Terraform static code analysis

To check the code quality and security of terraform codes we perform below code analysis. Each analysis have their own unique feature and they will provide us results on different dimension of code analysis.

In the below section we will analyse each tool in details with below listed parameters

1. Terraform Validate
   1. Source: Terraform Build -in tool
   2. Checks: Structure and Coherence. It finds typos, bad variable declaration some basic errors
   3. Use Case: Terraform validate can be used as a first level of screening, even though it cannot perform deep scan it will remove all the typos and basic errors and saves our time.
2. TFLINT (Static Code Analysis):
   1. Source: Tflint (Package)
   2. Checks: TFLINT will statically analyse our code and reports errors and misconfiguration with respect to cloud provider. TFLINT is best tool to perform static code analysis for AWS provider
   3. Use Case: TFLINT can be used as static code analysis tool. We can check and confirm that we used a right resource detail with respect to cloud provider and all the configuration with respect to resource are inline with the predefined syntax.
   4. How it works:
      1. TFLINT uses a set of rules.
      2. Each cloud provider has a specific set of rules and we need to initialize the rules before running tflint using “**tflint init**” command.
      3. code analysis is performed using the initialized rules.
3. TFSEC (Checking security compliance of the code):
   1. Source: TFSEC (Package)
   2. Checks: Security vulnerability in the code
   3. Use Case: TFSEC is mainly used to make sure that our code doesn’t have any sensitive data exposed to the world. For example it checks whether the code contains access key and token hard coded etc.
   4. How it works:
      1. TFSEC also contains a list of predefined rules which comes with it
      2. It checks the code against the defined rule.
      3. We can also add more rules to improve the quality of analysis.