

## Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



base: 12087e4 ▾



compare: cef74f7 ▾

-o 3 commits

 234 files changed

 1 contributor

-o Commits on Feb 15, 2022

v3.0 release (aka FY22 refresh) (#352)



DariuszPorowski committed 20 days ago 

-o Commits on Feb 17, 2022

after tests fixes (#355) ...



DariuszPorowski committed 18 days ago 

-o Commits on Feb 18, 2022

metadata fix



DariuszPorowski committed 17 days ago 

 Showing 234 changed files with 5,657 additions and 1,957 deletions.

Split

Unified

▼ 26  .github/workflows/ISSUE\_TEMPLATE/bug\_report.md 

... @@ -0,0 +1,26 @@

```
1 + ---  
2 + name: Bug report  
3 + about: Create a report to help us improve  
4 + title: ""  
5 + labels: "bug"  
6 + assignees: ""
```

```
7 + ---
8 +
9 + **Describe the bug**
10 + A clear and concise description of what
the bug is.
11 +
12 + **To Reproduce**
13 + Steps to reproduce the behavior:
14 +
15 + 1. '...'
16 + 2. '...'
17 + 3. '...'
18 +
19 + **Expected behavior**
20 + A clear and concise description of what
you expected to happen.
21 +
22 + **Screenshots**
23 + If applicable, add screenshots to help
explain your problem.
24 +
25 + **Additional context**
26 + Add any other context about the problem
here.
```

✓ 19 ████ .github/workflows/ISSUE\_TEMPLATE/feature\_request.md 

... @@ -0,0 +1,19 @@

```
1 + ---
2 + name: Feature or enhancement request
3 + about: Suggest an idea for this project
4 + title: ""
5 + labels: "enhancement"
6 + assignees: ""
7 +
8 +
9 + **Is your feature request related to a
problem? Please describe.**
10 + A clear and concise description of what
the problem is. Ex. I'm always frustrated
when [...]
11 +
12 + **Describe the solution you'd like**
13 + A clear and concise description of what
you want to happen.
14 +
```

```
15 + **Describe alternatives you've  
16 considered**  
17 + A clear and concise description of any  
18 alternative solutions or features you've  
19 considered.  
+  
+ **Additional context**  
+ Add any other context or screenshots  
about the feature request here.
```

▽ 123 █████ .github/workflows/ci-pipeline.yaml □

### Load diff

This file was deleted.

▽ 140 █████ .github/workflows/ci-workflow.yaml □

... @@ -0,0 +1,140 @@

```
1 + name: devops-ci-workflow  
2 +  
3 + # Runs on open PR to main, lints md and  
4 + json, creates copy of guides and pushes  
5 + to storage.  
6 +  
7 + on:  
8 +   pull_request:  
9 +     branches: [main]  
10 + env:  
11 +   ARCHIVE_FOLDER: openhack-  
12 + assets/archives/stagingbuilds/microsoft-  
13 + open-hack-devops  
14 +  
15 + jobs:  
16 +   build:  
17 +     runs-on: ubuntu-latest  
18 +  
19 +     # Steps represent a sequence of  
20 +     # tasks that will be executed as part of  
21 +     # the job  
22 +     steps:  
23 +       # Checks-out your repository under
```

```
$GITHUB_WORKSPACE, so your job can
access it
18 +     - uses: actions/checkout@v2
19 +
20 +     - name: "Setup Node"
21 +       uses: actions/setup-node@v2
22 +       with:
23 +         node-version: 12.x
24 +
25 +       # Lint your open-hack guides
26 +       - name: "Linting OpenHack Content"
27 +         uses: nosborn/github-action-
28 +           markdown-cli@v3
29 +           with:
30 +             # configuration file (JSON or
31 +             YAML)
32 +             config_file:
33 +               .markdownlint.json
34 +             # files, directories, or globs
35 +             files: ./portal
36 +
37 +       # Lint any additional guides you
38 +       have (any *.md files). Remove if not
39 +       needed.
40 +       - name: "Linting Resources"
41 +         uses: nosborn/github-action-
42 +           markdown-cli@v3
43 +           with:
44 +             # configuration file (JSON or
45 +             YAML)
46 +             config_file:
47 +               .markdownlint.json
48 +             # files, directories, or globs
49 +             files: ./resources
50 +             # ignore_files:
51 +               "resources/SupportingFiles/react-rating-
52 +                 ui-ts/*"
53 +
54 +       # Lint your JSON files [mostly for
55 +       content def]
56 +       - name: "Install dependencies"
57 +         run: npm install jsonlint -g
58 +
59 +       # Make sure to lint any language-
60 +       specific versions you've created(remove
61 +       unnecessary/non-existent paths)
```

```
49 +      - name: "Execute json linter lab-
  content-definitions"
50 +      run: |
51 +          jsonlint ./portal/en/lab-
  content-definition.json
52 +
53 +      # Install PanDoc
54 +      - name: Install Pandoc
55 +      run: sudo apt-get install pandoc
56 +
57 +      # Setup the output archive
  environment [You will need to map all of
  your output folders, remove any that
  don't make sense]
58 +      # Additionally: Don't forget to
  set chmod -R 755 on the new folders
59 +      - name: Setup Build Environment
60 +      run: |
61 +          mkdir output # create output
  dir
62 +          sudo chmod -R 755 output
63 +          mkdir output/portal
64 +          sudo chmod -R 755
  output/portal
65 +          mkdir output/portal/en
66 +          sudo chmod -R 755
  output/portal/en
67 +          mkdir output/portal/en/images
68 +          sudo chmod -R 755
  output/portal/en/images
69 +          mkdir output/resources
70 +          sudo chmod -R 755
  output/resources
71 +
72 +      # Create HTML files from your
  markdown. Map all language-specific
  versions, remove any that aren't
  existent.
73 +      - name: "Create EN files"
74 +      run: |
75 +          FILE_LIST_EN=$(printf '%s;''
  ./portal/en/*.md)
76 +          #echo ${FILE_LIST_EN}
77 +          IFS=';' read -ra ENFILES <<<
  ${FILE_LIST_EN}
78 +          for f in ${ENFILES[@]};
```

```

79      +           do
80      +               HTML_FULL_PATH=`basename
81      +                   ${f}`
82      +
83      +               NEW_FILE=${HTML_FULL_PATH%md/html}
84      +               #echo ${NEW_FILE}
85      +               pandoc --from
86      +                   markdown+raw_html --to html5
87      +                   ./portal/includes.yaml --metadata
88      +                   pagetitle="OPEN HACK GUIDE" --no-
89      +                   highlight --
90      +                   output=output/portal/en/${NEW_FILE} ${f}
91      +               cp ${f} output/portal/en/
92      +               done
93      +
94      +           #Copy Content Definitions to
95      +           #output artifact (again, map all specific
96      +           #languages, remove any non-existent)
97      +           - name: "Copy Lab content
98      +               Definitions"
99      +               run: |
100      +                   cp ./portal/en/lab-content-
101      +                   definition.json output/portal/en
102      +
103      +           #Copy images to output artifact
104      +           #(again, map all specific languages,
105      +           #remove any non-existent)
106      +           - name: "Copy Images"
107      +               run: |
108      +                   cp ./portal/en/images/*
109      +                   output/portal/en/images
110      +
111      +           #Copy resources (again, map any
112      +           #additional folders you want to archive,
113      +           #remove if not necessary). This/these
114      +           #folder(s) is/are never actually
115      +           #deployed, just archived.
116      +           - name: "Copy Resources"
117      +               run: |
118      +                   cp -r ./resources/*
119      +                   output/resources
120      +
121      +           #Create Archive
122      +           - name: "create archive"
123      +               uses: actions/upload-

```

```
        artifact@main
106 +      with:
107 +        name: output
108 +        path: output
109 +
110 +      - name: Get current time
111 +        uses: srfrnk/current-time@master
112 +        id: current-time
113 +        with:
114 +          format: YYYYMMDD
115 +
116 +      # Archive Deploy to storage [this
117 +      # is simply for backup and is placed into
118 +      # non-production storage]
119 +      - name: "Deploy Assets staging
120 +        builds archive"
121 +        env:
122 +          F_TIME: "${{ steps.current-
123 +          time.outputs.formattedTime }}"
124 +        run: |
125 +          STORAGEARCHIVE="https://${{{
126 +            secrets.STORAGE_ACCOUNT_NAME
127 +          }}.blob.core.windows.net/${{
128 +            env.ARCHIVE_FOLDER }}/$F_TIME${{
129 +            secrets.STORAGE_ACCOUNT_SAS_TOKEN }}"
130 +
131 +          #Download and install azcopy
132 +          pwd
133 +          wget -O azcopy.tar.gz
134 +          https://aka.ms/downloadazcopy-v10-linux
135 +          tar -xf azcopy.tar.gz
136 +          rm azcopy.tar.gz
137 +
138 +          #copy files
139 +          azcopy copy output
140 +            $STORAGEARCHIVE --recursive
141 +
142 +          echo "Content copied to Azure
143 +          blob storage"
144 +          echo $STORAGEARCHIVE
145 +
146 +          # Delete Archive [the archive is
147 +          never stored at GitHub, saving compute
148 +          space at GitHub]
149 +
150 +          # Note: If you want to test your
151 +          output, temporarily remove this step and
```

```
review the output at GitHub [make sure  
to put the step back when you are ready  
to release]  
137 +     - name: Delete Output  
138 +       uses: geekyeggo/delete-  
139 +         artifact@v1  
140 +           with:  
140 +             name: output
```

▽ 18 ████ .github/workflows/pull\_request\_template.md □

... @@ -0,0 +1,18 @@

```
1 + # Pull Request Template  
2 +  
3 + ## What are you trying to address  
4 +  
5 + - Describe the current behavior that you  
6 +   are modifying, and link to issue number  
7 + - If you don't have an issue, browse  
8 +   through existing issues to see if this is  
9 +   already tracked as an issue, to assign  
10 +    yourself to the issue and also verify  
11 +    that no one else is already working on  
12 +    the issue.  
13 +  
14 + ## Description of new changes  
15 +  
16 + - Write a detailed description of all  
17 +   changes - and if appropriate, why they  
18 +   are needed  
19 +  
20 + ## For all pull requests  
21 +  
22 + - [ ] Link to the issue you are solving  
23 +   (so it gets closed)  
24 + - [ ] Label the pull request with the  
25 +   appropriate area(s)  
26 + - [ ] Assign potential reviewers (you may  
27 +   also want to contact them on Teams to  
28 +   ensure timely reviews)  
29 + - [ ] Assign the project - FY22refresh  
30 + - [ ] Assign the pull request to yourself
```

▽ 306 ████ .github/workflows/release-pipeline.yaml → .github/workflows/release-workflow.yaml □

...

```
@@ -1,147 +1,159 @@
1 - name: devops-release-pipeline
2 -
3 - on:
4
5 - push:
6 - branches: [ main ]
7 - env:
8 - LAB_FOLDER: guides/microsoft-open-
9 - hack-devops-v2
10 - ARCHIVE_FOLDER: openhack-
11 - assets/archives/releases/microsoft-open-
12 - hack-devops-v2
13 -
14 - jobs:
15 - build:
16 - runs-on: ubuntu-latest
17 -
18 - steps:
19 - # Checks-out your repository under
20 - $GITHUB_WORKSPACE, so your job can
21 - access it
22 - - uses: actions/checkout@v2
23 -
24 - - name: 'Setup Node'
25 - - uses: actions/setup-node@v1
26 - - with:
27 - - node-version: 12.x
28 -
29 - - name: 'Linting OpenHack Content'
30 - - uses: nosborn/github-action-
31 - markdown-cli@v1.1.1
32 - - with:
33 - - # configuration file (JSON or
34 - YAML)
35 - - config_file:
36 - .markdownlint.json
37 - - # files, directories, or globs
38 - - files: ./portal
```

```
1 + name: devops-release-workflow
2 +
3 + #runs on push to main, archives a
4 + version [ARCHIVE_FOLDER] and also pushes
5 + a release to production storage
6 + [LAB_FOLDER]
7 +
8 + on:
9 +   push:
10 +     branches: [main]
11 +
12 + env:
13 +
14 + LAB_FOLDER: guides/microsoft-open-
15 + hack-devops
16 +
17 + ARCHIVE_FOLDER: openhack-
18 + assets/archives/releases/microsoft-open-
19 + hack-devops
20 +
21 +
22 + jobs:
23 +
24 + build:
25 +   runs-on: ubuntu-latest
26 +
27 + steps:
28 +
29 +   # Checks-out your repository under
30 +   $GITHUB_WORKSPACE, so your job can
31 +   access it
32 +   - uses: actions/checkout@v2
33 +
34 +   - name: "Setup Node"
35 +     uses: actions/setup-node@v2
36 +
37 +   with:
38 +
39 +     node-version: 12.x
40 +
41 +
42 +   # Once again, lint all your
43 +   content to ensure compliance
44 +   - name: "Linting OpenHack Content"
45 +     uses: nosborn/github-action-
46 +       markdown-cli@v3
47 +
48 +     with:
49 +
50 +       # configuration file (JSON or
51 +       YAML)
52 +       config_file:
```

```

30 -      - name: 'Linting Resources'
31 -          uses: nosborn/github-action-
32 -              markdown-cli@v1.1.1
33 -              with:
34 -                  # configuration file (JSON or
35 -                  YAML)
36 -                  config_file:
37 -                      .markdownlint.json
38 -                      # files, directories, or globs
39 -                      files: ./resources
40 -
41 -          # Install PanDoc
42 -          - name: Install Pandoc
43 -              run: sudo apt-get install pandoc
44 -
45 -          # Setup the output archive
46 -          environment
47 -          - name: Setup Build Environment
48 -              run: |
49 -                  mkdir output # create output
50 -                  dir
51 -                  sudo chmod -R 755 output
52 -
53 -                  mkdir output/portal
54 -                  sudo chmod -R 755
55 -                  output/portal
56 -                  mkdir output/portal/en
57 -                  sudo chmod -R 755
58 -                  output/portal/en
59 -                  mkdir output/portal/en/images
60 -                  sudo chmod -R 755
61 -                  output/portal/en/images
62 -
63 -                  mkdir output/resources
64 -                  sudo chmod -R 755
65 -                  output/resources
66 -
67 -          # Create HTML files

```

```

30 +      .markdownlint.json
31 +          # files, directories, or globs
32 +          files: ./portal
33 +
34 +          # Lint any additional resources
35 -          (or remove if not necessary)
36 +          - name: "Linting Resources"
37 +              uses: nosborn/github-action-
38 -                  markdown-cli@v3
39 +              with:
40 -                  # configuration file (JSON or
41 -                  YAML)
42 +                  config_file:
43 -                      .markdownlint.json
44 +                      # files, directories, or globs
45 -                      files: ./resources
46 +                      # ignore_files:
47 -                          "resources/SupportingFiles/react-rating-
48 -                          ui-ts/*"
49 +
50 +                      # Lint any json files, remove any
51 -                      non-existent/unnecessary paths
52 +                      - name: "Install dependencies"
53 +                          run: npm install jsonlint -g
54 +
55 +                      - name: "Execute json linter lab-
56 -                          content-definitions"
57 +                          run: |
58 -                              jsonlint ./portal/en/lab-
59 -                              content-definition.json
60 +
61 +                      # Install PanDoc
62 +                      - name: Install Pandoc
63 +                          run: sudo apt-get install pandoc
64 +
65 +                      # Setup the output archive
66 -                      environment, don't forget to set
67 -                      permissions. Build this to suit your
68 -                      needs for each language and outputs
69 +                      - name: Setup Build Environment
70 +                          run: |

```

```
58 -     - name: 'Create EN files'
59 -         run: |
60 -             FILE_LIST_EN=$(printf '%s;' \
61 -             ./portal/en/*.md)
62 -             #echo ${FILE_LIST_EN}
63 -             IFS=';' read -ra ENFILES <<<
64 -             ${FILE_LIST_EN}
65 -             for f in ${ENFILES[@]}; do
66 -                 HTML_FULL_PATH=`basename
67 -                 ${f}`
68 -                 #echo ${HTML_FULL_PATH}
69 -                 NEW_FILE=${HTML_FULL_PATH%/*}/html
70 -                 #echo ${NEW_FILE}
71 -                 pandoc --from
72 -                 markdown+raw_html --to html5
73 -                 ./portal/includes.yaml --title "OPEN
74 -                 HACK GUIDE" --no-highlight --
75 -                 output=output/portal/en/${NEW_FILE} ${f}
76 -                 cp ${f} output/portal/en/
77 -             done
78 -             #Copy Content Definitions
79 -             - name: 'Copy Lab content
80 -                 Definitions'
81 -                 run: |
82 -                     cp ./portal/en/lab-content-
83 -                     definition.json output/portal/en
84 -
85 -             #Copy images
86 -             - name: 'Copy Images'
87 -
88 -                 run: |
89 -                     cp ./portal/en/images/*
90 -                     output/portal/en/images
```

```
58 +         mkdir output # create output
      dir
59 +         sudo chmod -R 755 output
60 +         mkdir output/portal
61 +
62 +         sudo chmod -R 755
      output/portal
63 +         mkdir output/portal/en
64 +
65 +         sudo chmod -R 755
      output/portal/en
66 +         mkdir output/resources
67 +         sudo chmod -R 755
      output/resources
68 +
69 +         # Create HTML files for guides for
      each language. Remove any unnecessary
      steps
70 +
71 +         - name: "Create EN files"
72 +             run: |
73 +                 FILE_LIST_EN=$(printf '%s;' \
      ./portal/en/*.md)
74 +                 #echo ${FILE_LIST_EN}
75 +                 IFS=';' read -ra ENFILES << \
      ${FILE_LIST_EN}
76 +                 for f in ${ENFILES[@]}; do
77 +
78 +                     HTML_FULL_PATH=`basename \
      ${f}`
79 +
80 +                     #echo ${HTML_FULL_PATH}
81 +                     NEW_FILE=${HTML_FULL_PATH%/*}
82 +                     #echo ${NEW_FILE}
83 +                     pandoc --from
      markdown+raw_html --to html5
      ./portal/includes.yaml --metadata
      pagetitle="OPEN HACK GUIDE" --no-
      highlight --
      output=output/portal/en/${NEW_FILE} ${f}
84 +                     cp ${f} output/portal/en/
```

```

83 -      #Copy resources
84 -      - name: 'Copy Resources'
85 -          run: |
86 -
87 -              cp -r ./resources/*
88 -                  output/resources
89 -
90 -      #Create Archive
91 -
92 -          - name: 'create archive'
93 -              uses: actions/upload-
94 -                  artifact@main
95 -                  with:
96 -                      name: output
97 -                      path: output
98 -
99 -      - name: Get current time
100 -
101 -          uses: srfrnk/current-time@master
102 -          id: current-time
103 -          with:
104 -              format: YYYYMMDD
105 -
106 -      # Archive Deploy to storage
107 -      - name: 'Deploy Assets Archive'
108 -          Release'
109 -          env:
110 -              F_TIME: "${{ steps.current-
111 -                  time.outputs.formattedTime }}"
112 -          run: |
113 -              STORAGEARCHIVE="https://${{{
114 -                  secrets.STORAGE_ACCOUNT_NAME
115 -                  }}.blob.core.windows.net/${{{
116 -                  env.ARCHIVE_FOLDER }}}/${{F_TIME}}${{{
117 -                  secrets.STORAGE_ACCOUNT_SAS_TOKEN }}}"
118 -
119 -              #Download and install azcopy
120 -              pwd
121 -              wget -O azcopy.tar.gz
122 -                  https://aka.ms/downloadazcopy-v10-linux
123 -
124 -      done
125 -
126 -      #Copy Content Definitions [again,
127 -          map to your languages, remove any
128 -          unnecessary]
129 -      - name: "Copy Lab content
130 -          Definitions"
131 -          run: |
132 -              cp ./portal/en/lab-content-
133 -                  definition.json output/portal/en
134 -
135 -      #Copy images [again, map to your
136 -          languages, remove any unnecessary]
137 -      - name: "Copy Images"
138 -          run: |
139 -              cp ./portal/en/images/*
140 -                  output/portal/en/images
141 -
142 -      #Copy resources - set this section
143 -          to meet your needs on all the folders
144 -          you want to archive. These are not used
145 -          in production but are pushed to the
146 -          archive for history.
147 -      - name: "Copy Resources"
148 -          run: |
149 -              cp -r ./resources/*
150 -                  output/resources
151 -
152 -      #Create Archive
153 -      - name: "create archive"
154 -          uses: actions/upload-
155 -                  artifact@main
156 -          with:
157 -              name: output
158 -              path: output
159 -
160 -      - name: Get current time
161 -          uses: srfrnk/current-time@master
162 -          id: current-time
163 -          with:

```

```

111 -      tar -xf azcopy.tar.gz          111 +      format: YYYYMMDD
112 -      rm azcopy.tar.gz            112 +
113 -      #Copy the content to azure   113 +      # Archive Deploy to storage
114 -      blob storage               114 +      - name: "Deploy Assets Archive
115 -      echo "Copying content to Azure 115 +      Release"
116 -      blob storage               116 +      env:
117 -      #Copy the content to azure   117 +      F_TIME: "${{ steps.current-
118 -      blob storage               118 +      time.outputs.formattedTime }}"
119 -      echo "Copying content to Azure 119 +
120 -      blob storage               120 +      #Download and install azcopy
121 -      azcopy copy output          121 +      pwd
122 -      $STORAGEARCHIVE --recursive 122 +      wget -O azcopy.tar.gz
123 -      echo "Content copied to Azure 123 +      https://aka.ms/downloadazcopy-v10-linux
124 -      blob storage               124 +      tar -xf azcopy.tar.gz
125 -      echo $STORAGEARCHIVE          125 +      rm azcopy.tar.gz
126 -      # Deploy to Production - Release 126 +      #Copy the content to azure
127 -      - name: 'Deploy Assets Production 127 +      blob storage
128 -      Release'                   128 +      echo "Copying content to Azure
129 -      run: |                         129 +      blob storage"
130 -      DESTINATION="https://${{
131 -      secrets.STORAGE_ACCOUNT_NAME_PROD 130 +      #Copy the content to azure
132 -      }}.blob.core.windows.net/${{
133 -      env.LAB_FOLDER }}${{
134 -      secrets.STORAGE_ACCOUNT_SAS_TOKEN_PROD 131 +      blob storage
135 -      }}"                           132 +      echo "Copying content to Azure
136 -      #Download and install azcopy     133 +      blob storage"
137 -      pwd                            134 +      azcopy copy output
138 -      wget -O azcopy.tar.gz          135 +      $STORAGEARCHIVE --recursive
139 -      https://aka.ms/downloadazcopy-v10-linux 136 +      echo "Content copied to Azure
140 -      tar -xf azcopy.tar.gz          137 +      blob storage"
141 -      rm azcopy.tar.gz            148 +      echo $STORAGEARCHIVE

```

```

136 -      #Copy the content to azure
136 +      blob storage
137 -      echo "Copying content to Azure
137 +      blob storage lab folder"
138 -
138 +      # Deploy to Production - Release -
139 -      azcopy copy
139 +      make sure to map all of your languages,
140 -      'output/portal/en/*' $DESTINATION --
140 +      remove any unnecessary, non-existent.
140 +      recursive
140 +
140 +      - name: "Deploy Assets"
140 +      run: |
140 +
140 +      DESTINATION="https://${{{
140 +      secrets.STORAGE_ACCOUNT_NAME_PROD
140 +      }}}.blob.core.windows.net/${{{
140 +      env.LAB_FOLDER }}}${{{
140 +      secrets.STORAGE_ACCOUNT_SAS_TOKEN_PROD
140 +      }}}"
141 -      echo "Content copied to Azure
141 +      blob storage lab folder"
142 -
142 +      #Download and install azcopy
143 -      #Delete Archive
143 +      pwd
144 -      - name: Delete Output
144 +      wget -O azcopy.tar.gz
145 -      uses: geekyeggo/delete-
145 +      https://aka.ms/downloadazcopy-v10-linux
145 +      artifact@v1
146 -      with:
146 +      tar -xf azcopy.tar.gz
147 -      name: output
147 +
147 +      #Copy the content to azure
147 +      blob storage
148 -      echo "Copying content to Azure
148 +      blob storage lab folder"
149 -
150 +
151 +      azcopy copy
151 +      'output/portal/en/*' $DESTINATION --
151 +      recursive
152 +
153 +      echo "Content copied to Azure
153 +      blob storage lab folder"
154 +
155 +      #Delete Archive - nothing is
155 +      stored in GitHub
156 +      - name: Delete Output
157 +      uses: geekyeggo/delete-
157 +      artifact@v1
158 +      with:
159 +      name: output

```

```
... @@ -1,6 +1,8 @@
1 mqlint-config.json
2 .vscode/
3 **/*.DS_Store
4
5 # Word temporary
6 ~$*.doc*
```

```
1 mqlint-config.json
2 .vscode/
3 **/*.DS_Store
4 + output/
5 + output.sh
6
7 # Word temporary
8 ~$*.doc*
```

▼ 8 ████.markdownlint.json □

```
... @@ -1,6 +1,12 @@
1 {
2     "default": true,
3     "MD013": false,
4     "MD033": false,
5     - "MD007": { "indent": 4 }
6 }
```

```
1 {
2     "default": true,
3     "MD013": false,
4     + "MD024": false,
5     + "MD025": false,
6     "MD033": false,
7     + "MD034": false,
8     + "MD041": false,
9     + "MD007": {
10        "indent": 4
11    }
12 }
```

▼ 34 ██████ CI-pipeline.yaml □

Load diff

This file was deleted.

▼ 22 ████ README.md □

```
... @@ -1,22 +1,22 @@
1 # DevOps OpenHack
2
3 - This repo contains the master version of
   the DevOps OpenHack materials. If you
   would like to contribute see
   [Contributing](#contributing), below.
```

```
1 # DevOps OpenHack
2
3 + This repo contains the master version of
   the DevOps OpenHack materials. If you
   would like to contribute see
   [Contributing](#contributing), below.
```

```

4
5  ## Repo Structure
6
7  - * **portal** (English-language variant)
8  -   * **en**
9  -     * Markdown files for challenges
10 -       in the OpenHack portal
11 -         * JSON file defining lab
12 -           structure for Osgility portal
13 - * **resources**
14 -   * Coaches guide
15 -   * Additional resources for coaches
16 -     and OpenHack stakeholders
17 - * **solutions**
18 -   * validated solution code for
19 -     challenges
20
21  ### Instructions
22

```

```

4
5  ## Repo Structure
6
7  + - **portal** (English-language variant)
8  +   - **en**
9  +     - Markdown files for challenges
10 -       in the OpenHack portal
11 -         - JSON file defining lab
12 -           structure for Osgility portal
13 - * **resources**
14 -   - Coaches guide
15 -   - Additional resources for coaches
16 -     and OpenHack stakeholders
17 -   - **solutions**
18 -     - validated solution code for
19 -       challenges
20
21  ### Instructions
22

```

## 18 output.sh

```
@@ -0,0 +1,18 @@
```

```

1 +#!/bin/bash
2 +
3 + rm -rf output
4 + mkdir -p output/portal/en
5 + mkdir -p output/portal/en/images
6 +
7 + FILE_LIST_EN=$(printf '%s;' ./portal/en/*.md)
8 + echo ${FILE_LIST_EN}
9 + IFS=';' read -ra ENFILES <<<${FILE_LIST_EN}
10 + for f in ${ENFILES[@]}; do
11 +   HTML_FULL_PATH=$(basename ${f})

```

```
12 +     #echo ${HTML_FULL_PATH}
13 +     NEW_FILE=${HTML_FULL_PATH//md/html}
14 +     #echo ${NEW_FILE}
15 +     pandoc --from markdown+raw_html --to
16 +         html5 ./portal/includes.yaml --metadata
17 +             pagetitle="DevOps OpenHack -"
18 +             ${NEW_FILE/.html/}" --no-highlight --
19 +             output=output/portal/en/${NEW_FILE} ${f}
20 +             #cp ${f} output/portal/en/
21 +             done
22 +             cp ./portal/en/images/*
23 +             output/portal/en/images
```

▼ 19 portal/en/Challenge-00.md □

... @@ -0,0 +1,19 @@

```
1 + # Challenge 0 - Bring Your Own
2 + Subscription (BYOS)
3 +
4 +
5 + - No resources or configurations are
6 +   prepared in advance.
7 +
8 + ---
9 +
10 + This is Bring Your Own Subscription
11 + (BYOS) OpenHack. It means you have to do
12 + some preparation of the environment
13 + before you start working on real
14 + challenges.
15 +
16 + ## Challenge
17 +
18 + 1. Choose GitHub or Azure DevOps and
19 +   deploy only one lab environment based on
20 +   your team's decision!
21 +
22 + 1. Deployment steps and scripts are
23 +   located at [aka.ms/OpenHackBYOS]
24 +   (https://aka.ms/OpenHackBYOS) under
25 +   **devops** folder. This OpenHack is
26 +   tested and achievable using GitHub or
27 +   Azure DevOps.
```

```
17 + ## Success Criteria
18 +
19 + - OpenHack lab environment is
   provisioned.
```

▽ 82 portal/en/Challenge-01.md

```
...
@@ -0,0 +1,82 @@
1 + # Challenge 1 - Establish your plan
2 +
3 + **Reminders:**
4 +
5 + - Use an **InPrivate/Incognito** window
   in your browser to avoid any confusion
   with any other credentials you may use to
   access Azure resources.
6 + - The credentials you need to access the
   Azure subscription assigned to your team
   are available on the **OPEN HACK
   ENVIRONMENT** tab.
7 + - **GitHub path only!** Once you have
   configured your Git repository, you may
   see **security alerts** if you are an
   administrator in your GitHub
   organization. Do not be alarmed. These
   are intentional, and you will be required
   to resolve security issues in a later
   challenge.
8 +
9 + ---
10 +
11 + To have a successful DevOps strategy,
   your team needs to have a plan.
12 +
13 + Establishing a common plan for
   communication and collaboration is one of
   the main aspects of any collective work.
14 +
15 + Task boards are used to manage tasks
   better, prevent omissions, and provide
   project visibility to team members and
   management. The task board is a
   centralized information hub that provides
   a complete picture of your project. A
   task can be tracked from inception to
   completion.
```

```
16 +  
17 + **Resist the initial temptation to rush  
to the keyboard! Now is the time to pause  
and think as a team about the end-to-end  
deployment process that you would  
implement in this company.**  
18 +  
19 + ## Challenge  
20 +  
21 + Your challenge is to organize your team  
and establish a high-level view of the  
workflow you will implement during this  
event to achieve zero-downtime  
deployment.  
22 +  
23 + Based on your knowledge of the design  
architecture, your team must establish  
the fundamental processes needed to go  
from code to production. Focus only on a  
fundamental integration and deployment  
mechanism that considers the architecture  
of the infrastructure and application.  
24 +  
25 + Get familiar with the basic building  
blocks of a successful DevOps workflow,  
and be aware that your current plan may  
change throughout the event.  
26 +  
27 + Think about the following aspects:  
28 +  
29 + - Each part of API and infrastructure  
will potentially have a different life  
cycle.  
30 + - Your process should aim at minimizing  
the time it takes to deploy a feature  
into production.  
31 + - You want to be able to prioritize your  
tasks.  
32 + - Small changes are easier to deploy.  
33 +  
34 + Your Organization requires visibility  
into your project cycle. Design and  
implement a task board that meets the  
following organizational requirements:  
35 +  
36 + - All new issues need to be triaged
```

```
37 + - Once the issue has been triaged, it has  
38 to be moved into the Backlog  
39 + - Issues identified during Sprint  
planning will be shifted to the Sprint  
Backlog  
40 + - Issues that the Dev Team is actively  
working on should be moved into In  
Progress  
41 + - Issues with Pull Requests that need to  
be reviewed should be placed in Review In  
Progress  
42 + - Issues with PRs that have received  
approval and are ready to be merge have  
to be moved into the Review Approved  
43 +  
44 + ## Success Criteria  
45 +  
46 + - Explain to your coach your DevOps  
workflow. Then, be ready to answer the  
following questions:  
47 +  
48 + - What aspects of DevOps are you  
implementing?  
49 + - How are you splitting work amongst  
the team (who is going to be responsible  
for what)?  
50 + - How will you communicate amongst  
the team (Teams, Slack, verbal, ...)?  
51 +  
52 + - Demonstrate to your coach that you have  
implemented a basic task board to track  
work in progress that meets the  
organizational requirements.  
53 +  
54 + > NOTE  
55 + >  
56 + > To manage a project efficiently, you  
should consider implementing not only a  
task board, but other features like board  
automation, milestones/sprints, etc.  
However, this is not a part of the  
Challenge.  
57 +  
58 + ## GitHub Hints
```

```
59 +  
60 + GitHub introduces new Projects experience  
   (currently in Beta). Unfortunately, this  
   is not supported for this OpenHack -  
   please use the classic approach instead.  
61 +  
62 + ## References  
63 +  
64 + - <a href="https://docs.microsoft.com/en-  
   us/devops/what-is-devops"  
     target="_blank">What is DevOps?</a>  
65 + - <a href="https://developer-  
   tech.com/news/2016/jan/29/devops-  
   bridging-gap-between-business-and-  
   development/" target="_blank">Bridging  
   the gap between business and  
   development</a>  
66 + - <a href="https://docs.microsoft.com/en-  
   us/devops/plan/what-is-kanban"  
     target="_blank">What is Kanban?</a>  
67 + - <a  
     href="https://github.com/scgbear/kanban-  
     template/blob/main/docs/EngineeringPracti-  
     ces.md" target="_blank">Kanban Management  
     Best Practices</a>  
68 + - <a href="https://docs.microsoft.com/en-  
   us/devops/plan/planning-efficient-  
   workloads-with-devops"  
     target="_blank">Planning efficient  
   workloads with DevOps</a>  
69 +  
70 + - **GitHub**  
71 +  
72 + - <a  
     href="https://docs.github.com/en/issues/o-  
     rganizing-your-work-with-project-  
     boards/managing-project-boards/about-  
     project-boards/" target="_blank">Managing  
   GitHub Project Boards</a>  
73 + - <a  
     href="https://docs.github.com/en/issues/t-  
     racking-your-work-with-issues/about-  
     issues" target="_blank">Managing work  
   with GitHub Issues</a>  
74 + - <a  
     href="https://docs.github.com/en/issues/o-
```

```
rganizing-your-work-with-project-
boards/tracking-work-with-project-
boards/adding-issues-and-pull-requests-
to-a-project-board"
target="_blank">Adding issues and pull
requests to a project board</a>
75 +     - <a
      href="https://docs.github.com/en/issues/t
racking-your-work-with-issues/linking-a-
pull-request-to-an-issue"
      target="_blank">Linking a pull request to
an issue</a>
76 +
77 +     - **Azure DevOps**
78 +
79 +     - <a
      href="https://docs.microsoft.com/en-
us/azure/devops/organizations/settings/wo
rk/change-process-basic-to-agile?
view=azure-devops" target="_blank">Change
a project process from Basic to Agile</a>
80 +     - <a
      href="https://docs.microsoft.com/en-
us/azure/devops/boards/get-
started/customize-boards?view=azure-
devops&tabs=basic-process"
      target="_blank">Customize your Boards</a>
81 +     - <a
      href="https://docs.microsoft.com/en-
us/azure/devops/boards/work-
items/workflow-and-state-categories?
view=azure-devops&tabs=basic-process"
      target="_blank">How workflow states and
state categories are used in Azure DevOps
Backlogs and Boards</a>
82 +     - <a
      href="https://docs.microsoft.com/en-
us/azure/devops/boards/work-items/view-
add-work-items?view=azure-devops"
      target="_blank">View and add work items
using the Work Items page</a>
```

▽ 41 portal/en/Challenge-02.md 

... @@ -0,0 +1,41 @@

1 + # Challenge 2 - Setting up the

## Development Workflow

```
2 +
3 + **Reminders:**
4 +
5 + - Use an **InPrivate/Incognito** window
   in your browser to avoid any confusion
   with any other credentials you may use to
   access Azure resources.
6 + - Common references for your DevOps tool
   you can find under the **Cheat sheet**
   section.
7 +
8 + ---
9 +
10 + ## Challenge
11 +
12 + Configure and implement the mechanisms
   that will form the basis of the
   development workflow. You will need to
   set up the mechanism to prevent people
   from directly changing the code in your
   `main` branch without peer-reviewing
   (including code owners to establish
   mandatory reviewers). In addition, you
   would like to have clean commits history
   in your `main` branch and prevent people
   from merging their commits from the
   development phase.
13 +
14 + You are expected to implement a policy to
   enforce that code changes submitted must
   go through a peer review and formal
   approval process. Since the repository
   contains four APIs and IaC, code
   ownership should be defined per API and
   IaC. It should be mandatory to review the
   code by owner(s). The second policy must
   enforce that the commit history from your
   feature branch has not been visible in
   the `main` branch after the PR merging
   operation.
15 +
16 + > **NOTE:** You are expected to work on
   all the APIs of the MyDriving application
   and Infrastructure as Code part as well.
17 + >
```

```
18 + > **NOTE:** You are not expected to  
19   automate the build process yet. However,  
20   this will be required in the following  
21   challenges.  
22 + >  
23 + > **NOTE:** Discuss Git merging  
24   strategies with your Coach if you are  
25   unfamiliar with different merging  
26   opinions. The flat history might not  
27   always be the best strategy.  
28 +  
29 +  
30 + - Demonstrate to your coach that you have  
31   implemented a branch protection policy to  
32   prevent any code changes from being  
33   committed to the `main` branch without  
34   being reviewed. The policy must require  
35   at least two reviewers and including a  
36   code owner(s) review.  
37 + - Demonstrate to your coach that you have  
38   defined code owner(s) for each of the  
39   services.  
40 + - Demonstrate to your coach that you have  
41   implemented a branch protection policy to  
42   prevent development phase commits from  
43   being committed to the `main` branch.  
44 +  
45 +  
46 + ## References  
47 +  
48 + - About protected Branches  
49 + - About code owners  
50 + - About
```

```
      href="https://docs.github.com/en/reposi-  
      ries/configuring-branches-and-merges-in-  
      your-repository/defining-the-  
      mergeability-of-pull-requests/about-  
      protected-branches#require-linear-  
      history" target="_blank">About protected  
      branches - require linear history</a>  
35 +     - <a  
      href="https://docs.github.com/en/github/c-  
      ollaborating-with-pull-  
      requests/incorporating-changes-from-a-  
      pull-request/about-pull-request-  
      merges#squash-and-merge-your-pull-  
      request-commits" target="_blank">Squash  
      and merge your pull request commits</a>  
36 +  
37 +     - **Azure DevOps**  
38 +  
39 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/devops/repos/git/branch-  
      policies?view=azure-devops"  
      target="_blank">Improve code quality with  
      branch policies</a>  
40 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/devops/repos/git/branch-  
      policies?view=azure-devops#automatically-  
      include-code-reviewers"  
      target="_blank">Automatically include  
      code reviewers</a>  
41 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/devops/repos/git/merging-with-  
      squash?view=azure-devops"  
      target="_blank">Squash merge pull  
      requests</a>
```

▼ 134 portal/en/Challenge-03.md 

... @@ -0,0 +1,134 @@

```
1 + # Challenge 3 - Deploy Cloud  
    Infrastructure with Infrastructure as  
    Code (IaC)  
2 +  
3 +   **Reminders:**
```

```
4 +  
5 + - Use an **InPrivate/Incognito** window  
in your browser to avoid any confusion  
with any other credentials you may use  
to access Azure resources.  
6 + - Common references for your DevOps tool  
you can find under the **Cheat sheet**  
section.  
7 +  
8 + ---  
9 +  
10 + Infrastructure as Code (IaC) is the  
management of infrastructure (networks,  
virtual machines, load balancers, and  
connection topology) in a descriptive  
model, using the same versioning as  
DevOps team uses for source code. Like  
the principle that the same source code  
generates the same binary, an IaC model  
generates the same environment every  
time it is applied. IaC is a key DevOps  
practice and is used in conjunction with  
continuous delivery.  
11 +  
12 + Teams that implement IaC can rapidly  
deliver stable environments at scale,  
avoid manual configuration of  
environments, and enforce consistency by  
representing the desired state of their  
environments via code.  
13 +  
14 + ## Challenge  
15 +  
16 + Your Team just started preparing the  
official launch of the MyDriving  
solution. Currently, nothing is deployed  
to Azure. Your Team now needs to to  
deploy all the infrastructure resources  
required by the MyDriving application.  
17 +  
18 + You don't have to write the IaC code  
yourself, because one of the developers  
responsible for provisioning the Azure  
infrastructure prepared complete code to  
deploy all required resources in both  
[Bicep](https://docs.microsoft.com/en-
```

us/azure/azure-resource-manager/bicep/overview) and [Terraform] (<https://www.terraform.io/intro/index.html>) with similar functionality.

