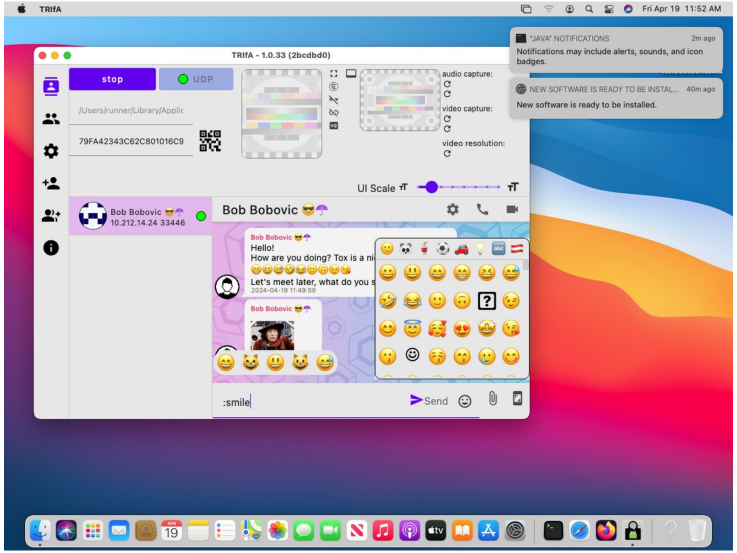
Workshop

Coding and working on security-focused apps, using Tox as example

IT-S NOW 2024

06.06.2024



/Users/runner/Library/Applic

79FA42343C62C801016C9

"JAVA" NOTIFICATIONS 2m ago
Notifications may include alerts, sounds, and icon badges.

NEW SOFTWARE IS READY TO BE INSTAL... 40m ago
New software is ready to be installed.

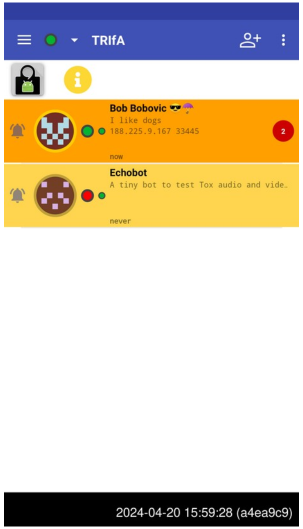
Bob Bobovic 😎☂️
10.212.14.24 33446

Bob Bobovic 😎☂️

Bob Bobovic 😎☂️
Hello!
How are you doing? Tox is a ni
😁😆😅🤣😂🙂🙃😉😘
Let's meet later, what do you s
2024-04-19 11:49:59

Bob Bobovic 😎☂️

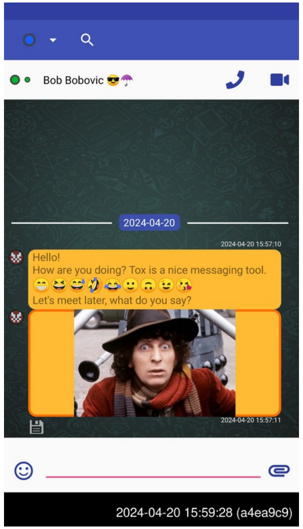
:smile



Bob Bobovic 😎☂️
I like dogs
188.225.9.167 33445
now
2

Echobot
A tiny bot to test Tox audio and vide..
never

2024-04-20 15:59:28 (a4ea9c9)



Bob Bobovic 😎☂️

2024-04-20

2024-04-20 15:57:10
Hello!
How are you doing? Tox is a nice messaging tool.
😁😆😅🤣😂🙂🙃😉😘
Let's meet later, what do you say?

2024-04-20 15:57:11

2024-04-20 15:59:28 (a4ea9c9)

Adopting the Noise Key Exchange in Tox

Improved security of Tox instant messaging with NoiseIK

Topics ...

