INSUFFICIENT LOGGING & MONITORING

@triplettmike1

Download slide deck from snowfroc.com

Post questions to slack

https://miketriplett.slack.com/archives/CU143MY67

Overview

A10-Insufficient Logging & Monitoring

A9-Using Components with Known Vulnerabilities

What's Next for Developers

Exploitability Prevalence Detectability Technical

- **Threat Agents/Attack Vectors**: Exploitation of insufficient logging and monitoring is the bedrock of nearly every major incident. Attackers rely on the lack of monitoring and timely response to achieve their goals without being detected.
- **Security Weakness**: This issue is included in the Top 10 based on an industry survey. One strategy for determining if you have sufficient monitoring is to examine the logs following penetration testing. The testers' actions should be recorded sufficiently to understand what damages they may have inflicted.
- **Impacts**: Most successful attacks start with vulnerability probing. Allowing such probes to continue can raise the likelihood of successful exploit to nearly 100%. In 2016, identifying a breach took an average of 191 days – plenty of time for damage to be inflicted.

https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2017/Top_10-2017_A10-Insufficient_Logging%252526Monitoring

Is the Application Vulnerable?

Insufficient logging, detection, monitoring and active response occurs any time:

- Auditable events, such as logins, failed logins, and high-value transactions are not logged.
- Warnings and errors generate no, inadequate, or unclear log messages.
- Logs of applications and APIs are not monitored for suspicious activity.
- Logs are only stored locally.
- Appropriate alerting thresholds and response escalation processes are not in place or effective.
- Penetration testing and scans by DAST tools (such as OWASP ZAP) do not trigger alerts.
- The application is unable to detect, escalate, or alert for active attacks in real time or near real time.

You are vulnerable to information leakage if you make logging and alerting events visible to a user or an attacker (see A3:2017-Sensitive Data Exposure).

How to Prevent

As per the risk of the data stored or processed by the application:

- Ensure all login, access control failures, and server-side input validation failures can be logged with sufficient user context to identify suspicious or malicious accounts, and held for sufficient time to allow delayed forensic analysis.
- Ensure that logs are generated in a format that can be easily consumed by a centralized log management solutions.
- Ensure high-value transactions have an audit trail with integrity controls to prevent tampering or deletion, such as append-only database tables or similar.
- Establish effective monitoring and alerting such that suspicious activities are detected and responded to in a timely fashion.
- Establish or adopt an incident response and recovery plan, such as NIST 800-61 rev 2 or later.

There are commercial and open source application protection frameworks such as OWASP AppSensor, web application firewalls such as ModSecurity with the OWASP ModSecurity Core Rule Set, and log correlation software with custom dashboards and alerting.

API10:2019 Insufficient Logging & Monitoring

Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems to tamper with, extract, or destroy data. Most breach studies demonstrate the time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.

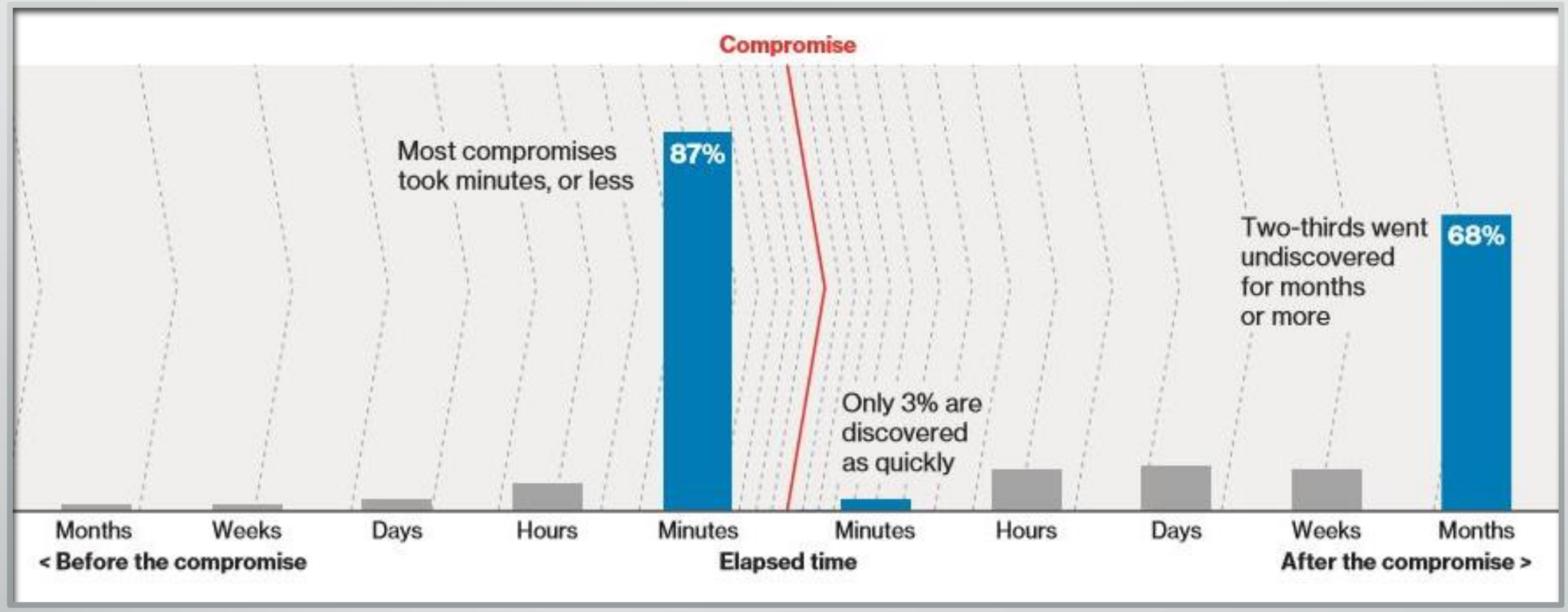
https://owasp.org/www-project-api-security/

Key Concepts

Four Pillars

- Log
  - The right information in the right place
- Monitor
  - Automate, but also manual review periodically
- Alert
  - Beware alert fatigue
- Respond
  - If tuned properly, every alert warrants a response

Dwell Time



"The faster the data breach can be identified and contained, the lower the costs."

** 2018 Verizon Data Breach Incident Report*

"In this year's study, organizations were able to reduce the days to identify the data breach from an average of approximately 201 in 2016 to 191 days and the average days to contain the data breach from 70 to 66 days. We attribute these improvements to investments in such enabling security technologies as security analytics, SIEM, enterprise wide encryption and threat intelligence sharing platforms."

- Average breach size: 25,575 records
- Average cost per record: $150
- $150 x 25,575 = $3,836,250

* *2017 Cost of Data Breach Study*, Ponemon Institute, IBM Security

Examples

Example: Marriott

- Dwell Time >= 4 years
  - **July 2014:** Hackers penetrate Starwood systems
  - **September 23, 2016:** Marriott officially completes acquisition of Starwood
  - **September 7, 2018:** Accenture, a contractor managing the Starwood database for Marriott, becomes aware of the breach due to an alert regarding an unusual SQL query
  - **November 2018:** Investigators find that a hacker has been present since at least July 2014*
  - DB monitoring tool triggered alert on admin account querying for number of rows in a table
    - Not a normal query => anomaly

* https://resources.infosecinstitute.com/lessons-learned-the-marriott-breach/

- Yahoo: 3 billion
- Marriott: 383 million
- Adult Friend Finder: 412 million
- eBay: 145 million
- Equifax: 147 million
- Heartland: 134 million
- Target: 110 million
- TJX: 94 million
- Uber: 57 million users / 600k drivers
- JP Morgan Chase: 76 million

