SerN et

Apple Dances Samba

sambaXP 2014 - May 14, 2014

Ralph Böhme

SerNet GmbH, Göttingen - Berlin



What's the story ?

OS X Mavericks: new default protocol SMB2

Pre-OS X Mavericks: AFP

OS X happily connects to any SMB server including Samba.

And it works – until it doesn't:

Apple dances Samba ?

SerN et



Apple dances Samba

SerN et

Challenges and Pitfalls

- Alternate data streams
- Character encoding
- Interoperability with Netatalk/AFP
 - Access to data stored used with Netatalk
 - Locking across SMB and AFP protocols
- Spotlight

Apple dances Samba • Challenges and Pitfalls

SerN

et

Goals:

- Develop a new VFS module for a true cross-protocol SMB/AFP fileserver
- Add Apple Spotlight search support to Samba

Code, work in progress:

- https://github.com/slowfranklin/samba/commits/vfs_apple
- https://github.com/slowfranklin/samba/commits/spotlight

Challenges and Pitfalls

SerN et

Alternate Data Streams (ADS)

OS X and Alternate Data Streams

In OS X every file can have optional data and metadata

SerN

61

- Legacy Mac OS Finder metadata (fixed size)
- An additional full fledged file fork called resource fork
- Extended Attibutes (xattrs)
- OS X SMB client checks whether SMB server supports ADS:
 - No: client packs all extra data in an ._ prefixed AppleDouble file
 - Yes: client sends data as ADS
- OS X SMB server supports ADS

SerN et

ADS used by OS X:

- Mac metadata stream: AFP_AfpInfo
 - Fixed size (60 bytes)
 - Contains Mac metadata like Type, Creator, creation date, attributes (visibility, locked, ...)
- Mac resource fork stream: AFP_Resource
 - Any size (like normal file)
- Extended attributes are also send as ADS

SerN et

Samba and ADS

- Samba without ADS and case sensivity: lots of possibly negative name lookups for ._ AppleDouble files which result in repeated directory rescans
- Opening folders with many files: performance goes south
- Samba supports ADS:
 - vfs_streams_xattr: size limitation (remember: resource fork), restricted xattr API (no partial reads/writes)
 - vfs_streams_depot: works, but incompatible with access from other network protocols



New VFS module vfs_apple to the rescue:

- Supports ADS
- Store resource fork (AFP_Resource) in filesystem xattr in case the OS and filesystem support this (Solaris/ZFS), otherwise resort to storing it in an AppleDouble helper file
- Store Mac OS metadata (AFP_AfpInfo) in an xattr
- Avoids costly lookups for ._ AppleDouble files
- (Optionally) Compatible with Netatalk

Challenges and Pitfalls



Character Encoding



SerN

et

NTFS illegal characters: / | * " <> ?

Name						
v 🚞	SambaXP 2014 / images **very nice**					
	🙀 Samba ".jpg					
	🙀 Samba *.jpg					
 Samba \o/.jpg Samba <.jpg 						
	📓 Samba I.jpg					
	Samba.jpg					

• OS X SMB Client maps them to Unicode private range [1]

On the server:

SerN

et

replacement glyph for characters in the Unicode private range.

vfs_apple to the rescue:

```
$ ls -1 "/Volumes/vfs_apple/SambaXP 2014 : images **very nice**"
Samba <.jpg
Samba >.jpg
Samba |.jpg
Samba ".jpg
Samba *.jpg
Samba.jpg
Samba \o:.jpg
```

SerN

et

- Employs a builtin mapping for vfs_catia, characters in their native encoding
- Works in both ways, even Window clients are happy:

SerN et

🐌 ▶ Netzwerk ▶ 10.10.10.139 ▶ r	normal 🕨 SambaXP 2014 • image	es ••very nice••	✓ C Sar
Name	Änderungsdatum	Тур	Größe
📔 Samba •	05.04.2014 08:37	JPEG-Bild	78 KB
🔄 Samba •	05.04.2014 08:37	JPEG-Bild	78 KB
🔄 Samba •	05.04.2014 08:37	JPEG-Bild	78 KB
🔄 Samba •	05.04.2014 08:37	JPEG-Bild	78 KB
🔄 Samba •o•	05.04.2014 08:37	JPEG-Bild	78 KB
🔄 Samba •	05.04.2014 08:37	JPEG-Bild	78 KB
📔 Samba	05.04.2014 08:37	JPEG-Bild	78 KB