Redesign of Tox’s cryptographic handshake

Code defensively and use dependencies responsibly

Using GitHub CI and the likes

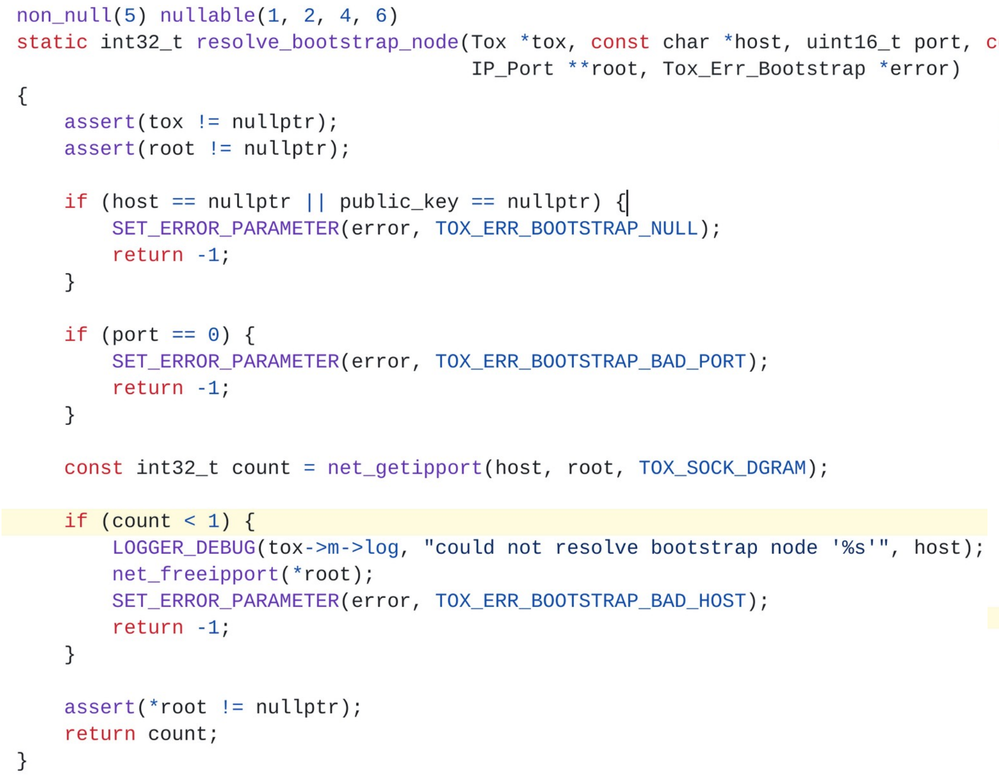
Make C code safer and better auditable

Redesign of Tox's cryptographic handshake

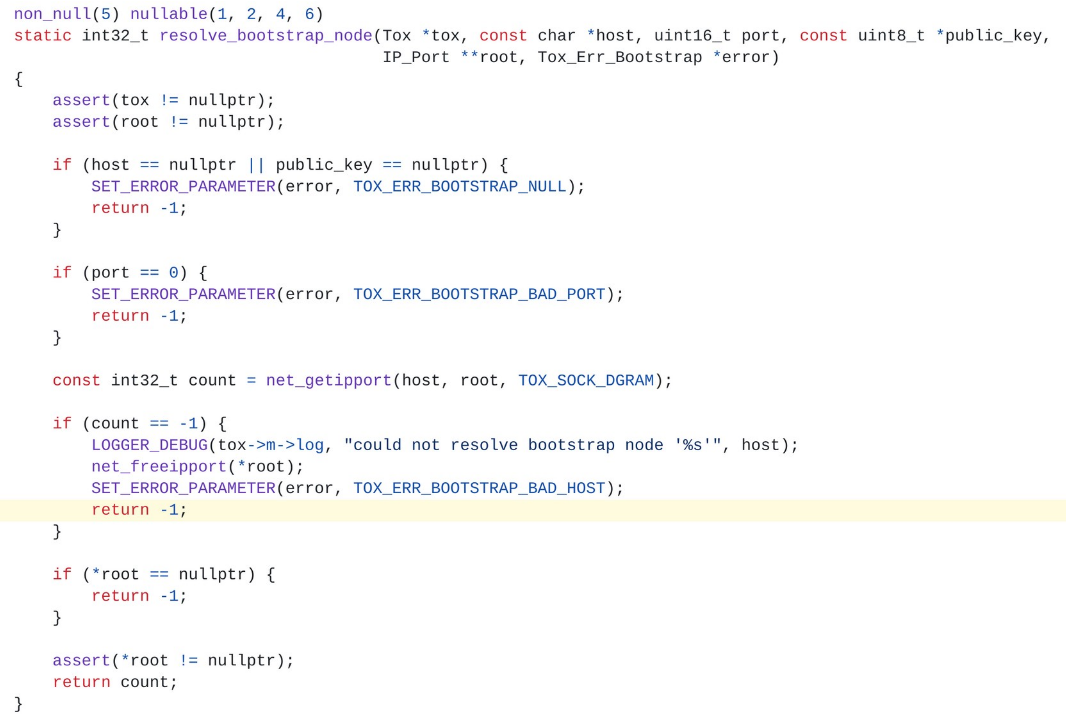
Adopting the Noise Key Exchange in Tox

Improved security of Tox instant messaging with NoiseIK

Make C code safer and better auditable

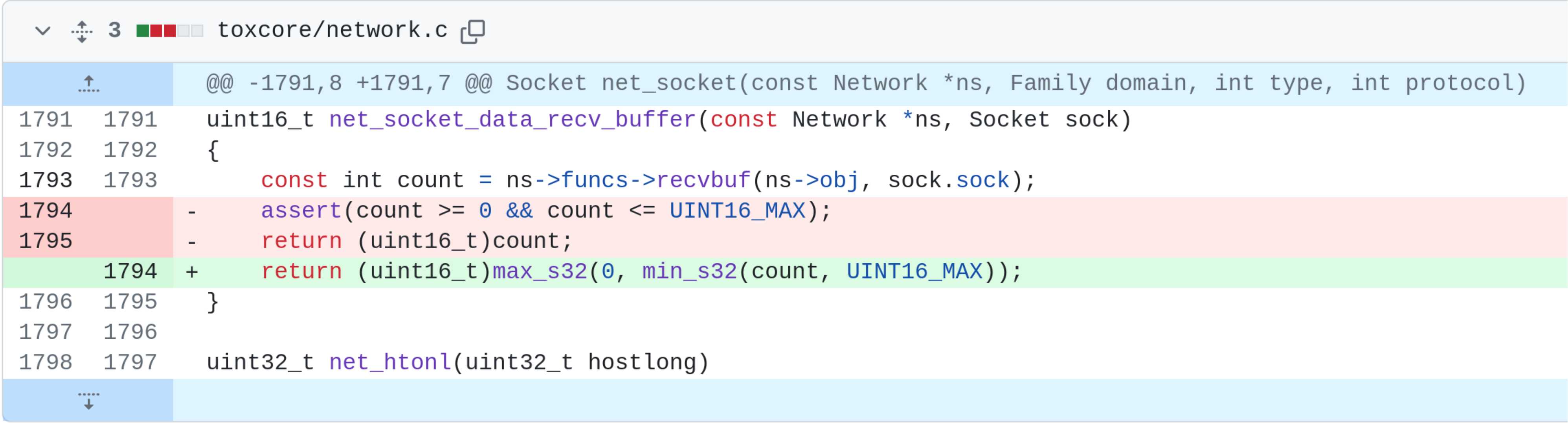


```c
non_null(5) nullable(1, 2, 4, 6)
static int32_t resolve_bootstrap_node(Tox *tox, const char *host, uint16_t port, const uint8_t *public_key,
                                      IP_Port **root, Tox_Err_Bootstrap *error)
{
    assert(tox != nullptr);
    assert(root != nullptr);

    if (host == nullptr || public_key == nullptr) {
        SET_ERROR_PARAMETER(error, TOX_ERR_BOOTSTRAP_NULL);
        return -1;
    }

    if (port == 0) {
        SET_ERROR_PARAMETER(error, TOX_ERR_BOOTSTRAP_BAD_PORT);
        return -1;
    }

    const int32_t count = net_getipport(host, root, TOX_SOCK_DGRAM);

    if (count < 1) {
        LOGGER_DEBUG(tox->m->log, "could not resolve bootstrap node '%s'", host);
