

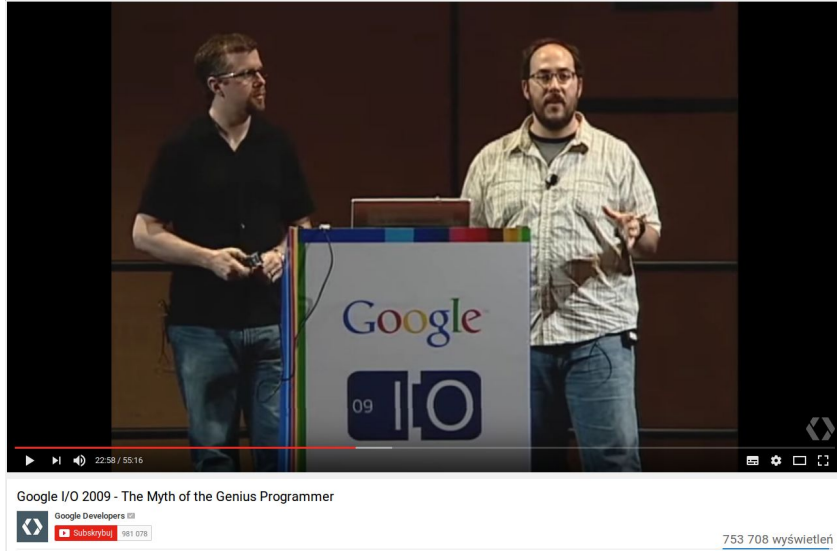
# Linux, Docker & GitLab CI

Łukasz Hryniuk

# Agenda

1. Bit about trust.
2. Overview of our project.
3. Tools: fabric, ccache.
4. Jenkins.
5. GitLabCI configuration.
6. GitLabCI++.
7. Q&A.

# Trust vs ~~control~~ constraints



The Myth of the Genius Programmer



One Hacker Way

# Server

# Problems we had

- manual building and testing
- manual environment configuration
- manual deploy
- one DEV environment

# Step-by-step solution

1. FreeBSD + Jenkins + shell
2. FreeBSD/Linux + Jenkins/GitLabCI + shell
3. FreeBSD/Linux + Jenkins/GitLabCI + Puppet
4. Linux + GitLabCI + Puppet (**today!**, 25.10)

# Fabric - execute remote tasks automatically

```
server/terminal_instalka/Makefile
server/terminal_instalka/getter.cpp
server/terminal_instalka/script
Aug 4 10:17:57 dev-compile0: Last login: Thu Aug 4 10:17:57 dev-compile0
Aug 4 10:17:58 dev-compile0: [09]mysql_conn
Aug 4 10:17:58 dev-compile0: scons -j4
Aug 4 10:17:58 dev-compile0: scons: Reading SConscript files...
Aug 4 10:17:58 dev-compile0: scons: done reading SConscript files.
Aug 4 10:17:58 dev-compile0: scons: Building targets...
Aug 4 10:17:58 dev-compile0: sent 55,891,231 bytes to 192.168.8.27:8125
Aug 4 10:17:58 dev-compile0: total size is 98,783,276
Aug 4 10:17:58 dev-compile0: rsync -avz server dev-compile0:~|
```



```
Note:
fab branch:develop sync build cpbin deploy syslog:grepfor='lhryniuk'
fab branch:develop sync build deploy syslog:grepfor='lhryniuk'
fab cpbin
fab deploy
fab names
fab sync
fab sync build
fab sync build:master_ut
fab sync build:master_ut test:master_ut
fab sync build deploy syslog
fab syslog
fab syslog:grepfor='lhryniuk'
fab syslog:lhryniuk
fab ssh-add ~/.ssh/id_rsa
```

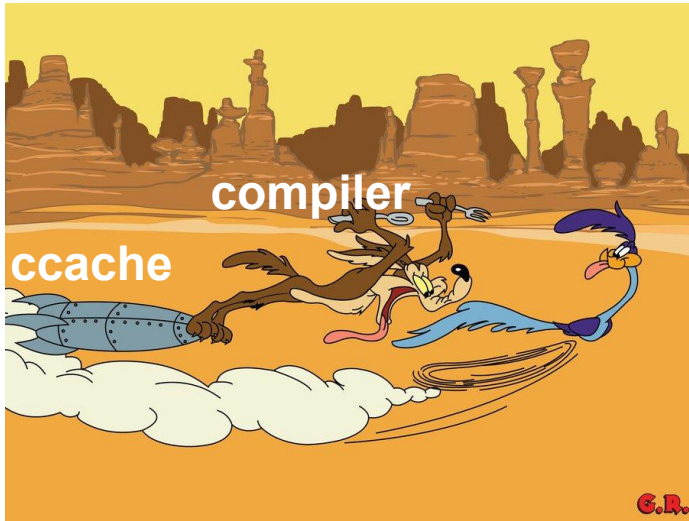
# Compilation





# ccache

- compiler cache for C family of languages
- keep object files between compilations
  - even if you run 'clean' compilation
- speed up dependent on number of changes
  - but for unchanged files stored obj's are used



# ccache

compilation times			
project	without ccache	with ccache	speed up
<b>server</b>	11m15.851s	0m45.086s	~1500%
<b>master_ut</b>	01m27.746s	0m08.707s	~1000%
<b>segment_ut</b>	10m09.432s	0m33.681s	~1800%

So where is this CI?



# Jenkins...

- 24.06 - 12.07 (18 days), 13 builds, **NO successes**
- compilation started only when MR is opened

# ...resurrected!

## Fixes

- **set correct version of dependencies**
- new configuration for GitLab build triggers

## Results

- 13.07 - 01.08 (**18 days**) 70 builds, **63 successes**

# Jenkins :(

- difficult to configure - everything requires a plugin
- UT coverage report only in XML
- only one job running at any point
- binaries are hard to find


# New sprint, Linux, “accidental” root access

- server must be migrated from FreeBSD to Linux
- all tests and regressions must run on new platform
- better compilation result notifications were needed (**the reason for checking GitLab CI**)
- new host to test server in Linux env (and subsequent **root access**)

# GitLab CI - installation

- `curl -L https://packages.gitlab.com/install/repositories/runner/gitlab-ci-multi-runner/script.rpm.sh | sudo bash`
- `sudo yum install gitlab-ci-multi-runner`
- `sudo gitlab-ci-multi-runner register`

## How to setup a new project specific runner

1. Install GitLab Runner software. Checkout the [GitLab Runner section](#) to install it
2. Specify following URL during runner setup: `http://gitlab/ci`
3. Use the following registration token during setup: 
4. Start runner!



## ...and configuration: **.gitlab-ci.yml**

```
segment_ut:
  stage: test
  script:
    - scons TEST_COVERAGE=TRUE CCACHE=TRUE -j4 segment_ut
    - ./tests/Segment/segment_ut --gtest_shuffle --gtest_output='xml:segment_ut.junit.xml'
    - chmod a+x ./tools/gcovr/gcovr
    - ./tools/gcovr/gcovr -p -r . -e tests -e messenger_common -e common --html -o segment_ut_cov.html
    - ./tools/gcovr/gcovr -p -r . -e tests -e messenger_common -e common
  artifacts:
    paths:
      - segment_ut_cov.html
      - segment_ut.junit.xml
    name: "${CI_BUILD_NAME}_${CI_BUILD_REF}"
    expire_in: 1 week
```

