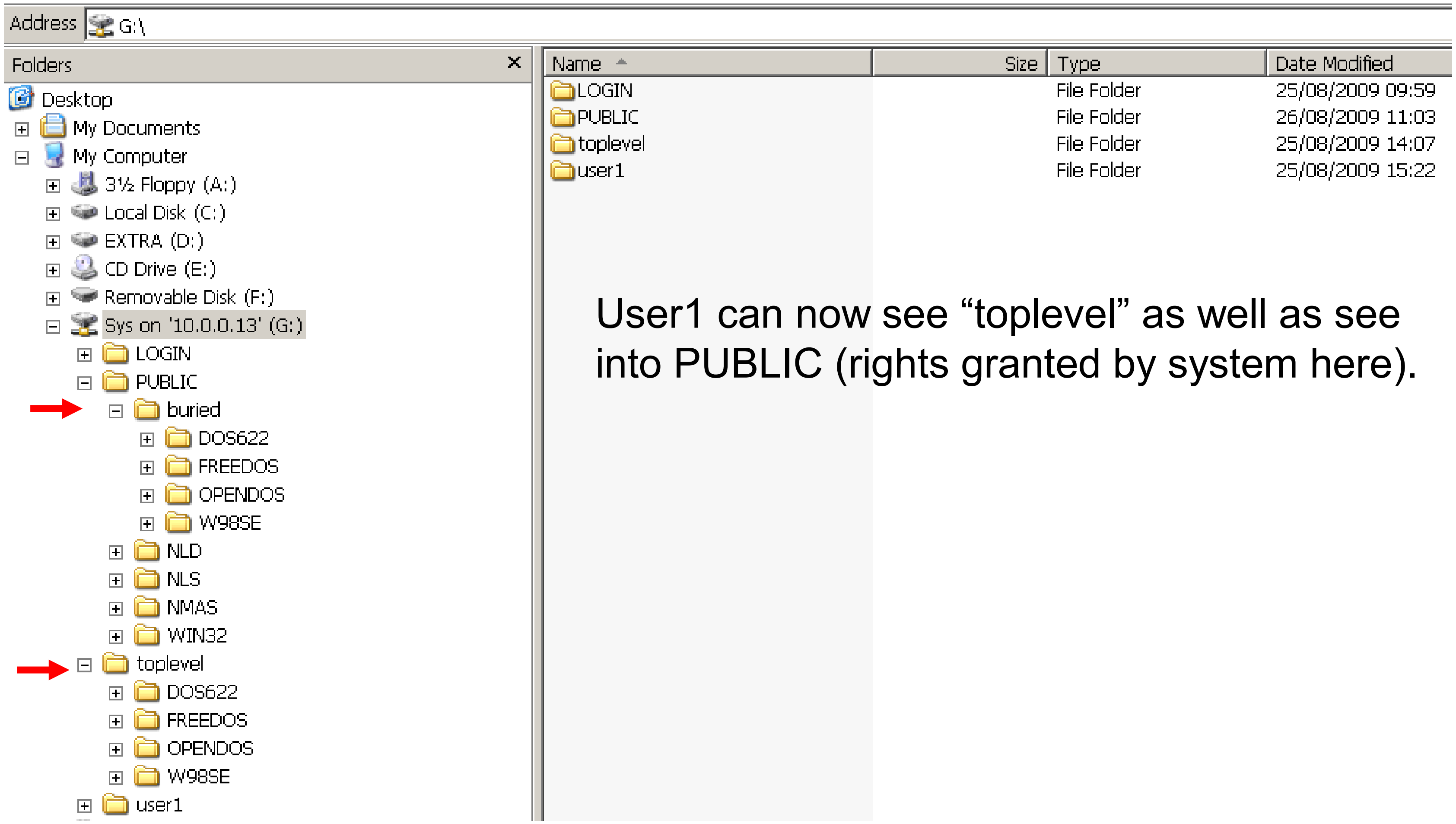


Trustees	S	R	W	C	E	M	F	A
user1.mw	☐	✓	☐	☐	☐	☐	✓	☐

Assign user1 as NCP trustee of SYS:toplevel with say RF rights

Seeing “toplevel” is independent of the mount operation

New view for user1


Address G:\

Folders

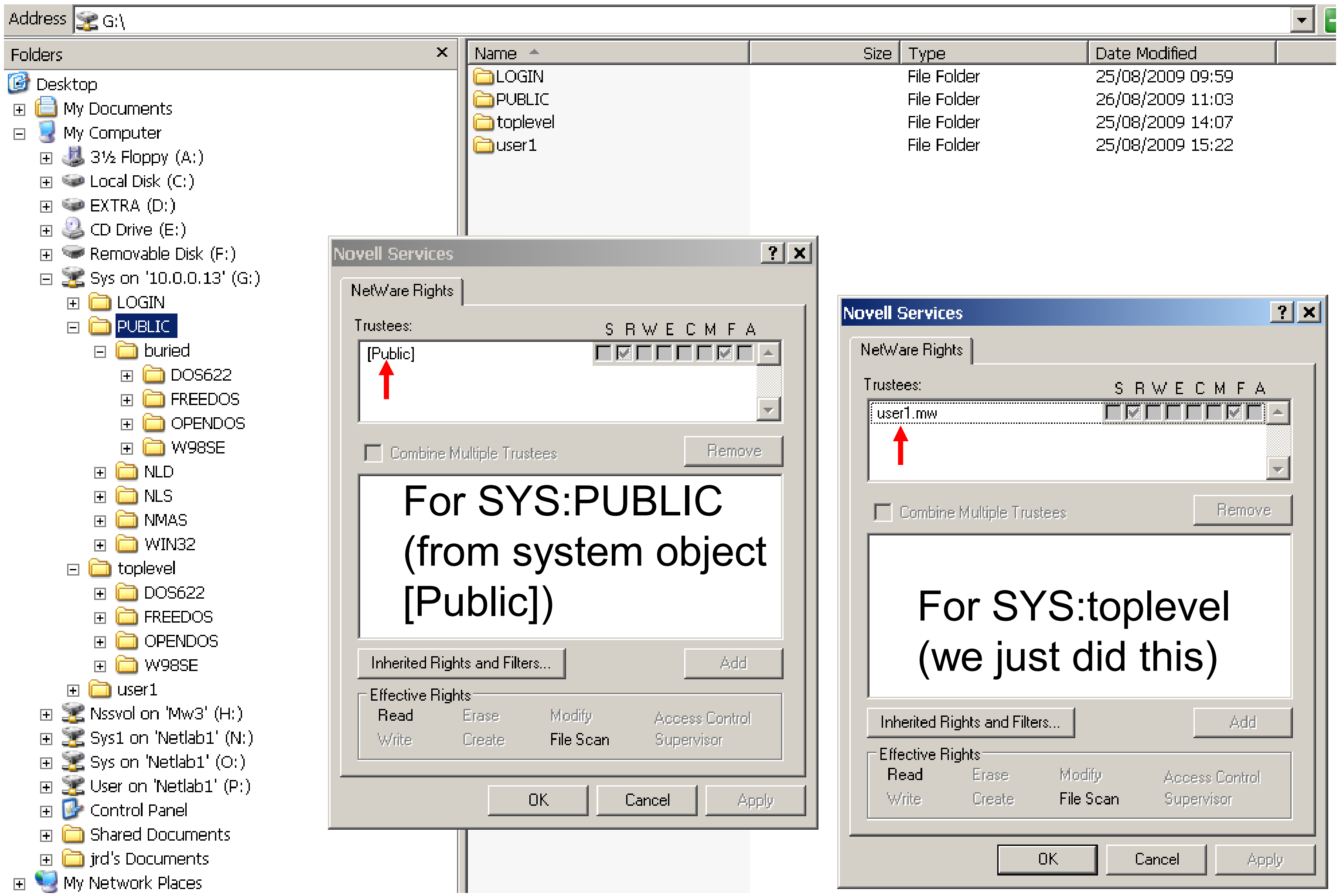
- Desktop
  - My Documents
  - My Computer
    - 3½ Floppy (A:)
    - Local Disk (C:)
    - EXTRA (D:)
    - CD Drive (E:)
    - Removable Disk (F:)
    - Sys on '10.0.0.13' (G:)
      - LOGIN
      - PUBLIC
        - buried
          - DOS622
          - FREEDOS
          - OPENDOS
          - W98SE
        - NLD
        - NLS
        - NMAS
        - WIN32
      - toplevel
        - DOS622
        - FREEDOS
        - OPENDOS
        - W98SE
      - user1

