# MaxSuite – Role-Driven App Activation

Version: 1.0

Module ID: 02

Document Type: Execution Policy and Role Control

---

## 1. Purpose

This document defines the role-based execution model for activating MaxSuite AppCapsules. It ensures that only verified, policy-mapped roles can initiate application execution under defined constraints.

## 2. Role Binding Workflow

- User identity is mapped to a MaxWorkRoles-compatible role.  
- The role is matched against capsule metadata (`role\_required`, `scope`, `capsule\_id`).  
- Policy signature is checked via MaxReg.  
- Capsule is activated only if signature and role-binding pass validation.  
- Any mismatch triggers a block-and-audit response.

## 3. Runtime Role Enforcement

The executing environment must enforce role compliance throughout runtime. Capsule escalation (e.g., from Viewer to Editor) is prohibited unless explicitly defined in policy.  
  
Runtime must:  
- Lock capsule to initial role context  
- Disable privileged API paths outside allowed scope  
- Trigger audit event on any policy deviation attempt

## 4. Multi-Role Capsule Design (Optional)

Capsules may support multiple roles (e.g., Viewer, Contributor, Moderator) with distinct execution profiles.  
- Must include role selector or UI binding logic  
- Each role profile must be independently auditable and signed  
- Transitions between roles require capsule re-authentication