CWRAP

Testing your full software stack

May 15th, 2014

Andreas Schneider

Red Hat Inc.
Samba Team

Who am I?

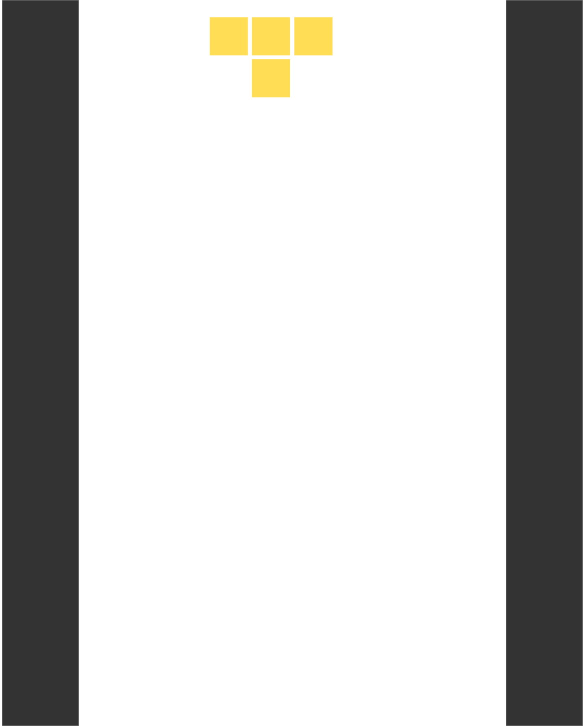
ABOUT ME

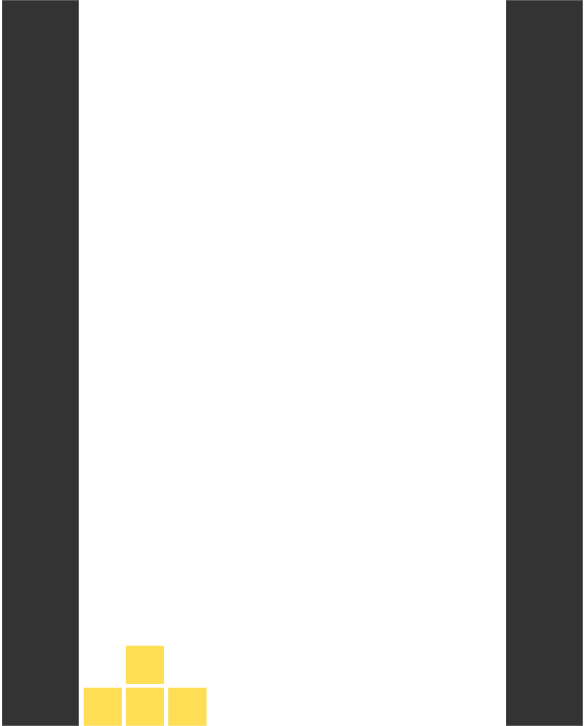
I'm a Free Software developer working on:

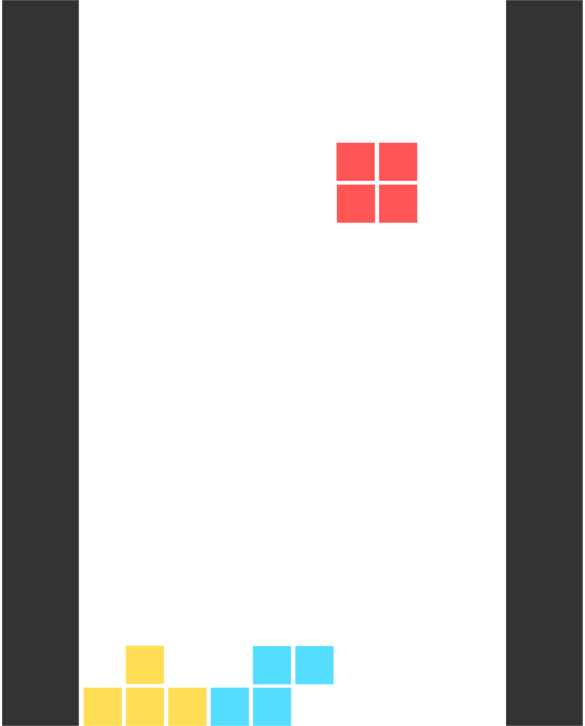
- → Samba - The domain controller and file server
- → FreeIPA - An integrated security information management solution for Linux
- → libssh - The SSH Library
- → cmocka - a unit testing framework for C
- → csync - a bidirectional client-only file synchronizer

