Samba and Btrfs

A Snapshot of Progress

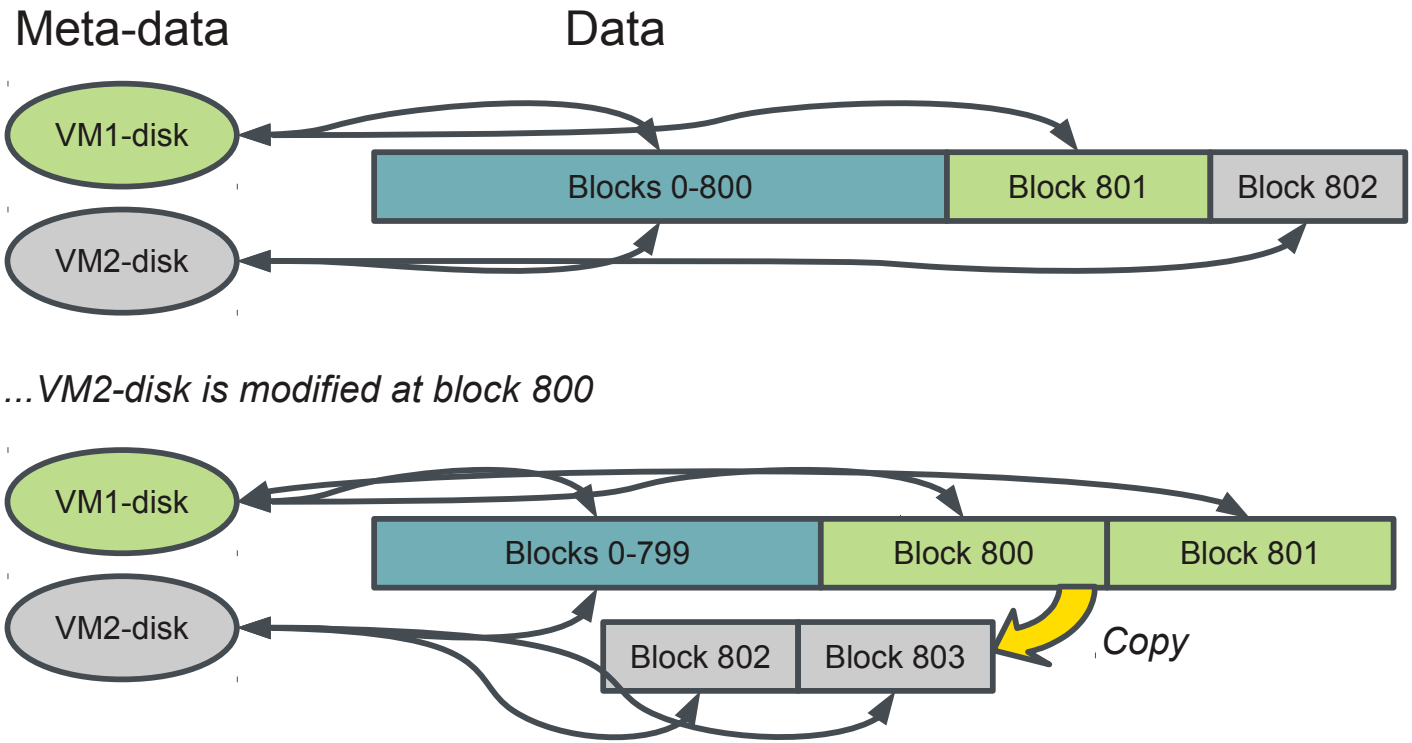
David Disseldorp
SUSE Labs & Samba Team
ddiss@suse.de

What is Btrfs?

- New filesystem for Linux
- Fully featured
  - Resiliency
    - Checksumming of data and meta-data
  - Redundancy
    - Multi-device support (mirrored data and meta-data, striped data)
  - Compression
  - Snapshots
  - De-duplication through shared data extents
    - Block, files or subvolume (snapshot) granularity

Shared Data Extents

Copy on Write



Benefits for Samba

- Shadow copies (snapshots)
  - Previous versions in Explorer
  - @GMT- token as path component
  - Management via File Server Remote VSS Protocol (FSRVP)
- Efficient server-side copy offload
- De-duplication
  - Data duplication avoidance from client to server

