- **Suggested Maturity Model:** Kubernetes Compliance Benchmarking

6. **kube-score**
   - **URL:** kube-score
   - **Description:** Static code analysis of your Kubernetes object definitions.
   - **Uses Method:** kube-score
   - **Suggested Maturity Model:** Code Quality and Security Scoring
7. **kube-hunter**
   - **URL:** kube-hunter
   - **Description:** Active scanner for Kubernetes (purple teaming).
   - **Uses Method:** kube-hunter
   - **Suggested Maturity Model:** Active Vulnerability Scanning
8. **Calico**
   - **URL:** Calico
   - **Description:** Open source networking and network security solution for containers.
   - **Uses Method:** Calico
   - **Suggested Maturity Model:** Network Security and Management
9. **Krane**
   - **URL:** Krane
   - **Description:** Simple Kubernetes RBAC static analysis tool.
   - **Uses Method:** krane
   - **Suggested Maturity Model:** Role-Based Access Control Analysis
10. **Starboard**
    - **URL:** Starboard
    - **Description:** Integrates security tools outputs into Kubernetes CRDs.
    - **Uses Method:** starboard
    - **Suggested Maturity Model:** Security Tool Integration and Reporting
11. **Gatekeeper**
    - **URL:** Gatekeeper
    - **Description:** Open policy agent gatekeeper for Kubernetes.
    - **Uses Method:** gatekeeper
    - **Suggested Maturity Model:** Policy Enforcement and Management
12. **Inspektor-gadget**
    - **URL:** Inspektor-gadget
    - **Description:** Collection of tools for debugging and inspecting Kubernetes.
    - **Uses Method:** inspector
    - **Suggested Maturity Model:** Debugging and Inspection Toolset

Containers

1. **Harbor**
   - **Description:** Trusted cloud native registry project.
   - **Uses Method:** Harbor
   - **Suggested Maturity Model:** Container Registry Security
2. **Anchore**
   - **Description:** Centralized service for inspection, analysis, and certification of container images.
   - **Uses Method:** Anchore
   - **Suggested Maturity Model:** Container Image Analysis and Certification
3. **Clair**
   - **Description:** Docker vulnerability scanner.
   - **Uses Method:** Clair
   - **Suggested Maturity Model:** Container Vulnerability Scanning
4. **Deepfence ThreatMapper**
   - **Description:** Powerful runtime vulnerability scanner for Kubernetes, virtual machines, and serverless.
   - **Uses Method:** ThreatMapper
   - **Suggested Maturity Model:** Runtime Vulnerability Scanning
5. **Docker Bench**
   - **Description:** Docker benchmarking against CIS.
   - **Uses Method:** Docker Bench
   - **Suggested Maturity Model:** Docker Security Benchmarking
6. **Falco**
   - **Description:** Container runtime protection.
   - **Uses Method:** Falco
   - **Suggested Maturity Model:** Runtime Security Monitoring
7. **Trivy**
   - **Description:** Comprehensive scanner for vulnerabilities in container images.
   - **Uses Method:** Trivy
   - **Suggested Maturity Model:** Container Image Vulnerability Scanning
8. **Notary**

- **Description:** Docker signing.
- **Uses Method:** Notary
- **Suggested Maturity Model:** Container Image Signing

9. **Cosign**
   - **Description:** Container signing.
   - **Uses Method:** Cosign
   - **Suggested Maturity Model:** Secure Container Signing
10. **Watchtower**
    - **Description:** Updates the running version of your containerized app.
    - **Uses Method:** Watchtower
    - **Suggested Maturity Model:** Continuous Container Update Management
11. **Grype**
    - **Description:** Vulnerability scanner for container images and filesystems.
    - **Uses Method:** Grype
    - **Suggested Maturity Model:** Comprehensive Container Scanning