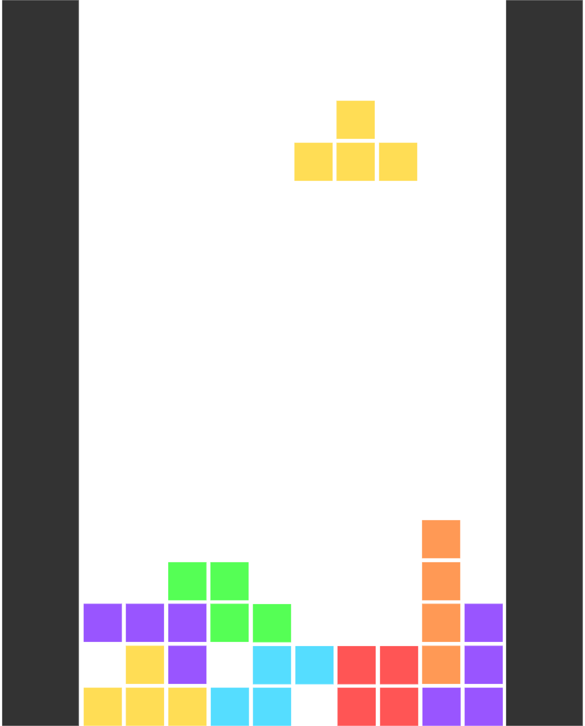
Twitter: @cryptomilk
Blog: **http://blog.cryptomilk.org/**
Git: **http://git.cryptomilk.org/**

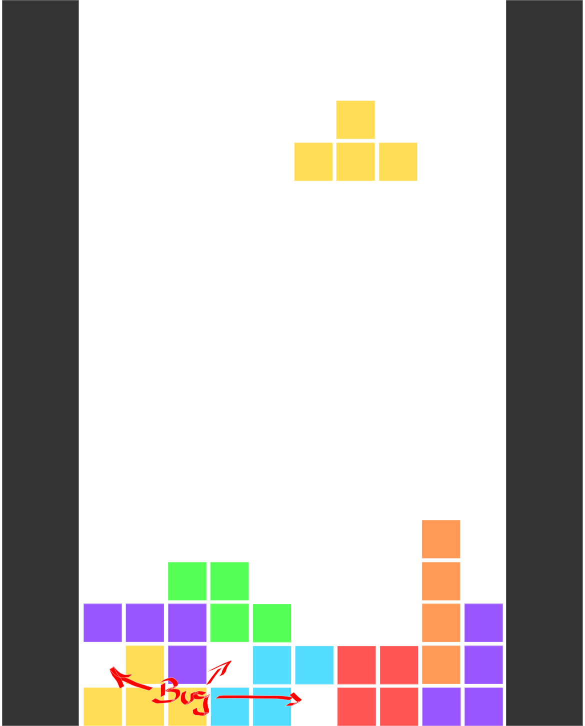
Do you remember this?

