Verificarea Monocypher cu FramaC/Eva

Adrian Manea

SLA, 410

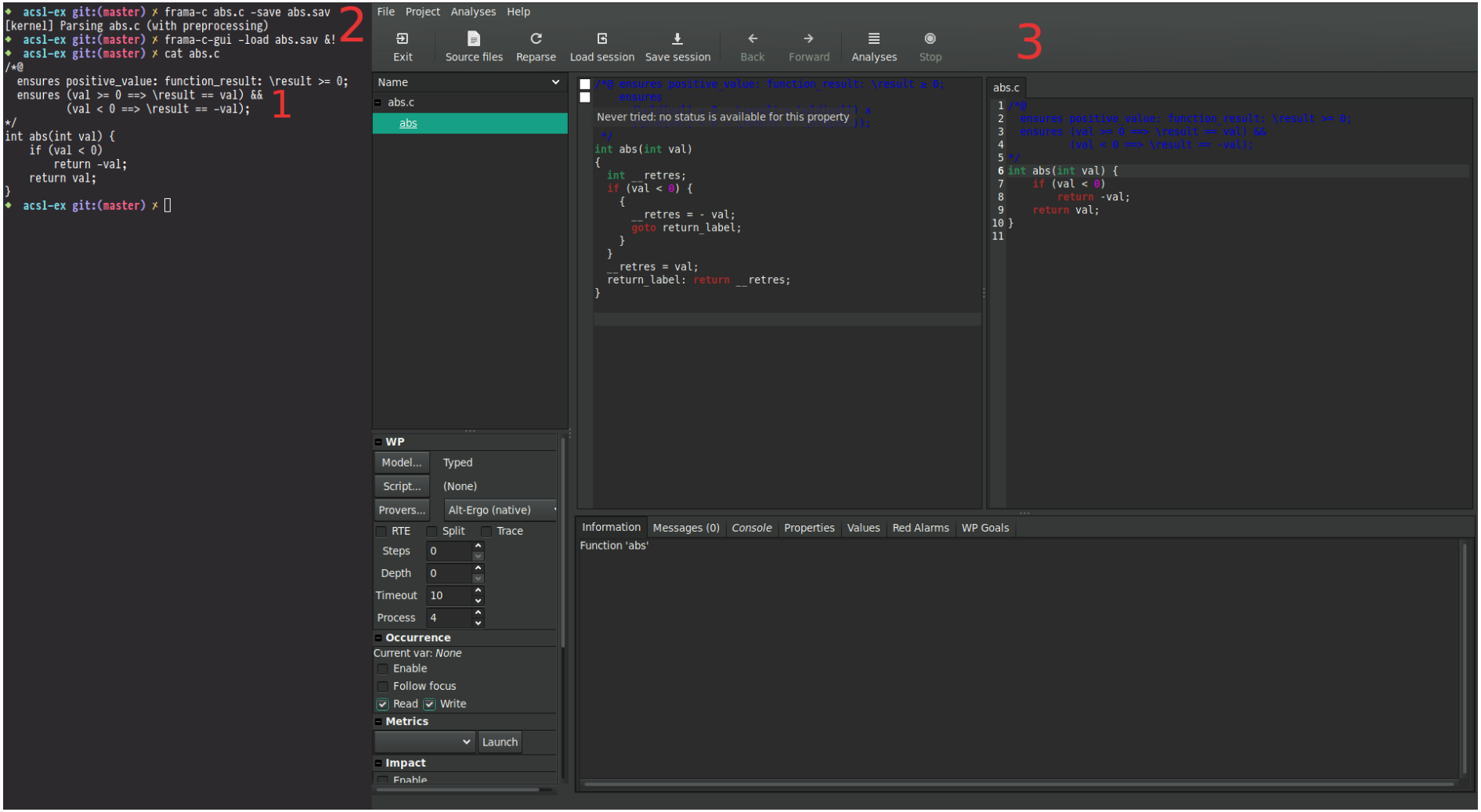
Uneltele de verificare

FramaC — implementat în OCaml, pentru verificarea programelor C.
Oferă o verificare automată sau cu adnotări în ACSL.
Face doar verificări „de bază“. Extins prin plugin-uri:

- WP = Weakest Precondition calculus;
- RTE = RunTime Evaluation;
- Eva = Evolved Value Analysis;
- poate integra demonstratorul Coq.

Detalii: [FramaC, 2019a], [Baudin, 2019], [FramaC, 2019b].