- 19 +
- 20 + To build the pipelines/workflows, you can draw inspiration from a bash script named `deploy.sh`, which contains all the steps required to run an end-to-end Bicep or Terraform deployment. Your task is to replicate these steps in a pipeline, while also respecting requirements described further.
- 21 +
- 22 + **\*\*Decision time!\*\*** Explore IaC code in your repository (you can find it under `iac` path) and select only the IaC tooling that best fits your team's skills or learning plans (Terraform or Bicep). The chosen technology will stay with you until the end of the OpenHack.
- 23 +
- 24 + > **\*\*NOTE:\*\*** Discuss Bicep vs. Terraform differences with your coach, like Stateless vs. Stateful deployment, ARM vs. API driven deployment, local/remote custom scripts execution inside IaC code.
- 25 +
- 26 + Your team's job is to leverage the IaC code selected and design a DevOps workflow to deploy the infrastructure needed by the MyDriving solution to Azure Cloud.
- 27 +
- 28 + The company's lead engineer wants to review proposed deployment changes of the infrastructure and provide a pre-deployment approval before these are pushed into production as this infrastructure is a critical part of your project and any small and incorrect change may cause a problem for the whole solution. However you are not expected to touch or change the existing IaC code in this challenge!

```
29 +  
30 + Furthermore, the lead of the project  
    requires high-quality IaC code as well.  
    Your team has to design a workflow to  
    validate the quality of the IaC code and  
    configuration of any potential changes  
    before the new code will be merged with  
    the `main` branch. Only linted code and  
    no faulty configuration should be pushed  
    to the `main` branch.  
31 +  
32 + > **NOTES:**  
33 + >  
34 + > - The IaC contains a full deployment  
    of cloud infrastructure and application  
    layer as well. Of course, this is a bad  
    practice, but this approach has been  
    implemented only for OpenHack purposes  
    for overall event flow simplification.  
    The good practice is always to separate  
    infrastructure and application parts.  
35 + >  
36 + > - Terraform requires remote state  
    configuration for a production workload.  
    A Resource Group and Storage Account is  
    already provisioned for this purpose on  
    your Azure Subscription. Look for  
    resources with _state_ in the name. If  
    your team has no experience with remote  
    states, start a discussion with your  
    coach about this concept. Terraform  
    configures remote state during the init  
    operation. This part requires  
    information to connect to the remote  
    state see next note.  
37 + >  
38 + > - The IaC code needs parameter values  
    to be provided as input. All required  
    details are set inside Repository  
    Secrets (GitHub) or Variable Groups  
    (Azure DevOps). More information can be  
    found in the sections below.  
39 +  
40 + ### GitHub  
41 +  
42 + If your team selected GitHub then basic
```

input data for your workflows is already provisioned inside Repository Secrets. Explore it first, and then try to use them inside your workflow.

```
43 +
44 + ##### Hints
45 +
46 + - IaC code during deployment requires RESOURCES_PREFIX. This variable is the name of your team and repository. You can hardcode this name or make your workflow more dynamic and generic with the following code:
47 +
48 +   ````yaml
49 +   jobs:
50 +     job1:
51 +       name: "Job 1"
52 +       outputs:
53 +         # Set output for next job
54 +         RESOURCES_PREFIX: ${{ steps.resources_prefix.outputs.result }}
55 +     steps:
56 +       # Get RESOURCES_PREFIX based on the repo name
57 +       - name: Get repo name
58 +         uses: actions/github-script@v5
59 +         id: resources_prefix
60 +         with:
61 +           result-encoding: string
62 +           script: return
63 +             context.repo.repo.toLowerCase()
64 +           # Usage for current job: ${{ steps.resources_prefix.outputs.result }}
65 +
66 +     job2:
67 +       name: "Job 2"
68 +       steps:
69 +         # Usage for next job: ${{
70 +           needs.job1.outputs.RESOURCES_PREFIX }}}
71 +         ````+
72 + ##### Azure DevOps
73 +
74 + If your team selected Azure DevOps then
```

basic parameter values for your pipelines is already provisioned inside Variable Groups. Explore it first, and then try to use them inside your workflow.

```
75 +
76 + ## Success Criteria
77 +
78 + - Demonstrate that the validation workflow is triggered only for proposed IaC code changes via a Pull Request.
79 + - Demonstrate that the validation workflow checks lint for the IaC.
80 + - Demonstrate that the validation workflow validates IaC configuration.
81 + - Demonstrate that the deployment workflow is triggered for a selected IaC only when changes to the IaC code are pushed to the `main` branch.
82 + - Demonstrate that the deployment workflow shows a preview for deployment or configuration changes.
83 + - Demonstrate that the deployment workflow has a manual approval process to see a preview of the changes before deploying into the **production** environment.
84 +
85 + ## References
86 +
87 + - <a href="https://docs.microsoft.com/en-us/devops/deliver/what-is-infrastructure-as-code" target="_blank">What is Infrastructure as Code?</a>
88 +
89 + - **Bicep**
90 +
91 + - <a href="https://docs.microsoft.com/en-us/azure/azure-resource-manager/bicep/overview" target="_blank">What is Bicep?</a>
92 + - <a href="https://docs.microsoft.com/en-
```

```
us/azure/azure-resource-
manager/bicep/linter#use-in-bicep-cli"
target="_blank">Use Bicep linter</a>
93 +     - <a
      href="https://docs.microsoft.com/en-
      us/cli/azure/deployment/sub?view=azure-
      cli-latest#az_deployment_sub_validate"
      target="_blank">Bicep sub-level
      deployment validate</a>
94 +     - <a
      href="https://docs.microsoft.com/en-
      us/azure/azure-resource-
      manager/bicep/deploy-what-if?tabs=azure-
      cli%2CCLI" target="_blank">Bicep
      deployment what-if operation</a>
95 +     - <a
      href="https://docs.microsoft.com/en-
      us/azure/azure-resource-
      manager/bicep/deploy-cli"
      target="_blank">How to deploy resources
      with Bicep and Azure CLI</a>
96 +
97 +   - **Terraform**
98 +
99 +     - <a
      href="https://www.terraform.io/intro/ind
      ex.html" target="_blank">Introduction to
      Terraform</a>
100 +     - <a
      href="https://www.terraform.io/docs/lan
      guage/state/remote.html"
      target="_blank">Terraform Remote
      State</a>
101 +     - <a
      href="https://www.terraform.io/docs/cli/
      commands/fmt.html"
      target="_blank">terraform fmt</a>
102 +     - <a
      href="https://www.terraform.io/docs/cli/
      commands/validate.html"
      target="_blank">terraform validate</a>
103 +     - <a
      href="https://www.terraform.io/docs/cli/
      commands/plan.html"
      target="_blank">terraform plan</a>
104 +     - <a
```

```
105     href="https://www.terraform.io/docs/cli/
           commands/apply.html"
           target="_blank">terraform apply</a>
+       - <a
+           href="https://www.terraform.io/docs/cli/
           commands/output.html"
           target="_blank">terraform output</a>
+
107   +   - **GitHub**
+
108   +
109   +   - <a
+
           href="https://docs.github.com/en/actions
           /security-guides/encrypted-
           secrets#using-encrypted-secrets-in-a-
           workflow" target="_blank">Using
           encrypted secrets in a workflow</a>
+
110   +   - <a
+
           href="https://docs.github.com/en/actions
           /learn-github-actions/workflow-syntax-
           for-github-actions#jobs"
           target="_blank">Workflow Job</a>
+
111   +   - <a
+
           href="https://docs.github.com/en/actions
           /learn-github-actions/workflow-syntax-
           for-github-actions#jobsjob_idneeds"
           target="_blank">Workflow Requiring
           dependent job</a>
+
112   +   - <a
+
           href="https://docs.github.com/en/actions
           /learn-github-actions/workflow-syntax-
           for-github-actions#jobsjob_idoutputs"
           target="_blank">Workflow Job outputs</a>
+
113   +   - <a
+
           href="https://docs.github.com/en/actions
           /deployment/using-environments-for-
           deployment" target="_blank">Using
           environments for deployment</a>
+
114   +   - <a
+
           href="https://github.com/marketplace/act
           ions/hashicorp-setup-terraform"
           target="_blank">GitHub Actions -
           Terraform</a>
+
115   +   - <a
+
           href="https://github.com/marketplace/act
           ions/azure-login" target="_blank">GitHub
           Actions - Azure Login</a>
```

```
116 +     - <a href="https://github.com/marketplace/actions/azure-cli-action" target="_blank">GitHub Actions - Azure CLI</a>
117 +     - <a href="https://github.com/marketplace/actions/bicep-build-output" target="_blank">GitHub Actions - Bicep Build</a>
118 +     - <a href="https://github.com/marketplace/actions/deployment-what-if" target="_blank">GitHub Actions - Azure Deployment What-If Action</a>
119 +     - <a href="https://github.com/marketplace/actions/deploy-azure-resource-manager-arm-template" target="_blank">GitHub Actions - Azure Resource Manager (ARM) deployment</a>
120 +
121 +   - Azure DevOps
122 +
123 +     - <a href="https://docs.microsoft.com/en-us/azure/devops/pipelines/library/variable-groups?view=azure-devops&tabs=yaml#use-a-variable-group" target="_blank">Use a variable group</a>
124 +     - <a href="https://docs.microsoft.com/en-us/azure/devops/pipelines/scripts/cli/pipeline-variable-group-secret-nonsecret-variables?view=azure-devops" target="_blank">Use a variable group's secret and nonsecret variables in an Azure Pipeline</a>
125 +     - <a href="https://docs.microsoft.com/en-us/azure/devops/pipelines/process/stages?view=azure-devops&tabs=yaml" target="_blank">Azure Pipelines Stages</a>
126 +     - <a href="https://docs.microsoft.com/en-
```

```
us/azure/devops/pipelines/process/phases  
?view=azure-devops&tabs=yaml"  
target="_blank">Azure Pipelines Jobs</a>  
127 +     - <a  
+         href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/process/deploy  
ment-jobs?view=azure-devops"  
+         target="_blank">Azure Pipelines  
Deployment jobs</a>  
128 +     - <a  
+         href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/process/variab  
les?view=azure-  
devops&tabs=yaml%2Cbatch#use-output-  
variables-from-tasks"  
+         target="_blank">Use output variables  
from tasks</a>  
129 +     - <a  
+         href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/process/variab  
les?view=azure-  
devops&tabs=yaml%2Cbatch#set-variables-  
in-scripts" target="_blank">Set  
variables in scripts</a>  
130 +     - <a  
+         href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/process/variab  
les?view=azure-  
devops&tabs=yaml%2Cbatch#set-variables-  
by-using-expressions"  
+         target="_blank">Set variables by using  
expressions</a>  
131 +     - <a  
+         href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/process/enviro  
nments?view=azure-devops#approvals"  
+         target="_blank">Environment  
Approvals</a>  
132 +     - <a  
+         href="https://marketplace.visualstudio.c  
om/items?itemName=charleszipp.azure-  
pipelines-tasks-terraform"  
+         target="_blank">Azure Pipelines Task -  
Terraform</a>  
133 +     - <a  
+         href="https://docs.microsoft.com/en-
```

```
us/azure/devops/pipelines/tasks/deploy/a
zure-cli?view=azure-devops"
target="_blank">Azure Pipelines Task -
Azure CLI</a>
134 + - <a
+ href="https://docs.microsoft.com/en-
us/azure/devops/pipelines/tasks/deploy/a
zure-resource-group-deployment?
view=azure-devops" target="_blank">Azure
Pipelines Task - Azure Resource Group
Deployment</a>
```

▽ 104 portal/en/Challenge-04.md

... @@ -0,0 +1,104 @@

```
1 + # Challenge 4 - Implement Continuous
+ Integration (CI) with Testing
2 +
3 + **Reminders:**+
4 +
5 + - Use an **InPrivate/Incognito** window
+ in your browser to avoid any confusion
+ with any other credentials you may use
+ to access Azure resources.
6 + - Common references for your DevOps tool
+ you can find under the **Cheat sheet**
+ section.
7 +
8 + ---
9 +
10 + No matter how well written the code or
+ experienced the dev team, the benefits
+ of continuous integration (CI)
+ automation will increase velocity and
+ quality of your code and prevent
+ unwanted scenarios such as "Well... it
+ works on my machine :)".
11 +
12 + ## Challenge
13 +
14 + In this challenge you will be expected
+ to create workflows that automate the
+ **building and unit testing** of at
+ least one of the APIs of your
+ application. You are not required to
+ publish code coverage reports, just test
```

results. Instead, focus exclusively on building and executing existing unit tests as part of your workflows. Any unit test failure should be highlighted as a comment in the PR. Carefully review the Readme for your chosen API as it may contain additional instructions. You will also need to consider the flow for a build failure and ensure that code for builds that fail doesn't get merged into the main branch.

```
15 +
16 + > **NOTE:** When selecting GitHub
Actions or Azure Pipeline tasks, be
prudent and select Actions/Tasks that
were created by the vendor as a first
choice. If you are going to use a third
party option make sure that you verify
that you are able to satisfy the success
criteria with your selection.
17 +
18 + > **NOTE:** Continuous Integration (CI)
uses best practices including automated
testing and ideally <a
href="https://trunkbaseddevelopment.com/"
target="_blank">trunk-based
development</a>. Automated testing will
be covered later and will extend the CI
setup completed here.
19 +
20 + ## Success Criteria
21 +
22 + - With your coach present, execute the
automation workflow, explain each step,
and show that each step completes
successfully.
23 + - Demonstrate that the workflow is
triggered for an API on a Pull Request
(PR) only when changes are made to code
for that API.
24 + - Demonstrate that the workflow(s)
complement existing branch protections
by performing verifications before a
Pull Request (PR) can be merged to your
protected branch.
25 + - Demonstrate that your workflow runs,
```

performs all of the required steps, and ultimately prevents the merging of the code when the workflow fails.

26 + - Demonstrate that your workflow can be triggered manually by developers on their feature branches for ad-hoc code validation - if build/test failed, then creates issue/bug in the backlog.

27 + - Demonstrate that your workflow adds a comment to your PR for each failed build during PR validation.

28 +

29 + > \*\*NOTE:\*\* You are required to make a change to the code that will result in the failure of a unit test. If you are uncertain, your coach will provide a breaking change to submit through your pipeline.

30 +

31 + **## Hints**

32 +

33 + To create well-formatted comments for PRs, you can use Markdown syntax and built-in variables. The code example for GitHub and Azure DevOps is below.

34 +

35 + **### GitHub**

36 +

37 + ````text`

38 + `### Unit Test `failure``

39 + `<details>`

40 + `<summary>Workflow details</summary>`

41 +

42 + `Workflow name: `#${github.workflow}``

43 + `Action: `#${github.event_name}``

44 + `Job: `#${github.job}``

45 + `PR: #${github.event.number}`

46 + `</details>`

47 +

48 + `Test details: [Run #${github.run_id}](${github.server_url})/${github.repository}/actions/runs/${github.run_id})`

49 + `Pusher: @#${github.actor}`

50 + `````

51 +

```
52 + ### Azure DevOps
53 +
54 + ```text
55 + ### Unit Test `failure`
56 + <details>
57 + <summary>Pipeline details</summary>
58 +
59 + Pipeline name: `$(Build.DefinitionName)`
60 + Action: `$(Build.Reason)`
61 + Job: `$(System.JobName)`
62 + PR:
63 +     [$(System.PullRequest.PullRequestId)]
64 +     ($(System.PullRequest.SourceRepositoryURI)/pullrequest/$(System.PullRequest.PullRequestId))
65 + </details>
66 +
67 + Test details: [Run #$(Build.BuildId)]
68 +     ($(System.CollectionUri)$(System.TeamProject)/_build/results?
69 +     buildId=$(Build.BuildId)&view=ms.vss-test-web.build-test-results-tab)
70 +
71 + Pusher: @<$(Build.RequestedForId)>
72 + ```
73 +
74 + ## References
75 +
76 + - <a href="https://docs.microsoft.com/en-us/devops/develop/what-is-continuous-integration" target="_blank">What is Continuous Integration (CI)?</a>
77 + - <a href="https://docs.microsoft.com/en-us/azure/devops/pipelines/apps/cd/azure-cicd-data-overview?view=azure-devops#what-is-cicd" target="_blank">What is CI/CD?</a>
78 + - <a href="https://docs.microsoft.com/en-us/devops/deliver/what-is-continuous-delivery" target="_blank">What is Continuous Delivery?</a>
79 +
80 + ### Workflow orchestration
81 +
82 +
```

```
77 +   - <a href="https://youtu.be/C7NFwlmAQu" target="_blank">What is Pipeline as Code? | One Dev Question: Abel Wang</a>
78 +
79 +   - **GitHub**
80 +
81 +     - <a href="https://docs.github.com/en/actions/automating-builds-and-tests/about-continuous-integration" target="_blank">About continuous integration (CI) workflows with GitHub Actions</a>
82 +     - <a href="https://docs.github.com/en/actions/advanced-guides/using-github-cli-in-workflows" target="_blank">Using GitHub CLI in workflows</a>
83 +     - <a href="https://cli.github.com/manual/" target="_blank">GitHub CLI manual</a>
84 +     - <a href="https://github.com/marketplace/actions/github-script" target="_blank">GitHub Script Action</a>
85 +     - <a href="https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-net" target="_blank">Building and testing .NET</a>
86 +     - <a href="https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-nodejs-or-python" target="_blank">Building and testing Node.js</a>
87 +     - <a href="https://docs.github.com/en/actions/automating-builds-and-tests/building-and-testing-java-with-maven" target="_blank">Building and testing Java with Maven</a>
88 +
89 +   - **Azure DevOps**
90 +
```

```
91 +     - <a
      href="https://marketplace.visualstudio.c
      om/items?
      itemName=mspremier.CreateWorkItem"
      target="_blank">Azure Pipelines Task -
      Create Work Item</a>
92 +     - <a
      href="https://marketplace.visualstudio.c
      om/items?itemName=CSE-DevOps.create-pr-
      comment-task" target="_blank">Azure
      Pipelines Task - Create Pull Request
      Comment</a>
93 +     - <a
      href="https://docs.microsoft.com/en-
      us/azure/devops/pipelines/ecosystems/dot
      net-core" target="_blank">Build, test,
      and deploy .NET Core apps</a>
94 +     - <a
      href="https://docs.microsoft.com/en-
      us/azure/devops/pipelines/ecosystems/jav
      ascript" target="_blank">Build, test,
      and deploy JavaScript and Node.js
      apps</a>
95 +     - <a
      href="https://docs.microsoft.com/en-
      us/azure/devops/pipelines/ecosystems/jav
      a" target="_blank">Build Java apps</a>
96 +     - <a
      href="https://docs.microsoft.com/en-
      us/azure/devops/pipelines/ecosystems/go"
      target="_blank">Build and test Go
      projects</a>
97 +
98 + ### Unit testing
99 +
100 +    - <a
      href="http://agiledata.org/essays/tdd.ht
      ml" target="_blank">Introduction to Test
      Driven Development (TDD)</a>
101 +    - <a
      href="https://docs.microsoft.com/en-
      us/dotnet/core/testing/unit-testing-
      with-dotnet-test"
      target="_blank">Writing and running
      units test with .NET Core</a>
102 +    - <a
```

```
    href="https://blog.risingstack.com/node-  
    hero-node-js-unit-testing-tutorial/"  
    target="_blank">Writing and running unit  
    tests with Node.js</a>  
103 + - <a  
    href="https://www.vogella.com/tutorials/  
    Mockito/article.html"  
    target="_blank">Writing and running unit  
    tests with Java</a>  
104 + - <a  
    href="https://blog.alexellis.io/golang-  
    writing-unit-tests/"  
    target="_blank">Writing and running unit  
    tests with GoLang</a>
```

▼ 81 portal/en/Challenge-05.md □

... @@ -0,0 +1,81 @@

```
1 + # Challenge 5 - Implement Continuous  
2 + Deployment (CD)  
3 + **Reminders:**  
4 +  
5 + - Use an **InPrivate/Incognito** window  
6 + in your browser to avoid any confusion  
7 + with any other Azure credentials.  
8 + - Common references for DevOps tools can  
9 + be found under the **Cheat sheet**  
10 + section.  
11 +  
12 + ---  
13 +  
14 + Now that you have successfully  
15 + implemented Continuous Integration (CI),  
16 + it is time to show that you can deploy a  
17 + container image for at least one of the  
18 + four APIs to Azure App Service.  
19 +  
20 + ## Challenge  
21 +  
22 + Earlier, each API of the MyDriving  
23 + application was built and deployed to an  
24 + Azure Container Registry using the  
25 + Terraform/Bicep scripts in the `iac`  
26 + directory. Mixing the infrastructure code  
27 + with the application is not recommended.
```

Your team has to separate the deployment of the application from IaC code and create a dedicated workflow for at least one API.

15 +  
16 + Using the tooling that you have implemented so far, you must build & push the images of each of the four APIs to a Container Registry and deploy each of them to your existing Azure App Service using Continuous Delivery (CD). Configuration steps will differ depending on whether your team decides to use Azure Container Registry or GitHub Packages to store your images.  
17 +  
18 + Ensure proper naming of your images. The tag format was already decided by your organization, so you should look for already deployed images in your subscription and replicate the naming convention.  
19 +  
20 + **### Hint**  
21 +  
22 + - The image tag format looks like `<Azure Container Registry URI>/<ACR repository name>:<Revision Number>`.  
23 +  
24 + **### Extended challenge (optional)**  
25 +  
26 + > \*\*NOTE:\*\* This extended, optional challenge is only for the teams who have chosen GitHub as their DevOps tool and want to explore GitHub Packages as Container Registry.  
27 +  
28 + By design, Azure Container Registry is the default registry for the MyDriving application. Container Registry is part of infrastructure architecture design. To make a change to GitHub Packages, your team has to leverage changes into the IaC part to set a new configuration per each APIs to use GitHub Packages.  
29 +

```
30 + ##### Hints
31 +
32 + - The image tag should take the format of
33 +   `ghcr.io/<Owner>/<Repository name>/<Image
34 +     Name>:<Revision Number>` where image name
35 +     follows the naming convention
36 +     `devopsoh/<Image Name>:<tag>`.
37 + - IaC code accepts optional parameters
38 +   like:
39 +     - Terraform:
40 +       `docker_registry_server_url` ,
41 +       `docker_registry_server_username` ,
42 +       `docker_registry_server_password`
43 +     - Bicep: `dockerRegistryServerUrl` ,
44 +       `dockerRegistryServerUsername` ,
45 +       `dockerRegistryServerPassword`
46 +     +
47 +     > **NOTE:** Your publish process should
48 +       not overwrite the existing image tag, so
49 +       consider how you will ensure unique
50 +       values for image tags.
51 +
52 + - The API(s) code is deployed to the
53 +   `staging` slot of the Azure App Service
54 +   every time code is pushed to `main` (if
55 +   the code compiles and all tests pass on
56 +   PRs before).
57 + > **NOTE:** Your branch/pull request
58 +   policy can be used to enforce workflow
59 +   and gates between CI and CD activities.
60 +
61 + ## References
62 +
63 + Continuous Delivery
64 +
65 + - <a href="https://docs.microsoft.com/en-
```

```
us/devops/deliver/what-is-continuous-
delivery" target="_blank">What is
Continuous Delivery?</a>
53 + - <a
54 href="https://continuousdelivery.com/#why-
-continuous-delivery" target="_blank">Why
Continuous Delivery?</a>
55 +
56 +
57 + - <a href="https://docs.microsoft.com/en-
us/azure/app-service/configure-custom-
container?pivots=container-linux"
target="_blank">Configure a custom
container for Azure App Service</a>
58 + - <a href="https://docs.microsoft.com/en-
us/azure/app-service/tutorial-custom-
container?pivots=container-linux"
target="_blank">Migrate custom software
to Azure App Service using a custom
container</a>
59 +
60 + GitHub
61 +
62 + - <a href="https://docs.microsoft.com/en-
us/azure/app-service/deploy-container-
github-action?tabs=service-principal"
target="_blank">Deploy a custom container
to App Service using GitHub Actions</a>
63 + - <a
64 href="https://github.com/marketplace/actions/azure-cli-action"
target="_blank">GitHub Actions - Azure
CLI Action</a>
65 + - <a
66 href="https://github.com/marketplace/actions/build-and-push-docker-images"
target="_blank">GitHub Actions - Build
and push Docker images</a>
67 + - <a
68 href="https://github.com/marketplace/actions/azure-webapp" target="_blank">GitHub
Actions - Azure WebApp</a>
69 + - <a
70 href="https://github.com/marketplace/actions/azure-app-service-settings"
```

```
target="_blank">GitHub Actions - Azure  
App Service Settings</a>  
67 +  
68 + Azure DevOps  
69 +  
70 + - <a href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/targets/webapp-  
on-container-linux?view=azure-  
devops&tabs=yaml" target="_blank">Deploy  
an Azure App custom container</a>  
71 + - <a href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/tasks/build/doc  
ker?view=azure-devops"  
target="_blank">Azure Pipelines - Docker  
task</a>  
72 + - <a href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/tasks/deploy/az  
ure-rm-web-app-deployment?view=azure-  
devops" target="_blank">Azure Pipelines -  
Azure App Service Deploy task</a>  
73 + - <a href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/tasks/deploy/az  
ure-rm-web-app-containers?view=azure-  
devops" target="_blank">Azure Pipelines -  
Azure Web App for Container task</a>  
74 + - <a href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/tasks/deploy/az  
ure-app-service-manage?view=azure-devops"  
target="_blank">Azure Pipelines - Azure  
App Service Manage task</a>  
75 +  
76 + Container Registries  
77 +  
78 + - <a href="https://docs.microsoft.com/en-  
us/azure/container-registry/container-  
registry-tutorial-quick-task"  
target="_blank">Build and deploy  
container images in the cloud with Azure  
Container Registry Tasks</a>  
79 + - <a href="https://docs.microsoft.com/en-  
us/azure/container-registry/container-  
registry-get-started-docker-cli?  
tabs=azure-cli" target="_blank">Push your  
first image to your Azure container  
registry using the Docker CLI</a>  
80 + - <a
```

```
81 href="https://docs.github.com/en/packages/working-with-a-github-packages-registry/working-with-the-container-registry" target="_blank">GitHub Packages - Working with the Container registry</a>
81 + - <a href="https://docs.github.com/en/packages/managing-github-packages-using-github-actions-workflows/publishing-and-installing-a-package-with-github-actions" target="_blank">GitHub Packages - Publishing and installing a package with GitHub Actions</a>
```

▽ 53 portal/en/Challenge-06.md

... @@ -0,0 +1,53 @@

```
1 + # Challenge 6 - Implement a Blue/Green deployment strategy
2 +
3 + **Reminders:**
4 +
5 + - Use an **InPrivate/Incognito** window in your browser to avoid any confusion with any other credentials you may use to access Azure resources.
6 + - Common references for your DevOps tool you can find under the **Cheat sheet** section.
7 +
8 + ---
9 +
10 + Classic deployment mechanisms require downtime, and if something goes wrong, you need to rollback or otherwise remediate the broken deployment.
11 +
12 + A blue/green deployment strategy involves having two environments and shifting traffic between them. Blue or green can be your current version with the other being the proposed next version. This allows you to 'instantaneously' shift between versions without downtime. In addition, if there is an issue with the new version, rolling back simply requires
```

reverting traffic to the previous deployment.

13 +  
14 + In the previous challenges, your team implemented workflow(s) for Continuous Integration, testing, and deployment to a Container Registry and to production Azure App Service. You also verified that updated container image was deployed and running in each API's respective web application. Unfortunately, during the deployment your team had interruptions in API(s) operation and availability. In this challenge you'll extend that capability to perform a Blue/Green deployment into your existing Azure App Service to minimize downtime during new version deployment.

15 +  
16 + > \*\*NOTE:\*\* Blue/Green deployment is different from flighting, canary testing and progressive exposure - those are covered in another challenge.

17 +  
18 + **## Challenge**  
19 +  
20 + Leveraging your selected orchestration tool, update your existing workflow(s) to automatically perform Blue/Green deployments into Azure App Service.

21 +  
22 + Think about your strategy as a team before you start developing your solution.

23 +  
24 + **### Hint**  
25 +  
26 + Before swapping the Blue and Green environments, ensure that the destination environment is online and available. Additional gates (e.g., health check) may be desired before switching. Health check endpoint is implemented for each of the APIs and you can use HTTP calls to check the response code during the deployment process (or choose a prebuilt extension

from the Marketplace, but be ready to explain your reasoning).

- 27 +
- 28 + **## Success Criteria**
- 29 +
- 30 + To successfully complete the challenge, implement Blue/Green deployments for each of the APIs.
- 31 +
- 32 + - Demonstrate that a code change in an API is deployed initially to a staging environment and automatically to production with "zero downtime".
- 33 + - Explain to your coach the logic that you are using for blue/green deployment and address the following points:
- 34 + - Describe the mechanism that validates that the container and web app is ready and reasons for choosing a particular approach
- 35 + - Demonstrate what triggers or constraints are used to determine when/whether it is safe to swap to the new version
- 36 + - Demonstrate the CI code that determines the version of the container in production (green) and version of the container in staging (blue)
- 37 +
- 38 + **## References**
- 39 +
- 40 + - <a href="https://martinfowler.com/bliki/BlueGreenDeployment.html" target="\_blank">Blue/Green deployment</a>
- 41 + - <a href="https://docs.microsoft.com/en-us/azure/app-service/deploy-staging-slots">Set up staging environments in Azure App Service</a>
- 42 + - <a href="https://medium.com/@sangeetavnsunchu/azure-slot-deployment-with-blue-green-deployment-model-b52cd5ffaf07" target="\_blank">Azure Slot Deployment with Blue-Green Deployment Model</a>
- 43 +

```
44 + - **GitHub**
45 +
46 +     - <a
47 +         href="https://github.com/marketplace/actions/url-health-check"
48 +             target="_blank">GitHub Actions - URL
49 +                 Health Check</a>
50 +
51 +     - <a
52 +         href="https://docs.microsoft.com/en-us/cli/azure/webapp/deployment/slot?view=azure-cli-latest#az_webapp_deployment_slot_swap"
53 +             target="_blank">az webapp deployment slot swap</a>
54 +
55 + - **Azure DevOps**
56 +
57 +     - <a
58 +         href="https://docs.microsoft.com/en-us/azure/devops/pipelines/process/environments?view=azure-devops#approvals"
59 +             target="_blank">Environment Approvals</a>
60 +
61 +     - <a
62 +         href="https://docs.microsoft.com/en-us/azure/devops/pipelines/process/approvals?view=azure-devops&tabs=check-pass#invoke-rest-api"
63 +             target="_blank">Define approvals and
64 +                 checks - Invoke REST API</a>
65 +
66 +     - <a
67 +         href="https://docs.microsoft.com/en-us/azure/devops/pipelines/tasks/deploy/azure-app-service-manage?view=azure-devops"
68 +             target="_blank">Azure Pipelines - Azure
69 +                 App Service Manage task</a>
```

▼ 122 portal/en/Challenge-07.md 

... @@ -0,0 +1,122 @@

```
1 + # Challenge 7 - DevSecOps basics - get
2 +     rid of secrets
3 +
4 +
5 + - **Reminders:**
```

1 + # Challenge 7 - DevSecOps basics - get  
2 + rid of secrets  
3 +  
4 +  
5 + - Use an \*\*InPrivate/Incognito\*\* window  
in your browser to avoid any confusion

with any other Azure credentials.

6 + - Common references for your DevOps tool  
you can find under the **\*\*Cheat sheet\*\***  
section.

7 +

8 + ---

9 +

10 + **## Background**

11 +

12 + Securely managing secrets (user  
accounts, passwords, connection strings,  
certificates, etc.) is essential to  
running secure applications in the  
cloud. As teams move to agile  
development with frequent  
deployments/releases, exposing secrets  
in code can be easy to overlook. When it  
comes to secrets, managing where the  
secrets are stored and how they are  
injected into code is only half of the  
problem. It is also crucial to make sure  
that secrets are updated routinely.

13 +

14 + **## Challenge**

15 +

16 + The application team of the fictitious  
insurance company is concerned that they  
might have leaked some information in  
the application that could expose the  
site to hackers. In this challenge, you  
will use the API repo provided to you.

17 + There are secrets exposed in the current  
codebase. This challenge will have you  
update your CI pipelines to catch leaked  
secrets and prevent a Pull Request that  
fails validation to be merged. Once  
detected, you will mitigate the issues  
by securely storing secrets across  
environments and removing them from any  
source code.

18 +

19 + **### Scan for secrets**

20 +

21 + The first part of this challenge will  
have you add credential scanning to your  
CI workflow. Your workflow should fail

if the credential scanning has a positive outcome. Inevitably you will discover that false positives are possible with credential scanning. Ensure that your credential scanning has filters that suppress any false positives that you discover.

```
22 +
23 + ### Hints
24 +
25 + In most cases, secret scanning tools are available on the market based on the patterns for common secrets like connection strings, access tokens, etc. However, if your codebase contains not commonly used secrets, the scanner may fail and not detect your secret. Therefore, it is not a "magic" solution that will solve your secret's problem - the most important thing is to raise awareness and avoid sensitive data in your code. Most tools allow you to extend and define your own rules/patterns. Below are some examples that may be helpful in this Challenge.
26 +
27 + #### Gitleaks
28 +
29 + Gitleaks custom rule for detecting common passwords based on keyword detection.
30 +
31 + ``toml
32 + [[rules]]
33 + id = "common-passwords"
34 + description = "Common Passwords"
35 + regex = '''(?
    i)1q2w3e|abc123|admin@123|admin123|alpha
    123|asdf|asdfgh|bienvenue|bienvenue123|c
    hangeme123|demopass|dragon|iloveyou|let
    ein|letmein!|letmein_apr|letmein_aug|let
    mein_dec|letmein_feb|letmein_jan|letmein
    _jul|letmein_jun|letmein_mar|letmein_may
    |letmein_nov|letmein_oct|letmein_sep|lov
    e|m0t2pass|m0t2passe|mot2pass|mot2passe|
    mustang|n0tallowed|n0tallowed123|notallo
```

```
wed123|oliver|p@55w0rd|p@55w0rd123|p@55w  
ord|p@55word123|p@5sw0rd|p@5sw0rd123|p@5  
sword|p@5sword123|p@s5w0rd|p@s5w0rd123|p  
@s5word|p@s5word123|p@55w0rd|p@ssw0rd|p@  
ssw0rd123|p4ssw0rd|password1|password123  
|qwert|qwertry|qwertry123|secret1|simplepa  
ss|support123|test123|trigger|trustno1|vi  
rus|welc0me123|welcome123'''
```

36

+ ````

37

+

38 + **##### GitHub Advanced Security -**

39 +

40 + GitHub Advanced Security secret scanning  
custom pattern for detecting common  
passwords based on keyword detection.

41 +

42 + ````plaintext

43 + (?)

```
i)1q2w3e|abc123|admin@123|admin123|alpha  
123|asdf|asdfgh|bienvenue|bienvenue123|c  
hangeme123|demopass|dragon|iloveyou|letm  
ein|letmein!|letmein_apr|letmein_aug|let  
mein_dec|letmein_feb|letmein_jan|letmein  
_jul|letmein_jun|letmein_mar|letmein_may  
|letmein_nov|letmein_oct|letmein_sep|10v  
e|m0t2pass|m0t2passe|mot2pass|mot2passe|  
mustang|n0tallowed|n0tallowed123|notallo  
wed123|oliver|p@55w0rd|p@55w0rd123|p@55w  
ord|p@55word123|p@5sw0rd|p@5sw0rd123|p@5  
sword|p@5sword123|p@s5w0rd|p@s5w0rd123|p  
@s5word|p@s5word123|p@55w0rd|p@ssw0rd|p@  
ssw0rd123|p4ssw0rd|password1|password123  
|qwert|qwertry|qwertry123|secret1|simplepa  
ss|support123|test123|trigger|trustno1|vi  
rus|welc0me123|welcome123
```

44 + ````

45 +

46 + **### Move secrets to Azure Key Vault**

47 +

48 + The second part of the challenge you  
will move your secrets to Azure Key  
Vault. Included in your repos are  
Infrastructure as Code (IaC) samples you  
can use to complete this step.

49 +

50 + > \*\*NOTE:\*\* If you can't see the secrets

in Key Vault (which you won't be able to), you can add an Access Policy to the Key Vault manually to give users in the Team access. However, be aware that any IaC run will reset the policies, which will need to be done again.

