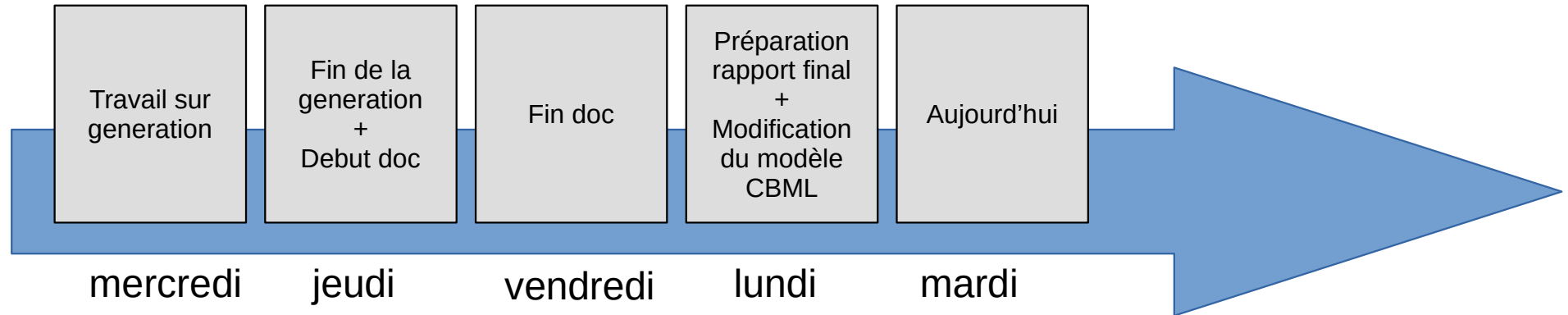


Réunion hebdomadaire 11

06/12/2022

Overview de la semaine



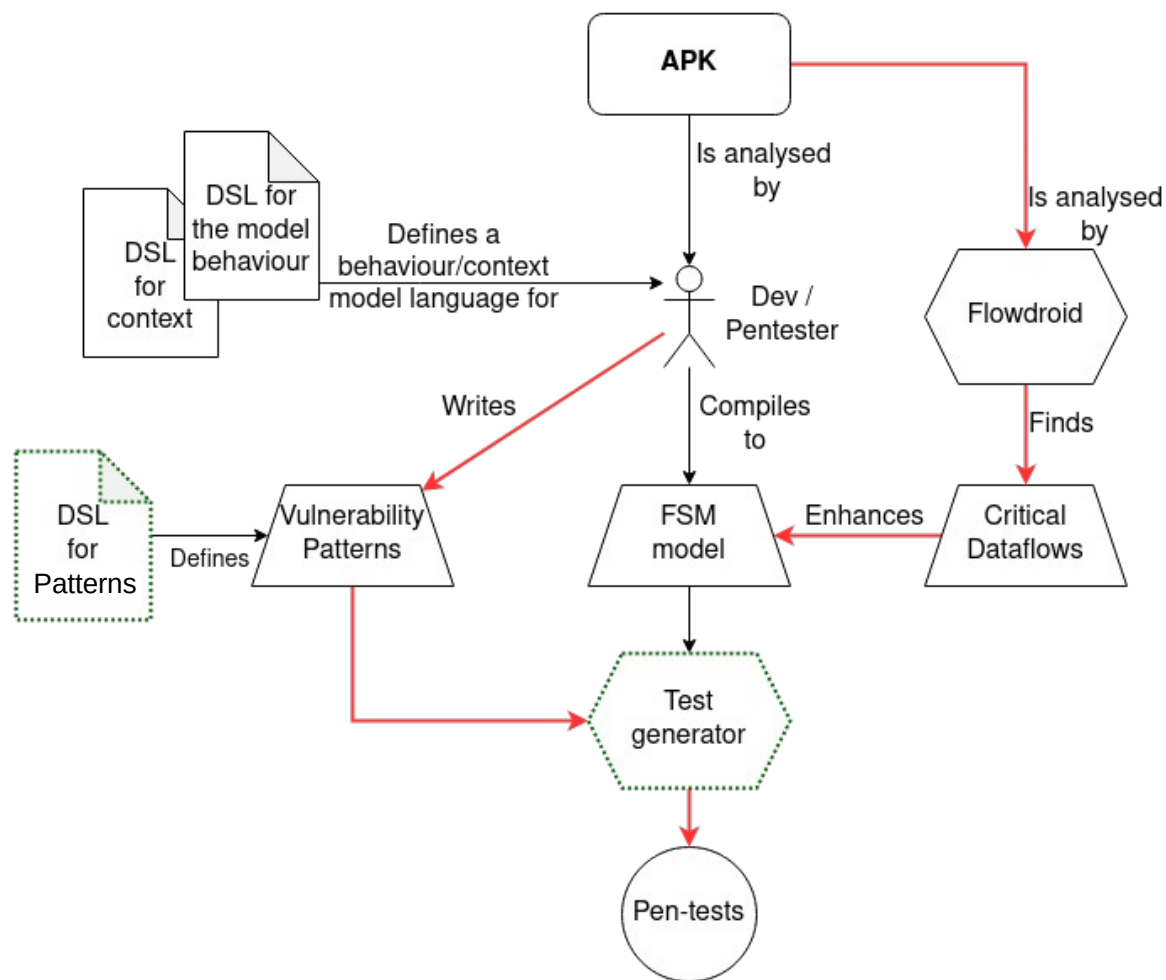
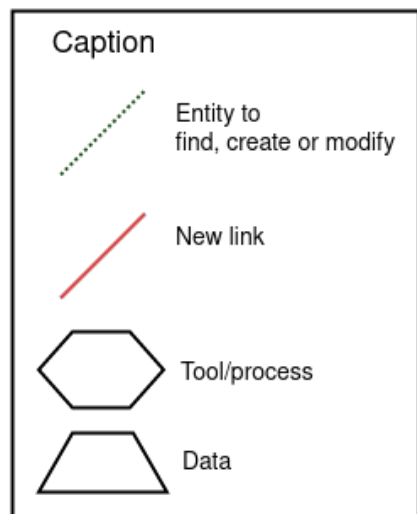
Ivan BAHEUX

