ICE Server Upgrade for Kubernetes Deployments

Product guide for prerelease

Document version 1841, produced on Friday, September 06, 2024.

main 90adc8bf40040649230176bbdd465f6261a2d8e0

Contents

List of Tables

1 Document History

Publication Date	Product Release	Notes
May 29, 2024	3.5.1	Updated ICE Server version reference to 3.5.41629.
April 15, 2024	3.5.0	Updated ICE Server version reference to 3.5.41160. Added prerequisite step to verify that in the helm values json file, that the sole FQDN value under `iceIngress` is within square brackets [ ].
September 20, 2023	3.4.0	Updated ICE Server version reference to 3.4.30527.
July 27, 2023	3.3.0	Updated ICE Server version reference to 3.3.28975.
July 24, 2023	3.3.0	Updated version reference to 3.3.28856.
April 25, 2023	3.2.0	Updated ICE Server version reference to 3.2.26273.
February 17, 2023	3.2.0	Updated ICE Server version reference to 3.2.24516.
December 1, 2022	3.2.0	Updated ICE Server version references to 3.2.0. Renamed this document to ***ICE Server Upgrade for Kubernetes Deployments***. The former name, ***ICE Server Upgrade Guide***, now applies to a new document for ICE OS deployments.
September 26, 2022	3.1.2	Updated ICE Server version references to 3.1.2.
June 23, 2022	3.1.1	Added section: *Step 11: Reinstall ice-helm-operator*
June 9, 2022	3.1.1	Updated ICE Server version references to 3.1.20179.
May 31, 2022	3.1.1	Added section: *Step 7: Remove existing Helm operator*. Updated some commands.
April 18, 2022	3.1.1	Made updates specific to upgrading ICE Server to 3.1.1.
March 17, 2022	3.1.0	Made updates specific to upgrading ICE Server from 3.0.0 to 3.1.0.
March 15, 2022	3.1.0	Document created.

2 Introduction

ICE Server™ is the management and provisioning server component of the Instant Connect Enterprise solution. It provides administrative functions like authentication, authorization, channel management and provisioning.

Refer to this guide when upgrading your ICE Server to version 3.5.0 from 3.x.y.

2.1 Overview

The process for upgrading an Instant Connect Enterprise system involves the following:

1. Make a pre-upgrade backup of the ICE Server cluster
2. Deploy the new software version onto the cluster.
3. Request a new cluster-specific license file from Instant Connect Support and install it into the ICE Server software. The new license file may be requested ahead of time, using the existing installation identifier.

These instructions assume the reader is familiar with Kubernetes and has a working knowledge of Helm.

> **Notes:**
>
> - Please consult the ***Server Administration Guide*** *> Appendix A - General Troubleshooting* for suggestions to resolving common issues which may be encountered.
> - We recommend an ad hoc backup of the server prior to beginning troubleshooting or upgrade processes. To create an ad hoc backup, please see the ***Server Administration Guide*** *> Appendix A - General Troubleshooting > Ad Hoc Server Backup*.

3 Prerequisites

Your existing ICE Server must be on major version 2 or 3 (e.g 2.0.0, 2.2.1, 3.0.0 etc). If it is older, please upgrade to at least version 2.0.9647 before continuing further. You may verify the ICE Server version by reviewing Build Info using ICE Desktop Client.

Your existing ICE Server installation should already be in a healthy operation state, i.e.:

- At least 500Gb of available disk space on the filesystem where persistent storage is located, as the upgraded MinIO storage bucket requires 150GB. The location of the persistent storage is dependent on the storage class's configuration.

  - If your VM has only one filesystem mount, it should have 500Gb in / (root filesystem).
  - If your VM has separate partition for /var and /opt, each should have 100Gb and 300Gb available space.

> ** WARNING ** Insufficient free disk space may result in upgrade failure and/or pod eviction. Either issue requires substantial system downtime from which to recover.