        net_freeipport(*root);
        SET_ERROR_PARAMETER(error, TOX_ERR_BOOTSTRAP_BAD_HOST);
        return -1;
    }

    assert(*root != nullptr);
    return count;
}
```



```c
non_null(5) nullable(1, 2, 4, 6)
static int32_t resolve_bootstrap_node(Tox *tox, const char *host, uint16_t port, const uint8_t *public_key,
                                      IP_Port **root, Tox_Err_Bootstrap *error)
{
    assert(tox != nullptr);
    assert(root != nullptr);

    if (host == nullptr || public_key == nullptr) {
        SET_ERROR_PARAMETER(error, TOX_ERR_BOOTSTRAP_NULL);
        return -1;
    }

    if (port == 0) {
        SET_ERROR_PARAMETER(error, TOX_ERR_BOOTSTRAP_BAD_PORT);
        return -1;
    }

    const int32_t count = net_getipport(host, root, TOX_SOCK_DGRAM);

    if (count == -1) {
        LOGGER_DEBUG(tox->m->log, "could not resolve bootstrap node '%s'", host);
        net_freeipport(*root);
        SET_ERROR_PARAMETER(error, TOX_ERR_BOOTSTRAP_BAD_HOST);
        return -1;
    }

    if (*root == nullptr) {
        return -1;
    }

    assert(*root != nullptr);
    return count;
}
```

