# DevSecOps – Applying security along the software development pipeline

## Scoping document for application selection

The purpose of this questionnaire is to help with the selection of the customer’s/partner’s application is interested in adding security along their deployment strategy in Azure.

The goal of the coding event is to achieve a proof of concept over the course of 3 days that demonstrates this functionality.   
An ideal application is micro-service oriented and already running in a containerized environment.

## Questionnaire

**Application description:**

* Provide a simple description of the application and its purpose (What the application is doing? What is it used for?). Ideally provide a high-level architectural diagram.

**Details about the application targeted for the hack:**

* OS / Components, dependencies / Languages & Frameworks?
* Describe the network and security constraints?
* What is the target deployment platform (Kubernetes, Service Fabric, etc.)?

**DevSecOps questions:**

* Development processes:
  + Describe your current development methodology
  + How long is your average release cycle?
  + Do you have branching and/or security policies? Code review policies?
  + How do you relate work items with a specific build?
  + How do you analyze your code for potential secrets exposure (passwords, keys, connection strings, certificates, …)?
  + What are the changes that you are looking to explore/experiment with during the hack? (for example, implementation of static code analysis).
* Build processes:
  + How do you control that only approved dependencies (packages, libraries) can be used in your software?
  + How do you ensure organization policies are applied on you all your pipelines?
  + How do you perform container scanning for known vulnerabilities?
  + Can you corelate releases of your software with the associated versions of the dependencies and components?
* How do you integrate the different tests in your pipeline?
  + When do you run your unit tests?
  + When do you run integration tests?
  + What is your strategy for penetration testing? Is it integrated in your pipeline?
  + How do you test your application against known vulnerabilities?
* Release management:
  + How do you manage your releases?
  + Do you have “deployment rings”?
  + How do you perform rollback?
  + How do you manage your secrets in production vs test or QA? How do you perform secret rotation?
  + How often do you perform infrastructure refresh?
* What integration do you have between planning, development and production? Example: Work item tracking integrated with source control.

**What tools (including custom tools/scripts) are used for:**

* Version control
* Code analysis
* Build and Continuous Integration
* Organization policies
* Testing frameworks throughout
* Deployment automation
* Monitoring
* Disaster Recovery
* Package/container management