Exemplu: FramaC CLI + GUI



Eva

Bazat pe **interpretarea abstractă** ([Cousot, 2008]).

Ideea: pornind de la *semantica concretă* a unei bucăți de cod, se *abstractizează*, pentru ușurarea calculelor.

Eva calculează valorile și intervalele valorilor unor variabile.

Face sub- sau supra-aproximări, deci produce *alarme*, nu erori.

Monocypher

Este o bibliotecă criptografică simplă și rapidă. ([LoupVaillant, 2019a])

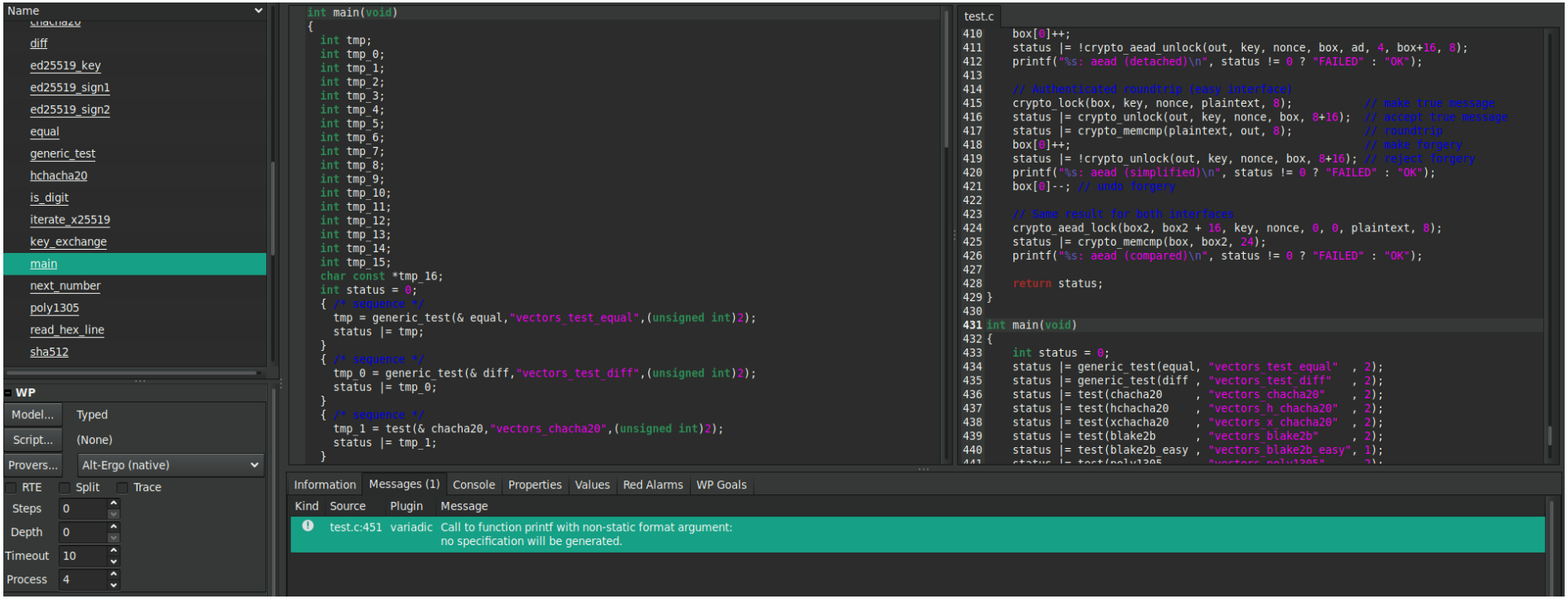
Folosește algoritmi criptografici clasici: `sha_512`, `ChaCha20` ([LoupVaillant, 2019b], [LoupVaillant, 2017]).

Rularea inițială

```
# rezultate stdout:
$ frama-c *.c
# rezultate log
$ frama-c *.c > log ; cat log | less
# rezultate salvate, apoi GUI
$ frama-c *.c -save firstrun.sav
$ frama-c-gui -load firstrun.sav
```

Se pot da opțiuni suplimentare de preprocesor și se încarcă în GUI rezultatul, pentru a vizualiza mai bine rezultatele.