Windows 8 client displaying files with illegal NTFS characters

Windows uses a • as replacement glyph for characters in the Unicode private range, not the

Challenges and Pitfalls





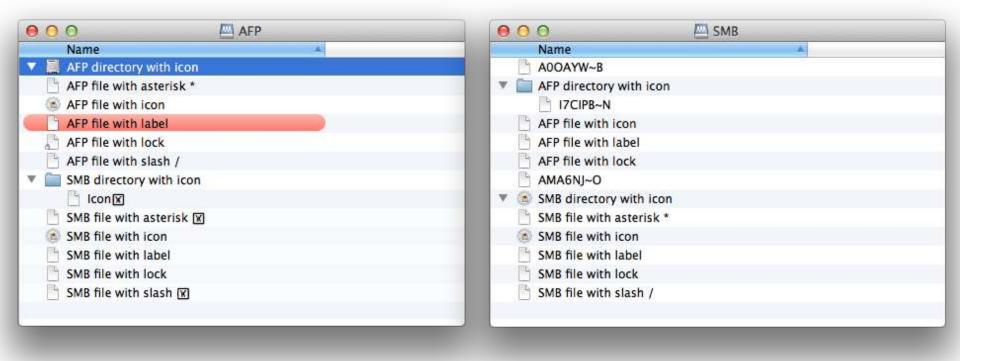
Interoperability



Interoperability?

SerN

et



One element displays correctly, can you find it?

vfs_apple, while we're at it lets add some sweet topping:

Map relevant named streams AFP_AfpInfo and AFP_Resource to corresponding Netatalk data structrures SerN

21

- Cross-protocol locking: AFP file sharing modes similar to SMB
 - SMB open modes can be checked against AFP sharing modes with a simple fcntl() byte-range lock
 - This is not yet implemented

0 0	O AFP	SMB with vfs_apple
	Name 🔺	Name 🔺
	AFP directory with icon	AFP directory with icon
E	AFP file with asterisk *	AFP file with asterisk *
6	AFP file with icon	AFP file with icon
	AFP file with label	AFP file with label
	AFP file with lock	AFP file with lock
	AFP file with slash /	💾 AFP file with slash /
. (8	SMB directory with icon	SMB directory with icon
P	SMB file with /	SMB file with /
1	SMB file with asterisk *	SMB file with asterisk *
	SMB file with icon	SMB file with icon
P	SMB file with label	SMB file with label
100	SMB file with lock	SMB file with lock
1		

With vfs_apple

SerN

et

Work in progress:

- Displaying directory icon via SMB somtimes not working
- Locked flag (SMB1: read only attribute, SMB2 ?)

Example smb.conf:

```
[SMB with vfs_apple]
   path = /Volumes/vfs_apple
   vfs objects = catia apple streams_xattr
   apple:ressource = file
   apple:metadata = netatalk
   apple:locking = netatalk
   apple:encoding = native
```







- Apple technology for indexing and searching data
- Introduced 2005 in Mac OS X 10.4, desktop search only
- Since Mac OS X 10.5 search queries can be sent over the network to connected AFP fileservers
- Since Mac OS X 10.7 networked Spotlight supports SMB connections to Apple's own SMB server SMBX
- Added Spotlight support to Netatalk in 2012/2013
- Reverse engineered as Spotlight implementation details are not published by Apple

SerN

ei

On the wire:

- Over the wire format supports marshalling simple (integer, bool, float) and complex (string, set, key/value dictionary) data
- Marshalled data stream contains full type information
- Therfore doesn't require IDL
- Call it SDR (Spotlight Data Representation)