File Share Shadow Copies
Introduction

- File Server Remote VSS Protocol
  - MS-FSRVP new with Windows Server 2012
  - DCE/RPC requests via \pipe\FssagentRpc
  - Introduced by Molly Brown at SNIA SDC 2011

- Clients can manage shadow copies remotely
  - Determine which shares may be shadow copied
  - Request the creation of a shadow copy for a given share
  - Request the exposure of a shadow copy as a new share

File Share Shadow Copies
Windows

- Volume Shadow Copy Service (VSS) ecosystem
  - Application consistent shadow-copies for backup
  - Providers, Requestors and Writers
    - File Share Shadow Copy provider
    - VSS aware applications "Writers" on client flush to disk prior to snapshot
- Microsoft File Server Shadow Copy Agent Service
  - diskshadow.exe
  - System Center Data Protection Manager (DPM) 2012

File Share Shadow Copies
Samba FSRVP Client Implementation

- fsrvp.idl
  - Based on preliminary MS-FSRVP documentation
- rpcclient
  - fss_is_path_supported, fss_create_expose, fss_delete, etc.
  - Shadow copy identifiers need to be retained by the client
    - Strangely no shadow copy enumeration RPC
- Smbtorture test suite
  - rpc.fsrvp

File Share Shadow Copies
Samba FSRVP Server Implementation

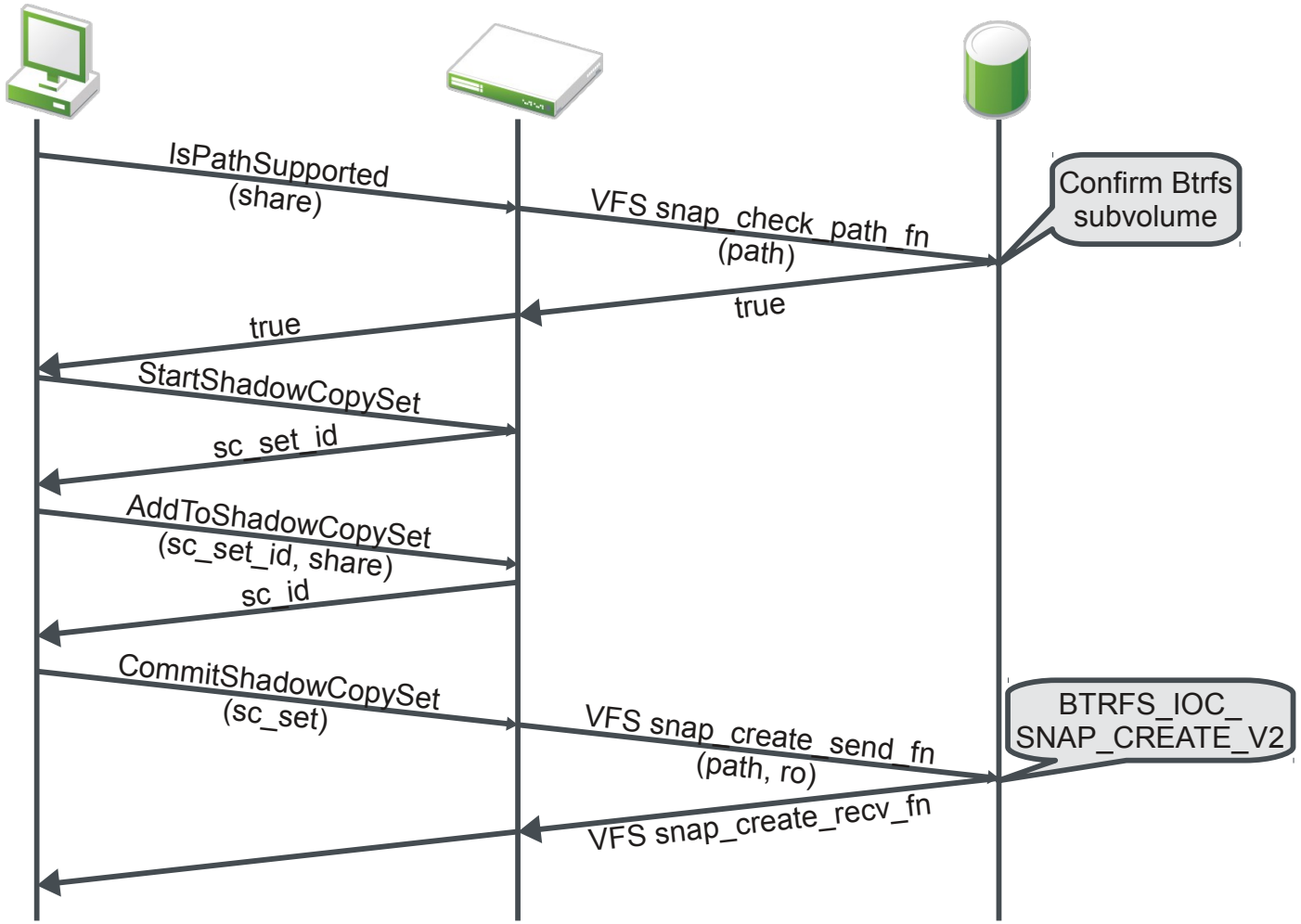
- Fssd forked on start-up
  - *rpc_daemon:fssd = fork*
  - *registry shares = yes*
- Snapshot requests propagated to VFS
  - Check whether path supports shadow copies
  - Asynchronous create/delete shadow copy requests
- Shadow copy shares defined in the registry
  - Clone of base share definition
  - libsmbconf