```
51 +
52 + ### Remove secrets from the source code
53 +
54 + In this step you will remove all secrets
   from your application code and configure
   your application code to read securely
   from Azure Key Vault. Once this is
   complete validate that the scanning no
   longer has any true positive results.
55 +
56 + ## Success Criteria
57 +
58 + - Demonstrate that your branch policy
   for `main` branch detects exposed
   secrets in code and fails the build
59 + - Demonstrate that the App Service with
   your APIs are configured to read secrets
   from Azure Key Vault
60 + - IaC code is updated to new
   configuration for security requirements
61 + - All exposed secrets have been
   identified and removed from code
62 +
63 + ## Extended Challenge (optional)
64 +
65 + > **NOTE:** This extended, optional
   challenge is only for the teams who want
   to explore key rotation of secrets in
   Azure Key Vault.
66 +
67 + Secret Rotation is a key feature that
   strengthens your overall security
   posture. This feature provides a
   backstop, should a secret be
   compromised, by limiting the lifetime
   that secrets are valid. In this part of
   the challenge you will implement a key
   rotation automation process using Azure
   Key Vault as the secret store. While
   there are passwordless solutions for
```

.NET and Azure SQL the polyglot nature of the applications services mean the team must stay with traditional password implementations for connections to the data store. In this challenge you will implement a password rotation pipeline to rotate the passwords using automation.

```
68 +
69 + ### Hints
70 +
71 + IaC and workflow for <a href="https://docs.microsoft.com/en-us/azure/key-vault/secrets/tutorial-rotation" target="_blank">Automate the rotation of a secret for resources that use one set of authentication credentials</a> you can find under your team's repo (see `support/sqlsecretrotation` path).
72 +
73 + ## Extended Challenge Success Criteria
74 +
75 + - An automated secret rotation solution for the Azure SQL Server is implemented, the code can use the new key and the old keys are deprecated.
76 +
77 + ## References
78 +
79 + ### Secret Scanning
80 +
81 + - <a href="https://docs.microsoft.com/en-us/azure/app-service/app-service-key-vault-references" target="_blank">Use Key Vault references for App Service and Azure Functions</a>
82 +
83 + - Azure DevOps**
84 +
85 + - <a href="https://docs.microsoft.com/en-us/azure/devops/repos/git/branch-policies?view=azure-devops&tabs=browser"
```

```
target="_blank">Improve code quality  
with branch policies</a>  
86 +     - <a  
     href="https://docs.microsoft.com/en-  
     us/azure/devops/pipelines/library/variab  
     le-groups?view=azure-  
     devops&tabs=yaml#link-secrets-from-an-  
     azure-key-vault" target="_blank">Key  
     Vault in Azure DevOps variable  
     groups</a>  
87 +     - <a  
     href="https://docs.microsoft.com/en-  
     us/azure/devops/pipelines/tasks/deploy/a  
     zure-key-vault?view=azure-devops"  
     target="_blank">Azure Key Vault task</a>  
88 +     - <a  
     href="https://marketplace.visualstudio.c  
     om/items?itemName=sariftools.scans"  
     target="_blank">SARIF SAST Scans Tab</a>  
89 +     - **CredScan (part of Secure  
     Development Tools aka Guardian)** (only  
     for Microsoft employees)  
90 +     - <a  
     href="https://marketplace.visualstudio.c  
     om/items?  
     itemName=secureddevelopmentteam.vss-  
     secure-development-tools "  
     target="_blank">Secure Development Tools  
(Guardian)</a>  
91 +     - <a  
     href="https://aka.ms/CredScan "  
     target="_blank">CredScan</a>  
92 +     - <a  
     href="https://eng.ms/docs/cloud-ai-  
     platform/developer-services/one-  
     engineering-system-1es/1es-  
     docs/credscan/credscan-overview#v2---  
     official-release-v2117---recommended"  
     target="_blank">CredScan recommended  
     version</a>  
93 +     - <a  
     href="https://secdevtools.azurewebsites.  
     net/helpPostAnalysis.html"  
     target="_blank">Getting started with the  
     Post-Analysis (Build Break)</a>  
94 +     - <a
```

```
95      href="https://strikecommunity.azurewebsites.net/articles/8148/credscan-false-positives-and-suppressions.html"
96      target="_blank">CredScan False Positives and Suppressions</a>
97      +     - <a
98      href="https://strikecommunity.azurewebsites.net/articles/8148/credscan-false-positives-and-suppressions.html"
99      target="_blank">Local Suppression</a>
100     +     - **Gitleaks**
101     +     - <a
102     href="https://github.com/zricethezav/gitLeaks" target="_blank">Gitleaks scanning tool</a>
103     +     - <a
104     href="https://marketplace.visualstudio.com/items?itemName=Foxholen1.Gitleaks"
105     target="_blank">Azure DevOps task for Gitleaks</a>
106     +     - <a
107     href="https://github.com/zricethezav/gitLeaks#configuration"
108     target="_blank">Your own secret detection rules for Gitleaks</a>
109     +     - <a
110     href="https://github.com/zricethezav/gitLeaks#configuration"
111     target="_blank">Gitleaks suppressing false positives with allow patterns</a>
112     +
113     +     - **GitHub**
114     +     - <a
115     href="https://github.com/marketplace/actions/azure-key-vault-get-secrets"
116     target="_blank">GitHub Action for Azure Key Vault</a>
117     +     - **GitHub Advanced Security (GHAS)**
118     +     - <a
119     href="https://docs.github.com/en/code-security/secret-scanning"
120     target="_blank">GitHub Advanced Security Secret Scanning</a>
121     +     - <a
122     href="https://docs.github.com/en/enterpr
```

```
ise-cloud@latest/code-security/secret-
scanning/defining-custom-patterns-for-
secret-scanning"
target="_blank">Defining custom patterns
for secret scanning</a>
107 +     - **Gitleaks**
108 +     - <a
      href="https://github.com/zricethezav/git
      leaks" target="_blank">Gitleaks scanning
      tool</a>
109 +     - <a
      href="https://github.com/marketplace/act
      ions/gitleaks-scanner"
      target="_blank">GitHub Action for
      Gitleaks</a>
110 +     - <a
      href="https://github.com/zricethezav/git
      leaks#configuration"
      target="_blank">Your own secret
      detection rules for Gitleaks</a>
111 +     - <a
      href="https://github.com/zricethezav/git
      leaks#configuration"
      target="_blank">Gitleaks suppressing
      false positives with allow patterns</a>
112 +     - <a
      href="https://docs.github.com/en/code-
      security/code-scanning/integrating-with-
      code-scanning/uploading-a-sarif-file-to-
      github" target="_blank">Uploading a
      SARIF file to GitHub</a>
113 +
114 + ### Secret Rotation
115 +
116 + - <a
      href="https://docs.microsoft.com/en-
      us/azure/key-vault/secrets/tutorial-
      rotation" target="_blank">Automate the
      rotation of a secret for resources that
      use one set of authentication
      credentials</a>
117 + - <a
      href="https://docs.microsoft.com/en-
      us/azure/key-vault/secrets/tutorial-
      rotation-dual?tabs=azure-cli"
      target="_blank">Automate the rotation of
```

```
    a secret for resources that have two  
    sets of authentication credentials</a>  
118 +  
119 + ### DevSecOps (takeaways after the  
    event)  
120 +  
121 + - <a  
    href="https://azure.microsoft.com/en-  
    us/resources/6-tips-to-integrate-  
    security-into-your-devops-practices/"  
    target="_blank">6 tips to integrate  
    security into your DevOps practices</a>  
122 + - <a  
    href="https://resources.github.com/downl  
    oads/GitHub-eBook-Built-in-Security-  
    Demo-Day-Promo.pdf"  
    target="_blank">GitHub e-Book  
    DevSecOps</a>
```

▽ 100 [■■■■■] portal/en/Challenge-08.md 

... @@ -0,0 +1,100 @@

```
1 + # Challenge 8 - Integrating quality and  
    security checks  
2 +  
3 + **Reminders:**  
4 +  
5 + - Use an **InPrivate/Incognito** window  
    in your browser to avoid any confusion  
    with any other Azure credentials.  
6 + - Common references for your DevOps tool  
    you can find under the **Cheat sheet**  
    section.  
7 +  
8 + ---  
9 +  
10 + At the completion of Challenge 7, you  
    implemented workflow(s) to perform unit  
    testing, continuous integration,  
    Blue/Green deployments, and basic  
    security features to protect your code  
    from secrets leaking. In addition, you  
    performed essential gating to verify  
    that the staging environment was online  
    before performing the Blue/Green swap.  
    However, there are always opportunities
```

to improve automated quality and security gates.

+  
+ Depending upon the business need, it is common to perform one or more of the following in your workflows:

+  
+ - **\*\*DevSecOps\*\***  
+  
+ - **\*\*Dependency Scanning\*\*** - Open source components are used in the majority of modern applications. This greatly accelerates development but can also introduce vulnerability or license issues. Dependency scanning can check for issues as part of your code commit or during CI/CD workflows.

+ - **\*\*Static Application Security Testing (SAST)\*\*** - Computers are useful in identifying patterns, including patterns in code that may be vulnerable, difficult to maintain, perform poorly or are otherwise buggy. Static Code Analysis can identify "code smells" and improve the value of your code reviews.

+ - **\*\*Dynamic Application Security Testing (DAST)\*\*** - Dynamic analysis focuses on the "outside" of the application and runs simulated attacks against the frontend, like an attacker would. DAST tools are able to automatically perform most common attacks (like SQL injection) and then look at test results and identify potential vulnerabilities in the application.

+ - **\*\*Variant Analysis\*\*** - Like Static Code Analysis, Variant Analysis can identify "code smells" and improve the value of your code reviews. However, Variant Analysis can also reduce false positives and otherwise provide more meaningful insights to your code. As a result, variant Analysis has proven extremely useful in security analysis scenarios.

```
20 +  
21 + - **Code Quality**  
22 +  
23 +     - **Manual Approvals** - Although  
    manual intervention is against some of  
    the core tenants of automation, it is  
    still required in some scenarios. In  
    this case, moving from a pre-production  
    environment to production requires the  
    approval of a specific individual or  
    team.  
24 +     - **Code Coverage** - Unit Tests can  
    provide a safety net and indicator of  
    code quality. However, unit tests  
    require ongoing maintenance and can give  
    a false sense of security. Code Coverage  
    helps you understand how much your code  
    is being tested and monitor whether  
    testing declines as code changes.  
25 +     - **Integration Testing** - After  
    deployment to a pre-production  
    environment, automated tests can run  
    against an integrated system, verifying  
    "units" of code and the completely  
    integrated system.  
26 +  
27 + ## Challenge  
28 +  
29 + In this challenge you will plan and  
    improve your workflow to support **_one  
    or more_* quality and security checks.  
    You must complete the implementation of  
    **_one of these enhancements from each  
    category below_* (DevSecOps + Code  
    Quality). Challenges in each section has  
    been arranged in increasing degrees of  
    complexity.  
30 +  
31 + ### DevSecOps Enhancements  
32 +  
33 + - _Dependency Scanning_ is frequently  
    performed as part of your commit  
    workflow or during PR or CI builds. If  
    you're using GitHub, implement security  
    alerts and automated security updates.  
    For other repositories, integrate
```

- [WhiteSource Bolt]  
(<https://marketplace.visualstudio.com/items?itemName=whitesource.whiteSource-bolt-v2>) or another vulnerability scanner into your PR or CI workflow.
- 34 + - Static Application Security Testing is generally performed as part of PR or CI builds and reporting is used to support peer reviews. Different computer languages may leverage different tools for static analysis.
- 35 + - Dynamic Application Security Testing can be executed when new version of the application is deployed to verify that no "easy" security flaws were introduced. The OWASP Foundation maintains a [list of vulnerability scanning tools]([https://owasp.org/www-community/Vulnerability\\_Scanning\\_Tools](https://owasp.org/www-community/Vulnerability_Scanning_Tools)) and also provides the [ZAP (Zed Attack Proxy)](<https://owasp.org/www-project-zap/>) tool as a flagship project.
- 36 + - Variant Analysis is more intensive than static analysis and is frequently decoupled from the pipeline. Variant analysis tools like [CodeQL] (<https://codeql.github.com/docs/>) can typically be configured to discover variants of vulnerabilities in your public repositories.
- 37 +
- 38 + **### Code Quality Enhancements**
- 39 +
- 40 + - Manual Approval is very common in enterprise environments, relying on a specific individual or team to review and approve a pre-production deployment before pushing to production.
- 41 + - Code Coverage is generally performed during unit testing to measure how much of your code is actually tested. Integrate code coverage reporting into your PR or CI workflow. You may also implement a gate such that the workflow fails if the amount of code coverage falls as code is changed.

42 + - Integration Tests are generally run against a pre-production environment to verify your components integrate well across your code base. The technology will be different than unit testing (e.g., Selenium et al) and should be implemented after deployment.

43 +

44 + > \*\*NOTE:\*\* Different source repositories and orchestration tools will have other mechanisms to implement these capabilities. Depending upon your tools, some of these may be significantly easier or more difficult to achieve.

45 +

46 + **## Success Criteria**

47 +

48 + - Demonstrate one or more of the above enhancements (one for each category DevSecOps + Code Quality). This will frequently require pushing changes through your workflow and showing and explaining the steps. It may also require demonstrating integration with external tools.

49 + - For a different enhancement, explain a design, the specific tools you would leverage, how they fit into your process, and how you would integrate them.

50 +

51 + **## References**

52 +

53 + - Dependency Scanning

54 + - GitHub

55 + - <a href="https://docs.github.com/en/code-security/supply-chain-security/keeping-your-dependencies-updated-automatically" target="\_blank">Dependabot - Keeping your dependencies updated automatically</a>

56 + - <a href="https://docs.github.com/en/code-security/supply-chain-security/managing-

```
vulnerabilities-in-your-projects-
dependencies/about-alerts-for-
vulnerable-dependencies"
target="_blank">About alerts for
vulnerable dependencies</a>
57 +     - <a
+         href="https://docs.github.com/en/code-
+             security/supply-chain-security/managing-
+                 vulnerabilities-in-your-projects-
+                     dependencies/configuring-dependabot-
+                         security-updates"
+                         target="_blank">Configuring Dependabot
+                             security updates</a>
58 +     - <a
+         href="https://whitesource.atlassian.net/
+             wiki/spaces/WD/pages/556007950/WhiteSour
+                 ce+Bolt+for+GitHub"
+                 target="_blank">Whitesource Bolt for
+                     GitHub</a>
59 +     - Azure DevOps
60 +     - <a
+         href="https://whitesource.atlassian.net/
+             wiki/spaces/WD/pages/1641644045/WhiteSou
+                 rce+Bolt+for+Azure+Pipelines"
+                 target="_blank">Whitesource Bolt for
+                     Azure Pipelines</a>
61 +     - <a
+         href="https://marketplace.visualstudio.c
+             om/items?itemName=dependency-
+                 check.dependencycheck"
+                 target="_blank">OWASP Dependency
+                     Check</a>
62 +   - Static Application Security Testing
63 +     - <a
+         href="https://en.wikipedia.org/wiki/List
+             _of_tools_for_static_code_analysis"
+             target="_blank">Static Code Analysis
+                 tools (Wikipedia)</a>
64 +     - GitHub
65 +       - <a
+           href="https://github.com/marketplace?
+               type=actions&query=lint"
+               target="_blank">Linters in the GitHub
+                   Actions marketplace</a>
66 +       - <a
+           href="https://sonarcloud.io/documentatio
```

```
n/getting-started/github/"  
target="_blank">SonarCloud - Getting  
started with GitHub</a>  
67 +     - <a  
+         href="https://github.com/marketplace/act  
ions/sonarcloud-scan"  
+         target="_blank">SonarCloud for GitHub  
Actions</a>  
68 +     - Azure DevOps  
69 +     - <a  
+         href="https://sonarcloud.io/documentatio  
n/getting-started/azure-devops/"  
+         target="_blank">SonarCloud - Getting  
started with Azure DevOps</a>  
70 +     - <a  
+         href="https://azuredavolabs.com/labs/v  
stsextend/sonarcloud/"  
+         target="_blank">Driving continuous  
quality of your code with SonarCloud</a>  
71 +   - Dynamic Application Security Testing  
72 +     - <a href="https://owasp.org/www-  
project-zap/" target="_blank">OWASP Zed  
Attack Proxy (ZAP)</a>  
73 +     - GitHub  
74 +     - <a  
+         href="https://github.com/marketplace/act  
ions/owasp-zap-baseline-scan"  
+         target="_blank">GitHub Action - OWASP  
ZAP Baseline Scan</a>  
75 +     - <a  
+         href="https://github.com/marketplace/act  
ions/owasp-zap-full-scan"  
+         target="_blank">GitHub Action - OWASP  
ZAP Full Scan</a>  
76 +     - Azure DevOps  
77 +     - <a  
+         href="https://marketplace.visualstudio.c  
om/items?itemName=CSE-DevOps.zap-  
scanner" target="_blank">OWASP/ZAP  
Scanning extension for Azure DevOps</a>  
78 +   - Variant Analysis  
79 +     - <a  
+         href="https://codeql.github.com/docs/"  
+         target="_blank">Variant Analysis with  
CodeQL</a>  
80 +     - <a href="https://lgtm.com/"
```

```
target="_blank">Continuous security  
analysis (with GitHub and SEMMLE)</a>  
81 + - Integration Testing  
82 +     - <a  
     href="https://softwaretestingfundamental  
s.com/integration-testing/"  
     target="_blank">Integration testing</a>  
83 +     - <a  
     href="https://www.guru99.com/unit-test-  
vs-integration-test.html"  
     target="_blank">Integration tests vs  
Unit tests</a>  
84 +     - <a  
     href="https://docs.microsoft.com/en-  
us/aspnet/core/test/integration-tests?  
view=aspnetcore-3.1"  
     target="_blank">Integration testing in  
.NET Core</a>  
85 +     - <a  
     href="https://www.cloudbees.com/blog/tes  
ting-in-go" target="_blank">Integration  
testing in Go</a>  
86 +     - <a  
     href="https://www.codementor.io/@olatund  
egaruba/integration-testing-supertest-  
mocha-chai-6zbh6sefz"  
     target="_blank">Integration testing with  
Node.JS</a>  
87 + - Code Coverage  
88 +     - <a href="https://coveralls.io/"  
     target="_blank">CoverAlls</a>  
89 +     - <a href="https://codecov.io/"  
     target="_blank">CodeCov</a>  
90 +     - <a  
     href="https://www.jacoco.org/jacoco/"  
     target="_blank">Jacoco</a>  
91 +     - GitHub  
92 +         - <a  
     href="https://github.com/marketplace/act  
ions/coveralls-github-action"  
     target="_blank">CoverAlls for GitHub  
Actions</a>  
93 +         - Azure DevOps  
94 +             - <a  
     href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/ecosystems/eco
```

```
systems?view=azure-devops"
target="_blank">Azure Pipelines
ecosystem (per-language code coverage
docs)</a>
95 + - Manual Approvals
96 +     - GitHub
97 +         - <a
98 href="https://docs.github.com/en/actions
99 /managing-workflow-runs/reviewing-
100 deployments" target="_blank">Reviewing
deployments</a>
+     - Azure DevOps
+         - <a
100 href="https://docs.microsoft.com/en-
us/azure/devops/pipelines/release/approval
s/?view=azure-devops"
target="_blank">Release gates and
approvals for Azure Pipelines</a>
+         - <a
100 href="https://docs.microsoft.com/en-
us/azure/devops/pipelines/tasks/utility/
manual-intervention?view=azure-devops"
target="_blank">Manual Intervention task
for Azure Pipelines</a>
```

▼ 87 portal/en/Challenge-09.md

... @@ -0,0 +1,87 @@

```
+ # Challenge 9 - Implement a Load Testing
+ & Monitoring solution with alerting
2 +
3 + **Reminders:**+
4 +
5 + - Use an **InPrivate/Incognito** window
+ in your browser to avoid any confusion
+ with any other Azure credentials.
6 + - Common references for your DevOps tool
+ you can find under the **Cheat sheet**
+ section.
7 +
8 + ---
9 +
10 + ## Challenge
11 +
12 + Your challenge is to define and implement
+ a Load Testing & Monitorin stratergy for
```

your APIs. First, carefully select the monitoring tool(s) you want to use.

- 13 +
- 14 + You are expected to achieve the following goals:
  - 15 +
  - 16 + - Implement a solution that performs Load Testing of your APIs.
  - 17 + - Implement a solution that monitors the health and performance of your APIs. You will have to collect data regarding the performance of your web applications and the APIs hosted in each web application.
  - 18 + - Define the performance baseline for the APIs running in Azure and implement a mechanism that will automatically raise an alert and create an incident in your work item tracking system if performance degradation is observed.
  - 19 + - Build a dashboard that will allow you to visualize the global health of your environment.
  - 20 +
  - 21 + > \*\*NOTE:\*\*
  - 22 + >
  - 23 + > It is recommended to discuss the capabilities of the tools that you have selected with your team. Keep in mind that your coach is here to help you make an educated decision.
  - 24 + >
  - 25 + > A sample JMX (JMeter XML) file is provided in the resources section on the teams repo.
  - 26 +
  - 27 + **## Success Criteria**
  - 28 +
  - 29 + Show your coach Load testing solution and the aggregated view of your application and infrastructure that includes the following:
  - 30 +
  - 31 + - Web application monitoring that displays the CPU utilization for each production web application
  - 32 + - The average response time is over 1

```
        minute for each production web
        application
33    + - A work item is created when API
        performance degradation is observed
34    + - The top 10 queries by duration over the
        last 24 hours for the **mydrivingDB**
        database
35    + - The percentage of DTU utilized on the
        **mydrivingDB** database
36    +
37    + Show your coach that an incident is
        automatically created when an alert is
        raised.
38    +
39    + ## References
40    +
41    + ### Azure resources
42    +
43    + - Azure Load Testing
44    +
45    +     - <a
        href="https://docs.microsoft.com/en-
        us/azure/load-testing/overview-what-is-
        azure-load-testing" target="blank">What
        is Azure Load Testing?</a>
46    +     - <a
        href="https://docs.microsoft.com/en-
        us/azure/load-testing/quickstart-create-
        and-run-load-test" target="blank">Create
        and run a load test with Azure Load
        Testing</a>
47    +     - <a
        href="https://docs.microsoft.com/en-
        us/azure/load-testing/how-to-
        parameterize-load-tests"
        target="blank">Conduct configurable load
        tests with secrets and environment
        variables</a>
48    +     - <a
        href="https://jmeter.apache.org/download_
        jmeter.cgi" target="blank">Apache
        JMeter</a>
49    +     - <a
        href="https://docs.microsoft.com/en-
        us/azure/load-testing/tutorial-identify-
        bottlenecks-azure-portal"
```

```
target="blank">Run a load test to  
identify performance bottlenecks in a web  
app</a>  
50 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/load-testing/tutorial-cicd-  
      github-actions" target="blank">Identify  
      performance regressions with Azure Load  
      Testing and GitHub Actions</a>  
51 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/load-testing/tutorial-cicd-  
      azure-pipelines" target="blank">Identify  
      performance regressions with Azure Load  
      Testing and Azure Pipelines</a>  
52 +  
53 +     - **Azure Monitor**  
54 +  
55 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/azure-monitor/overview"  
      target="blank">Azure Monitor overview</a>  
56 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/azure-monitor/logs/log-  
      analytics-tutorial" target="_blank">Log  
      Analytics tutorial</a>  
57 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/azure-  
      monitor/essentials/monitor-azure-  
      resource" target="_blank">Monitor Azure  
      resources with Azure Monitor</a>  
58 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/azure-monitor/app/azure-web-  
      apps" target="_blank">Application  
      Monitoring for Azure App Service</a>  
59 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/azure/azure-monitor/insights/azure-  
      sql" target="_blank">Monitor Azure SQL  
      Database using Azure SQL Analytics</a>  
60 +     - <a  
      href="https://docs.microsoft.com/en-  
      us/Azure/app-service/web-sites-monitor"
```

```
target="_blank">Monitor apps in Azure App  
Service</a>  
61 + - <a  
62 href="https://docs.microsoft.com/en-  
us/azure/app-service/faq-availability-  
performance-application-issues"  
target="_blank">Application performance  
FAQs for Web Apps in Azure</a>  
62 + - <a  
63 href="https://docs.microsoft.com/en-  
us/azure/azure-monitor/essentials/diagnostic-settings"  
target="_blank">Create diagnostic  
settings to send Azure Monitor platform  
logs and metrics to different  
destinations</a>  
63 + - <a  
64 href="https://docs.microsoft.com/en-  
us/azure/azure-monitor/alerts/alerts-overview" target="_blank">Overview of  
alerts in Microsoft Azure</a>  
64 + - <a  
65 href="https://docs.microsoft.com/en-  
us/azure/azure-monitor/alerts/alerts-log"  
target="_blank">Create, view, and manage  
log alerts using Azure Monitor</a>  
65 +  
66 + - Azure Logic Apps  
67 +  
68 + - <a  
69 href="https://docs.microsoft.com/en-  
us/azure/logic-apps/logic-apps-overview"  
target="_blank">What is Azure Logic Apps?  
</a>  
69 + - <a  
70 href="https://docs.microsoft.com/en-  
us/connectors/visualstudioteamservices/"  
target="_blank">Azure DevOps connector  
for Azure Logic Apps</a>  
70 + - <a  
71 href="https://docs.microsoft.com/en-  
us/connectors/github/"  
target="_blank">GitHub connector for  
Azure Logic Apps</a>  
71 +  
72 + - Kusto documentation and references**:
```

```
73 +
74 +      - The solutions in Azure will
    evolve to use <a
    href="https://docs.microsoft.com/en-
    us/azure/data-explorer/kusto/query/"
    target="_blank">Azure Data Explorer query
    language</a> (also know as <a
    href="https://docs.microsoft.com/en-
    us/azure/data-explorer/kusto/query/"
    target="_blank">Kusto Query Language
    (KQL)</a>)
75 +      - Syntax for regular expressions
    supported by <a
    href="https://docs.microsoft.com/en-
    us/azure/data-explorer/kusto/query/re2"
    target="_blank">Azure Data Explorer</a>
76 +
77 +  - **Regular Expressions**
78 +
79 +      - You can test your regular
    expression on this site: <a
    href="https://regexr.com/"
    target="_blank">RegExr: Learn, Build, &
    Test RegEx</a>
80 +      - You can use regular expressions
    in the `extract()` function in Kusto. For
    example, this code will extract and
    convert to a double the response time
    from the container output showed above:
81 +
82 +      ````sh
83 +
84 +      todouble(extract(@"CreateTripPoint ([0-
    9.]*)ms", 1, LogEntry))
85 +
86 +  - **Application availability and
    responsiveness**
87 +      - <a
    href="https://docs.microsoft.com/en-
    us/azure/azure-monitor/app/monitor-web-
    app-availability#alerts"
    target="_blank">Availability alerts of
    any web site</a>
```

## Load diff

This file was deleted.

▼ ↑ 22 portal/en/Challenge-8.md → portal/en/Challenge-10.md ⌂

```
... @@ -1,12 +1,11 @@
1 - # Implement phased rollout with rollback
2 -
3 - **Resist the initial temptation to rush
   on the keyboard! This is the time to
   pause and think as a team about your end-
   to-end deployment process that you would
   implement in this company.**
4
5 **Reminders:** 
6
7 - - Use an InPrivate/Incognito window in
   your browser to avoid any confusion with
   any other credentials that you may use to
   access Azure resources.
8
9 - -----
10
11 The executive committee is hesitant to
   deploy new features and would like to be
   able to validate any changes on a small
   subset of customers before deployment
   into production.
12
30
31 #### Stretch goal
32
33 - If you have implemented challenge 6,
   automate your rollback based on
   monitoring the environment.
34
35 ## Reference
```

```
1 + # Challenge 10 - Implement phased rollout
   with rollback
2
3 **Reminders:** 
4
5 + - Use an **InPrivate/Incognito** window
   in your browser to avoid any confusion
   with any other Azure credentials.
6 + - Common references for your DevOps tool
   you can find under the **Cheat sheet**
   section.
7
8 + ---
9
10 The executive committee is hesitant to
   deploy new features and would like to be
   able to validate any changes on a small
   subset of customers before deployment
   into production.
11
29
30 #### Stretch goal
31
32 + If you have implemented Challenge 9,
   automate your rollback based on
   monitoring the environment.
33
34 ## Reference
```

```
36
37 - - Topical References
38
39     - <a href="https://docs.microsoft.com/azure/app-service/deploy-staging-slots#route-traffic" target="_blank">Route traffic with Azure App Service and Deployment slots</a>
40     - <a href="https://docs.microsoft.com/azure/devops/migrate/phase-rollout-with-rings" target="_blank">Explore how to progressively expose your Azure DevOps extension releases in production to validate, before impacting all users</a>
41     - <a href="https://docs.microsoft.com/azure/architecture/framework/devops/deployment" target="_blank">Deployment considerations for DevOps</a>
42     - GitHub
43
44     - <a href="https://github.com/marketplace/actions/rollback-release" target="_blank">GitHub Action Rollback Release</a>
45     - <a href="https://help.github.com/actions/reference/context-and-expression-syntax-for-github-actions" target="_blank">Context and expression syntax for GitHub Actions</a>
46     - <a href="https://mattvsts.github.io/2019/07/">
```

```
35
36 + - **Topical References**
37 +
38     - <a href="https://docs.microsoft.com/azure/app-service/deploy-staging-slots#route-traffic" target="_blank">Route traffic with Azure App Service and Deployment slots</a>
39     - <a href="https://docs.microsoft.com/azure/devops/migrate/phase-rollout-with-rings" target="_blank">Explore how to progressively expose your Azure DevOps extension releases in production to validate, before impacting all users</a>
40     - <a href="https://docs.microsoft.com/azure/architecture/framework/devops/deployment" target="_blank">Deployment considerations for DevOps</a>
41     - <a href="https://www.c-sharpcorner.com/blogs/doing-canary-deployments-using-azure-web-app-deployment-slots" target="_blank">Canary Deployments Using Azure Web App Deployment Slots</a>
42 +
43 + - **GitHub**
44 +
45     - <a href="https://github.com/marketplace/actions/rollback-release" target="_blank">GitHub Action Rollback Release</a>
46     - <a href="https://help.github.com/actions/reference/context-and-expression-syntax-for-github-actions" target="_blank">Context and expression syntax for GitHub Actions</a>
47 +
48 + - **Azure Pipelines**
49 +
50     - <a href="https://mattvsts.github.io/2019/07/">
```

[07/how-to-implement-rollback-strategies-in-azure-pipelines/" target="\\_blank">How to implement rollback strategies in Azure Pipelines</a>](#)

[07/how-to-implement-rollback-strategies-in-azure-pipelines/" target="\\_blank">How to implement rollback strategies in Azure Pipelines</a>](#)

▼ 89  portal/en/Challenge-2.md 

**Load diff**

This file was deleted.

▼ 57  portal/en/Challenge-3.md 

**Load diff**

This file was deleted.

▼ 108  portal/en/Challenge-4.md 

**Load diff**

This file was deleted.

▼ 48  portal/en/Challenge-5.md 

**Load diff**

This file was deleted.

▼ 74  portal/en/Challenge-6.md 

## Load diff

This file was deleted.

✓ 78 ██████████ portal/en/Challenge-7.md 

## Load diff

This file was deleted.

✓ 116 ██████████ portal/en/Overview.md 

## Load diff

This file was deleted.

✓ 131 ██████████ portal/en/README.md 

... @@ -0,0 +1,131 @@

```
1 + # DevOps OpenHack
2 +
3 + During this OpenHack, your team will
4 + implement DevOps practices to achieve
5 + Secure Zero Downtime Deployment
6 + targeting an application and APIs hosted
7 + in <a
8 + href="https://docs.microsoft.com/en-
9 + us/azure/app-service/overview#app-
10 + service-on-linux" target="_blank">Azure
11 + App Service on Linux</a>, which is
12 + hosting a series of Linux-based
13 + containers.
14 +
15 + Any resemblance to actual scenarios,
16 + issues, or pain points that you are
```

facing is **\*\*not\*\*** purely coincidental and definitions of the challenges presented to you during this event are inspired by real life.

```
6 +
7 + ## OpenHack lab environment provisioning
8 +
9 + > **Attention!** No resources or
   configurations are prepared in advance.
   You or your team are responsible for the
   GitHub or Azure DevOps provisioning!
10 +
11 + Deployment steps and scripts are located
   at [aka.ms/OpenHackBYOS]
   (https://aka.ms/OpenHackBYOS) under
   **devops** folder. This OpenHack is
   tested and achievable using GitHub or
   Azure DevOps.
12 +
13 + > **NOTE:** Choose GitHub or Azure
   DevOps and deploy only one lab
   environment based on your team's
   decision!
14 +
15 + ## Context
16 +
17 + Your team is the IT team of a fictitious
   insurance company. The company is
   offering its customers the ability to
   evaluate their driving skills. A mobile
   application collects the data from the
   car and sends them to a set of APIs,
   which consider the trip that has just
   been completed. Your customers can
   connect to a web application that uses
   the same APIs to review their trips and
   driving scores. Any downtime of the APIs
   would significantly impact your
   business.
18 +
19 + **Your mission is to update the
   infrastructure and APIs to incorporate
   business requirements while keeping the
   application functional.**
20 +
21 + Your team's success depends on your
```

ability to perform those updates and minimize the downtime of your application after your release production environment.

