You're right — permissions are crucial for successful integration between **CodeBuild**, **CodePipeline**, and **EC2**. Below are the **minimum required IAM policies** for each service to interact smoothly.

**✅ 1. CodeBuild IAM Role Policies**

This role is assumed by CodeBuild and should have:

**Policy: codebuild-service-role-policy**

json

CopyEdit

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": [

"logs:\*",

"cloudwatch:\*",

"s3:\*",

"codebuild:\*"

],

"Resource": "\*"

},

{

"Effect": "Allow",

"Action": [

"secretsmanager:GetSecretValue"

],

"Resource": "\*"

},

{

"Effect": "Allow",

"Action": [

"ssm:SendCommand"

],

"Resource": "\*"

},

{

"Effect": "Allow",

"Action": [

"iam:PassRole"

],

"Resource": "\*"

}

]

}

🔸 **Note**: Restrict "Resource": "\*" by scoping to your actual SSM and Secrets resources if needed for security.

**✅ 2. EC2 IAM Role Policies**

If your EC2 instance interacts with SSM (for receiving commands), attach this policy:

**Managed Policy: AmazonSSMManagedInstanceCore**

It allows EC2 to:

* Register with SSM
* Receive commands
* Access S3 (for SSM document downloads, if needed)

**✅ 3. CodePipeline IAM Role Policies**

The CodePipeline service role must be able to:

* Access S3
* Trigger CodeBuild
* Pass roles

**Policy: codepipeline-service-role-policy**

json

CopyEdit

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": [

"codebuild:BatchGetBuilds",

"codebuild:StartBuild",

"codecommit:Get\*",

"codecommit:GitPull",

"s3:GetObject",

"s3:GetObjectVersion",

"s3:PutObject",

"iam:PassRole"

],

"Resource": "\*"

}

]

}

**✅ 4. SSM Permission Note (for send-command)**

Your CodeBuild role must have:

json

CopyEdit

{

"Effect": "Allow",

"Action": "ssm:SendCommand",

"Resource": [

"arn:aws:ec2:<region>:<account-id>:instance/<instance-id>",

"arn:aws:ssm:<region>::document/AWS-RunShellScript"

]

}

And make sure the EC2 is:

* Running
* In a public subnet (or has access to AWS SSM endpoints)
* Has AmazonSSMManagedInstanceCore policy