https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html

Best practices

Application

Traces
Events

Traces
Events

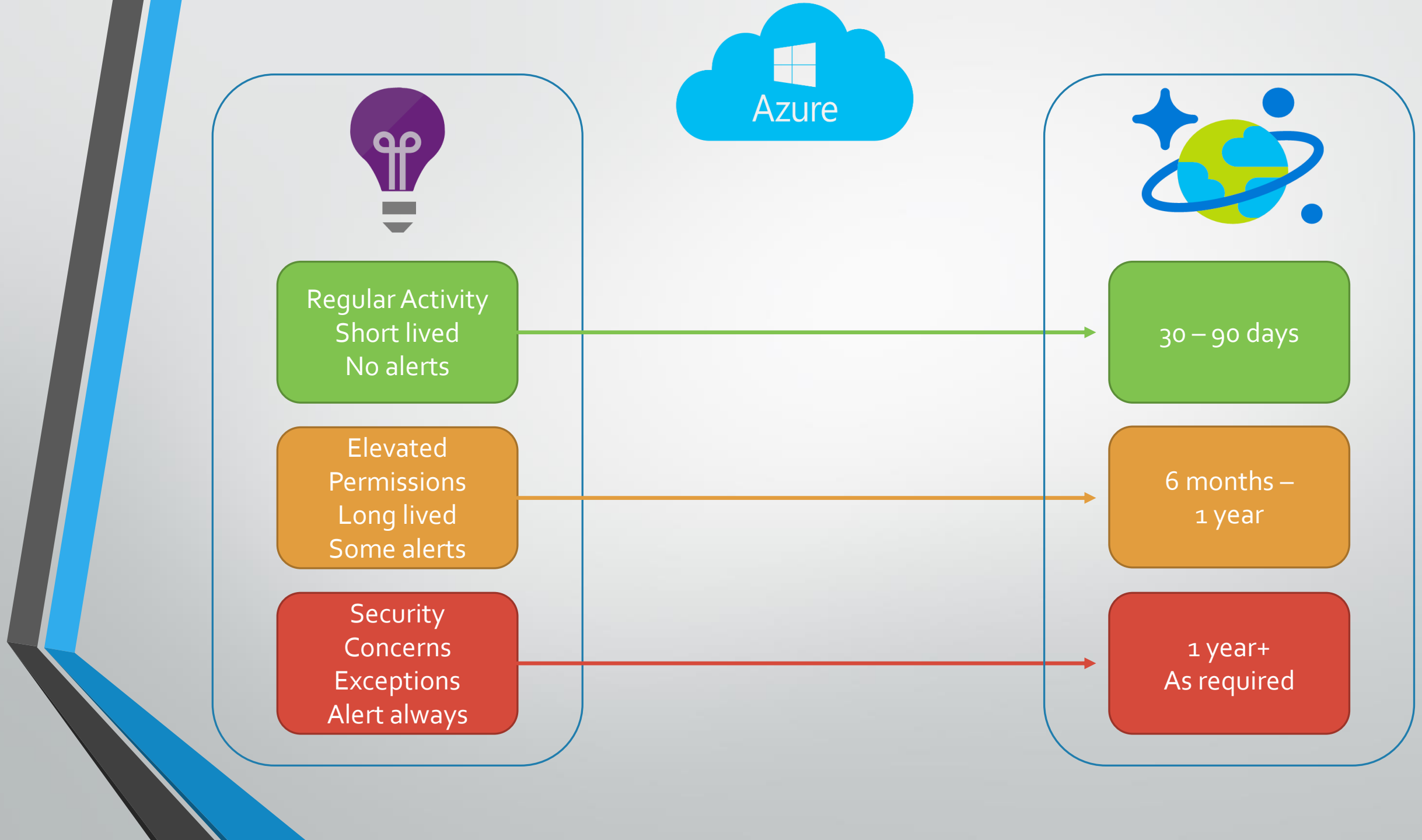
Events
Exceptions

Azure

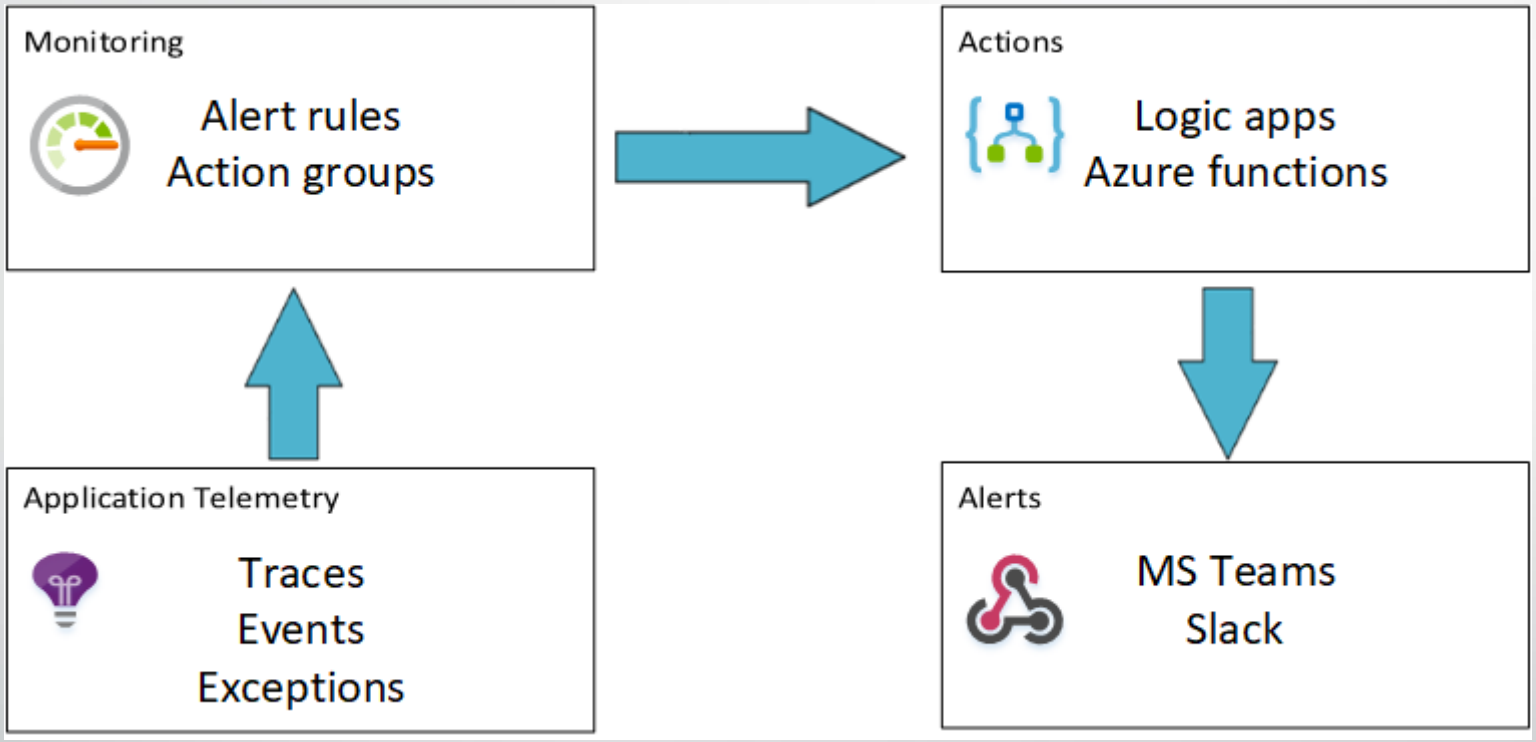
Regular Activity
Short lived
No alerts

Elevated Permissions
Long lived
Some alerts

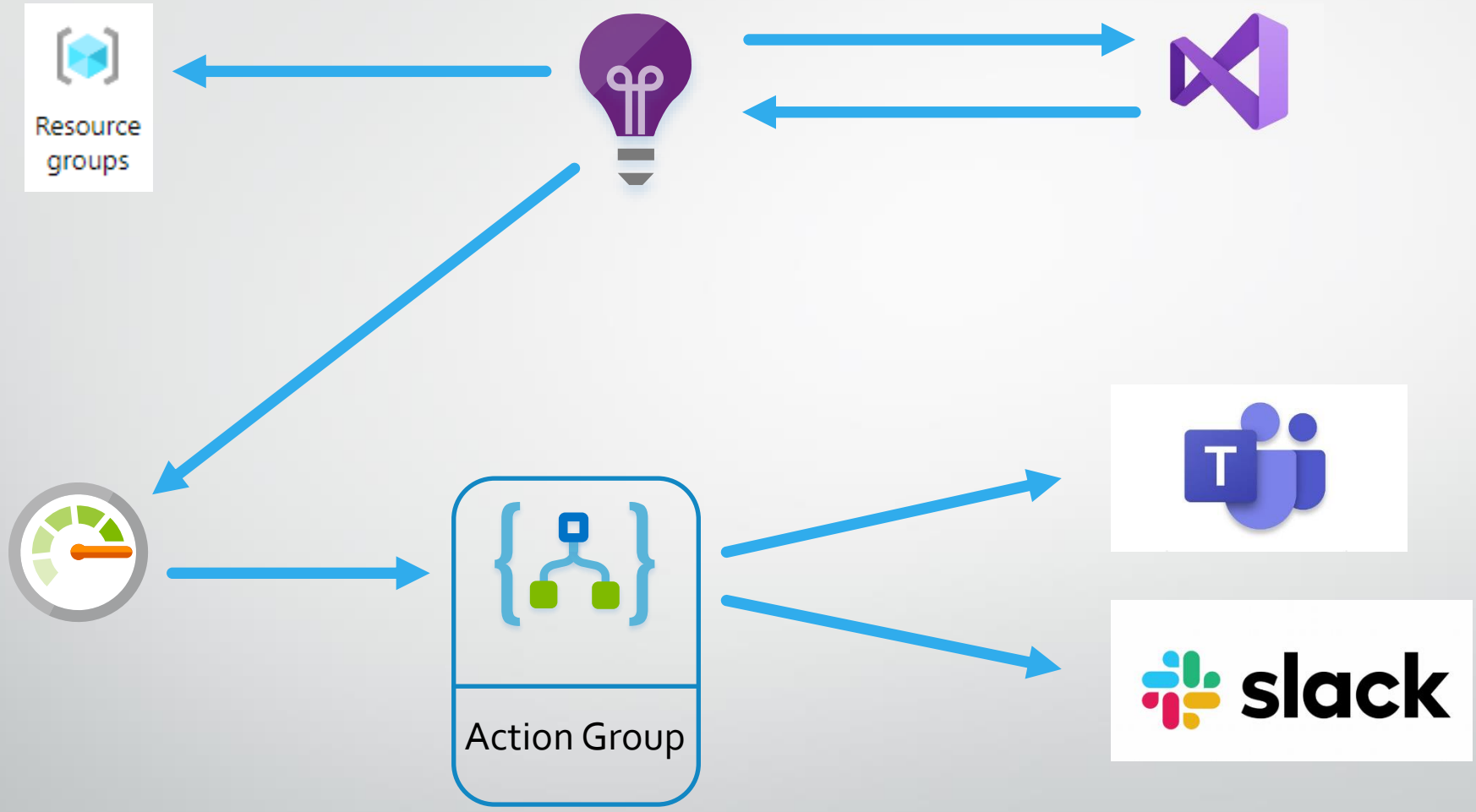
Security Concerns
Exceptions
Alert always


Azure
Regular Activity
Short lived
No alerts
Elevated
Permissions
Long lived
Some alerts
Security
Concerns
Exceptions
Alert always
30 – 90 days
6 months –
1 year
1 year+
As required