Multi-Cloud

1. **Cloudsploit**
   - **Description:** Detection of security risks in cloud infrastructure.
   - **Uses Method:** Cloudsploit
   - **Suggested Maturity Model:** Cloud Security Risk Assessment
2. **ScoutSuite**
   - **Description:** NCCgroup multicloud scanning tool.
   - **Uses Method:** ScoutSuite
   - **Suggested Maturity Model:** Multicloud Security Scanning
3. **CloudCustodian**
   - **Description:** Multicloud security analysis framework.
   - **Uses Method:** CloudCustodian
   - **Suggested Maturity Model:** Cloud Security Governance
4. **CloudGraph**
   - **Description:** GraphQL API + Security for AWS, Azure, GCP, and K8s.
   - **Uses Method:** CloudGraph
   - **Suggested Maturity Model:** Cloud Resource Visualization and Security
5. **Steampipe**
   - **Description:** Query your cloud, code, logs & more with SQL. Open-source benchmarks & dashboards for security & insights.
   - **Uses Method:** Steampipe
   - **Suggested Maturity Model:** Cloud Data Querying and Security Insights

AWS

1. **Dragoneye**
   - **Description:** Dragoneye Indeni AWS scanner.
   - **Uses Method:** Dragoneye
   - **Suggested Maturity Model:** AWS Security Scanning
2. **Prowler**
   - **Description:** Command line tool for AWS security assessment, auditing, hardening, and incident response.
   - **Uses Method:** Prowler
   - **Suggested Maturity Model:** AWS Security Assessment and Auditing
3. **AWS Inventory**
   - **Description:** Helps to discover all AWS resources created in an account.
   - **Uses Method:** AWS Inventory
   - **Suggested Maturity Model:** AWS Resource Management and Security
4. **PacBot**
   - **Description:** Policy as Code Bot (PacBot).
   - **Uses Method:** PacBot
   - **Suggested Maturity Model:** Policy as Code for AWS
5. **Komiser**
   - **Description:** Monitoring dashboard for costs and security.
   - **Uses Method:** Komiser
   - **Suggested Maturity Model:** AWS Cost and Security Monitoring
6. **Cloudsplaining**
   - **Description:** IAM analysis framework.
   - **Uses Method:** Cloudsplaining
   - **Suggested Maturity Model:** AWS IAM Security Analysis
7. **ElectricEye**
   - **Description:** Continuously monitor your AWS services for configurations.

- **Uses Method:** ElectricEye
- **Suggested Maturity Model:** Continuous AWS Security Monitoring

8. **Cloudmapper**
   - **Description:** Helps you analyze your Amazon Web Services environments.
   - **Uses Method:** CloudMapper
   - **Suggested Maturity Model:** AWS Environment Analysis
9. **Cartography**
   - **Description:** Consolidates AWS infrastructure assets and the relationships between them in an intuitive graph.
   - **Uses Method:** Cartography
   - **Suggested Maturity Model:** AWS Asset Visualization and Management
10. **Policy Sentry**
    - **Description:** IAM Least Privilege Policy Generator.
    - **Uses Method:** Policy Sentry
    - **Suggested Maturity Model:** AWS IAM Policy Optimization

... (and so on for each tool in the AWS category)

Google Cloud Platform

1. **Forseti**
   - **Description:** Complex security orchestration and scanning platform.
   - **Uses Method:** Forseti
   - **Suggested Maturity Model:** GCP Security Orchestration
2. **GCP Insights**
   - **Description:** Visualize GCP inventory and permissions through relationship graphs.
   - **Uses Method:** GCP Insights
   - **Suggested Maturity Model:** GCP Security Visualization
3. **GCP Compliance**
   - **Description:** Check compliance of GCP configurations to security best practices.
   - **Uses Method:** GCP Compliance
   - **Suggested Maturity Model:** GCP Compliance Assessment

Microsoft Azure

1. **Azure Insights**
   - **Description:** Visualize Azure inventory and permissions through relationship graphs.
   - **Uses Method:** Azure Insights
   - **Suggested Maturity Model:** Azure Security Visualization
2. **Azure Compliance**
   - **Description:** Check compliance of Azure configurations to security best practices.
   - **Uses Method:** Azure Compliance
   - **Suggested Maturity Model:** Azure Compliance Assessment