Name	Size	Type	Date Modified
LOGIN		File Folder	25/08/2009 09:59
PUBLIC		File Folder	26/08/2009 11:03
toplevel		File Folder	25/08/2009 14:07
user1		File Folder	25/08/2009 15:22

User1 can now see “toplevel” as well as see into PUBLIC (rights granted by system here).

MindWorks UK

Source of trustee rights


Address G:\

Folders

- Desktop
  - My Documents
  - My Computer
    - 3½ Floppy (A:)
    - Local Disk (C:)
    - EXTRA (D:)
    - CD Drive (E:)
    - Removable Disk (F:)
    - Sys on '10.0.0.13' (G:)
      - LOGIN
      - PUBLIC
        - buried
          - DOS622
          - FREEDOS
          - OPENDOS
          - W98SE
        - NLD
        - NLS
        - NMAS
        - WIN32
      - toplevel
        - DOS622
        - FREEDOS
        - OPENDOS
        - W98SE
      - user1
    - Nssvol on 'Mw3' (H:)
    - Sys1 on 'Netlab1' (N:)
    - Sys on 'Netlab1' (O:)
    - User on 'Netlab1' (P:)
    - Control Panel
    - Shared Documents
    - jrd's Documents
  - My Network Places

Name	Size	Type	Date Modified
LOGIN		File Folder	25/08/2009 09:59
PUBLIC		File Folder	26/08/2009 11:03
toplevel		File Folder	25/08/2009 14:07
user1		File Folder	25/08/2009 15:22

Novell Services

NetWare Rights

Trustees:	S	R	W	E	C	M	F	A
[Public]		✓					✓	

Combine Multiple Trustees — Remove

For SYS:PUBLIC (from system object [Public])

Inherited Rights and Filters... — Add

Effective Rights: Read, Erase, Modify, Access Control, Write, Create, File Scan, Supervisor

OK — Cancel — Apply

Novell Services

NetWare Rights

Trustees:	S	R	W	E	C	M	F	A
user1.mw		✓					✓	

Combine Multiple Trustees — Remove

For SYS:toplevel (we just did this)

Inherited Rights and Filters... — Add

Effective Rights: Read, Erase, Modify, Access Control, Write, Create, File Scan, Supervisor

OK — Cancel — Apply

MindWorks UK

Reason for trustee rights

Parent directory *_/usr/novell/sys_* holds a list of file names and their pointers to inodes/data for items within that directory. We need permission to read all or part that list.

User1 had NCP rights to read only selected parts of that top level directory, omitting name “toplevel”

NCP says: if no permission then no visibility. A feature!

A Linux user has the POSIX rights and will see directory name “toplevel” but will require appropriate rights to see its list/contents. Typical of Unix.

©MindworksUK 2009

MindWorks UK

File system internal topology

Superblock:

Partition (file system) layout numbers, pointers to cylinder groups, etc

Replicated for safety and recovery

Directory file data:

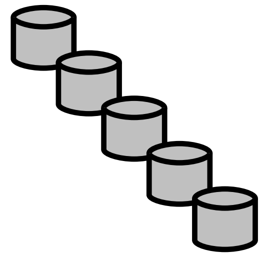
Filename1, pointer to inode1
Filename2, pointer to inode2
Filename3, pointer to inode3
Filename4, pointer to inode4
Filename5, pointer to inode5
Filename6, pointer to inode6
…

File's metadata

Inode data:
bookkeeping details

Pointers to disk blocks

File's data



Bulk storage

A directory is ordinary file, with binary data. Filenames live here, not in an inode.

