

IPWorks Scaling Guide for CEE

USER GUIDE

Copyright

© Ericsson AB 2017, 2018. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
1.1	Prerequisites	1
1.2	Preparation	2
1.3	Related Information	10
2	Scaling Procedure	11
2.1	Creating Backup before Scaling	11
2.2	Operation of Scale-Out through CLI	11
2.3	Operation of Scale-In through CLI	21
2.4	Operation of Scale Out/In through VNF-LCM	27
2.5	Scaling Health Check	28
2.6	Creating the Final Backup	31
	Reference List	33





1 Introduction

This document describes how to perform the scaling operations for IPWorks with Cloud Execution Environment (CEE). Scaling operations can only be performed on Standard deployment configuration. IPWorks DHCP or DNS service does not support scalability.

Scaling operations include:

- Scale-out, with the creation of new Payloads (PLs) by instantiation of new Virtual Machines (VMs), joins them to the cluster and increases the number of PLs in the system.

The maximum number of PLs after scale-out operation is 10.

- Scale-in, with the removal of existing PLs from the cluster and deletion the associated VM instances, decreases the number of PLs in the system.

The minimum number of PLs after scale-in operation is 2.

For both scaling operation, the number of SC is always 2.

1.1 Prerequisites

This section states the prerequisites for performing the scaling procedure. It is assumed that users of this document are familiar with performing operations in cloud environment.

1.1.1 Prerequisites for IPWorks VNF

The following conditions must apply in IPWroks VNF:

- The scaling functionality is available from IPWorks 1.9 and higher. Make sure that the IPWorks is upgraded or deployed to the correct version first.
- The CEE version must be R6A or higher.
- The scaling functionality is enabled.

```
#cmw-configuration --status SCALING
```

Enable

- No other maintenance activity is in progress in IPWorks VNF.
- Scheduled backups must not be active. To disable scheduled backups, refer to **BRF-C Management Guide**.



- VNF is in a healthy state to perform a scaling, follow section Section 2.5 Scaling Health Check on page 28 to check the health status.
- Scaling (both scale-out and scale-in) operation activities are recommended to be performed at off-peak hours.
- The eVIP configuration must be based on the IPWorks eVIP template . Other eVIP configuration is not supported.
- Enough available IP addresses of Signaling network (<IPW_SIG_SP1_NW>) and Data network (<IPW_DATA_SP1_NW>) are reserved for the use of scaled-out PL VM.

If there is not enough IP address for scaling-out, expand the Signaling network and Data network.

1.1.2 Conditions

By default, most actions are performed on the Atlas, and some actions are performed on Service Controllers (SCs), unless otherwise specified.

1.2 Preparation

1.2.1 Configuring SS7 to Support Scaling Operations

Note: This section is only available for the upgraded IPWorks.

Ensure that SS7 configuration meets the scaling requirement.

This section describes how to precheck the SS7 configuration for supporting the scaling function.

Prerequisites:

This section is only applicable when the following conditions are satisfied, if none of the condition is met, skip the section.

- IPWorks uses SS7 stack. The SS7 stack is used by the following IPWorks functions/scenarios:
 - IPWorks ENUM Number Portability (NP)
 - Wi-Fi Mobility Management (WiFi MM)
 - Diameter over SCTP
 - Secure Entitlement Server (SES) Support
 - Wi-Fi AAA



Before do the precheck, start the Signaling Manager on the SC-1:

1. Log on to the SC-1.

```
# ssh root@<SC-1 IP address>
```

2. Find the path to PSO storage where SS7 configuration files are stored:

```
# cat /usr/share/psd/storage-paths/config
```

```
<path to config PSD storage>
```

3. Create links to the path where SS7 configuration files are stored, If the path /opt/sign/etc already exists, skip it.

```
# ln -s <path to config PSD storage>/ss7caf-ana90137/etc
/opt/sign/etc
```

4. Start Signaling Manager on the SC-1.

```
# /opt/sign/EABss7050/bin/signmgrui -own.conf /opt/sign/etc/signmgr.cnf &
```

Note:

- If the JAVA cannot be found, use the command **export JAVA_HOME=/opt/sign/EABss7069/jre**
- If no X11 DISPLAY variable was set, try to log out the SC-1, and then log on again by using the -X option:

```
# ssh -X root@<SC-1 IP Address>
```

5. Select **Tools > Expert Mode** and **Tools > Configuration Mode > Initial**.

Note: **Expert Mode** enables all the properties to be visible in the Signaling Manager.

Procedure:

To configure SS7 to support Scaling operations, do the following:

Attention!

Traffic will be lost for approximately 30 seconds during the SS7 stack restart. If SS7 configuration is modified, the SS7 stack is required to restart to take effect the change.

1. Back up SS7 configuration files on SC-1.



```
#cp /opt/sign/etc/active.om.cim /opt/sign/etc/active.om.cim.pr  
echeck.bak
```

2. LDE MIP address is used as Common Parts manager address. Refer to the following table that complies with the rule:

Navigation Pane	Operation Pane Properties	Value
System Components > System Components	CP Manager Address	ss7cafcpmaddress:6669
System Components > System Components > CP > CP	If Alias	On
System Components > System Components > ECM > ECM	Connection Time Wait	25