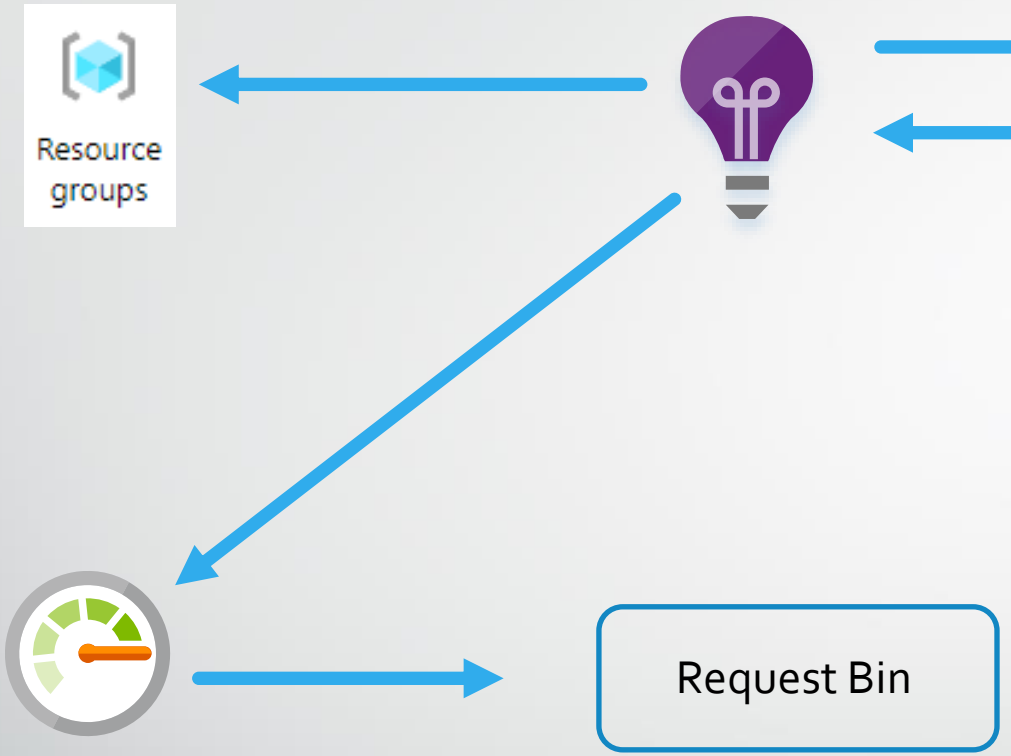
Azure Organization & Flow



Design Concept

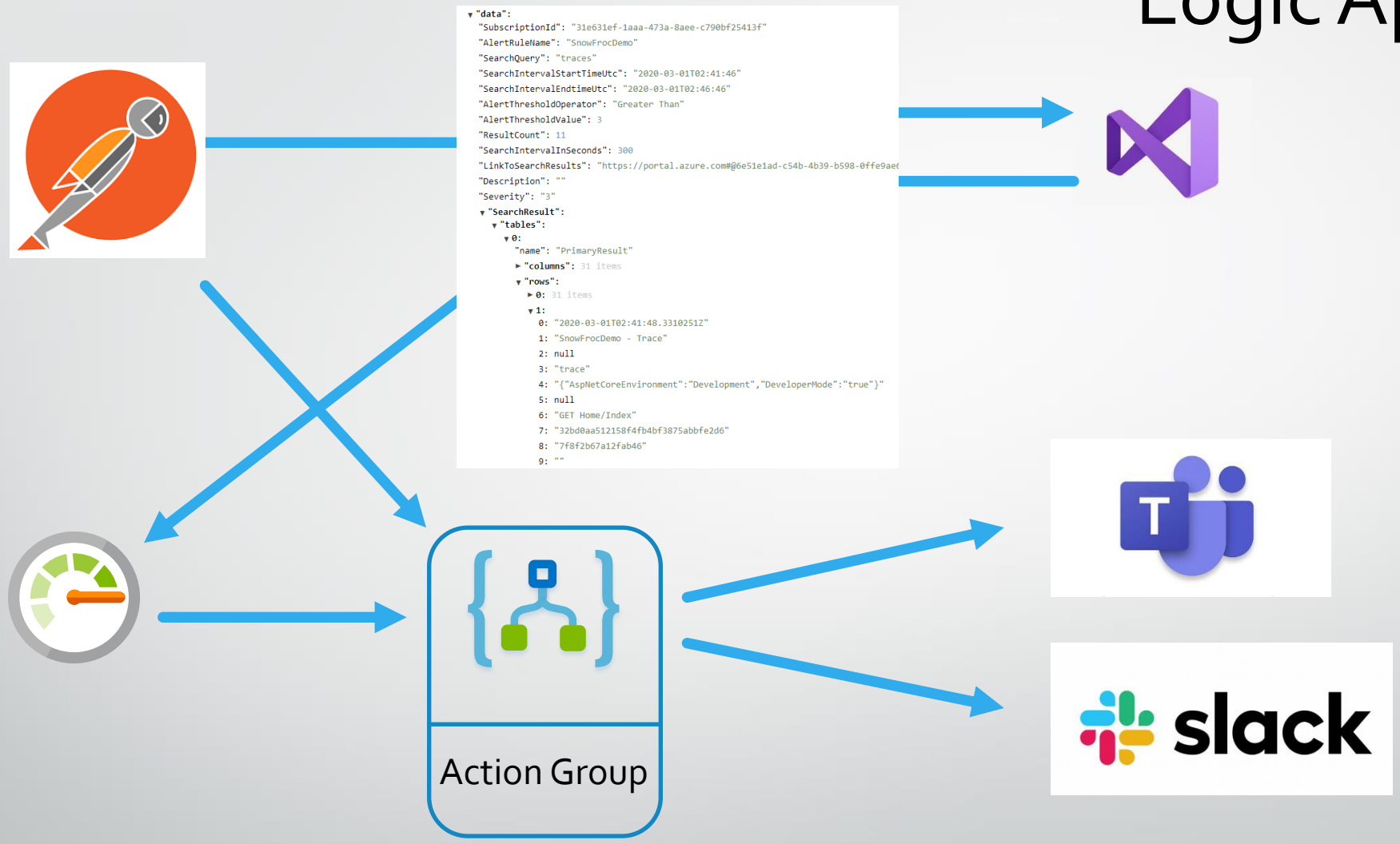


Early Dev – Request Bin




```
▼ "data":
  "SubscriptionId": "31e631ef-1aaa-473a-8aee-c790bf25413f"
  "AlertRuleName": "SnowFrocDemo"
  "SearchQuery": "traces"
  "SearchIntervalStartTimeUtc": "2020-03-01T02:41:46"
  "SearchIntervalEndtimeUtc": "2020-03-01T02:46:46"
  "AlertThresholdOperator": "Greater Than"
  "AlertThresholdValue": 3
  "ResultCount": 11
  "SearchIntervalInSeconds": 300
  "LinkToSearchResults": "https://portal.azure.com#@6e51e1ad-c54b-4b39-b598-0ffe9ae6
  "Description": ""
  "Severity": "3"
  ▼ "SearchResult":
    ▼ "tables":
      ▼ 0:
          "name": "PrimaryResult"
        ▶ "columns": 31 items
        ▼ "rows":
          ▶ 0: 31 items
          ▼ 1:
              0: "2020-03-01T02:41:48.3310251Z"
              1: "SnowFrocDemo - Trace"
              2: null
              3: "trace"
              4: "{"AspNetCoreEnvironment":"Development","DeveloperMode":"true"}"
              5: null
              6: "GET Home/Index"
              7: "32bd0aa512158f4fb4bf3875abbfe2d6"
              8: "7f8f2b67a12fab46"
              9: ""
```

Logic App Dev



Tools

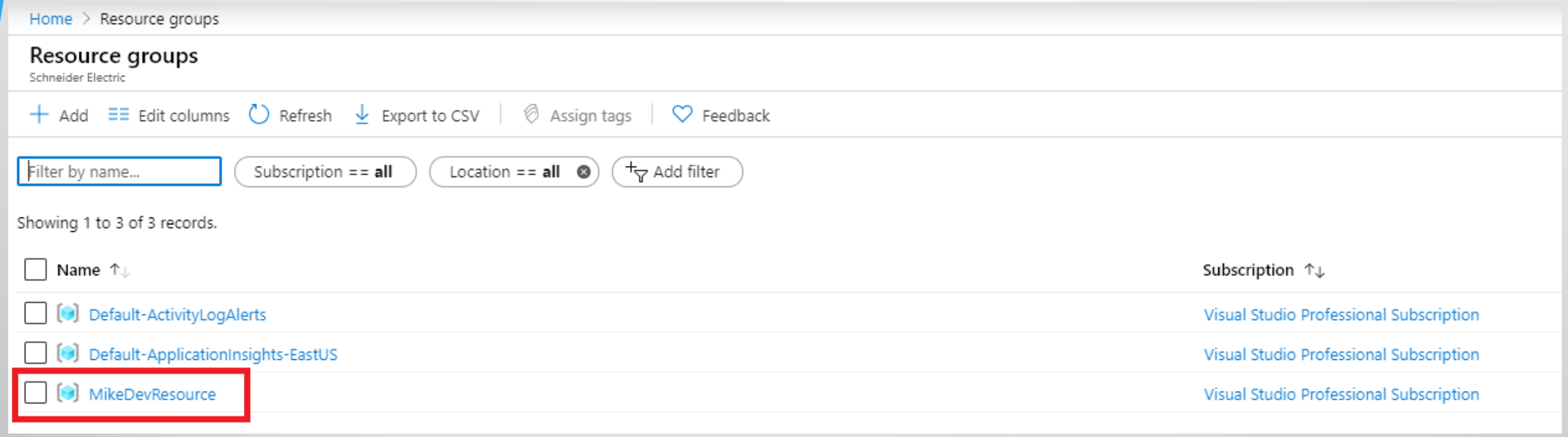
- Visual Studio 2019
- ASP.NET Core
- Azure Resource Group
- Azure Action Group
- Azure Logic App
- Request Bin

