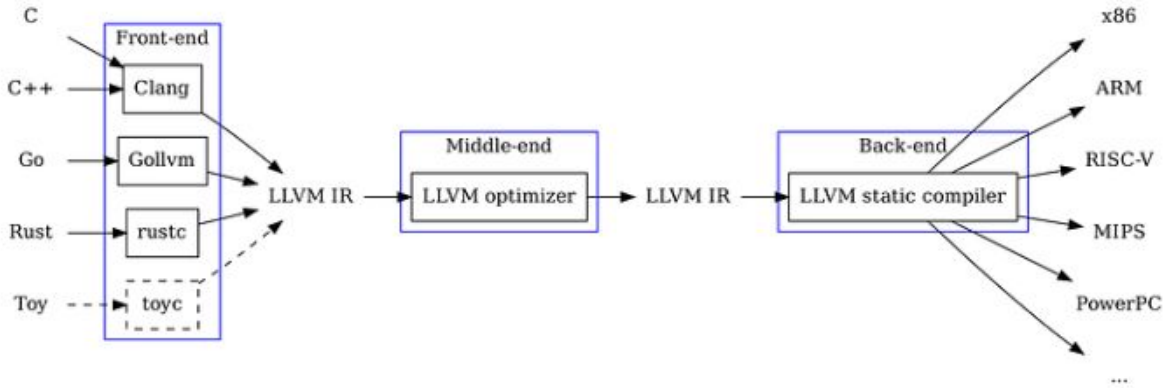
Traçabilité du processus de compilation d'un compilateur llvm, dans le cadre d'une compilation obfuscante

