ICE Server Administration Guide

Product guide for prerelease

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

main 90adc8bf40040649230176bbdd465f6261a2d8e0

Contents

1 Introduction **7**
1.1 ICE OS Patching Policy 7
1.2 ICE Telephony 7

2 Security and Certificates **7**

3 Command Line Monitoring **8**
3.1 Pod Container Restarts 8

4 Adjust Pod Affinity Settings **8**
4.1 ElasticSearch Configuration Update 8
4.2 Rallypoint Configuration Update 8

5 High Availability **9**

6 Appendix A: General Troubleshooting Guide **12**
6.1 Ad Hoc Server Backup 13

7 Appendix B: Elasticsearch **14**
7.1 Additional Steps for Multi-node Setups 14
7.2 Repair 17

8 Appendix C: Internal and External Firewall Rules **17**
8.1 Internal and External Firewall Rules 18
8.1.1 Internal (Host) 18
8.1.2 External 18

9 Appendix D: Restart Static Reflectors **18**
9.1 Reconnect the default (cluster)Static Reflector 18
9.2 Reconnect the external (docker host) Static Reflector 18

10 Appendix E: Restart Patch Servers **19**
10.1 Reconnect the default (cluster) Patch Server 19
10.2 Reconnect the external (docker host) Patch Server 19

11 Appendix F: Client Configuration File **19**
11.1 File Specifications 20
11.2 File Parameters and Values 21
11.2.1 applicationAutoLaunch 21

11.2.2 audioSettingsAudioDevice . . . 21
11.2.3 audioSettingsToneEnabled . . . 21
11.2.4 audioSettingsToneLevels . . . 23
11.2.5 audioSettingsVibration . . . 24
11.2.6 audioSettingsWiredHeadsetType . . . 24
11.2.7 channelMode . . . 24
11.2.8 connectivitySettingsCheckInterval . . . 25
11.2.9 connectivitySettingsNetworkInterface . . . 25
11.2.10 connectivitySettingsTrellisware . . . 25
11.2.11 desktopLocation . . . 26
11.2.12 logBufferSizeInDays . . . 26
11.2.13 loginUsername . . . 26
11.2.14 monitoringSettingsCrashes . . . 27
11.2.15 monitoringSettingsLocation . . . 27
11.2.16 operatingModes . . . 27
11.2.17 serverKey . . . 28
11.2.18 telephonyAsAChannel . . . 28
11.2.19 timelines . . . 29
11.2.20 verboseLogging . . . 29

12 Appendix G: Helm Chart Objects 30

13 Appendix H: Vector Logging Integration 31
13.1 Vector and Amazon CloudWatch . . . 32
13.1.1 A. Creating a Log Group on Amazon AWS. . . . 32
13.1.2 B. Obtaining the Access Key and AWS Region . . . 33
13.1.3 C. Configuring a Vector Agent for CloudWatch on ICE OS . . . 34
13.1.4 D. Verifying Log Reception . . . 35
13.2 Vector and Azure Monitor Logs . . . 35
13.2.1 A. Configuring Azure Log Monitoring . . . 35
13.2.2 B. Configuring the Vector Agent . . . 35
13.2.3 C. Verifying Log Reception . . . 36
13.3 Vector and Splunk HEC v9.1 . . . 37
13.3.1 Installing Docker . . . 37
13.3.2 Installing Docker-Compose . . . 38
13.3.3 Generating SSL . . . 38
13.3.4 Create Docker-Compose file . . . 38
13.3.5 Start Splunk HEC with Docker-Compose . . . 40

13.3.6 Verifying Connectivity 42
13.3.7 Stop Splunk HEC with Docker-Compose 42
13.3.8 Configuring HEC on Splunk 42
13.3.9 Configuring a Vector Agent for Splunk HEC on ICE OS 45

List of Tables

1 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.

1.1 ICE OS Patching Policy

ICE OS uses an embedded Linux kernel, and so encounters fewer vulnerabilities in comparison to server/desktop operating systems. As part of its security strategy, ICE OS is read-only and immutable, so it cannot be patched in the same way as some other operating systems, e.g., Red Hat, Windows. Each new ICE product release includes a new ICE OS version.

Instant Connect requires customers to be on the latest General Availability (GA) product release in order to receive security vulnerability support. If a vulnerability were discovered, Instant Connect would issue an updated ICE OS version to address it. The update would be for the latest GA only, and not for any older product releases.

1.2 ICE Telephony

ICE Telephony integrates Instant Connect Enterprise's push-to-talk communications with your SIP PBX as a registrar or as SIP Trunk, enabling advanced voice communication feature. For example:

- A telephone caller can dial an Instant Connect user (using ICE Desktop or ICE Android) and establish a full-duplex phone call with them.
- An appropriately configured Instant Connect user can use their client software to place a dial call. In this regard, the ICE Desktop and ICE Android clients function as a "soft phone." A telephone caller can dial directly into a channel that's been configured to accept outside callers. The telephone caller can speak on the channel by pressing the * key to request the floor, and the # to relinquish it.

Please refer to **ICE Telephony Administration Guide** for additional information, including instructions for installing a local patch server and a static reflector.

2 Security and Certificates

Refer to the **ICE Security Guide** for instructions on security configuration.

3 Command Line Monitoring

In cases where browser-based monitoring (i.e., Grafana) is not available, the following command line option exists:

3.1 Pod Container Restarts

To monitor pod container restarts, enter the following command line:

```
watch kubectl get pods -A
```

The resulting table shows `ice-rallypoint` pod restarts (the fifth column):

```
ice-rallypoint   patch-6445f47d45-xtp2s          4/4   Running   4   26d
ice-rallypoint   rallypoint-59fff8dfd5-lxf57     4/4   Running   5   26d
ice-rallypoint   reflector-797b9f8fdb-2tdnn      4/4   Running   4   26d
```

The number of restarts should be minimal. If there are more than a few in an hour, then the host may be overloaded.

4 Adjust Pod Affinity Settings

If you are running ICE Server on multi-workers K8s cluster, you should update the pod affinity for a more robust failover. Skip this section if your cluster has only one worker node.

4.1 ElasticSearch Configuration Update

Please refer to 'Appendix B: Elasticsearch' below.

4.2 Rallypoint Configuration Update

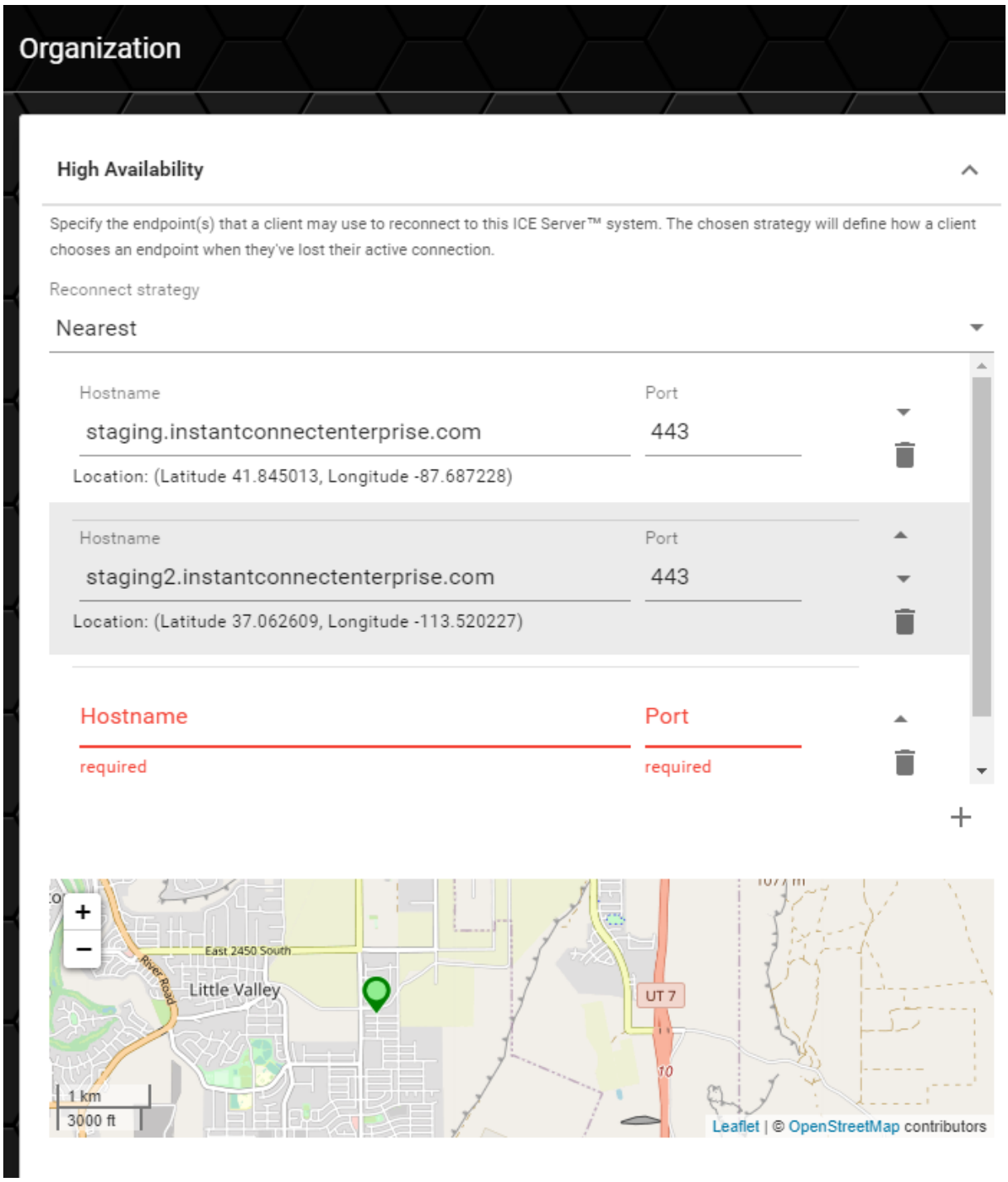
Run the following command once to adjust Rallypoint's failover setting, if your cluster has three (3) worker nodes. Adjust the number according to the number of worker nodes.

```
kubectl -n ice-rallypoint patch deploy rallypoint -p '{"spec":{"replicas
    ":3,"template":{"spec":{"affinity":{"podAntiAffinity":{"
    preferredDuringSchedulingIgnoredDuringExecution":[{"podAffinityTerm":{"
    labelSelector":{"matchLabels":{"app":"rallypoint"}},"topologyKey":"
    kubernetes.io/hostname"},"weight":100}]}}}}}'
```

> **Note:** The above command is on one single line. To cut-and-paste correctly, please paste it into a text editor, remove the paragraph break, then copy the edited text into the terminal window to run.