- Verify all existing nodes are in Ready state:

```
kubectl get nodes -o wide
```

- Verify all existing helmreleases are in Succeeded PHASE with deployed RELEASESTATUS:

```
kubectl -n ice-release get hr
```

If any helmrelease is showing ChartFetched phase and/or failed status, you may reset the status by deleteing the ice-helm-operator pod:

```
kubectl -n ice-release delete pod -l app=helm-operator
```

> You may ignore ice-minio helm release's status at this time.

- Within the helm values json file, verify the sole FQDN value under iceIngress is within square brackets [ ].

1. Open the 3.3.0 or 3.4.0 helm values json file.
2. Find the host value under iceIngress and verify it is enclosed within square brackets [ ]. If there are brackets, then continue on with the upgrade process. In the example below, though, the brackets are missing:

```
"iceIngress": {
  "enabled": true,
  "values": {
    "host": "ice-azure-demo.icnow.app",
    ...
```

3. If the brackets are missing, add them as seen in the example below:

```
"iceIngress": {
      "enabled": true,
      "values": {
        "host": [ "ice-azure-demo.icnow.app" ],
        ...
```

4. Save the file, then continue on with the upgrade process.

> **Note:** If you accidentally perform the upgrade without performing this prerequisite step, issue the following command to delete the ICE Ingress helm release…
>
> `k -n ice-release delete hr ice-ingress`
>
> …and then perform steps 1-4 above.

4 Upgrade Steps

Follow the steps described in this section to upgrade ICE Server.

The commands described in this process should not be interrupted. Please make sure the installation host executing them does not have an idle session timeout configured that might cause a long-running command to fail.

For added security, all commands in this document should be executed from a non-root user account, with sudo access defined in `/etc/sudoers` to run specific commands as needed.

4.1 Step 1: Request new ICE Server 3.5.0 license

You should request a new ICE Server 3.5.0 license using the existing installation ID as shown on ICE Desktop. You will need to apply it when the server upgrade process is completed.

4.2 Step 2: Update the helm repository

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

Helm version 3.6.2 or newer will require the helm repository to be redefined:

```
helm repo rm ice-release-helm

helm \
  repo \
  add \
  ice-release-helm \
  https://ic.repo.dillonkane.com:443/artifactory/ice-release-helm \
  --username instantconnect-customer \
  --password sazkax-jibzuc-5pEpgi
```

Update the helm repository to show the latest, available ICE charts:

```
helm repo update
```

4.3 Step 3: Adjust existing installation parameters

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

Some existing installation parameters must be adjusted for a successful upgrade.

4.3.1 Install `jq`

Install `jq` using `apt` on Ubuntu. For RHEL, you must enable the ELRepo yum repository to use `yum` for install. `jq` is also available in Snap.

4.3.2 Save existing installation parameters

Use `helm get values` to retrieve the existing installation parameters:

```
helm -n ice-release \
    get values ice-helm-operator -o json | jq . > pre351upgrade.json
```

If this is a Geo-redundant setup, you must perform the same command on *both* DCs:

```
# From DC1
helm --kubeconfig ~/.kube/dc1 \
  -n ice-release get values ice-helm-operator -o json \
  | jq . > pre351upgrade.dc1.json

# From DC2
helm --kubeconfig ~/.kube/dc2 \
  -n ice-release get values ice-helm-operator -o json \
  | jq . > pre351upgrade.dc2.json
```

> **Note:** If using an international keyboard, the multi-line command(s) above may introduce unexpected characters. Please make sure the copy-and-pasted commands match the above.

4.3.3 Adjust the installation parameters

For each pre-upgrade json file:

4.3.3.1 Make a copy of `pre351upgrade.json` as `351upgrade.json` If this is a Geo-redundant setup, you should copy:

```
cp pre351upgrade.dc1.json 351upgrade.dc1.json
cp pre351upgrade.dc2.json 351upgrade.dc2.json
```

4.3.3.2 Review and adjust the installation parameters Please make sure the values such as FQDN, multicast interface name, and others are correct.

