



# Bringing the Security Mindset to Your Team

Greta Jocyte

Sr Technical Program Manager  
Microsoft

[linkedin.com/in/gretajocyte](https://www.linkedin.com/in/gretajocyte)



# My Career



# Did you know?

**4.35 Million USD**

Average cost of a data breach globally

**5.57 Million USD**

Average cost of an organizations data breach with high compliance failures

**227 Days**

Average time it takes to remediate a data breach

TECH · TMOBILE

**T-Mobile to pay \$350 million to customers due to data breach**

Technology

**Meta Fined \$277 Million for Leak of Half a Billion Users**

- Fine by Irish watchdog follows probe of EU privacy violations
- Penalty is third-largest under the EU's GDPR data law

**Instagram fined \$400 million for failing to protect children's data**

By Reuters

**What can you and your team do?**

# Engineering Best Practices

---

- OWASP Top 10 or SANS Top 25, CSE Engineering Playbook
- Pull Request Policy
- Logging & Error Handling
- Vulnerability mgmt. policy





## Code With Engineering Playbook

[CSE Code-With Engineering Playbook](#)[Who We Are](#)[Engineering Fundamentals Checklist](#)[Structure of a Sprint](#)[Accessibility](#)[Agile development](#)[Automated testing](#)[Code reviews](#)[FAQ](#)[Inclusion in Code Review](#)[Pull Requests](#)[Code Review Tools](#)[Evidence and measures](#)[Process guidance](#)[Pull request template](#)[Recipes](#)[Continuous delivery](#)[Continuous integration](#)[Design](#)[Developer experience](#)[Documentation](#)[Engineering feedback](#)[Machine learning](#)[Observability](#)[Privacy](#)[Reliability](#)[Resources](#)

## Pull Requests

Changes to any main codebase - main branch in Git repository, for example - must be done using pull requests (PR).

Pull requests enable:

- Code inspection - see [Code Reviews](#)
- Running automated qualification of the code
  - Linters
  - Compilation
  - Unit tests
  - Integration tests etc.

The requirements of pull requests can and should be enforced by policies, which can be set in the most modern version control and work item tracking systems. See [Evidence and Measures section](#) for more information.

## General Process

1. Implement changes based on the well-defined description and acceptance criteria of the task at hand
2. Then, before creating a new pull request: \* Make sure the code conforms with the agreed coding conventions \* This can be partially automated using linters \* Ensure the code compiles and runs without errors or warnings \* Write and/or update tests to cover the changes and make sure all new and existing tests pass \* Write and/or update the documentation to match the changes
3. Once convinced the criteria above are met, create and submit a new pull request adhering to the [pull request template](#)

**Have I reviewed our best  
practices before committing  
my code?**

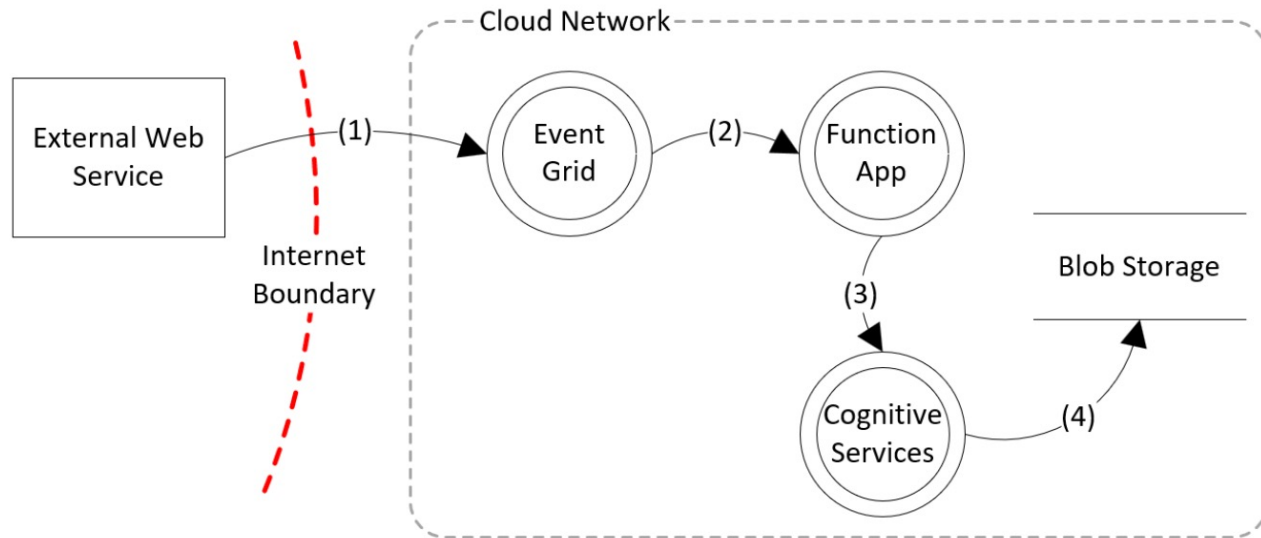
# Threat Modelling

- For major architectural changes
- Use a framework (STRIDE, MITRE ATT&CK)
- Upskill *at least one dev*





## Diagram



## Assets

Asset	Entry Point	Trust Level
Azure Event Grid	HTTP endpoint	Access Key or Shared Access Token
Azure Function	HTTP endpoint	AAD identity
Azure Cognitive Services	HTTP endpoint	AAD identity
Azure Blob Storage	HTTP endpoint	AAD identity
Image files	Azure Storage public endpoint	AAD identity or Shared Key

## Threats and Mitigations

#	Threat	Node	Mitigation	Status
1	Plaintext data in transit could be intercepted via a man-in-the-middle (MitM) attack. Sensitive data could be exposed or tampered to allow further exploits.	Data flow # 1, 2, 3, 4	Use TLS to encrypt all HTTP network traffic. Use other mechanisms, such as IPSec, to encrypt non-HTTP network traffic that contains customer or confidential data. Applications should never accept unencrypted requests.	Mitigated. Services set to only accept TLS 1.2 connections.

