Reproducible Builds All The Way Down

Vagrant Cascadian <vagrant@reproducible-builds.org>

OSFC 2023-10-11

Who am I

	Vagrant
debian user	2001
debian developer	2010
reproducible builds	2015

Reproducible Builds

`https://reproducible-builds.org/docs/definition/`

A build is reproducible if given the same source code, build environment and build instructions, any party can recreate bit-by-bit identical copies of all specified artifacts.

Spelling it out

Reproducible Builds provides...

- strong confidence...

Spelling it out

Reproducible Builds provides...

- strong confidence...
- that a binary was produced from a given source...

Spelling it out

Reproducible Builds provides...

- strong confidence...
- that a binary was produced from a given source...
- ...probably!

Freedom to iterate

Benefits of Reproducible Builds

- ...

Freedom to iterate

Benefits of Reproducible Builds

- ...
- Security

Freedom to iterate

Benefits of Reproducible Builds

- ...
- Security
- Code refactoring

Freedom to iterate

Benefits of Reproducible Builds

- ...
- Security
- Code refactoring
- Build Caching

Scope: Debian

Debian

- The Universal Operating System

Scope: Debian

Debian

- The Universal Operating System
- ~34000 source packages and counting

Scope: Debian

Debian

- The Universal Operating System
- ~34000 source packages and counting
- 380 million lines of code ... and counting!

Scope: Debian

Debian

- The Universal Operating System
- ~34000 source packages and counting
- 380 million lines of code ... and counting!
- ~95% reproducible

Scope: Open Source Firmware

Firmware projects

- hardware specific

Scope: Open Source Firmware

Firmware projects

- hardware specific
- often limited functionality and scope

Scope: Open Source Firmware

Firmware projects

- hardware specific
- often limited functionality and scope
- smaller size, quicker development cycles

Scope: Open Source Firmware

Firmware projects

- hardware specific
- often limited functionality and scope
- smaller size, quicker development cycles
- 100% reproducible is achievable

Open Source Firmware, yay!

Open Source Firmware projects make a great showcase for the viability of 100% reproducibility!

Approaching 100% reproducible

- Coreboot 100%

Approaching 100% reproducible

- Coreboot 100%
- U-Boot 100% (more later)

Approaching 100% reproducible

- Coreboot 100%
- U-Boot 100% (more later)
- OpenSBI 100% (more later)

Approaching 100% reproducible

- Coreboot 100%
- U-Boot 100% (more later)
- OpenSBI 100% (more later)
- TrustedFirmware-A 100% (more later)

Approaching 100% reproducible

- Coreboot 100%
- U-Boot 100% (more later)
- OpenSBI 100% (more later)
- TrustedFirmware-A 100% (more later)
- OpenWRT 96%-100%

Approaching 100% reproducible

- Coreboot 100%
- U-Boot 100% (more later)
- OpenSBI 100% (more later)
- TrustedFirmware-A 100% (more later)
- OpenWRT 96%-100%
- Yocto 99.98%

Happy U-Boot Anniversary, Vagrant!

https://bugs.debian.org/726699

On Thu, Oct 17, 2013 at 09:48:50PM -0700, Vagrant Cascadian wrote:
> A newer version of u-boot is available (2013.10), with support for a few new
> platforms such as BeagleBone Black and Wandboard...
>
> I've done some work towards updating to the new version and could help with
> an upload or two if needed... but probably not long-term maintenance.

There's basically no one willing to do long-term maintenance, so
have a blast!

U-Boot, the great enticement

Debian U-Boot packages were marked reproducible
`https://alioth-lists.debian.net/pipermail/reproducible-builds/Week-of-Mon-20150727/002492.html`

- ...

U-Boot, the great enticement

Debian U-Boot packages were marked reproducible
`https://alioth-lists.debian.net/pipermail/reproducible-builds/Week-of-Mon-20150727/002492.html`

- ...
- I knew there were timestamps!