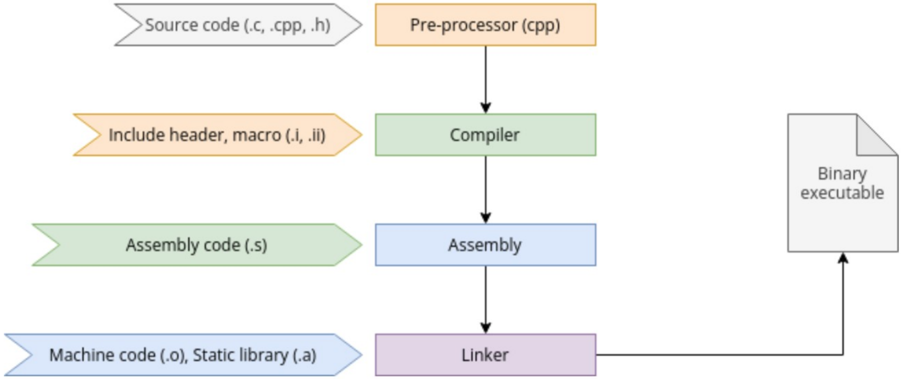
Make C code safer and better auditable



3 🟩🟥🟥⬜⬜ toxcore/network.c

```
@@ -1791,8 +1791,7 @@ Socket net_socket(const Network *ns, Family domain, int type, int protocol)
  uint16_t net_socket_data_recv_buffer(const Network *ns, Socket sock)
  {
      const int count = ns->funcs->recvbuf(ns->obj, sock.sock);
-     assert(count >= 0 && count <= UINT16_MAX);
-     return (uint16_t)count;
+     return (uint16_t)max_s32(0, min_s32(count, UINT16_MAX));
  }

  uint32_t net_htonl(uint32_t hostlong)
```

Phases of the C compiler


Source code (.c, .cpp, .h)
Pre-processor (cpp)
Include header, macro (.i, .ii)
Compiler
Assembly code (.s)
Assembly
Machine code (.o), Static library (.a)
Linker
Binary executable

(Jayashree Huttanagoudar, CC BY-SA 4.0)

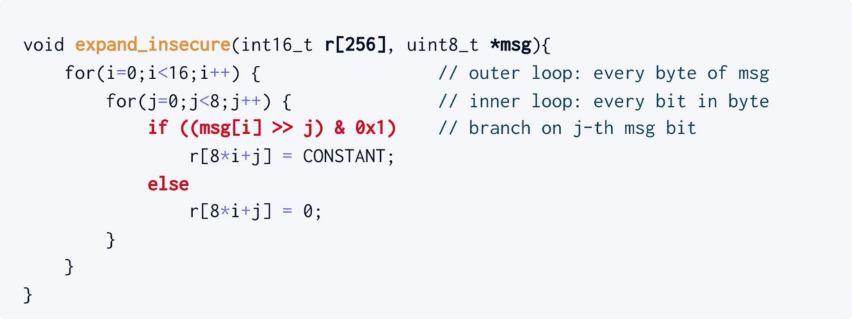
```
gcc -E main.c -o main.i        # C Preprocessor
gcc -S main.i -o main.s        # convert to assembly language
gcc -c main.s -o main.o        # Assembler
gcc main.o -o mybinary         # Linker
```

C compiler foo

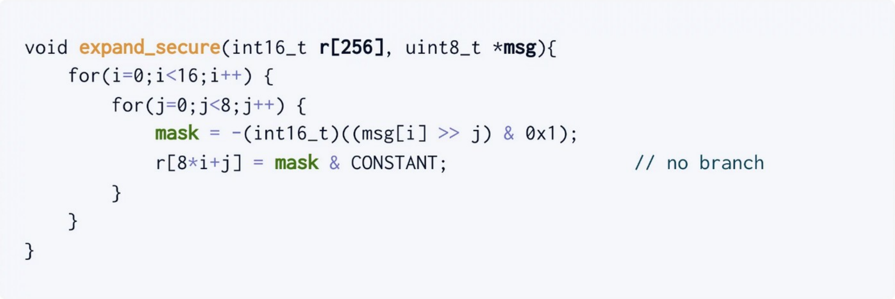
PQShield plugs timing leaks in Kyber / ML-KEM to improve PQC implementation maturity