- **Is my data encrypted at rest?**
- **Is my data encrypted in transit?**
- **Who has access and how is this enforced?**
- **....**



# Automate

- **Container dependency scanning**
- **Static Code Scanning Tools**
- **Git Commit Hooks (Cred Scanners, Linting)**
- **Secrets Rotation**
- **Use pre-built security frameworks**

## Synopsis

```
git secrets --scan [-r|--recursive] [--cached] [--no-index] [--untracked] [<files>...]  
git secrets --scan-history  
git secrets --install [-f|--force] [<target-directory>]  
git secrets --list [--global]  
git secrets --add [-a|--allowed] [-l|--literal] [--global] <pattern>  
git secrets --add-provider [--global] <command> [arguments...]  
git secrets --register-aws [--global]  
git secrets --aws-provider [<credentials-file>]
```

## Description

`git-secrets` scans commits, commit messages, and `--no-ff` merges to prevent adding secrets into your git repositories. If a commit, commit message, or any commit in a `--no-ff` merge history matches one of your configured prohibited regular expression patterns, then the commit is rejected.

## Installing git-secrets

`git-secrets` must be placed somewhere in your PATH so that it is picked up by `git` when running `git secrets`.

### \*nix (Linux/macOS)

You can use the `install` target of the provided Makefile to install `git secrets` and the man page. You can customize the install path using the `PREFIX` and `MANPREFIX` variables.

```
make install
```

## Windows

# Container Scanning Tools

1. [Trivy](#) - a simple and comprehensive vulnerability scanner for containers (doesn't support Windows containers)
2. [Aqua](#) - dependency and container scanning for applications running on AKS, ACI and Windows Containers. Has an integration with AzDO pipelines.
3. [Dependency-Check Plugin for SonarQube](#) - OnPrem dependency scanning
4. [Mend \(previously WhiteSource\)](#) - Open Source Scanning Software

**What are our automated tools  
*not* catching?**



# Consider Infrastructure

- **Kubernetes security best practices**
- **Runtime security & Binary authorization**
- **Cloud Provider Security Configuration**

🔍 Search

- ▶ Home
- ▶ Getting started
- ▼ Concepts
  - ▶ Overview
  - ▶ Cluster Architecture
  - ▶ Containers
  - ▶ Windows in Kubernetes
  - ▶ Workloads
  - ▶ Services, Load Balancing, and Networking
  - ▶ Storage
  - ▶ Configuration
  - ▼ Security
    - Overview of Cloud Native Security
    - Pod Security Standards
    - Pod Security Admission
    - Pod Security

## Authentication & Authorization

- `system:masters` group is not used for user or component authentication after bootstrapping.
- The kube-controller-manager is running with `--use-service-account-credentials` enabled.
- The root certificate is protected (either an offline CA, or a managed online CA with effective access controls).
- Intermediate and leaf certificates have an expiry date no more than 3 years in the future.
- A process exists for periodic access review, and reviews occur no more than 24 months apart.
- The [Role Based Access Control Good Practices](#) is followed for guidance related to authentication and authorization.

After bootstrapping, neither users nor components should authenticate to the Kubernetes API as `system:masters`. Similarly, running all of kube-controller-manager as `system:masters` should be avoided. In fact, `system:masters` should only be used as a break-glass mechanism, as opposed to an admin user.

## Network security

- CNI plugins in-use supports network policies.
- Ingress and egress network policies are applied to all workloads in the cluster.
- Default network policies within each namespace, selecting all pods, denying everything, are in place.
- If appropriate, a service mesh is used to encrypt all communications inside of the cluster.
- The Kubernetes API, kubelet API and etcd are not exposed publicly on Internet.



## Identity and access management

Recommendation	Comments	Defender for Cloud
Centralize VM authentication.	You can centralize the authentication of your Windows and Linux VMs by using <a href="#">Azure Active Directory authentication</a> .	-

## Monitoring

Recommendation	Comments	Defender for Cloud
Monitor your VMs.	You can use <a href="#">Azure Monitor for VMs</a> to monitor the state of your Azure VMs and virtual machine scale sets. Performance issues with a VM can lead to service disruption, which violates the security principle of availability.	-

## Networking


Recommendation	Comments	Defender for Cloud
Restrict access to management ports.	Attackers scan public cloud IP ranges for open management ports and attempt "easy" attacks like common passwords and known unpatched vulnerabilities. You can use <a href="#">just-in-time (JIT) VM access</a> to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy connections to VMs when they're needed.	-
Limit network access.	Network security groups allow you to restrict network access and control the number of exposed endpoints. For more information, see <a href="#">Create, change, or delete a network security group</a> .	-

**Is my change to our  
infrastructure in compliance  
with our security policies?**



# Next Steps

---

- Evaluate where your team is on the journey to “shifting left”
  - Take a look at the CSE Engineering Playbook, OWASP, SANS, and other security resources for guidance
  - Get at least one person on your team upskilled on security fundamentals
- 



# THANK YOU

Search for “CSE Engineering Playbook”



[linkedin.com/in/gretajocyte](https://www.linkedin.com/in/gretajocyte)