**SonarQube**

Tuesday, May 28, 2024

9:26 AM

**Installing SonarQube:**

**-------------------------------**

* + AWS Account: **lz-dev**
  + EKS Cluster: **sbx-eks-cluster**
  + SonarQube Installation Document:
    - <https://artifacthub.io/packages/helm/sonarqube/sonarqube>

**SonarQube Dev Settings:**

**-----------------------------------**

<https://sonarqube.sbx.lz.us-cert.gov/>

**Default Sonarqube username:** admin

**Default Sonarqube password:** admin

**New Password:** SonarqubePassword1

**Registered E-mail:** khaled.elsayed@associates.cisa.dhs.gov

**Adding SonarQube Helm Chart:**

**--------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ✗ ❯ helm repo add sonarqube** [**https://SonarSource.github.io/helm-chart-sonarqube**](https://SonarSource.github.io/helm-chart-sonarqube)

"sonarqube" has been added to your repositories

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ✗ ❯ helm repo update**

Hang tight while we grab the latest from your chart repositories...

...Successfully got an update from the "sonarqube" chart repository

...Successfully got an update from the "grafana" chart repository

...Successfully got an update from the "cloudbees" chart repository

...Successfully got an update from the "jfrog" chart repository

Update Complete. ⎈Happy Helming!⎈

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ✗ ❯ helm repo list**

NAME         URL

cloudbees        <https://charts.cloudbees.com/public/cloudbees>

jfrog         <https://charts.jfrog.io>

grafana         <https://grafana.github.io/helm-charts>

**sonarqube**        [**https://SonarSource.github.io/helm-chart-sonarqube**](https://SonarSource.github.io/helm-chart-sonarqube)

**Search for the Latest Helm Version for SonarQube:**

**----------------------------------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Scripts & Logs/Test-1 ❯ helm search repo sonarqube --versions**

NAME         CHART VERSION        APP VERSION        DESCRIPTION

sonarqube/sonarqube         10.5.1+2816         10.5.1         SonarQube is a self-managed, automatic code rev...

sonarqube/sonarqube         10.5.0+2748         10.5.0         SonarQube is a self-managed, automatic code rev...

sonarqube/sonarqube         10.4.1+2389         10.4.1         SonarQube is a self-managed, automatic code rev...

sonarqube/sonarqube         10.4.0+2288         10.4.0         SonarQube is a self-managed, automatic code rev...

sonarqube/sonarqube         10.3.0+2009         10.3.0         SonarQube is a self-managed, automatic code rev...

sonarqube/sonarqube         10.2.1+800         10.2.1         SonarQube is a self-managed, automatic code rev...

**Pulling the SonarQube Helm Chart Locally:**

**-----------------------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ✗ ❯ helm pull sonarqube/sonarqube**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ✗ ❯ ls -l**

total 144

-rw-r--r-- 1 workspaces\khaled.elsayed workspaces\domain users 143566 May 28 10:50 **sonarqube-10.5.1+2816.tgz**

**Export AWS Credentials according to the AWS Account:**

**-----------------------------------------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ❯ export AWS\_ACCESS\_KEY\_ID="ASIAXKP3V7QBOJW4SZVE"**

**export AWS\_SECRET\_ACCESS\_KEY="aeAD166zDRNfsRxApjSZHeg8XH91I80/FQS8p0sb"**

**export AWS\_SESSION\_TOKEN=""**

**Verify that you are working with the Correct AWS Account:**

**----------------------------------------------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/helm (master) ❯ aws sts get-caller-identity**

{

"UserId": "AROAXKP3V7QBAFX27APXT:kelsayed@us-cert.gov",

"Account": "**503575870466**",

"Arn": "arn:aws-us-gov:sts::503575870466:assumed-role/AWSReservedSSO\_LZ-DSO-EKS-Cluster-Admin-Role\_fa289c6a354c4b7d/kelsayed@us-cert.gov"

}

**Use the SonarQube Terraform Code:**

**----------------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/terraform ❯ terraform init**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/terraform ❯ terraform plan**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/terraform ❯ terraform apply --auto-approve**

**Terraform** was **completed successfully** as shown below.

-------------------------------------------------------------------------

helm\_release.sonarqube: Still creating... [2m30s elapsed]

helm\_release.sonarqube: Still creating... [2m40s elapsed]

helm\_release.sonarqube: Creation complete after 2m42s [id=sonarqube]

**Apply complete! Resources: 5 added, 0 changed, 0 destroyed.**

**Outputs:**

rds-db-username = <sensitive>

rds-endpoint = "sbx-sonarqube-0.cn3duv0fhocf.us-gov-west-1.rds.amazonaws.com:5432"

-------------------------------------------------------------------------

**Get All the Configurations of SonarQube:**

**----------------------------------------------------------**

**khaled.elsayed at ~/Documents/Repos/Terraform/sonarqube-master/terraform (master) ✗ ❯ kubectl get all -n sonarqube**

NAME READY STATUS RESTARTS AGE

**pod/sonarqube-sonarqube-0 1/1 Running 0 8m34s**

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

**service/sonarqube-sonarqube LoadBalancer 172.20.18.58 internal-aa83f57c26419439488e7f6a876ab7e6-621708594.us-gov-west-1.elb.amazonaws.com 443:31299/TCP 8m34s**

NAME READY AGE

**statefulset.apps/sonarqube-sonarqube 1/1 8m34s**

**SAML Integration:**

**----------------------------**

**This happens through the Network and Access Art (NAART)**

**Send an E-mail to the following people to start the SAML integration process.**

**Technical People:**

* + Keller, Robert (CTR) <ROBERT.KELLER@associates.cisa.dhs.gov>
  + Heah, Nicholas (CTR) <nicholas.heah@associates.cisa.dhs.gov>
  + Bonner, Lamarris (CTR) <lamarris.bonner@associates.cisa.dhs.gov>

**Product Owner & Proxy Product Owner:**

* + **Government Lead:**
    - RAYMOND.PIERCE@cisa.dhs.gov
    - Crawford, Annamarie <annamarie.crawford@cisa.dhs.gov>
  + **Proxy Product Owner**
    - Willars, Matthew (CTR) <matthew.willars@associates.cisa.dhs.gov>

**Scrum Masters:**

* + Ngwa, Benjamin (CTR) <BENJAMIN.NGWA@associates.cisa.dhs.gov>
  + Gupta, Gyanendra (CTR) <GYANENDRA.GUPTA@associates.cisa.dhs.gov>

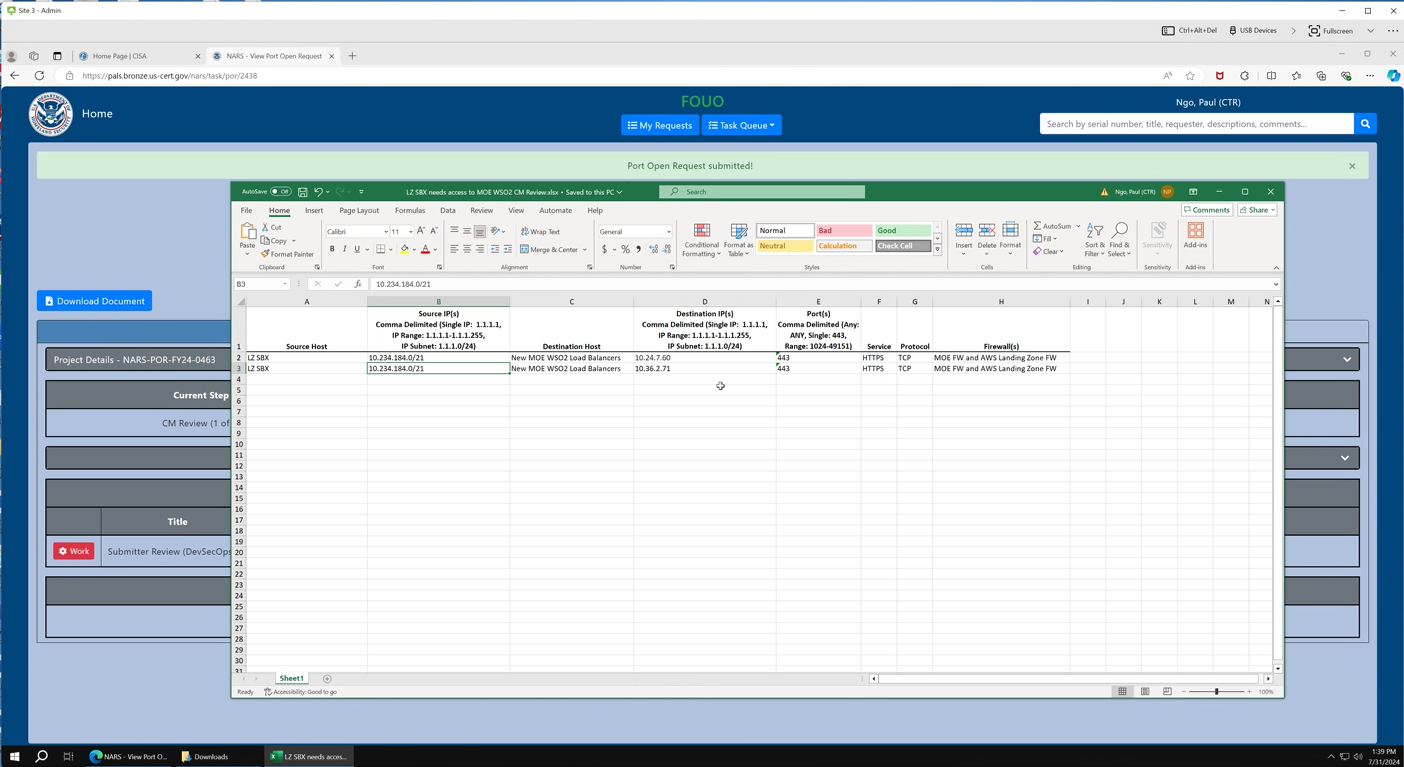
**We should ask them to help us in integrating the Pipeline Services in the SBX environment with the WSO2 SSO.**