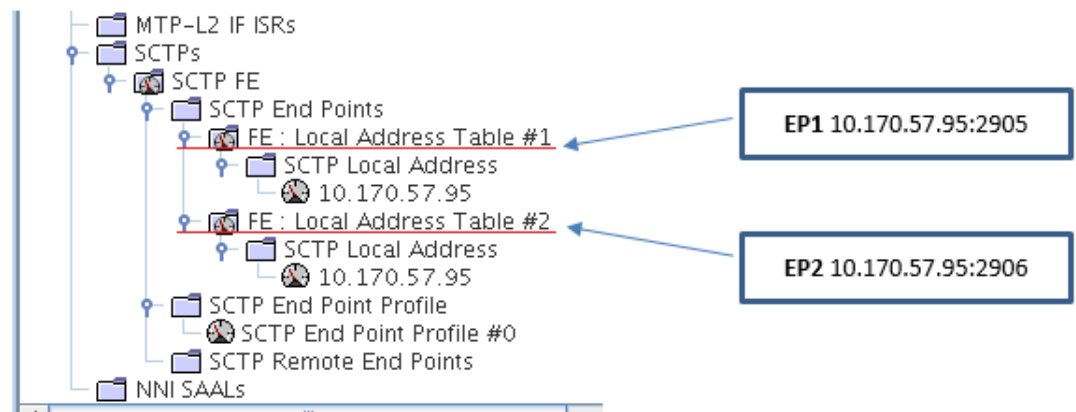
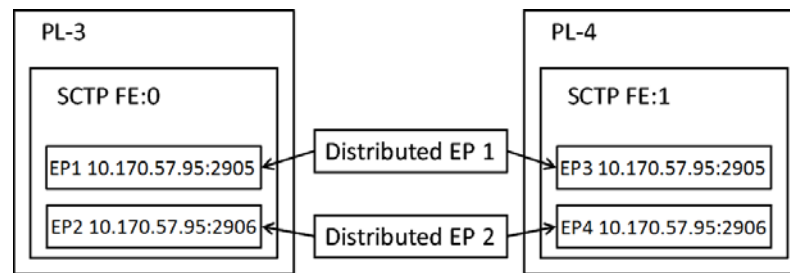
3. Asynchronous connection is enabled for BE, FE and NMP processes. Refer to the following table that complies with the rule:

Navigation Pane	Operation Pane Properties	Value
System Components > System Components > CP > CP	Msg Conn Time Wait	25
System Components > System Components > ECM > ECM> Process Classes > SCTP FEP	Command	"-w 5" should be added to launching command. For example: <code>/opt/sign/EABss7052/bin /fe_sctp -e 255 -u 161 -a 1 -o 5 -w 5</code>
System Components > System Components > ECM > ECM> Process Classes > GEN RP	Command	"-w 5" should be added to launching command. For example: <code>/opt/sign/EABss7053/bin /be -b 3 -u 161 -a 5 -o 1 -d 0 -w 5</code>
System Components > System Components > ECM > ECM> Process Classes > NMP	Command	"-w 5" should be added to launching command. For example: <code>/opt/sign/EABss7053/bin /be -b 2 -e 255 -u 161 -a 1 -w 5</code>

4. Only SCTP Distributed End Points (EPs) are configured in SCTP configuration.

Note: For Diameter over SCTP scenario, skip the step.

As the below SCTP Distributed End Point configuration (recommended by IPWorks) shows, two Distributed EPs are configured for SCTP FE. These two Distributed EPs use the same eVIP (e.g., 10.170.57.95) and different port (2905 and 2906).



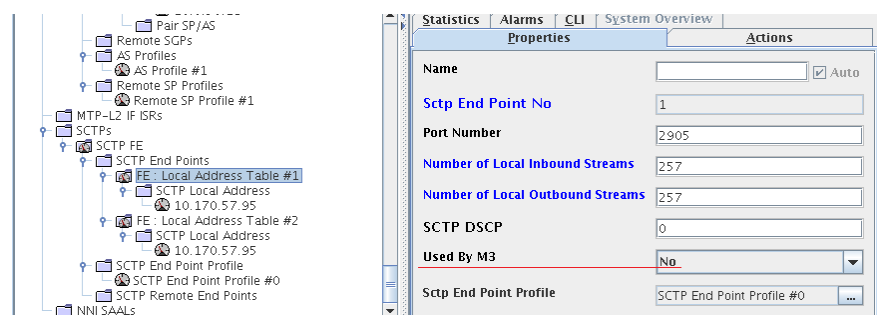
For SCTP Distributed End Points, check if below configurations are satisfied:

- a. Enable usage of eVIP functionality in SS7 CAF and usage of SCTP Distributed End Points feature.

Navigation Pane	Operation Pane Properties	Value
System Components > System Components > CP > CP	EVIP	on
M3UA IETF > M3UA	Distributed End Point Support	on

- b. For all SCTP End Points, the option **Used By M3** option is set to **No**.

For example:





For more information about the configurations of Distributed End Points, you can check the IPWorks SS7 configuration templates . Perform one of the following procedures depending on the IPWorks service to be used:

- For ENUM NP, refer to the section Starting From Template Configuration File in [Configure SS7 for ENUM Number Portability](#).
- For AAA, refer to the section Starting From Template Configuration File in [Configure SS7 for AAA](#).

5. If the SS7 configuration is modified, in order to make the change take effect, need to validate and restart the SS7 stack, do the following:
 - a. Validate the configuration by selecting **Edit > Validate**.
 - b. If there are validation errors, click **Results** to view error description and go to the respective configuration.
 - c. Select **Tools > Process View... > Configure** in the process view dialog box, and select **Initial Configuration** to make any update take effect.
 - d. Restart SS7 Stack on SC-1.

