Capture of silent security patches and reports

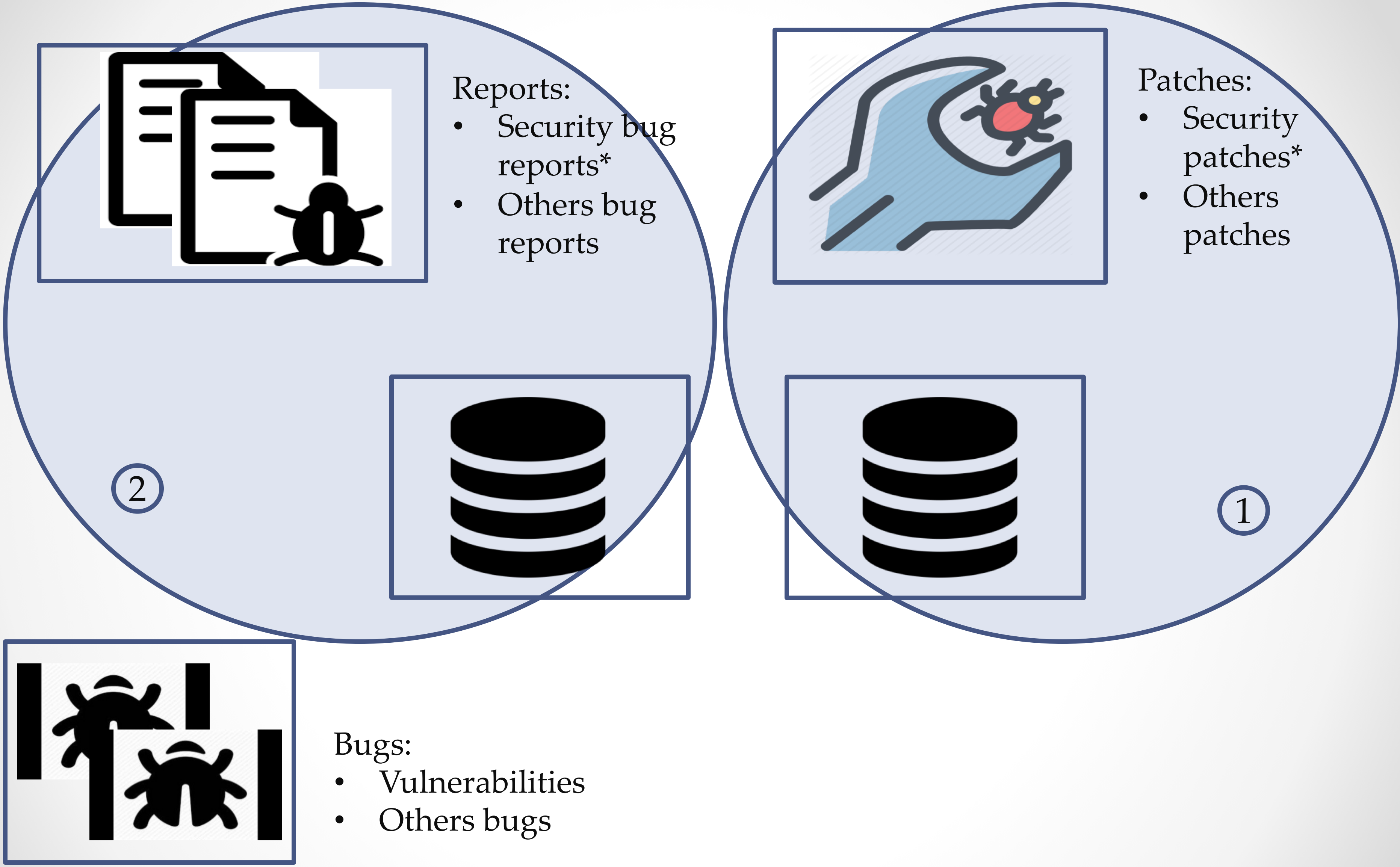
Délwendé D. A. Sawadogo (UQAM)

Naouel Moha (UQAM)

Tégawendé F. Bissyande (Univ Lu, SnT)

Second meeting of the SE@MTL community

June 6, 2019



Reports:

- Security bug reports*
- Others bug reports

Patches:

- Security patches*
- Others patches

Bugs:

- Vulnerabilities
- Others bugs

« Silent security patches»

- Non flagged
- Suppose to be a patch without security impact

« Silent security bug repports»

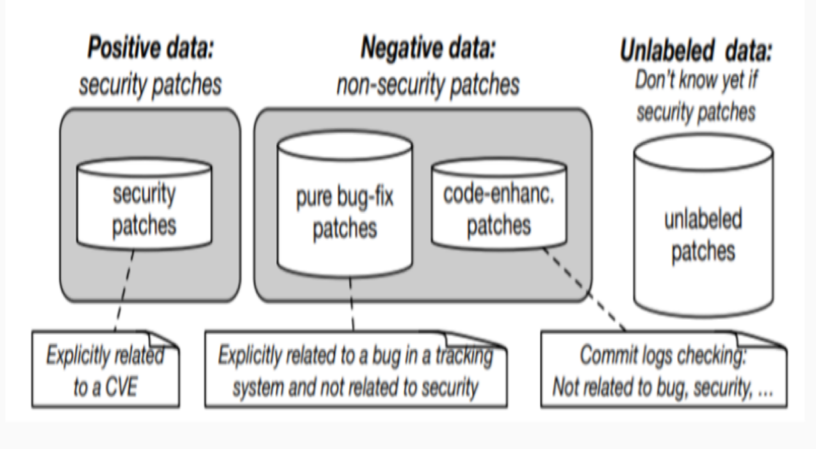
- Non flagged
- Suppose to be a bug repport without security impact