Policy as Code

1. **Open Policy Agent (OPA)**
   - **Description:** General-purpose policy engine for unified, context-aware policy enforcement across the stack.
   - **Uses Method:** OPA
   - **Suggested Maturity Model:** Cross-Platform Policy Enforcement
2. **Kyverno**
   - **Description:** Policy engine designed for Kubernetes.
   - **Uses Method:** Kyverno
   - **Suggested Maturity Model:** Kubernetes Policy Management
3. **Inspec**
   - **Description:** Open-source testing framework for infrastructure with language for compliance, security, and policy requirements.
   - **Uses Method:** Inspec
   - **Suggested Maturity Model:** Infrastructure Compliance Testing
4. **Cloud Formation Guard**
   - **Description:** Cloud Formation policy as code.
   - **Uses Method:** CF-Guard
   - **Suggested Maturity Model:** AWS CloudFormation Policy Management
5. **CNSpec**
   - **Description:** Cloud-native and powerful Policy as Code engine for security and compliance assessment across various infrastructures.
   - **Uses Method:** CNSpec
   - **Suggested Maturity Model:** Multi-Infrastructure Security Assessment

Chaos Engineering

1. **Chaos Mesh**
   - **Description:** Cloud-native Chaos Engineering platform that orchestrates chaos on Kubernetes environments.
   - **Uses Method:** Chaos Mesh
   - **Suggested Maturity Model:** Kubernetes Resilience Testing
2. **Chaos Monkey**
   - **Description:** Randomly terminates instances in production to ensure service resilience.
   - **Uses Method:** Chaos Monkey
   - **Suggested Maturity Model:** Production Resilience Assurance
3. **Chaos Engine**
   - **Description:** Tool designed to intermittently destroy or degrade application resources in cloud infrastructure.
   - **Uses Method:** Chaos Engine
   - **Suggested Maturity Model:** Cloud Infrastructure Resilience Testing
4. **Chaoskube**
   - **Description:** Test how your system behaves under arbitrary pod failures.
   - **Uses Method:** Chaoskube
   - **Suggested Maturity Model:** Kubernetes Pod Failure Testing
5. **Kube-Invaders**
   - **Description:** Gamified chaos engineering tool for Kubernetes.
   - **Uses Method:** Kube-Invaders
   - **Suggested Maturity Model:** Kubernetes Gamified Resilience Testing

Infrastructure as Code Security

1. **KICS**
   - **Description:** Checkmarx security testing opensource for Infrastructure as Code (IaC).
   - **Uses Method:** KICS
   - **Suggested Maturity Model:** Open Source IaC Security Testing
2. **Checkov**
   - **Description:** Static code analysis tool for infrastructure-as-code.
   - **Uses Method:** Checkov
   - **Suggested Maturity Model:** Static Code Analysis for IaC
3. **tfsec**
   - **Description:** Uses static analysis of Terraform templates to spot potential security issues.
   - **Uses Method:** tfsec
   - **Suggested Maturity Model:** Terraform Security Analysis
4. **terrascan**
   - **Description:** Static code analyzer for Infrastructure as Code.
   - **Uses Method:** terrascan
   - **Suggested Maturity Model:** IaC Code Scanning
5. **cfsec**
   - **Description:** Scans CloudFormation configuration files for security issues.
   - **Uses Method:** cfsec
   - **Suggested Maturity Model:** CloudFormation Security Scanning
6. **cfn_nag**
   - **Description:** Looks for insecure patterns in CloudFormation.
   - **Uses Method:** cfn_nag
   - **Suggested Maturity Model:** CloudFormation Pattern Analysis
7. **Sysdig IaC Scanner Action**
   - **Description:** Scans your repository with Sysdig IAC Scanner and reports vulnerabilities.
   - **Uses Method:** Sysdig IaC Scanner
   - **Suggested Maturity Model:** Repository Vulnerability Scanning
8. **Terraform Compliance for AWS**
   - **Description:** Checks compliance of Terraform configurations to AWS security best practices.
   - **Uses Method:** Terraform Compliance
   - **Suggested Maturity Model:** AWS Terraform Compliance
9. **Terraform Compliance for Azure**
   - **Description:** Checks compliance of Terraform configurations to Azure security best practices.
   - **Uses Method:** Terraform Compliance
   - **Suggested Maturity Model:** Azure Terraform Compliance
10. **Terraform Compliance for GCP**
    - **Description:** Checks compliance of Terraform configurations to GCP security best practices.
    - **Uses Method:** Terraform Compliance
    - **Suggested Maturity Model:** GCP Terraform Compliance
11. **Terraform Compliance for OCI**