First, restart one SS7 stack:

```
amf-adm restart safSu=SC-1,safSg=2N,safApp=ERIC-ss7caf.mgmt
```

```
amf-adm restart safSu=SC-2,safSg=2N,safApp=ERIC-ss7caf.mgmt
```

```
amf-adm restart safSu=PL-3,safSg=2N,safApp=ERIC-ss7caf.netwcontrol
```

```
amf-adm restart safSu=PL-3,safSg=NWA,safApp=ERIC-ss7caf.core
```

After 1 minute, restart other ss7 stack:

```
amf-adm restart safSu=PL-4,safSg=2N,safApp=ERIC-ss7caf.netwcontrol
```

```
amf-adm restart safSu=PL-4,safSg=NWA,safApp=ERIC-ss7caf.core
```

- e. Select **File > Connect** and make sure that the status is **Active** in the status bar.
- f. Save the configuration file as another name by selecting **File> Save As**.
- g. Verify stack configuration.



- If IPWorks AAA is deployed:

Refer to the Verify Stack Configuration section in *Configure SS7 for AAA*.

- If IPWorks ENUM is deployed:

Refer to the Verify Stack Configuration section in *Configure SS7 for ENUM Number Portability*.

If verification of stack configuration fails, and the issue cannot be fixed. You need to restore the SS7 configuration with the following step:

Close the Signaling Manager, and then restore the SS7 configuration by using the SS7 configuration backup file.

```
#cp /opt/sign/etc/active.om.cim.precheck.bak
/opt/sign/etc/active.om.cim
```

Repeat the steps from Step d to Step g.

1.2.2 Configuring eVIP to Support Scaling Operations

Note: This procedure is ONLY applicable to the upgraded IPWorks, to ensure that eVIP configuration meets requirement of scaling operation.

Check the eVIP configuration by using ECLI, if only EvipNode=3 and EvipNode=4 are configured under *EvipCluster*, you need to further configure eVIP in Atlas CLI and ECLI.

View the information of *EvipCluster* configuration, for example:

```
# /opt/com/bin/cliss
```

```
>ManagedElement=<Node Name>,Transport=1,Evip=1,EvipDeclarations=1
,EvipCluster=1
```

```
(EvipCluster=1)>show
```

```
EvipCluster=1
  commandsForAllUndesignated
    "4:set_local_port_range"
    "3:set_default_route_ipv6_sig"
    "2:set_default_route_ipv4_sig"
    "1:flush_ipv6_default"
    "0:flush_route_cache"
  primaryInterface="eth0"
  EvipNode=4
  EvipNode=3
```



To further configure eVIP, do the following:

1. Ensure that both Data Nodes are active.

```
# /etc/init.d/ipworks.mysql show-status
```

```
Connected to Management Server at: localhost:1186
Cluster Configuration
```

```
-----
[ndbd(NDB)]      2 node(s)
id=27   @169.254.100.1  (mysql-5.6.31 ndb-7.4.12, Nodegroup: 0, *)
id=28   @169.254.100.2  (mysql-5.6.31 ndb-7.4.12, Nodegroup: 0)
```

```
[ndb_mgmd(MGM)] 2 node(s)
id=1     @169.254.100.1  (mysql-5.6.31 ndb-7.4.12)
id=2     @169.254.100.2  (mysql-5.6.31 ndb-7.4.12)
```

```
[mysqld(API)]   24 node(s)
id=3     @169.254.100.1  (mysql-5.6.31 ndb-7.4.12)
id=4 (not connected, accepting connect from SC-2)
```

```
...
```

2. Log on to the Atlas CLI, then stop the SC-1.

```
#ssh atlasadm@<Atlas_Addr>
```

```
atlasadm@atlas:~$source openrc
```

```
atlasadm@atlas:~$nova stop <instance-name of SC-1>
```

The instance-name can be found by using command:

```
atlasadm@atlas:~$nova list
```

For example:

ID	Name	Status	Task State	Power State	Network
0ab6ab6e-5b25-4f71-9894-105866d078a8	sub14_22_PL-3	ACTIVE	-	Running	su
eff38341-4a6d-40ec-8f2a-fe30288fdaf0	sub14_22_PL-4	ACTIVE	-	Running	su
02a6436e-73d7-41ec-bf22-bbf01d29e4e0	sub14_22_SC-1	ACTIVE	-	Running	su
ee9ea2c3-cbb5-46fb-9e4f-c0a67f99011c	sub14_22_SC-2	ACTIVE	-	Running	su

As the example shows, the instance name of SC-1 is sub14_22_SC-1. So the nova stop command is:

```
atlasadm@atlas:~/ $ nova stop sub14_22_SC-1
```

3. When SC-1 is shutdown, log on to SC-2, and add an eVIP node for SC-1.

```
SC-2:~# /opt/com/bin/cliss
```



- ```
> ManagedElement=<Node Name>,Transport=1,Evip=1,EvipDeclarations=1,EvipCluster=1

(EvipCluster=1)>configure

(config-EvipCluster=1)>EvipNode=1

(config-EvipNode=1)>hostname="SC-1"

(config-EvipNode=1)>commit

(config-EvipNode=1)>exit
```
4. Start the SC-1 in Atlas CLI.
 

```
atlasadm@atlas:~$nova start <instance-name of SC-1>
```
  5. Wait until services on SC-1 are up. MySQL Data Node must be started.  
To see the MySQL cluster status:
 

```
/etc/init.d/ipworks.mysql show-status
```
  6. Stop the SC-2 in Atlas CLI.
 

```
atlasadm@atlas:~$nova stop <instance-name of SC-2>
```
  7. When SC-2 is shutdown, log on to SC-1, and add an eVIP node for SC-2.
 

```
SC-1:~# /opt/com/bin/cliss

> ManagedElement=<Node Name>,Transport=1,Evip=1,EvipDeclarations=1,EvipCluster=1

(EvipCluster=1)>configure

(config-EvipCluster=1)>EvipNode=2

(config-EvipNode=2)>hostname="SC-2"

(config-EvipNode=2)>commit

(config-EvipNode=2)>exit
```
  8. Start the SC-2 in Atlas CLI, and wait until services on SC-2 are up.
 

```
$nova start <instance-name of SC-2>
```
  9. Add other eVIP configuration on SC-1 or SC-2 by executing script.
 

```
#/opt/ipworks/common/scripts/add_evip_configuration.py

Add evip node ...
Add evip lbe for EvipAlb=ipw_sig_sp
```



```
Add evip se for EvipAlb=ipw_sig_sp
Add evip fee for EvipAlb=ipw_sig_sp
Add evip lbe for EvipAlb=ipw_data_sp
Add evip se for EvipAlb=ipw_data_sp
Add evip fee for EvipAlb=ipw_data_sp
Done
```