**SAML Integration Process:**

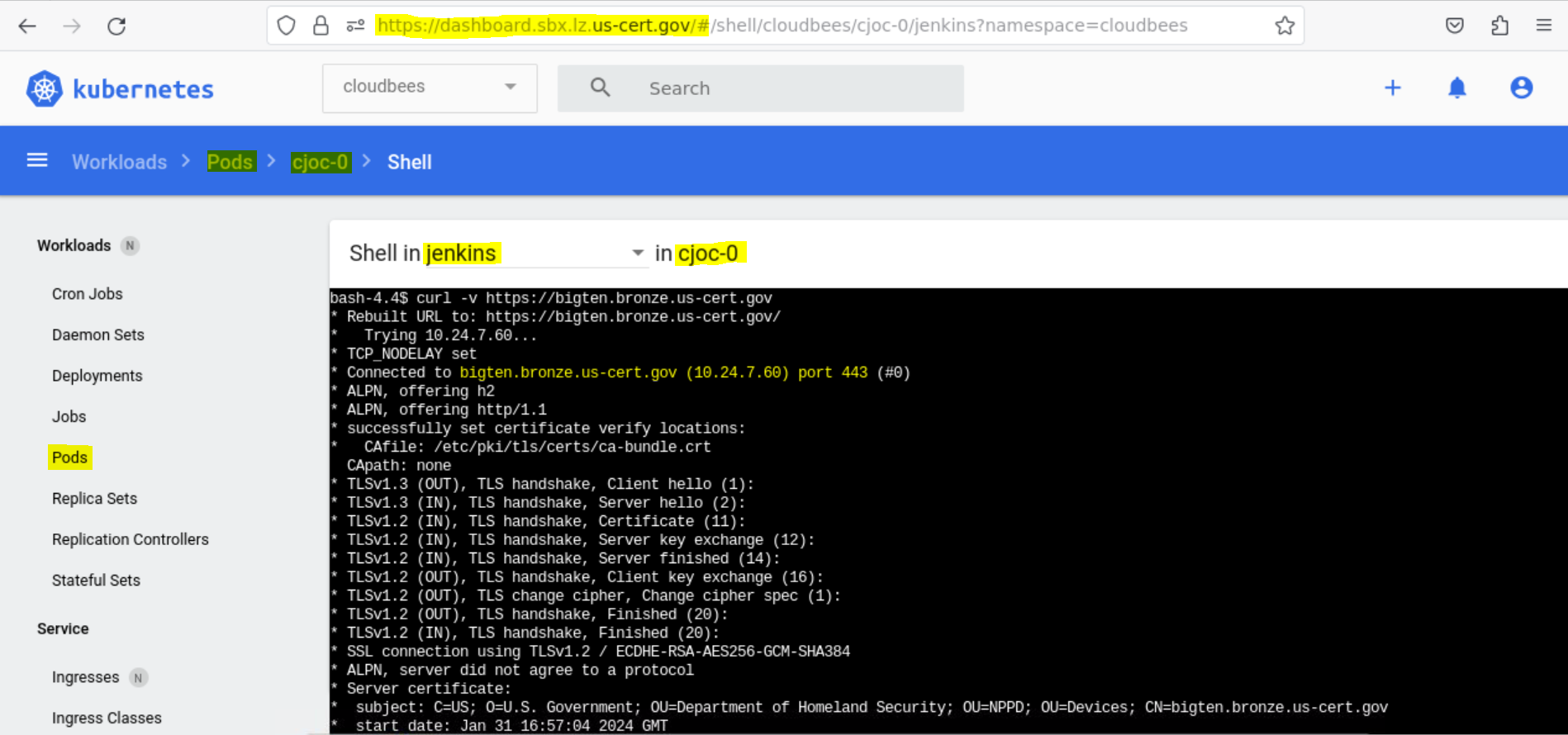
**-------------------------------------**

**We only need to submit this POR Request ONCE for All DevSecOps Services.**

* + **Submit a Port Open Request (POR) for All DevSecOps Services to open port 443 as shown below.**
  + **The POR Request need to have the following inputs in a Spreadsheet as shown below.**
    - **Source Host: LZ SBX**
    - **Source IP Address: 10.234.184.0/21**
      1. **This is the CIDR IP Range for the VPC that hosts all DevSecOps services in the lz-dev AWS account.**
    - **Destination Host: New MOE WSO2 Load Balancers**
    - **Destination IP Addresses:**
      1. **10.24.7.60**
         * **This is the IP Address of the SAML Integration Server in the MOE.**
      2. **10.36.2.71**
         * **This is the IP Address of the SAML Integration Server in the MOE.**
    - **Port: 443**
    - **Service: HTTPS**
    - **Protocol: TCP**
    - **Firewall: MOE FW and AWS Landing Zone FW**



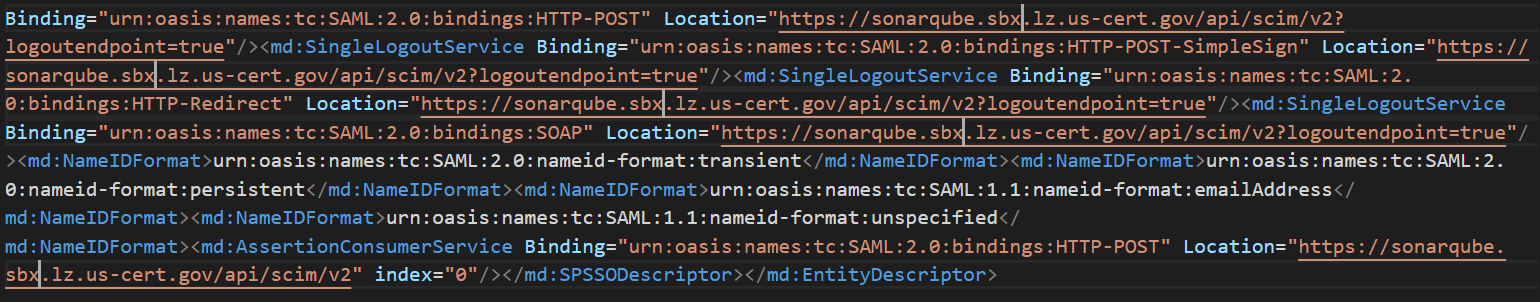
* + **Verify that Port "443" has been opened after the POR request has been completed**



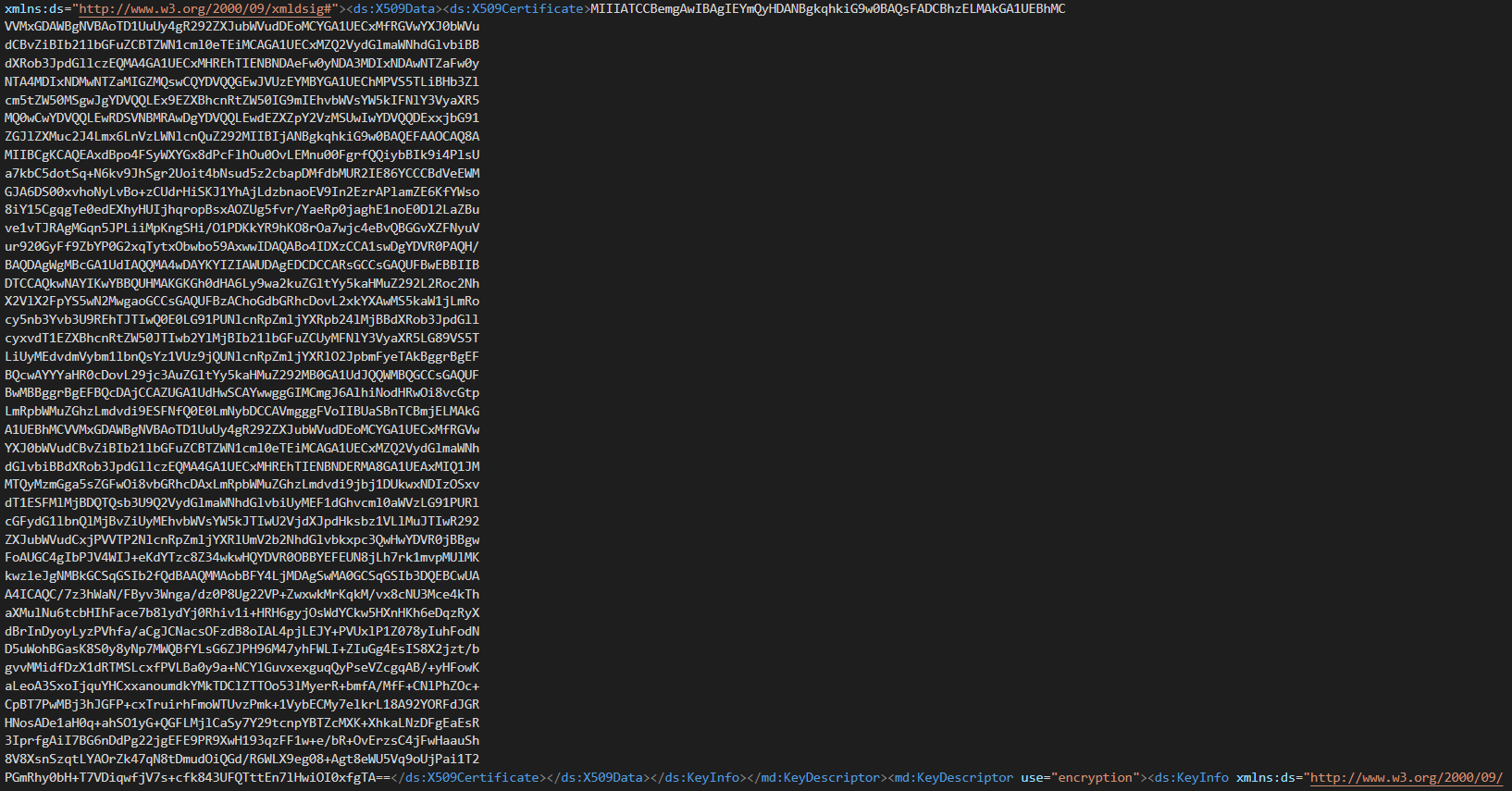
* + **Update the Metadata File for Sonarqube DevSecOps service with the following inputs**