03/06/2024 Author: Dr Antoon Purnal

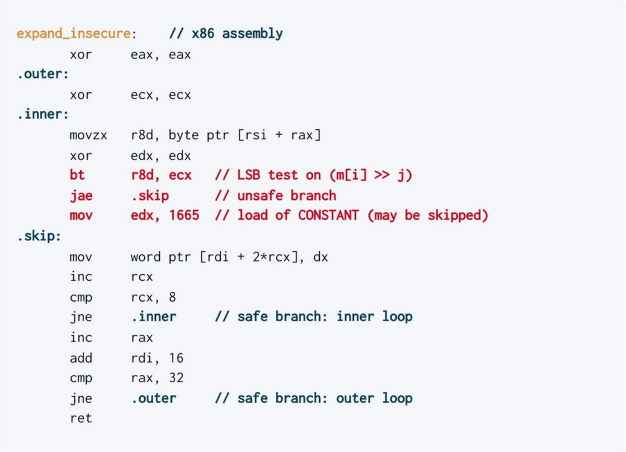
https://pqshield.com/pqshield-plugs-timing-leaks-in-kyber-ml-kem-to-improve-pqc-implementation-maturity/



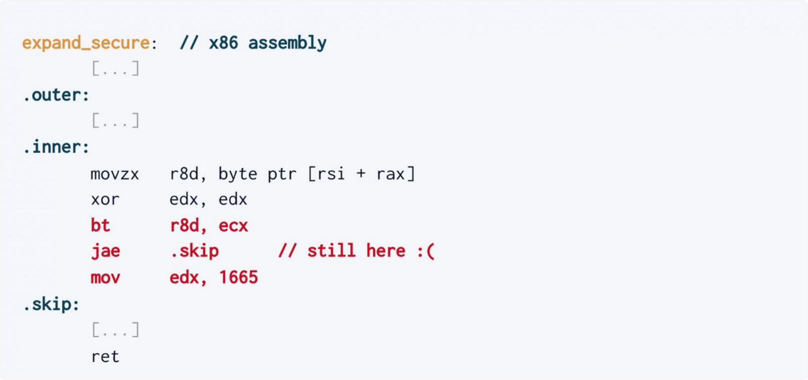
```
void expand_insecure(int16_t r[256], uint8_t *msg){
    for(i=0;i<16;i++) {                    // outer loop: every byte of msg
        for(j=0;j<8;j++) {                 // inner loop: every bit in byte
            if ((msg[i] >> j) & 0x1)       // branch on j-th msg bit
                r[8*i+j] = CONSTANT;
            else
                r[8*i+j] = 0;
        }
    }
}
```



```
void expand_secure(int16_t r[256], uint8_t *msg){
    for(i=0;i<16;i++) {
        for(j=0;j<8;j++) {
            mask = -(int16_t)((msg[i] >> j) & 0x1);
            r[8*i+j] = mask & CONSTANT;                  // no branch
        }
    }
}
```



```
expand_insecure:    // x86 assembly
        xor     eax, eax
.outer:
        xor     ecx, ecx
.inner:
        movzx   r8d, byte ptr [rsi + rax]
        xor     edx, edx
        bt      r8d, ecx   // LSB test on (m[i] >> j)
        jae     .skip      // unsafe branch
        mov     edx, 1665  // load of CONSTANT (may be skipped)
.skip:
        mov     word ptr [rdi + 2*rcx], dx
        inc     rcx
        cmp     rcx, 8
        jne     .inner     // safe branch: inner loop
        inc     rax
        add     rdi, 16
        cmp     rax, 32
        jne     .outer     // safe branch: outer loop
        ret
```



```
expand_secure:  // x86 assembly
        [...]
.outer:
        [...]
.inner:
        movzx   r8d, byte ptr [rsi + rax]
        xor     edx, edx
        bt      r8d, ecx
        jae     .skip      // still here :(
        mov     edx, 1665
.skip:
        [...]
        ret
```

Toxcore - Dependencies

- toxcore + toxencryptsave
  - libsodium
    https://github.com/jedisct1/libsodium
    Libsodium v1.0.12 and v1.0.13 Security Assessment in 2017
    https://www.privateinternetaccess.com/blog/2017/08/libsodium-v1-0-12-and-v1-0-13-security-assessment/

Cryptography Engineering

- toxav
  - libvpx https://github.com/webmproject/libvpx
  - libopus https://github.com/xiph/opus
  - x264* https://code.videolan.org/videolan/x264/-/tree/stable?ref_type=heads
  - libav* https://github.com/FFmpeg/FFmpeg

* Zoxcore - toxcore experiment fork (experimental H.264 support and other upgrades)
https://github.com/zoff99/c-toxcore

Toxcore - Dependencies (2)

- libvpx https://github.com/webmproject/libvpx
- libopus https://github.com/xiph/opus
    - yasm https://github.com/yasm/yasm

- x264* https://code.videolan.org/videolan/x264/-/tree/stable?ref_type=heads
- libav* https://github.com/FFmpeg/FFmpeg
    - nasm https://www.nasm.us/pub/nasm/releasebuilds/2.13.02/nasm-2.13.02.tar.bz2
    - yasm https://github.com/yasm/yasm

[*] Zoxcore - toxcore experiment fork (experimental H.264 support and other upgrades)
https://github.com/zoff99/c-toxcore

Nice Things (about Toxcore) ...