4.3.3.2.1 Disable ICE Logging chart If you see the following "iceLogging" block, replace the whole block to:

```
    "iceLogging": {
      "enabled": false
    }
```

Preserve any comma (,) at the last curly bracket (}), if any.

4.3.3.2.2 Disable ICE Sonobuoy chart option If you see the following "sonobuoy" block, replace the whole block to:

```
    "sonobuoy": {
      "enabled": false
    }
```

Preserve any comma (,) at the last curly bracket (}), if any.

4.3.3.2.3 Update Instant Connect Enterprise chart option If you see the following "instantConnectEnterprise" block, change the "elasticsearchOperator" value from **false** to **true**, so it looks like this:

```
      "instantConnectEnterprise": {
         "values": {
            "config": {
               "include": {
                  "elasticsearchOperator": true
...
```

4.3.3.2.4 Update ICE Monitoring chart option If you see the following "iceMonitoring" block, pay particular attention to the "prometheus-operator" line. Remove that line, and then remove any one curly bracket (}) before the ending } **,**.

```
    "iceMonitoring": {
      "values": {
        "prometheus-operator": {
          "grafana": {
            "grafana.ini": {
              "server": {
                "domain": ....
```

```
            }
          },
          "ingress": {
            "enabled": true,
            "hosts": [
              .....
            ]
          }
        }
      }
    }
  }
```

For example, the above code block should be modified to:

```
  "iceMonitoring": {
    "values": {

        "grafana": {
          "grafana.ini": {
            "server": {
              "domain": ....
            }
          },
          "ingress": {
            "enabled": true,
            "hosts": [
              .....
            ]
          }
        }

    }
  }
```

Preserve any comma (,) at the last curly bracket (}), if any.

4.3.3.2.5 Update MinIO Delete existing MinIO deployment

If necessary, please export all CDR (Call Data Records) as CSV in Grafana, as well as any custom Grafana dashboards. The upgrade process requires a larger MinIO object storage.

```
kubectl -n ice-release delete hr ice-monitoring
kubectl -n ice-release delete hr ice-minio
```

Update MinIO deployment value for Non-Geo Redundant setup

For a non-Geo Redundant setup that uses `https` to connect, add the following section into `351 upgrade.json`, immediately before the `"iceLogging": {` line. You do not need to add the

above if you are not using `https`.

```
    "iceMinio": {
      "enabled": true,
      "values": {
        "config": {
          "ingress": {
            "hosts": [
              "___CLUSTER___INGRESS___FQDN___"
            ]
          }
        }
      }
    },
```

Update MinIO deployment value for Geo Redundant setup

You must edit `351upgrade.dc1.json` and `351upgrade.dc2.json` to enable MinIO replication. The ***Geo Redundancy Guide*** has a complete, generic example on how to perform this task. The following is a simplified working example.

For DC1, modify `351upgrade.dc1.json` by adding this `iceMinio` block immediately before the `"iceLogging": {` line:

```
    "iceMinio": {
      "enabled": true,
      "values": {
        "config": {
          "ingress": {
            "hosts": [
              "___DC1___INGRESS___FQDN___"
            ]
          }
        },
        "multidatacenter": {
          "currentDC": "dc1",
          "enabled": true,
          "remoteDC": "dc2"
        }
      }
    },
```

For DC2, modify `351upgrade.dc2.json` by adding this `iceMinio` block immediately before the `"iceLogging": {` line:

```
    "iceMinio": {
      "enabled": true,
      "values": {
        "config": {
          "ingress": {
```

```
            "hosts": [
              "___DC2___INGRESS___FQDN___"
            ]
          }
        },
        "multidatacenter": {
          "currentDC": "dc2",
          "enabled": true,
          "remoteDC": "dc1"
        }
      }
    },
```