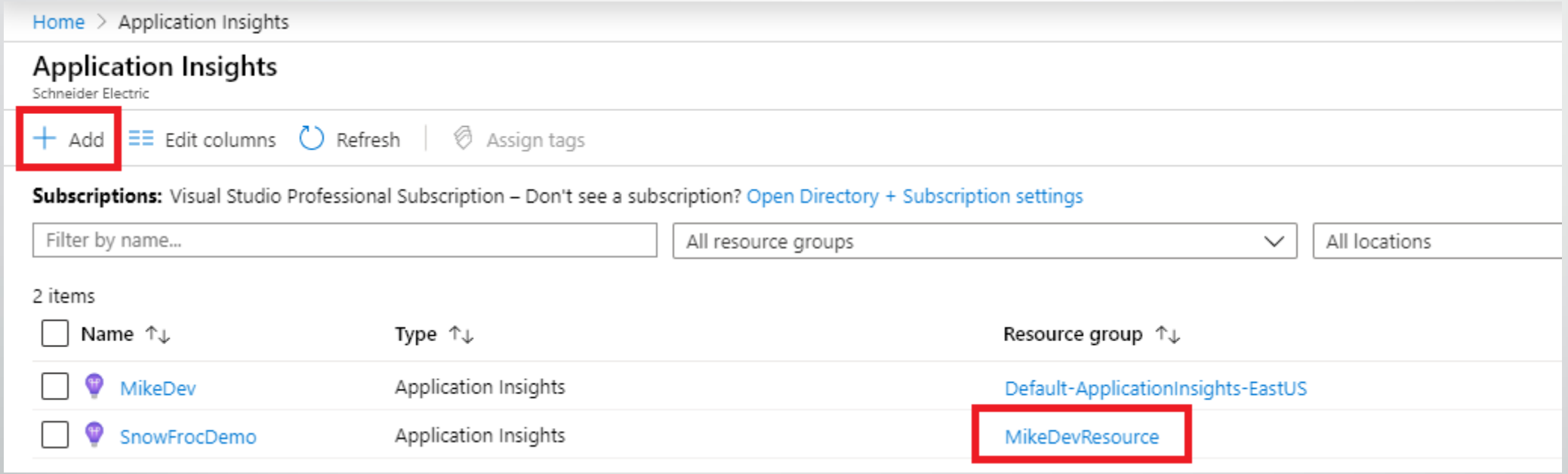
- Application Insights
- Azure Monitor
- Azure Alert
- MS Teams
- Slack
- Postman

- OWASP API Security Top 10
  - https://owasp.org/www-project-api-security/
- OWASP Top 10
  - https://owasp.org/www-project-top-ten/
- A10-Insufficient Logging & Monitoring
  - https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2017/Top_10-2017_A10-Insufficient_Logging%252526Monitoring
- 2017 Cost of Data Breach Study
  - https://www.ibm.com/downloads/cas/ZYKLN2E3
- Message Card playground
  - https://messagecardplayground.azurewebsites.net/

