SAFE RIVER
Safety & Security Forge

Software Un-Security Exploitation Evaluation
CadRiver project

Christèle Faure

Directeur scientifique

SAFE RIVER
Safety & Security Forge

Project objectives

- **Static tool development**
  - Security audit help
  - Input:
    - Audit objective (security policy)
    - Piece of software (source code)
  - Output:
    - Elements that contribute to the audit objective
    - Facts on these elements
  - Arguments to show
    - Insecurity: problems found
    - Security w.r.t. the security objective
- **Extension of vulnerability detection tools**
  - RATS, ITS4, Fortify
  - PolySpace, Astrée
  - Database of known vulnerabilities

SAFE RIVER
Safety & Security Forge

Use cases *(1)*

- **Objective: Attack surface**
- **Input:**
  - Source code
  - Specification of the attack surface
- **Output:**
  - List of entry/output points (I/O): actual attack surface
  - To be compared with expected ones

SAFE RIVER
Safety & Security Forge

Use case *(2)*

- **Objective: Accesses to assets**
- **Input:**
  - Source code
  - List of assets (cryptographic key, password, bank account)
- **Output:**
  - List of entry/output points
  - Accesses to assets (read / write)
  - Rights and access modes

SAFE RIVER
Safety & Security Forge

Use case *(3)*

- **Objective: Information leak**
- **Input:**
  - Source code
  - List of assets
- **Output:**
  - List of entry/output points
  - Impact of assets on output (O ports)
  - Influence of inputs (I ports) on assets

SAFE RIVER
Safety & Security Forge

Use case *(4)*

- **Objective: Correctness of protections**
- **Input:**
  - Source code
  - List of assets
  - List of protections
- **Output:**
  - List of sensible flows w.r.t. assets
  - Location of protections on the source code
  - Presence/miss of protections on sensible flows
  - List of unprotected assets

SAFE RIVER
Safety & Security Forge

Use case *(5)*

- **Objective: Vulnerability instruction**
- **Input:**
  - Source code
  - List of vulnerabilities
    - Present in the target software
    - Computed by other tools
- **Output:**
  - List of entry/output points
  - Influence of entries on the vulnerabilities
  - Impact of the vulnerabilities on the output
  - Simulation of the vulnerability effect ?
    - In terms of execution flow
    - In terms of memory

SAFE RIVER
Safety & Security Forge

Example Input

Inputs:

- **Vulnerability (2, non verified line)**
- **Asset (precious)**
- **Source code:**

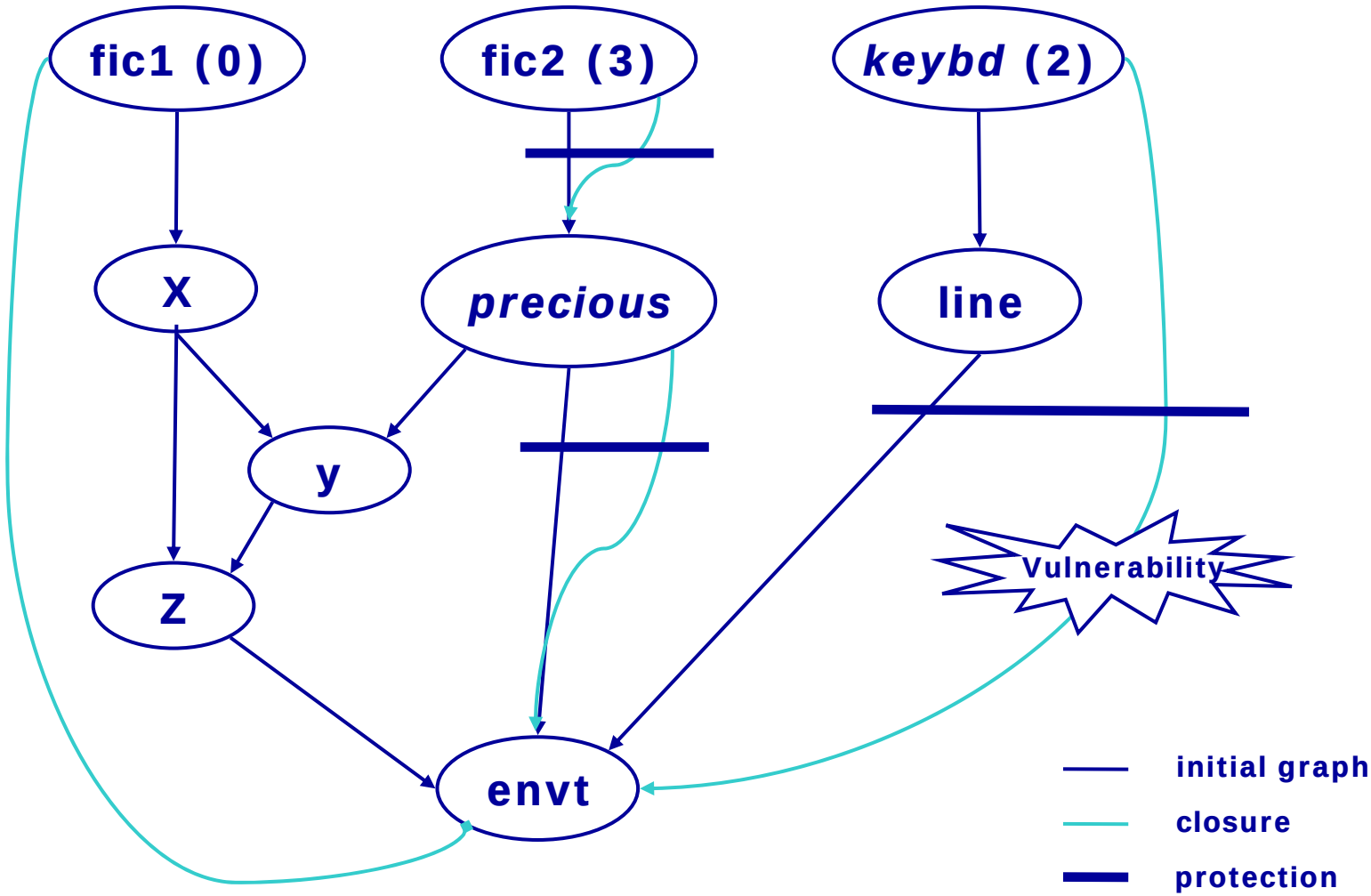
```
0: x=getc(fic1);
1: gets(line);
2: system(line);
3: fscanf(fic2,precious);
4: y=compute(precious,x);
5: z=makefullcomputation(x,y);
6: printf(z);
```

