

Modify Load Balancer health probes

- [Overview](#)
- [Problem](#)
- [Solution](#)
- [Pre-requisites](#)
- [Procedure](#)
 - [Steps to create & manage Incident Task](#)
 - [Create Incident Task](#)
- [Steps](#)
- [Related articles](#)

Overview

Platform:	Azure
Owner of this SOP:	Fully Managed POD A
Cloud Services:	Instance Management

Problem

When requesters are asking to modify the load balancer health probes.

tester	Reviewer
<div>@ Ramkumar Samudram (Deactivated) - Win</div> <div>@ Asif Patel (Deactivated) - Linux</div>	

Solution

Cloud Services operators will perform basic health checks and update the associated record as required. In most cases, the record will be an Incident Task (which may be assigned or manually created).

Pre-requisites

- The operator needs to have access to the **Cloud Factory ServiceNow Dashboard**. If this access is missing raise a request for [Access to ServiceNow](#) and request the group "Cloud Factory - Dashboard Access".
- Must be a member of the "Cloud Svcs- CloudOne" assignment group to work on ServiceNow tickets. As well access to the required report in this SOP. Use the same [Access to ServiceNow](#) request to gain this group if needed.
- The operator needs to be part of **OperationsCloudOneAccess** role in AWS & **CloudOne Operator** role in Azure . If this is missing raise a [Manage Cloud Access](#) request to get this custom role.

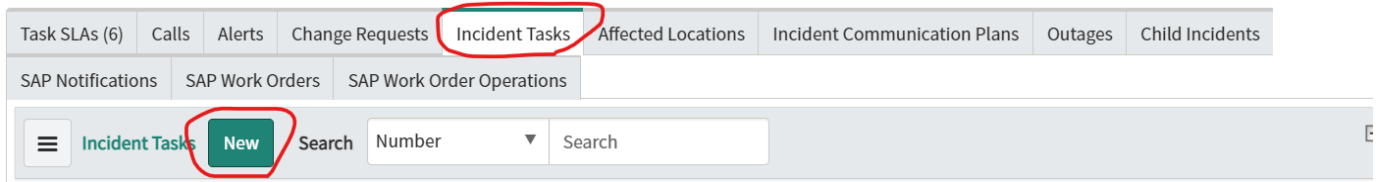
Procedure

Steps to create & manage Incident Task

Create Incident Task

Follow the below steps to create incident task. This step is required when the operator has received only an Incident.

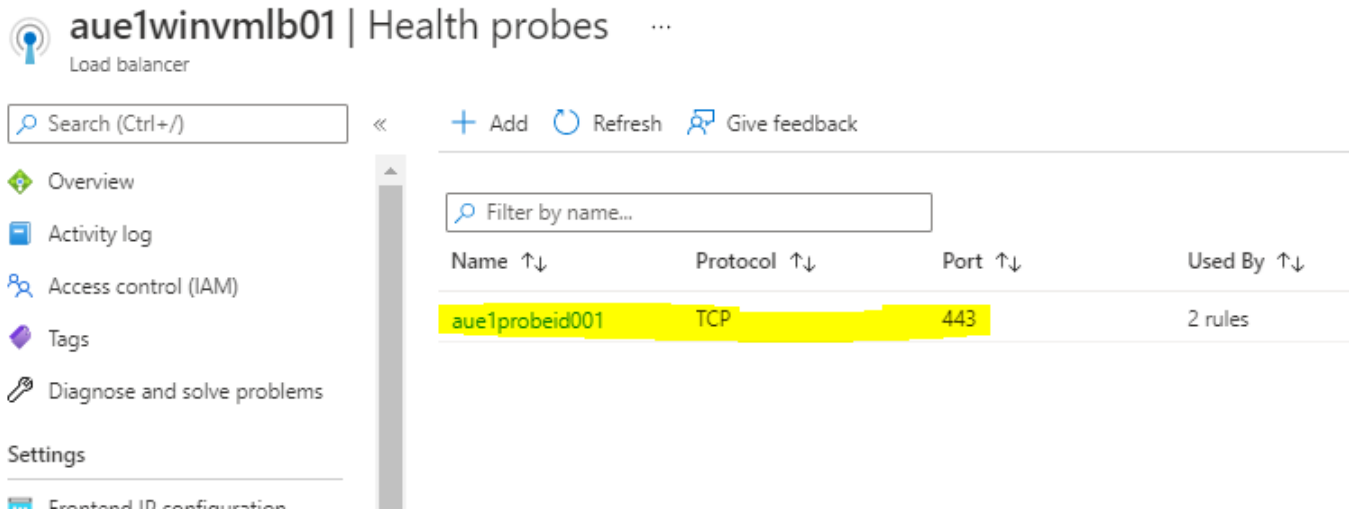
On *Incident Tasks* Tab of the Incident ticket, click *New* and assign it to "CloudOne" team. Populate the Assignment group and Configuration item (CI). (Hint the CI is the VM on which the health check is being performed)