## 1.3 Related Information

Trademark information, typographic conventions, definitions, and explanations of acronyms and terminology can be found in the following documents:

- [Glossary of Terms and Acronyms](#)
- [Trademark Information](#)
- [Typographic Conventions](#)



## 2 Scaling Procedure

Following topics are included in this section:

- Create backup, see Section 2.1 Creating Backup before Scaling on page 11.
- Scale-out/in through one of the methods:
  - Using CLI, see Section 2.2 Operation of Scale-Out through CLI on page 11 and Section 2.3 Operation of Scale-In through CLI on page 21
  - Using VNF-LCM, see Section 2.4 Operation of Scale Out/In through VNF-LCM on page 27
- Perform health check, Section 2.5 Scaling Health Check on page 28
- Create the final backup, see Section 2.6 Creating the Final Backup on page 31.

Create a compute node in IPWorks VNF as part of the scale-out operation is out of the scope of this document. However, the backup creation before and after scaling operation are required and this is a part of the scaling procedure.

### 2.1 Creating Backup before Scaling

Both System data backup and user data backup must be executed before scaling. For more details, refer to the documents [Backup](#) and [Restore](#).

### 2.2 Operation of Scale-Out through CLI

#### 2.2.1 Overview

Scale-out, means new VMs are instantiated. Those instances are added to the IPWorks VNF automatically.

Follow the instructions given by the cloud management system about how to create a VM instance. There are two phases to complete scale-out operation:

1. Use OpenStack related command to add a new PL in the cloud, refer to Step 2.
2. Monitor the scale-out progress in ECLI until the scale-out process has ended, refer to Section 2.2.4 Monitoring the Scale-Out Progress on page 19.

Followed the operation needed for different IPWorks:

- For deployed IPWorks stack, check if it is scalable or not. Refer to Section 2.2.2 Deploying IPWorks Stack on page 12.



- If it supports scaling, follow Section 2.2.3 Operating Scale-Out on page 16 to execute scaling operation.
  - If it does not support scaling, follow Section 2.2.2 Deploying IPWorks Stack on page 12 to support scaling function first, and then follow to execute scaling operation.
- For IPWorks 1.9 (or higher) which already supports scaling, just follow Section 2.2.3 Operating Scale-Out on page 16 to execute scaling operation.

**Note:** For scale-out operation, due to the elastic nature of the VNF, the PLs in the cluster could be hosted by any active VMs allocated to the VNF. There is no strict mapping between the VM name and the PL name.

## 2.2.2 Deploying IPWorks Stack

To deploy IPWorks stack:

1. Connect to Atlas.

```
#ssh atlasadm@<ATLAS_VM_IP_ADDRESS>
```

```
atlasadm@atlas:~$source openrc
```

These steps are same as the steps in section **Deploy IPWorks By Using Atlas CLI** in document *IPWorks Deployment Guide*.

2. Check if IPWorks stack supports scaling.

```
atlasadm@atlas:~$openstack stack resource list <IPW_STACK_NAME>
```

If it returns scaling resource with OS::Heat::ResourceGroup in resource\_type, it means that the deployed IPWorks stack supports scaling.

```
atlasadm@atlas:/var/archives/epengta$ openstack stack resource list sub23_mini_lsv21_pretest
```

| resource_name     | physical_resource_id                 | resource_type           | resource_status |
|-------------------|--------------------------------------|-------------------------|-----------------|
| ipw_PL-3          | d2b91f51-e25f-4995-88f2-8d71b22189d3 | OS::Nova::Server        | CREATE_COMPLETE |
| ipw_PL-4          | 7a214535-4117-4af6-b467-3b5232036db5 | OS::Nova::Server        | CREATE_COMPLETE |
| ipw_SC-1          | 9509bc52-94f1-486e-aalc-86e9fe8306c6 | OS::Nova::Server        | CREATE_COMPLETE |
| ipw_SC-2          | cf7b7cc2-f38b-4464-8fe6-28ea915982e2 | OS::Nova::Server        | CREATE_COMPLETE |
| ipw_oam_sp_subnet | cf394677-a6fe-490c-bb98-730d4400c833 | OS::Neutron::Subnet     | CREATE_COMPLETE |
| ...               |                                      |                         |                 |
| ipw_sig_sp_subnet | bcd1e25-2833-464d-b4f6-8b356b3dfefa  | OS::Neutron::Subnet     | CREATE_COMPLETE |
| scaling           | 05898345-d4ff-46c9-ac36-babc12f78e9e | OS::Heat::ResourceGroup | UPDATE_COMPLETE |

**Note:** If the IPWorks stack support scaling, a file named ipw\_scaling\_group.yaml is in /home/atlasadm/temp/mode22/. Then skip all the rest steps in this section and go to Section 2.2.3 Operating Scale-Out on page 16 to continue scaling operation directly.