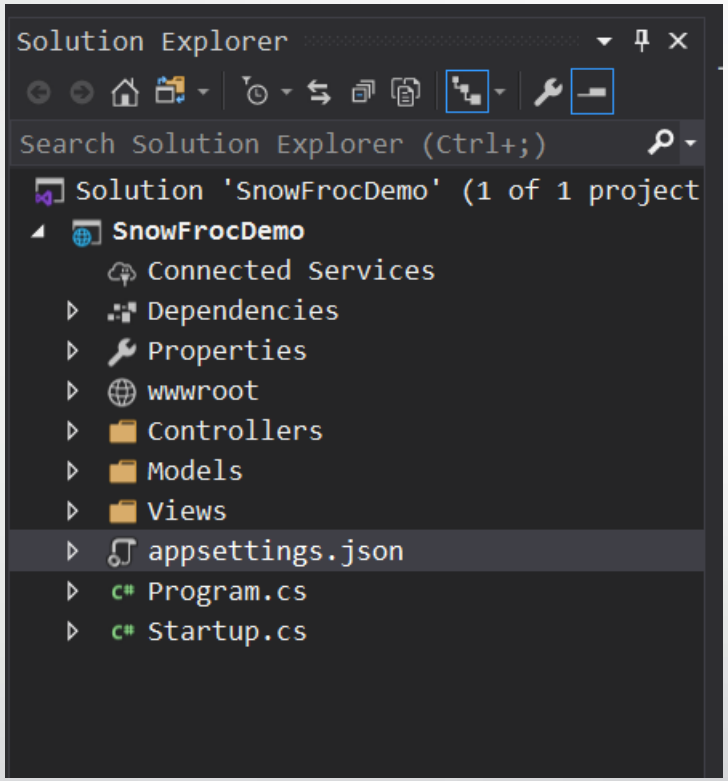
How to Guide

Create Resource Group

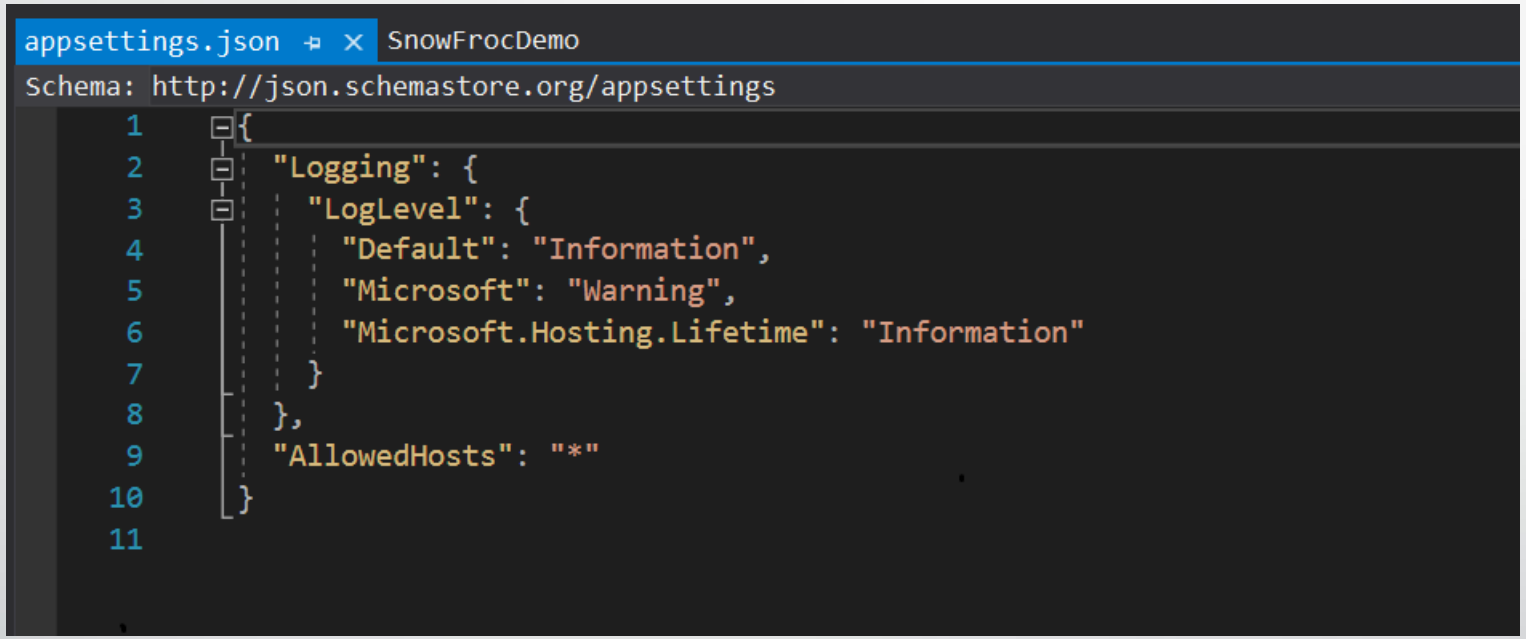


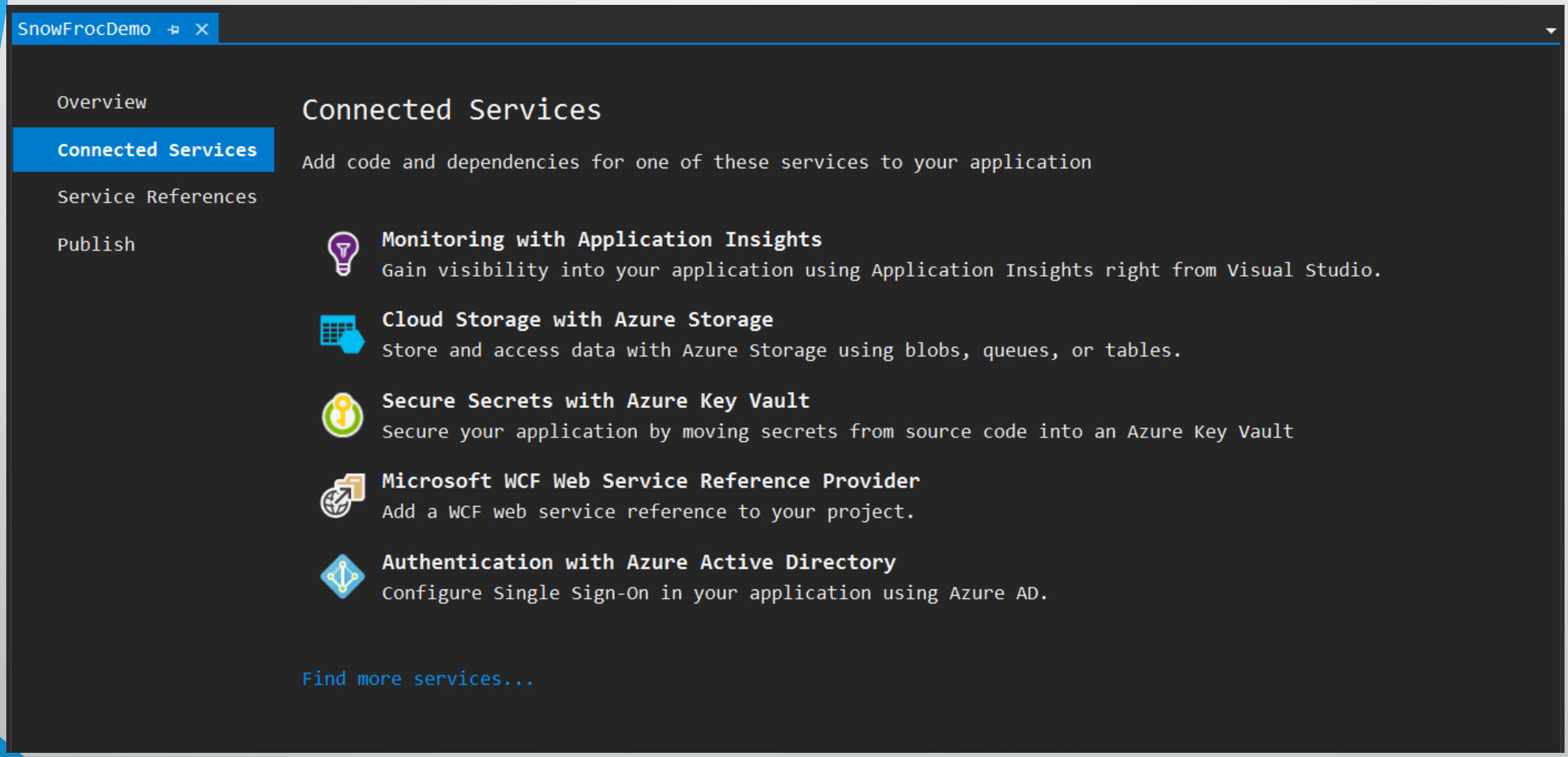

Home > Application Insights
Application Insights
Schneider Electric
Add
Edit columns
Refresh
Assign tags
Subscriptions: Visual Studio Professional Subscription – Don't see a subscription? Open Directory + Subscription settings
Filter by name...
All resource groups
All locations
2 items
Name
Type
Resource group
MikeDev
Application Insights
Default-ApplicationInsights-EastUS
SnowFrocDemo
Application Insights
MikeDevResource

Solution explorer before adding app insights



App settings before adding app insights




SnowFrocDemo
Overview
Connected Services
Service References
Publish
Connected Services
Add code and dependencies for one of these services to your application
Monitoring with Application Insights
Gain visibility into your application using Application Insights right from Visual Studio.
Cloud Storage with Azure Storage
Store and access data with Azure Storage using blobs, queues, or tables.
Secure Secrets with Azure Key Vault
Secure your application by moving secrets from source code into an Azure Key Vault
Microsoft WCF Web Service Reference Provider
Add a WCF web service reference to your project.
Authentication with Azure Active Directory
Configure Single Sign-On in your application using Azure AD.
Find more services...

Application…Configuration | SnowFrocDemo

Application Insights

Gain insights through telemetry, analytics and smart

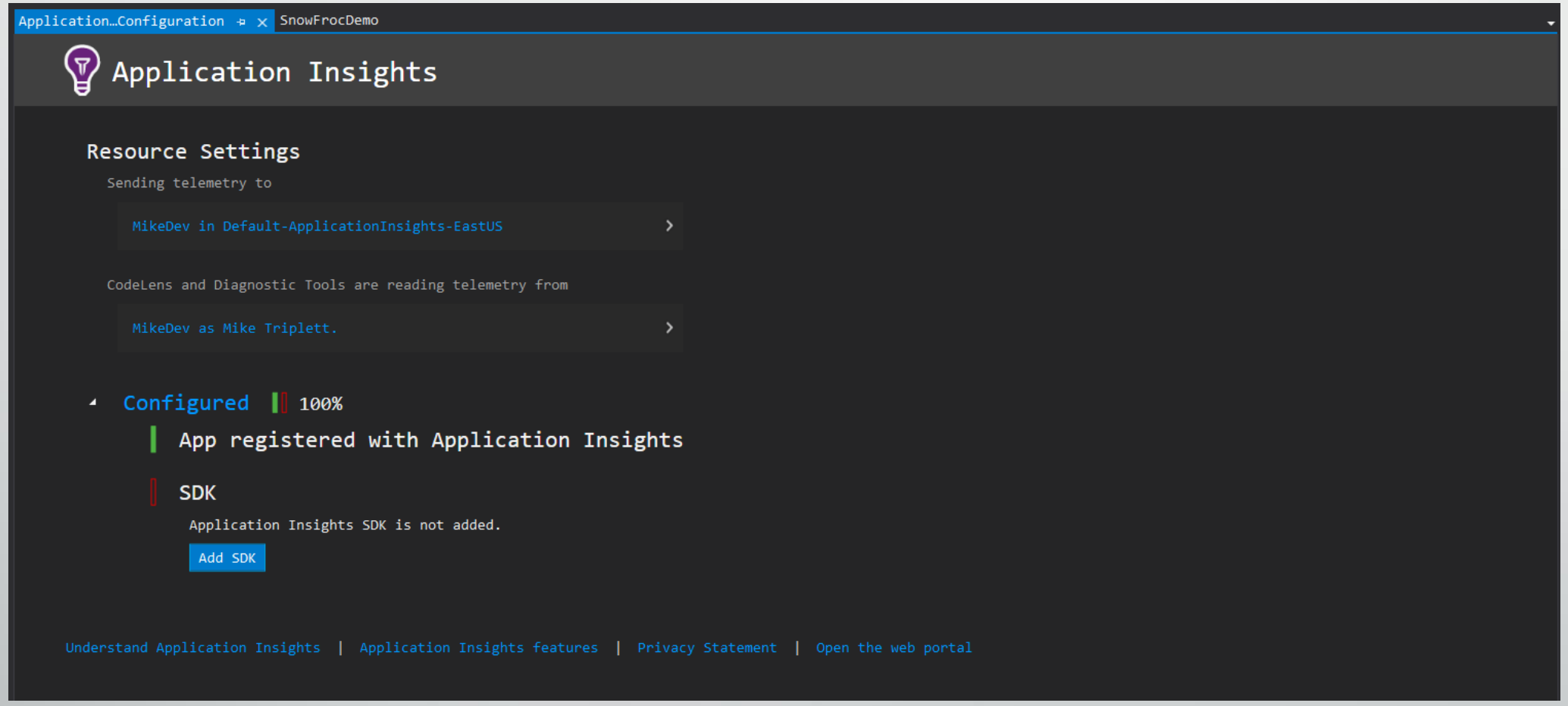
Detect
and diagnose exceptions and application performance issues

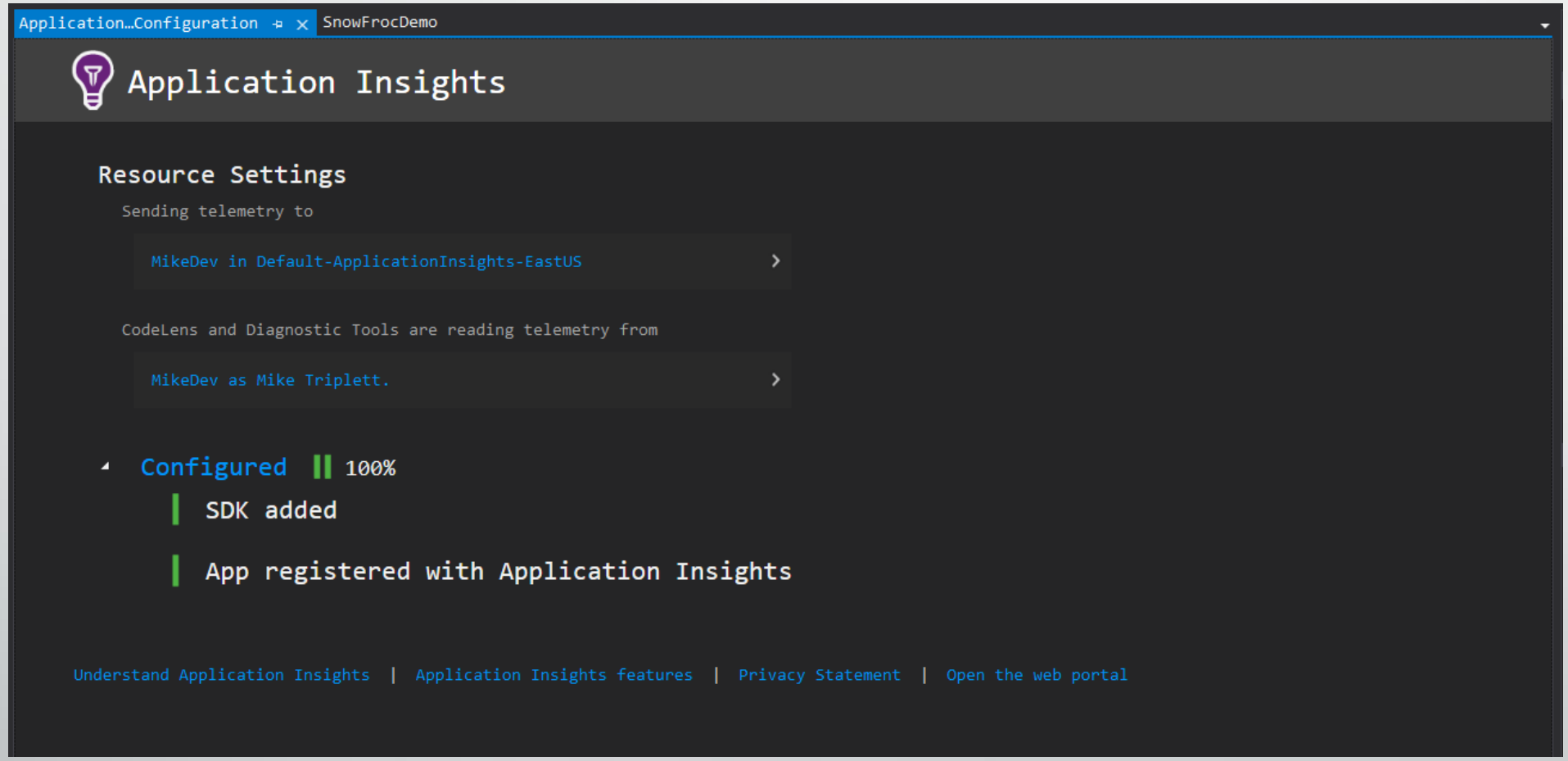
Monitor
websites on Azure, hosted containers, on-premises and with other cloud providers

Integrate
with your DevOps pipeline using Visual Studio, VSTS, GitHub, and web hooks

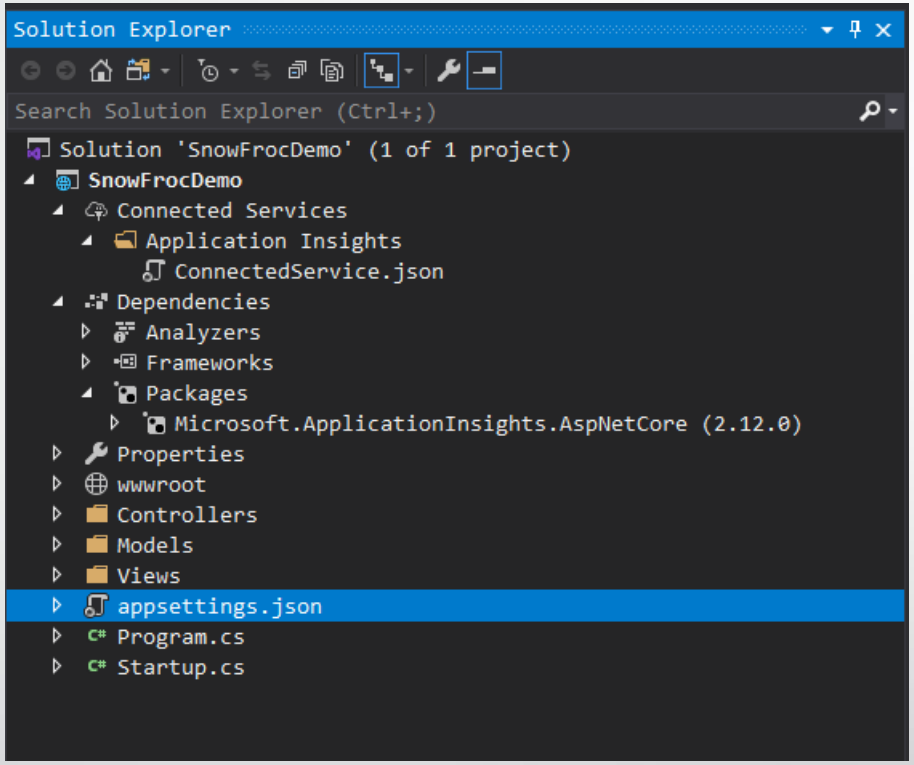
Get Started

Understand Application Insights | Application Insights features | Privacy Statement

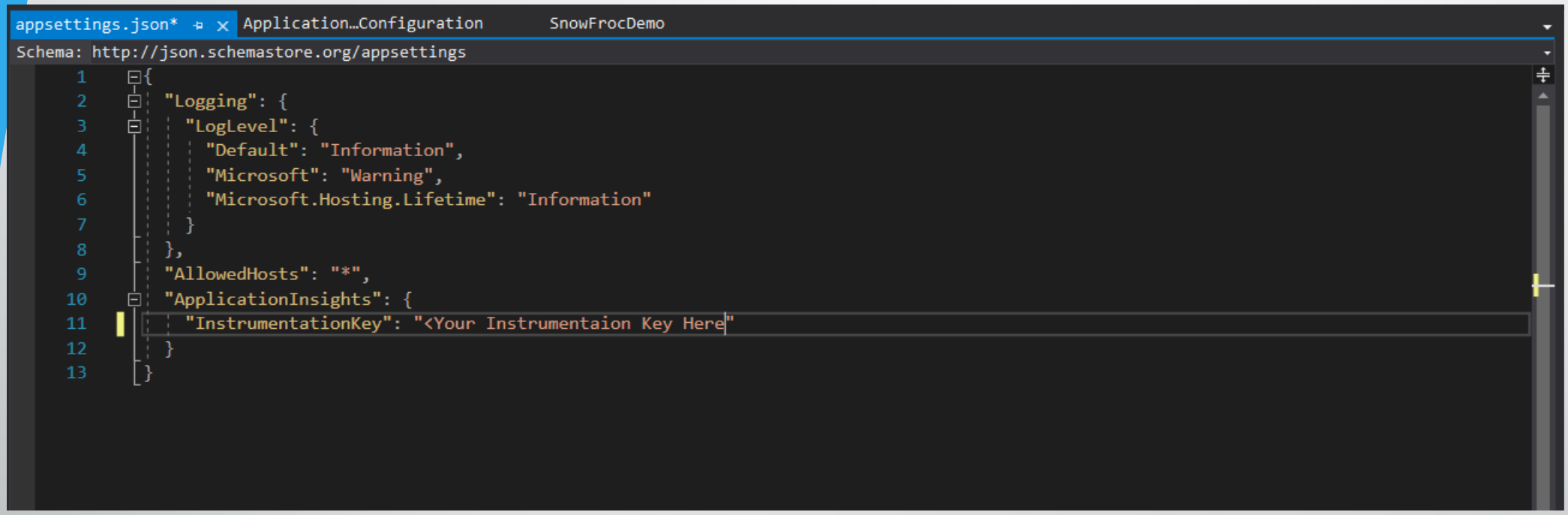

Application…Configuration
SnowFrocDemo
Application Insights
Resource Settings
Sending telemetry to
MikeDev in Default-ApplicationInsights-EastUS
CodeLens and Diagnostic Tools are reading telemetry from
MikeDev as Mike Triplett.
Configured
100%
App registered with Application Insights
SDK
Application Insights SDK is not added.
Add SDK
Understand Application Insights | Application Insights features | Privacy Statement | Open the web portal


Application…Configuration
SnowFrocDemo
Application Insights
Resource Settings
Sending telemetry to
MikeDev in Default-ApplicationInsights-EastUS
CodeLens and Diagnostic Tools are reading telemetry from
MikeDev as Mike Triplett.
Configured
100%
SDK added
App registered with Application Insights
Understand Application Insights | Application Insights features | Privacy Statement | Open the web portal

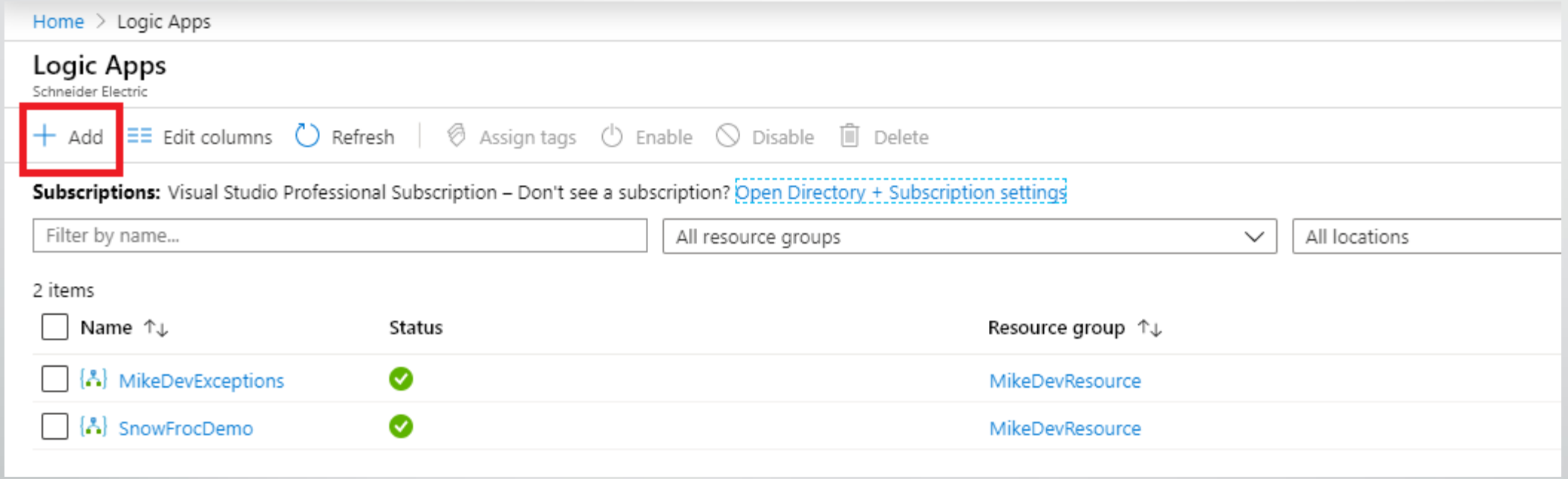
Solution explorer after adding app insights



App settings after adding app insights



Add logic app



Home > Monitor - Overview

Monitor - Overview

Microsoft

Documentation

Search (Ctrl+/)

- Overview
- Activity log
- Alerts
- Metrics
- Logs
- Service Health
- Workbooks

Insights

- Applications
- Virtual Machines (preview)
- Storage Accounts (preview)
- Containers
- Networks (preview)
- Cosmos DB (preview)
- More

Settings

- Diagnostics settings
- Autoscale

Support + Troubleshooting

- Usage and estimated costs
- Advisor recommendations
- New support request

What's new | Get started | Tutorials & Demos

Monitor your applications and infrastructure

Get full stack visibility, find and fix problems, optimize your performance, and understand customer behavior all in one place. Learn more

Monitor & Visualize Metrics

Metrics are numerical values available from Azure Resources helping you understand the health, operation & performance of your systems.

Explore Metr...

Query & Analyze Logs

Logs are activity logs, diagnostic logs and telemetry from monitoring solutions; Analytics queries help with troubleshooting & visualizations.