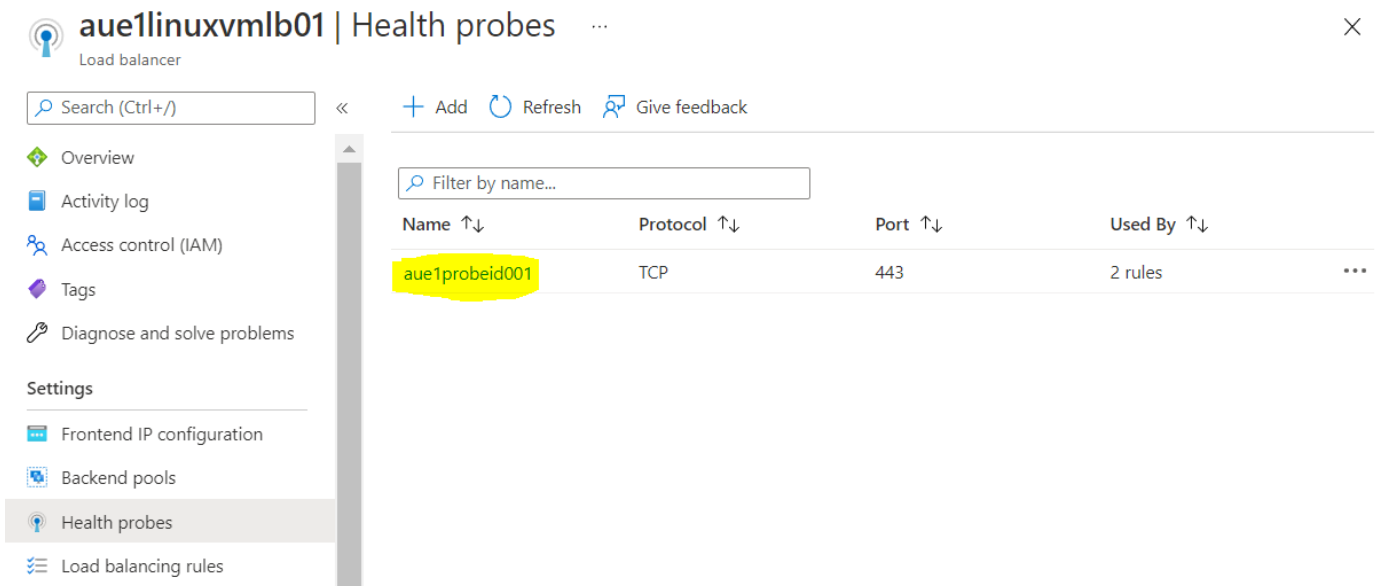
Steps

- Log on to Azure Portal and check the existing health probe.

Windows:



Linux:



- Clone the respective repo and change the required variables into .tfvars file.

Windows:

```
WINVMWITHLB
├── winvmwithlb
│   ├── .idea
│   ├── .gitignore
│   ├── doc
│   ├── env_vars
│   └── npe.auto.tfvars
├── scripts
│   ├── apply-workspace.sh
│   ├── cyberark.py
│   ├── cyberark.sh
│   ├── env.sh
│   ├── fetch_secrets.sh
│   ├── seed_creds.py
│   ├── terraform-apply.sh
│   ├── terraform-destroy.sh
│   ├── terraform-plan.sh
│   ├── update_workspace.py
│   ├── update_workspace.sh
│   ├── upload_variables.py
│   ├── validate.sh
│   └── win_secrets_inject.py
├── templates
├── .gitlab-ci.yml
├── .pre-commit-config.yaml
├── .terraform-docs.yaml
├── data.tf
├── OUTLINE
└── TIMELINE

winvmwithlb > env_vars > npe.auto.tfvars
153     name = "HTTPS" #####
154     protocol = "TCP" #####
155     frontend_ip_name = "frontendip001"
156     probe_id = "aue1probeid001"
157 },
158 "RDP" = {
159     backend_port = 3389 #####
160     frontend_port = 3389 #####
161     name = "RDP" ##### SP
162     protocol = "TCP" #####
163     frontend_ip_name = "frontendip001"
164     probe_id = "aue1probeid002"
165 },
166 }
167
168 lb_probe = {
169     "tcp" = {
170         name = "aue1probeid001"
171         protocol = "Tcp" #####
172         port = "443" #####
173         request_path = "" #####
174         interval_in_seconds = 15 #####
175         number_of_probes = 2 #####
176     }
177     "RDP" = {
178         name = "aue1probeid002"
179         protocol = "Tcp" #####
180         port = "3389" #####
181         request_path = "" #####
182         interval_in_seconds = 15 #####
183         number_of_probes = 2 #####
184     }
185 }
```