U-Boot, the great enticement

Debian U-Boot packages were marked reproducible
`https://alioth-lists.debian.net/pipermail/reproducible-builds/Week-of-Mon-20150727/002492.html`

- ...
- I knew there were timestamps!
- Reproducible Builds tests only performed on x86

U-Boot, the great enticement

Debian U-Boot packages were marked reproducible
`https://alioth-lists.debian.net/pipermail/reproducible-builds/Week-of-Mon-20150727/002492.html`

- ...
- I knew there were timestamps!
- Reproducible Builds tests only performed on x86
- Started with two or three little arm boards...

U-Boot, the great enticement

Debian U-Boot packages were marked reproducible
`https://alioth-lists.debian.net/pipermail/reproducible-builds/Week-of-Mon-20150727/002492.html`

- ...
- I knew there were timestamps!
- Reproducible Builds tests only performed on x86
- Started with two or three little arm boards...
- Reproducible Builds Zoo!

U-boot: Timestamps

5847084f6bbd0778afb29f0574085d4210ea8cff
Respect SOURCE_DATE_EPOCH when building FIT images.

Deterministic time?

SOURCE_DATE_EPOCH
`https://reproducible-builds.org/docs/source-date-epoch/`
Supported in GCC, Clang, and more!

U-Boot: Revenge of Timestamps

8664ab7debabfb6e1049c81030c2a18fd3eecb58
Set time and umask on multi-dtb fit images to ensure reproducibile builds.

- ...

U-Boot: Revenge of Timestamps

8664ab7debabfb6e1049c81030c2a18fd3eecb58
Set time and umask on multi-dtb fit images to ensure reproducibile builds.

- ...
- gzip has –no-name

U-Boot: Revenge of Timestamps

8664ab7debabfb6e1049c81030c2a18fd3eecb58
Set time and umask on multi-dtb fit images to ensure reproducibile builds.

- ...
- gzip has –no-name
- lzop has no way around it

U-Boot: Revenge of Timestamps

8664ab7debabfb6e1049c81030c2a18fd3eecb58
Set time and umask on multi-dtb fit images to ensure reproducibile builds.

- ...
- gzip has –no-name
- lzop has no way around it
- a light "touch" fixes future compression choices!

U-Boot: Revenge of the Revenge of the Timestamps

878e2a50b50199cb06ee28df53151e396a29d838
Set time and umask on fit-dtb.blob to ensure reproducibile builds.

- Sound familiar?

U-Boot: Revenge of the Revenge of the Timestamps

878e2a50b50199cb06ee28df53151e396a29d838
Set time and umask on fit-dtb.blob to ensure reproducibile builds.

- Sound familiar?
- New files...

U-Boot: Revenge of the Revenge of the Timestamps

878e2a50b50199cb06ee28df53151e396a29d838
Set time and umask on fit-dtb.blob to ensure reproducibile builds.

- Sound familiar?
- New files...
- ...identical problems!

Noooo Timestamps

There are no timestamps like...
No Timestamps!

U-Boot: build paths

https://patchwork.ozlabs.org/project/uboot/patch/20220818173133.12552-1-vagrant@debian.org/

Makefile: Use relative paths for debugging symbols.

- -ffile-prefix-map and –debug-prefix-map

U-Boot: build paths

https://patchwork.ozlabs.org/project/uboot/patch/20220818173133.12552-1-vagrant@debian.org/

Makefile: Use relative paths for debugging symbols.

- -ffile-prefix-map and –debug-prefix-map
- Worked around in Debian u-boot packages.

U-Boot: build paths

https://patchwork.ozlabs.org/project/uboot/patch/20220818173133.12552-1-vagrant@debian.org/

Makefile: Use relative paths for debugging symbols.

- -ffile-prefix-map and –debug-prefix-map
- Worked around in Debian u-boot packages.
- Fixed in upstream gcc 13 https://gcc.gnu.org/bugzilla/show_bug.cgi?id=93371

U-Boot: Running kernel

3a0654ecd0d6a39406e6fe91f7a40ce589594ae9
efi_loader: correctly identify binary name

- ...

U-Boot: Running kernel

3a0654ecd0d6a39406e6fe91f7a40ce589594ae9
efi_loader: correctly identify binary name

- ...
- Embedded BOOTX64.EFI or BOOTIA32.EFI

U-Boot: Running kernel

3a0654ecd0d6a39406e6fe91f7a40ce589594ae9
efi_loader: correctly identify binary name

- ...
- Embedded BOOTX64.EFI or BOOTIA32.EFI
- Used host architecture rather than target architecture

U-Boot: randomness

aaa91a4e4b8a5d74f1317e18aa47d2a7a72e0c43
fit_image: Use calloc() to fix reproducibility issue

- ...

U-Boot: randomness

aaa91a4e4b8a5d74f1317e18aa47d2a7a72e0c43
fit_image: Use calloc() to fix reproducibility issue

- ...
- Unitialized memory areas embedded into the build

U-Boot: locale

42ffa51fd46bc6fd4bf2c244f00a80df31d01596
Use C locale when setting CC_VERSION_STRING and LD_VERSION_STRING.

- ...

U-Boot: locale

42ffa51fd46bc6fd4bf2c244f00a80df31d01596
Use C locale when setting CC_VERSION_STRING and LD_VERSION_STRING.

- ...
- Only unreproducible when italitian locales used!!

U-Boot: locale

42ffa51fd46bc6fd4bf2c244f00a80df31d01596
Use C locale when setting CC_VERSION_STRING and LD_VERSION_STRING.

- ...
- Only unreproducible when italitian locales used!!
- Tried many, many other locales

U-Boot: locale

42ffa51fd46bc6fd4bf2c244f00a80df31d01596
Use C locale when setting CC_VERSION_STRING and LD_VERSION_STRING.

- ...
- Only unreproducible when italitian locales used!!
- Tried many, many other locales
- GNU ld (GNU Binutils for Debian) 2.26 ...

U-Boot: locale

42ffa51fd46bc6fd4bf2c244f00a80df31d01596
Use C locale when setting CC_VERSION_STRING and LD_VERSION_STRING.

- ...
- Only unreproducible when italitian locales used!!
- Tried many, many other locales
- GNU ld (GNU Binutils for Debian) 2.26 ...
- ld di GNU (GNU Binutils for Debian) 2.26

U-Boot: locale

42ffa51fd46bc6fd4bf2c244f00a80df31d01596
Use C locale when setting CC_VERSION_STRING and LD_VERSION_STRING.

- ...
- Only unreproducible when italitian locales used!!
- Tried many, many other locales
- GNU ld (GNU Binutils for Debian) 2.26 ...
- ld di GNU (GNU Binutils for Debian) 2.26
- Only language that happened to have translations for the relevent string

opensbi

https://github.com/riscv-software-src/opensbi/pull/229
Build paths embedded, new uses of FILE

- ...

opensbi

https://github.com/riscv-software-src/opensbi/pull/229
Build paths embedded, new uses of FILE

- ...
- Proposed -ffile-prefix-map to workaround

opensbi

https://github.com/riscv-software-src/opensbi/pull/229
Build paths embedded, new uses of FILE

- ...
- Proposed -ffile-prefix-map to workaround
- Actually fixed by improved error handling not using FILE

trustedfirmware-a (or did you say arm-trusted-firmware)

Also embedding build paths
debian/rules: Use -ffile-prefix-map in TF_CFLAGS.

- ...

trustedfirmware-a (or did you say arm-trusted-firmware)

Also embedding build paths
debian/rules: Use -ffile-prefix-map in TF_CFLAGS.

- ...
- Regression and only partially fixed at the moment

Types of reproducibility issues discovered

Let us Review...

- ...

Types of reproducibility issues discovered

Let us Review...

- ...
- Timestamps

Types of reproducibility issues discovered

Let us Review...

- ...
- Timestamps
- Build Paths

Types of reproducibility issues discovered

Let us Review...

- ...
- Timestamps
- Build Paths
- Locale dependent strings

Types of reproducibility issues discovered

Let us Review...

- ...
- Timestamps
- Build Paths
- Locale dependent strings
- Also Timestamps

Types of reproducibility issues discovered

Let us Review...

- ...
- Timestamps
- Build Paths
- Locale dependent strings
- Also Timestamps
- Let us not forget timestamps!

Types of reproducibility issues discovered

Let us Review...