- **Description:** Checks compliance of Terraform configurations to OCI security best practices.
- **Uses Method:** Terraform Compliance
- **Suggested Maturity Model:** OCI Terraform Compliance

Orchestration

1. **StackStorm**
   - **Description:** Platform for integration and automation across services and tools supporting event-driven security.
   - **Uses Method:** StackStorm
   - **Suggested Maturity Model:** Event-Driven Security Automation
2. **Camunda**
   - **Description:** Workflow and process automation.
   - **Uses Method:** Camunda
   - **Suggested Maturity Model:** Process Automation for Security
3. **DefectDojo**
   - **Description:** Security orchestration and vulnerability management platform.
   - **Uses Method:** DefectDojo
   - **Suggested Maturity Model:** Vulnerability Management Orchestration
4. **Faraday**
   - **Description:** Security suite for Security Orchestration, vulnerability management, and centralized information.
   - **Uses Method:** Faraday
   - **Suggested Maturity Model:** Comprehensive Security Orchestration

Controls

Development Environment

Security Control: Secure Code Training

- **Description:** Training developers in secure coding to reduce security bugs and enhance system design.
- **Control(s):** CIS8, APRA234, NIST 800-53B, SSDF1.1, ISO27001

Security Control: Source Code Versioning

- **Description:** Using Version Control Systems for peer review, auditable history, and consistent work patterns.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: .gitignore

- **Description:** Preventing accidental commits of sensitive data with .gitignore files.
- **Control(s):** APRA234, CIS8, NIST 800-53B, SSDF1.1

Security Control: Pre-Commit Hook Scans

- **Description:** Using Pre-Commit Hooks for security scans to provide timely feedback and prevent vulnerabilities.
- **Control(s):** APRA234, CIS8, NIST 800-53B, SSDF1.1

Security Control: Commit Signing

- **Description:** Signing all commits to verify author authenticity.
- **Control(s):** APRA234, CIS8, NIST 800-53B, SSDF1.1

Security Control: IDE Plugins

- **Description:** Implementing IDE plugins to highlight security issues in real-time.
- **Control(s):** APRA234, CIS8, NIST 800-53B, SSDF1.1

Security Control: Local Software Composition Analysis

- **Description:** Finding and fixing libraries with known security issues.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: Local Static Code Analysis

- **Description:** Identifying and resolving security vulnerabilities in source code.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: Local Sensitive Data Analysis

- **Description:** Auditing repositories for secrets, credentials, and API keys.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: Application Baseline

- **Description:** Creating a comprehensive "recipe" for building applications considering risk, compliance, and technical components.
- **Control(s):** APRA234, CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Source Code Management (SCM)

Security Control: Source Code Management

- **Description:** Using centralized SCM systems like Bitbucket, GitHub, or Gitlab.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: User Roles

- **Description:** Creating unique user and team roles for tailored source code access.
- **Control(s):** APRA234, CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Security Control: SSH

- **Description:** Accessing repositories using SSH protocol instead of HTTPS.
- **Control(s):** APRA234, CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Security Control: GPG Key

- **Description:** Adding a GPG key to SCM providers for identity verification.
- **Control(s):** APRA234, CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Security Control: Multi-Factor Authentication

- **Description:** Ensuring MFA is used for SCM interactions.
- **Control(s):** APRA234, CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Security Control: Server Side Git Hook

- **Description:** Utilizing server-side git hooks for automatic scans.
- **Control(s):** APRA234, CIS8, NIST 800-53B, SSDF1.1

Security Control: Developer Collaboration

- **Description:** Using collaboration tools for documenting software changes.
- **Control(s):** APRA234, CIS8, NIST 800-53B, SSDF1.1

Security Control: Pull Requests

- **Description:** Enforcing pull or merge requests for code verification.
- **Control(s):** APRA234, CIS8, ISO27001, NIST 800-53B, SSDF1.1

Security Control: Peer Reviews

- **Description:** Enforcing peer reviews to enhance code quality and security.
- **Control(s):** APRA234, CIS8, ISO27001, NIST 800-53B, SSDF1.1