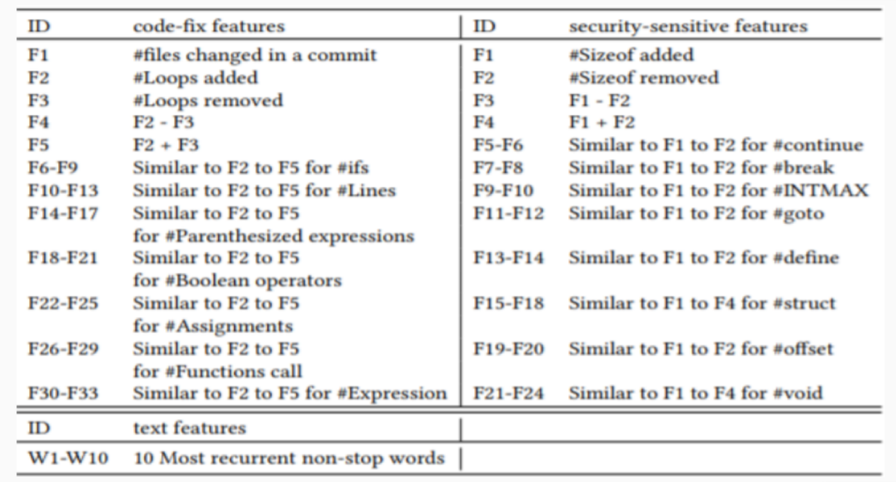
Why?

(1) Silent security patches detection

- Binary classification: we aimed to detect silent vulnerabilities in a set of fixes.
- We used text analysis on commits logs and code analysis on commits diffs.
- We trained a prediction model based on machine learning algorithms.

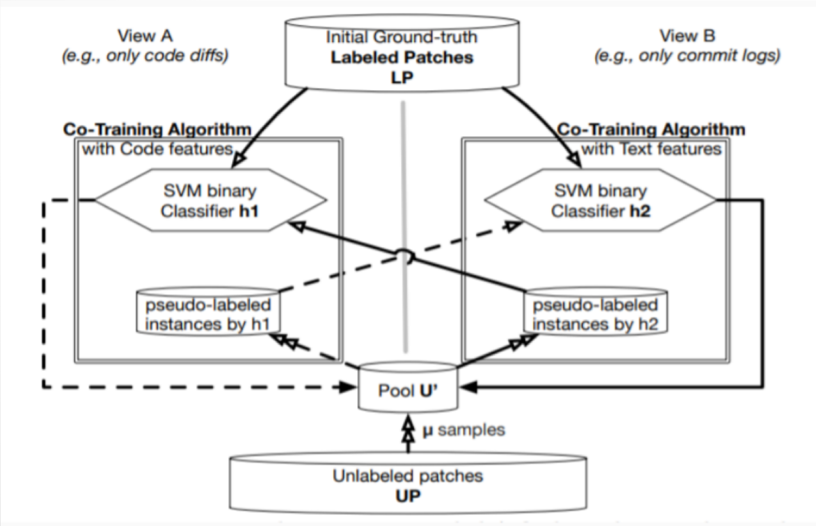
(1)Silent security patches detection


Positive data:
security patches
security patches
Explicitly related to a CVE
Negative data:
non-security patches
pure bug-fix patches
code-enhanc. patches
Explicitly related to a bug in a tracking system and not related to security
Unlabeled data:
Don't know yet if security patches
unlabeled patches
Commit logs checking: Not related to bug, security, ...



ID	code-fix features	ID	security-sensitive features
F1	#files changed in a commit	F1	#Sizeof added
F2	#Loops added	F2	#Sizeof removed
F3	#Loops removed	F3	F1 - F2
F4	F2 - F3	F4	F1 + F2
F5	F2 + F3	F5-F6	Similar to F1 to F2 for #continue
F6-F9	Similar to F2 to F5 for #ifs	F7-F8	Similar to F1 to F2 for #break
F10-F13	Similar to F2 to F5 for #Lines	F9-F10	Similar to F1 to F2 for #INTMAX
F14-F17	Similar to F2 to F5 for #Parenthesized expressions	F11-F12	Similar to F1 to F2 for #goto
F18-F21	Similar to F2 to F5 for #Boolean operators	F13-F14	Similar to F1 to F2 for #define
F22-F25	Similar to F2 to F5 for #Assignments	F15-F18	Similar to F1 to F4 for #struct
F26-F29	Similar to F2 to F5 for #Functions call	F19-F20	Similar to F1 to F2 for #offset
F30-F33	Similar to F2 to F5 for #Expression	F21-F24	Similar to F1 to F4 for #void

ID	text features
W1-W10	10 Most recurrent non-stop words



- ✓ we investigate the discriminative power of a variety of features to clarify the possibility of a learning process.
- ✓ We propose a semi-supervised approach with Co-Training which we demonstrate to yield high precision (95%) and recall (88%).
- ✓ we show that our approach can help to flag patches that were unlabeled until now.

(2) Security bug reports detection

- Mining 204 Open sources projects with arround 2000 labelled (positive and negative) vulnerabilities commits (Ponta et al., 2019) ✓

  Extraction of Bug reports by links in these commits. ✓

  Training an automatic learning model to identify security sensitive bugs reports. Loading

Bibliography

(1) Ponta et al., A Manually-Curated Dataset of Fixes to Vulnerabilities of Open-Source Software, arXiv preprint arXiv:1902.02595v (2019)