- ...
- Timestamps
- Build Paths
- Locale dependent strings
- Also Timestamps
- Let us not forget timestamps!
- `https://reproducible-builds.org/docs/env-variations/`

diffocope

`https://diffoscope.org`

- Recursive and human-readable "diff"

diffocope

`https://diffoscope.org`

- Recursive and human-readable "diff"
- locates and diagnoses reproducibility issues

diffoscope

`https://diffoscope.org`

- Recursive and human-readable "diff"
- locates and diagnoses reproducibility issues
- used for analysing why something is reproducible!

diffoscope

`https://diffoscope.org`

- Recursive and human-readable "diff"
- locates and diagnoses reproducibility issues
- used for analysing why something is reproducible!
- not used for determining whether something is reproducible!

diffoscope example

51431	13611);	51438	13542);
51432	INSERT INTO "targets" VALUES('ttu.ee',13611);	51439	INSERT INTO "targets" VALUES('ttu.ee',13542);
51433	[9300 lines removed]	51440	[9314 lines removed]
60733	CREATE TABLE git_commit	60754	CREATE TABLE git_commit
60734	(git_commit TEXT);	60755	(git_commit TEXT);
60735	INSERT INTO "git_commit" VALUES('cd09fb8c2161a8d1280b848eaab3b14d35fe3044');	60756	INSERT INTO "git_commit" VALUES('e78fe5d803208bf6c877dc675cdb4f1b719e7519');
60736	COMMIT;	60757	COMMIT;

install.rdf

Offset 5, 15 lines modified		Offset 5, 15 lines modified	
5	<Description about="urn:mozilla:install-manifest">	5	<Description about="urn:mozilla:install-manifest">
6	<em:name>HTTPS-Everywhere</em:name>	6	<em:name>HTTPS-Everywhere</em:name>
7	<em:creator>Mike Perry, Peter Eckersley, & Yan Zhu</em:creator>	7	<em:creator>Mike Perry, Peter Eckersley, & Yan Zhu</em:creator>
8	<em:aboutURL>chrome://https-everywhere/content/about.xul</em:aboutURL>	8	<em:aboutURL>chrome://https-everywhere/content/about.xul</em:aboutURL>
9	<em:id>https-everywhere@eff.org</em:id>	9	<em:id>https-everywhere@eff.org</em:id>
10	<em:type>2</em:type> <!-- type: Extension -->	10	<em:type>2</em:type> <!-- type: Extension -->
11	<em:description>Encrypt the Web! Automatically use HTTPS security on many sites.</em:description>	11	<em:description>Encrypt the Web! Automatically use HTTPS security on many sites.</em:description>
12	<em:version>5.0.6</em:version>	12	<em:version>5.0.7</em:version>
13	<em:multiprocessCompatible>true</em:	13	<em:multiprocessCompatible>true</em:

diffoscope, supported file types

Android APK files, Android boot images, Ar(1) archives, Berkeley DB database files, Bzip2 archives, Character/block devices, ColorSync colour profiles (.icc), Coreboot CBFS filesystem images, Cpio archives, Dalvik .dex files, Debian .buildinfo files, Debian .changes files, Debian source packages (.dsc), Device Tree Compiler blob files, Directories, ELF binaries, Ext2/ext3/ext4/btrfs filesystems, FreeDesktop Fontconfig cache files, FreePascal files (.ppu), Gettext message catalogues, GHC Haskell .hi files, GIF image files, Git repositories, GNU R database files (.rdb), GNU R Rscript files (.rds), Gnumeric spreadsheets, Gzipped files, ISO 9660 CD images, Java .class files, JavaScript files, JPEG images, JSON files, LLVM IR bitcode files, MacOS binaries, Microsoft Windows icon files, Microsoft Word .docx files, Mono 'Portable Executable' files, Ogg Vorbis audio files, OpenOffice .odt files, OpenSSH public keys, OpenWRT package archives (.ipk), PDF documents, PGP signed/encrypted messages, PNG images, PostScript documents, RPM archives, Rust object files (.deflate), SQLite databases, SquashFS filesystems, Statically-linked binaries, Symlinks, Tape archives (.tar), Tcpdump capture files (.pcap), Text files, TrueType font files, XML binary schemas (.xsb), XML files, XZ compressed files, etc.