# First build

Build #23730 for commit 5a592ec2 from proto/gitlab-ci-test

✓ build\_server ✓ master\_ut ✗ segment\_ut ✓ worker\_ut

passed 2 minutes 40 seconds 7 days ago

**dependent builds**

Running with gitlab-ci-multi-runner 1.4.0 (5d9b2f)  
Using Shell executor...  
Running on  
Fetching changes...  
Removing .sconsign.dblite  
Removing Workers/Contacts/WorkerContacts  
Removing Workers/Contacts/build\_worker\_soa/  
Removing Workers/Files/WorkerFiles  
Removing Workers/Files/build\_worker\_files/  
Removing master/master  
Removing master/obj/  
Removing messenger\_common/  
Removing rating/obj/  
Removing rating/rating  
Removing segment/obj/  
Removing segment/segment  
HEAD is now at 3b2a0e0 Added .gitlab.yml  
From http://gitlab/trans-messenger/server  
+ 3b2a0e0...5a592ec proto/gitlab-ci-test -> origin/proto/gitlab-ci-test (forced update)  
Checking out 5a592ec2 as proto/gitlab-ci-test...  
\$ cp -R /messenger\_common .  
\$ cd messenger\_common && git checkout 'cat ../common.required' && cd ..  
Already on 'ccache\_on\_66'  
\$ scons CCACHE=TRUE -j4  
scons: Reading SConscript files ...  
scons: done reading SConscript files.  
scons: Building targets ...  
ccache g++-4.8 -o master/obj/messenger\_common/src/3rdParty/7zipSdk/Alloc.o -c -std=c++1y -O0 -g -Wall -Wextra -Werror -finput-cha  
rset=UTF-8 -D 7ZIP\_ST -I/usr/include/mysql -I. -isystem/usr/include/queue messenger\_common/src/3rdParty/7zipSdk/Alloc.cpp  
ccache g++-4.8 -o master/obj/messenger\_common/src/3rdParty/7zipSdk/LzFind.o -c -std=c++1y -O0 -g -Wall -Wextra -Werror -finput-cha  
rset=UTF-8 -D 7ZIP\_ST -I/usr/include/mysql -I. -isystem/usr/include/queue messenger\_common/src/3rdParty/7zipSdk/LzFind.cpp  
ccache g++-4.8 -o master/obj/messenger\_common/src/3rdParty/7zipSdk/LzmaDec.o -c -std=c++1y -O0 -g -Wall -Wextra -Werror -finput-c  
harset=UTF-8 -D 7ZIP\_ST -I/usr/include/mysql -I. -isystem/usr/include/queue messenger\_common/src/3rdParty/7zipSdk/LzmaDec.cpp  
ccache g++-4.8 -o master/obj/messenger\_common/src/3rdParty/7zipSdk/LzmaEnc.o -c -std=c++1y -O0 -g -Wall -Wextra -Werror -finput-c  
harset=UTF-8 -D 7ZIP\_ST -I/usr/include/mysql -I. -isystem/usr/include/queue messenger\_common/src/3rdParty/7zipSdk/LzmaEnc.cpp  
ccache g++-4.8 -o master/obj/messenger\_common/src/3rdParty/7zipSdk/LzmaLib.o -c -std=c++1y -O0 -g -Wall -Wextra -Werror -finput-c

output

## commit info

Commit 5a592ec2  
Branch: proto/gitlab-ci-test  
Author: lhryniuk  
Message: Added .gitlab.yml

### 3 other builds for

5a592ec2:		
✓	worker_ut	success
✗	segment_ut	failed
✓	master_ut	success

Commits 4,912 Network Compare Branches 36 Tags 94

proto/gitlab-ci-test - server

+ Create Merge Request Filter by commit message

29 Jul, 2016	Test fpm	✗ f2056d2c
1 commit	by Łukasz Hryniuk 3 days ago	Browse Files
27 Jul, 2016	Added printing coverage of common_ut	✗ b93c5828
2 commits	by Łukasz Hryniuk 5 days ago	Browse Files
	Added .gitlab.yml	✗ ca893d01
	by Łukasz Hryniuk 5 days ago	Browse Files
26 Jul, 2016	Added common branch with ccache enabled.	b6be9a56
2 commits	by Łukasz Hryniuk 6 days ago	Browse Files
	Prevented premature free() call in segment_ut.	5925ded3
	by Łukasz Hryniuk 6 days ago	Browse Files
25 Jul, 2016	Fixed free() call on a pointer used in a test.	28df4d8
4 commits	by Łukasz Hryniuk 6 days ago	Browse Files
	Fixed time setting in EmailTagsCreatorTester.	3361be11
		Browse Files

## tests

✓ passed	#23733	5a592ec2	proto/gitlab-ci-test	test	worker_ut	20 seconds	7 days ago	C
✗ failed	#23732	5a592ec2	proto/gitlab-ci-test	test	segment_ut	2 minutes 7 seconds	7 days ago	C
✓ passed	#23731	5a592ec2	proto/gitlab-ci-test	test	master_ut	25 seconds	7 days ago	C
✓ passed	#23730	5a592ec2	proto/gitlab-ci-test	test	build_server	2 minutes 40 seconds	7 days ago	C

## Select executor

- [Help me select executor](#)
- [Shell](#)
- [Docker and Docker-SSH](#)
- [Parallels](#)
- [VirtualBox](#)
- [SSH](#)