Rularea inițială în GUI



Verificarea cu Eva

```
$ frama-c -load parsed.sav -val -val-builtins-auto \
      -no-val-show-progress -memexec-all -save value.sav > log
```

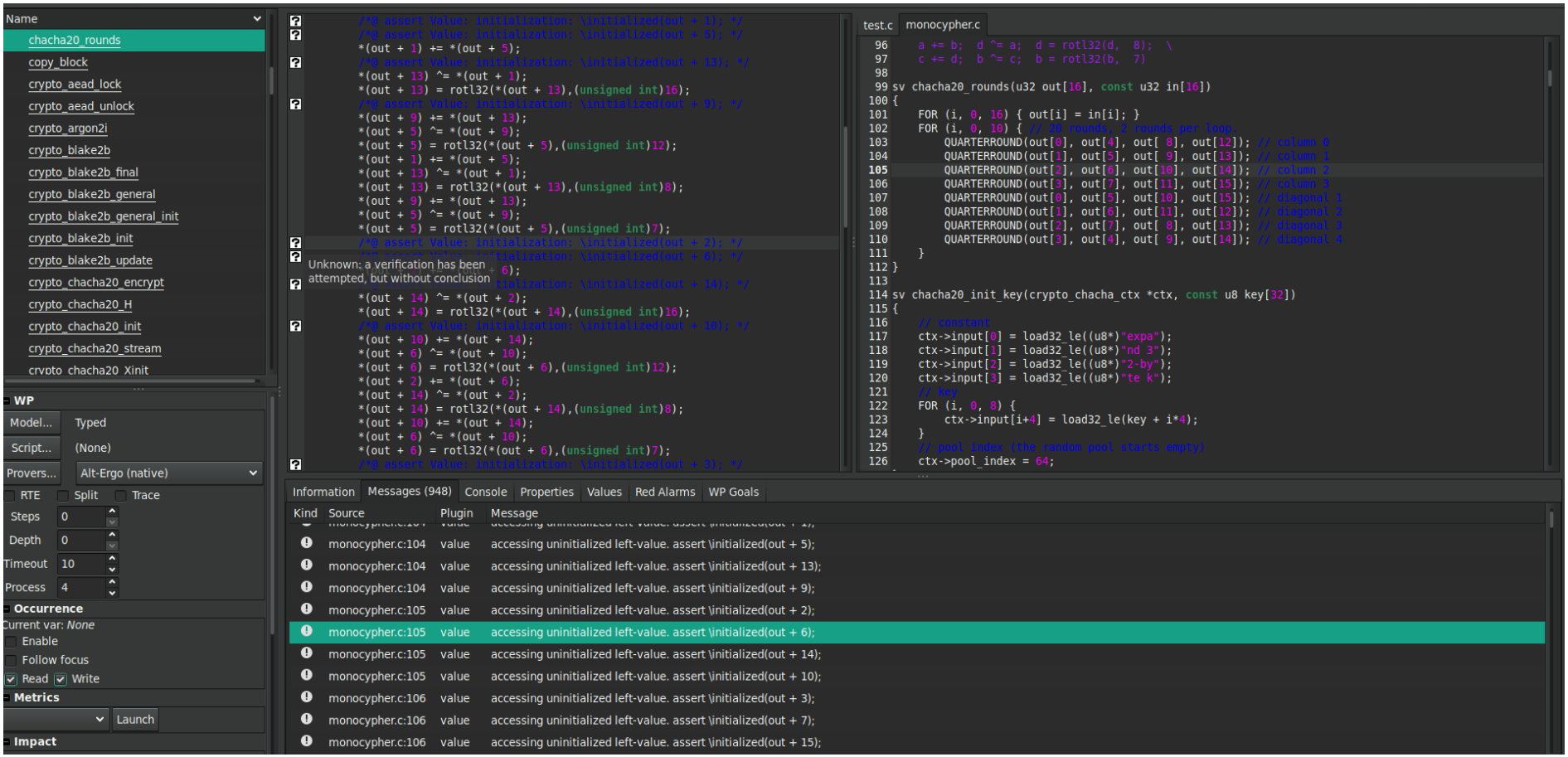
- `-val` = Eva;
- `-val-builtins-auto` = verificările standard;
- `-no-val-show-progress` = nu se afișează mesaje cînd se intră în fiecare funcție;
- `-memexec-all` = se păstrează în cache valorile calculate, pentru refolosire.