If you do not use FQDN on ICE Server, you may skip the `config` block. For example, if FQDN is not used, `351upgrade.dc2.json` may look like the following:

```
    "iceMinio": {
      "enabled": true,
      "values": {
        "multidatacenter": {
          "currentDC": "dc2",
          "enabled": true,
          "remoteDC": "dc1"
        }
      }
    },
```

4.3.3.2.6 Update ICE Minio replication option (Geo Redundant setup only) For DC1, modify `351upgrade.dc1.json`: The `iceMinio` block should look something like the following:

```
    "iceMinio": {
      "enabled": true,
      "values": {
        "config": {
          "ingress": {
            "hosts": [
              "___DC1___INGRESS___FQDN___"
            ]
          }
        },
        "multidatacenter": {
          "currentDC": "dc1",
          "enabled": true,
          "remoteDC": "dc2"
        }
      }
    }
```

Preserve any comma (,) at the last curly bracket (}), if any.

For DC2, modify `351upgrade.dc2.json`: The `iceMinio` block should look something like the following:

```
    "iceMinio": {
      "enabled": true,
      "values": {
        "config": {
          "ingress": {
            "hosts": [
              "___DC2___INGRESS___FQDN___"
            ]
          }
        },
        "multidatacenter": {
          "currentDC": "dc2",
          "enabled": true,
          "remoteDC": "dc1"
        }
      }
    }
```

Preserve any comma (,) at the last curly bracket (}), if any.

4.3.3.3 Verify the `351upgrade.json` file

You may verify the correctness of `351upgrade.json` using `jq`:

```
# non-Geo
jq . 351upgrade.json

# Geo - DC1
jq . 351upgrade.dc1.json

# Geo - DC2
jq . 351upgrade.dc2.json
```

There should not be any syntax error reported. Otherwise, repeat the above steps to make sure the necessary changes are made correctly.

4.4 Step 4: Remove ICE Logging chart

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

ICE Logging chart was deprecated in ICE Server 2.2.0. You will need to remove it for upgrade to work correctly:

```
kubectl -n ice-release delete hr ice-logging
```

You should see the pods in `ice-logging` namespace being removed.

4.5 Step 5: Save existing ICE Server data

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

The upgrade process updates the existing ICE Server database. It is recommended to save the existing database with `kubectl` to ensure you have a backup copy of the data.

4.5.1 Cassandra database

Flush each Cassandra node in the cluster:

```
kubectl -n ice-cassandra -c cassandra \
    exec -ti ice-dc1-rack1-0 -- \
    nodetool flush
```

4.5.1.1 For ICE Server 2.0.0 Save a snapshot of each Cassandra node in the cluster:

```
export MY_SNAPSHOT="ice-dc1-rack1-0-$(date "+%Y%m%d")"

kubectl -n ice-cassandra -c cassandra \
    exec -ti ice-dc1-rack1-0 -- \
    nodetool snapshot -t "${MY_SNAPSHOT}"

kubectl -n ice-cassandra -c cassandra \
    exec -ti ice-dc1-rack1-0 -- \
   tar zcvf /tmp/${MY_SNAPSHOT}.tgz /var/lib/cassandra

kubectl -n ice-cassandra -c cassandra \
  cp \
  ice-dc1-rack1-0:/tmp/${MY_SNAPSHOT}.tgz \
  /var/tmp/${MY_SNAPSHOT}.tgz
```

Save `/var/tmp/${MY_SNAPSHOT}.tgz` to a safe place.

4.5.1.2 For ICE Server 2.1.2 You should copy the Cassandra database backups (in MinIO) to another safe location, outside of the existing K8s cluster nodes. Please review the port-forward step in Cassandra Restore Product Guide on how to perform this step.