3. Go to IPWorks VNF Utility directory.





**Note:** The folder must be the one which is used to deploy the IPWorks VNF before.

```
atlasadm@atlas:~$cd <IPW_VNF_UTIL_DIR>
```

```
atlasadm@atlas:~$cd <IPW_VNF_UTIL_DIR>/temp/mode22/
```

4. Check if there is a `ipw_scaling_group.yaml` file.
  - For the new deployed IPWorks 1.9 or higher, the `ipw_scaling_group.yaml` file is existed.
  - For the upgraded IPWorks, the `ipw_scaling_group.yaml` file is not existed. Create a new `ipw_scaling_group.yaml` according to the following example:



```

heat_template_version: '2013-05-23'
description: Generic Scaling file for vIPWorks
parameters:
 index:
 description: availability zone index
 type: number
 availability_zones_for_scaling:
 description: Availability zone used for instance creation
 type: comma_delimited_list
 stack_name:
 description: Name of vIPWorks stack_name
 type: string
 image_for_scaling:
 description: Name of the Glance image to create PL volumes
 type: string
 flavor_for_scaling:
 description: Name of the PL flavor
 type: string
 instance_number:
 description: Number of the instance
 type: string
 pl_ha_policy:
 description: HA policy for PL VMs
 type: string
 default: 'ha-offline'
resources:
 ###Creating Neutron ports###
 PL-x_eth0:
 type: OS::Neutron::Port
 properties:
 name: { list_join: ['_'], [{ get_param: stack_name }, PL, { get_param: instance_number }] }
 network: { list_join: ['_'], [{ get_param: stack_name }, 'int_sp'] }
 port_security_enabled: false
 PL-x_eth1:
 type: OS::Neutron::Port
 properties:
 name: { list_join: ['_'], [{ get_param: stack_name }, PL, { get_param: instance_number }] }
 network: { list_join: ['_'], [{ get_param: stack_name }, 'sig_sp'] }
 port_security_enabled: false
 PL-x_eth2:
 type: OS::Neutron::Port
 properties:
 name: { list_join: ['_'], [{ get_param: stack_name }, PL, { get_param: instance_number }] }
 network: { list_join: ['_'], [{ get_param: stack_name }, 'data_sp'] }
 port_security_enabled: false
 PL-x:
 type: OS::Nova::Server
 depends_on:
 - PL-x_eth0
 - PL-x_eth1
 - PL-x_eth2
 properties:
 availability_zone: { get_param: [availability_zones_for_scaling, { get_param: index }] }
 name: { list_join: ['_'], [{ get_param: stack_name }, Scale, { get_param: instance_number }] }
 image: { get_param: image_for_scaling }
 flavor: { get_param: flavor_for_scaling }
 metadata: { 'ha-policy': { get_param: pl_ha_policy } }
 networks:
 - port: { get_resource: PL-x_eth0 }
 - port: { get_resource: PL-x_eth1 }
 - port: { get_resource: PL-x_eth2 }

```

**Note:** This file is the HOT scaling group template and it must not be modified. It is used by the main HOT template yaml file directly.

## 5. Get the onboarding HOT yaml file.

- a. Identify the stack name <stack\_name> to be used in next command:

```
atlasadm@atlas:~$openstack stack list
```



- b. Obtain the onboarding HOT yaml file:

```
atlasadm@atlas:~$openstack stack template show
<stack_name> >ipw_hot_onboarding.yaml
```

6. Modify HOT template file to support scaling.

Modify the content of the onboarding HOT `ipw_hot_onboarding.yaml`:

- a. Add this structure at the end of resources section.

---

### Warning!

It is important to keep the right index of the parameters for any modification of your onboarding HOT file.

---

```
-----###
VNF Scaling part
-----###
scaling:
 type: OS::Heat::ResourceGroup
 properties:
 count: { get_param: number_of_total_scaled_vms }
 removal_policies:
 - resource_list:
 get_param: list_of_vms_to_scale_in
 resource_def:
 properties:
 index: "%index%"
 availability_zones_for_scaling: { get_param: availability_zones_for_scal
 stack_name: { get_param: VNF_NAME }
 image_for_scaling: { get_param: PL_IMAGE }
 flavor_for_scaling: { get_param: PL_FLAVOR_NAME }
 instance_number: "%index%"
 type: ipw_scaling_group.yaml
```

**Note:** The `ipw_scaling_group.yaml` is used, and the `flavor_for_scaling` is the one defined in parameter **PL\_FLAVOR\_NAME** in [IPWorks Deployment Guide](#).

- b. Add the following parameters to the parameter group and parameter section.

— In parameter group section, add the following:

```
- description: IPWorks Scaling related Parameters
 label: IPWorks Scaling Parameters
 parameters: [availability_zones_for_scaling, list_of_vms_to_scale_in,
number_of_total_scaled_vms]
```

— In parameter section, add the following:



```
availability_zones_for_scaling: {default: [], description: availability zone to be used for scaled out VMs, type: con
list_of_vms_to_scale_in:
 default: []
 description: List of PLs to be scaled in
 type: comma_delimited_list
number_of_total_scaled_vms: {default: 0, description: The number of PLs to be scaled,
 type: number}
```

After all modification steps, the new IPWorks HOT yaml file which supports scaling is `ipw_hot_onboarding.yaml`. The new HOT scaling group yaml file is `ipw_scaling_group.yaml`. Both yaml files must be in the same folder. Users can follow the Section 2.2.3 Operating Scale-Out on page 16 to execute scaling operations.