* + **Update the URL for Artifactory Service**
    - [**https://sonarqube.sbx.lz.us-cert.gov**](https://sonarqube.sbx.lz.us-cert.gov)

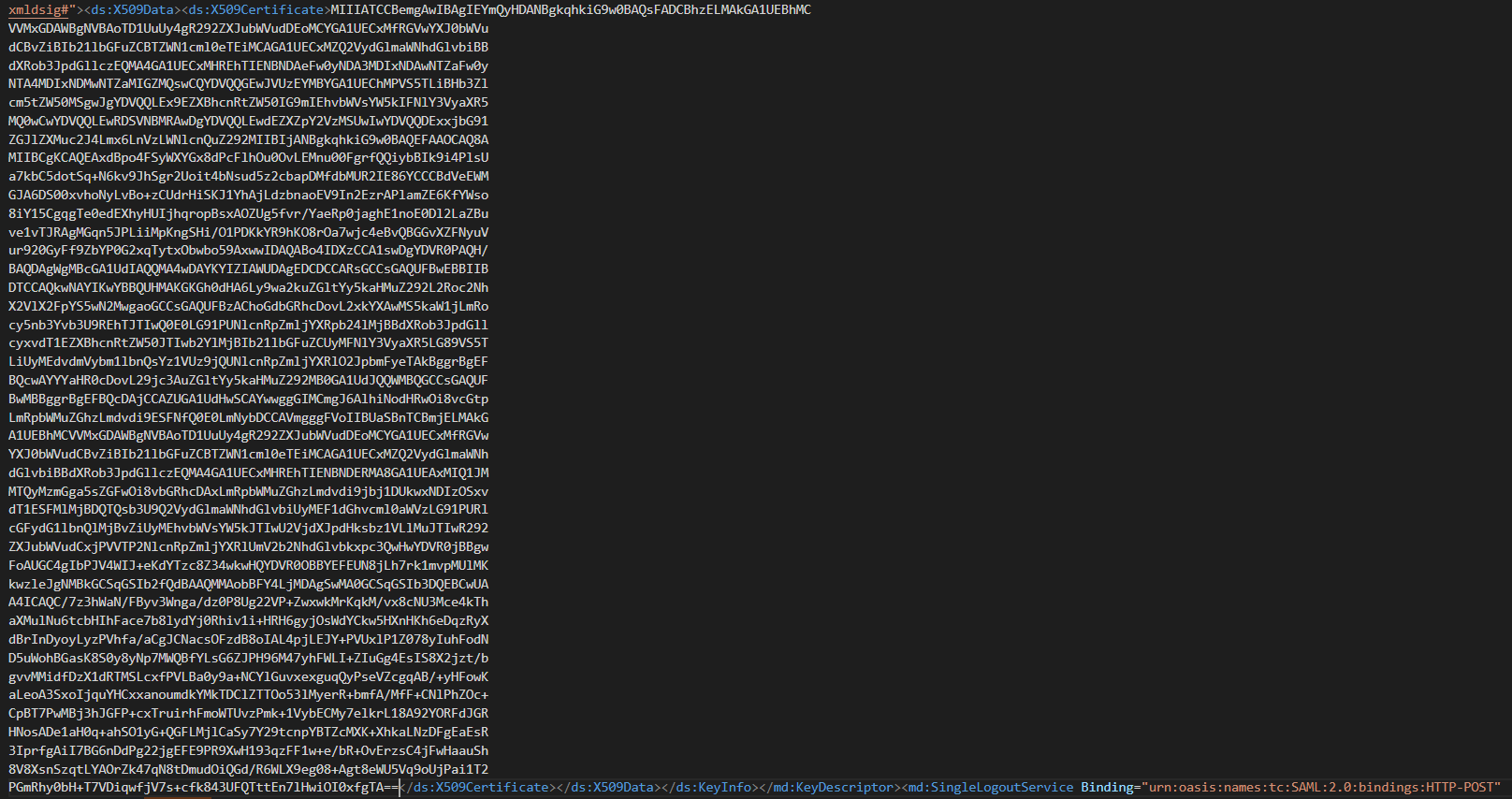




* + **Update the DHS Certificate for Sonarqube Service**
    - **This is the Same DHS Certificate that is being installed in lz-dev account in AWS Certificate Manager.**



* + **Update the DHS Certificate of Sonarqube Service in the Second Section as well.** 
    - **Note that we use the Public Certificate in both sections. The Certificate Key is NOT used at all.**

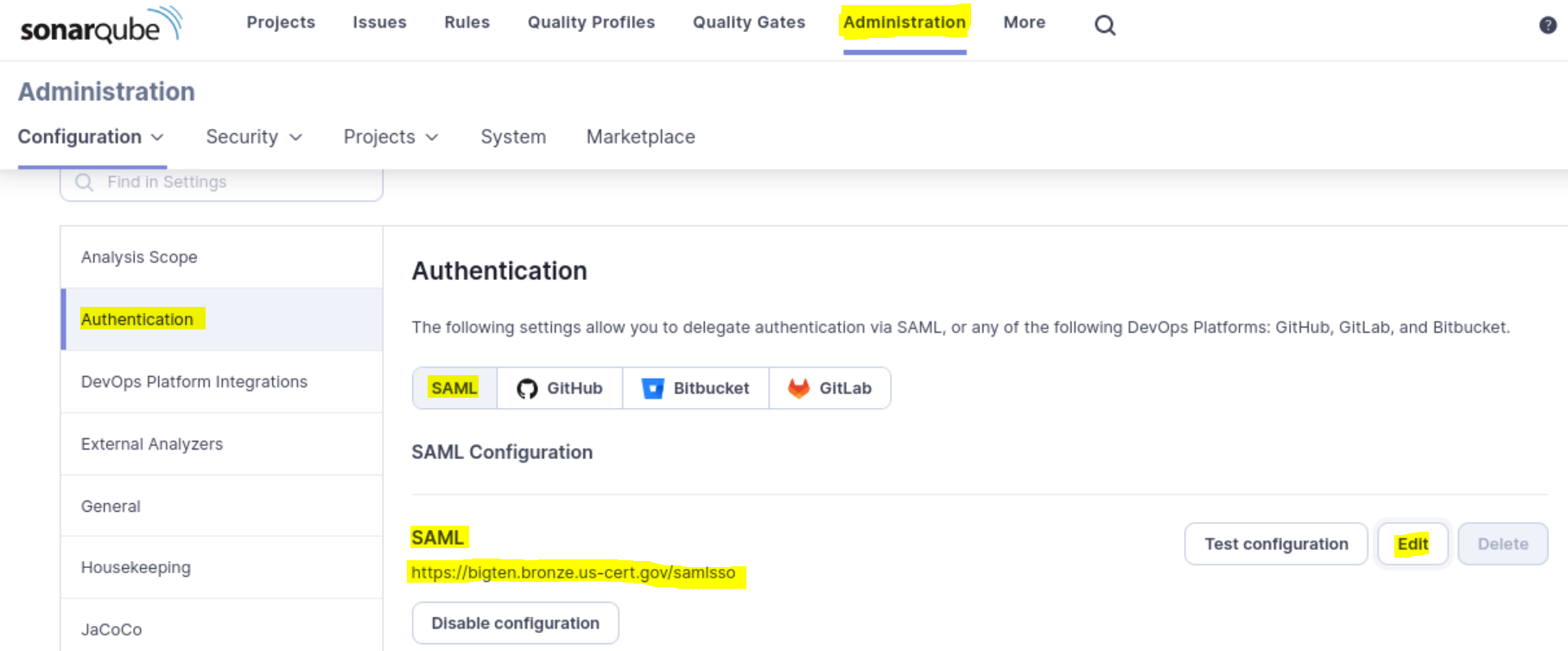


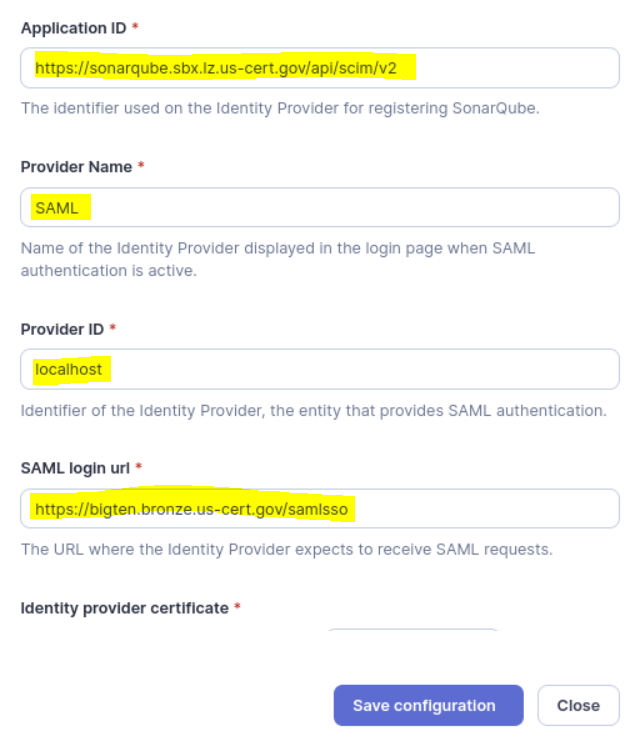
* + **Follow this Document to configure SAML for each Service.** 
    - [**https://confluence.brass.us-cert.gov/display/POPTART/LZ+Pipeline+Service+Integration+with+SSO+-+WSO2**](https://confluence.brass.us-cert.gov/display/POPTART/LZ+Pipeline+Service+Integration+with+SSO+-+WSO2)