4.5.2 Custom Grafana dashboards

Export any custom Grafana dashboards so they can be re-imported later, otherwise they will be lost. To export custom Grafana dashboards:

- Navigate to the dashboard.
- Click on the "Share Dashboard or Panel" icon to the right of the dashboard title at the top right corner.
- Select the "Export" tab.
- Select the "Export for sharing externally" radio button, then select "Save to file".

4.6 Step 6: Remove ICE Monitoring chart

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

You will need to remove the existing ICE Monitoring chart for Grafana upgrade to work correctly:

```
kubectl -n ice-release delete hr ice-prometheus
kubectl -n ice-release delete hr ice-monitoring
```

You should see the pods in `ice-metrics` namespace being removed. If they do not, run the following command to purge them:

```
kubectl -n ice-metrics delete all --all
```

4.7 Step 7: Uninstall the existing Helm operator

Uninstall the existing Helm operator via the following command:

```
# Non-Geo
helm -n ice-release uninstall ice-helm-operator

# Geo
helm --kubeconfig ~/.kube/dc1 -n ice-release uninstall ice-helm-operator
helm --kubeconfig ~/.kube/dc2 -n ice-release uninstall ice-helm-operator
```

Run the following command (repeatedly, if necessary) to verify the `helm-operator` pod is no longer available, which confirms the Helm operator was uninstalled:

```
kubectl get pods -n ice-release -l app=helm-operator
```

Once the above command indicates the pod is not found, run the following command:

```
# Non-Geo
kubectl delete crd helmreleases.helm.fluxcd.io

# Geo
kubectl --kubeconfig ~/.kube/dc1 delete crd helmreleases.helm.fluxcd.io
kubectl --kubeconfig ~/.kube/dc2 delete crd helmreleases.helm.fluxcd.io
```

4.8 Step 8: Release MinIO storage

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

If you are using cloud storage that supports dynamic volume resizing, you may skip this section.

If you are using local storage that does not support dynamic volume resizing, such as the local storage provisioner on Microk8s, or Rancher's local storage provisioner as mentioned in the Kubernetes product guide, the following steps must be performed:

- Save your existing MinIO data, if desired. See the *Save existing ICE Server data* section for details.
- Delete the `ice-minio` helmrelease using `kubectl`:

```
kubectl -n ice-release delete hr ice-minio
```

You should see the pods in `ice-minio` namespace being removed.

4.9 Step 9: Check Helm chart value file

To ensure your json files are properly formatted, use `jq` and make sure there is no formatting error reported.

```
# non-Geo
jq . 351upgrade.json

# Geo - DC1
jq . 351upgrade.dc1.json

# Geo - DC2
jq . 351upgrade.dc2.json
```

4.10 Step 10: Upgrade the ICE Server Helm chart

4.10.1 Required actions prior to Helm chart upgrade

For the upgrade to ICE Server 3.5.0, due to recent updates to some Kubernetes CustomResourceDefinitions (CRDs) and operators, some additional actions are required prior to upgrading the ICE Server Helm chart.

1. Remove Instant Connect ingress objects:

```
kubectl delete ing --all -A
```

2. Stop Cassandra:

```
# Non-Geo
kubectl -n ice-cassandra scale deployment \
    ice-release-ice-cassandra-ice-cassandra-cassandra-operator \
    --replicas=0

# Geo
kubectl --kubeconfig ~/.kube/dc1 -n ice-cassandra \
    scale deployment \
    ice-release-ice-cassandra-ice-cassandra-cassandra-operator \
    --replicas=0
kubectl --kubeconfig ~/.kube/dc2 -n ice-cassandra \
    scale deployment \
    ice-release-ice-cassandra-ice-cassandra-cassandra-operator \
    --replicas=0
```

3. Remove default Ingress:

```
# Non-Geo
kubectl -n ice-release delete ing placeholder

# Geo
kubectl --kubeconfig ~/.kube/dc1 -n ice-release delete ing placeholder
kubectl --kubeconfig ~/.kube/dc2 -n ice-release delete ing placeholder
```