### 2.2.3 Operating Scale-Out

1. Before the scale-out operation, stack name must be known. To determine the stack name, use the following command:

```
atlasadm@atlas:~$openstack stack-list
```

2. Perform IPWorks VNF scale-out.

There are two ways to perform the scale-out operation:

- Scale-out without specifying availability zone. Scaled-out PLs will be launched in availability zone specified by CEE:

```
atlasadm@atlas:~$ openstack stack update <stack_name> -t
<ipw_hot_onboarding.yaml> -e <env_file.yaml> --parameter
"number_of_total_scaled_vms=<number_of_scaled_vms>"
--rollback enabled
```

Where:

|                            |                                                                                                                                               |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| number_of_total_scaled_vms | The parameter specifies the total number of scaled PLs to be presented in IPWorks VNF after the scale-out operation. The maximum number is 8. |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|

**Note:**

- You can find the `yaml` from the original installation directory `IPW_VNF_UTIL_DIR` on the Atlas machine.

For example:

```
<ipw_hot_onboarding.yaml>: <IPW_VNF_UTIL_DIR>/temp
/mode22/<ipw-vnf-22*>.yaml
```

```
<env_file.yaml>: <IPW_VNF_UTIL_DIR>/tmp/<stack-name>
>/<stack-name>_env.yaml
```

If the proceeding files do not exist, refer to section **Deploying IPWorks By Atlas GUI** in document *IPWorks Deployment Guide* to generate.

- The `number_of_total_scaled_vms` includes all the scaled-out PLs in previous scale-out operations.

For example:

The first scale-out operation to scale out 2 PLs specifies "`number_of_total_scaled_vms=2`". The second scale-out operation to scale out 1 additional PL must specify "`number_of_total_scaled_vms=3`".

- Scale-out with specifying availability zone:

```
atlasadm@atlas:~$ openstack stack update <stack_name> -t
<ipw_hot_onboarding.yaml> -e <env_file.yaml> --parameter
"number_of_total_scaled_vms=<number_of_scaled_vms>;availabi
lity_zones_for_scaling=<available_zone list>" --rollback
enabled
```

Where:

|                                             |                                                                                                                                               |
|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| <code>number_of_total_scaled_vms</code>     | The parameter specifies the total number of scaled PLs to be presented in IPWorks VNF after the scale-out operation. The maximum number is 8. |
| <code>availability_zones_for_scaling</code> | The parameter specifies the list of availability zones for scaled-out PLs to launch in the availability zone one by one.                      |

For example,

- When specifying "`number_of_total_scaled_vms=2`" and "`availability_zones_for_scaling=nova:compute-1-11.domain.tld,nova:compute-1-12.domain.tld`", the `Scale_0` VM will be launched in availability zone "`nova:compute-1-11.domain.tld`" and `Scale_1` VM will be launched in availability zone "`nova:compute-1-12.domain.tld`".
- Each scale-out command must include availability zone lists used in all the previous scale-out operations.



The first scale-out operation to scale out 2 PLs specifies “number\_of\_total\_scaled\_vms=2” and “availability\_zones\_for\_scaling=nova:compute-1-11.domain.tld,nova:compute-1-12.domain.tld”. The second scale-out operation to scale out 1 additional PL must specify “number\_of\_total\_scaled\_vms=3” and “availability\_zones\_for\_scaling=nova:compute-1-11.domain.tld,nova:compute-1-12.domain.tld,nova:compute-1-13.domain.tld”.

#### Note:

- The number\_of\_total\_scaled\_vms must be same as the size of availability\_zones\_for\_scaling. Otherwise, the scaled-out VM will be launched in the specified availability zones first, and the rest VM will be launched in available zone scheduled by CEE.
- If the availability\_zones\_for\_scaling list specifies the different availability zones for already scale-out VMs, the already scale-out VM will be rebuilt and moved to the new availability zone.

After the successful scale-out operation, the active PLs are displayed in Nova list.

- An onboarding stack with 2 PLs was previously scaled-out with one additional PL. This initial situation has the following instanced VMs: SC-1, SC-2, PL-3, PL-4 and PL-5, where PL-5 is the scaled VM that corresponds to Scale\_0. The following output is in a nova list:

|                                      |                                  |        |   |         |       |
|--------------------------------------|----------------------------------|--------|---|---------|-------|
| e6a56147-c7f0-4907-a1b1-602ccc2642ab | sub23_mini_lsv21_pretest_PL-3    | ACTIVE | - | Running | sub23 |
| 0a50af05-5e9a-4446-8c84-750e07c05790 | sub23_mini_lsv21_pretest_PL-4    | ACTIVE | - | Running | sub23 |
| 3b96becf-9294-479d-8d69-9ce6a514e3ec | sub23_mini_lsv21_pretest_SC-1    | ACTIVE | - | Running | sub23 |
| f8b225bd-e235-44ff-b7e3-d9db9284709a | sub23_mini_lsv21_pretest_SC-2    | ACTIVE | - | Running | sub23 |
| 10b50afa-7901-4734-8d9a-f568af49ea59 | sub23_mini_lsv21_pretest_Scale_0 | ACTIVE | - | Running | sub23 |

- To add a new PL to the cluster (PL-6):

```
atlasadm@atlas:~$ openstack stack update sub23_m
ini_lsv21_pretest -t ipw_hot_onboarding.yaml -e
<env_file.yaml> --parameter "number_of_total_scaled_vms=2;
availability_zones_for_scaling=nova:compute-1-11.domain.tld
,nova:compute-1-12.domain.tld" --rollback enabled
```