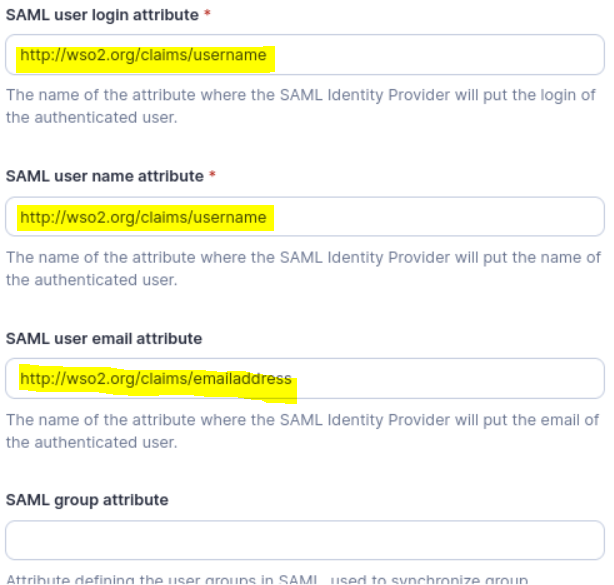
* + **Complete the Following Configuration inside Sonarqube.**

* + **Application ID:** <https://sonarqube.pb.lz.us-cert.gov/api/scim/v2>
  + **Provider Name:** SAML
  + **Provider ID:** localhost
  + **SAML login url:** <https://bigten.bronze.us-cert.gov/samlsso>
  + **SAML user login attribute:** <http://wso2.org/claims/username>
  + **SAML user name attribute:** <http://wso2.org/claims/username>
  + **SAML user email attribute:** <http://wso2.org/claims/emailaddress>

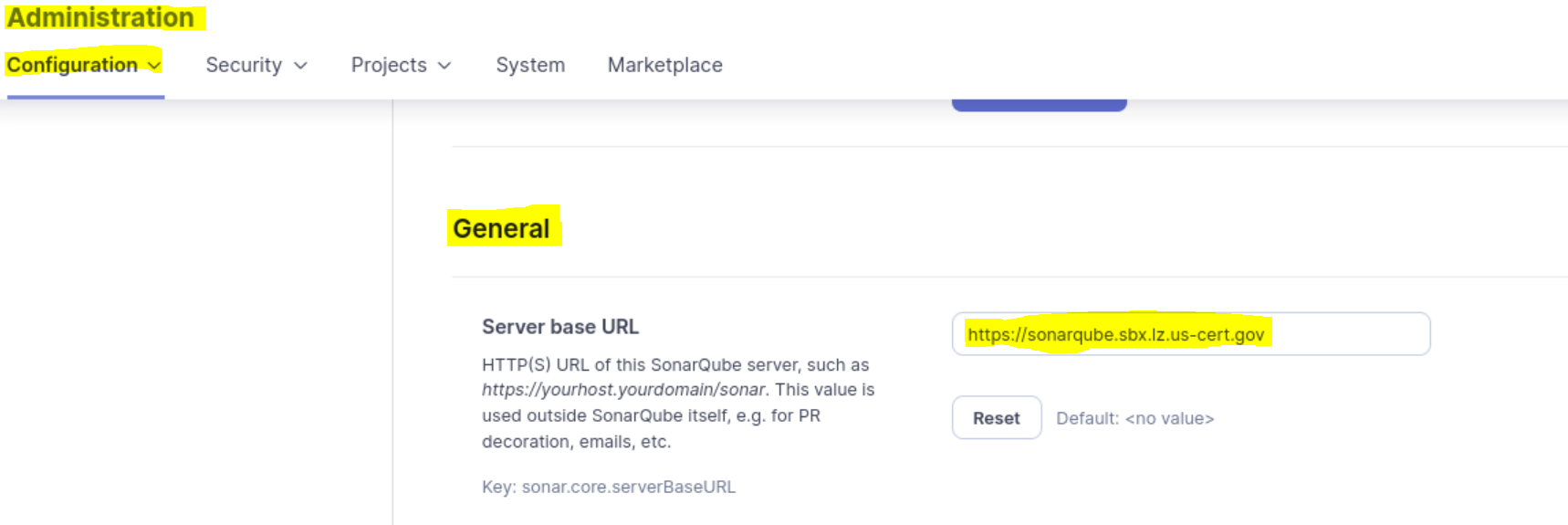
**Click on Save configuration.**







* + **Add the following configuration** 
    - **Go to Administration --> General**
    - **Server base URL:** [**https://sonarqube.sbx.lz.us-ert.gov**](https://sonarqube.sbx.lz.us-ert.gov)



**SonarQube Errors:**

**--------------------------**

aws\_security\_group.sonarqube-sg: Still destroying... [id=sg-032f70cf719698536, 14m40s elapsed]

aws\_security\_group.sonarqube-sg: Still destroying... [id=sg-032f70cf719698536, 14m50s elapsed]

aws\_security\_group.sonarqube-sg: Still destroying... [id=sg-032f70cf719698536, 15m0s elapsed]

╷

│ Error: deleting RDS Subnet Group (sbx-sonarqube-subnet-group): operation error RDS: DeleteDBSubnetGroup, https response error StatusCode: 400, RequestID: c33dc2bb-3405-4458-9011-885122762bc6, InvalidDBSubnetGroupStateFault: Cannot delete the subnet group 'sbx-sonarqube-subnet-group' because at least one database instance: sbx-sonarqube-0 is still using it.

│

│

╵

╷

│ Error: deleting RDS DB Parameter Group (sbx-sonarqube-param-group): operation error RDS: DeleteDBParameterGroup, https response error StatusCode: 400, RequestID: c0cf0719-b707-4bcc-b49e-2b1ad54cbc44, InvalidDBParameterGroupState: One or more database instances are still members of this parameter group sbx-sonarqube-param-group, so the group cannot be deleted

│

│

╵

╷

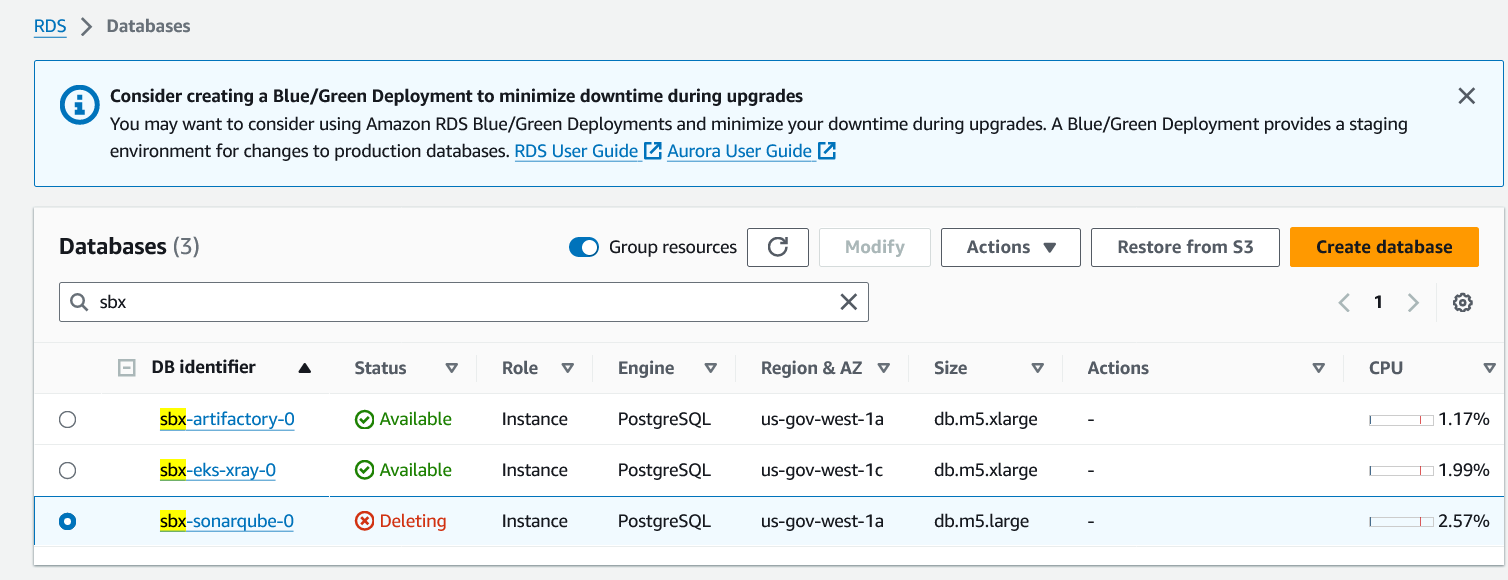
│ Error: deleting Security Group (sg-032f70cf719698536): DependencyViolation: resource sg-032f70cf719698536 has a dependent object