```
22 +
23 + ## Architecture
24 +
25 + The application is composed of :
26 +
27 + - **Tripviewer** (.NET Core): The team website your customers use to review their driving scores and trips is being simulated against the APIs.
28 + - **Trips** (Go): The trips API is where the mobile application sends the trip data from the OBD device to be stored.
29 + - **POI**: (.NET Core) The POI (Points Of Interest) API is collecting the points of the trip when a hard stop or hard acceleration has detected.
30 + - **User** (Node): The user profile API is used by the application to read the user's information.
31 + - **UserJava** (Java): The user-java API is used by the application to create and modify the users.
32 +
33 + A simulator runs throughout the entire event after you spin up infrastructure to send your APIs the data your users generate in real life.
34 +
35 + The following schema shows the overall architecture:
36 +
37 + ![[My Driving architecture]
      (./images/MyDriving-architecture.png)]
38 +
39 + ### Keep the lights up
40 +
41 + There is no "maintenance window". Your goal throughout this OpenHack is to implement a strategy for a zero-downtime deployment.
42 +
43 + ## Make decision
```

```
44 +  
45 + Select the tooling that best fits your  
team's skills or learning plans (GitHub  
or Azure DevOps). To provision lab  
environment and do initial DevOps  
platform setup, go to **OpenHack lab  
environment provisioning** section.  
46 +  
47 + ## Cheat sheet  
48 +  
49 + In this section, you will find a list of  
information to help you get started with  
the technologies used during the DevOps  
OpenHack.  
50 +  
51 + ### Obtaining the credentials of your  
team environment  
52 +  
53 + 1. Navigate to the **OPEN HACK  
ENVIRONMENT** tab.  
54 + 2. The usernames and passwords provided  
can be used to access your Azure  
subscription.  
55 +  
56 + ### Database credentials  
57 +  
58 + The existing Azure SQL Database  
credentials are as follows:  
59 +  
60 + - Username: **demousersa**  
61 + - Password: **demo!P@55w0rd123**  
62 +  
63 + ### DevOps tool references  
64 +  
65 + General references to common things used  
for almost every challenge  
66 +  
67 + #### GitHub References  
68 +  
69 + General GitHub references to common  
things used for almost every challenge.  
70 +  
71 + - <a  
    href="https://docs.github.com/en/actions  
/learn-github-actions/workflow-syntax-  
for-github-actions"
```

```
target="_blank">Workflow syntax for  
GitHub Actions</a>  
72 + - <a  
    href="https://docs.github.com/en/actions  
/learn-github-actions/environment-  
variables" target="_blank">Environment  
variables</a>  
73 + - <a  
    href="https://docs.github.com/en/actions  
/learn-github-actions/contexts"  
    target="_blank">Contexts</a>  
74 + - <a  
    href="https://docs.github.com/en/actions  
/learn-github-actions/expressions"  
    target="_blank">Expressions</a>  
75 + - <a  
    href="https://docs.github.com/en/actions  
/learn-github-actions/events-that-  
trigger-workflows"  
    target="_blank">Events that trigger  
workflows</a>  
76 + - <a  
    href="https://docs.github.com/en/actions  
/learn-github-actions/workflow-commands-  
for-github-actions"  
    target="_blank">Workflow commands</a>  
77 +  
78 + ##### Azure DevOps References  
79 +  
80 + General Azure DevOps references to  
common things used for almost every  
challenge.  
81 +  
82 + - <a  
    href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/yaml-schema"  
    target="_blank">Azure Pipelines YAML  
schema reference</a>  
83 + - <a  
    href="https://docs.microsoft.com/en-  
us/azure/devops/pipelines/build/variable  
s?view=azure-devops&tabs=yaml"  
    target="_blank">Use predefined  
variables</a>  
84 + - <a  
    href="https://docs.microsoft.com/en-
```

```
us/azure/devops/pipelines/process/conditions?tabs=yaml&view=azure-devops"
target="_blank">Specify conditions</a>
85 + - <a
     href="https://docs.microsoft.com/en-
us/azure/devops/pipelines/process/variables?view=azure-devops&tabs=yaml%2Cbatch"
     target="_blank">Define variables</a>
86 + - <a
     href="https://docs.microsoft.com/en-
us/azure/devops/pipelines/repos/azure-
repos-git?view=azure-
devops&tabs=yaml#ci-triggers"
     target="_blank">CI triggers</a>
87 + - <a
     href="https://docs.microsoft.com/en-
us/azure/devops/pipelines/repos/azure-
repos-git?view=azure-
devops&tabs=yaml#pr-triggers"
     target="_blank">PR triggers</a>
88 +
89 + ## Command line references
90 +
91 + The latest [GitHub CLI version can be
   found here](https://cli.github.com) if a
   suitable version 2.5.0 or later is not
   already installed.
92 +
93 + The latest [Azure CLI version can be
   found here]
   (https://docs.microsoft.com/en-
us/cli/azure/install-azure-cli?
   view=azure-cli-latest) if a suitable
   version 2.33.1 or later is not already
   installed.
94 +
95 + The latest [Terraform can be found here]
   (https://www.terraform.io/downloads.html
   ) if a suitable version 1.1.5 or later
   is not already installed.
96 +
97 + ## Glossary
98 +
99 + Common term definitions:
100 +
101 + ### Azure App Service on Linux
```

102 +  
103 + [Azure App Service]  
(<https://docs.microsoft.com/en-us/azure/app-service/overview>) is a  
fully managed compute platform that is  
optimized for hosting websites and web  
applications. Customers can use [App  
Service on Linux]  
(<https://docs.microsoft.com/en-us/azure/app-service/overview#app-service-on-linux>) to host web apps  
natively on Linux for supported  
application stacks.  
104 +  
105 + **### ACR - Azure Container Registry**  
106 +  
107 + This is the private registry for Docker  
containers in Azure. The containers  
stored in an [Azure Container Registry]  
(<https://docs.microsoft.com/en-us/azure/container-registry/container-registry-intro>) can only be accessed by  
the resources that have been granted  
access.  
108 +  
109 + **### ACI - Azure Container Instances**  
110 +  
111 + [Azure Container Instances]  
(<https://docs.microsoft.com/en-us/azure/container-instances/container-instances-overview>) offers the fastest  
and simplest way to run a container in  
Azure, without having to manage any  
virtual machines and without having to  
adopt a higher-level service.  
112 +  
113 + Azure Container Instances is a great  
solution for any scenario that can  
operate in isolated containers,  
including simple applications, task  
automation, and build jobs.  
114 +  
115 + **### Azure SQL Database**  
116 +  
117 + [Azure SQL Database]  
(<https://docs.microsoft.com/en>-

us/azure/azure-sql/database/sql-database-paas-overview) is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user involvement.

118 +  
119 + **### GitHub**  
120 +  
121 + [GitHub](https://github.com) is a web-based version-control and collaboration platform for software developers. GitHub simplifies the process of working with other people and makes it easy to collaborate on projects.  
122 +  
123 + **### Azure DevOps**  
124 +  
125 + [Azure DevOps]  
(https://docs.microsoft.com/en-us/azure/devops/user-guide/what-is-azure-devops?view=azure-devops) provides developer services for support teams to plan work, collaborate on code development, and build and deploy applications. Azure DevOps supports a culture and set of processes that bring developers and project managers and contributors together to complete software development. It allows organizations to create and improve products at a faster pace than they can with traditional software development approaches.  
126 +  
127 + **### Docker container**  
128 +  
129 + A [container]  
(https://www.docker.com/resources/what-container) is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. A Docker container image is a lightweight,

standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings.

130

+

131

+ Container images become containers at runtime and in the case of Docker containers - images become containers when they run on Docker Engine (and in the case of this application and its APIs - Docker Engine running in Azure App Service on Linux). Available for both Linux and Windows-based applications, containerized software will always run the same, regardless of the infrastructure. Containers isolate software from its environment and ensure that it works uniformly despite differences for instance between development and staging.

▽ BIN **-229 KB** portal/en/images/DevOps OpenHack architecture.png 

Deleted file not rendered

▽ BIN **+88.1 KB** portal/en/images/MyDriving-architecture.png 



▽ 73  portal/en/lab-content-definition.json 

```
...     @@ -1,65 +1,86 @@
1      {
2      - "Title": "OpenHack - DevOps V2 Zero
3          Downtime Deployment",
4      - "Description": "OpenHack challenges
5          for performing Zero Downtime
6          Deployments",
7      - "OverviewUrl": "Overview.html",
```

```
1      {
2      + "Title": "DevOps OpenHack - Secure Zero
3          Downtime Deployments",
4      + "Description": "OpenHack challenges for
5          performing Secure Zero Downtime
6          Deployments",
7      + "OverviewUrl": "README.html",
```

```

5  - "ContentVersion": "1.0",
6  "ContentPackage": {
7    "ContentItems": [
8      {
9        "Order": 1,
10       "Title": "Challenge 1: Establish
11      your plan",
12      "Description": "Establish your
13      plan",
14      "Url": "Challenge-1.html"
15    },
16    {
17      "Order": 2,
18      "Follows": 1,
19      "Title": "Challenge 2: Setting up
20      development workflow",
21      "Description": "Setting up
22      development workflow",
23      "Url": "Challenge-2.html"
24    },
25    {
26      "Order": 3,
27      "Follows": 2,
28      "Title": "Challenge 3: Implement
29      continuous testing",
30      "Description": "Implement
31      continuous testing",
32      "Url": "Challenge-3.html"
33    },
34    {
35      "Order": 4,
36      "Follows": 3,
37      "Title": "Challenge 4: Implement
38      continuous deployment (CD)",

```

```

5  + "ContentVersion": "3.0",
6  "ContentPackage": {
7    "ContentItems": [
8      {
9        "Order": 0,
10       "Title": "Challenge 0: Bring Your
11      Own Subscription (BYOS)",
12       "Description": "Bring Your Own
13      Subscription (BYOS)",
14       "Url": "Challenge-00.html"
15     },
16     {
17       "Order": 1,
18       "Follows": 0,
19       "Title": "Challenge 1: Establish
20      your plan",
21       "Description": "Establish your
22      plan",
23       "Url": "Challenge-01.html"
24     },
25     {
26       "Order": 2,
27       "Follows": 1,
28       "Title": "Challenge 2: Setting up
29      development workflow",
30       "Description": "Setting up
31      development workflow",
32       "Url": "Challenge-02.html"
33     },
34     {
35       "Order": 3,
36       "Follows": 2,
37       "Title": "Challenge 3: Deploy
38      Cloud Infrastructure with Infrastructure
      as Code",
39       "Description": "Deploy Cloud
      Infrastructure with Infrastructure as
      Code",
40       "Url": "Challenge-03.html"
41     },
42     {
43       "Order": 4,
44       "Follows": 3,
45       "Title": "Challenge 4: Implement
46      Continuous Integration (CI) with
47      Testing",
48     }

```

```
32 -     "Description": "Implement  
continuous deployment (CD)",  
33 -     "Url": "Challenge-4.html"  
34     },  
35     {  
36         "Order": 5,  
37         "Follows": 4,  
38 -     "Title": "Challenge 5: Implement  
a Blue/Green deployment strategy",  
39 -     "Description": "Implement a Blue/Green deployment strategy",  
40 -     "Url": "Challenge-5.html"  
41     },  
42     {  
43         "Order": 6,  
44         "Follows": 5,  
45 -     "Title": "Challenge 6: Implement  
a monitoring solution for your MyDriving  
APIs",  
46 -     "Description": "Implement a monitoring solution for your MyDriving  
APIs",  
47 -     "Url": "Challenge-6.html"  
48     },  
49     {  
50         "Order": 7,  
51         "Follows": 5,  
52 -     "Title": "Challenge 7:  
Integrating quality and security gates",  
53 -     "Description": "Integrating  
quality and security gates",  
54 -     "Url": "Challenge-7.html"  
55     },  
56     {  
57         "Order": 8,  
58         "Follows": 5,  
59 -     "Title": "Challenge 8: Implement  
phased rollout with rollback",
```

```
39 +     "Description": "Implement  
Continuous Integration (CI) with  
Testing",  
40 +     "Url": "Challenge-04.html"  
41     },  
42     {  
43         "Order": 5,  
44         "Follows": 4,  
45 +     "Title": "Challenge 5: Implement  
Continuous Deployment (CD)",  
46 +     "Description": "Implement  
Continuous Deployment (CD)",  
47 +     "Url": "Challenge-05.html"  
48     },  
49     {  
50         "Order": 6,  
51         "Follows": 5,  
52 +     "Title": "Challenge 6: Implement  
a Blue/Green deployment strategy",  
53 +     "Description": "Implement a Blue/Green deployment strategy",  
54 +     "Url": "Challenge-06.html"  
55     },  
56     {  
57         "Order": 7,  
58         "Follows": 6,  
59 +     "Title": "Challenge 7: DevSecOps  
basics - get rid of secrets",  
60 +     "Description": "DevSecOps basics  
- get rid of secrets",  
61 +     "Url": "Challenge-07.html"  
62     },  
63     {  
64         "Order": 8,  
65         "Follows": 7,  
66 +     "Title": "Challenge 8:  
Integrating quality and security checks",  
67 +     "Description": "Integrating  
quality and security checks",  
68 +     "Url": "Challenge-08.html"  
69     },  
70     {  
71         "Order": 9,  
72 +     "Follows": 7,
```

```
73      +         "Title": "Challenge 9: Implement  
74          a Load Testing & Monitoring solution with  
75          alerting",  
76      +         "Description": "Implement a Load  
77          Testing & Monitoring solution with  
78          alerting",  
79      +         "Url": "Challenge-09.html"  
80      +     },  
81      +     {  
82      +         "Order": 10,  
83      +         "Follows": 7,  
84      +         "Title": "Challenge 10: Implement  
85          phased rollout with rollback",  
86      +         "Description": "Implement phased  
87          rollout with rollback",  
88      +         "Url": "Challenge-10.html"  
89      +     }  
90  ]  
91 }  
92 - }
```

16 portal/includes.yaml

```
... @@ -1,8 +1,8 @@
1 - ---
2 - header-includes:
3 - - <script
4 -   src="https://code.jquery.com/jquery-
5 -   3.2.1.min.js" integrity="sha256-
6 -   hwg4gsxgFZhOsEEamOYGBf13FyQuiTwIaQgxVSNg
7 -   t4=" crossorigin="anonymous"></script>
8 - - <link rel="stylesheet"
9 -   href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
10 -   integrity="sha384-
11 -   BVYiSIFeK1dGmJRAkycuHAHRg320mUcww7on3RYd
12 -   g4Va+PmSTsz/K68vbdEjh4u"
13 -   crossorigin="anonymous">
14 - - <link rel="stylesheet"
15 -   href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap-theme.min.css"
16 -   integrity="sha384-
17 -   rHyOn1iRsVXV4nD0JutlnGaslCJuC7uwjduW9SVrL
18 -   vRYooPp2bwYgmgJQIXwl/Sp"
19 -   crossorigin="anonymous">
20 - - <script
```

```
1 + ---  
2 + header-includes:  
3 + - <script  
    src="https://ajax.aspnetcdn.com/ajax/jQuery/jquery-3.6.0.min.js"  
    crossorigin="anonymous"></script>  
  
4 + - <link rel="stylesheet"  
    href="https://ajax.aspnetcdn.com/ajax/bootstrap/3.4.1/css/bootstrap.min.css"  
    crossorigin="anonymous">  
  
5 + - <link rel="stylesheet"  
    href="https://ajax.aspnetcdn.com/ajax/bootstrap/3.4.1/css/bootstrap-theme.min.css"  
    crossorigin="anonymous">  
  
6 + - <script
```

```
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"
integrity="sha384-Tc5IQib027qvyjSMFhjOMaLkfuWVxZxUPnCJA712m
CWNIPG9mGCD8wGNICPD7Txa"
crossorigin="anonymous">></script>
7 - <link rel="stylesheet"
 href="https://opsgilitylabs.blob.core.windows.net/lablayout/labstyles.css"></link>
8 - --- ⊖
```

```
src="https://ajax.aspnetcdn.com/ajax/bootstrap/3.4.1/bootstrap.min.js"
crossorigin="anonymous">></script>
7 + - <link rel="stylesheet"
 href="https://opsgilitylabs.blob.core.windows.net/lablayout/labstyles.css"
crossorigin="anonymous">></link>
8 + ---
```

▼ BIN **-50.9 KB** resources/Attendee\_Takeaway.docx

Binary file not shown.

▼ 74 resources/CHANGELOG.md

```
... @@ -1,6 +1,76 @@
1 - # Changelog for the DevOps OpenHack
2
3 - ## Changelog since v1.0.0
```

```
1 + # DevOps OpenHack - Changelog
2
3 + ## Purpose
4 +
5 + This document aims to highlight
6 + significant changes to the OpenHack over
7 + time.
8 + Add a section with the release
9 + date/version for each significant content
10 + refresh and summarize the fundamental
11 + changes as a bulleted list.
12 +
13 + ## v3.0.0 [Feb 14th, 2022]
14 +
15 + ### General
16 +
17 + Challenges contain GitHub naming
18 + convention, e.g., workflow instead of
19 + pipeline. This is to give the impression
20 + that GitHub is the 1st choice platform
21 + (aligned with the current Microsoft Sales
22 + & Marketing strategy).
23 +
24 + ### Lab environment provisioning / BYOS
25 +
26 + Any resources pre-provisioning and
```

environment pre-configuration have been deprecated. Instead, teams during the OpenHack are responsible for setting the initial setup configuration based on the BYOS deployment guide and scripts. BYOS deployment scripts and guide contains GitHub and Azure DevOps configuration.

```
17 +
18 + ### YAML approach
19 +
20 + The classic UI approach for building
   Azure DevOps Pipelines has been
   deprecated. Instead, YAML is the only
   choice for building GitHub Workflows and
   Azure DevOps Pipelines.
21 +
22 + ### Jenkins
23 +
24 + Jenkins is no longer available for pickup
   as a learning path. Only GitHub and Azure
   DevOps learning paths are available.
25 +
26 + ### Challenge 1: Establish your plan
27 +
28 + Challenge 1 has been enhanced and added a
   hands-on experience for basic task board
   custom configuration.
29 +
30 + ### Challenge 2: Setting up the
      Development Workflow
31 +
32 + Challenge 2 has adjusted to the common
   approach available on both platforms. For
   example, "linked work item enforcement"
   is not currently available on GitHub, so
   the challenge success criteria were
   confusing for attendees and not
   achievable in the standard approach. This
   point has been replaced with "flat
   history" in the PR.
33 +
34 + ### Challenge 3: Deploy Cloud
      Infrastructure with Infrastructure as
      Code (IaC)
35 +
36 + It is a new Challenge to introduce the
```

Infrastructure as Code (IaC) concept. The challenge contains two options for choosing based on the team's learning needs - Terraform and Bicep.

37 +  
38 + **### Challenge 4: Implement Continuous Integration (CI) with Testing**  
39 +  
40 + This is old Challenge 3, a bit adjusted and refreshed without big changes. Added feature to show how to automatically report custom comments to the PR/Issue based on workflow behavior.  
41 +  
42 + **### Challenge 5: Implement Continuous Deployment (CD)**  
43 +  
44 + This is old Challenge 4, slightly adjusted and refreshed without big changes. Added optional extended challenge for GitHub learning path - GitHub Packages as Container Registry instead of Azure Container Registry.  
45 +  
46 + **### Challenge 6: Implement a Blue/Green deployment strategy**  
47 +  
48 + This is old Challenge 5. The concept remains the same, but the approach for health checks has changed from custom code to platform-native solutions.  
49 +  
50 + **### Challenge 7: DevSecOps basics – get rid of secrets**  
51 +  
52 + It is a new Challenge to introduce DevSecOps basics. The goal is to use secret scanning tools to discover hardcoded secrets in the source code. Then, remediate it by implementing secure storage to pull secrets (Key Vault) and next remove them from the source code.  
53 +  
54 + The challenge contains an optional extended challenge to implement a secret rotation strategy for Azure SQL Server.

```
4
5      - Use of a container to provision the
6      infrastructure used during the OpenHack
7      - Helm charts refactored to enable the
8      use of helm package for the API.
9      [@mathieu-benoit]
10     (https://github.com/mathieu-benoit)
```

```
55    +
56    + ### Challenge 8: Integrating quality and
57    security checks
58    +
59    + This is old Challenge 7, slightly
60    adjusted and refreshed without big
61    changes. Load Testing moved to Challenge
62    9.
63    +
64    + ### Challenge 9: Implement a Load Testing
65    & Monitoring solution with alerting
66    +
67    + This is old Challenge 6. There are no
68    significant changes in the Monitoring
69    area. Added Azure Load Test service for
70    the load testing scenario.
71    +
72    +
73    + ## v2.0.0
74    +
75    + - Kubernetes has been deprecated and
76    replaced by App Service for the
77    application hosting platform.
78    +
79    + - Added GitHub as 3rd option (not fully
80    supported yet)
81    +
82    +
83    + ## v1.0.1
84    +
85    + - Use of a container to provision the
86    infrastructure used during the OpenHack
87    - Helm charts refactored to enable the
88    use of helm package for the API.
89    [@mathieu-benoit]
90    (https://github.com/mathieu-benoit)
```

✓ 167  resources/COACHES\_NOTES.md 

... @@ -0,0 +1,167 @@

```
1  + # Notes for Coaches
2  +
3  + This document is a set of references and
```

notes that the coaches of the DevOps OpenHack can use at each stage of the event.

```
4 +
5 + ## Environment setup
6 +
7 + - **Azure Accounts**: The attendees have
   been provided a specific account for the
   event. It does not apply to the BYOS
   delivery model - it's only for Opsgility
   managed delivery. It is highly
   recommended that they use their browser
   InPrivate/Incognito mode.
8 +
9 + - **GitHub Accounts**: The attendees can
   use their own GitHub accounts. If they
   do not want to, they have to setup a new
   one on their own.
10 +
11 + - **GitHub Enterprise Organization**:
   Dedicated GitHub Enterprise Organization
   has been created upfront with all
   enterprise features and an Advanced
   Security license. Coach has to invite
   attendees to the GitHub Enterprise
   Organization as standard members before
   GitHub Lab Environment deployment has
   been triggered.
12 +
13 + - **Azure DevOps Organization**: The
   attendees have to create a new Azure
   DevOps organization dedicated only for
   the OpenHack. The participants will have
   to add the hackers' accounts of their
   team. Each Azure DevOps has a
   subscription limit of 5 free accounts,
   and the accounts shall be set using the
   Basic access level.
14 +
15 + - **Azure DevOps**: You can suggest to
   the teams to rename their accounts from
   Hacker1 etc., to their names to make the
   rest of the event easier. For example,
   when they will create PRs, they will
   know who created them. It does not apply
```

to the BYOS delivery model - it's only for Opsgility managed delivery.

16 +  
17 + **## Coaching Tips and Tricks**  
18 +  
19 + In this section, you will find non-technical related stuff like general structure of the event, Do's and Don'ts, how to coach, etc. Follow this guide:  
[Coaching Tips and Tricks]  
(COACHING\_TIPS\_AND\_TRICKS.md)  
20 +  
21 + **## Challenges flow**  
22 +  
23 + - Challenge 5 is a minimum challenge requirement to achieve Badge/TSI credit.  
24 + - Challenge 5 & 7 have optional additional challenges - it's recommended only for experienced teams where all participants want to learn more advanced scenarios.  
25 + - After Challenge 7, the team can choose the favorite path based on learning needs.  
26 +  
27 + ![DevOps OpenHack - flow]  
(./images/DevOpsOpenHack-flowchart.png)  
28 +  
29 + **## Do not use two-factor auth with the hacker Azure test ID's**  
30 +  
31 + Tell the attendees to click the "skip for 14 days" link after the initial sign-in to the Azure portal or DevOps portal. Two-factor auth can cause unintended behavior between the hacker accounts and MSFT accounts.  
32 +  
33 + **## GitHub**  
34 +  
35 + [GitHub Codespaces]  
(<https://github.com/features/codespaces>) is not mentioned anywhere in the OpenHack content, but feel free to advise your team they can try this feature if they are interested.

```
36 +  
37 + ## Challenge 0 (BYOS)  
38 +  
39 + It might be too difficult for attendees  
+ with limited CLI & scripting experience.  
40 +  
41 + The coach should evaluate the skill  
level of the team and either help with  
the provisioning or do it by themselves  
and share in real-time all the steps  
with attendees.  
42 +  
43 + ## Challenge 1  
44 +  
45 + Before creating a project lifecycle and  
task board, you must deploy resources  
and applications following the steps in  
the **Overview** markdown.  
46 +  
47 + This challenge is about creating a  
project lifecycle and a task board.  
48 +  
49 + Everybody in the team should be engaged,  
and if you see anyone left aside, you  
should facilitate their integration into  
the team.  
50 +  
51 + Coaches, if your team wants to use  
Microsoft Teams, you will need to create  
a team and invite the Opsgility accounts  
to that team. It does not apply to the  
BYOS delivery model - it's only for  
Opsgility managed delivery.**  
52 +  
53 + If your team chooses an app like Teams  
or Slack, you can suggest the plugins  
for integration. It is not part of the  
challenge itself, but nice to have.  
54 +  
55 + - [GitHub & Teams]  
+ (https://teams.github.com/)  
56 + - [GitHub & Slack]  
+ (https://slack.com/apps/A01BP7R4KNY-github)  
57 + - [Azure DevOps & Teams]  
+ (https://marketplace.visualstudio.com/it)
```

```
ems?itemName=ms-vsts.vss-services-teams)
58 + - [Azure Boards & Slack]
      (https://slack.com/apps/AKR9QDD1D-azure-boards)
59 + - [Azure Repos & Slack]
      (https://slack.com/apps/AKPEA4W4Q-azure-repos)
60 + - [Azure Pipelines & Slack]
      (https://slack.com/apps/AFH4Y66N9-azure-pipelines)
61 +
62 + Pair programming is strongly recommended
  because it is more efficient and
  facilitates teamwork and learning.
  [VScode LiveShare]
  (https://docs.microsoft.com/en-us/visualstudio/liveshare/) allows the
  team to pair-program from their personal
  editors.
63 +
64 + ## Challenge 2
65 +
66 + Quick and easy challenge, but can
  present a good opportunity to talk about
  branch restrictions and what's the
  impact on development velocity. Rules
  too strict can significantly slow the
  project down and the tax of PR reviews
  is not negligible. On the other hand
  there are significant benefits to
  protecting the `main` branch from
  unsupervised commits.
67 +
68 + ### ADO
69 +
70 + There's no expectation to do anything
  with code in this challenge.
71 +
72 + The solution here is different from
  GitHub (below), because it allows to
  enforce squash merging on a branch
  level.
73 +
74 + ### GitHub
75 +
76 + Beware that when specifying paths in the
```

`CODEOWNERS` file, all characters matter  
- especially `/` and `\*`.

77 +

78 + - `/apis/\*` means files in just the one folder, `/apis/` means this folder and subfolders.

79 + - `README.md` means all readme files in any folder, `/README.md` means only the root readme.

80 + - `apis/` means any file in any apis directory anywhere in the repository.

81 +

82 + See [examples]  
<https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-code-owners#example-of-a-codeowners-file>) for more details.

83 +

84 + If there are syntax errors in the file, they **\*\*will not be detected\*\*** and the code review rules will not work.

85 +

86 + Observe if the team decides to enforce squash merging on the repository level. The solution is acceptable without it, but encourage them to think about both options.

87 +

88 + **## Challenge 3**

89 +

90 + The Bicep path is more complicated than Terraform. If the team does not have experience in ARM and Infrastructure as Code, then Terraform is highly recommended.

91 +

92 + Be aware of where the participants are running the deployment scripts from i.e desktop or az pipeline make sure there are az login commands executed to ensure proper deployment.

93 +

94 + If the team name is hardcoded, then RESOURCES\_PREFIX is no longer required.

95 +

96 + The web apps might take some time to  
97 spin up especially the java one. If the  
98 pipeline/workflow fails to get the sites  
99 health endpoints during the test step,  
run the following code replacing  
`<ProjectName>` with the name of your  
99 project:  
97 +  
98 + ````shell  
99 + ./smokeTest.ps1 -HostNames  
100 <ProjectName>poi.azurewebsites.net/api/h  
ealthcheck/poi,  
101 <ProjectName>trips.azurewebsites.net/api  
/healthcheck/trips,  
102 <ProjectName>userjava.azurewebsites.net/  
api/healthcheck/user-java,  
103 <ProjectName>userprofile.azurewebsites.n  
et/api/healthcheck/user`  
100 + ````  
101 +  
102 + or use your browser to access the above  
endpoints. Once the websites **return** a  
valid json re-run the pipeline/workflow.  
103 +  
104 + When re-running the Terraform  
deployment, make sure you also delete  
the terraform.tfstate file **in** the Azure  
storage account **in** the state resource  
group.  
105 +  
106 + If there are already resources present  
**in** the target resource groups with the  
same names as the new resources (for  
instance because they were deployed  
previously with Bicep or Azure Portal),  
you'll get an error like this: **\\*Error:**  
A resource with the ID  
"/subscriptions/\_\\*\\*/resourceGroups/dev  
ops09248rg" already exists - to be  
managed via Terraform this resource  
needs to be imported into the State.\_  
The easiest thing is to remove all  
existing resources and deploy again.  
107 +  
108 + ### ADO  
109 +

110 + Add Pipeline permissions to the production environment, grant access permissions to all pipelines in the service connection (See [Manage service connections]  
<https://docs.microsoft.com/en-us/azure/devops/pipelines/library/service-endpoints?view=azure-devops&tabs=yaml>), and get the [Azure Pipelines Terraform Task extension]  
<https://marketplace.visualstudio.com/items?itemName=charleszipp.azure-pipelines-tasks-terraform>).

111 +

112 + You will need to permit the pipeline to access additional resources during the initial run of each pipeline.

113 +

114 + ### GitHub

115 +

116 + If using a Personal Access Token (PAT) to login, make sure it has the `workflow` permissions - otherwise GitHub will refuse a commit which creates Actions.

117 +

118 + All workflows can be triggered manually, which is good for development and testing, but don't forget to validate the branch, PR and path conditions.

119 +

120 + ## Challenge 4

121 +

122 + Encourage the use of a CI pipeline per API source repo. Smaller pipelines will run faster to provide a quicker feedback loop and partition the code builds for added isolation. The build definition can then be part of each api code base.

123 +

124 + Make sure the main branch is not included in the CI pipeline.

125 +

126 + For the part of the challenge that requires adding a comment to the PR for failed builds in ADO version ensure that the Build Service (hacker) has

"contribute to pull requests" set to allow.

127 +  
128 + ! [manage ADO repo permissions]  
  (./images/managrepo.png)  
129 +  
130 + ! [set build service permissions]  
  (./images/repoBuildSvcPerm.png)  
131 +  
132 + ## Challenge 5  
133 +  
134 + When using Azure Container Registry,  
  there are two ways to build the Docker  
  image: on the build agent or using ACR  
  Tasks. Both approaches are feasible,  
  just encourage the team to discuss  
  implications of each of them and make an  
  informed decision.  
135 +  
136 + ## Challenge 7  
137 +  
138 + Provided challenge does not use Managed  
  Identities because of application  
  architecture but having discussion about  
  Managed Identity usage instead of  
  keys/passwords (benefits,  
  implementation, etc.) is a good idea to  
  start with team members.  
139 +  
140 + > \*\*NOTE:\*\* If your team consists only  
  Microsoft employees, and they have  
  chosen Azure DevOps, then the preferred  
  way is Cred Scan.  
141 +  
142 + ## Challenge 8  
143 +  
144 + If the Team decided to use 3rd party  
  GitHub Apps like WhiteSource Bold,  
  someone with GitHub Organization owner  
  right has to approve that request. Each  
  Coach has admin permission and can do  
  that for Team.  
145 +  
146 + ## Challenge 9  
147 +  
148 + A sample JMX file has been provided in  
  the teams repo resources section. This

```

file is already parameterized and
accepts uris for each API.

149 +
150 + Teams can create a new Azure Load
    Testing Resource and create a test. In
    there, please ensure they add parameters
    with each of the endpoints for each api.
    This should be the domain only as the
    path is specified in the test.

151 +
152 + The names of the parameters can be found
    in the MyDrivingTest.jmx file in the
    User Defined Variables section.

153 +
154 + For example the User Java parameter name
    is UserJavaURI.

155 +
156 + ````xml
157 + <Arguments guiclass="ArgumentsPanel"
    testclass="Arguments" testname="User
    Defined Variables" enabled="true">
158 +     <collectionProp
        name="Arguments.arguments">
159 +         <elementProp
            name="UserJavaUri"
            elementType="Argument">
160 +             <stringProp
                name="Argument.name">UserJavaUri</string
                Prop>
161 +             <stringProp
                name="Argument.value">${__BeanShell(Syst
                em.getenv("UserJavaUri"))}</stringProp>
162 +             <stringProp
                name="Argument.metadata">=</stringProp>
163 +         </elementProp>
164 +         ...
165 +     </collectionProp>
166 + </Arguments>
167 + ````
```

▽ 94 resources/COACHING\_TIPS\_AND\_TRICKS.md □

... @@ -0,0 +1,94 @@

```

1 + # Coaching Tips and Tricks
2 +
3 + In this section, you will find non-
```

technical related stuff like general structure of the event, Do's and Don'ts, how to coach, etc.

- 4 +
- 5 + **## Agenda for days of the OpenHack**
- 6 +
- 7 + OpenHacks vary in the number of days allocated to each event. However, they all follow a similar pattern in terms of coaching:
- 8 +
- 9 + - Start of the Day - standup. The purpose is to update the coaches with any last-minute changes or updates that might impact their teams.
- 10 + - OpenHack - working with teams.
- 11 + - End of the Day - retrospective. The purpose of this is to gather feedback from coaches and have a check-in to measure each team's progress.
- 12 + > Note: This is not a competition, and teams are not measured against one another. The goal of the check-in is to see if any team is falling behind and needs additional attention or resources.
- 13 +
- 14 + **## The role of a Coach**
- 15 +
- 16 + The coach at the OpenHack is a guide/enabler, not a teacher. Avoid lecturing students on each challenge and pertaining topics, unless they ask for information and have also read the references at the bottom of each challenge `\*`.
- 17 +
- 18 + Ask pointed questions aimed at guiding the participants towards finding a solution. In some cases, this can cause frustration, which is understandable.
- 19 +
- 20 + As a coach, you should also manage frustration levels on a team by team basis. If your team is truly stuck and have tried to find a solution on their own, you can provide a stronger hint and

adjust as needed. We want them to learn, but also to walk away with a positive experience.

- 21 +
- 22 + **``\*``** The exception to this being around technology what is used in the challenge, but it is not the main theme of the OpenHack. If your team does not have a strong background in it, it's ok to give them a short introduction and teach them about basics. Also, if your team does require this, please include directly related stuff to your descriptions. This is an excellent opportunity to grab a whiteboard! If you don't feel confident explaining that technology, please find another coach to assist. The lead coach can help you with this as well.
- 23 +
- 24 + **## Do's and Don'ts**
- 25 +
- 26 + This section will cover the Do's and Don'ts of coaching.
- 27 +
- 28 + **### Do's**
- 29 +
- 30 + - Give suggestions and guide teams to find solutions on their own.
- 31 + - Encourage teamwork and participation by everyone at your table.
- 32 + - Be engaged and answer questions that may come up. If you don't know the answer, ask a fellow coach or the lead coach.
- 33 + - Pay attention to personalities in your group. It's key for everyone to be respectful and work as a team, if any issues arise, bring them up to the lead coach and PM lead immediately.
- 34 + - Identify high skilled individuals at your table and ask them to help you ensure everyone learns.
- 35 + - Encourage your team to work on a shared monitor (each table should have one.) Also, ensure that they are rotating who is "driving".

- + - Encourage pair programming / mob programming.
- + - Recommend that teams use the visual (classic) pipeline editor instead of YAML. While YAML is preferred for production builds, the classic editor provides an easier onramp for folks that are learning.
- + - Encourage the team to use/read the references at the end of each challenge. Often times if they are struggling, it's because they didn't take time to read the references. It's OK to *\_not\_* jump right into a solution.
- +  
+ **### Don'ts**
- +  
+ - Don't give out direct answers. Instead, guide people to finding the solutions themselves.
- + - Don't allow one person on the team to do all the work.
- + - Don't pass challenges unless everyone on the team demonstrates an understanding of the challenge solution. A good way to approach this is to let the team know that they either all pass together, or no one passes a challenge.
- + - Don't check your email or do work that is not part of the OpenHack. The goal of a coach is to provide a great experience by being engaged and available to your team.
- + - Don't leave for long periods. While it's ok to take a break and socialize with other coaches (it's even encouraged!), don't be away for too long. If you need to be away, please let the lead coach know so arrangements can be made.
- + - Don't spend a lot of time teaching something that is not directly related to the challenges. If attendees ask about something off-topic, give it a few minutes and then move them back on track. This is a good time to explore, and we

want them to, but you need to make sure they stay focused on the DevSecOps journey.

- 48 +
- 49 + **## Team Building**
- 50 +
- 51 + It's essential to have the folks at your table work as a team. While they could be from the same company/organization, in many cases, they will not be and be a group meeting for the first time.
- 52 +
- 53 + In order to bring everyone to the table, we've found introductions at the start of the day to be key. A recommended approach is to introduce yourself, then have each team member introduce themselves and provide the following information like:
- 54 +
- 55 + - Name and Role
- 56 + - Experience with DevOps and DevSecOps
- 57 + - One interesting fact about themselves (non-work related)
- 58 +
- 59 + It's also important to make it clear that they will be passed on challenges as a team if one person doesn't understand the solution or have issues, then the team does not pass that challenge.
- 60 +
- 61 + Moreover, it helps to be explicit about recommending pair programming/mob programming and to ensure everyone gets a chance to drive/share.
- 62 +
- 63 + Another important note for the team is to suggest they share findings. If they find a solution that's interesting, share it on the screen and explain it to the group. Often one person will work through lunch and solve an issue. It's a good idea when the team reconvenes to ask that person to explain to the group their solution.
- 64 +
- 65 + Each team should have a dedicated chat

mechanism. For internal teams, Microsoft Teams should be used. For teams with customers, any chat system will work (Slack, Teams, discord, etc.) Allow the team to decide what to use and also to create the channel/team. This gives the team more ownership over the process.

- 66 +
- 67 + **## Working through Challenges**
- 68 +
- 69 + The bulk of the event will be your team working on challenges for the OpenHack. The goal is for them to add security into DevOps workflow to increase overall security, minimize risk, and meet company security compliance.
- 70 +
- 71 + A good workflow for each challenge includes the following:
- 72 +
- 73 + - Ask the team to read the challenge as a group.
- 74 + - Recommend they discuss their strategy for solving the challenge before starting to work.
- 75 + - Ask them to read the references as that has information directly related to each challenge.
- 76 + - Before starting to work, per a good DevOps culture, the team should create work items to represent the work they will perform. As a coach, this is a good check-in opportunity for you.
- 77 + - Each challenge should be treated as a sprint with a sprint planning to start, then retro at the end.
- 78 + - As the team completes a challenge, they may ask for questions or clarifications from you.
- 79 + - When they are finished, go through the completion criteria at the bottom of each challenge to ensure they have met the goals.
- 80 + - After completing, have a quick retrospective to talk about the challenge. Specifically, ask for any

feedback to improve their experience in upcoming challenges. This can be technical, but also workflow related.

81 + - If a team member mentions they didn't do enough in this challenge and others did all the work, that's a good indicator that you need to ensure they split up the work in the next challenge.

82 +

83 + **## How to Get Help**

84 +

85 + As you arrive at your table, get to know the coaches near you. They can assist you when stuck and will be a valuable asset. However, if you need extra help and attention, please do the following:

86 +

87 + - Utilize the Teams channel (dedicated per OpenHack) and ask your question there.

88 + - Find the lead coach or PM lead and ask them for assistance. They will help direct you towards experienced coaches who can assist.

89 +

90 + **## Submit Feedback**

91 +

92 + One of the coach responsibilities is to collect feedback from the team. Below you can find places where you can put your feedback:

93 +

94 + - OpenHack content-related feedback send to your Lead Coach. He/She is responsible for collect, validate, aggregate, and finally send to the OpenHack development team.

✓ BIN **-21.2 KB** resources/Change-Log-DevOps-OpenHack.docx 

Binary file not shown.

✓ 2  ...ons\_Bring Your Own Azure Subscriptions.md → resources/DEPLOYMENT.md 

... @@ -1 +1 @@

```
1 - # Deployment instructions are located at  
[aka.ms/OpenHackBYOS]  
(https://aka.ms/OpenHackBYOS)
```

```
1 + # Deployment instructions are located at  
[aka.ms/OpenHackBYOS]  
(https://aka.ms/OpenHackBYOS)
```

✓ BIN -55.4 KB resources/DevOps 2.0 - OH One Pager.docx 

Binary file not shown.

✓ BIN +118 KB resources/DevOps OH 3.0 - One Pager.docx 

Binary file not shown.

✓ 1 ■■■■■ resources/DevOpsOpenHack-flowchart.drawio 

... @@ -0,0 +1 @@

```
1 + <mxfile host="Electron" modified="2022-  
01-26T19:52:06.199Z" agent="5.0 (Windows  
NT 10.0; Win64; x64) AppleWebKit/537.36  
(KHTML, like Gecko) draw.io/16.4.0  
Chrome/96.0.4664.110 Electron/16.0.7  
Safari/537.36"  
etag="uTPsnKH8bBsBrnKbCiev"  
version="16.4.0" type="device"><diagram  
name="flowchart" id="edf60f1a-56cd-e834-  
aa8a-  
f176f3a09ee4">5VtJd+I4EP41HMPzDhw7k05Zeiy  
9j8N0TnnCFlgT23LLMkv/+pGwvCEgYhedS7DKkiVV  
1Vf1qex07GG8/EJAGv6FAxh1LCNYduxRx7JM07DYD  
5esCknf7RWCUGUB6FQLxughFEJDSHMUwKzVkwIcUZ  
S2hT50EujiTlgwQghftblMctWdNwQxKgrEPILn6Lwp  
oW07CqOW/QTQLabVhcWcc/LcZwXki5ktwAos7MSgf  
I7pmIQjwoiGynzr2kGBMi6t40YQRV2upsWlc5x13q  
yUTmFCVAX+8TV5f/xyB60n16+jN+Gf6MLQFTGG4OY  
hyoQuxWroqlQMDpivRxISGeIYT9pRa+rhwAOTzGKx  
V9/mKccqEjhP+ByldCcODnGImCmkcibuVbvj4jBL8  
VtmB35a3Knaf4Zz4cM/+HOFMgMwg3acH8UC+18YMQ  
pNFII4hJSvWgcAIUDRv+w0Q7jer+tV2YBfCFAeYxZ  
GsMmZboJJpZFW11ZryfvFyx1HbnUZ44YfsMd2MP+y  
V91mEiMJxCtZKXLBuTWPsvf0cEggXe3U17to9txgi  
goMjALGokVbhKwlygzDMupF1TVu8v5fSlM7/v9bZWx  
l+uu2Gxx5fnccfyIraLxwlhVzN+Fca0wquYK0ZChH  
B6ynwZ/sR8vscUEsRWCE1T/q0WbkVBw1BTtIR10jk  
32urskPHakDG3Yca6JmZcT1fMDHGEuWUDOAX521HO  
iiVbEUuWXhmkXHfdXsMQRBFM2PIsg01rdLtd9teRz  
PgOGq7v/Vszx1W93+rr6v3n8XJLNWNYPa283JIZRp
```