5 High Availability

The ICE administrator can specify the endpoint (ICE Server FQDN) a client may use to reconnect to their ICE Server system. The administrator can also choose a connection strategy for determining which endpoint to use when the client has lost their active connection.


Organization
High Availability
Specify the endpoint(s) that a client may use to reconnect to this ICE Server™ system. The chosen strategy will define how a client chooses an endpoint when they've lost their active connection.
Reconnect strategy
Nearest
Hostname
staging.instantconnectenterprise.com
Port
443
Location: (Latitude 41.845013, Longitude -87.687228)
Hostname
staging2.instantconnectenterprise.com
Port
443
Location: (Latitude 37.062609, Longitude -113.520227)
Hostname
required
Port
required
East 2450 South
River Road
Little Valley
UT 7
1 km
3000 ft
Leaflet | © OpenStreetMap contributors

Field / icon	Description
Reconnect strategy	Choose the strategy the client will use to determine which endpoint to use when the client has lost their active connection.• **Preferred** - The endpoints are specified in order of connection preference. A client will always try to connect to the highest-ranked endpoint first; if that connection fails it will try the second ranked endpoint and so on.• **Nearest** - Clients will attempt to connect to the endpoint physically nearest to its current location. When geolocation is unavailable the client will connect in order of preference.• **Random** - Client will randomly choose an endpoint to connect to.• **Identity** - Clients will only connect tot he endpoint the user entered on the login screen.
Endpoint Hostname / Port	Fully Qualified Domain Name (FQDN) of the ICE server and the IP Port number the client will use for the connection.
Endpoint Location	Location Latitude / Longitude of the ICE Server the client will use to determine the nearest ICE server for the connection. The administrator can use the map and map pin to set the location for the ICE server.
	Use the Up / Down arrows on the Endpoint record to create the preferred connection list for the client to use with the preferred connection strategy. The Up arrow will move the endpoint up the list, the Down arrow will move the endpoint down the list.
	Use the Trash Can icon to delete an Endpoint record from the list.

Field / icon	Description
	Use the Plus Sign icon to create a new row in the list for another Endpoint record.

6 Appendix A: General Troubleshooting Guide

Issue	**Suggestion**
How to find the installation ID?	See the License page on the **ICE Desktop**
What is the approximate time required to complete the successful install to plan the activity with network and system administrator?	Less than 30 minutes on a properly configured Kubernetes cluster
When installing using `helm`, this error message is reported: **Error: Kubernetes cluster unreachable**	Make sure the Kubernetes cluster is accessible by running `kubectl get nodes`. Make sure the environment variable `$KUBECONFIG` is defined and pointing to a valid Kubernetes `KUBECONFIG` file, typically `${HOME}/.kube/config`
I tried to install ICE Server using ssh. The session timed out and got disconnected before the install has finished.	The ICE Server installation may continue to run when your session is disconnected. Simply resume installation from where you left off
How to check if the ICE Server charts and add-ons are installed?	Run `helm ls -A` then `kubectl get pods -A` to look for pods that failed to start. Consult technical support if there is any pod that shows large number of restarts.
What version is installed currently?	Choose HELP → BUILD INFO on the **ICE Desktop**
ICE Server superuser password is lost. How to reset it?	Use another administrator account to reset the password
How to request a license?	Contact ICE License Support with installation ID.

Issue	Suggestion
License file upload failed	Make sure the license file received from ICE License Support is saved as-is, without any modification. Make sure the installation ID in the license file matches what is displayed on the **ICE Desktop** license page.
How to increase the licensed feature counts?	Request a new license from ICE Sales Support
Is a new license necessary if the product is reinstalled?	Yes. Any new installation (including reinstallation on the same Kubernetes cluster) will require a new license.
The hosting VM is rebooted. Is manual restart of the ICE Server necessary?	ICE Server would start automatically. There is no need to run traditional Linux OS commands such as 'service start….' , 'systemctl …', etc. Run `watch kubectl get pods -A` to monitor pod restart status. The pods may take a few minutes to up to 15 minutes (on slower system) to complete restart. Typically, restarting the host VM is not recommended, as it rarely would automatically resolve any pod issue.
What is the approximate time required to complete the successful upgrade to plan the activity with network and system administrator?	Upgrade typically only requires a brief, transient outage of less than one minute. Active users typically do not need to log out during the upgrade process.
`watch kubectl get pods -A` is showing spordic `etcdserver timeout` errors	Your hosting hardware's storage devices may be too slow. Review disk I/O latency of your hosting hardware, upgrade storage devices as needed
Is it possible to change IP address and/or hostname after ICE Server is installed?	After the cluster is installed, changing IP address and/or hostnames is not recommended

6.1 Ad Hoc Server Backup

In addition to scheduled backups, we recommend an ad hoc backup of the server prior to beginning troubleshooting or upgrade processes. To create an ad hoc backup:

```
cat <<EOF | kubectl apply -f -
apiVersion: db.orange.com/v1alpha1
```

```
kind: CassandraBackup
metadata:
  labels:
    app: cassandra
  name: cassandra-backup-$(date +"%s")
  namespace: ice-cassandra
spec:
  cassandraCluster: ice
  datacenter: dc1
  secret: minio-access-secret
  storageLocation: oracle://backup
  snapshotTag: '$(date +"%s")'
EOF
```

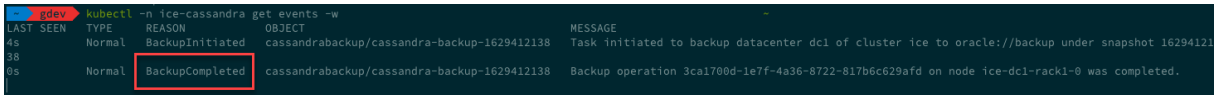
Example output:

```
cassandrabackup.db.orange.com/cassandra-backup-1629411753 created
```

Wait for the backup to complete. Check progress by using the following command:

```
kubectl -n ice-cassandra get events -w
```

Events will display as they come in. The following example shows the backup process has completed:



7 Appendix B: Elasticsearch

ElasticSearch provides channel and people search capabilities in Instant Connect.

7.1 Additional Steps for Multi-node Setups

To ensure your multi-node cluster runs seamlessly during failover state, complete the following additional steps. The example below assumes the cluster has three (3) nodes.

1. Create a new file called `es_nodeport.yaml`:

```
{
  "apiVersion": "v1",
  "kind": "Service",
  "metadata": {
    "labels": {
      "common.k8s.elastic.co/type": "elasticsearch",
```

```
      "elasticsearch.k8s.elastic.co/cluster-name": "elasticsearch-arcus"
    },
    "name": "ice-arcus-es-client-np",
    "namespace": "ice-arcus",
    "selfLink": "/api/v1/namespaces/ice-arcus/services/ice-arcus-es-client
       -np"
  },
  "spec": {
    "externalTrafficPolicy": "Cluster",
    "ports": [
      {
        "name": "arcus-es",
        "nodePort": 30029,
        "port": 9200,
        "protocol": "TCP",
        "targetPort": 9200
      }
    ],
    "selector": {
      "common.k8s.elastic.co/type": "elasticsearch",
      "elasticsearch.k8s.elastic.co/cluster-name": "elasticsearch-arcus"
    },
    "sessionAffinity": "None",
    "type": "NodePort"
  }
}
```

2. Create the `nodeport` service:

```
kubectl -n ice-arcus create -f es_nodeport.yaml
```

3. Scale up the Elasticsearch deployment using `kubectl` on any one node:

```
# the following command must be on a single line
ESS=$(kubectl -n ice-arcus get secrets elasticsearch-arcus-es-elastic-user
    -o jsonpath --template '{.data.elastic}' | base64 -d)

# the following command must be on a single line
LIP=$(ip route get 1 | awk '{print $NF;exit}')

# the following command must be on a single line
kubectl -n ice-arcus patch elasticsearches.elasticsearch.k8s.elastic.co
   elasticsearch-arcus --type='json' --patch='[{"op":"replace","path":"/
   spec/nodeSets/0/count","value":3}]'
```

4. You should see Elasticsearch scales up to three nodes with green status:

```
$ kubectl -n ice-arcus \
    get elasticsearches.elasticsearch.k8s.elastic.co  \
    elasticsearch-arcus
```

```
NAME                  HEALTH   NODES   VERSION   PHASE   AGE
elasticsearch-arcus   green    3       7.6.2     Ready   2d22h
```

5. Define replicas for each Elasticsearch index:

```
for INDEX in $(curl -k --user elastic:${ESS}  https://${LIP}:30029/_cat/
   indices 2>/dev/null | awk '{print $3}')
do
    curl -k --user elastic:${ESS} -XPUT \
        "https://${LIP}:30029/${INDEX}/_settings?pretty" \
        -H 'Content-Type: application/json' \
        -d' { "number_of_replicas": 0 }'
done
```