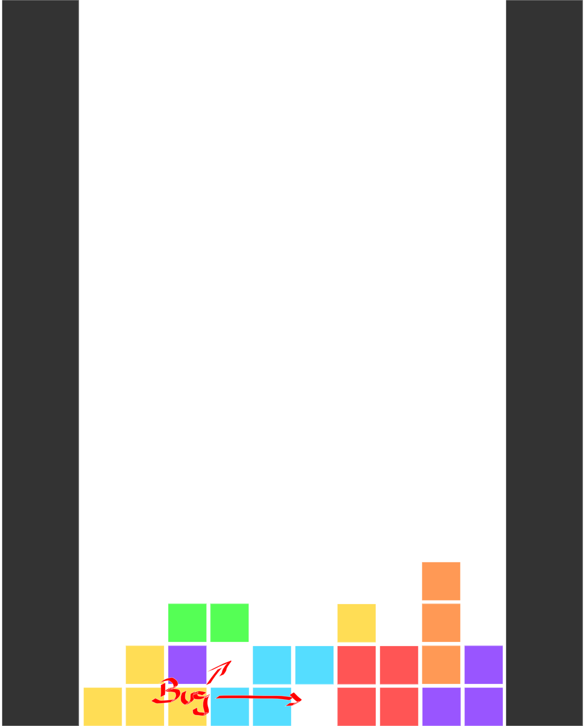






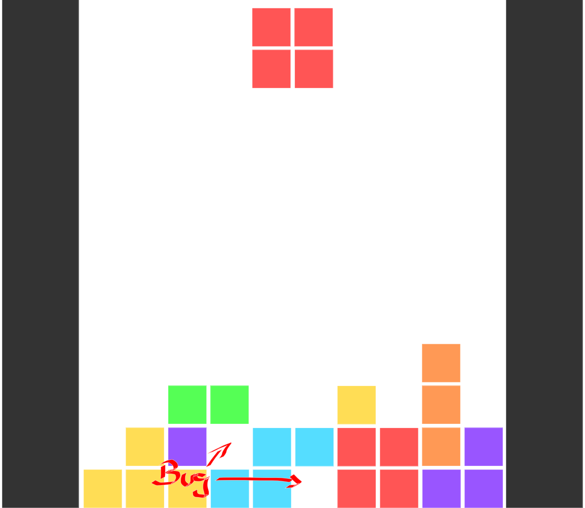



Bug

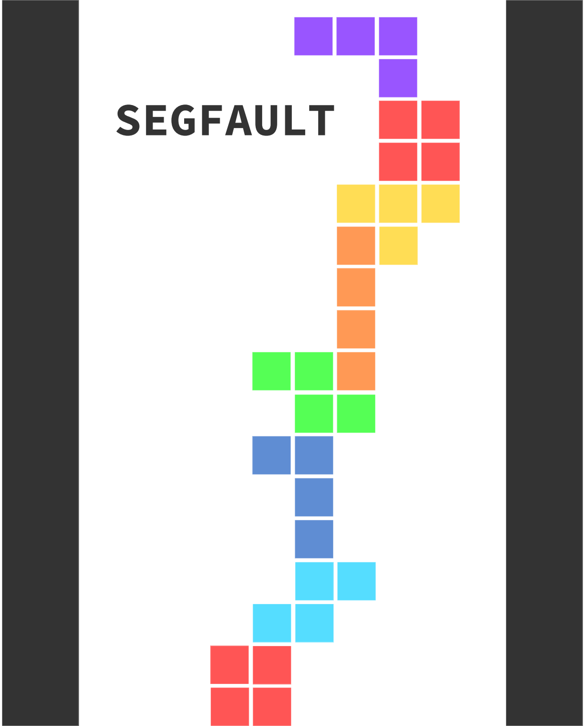


Bug

TEST YOUR CODE



SEGFAULT



WHAT IS THIS TALK ABOUT?

TETRIS?

Is this about Tetris?

SAMBA?

~~Is this about Tetris?~~
Is this about Samba?

THIS IS ABOUT CWRAP

~~Is this about Tetris?~~

(Is this about Samba?)

This is about new project called cwrap

WHY DO YOU NEED CWRAP?

WHAT DOES IT TRY TO SOLVE?

Let's assume

- → you have a client and a server application,
- → which requires Unix accounts,
- → and it requires a user change once logged in.

WHAT DOES IT TRY TO SOLVE?

Example: You want to test

→ a SSH client/server.

WHAT DOES IT TRY TO SOLVE?

What are the issues testing it?

→ Server requires to run on a privileged port.

→ You need a network with several machines for several servers.

→ A you need the same user accounts on each machine.

→ You need to be root to switch users.

WHAT DO WE TRY TO SOLVE?

You want to be able to test:

- → Everything on one machine.
- → Run everything as a normal user.
- → Run everything without network interfaces.
- → To just call 'make test'.

WHAT IS CWRAP?

WHAT IS CWRAP?

cwrap is..

a set of tools to create a fully isolated network environment to test client/server components on a single host, complete with synthetic account information, hostname resolution and privilege separation support. The heart of cwrap consists of three libraries you can preload to any executable.

WHAT IS CWRAP?

cwrap is ..

like The Matrix, where reality is simulated and everything is a lie

WHAT IS PRELOADING?

- → Preloading is a feature of the dynamic linker.
- → It is available on most Unices.
- → It loads a user specified, shared library before all others.
- → The library to preload is defined by the environment variable `LD_PRELOAD=libwurst.so`.
- → The symbols of the preloaded library are bound first.
- → See ›man ld.so‹

SYMBOL BINDING

Your application uses a function from a library.