```
ktshQkZb5oer5bdmATNvvoDwP75jCwDUndL+zYdks  
kbDfFCUjoqSJhoBUsbFcyzd/414pRqpaxHa0sU667  
YZnFE8Y7YuAzn1gFoIMnFLE1dyxP0tGIyGOJ3n2P  
k/dEpJ2MaQLhypHNVT1LxaqZK0fgYbz+z/TMV19bz  
ZeeN+uWzzHSzG2aK1Eq43LXXY9AV2Dm6FLDP2GET/  
P1s71GW2Xcgcb1KsVIyqveUTIWDV6JbyDpn6PF67  
jMYuiifWrljt8YSYMJC8c81RQMVDRDAGERIA0cEy/  
JrKHMhX5Ix50s1JPti4zkZwcG/NYxxtC0CXPczaqu  
VQR6/CkL3NXgWLD9C8ZvANUc0U3o+c19gfmbvSBxC  
hWdKxP3EoQZ6G6/ubz2meDmKUoDiP1zv+kSMCeFEP  
43v1QwTnPJhvOFVu4iut1P2CmG0/+HvUurb1vMRaJ  
4D5F9ciyrL8SB0qdt5RJkwJ9mGmwK3qFz3POY1QAj  
cxsfFhYFQobRXC6N28fT8Hc/q0pmPNBS4a0asmwzIu  
aRFlnf8nQ04lx1A4vws0VKcgm0HrGzSmIthXFC4PD  
VQWHxpVGR66vNMHRux86vomF/s2x40pXVrwaHFSLW  
o5e5UZXZuTa1Bs1sZir1xnK2VYQkyn506VJtVcn6h  
VLbaLgQDUKXo4uyzWhne+yQJV1F0pE53ndpU35aGD  
ePF/J3015rXkm+xgszlU94rh6sThPZhqv5cw7spxq  
+eCMhtSLF3jWbkPad2hIUoSta1jS0+tDJXdb1QHjj  
0c3nq5YglSXhn2YAIjkjnmlHXMM0bv561Pubg0BcvF  
bu6cXeXTX23ixJD06e+Um4MvVr2hejfukrguT6q0  
B6rsAtY8GjoQXXCL6vZ6AtV4ad+q5eK0c6gRIqn55  
oBskDzivCUj27x6SZl8/SMovJq4JyQ0/47kLSJYU8  
d4gWa77AEjyYsKdY3KzQKIDJmVS/5QEMk6jCKUzvc  
vub1/wdQlr1v8PWnxrVv+/rf30Pw==</diagram>  
</mxfile> ⊖
```

▼ BIN -4.61 MB (63%) resources/DevOps\_Tech Scenario\_Attendee\_Coach\_Deck.pptx 

Binary file not shown.

▼ 1  resources/MyDriving-architecture.drawio 

... @@ -0,0 +1 @@

```
+ <mxfile host="Electron" modified="2021-  
10-04T07:18:51.964Z" agent="5.0 (Windows  
NT 10.0; Win64; x64) AppleWebKit/537.36  
(KHTML, like Gecko) draw.io/15.4.0  
Chrome/91.0.4472.164 Electron/13.5.0  
Safari/537.36"  
etag="zuNVU1dFDXgfNs9wxUb7"  
version="15.4.0" type="device"><diagram  
id="08xRjSSHFDWdbHYYJyld" name="Page-  
1">7VrbcqM4EP0aV+0+TAoJg+ExtsfZTGvrM+vs5c  
21gAyagEQJ+TZfp8IIDIGkjoMxM3EeYtSo7abP0aj
```

```
V0APjchPDYOT/SV0U9HTN3fTApKfrfc2yxUdi2aYW  
y9BSg8ewKyftDTP8HwWe0rrELopLEzm1AcdR2ehQQ  
pDDSzbgIF2Xpy1oUP7VChpIMcwcGKjW/7DLfWnta9  
r+xB8Iez6vXF8Is8nSEPVQpeuCCXzugTGj1KdH4Wa  
MgiR5WV5Sv+kzz/PAGCL8E1cv1td/nx7/HxAy/Jvz  
b+HTrfbwKUvzCgZLecUyWr7NUrD2MUezCDrJeC1g7  
oGRz8NAjPrcEEJ17j1LTke04AyYSCUCPUI0Dv4iH  
IHHASVCTFn9C1PMBAWF8Y+cqVHejrz6e1A04bTqbj  
QkZoEmZcVYhxtCiaZlBtEQ8TZVkyRZwEAqYtkaN+S  
OVnv8R5kIPoFqA0gjVByzMu/ew+DOJBIVAUVBZTrK  
BKGGWIrLGDQtfsAEvHx2x0my83vCmQwj1IxLPAmyW  
IRr4hiwncRG60eMREWGGCPCEOAFsJnhModJhRgpX2  
CQ09cVYAfxx/4fc1QcrUwiuZxG14shiLeuYx2nsQa  
X8UrL/tOGwCSTJDw4p7GmG0ahMDS9I4S+LBQ4V31f  
IhdN3HOJ1zL2PMTDTBCtyqMACojzIFKiMzWOB90hQ  
//iFx/gSvYPEKOSBZizWNFD/v7ETIrmjVqNNsqQkB  
B6IE1S9RHhadfgidfa88Fz0CB5/6v2w8LTlk7YHbm  
cIZau9sFnbtMPvnRadvv14hIlfUzHIOKzpG18TdF  
XCaGFHGfepRIhZ0SiMJ2jf+e+VZCAZecvrmsrCkLbV  
vT1HpSpHuEiHudbAf23sIyxUkudhE+C11M18xBLyR  
omM7jkHmIv75EJL16kQgMBZDjVXn/UQerdL1PuL8n  
kNiZVBhkVMrUNFLpVmFHHsc7CDPsLGFeI8FbCHV6w  
gyaJs77gPWBdZGYDW6Bat9gbURWPV0wareg80ty/  
AKE28y6kRp5UIOH2G8q6tmX+/mEzk+UV1lmdqVZpc  
WRqNuV1LT57F01uZRpdZSoTSd2jYAio5V3RR7dcJv  
AU1rp7DT1VPWufRWX05Z1XrcHLZcTamdha7cn59nV  
Zvl96F86diyq3YkLrAeA2vHimS113GB9RhYG98svw  
tWdeszw+FS/DLtRpVKEZMhJog1xdQ4G8xvScwh0V2  
vyjLKa6Ne04Ufdq6ynWMrNVXWee6kAiHjFV3tbAV1  
FZTnBDCosfPgY1ISY00D0RfEfLxY7UNLMa1Taj1gR  
3sCkeqviRS7wh1zcX3TG0G3KFN1A0LM35bIWn7Zzq  
K4s61RowFeadW/UyGid7eIPVZwL/QaDtH58aLMxNZ  
+ZVS/QLs87ab9e6WXCfgVgu8aXyjVM8bs/KYIu/G  
tMWbn+IxxU/Em5YaMma//G6BYbfLG3DAG30/YIF5X  
m6190y0ek8Cwsv3JJVbyWtEK4zWH/aFiIFeqS80dZ  
NpNdO1F8P967gppvuXmsHnHw==</diagram>  
</mxfile> 
```

✓ 55 resources/Notes-for-Jenkins-Coaches.md

### Load diff

This file was deleted.

⌄ 398 ██████████ resources/Notes-for-coaches.md ⌄

Load diff

This file was deleted.

⌄ BIN -74.5 KB resources/Portal diagrams.pptx ⌄

Binary file not shown.

⌄ 1 ██████████ resources/deployment/README.md ⌄

Load diff

This file was deleted.

⌄ BIN +56.5 KB resources/images/DevOpsOpenHack-flowchart.png ⌄



⌄ BIN +139 KB resources/images/managrepo.png ⌄



⌄ BIN +106 KB resources/images/reposbuildsvcpem.png ⌄



▼ 34 ██████████ resources/pipelines/Build/POI/zddDemo-POI-Build.yaml 

**Load diff**

This file was deleted.

▼ 29 ██████████ resources/pipelines/Build/POI/zddDemo-POI-CI.yaml 

**Load diff**

This file was deleted.

▼ 56 ██████████ resources/pipelines/Build/POI/zddDemo-POI-Int-Test.yaml 

**Load diff**

This file was deleted.

▼ 27 ██████████ resources/pipelines/Build/Trips/zddDemo-Trips-Build.yaml 

**Load diff**

This file was deleted.

▼ 16 ██████████ resources/pipelines/Build/Trips/zddDemo-Trips-CI.yaml 

### Load diff

This file was deleted.

▼ 94 ██████████ resources/pipelines/Build/Trips/zddDemo-Trips-Int-Test.yaml □

### Load diff

This file was deleted.

▼ 11 ██████████ resources/pipelines/Build/User-Java/zddDemo-UserJava-Build.yaml □

### Load diff

This file was deleted.

▼ 27 ██████████ resources/pipelines/Build/User-Java/zddDemo-UserJava-CI.yaml □

### Load diff

This file was deleted.

▼ 27 ██████████ resources/pipelines/Build/User/zddDemo-User-Build.yaml □

### Load diff

This file was deleted.

✓ 30 ██████ resources/pipelines/Build/User/zddDemo-User-CI.yaml 

**Load diff**

This file was deleted.

✓ BIN **-307 KB** resources/pipelines/Release/POI/Artifacts.png 

Deleted file not rendered

✓ BIN **-370 KB** resources/pipelines/Release/POI/Deploy-staging-AgentStep.png 

Deleted file not rendered

✓ BIN **-363 KB** resources/pipelines/Release/POI/Deploy-staging-Step1.png 

Deleted file not rendered

✓ BIN **-398 KB** resources/pipelines/Release/POI/Deploy-staging-Step2.png 

Deleted file not rendered

✓ BIN **-378 KB** resources/pipelines/Release/POI/Deploy-staging-Step3.png 

Deleted file not rendered

✓ BIN **-368 KB** resources/pipelines/Release/POI/RollBack-AgentStep.png 

Deleted file not rendered

✓ BIN **-351 KB** resources/pipelines/Release/POI/RollBack-Step1.png 

Deleted file not rendered

✓ BIN **-377 KB** resources/pipelines/Release/POI/RollBack-Step2.png 

Deleted file not rendered

▼ 18 ██████ resources/pipelines/Release/POI/SetVariableGroup-script.sh 

**Load diff**

This file was deleted.

▼ BIN **-371 KB** resources/pipelines/Release/POI/Upgrade-Stating-to-Prod-AgentStep.png 

Deleted file not rendered

▼ BIN **-352 KB** resources/pipelines/Release/POI/Upgrade-Stating-to-Prod-Step1.png 

Deleted file not rendered

▼ BIN **-377 KB** resources/pipelines/Release/POI/Upgrade-Stating-to-Prod-Step2.png 

Deleted file not rendered

▼ BIN **-414 KB** resources/pipelines/Release/POI/Upgrade-Stating-to-Prod-Step3.png 

Deleted file not rendered

▼ BIN **-379 KB** resources/pipelines/Release/POI/Upgrade-Stating-to-Prod-Step4.png 

Deleted file not rendered

▼ BIN **-342 KB** resources/pipelines/Release/POI/Verification-AgentStep.png 

Deleted file not rendered

▼ BIN **-356 KB** resources/pipelines/Release/POI/Verification-ManualStep.png 

Deleted file not rendered

▼ 21 ██████ resources/pipelines/Release/POI/VerifyProductionSlot-script.sh 

## Load diff

This file was deleted.

✓ BIN **-312 KB** resources/pipelines/Release/Trips/Artifacts.png 

Deleted file not rendered

✓ BIN **-373 KB** resources/pipelines/Release/Trips/Deploy-staging-AgentStep.png 

Deleted file not rendered

✓ BIN **-355 KB** resources/pipelines/Release/Trips/Deploy-staging-Step1.png 

Deleted file not rendered

✓ BIN **-403 KB** resources/pipelines/Release/Trips/Deploy-staging-Step2.png 

Deleted file not rendered

✓ BIN **-382 KB** resources/pipelines/Release/Trips/Deploy-staging-Step3.png 

Deleted file not rendered

✓ BIN **-372 KB** resources/pipelines/Release/Trips/RollBack-AgentStep.png 

Deleted file not rendered

✓ BIN **-354 KB** resources/pipelines/Release/Trips/RollBack-Step1.png 

Deleted file not rendered

✓ BIN **-380 KB** resources/pipelines/Release/Trips/RollBack-Step2.png 

Deleted file not rendered

✓ 18 ██████ resources/pipelines/Release/Trips/SetVariableGroup-script.sh 

## Load diff

This file was deleted.

✓ BIN **-375 KB** resources/pipelines/Release/Trips/Upgrade-Stating-to-Prod-AgentStep.png 

Deleted file not rendered

✓ BIN **-355 KB** resources/pipelines/Release/Trips/Upgrade-Stating-to-Prod-Step1.png 

Deleted file not rendered

✓ BIN **-380 KB** resources/pipelines/Release/Trips/Upgrade-Stating-to-Prod-Step2.png 

Deleted file not rendered

✓ BIN **-421 KB** resources/pipelines/Release/Trips/Upgrade-Stating-to-Prod-Step3.png 

Deleted file not rendered

✓ BIN **-380 KB** resources/pipelines/Release/Trips/Upgrade-Stating-to-Prod-Step4.png 

Deleted file not rendered

✓ BIN **-345 KB** resources/pipelines/Release/Trips/Verification-AgentStep.png 

Deleted file not rendered

✓ BIN **-359 KB** resources/pipelines/Release/Trips/Verification-ManualStep.png 

Deleted file not rendered

✓ 21 ██████ resources/pipelines/Release/Trips/VerifyProductionSlot-script.sh 

## Load diff

This file was deleted.

✓ BIN -315 KB resources/pipelines/Release/User-Java/Artifacts.png 

Deleted file not rendered

✓ BIN -356 KB resources/pipelines/Release/User-Java/Deploy-staging-Step1.png 

Deleted file not rendered

✓ BIN -405 KB resources/pipelines/Release/User-Java/Deploy-staging-Step2.png 

Deleted file not rendered

✓ BIN -385 KB resources/pipelines/Release/User-Java/Deploy-staging-Step3.png 

Deleted file not rendered

✓ BIN -374 KB resources/pipelines/Release/User-Java/RollBack-AgentStep.png 

Deleted file not rendered

✓ BIN -356 KB resources/pipelines/Release/User-Java/RollBack-Step1.png 

Deleted file not rendered

✓ BIN -383 KB resources/pipelines/Release/User-Java/RollBack-Step2.png 

Deleted file not rendered

✓ 18 ██████ resources/pipelines/Release/User-Java/SetVariableGroup-script.sh 

[Load diff](#)

This file was deleted.

✓ BIN -377 KB resources/pipelines/Release/User-Java/Upgrade-Stating-to-Prod-AgentStep.png 

Deleted file not rendered

✓ BIN -356 KB resources/pipelines/Release/User-Java/Upgrade-Stating-to-Prod-Step1.png 

Deleted file not rendered

✓ BIN -383 KB resources/pipelines/Release/User-Java/Upgrade-Stating-to-Prod-Step2.png 

Deleted file not rendered

✓ BIN -420 KB resources/pipelines/Release/User-Java/Upgrade-Stating-to-Prod-Step3.png 

Deleted file not rendered

✓ BIN -382 KB resources/pipelines/Release/User-Java/Upgrade-Stating-to-Prod-Step4.png 

Deleted file not rendered

✓ BIN -346 KB resources/pipelines/Release/User-Java/Verification-AgentStep.png 

Deleted file not rendered

✓ BIN -361 KB resources/pipelines/Release/User-Java/Verification-ManualStep.png 

Deleted file not rendered

✓ 21 ██████ resources/pipelines/Release/User-Java/VerifyProductionSlot-script.sh 

### Load diff

This file was deleted.

✓ BIN -312 KB resources/pipelines/Release/User/Artifacts.png 

Deleted file not rendered

✓ BIN -374 KB resources/pipelines/Release/User/Deploy-staging-AgentStep.png 

Deleted file not rendered

✓ BIN -358 KB resources/pipelines/Release/User/Deploy-staging-Step1.png 

Deleted file not rendered

✓ BIN -404 KB resources/pipelines/Release/User/Deploy-staging-Step2.png 

Deleted file not rendered

✓ BIN -383 KB resources/pipelines/Release/User/Deploy-staging-Step3.png 

Deleted file not rendered

✓ BIN -373 KB resources/pipelines/Release/User/RollBack-AgentStep.png 

Deleted file not rendered

✓ BIN -355 KB resources/pipelines/Release/User/RollBack-Step1.png 

Deleted file not rendered

✓ BIN -382 KB resources/pipelines/Release/User/RollBack-Step2.png 

Deleted file not rendered

✓ 18 ██████ resources/pipelines/Release/User/SetVariableGroup-script.sh 

[Load diff](#)

This file was deleted.

✓ BIN -375 KB resources/pipelines/Release/User/Upgrade-Stating-to-Prod-AgentStep.png 

Deleted file not rendered

✓ BIN -356 KB resources/pipelines/Release/User/Upgrade-Stating-to-Prod-Step1.png 

Deleted file not rendered

✓ BIN -382 KB resources/pipelines/Release/User/Upgrade-Stating-to-Prod-Step2.png 

Deleted file not rendered

✓ BIN -419 KB resources/pipelines/Release/User/Upgrade-Stating-to-Prod-Step3.png 

Deleted file not rendered

✓ BIN -381 KB resources/pipelines/Release/User/Upgrade-Stating-to-Prod-Step4.png 

Deleted file not rendered

✓ BIN -345 KB resources/pipelines/Release/User/Verification-AgentStep.png 

Deleted file not rendered

✓ BIN -361 KB resources/pipelines/Release/User/Verification-ManualStep.png 

Deleted file not rendered

✓ 20 ██████ resources/pipelines/Release/User/VerifyProductionSlot-script.sh 

### Load diff

This file was deleted.

✓ 5 ██████ resources/solutions/README.md 

... @@ -0,0 +1,5 @@

```
1 + # Solutions
2 +
3 + For Azure DevOps go to: [Azure DevOps]
4 + (ado/README.md)
5 + For GitHub go to: [GitHub](gh/README.md)
```

✓ 12 ██████ resources/solutions/ado/README.md 

... @@ -0,0 +1,12 @@

```
1 + # Azure DevOps
2 +
3 + - [Challenge-1](ch1)
4 + - [Challenge-2](ch2)
5 + - [Challenge-3](ch3)
6 + - [Challenge-4](ch4)
7 + - [Challenge-5](ch5)
8 + - [Challenge-6](ch6)
9 + - [Challenge-7](ch7)
10 + - [Challenge-8](ch8)
11 + - [Challenge-9](ch9)
12 + - [Challenge-10](ch10)
```

⌄ 21 resources/solutions/ado/ch1/README.md 

... @@ -0,0 +1,21 @@

```
1 + # Challenge 1
2 +
3 + ## Board
4 +
5 + ! [Board](ado-ch1-board.png)
6 +
7 + ## Board Details
8 +
9 + ! [Part 1](ado-ch1-board01.png)
10 +
11 + ! [Part 2](ado-ch1-board02.png)
12 +
13 + ! [Part 3](ado-ch1-board03.png)
14 +
15 + ! [Part 4](ado-ch1-board04.png)
16 +
17 + ! [Part 5](ado-ch1-board05.png)
18 +
19 + ! [Part 6](ado-ch1-board06.png)
20 +
21 + ! [Part 7](ado-ch1-board07.png)
```

⌄ BIN +95 KB resources/solutions/ado/ch1/ado-ch1-board.png 

✓ BIN +124 KB resources/solutions/ado/ch1/ado-ch1-board01.png 

✓ BIN +144 KB resources/solutions/ado/ch1/ado-ch1-board02.png 

✓ BIN +144 KB resources/solutions/ado/ch1/ado-ch1-board03.png 

✓ BIN +144 KB resources/solutions/ado/ch1/ado-ch1-board04.png 

⌄ BIN +145 KB resources/solutions/ado/ch1/ado-ch1-board05.png 

⌄ BIN +146 KB resources/solutions/ado/ch1/ado-ch1-board06.png 

▽ BIN +121 KB resources/solutions/ado/ch1/ado-ch1-board07.png 

▽ 38  resources/solutions/ado/ch2/README.md 

... @@ -0,0 +1,38 @@

1 + # Challenge 2  
2 +  
3 + This challenge expects participants to

use the Azure DevOps user interface and doesn't require touching any code.

4 +  
5 + **## Mandatory reviews for `main` and prevent direct pushes**  
6 +  
7 + 1. No direct push to `main`  
8 + 1. At least two reviewers  
9 + 1. Require code owner's review  
10 + 1. Commit history not visible in `main` after merging  
11 +  
12 + Go to the project in ADO and under **\*\*Repos\*\*** select **\*\*Branches\*\***.  
13 +  
14 + ![[ADO protection rules]](ado-ch2-protection-rules1.png)  
15 +  
16 + Hover over the `main` (or `master` for older repos) branch and click the three dots on the right hand side. Select **\*\*Branch policies\*\***.  
17 +  
18 + ![[ADO protection rules]](ado-ch2-protection-rules2.png)  
19 +  
20 + Configure reviewers and merge types according to the challenge goals:  
21 +  
22 + ![[ADO protection rules]](ado-ch2-protection-rules3.png)  
23 +  
24 + **## Code owners per service**  
25 +  
26 + 1. Define code ownership per API and IaC  
27 +  
28 + Code owners are defined on the same screen as branch policies, in the **\*\*Automatically included reviewers\*\*** section.  
29 +  
30 + Click the "+" button on the right hand side and create policies for individual APIs and the IaC folder.  
31 +  
32 + ![[ADO protection rules]](ado-ch2-

```
    protection-rules4.png)
33 + 
34 + Multiple paths can be combined in one
rule:
35 +
36 + ````text
37 + /apis/poi/*;/apis/userprofile/*;/iac/*
38 + ````
```

▽ BIN +29.5 KB resources/solutions/ado/ch2/ado-ch2-protection-rules1.png 

▽ BIN +12.8 KB resources/solutions/ado/ch2/ado-ch2-protection-rules2.png 

▽ BIN +58.5 KB resources/solutions/ado/ch2/ado-ch2-protection-rules3.png 

✓ BIN +16.5 KB resources/solutions/ado/ch2/ado-ch2-protection-rules4.png 

⌄ 89 resources/solutions/ado/ch3/.azure-pipelines/pipeline.bicep.deploy.old.yml

... @@ -0,0 +1,89 @@

```
1 + trigger:
2 +   branches:
3 +     include:
4 +       - main
5 +   paths:
6 +     include:
7 +       - iac/bicep
8 +
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - name: ServiceConnectionName
15 +     value: AzureServiceConnection
```

```
16 +   - name: workDir
17 +     value:
18 +       "$(System.DefaultWorkingDirectory)/iac/bi
19 +       cep"
20 +   stages:
21 +     - stage: Preview
22 +       jobs:
23 +         - job: PreviewChanges
24 +           displayName: Preview
25 +             infrastructure changes
26 +             steps:
27 +               - task: AzureCLI@2
28 +                 name: RunWhatIf
29 +                 displayName: Run what-if
30 +                 inputs:
31 +                   azureSubscription:
32 +                     $(ServiceConnectionName)
33 +                     scriptType: "bash"
34 +                     scriptLocation:
35 +                       "inlineScript"
36 +                         inlineScript: |
37 +                           az deployment sub what-if
38 +                           \
39 +                           --name $(Build.BuildId)
40 +                           \
41 +                           --template-file
42 +                           main.bicep \
43 +                           --location $(LOCATION)
44 +                           \
45 +                           --parameters
46 +                           resourcesPrefix=$(RESOURCES_PREFIX)
47 +                           workingDirectory:
48 +                             $(workDir)
49 +
50 +   - stage: Deploy
51 +     jobs:
52 +       - deployment: DeployInfra
53 +         displayName: Deploy
54 +           infrastructure
55 +             environment: production
56 +             strategy:
57 +               runOnce:
58 +                 deploy:
59 +                   steps:
60 +                     - checkout: self
61 +                     - task: AzureCLI@2
```

```
50 +           name: Deployment
51 +           displayName: Run
52 +           deployment
53 +           inputs:
54 +           azureSubscription:
55 +             $(ServiceConnectionName)
56 +             scriptType: "bash"
57 +             scriptLocation:
58 +               "inlineScript"
59 +             inlineScript: |
60 +
61 +               deployment_output=$(az deployment sub
62 +                 create \
63 +                   --name
64 +                     $(Build.BuildId) \
65 +                     --template-file
66 +                       main.bicep \
67 +                         --location
68 +                           $(LOCATION) \
69 +                             --parameters
70 +                               resourcesPrefix=$(RESOURCES_PREFIX))
71 +
72 +           hostnames=$(echo
73 +             $deployment_output | jq -r -c
74 +               '.properties.outputs | map(.value) |
75 +                 join(",")')
76 +           echo
77 +             "##vso[task.setvariable
78 +               variable=hostnames;isOutput=true]$hostnam
79 +               es"
80 +
81 +           workingDirectory:
82 +             $(workDir)
83 +
84 +           - stage: Test
85 +             jobs:
86 +               - job: SmokeTest
87 +                 displayName: Smoke test
88 +                 variables:
89 +                   hostnames: $[
90 +                     stageDependencies.Deploy.DeployInfra.outp
91 +                       uts['DeployInfra.Deployment.hostnames'] ]
92 +
93 +           steps:
94 +             - task: PowerShell@2
95 +               name: RunSmokeTests
96 +               displayName: Run smoke tests
97 +               inputs:
```

```
78 +         filePath:
79 +             "$(workDir)/smokeTest.ps1"
80 +             arguments: "-HostNames
81 +                 $(hostnames)"
82 +             pwsh: true
83 +             workingDirectory:
84 +                 $(workDir)
85 +             - task: PublishTestResults@2
86 +                 displayName: Publish Test
87 +                     Results
88 +                     condition:
89 +                         succeededOrFailed()
```

▽ 64 resources/solutions/ado/ch3/.azure-pipelines/pipeline.bicep.deploy.yml 

... @@ -0,0 +1,64 @@

```
1 + trigger:
2 + branches:
3 +     include:
4 +         - main
5 + paths:
6 +     include:
7 +         - iac/bicep
8 +
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - name: ServiceConnectionName
15 +     value: AzureServiceConnection
16 +   - name: workDir
17 +     value:
18 +       "$(System.DefaultWorkingDirectory)/iac/bi
19 + cep"
20 +
21 + stages:
22 +   - stage: Preview
23 +     jobs:
24 +       - job: Preview
```

```
23 +         displayName: Preview Changes
24 +     steps:
25 +         - task: AzureCLI@2
26 +             name: RunWhatIf
27 +             displayName: What-If
28 +             inputs:
29 +                 azureSubscription:
30 +                     $(ServiceConnectionName)
31 +                     scriptType: "bash"
32 +                     scriptLocation:
33 +                         "inlineScript"
34 +                         inlineScript: |
35 +                             az deployment sub what-if
36 +                             \
37 +                             \
38 +                             --name $(Build.BuildId)
39 +                             \
40 +                             \
41 +                             --template-file
42 +                             main.bicep \
43 +                             \
44 +                             --location $(LOCATION)
45 +                             \
46 +                             \
47 +                             --parameters
48 +                             resourcesPrefix=$(RESOURCES_PREFIX)
49 +                             \
50 +                             \
51 +                             - stage: Deploy
52 +                             jobs:
53 +                                 - deployment: Deploy
54 +                                     displayName: Deployment
55 +                                     environment: production
56 +                                     strategy:
57 +                                         runOnce:
58 +                                             deploy:
59 +                                                 steps:
60 +                                                     - checkout: self
61 +                                                     \
62 +                                                     - task: AzureCLI@2
63 +                                                         name: Deployment
64 +                                                         displayName: Deployment
65 +                                                         inputs:
66 +                                                             azureSubscription:
67 +                     $(ServiceConnectionName)
68 +                     scriptType: "bash"
69 +                     scriptLocation:
70 +                         "inlineScript"
71 +                         inlineScript: |
```

```
59      +               az deployment sub
60      +             create \
61      +               --name
62      +               $(Build.BuildId) \
63      +               --template-file
64      +               main.bicep \
64      +               --location
64      +               $(LOCATION) \
64      +               --parameters
64      +               resourcesPrefix=$(RESOURCES_PREFIX)
64      +               workingDirectory:
64      +               $(workDir)
```

▽ 35 resources/solutions/ado/ch3/.azure-pipelines/pipeline.bicep.pr.yml

... @@ -0,0 +1,35 @@

```
1  + trigger:
2  +   - none
3  + # trigger and path filter are expected be
4  +   # configured in the ADO UI
5  +
6  + pool:
7  +   vmImage: ubuntu-latest
8  +
9  + variables:
10 +   - group: openhack
11 +   - name: ServiceConnectionName
12 +     value: AzureServiceConnection
13 +   - name: workDir
14 +     value:
15 +       "$(System.DefaultWorkingDirectory)/iac/bi
16 +       cep"
17 +
18 + steps:
19 +   - script: |
20 +     az bicep build --file main.bicep
21 +     name: Lint
22 +     displayName: "Bicep Lint"
23 +     workingDirectory: $(workDir)
24 +
25 +   - task: AzureCLI@2
26 +     name: Validate
27 +     displayName: "Bicep Validate"
28 +     inputs:
29 +       azureSubscription:
30 +         $(ServiceConnectionName)
```

```
27 +     scriptType: "bash"
28 +     scriptLocation: "inlineScript"
29 +     inlineScript: |
30 +       az deployment sub validate \
31 +         --name $(Build.BuildId) \
32 +         --template-file main.bicep \
33 +         --location "$(LOCATION)" \
34 +         --parameters
35 +           resourcesPrefix=$(RESOURCES_PREFIX)
35 +           workingDirectory: $(workDir)
```

▽ 123 ████ resources/solutions/ado/ch3/.azure-pipelines/pipeline.terraform.deploy.old.yml □

... @@ -0,0 +1,123 @@

```
1 + trigger:
2 +   branches:
3 +     include:
4 +       - main
5 +   paths:
6 +     include:
7 +       - iac/terraform
8 +
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - group: tfstate
15 +   - name: ServiceConnectionName
16 +     value: AzureServiceConnection
17 +   - name: workDir
18 +     value:
19 +       "$(System.DefaultWorkingDirectory)/iac/terraform"
20 + stages:
21 +   - stage: Preview
22 +     jobs:
23 +       - job: PreviewChanges
24 +         displayName: Preview
25 +         infrastructure changes
26 +         steps:
27 +           - task: TerraformInstaller@0
28 +             displayName: Install
29 +             Terraform
30 +             inputs:
31 +               terraformVersion: "latest"
```

```
29 +           - task: TerraformCLI@0
30 +             displayName: Run Terraform
31 +             Init
32 +             inputs:
33 +               command: "init"
34 +               workingDirectory:
35 +                 "$(workDir)"
36 +               backendType: "azurerm"
37 +               backendServiceArm:
38 +                 "$(ServiceConnectionName)"
39 +               backendAzureRmResourceGroupName:
40 +                 "$(TFSTATE_RESOURCES_GROUP_NAME)"
41 +               backendAzureRmStorageAccountName:
42 +                 "$(TFSTATE_STORAGE_ACCOUNT_NAME)"
43 +               backendAzureRmContainerName:
44 +                 "$(TFSTATE_STORAGE_CONTAINER_NAME)"
45 +               backendAzureRmKey:
46 +                 "$(TFSTATE_KEY)"
47 +               allowTelemetryCollection:
48 +                 true
49 +           - task: TerraformCLI@0
50 +             displayName: Run Terraform
51 +             Plan
52 +             inputs:
53 +               command: "plan"
54 +               workingDirectory:
55 +                 "$(workDir)"
56 +               environmentServiceName:
57 +                 "$(ServiceConnectionName)"
58 +               commandOptions: '-'
59 +                 detailed-exitcode -
60 +                   var="location=$(LOCATION)" -
61 +                   var="resources_prefix=$(RESOURCES_PREFIX
62 + )"
63 +               publishPlanResults:
64 +                 tfplan"
65 +               allowTelemetryCollection:
66 +                 true
67 +           - script: |
68 +             echo
69 +             ##vso[task.setvariable
70 +             variable=TERRAFORM_PLAN_HAS_CHANGES;isOutpu
71 +             tput=true]$TERRAFORM_PLAN_HAS_CHANGES"
```



```
81     backendAzureRmResourceGroupName:
81     +"$(TFSTATE_RESOURCES_GROUP_NAME)"
82     +
82     backendAzureRmStorageAccountName:
82     +"$(TFSTATE_STORAGE_ACCOUNT_NAME)"
83     +
83     backendAzureRmContainerName:
83     +"$(TFSTATE_STORAGE_CONTAINER_NAME)"
84     +
84     backendAzureRmKey:
84     +"$(TFSTATE_KEY)"
85     +
85     allowTelemetryCollection: true
85     + - task: TerraformCLI@0
86     +
86     displayName: Run
87     Terraform Apply
87     +
88     inputs:
88     + command: "apply"
89     +
89     workingDirectory:
90     +"$(workDir)"
90     +
91     environmentServiceName:
91     +"$(ServiceConnectionName)"
91     +
91     runAzLogin: true
92     +
92     commandOptions: '-'
92     var="location=$(LOCATION) -"
92     var="resources_prefix=$(RESOURCES_PREFIX)"
92     )'"
93     +
93     allowTelemetryCollection: true
94     +
94     - script: |
95     +
95     deployment_output=$(terraform output -
95     json)
96     +
96     hostnames=$(echo
96     "${deployment_output}" | jq -r -c
96     'map(.value) | join(",")')
97     +
97     echo
97     "##vso[task.setvariable
97     variable=hostnames;isOutput=true]$hostna
97     mes"
98     +
98     workingDirectory:
98     +"$(workDir)"
99     +
99     name: DeployOutput
100    +
101    - stage: Test
102    + jobs:
```

```
103 +     - job: SmokeTest
104 +         displayName: Smoke test
105 +         variables:
106 +             hostnames: $[
107 +                 stageDependencies.Deploy.DeployInfra.outputs['DeployInfra.DeployOutput.hostnames']
108 +             ]
109 +             steps:
110 +                 - task: PowerShell@2
111 +                     name: RunSmokeTests
112 +                     displayName: Run smoke tests
113 +                     inputs:
114 +                         filePath:
115 +                             "$(workDir)/smokeTest.ps1"
116 +                         arguments: "-HostNames
117 +                             $(hostnames)"
118 +                         pwsh: true
119 +                         workingDirectory:
120 +                             $(workDir)
121 +                         task: PublishTestResults@2
122 +                         displayName: Publish Test
123 +                         Results
124 +                         condition:
125 +                             succeededOrFailed()
126 +                         inputs:
127 +                             testResultsFormat: " NUnit"
128 +                             testResultsFiles:
129 +                                 "testResults.xml"
130 +                             searchFolder: $(workDir)
131 +                             mergeTestResults: true
```

▽ 99 resources/solutions/ado/ch3/.azure-pipelines/pipeline.terraform.deploy.yml ⌂

... @@ -0,0 +1,99 @@

```
1 + trigger:
2 + branches:
3 +     include:
4 +         - main
5 + paths:
6 +     include:
7 +         - iac/terraform
8 +
9 + pool:
10 +     vmImage: ubuntu-latest
11 +
12 + variables:
```

```
13 +   - group: openhack
14 +   - group: tfstate
15 +   - name: ServiceConnectionName
16 +     value: AzureServiceConnection
17 +   - name: workDir
18 +     value:
19 +       "$(System.DefaultWorkingDirectory)/iac/te
20 +       rraform"
21 +
22 + stages:
23 +   - stage: Preview
24 +     jobs:
25 +       - job: Preview
26 +         displayName: Preview Changes
27 +       steps:
28 +         - task: TerraformInstaller@0
29 +           displayName: Install
30 +
31 +         Terraform
32 +           inputs:
33 +             terraformVersion: "latest"
34 +
35 +           - task: TerraformCLI@0
36 +             displayName: Run Terraform
37 +
38 +             Init
39 +               inputs:
40 +                 command: "init"
41 +                 workingDirectory:
42 +                   "$(workDir)"
43 +                 backendType: "azurerm"
44 +                 backendServiceArm:
45 +                   "$(ServiceConnectionName)"
46 +
47 +                 backendAzureRmResourceGroupName:
48 +                   "$(TFSTATE_RESOURCES_GROUP_NAME)"
49 +
50 +                 backendAzureRmStorageAccountName:
51 +                   "$(TFSTATE_STORAGE_ACCOUNT_NAME)"
52 +
53 +                 backendAzureRmContainerName:
54 +                   "$(TFSTATE_STORAGE_CONTAINER_NAME)"
55 +
56 +                 backendAzureRmKey:
57 +                   "$(TFSTATE_KEY)"
58 +
59 +                 allowTelemetryCollection:
60 +                   true
61 +
62 +               - task: TerraformCLI@0
```