4. Remove Cert-Manager and untracked resources:

> **Note:** If you see 'resource not found' errors for the second command, that means the first command executed successfully, so you may ignore such those errors.

```
kubectl -n ice-release delete hr ice-certificates
kubectl delete $(kubectl get crd -o name | grep cert-manager.io)
```

5. Remove Elasticsearch and untracked resources:

> **Note:** If you see 'resource not found' errors for the subsequent commands, that means the first command executed successfully, so you may ignore such those errors.

```
# Non-Geo
kubectl -n ice-release delete hr ice-elasticsearch
kubectl delete $(kubectl get crd -o name | grep elastic.co)
kubectl -n ice-arcus delete statefulsets.apps elastic-operator
kubectl -n ice-arcus delete svc elastic-webhook-server ice-release-ice-
    arcus-ice-elasticsearch-exporter
kubectl -n ice-arcus delete sa elastic-operator
kubectl delete clusterrole elastic-operator elastic-operator-edit elastic-
    operator-view
kubectl delete clusterrolebinding elastic-operator
kubectl delete hr elastic-operator

# Geo
for KCF in ~/.kube/dc1 ~/.kube/dc2
do
export KUBECONFIG=${KCF}
kubectl -n ice-release delete hr ice-elasticsearch
kubectl delete $(kubectl get crd -o name | grep elastic.co)
kubectl -n ice-arcus delete statefulsets.apps elastic-operator
kubectl -n ice-arcus delete svc elastic-webhook-server ice-release-ice-
    arcus-ice-elasticsearch-exporter
kubectl -n ice-arcus delete sa elastic-operator
kubectl delete clusterrole elastic-operator elastic-operator-edit elastic-
    operator-view
kubectl delete clusterrolebinding elastic-operator
kubectl delete hr elastic-operator
done
```

6. Remove license-cleanup-template cronjob:

```
# Non-Geo
kubectl -n ice-arcus delete cronjob license-cleanup-template

# Geo
kubectl --kubeconfig ~/.kube/dc1 -n ice-arcus delete cronjob license-
    cleanup-template
kubectl --kubeconfig ~/.kube/dc2 -n ice-arcus delete cronjob license-
    cleanup-template
```

7. (For Kubernetes v.19.x and v.20.x only) Pre-define Elasticsearch custom resources:

This step only applies if you continue using Kubernetes version v.1.19.x or v1.20.x. You may check your existing `Server Version` with kubectl:

```
$ kubectl version
```

```
Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.1",
    GitCommit:"632ed300f2c34f6d6d15ca4cef3d3c7073412212", GitTreeState:"
    clean", BuildDate:"2021-08-19T15:45:37Z", GoVersion:"go1.16.7",
    Compiler:"gc", Platform:"linux/amd64"}
Server Version: version.Info{Major:"1", Minor:"21", GitVersion:"v1.21.4-
    k3s1", GitCommit:"24d3954c6cd8b757150dd3b0ebf02ad005f07523",
    GitTreeState:"clean", BuildDate:"2021-08-12T22:35:54Z", GoVersion:"go1
    .16.6", Compiler:"gc", Platform:"linux/amd64"}
```

If the `Server Version` shows `v1.21.x`, or if you are migrating to ICE OS, then skip this step.

Contact Technical Support to download a copy of `elasticsearches.elasticsearch.k8s.elastic.co.yaml`. Then, use `kubectl` to create the required custom resources:

```
kubectl -n ice-arcus apply -f elasticsearches.elasticsearch.k8s.elastic.co
    .yaml
```

If this is a Geo Redundant setup, you must apply `elasticsearches.elasticsearch.k8s.elastic.co.yaml` to *both* data centers.