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux
- GNU Guix

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux
- GNU Guix
- NixOS

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux
- GNU Guix
- NixOS
- FreeBSD

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux
- GNU Guix
- NixOS
- FreeBSD
- NetBSD

try diffoscope

`https://diffoscope.org`

Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux
- GNU Guix
- NixOS
- FreeBSD
- NetBSD
- Homebrew

try diffoscope

`https://diffoscope.org`

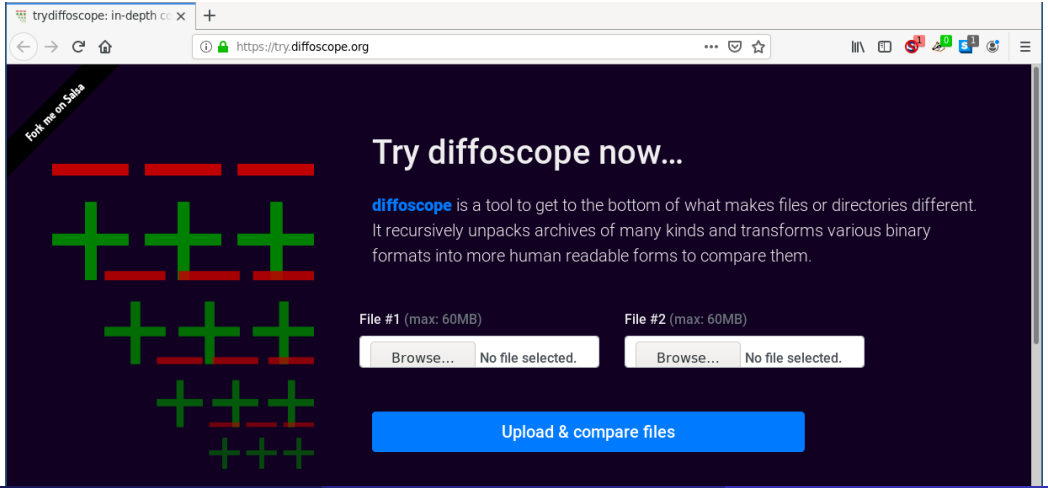
Available on many platforms:

- Debian
- Fedora
- OpenSUSE
- Archlinux
- GNU Guix
- NixOS
- FreeBSD
- NetBSD
- Homebrew
- PyPI

try diffoscope online

And on the World Wide Web!

`https://try.diffoscope.org`



Reprotest

reprotest

- builds something twice with many variations

Reprotest

reprotest

- builds something twice with many variations
- `https://salsa.debian.org/reproducible/reprotest`

Reprotest

reprotest

- builds something twice with many variations
- `https://salsa.debian.org/reproducible/reprotest`
- if unreproducible: "bisect" the variations

So you want to have Reproducible builds

`https://reproducible-builds.org/docs/recording/`
Providing sufficient information for independent verification:

- ...

So you want to have Reproducible builds

`https://reproducible-builds.org/docs/recording/`
Providing sufficient information for independent verification:

- ...
- "toolchain" packages at specific versions

So you want to have Reproducible builds

`https://reproducible-builds.org/docs/recording/`
Providing sufficient information for independent verification:

- ...
- "toolchain" packages at specific versions
- SOURCE_DATE_EPOCH (seconds since 1970-01-01)

So you want to have Reproducible builds

`https://reproducible-builds.org/docs/recording/`
Providing sufficient information for independent verification:

- ...
- "toolchain" packages at specific versions
- SOURCE_DATE_EPOCH (seconds since 1970-01-01)
- Works best with Free and Open Source Software!

To Catch a Regression

Automatic Testing (Continuous Integration, Quality Assurance, etc.)

Make it happen

https://reproducible-builds.org/contribute/

Keeping the lights on

`https://reproducible-builds.org/donate/`

If you are feeling spontaneous

https://reproducible-builds.org/events/hamburg2023/

Thanks

Open Technology Fund
Civil Infrastructure Platform
Mullvad VPN
Protocol Labs
Siemens

Copyright and attributions