File Share Shadow Copies
Samba FSRVP Server Implementation

- vfs_btrfs
  - Issues Btrfs ioctls for snapshot creation and destruction
- vfs_snapper
  - System wide snapshot management and rollback
    - Module work in progress
- vfs_shadow_copy
  - Presents FSRVP snapshots as previous versions in Explorer
  - “@GMT-” path component

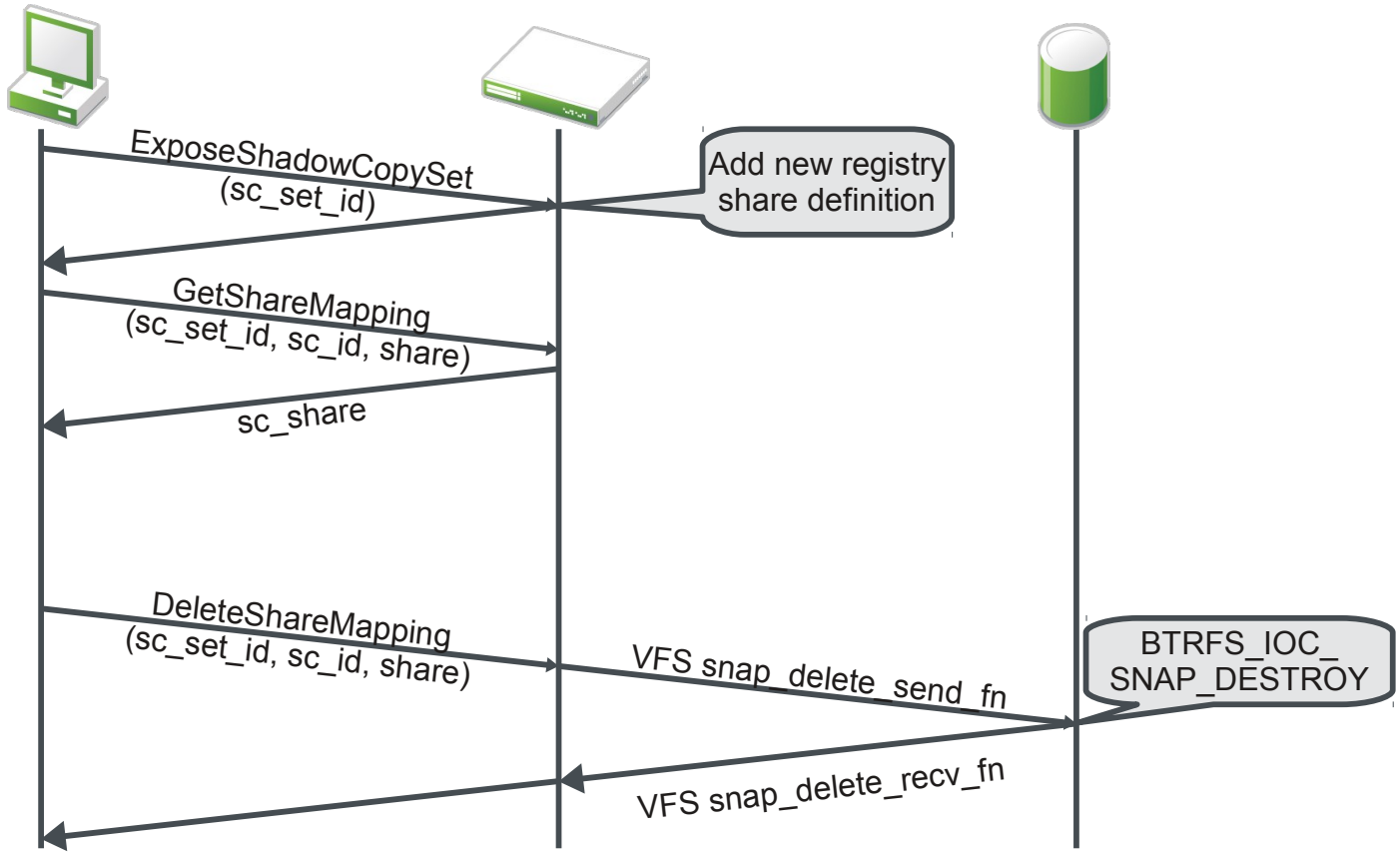
FSRVP on the Wire

Shadow Copy Creation



FSRVP on the Wire

Shadow Copy Exposure & Deletion



FSRVP Client Demo

- Server = Windows Server 2012 Beta
- Client = Samba rpcclient

FSRVP Server Demo

- Server = Samba with FSRVP changes
- Client = Samba rpcclient

Server-Side Copy

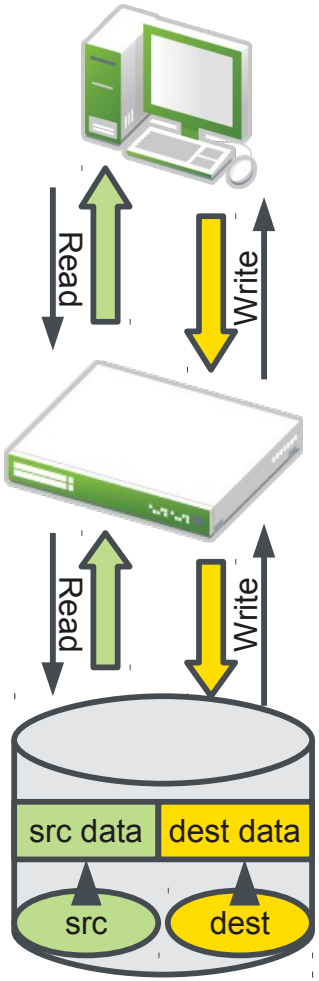
- Regular copy
  - Client reads data from the server
  - Client writes same data back to the server
  - Disk and network round trip
- Server-side copy
  - Client offloads the copy operation to the server
  - Server reads data from disk, then writes back
  - Disk round trip only
- Server-side copy SMBs
  - FSCTL_SRV_COPYCHUNK
    - FSCTL_SRV_REQUEST_RESUME_KEY (obtain source file identifier)
  - FSCTL_OFFLOAD_READ / FSCTL_OFFLOAD_WRITE

Server-Side Copy
Samba with Btrfs Enhancements

- Client offloads the copy operation to the server
  - SMB2 FSCTL_SRV_COPYCHUNK supported
    - Multiple chunks per request, 16M maximum offload per request
- Samba request Btrfs clone the range of bytes
  - BTRFS_IOC_CLONE_RANGE
  - Meta-data operation only, no network or disk round trip
    - Duplicate data does not consume disk space
  - Must be Btrfs block-size aligned

