Review: RBAC

- RBAC supports both Static Separation of Duty (SSoD) and Dynamic Separation of Duty (DSoD). Let {RS, n} denote a constraint that no user should be assigned to n or more of the roles in the set RS. For simplicity, assume that there is no role hierarchy defined.
- Let {(r1, r2, r3, r4), 3} be a defined SSoD. Which of the following UA (User Assignment) sets are valid?
  - UA1 = {(*u*1, *r*1), (*u*2, *r*1), (*u*3, *r*1), (*u*1, *r*2), (*u*4, *r*2), (*u*5,*r*2), (*u*1,*r*3), (*u*2,*r*3), (*u*3,*r*3), (*u*4, *r*4)}
  - UA2 = {(*u*1, *r*1), (*u*3, *r*1), (*u*5, *r*1), (*u*1, *r*2), (*u*2, *r*2), (*u*3, *r*2), (*u*5, *r*2), (*u*2, *r*3), (*u*4, *r*3)}

- UA1 = {(*u*1*, r*1), (*u*2*, r*1), (*u*3*, r*1), (*u*1*, r*2), (*u*4*, r*2), (*u*5*,r*2), (*u*1*,r*3), (*u*2*,r*3), (*u*3*,r*3), (*u*4*, r*4)}

  – Invalid UA since u1 has 3 roles

- UA2 = {(*u*1*, r*1), (*u*3*, r*1), (*u*5*, r*1), (*u*1*, r*2), (*u*2*, r*2), (*u*3*, r*2), (*u*5*, r*2), (*u*2*, r*3), (*u*4*, r*3)}

  – valid UA since no users has 3 or more roles

What are the implications of having both the following pairs of SSoD and DSoD present in a system at the same time?

- ({*r*1, *r*2, *r*3, *r*4}, 3)∈ *SSoD* and ({*r*1, *r*2, *r*3, *r*4}, 3) ∈ *DSoD*
  - Fine although one of them is redundant.
- ({*r*1, *r*2, *r*3, *r*4}, 3)∈ *SSoD* and ({*r*1, *r*2, *r*3, *r*4}, 2) ∈ *DSoD*
  - Fine since DSoD is more stringent.
- ({*r*1, *r*2, *r*3, *r*4}, 2)∈ *SSoD* and ({*r*1, *r*2, *r*3, *r*4}, 3) ∈ *DSoD*
  - Fine though DSoD is redundant because it will never happen.