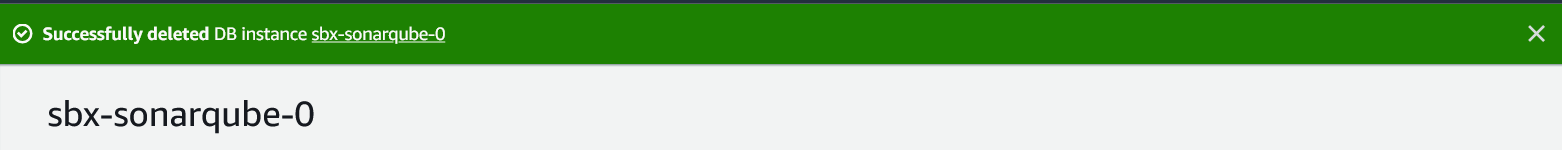
│         status code: 400, request id: b36514d5-be00-4712-87d9-1afb855457d3

│

**Solution:**

**I deleted the RDS Database manually from the AWS Console as shown below.**





**============================================================================================================================**

aws\_db\_instance.sonarqube-db: Still creating... [4m40s elapsed]

aws\_db\_instance.sonarqube-db: Still creating... [4m50s elapsed]

aws\_db\_instance.sonarqube-db: Creation complete after 5m0s [id=db-CKZAFIWKJSUSP3ZSYNLFDANIGU]

data.template\_file.sonarqube\_values: Reading...

╷

**│ Error: failed to render : <template\_file>:60,26-41: Unknown variable; There is no variable named "EKS\_BIND\_PASSWD".**

│

│ with data.template\_file.sonarqube\_values,

│ on main.tf line 124, in data "template\_file" "sonarqube\_values":

│ 124: data "template\_file" "sonarqube\_values" {

│

**Solution:**

**I defined the "EKS\_BIND\_PASSWD" environment variable in the Terraform code** in the **main.tf.**

data "template\_file" "sonarqube\_values" {

template = file("../${var.env}-sonarqube-values.yaml")

vars = {

# RDS\_USER = aws\_db\_instance.sonarqube-db.username

RDS\_DB = var.rds-db

RDS\_USER = var.rds-db-user

RDS\_PASSWD = var.rds-db-password

RDS\_URL = aws\_db\_instance.sonarqube-db.address

**EKS\_BIND\_PASSWD = var.eks-bind-password**

HTTP\_PROXY = var.http-proxy

HOST\_NAME = var.host-name

ARN\_CERTIFICATE = var.arn-certificate

}

}

**=========================================================================================================================**

**Error:**

9m27s Normal Pulled pod/sonarqube-sonarqube-0 Container image "sonarqube:10.6.0-community" already present on machine

9m26s Normal Created pod/sonarqube-sonarqube-0 Created container init-fs

9m26s Normal Started pod/sonarqube-sonarqube-0 Started container init-fs

9m26s Normal Pulled pod/sonarqube-sonarqube-0 Container image "sonarqube:10.6.0-community" already present on machine

9m25s Normal Started pod/sonarqube-sonarqube-0 Started container sonarqube

9m25s Normal Created pod/sonarqube-sonarqube-0 Created container sonarqube

9m25s Normal Pulled pod/sonarqube-sonarqube-0 Container image "sonarqube:10.6.0-community" already present on machine

**8m47s Warning Unhealthy pod/sonarqube-sonarqube-0 Startup probe failed: Get "**[**http://10.234.184.19:9000/api/system/status**](http://10.234.184.19:9000/api/system/status)**": context deadline exceeded (Client.Timeout exceeded while awaiting headers)**

**=========================================================================================================================**

**Error:**

2024.07.09 23:09:36 INFO ce[][c.z.h.HikariDataSource] HikariPool-1 - Start completed.

2024.07.09 23:09:39 INFO ce[][o.s.s.p.ServerFileSystemImpl] SonarQube home: /opt/sonarqube

2024.07.09 23:09:40 INFO ce[][o.s.c.c.CePluginRepository] Load plugins

2024.07.09 23:09:42 INFO ce[][o.s.c.c.ComputeEngineContainerImpl] Running Community edition

2024.07.09 23:09:42 INFO ce[][o.s.ce.app.CeServer] Compute Engine is started

2024.07.09 23:09:43 INFO app[][o.s.a.SchedulerImpl] Process[ce] is up

2024.07.09 23:09:43 INFO app[][o.s.a.SchedulerImpl] SonarQube is operational

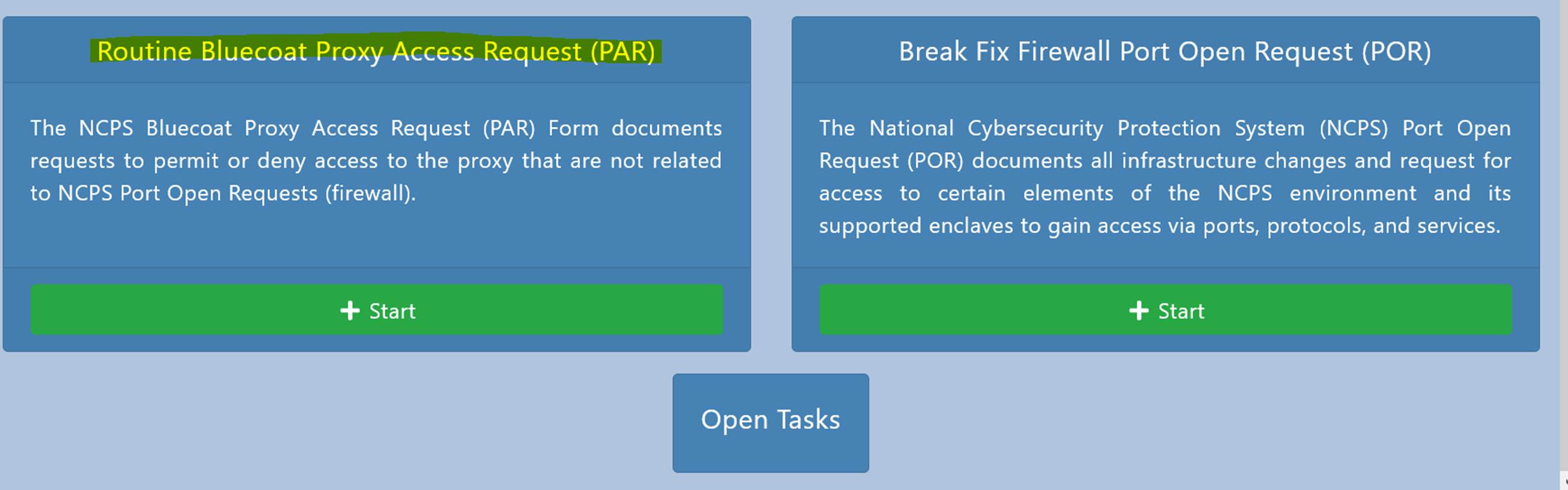
**2024.07.09 23:12:34 ERROR web[de9bc0ea-9de7-4153-8eff-7bdeffad5446][o.s.s.p.UpdateCenterClient] Fail to connect to update center**

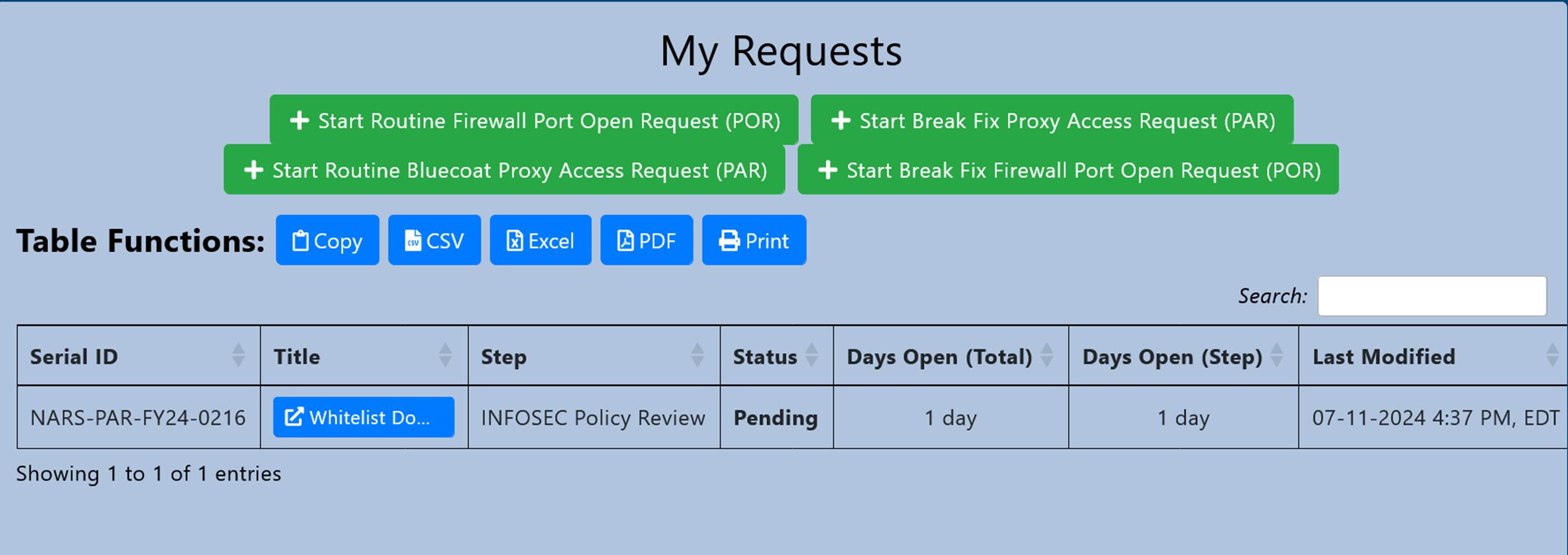
**org.sonar.api.utils.SonarException: Fail to download:** [**https://downloads.sonarsource.com/sonarqube/update/update-center.properties**](https://downloads.sonarsource.com/sonarqube/update/update-center.properties)