6. Verify each index is now replicated across all three nodes:

```
$ curl -k --user elastic:${ESS}  https://${LIP}:30029/_cat/shards
channels          0 r STARTED   5 37.4kb 10.90.0.9  elasticsearch-arcus-es
   -member-0
channels          0 r STARTED   5 37.4kb 10.90.2.21 elasticsearch-arcus-es
   -member-2
channels          0 p STARTED   5 37.4kb 10.90.1.22 elasticsearch-arcus-es
   -member-1
geofence          0 r STARTED   0   283b 10.90.0.9  elasticsearch-arcus-es
   -member-0
geofence          0 p STARTED   0   283b 10.90.2.21 elasticsearch-arcus-es
   -member-2
geofence          0 r STARTED   0   283b 10.90.1.22 elasticsearch-arcus-es
   -member-1
persons           0 r STARTED   4 87.8kb 10.90.0.9  elasticsearch-arcus-es
   -member-0
persons           0 r STARTED   4 87.8kb 10.90.2.21 elasticsearch-arcus-es
   -member-2
persons           0 p STARTED   4 87.8kb 10.90.1.22 elasticsearch-arcus-es
   -member-1
filedata          0 p STARTED   0   283b 10.90.0.9  elasticsearch-arcus-es
   -member-0
filedata          0 r STARTED   0   283b 10.90.2.21 elasticsearch-arcus-es
   -member-2
filedata          0 r STARTED   0   283b 10.90.1.22 elasticsearch-arcus-es
   -member-1
auditlog-02042022 0 p STARTED  30 43.2kb 10.90.0.9  elasticsearch-arcus-es
   -member-0
auditlog-02042022 0 r STARTED  30 43.2kb 10.90.2.21 elasticsearch-arcus-es
   -member-2
auditlog-02042022 0 r STARTED  30 43.2kb 10.90.1.22 elasticsearch-arcus-es
   -member-1
auditlog-02012022 0 r STARTED 102 61.7kb 10.90.0.9  elasticsearch-arcus-es
   -member-0
```

```
auditlog-02012022 0 p STARTED 102 61.7kb 10.90.2.21 elasticsearch-arcus-es
    -member-2
auditlog-02012022 0 r STARTED 102 61.7kb 10.90.1.22 elasticsearch-arcus-es
    -member-1
textmessage       0 p STARTED   0   283b 10.90.0.9  elasticsearch-arcus-es
    -member-0
textmessage       0 r STARTED   0   283b 10.90.2.21 elasticsearch-arcus-es
    -member-2
textmessage       0 r STARTED   0   283b 10.90.1.22 elasticsearch-arcus-es
    -member-1
```

> **Note:** In a Geo-redundant setup, the above procuredure should be performed in each data center.

7.2 Repair

On rare occasions, Elasticsearch may fall out of sync. The most likely example is becoming out of sync with Cassandra after an unexpected system down event, and typically it would recover itself in no more than 24 hours.

If you are experiencing data inconsistency issues, e.g., people or channels not appearing in search results, or find that some users' online/offline status indicators do not match their true values, then it may be helpful to execute the following re-sync procedure:

```
kubectl \
  -n ice-arcus \
  create job \
  resync-$(date "+%Y%m%d-%H%M") \
  --from=cronjob/elastic-sync-DATACENTERNAME
```

> **Note:** In a Geo-redundant setup, the above command should be run in each data center.

8 Appendix C: Internal and External Firewall Rules

Please note the following ports:

- **80**: http
- **443**: https
- **7443**: Rallypoint (whether secure or unsecure)

8.1 Internal and External Firewall Rules

8.1.1 Internal (Host)

```
sudo firewall-cmd --zone=public --permanent --add-port
   ={6443,2379-2380,10250-10252,10255,30000-32767}/tcp
  sudo firewall-cmd --zone=public --permanent --add-port=8472/udp
  sudo firewall-cmd --zone=public --permanent --add-masquerade --permanent
  sudo firewall-cmd --zone=public --permanent --add-port=80/tcp
  sudo firewall-cmd --zone=public --permanent --add-port=443/tcp
  sudo firewall-cmd --zone=public --permanent --add-port=7443/tcp
  sudo firewall-cmd --permanent --direct --add-rule ipv4 filter INPUT 0 -m
      pkttype --pkt-type multicast -j ACCEPT
  sudo firewall-cmd --zone=public --permanent --add-protocol=igmp
  sudo firewall-cmd --zone=trusted --permanent --add-interface cni0
  sudo firewall-cmd --reload
  sudo sysctl --system
  echo " Adding cni "
  sudo firewall-cmd --zone=trusted --permanent --add-interface cni0
  echo " Adding cni "  echo " your firewall is configued as "
  sudo firewall-cmd --list-all --zone trusted
  sudo firewall-cmd --list-all --zone public
```

8.1.2 External

```
sudo firewall-cmd --zone=public --permanent --add-port=80/tcp
  sudo firewall-cmd --zone=public --permanent --add-port=443/tcp
  sudo firewall-cmd --zone=public --permanent --add-port=7443/tcp
```

9 Appendix D: Restart Static Reflectors

The administrator should verify on ICE Desktop that all valid Static Reflectors are in connected state.

9.1 Reconnect the default (cluster)Static Reflector

Use `kubectl` to restart it:

```
kubectl -n ice-rallypoint delete pod -l app=reflector
```

9.2 Reconnect the external (docker host) Static Reflector

On the docker host, use `docker` to restart it:

```
docker restart reflector-agent
docker restart reflector
```

10 Appendix E: Restart Patch Servers

The administrator should verify on ICE Desktop that all valid Patch Servers are in connected state.

10.1 Reconnect the default (cluster) Patch Server

Use `kubectl` to restart it:

```
kubectl -n ice-rallypoint delete pod -l app=patch
```

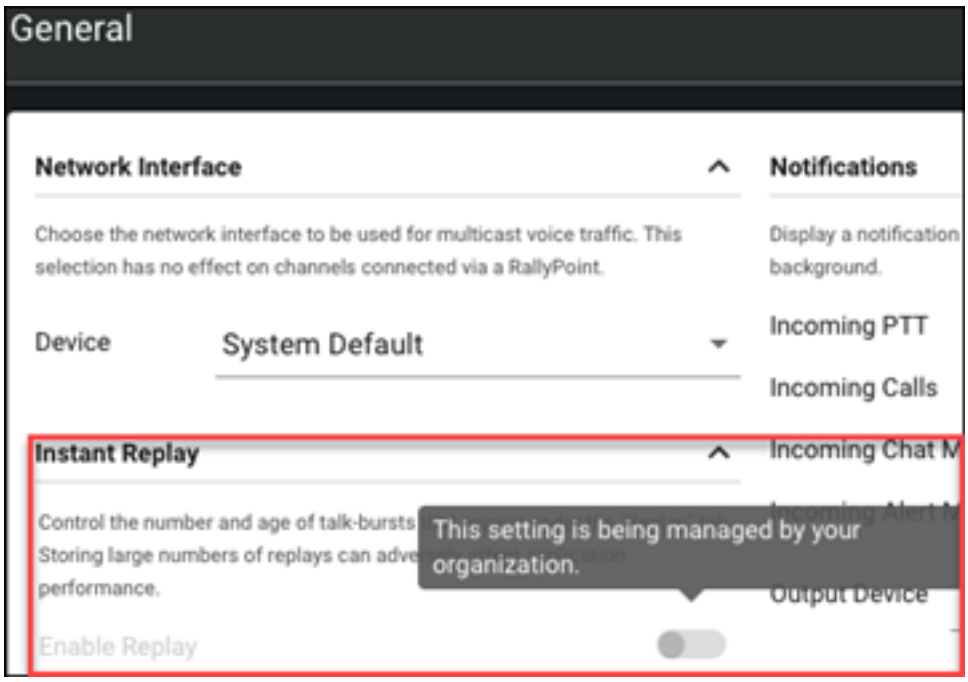
10.2 Reconnect the external (docker host) Patch Server

On the docker host, use `docker` to restart it:

```
docker restart patch-agent
docker restart patch
```

11 Appendix F: Client Configuration File

ICE desktop and mobile client configurations can be overridden on devices via a manually edited .json file placed at a specified location on the relevant devices. Configuration files are sent to devices via your mobile device management (MDM) service (or manual upload). The desktop and mobile clients check the relevant file location on launching. To be recognized by the clients, the file *must* reflect the specifications, parameters, and values detailed below. If an appropriate configuration file is found, then the values therein are applied, overriding any prior configuration settings by the administrator or user. Any feature/setting configured from the file is locked to the user and a message displays: “This setting is being managed by your organization.”



> **Note:** Due to the inherent risks of overriding configurations via a manually edited text file, please proceed with caution.

11.1 File Specifications

File name: app_config.json

File type: .json

File location:

Operating System (OS)	OS Type	File Location
Android	Mobile	`/Android/data/com.dillonkane.ice.flutter/files`
iOS	Mobile	`/ICE Mobile`
MacOS	Desktop	`~/Library/Application Support/ICE Desktop`
Windows	Desktop	`C:\Users\\[USERNAME]\AppData\Roaming\ICE Desktop`

Required file content: Client configuration files *must* include the following parameter and values:

```
"monitoringSettingsLocation": {
   "accuracy": "high",
   "changeMeters": 0,
   "shareLocation": true,
   "updateIntervalSeconds": 60
}
```

11.2 File Parameters and Values

11.2.1 applicationAutoLaunch

Feature / Setting:

- **Mobile:** For Android only, on rebooting the device, the ICE mobile client will auto-launch.

