**Multilevel Policy Test Case**

**1. Basic Test Case**

**Subject:** Student\_A (rank=1), TA (rank=2)

**Resource:** R\_File1 (rank=1), R\_File2 (rank=2)

**Action:** Read

**Rule Property:** no read up

**2. Test case with Error Type #1 <access conflict>**

**Subject:** Student\_A (rank=1, rank=2), TA (rank=2) [Note: we assign two ranks to Student\_A to check the conflict]

**Resource:** R\_File1 (rank=1), R\_File2 (rank=2)

**Action:** Read

**Rule Property:** no read up

**Subjects and Objects in policy:** Student\_A, TA, R\_File1 and R\_File2.

**Property (used to check the error):**

(Student\_A = 1) & (Student\_A = 2) & (R\_File2 = 2) & (Action = read) -> decision = Permit

**Result on ACPT:** N/A.

**3. Test case with Error Type #2 <no object>**

**Subject:** Student\_A (rank=1), TA (rank=2)

**Resource:** R\_File1 (rank=1), R\_File2 (rank=2), R\_File3 (we just add R\_File3 to resources, without adding it to the policy)

**Action:** Read

**Rule Property:** no read up

**Subjects and Objects in Policy:** Student\_A, TA, R\_File1 and R\_File2.

**Property (used to check the error):**

(Student\_A = 1) & (File = R\_File3) & (Action = read) -> decision = Permit

(TA = 2) & (File = R\_File3) & (Action = read) -> decision = Permit

**Result on ACPT:**

(Student\_A = 1) & (File = R\_File3) & (Action = read) -> decision = Permit is **False**

(TA = 2) & (File = R\_File3) & (Action = read) -> decision = Permit is **False**

**4. Test case with Error Type #3 <undecided>**

**Subject:** Student\_A (rank=1), TA (rank=2), Student\_B (rank=1) (we assign the rank to Student\_B, but Student\_B is not included in the policy)

**Resource:** R\_File1 (rank=1), R\_File2 (rank=2)

**Action:** Read

**Rule Property:** no read up

**Subjects and Objects in Policy:** Student\_A, TA, R\_File1 and R\_File2.

**Property (used to check the error):**

(StudentB = 1) & (R\_File1 = 1) & (Action = read) -> decision = Permit

(StudentB = 1) & (R\_File2 = 2) & (Action = read) -> decision = Permit

**Result on ACPT:**

(StudentB = 1) & (R\_File1 = 1) & (Action = read) -> decision = Permit **is False**

(StudentB = 1) & (R\_File2 = 2) & (Action = read) -> decision = Permit **is False**