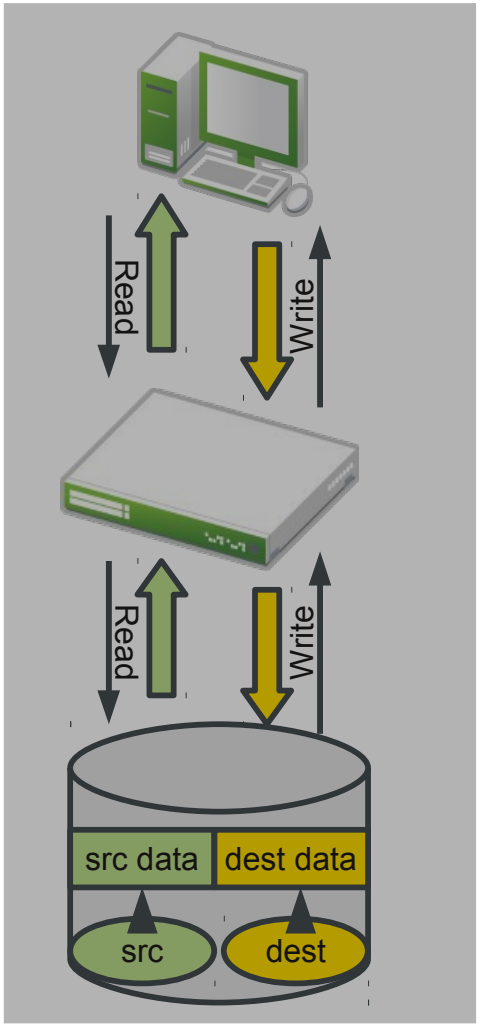
Server-Side Copy on the Wire

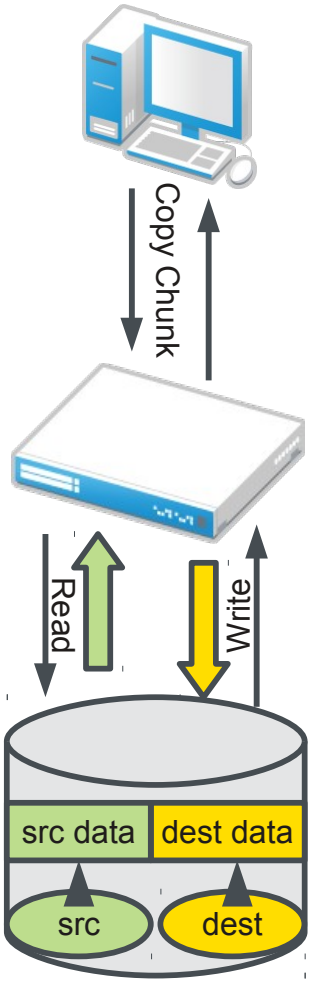
Client-Server exchange



Server-Side Copy on the Wire

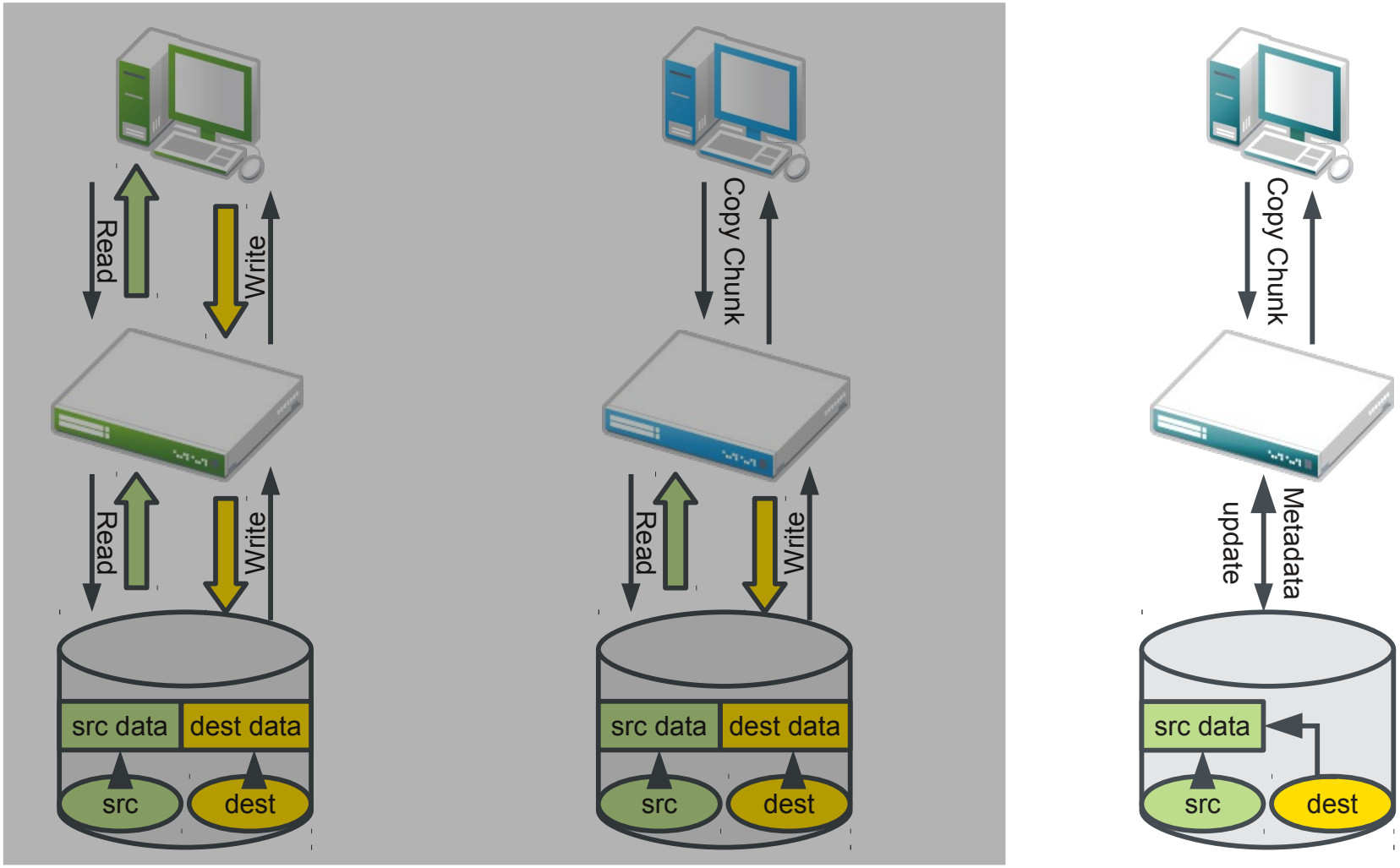
Client-Server exchange





Server-Side Copy on the Wire

Client-Server exchange



Server-Side Copy Demo

- Client = Windows Server 2012 Beta
- Server = Samba with server-side copy changes

Implementation State
Server-Side Copy

- Changes tested and (mostly) reviewed
  - Disjoint between SMB1 and SMB2 ioctl handlers
- Potential improvements
  - Encourage client Copy Chunk request alignment
    - IOCTL_STORAGE_QUERY_PROPERTY FileFSSectorSizeInformation
  - Linux CIFS kernel client support
  - Add support for new offload read/write ioctls
  - Support for other filesystems where possible
    - OCFS2, ZFS, etc.

Implementation State
FSRVP Server

- Further work required
  - Privilege checks
  - Thorough asynchronous RPC-server testing and review
  - Asynchronous dispatch of Btrfs snapshot ioctls
  - Store persistent FSRVP state in a TDB
  - Request and state timeouts

- Support for other filesystems
  - LVM + XFS, ZFS, etc

Questions?