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Paul ENJALBERT

Tuteur:
Jean-Christophe BACH
Fabien DAGNAT

Contexte du stage



Chaîne de compilation d'un code compilé avec llvm

Travaux effectués

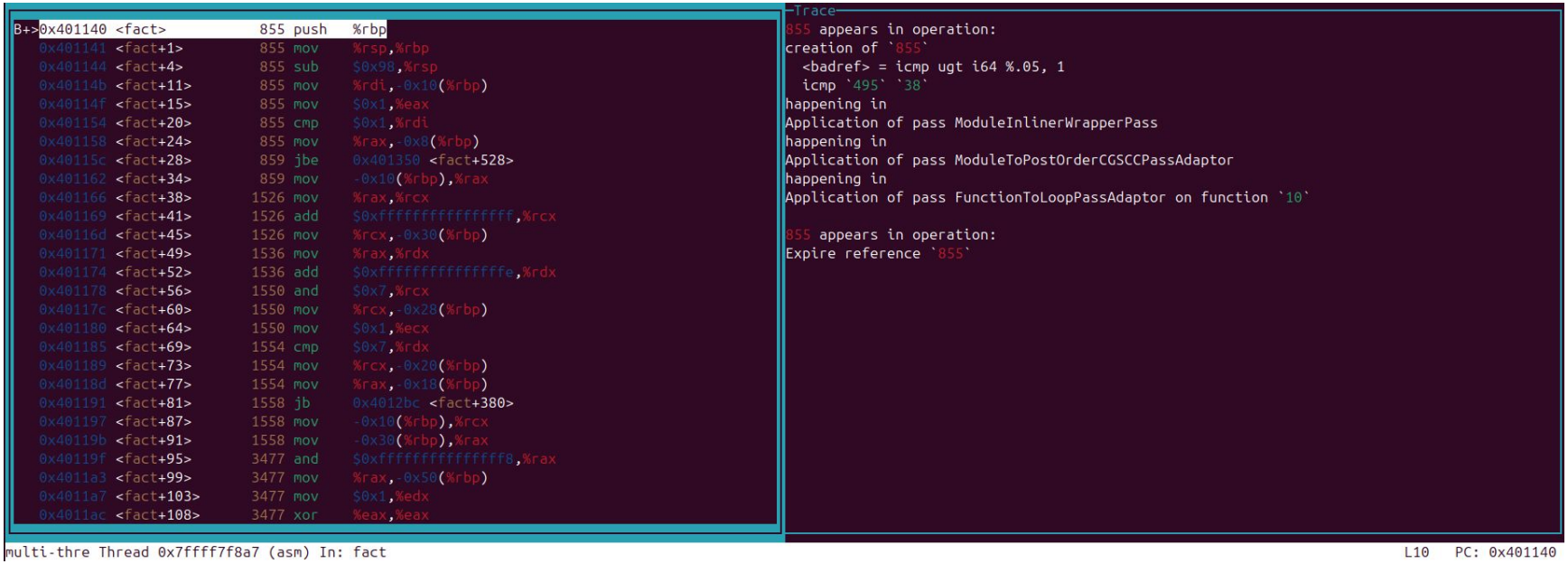
1. Reproductibilité

- Compiler le code llvm modifié (qomptrace) -> LLVM 14

- Compilation d’un code et l’obtention d’une trace


```
[
  {
    "type": "event",
    "event-type": "creation",
    "name": "declare i32 @BlockSort(i32*, i8*, i32)\n
    "value-id": "4",
    "operands": []
  },
  {
    "type": "event",
    "event-type": "creation",
    "name": "\n",
    "value-id": "8"
  },
  {
    "type": "event",
    "event-type": "creation",
    "name": "i32 undef",
    "value-id": "12",
    "operands": []
  },
  {
    "type": "event",
    "event-type": "creation",
    "name": "argument",
    "value-id": "18"
  },
  {
```

- Utilisation d’un plugin gdb afin d’afficher les informations de debug



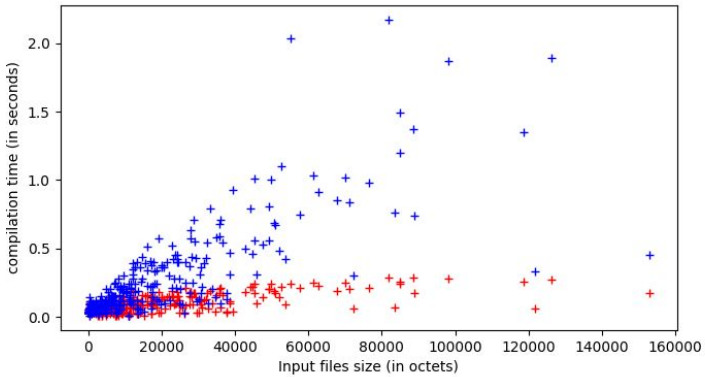
Travaux effectués

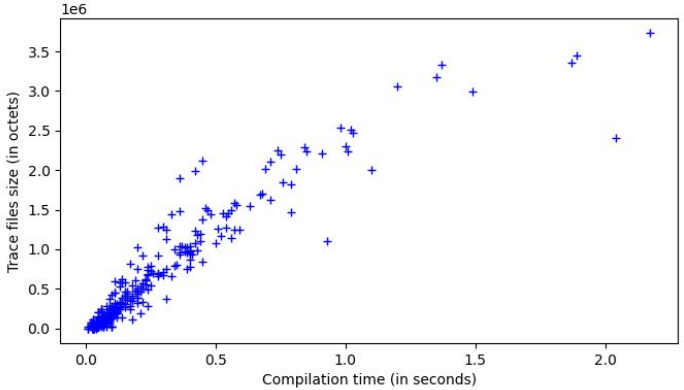
2. Evaluation

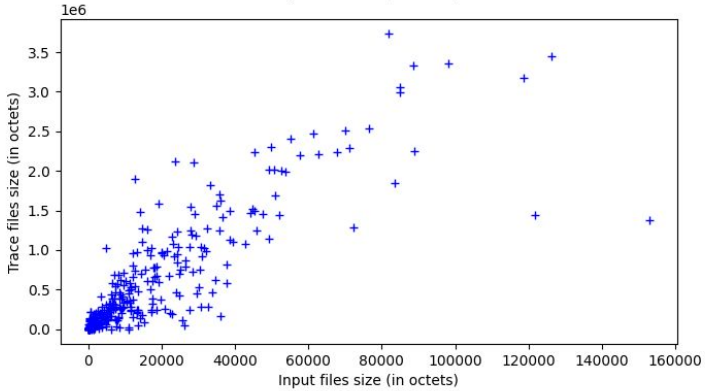
- Evaluation du temps de compilation avec la llvm test suite

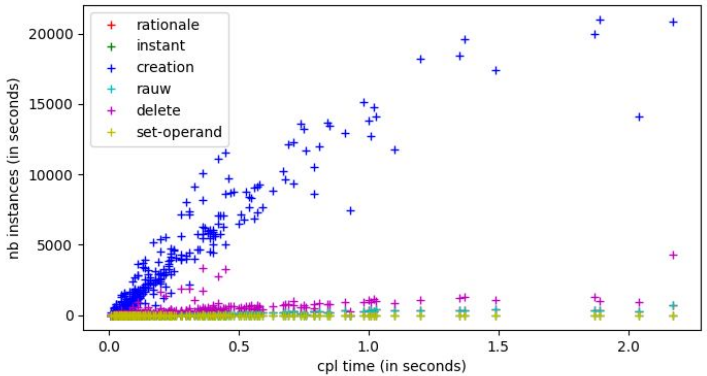
- Mise en place d'une campagne de test qui compare:

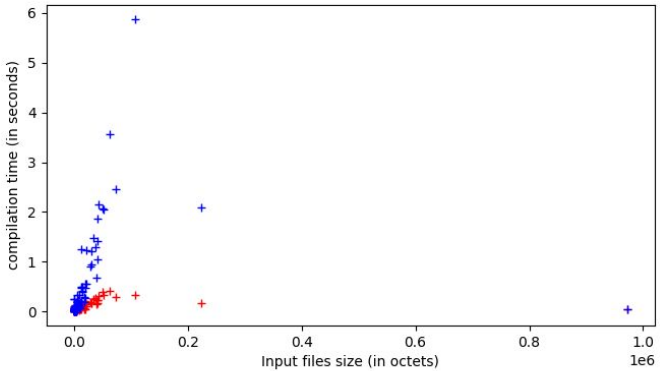
    1. Temps de compilation
    2. Tailles des fichiers de trace
    3. Contenus des fichiers de trace

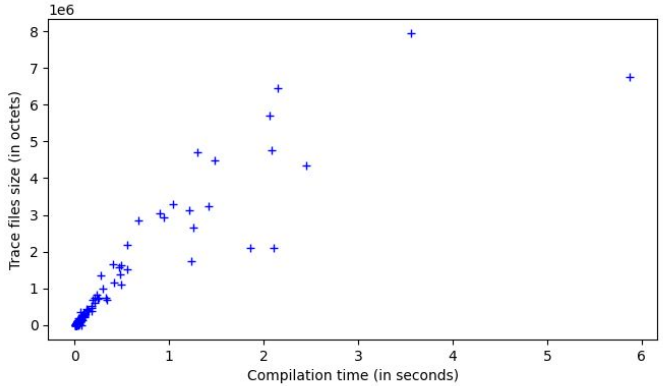

compilation time (in seconds)
Input files size (in octets)

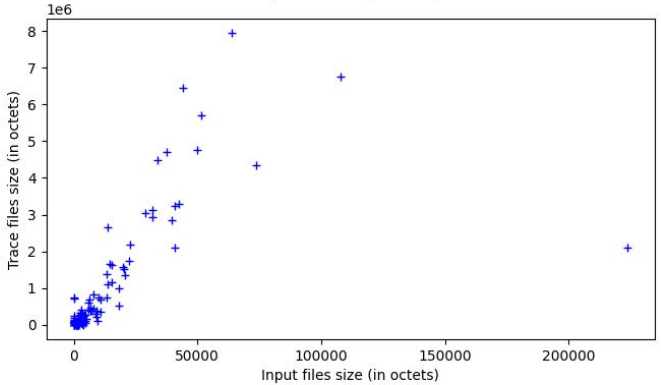

1e6
Trace files size (in octets)
Compilation time (in seconds)

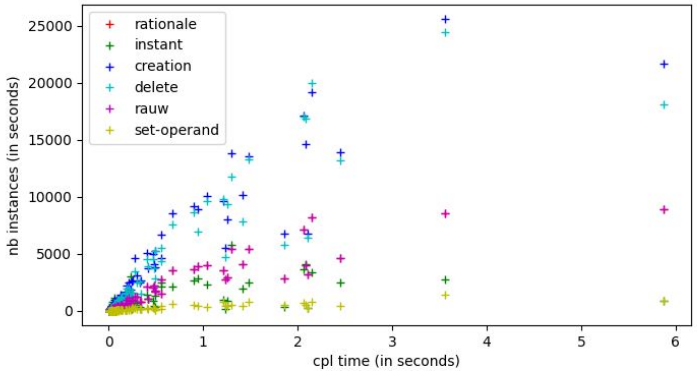

1e6
Trace files size (in octets)
Input files size (in octets)


rationale
instant
creation
rauw
delete
set-operand
nb instances (in seconds)
cpl time (in seconds)


compilation time (in seconds)
Input files size (in octets)
1e6


1e6
Trace files size (in octets)
Compilation time (in seconds)


1e6
Trace files size (in octets)
Input files size (in octets)


rationale
instant
creation
delete
rauw
set-operand
nb instances (in seconds)
cpl time (in seconds)

Travaux effectués

3. Portabilité

- Portabilité des modifications de qomptrace sur LLVM 20

- Changement important entre ces deux versions : opaque pointers

Travaux effectués

4. Autres

- Image Docker

- Correction d’un bug lors de l’utilisation de passes d’optimisations

- Nommage des fichiers de trace en fonction du fichier d’entrée + répertoire de stockage

Suite du stage

- Faire marcher la campagne d’évaluation sur la llvm test suite
- Portabilité des passes d’obfuscation de ollvm sur LLVM 20
- Ajouter de nouvelles informations de trace
- Intégrer la mesure de la mémoire dans la campagne d’évaluation