1. Linker first looks if the application provides it.
2. Preloaded libraries are searched in the given order. `LD_PRELOAD=libwurst.so:libbrot.so`).
3. Libraries in linking order are searched (see `ldd /path/to/your/application`).

→ Trace with: `LD_DEBUG=symbols ls`

SYMBOL BINDING

Your application uses the function open(2).

1. Your application doesn't provide it.
2. `LD_PRELOAD=libcwrap.so` provides open(2).
3. Linked libc.so provides open(2).

→ The open(2) symbol from libcwrap.so gets bound!

THE PRELOADABLE LIBRARIES OF CWRAP

cwrap provides three libraries you can preload:

→ socket_wrapper
  Fakes socket communication

→ nss_wrapper
  Fakes nsswitch calls

→ uid_wrapper
  Fakes privilege separation

THE WRAPPERS IN DETAIL

SOCKET_WRAPPER

When preloaded and enabled:

- → Redirects all network communication to happen over Unix sockets.
- → Support for IPv4 and IPv6 socket and addressing emulation.
- → Ablility to capture network traffic in pcap format.

The idea and the first incarnation of socket_wrapper has been written by Jelmer Vernooij in 2005.
It made it possible to run the Samba torture suite against smbd in 'make test'.

SOCKET_WRAPPER

It wraps nearly all libc function responsible for socket communication.

- → socket(2), setsockopt(2), getsockopt(2)
- → bind(2), listen(2), accept(2)
- → write(2), read(2), writev(2), readv(2)
- → send(2), recv(2), sendto(2), recvfrom(2), sendmsg(2), recvmsg(2)
- → close(2)
- → ...

SOCKET_WRAPPER TODO

TODO or IN_PROGRESS

- → IP_PKTINFO support for recvmsg() (IN_PROGRESS).
- → Support for fd-passing in sendmsg()/recvmsg().
- → More tests.

SOCKET_WRAPPER DEMO

NSS_WRAPPER

When preloaded and enabled:

- → Provides account information for users and groups.
- → Can do network name resolution using a hosts file.
- → Allows loading and testing of NSS modules.

NSS_WRAPPER

It wraps most nsswitch functions.

- → getpwnam(3), getpwnam_r(3), getpwuid(3), getpwuid_r(3)
- → getgrnam(3), getgrnam_r(3), getgrgid(3), getgrgid_r(3)
- → initgroups(3)
- → getaddrinfo(3), getnameinfo(3)
- → gethostbyname(3), gethostbyname_r(3), gethostbyaddr(3), gethostbyaddr_r(3)
- → gethostname(3)
- → ...

NSS_WRAPPER TODO

→ Support for hosts nsswitch modules.

NSS_WRAPPER DEMO

UID_WRAPPER

When preloaded and enabled:

- → Allows uid switching as a normal user.
- → You can start any application making it believe it is running as root.
- → Support for user/group changing in the local thread using the syscalls (like glibc).

uid_wrapper has been completely rewritten to support the last two features.

UID_WRAPPER

It wraps most user switching functions

- → setuid(2), seteuid(2), setreuid(2), setresuid(2)
- → getuid(2), geteuid(2)
- → setgid(2), setegid(2), setregid(2), setresgid(2)
- → getgid(2), getegid(2)
- → setgroups(2), getgroups(2)
- → ...

UID_WRAPPER DEMO

HOW ARE THE WRAPPERS TESTED?

The wrappers are all tested using cmocka, a unit testing framwork for C with support for mock objects (**http://cmocka.org**).

- → You can simply test them using 'make test'.
- → All wrappers have a code coverage above 75%.
- → There are nightly build of the master branch.
- → Testing Dashboard: **http://mock.cryptomilk.org**

SUPPORTED PLATFORMS

- → Linux (openSUSE, CentOS)
- → BSD (FreeBSD)
- → Solaris (OpenIndiana)

If you want a platform supported, create a nightly build and submit results to the dashboard.

WHO USES CWRAP?

- → Samba (Doesn't use the preloaded version yet)
- → MIT KRB5 (Work in progress by Nalin Dahyabhai)
- → libssh (Test libssh against OpenSSH, I'm working on it)
- → SSSD (Jakub Hrozek wants to use in SSSD)

LINKS

- → Website: **http://cwrap.org/**
- → Mailinglist: **https://lists.samba.org/mailman/listinfo/samba-technical**
- → IRC: #cwrap @ freenode

Questions & Answers

http://cwrap.org/
http://cmocka.org/