```
45 +         displayName: Run Terraform
46 +         Plan
47 +         inputs:
48 +           command: "plan"
49 +           workingDirectory:
50 +             "$(workDir)"
51 +           environmentServiceName:
52 +             "$(ServiceConnectionName)"
53 +           commandOptions: '-detailed-
54 +             exitcode -var="location=$(LOCATION)" -
55 +             var="resources_prefix=$(RESOURCES_PREFIX)
56 +             "
57 +           publishPlanResults:
58 +             "tfplan"
59 +           allowTelemetryCollection:
60 +             true
61 +
62 +           - script: |
63 +             echo
64 +               ##vso[task.setvariable
65 +               variable=TERRAFORM_PLAN_HAS_CHANGES;isOut
66 +               put=true]$TERRAFORM_PLAN_HAS_CHANGES"
67 +             echo
68 +               ##vso[task.setvariable
69 +               variable=TERRAFORM_PLAN_HAS_DESTROY_CHANG
70 +               ES;isOutput=true]$TERRAFORM_PLAN_HAS_DEST
71 +               ROY_CHANGES"
72 +
73 +           name: "PlanStatus"
74 +           displayName: "Set Variables
75 +             with Terraform Plan Exit Codes for next
76 +             stage"
77 +
78 +           - stage: Deploy
79 +             dependsOn: Preview
80 +             condition: and(succeeded(),
81 +               eq(dependencies.Preview.outputs['Preview.
82 +                 PlanStatus.TERRAFORM_PLAN_HAS_CHANGES'],
83 +                 'true'))
84 +             jobs:
85 +               - deployment: Deploy
86 +                 displayName: Deployment
87 +                 environment: production
88 +                 strategy:
89 +                   runOnce:
90 +                     deploy:
91 +                       steps:
```

```
71 +           - checkout: self
72 +
73 +           - task:
74 +               TerraformInstaller@0
75 +                   displayName: Install
76 +                   Terraform
77 +
78 +               - task: TerraformCLI@0
79 +                   displayName: Terraform
80 +                   Init
81 +                   inputs:
82 +                       command: "init"
83 +                   workingDirectory:
84 +                       "${workDir}"
85 +                   backendType:
86 +                       "azurerm"
87 +                   backendServiceArm:
88 +                       "${ServiceConnectionName}"
89 +
90 +                   backendAzureRmResourceGroupName:
91 +                       "${TFSTATE_RESOURCES_GROUP_NAME}"
92 +
93 +                   backendAzureRmStorageAccountName:
94 +                       "${TFSTATE_STORAGE_ACCOUNT_NAME}"
95 +                   backendAzureRmKey:
96 +                       "${TFSTATE_KEY}"
97 +
98 +                   allowTelemetryCollection: true
99 +
100 +               - task: TerraformCLI@0
101 +                   displayName: Terraform
102 +                   Apply
103 +                   inputs:
104 +                       command: "apply"
105 +                   workingDirectory:
106 +                       "${workDir}"
107 +
108 +                   environmentServiceName:
109 +                       "${ServiceConnectionName}"
110 +                   runAzLogin: true
```

```
98      +           commandOptions: '-'
98      +           var="location=$(LOCATION)" -
98      +           var="resources_prefix=$(RESOURCES_PREFIX)"
98      +
99      +           allowTelemetryCollection: true
```

▽ 40 resources/solutions/ado/ch3/.azure-pipelines/pipeline.terraform.pr.yml □

... @@ -0,0 +1,40 @@

```
1  + trigger:
2  +   - none
3  + # trigger and path filter are expected be
4  +   # configured in the ADO UI
5  +
6  + pool:
7  +   vmImage: ubuntu-latest
8  +
9  + variables:
10 +   - name: workDir
10 +     value:
10 +       "$(System.DefaultWorkingDirectory)/iac/te
10 +       rraform"
11 +
12 + steps:
13 +   - task: TerraformInstaller@0
13 +     displayName: Install Terraform
14 +     inputs:
14 +       terraformVersion: "latest"
15 +
16 +   - task: TerraformCLI@0
16 +     displayName: Terraform Lint
16 +     inputs:
16 +       command: "fmt"
16 +       workingDirectory: "$(workDir)"
16 +       commandOptions: "-check -recursive"
16 +       allowTelemetryCollection: true
17 +
18 +   - task: TerraformCLI@0
18 +     displayName: Terraform Init
18 +     inputs:
18 +       command: "init"
18 +       commandOptions: "-backend=false"
18 +       workingDirectory: "$(workDir)"
18 +       backendType: "selfConfigured"
18 +       allowTelemetryCollection: true
```

```
34 +  
35 +   - task: TerraformCLI@0  
36 +     displayName: Terraform Validate  
37 +     inputs:  
38 +       command: "validate"  
39 +       workingDirectory: "$(workDir)"  
40 +       allowTelemetryCollection: true
```

▽ 84 resources/solutions/ado/ch3/README.md 

... @@ -0,0 +1,84 @@

```
1 + # Challenge 3  
2 +  
3 + This challenge doesn't expect the  
4 + participants to edit the IoC code or the  
5 + application. Their task is to build  
6 + pipelines for automated deployment to  
7 + Azure.  
8 +  
9 + All pipelines are implemented in the  
10 + `azure-pipelines` folder. There are two  
11 + versions: _Bicep_ and _Terraform_. Each  
12 + has two flavors: _deploy_ and _pr_.  
13 +  
14 + On the first run, you might be asked to  
15 + permit access to variable group and  
16 + service connection. Also, if the pipeline  
17 + gets stuck in Pending state, check if the  
18 + lacking permissions are not blocking it.  
19 +  
20 + (ado-ch3-  
21 + permission1.png)  
22 +  
23 + (ado-ch3-  
24 + permission2.png)  
25 +  
26 + ## Build agents  
27 +  
28 + Due to a recent change in Azure DevOps,  
29 + it's possible that newly created ADO  
30 + project [might not get the free grant]  
31 + (https://devblogs.microsoft.com/devops/change-in-azure-pipelines-grant-for-private-projects/) of build minutes.  
32 +  
33 + There's a process to request it, but it
```

takes 2-3 business days to review, so start it in advance.

+  
+ Alternatives:  
+  
+ - Use your Azure Subscription to buy parallel jobs in ADO  
(`[https://dev.azure.com/\[organization name\]/\\_settings/billing](https://dev.azure.com/[organization name]/_settings/billing)`).  
+ - Provision a [custom build agent] (<https://docs.microsoft.com/azure/devops/pipelines/agents/v2-linux?view=azure-devops>) in your Azure Subscription with the following tools installed:  
+ - [pwsh] (<https://docs.microsoft.com/powershell/scripting/install/installing-powershell-on-linux?view=powershell-7.1>) (PowerShell Core)  
+ - [az] (<https://docs.microsoft.com/cli/azure/install-azure-cli-linux?pivots=apt>) (Azure CLI)  
+ - [sqlcmd] ([https://docs.microsoft.com/sql/linux/sql-server-linux-setup-tools?view\\$sql-server-ver15#ubuntu](https://docs.microsoft.com/sql/linux/sql-server-linux-setup-tools?view$sql-server-ver15#ubuntu)) (SQL Server tools)  
+ - unzip  
+ - jq  
+ - terraform (if taking the TF route)  
+ - docker (for future challenges)  
+  
+ ## Challenge  
+  
+ There are two kinds of pipelines that need to be built: `_validation_` and `_deployment_`. The validation pipeline should run tests on pull requests, but not deploy anything. The deployment itself is handled by the second pipeline, which would be triggered any time there's a merge into the ``main`` branch.  
+  
+ Validation:  
+  
+ 1. Trigger validation on pull requests.

```
36 + 1. Validate the IaC code with linter.  
37 +  
38 + Deployment:  
39 +  
40 + 1. Trigger deployment when changes in IaC  
are merged into `main`.  
41 + 1. Before deploying, show a preview of  
changes.  
42 + 1. Wait for approval after showing  
changes.  
43 +  
44 + Azure access has been pre-provisioned in  
Service Connections  
(`AzureServiceConnection`) and all  
necessary values are stored in Variable  
Groups. The team is not expected to set  
up any configuration, but they're  
expected to explore, discover what  
variables are available and use them in  
their pipelines.  
45 +  
46 + First step is to create pipelines for  
validation and deployment. You can look  
at the solution files for inspiration.  
47 +  
48 + > **Tip:** When creating a pipeline from  
an existing YAML file, save it first and  
then rename to something more  
appropriate.  
49 +  
50 + ! [ado-ch3-pipelines1](ado-ch3-  
pipelines1.png)  
51 +  
52 + ! [ado-ch3-pipelines2](ado-ch3-  
pipelines2.png)  
53 +  
54 + The key to solving this challenge is to  
properly configure the triggering branch  
(`iac/bicep` or `iac/terraform`) and the  
manual approval step - without it the  
pipeline will not stop after preview.  
First of all, an environment called  
`production` is expected to exist in  
Azure DevOps.  
55 +  
56 + ! [ado-ch3-environment1](ado-ch3-
```

```
    environment1.png)
57 +
58 + Then the ADO user interface can be used
  to configure mandatory PR validation with
  one of the pipelines.
59 +
60 + Go to **Repos -> Branches** and select
  the `main` branch to configure **Branch
  policies**:
61 +
62 + (ado-ch3-branch1.png)
63 +
64 + Click the **+** button next to **Build
  validation** and create new validation
  policy.
65 +
66 + For Bicep:
67 +
68 + (ado-ch3-pr-bicep.png)
69 +
70 + For Terraform:
71 +
72 + (ado-ch3-pr-
  terraform.png)
73 +
74 + Both created:
75 +
76 + (ado-ch3-pr-
  buildvalidation.png)
77 +
78 + Once the policy is configured, it won't
  be possible to push directly to `main`,
  but pull request will be required. To
  verify that the PR check works properly,
  create a new branch, make a change in the
  IaC code, push and create a pull request.
79 +
80 + (ado-ch3-
  pr-buildvalidation2.png)
81 +
82 + When the pull request gets merged,
  pipeline trigger (from the pipeline YAML)
  should kick in and start deployment.
83 +
84 + (ado-ch3-deploy1.png)
```

✓ BIN +13.5 KB resources/solutions/ado/ch3/ado-ch3-branch1.png 

✓ BIN +5.86 KB resources/solutions/ado/ch3/ado-ch3-deploy1.png 

✓ BIN +9.21 KB resources/solutions/ado/ch3/ado-ch3-environment1.png 

▽ BIN +5.69 KB resources/solutions/ado/ch3/ado-ch3-permission1.png 

▽ BIN +11 KB resources/solutions/ado/ch3/ado-ch3-permission2.png 

▽ BIN +7.64 KB resources/solutions/ado/ch3/ado-ch3-pipelines1.png 

✓ BIN +11.1 KB resources/solutions/ado/ch3/ado-ch3-pipelines2.png 

✓ BIN +24.1 KB resources/solutions/ado/ch3/ado-ch3-pr-bicep.png 

✓ BIN +17.3 KB resources/solutions/ado/ch3/ado-ch3-pr-buildvalidation.png 

▽ BIN +7.73 KB resources/solutions/ado/ch3/ado-ch3-pr-buildvalidation2.png 

▽ BIN +24.2 KB resources/solutions/ado/ch3/ado-ch3-pr-terraform.png 

✓ 70 resources/solutions/ado/ch4/.azure-pipelines/pipeline.pr.api.poi.yml

... @@ -0,0 +1,70 @@

```
1 + trigger:  
2 +   - none  
3 +  
4 + pool:
```

```
5 +   vmImage: ubuntu-latest
6 +
7 + variables:
8 +   - name: WORKDIR
9 +     value:
10 +       "$(System.DefaultWorkingDirectory)/apis/p
11 +         oi"
12 +
13 + steps:
14 +   - task: UseDotNet@2
15 +     displayName: "Setup .NET Core
16 +       $(DOTNET_VERSION)"
17 +     inputs:
18 +       packageType: "sdk"
19 +       version: $(DOTNET_VERSION)
20 +
21 +   - task: DotNetCoreCLI@2
22 +     displayName: "Restore dependencies"
23 +     inputs:
24 +       command: "restore"
25 +       projects: "$(WORKDIR)/*.sln"
26 +       feedsToUse: "select"
27 +
28 +   - task: DotNetCoreCLI@2
29 +     displayName: "Build the App"
30 +     inputs:
31 +       command: "build"
32 +       arguments: "--no-restore"
33 +       workingDirectory: "$(WORKDIR)"
34 +
35 +   - task: DotNetCoreCLI@2
36 +     name: UnitTest
37 +     displayName: "Run Unit Tests"
38 +     inputs:
39 +       command: "test"
40 +       arguments: '--no-build --filter
41 +         "FullyQualifiedNames~UnitTest"'
42 +       workingDirectory: "$(WORKDIR)"
43 +
44 +   - task: CreateWorkItem@1
45 +     displayName: "Create an issue (bug)"
46 +     condition: and(failed(),
47 +       ne(variables['Build.Reason'],
48 +         'PullRequest'))
```

```
45 +     inputs:
46 +       workItemType: "Bug"
47 +       title: "Unit Test failure"
48 +       assignedTo:
49 +         $(Build.RequestedForId)"
50 +       fieldMappings: |
51 +         System Info=Test details:
52 +           $($System.CollectionUri)$(System.TeamProje
53 +           ct)/_build/results?
54 +             buildId=$(Build.BuildId)&view=ms.vss-
55 +               test-web.build-test-results-tab
56 +             associate: true
57 +
58 +             - task: CreatePRCommentTask@1
59 +               displayName: "Create a PR comment"
60 +               condition: and(failed(),
61 +                 eq(variables['Build.Reason'],
62 +                   'PullRequest'))
63 +               inputs:
64 +                 AuthType: "system"
65 +                 Comment: |
66 +                   ### Unit Test `failure`  

67 +                     <details>
68 +                       <summary>Pipeline
69 +                         details</summary>
70 +                       Pipeline name:  

71 +                         `$(Build.DefinitionName)`  

72 +                         Action: `$(Build.Reason)`  

73 +                         Job: `$(System.JobName)`  

74 +                         PR:  

75 +                           [$(System.PullRequest.PullRequestId)]
76 +                             ($(System.PullRequest.SourceRepositoryURI
77 +                               /pullrequest/$(System.PullRequest.PullRe
78 +                                 questId)))
79 +                           </details>
80 +
81 +             Test details: [Run
82 +               #$(Build.BuildId)]
83 +               $($System.CollectionUri)$(System.TeamProj
84 +                 ect)/_build/results?
85 +                   buildId=$(Build.BuildId)&view=ms.vss-
86 +                     test-web.build-test-results-tab)
87 +             Pusher:
88 +               @<$(Build.RequestedForId)>
```

... @@ -0,0 +1,79 @@

```
1  + trigger:
2  +   - none
3  +
4  + pool:
5  +   vmImage: ubuntu-latest
6  +
7  + variables:
8  +   - name: WORKDIR
9  +     value:
10    "$(System.DefaultWorkingDirectory)/apis/trips"
11    - name: GOLANG_VERSION
12      value: "1.16"
13 +
14 + steps:
15 +   - task: GoTool@0
16   +   displayName: "Setup Go $(GOLANG_VERSION)"
17   +   inputs:
18     +     version: "$(GOLANG_VERSION)"
19 +
20 +   - task: Go@0
21   +   displayName: "Get dependencies"
22   +   inputs:
23     +     command: "get"
24     +     workingDirectory: "$(WORKDIR)"
25 +
26 +   - task: Go@0
27   +   displayName: "Build the app"
28   +   inputs:
29     +     command: "build"
30     +     workingDirectory: "$(WORKDIR)"
31 +
32 +   - script: |
33   +     chmod +x install_gotestsum.sh
34   +     ./install_gotestsum.sh
35   +     displayName: "Setup gotestsum"
36   +     workingDirectory: "$(WORKDIR)"
37 +
38 +   - script: ./gotestsum --format
39   +     standard-verbose --junitfile
40   +     unittest_results.xml -- ./tripsgo -run
```

```
        Unit -coverprofile=unittest_coverage.out
        -covermode=count
38    +   displayName: "Run Unit Tests"
39    +   workingDirectory: "$(WORKDIR)"
40    +
41    +   - task: PublishTestResults@2
42    +     displayName: "Publish Test Results"
43    +     condition: succeededOrFailed()
44    +     inputs:
45    +       testResultsFormat: "JUnit"
46    +       testResultsFiles:
47    +         "unittest_results.xml"
48    +       searchFolder: "$(WORKDIR)"
49    +       mergeTestResults: true
50    +       failTaskOnFailedTests: true
51    +
52    +   - task: CreateWorkItem@1
53    +     displayName: "Create an issue (bug)"
54    +     condition: and(failed(),
55    +       ne(variables['Build.Reason'],
56    +         'PullRequest'))
57    +     inputs:
58    +       workItemType: "Bug"
59    +       title: "Unit Test failure"
60    +       assignedTo:
61    +         "$(Build.RequestedForId)"
62    +       fieldMappings: |
63    +         System Info=Test details:
64    +           $(System.CollectionUri)$(System.TeamProject)/_build/results?
65    +             buildId=$(Build.BuildId)&view=ms.vss-test-web.build-test-results-tab
66    +             associate: true
67    +
68    +   - task: CreatePRCommentTask@1
69    +     displayName: "Create a PR comment"
70    +     condition: and(failed(),
71    +       eq(variables['Build.Reason'],
72    +         'PullRequest'))
```

```
71 +  
72 +     Pipeline name:  
73 +     `$(Build.DefinitionName)`  
74 +     Action: `$(Build.Reason)`  
75 +     Job: `$(System.JobName)`  
76 +     PR:  
77 +  
78 +     Test details: [Run  
79 +         #$(Build.BuildId)]  
    +     ($(System.CollectionUri)$(System.TeamProject)/_build/results?  
    +         buildId=$(Build.BuildId)&view=ms.vss-test-web.build-test-results-tab)  
79 +     Pusher:  
    +     @<$(Build.RequestedForId)>
```

▽ 56 resources/solutions/ado/ch4/.azure-pipelines/pipeline.pr.api.userjava.yml 

... @@ -0,0 +1,56 @@

```
1 + trigger:  
2 +   - none  
3 +  
4 + pool:  
5 +   vmImage: ubuntu-latest  
6 +  
7 + variables:  
8 +   - name: WORKDIR  
9 +     value:  
10 +       "$(System.DefaultWorkingDirectory)/apis/user-java"  
11 +       - name: JAVA_VERSION  
12 +         value: "11"  
13 +  
14 + steps:  
15 +   - task: Maven@3  
16 +     inputs:  
17 +       mavenPomFile: "$(WORKDIR)/pom.xml"  
18 +       goals: "test"  
19 +       publishJUnitResults: true  
    +       testResultsFiles: "**/surefire-reports/TEST-*.xml"
```

```
20 +         javaHomeOption: "JDKVersion"
21 +         jdkVersionOption:
22 +             "1.$(JAVA_VERSION)"
23 +         mavenVersionOption: "Default"
24 +         mavenOptions: "-Xmx3072m"
25 +         mavenAuthenticateFeed: false
26 +         effectivePomSkip: false
27 +         sonarQubeRunAnalysis: false
28 +
29 +     - task: CreateWorkItem@1
30 +         displayName: "Create an issue (bug)"
31 +         condition: and(failed(),
32 +             ne(variables['Build.Reason'],
33 +                 'PullRequest'))
34 +         inputs:
35 +             workItemType: "Bug"
36 +             title: "Unit Test failure"
37 +             assignedTo:
38 +                 "$(Build.RequestedForId)"
39 +             fieldMappings: |
40 +                 System Info=Test details:
41 +                     $(System.CollectionUri)$(System.TeamProject)/_build/results?
42 +                     buildId=$(Build.BuildId)&view=ms.vss-test-web.build-test-results-tab
43 +                     associate: true
44 +
45 +     - task: CreatePRCommentTask@1
46 +         displayName: "Create a PR comment"
47 +         condition: and(failed(),
48 +             eq(variables['Build.Reason'],
49 +                 'PullRequest'))
50 +         inputs:
51 +             AuthType: "system"
52 +             Comment: |
53 +                 ### Unit Test `failure`  

54 +                 <details>  

55 +                 <summary>Pipeline  

56 +                     details</summary>
57 +
58 +             Pipeline name:
59 +                 `$(Build.DefinitionName)`
60 +             Action: `$(Build.Reason)`
61 +             Job: `$(System.JobName)`
62 +             PR:
63 +                 [$(System.PullRequest.PullRequestId)]
```

```
$(System.PullRequest.SourceRepositoryURI  
)/pullrequest/${System.PullRequest.PullRe  
questId})  
53 +     </details>  
54 +  
55 +     Test details: [Run  
#$(Build.BuildId)]  
($System.CollectionUri)$(System.TeamProj  
ect)/_build/results?  
buildId=$(Build.BuildId)&view=ms.vss-  
test-web.build-test-results-tab)  
56 +     Pusher:  
@<$(Build.RequestedForId)>
```

✓ 67 resources/solutions/ado/ch4/.azure-pipelines/pipeline.pr.api.userprofile.yml 

... @@ -0,0 +1,67 @@

```
+ trigger:  
2 +   - none  
3 +  
4 + pool:  
5 +   vmImage: ubuntu-latest  
6 +  
7 + variables:  
8 +   - name: WORKDIR  
9 +     value:  
"$(System.DefaultWorkingDirectory)/apis/u  
serprofile"  
10 +    - name: NODE_VERSION  
11 +      value: "12"  
12 +  
13 + steps:  
14 +   - task: NodeTool@0  
15 +     displayName: "Setup Node  
$(NODE_VERSION)"  
16 +     inputs:  
17 +       versionSpec: "$(NODE_VERSION)"  
18 +  
19 + # npm ci is similar to npm install,  
+ except it's meant to be used in automated  
+ environments  
20 + #  
+   https://docs.npmjs.com/cli/v7/commands/np  
+ m-ci  
21 +   - script: npm ci  
22 +     displayName: "Install CI"
```

```
    dependencies"
23 +     workingDirectory: "$(WORKDIR)"
24 +
25 +     - script: npm run test
26 +       displayName: "Run Tests"
27 +       workingDirectory: "$(WORKDIR)"
28 +
29 +     - task: PublishTestResults@2
30 +       displayName: "Publish Test Results"
31 +       condition: succeededOrFailed()
32 +       inputs:
33 +         testResultsFormat: "JUnit"
34 +         testResultsFiles: "userprofile-
      report.xml"
35 +         searchFolder: "$(WORKDIR)/reports"
36 +         mergeTestResults: true
37 +         failTaskOnFailedTests: true
38 +
39 +     - task: CreateWorkItem@1
40 +       displayName: "Create an issue (bug)"
41 +       condition: and(failed(),
        ne(variables['Build.Reason'],
          'PullRequest'))
42 +       inputs:
43 +         workItemType: "Bug"
44 +         title: "Unit Test failure"
45 +         assignedTo:
        "$(Build.RequestedForId)"
46 +         fieldMappings: |
47 +           System Info=Test details:
        $(System.CollectionUri)$(System.TeamProje
        ct)/_build/results?
        buildId=$(Build.BuildId)&view=ms.vss-
        test-web.build-test-results-tab
48 +         associate: true
49 +
50 +     - task: CreatePRCommentTask@1
51 +       displayName: "Create a PR comment"
52 +       condition: and(failed(),
        eq(variables['Build.Reason'],
          'PullRequest'))
53 +       inputs:
54 +         AuthType: "system"
55 +         Comment: |
56 +           ### Unit Test `failure`  

57 +           <details>
```

```
58 +         <summary>Pipeline  
59 +         details</summary>  
60 +     Pipeline name:  
61 +         `$(Build.DefinitionName)`  
62 +         Action: `$(Build.Reason)`  
63 +         Job: `$(System.JobName)`  
63 +         PR:  
64 +             [$(System.PullRequest.PullRequestId)]  
64 +             ($(System.PullRequest.SourceRepositoryURI  
64 +             )/pullrequest/$(System.PullRequest.PullRe  
64 +             questId))  
64 +         </details>  
65 +     +  
66 +         Test details: [Run  
66 +             #$(Build.BuildId)]  
66 +             ($(System.CollectionUri)$(System.TeamProj  
66 +             ect)/_build/results?  
66 +             buildId=$(Build.BuildId)&view=ms.vss-  
66 +             test-web.build-test-results-tab)  
67 +         Pusher:  
67 +             @<$(Build.RequestedForId)>
```

▽ 17 resources/solutions/ado/ch4/README.md 

... @@ -0,0 +1,17 @@

```
1 + # Challenge 4  
2 +  
3 + All pipelines are implemented in the  
3 + `azure-pipelines` folder. There are four  
3 + pipelines for each API: *POI*, *Trips*,  
3 + *UserJava*, and *User*  
4 +  
5 + On the first run, you might be asked to  
5 + permit access to variable group and  
5 + service connection. Also, if the pipeline  
5 + gets stuck in Pending state, check if the  
5 + lacking permissions are not blocking it.  
6 +  
7 + ! [ado-ch3-permission1](../ch3/ado-ch3-  
7 + permission1.png)  
8 +  
9 + ! [ado-ch3-permission2](../ch3/ado-ch3-  
9 + permission2.png)  
10 +  
11 + ## Challenge
```

```
12 +
13 + A *pull request* workflow needs to be
14 + created, which builds the API, unit tests
15 + the API, and publishes test results. It
16 + is triggered by pull request creation.
17 +
18 + The solution is split into four files for
19 + each API.
20 +
21 + Follow the same validation steps as <a
22 + href="../ch3/README.md"
23 + target="_blank">Challenge 3</a>
```

▼ 86 resources/solutions/ado/ch5/.azure-pipelines/pipeline.deploy.api.poi.yml

... @@ -0,0 +1,86 @@

```
1 + trigger:
2 +   branches:
3 +     include:
4 +       - main
5 +   paths:
6 +     include:
7 +       - apis/poi
8 +
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - name: AzureServiceConnectionName
15 +     value: AzureServiceConnection
16 +   - name:
17 +     ContainerRegistryServiceConnectionName
18 +   - value: AzureContainerRegistry
19 +   - name: WORKDIR
20 +     value:
21 +       "$(Build.SourcesDirectory)/apis/poi/web"
22 +   - name: DOCKER_IMAGE_BASE_NAME
23 +     value: "devopsoh/api-poi"
24 +   - name: API_NAME
25 +     value: "poi"
26 +
27 + stages:
28 +   - stage: Build
29 +     displayName: Build
30 +     jobs:
```



```

$(AzureServiceConnectionName)
67 +           appType:
    webAppContainer
68 +           WebAppName:
    "$(RESOURCES_PREFIX)$(API_NAME)"
69 +           ResourceGroupName:
    "$(RESOURCES_PREFIX)rg"
70 +           DockerNamespace:
    "$(RESOURCES_PREFIX)cr.azurecr.io"
71 +           DockerRepository:
    $(DOCKER_IMAGE_BASE_NAME)
72 +           DockerImageTag:
    $(Build.BuildId)
73 +
74 +           - task:
        AzureAppServiceManage@0
75 +           inputs:
76 +           azureSubscription:
    $(AzureServiceConnectionName)
77 +           Action: "Stop Azure
      App Service"
78 +           WebAppName:
    "$(RESOURCES_PREFIX)$(API_NAME)"
79 +           ResourceGroupName:
    "$(RESOURCES_PREFIX)rg"
80 +
81 +           - task:
        AzureAppServiceManage@0
82 +           inputs:
83 +           azureSubscription:
    $(AzureServiceConnectionName)
84 +           Action: "Start Azure
      App Service"
85 +           WebAppName:
    "$(RESOURCES_PREFIX)$(API_NAME)"
86 +           ResourceGroupName:
    "$(RESOURCES_PREFIX)rg"

```

▽ 52 resources/solutions/ado/ch5/README.md 

... @@ -0,0 +1,52 @@

```

1 + # Challenge 5
2 +
3 + 1. When there's push to `main`, each of
   the APIs should have a container image
   published to Container Registry.

```

```
4 + 1. Publishing shouldn't be triggered by
5   pull requests.
6 +
7 + This challenge builds on the previous
8   one, so the CI part should be already in
9   place. The team is expected to leverage
10  it to make sure that image is built and
11  pushed to the container registry only
12  when all code tests passed earlier.
13 + There's a sample implementation in the
14  `pipeline.deploy.api.poi.yml` file, which
15  deploys the POI API. Similar approach is
16  to be taken for the other APIs (bonus
17  praise if the team leverages templates
18  and makes their pipeline implementation
19  reusable).
20 +
21 + To demonstrate different approaches, this
22  solution leverages the `Docker` task for
23  ADO, unlike GitHub, which used Azure CLI.
24  When trying to use it, you will notice
25  that there's a connection missing:
26 +
27 + (ch5-ado-
28  authorize-connection1.png)
29 +
30 + That's because the team needs to create
31  an ADO service connection for the Azure
32  Container Registry deployed earlier.
33 +
34 + 1. Go to **Project settings**.
35 + 1. Click **Service connections**.
36 + 1. Click **New service connection**.
37 + 1. Select **Docker Registry** and click
38  **Next**.
39 +
40 + 1. Configure a connection for your Azure
41  Container Registry using the options
42  documented [here]
43  (https://docs.microsoft.com/azure/devops/pipelines/ecosystems/containers/push-image?view=azure-devops#azure-container-registry) and [here]
```

```
(https://docs.microsoft.com/azure/devops/
pipelines/library/service-endpoints?
view=azure-devops&tabs=yaml#docker-hub-
or-others).
22 +   1. Enter `AzureContainerRegistry` as
the **connection name**.
23 + 1. Save.
24 +
25 + The Docker task uses local Docker
installation to build the image and then
pushes it into the connected container
registry.
26 +
27 + > **Note:** When using a self-hosted
build agent, make sure that the user
account running the ADO service has
permissions to work with Docker.
28 + >
29 + > `sudo usermod -a -G docker [user]`
30 +
31 + One of the key objectives in this
challenge is to use unique tag and don't
overwrite an existing one. This is
achieved by using the built-in variable
`$(Build.BuildId)` which contains the ID
of this run.
32 +
33 + ````yml
34 + - task: Docker@2
35 +   displayName: Build an image
36 +   inputs:
37 +     containerRegistry:
$(ContainerRegistryServiceConnectionName)
38 +     repository: $(DOCKER_IMAGE_BASE_NAME)
39 +     command: "build"
40 +     Dockerfile: "$(WORKDIR)/Dockerfile"
41 +     buildContext: "$(WORKDIR)"
42 +     arguments: "--build-arg
      build_version=$(Build.BuildId)"
43 +     tags: $(Build.BuildId)
44 +
45 + - task: Docker@2
46 +   displayName: Push an image
47 +   inputs:
48 +     containerRegistry:
$(ContainerRegistryServiceConnectionName)
```

```
49 +     repository: $(DOCKER_IMAGE_BASE_NAME)
50 +     command: "push"
51 +     tags: $(Build.BuildId)
52 +     ``
```

▽ BIN +19.1 KB resources/solutions/ado/ch5/ch5-ado-authorize-connection1.png 

▽ 106 ██████ resources/solutions/ado/ch6/.azure-pipelines/pipeline.deploy.api.poi.yml 

... @@ -0,0 +1,106 @@

```
1 + trigger:
2 + branches:
3 +   include:
4 +     - main
5 + paths:
6 +   include:
7 +     - apis/poi
8 +
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - name: WORKDIR
15 +     value:
16 +       "$(Build.SourcesDirectory)/apis/poi/web"
17 +     - name: DOCKER_IMAGE_BASE_NAME
18 +       value: "devopsoh/api-poi"
19 +     - name: API_NAME
20 +       value: "poi"
21 +
22 + stages:
23 +   - stage: Build
24 +     displayName: Build
25 +     jobs:
26 +       - job: Build
27 +         displayName: Build
28 +         steps:
```

```
29 +           - task: Docker@2
30 +             displayName: Build an image
31 +             inputs:
32 +               containerRegistry:
33 +                 "AzureContainerRegistry"
34 +               repository:
35 +                 "$(DOCKER_IMAGE_BASE_NAME)"
36 +               command: "build"
37 +               Dockerfile:
38 +                 "$(WORKDIR)/Dockerfile"
39 +               buildContext: "$(WORKDIR)"
40 +               arguments: "--build-arg
41 +                 build_version=$(Build.BuildId)"
42 +               tags: $(Build.BuildId)
43 +
44 +           - task: Docker@2
45 +             displayName: Push an image
46 +             inputs:
47 +               containerRegistry:
48 +                 "AzureContainerRegistry"
49 +               repository:
50 +                 "$(DOCKER_IMAGE_BASE_NAME)"
51 +               command: "push"
52 +               tags: $(Build.BuildId)
53 +
54 +           - stage: Staging
55 +             displayName: "Staging (Blue)"
56 +             jobs:
57 +               - deployment: Staging
58 +                 displayName: Staging
59 +                 environment: poi-staging
60 +               strategy:
61 +                 runOnce:
62 +                   deploy:
63 +                     steps:
64 +                       - task:
```

```
      "$(RESOURCES_PREFIX)$(API_NAME)"  
65 +           deployToSlotOrASE:  
66 +             true  
67 +             ResourceGroupName:  
68 +               "$(RESOURCES_PREFIX)rg"  
69 +             SlotName: staging  
70 +             DockerNamespace:  
71 +               "$(RESOURCES_PREFIX)cr.azurecr.io"  
72 +             DockerRepository:  
73 +               $(DOCKER_IMAGE_BASE_NAME)  
74 +             DockerImageTag:  
75 +               $(Build.BuildId)  
76 +             - task:  
77 +               AzureAppServiceManage@0  
78 +                 inputs:  
79 +                   azureSubscription:  
80 +                     "AzureServiceConnection"  
81 +                   Action: "Stop Azure  
82 +                     App Service"  
83 +                   WebAppName:  
84 +                     "$(RESOURCES_PREFIX)$(API_NAME)"  
85 +                   SpecifySlotOrASE:  
86 +                     true  
87 +                   ResourceGroupName:  
88 +                     "$(RESOURCES_PREFIX)rg"  
89 +                   Slot: staging  
90 +                   - stage: Production  
91 +                     displayName: "Production (Green)"  
92 +                   jobs:
```

```
93 +     - deployment: Production
94 +         displayName: Production
95 +         environment: poi-production
96 +         strategy:
97 +             runOnce:
98 +                 deploy:
99 +                     steps:
100 +                         - task:
101 +                             AzureAppServiceManage@0
102 +                                 inputs:
103 +                                     azureSubscription:
104 +                                         "AzureServiceConnection"
105 +                                         Action: "Swap Slots"
106 +                                         WebAppName:
107 +                                             "$(RESOURCES_PREFIX)$(API_NAME)"
108 +                                         ResourceGroupName:
109 +                                             "$(RESOURCES_PREFIX)rg"
110 +                                         SourceSlot:
111 +                                             "staging"
```

▽ 98 [ ] resources/solutions/ado/ch6/README.md

... @@ -0,0 +1,98 @@

```
1 + # Challenge 6
2 +
3 + This challenge builds on the previous
4 + one, so the CD part should be already in
5 + place. The team is expected to extend it
6 + to make sure that code is pushed to
7 + production only if staging deployment and
8 + its running succeeds.
9 +
10 + Same as in previous exercise there's a
11 + sample implementation in the
12 + `pipeline.deploy.api.poi.yml`, which
13 + deploys the POI API first to a staging
14 + slot of the POI web app.
15 +
16 + Instead of directly deploying to
17 + production the container built is first
18 + deployed in a staging slot
19 +
20 + ````yml
21 +     - stage: Staging
22 +         displayName: "Staging (Blue)"
23 +         jobs:
```

```
13 +     - deployment: Staging
14 +         displayName: Staging
15 +         environment: poi-staging
16 +         strategy:
17 +             runOnce:
18 +                 deploy:
19 +                     steps:
20 +                         - task:
21 +                             AzureRmWebAppDeployment@4
22 +                                 inputs:
23 +                                     ConnectionType:
24 +                                         "AzureRM"
25 +                                         azureSubscription:
26 +                                             "AzureServiceConnection"
27 +                                         appType:
28 +                                             webAppContainer
29 +                                         WebAppName:
30 +                                             "$(RESOURCES_PREFIX)$(API_NAME)"
31 +                                         deployToSlotOrASE:
32 +                                             true
33 +                                         ResourceGroupName:
34 +                                             "$(RESOURCES_PREFIX)rg"
35 +                                         SlotName: staging
36 +                                         DockerNamespace:
37 +                                             "$(RESOURCES_PREFIX)cr.azurecr.io"
38 +                                         DockerRepository:
39 +                                             $(DOCKER_IMAGE_BASE_NAME)
40 +                                         DockerImageTag:
41 +                                             $(Build.BuildId)
42 +                                         - task:
```

```
AzureAppServiceManage@0
43 +           inputs:
44 +             azureSubscription:
45 +               "AzureServiceConnection"
46 +             Action: "Start Azure
47 +               App Service"
48 +             WebAppName:
49 +               "$(RESOURCES_PREFIX)$(API_NAME)"
50 +             SpecifySlotOrASE:
51 +               true
52 +             ResourceGroupName:
53 +               "$(RESOURCES_PREFIX)rg"
54 +             Slot: staging
55 +             ``
56 +             +
57 +             + Then to test whether the staging slot is
58 +               up and running you will have to:
59 +             +
60 +             + ![[Service Connection creation]](ado-
61 +               ch6-genericsvc.png)
62 +             +
63 +             + 2. Create a [Rest Call task]
64 +               (https://docs.microsoft.com/en-
65 +                 us/azure/devops/pipelines/tasks/utility/h
66 +                   ttp-rest-api?view=azure-devops) using
67 +                     that service connection as a gate to move
68 +                       to the production deployment for example
69 +             +
70 +             ``yaml
71 +             # Invoke REST API
72 +             # Invoke a REST API as a part of your
73 +               pipeline.
74 +             - task: InvokeRESTAPI@1
75 +               inputs:
76 +                 #connectionType:
77 +                   'connectedServiceName' # Options:
78 +                     connectedServiceName,
79 +                     connectedServiceNameARM
80 +                     +
81 +                     #serviceConnection: # Required
82 +                     when connectionType ==
83 +                       ConnectedServiceName
84 +                     +
85 +                     #azureServiceConnection: #
86 +                     Required when connectionType ==
87 +                     ``
```