Cîteva linii din log

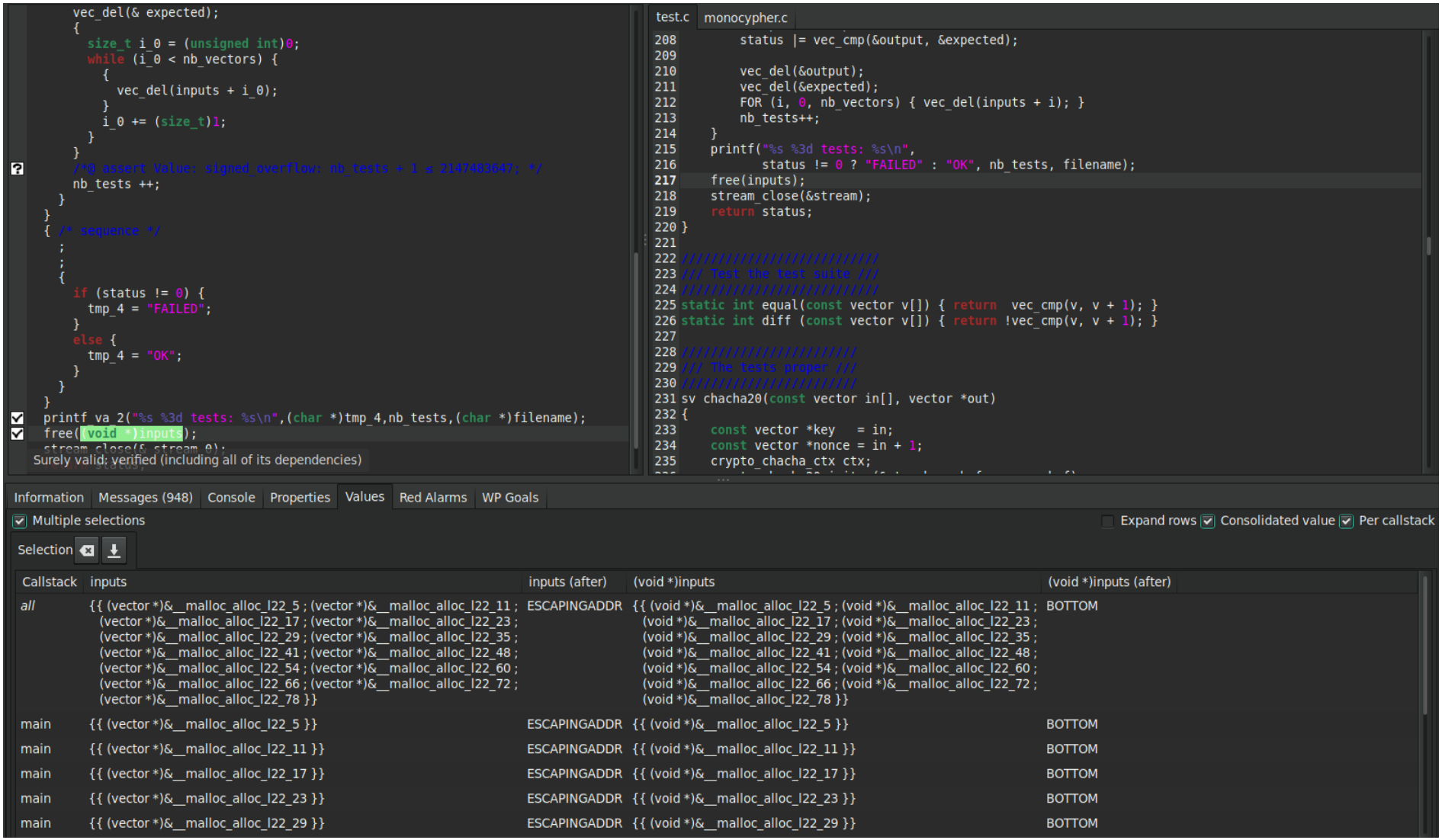
```
[value] Analyzing a complete application starting at main
[value] Computing initial state
[value] Initial state computed
[value:initial-state] Values of globals at initialization
__fc_errno in [--..--]
__fc_stderr in {{ NULL ; &S___fc_stderr[0] }}
__fc_strtok_ptr in {0}
K[0] in {4794697086780616226}
[1] in {8158064640168781261}
...
blake2b_compress_sigma[0][0] in {0}
[0][1] in {1}
...
[value:alarm] test.c:126: Warning:
out of bounds write. assert \valid(v->buf + v->size);
```

```
 [value:alarm] test.c:121: Warning:
function memcpy: precondition 'valid_src' got status unknown.
 [value:alarm] monocypher.c:839: Warning:
 signed overflow. assert g3 * 19 leq 2147483647;
 [kernel] monocypher.c:494:
more than 200(253) locations to update in array. Approximating.
```

