

# Making a Product with Samba



**Jeremy Allison**  
**Samba Team/Google Open**  
**Source Programs Office**

**[jra@samba.org](mailto:jra@samba.org)**  
**[jra@google.com](mailto:jra@google.com)**

# What is Samba ?

- File server (SMB/CIFS/POSIX CIFS/SMB2)
- Clustered file server (CTDB)
- Print server
- NetBIOS over TCP server
- Single sign on for Windows AD/NT Domains
- Authentication server for AD (Samba4) and NT-Domains (Samba3 and 4).
- LDAP server (Samba4)
- Kerberos server (Samba4)

# What is Samba (continued)?

- Set of libraries.
  - Client libraries - libsmbclient (SMB1 - SMB2 under development)
  - Memory allocation - talloc
  - Database - tdb/ldb
  - Clustering - ctdb
  - Asynchronous events - tevent
- Linux kernel client.
  - SMB1/CIFS
  - SMB2 under development

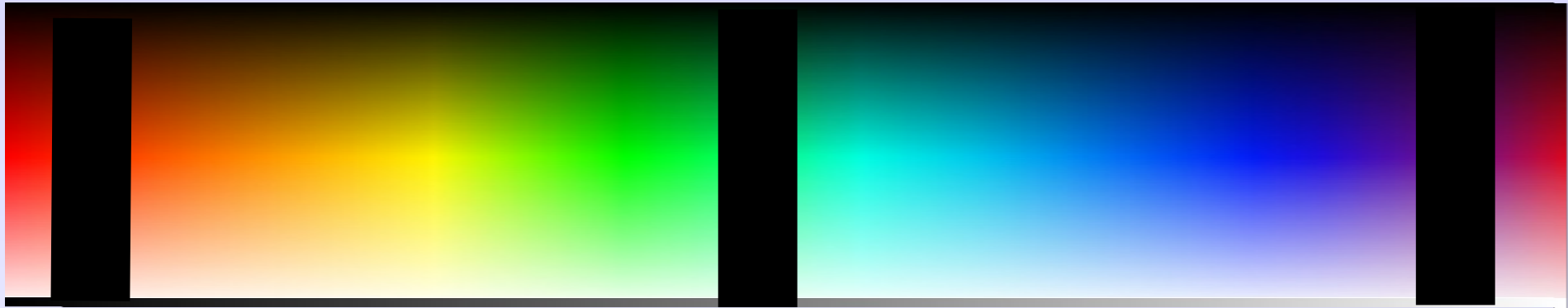
# But is Samba a product ? Or is it a set of technologies ?



**But is Samba a product ?  
Or is it a set of  
technologies ?**



# Samba is a spectrum of solutions people can use



Plug and play/  
Integrate and  
forget

Semi-customizable  
solution

Technology kit  
of parts

# Plug and play users

- Commodity component
  - We need a CIFS/SMB/SMB2/AD-member server as part of our product.
- Embedded solution
  - File serving not a primary target of the product.
  - Media players, access points/gateway boxes, phones, *really* low end NAS (plug in your own USB disk).
  - Copy the bits from samba.org, hack it up as needed and ship.

# Mid-level users

- Linux/UNIX Distributions.
  - Red Hat, SuSE/Novell, Debian/Ubuntu
  - HP/AIX/Solaris/OS X
- Interoperability isn't what they sell, but it is a core part of the product.
  - All ship some version of Samba fileserver.
  - Gnome/KDE use libsmbclient.
  - All Linux distributions use the CIFS VFS kernel client.



# More mid-level users

- Mid-level NAS vendors.
  - Vendors/OEM's for which file serving/member server/authentication integration is an important part of the product.
- Often have custom versions of the Samba code in internal trees.
  - Patch source, fix their own bugs.
- Apply engineering effort to triage bugs, report back to Samba bugzilla.
  - Usually have a relationship/part of Samba Team.

# High end users

- Require customized versions of Samba, tightly integrated with external (often proprietary) code.
  - Clustered NAS.
  - High end NAS servers.
  - Print Servers.
  - WAN acceleration.
  - Active Directory Domain service.
  - Secret projects (say no more... :-).

# High end users (continued)

- Often employ Samba Team members/engineering talent which ends up on the Team.
- Drive the long-term development of Samba.
  - Create new features.
  - Drive the direction of the overall project.
- Have the most “*interesting*” legal and licensing issues :-).

# Talking to the Samba Team

*“I'm surprised. You have the reputation of being a rabid bunch of Free Software zealots, yet you're some of the most business-friendly Open Source people I've ever met.*

*Whoever does your PR sucks”:*

Samba OEM who shall go un-named,  
SambaXP 2008.

# “But I have proprietary code..”

- This is not usually a problem.
  - We have yet to find a vendor who could not work with us to integrate their proprietary code.
  - It may be more work than they are willing to do in order to get the value that Samba offers them.
- Samba *will* change our code to aid integration with proprietary code.
  - So long as this is an improvement.
  - Samba is not hostile to proprietary code solutions, but we prefer to develop our code as Free Software.