```
    ConnectedServiceNameARM
68      #method: 'POST' # Options:
69        OPTIONS, GET, HEAD, POST, PUT, DELETE,
70        TRACE, PATCH
71      #headers: '{Content-
72        Type:application/json, PlanUrl:
73        $(system.CollectionUri), ProjectId:
74        $(system.TeamProjectId), HubName:
75        $(system.HostType), PlanId:
76        $(system.PlanId), JobId: $(system.JobId),
77        TimelineId: $(system.TimelineId),
78        TaskInstanceId: $(system.TaskInstanceId),
79        AuthToken: $(system.AccessToken)}'
80      #body: # Required when method != GET && Method != HEAD
81      #urlSuffix: # Optional
82      #waitForCompletion: 'false' #
83      Options: true, false
84      #successCriteria: # Optional
85      ````yml
86      - stage: Production
87        displayName: "Production (Green)"
88        jobs:
89          - deployment: Production
90            displayName: Production
91            environment: poi-production
92            strategy:
93              runOnce:
94                deploy:
95                  steps:
96                    - task:
97                      AzureAppServiceManage@0
98                      inputs:
99                        azureSubscription:
100                          "AzureServiceConnection"
101                        Action: "Swap Slots"
```

```
95 +             WebAppName :  
96 +                 "$(RESOURCES_PREFIX)$(API_NAME)"  
96 +             ResourceGroupName :  
97 +                 "$(RESOURCES_PREFIX)rg"  
97 +             SourceSlot: "staging"  
98 +             ...
```

▽ BIN +25.7 KB resources/solutions/ado/ch6/ado-ch6-genericsvc.png 

▽ BIN +48 KB resources/solutions/ado/ch6/ado-ch6-invokerest.png 



✓ 81 ████ resources/solutions/ado/ch7/.azure-pipelines/pipeline.bicep.deploy.yml □

... @@ -0,0 +1,81 @@

```
1 + trigger:
2 +   branches:
3 +     include:
4 +       - main
5 +   paths:
6 +     include:
7 +       - iac/bicep
8 +
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - group: keyvault # option 1: Variable
15 +     Group linked to KV
16 +   - name: ServiceConnectionName
17 +     value: AzureServiceConnection
18 +   - name: workDir
19 +     value:
20 +       "$(System.DefaultWorkingDirectory)/iac/bi
21 +       cep"
22 +
23 + stages:
24 +   - stage: Preview
25 +     jobs:
26 +       - job: Preview
27 +         displayName: Preview Changes
28 +         steps:
29 +           # option 2: Query KV
30 +           # - task: AzureKeyVault@2
31 +           #   inputs:
32 +           #     azureSubscription:
```

```
      "$(ServiceConnectionName)"  
30 +          #     KeyVaultName:  
      "$(RESOURCES_PREFIX)kv"  
31 +          #     SecretsFilter: "SQL-  
      PASSWORD"  
32 +          #     RunAsPreJob: true  
33 +  
34 +      - task: AzureCLI@2  
35 +          name: RunWhatIf  
36 +          displayName: What-If  
37 +          inputs:  
38 +              azureSubscription:  
      $(ServiceConnectionName)  
39 +                  scriptType: "bash"  
40 +                  scriptLocation:  
      "inlineScript"  
41 +                  inlineScript: |  
42 +                      az deployment sub what-if  
      \  
43 +                          --name $(Build.BuildId)  
      \  
44 +                          --template-file  
      main.bicep \  
45 +                          --location $(LOCATION)  
      \  
46 +                          --parameters  
      resourcesPrefix=$(RESOURCES_PREFIX)  
      sqlServerAdminPassword=$(SQL-PASSWORD)  
47 +                          workingDirectory:  
      $(workDir)  
48 +  
49 +      - stage: Deploy  
50 +          jobs:  
51 +              - deployment: Deploy  
52 +                  displayName: Deployment  
53 +                  environment: production  
54 +                  strategy:  
55 +                      runOnce:  
56 +                          deploy:  
57 +                          steps:  
58 +                              - checkout: self  
59 +  
60 +                          # option 2: Query KV  
61 +                          # - task: AzureKeyVault@2  
62 +                          #     inputs:  
63 +                          #     azureSubscription:
```

```
      "$(ServiceConnectionName)"  
64      #      KeyVaultName:  
      "$(RESOURCES_PREFIX)kv"  
65      #      SecretsFilter:  
      "SQL-PASSWORD"  
66      #      RunAsPreJob: true  
67      +  
68      - task: AzureCLI@2  
69      +      name: Deployment  
70      +      displayName: Deployment  
71      +      inputs:  
72      +          azureSubscription:  
      $(ServiceConnectionName)  
73      +          scriptType: "bash"  
74      +          scriptLocation:  
      "inlineScript"  
75      +          inlineScript: |  
76      +              az deployment sub  
      create \  
77      +              --name  
      $(Build.BuildId) \  
78      +              --template-file  
      main.bicep \  
79      +              --location  
      $(LOCATION) \  
80      +              --parameters  
      resourcesPrefix=$(RESOURCES_PREFIX)  
      sqlServerAdminPassword=$(SQL-PASSWORD)  
81      +              workingDirectory:  
      $(workDir)
```

▼ 142  resources/solutions/ado/ch7/.azure-pipelines/pipeline.pr.api.poi.yml 

... @@ -0,0 +1,142 @@

```
1  + trigger:  
2  +   - none  
3  +  
4  + pool:  
5  +   vmImage: ubuntu-latest  
6  +  
7  + variables:  
8  +   - name: WORKDIR  
9  +     value:  
      "$(System.DefaultWorkingDirectory)/apis/  
      poi"  
10 +    - name: DOTNET_VERSION
```

```
11 +     value: "3.1.x"
12 +
13 +   jobs:
14 +     - job: Test
15 +       displayName: Test
16 +       steps:
17 +         - task: UseDotNet@2
18 +           displayName: "Setup .NET Core
19 +             $(DOTNET_VERSION)"
20 +           inputs:
21 +             packageType: "sdk"
22 +             version: $(DOTNET_VERSION)
23 +
24 +         - task: DotNetCoreCLI@2
25 +           displayName: "Restore
26 +             dependencies"
27 +           inputs:
28 +             command: "restore"
29 +             projects: "$(WORKDIR)/*.sln"
30 +             feedsToUse: "select"
31 +
32 +         - task: DotNetCoreCLI@2
33 +           displayName: "Build the App"
34 +           inputs:
35 +             command: "build"
36 +             arguments: "--no-restore"
37 +             workingDirectory: "$(WORKDIR)"
38 +
39 +         - task: DotNetCoreCLI@2
40 +           name: UnitTest
41 +           displayName: "Run Unit Tests"
42 +           inputs:
43 +             command: "test"
44 +             arguments: '--no-build --
45 +               filter "FullyQualifiedName~UnitTest"'
46 +             workingDirectory: "$(WORKDIR)"
47 +
48 +         - task: CreateWorkItem@1
49 +           displayName: "Create an issue
50 +             (bug)"
51 +           condition: and(failed(),
52 +             ne(variables['Build.Reason'],
53 +               'PullRequest'))
54 +           inputs:
55 +             workItemType: "Bug"
56 +             title: "Unit Test failure"
```

```

51 +           assignedTo:
52 +             "$(Build.RequestedForId)"
53 +           fieldMappings: |
54 +             System Info=Test details:
55 +               $(System.CollectionUri)$(System.TeamProj
56 +               ect)/_build/results?
57 +               buildId=$(Build.BuildId)&view=ms.vss-
58 +               test-web.build-test-results-tab
59 +           associate: true
60 +
61 +           - task: CreatePRCommentTask@1
62 +             displayName: "Create a PR
63 +             comment"
64 +             condition: and(failed(),
65 +             eq(variables['Build.Reason'],
66 +             'PullRequest'))
67 +             inputs:
68 +               AuthType: "system"
69 +               Comment: |
70 +                 ### Unit Test `failure`  

71 +                 <details>
72 +                   <summary>Pipeline
73 +                     details</summary>
74 +                   </details>
75 +                   Pipeline name:
76 +                     `$(Build.DefinitionName)`  

77 +                     Action: `$(Build.Reason)`  

78 +                     Job: `$(System.JobName)`  

79 +                     PR:
80 +                       [$(System.PullRequest.PullRequestId)]
81 +                         ($(System.PullRequest.SourceRepositoryUR
82 +                           I)/pullrequest/$(System.PullRequest.Pull
83 +                           RequestId))
84 +                     </details>
85 +
86 +           Test details: [Run
87 +             #$(Build.BuildId)]
88 +             ($(System.CollectionUri)$(System.TeamPro
89 +               ject)/_build/results?
90 +               buildId=$(Build.BuildId)&view=ms.vss-
91 +               test-web.build-test-results-tab)
92 +           Pusher:
93 +             @<$(Build.RequestedForId)>
94 +
95 +           - job: SecretsScanningCredScan
96 +             displayName: "Secrets Scanning - "

```

```
    CredScan"
77 +   pool:
78 +     vmImage: windows-latest
79 +   steps:
80 +     - checkout: self
81 +
82 +     - task: CredScan@3
83 +       displayName: "Scan source code
for committed credentials"
84 +       inputs:
85 +         toolVersion: "2.2.7.8"
86 +         batchSize: "32"
87 +
88 +     - task: SdtReport@2
89 +       displayName: "Generates a report
from the CredScan results"
90 +       inputs:
91 +         GdnExportAllTools: false
92 +         GdnExportGdnToolCredScan: true
93 +
94 +         GdnExportGdnToolCredScanSeverity:
95 +           "Warning"
96 +
97 +     - task:
98 +       PublishSecurityAnalysisLogs@3
99 +         displayName: "Publishes artifact
to CodeAnalysisLogs"
100 +        inputs:
101 +          ArtifactName:
102 +            "CodeAnalysisLogs"
103 +          ArtifactType: "Container"
104 +          PublishProcessedResults: true
105 +          AllTools: false
106 +          AntiMalware: false
107 +          APIScan: false
108 +          Armory: false
109 +          Bandit: false
110 +          BinSkim: false
111 +          CodesignValidation: false
112 +          CredScan: true
113 +          CSRF: false
114 +          ESLint: false
115 +          Flawfinder: false
116 +          FortifySCA: false
117 +          FxCop: false
118 +          ModernCop: false
```

```

115 +     MSRD: false
116 +     PoliCheck: false
117 +     RoslynAnalyzers: false
118 +     SDLNativeRules: false
119 +     Semmle: false
120 +     SpotBugs: false
121 +     TSLint: false
122 +     WebScout: false
123 +     ToolLogsNotFoundAction:
124 +         "Standard"
125 +         - task: PostAnalysis@2
126 +             displayName: "Break the build"
127 +             inputs:
128 +                 GdnBreakPolicyMinSev:
129 +                     "Warning"
130 +                     GdnBreakAllTools: false
131 +                     GdnBreakGdnToolCredScan: true
132 +
133 +                     GdnBreakGdnToolCredScanSeverity:
134 +                         "Warning"
135 +                         - job: SecretsScanningGitleaks
136 +                             displayName: "Secrets Scanning - "
137 +                             Gitleaks"
138 +                             steps:
139 +                                 - task: Gitleaks@2
140 +                                     inputs:
141 +                                         scanlocation:
142 +                                             "$(Build.SourcesDirectory)"

```

▽ 116 resources/solutions/ado/ch7/.azure-pipelines/pipeline.terraform.deploy.yml □

... @@ -0,0 +1,116 @@

```

1 + trigger:
2 +   branches:
3 +     include:
4 +       - main
5 +     paths:
6 +       include:
7 +         - iac/terraform

```

```
8 + 
9 + pool:
10 +   vmImage: ubuntu-latest
11 +
12 + variables:
13 +   - group: openhack
14 +   - group: keyvault # option 1: Variable
15 +     Group linked to KV
16 +   - group: tfstate
17 +   - name: ServiceConnectionName
18 +     value: AzureServiceConnection
19 +   - name: workDir
20 +     value:
21 +       "$(System.DefaultWorkingDirectory)/iac/t
22 + erraform"
23 +
24 + stages:
25 +   - stage: Preview
26 +     jobs:
27 +       - job: Preview
28 +         displayName: Preview Changes
29 +         steps:
30 +           - task: TerraformInstaller@0
31 +             displayName: Install
32 +             Terraform
33 +             inputs:
34 +               terraformVersion: "latest"
35 +
36 +           - task: TerraformCLI@0
37 +             displayName: Run Terraform
38 +             Init
39 +             inputs:
40 +               command: "init"
41 +               workingDirectory:
42 +                 "$(workDir)"
43 +               backendType: "azurerm"
44 +               backendServiceArm:
45 +                 "$(ServiceConnectionName)"
46 +
47 +               backendAzureRmResourceGroupName:
48 +                 "$(TFSTATE_RESOURCES_GROUP_NAME)"
49 +
50 +               backendAzureRmStorageAccountName:
51 +                 "$(TFSTATE_STORAGE_ACCOUNT_NAME)"
52 +
53 +               backendAzureRmContainerName:
```

```
        "$(TFSTATE_STORAGE_CONTAINER_NAME)"  
42    +           backendAzureRmKey:  
43    +           "$(TFSTATE_KEY)"  
44    +           allowTelemetryCollection:  
45    +               true  
46    +  
47    +               # option 2: Query KV  
48    +               # - task: AzureKeyVault@2  
49    +               #   inputs:  
50    +               #     azureSubscription:  
51    +                   "$(ServiceConnectionName)"  
52    +               #     KeyVaultName:  
53    +                   "$(RESOURCES_PREFIX)kv"  
54    +               #     SecretsFilter: "SQL-  
55    +                   PASSWORD"  
56    +               #     RunAsPreJob: true  
57    +  
58    +               - task: TerraformCLI@0  
59    +                   displayName: Run Terraform  
60    +                   Plan  
61    +                   inputs:  
62    +                   command: "plan"  
63    +                   workingDirectory:  
64    +                       "$(workDir)"  
65    +                   environmentServiceName:  
66    +                       "$(ServiceConnectionName)"  
67    +                   commandOptions: '-  
68    +                       detailed-exitcode -  
69    +                       var="location=$(LOCATION)" -  
70    +                       var="resources_prefix=$(RESOURCES_PREFIX)  
71    +                           )" -  
72    +                       var="mssql_server_administrator_login_pa  
73    +                           ssword=$(SQL-PASSWORD)"  
74    +                   publishPlanResults:  
75    +                       "tfplan"  
76    +                   allowTelemetryCollection:  
77    +               true  
78    +  
79    +               - script: |  
80    +                   echo  
81    +                   "##vso[task.setvariable  
82    +                       variable=TERRAFORM_PLAN_HAS_CHANGES;isOut  
83    +                       put=true]$TERRAFORM_PLAN_HAS_CHANGES"  
84    +                   echo  
85    +                   "##vso[task.setvariable  
86    +                       variable=TERRAFORM_PLAN_HAS_DESTROY_CHAN
```

```
GES;isOutput=true]$TERRAFORM_PLAN_HAS_DE  
STROY_CHANGES"  
66      name: "PlanStatus"  
67      displayName: "Set Variables  
with Terraform Plan Exit Codes for next  
stage"  
68  
69      - stage: Deploy  
70      dependsOn: Preview  
71      condition: and(succeeded(),  
eq(dependencies.Preview.outputs['Preview  
.PlanStatus.TERRAFORM_PLAN_HAS_CHANGES'])  
, 'true'))  
72      jobs:  
73          - deployment: Deploy  
74          displayName: Deployment  
75          environment: production  
76          strategy:  
77              runOnce:  
78                  deploy:  
79                  steps:  
80                      - checkout: self  
81  
82                      - task:  
TerraformInstaller@0  
83                      displayName: Install  
Terraform  
84                      inputs:  
85                      terraformVersion:  
"latest"  
86  
87                      - task: TerraformCLI@0  
88                      displayName: Terraform  
Init  
89                      inputs:  
90                      command: "init"  
91                      workingDirectory:  
"${workDir}"  
92                      backendType:  
"azurerm"  
93                      backendServiceArm:  
"${ServiceConnectionName}"  
94  
backendAzureRmResourceGroupName:  
"${TFSTATE_RESOURCES_GROUP_NAME}"  
95
```

```

96   backendAzureRmStorageAccountName:
97     +"$(TFSTATE_STORAGE_ACCOUNT_NAME)"
98   +
99     allowTelemetryCollection: true
100    +
101      # option 2: Query KV
102      # - task:
103        AzureKeyVault@2
104        #
105          # inputs:
106          #   azureSubscription:
107            +"$(ServiceConnectionName)"
108            #   KeyVaultName:
109            +"$(RESOURCES_PREFIX)kv"
110            #   SecretsFilter:
111            "SQL-PASSWORD"
112            #   RunAsPreJob: true
113            +
114            #   task: TerraformCLI@0
115            #   displayName: Terraform
116            Apply
117            inputs:
118              command: "apply"
119            workingDirectory:
120              +"$(workDir)"
121            +
122              environmentServiceName:
123                +"$(ServiceConnectionName)"
124                runAzLogin: true
125                commandOptions: '-'
126                  var="location=$(LOCATION)" -
127                  var="resources_prefix=$(RESOURCES_PREFIX)" -
128                  var="mssql_server_administrator_login_password=$(SQL-PASSWORD)"
129                +
130                  allowTelemetryCollection: true

```

▽ 74 resources/solutions/ado/ch7/README.md 

... @@ -0,0 +1,74 @@

1 + # Challenge 7

```
2 +
3 + ## Scan for secrets
4 +
5 + Real secrets to find are located in two
6 + files: `iac/terraform/locals.defaults.tf`  

7 + and `iac/bicep/variables.json`. The rest
8 + are false-positive and have to be
9 + suppressed/allowlisted.
10 +
11 + > Because secrets are in Terraform and
12 + Bicep files, and your team has chosen the
13 + IaC path at the beginning of the
14 + OpenHack, it's OK to mark it as a false-
15 + positive for opposite IaC to your team's
16 + choice.
17 +
18 + #### CredScan
19 +
20 + TODO
21 +
22 + #### Gitleaks
23 +
24 + TODO
25 +
26 + #### Custom rules in TOML
27 +
28 + TODO
29 +
30 + #### Move secrets to Azure Key Vault
31 +
32 + Follow this guide:
33 + [https://docs.microsoft.com/en-
34 + us/azure/app-service/app-service-key-
35 + vault-references]
36 + (https://docs.microsoft.com/en-
37 + us/azure/app-service/app-service-key-
38 + vault-references)
39 +
40 + #### Remove secrets from the source code
41 +
42 + 1. Remove secrets from the IaC source
43 + code
44 + 1. Update IaC deploy pipeline
45 + 1. Connect Variable Group to Key Vault
46 + or query Key Vault using a task
47 + 2. Pass secret as a parameter
```

```
31      +      - Bicep: `sqlServerAdminPassword`  
32      +      - Terraform:  
33          `mssql_server_administrator_login_password`  
34  
35      + ## (optional) Secret Rotation  
36      +  
37      + The solution for the Secret Rotation will  
38      + require the team to create a KeyVault  
39      + with the name {RESOURCES_PREFIX}kv.  
40      +  
41      + `az keyvault create --name  
42      + {RESOURCES_PREFIX}kv -g  
43      + {RESOURCES_PREFIX}rg`  
44      +  
45      + They will then need to create the ado  
46      + pipeline to deploy the dependent  
47      + resources listed below for the rotation  
48      + to work. The pipeline is located in the  
49      + team's repo  
50          (`support/sqlsecretrotation/.azure-  
51          pipelines/pipeline.bicep.sqlsecret.yml`).  
52      +  
53      + - {RESOURCES_PREFIX}sqlsecrotrg  
54      + - {RESOURCES_PREFIX}sqlsecrotst  
55      + - {RESOURCES_PREFIX}sqlsecrotplan  
56      + - {RESOURCES_PREFIX}sqlsecrotfunc  
57      + - {RESOURCES_PREFIX}sqlsecrotappi  
58      + - EventGrid System Topic in the  
59      + {RESOURCES_PREFIX}rg  
60      +  
61      + Once the Keyvault is created and the  
62      + supporting services are created using the  
63      + ado pipeline they will need to create the  
64      + secret and add the tags properly.  
65      +  
66      + `az keyvault secret set --name SQL-  
67      + PASSWORD --vault-name  
68      + {RESOURCES_PREFIX}kv --value  
69      + "demo!P@55w0rd123" --tags  
70      + "CredentialId=demouserssa"  
71      + "ProviderAddress=  
72      + {MydrvingDB_Azure_ResourceID}"  
73      + "ValidityPeriodDays=90"`  
74      +  
75      + Finally configure the poi app service to
```

```
      read the password value from the secret
      in keyvault with Key Vault references for
      App Service.

54  +
55  + - Enable system assigned identity on the
      poi app service.
56  + !*[Enable system assigned identity on the
      poi app service.]*
      (enable_system_identity_poi.png "Enable
      system assigned identity on the poi app
      service.")

57  +
58  + - Add Access Policy in Keyvault
      {RESOURCES_PREFIX}kv for the system
      assigned identity for the poi app
      service.
59  + !*[Add Keyvault Access Policy.]*
      (add_access_policy.png "Add Keyvault
      Access Policy.")

60  +
61  + - Add keyvault reference to the poi app
      service application settings.
62  + @Microsoft.KeyVault(VaultName=
      {RESOURCES_PREFIX}kv;SecretName=SQL-
      PASSWORD)
63  +
64  + !*[Add keyvault reference to the poi app
      service application settings]*
      (update_sql_pwd_kv.png "Add keyvault
      reference to the poi app service
      application settings.")
65  + !*[Verify keyvault app settings]*
      (appservice_keyvault_ref.png "Verify
      keyvault app settings.")

66  +
67  + ### Testing
68  +
69  + - Set the secret to expire within 29 days
      of the current date.
70  + !*[Set the secret to expire within 29 days
      of the current date]*
      (secret_exp_test_date.png "Set the secret
      to expire within 29 days of the current
      date.")

71  + - Save the change and wait for the
      rotation to happen. This may take a few
```

- minutes to complete.
- 72 + - Once the rotation is complete you should see multiple versions of the secret one current and many historical depending on the number of time tested
- 73 + ! [Once the rotation is complete you should see multiple versions of the secret one current and many historical depending on the number of time tested]  
(secret\_rotation\_versions.png "Once the rotation is complete you should see multiple versions of the secret one current and many historical depending on the number of time tested.")
- 74 + - Validate the poi/api endpoint for the app service is still service data.

▽ BIN +214 KB resources/solutions/ado/ch7/add\_access\_policy.png 

▽ BIN +37.2 KB resources/solutions/ado/ch7/appservice\_keyvault\_ref.png 

▽ BIN +142 KB resources/solutions/ado/ch7/enable\_system\_identity\_poi.png 

✓ BIN +186 KB resources/solutions/ado/ch7/secret\_exp\_test\_date.png 

✓ BIN +176 KB resources/solutions/ado/ch7/secret\_rotation\_versions.png 

✓ BIN +225 KB resources/solutions/ado/ch7/update\_sql\_pwd\_kv.png 

✓ 11 ■■■■■ resources/solutions/ado/ch8/README.md 

... @@ -0,0 +1,11 @@

```
1 + # Challenge 8
2 +
3 + ## Dependency Scanning
4 +
5 + #### WhiteSource Bolt
6 +
7 + Follow instructions in [WhiteSource Bolt
8 + for Azure Pipelines]
9 + (https://whitesource.atlassian.net/wiki/spaces/WD/pages/1641644045/WhiteSource+Bolt+for+Azure+Pipelines)
10 +
11 + #### OWASP Dependency Check
```

Check]  
(<https://marketplace.visualstudio.com/items?itemName=dependency-check.dependencycheck>)

▽ 12 resources/solutions/gh/README.md □

... @@ -0,0 +1,12 @@

```
1 + # GitHub
2 +
3 + - [Challenge-1](ch1)
4 + - [Challenge-2](ch2)
5 + - [Challenge-3](ch3)
6 + - [Challenge-4](ch4)
7 + - [Challenge-5](ch5)
8 + - [Challenge-6](ch6)
9 + - [Challenge-7](ch7)
10 + - [Challenge-8](ch8)
11 + - [Challenge-9](ch9)
12 + - [Challenge-10](ch10)
```

▽ 21 resources/solutions/gh/ch1/README.md □

... @@ -0,0 +1,21 @@

```
1 + # Challenge 1
2 +
3 + ## Board
4 +
5 + ! [Board](gh-ch1-board.png)
6 +
7 + ## Board Details
8 +
9 + ! [Part 1](gh-ch1-board01.png)
10 +
11 + ! [Part 2](gh-ch1-board02.png)
12 +
13 + ! [Part 3](gh-ch1-board03.png)
14 +
15 + ! [Part 4](gh-ch1-board04.png)
16 +
17 + ! [Part 5](gh-ch1-board05.png)
18 +
19 + ! [Part 6](gh-ch1-board06.png)
20 +
21 + ! [Part 7](gh-ch1-board07.png)
```

✓ BIN +57.9 KB resources/solutions/gh/ch1/gh-ch1-board.png 

✓ BIN +47.2 KB resources/solutions/gh/ch1/gh-ch1-board01.png 

✓ BIN +45.7 KB resources/solutions/gh/ch1/gh-ch1-board02.png 

✓ BIN +20.5 KB resources/solutions/gh/ch1/gh-ch1-board03.png 

✓ BIN +21.5 KB resources/solutions/gh/ch1/gh-ch1-board04.png 

⌄ BIN +24.4 KB resources/solutions/gh/ch1/gh-ch1-board05.png 

⌄ BIN +27.9 KB resources/solutions/gh/ch1/gh-ch1-board06.png 

✓ BIN +22.6 KB resources/solutions/gh/ch1/gh-ch1-board07.png 

✓ 5 ██████ resources/solutions/gh/ch2/.github/CODEOWNERS 

 This CODEOWNERS file is valid.

```
... @@ -0,0 +1,5 @@
```

```
1 + /apis/poi/ @DariuszPorowski
2 + /apis/trips/ @DariuszPorowski
3 + /apis/user-java/ @DariuszPorowski
4 + /apis/userprofile/ @DariuszPorowski
5 + /iac/ @DariuszPorowski ⊖
```

▽ 44 resources/solutions/gh/ch2/README.md 

```
... @@ -0,0 +1,44 @@
```

```
1 + # Challenge 2
2 +
3 + This challenge expects participants to
   use both the GitHub interface and
   configuration in code to accomplish the
   goals.
4 +
5 + ## Mandatory reviews for `main` and
   prevent direct pushes
6 +
7 + 1. No direct push to `main`
8 + 1. At least two reviewers
9 + 1. Require code owner's review
10 + 1. Commit history not visible in `main`
    after merging
11 +
12 + Go to the repository within the GitHub
   organization and click **Settings**.
13 +
14 + In **Branches** click **Add rule**.
15 +
16 + ![[GH protection rules](gh-ch2-protection-
   rules1.png)]
17 +
18 + Enter `main` to the **Branch name
   pattern** field and set the rule to
   require pull requests before merging,
   with 2 approvals and review from Code
   Owners. Ideally, this should be enforced
   for administrators too.
19 +
20 + ![[GH protection rules](gh-ch2-protection-
   rules2.png)]
21 +
```

```
22 + This effectively disables the ability to
23 + push directly to `main`.
24 +
25 +
26 + Additionaly, it's possible to change what
27 + merge options are available on the
28 + repository level (so across all branches)
29 + - in the **Settings** screen of the repo,
30 + scroll down and see the **Merge button**
31 + section:
32 +
33 +
34 + (gh-ch2-protection-
35 + rules3.png)
36 +
37 + ## Code owners per service
38 +
39 + 1. Define code ownership per API and IaC
40 +
41 + Create a `CODEOWNERS` file in the
42 + `.github` folder within repository root.
43 +
44 + The contents should specify paths to
45 + individual components (individual APIs
46 + and IaC) and respective owner accounts.
47 +
48 + ```text
49 + /apis/poi/ @DariuszPorowski
50 + /apis/trips/ @msimecek
51 + /apis/user-java/ @DariuszPorowski
52 + @msimecek
53 + /apis/userprofile/ @DariuszPorowski
54 + /iac/ @DariuszPorowski
55 + ```
56 +
57 + Usernames have to begin with `@` and
58 + multiple users are separated with a
59 + space.
```

✓ BIN +20.9 KB resources/solutions/gh/ch2/gh-ch2-protection-rules1.png 

✓ BIN +79.2 KB resources/solutions/gh/ch2/gh-ch2-protection-rules2.png 

✓ BIN +21.5 KB resources/solutions/gh/ch2/gh-ch2-protection-rules3.png 

✓ 115 ██████ resources/solutions/gh/ch3/.github/workflows/workflow.bicep.deploy.old.yml 

... @@ -0,0 +1,115 @@

```
1 + name: "Deploy - Bicep"
2 +
3 + on:
4 +   push:
5 +     branches:
6 +       - main
7 +     paths:
8 +       - "iac/bicep/**"
9 +   workflow_dispatch:
10 +
11 + # Set envs
12 + env:
13 +   WORKDIR: "iac/bicep"
14 +
15 + # Set defaults for GitHub Actions runner
16 + defaults:
17 +   run:
18 +     working-directory: "iac/bicep"
19 +
20 + jobs:
21 +   preview:
22 +     name: "Preview Changes"
23 +     runs-on: ubuntu-latest
24 +     outputs:
25 +       resources-prefix: ${{ steps.inputdata.outputs.resources-prefix }}
```

```
        }}

26 +
27 +   steps:
28 +     # Checkout the repository to the
29 +     # GitHub Actions runner
30 +     - name: Checkout
31 +       uses: actions/checkout@v2
32 +
33 +     # Generate RESOURCES_PREFIX based
34 +     # on the repo name
35 +     - name: "Get Resources Prefix"
36 +       run: |
37 +         RESOURCES_PREFIX=$(echo '${
38 +           toJson(github.repository) }' | jq -r -c
39 +           'split("/")')[1]')
40 +
41 +       # Set for current Job
42 +       echo
43 +         "RESOURCES_PREFIX=${RESOURCES_PREFIX}"
44 +         >> $GITHUB_ENV
45 +
46 +       # Set for next Job
47 +       echo "::set-output
48 +         name=resources-
49 +         prefix::${RESOURCES_PREFIX}"
50 +
51 +       # Login to Azure with Service
52 +       # Principal
53 +       - name: "Login to Azure"
54 +         uses: Azure/login@v1
55 +         with:
56 +           creds: ${{
57 +             secrets.AZURE_CREDENTIALS }}
58 +
59 +       # Preview changes
60 +       - name: "Run what-if"
61 +         uses: Azure/cli@v1
62 +         with:
63 +           inlineScript: |
64 +             az deployment sub what-if \
65 +               --name ${{
66 +                 github.run_id
67 +               }} \
68 +               --template-file ${{
69 +                 env.WORKDIR }}/main.bicep \
70 +               --location "${{
71 +                 env.WORKDIR }}'
```

```
      secrets.LOCATION }}}" \
59      --parameters
      resourcesPrefix=${{ env.RESOURCES_PREFIX
}}}
60
61  +  deploy:
62  +    name: "Deploy"
63  +    needs: preview
64  +    runs-on: ubuntu-latest
65  +    environment: production
66  +    outputs:
67  +      hostnames: ${{
      steps.deployment.outputs.hostnames }}
68 +
69  +    steps:
70  +      # Checkout the repository to the
      GitHub Actions runner
71  +      - name: Checkout
72  +        uses: actions/checkout@v2
73 +
74  +      # Login to Azure with Service
      Principal
75  +      - name: "Azure Login"
76  +        uses: Azure/login@v1
77  +        with:
78  +          creds: ${{
      secrets.AZURE_CREDENTIALS }}
79 +
80  +      # Deploy
81  +      - name: "Run deployment"
82  +        uses: Azure/cli@v1
83  +        id: deployment
84  +        with:
85  +          inlineScript: |
86  +            deployment_output=$(az
      deployment sub create \
87  +              --name ${{
      github.run_id
}}) \
88  +              --template-file ${{
      env.WORKDIR }}/main.bicep \
89  +              --location "${{
      secrets.LOCATION }}"
90  +              --parameters
      resourcesPrefix=${{
      needs.preview.outputs.resources-prefix
}})
```

```

91      +           hostnames=$(echo
92      +             "${deployment_output}" | jq -r -c
93      +               '.properties.outputs | map(.value) |
94      +                 join(",")')
95      +           echo "::set-output
96      +             name=hostnames::${hostnames}"
97      +
98      +   test:
99      +     name: "Test"
100     +     needs: deploy
101     +     runs-on: ubuntu-latest
102     +
103     +     # Checkout the repository to the
104     +       GitHub Actions runner
105     +       - name: Checkout
106     +         uses: actions/checkout@v2
107     +
108     +     # Run Pester based smoke test
109     +       - name: "Run smoke tests"
110     +         shell: pwsh
111     +         run: ./smokeTest.ps1 -HostNames
112     +           ${needs.deploy.outputs.hostnames }
113     +
114     +     # Publish JUnit Test Results
115     +       - name: Publish test results
116     +         if: success() || failure()
117     +         uses: dorny/test-reporter@v1.5.0
118     +         with:
119     +           name: Test results
120     +           path: "${env.WORKDIR
121     +             }}/testResultsJUnit.xml"
122     +           reporter: java-junit

```

✓ 86 resources/solutions/gh/ch3/.github/workflows/workflow.bicep.deploy.yml 

... @@ -0,0 +1,86 @@

```

1  + name: "Deploy - IaC (Bicep)"
2  +
3  + # run on push to main branch only
4  + on:
5  +   push:
6  +     branches:
7  +       - main
8  +     paths:
9  +       - "iac/bicep/**"
10 + workflow_dispatch:

```

```
11 +  
12 + # Set envs  
13 + env:  
14 +   WORKDIR: "iac/bicep"  
15 +   # RESOURCES_PREFIX: "devopsoh44707" #  
16 +     hardcoded or dynamic based on repo name  
17 +   # LOCATION: "westus2" # hardcoded or  
18 +     get from secrets  
19 +  
20 + # Set defaults for GitHub Actions runner  
21 + defaults:  
22 +   run:  
23 +     working-directory: "iac/bicep"  
24 +  
25 +   jobs:  
26 +     preview:  
27 +       name: "Preview Changes"  
28 +       runs-on: ubuntu-latest  
29 +     outputs:  
30 +       RESOURCES_PREFIX: ${  
31 +         steps.resources_prefix.outputs.result }  
32 +  
33 +     steps:  
34 +       # Checkout the repository to the  
35 +       # GitHub Actions runner  
36 +         - name: Checkout  
37 +           uses: actions/checkout@v2  
38 +  
39 +         # Get RESOURCES_PREFIX based on the  
40 +         # repo name  
41 +         - name: Get repo name  
42 +           uses: actions/github-script@v5  
43 +           id: resources_prefix  
44 +           with:  
45 +             result-encoding: string  
46 +             script: return  
47 +               context.repo.repo.toLowerCase()  
48 +  
49 +         # Login to Azure with Service  
50 +         Principal  
51 +         - name: "Login to Azure"  
52 +           uses: Azure/login@v1  
53 +           with:  
54 +             creds: ${  
55 +               secrets.AZURE_CREDENTIALS }  
56 +
```

```
49 +     # Preview changes
50 +     - name: "What-If"
51 +       uses: Azure/cli@v1
52 +       with:
53 +         inlineScript: |
54 +           az deployment sub what-if \
55 +             --name ${{ github.run_id }}
56 +             \
57 +             --template-file ${{
58 +               env.WORKDIR }}/main.bicep \
59 +               --location "${{
60 +                 secrets.LOCATION }}" \
61 +                   --parameters
62 +                     resourcesPrefix=${{
63 +                       steps.resources_prefix.outputs.result }}
64 +
65 +
66 +     deploy:
67 +       name: "Deploy"
68 +       needs: preview
69 +       runs-on: ubuntu-latest
70 +       environment: production
71 +
72 +     steps:
73 +       # Checkout the repository to the
74 +       # GitHub Actions runner
75 +       - name: Checkout
76 +         uses: actions/checkout@v2
77 +
78 +       # Login to Azure with Service
79 +       # Principal
80 +       - name: "Azure Login"
81 +         uses: Azure/login@v1
82 +         with:
83 +           creds: ${{
84 +             secrets.AZURE_CREDENTIALS }}
```

```
85      +           --location "${{  
86      +             secrets.LOCATION }}}" \  
86      +           --parameters  
87      +             resourcesPrefix=${{  
88      +               needs.preview.outputs.RESOURCES_PREFIX }}
```

▽ 61 resources/solutions/gh/ch3/.github/workflows/workflow.bicep.pr.yml

... @@ -0,0 +1,61 @@

```
1  + name: "PR - IaC (Bicep)"  
2  +  
3  + # run on pr to main branch only  
4  + on:  
5  +   pull_request:  
6  +     branches:  
7  +       - main  
8  +     paths:  
9  +       - "iac/bicep/**"  
10 +   workflow_dispatch:  
11 +  
12 + # Set envs  
13 + env:  
14 +   WORKDIR: "iac/bicep"  
15 +   # RESOURCES_PREFIX: "devopsoh44707" #  
16 +   # LOCATION: "westus2" # hardcoded or  
17 +   get from secrets  
18 + # Set defaults for GitHub Actions runner  
19 + defaults:  
20 +   run:  
21 +     working-directory: "iac/bicep"  
22 +  
23 + jobs:  
24 +   codequalitycheck:  
25 +     name: "Code Quality Check"  
26 +     runs-on: ubuntu-latest  
27 +     steps:  
28 +       # Checkout the repository to the  
29 +       # GitHub Actions runner  
30 +       - name: Checkout  
31 +         uses: actions/checkout@v2  
32 +       # Get RESOURCES_PREFIX based on the  
33 +       # repo name  
34 +       - name: Get repo name
```

```

34   +     uses: actions/github-script@v5
35   +     id: resources_prefix
36   +     with:
37   +       result-encoding: string
38   +       script: return
39   +         context.repo.repo.toLowerCase()
40   +
41   +       # Login to Azure with Service
42   +       Principal
43   +         - name: Azure Login
44   +           uses: Azure/login@v1
45   +           with:
46   +             creds: ${{ secrets.AZURE_CREDENTIALS }}
47   +
48   +       # Checks that all Bicep
49   +       configuration files adhere to a canonical
50   +       format
51   +
52   +       - name: Bicep Lint
53   +         uses: Azure/cli@v1
54   +         with:
55   +           inlineScript: az bicep build --
56   +             file ${ env.WORKDIR }/main.bicep
57   +
58   +       # Validate whether a template is
59   +       valid at subscription scope
60   +
61   +       - name: Bicep Validate
62   +         uses: Azure/cli@v1
63   +         with:
64   +           inlineScript: |
65   +             az deployment sub validate \
66   +               --name ${ github.run_id } \
67   +               \
68   +               --template-file ${ env.WORKDIR }/main.bicep \
69   +               \
70   +               --location "${ secrets.LOCATION }" \
71   +               \
72   +               --parameters
73   +               resourcesPrefix=${ steps.resources_prefix.outputs.result }

```

✓ 128 resources/solutions/gh/ch3/.github/workflows/workflow.terraform.deploy.old.yaml 

... @@ -0,0 +1,128 @@

```

1 + name: "Deploy - Terraform"
2 +

```

```
3 + on:
4 +   push:
5 +     branches:
6 +       - main
7 +     paths:
8 +       - "iac/terraform/**"
9 +   workflow_dispatch:
10 +
11 + # Using env secrets is the preferred
12 +   approach for CI/CD tools. AZ login only
13 +   works for user accounts and not for
14 +   service principals.
15 +
16 + env:
17 +   ARM_CLIENT_ID: "${{
18 +     secrets.ARM_CLIENT_ID }}"
19 +   ARM_CLIENT_SECRET: "${{
20 +     secrets.ARM_CLIENT_SECRET }}"
21 +   ARM_SUBSCRIPTION_ID: "${{
22 +     secrets.ARM_SUBSCRIPTION_ID }}"
23 +   ARM_TENANT_ID: "${{
24 +     secrets.ARM_TENANT_ID }}"
25 +   WORKDIR: "iac/terraform"
26 +
27 + # Set defaults for GitHub Actions runner
28 + defaults:
29 +   run:
30 +     working-directory: "iac/terraform"
31 +
32 + jobs:
33 +   preview:
34 +     name: "Preview Changes"
35 +     runs-on: ubuntu-latest
36 +     outputs:
37 +       tfplan-exitcode: ${{
38 +         steps.tfplan.outputs.exitcode }}
39 +       resources-prefix: ${{
40 +         steps.inputdata.outputs.resources-prefix
41 +       }}
42 +
43 + steps:
44 +   # Checkout the repository to the
45 +   # GitHub Actions runner
46 +   - name: Checkout
47 +     uses: actions/checkout@v2
48 +
49 +   # Generate RESOURCES_PREFIX based
```

```
    on the repo name
38 +      - name: "Get Resources Prefix"
39 +        run: |
40 +          RESOURCES_PREFIX=$(echo '${{ toJson(github.repository) }}' | jq -r -c
41 +            'split("/")[1]')
42 +          # Set for current Job
43 +          echo
44 +            "RESOURCES_PREFIX=${RESOURCES_PREFIX}"
45 +            >> $GITHUB_ENV
46 +          # Set for next Job
47 +          echo "::set-output
48 +            name=resources-
49 +            prefix::${RESOURCES_PREFIX}"
50 +          id: inputdata
51 +          #
52 +          # Install the latest version of
53 +          # Terraform CLI
54 +          - name: Setup Terraform
55 +            uses: hashicorp/setup-
56 +              terraform@v1
57 +              #
58 +              # Initialize a new or existing
59 +              # Terraform working directory by creating
60 +              # initial files, loading any remote state,
61 +              # downloading modules, etc.
62 +              - name: Terraform Init
63 +                run: terraform init -backend-
64 +                  config=resource_group_name="${
65 +                    secrets.TFSTATE_RESOURCES_GROUP_NAME }"
66 +                  -backend-
67 +                  config=storage_account_name="${
68 +                    secrets.TFSTATE_STORAGE_ACCOUNT_NAME }"
69 +                  -backend-config=container_name="${
70 +                    secrets.TFSTATE_STORAGE_CONTAINER_NAME
71 +                  }" -backend-config=key="${
72 +                    secrets.TFSTATE_KEY }"
73 +                  #
74 +                  # Generates an execution plan for
75 +                  # Terraform
76 +                  - name: Terraform Plan
77 +                    run: terraform plan
78 +                    # An exit code of 2 indicates
79 +                    # there were changes detected and will
80 +                    # throw an error. Just continue for now,
81 +                    # we'll check next step
```

```
59 +      - name: Terraform Plan
60 +        run: terraform plan -detailed-
61 +          exitcode -var="location=${{ secrets.LOCATION }}"
62 +          var="resources_prefix=${{ env.RESOURCES_PREFIX }}"
63 +          || exit 0
64 +      id: tfplan
65 +
66 +    deploy:
67 +      name: "Deploy"
68 +      if: needs.preview.outputs.tfplan-
69 +          exitcode == 2
70 +      needs: preview
71 +      runs-on: ubuntu-latest
72 +      environment: production
73 +      outputs:
74 +        hostnames: ${{ steps.tfoutput.outputs.hostnames }}
75 +
76 +    steps:
77 +      # Checkout the repository to the
78 +      # GitHub Actions runner
79 +      - name: Checkout
80 +        uses: actions/checkout@v2
81 +
82 +      # Login to Azure with Service
83 +      # Principal
84 +      # This step is not required by
85 +      # Terraform itself, but Terraform code
86 +      # contains resources with "az" commands to
87 +      # execute on agent locally.
88 +      - name: Azure Login
89 +        uses: Azure/login@v1
90 +        with:
91 +          creds: ${{ secrets.AZURE_CREDENTIALS }}
```

Terraform working directory by creating initial files, loading any remote state, downloading modules, etc.