4.10.2 Single Data Center installation

Use the following `helm` command to upgrade existing ICE Server to 3.5.0:

```
# Non-Geo
helm \
  upgrade -i \
  -n ice-release \
  ice-helm-operator \
  ice-release-helm/ice-helm-operator-release-3-5-0 \
  -f 351upgrade.json \
  --set charts.iceLogging.enabled=false \
  --set charts.instantConnectEnterprise.values.config.include\
  elasticsearchOperator=true \
  --version 3.5.41629
```

Verify the upgrade was successful by checking the helm release number. The instant-connect-enterprise chart should show version 3.5.41629.

```
# Non-Geo
helm -n ice-release ls
```

4.10.2.1 Geo Redundant / Multiple Data Center installations

- Upgrade should be done only when both sites are online and can connect to each other.

- This upgrade process does not require any node to be shutdown. However, it is recommended to limit user access to DC2.

4.10.2.2 Upgrade DC1

On DC1, use the following `helm` command to upgrade existing ICE Server to 3.5.0:

```
# Geo: DC1
helm \
  --kubeconfig ~/.kube/dc1 \
  upgrade -i \
  -n ice-release \
  ice-helm-operator \
  ice-release-helm/ice-helm-operator-release-3-5-0 \
  -f ice-dc1-351-values.json \
  --set charts.iceLogging.enabled=false \
  --set charts.instantConnectEnterprise.values.config.include.
     elasticsearchOperator=true \
  --version 3.5.41629
```

Verify the upgrade was successful by checking the helm release number on DC1. The instant-connect-enterprise chart should show version 3.5.41629.

```
# Geo: DC1
helm --kubeconfig ~/.kube/dc1 -n ice-release ls
```

4.10.2.3 Perform Cassandra data sync

On DC1, use the following `kubectl` commands to force a data sync

```
# Geo: DC1
kubectl --kubeconfig ~/.kube/dc1 -n ice-cassandra \
    -c cassandra \
    exec -ti ice-dc1-rack1-0 -- \
    nodetool flush

kubectl --kubeconfig ~/.kube/dc1 -n ice-cassandra \
    -c cassandra \
    exec -ti ice-dc1-rack1-0 -- \
    nodetool repair
```

Verify the upgrade was successful by checking the helm release number on DC1. The instant-connect-enterprise chart should show version 3.5.41629. true:

```
# Geo: DC1
helm --kubeconfig ~/.kube/dc1 -n ice-release ls
```

4.10.2.4 Upgrade DC2 On DC2, Using the following `helm` command to upgrade existing ICE Server to 3.5.0:

```
# Geo: DC2
helm \
  --kubeconfig ~/.kube/dc2 \
  upgrade -i \
  -n ice-release \
  ice-helm-operator \
  ice-release-helm/ice-helm-operator-release-3-5-0 \
  -f ice-dc2-351-values.json \
  --set charts.iceLogging.enabled=false \
  --set charts.instantConnectEnterprise.values.config.include.
     elasticsearchOperator=true \
  --version 3.5.41629
```

Verify the upgrade was successful by checking the helm release number on DC1. The instant-connect-enterprise chart should show version 3.5.41629.

```
# Geo: DC2
helm --kubeconfig ~/.kube/dc2 -n ice-release ls
```

4.10.3 Monitoring Upgrade Progress

The `helm upgrade -i` command will download more than a dozen Docker containers from Instant Connect's private Docker repository (hosted at hub.docker.com). This comprises 10GB or more of data and may take up to one hour, depending on the quality if your Internet connection (typically 10-20 minutes on most corporate networks).

> **This command will take a considerable amount of time to complete.**
>
> To monitor the progress of the installation, run the following commands in a separate terminal session:
>
> `watch kubectl get pods -A`
>
> You should see many pods created in multiple namespaces with `ice-` prefix:
>
> *ice-cassandra ice-cassandra-system ice-arcus ice-rallypoint ice-release*
>
> If the installation does not appear to be progressing, run the following command to review any error being reported:
>
> `kubectl -n ice-release logs --selector app=helm-operator -f`

Upon successful completion, Helm will print a success message to the console.