Values: true, false (Boolean)

Example:

```
"applicationAutoLaunch": false
```

11.2.2 audioSettingsAudioDevice

Feature / Setting:

- **Mobile:** Settings > Audio > Default Audio Device > Device

Values: earpiece, speaker (String)

Example:

```
"audioSettingsAudioDevice": "speaker"
```

11.2.3 audioSettingsToneEnabled

Feature / Setting:

- **Desktop:** Settings > General >
  - Error Sounds > Network Channel Error
  - Other Sounds >
    - Alert Received

        * Channel Added
        * Incoming Text Message
        * Outgoing Text Message
        * Telephony Incoming Call
        * Telephony Outgoing Call
    - Push To Talk Sounds >
        * Push to Talk Denied
        * Push to Talk Ended
        * Push to Talk Granted
        * Push to Talk Received

- **Mobile:** Audio >
    - Error Sounds > Network Error
    - Other Sounds >
        * Alert Received
        * Channel Added
        * Incoming Text Message
        * Outgoing Text Message
        * Telephone Call
    - Push To Talk Sounds >
        * Push to Talk Denied
        * Push to Talk Ended
        * Push to Talk Granted

Values:

- `desktopIncomingCall`: true, false (Desktop, Boolean)
- `desktopIncomingTextMessage`: true, false (Desktop, Boolean)
- `desktopOutgoingCall`: true, false (Desktop, Boolean)
- `desktopOutgoingTextMessage`: true, false (Desktop, Boolean)
- `desktopPttReceived`: true, false (Desktop, Boolean)
- `errorChannel`: true, false (Both, Boolean)
- `otherAlert`: true, false (Both, Boolean)
- `otherChannelAdded`: true, false (Both, Boolean)
- `otherIncomingTextMessage`: true, false (Mobile, Boolean)
- `otherOutgoingTextMessage`: true, false (Mobile, Boolean)
- `otherPrivateCall`: true, false (Mobile, Boolean)
- `pttDenied`: true, false (Both, Boolean)

- `pttEnded`: true, false (Both, Boolean)
- `pttGranted`: true, false (Both, Boolean)

Example:

```
"audioSettingsToneEnabled": {
   "desktopIncomingCall": true,
   "desktopIncomingTextMessage": false,
   "desktopOutgoingCall": false,
   "desktopOutgoingTextMessage": false,
   "desktopPttReceived": false,
   "errorChannel": false,
   "otherAlert": false,
   "otherChannelAdded": true,
   "otherIncomingTextMessage": false,
   "otherOutgoingTextMessage": false,
   "otherPrivateCall": true,
   "pttDenied": false,
   "pttEnded": false,
   "pttGranted": false
}
```

11.2.4 audioSettingsToneLevels

Feature / Setting:

- **Desktop:** Settings > General >
  - Error Sounds (Volume)
  - Other Sounds (Volume)
  - Push To Talk Sounds (Volume)
- **Mobile:** Audio >
  - Error Sounds > Tone volume
  - Other Sounds > Tone volume
  - Push To Talk Sounds > Tone volume

Values:

- `error`: Range from 0 to 1 using decimals: 0, 0.1, 0.2, 0.3 etc (Number)
- `other`: Range from 0 to 1 using decimals: 0, 0.1, 0.2, 0.3 etc (Number)
- `ptt`: Range from 0 to 1 using decimals: 0, 0.1, 0.2, 0.3 etc (Number)

Example:

```
"audioSettingsToneLevels": {
   "error": 0.9,
   "other": 0.1,
   "ptt": 0.75
}
```

11.2.5 audioSettingsVibration

Feature / Setting:

- **Mobile:** Audio > Haptic Feedback > Enable vibrations

Values: true, false (Boolean)

Example:

```
"audioSettingsVibration": false
```

11.2.6 audioSettingsWiredHeadsetType

Feature / Setting: ***Where/what is this?***a

- **Mobile:** For Android only, specify the allowed brand/model of wired accessory for PTT.

Values: milicomUHA, normal, savoxRSM30, savoxSH01 (String)

Example:

```
"audioSettingsWiredHeadsetType": normal
```

11.2.7 channelMode

Feature / Setting:

- **Mobile:** Settings > History > Channel mode > Radio mode

Values: radio, regular (String)

Example:

```
"channelMode": regular
```

11.2.8 connectivitySettingsCheckInterval

Feature / Setting:

- **Mobile:** Settings > Network Connection > Network Check

Values: Range is from 1 to any number, but recommend: 30, 60, 90, 120

Example:

```
"connectivitySettingsCheckInterval": 60
```

11.2.9 connectivitySettingsNetworkInterface

Feature / Setting:

- **Desktop:** Settings > General > Network Interface
- **Mobile:** Settings > Network Connection > Network Interface

Values: Values loaded from server (String)

Example:

```
"connectivitySettingsNetworkInterface": "en0"
```

11.2.10 connectivitySettingsTrellisware

Feature / Setting:

- **Mobile:** Settings > Asset Discovery > Discover Trellisware

Values:

- `enabled`: true, false (Boolean)

Example:

```
"connectivitySettingsTrellisware": {
   "enabled": false
}
```

11.2.11 desktopLocation

Feature / Setting:

- **Desktop:** Settings > Location > Share my location with others
  - Automatically
  - Using a location I specify
  - Never

Values:

- `sharing`: AUTO, MANUAL, OFF (String)

Example:

```
"desktopLocation": {
   "sharing": "AUTO"
}
```

11.2.12 logBufferSizeInDays

Feature / Setting:

- **Mobile:** Settings > Analytics > Maximum Logs Limit Days

Values: Range is from 1 to any number, but recommend: 1-10

Example:

```
"logBufferSizeInDays": 4
```

11.2.13 loginUsername

Feature / Setting: The value is used for login to both the desktop and mobile client. The 'Username' field on the client login UI displays the value and cannot be edited by the user.

Values: Any string, e.g., `andrii@test4.com`

Example:

```
"loginUsername": "andrii@test4.com"
```

11.2.14 monitoringSettingsCrashes

Feature / Setting:

- **Mobile:** Settings > Analytics > Report Crashes

Values: true, false (Boolean)

Example:

```
"monitoringSettingsCrashes": false
```

11.2.15 monitoringSettingsLocation

> **Note:** Client configuration files *must* include this parameter, see the *'File specifications'* section above.

Feature / Setting:

- **Mobile:** Settings > Location Tracking

Values:

- `accuracy`: balanced, high, low, powersave (String)
- `changeMeters`: Range is from 1 to any number, but recommend: 0, 5, 20, 36, 50, 100
- `shareLocation`: background, foreground, none (String)
- `updateIntervalSeconds`: Range is from 1 to any number, but recommend: 15, 30, 60, 120, 300

Example:

```
"monitoringSettingsLocation": {
   "accuracy": "high",
   "changeMeters": 0,
   "shareLocation": true,
   "updateIntervalSeconds": 60
}
```

11.2.16 operatingModes

Feature / Setting:

- **Mobile:** Settings > Location Tracking > Operating Modes

Values:

- `emergencyAlertButton`: true, false (Boolean)
- `onScreenPttButton`: true, false (Boolean)
- `persistentRxDisplay`: true, false (Boolean)
- `radioMode`: true, false (Boolean)
- `silentModeEnabled`: true, false (Boolean)
- `tacticalEnabled`: true, false (Boolean)
- `telephonyAsAChannel`: true, false (Boolean)

Example:

```
"operatingModes": {
   "emergencyAlertButton": false,
   "onScreenPttButton": false,
   "persistentRxDisplay": false,
   "radioMode": true,
   "silentModeEnabled": false,
   "tacticalEnabled": false
   "telephonyAsAChannel": true,
}
```

11.2.17 serverKey

Feature / Setting: The value is used for login to both the desktop and mobile client. The 'Address' field on the client login UI displays the value and cannot be edited by the user.

Values: Any string, e.g., `test.icnow.app`

Example:

```
"serverKey": "develop.icnow.app"
```

11.2.18 telephonyAsAChannel

Feature / Setting:

- **Mobile:** Settings > History > Channel mode > Telephony As a Channel

Values: true, false (Boolean)

Example:

```
"telephonyAsAChannel": false
```

11.2.19 timelines

Feature / Setting:

- **Desktop:** Settings > General > Instant Replay
  - Enable Replay
  - Oldest Replay
  - Max Stored Replays
- **Mobile:** Settings > History
  - Instant Replay
  - Oldest Replay
  - Maximum History Limit

Values:

- `enabled`: true, false (Boolean)
- `historyAgeHours`: Range is from 1 to any number, but recommend: 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 (Number)
- `historyLimit`: 25, 50, 75, 100 (Number)

Example:

```
"timelines": {
   "enabled": false,
   "historyAgeHours": 4,
   "historyLimit": 50
}
```

11.2.20 verboseLogging

Feature / Setting:

- **Desktop:** Settings > General > Crash Reporting > Verbose Logging
- **Mobile:** Settings > Analytics > Verbose Logging

Values: true, false (Boolean)

Example:

```
"verboseLogging": false
```

12 Appendix G: Helm Chart Objects

Resource Name	Namespace	Description
Cassandra	ice-cassandra	User data on the ICE Server is stored in Apache Cassandra™, a distributed, wide column store, NoSQL database management system designed to handle large amounts of data across many servers, providing high availability with no single point of failure. Cassandra offers robust support for clusters spanning multiple datacenters.
client-bridge	ice-arcus	Client-bridge is a component of the ICE Server which acts as a gateway and its main responsibilities are authorization, authentication and traffic routing.
elastic	ice-arcus	Elasticsearch is a distributed, RESTful search and analytics engine for lightning fast search, fine-tuned relevancy, and powerful analytics that scale with ease
Grafana / Promethus (ICE Monitoring)	ice-metrics	Grafana is a multi-platform open source analytics and interactive visualization web application. Prometheus is a free software application used for event monitoring and alerting.
Kafka	ice-kafka	Apache Kafka™ is a stream-processing software that provides a unified, high-throughput, low-latency platform for handling real-time data feed
MinIO (ICE Minio)	ice-minio	MinIO is an Amazon S3 compatible server-side software storage stack, it can handle unstructured data such as photos, videos, log files, backups, and container images
modelmanger	ice-arcus	ModelManager is a component of the ICE Server that establishes all of the required keyspaces, tables, columns and topics within Cassandra and Kafka. It also provides automatic database migration when upgrading the ICE Server.