Easy to compile on almost any platform

use make or cmake or just use the single file toxcore amalgamation
https://github.com/zoff99/c-toxcore/tree/zoff99/zoxcore_local_fork/amalgamation

No Access to Storage / Disk

Toxcore itself does not read / write or access any storage itself

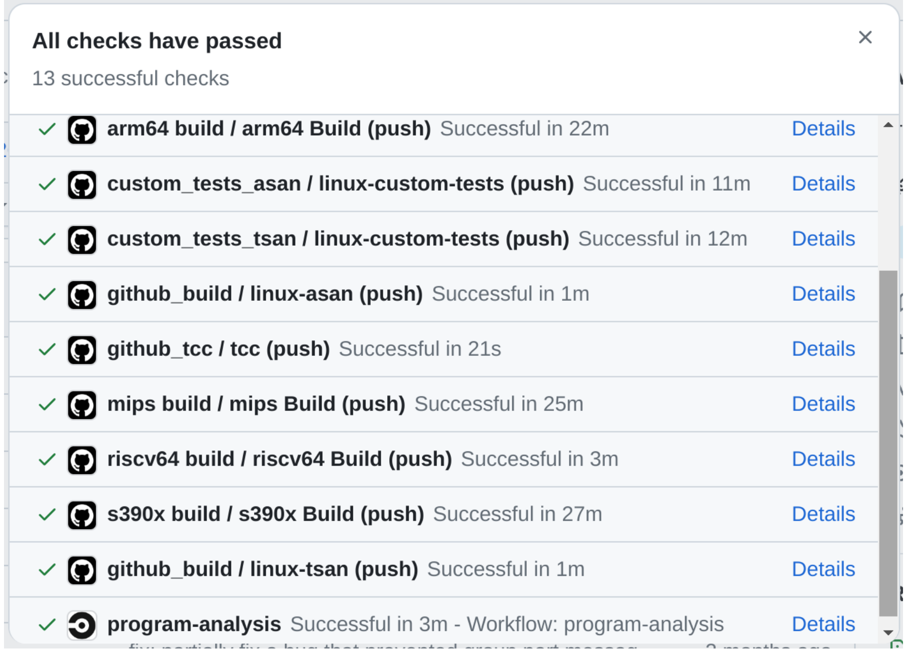
Does not do Anything on it's own

a client application needs to trigger actions (iterate) in Toxcore

No internal Threads are created

Toxcore runs on the Thread(s) given to by a client application

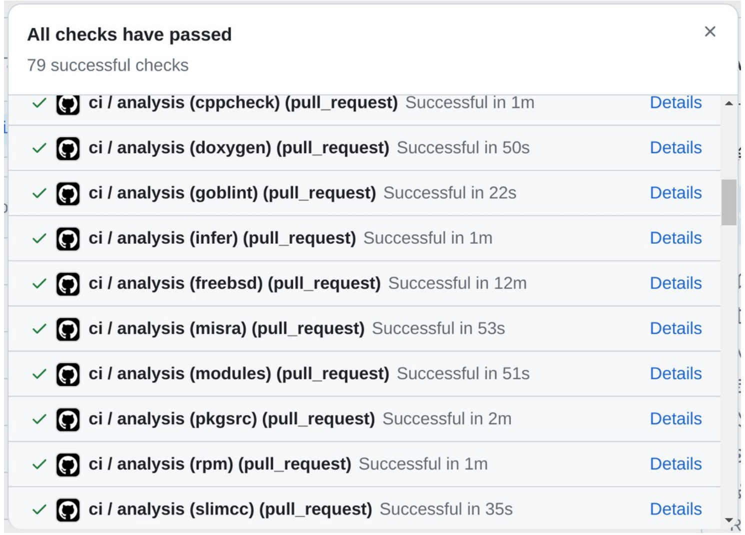
Nice Things (about Toxcore) ...