Security Control: CODEOWNERS

- **Description:** Creating a CODEOWNERS file to identify repository owners.
- **Control(s):** APRA234, CIS8, ISO27001, NIST 800-53B, SSDF1.1

Security Control: SECURITY.md

- **Description:** Creating a SECURITY.md file for reporting security issues.
- **Control(s):** APRA234, CIS8, ISO27001, NIST 800-53B, SSDF1.1

Security Control: .github Repository

- **Description:** Creating a .github repository for standard files across the organization.
- **Control(s):** APRA234, CIS8, ISO27001, NIST 800-53B, SSDF1.1

CI/CD Pipelines and Automation

Security Control: CI/CD Pipeline

- **Description:** Implementing a CI/CD pipeline for continuous integration and deployment.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, SSDF1.1

Security Control: Application Environments

- **Description:** Creating separate environments for development, staging, and production.
- **Control(s):** CIS8, ISM GSD, ISO27001, SSDF1.1

Security Control: Application Data Separation

- **Description:** Ensuring data separation between dev/test and production environments.
- **Control(s):** CIS8, ISM GSD, ISO27001, SSDF1.1

Security Control: CI/CD Administration

- **Description:** Enforcing user or team roles for CI/CD pipeline management.
- **Control(s):** CIS8, ISM GSD, ISO27001, SSDF1.1

Security Control: Credential Store

- **Description:** Creating a secure place for storing sensitive credentials.
- **Control(s):** APRA234, CIS8, ISM GSD, NIST 800-53.2b, SSDF1.1

Security Control: Centralized Software Composition Analysis

- **Description:** Scanning source code for vulnerable libraries from within a CD stage.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53.2a, SSDF1.1

Security Control: Centralized Static Code Analysis

- **Description:** Scanning source code for vulnerabilities from within a CD stage.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53.2b, SSDF1.1

Security Control: Centralized Sensitive Data Analysis

- **Description:** Scanning source code for secrets and credentials from within a CD stage.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: DAST

- **Description:** Scanning running applications for vulnerabilities.
- **Control(s):** CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Security Control: Transient Test Compute

- **Description:** Ensuring up-to-date compute resources in CI/CD pipelines.
- **Control(s):** CIS8, ISM GSD, ISO27001, SSDF1.1

Security Control: Harden Transient Compute

- **Description:** Hardening transient compute resources used in pipelines.
- **Control(s):** CIS8, ISM GSM, ISM GOSH, SSDF1.1

Valid SSL Certificate

- **Description:** Create and use a valid SSL certificate for each application URL, or implement a wildcard cert.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Encrypt Traffic

- **Description:** Encrypt all traffic that's public facing.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1, DSOMM: Infrastructure Hardening Level 1

Redirect to HTTPS

- **Description:** Configure web service to redirect all inbound requests to port 80 to the secure HTTPS endpoint.
- **Control(s):** CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1, DSOMM: Application Hardening Level 1

HSTS

- **Description:** Enable HSTS in your webserver, load balancer or CDN.
- **Control(s):** CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1, DSOMM: Application Hardening Level 1

CSP

- **Description:** Enable content security policy (CSP) in the webserver, load balancer or CDN.
- **Control(s):** CIS8, ISM GSD, ISO27001, NIST 800-53B, DSOMM: Application Hardening Level 1

Use Current Software

- **Description:** Use the most recent versions of application components, languages, frameworks and operating systems.
- **Control(s):** CIS8, ISM GSD, ISO27001, SSDF1.1, DSOMM: Application Hardening Level 1

Alternative Deployment

- **Description:** Have tested and working alternative way to deploy changes to your application other than using your standard process with GitHub or Bitbucket in case they go down. This must include the ability to push to PROD from local in emergencies.
- **Control(s):** CIS8, NIST 800-53B, SSDF1.1

security.txt

- **Description:** Create a security.txt file in the root of your application so people know how to contact you about security issues.
- **Control(s):** CIS8, ISM GSD, SSDF1.1

X-Forwarded-By

- **Description:** Configure your webservers, load balancers & web proxies to include the X-Forwarded-By: header.
- **Control(s):** APRA234 ATM D-2-d-i, CIS8, NIST 800