Resource Name	Namespace	Description
patching	ice-rallypoint	Supports channel patching
platform-services	ice-arcus	Platform Services is an component of the ICE Server that is responsible for business logic execution.
Rallypoint	ice-rallypoint	RallyPoints™ is a component of the ICE Server that bridges media (voice) traffic between networks (e.g. over the internet) without using multicast traffic.
reflector	ice-rallypoint	A reflector contains a set of reflections within one multicast domain. A reflection defines a channel to be reflected within that domain by specifying the transmission and receiving address
rest-bridge	ice-arcus	RESTful API
telephony	ice-arucs	ICE Telephony provides IP Telephony service thru DN and SIP bridge
server-bridge	ice-arcus	Server Bridge is a component of the ICE Server that provides SIP (Session Initiation Protocol) service to set up real-time multimedia sessions between groups of participants
zookeeper	ice-kafka	Apache Zookeeper™ is a centralized service to maintain naming and configuration data and to provide flexible and robust synchronization within distributed systems. Zookeeper keeps track of status of the Kafka cluster nodes and it also keeps track of Kafka topics, partitions etc, and keeps them in sync.

13 Appendix H: Vector Logging Integration

The ICE Server can be configured to stream logs to an external logging repository using Vector and a Vector agent. Vector is an open-source log aggregator. It allows the configuration of observability pipelines by fetching logs from many sources, transforming the data as needed, and routing it to a

destination. Use the instructions in this section to leverage Vector for the integration of streaming logs.

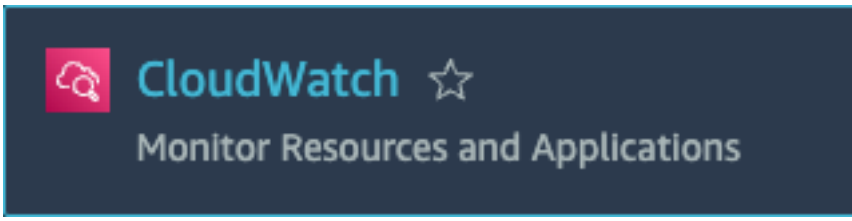
13.1 Vector and Amazon CloudWatch

Use the following procedures to configure log streaming using Vector on Amazon CloudWatch.

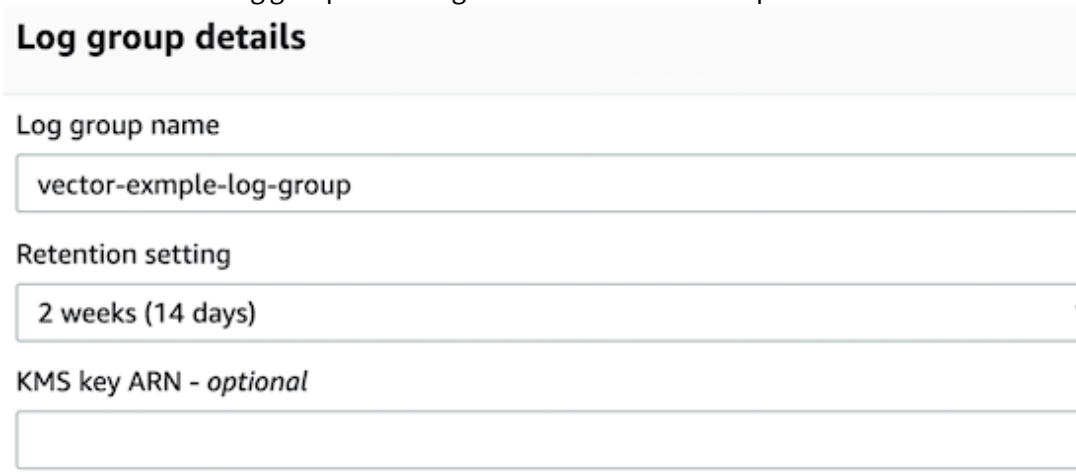
13.1.1 A. Creating a Log Group on Amazon AWS.

To create a log group on Amazon AWS

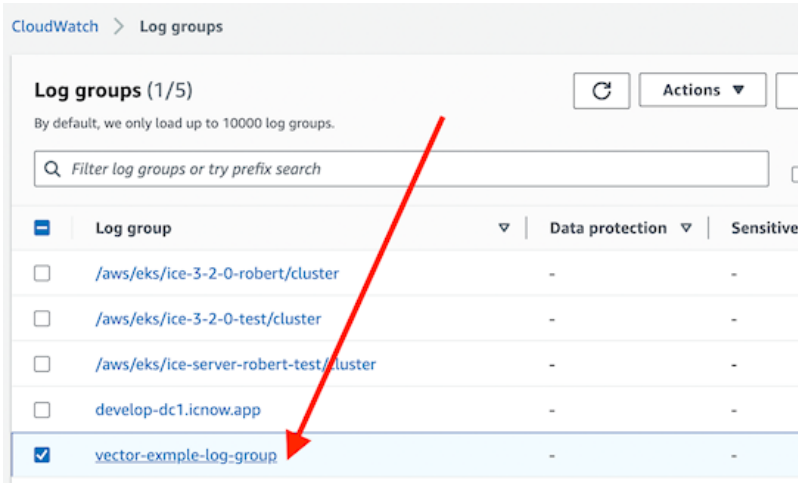
1. Log in to Amazon AWS using SSO. The Instant Connect SSO link is https://instantconnectnow.awsapps.com/start
2. Type `cloudwatch` in search box. 

3. In the search results, click **CloudWatch.** 

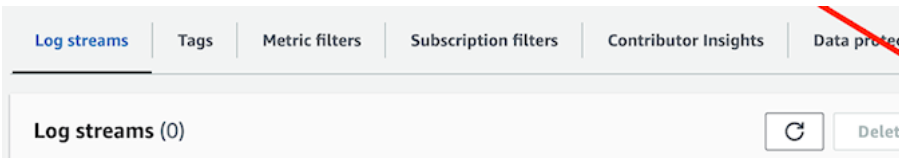
4. In the left pane, expand **Logs** and click **Log groups.**
5. In the right pane, click the **Create log group** button. 

The Create log group page opens.
6. Complete the required fields to create a log group according to the standards and requirements of your organization. 

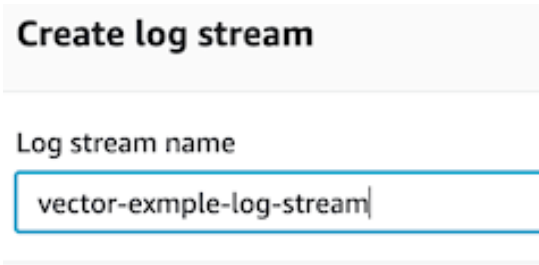
7. Once completed, click the **Create** button. The Log groups screen opens.



8. Click the name of the group you just created.
   The configuration page for your log group opens.



9. Click the **Create log stream** button.
   The Create log stream window appears.

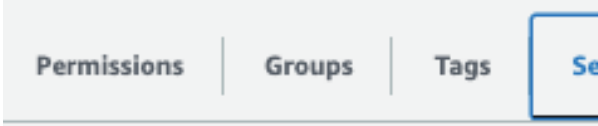


10. Optionally edit the Log stream name, then click the **Create** button.

13.1.2 B. Obtaining the Access Key and AWS Region

To obtain the AWS_ACCESS_KEY and AWS_ACCESS_SECRET_KEY

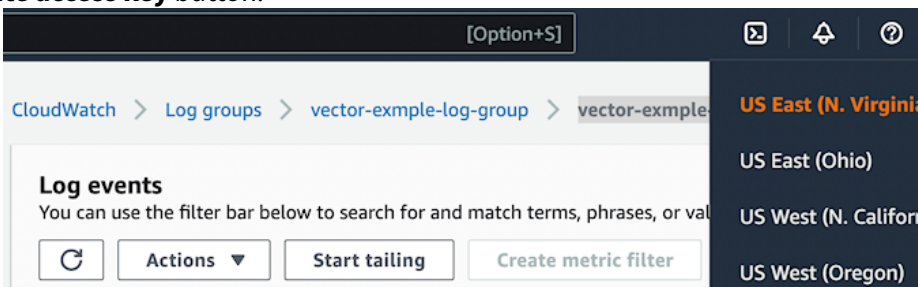
1. In **Identity and Access Management (IAM),** create a service account user with appropriate access.



2. In the IAM user detail page, click the **Security Credentials** tab

3. Under **Access Keys,** click the **Create access key** button.

4. Copy the **AWS region** for later use.



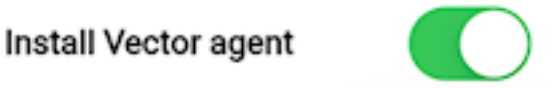
13.1.3 C. Configuring a Vector Agent for CloudWatch on ICE OS

To configure the Vector agent

> [!NOTE]
>
> For optimal performance, set up a different data source for each ICE Server or ICE Server Geo setup.