Creates *a tree* of names

One inode per file, usually a fixed number at f/s creation

MindWorks UK

Reason for trustee rights

Command *mount -o bind* revises the inode pointer of “toplevel” to point to the list (data) of “DOS”

The original list pointer is safely tucked away for use by command *umount*

Thus after the mount, “toplevel” shows the contents of directory “DOS” but not the directory name “DOS” itself (which is held in its parent directory)

Notes, after the mount

- **Subdir name DOS does <u>not</u> appear, only its contents**
- **The path to DOS does <u>not</u> appear**
  - **Enhanced Security**
- **The source volume holding DOS need <u>not</u> be made visible to NCP clients**
  - **Simplified view of the storage farm:**
    - Shorter login scripts (fewer drive letters, etc)
    - Volumes may be hidden from NCP

Notes, after the mount

Linux apps have same access as users

NSS rights apply to NSS material

POSIX rights apply to POSIX material

LUM enable your applications to see NSS contents

Notes, after the mount

There are no symbolic links involved

Thus going into a subdir and then saying “cd ..” takes one up one level, to where one went in

No client nor application knows about this work, unlike CIFS Junctions. Only the kernel knows.

No messy side effects such as duplication of directory names and name collisions as with Shadow Volumes

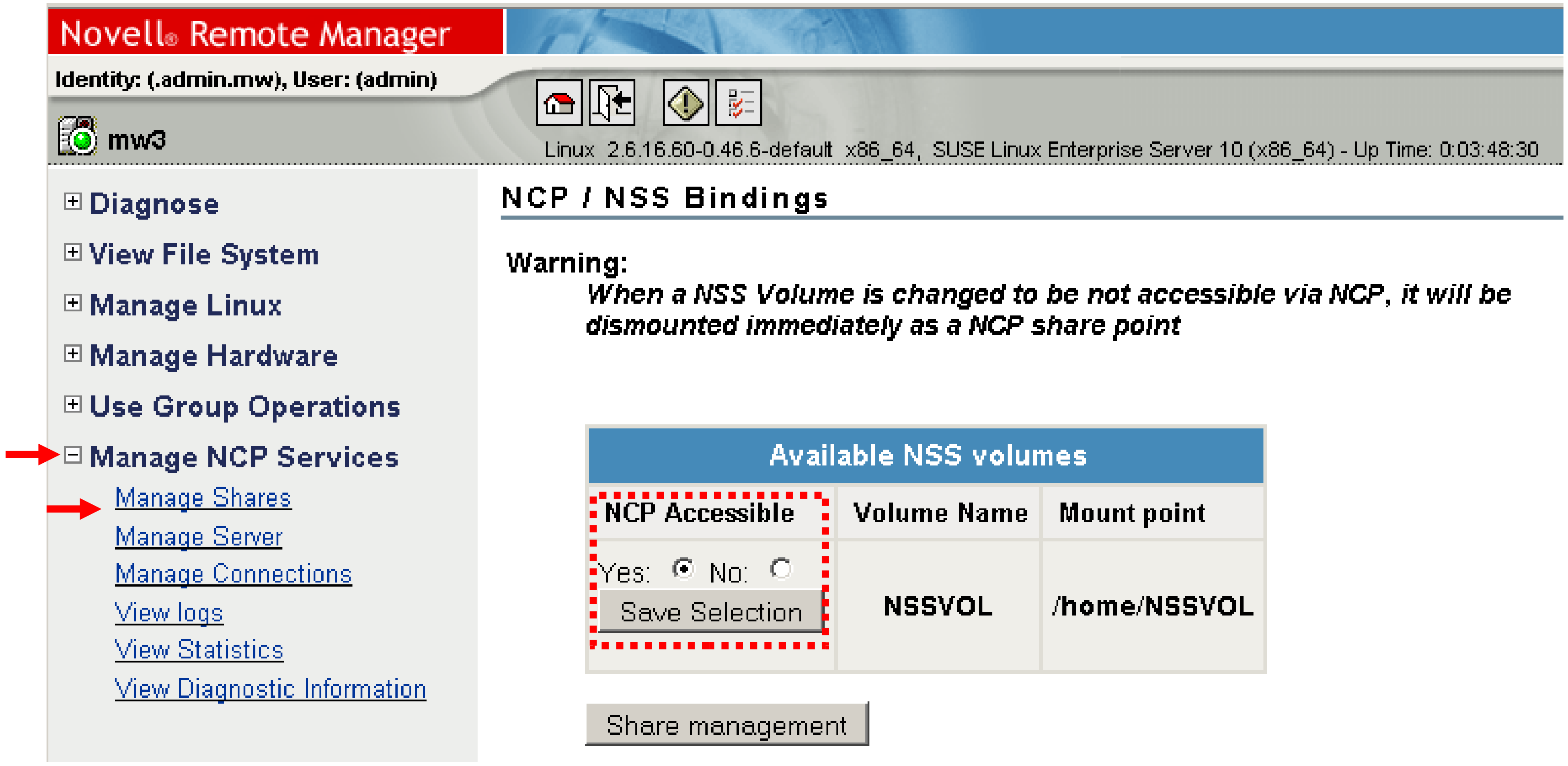
Notes, after the mount

CPU consumption is zero

Quota calculations may be confused

The result does not depend on how one gets there at user level

Hiding whole volumes from NCP


Novell® Remote Manager

Identity: (.admin.mw), User: (admin)

mw3

Linux 2.6.16.60-0.46.6-default x86_64, SUSE Linux Enterprise Server 10 (x86_64) - Up Time: 0:03:48:30

- ⊞ Diagnose
- ⊞ View File System
- ⊞ Manage Linux
- ⊞ Manage Hardware
- ⊞ Use Group Operations
- ⊟ Manage NCP Services
  - Manage Shares
  - Manage Server
  - Manage Connections
  - View logs
  - View Statistics
  - View Diagnostic Information

NCP / NSS Bindings

Warning:

When a NSS Volume is changed to be not accessible via NCP, it will be dismounted immediately as a NCP share point

Available NSS volumes

NCP Accessible	Volume Name	Mount point
[✓ Yes:] No: Save Selection	NSSVOL	/home/NSSVOL

Share management

