Viasat Project Name – “CodeProfiler”

Use Case: As we add expanding new features, the CI/CD team is catching up with the new features, and there is always 6-9 months of lag time for feature level testing. Also, we cannot expand CI/CD team linearly in proportion to the new feature additions. The number of data centers/deployment increases significantly for VS3/4/modernization; my goal is to focus CI/CD team only on System-level end-to-end testing, Customer issue creation, performance (to scale), and deployment pipeline.

It's tough to motivate the development team to do the functional automation as new feature deliverables are always in phase or more than the current workload.

So, to catch up, I’m thinking of initiating the project called “CodeProfiler” (it's not static code analyzer); The CodeProfiler uses golang advanced concepts like

- Golang Decorator
- Hooks to runtime compiler
- Extending Golang test_* built-in code etc.

What does CodeProfiler does?

Moving forward, the dev team will write Variadic functions with event structures. Most analyzers just vary the parameter and validate, but the CodeProfiler event structures pass the different states of the caller and make sure the function(s) behave as expected. In the case of non-Variadic functions, the dev team will have to write the wrapper. The event structure will have the verification code. The result event structures are created and validated at runtime using golang assert & another built-in mechanism based on the verification code. Its very well possible every function output would have specific code. Eventually, this code will give the map for ML, and the existing anomaly detection can be leveraged.

Expected output:

Build the framework of CodeProfiler and put together a POC for a couple of new features.

Mentor: Gopi Selvarajan.
Requested Resource: 2-3