In a nova list output, this is the expected output after stack update where PL-6 is the scaled VM that corresponds to Scale\_1:



|                                      |                                  |        |   |
|--------------------------------------|----------------------------------|--------|---|
| e6a56147-c7f0-4907-a1b1-602ccc2642ab | sub23_mini_lsv21_pretest_PL-3    | ACTIVE | - |
| 0a50af05-5e9a-4446-8c84-750e07c05790 | sub23_mini_lsv21_pretest_PL-4    | ACTIVE | - |
| 3b96becf-9294-479d-8d69-9ce6a514e3ec | sub23_mini_lsv21_pretest_SC-1    | ACTIVE | - |
| f8b225bd-e235-44ff-b7e3-d9db9284709a | sub23_mini_lsv21_pretest_SC-2    | ACTIVE | - |
| 10b50afa-7901-4734-8d9a-f568af49ea59 | sub23_mini_lsv21_pretest_Scale_0 | ACTIVE | - |
| a1fec792-d6ca-43c8-be33-1ac5fd43aa1a | sub23_mini_lsv21_pretest_Scale_1 | ACTIVE | - |

### 3. Monitor the scale-out status:

```
atlasadm@atlas:~$openstack stack list
```

or

```
atlasadm@atlas:~$openstack resource list <IPW_STACK_NAME>
```

Check the resource scaling status, if it shows UPDATE\_COMPLETE, it means that the scale-out operation is successful.

---

## Attention!

### Risk of data loss or data corruption!

The environment file must be the one which is generated when deploy the IPWorks, or which has been updated after any openstack stack update command related SC VM (for example, SC recovery). Otherwise, it will cause IPWorks VNF SC or PL rebuilding and data lost. For SC image, change SC\_IMAGE value in environment yaml file to UUID instead of image name to avoid SC rebuilding, which will crush the IPWorks. So when deployment, it is recommended to fill SC image UUID instead of SC image name. Never remove resources created by OpenStack manually (by commands like nova, neutron) because it can corrupt the database of OpenStack.

---

## 2.2.4 Monitoring the Scale-Out Progress

To monitor the scale-out progress, do the following:

1. Log in to ECLI.

```
#ssh <user>@<OAM_MIP> -p <port> -s -t cli
```

2. Navigate to the Scaling Management model information:

```
>ManagedElement=<Node name>,SystemFunctions=1,SysM=1,CrM=1
```

3. Verify that the scale-out process has started.

```
(CrM=1)>show -r
```



```
(CrM=1)>show -r
CrM=1
 autoRoleAssignment=ENABLED
 ...
 ComputeResourceRole=PL-5
 adminState=UNLOCKED
 instantiationState=INSTANTIATED
 operationalState=ENABLED
 provides="ManagedElement=1, SystemFunctions=1, SysM=1, CrM=1, Role=Default-Role"
 uses="ManagedElement=1, Equipment=1, ComputeResource=PL-5"
 ComputeResourceRole=PL-6
 adminState=UNLOCKED
 instantiationState=INSTANTIATING
 operationalState=ENABLED
 provides="ManagedElement=1, SystemFunctions=1, SysM=1, CrM=1, Role=Default-Role"
 uses="ManagedElement=1, Equipment=1, ComputeResource=PL-6"
 ...
```

This example shows that instantiationState has changed to INSTANTIATING for node PL-6. It means that the scale-out has started.

4. Continue to monitor the progress until the scale-out process has ended and the added node has joined the cluster:

```
(CrM=1)>show -m ComputeResourceRole -p instantiationState,operationalState
```

For example:

```
(CrM=1)>show -m ComputeResourceRole -p instantiationState,operationalState
```

```
ComputeResourceRole=PL-3
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-4
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-5
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-6
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=SC-1
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=SC-2
 instantiationState=INSTANTIATED
 operationalState=ENABLED
(CrM=1)>
```





## 2.3 Operation of Scale-In through CLI

### 2.3.1 Overview

Scale-in means that the existing instances are removed from the cluster and then its corresponding VMs are deleted. There are two kinds of scale-in operation:

- Graceful scale-in (Recommended): User removes the resource in SC ECLI first and then scale in the VM in CEE by using HOT template. The traffic is gracefully switched to active PL.

For the operation steps of graceful scale-in, refer to Section 2.3.2 Graceful Scale-In Operation on page 21.

- Forceful scale-in: Users must remove the resource in ECLI to clean the dirty data if one of the following case occurs:

- Graceful scale-in fails.
- The VM is down or removed accidentally.

For the operation steps of forceful scale-in, refer to Section 2.3.3 Forceful Scale-In Operating on page 21.

#### Note:

- Scale-in only can be operated on the PLs which have been scaled out before.
- For scale-in operation, any of the existing PL in the cluster could be removed except PL3 and PL4.
- If the Signaling Manager is open during scale-in, after scale-in is finished, users need to restart the Signaling Manager to get the latest state of the SS7 stack.

### 2.3.2 Graceful Scale-In Operation

To perform the graceful scale-in, do the following process:

1. Section 2.3.4 Remove PL from Cluster on page 22.

If the PL is failed to be removed due to some unexpected reason, go to the Section 2.3.3 Forceful Scale-In Operating on page 21 directly.