The volume persists and is mounted, but NCP callers cannot see it

Automatic mounting at boot

If these *mount -o bind* commands are added to */etc/fstab*, then they will fail as the system boots because NSS is not running when *fstab* is read

Workaround is to create a new start/stop script, say copy from the SUSE skeleton, which does the mount operations after NSS has started

Start/stop script intro section

```
### BEGIN INIT INFO
# Provides:          mymount
# Required-Start:    $syslog $remote_fs nss
# Should-Start:
# Required-Stop:     $syslog $remote_fs nss
# Should-Stop:
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Short-Description: Remount NSS dirs
# Description:       Remount NSS dirs using option -bind
### END INIT INFO
```

©MindworksUK 2009

Start/stop script center section

```
case "$1" in
    start)
        echo -n “Remounting NSS directories"
        test `mount | grep /usr/novell/sys/toplevel` ||  \
         mount -o bind  /home/NSSVOL/Portlock/NWDSK/DOS \
   /usr/novell/sys/toplevel                        (repeat as req’d)
        rc_status -v
        ;;
    stop)
        echo -n "Dismounting exported NSS directories"
        umount /usr/novell/sys/toplevel            (repeat as req’d)
        rc_status -v
        ;;
```

----

mount **lists its existing mount points,** ***grep*** **seeks one, its result of success or failure is passed back to** ***test***

Only if failure is the OR (||) clause done to perform the full ***mount -o bind*** **command**

In short, don’t mount again if already mounted

Summary

We made subdirectories, or just individual files, appear at other places in the user level file tree

- **without copying**
- **without symbolic links or hard links**
- **without CIFS Junctions**
- **without Shadow Volumes/DST**
- **without revealing names of superior directories just to get to the interesting area**
- **without complexity or side effects**
- **without system loading**

©MindworksUK 2009

Summary

For both NSS and POSIX file systems (source and destination)

For all methods of access, users and applications alike

Without spending money or installing a new product

MindWorks UK

MindWorks Inc. Ltd
210 Burnley Road
Weir
Bacup
OL13 8QE UK

Telephone:	+44 (0) 170 687 1900
Fax:	+44 (0) 170 687 8203
Web:	www.mindworksuk.com
Email:	training@mindworksuk.com

©MindworksUK 2009