Search Logs

Setup Alert & Actions

Alerts notify you of critical conditions and potentially take corrective automated actions based on triggers from metrics or logs.

Create Alert

Quick Starts

Learn how to collect data from...

- Azure VMs
- Linux Computers
- Windows Computers
- Azure Kubernetes
- Docker & Windows Containers

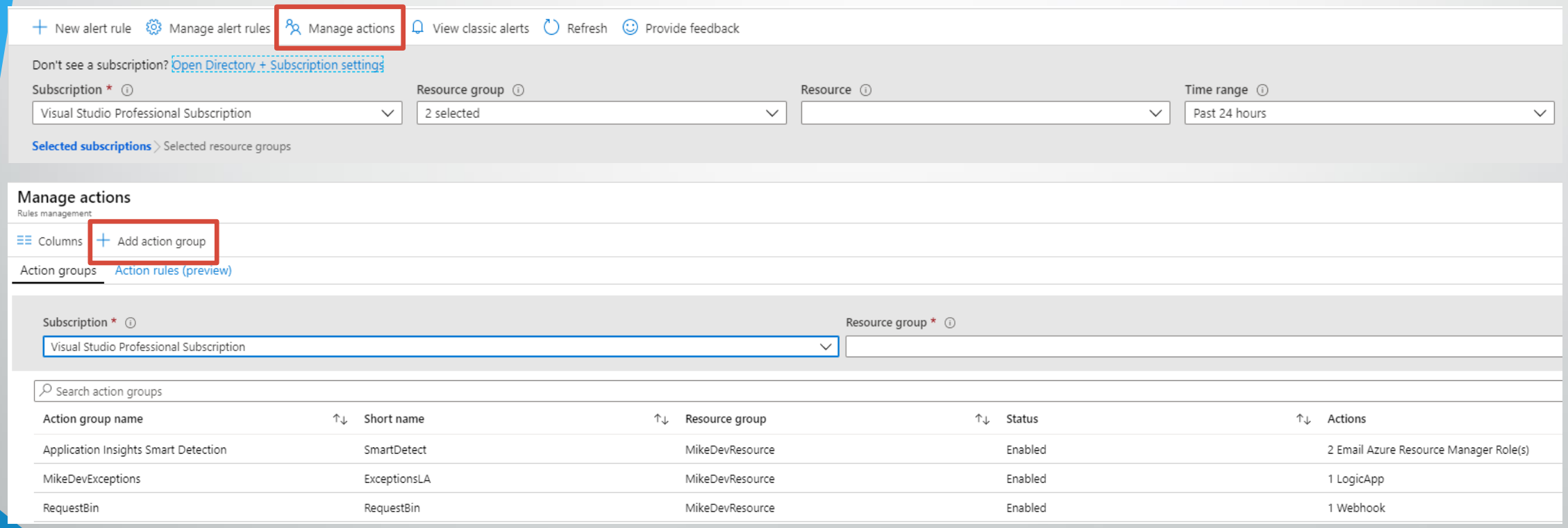
Learn how to monitor...

- Azure Web Apps
- Azure Cloud Services
- Docker Apps
- Azure Functions
- Service Fabric Apps

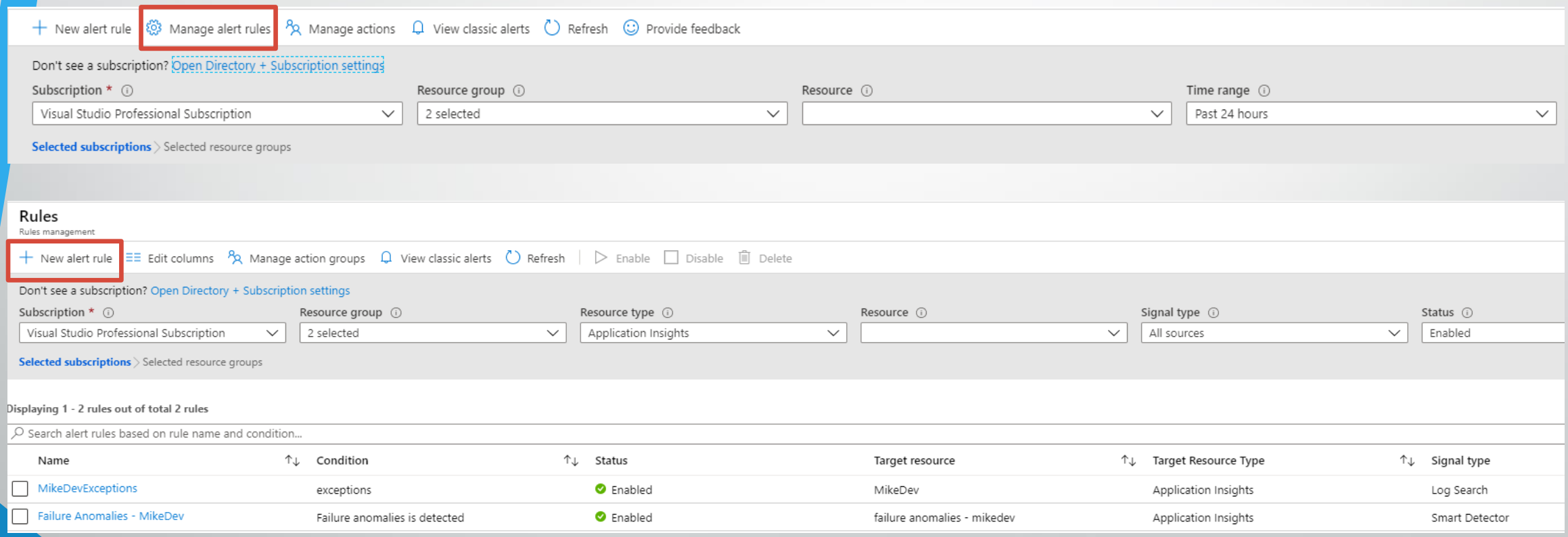
Learn how to onboard...

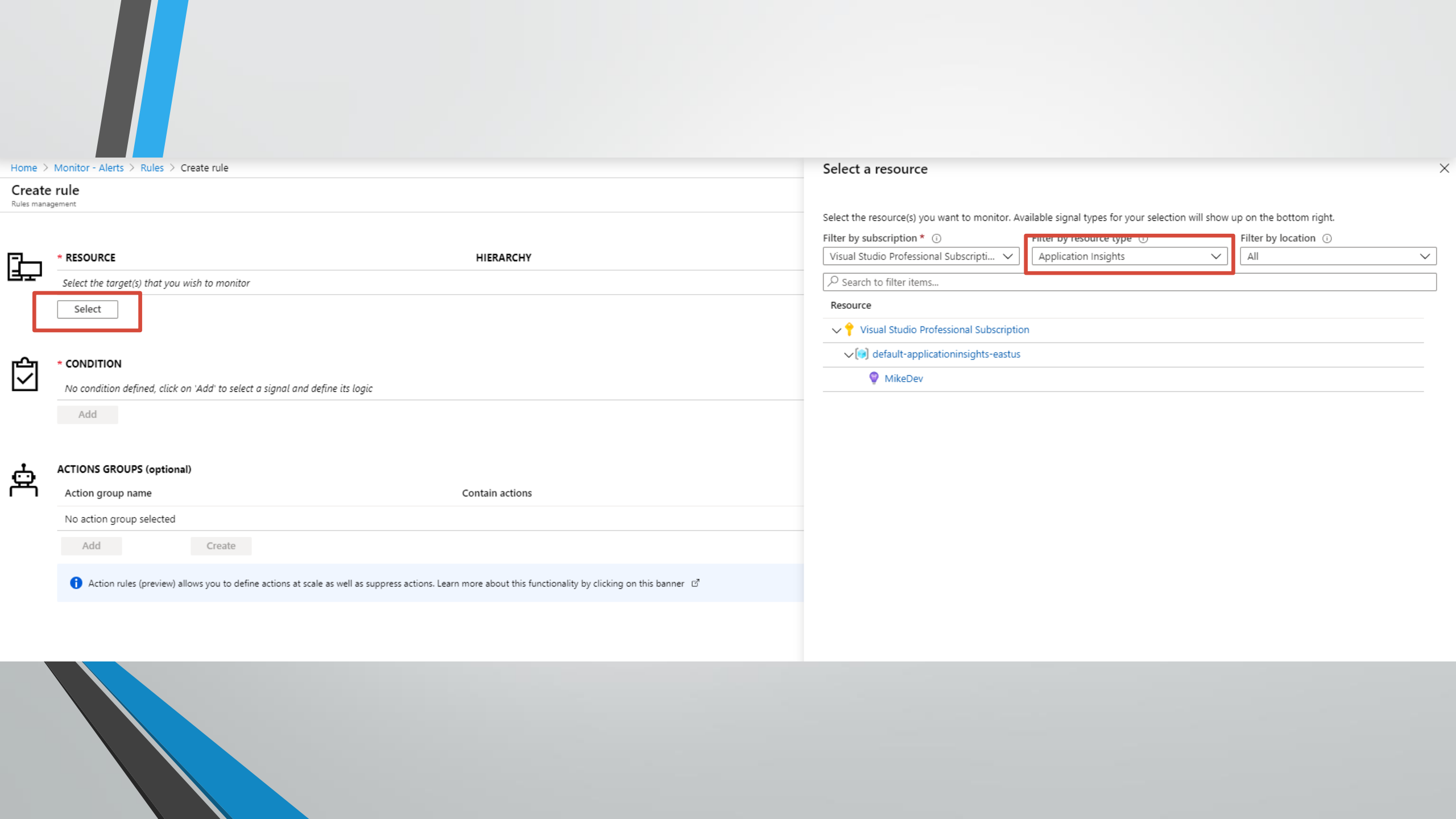
- ASP.NET Apps from Visual Studio
- NodeJS Apps
- Java Apps from Eclipse
- Mobile Apps from VS App Center