# Dockerfile and more gitlab-runners

```
1 FROM centos:latest
2
3 RUN yum groupinstall -y "Development Tools" && \
4     yum -y install scons \
5         libstdc++-static \
6         libkqueue-devel \
7         mariadb-devel \
8         libicu-devel \
9         libmemcached-devel \
10        boost-devel \
11        python-devel \
12        ccache \
13        which \
14        ruby-devel && gem install fpm
```

1b8bb210 

lhryniuk-local

Remove runner

#12

● bdfd1bc7 

Remove runner









#10

● b43c10f0 

Remove runner

#9

**^TOTAL.\*\s+(\d+\%)\$** + a bit of code =

✓ passed	#24934	a74ff413	proto/gitlab-ci-test	test	worker_ut	49 seconds	4 days ago	71.0%	
✗ failed	#24933	a74ff413	proto/gitlab-ci-test	test	segment_ut	3 minutes 13 seconds	4 days ago	19.0%	
✓ passed	#24932	a74ff413	proto/gitlab-ci-test	test	master_ut	56 seconds	4 days ago	13.0%	
✓ passed	#24922	a74ff413	proto/gitlab-ci-test	build	worker-files	42 seconds	4 days ago		
✓ passed	#24921	a74ff413	proto/gitlab-ci-test	build	worker-contactlist	1 minute 9 seconds	4 days ago		
✓ passed	#24920	a74ff413	proto/gitlab-ci-test	build	segment	3 minutes 7 seconds	4 days ago		
✓ passed	#24919	a74ff413	proto/gitlab-ci-test	build	rating	37 seconds	4 days ago		
✓ passed	#24918	a74ff413	proto/gitlab-ci-test	build	master	51 seconds	4 days ago		

coverage!

artifacts!

GitLab Messenger · GitLab · 11:42

██████████ pushed to branch `task/blocking_socket_function_test` of `trans-messenger/segment` ([Compare changes](#))  
- made test more bdd style (7ef1c2)

GitLab Messenger · GitLab · 11:44

██████████ pushed to branch `task/blocking_socket_function_test` of `trans-messenger/segment` ([Compare changes](#))  
- Made test more bdd style. (b1af00)  
- Removed redundant space. (85212a)

GitLab Messenger · GitLab · 11:48

██████████ pushed new branch `task/performance-test-login` to `trans-messenger/segment`

GitLab Messenger · GitLab · 11:49

██████████ removed branch `task/performance-test-login` from `trans-messenger/segment`

GitLab Messenger · GitLab · 11:52

`trans-messenger/segment`: Commit 7ef1c206 of `task/blocking_socket_function_test` branch ██████████ in 596 second(s)

GitLab Messenger · GitLab · 11:56

`trans-messenger/segment`: Commit 85212a9d of `task/blocking_socket_function_test` branch ██████████ in 696 second(s)

GitLab Messenger · GitLab · 11:58

`trans-messenger/segment`: Commit 79fde76d of `task/performance-test-login` branch ██████████ in 623 second(s)

HipChat notifications

# What we had after first stage?

- several builds concurrently, inside Docker container
- build on push
- binaries are built separately
- tests are built and run only after success of the previous stage
- code coverage next to build status
- RPMs generation
- test artifacts in XML and HTML
- notifications about commits and builds
- **we can build binaries on Windows and run it on Linux**

# Second stage

- Docker image in our own registry
- configs pull - Puppet
- RPM auto-push to the repo
- local DEV environment: Vagrant/Docker

# More: Beamer slides generation and docs ;-)

Status	Build ID	Commit	Ref	Stage	Name	Duration	Finished at	
✓ passed	#39749	e0d197ee	master	test	pdf	8 seconds	11 days ago	📄 ↻
✗ failed	#39748	725aafd9	master	test	pdf	3 seconds	11 days ago	↻

Name	Size
📄 presentation.pdf	11 KB





# What we want to have

- SonarQube integration (test results/coverage/static code analysis)
- Coding Standards checks (git hooks maybe - **too much control(?)**)
- ...and autonotification about new package ready to deploy
- auto deploy and functional testing
- changelog generation
- more Docker (Docker swarm?)

# Questions?

# Thanks!