All checks have passed

13 successful checks

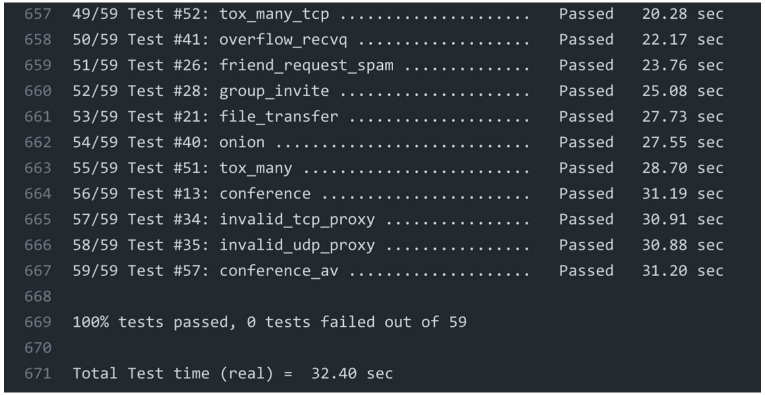
- ✓ **arm64 build / arm64 Build (push)** Successful in 22m Details
- ✓ **custom_tests_asan / linux-custom-tests (push)** Successful in 11m Details
- ✓ **custom_tests_tsan / linux-custom-tests (push)** Successful in 12m Details
- ✓ **github_build / linux-asan (push)** Successful in 1m Details
- ✓ **github_tcc / tcc (push)** Successful in 21s Details
- ✓ **mips build / mips Build (push)** Successful in 25m Details
- ✓ **riscv64 build / riscv64 Build (push)** Successful in 3m Details
- ✓ **s390x build / s390x Build (push)** Successful in 27m Details
- ✓ **github_build / linux-tsan (push)** Successful in 1m Details
- ✓ **program-analysis** Successful in 3m - Workflow: program-analysis Details



All checks have passed

79 successful checks

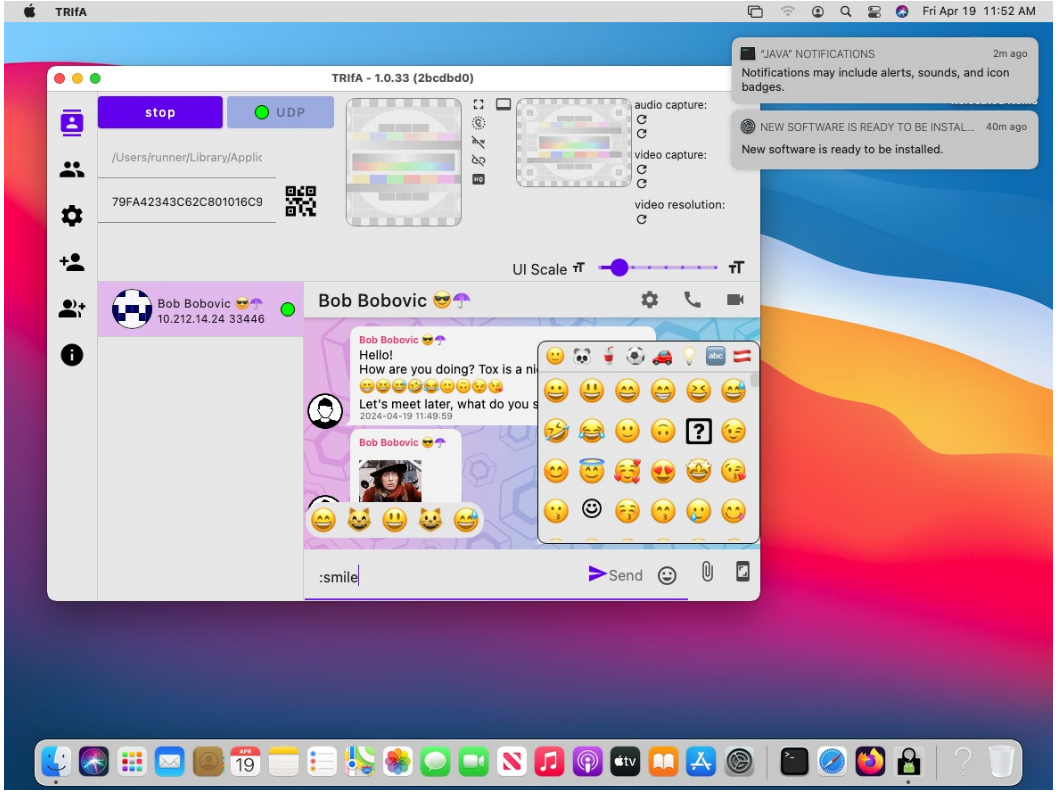
- ✓ **ci / analysis (cppcheck) (pull_request)** Successful in 1m Details
- ✓ **ci / analysis (doxygen) (pull_request)** Successful in 50s Details
- ✓ **ci / analysis (goblint) (pull_request)** Successful in 22s Details
- ✓ **ci / analysis (infer) (pull_request)** Successful in 1m Details
- ✓ **ci / analysis (freebsd) (pull_request)** Successful in 12m Details
- ✓ **ci / analysis (misra) (pull_request)** Successful in 53s Details
- ✓ **ci / analysis (modules) (pull_request)** Successful in 51s Details
- ✓ **ci / analysis (pkgsrc) (pull_request)** Successful in 2m Details
- ✓ **ci / analysis (rpm) (pull_request)** Successful in 1m Details
- ✓ **ci / analysis (slimcc) (pull_request)** Successful in 35s Details



```
49/59 Test #52: tox_many_tcp .....................   Passed   20.28 sec
50/59 Test #41: overflow_recvq ...................   Passed   22.17 sec
51/59 Test #26: friend_request_spam ..............   Passed   23.76 sec
52/59 Test #28: group_invite .....................   Passed   25.08 sec
53/59 Test #21: file_transfer ....................   Passed   27.73 sec
54/59 Test #40: onion ............................   Passed   27.55 sec
55/59 Test #51: tox_many .........................   Passed   28.70 sec
56/59 Test #13: conference .......................   Passed   31.19 sec
57/59 Test #34: invalid_tcp_proxy ................   Passed   30.91 sec
58/59 Test #35: invalid_udp_proxy ................   Passed   30.88 sec
59/59 Test #57: conference_av ....................   Passed   31.20 sec

100% tests passed, 0 tests failed out of 59

Total Test time (real) =  32.40 sec
```