You can view the Kubernetes resources that were deployed as part of the installation by executing the command `kubectl -n $NAMESPACE get $RESOURCE_TYPE`, where `$NAMESPACE` is one of the namespaces created in Step 1, and `$RESOURCE_TYPE` is a valid Kubernetes object (*i.e.*, `pods`, `services`, `deployments.apps`, `jobs`, or `all`). For a list of the Kubernetes objects created as part of this release, see the appendix at the end of this document.

Validate the installation status of the ICE Helm chart by using the command:

```
helm -n ice-release ls -o yaml
```

This should indicate that `instant-connect-enterprise` is in the `deployed` state.

Verify all existing helmreleases are in `Succeeded` phase with `Succeeded` status:

```
kubectl -n ice-release get hr
```

Congratulations! You have successfully upgraded Instant Connect Enterprise.

4.11 Step 11: Reinstall `ice-helm-operator`

> **Note:** For Geo-redundant setups, this section applies to *both* data centers.

1. Save `helm-values`:

   ```
   helm -n ice-release get values ice-helm-operator -o json | jq . > helm
       -values.json
   ```

2. Uninstall `ice-helm operator`:

   ```
   # delete helm operator
   helm -n ice-release uninstall ice-helm-operator
   ```

3. Do not proceed until confirming the `ice-helm-operator` pod was removed, so run this command…

   ```
   kubectl get pods -n ice-release -l app=helm-operator
   ```

   …until you get this output:

   ```
   No resources found in ice-release namespace.
   ```

4. Remove `helmreleases` CRD:

   ```
   kubectl delete crd helmreleases.helm.fluxcd.io
   ```

5. Run `helmupgrade` (use appropriate version and chart names):

```
helmupgrade -i -n ice-release ice-helm-operator \
    ice-release-helm/ice-helm-operator-release-<PUT RELEASE VERSION
       HERE> \
    --version <PUT VERSION HERE> \
    -f helm-values.json
```

6. Confirm the `ice-helm-operator` pod is running, so run this command…

```
kubectl get pods -n ice-release -l app=helm-operator
```

…until you get this output:

```
NAME                                READY   STATUS    RESTARTS   AGE
ice-helm-operator-d768987fc-ws8nl   1/1     Running   1          13d
```

4.12 Step 12: Apply the ICE Server 3.5.0 license

Congratulations, you have successfully upgraded the ICE Server!

Use ICE Desktop Client 3.5.0 to log into your ICE Server 3.5.0, verify the build information is showing the upgraded ICE Server version, then apply the new ICE Server 3.5.0 license.

4.13 Step 13: ICE Telephony Upgrade

ICE Server 3.5.0, you may continue to run ICE Telephony using an external docker host like in prior release. This section is applicable only if you have not set up ICE Telephony to run on the ICE Server.

First, stop the ICE Telephony docker container:

```
docker stop telephony && docker rm telephony
```

To install ICE Telephony 3.5.0. please see ***ICE Server Administration Guide*** > *Connect to the Instant Connect Docker repository* and follow the instructions.

4.14 Step 14: Verify Rallypoint

The administrator should verify on ICE Desktop that a valid Rallypoint is set in the default Channel setup page.

4.15 Step 15: Verify ICE Monitoring Chart

If pods in `ice-metrics` namespace are crashing, you may issue the following command to clear them:

```
# non-Geo
kubectl -n ice-metrics delete replicasets --all

# Geo - DC1
kubectl --kubeconfig ~/.kube/dc1 -n ice-metrics delete replicasets --all

# Geo - DC2
kubectl --kubeconfig ~/.kube/dc2 -n ice-metrics delete replicasets --all
```

4.16 Step 16: Recreate Grafana Accounts

Log in as an admin to Grafana and recreate your previous accounts. Please see ***ICE Server Administration Guide*** > *Monitoring* > *Grafana Dashboards* for instructions on retrieving the new Grafana admin password.