**at org.sonar.core.util.DefaultHttpDownloader.failToDownload(DefaultHttpDownloader.java:153)**

**Solution:**

**We need to submit a PAR request to Boundary Protection Team (Anthony Miller) to allow the Pod to reach the internet to download packages from** [**https://downloads.sonarsource.com/**](https://downloads.sonarsource.com/)

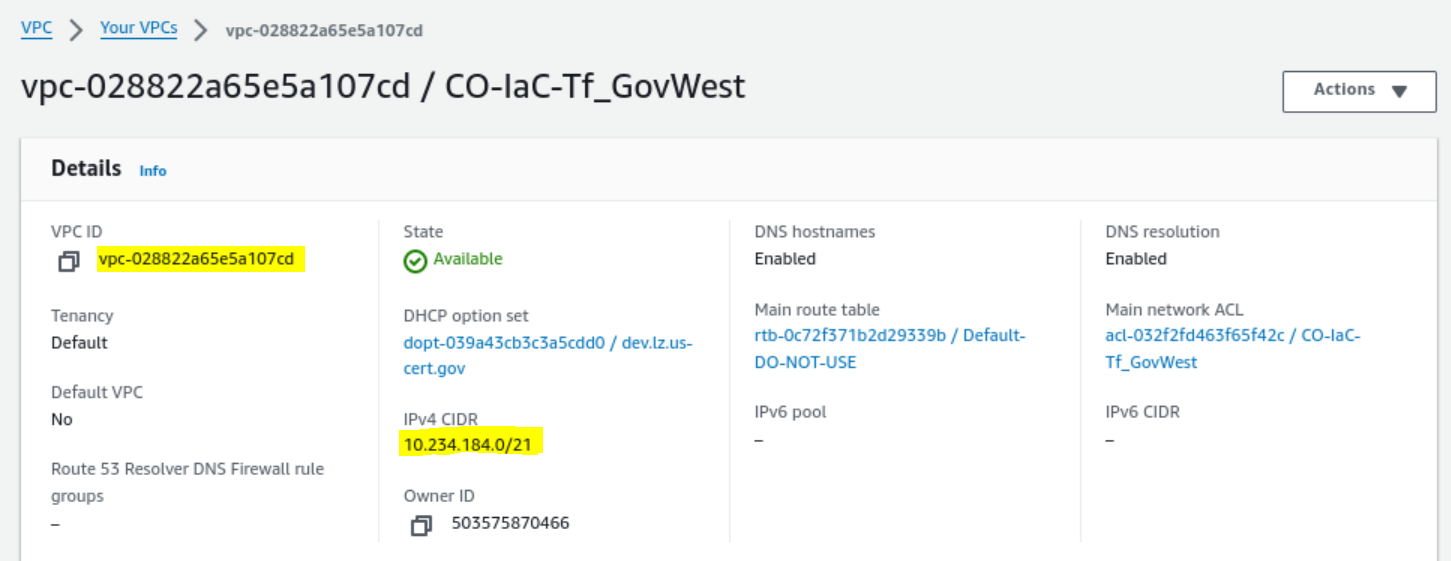




**Note: We need DONOT need to add a Security Group for Adding this CIDR IP Range to allow the Pod traffic to reach the Internet.**

**This should be solved by the above PAR Request through the Boundary Protection team, which allows the Pod to reach the internet.**

* + **10.234.184.0/21 --> This CIDR IP Range is used for the VPC that hosts the SBX EKS Cluster inside "lz-dev" account.**



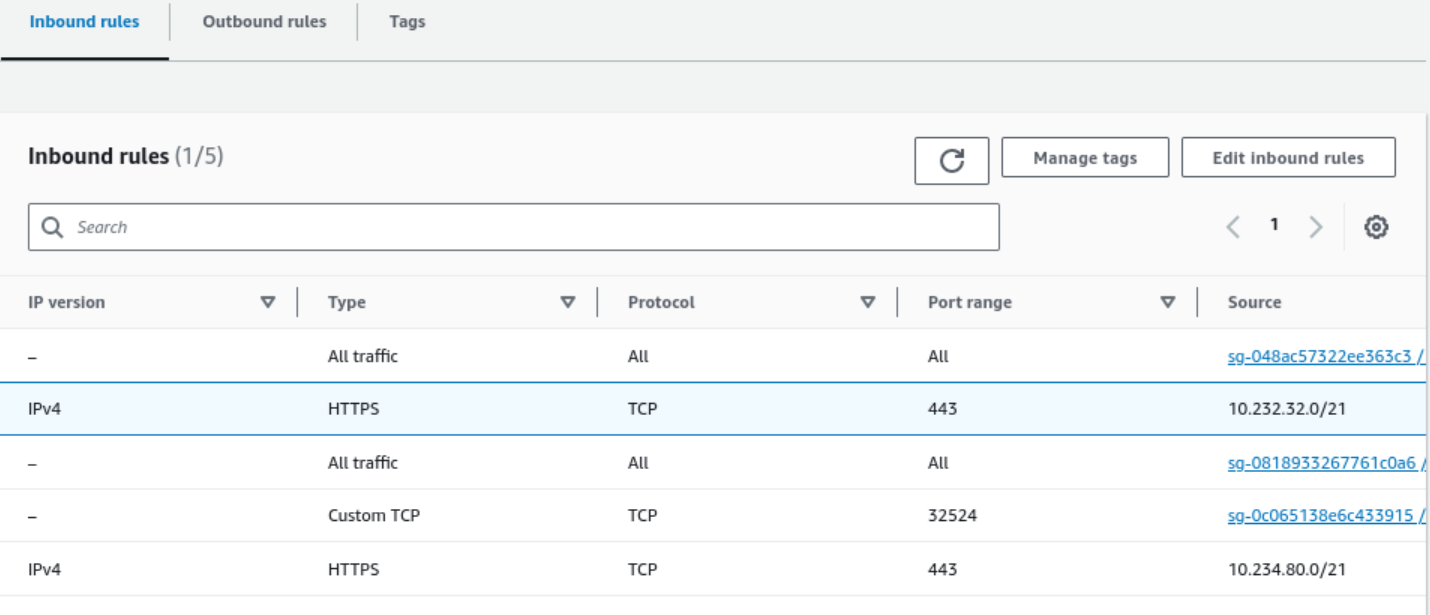
**These are the IP Addresses that are being configured during the provisioning of the EKS Cluster.**

**Our Terraform Code adds ONLY the two Ingress rules for these two subnets:**

* + **10.232.32.0/21 -->** This is the Subnet of **VPC** that hosts the "**DSO Production**" EKS Cluster, to allow **Vince** to **accept** traffic from DSO AWS Account and interact with all **DeSecOps** services that lives in the "**LZ-DSO**" AWS Account.

* + **10.234.80.0/21 -->** This is the Subnet of the **VPC** that hosts all our **AWS Workspaces**.

* + The **rest** of the **Security Group rules** are created by default, and they aren't configured in our Terraform code.
  + Also, the **Outbound rule** was created by **default**, and it isn't configured in our Terraform code.



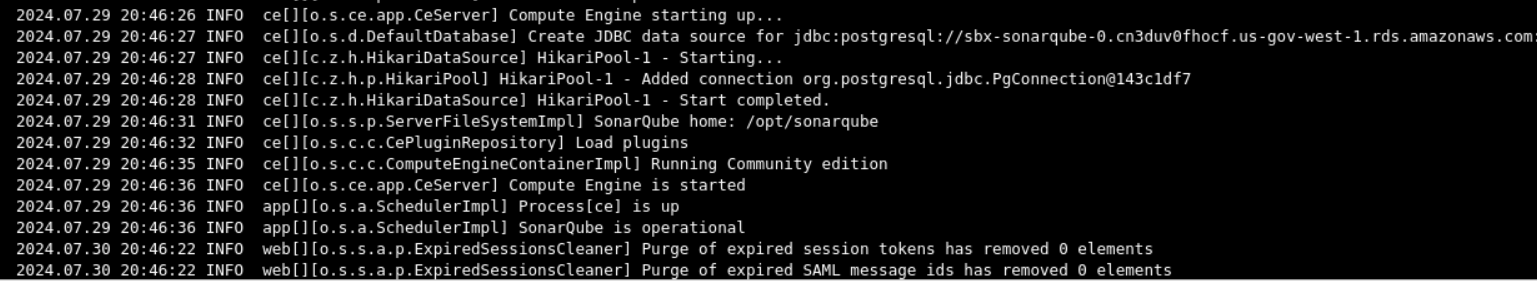


**Another Way to Fix this Issue:**

**------------------------------------------**

**Ask Anthony Miller from the Boundary Protection Team to Add this CIDR IP Range inside the PAR Request to allow the Pod to reach the internet and download any necessary packages from "**[**https://downloads.sonarsource.com**](https://downloads.sonarsource.com)**"**