TRIfA Material - Linux, Windows, MacOS ...

GitHub https://github.com/Zoxcore/trifa_material



"JAVA" NOTIFICATIONS 2m ago
Notifications may include alerts, sounds, and icon badges.

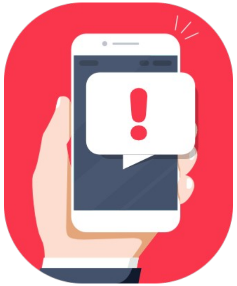
NEW SOFTWARE IS READY TO BE INSTAL... 40m ago
New software is ready to be installed.

Bob Bobovic 😎☂️
Hello!
How are you doing? Tox is a ni
😁😆😅🤣😂🙂🙃😉😘
Let's meet later, what do you s
2024-04-19 11:49:59

Bob Bobovic 😎☂️

:smile

triggers Push Notifications



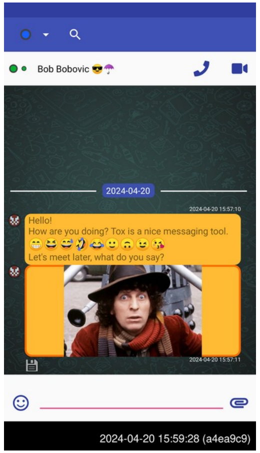
TRIfA - Android

GET IT ON F-Droid

https://f-droid.org/packages/com.zoffcc.applications.trifa/

GET IT ON Google Play

https://play.google.com/store/apps/details?id=com.zoffcc.applications.trifa


Bob Bobovic 😎☂

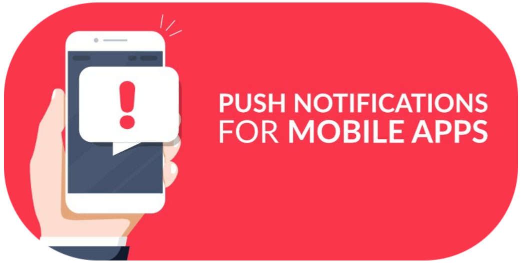
2024-04-20

2024-04-20 15:57:10

Hello!
How are you doing? Tox is a nice messaging tool.
😁😆😅🤣😂🙂🙃😉😘
Let's meet later, what do you say?

2024-04-20 15:57:11

2024-04-20 15:59:28 (a4ea9c9)



https://zoff99.github.io/ToxAndroidRefImpl/PUSH_NOTIFICATION.html

getting in touch ...

- Github

  https://github.com/zoff99/c-toxcore

- Tox Public Group

  154b3973bd0e66304fd6179a8a54759073649e09e6e368f0334fc6ed666ab762

  or

  https://trifagrp.tox.zoff.cc/

  or

  QR Code ------------------------------->

- Matrix

  https://matrix.to/#/#trifa:matrix.org

- Email

  tox@zoff.cc