Code: http://git.samba.org/?p=ddiss/samba.git
Slides: http://www.samba.org/~ddiss/

Thank you.

References

- Demonstrated Samba code
  - git://git.samba.org/ddiss/samba.git async_fsrvp_srv_wip_sxp2012
- MS-FSRVP preliminary documentation
  - http://msdn.microsoft.com/us-en/library/hh554852.aspx
- SNIA SDC 2011 - Advancements in backup - Molly Brown
  - http://www.snia.org/sites/default/files2/SDC2011/presentations/tuesday/MollyBrown_Advancements_In_Backup.pdf
- Btrfs Wiki
  - btrfs.wiki.kernel.org
- I Can't Believe This is Butter! A tour of btrfs by Avi Miller
  - http://www.youtube.com/watch?v=hxWuaozpe2I

References

- WS8 Beta Storage Availability White Paper
  - http://download.microsoft.com/download/6/7/6/676CCB32-8ECC-4793-A698-68D0AFD2F1A9/WS8%20Beta%20Storage%20and%20Availability%20White%20Paper.pdf
- Offline Deduplication for Btrfs
  - http://thread.gmane.org/gmane.comp.file-systems.btrfs/8448


suse®

This document could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein. These changes may be incorporated in new editions of this document. SUSE may make improvements in or changes to the software described in this document at any time.