```
91      - name: Terraform Init
92      run: terraform init -backend-
93      config=resource_group_name="${
94      secrets.TFSTATE_RESOURCES_GROUP_NAME }"
95      -backend-
96      config=storage_account_name="${
97      secrets.TFSTATE_STORAGE_ACCOUNT_NAME }"
98      -backend-config=container_name="${
99      secrets.TFSTATE_STORAGE_CONTAINER_NAME }"
100     -backend-config=key="${
101     secrets.TFSTATE_KEY }"
102
103
104
105
106
107
108
109
110
111
112
```

# Terraform Apply

```
- name: Terraform Apply
run: terraform apply -auto-
approve -var="location=${{
secrets.LOCATION }}" -
var="resources_prefix=${{
needs.preview.outputs.resources-prefix
}}"
```

# Terraform Output

```
# Grab TF output values, transform
and set as Job output to use for the
next Job
```

- name: Terraform Output

```
run: |
deployment_output=$(terraform
output -json)
hostnames=$(echo
"${deployment_output}" | jq -r -c
'map(.value) | join(",")')
echo "::set-output
name=hostnames::${hostnames}"
id: tfoutput
```

+ test:

```
+ name: "Test"
+ needs: deploy
+ runs-on: ubuntu-latest
+ steps:
+   # Checkout the repository to the
GitHub Actions runner
```

```
113 +     - name: Checkout
114 +       uses: actions/checkout@v2
115 +
116 +     # Run Pester based smoke test
117 +     - name: "Run smoke tests"
118 +       shell: pwsh
119 +       run: ./smokeTest.ps1 -HostNames
120 +         ${{ needs.deploy.outputs.hostnames }}
121 +
122 +     # Publish JUnit Test Results
123 +     - name: Publish test results
124 +       if: success() || failure()
125 +       uses: dorny/test-reporter@v1.5.0
126 +       with:
127 +         name: Test results
128 +         path: "${{ env.WORKDIR
} }}/testResultsJUnit.xml"
+         reporter: java-junit
```

✓ 91 resources/solutions/gh/ch3/.github/workflows/workflow.terraform.deploy.yaml 

... @@ -0,0 +1,91 @@

```
1 + name: "Deploy - IaC (Terraform)"
2 +
3 + # run on push to main branch only
4 + on:
5 +   push:
6 +     branches:
7 +       - main
8 +     paths:
9 +       - "iac/terraform/**"
10 +   workflow_dispatch:
11 +
12 + # Using env secrets is the preferred
13 +   approach for CI/CD tools. AZ login only
14 +   works for user accounts and not for
15 +   service principals.
16 + env:
17 +   ARM_CLIENT_ID: "${{
18 +     secrets.ARM_CLIENT_ID }}"
19 +   ARM_CLIENT_SECRET: "${{
20 +     secrets.ARM_CLIENT_SECRET }}"
21 +   ARM_SUBSCRIPTION_ID: "${{
22 +     secrets.ARM_SUBSCRIPTION_ID }}"
23 +   ARM_TENANT_ID: "${{
24 +     secrets.ARM_TENANT_ID }}"
```

```
18 +   WORKDIR: "iac/terraform"
19 +   # RESOURCES_PREFIX: "devopsoh44707" #
20 +   # hardcoded or dynamic based on repo name
21 +
22 +   # Set defaults for GitHub Actions runner
23 +   defaults:
24 +     run:
25 +       working-directory: "iac/terraform"
26 +
27 +   jobs:
28 +     preview:
29 +       name: "Preview Changes"
30 +       runs-on: ubuntu-latest
31 +       outputs:
32 +         TFPLAN_EXITCODE: ${{ steps.tfplan.outputs.exitcode }}
33 +         RESOURCES_PREFIX: ${{ steps.resources_prefix.outputs.result }}
34 +
35 +     steps:
36 +       # Checkout the repository to the
37 +       # GitHub Actions runner
38 +       - name: Checkout
39 +         uses: actions/checkout@v2
40 +
41 +       # Get RESOURCES_PREFIX based on the
42 +       # repo name
43 +       - name: Get repo name
44 +         uses: actions/github-script@v5
45 +         id: resources_prefix
46 +         with:
47 +           result-encoding: string
48 +           script: return
49 +             context.repo.repo.toLowerCase()
50 +
51 +       # Install the latest version of
52 +       # Terraform CLI
53 +       - name: Setup Terraform
54 +         uses: hashicorp/setup-terraform@v1
55 +
56 +       # Initialize a new or existing
57 +       # Terraform working directory by creating
```

```
      initial files, loading any remote state,
      downloading modules, etc.

53  +      - name: Terraform Init
54  +          run: terraform init -backend-
      config=resource_group_name="${{
      secrets.TFSTATE_RESOURCES_GROUP_NAME }}"
      -backend-config=storage_account_name="${{
      secrets.TFSTATE_STORAGE_ACCOUNT_NAME }}"
      -backend-config=container_name="${{
      secrets.TFSTATE_STORAGE_CONTAINER_NAME
}}" -backend-config=key="${{
      secrets.TFSTATE_KEY }}"

55  +
56  +      # Generates an execution plan for
      Terraform
57  +      # An exit code of 2 indicates there
      were changes detected and will throw an
      error on GitHub Action. "|| exit 0"
      overrides it. Just continue for now,
      we'll check next step
58  +      - name: Terraform Plan
59  +          run: terraform plan -detailed-
      exitcode -var="location=${{
      secrets.LOCATION }}" -
      var="resources_prefix=${{
      steps.resources_prefix.outputs.result }}"
      || exit 0
60  +          id: tfplan
61  +
62  +      deploy:
63  +          name: "Deploy"
64  +          needs: preview
65  +          if:
      needs.preview.outputs.TFPLAN_EXITCODE ==
      2 # it is not part of the challenge, but
      nice to have
66  +          runs-on: ubuntu-latest
67  +          environment: production
68  +
69  +      steps:
70  +          # Checkout the repository to the
      GitHub Actions runner
71  +          - name: Checkout
72  +              uses: actions/checkout@v2
73  +
74  +          # Login to Azure with Service
```

```

Principal
75 +      # This step is not required by
    Terraform itself, but Terraform code
    contains resources with "az" commands to
    execute on agent locally.
76 +      - name: Azure Login
77 +      uses: Azure/login@v1
78 +      with:
79 +      creds: ${{{
    secrets.AZURE_CREDENTIALS }}}
80 +
81 +      # Install the latest version of
    Terraform CLI
82 +      - name: Setup Terraform
83 +      uses: hashicorp/setup-
    terraform@v1
84 +
85 +      # Initialize a new or existing
    Terraform working directory by creating
    initial files, loading any remote state,
    downloading modules, etc.
86 +      - name: Terraform Init
87 +      run: terraform init -backend-
    config=resource_group_name="${
    secrets.TFSTATE_RESOURCES_GROUP_NAME }}"
    -backend-config=storage_account_name="${
    secrets.TFSTATE_STORAGE_ACCOUNT_NAME }}"
    -backend-config=container_name="${
    secrets.TFSTATE_STORAGE_CONTAINER_NAME
}" -backend-config=key="${
    secrets.TFSTATE_KEY }"
88 +
89 +      # Terraform Apply
90 +      - name: Terraform Apply
91 +      run: terraform apply -auto-
    approve -var="location=${
    secrets.LOCATION }" -
    var="resources_prefix=${
    needs.preview.outputs.RESOURCES_PREFIX
}"

```

✓ 41 resources/solutions/gh/ch3/.github/workflows/workflow.terraform.pr.yml 

... @@ -0,0 +1,41 @@

```

1 + name: "PR - IaC (Terraform)"
2 +

```

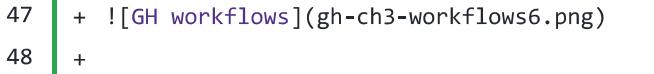
```
3 + # run on pr to main branch only
4 + on:
5 +   pull_request:
6 +     branches:
7 +       - main
8 +     paths:
9 +       - "iac/terraform/**"
10 + workflow_dispatch:
11 +
12 + # Set defaults for GitHub Actions runner
13 + defaults:
14 +   run:
15 +     working-directory: "iac/terraform"
16 +
17 + jobs:
18 +   codequalitycheck:
19 +     name: "Code Quality Check"
20 +     runs-on: ubuntu-latest
21 +     steps:
22 +       # Checkout the repository to the
23 +       # GitHub Actions runner
24 +       - name: Checkout
25 +         uses: actions/checkout@v2
26 +       # Install the latest version of
27 +       # Terraform CLI
28 +       - name: Setup Terraform
29 +         uses: hashicorp/setup-
30 +           terraform@v1
31 +       # Checks that all Terraform
32 +       # configuration files adhere to a canonical
33 +       # format
34 +       - name: Terraform Lint
35 +         run: terraform fmt -check -
36 +           recursive
37 +       # Initialize a new or existing
38 +       # Terraform working directory by creating
39 +       # initial files, loading any remote state,
40 +       # downloading modules, etc.
41 +       # for validation remote state is
42 +       # not required => -backend=false
43 +       - name: Terraform Init
44 +         run: terraform init -
45 +           backend=false
```

```
38 +  
39 +      # Validate terraform configuration  
40 +      - name: Terraform Validate  
41 +          run: terraform validate
```

▽ 75 resources/solutions/gh/ch3/README.md

... @@ -0,0 +1,75 @@

```
1 + # Challenge 3  
2 +  
3 + This challenge doesn't expect the  
4 + participants to edit the IaC code or the  
5 + application. Their task is to build  
6 + workflows for automated deployment to  
7 + Azure.  
8 +  
9 + All workflows are implemented in the  
10+ `./github/workflows` folder. There are two  
11+ versions: _Bicep_ and _Terraform_. Each  
12+ has two flavors: _deploy_ and _pr_.  
13+  
14+ ## Challenge  
15+  
16+ There are two kinds of workflows that  
17+ need to be built: _validation_ and  
18+ _deployment_. The validation workflow  
19+ should run tests on pull requests, but  
20+ not deploy anything. The deployment  
21+ itself is handled by the second workflow,  
22+ which would be triggered any time there's  
23+ a merge into the `main` branch.  
24+  
25+ Validation:  
26+  
27+ 1. Trigger validation on pull requests.  
28+ 1. Validate the IaC code with linter.  
29+  
30+ Deployment:  
31+  
32+ 1. Trigger deployment when changes in IaC  
33+ are merged into `main`.  
34+ 1. Before deploying, show a preview of  
35+ changes.  
36+ 1. Wait for approval after showing  
37+ changes.  
38+
```

```
22 + The solution is split into two files per
23 + chosen technology (Bicep or Terraform):
24 + - `workflow.x.pr.yml` implements the
25 + validation part
26 + - `workflow.x.deploy.yml` contains the
27 + deployment part
28 +
29 + > Ensure that the workflow is using the
30 + correct `RESOURCES_PREFIX` value for the
31 + team. By default the solution presented
32 + here, parses the repository name and uses
33 + it as the prefix.
34 +
35 + ### PR
36 +
37 + The PR trigger in YAML code is set up in
38 + a way that the pull request workflow is
39 + triggered automatically. Ensure that the
40 + branch and paths are correct:
41 +
42 + ```yaml
43 + on:
44 +   pull_request:
45 +     branches:
46 +       - main
47 +     paths:
48 +       - "iac/bicep/**"
49 +   workflow_dispatch:
50 +     ````
51 +
52 + To test the functionality, create a new
53 + branch, push some changes in the IaC
54 + files and create a pull request against
55 + the `main` branch. After a few seconds,
56 + the automated check should trigger:
57 +
58 + 
```

```
49 + ### Deployment
50 +
51 + Merging the pull request should trigger
   the second pipeline - deployment.
52 +
53 + Preview of changes is done by using the
   `what-if` command of Azure CLI
   deployment. This step doesn't stop and
   wait for approval on its own, it only
   lists what's going to happen during
   deployment.
54 +
55 + In order to complete this challenge, it's
   necessary to set approvals in deployment
   environment on GitHub. The solution
   provided in the `./github/workflows`
   folder expects to find an environment
   called `production`.
56 +
57 + ````yml
58 + deploy:
59 +   name: "Deploy"
60 +   needs: preview
61 +   runs-on: ubuntu-latest
62 +   environment: production
63 + ````+
64 +
65 + ![GH workflows](gh-ch3-workflows1.png)
66 +
67 + Then configure a protection rule - add
   required reviewers.
68 +
69 + ! [GH workflows](gh-ch3-workflows2.png)
70 +
71 + When the workflow is started, it will
   stop after the "Preview Changes" job and
   wait for confirmation.
72 +
73 + ! [GH workflows](gh-ch3-workflows3.png)
74 +
75 + ! [GH workflows](gh-ch3-workflows4.png)
```

✓ BIN +12 KB resources/solutions/gh/ch3/gh-ch3-workflows1.png 

✓ BIN +23.9 KB resources/solutions/gh/ch3/gh-ch3-workflows2.png 

✓ BIN +17.8 KB resources/solutions/gh/ch3/gh-ch3-workflows3.png 

▽ BIN +18.2 KB resources/solutions/gh/ch3/gh-ch3-workflows4.png 

▽ BIN +75.3 KB resources/solutions/gh/ch3/gh-ch3-workflows5.png 

✓ BIN +22.5 KB resources/solutions/gh/ch3/gh-ch3-workflows6.png 

✓ 92 ██████ resources/solutions/gh/ch4/.github/workflows/workflow.pr.api.poi.yml 

... @@ -0,0 +1,92 @@

```
1 + name: "PR - api-poi (.NET Core)"
2 +
3 + # run on pr to main branch and manually
4 + on:
```

```
5 +   pull_request:
6 +     branches:
7 +       - main
8 +     paths:
9 +       - "apis/poi/**"
10 +    workflow_dispatch:
11 +
12 + # Set envs
13 + env:
14 +   DOTNET_VERSION: "3.1.x"
15 +   WORKDIR: "apis/poi"
16 +
17 + # Set defaults for GitHub Actions runner
18 + defaults:
19 +   run:
20 +     working-directory: "apis/poi"
21 +
22 + jobs:
23 +   buildtest:
24 +     name: "Build and Test"
25 +     runs-on: ubuntu-latest
26 +     steps:
27 +       - uses: actions/checkout@v2
28 +
29 +       - name: "Setup .NET Core ${{ env.DOTNET_VERSION }}"
30 +         uses: actions/setup-dotnet@v1
31 +         with:
32 +           dotnet-version: ${{ env.DOTNET_VERSION }}
33 +
34 +       - name: "Restore dependencies"
35 +         run: dotnet restore
36 +
37 +       - name: "Build the App"
38 +         run: dotnet build --no-restore
39 +
40 +       - name: "Run Unit Tests"
41 +         id: unittest
42 +         run:
43 +           dotnet test --logger
44 +             "trx;LogFileName=UnitTestResults.trx" --
```

```
45 +         - name: "Publish test results"
46 +             if: success() || failure()
47 +             uses: dorny/test-reporter@v1
48 +             with:
49 +               name: Test results
50 +               path: "${{ env.WORKDIR
      }}}/TestResults/UnitTestResults.trx"
51 +               reporter: dotnet-trx
52 +               fail-on-error: false
53 +
54 +         - name: "Create an issue (bug)"
55 +             if: failure() &&
56 +               github.event_name != 'pull_request'
57 +             run: |
58 +               body='### Unit Test `failure`'
59 +               <details>
60 +                 <summary>Workflow
61 +                   details</summary>
62 +               Workflow name: `${{
63 +                 github.workflow }}`
64 +               Action: `${{
65 +                 github.event_name
66 +               }}` 
67 +               Job: `${{
68 +                 github.job }}` 
69 +               </details>
70 +
71 +         Test details: [Run ${{
72 +           github.run_id }}](${{
73 +             github.server_url
74 +           }}/${{
75 +             github.repository
76 +           }})/actions/runs/${{
77 +             github.run_id }})
78 +           Pusher: @${{
79 +             github.actor }}'
80 +
81 +           title='Unit Tests failure'
82 +           gh issue create --title
83 +             "${{title}}" --body "${{body}}" --label bug -
84 +             -assignee ${{
85 +               github.actor }}}
86 +           env:
87 +             GITHUB_TOKEN: ${{
88 +               secrets.GITHUB_TOKEN }}
89 +
90 +         - name: "Create a PR comment"
91 +             if: failure() &&
92 +               github.event_name == 'pull_request'
93 +             run: |
94 +               body='### Unit Test `failure`'
95 +               <details>
```

```
79 +           <summary>Workflow
80 +             details</summary>
81 +
82 +           Workflow name: `${{{
83 +             github.workflow }}`+
84 +           Action: `${{ github.event_name
85 +             }}`+
86 +
87 +           Test details: [Run #${{
88 +             github.run_id }}](${{
89 +               github.server_url
90 +             }}/${{
91 +               github.repository
92 +             }}/actions/runs/${{
93 +               github.run_id }})+
```

▽ 142 resources/solutions/gh/ch4/.github/workflows/workflow.pr.api.poiv2.yml

... @@ -0,0 +1,142 @@

```
1 + name: "PR - api-poi (.NET Core) v2"
2 +
3 + on:
4 +   pull_request:
5 +     branches:
6 +       - main
7 +     paths:
8 +       - "apis/poi/**"
9 +   workflow_dispatch:
10 +
11 + # Set envs
12 + env:
13 +   DOTNET_VERSION: "3.1.x"
14 +
15 + # Set defaults for GitHub Actions runner
16 + defaults:
17 +   run:
18 +     working-directory: "apis/poi"
19 +
20 + jobs:
```

```
21 +   buildtest:
22 +     name: "Build and Test"
23 +     runs-on: ubuntu-latest
24 +     outputs:
25 +       test_results: ${{ steps.unittestoutput.outputs.test_results }}
26 +       test_outcome: ${{ steps.unittestoutput.outputs.test_outcome }}
27 +     steps:
28 +       - uses: actions/checkout@v2
29 +
30 +       - name: "Setup .NET Core ${{ env.DOTNET_VERSION }}"
31 +         uses: actions/setup-dotnet@v1
32 +         with:
33 +           dotnet-version: ${{ env.DOTNET_VERSION }}
34 +
35 +       - name: "Restore dependencies"
36 +         run: dotnet restore
37 +
38 +       - name: "Build the App"
39 +         run: dotnet build --no-restore
40 +
41 +       - name: "Run Unit Tests"
42 +         id: unittest
43 +         run: |
44 +           dotnet test --no-build --
45 +             filter "FullyQualifiedName~UnitTest" |&
46 +             tee test_results.txt
47 +
48 +           # break if 'dotnet test' failed
49 +           test ${PIPESTATUS[0]} -eq 0
50 +           continue-on-error: true
51 +
52 +       - name: "Transform Unit Tests output"
53 +         id: unittestoutput
54 +         if: steps.unittest.outcome == 'failure'
55 +         run: |
56 +           test_results=$(cat
```

```
55 +     test_results="\${test_results//%'/'%25}"
56 +     "
57 +     test_results="\${test_results//'$'\n'/'%0A
58 +     '}"
59 +     echo "::set-output
60 +         name=test_results::\${test_results}"
61 +     echo "::set-output
62 +         name=test_outcome::\${{
63 +             steps.unittest.outcome }}"
64 +
65 +     reportbug:
66 +         name: "Report Bug for non PR runs"
67 +         needs: buildtest
68 +         if:
69 +             needs.buildtest.outputs.test_outcome ==
70 +                 'failure' && github.event_name !=
71 +                     'pull_request'
72 +             runs-on: ubuntu-latest
73 +             steps:
74 +                 - name: "Create an issue (bug)"
75 +                     uses: actions/github-
76 +                         script@v5.0.0
77 +                     env:
78 +                         UNIT_TEST_RESULTS: "\${{
79 +                             needs.buildtest.outputs.test_results }}"
80 +                     with:
81 +                         script: |
82 +                             const body = `#### Unit
83 + Tests \`$\{
84 +     needs.buildtest.outputs.test_outcome
85 + }\`"
86 +                         <details>
87 +                         <summary>Workflow
88 +                             details</summary>\n
89 +                         Workflow name - \`$\{
90 +                             github.workflow }\`\
91 +                         Action - \`$\{
92 +                             github.event_name }\`\
93 +                         Job - \`$\{
94 +                             github.job }\`\
95 +                         PR - #\$\{
96 +                             github.event.number }\`
```

```
80 +         </details>\n81 +         <details>\n82 +             <summary>Unit Tests\n83 +             details</summary>\n84 +             ````\n85 +             ${process.env.UNIT_TEST_RESULTS}\n86 +             ````\n87 +             </details>\n88 +             Pusher: @${{ github.actor }}\n89 +             Action: \`${{ github.event_name }}\n90 +             ```\n91 +             const title = `Unit Tests\n92 +             ${{ needs.buildtest.outputs.test_outcome }}`\n93 +\n94 +             github.rest.issues.create(\n95 +                 owner: context.repo.owner,\n96 +                 repo: context.repo.repo,\n97 +                 title: title,\n98 +                 body: body,\n99 +                 labels: ['bug'],\n100 +                 assignees: [context.actor]\n101 +             ))\n102 +\n103 +             # break the workflow\n104 +             - name: "Unit Tests Status"\n105 +             env:\n106 +                 UNIT_TEST_RESULTS: "${{\n107 +                     needs.buildtest.outputs.test_results }}"\n108 +             run: |\n109 +                 echo "${{\n110 +                     env.UNIT_TEST_RESULTS }}"\n111 +                 exit 1\n112 +\n113 +             reportprcomment:\n114 +                 name: "Report to PR comment fo PR\n115 +                     runs"\n116 +                 needs: buildtest\n117 +                 if:\n118 +                     needs.buildtest.outputs.test_outcome ==\n119 +                         'failure' && github.event_name ==\n120 +                             'pull_request'\n121 +                 runs-on: ubuntu-latest\n122 +\n123 +             steps:
```

```
114 +     - name: "Create a PR comment"
115 +       uses: actions/github-
116 +         script@v5.0.0
117 +       env:
118 +         UNIT_TEST_RESULTS: "${{
119 +           needs.buildtest.outputs.test_results }}"
120 +       with:
121 +         script: |
122 +           const body = `##### Unit
123 +             Tests \`${{
124 +               needs.buildtest.outputs.test_outcome
125 +             }}\`  

126 +             <details>
127 +               <summary>Details</summary>\n
128 +               \```\n
129 +             </details>\n
130 +             Pusher: @${{ github.actor
131 +               }}, Action: \`${{ github.event_name
132 +                 }}\`;;
133 +
134 +             github.rest.issues.createComment({
135 +               owner: context.repo.owner,
136 +               repo: context.repo.repo,
137 +               issue_number:
138 +                 context.issue.number,
139 +               body: body
140 +             })
141 +
142 +             # break the workflow
143 +             - name: "Unit Tests Status"
144 +               env:
145 +                 UNIT_TEST_RESULTS: "${{
146 +                   needs.buildtest.outputs.test_results }}"
147 +               run: |
148 +                 echo "${{{
149 +                   env.UNIT_TEST_RESULTS }}"
150 +               exit 1
```

94 resources/solutions/gh/ch4/.github/workflows/workflow.pr.api.trips.yml

... @@ -0,0 +1,94 @@

```
1 | + name: "PR - api-trips (GoLang)"
```

```
2 +  
3 + # run on pr to main branch and manually  
4 + on:  
5 +   pull_request:  
6 +     branches:  
7 +       - main  
8 +     paths:  
9 +       - "apis/trips/**"  
10 + workflow_dispatch:  
11 +  
12 + # Set envs  
13 + env:  
14 +   GOLANG_VERSION: "1.16"  
15 +   WORKDIR: "apis/trips"  
16 +  
17 + # Set defaults for GitHub Actions runner  
18 + defaults:  
19 +   run:  
20 +     working-directory: "apis/trips"  
21 +  
22 + jobs:  
23 +   buildtest:  
24 +     name: "Build and Test"  
25 +     runs-on: ubuntu-latest  
26 +     steps:  
27 +       - uses: actions/checkout@v2  
28 +  
29 +       - name: "Setup Go ${{  
30 +         env.GOLANG_VERSION }}"  
31 +         uses: actions/setup-go@v2  
32 +         with:  
33 +           go-version: ${{  
34 +             env.GOLANG_VERSION }}  
35 +  
36 +       - name: "Get dependencies"  
37 +         run: go get  
38 +  
39 +       - name: "Build the app"  
40 +         run: go build  
41 +  
42 +       - name: "Setup gotestsum"  
43 +         run: |  
44 +           chmod +x install_gotestsum.sh  
45 +           ./install_gotestsum.sh  
46 +  
47 +       - name: "Run Unit Tests"  
48 +         run: go test -v ./...  
49 +
```

```
46 +         run: ./gotestsum --format
47 +             standard-verbose --junitfile
48 +                 unittest_results.xml -- ./tripsgo -run
49 +                     Unit -coverprofile=unittest_coverage.out
50 +                         -covermode=count
51 +
52 +         - name: "Publish test results"
53 +             if: success() || failure()
54 +             uses: dorny/test-reporter@v1
55 +             with:
56 +                 name: Test results
57 +                 path: "${{ env.WORKDIR
58 + }}}/unittest_results.xml"
59 +             reporter: java-junit
60 +
61 +         - name: "Create an issue (bug)"
62 +             if: failure() &&
63 +                 github.event_name != 'pull_request'
64 +             run: |
65 +                 body='### Unit Test `failure`'
66 +                 <details>
67 +                     <summary>Workflow
68 +                         details</summary>
69 +                     Workflow name: `${{
70 +                         github.workflow }}`
71 +                     Action: `${{
72 +                         github.event_name
73 + }}`'
74 +                     Job: `${{
75 +                         github.job }}`'
76 +                 </details>
77 +
78 +         - name: "Test details: [Run ${{
79 +                         github.run_id }}](${{
80 +                         github.server_url
81 + }})/${{
82 +                         github.repository
83 + }}/actions/runs/${{
84 +                         github.run_id }})"
85 +             Pusher: @${{
86 +                         github.actor }}'
87 +
88 +         - name: "Create a PR comment"
89 +             title='Unit Tests failure'
90 +             gh_issue_create --title
91 +                 "${{title}}" --body "${{body}}" --label bug -
92 +                     -assignee ${{
93 +                         github.actor }}'
94 +
95 +         env:
96 +             GITHUB_TOKEN: ${{
97 +                 secrets.GITHUB_TOKEN }}'
98 +
99 +         - name: "Create a PR comment"
100 +             title='Unit Tests failure'
101 +             gh_issue_create --title
102 +                 "${{title}}" --body "${{body}}" --label bug -
103 +                     -assignee ${{
104 +                         github.actor }}'
```

```
77      +         if: failure() &&
78      +             github.event_name == 'pull_request'
79      +                 run: |
80      +                     body='### Unit Test `failure`'
81      +                         <details>
82      +                             <summary>Workflow
83      +                                 details</summary>
84      +
85      +                         Workflow name: `${
86      +                             github.workflow }`'
87      +                             Action: `${
88      +                                 github.event_name
89      +                             }`'
90      +                             Job: `${
91      +                                 github.job }`'
92      +                             PR: #${{
93      +                                 github.event.number }}'
94      +                         </details>
95      +
96      +                         Test details: [Run #${{
97      +                             github.run_id }}](${{
98      +                                 github.server_url
99      +                             }}/${{
100     +                                 github.repository
101     +                             }}/actions/runs/${{
102     +                                 github.run_id }})
103     +                         Pusher: @${{
104     +                                 github.actor }}'
105     +
106     +                         gh pr comment ${{
107     +                             github.event.number }} --body "${body}"
108     +                         env:
109     +                             GITHUB_TOKEN: ${{
110     +                                 secrets.GITHUB_TOKEN }}'
```

▽ 85 resources/solutions/gh/ch4/.github/workflows/workflow.pr.api.userjava.yml

... @@ -0,0 +1,85 @@

```
1  + name: "PR - api-userjava (Java Maven)"
2  +
3  + # run on pr to main branch and manually
4  + on:
5  +   pull_request:
6  +     branches:
7  +       - main
8  +     paths:
9  +       - "apis/user-java/**"
10 +    workflow_dispatch:
11 +
12 +    # Set envs
13 +    env:
14 +      JAVA_VERSION: "11"
15 +      WORKDIR: "apis/user-java"
```

```
16 +  
17 + # Set defaults for GitHub Actions runner  
18 + defaults:  
19 +   run:  
20 +     working-directory: "apis/user-java"  
21 +  
22 + jobs:  
23 +   buildtest:  
24 +     name: "Build and Test"  
25 +     runs-on: ubuntu-latest  
26 +     steps:  
27 +       - uses: actions/checkout@v2  
28 +  
29 +       - name: "Setup JDK ${{  
env.JAVA_VERSION }}"  
30 +         uses: actions/setup-java@v2  
31 +         with:  
32 +           java-version: ${{  
env.JAVA_VERSION }}  
33 +           distribution: "adopt"  
34 +           cache: maven  
35 +  
36 +       - name: "Run Tests with Maven"  
37 +         run: mvn test  
38 +  
39 +       - name: "Publish test results"  
40 +         uses: dorny/test-reporter@v1  
41 +         if: success() || failure()  
42 +         with:  
43 +           name: Test results  
44 +           path: "${{ env.WORKDIR  
}}/target/surefire-reports/TEST-*.xml"  
45 +           reporter: java-junit  
46 +  
47 +       - name: "Create an issue (bug)"  
48 +         if: failure() &&  
github.event_name != 'pull_request'  
49 +         run: |  
50 +           body='### Unit Test `failure`  
51 +           <details>  
52 +             <summary>Workflow  
details</summary>  
53 +  
54 +             Workflow name: `${{  
github.workflow }}`  
55 +             Action: `${{ github.event_name
```

```
    }`  
56 +     Job: `#${github.job}`  
57 +   </details>  
58 +  
59 +     Test details: [Run #${github.run_id}](${github.server_url})/${github.repository}/actions/runs/${github.run_id}  
60 +     Pusher: @$#{github.actor}'  
61 +  
62 +       title='Unit Tests failure'  
63 +       gh issue create --title  
" ${title}" --body "${body}" --label bug -  
-assignee ${github.actor}  
64 +     env:  
65 +       GITHUB_TOKEN: ${secrets.GITHUB_TOKEN}  
66 +  
67 +       - name: "Create a PR comment"  
68 +       if: failure() &&  
github.event_name == 'pull_request'  
69 +       run: |  
70 +         body='### Unit Test `failure`'  
71 +         <details>  
72 +           <summary>Workflow  
details</summary>  
73 +  
74 +           Workflow name: `#${github.workflow}`  
75 +           Action: `#${github.event_name}`  
76 +           Job: `#${github.job}`  
77 +           PR: #${github.event.number}  
78 +           </details>  
79 +  
80 +     Test details: [Run #${github.run_id}](${github.server_url})/${github.repository}/actions/runs/${github.run_id})  
81 +     Pusher: @$#{github.actor}'  
82 +  
83 +       gh pr comment ${github.event.number} --body "${body}"  
84 +     env:  
85 +       GITHUB_TOKEN: ${secrets.GITHUB_TOKEN}
```

✓ 86  resources/solutions/gh/ch4/.github/workflows/workflow.pr.api.userprofile.yml [Open](#)

[Load diff](#)

✓ 33  resources/solutions/gh/ch4/README.md [Open](#)

[Load diff](#)

✓ 126  resources/solutions/gh/ch5/.github/workflows/workflow.deploy.api.poi.yml [Open](#)

[Load diff](#)

✓ 85  resources/solutions/gh/ch5/README.md [Open](#)

[Load diff](#)

✓ BIN  resources/solutions/gh/ch5/gh-ch5-workflow1.png [Open](#)

✓ 158  resources/solutions/gh/ch6/.github/workflows/workflow.deploy.api.poi.yml [🔗](#)

 [Load diff](#)

✓ 125  resources/solutions/gh/ch6/README.md [🔗](#)

 [Load diff](#)

✓ BIN  resources/solutions/gh/ch6/gh-ch6-pat.png [🔗](#)

✓ BIN  resources/solutions/gh/ch6/gh-ch6-reposecrets.png 

▼ 95  resources/solutions/gh/ch7/.github/workflows/workflow.bicep.deploy.yml [!\[\]\(581052d0767602aebb471db3216c7f44\_img.jpg\)](#)

[Load diff](#)

▼ 147  resources/solutions/gh/ch7/.github/workflows/workflow.deploy.api.poi.ghas.yml [!\[\]\(213843e43059380df1518837f751103c\_img.jpg\)](#)

[Load diff](#)

▼ 94  resources/solutions/gh/ch7/.github/workflows/workflow.pr.api.poi.gitleaks.yml [!\[\]\(6e25d00eb33a8b510d97d22411aa9c77\_img.jpg\)](#)

[Load diff](#)

▼ 106  resources/solutions/gh/ch7/.github/workflows/workflow.terraform.deploy.yaml [🔗](#)

[Load diff](#)

▼ 79  resources/solutions/gh/ch7/README.md [🔗](#)

[Load diff](#)

▼ 8  resources/solutions/gh/ch8/README.md [🔗](#)

[Load diff](#)