Logging

- **Description:** Collect application logs in realtime and send to centralized storage or SIEM.
- **Control(s):** CIS8 16.11, APRA234, ISM GSM, NIST 800, SSDF1.1, DSOMM: Logging Level 1

WAF

- **Description:** Implement a web application firewall (WAF) to protect your application from known attacks.
- **Control(s):** APRA234, CIS8, NIST 800-53.2a

CDN

- **Description:** Use a content delivery network (CDN) whenever possible to add availability and security to your applications.
- **Control(s):** APRA234, CIS8, ISM GN, NIST 800-53.2a, DSOMM: Application Hardening Level 1

Harden Operating System

- **Description:** Harden operating system using industry best practices from CIS, ISM, etc.
- **Control(s):** CIS8, ISM GSM, ISM GOSH, SSDF1.1

Encrypt Storage

- **Description:** Encrypt all filesystems, disks and cloud storage.
- **Control(s):** CIS8, NIST 800-50b, SSDF1.1, DSOMM: Infrastructure Hardening Level 1

SBOM

- **Description:** Generate a real-time software bill-of-materials (SBOM).
- **Control(s):** CIS8, ISM GSD, NIST 800-53B, SSDF1.1, DSOMM: Build Level 2

Monitor Application

- **Description:** Monitor your application in real-time so you know when its state changes for the worse (or better). This includes uptime, performance and security monitoring.
- **Control(s):** CIS8, NIST 800-53B, SSDF1.1, DSOMM: Monitoring Levels 1 - 4

Cloud Security Posture

- **Description:** If your application is deployed in the cloud or uses cloud native services then a solution should be employed to verify that those cloud resources are secure and follow best practices.
- **Control(s):** CIS8, NIST 800-53B, SSDF1.1

Centralized Container Analysis

- **Description:** Scan any containers built for deployment for vulnerabilities.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53.2a, SSDF1.1, DSOMM: Build Level 2

IaC

- **Description:** Use infrastructure as code to build all application environments.
- **Control(s):** CIS8, ISM GSM, ISM GOSH, SSDF1.1, DSOMM: Infrastructure Hardening Level 3

TLS 1.3

- **Description:** Use TLS 1.3 instead of earlier versions. TLS 1.2 is still okay, but you should enable 1.3 soon as its more secure than any of the earlier versions.
- **Control(s):** APRA234, CIS8, ISM GSD, ISO27001, NIST 800-53B, SSDF1.1

Organizational Techniques

Penetration Testing

- **Description:** Have your application pentested regularly.
- **Control(s):** CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Threat Modeling

- **Description:** Build a collaborative way for developers and security staff to understand the threat landscape for an individual application.

- **Control(s):** CIS8, ISM GSD, NIST 800-53B, SSDF1.1, DSOMM: Designing Levels 1 - 4

SIEM

- **Description:** Implement a SIEM and send all application, system and cloud logs to it.
- **Control(s):** CIS8, NIST 800-53B, SSDF1.1

Attack Surface Management

- **Description:** Identify public facing resources via automation.
- **Control(s):** CIS8, NIST 800-53B, SSDF1.1, DSOMM: Logging Level 4, DSOMM: Dynamic Depth for Applications Level 2

Sovereignty

- **Description:** Require that all code is written in, stored in, or otherwise served from a location and/or sovereignty that aligns with your org's requirements.
- **Control(s):** ISM GCSR, ISO27001

Vulnerability Disclosure

- **Description:** Create and publish a set of procedures to let people contact you when they find security issues in your app.
- **Control(s):** CIS8, ISM GSD, SSDF1.1

Bug Bounty

- **Description:** Setup a bug bounty program to incentivize security researchers to tell you about vulnerabilities they find.
- **Control(s):** CIS8, ISM GSD, NIST 800-53B, SSDF1.1

Licensing

- **Description:** Track licensing of all software that your organizations uses or depends on. Utilize a license tracking solution to enable searching for license types.
- **Control(s):** ?

Vulnerability Aggregation and Management

- **Description:** Implement a system that aggregates vulnerability data from multiple tools and allows teams to prioritize, collaborate on, and manage the lifecycle of said vulnerabilities.
- **Control(s):** CIS8, ISM GSD, SSDF1.1