# How to fail using Samba

- Hide who you are...
  - Use anonymous gmail/yahoo/hotmail accounts to ensure no one knows you're an important OEM user.
- Silently hope your bug will get fixed...
  - If you're seeing it, it *must* be a common bug. Don't submit a bug report, just keep waiting to see if the Team notices the problem. Surely they will..
- Go via a chain of OEM's...
  - Get Samba embedded in a chain of relicensed products, preferably from an anonymous third-party vendor.

# How to fail using Samba (continued)

- Don't bother with licensing...
  - After all, it's all free anyway, and no one really cares about that stuff.
- Who needs the Samba Team anyway...
  - They don't know anything about this stuff and probably wouldn't be able to understand *our* use of Samba. No point talking to them. Anyway, aren't they all communists ?
- Don't feed back our changes...
  - How hard can it be to forward port changes ?

# How to be successful with Samba (the basics)

- Talk to the Samba Team.
  - Despite our fearsome reputation, we are very business-friendly.
  - Many of us make our living doing economic activity around Samba.
  - Yes, we are Free Software people, but this is for *OUR* software, not yours.
    - The worst we're going to do is tease you about closed-source software.
  - Following the license is easy, and we will try and help you find a way to do what you need with our software (\*).

\* Unless what you need to do is violate our license and rip us off :-).



# Being successful with Samba: step one

- Let us (the Team) know that you exist and are planning to work with Samba.
  - We've usually heard most requirements before, and can often guide you in the right direction to help you decide if Samba is a good fit for you or not.
  - Meet with us to determine your needs
    - NDA's are possible.
- Decide where you fit on the spectrum of users.

# Being successful with Samba: Step two

- Work out if the license works for your product.
  - Samba is GPLv3.
  - Lawyers like to talk to lawyers: Ours are at <http://www.softwarefreedom.org/>.
  - Shipping the Samba source is the easiest way to comply with the license.
  - Allow users to update versions if updates are available for the product.
    - Note – this does not mean you have to *support* user-installed versions.

# Being successful with Samba: Step three

- Work out how to interface Samba with any proprietary value-added code you have.
  - Easiest route - stick your proprietary code in the kernel.
  - Interfacing with proprietary user-space code is harder, but possible.
  - Samba VFS modules are a good way to interface with external code.
- Talk to us. We will help you structure your code in order to achieve your goals.

# Being successful with Samba: Step four

- Work out your preferred level of engagement with the source code.
  - Consulting companies exist with *extensive* Samba experience, and the ability to customize any part of the code.
  - Companies in the past have set engineer performance targets “to become a Samba Team member by XX-XX-XXX date”.
  - Plan to integrate your code changes back into the main source tree.

# Being successful with Samba: Step five

- Decide how you want to get technical support.
  - Everyone will need technical support with Samba at some point.
  - Either do-it-yourself, with help from [samba-technical@samba.org](mailto:samba-technical@samba.org).
  - Or pay a consulting company / Linux distribution / secondary OEM to solve problems for you.
- Don't wait until you have a critical client issue before deciding on your support path.

# “But is it safe ?” : Security issues



# Security issues

- What happens on a Friday afternoon when someone posts a YouTube video showing what they claim is a zero-day exploit in Samba ?
  - Yes, this really happened to us.
- Samba has developer timezone coverage that gives us 24/7 monitoring of security issues on the [security@samba.org](mailto:security@samba.org) mailing list.
  - We take security very seriously.
  - We co-ordinate with the vendor-sec security lists and practice “responsible disclosure”.

# More on security

- Samba has had many security bugs over the years.
- What matters is not so much the bugs themselves, but the speed and accuracy of the fixes that are produced.
- This is something we have always prided ourselves on.
  - Hopefully an anecdote from the early days of Samba will illustrate.



# Certified Samba Security

- In January 2008, the US Software Quality control vendor Coverity, in conjunction with the US Department of Homeland security, scanned the Samba source code and certified Samba at Coverity “Rung 2” quality level.
  - No other SMB/CIFS/SMB2 server has been publicly assessed at this level of quality.
  - We actively scan for Coverity bugs, and fix them as soon as they are reported.

# Why work with Samba and the Samba Team ?

- Most commonly the technologies that Samba implements:
  - Aren't your core competency.
  - Are horribly complex.
  - Are needed for feature completeness of your product.
- The Samba Team has more experience in these technologies than anyone outside of Microsoft.
  - And in some of the older protocols, more than anyone inside Microsoft :-).

**But I don't want to work  
with Samba !**



What-me worry ?

# But I don't want to work with Samba !

- Ok, so you've made the wrong choice. Can happen to anyone :-).
  - At least work with smb torture and the client libraries to do torture testing on the SMB/CIFS/SMB2 server you decide to use.
  - Your competitors will be doing this, and if they're evil they may publish the results.
- Keep up with the Samba community to ensure you are aware of new tests and features.

# Questions and Comments ?

Email: [jra@samba.org](mailto:jra@samba.org)  
[jra@google.com](mailto:jra@google.com)