Rafinarea verificării în GUI

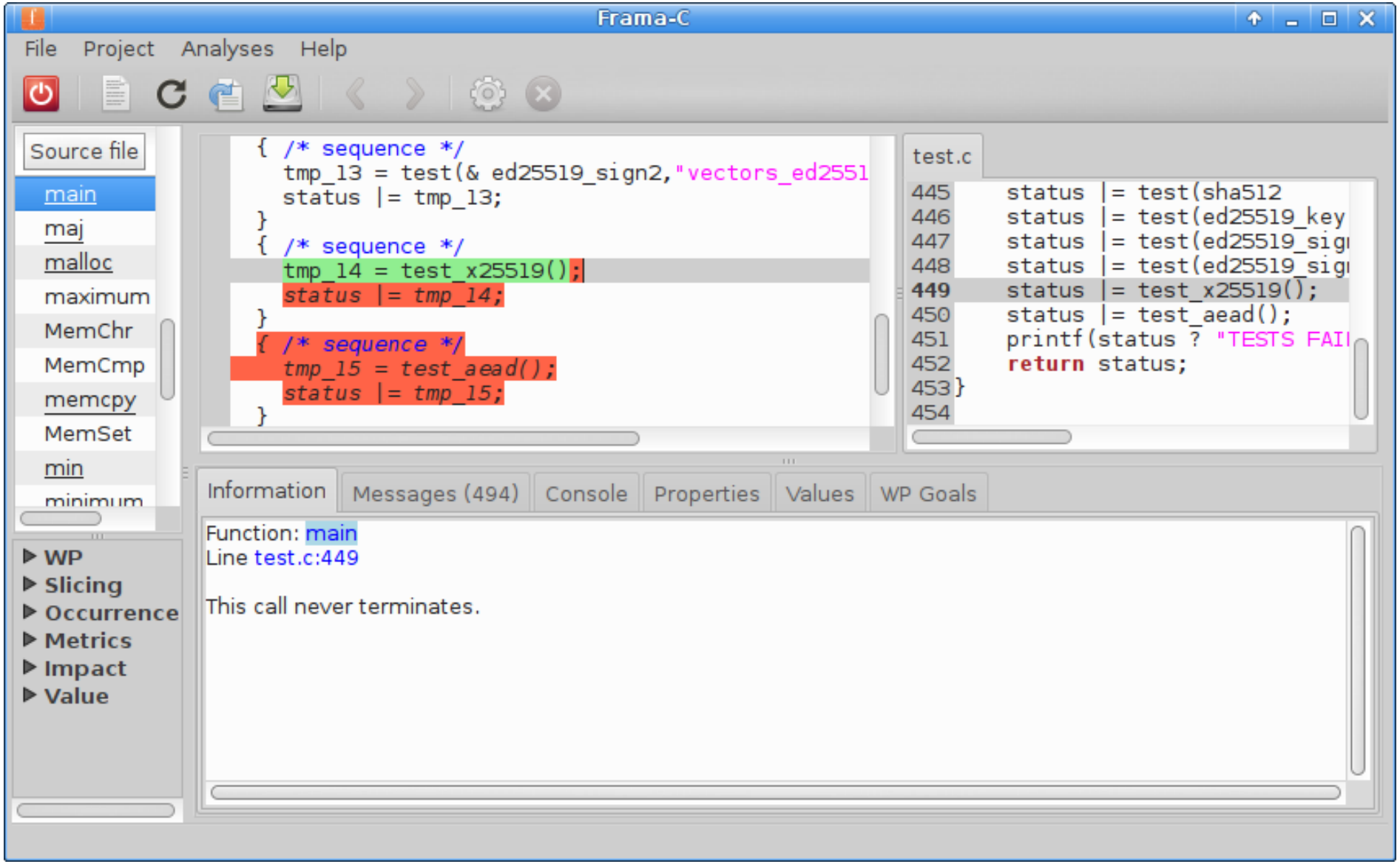


Rafinarea verificării în GUI

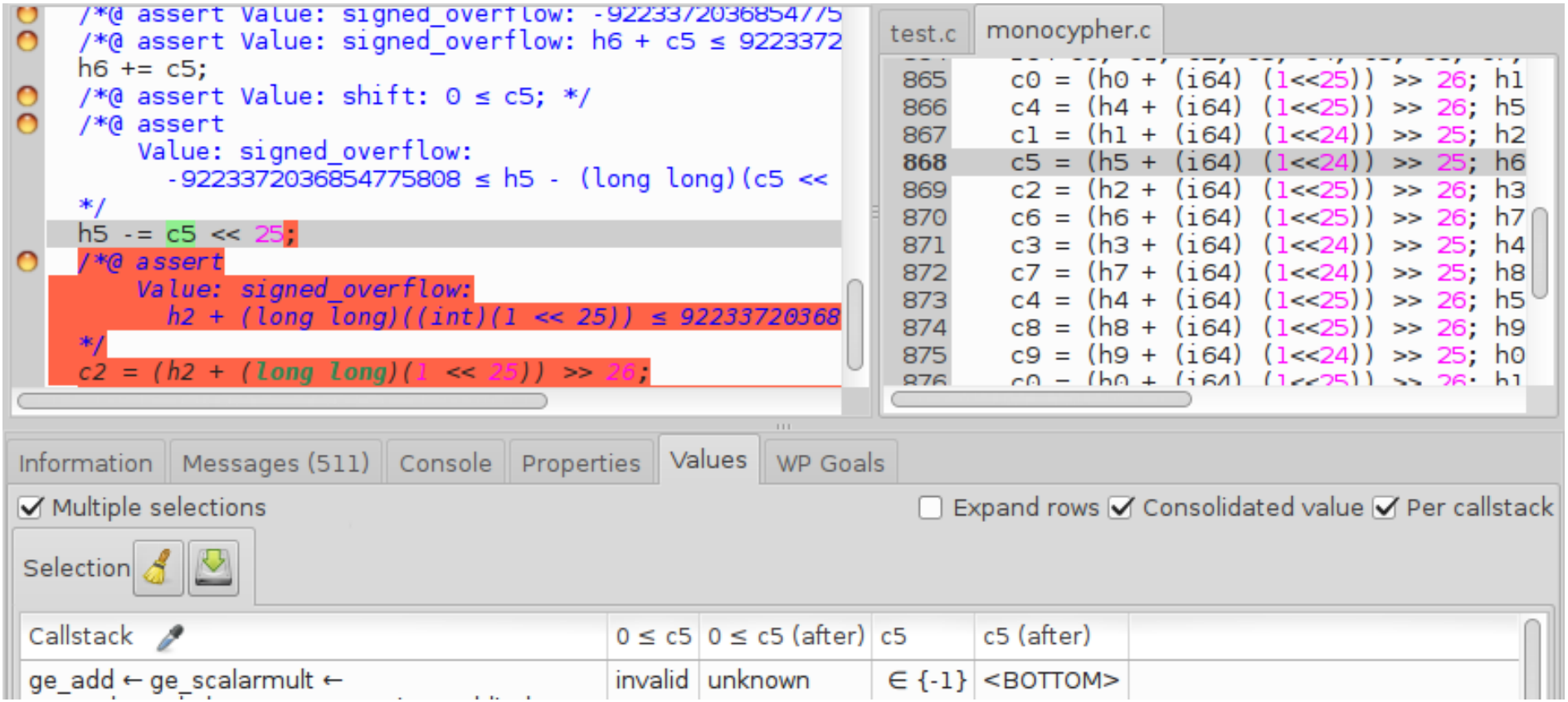


Concluzii: NO BUGS

Neterminare în versiuni anterioare ([Maroneze, 2017]):



Neterminarea în versiuni anterioare




Baudin, P. h. (2019).
FramaC.
site oficial.
online, accesat aprilie-mai 2019.


Cousot, P. (2008).
Abstract interpretation.
site oficial.
online, accesat aprilie-mai 2019.


FramaC (2019a).
ACSL.
site oficial.
online, accesat aprilie-mai 2019.


FramaC, G. (2019b).
Eva.
manual oficial.
online, accesat aprilie-mai 2019.


LoupVaillant (2017).
How I Implemented My Own Crypto.
eseu online.
online, accesat aprilie-mai 2019.


LoupVaillant (2019a).
Monocypher.
GitHub.
online, accesat aprilie-mai 2019.


LoupVaillant (2019b).
Monocypher.
site oficial.
online, accesat aprilie-mai 2019.


Maroneze, A. (2017).
A simple Eva tutorial.
blogul FramaC.
online, accesat aprilie-mai 2019.