* + **10.234.80.0/21**
  + **Note the main IP address that was shown inside Splunk being blocked access to reach this website is 10.234.81.16, which is the Same IP address of My Amazon Workspace.**
  + **Note:** 
    - **I asked Anthony to add this CIDR range (10.234.81.16) to the Proxy Policy as a Test, and it fixed the issue as well and I no longer see the Network Unreachable Error inside the Pod. Then, he removed that IP Address again from the Proxy Policy.**
    - **Also, We should ask to add the Whole CIDR Range. (10.234.80.0/21).**
    - **The issue was resolved by itself, without adding this IP "10.234.80.0/21" by the Boundary Protection team to the Proxy Policy.**

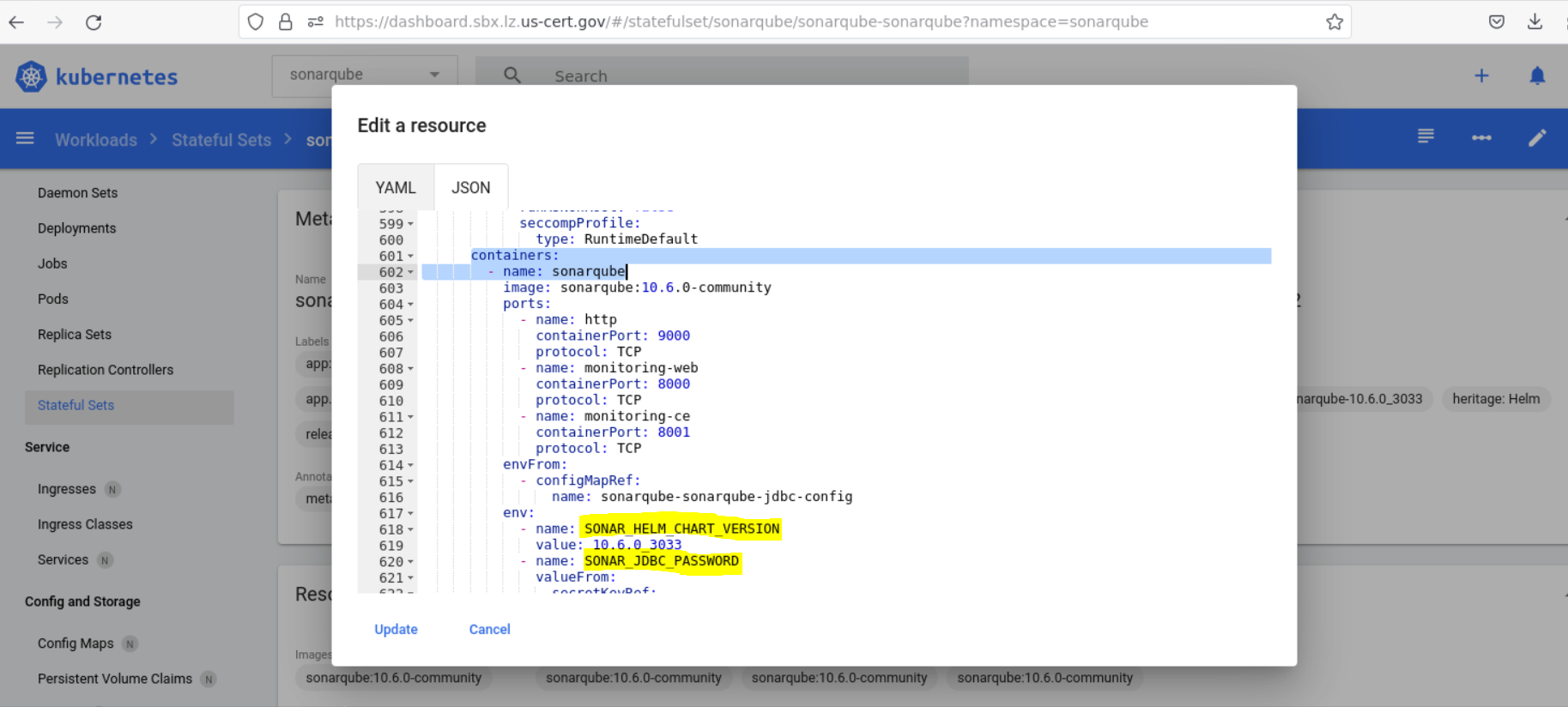


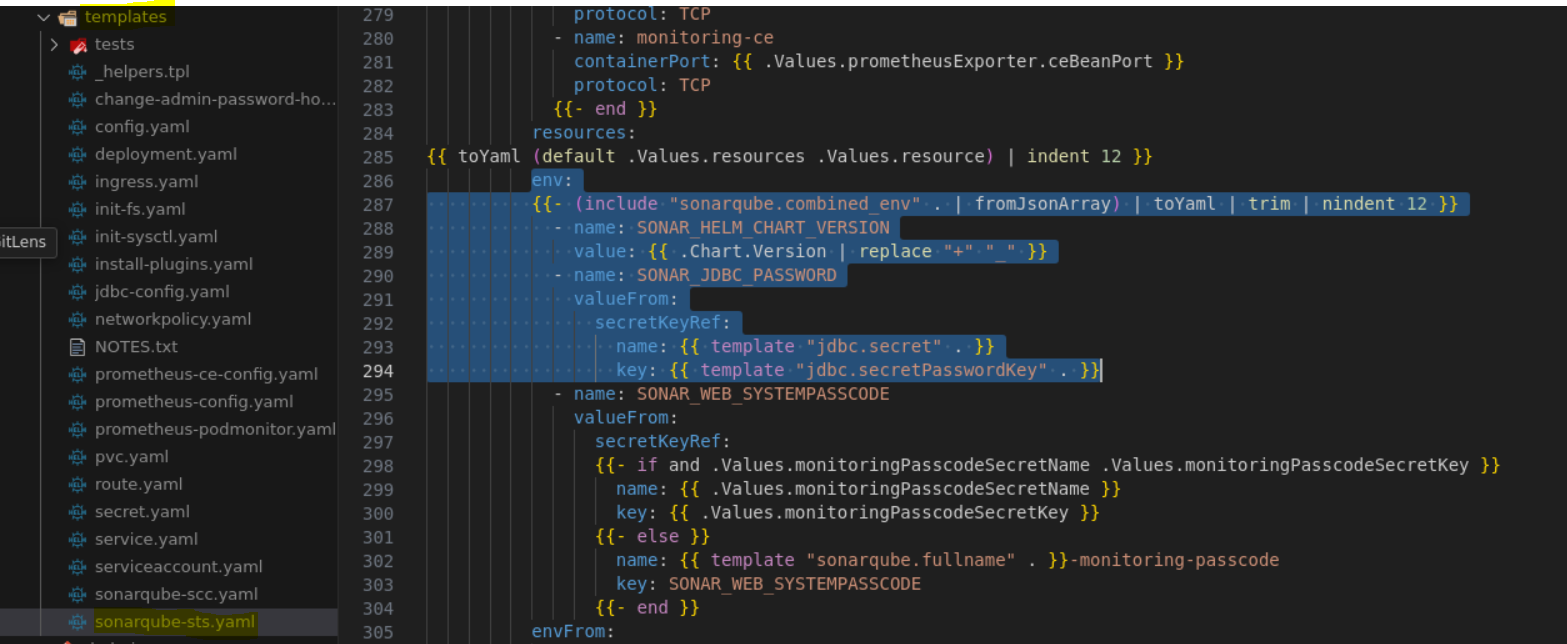
**SonarQube Helm Configuration:**

**---------------------------------------------**

**The Configurations shown below come from the Helm Templates that are found in the Sonarqube Helm Chart.**

* + **Example: This Sonarqube Statefulset Configuration is coming from the Following Helm Statefulset Helm Template "sonarqube-sts.yaml".**





**So, If you want to inject any specific configurations inside the Pod, we can do that by adding the required configurations in the Statefulset Configuration of Sonarqube.**

* + **Example:** We can **inject and add** the **Proxy configurations** inside the **Sonarqube Pod**, so that it would have access to **internet** and do **curl command**, without the need to **export** the **proxy** inside the pod before executing the curl command. **export https\_proxy=http://proxy.lz.us-cert.gov:8080.**
  + **This would allow the Pod to have this proxy configurations every time it is running.**

**- name: http\_proxy**

**value:** [**http://proxy.lz.us-cert.gov:8080**](http://proxy.lz.us-cert.gov:8080)

**- name: https\_proxy**

**value:** [**http://proxy.lz.us-cert.gov:8080**](http://proxy.lz.us-cert.gov:8080)

**- name: no\_proxy**

**value: .sbx.lz.us-cert.gov**

**This is the Statefulset File Before adding the Proxy Configurations:**

**-------------------------------------------------------------------------------------------**

**containers:**

**- name: sonarqube**

image: sonarqube:10.6.0-community

ports:

- name: http

containerPort: 9000

protocol: TCP

- name: monitoring-web

containerPort: 8000

protocol: TCP

- name: monitoring-ce

containerPort: 8001

protocol: TCP

envFrom:

- configMapRef:

name: sonarqube-sonarqube-jdbc-config

**env:**

**- name: SONAR\_HELM\_CHART\_VERSION**

**value: 10.6.0\_3033**

**- name: SONAR\_JDBC\_PASSWORD**

**valueFrom:**

**secretKeyRef:**

**name: sonarqube-sonarqube**

**key: jdbc-password**

**- name: SONAR\_WEB\_SYSTEMPASSCODE**

**valueFrom:**

**secretKeyRef:**

**name: sonarqube-sonarqube-monitoring-passcode**

**key: SONAR\_WEB\_SYSTEMPASSCODE**

**- name: SONAR\_WEB\_CONTEXT**

**value: /**

**- name: SONAR\_WEB\_JAVAOPTS**

**value: >-**

**-javaagent:/opt/sonarqube/data/jmx\_prometheus\_javaagent.jar=8000:/opt/sonarqube/conf/prometheus-config.yaml**

**- name: SONAR\_CE\_JAVAOPTS**

**value: >-**

**-javaagent:/opt/sonarqube/data/jmx\_prometheus\_javaagent.jar=8001:/opt/sonarqube/conf/prometheus-ce-config.yaml**

**This would be the Final Statefulset File After adding the Proxy Configurations:**

**-----------------------------------------------------------------------------------------------------------**

**containers:**

**- name: sonarqube**

image: sonarqube:10.6.0-community

ports:

- name: http

containerPort: 9000

protocol: TCP

- name: monitoring-web

containerPort: 8000

protocol: TCP

- name: monitoring-ce

containerPort: 8001

protocol: TCP

envFrom:

- configMapRef:

name: sonarqube-sonarqube-jdbc-config

**env:**

**- name: SONAR\_HELM\_CHART\_VERSION**

**value: 10.6.0\_3033**

**- name: SONAR\_JDBC\_PASSWORD**

**valueFrom:**

**secretKeyRef:**

**name: sonarqube-sonarqube**

**key: jdbc-password**

**- name: SONAR\_WEB\_SYSTEMPASSCODE**

**valueFrom:**

**secretKeyRef:**

**name: sonarqube-sonarqube-monitoring-passcode**

**key: SONAR\_WEB\_SYSTEMPASSCODE**

**- name: SONAR\_WEB\_CONTEXT**

**value: /**

**- name: SONAR\_WEB\_JAVAOPTS**

**value: >-**

**-javaagent:/opt/sonarqube/data/jmx\_prometheus\_javaagent.jar=8000:/opt/sonarqube/conf/prometheus-config.yaml**

**- name: SONAR\_CE\_JAVAOPTS**

**value: >-**

**-javaagent:/opt/sonarqube/data/jmx\_prometheus\_javaagent.jar=8001:/opt/sonarqube/conf/prometheus-ce-config.yaml**

**- name: http\_proxy**

**value:** [**http://proxy.lz.us-cert.gov:8080**](http://proxy.lz.us-cert.gov:8080)

**- name: https\_proxy**

**value:** [**http://proxy.lz.us-cert.gov:8080**](http://proxy.lz.us-cert.gov:8080)

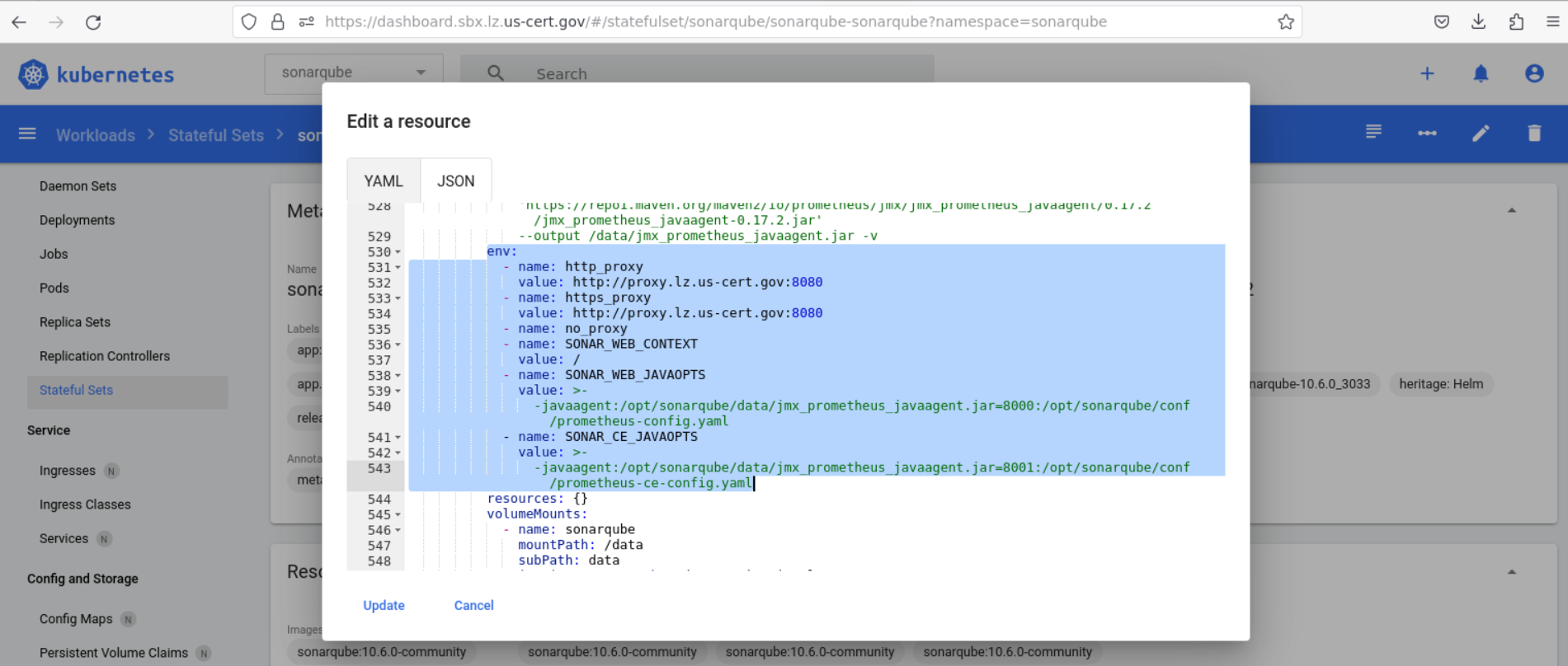
**- name: no\_proxy**

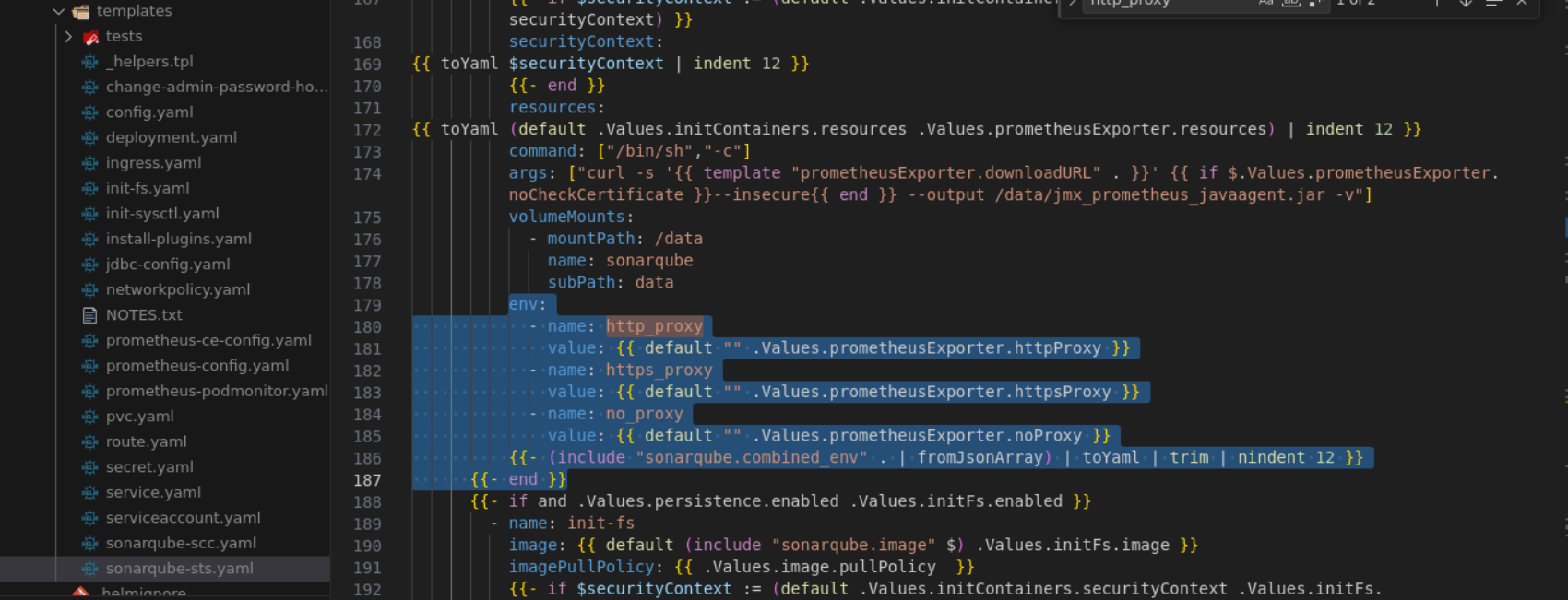
**value: .sbx.lz.us-cert.gov**

**Another Example for Helm Configuration Template:**

**-----------------------------------------------------------------------**

* + **Example: This Sonarqube Statefulset Configuration is coming from the Following Helm Statefulset Helm Template "sonarqube-sts.yaml", and "\_helpers.tpl".**





And Then those values are coming from the "**sonarqube.combined\_env**" template, which is defined in this File "**\_helpers.tpl**".

- name: SONAR\_WEB\_CONTEXT

value: /

- name: SONAR\_WEB\_JAVAOPTS

value: >-

-javaagent:/opt/sonarqube/data/jmx\_prometheus\_javaagent.jar=8000:/opt/sonarqube/conf/prometheus-config.yaml

- name: SONAR\_CE\_JAVAOPTS

value: >-

-javaagent:/opt/sonarqube/data/jmx\_prometheus\_javaagent.jar=8001:/opt/sonarqube/conf/prometheus-ce-config.yaml