SAFE RIVER
Safety & Security Forge

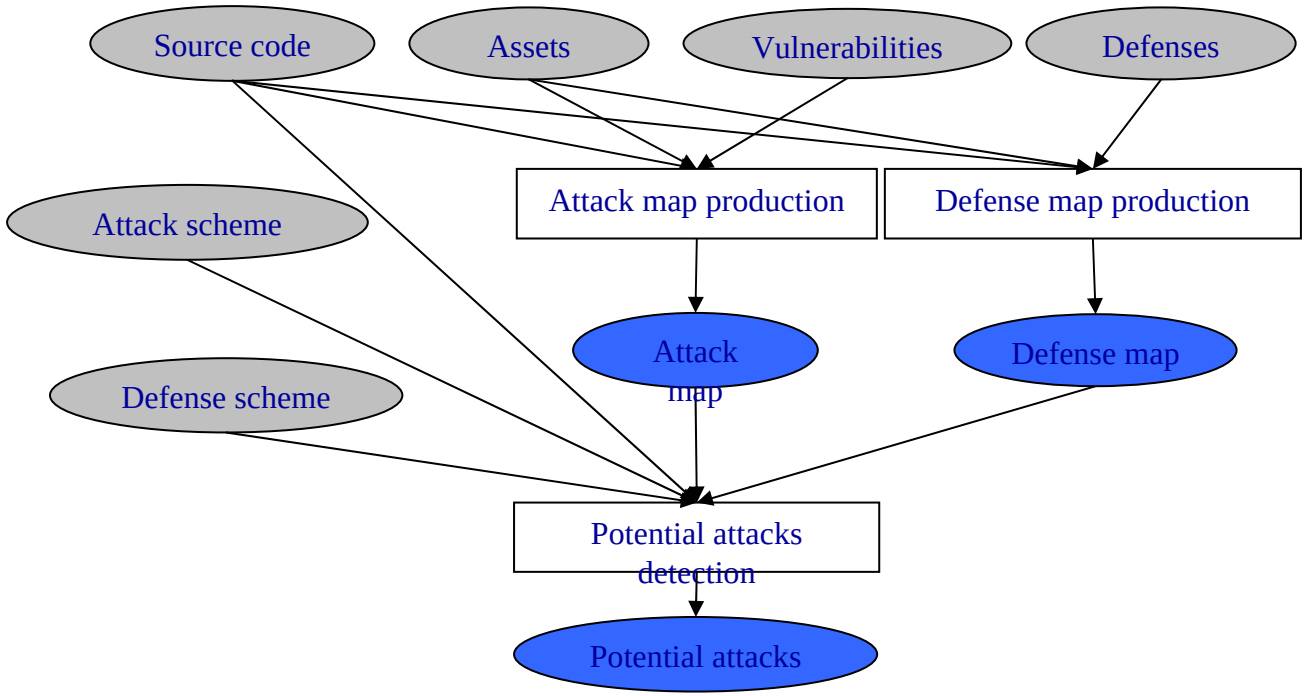
Exemple Output (1)

	Map	Definition	Use
0	IN(getc)	X	fic1
1	IN(gets)	line	*keyboard*
2	OUT(system) Vuln(non verified line)	*envt*	line
3	IN(fscanf) Access(precious)	precious	fic2
4	Access(precious)	y	precious, x
5		z	x,y
6	OUT(printf)	*envt*	z

Example output (2)



CadRiver


Source code
Assets
Vulnerabilities
Defenses
Attack map production
Defense map production
Attack scheme
Attack map
Defense map
Defense scheme
Potential attacks detection
Potential attacks

SAFE RIVER
Safety & Security Forge

Underlying technologies

- **Data-flow, control-flow, semantic analysis**
- **Exploration of the source code**
  - Computation of interfaces
  - Search for occurrences of declared assets
  - Search for present vulnerabilities
  - Search for protections
- **Dependencies computation**
  - Inter-procedural
  - Aliasing
  - Projected on target paths
  - Closed by target variables
- **Conformance with objective**

SAFE RIVER
Safety & Security Forge

Development in progress

- **Internal development**
- **Related studies:**
  - Airbus, Aréva: extension of the development process
  - Thales RSS: robustness proved by PolySpace
  - Lafosec: security for functional languages (ANSSI, LIP6, INRIA …)
- **Tool development projects:**
  - Baccarat: LIP6, Magillem, Oppida
  - Autostat (Systematic/Aerospace valley)?: CEA, SERMA, Continental, IRIT, Actia, ATOS …
  - ITEA?: CEA, Verimag, EADS IW, Onera, IMDEA
  - Célar?: