Chad Ballay

CYBR 450-342N

12/10/2020

**Week 5 Milestone**

**Abstract**

Guarding database credentials, tokens, passwords and other secrets becomes a non-trivial risk when scaling above a single developer. Securely storing these secrets yet providing a low inertia way for authentication, authorization, and access to use these secrets is key to encouraging developer engagement.

In previous times, teams would have their own walled garden for their codebase and implementation. There would also be a high degree of geographic colocation as well as time in role tenure established. That is no longer the case. Code bases are shared publicly. Developers will get onboarded/offboarded more frequently. Finally, the practice of being able to pass a post-it note with the credentials no longer is viable.

This paper will focus on mechanisms to increase the adoption rate and decrease the time to functional. To do so I will cover several approaches for increasing insight into the environment, demonstrating value, and finally positioning to ensure compliance with the policy of adoption. I will use a generic reference architecture as illustrative, but the recommendations will be non-implementation specific.

**Outline**

1. Introduction
2. Scenario and Overview
   1. Reliance on Wisdom of the Crowd
      1. Reduce cost of research
      2. Enables adoption of standardize toolsets
      3. Genericize the skillsets needed by employees
   2. Defining what is in scope
      1. 12 Factor App
      2. What are secrets
      3. What are configs
   3. Defining the reference architecture used
      1. CI/CD pipeline
      2. Source code repository
      3. K8 cluster
3. Track the inputs of secrets
   1. Source Code scanning of existing repositories
   2. User facing scanning upon commit
4. Communicate the message
   1. Policy written and broadly distributed
   2. Remedial notification process
5. Monitor the utilization
   1. Scan coverage of code base
      1. Truffle Hog
   2. Git commit statistics
   3. Remediation notifications communicated