1. In the ICE OS Configuration Wizard, click the **External Log Store** tab.

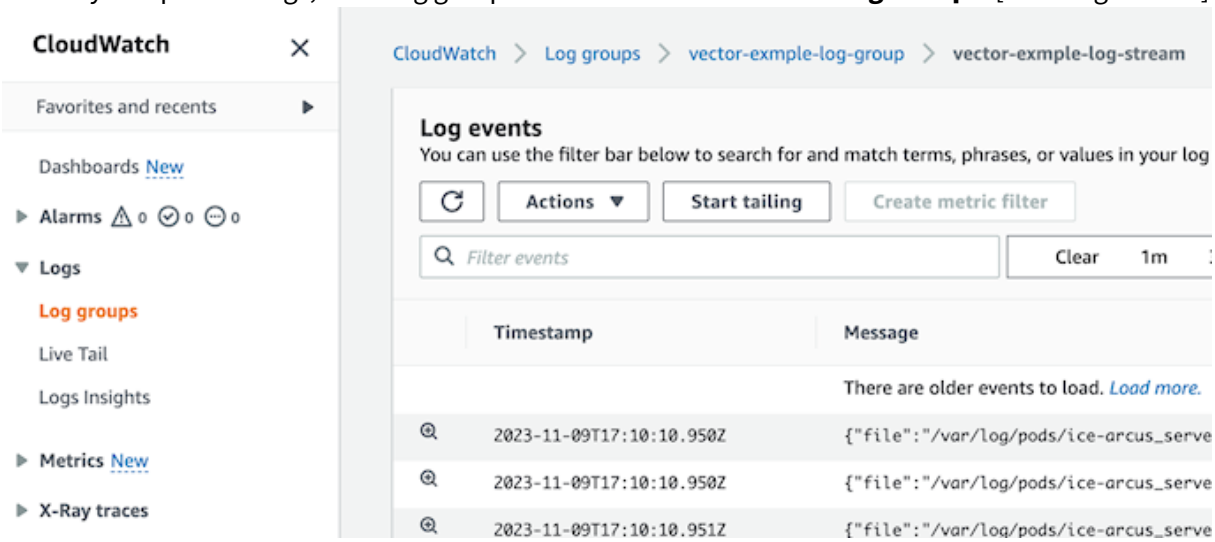
2. Ensure **Install Vector Agent** is selected.



3. Select the **AmazonCloudWatchLogs** Vector Sink Type.
4. Enter the **AWS CloudWatch Log Region.** Refer to the previous procedure to obtain.
5. Enter the **Log Group Name.** Refer to the first procedure (A) to obtain.
6. Enter the **Log Stream Name.** Refer to the first procedure (A) to obtain.
7. Enter **Access Key ID.** Refer to the previous procedure to obtain.
8. Enter **Secret Access Key.** Refer to the previous procedure to obtain.
9. Click the **Apply** button. The logs from ICE Server are now streaming into AWS CloudWatch. (It is not uncommon to see a delay of a few minutes, before the log messages are streamed into Azure Monitor Logs).

13.1.4 D. Verifying Log Reception

To verify reception of logs, click Log groups and follow **CloudWatch > Log Group >** [Your Log Stream].



13.2 Vector and Azure Monitor Logs

The Azure Monitor Log provides an option for viewing Vector logs. Use the following instructions to configure Vector log streaming to an Azure Logs Analytics workspace.

13.2.1 A. Configuring Azure Log Monitoring

To configure Azure log monitoring

1. Log in to your Azure Portal.
2. Navigate to Log Analytics workspaces
3. Create a new Log Analytics workspace.
4. Obtain the **Workspace ID** and **Primary Key.**

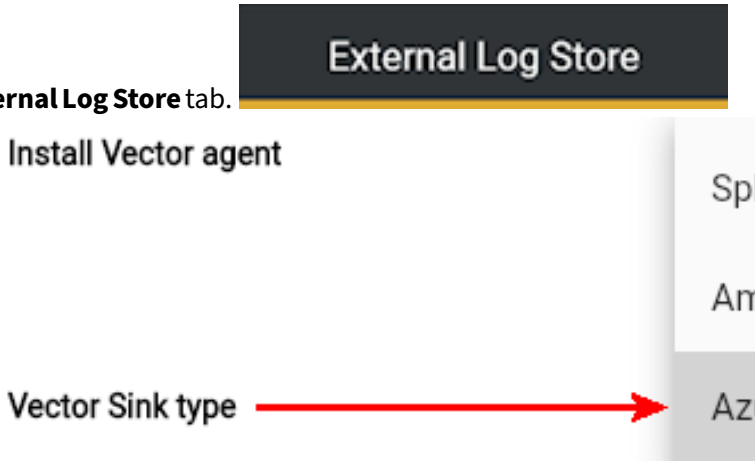
13.2.2 B. Configuring the Vector Agent

To configure the Vector agent

> [!NOTE]
>
> For optimal performance, set up a different data source for each ICE Server or ICE Server Geo setup.

> Similar to Secure LDAP, you should enable TLS only if Azure Monitor Logs is using server identity certificates issued by self-signed, private or Enterprise CA.

1. In the ICE OS Configuration Wizard, click the **External Log Store** tab.



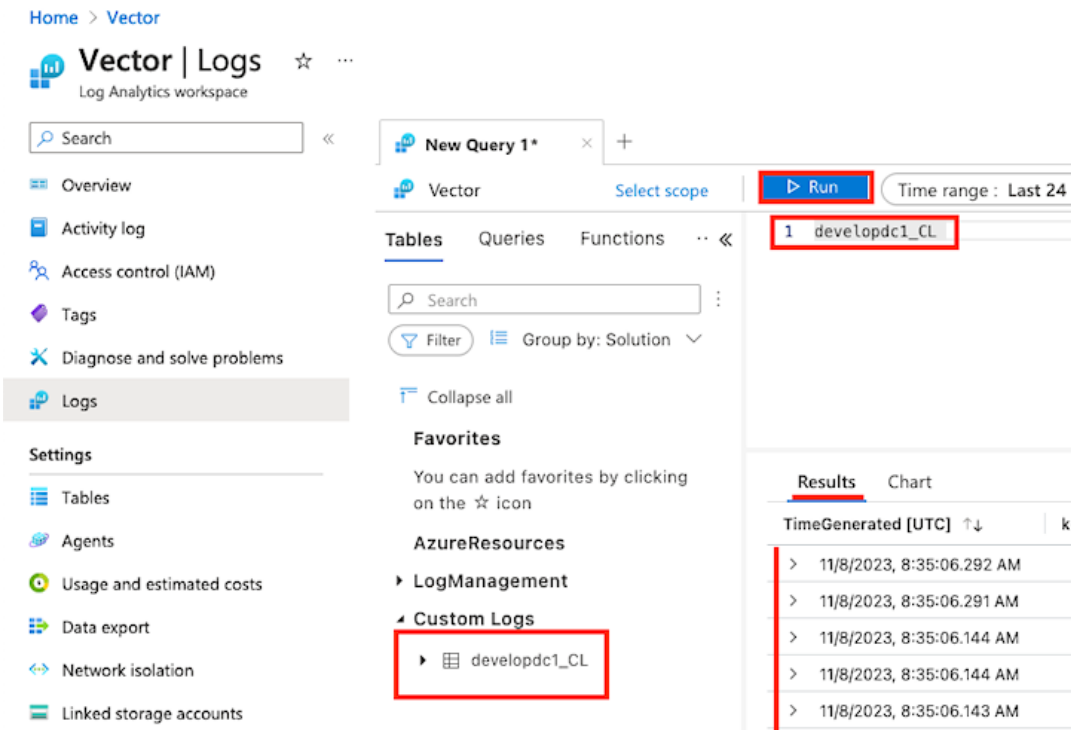
2. Select the **AzureMonitorLogs** Vector Sink Type.
3. Enter the **Customer ID.** Customer **ID is identical** to the **Workspace ID** of the desired Log Analytics Workspace.
4. Copy the **Shared Key.** Refer to the first procedure (A) to obtain.
5. Enter the **Table Name.** Table Name is the unique identifier used to group your ICE Server messages. If a table does not exist in the Log Analytics Workspace, it will be created automatically.
6. Click the **Apply** button. Logs from the ICE Server are now streaming into the Azure Monitor Logs. (It is not uncommon to see a delay of a few minutes, before the log messages are streamed into Azure Monitor Logs).

13.2.3 C. Verifying Log Reception

Check that the specified table has been created

To verify log reception

1. Navigate to **Log Analytics Workspace > Logs tab > Custom Logs.**
2. Verify that a table with the **Table Name** you created exists.
3. Create a new query and copy the name of the table into the Query field.



4. Click the **Run** button.
   The Results tab populates with the results of the query.

13.3 Vector and Splunk HEC v9.1

The document provides a working example of ICE Server log transmission to Splunk HEC (http event collector) through a Vector agent.

13.3.1 Installing Docker

Note that running `docker` without `sudo` requires the addition of the associated user ID into `docker` group.

```
sudo groupadd docker
sudo usermod -aG docker ${USER}
```

Enter the following to run the installer. Do not use Snap to install Docker.

```
sudo apt -y update
sudo apt -y install apt-transport-https ca-certificates curl software-
    properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add
    -
sudo add-apt-repository "deb [arch=amd64] \
  https://download.docker.com/linux/ubuntu focal stable"
```

```
apt-cache policy docker-ce
#
#
#
sudo apt-get -y install init-system-helpers
sudo apt -y install docker-ce
sudo systemctl restart docker
sudo systemctl status docker
```

13.3.2 Installing Docker-Compose

Enter the following to run the installer.

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/
    docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-
    compose

sudo chmod +x /usr/local/bin/docker-compose
docker-compose --version
```

Enter the following to verify the service is running:

```
systemctl restart docker
docker-compose down -v
docker-compose up -d
```

13.3.3 Generating SSL

For TLS access, you will need to create a server certificate and key pair.

13.3.4 Create Docker-Compose file

In the following example, a self-signed CA and self-signed certs are employed with Splunk 9.1.3.

> [!IMPORTANT]
>
> Do not mix Splunk versions.

```
sudo mkdir -p /home/splunk/etc
cd /home/splunk

docker run -it --rm -e SPLUNK_PASSWORD=UjT5sNCWIIwEaaI2zUBmOvkMlVToGbtW
    splunk/splunk:9.1.3 create-defaults > default.yml
```

```
cat <<EOF > /home/splunk/docker-compose.yml

version: "3.6"

services:
  splunk:
    image: splunk/splunk:9.1.3
    container_name: ice-splunk-hec-logs
    volumes:
      - /home/splunk/etc:/opt/splunk/etc
      - /home/splunk/tls:/vector/tls
      - /home/splunk/tls/splunk-cacert.pem:/opt/splunk/etc/auth/cacert.pem
      - /home/splunk/tls/splunk-cert-chain.pem:/opt/splunk/etc/auth/ca.pem
      - /home/splunk/tls/splunk-cert-chain.pem:/opt/splunk/etc/auth/server
          .pem
    environment:
      - DEBUG=true
      - SPLUNK_START_ARGS=--accept-license
      - SPLUNK_HEC_SSL=true
      - SPLUNK_HEC_TOKEN=zJTLfiO0VyFHEtKb7PweSbQ9qJoTY78i
      - SPLUNK_LICENSE_URI=free
      - SPLUNK_PASSWORD
      - SPLUNK_HTTP_ENABLESSL=true
      - SPLUNK_HTTP_ENABLESSL_CERT=/vector/tls/splunk-mysite.pem
      - SPLUNK_HTTP_ENABLESSL_PRIVKEY=/vector/tls/splunk-private.key
    ports:
      - "8000:8000"
      - "9997:9997"
      - "8088:8088"
      - "1514:1514"

EOF
```

In the above example:

- Web Console Access (https://splunk.icnow.app:8000)
  - /home/splunk/tls/splunk-mysite.pem should have the following content in this specific order
    - SPLUNK'S SERVER CERT
    - SPLUNK'S INTERMEDIATE CA CERT (if applicable)
    - SPLUNK'S ROOT CA CERT
  - /home/splunk/tls/splunk-**private**.key should contain
    - PRIVATE KEY RELATED TO SPLUNK'S SERVER CERT
- HEC TLS Access (https://splunk.icnow.app:8088)

- `/home/splunk/tls/splunk-cacert.pem` should have the following content in this specific order
  - SPLUNK'S INTERMEDIATE CA CERT (if applicable)
  - SPLUNK'S ROOT CA CERT
- `/home/splunk/tls/splunk-cert-chain.pem` should have the following content in this specific order
  - SPLUNK'S SERVER CERT
  - PRIVATE KEY RELATED TO SPLUNK'S SERVER CERT
  - SPLUNK'S INTERMEDIATE CA CERT (if applicable)
  - SPLUNK'S ROOT CA CERT

13.3.5 Start Splunk HEC with Docker-Compose

If you are installing the first time, or re-installing after a failed/botched install, you can use the following to delete references to old resources:

```
cd /home/splunk
docker-compose down --remove-orphans
docker container rm ice-splunk-hec-logs -f
#
# only if you feel brute & personal against Splunk!
#
docker volume prune -a -f
docker system prune -a -f
```

You may specify the Splunk `admin` password via command line, or in `docker-compose.yml`. For example, using randomly generated password `UjT5sNCWIIwEaaI2zUBmOvkMlVToGbtW`

```
cd /home/splunk

SPLUNK_PASSWORD=UjT5sNCWIIwEaaI2zUBmOvkMlVToGbtW docker-compose up -d
```

Run the following to view the logs:

```
docker logs -f ice-splunk-hec-logs
```

Watch out for file permission issues. It may be necessary to exec into the docker container to ensure the certs are readable in `/vector/tls`.

If the installation is successful, an ansible summary appears:

```

RUNNING HANDLER [splunk_common : Wait for splunkd management port]
    *************
```

```
ok: [localhost]

PLAY RECAP
    *********************************************************************
localhost                  : ok=82   changed=16   unreachable=0    failed
    =0    skipped=68   rescued=0    ignored=0

Thursday 07 March 2024  01:50:49 +0000 (0:00:00.716)       0:01:10.815
    ********
===============================================================================

splunk_common : Restart the splunkd service - Via CLI ------------------
    17.58s
splunk_common : Start Splunk via CLI -----------------------------------
    7.59s
splunk_common : Update Splunk directory owner --------------------------
    3.26s
splunk_common : Update /opt/splunk/etc ---------------------------------
    2.34s
splunk_common : Get Splunk status --------------------------------------
    2.21s
Gathering Facts --------------------------------------------------------
    2.02s
splunk_common : Generate user-seed.conf (Linux) ------------------------
    1.33s
splunk_common : Test basic https endpoint ------------------------------
    1.31s
splunk_standalone : Setup global HEC -----------------------------------
    1.09s
splunk_standalone : Get existing HEC token -----------------------------
    1.09s
splunk_common : Activate free license ----------------------------------
    1.08s
Check for required restarts --------------------------------------------
    1.06s
splunk_standalone : Check for required restarts ------------------------
    1.05s
splunk_common : Cleanup Splunk runtime files ---------------------------
    1.05s
splunk_common : Check current license group ----------------------------
    1.01s
splunk_standalone : Update HEC token configuration ---------------------
    1.01s
splunk_common : Wait for splunkd management port -----------------------
    0.85s
splunk_common : Check for scloud ---------------------------------------
    0.79s
splunk_common : Hash the password --------------------------------------
    0.75s
splunk_common : Wait for splunkd management port -----------------------
    0.72s
```

```
================================================================================


Ansible playbook complete, will begin streaming splunkd_stderr.log
```

13.3.6 Verifying Connectivity

To verify connectivity

1. open your browser to `https://splunk.icnow.app:8000` (not 8088!) The connection should be secure.
2. Ensure you import the self-signed root CA cert into your workstation's truststore and trust it.
3. Navigate to the default HEC page: https://splunk.icnow.app:8000/en-US/manager/launcher/http-eventcollector
4. To test HEC TLS, hit port 8088 (e.g. https://splunk.icnow.app:8088/) You may not get a valid response, but you should see the connection is secure.

Note the default HEC token, and then run a test curl:

```
$ curl https://splunk.icnow.app:8088/services/collector/event \
  -H "Authorization: Splunk zJTLfiO0VyFHEtKb7PweSbQ9qJoTY78i" \
  -d '{"event": "hello world"}'

{"text": "Success", "code": 0}
```

> [!CAUTION]
>
> When you test using `curl` you would add path after 8088. However, Vector sink add such automatically. So in the ICE interview, you only need to enter `https://splunk.icnow.app:8088`

13.3.7 Stop Splunk HEC with Docker-Compose

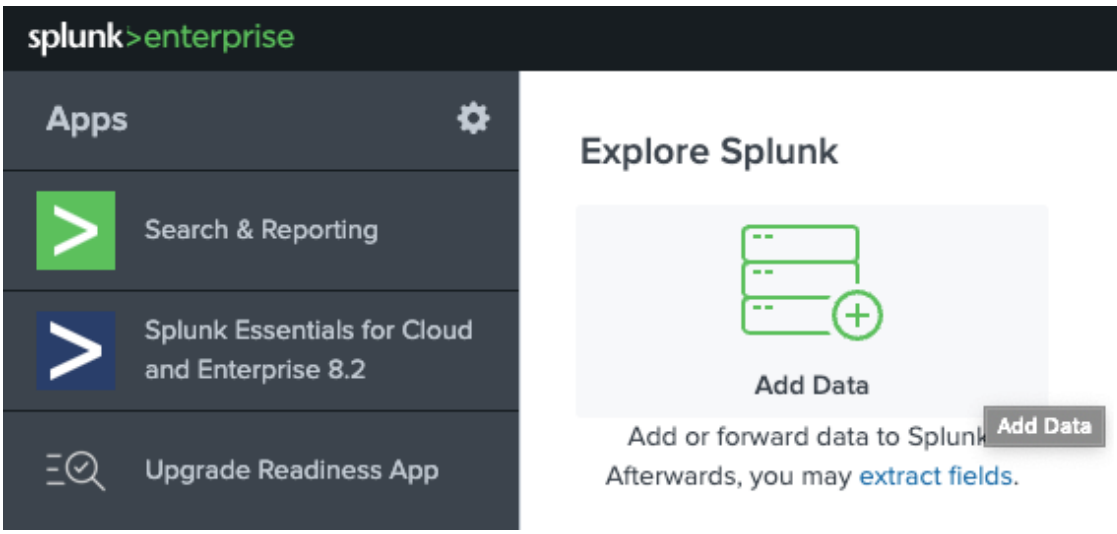
To stop Splunk HEC using Docker-Compose

```
docker-compose down --remove-orphans
```

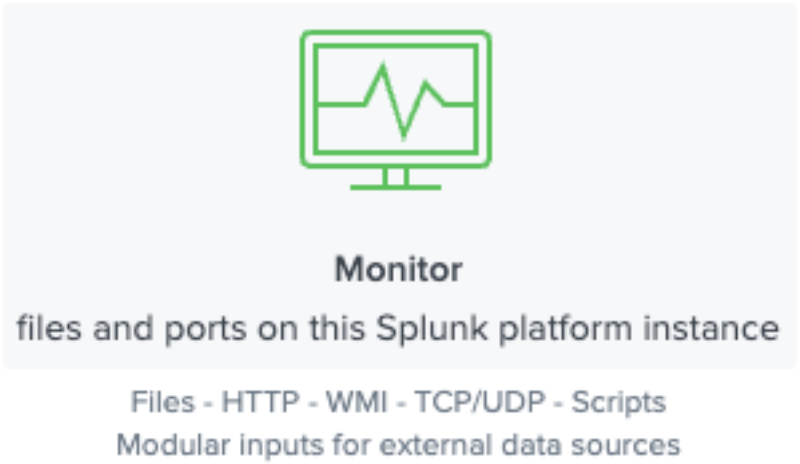
13.3.8 Configuring HEC on Splunk

The Splunk admin console may be access using IP or FQDN to port 8000.

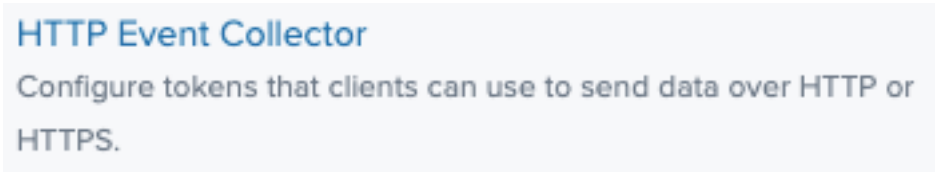
1. Click **Add Data**

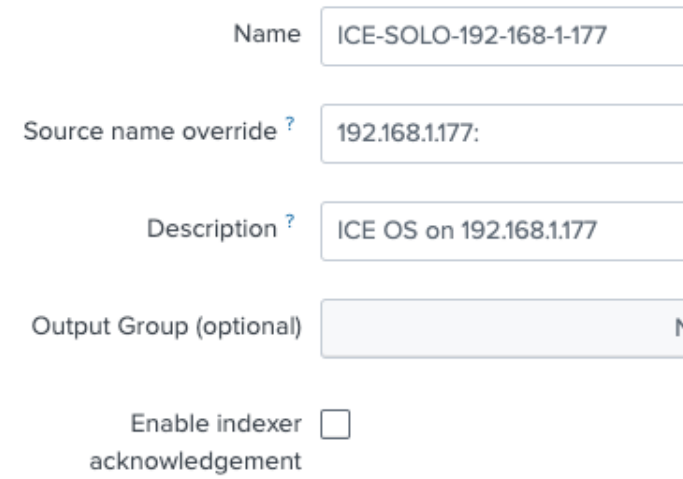


2. Click **Monitor.**

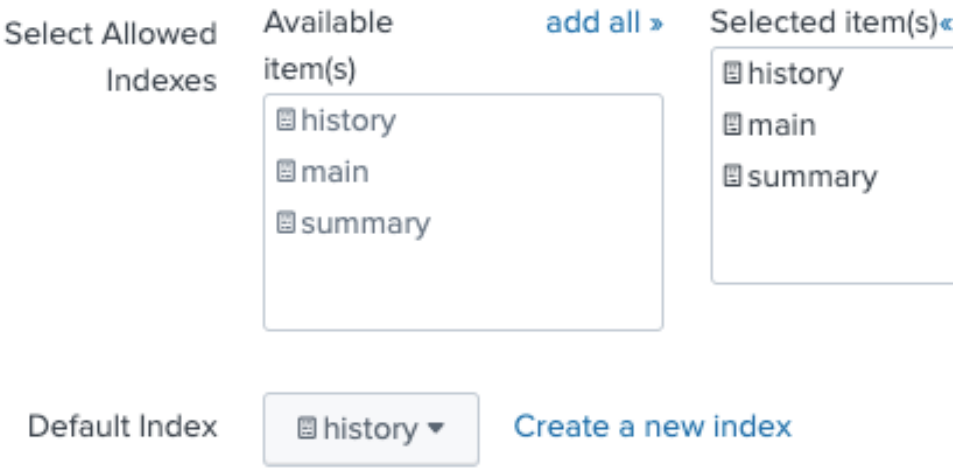


3. Click **HTTP Event Collector.**

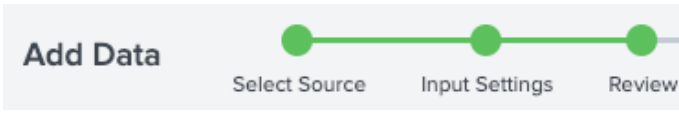




4. Complete the required fields to configure the token.



5. Customize indices as needed.



6. Review the configuration and click the **Submit** button.

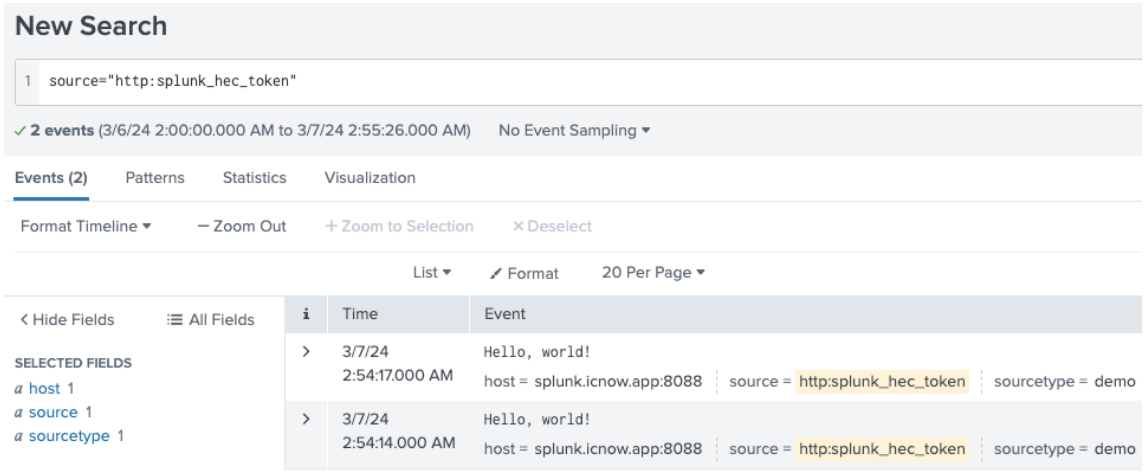
7. Click **Start Searching** to test the configuration.

8. On any host trusting the self-signed root CA cert, run `curl` using the token generated by Splunk HEC. You should see `{"text":"Success","code":0}` output. For example:

```
$ curl https://splunk.icnow.app:8088/services/collector/event -H '
   Authorization: Splunk  zJTLfiO0VyFHEtKb7PweSbQ9qJoTY78i' -d '{"
   sourcetype": "demo", "event":"Hello, world!"}'

{"text":"Success","code":0}
```

9. You should see the test data appearing in your search. The most basic search query is `source="NAME_OF_THE_DATA_SOURCE"(index="history"OR index="main"OR index=`




```
"summary")
```

13.3.9 Configuring a Vector Agent for Splunk HEC on ICE OS

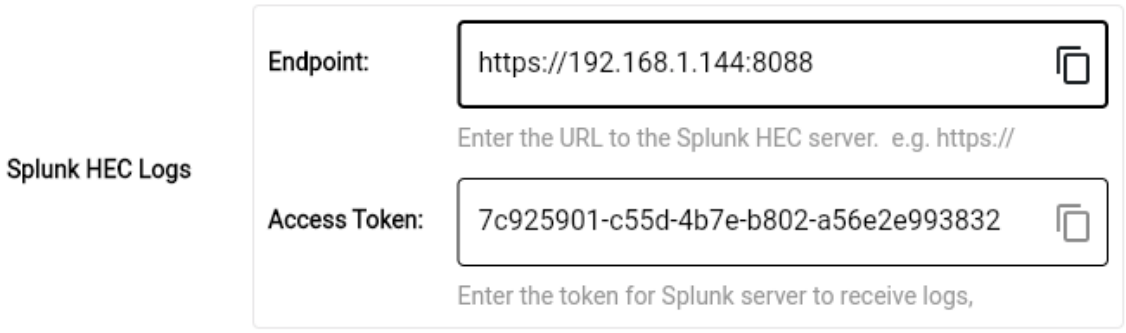
> [!NOTE]
>
> For optimal performance, set up a different data source for each ICE Server or ICE Server Geo setup.

> [!CAUTION]
>
> Similar to Secure LDAP, you should enable TLS only if the Splunk server is using server identity certificates issued by self-signed, private or Enterprise CA.

To configure a Vector agent for Splunk HEC on ICE OS

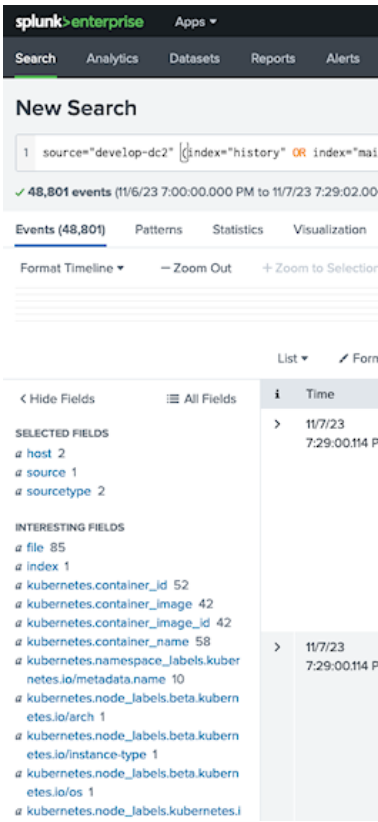
1. In the ICE OS Configuration Wizard, click the **External Log Store** tab.

2. Select the **SplunkHECLogs** Vector Sink Type.
3. Enter the Splunk HEC Logs **Endpoint.**



Note: The evaluation version of Splunk uses port 8088/TCP. Licensed installations may use port 443/TCP.

4. Copy the Access Token into the **Access Token** field.



5. Click the **Apply** button. Logs from the ICE Server are now streaming into Splunk.

Note: It is not uncommon to see a delay of a few minutes, before the log messages are streamed from the ICE Server into Splunk HEC