Linux:

```

env_vars > npe.auto.tfvars > lb_probe
50 }
51
52 lb_probe = {
53   "tcp" = {
54     name           = "aue1probeid001"      ### SPECIFIES THE NAME OF THE PROBE
55     protocol       = "Tcp"                 ### SPECIFIES THE PROTOCOL OF THE PROBE
56     port           = "443"                 ### PORT ON WHICH THE PROBE QUERY IS SENT
57     request_path    = ""                   ### The URI USED FOR REQUESTING THE PROBE
58     interval_in_seconds = 15               ### The interval, in seconds between two consecutive probes
59     number_of_probes = 2                   ### The number of failed probe attempts before the probe is considered failed
60   },
61   "tcp_1" = {
62     name           = "aue1probeid002"      ### SPECIFIES THE NAME OF THE PROBE
63     protocol       = "Tcp"                 ### SPECIFIES THE PROTOCOL OF THE PROBE
64     port           = "80"                 ### PORT ON WHICH THE PROBE QUERY IS SENT
65     request_path    = ""                   ### The URI USED FOR REQUESTING THE PROBE
66     interval_in_seconds = 15               ### The interval, in seconds between two consecutive probes
67     number_of_probes = 2                   ### The number of failed probe attempts before the probe is considered failed
68   }
69 }

```

- commit and push your changes
- verify that the pipeline executes as expected and that a terraform workspace is created and run
- verify the terraform plan in your TF workspace
- if you are satisfied with the plan output, Initiate the apply stage in pipeline

Policy checked
Queued from GitLab CI/CD Pipeline
CURRENT

API Integration triggered a run from API 5 minutes ago

Run Details

Plan finished 5 minutes ago

Resources: 2 to add, 2 to change, 0 to destroy

Started 5 minutes ago > Finished 4 minutes ago

+ 2 to create

~ 2 to change

Filter resources by address...

Terraform 1.1.2
Download raw log

Applied

Queued from GitLab CI/CD Pipeline

CURRENT

API Integration triggered a run from API 5 minutes ago

Run Details

Plan finished 7 minutes ago

Resources: 2 to add, 2 to change, 0 to destroy

Cost estimation finished 7 minutes ago

Resources: 6 of 47 estimated · \$371.61/mo · +\$0.00

Policy check passed 7 minutes ago

Policies: 13 passed, 0 failed

Apply finished a few seconds ago

Resources: 2 added, 2 changed, 0 destroyed

- Log on to the Azure portal and verify that whether the new port is added to the health probe as per requirement.

Windows:



aeu1winvmlb01 | Health probes ...

Load balancer



Give feedback



Overview



Activity log



Access control (IAM)



Tags



Diagnose and solve problems

Settings



Frontend IP configuration



Backend pools



Health probes



Load balancing rules



Filter by name...

Name ↑↓

Protocol ↑↓

Port ↑↓

Used By ↑↓

aeu1probeid001

TCP

443

HTTPS

aeu1probeid002

TCP

3389

RDP

Linux:



aeu1linuxvmlb01 | Health probes ...

Load balancer



Give feedback



Overview



Activity log



Access control (IAM)



Tags



Diagnose and solve problems

Settings



Frontend IP configuration



Backend pools



Health probes



Load balancing rules



Filter by name...

Name ↑↓

Protocol ↑↓

Port ↑↓

Used By ↑↓

aeu1probeid001

TCP

443

Port_443_Rule



aeu1probeid002

TCP

80

Port_80_Rule



Related articles

- [AWS VPC Endpoints configuration change](#)
- [Renewal / Upload SSL certificate for existing Application](#)
- [Monitoring GitLab Access Audit Report](#)
- [Azure Storage Account Blueprint SOP - BPAZR022](#)
- [Azure App Service FTP Credential Sharing and Reset Procedure](#)