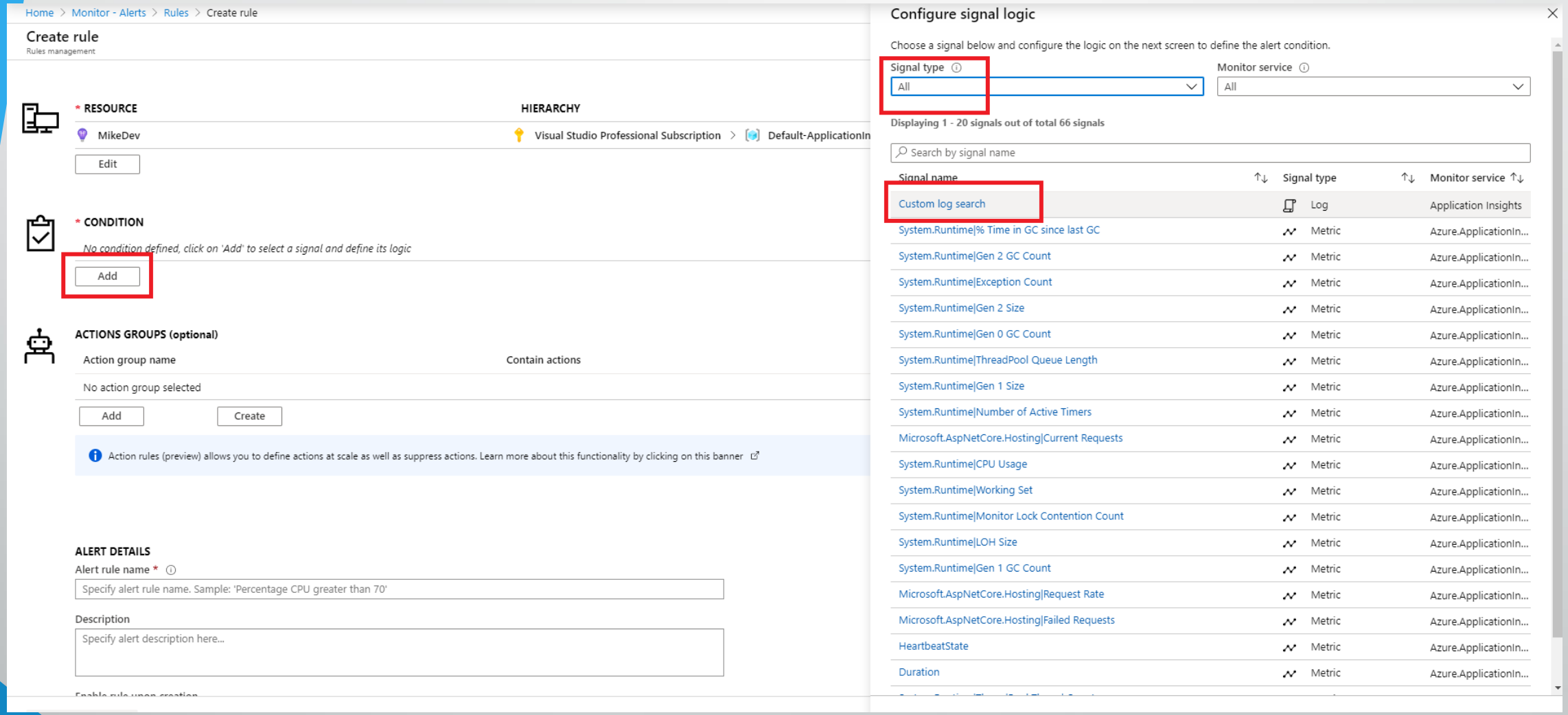
Add logic app to action group



Create new alert rule



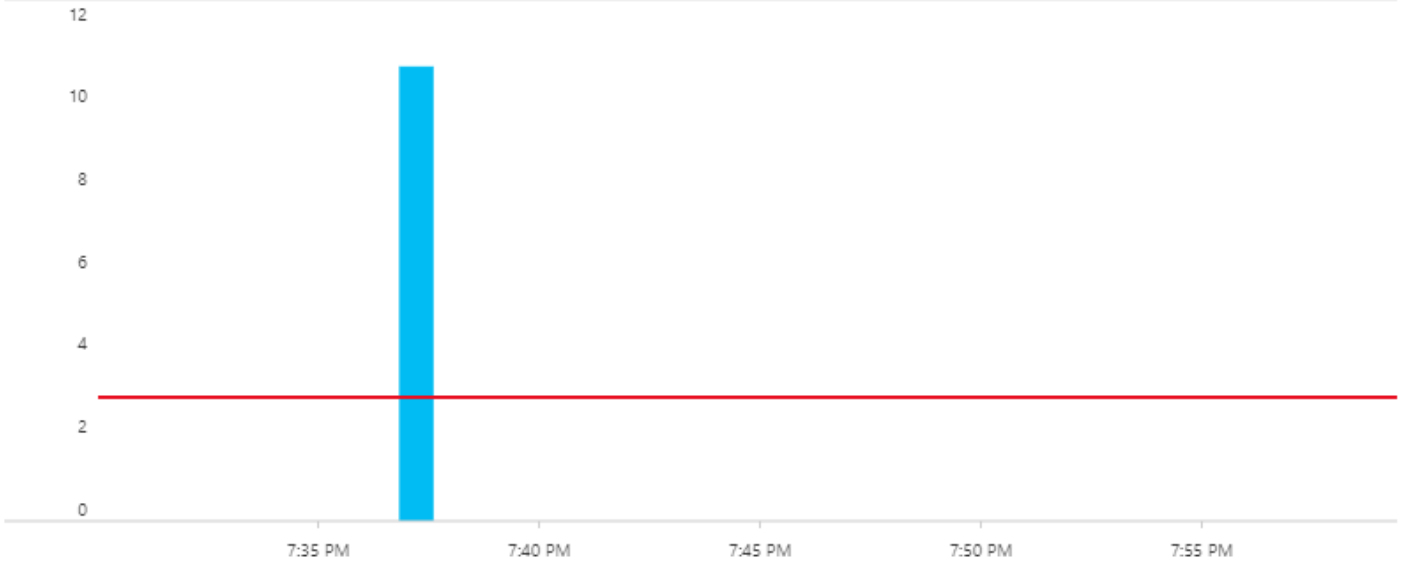

Home > Monitor - Alerts > Rules > Create rule
Create rule
Rules management
* RESOURCE
HIERARCHY
Select the target(s) that you wish to monitor
Select
* CONDITION
No condition defined, click on 'Add' to select a signal and define its logic
Add
ACTIONS GROUPS (optional)
Action group name
Contain actions
No action group selected
Add
Create
Action rules (preview) allows you to define actions at scale as well as suppress actions. Learn more about this functionality by clicking on this banner
Select a resource
Select the resource(s) you want to monitor. Available signal types for your selection will show up on the bottom right.
Filter by subscription *
Visual Studio Professional Subscripti...
Filter by resource type
Application Insights
Filter by location
All
Search to filter items...
Resource
Visual Studio Professional Subscription
default-applicationinsights-eastus
MikeDev


Home > Monitor - Alerts > Rules > Create rule
Create rule
Rules management
* RESOURCE
HIERARCHY
MikeDev
Visual Studio Professional Subscription > Default-ApplicationIn
Edit
* CONDITION
No condition defined, click on 'Add' to select a signal and define its logic
Add
ACTIONS GROUPS (optional)
Action group name
Contain actions
No action group selected
Add
Create
Action rules (preview) allows you to define actions at scale as well as suppress actions. Learn more about this functionality by clicking on this banner
ALERT DETAILS
Alert rule name *
Specify alert rule name. Sample: 'Percentage CPU greater than 70'
Description
Specify alert description here...
Configure signal logic
Choose a signal below and configure the logic on the next screen to define the alert condition.
Signal type
All
Monitor service
All
Displaying 1 - 20 signals out of total 66 signals
Search by signal name
Signal name
Signal type
Monitor service
Custom log search
Log
Application Insights
System.Runtime|% Time in GC since last GC
Metric
Azure.ApplicationIn...
System.Runtime|Gen 2 GC Count
Metric
Azure.ApplicationIn...
System.Runtime|Exception Count
Metric
Azure.ApplicationIn...
System.Runtime|Gen 2 Size
Metric
Azure.ApplicationIn...
System.Runtime|Gen 0 GC Count
Metric
Azure.ApplicationIn...
System.Runtime|ThreadPool Queue Length
Metric
Azure.ApplicationIn...
System.Runtime|Gen 1 Size
Metric
Azure.ApplicationIn...
System.Runtime|Number of Active Timers
Metric
Azure.ApplicationIn...
Microsoft.AspNetCore.Hosting|Current Requests
Metric
Azure.ApplicationIn...
System.Runtime|CPU Usage
Metric
Azure.ApplicationIn...
System.Runtime|Working Set
Metric
Azure.ApplicationIn...
System.Runtime|Monitor Lock Contention Count
Metric
Azure.ApplicationIn...
System.Runtime|LOH Size
Metric
Azure.ApplicationIn...
System.Runtime|Gen 1 GC Count
Metric
Azure.ApplicationIn...
Microsoft.AspNetCore.Hosting|Request Rate
Metric
Azure.ApplicationIn...
Microsoft.AspNetCore.Hosting|Failed Requests
Metric
Azure.ApplicationIn...
HeartbeatState
Metric
Azure.ApplicationIn...
Duration
Metric
Azure.ApplicationIn...

Configure signal logic

6 evaluations done in last 30 minutes

Time range: last 30 minutes



Search query *

customEvents

View result of query in Azure Monitor - Logs

Query to be executed : *customEvents ↵| count*
For time window : *2/17/2020, 7:47:00 PM - 2/17/2020, 7:52:00 PM*

It may take in the range of 9 minutes, to have the logs available for provided query Learn more

Alert logic

Based on	Operator	Threshold value *
Number of results	Greater than	3

Trigger the action group containing the logic app