Semaine 49

Rappel

But :

- Permettre à un développeur de tester puis de démontrer la présence des vulnérabilités définies par les patterns en fonction du contexte de l'application



Génération

- La génération semble finie :

```
-----  
state -> START  
event -> APP_STARTED  
state -> ENTER_PHRASE_ACTIVITY  
event -> TRANSLATE_BUTTON_CLICKED  
state -> TRANSLATE  
event -> TRANSLATION_SUCCEEDED with context _4g AND france  
state -> TRANSLATION_ACTIVITY  
event -> NONE  
state -> DISPLAY_TRANSLATION  
event -> BACK_BUTTON_CLICKED  
state -> ENTER_PHRASE_ACTIVITY  
event -> TRANSLATE_BUTTON_CLICKED  
state -> TRANSLATE  
event -> TERMINATE_BUTTON_CLICKED with context spain AND fast3g  
state -> EXIT  
-----
```

```
-----  
state -> START  
event -> APP_STARTED  
state -> ENTER_PHRASE_ACTIVITY  
event -> TRANSLATE_BUTTON_CLICKED  
state -> TRANSLATE  
event -> TRANSLATION_SUCCEEDED with context germany AND high_latency  
state -> TRANSLATION_ACTIVITY  
event -> NONE  
state -> DISPLAY_TRANSLATION  
event -> BACK_BUTTON_CLICKED  
state -> ENTER_PHRASE_ACTIVITY  
event -> TRANSLATE_BUTTON_CLICKED  
state -> TRANSLATE  
event -> TERMINATE_BUTTON_CLICKED with context germany AND high_latency  
state -> EXIT  
-----
```

- Quelques points à revoir ont été indiqués dans la documentation
- Couverture de code : AllTransitions+AllContexts

```
-----  
state -> START  
event -> APP_STARTED  
state -> ENTER_PHRASE_ACTIVITY  
event -> TRANSLATE_BUTTON_CLICKED  
state -> TRANSLATE  
event -> TRANSLATION_SUCCEEDED with context _4g AND uk  
state -> TRANSLATION_ACTIVITY  
event -> NONE  
state -> DISPLAY_TRANSLATION  
event -> TERMINATE_BUTTON_CLICKED  
state -> EXIT  
-----
```

Documentation

- Générée avec JavaDoc :

generate

```
public Set<TestPath> generate(fr.lc.is.castav.cbml.cbml.StateMachine sm,  
                             fr.lc.is.castav.cdml.cdml.ContextModel contextModel)
```

Description copied from interface: TestSubpathGenerator

Generates the set of subpaths for the statemachine and contextmodel sent in parameter The set of tests generated are considered subpaths as they may contain super states that link to other statemachines (See @link TestPathAggregator.java for full tests)

Specified by:

generate in interface TestSubpathGenerator

Parameters:

sm - Statemachine from which we generate every path

contextModel - ContextModel (*.cdl file) defining the possible contexts/situation and their meaning

- Pas la plus pratique mais se génère sans plugin additionnel

```
/**  
 * Generates the set of subpaths for the statemachine and contextmodel sent in parameter  
 * The set of tests generated are considered subpaths as they may contain super states that link to other statemachines  
 * (See @link TestPathAggregator.java for full tests)  
 *  
 * @param sm Statemachine from which we generate every path  
 * @param contextModel ContextModel (*.cdl file) defining the possible contexts/situation and their meaning  
 */  
public Set<TestPath> generate(StateMachine sm, ContextModel contextModel);
```

Contribution

Pour l'instant :

- Mise à jour de CBML (Context-Behaviour Modelling Language)
 - Ajout de Exported + (Signature/SignatureOrSystem/Dangerous/normal)
 - Pour chaque statemachine
- Dans la suite :
 - Ajout de Sink/Source dans chaque State

Travail à venir

Prévisionnel pour la suite :

Fin de la MaJ de CBML d'ici ce soir (au pire).

Choisir l'interface entre Vpat et le modèle (XML ou code direct)

Implementer Vpat avec le modèle