	W 5	Spot	liç	ht	RPC	da	ita											
		v a	rra	у,	toc	in	dex	: 1,	, ch	ild	ren	: 1						
			a	ray	1, 1	oc	ind	dex:	2,	ch:	ild	ren:	3					
			v	st	rin	g,	toc	in	dex:	з,	st	rin	g:	'fe	tch	Prop	ertiesForC	ontext:'
					sti	ring	9:	fet	chP	rope	ert	iesl	For	Cont	tex	t:'		
			v	in	t64													
					int	64	: 0:	x000	000	000	000	000	0					
					int	:64	: 0:	×000	000	000	000	000	3					
	v (Comr	lex	t	nes	T	C (3 е	ntri	es	0.0000.0		20					
	0.80 P							s (3										
			nkn						.,									
		- 25	nkn															
		100		-	+.	1	tvo	o	arra	v	off	cot	• 1	6				
						2.2			arra									
						2.6	- C								fro	+. 2	2	
			• •	au	byt	ec	oun		, ·	ype	. 5	-	ny,	01	rse	t: 3	2	
00e0	00	00	e0	00	00	00	00	00	02	00	00	00	00	00	bc	67		g
00f0		23	12.2	1.00	10.00	- 7.9	1.200		1000	0707	- T		92		2.7		.#7t@Z	5
0100	15151	00	10.12	00	00		15	1.7.7	00	1000	1.20	00	1000	00	12.2		•••••	0.0000000000000000
0110	1000	6b	5.57	2.7	00	- 7.7	88	1.7.7.1	00	1507	12.2	00	1000	00	122	12.2	. k	
0120	100	00	557	2.2	00	-7.7	88	1.7.7			10000	1-20-0	32	0.000	928622	1000		
	64	6d	10	00	00	00	0c	00	00	00	01	00	00	02	01	00	dm	
22020	-	-							-				-				ulli	
0140	00		01				02		00			00	00		03	00		
0140 0150	00	00	05	00	00	07	02	00	00	00	66	00 65	74	63	03 68	00 50		fetchP
0140 0150 0160	00 72	00 6f	05 70	00 65	00 72	07 74	02 69	00 65	00 73	00 46	66 6f	00 65 72	74 43	63 6f	03 68 6e	00 50 74	ropertie	fetchP sForCont
0140 0150 0160 0170	00 72 65	00 6f 78	05 70 74	00 65 3a	00 72 00	07 74 00	02 69 00	00 65 00	00 73 00	00 46 00	66 6f 03	00 65 72 00	74 43 00	63 6f 84	03 68 6e 02	00 50 74 00		fetchP sForCont
0140 0150 0160 0170 0180	00 72 65 00	00 6f 78 00	05 70 74 00	00 65 3a 00	00 72 00 00	07 74 00 00	02 69 00 00	00 65 00 00	00 73 00 00	00 46 00 00	66 6f 03 00	00 65 72 00 00	74 43 00 00	63 6f 84 00	03 68 6e 02 00	00 50 74 00	ropertie	fetchP sForCont
0140 0150 0160 0170 0180 0190	00 72 65 00 00	00 6f 78 00 00	05 70 74 00 04	00 65 3a 00 00	00 72 00 00	07 74 00 00 88	02 69 00 00 00	00 65 00 00 00	00 73 00 00 00	00 46 00 00 00	66 6f 03 00 02	00 65 72 00 00 00	74 43 00 00 00	63 6f 84 00 0a	03 68 6e 02 00 01	00 50 74 00 00	ropertie ext:	fetchP sForCont
0140 0150 0160 0170 0180 0190 0190	00 72 65 00 00	00 6f 78 00 00 00	05 70 74 00 04 03	00 65 3a 00 00 00	00 72 00 00 00 00	07 74 00 00 88 0a	02 69 00 00 00 03	00 65 00 00 00 00	00 73 00 00 00 00	00 46 00 00 00 00	66 6f 03 00 02 04	00 65 72 00 00 00 00	74 43 00 00 00 00	63 6f 84 00 0a 0c	03 68 02 00 01 02	00 50 74 00 00 00	ropertie ext:	fetchP sForCont
0130 0140 0150 0160 0170 0180 0190 01a0 01a0 01b0 01c0	00 72 65 00 00 00	00 6f 78 00 00	05 70 74 00 04 03 00	00 65 3a 00 00 00 00	00 72 00 00 00 00 00	07 74 00 88 0a 00	02 69 00 00 00 03 00	00 65 00 00 00 00 00	00 73 00 00 00 00 00	00 46 00 00 00 00	66 6f 03 00 02 04	00 65 72 00 00 00 00	74 43 00 00 00	63 6f 84 00 0a 0c	03 68 02 00 01 02	00 50 74 00 00 00	ropertie ext:	fetchP sForCont

SerN

On the wire, SMB:

- SDR (remember: Spotlight Data Representation) wrapped inside DCE/RPC
- Uses mdssvc pipe
- Good news: Spotlight on the wire data mostly the same in AFP and SMB
- Added Wireshark dissector, already available, calls into AFP dissector for SDR

```
SMB (Server Message Block Protocol)
SMB Pipe Protocol
Distributed Computing Environment / Remote Procedure Call (DCE/RPG)
Spotlight metadata search service, cmd
    Operation: cmd (2)
    [Response in frame: 294]
  Policy Handle
    Unkn1: 0
    Device Id: 822083610
    Unkn3: 23
    Unkn4: 0
    Flags: 1795162113
  Request Blob
       Length: 136
       Size: 136
       Endianness: Little Endian
       ToC Offset: 88 Bytes, Query length: 120 Bytes
    Spotlight RPC data
       v array, toc index: 1, children: 1
         v array, toc index: 2, children: 3
            v string, toc index: 3, string: 'fetchPropertiesForCont
                 string: 'fetchPropertiesForContext:'
            int64
                 int64: 0x00000000000000000
                 int64: 0x0000000000000000
     Complex types ToC (3 entries)
         Number of entries (3)
         unknown
         unknown
         1: count: 1, type: array, offset: 16
         2: count: 3, type: array, offset: 24
         3: pad byte count: 6, type: string, offset: 32
```

SerN et

On the server:

- Receive SDR data stream
- Unpack query which is in string format. Examples:
 - *=="main*"cdw||kMDItemTextContent=="main*"cdw
 - _kMDItemGroupId=="13"
 - kMDItemContentTypeTree=="public.tiff"cd
- Use a scanner and a parser for translating this to SPARQL and throw it at Gnome Tracker



SPARQL? Tracker?

SerN et

Gnome Tracker:

- "Tracker is a search engine, search tool and metadata storage system." [2]
- Crawls and monitors filessystem, stores metadata in a index database
- Robust, mature, capable, well designed, standard on Linux
- Provides a library for querying the store (sync and async)
- Query language is SPARQL

SerN et

SPARQL:

- "SPARQL (a recursive acronym for SPARQL Protocol and RDF Query Language) is an RDF query language, that is, a query language for databases, able to retrieve and manipulate data stored in Resource Description Framework format." [3]
- RDF? Nevermind!
- SPARQL supports all relevant logic expressions queries, therefor Spotlight search strings can be transformed one-to-one

SPARQL examples:

SELECT DISTINCT ?url WHERE { ?obj nie:url ?url FILTER(regex(?url, '^file:///Volumes/spotlight/')) . ? obj fts:match 'main*'} SerN

et

SELECT DISTINCT ?url WHERE { ?obj nie:url ?url FILTER(regex(?url, '^file:///Volumes/spotlight/')) . ? obj nie:mimeType 'image/tiff'}

Demo Time

SerN et

Spotlight in action

SerN et

Thank you!

Ralph Böhme, rb@sernet.de

SerNet GmbH						
Bahnhofsallee 1b	Schützenstr. 18					
37081 Göttingen	10117 Berlin					
tel +49 551 370000-0	+49 30 5 779 779 0					
fax +49 551 370000-9	+49 30 5 779 779 9					
37081 Göttingen tel +49 551 370000-0	10117 Berlin +49 30 5 779 779 0					

http://www.sernet.de

© 2014, SerNet GmbH

References

SerN et

- [1] http://support.microsoft.com/kb/q117258
- [2] https://wiki.gnome.org/Projects/Tracker
- [3] http://en.wikipedia.org/wiki/SPARQL