2. Section 2.3.5 Remove VM Instance on page 23.
3. Section 2.3.6 Monitoring the Scale-In Progress on page 26

### 2.3.3 Forceful Scale-In Operating

To perform forceful scale-in, do the following process:



1. If the PL to be scale-in is up, stop IPWorks service running on this PL. Take PL-5 as example:

```
#ipw-ctr status all | grep PL-5 -A20
```

```
#ipw-ctr stop <running_services> PL-5
```

For example: `#ipw-ctr stop aaa_radius_stack PL-5`

2. Section 2.3.5 Remove VM Instance on page 23.
3. Section 2.3.4 Remove PL from Cluster on page 22
4. Section 2.3.6 Monitoring the Scale-In Progress on page 26

### 2.3.4 Remove PL from Cluster

1. Log in to the COM CLI.

```
#ssh <user>@<OAM_MIP> -p <port> -s -t cli
```

2. Switch to configuration mode.

```
> configure
```

3. Go to compute ComputeResourceRole MO of the PL to be scaled in.

```
(config)>ManagedElement=<Node name>,SystemFunctions=1,SysM=1,
CrM=1,ComputeResourceRole=PL-<N>
```

For example:

```
(config)> ManagedElement=1,SystemFunctions=1,SysM=1,CrM=1,Co
mputeResourceRole=PL-5
```

4. Request to remove the PL.

```
(config-ComputeResourceRole=PL-<N>)> no provides
```

For example:

```
(config-ComputeResourceRole=PL-5)> no provides
```

5. Commit the change.

```
(config-ComputeResourceRole=PL-<N>)> up
```

```
(config-CrM=1)> commit
```

For example:

```
(config-ComputeResourceRole=PL-5)> up
```

```
(config-CrM=1)> commit
```



6. Monitor if the PL is removed.

Follow steps in Section 2.3.6 Monitoring the Scale-In Progress on page 26 to verify the transient states the PL goes through until this is removed.

## 2.3.5 Remove VM Instance

To remove the VM instances of the specific PLs:

1. Determine scale index of PL to be scaled-in.

**Note:** For the graceful scale-in, after the PL is removed from cluster in Section 2.3.4 Remove PL from Cluster on page 22, it appears as shutoff in Atlas. Proceed to Step 2 to scale in the shutoff node.

For the forceful scale-in, use internal PL MAC address to find the scale-in index in Atlas. Proceed with the following steps (take PL-5 as example).

- a. Find the internal eth0 MAC address of PL-5.

```
SC-x:~ # lde-config -p | grep "interface 5" | grep eth0
| awk '{print $5}'
```

**Note:** "interface x" corresponds to PL-x, and x stands for 5-12.

Example output:

The return MAC address is fa:16:3e:1b:0f:c4

- b. Login to Atlas CLI:

```
#ssh atlasadm@<Atlas_Addr>
atlasadm@atlas:~$source openrc
```

- c. Find the scale index for the PL.

```
$neutron port-list | grep <PL INTERNAL MAC address> |
awk -F"eth0" '{print $1}' | awk -F" " '{print $NF}'
```

In this example, the PL-5 internal MAC address is fa:16:3e:1b:0f:c4, so the command is:

```
$neutron port-list | grep "fa:16:3e:1b:0f:c4" | awk
-F"eth0" '{print $1}' | awk -F" " '{print $NF}'
```

Output:

```
sub23_mini_lsv21_pretest_Scale_0
```

In this example, the name of the scaled VM is sub23\_mini\_lsv21\_pretest\_Scale\_0, and the scale index is the trailing digit 0.



To double-check the scale-in index is correct, and the scale-in PL is the expected blade server, run the following command:

```
$nova console-log <VM_Name> | egrep "login|host"
```

Take the following as example:

```
$nova console-log sub23_mini_lsv21_pretest_Scale_0 |
egrep "login|host"
```

```
[0.000000] Linux version 3.12.61-52.89-default (geeko@buildhost) (gcc version 4.8
[0.103019] PCI: Using host bridge windows from ACPI; if necessary, use "pci=nocrs
[0.128637] PCI host bridge to bus 0000:00
[14.603698] systemd[1]: Set hostname to <linux>.
linux login: [234.289322] reboot: Restarting system
[0.000000] Linux version 3.12.61-52.89-default (geeko@buildhost) (gcc version 4.8
[0.133296] PCI: Using host bridge windows from ACPI; if necessary, use "pci=nocrs
[0.177178] PCI host bridge to bus 0000:00
[22.613225] systemd[1]: Set hostname to <linux>.
[[32m OK [0m] Reached target Node with type payload on host PL-5.
PL-5 login:
```

From the output PL-5 login, check if the PL-5 is the expected scale-in blade server, and the VM name of sub23\_mini\_lsv21\_pretest\_Scale\_0 corresponds to PL-5.

If users want to scale-in more than one scaled-out PLs, the users can repeat this step before go to next step.

## 2. Perform the scale-in to the desired number of PLs.

— To scale-in the specific PLs, execute the following command:

```
atlasadm@atlas:~$openstack stack update <stack_name> -t
<ipw_hot_onboarding.yaml> -e <ipw_env_file.yaml> --param
eter "number_of_total_scaled_vms=<number_of_scaled_vms>;
availability_zones_for_scaling=<available_zone_list>;list_
of_vms_to_scale_in=<list_of_vms_to_scale_in>" --rollback
enabled
```

Where:



|                                |                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| number_of_total_scaled_vms     | It corresponds to the number of scaled PLs to be presented in the system after the scale in operation.                                                                                                                                                                                                                                                                                                                                   |
| availability_zones_for_scaling | <p>The parameter specifies the list of availability zones for scaled-out PLs to launch in the availability zone one by one.</p> <p>For scale-in operation, the availability zones listed in this parameter must match the scale index order of remaining scale-out PL after scale-in operation (from lower ID to higher ID). If all scaled-out PLs are to be scaled-in, the parameter availability_zones_for_scaling can be ignored.</p> |
| list_of_vms_to_scale_in        | <p>It is a list of numbers separated by comma, with each number corresponding to the scale index of PL removed from cluster in a scale-in process. For example, 0,1,2.</p> <p>This is an optional parameter.</p> <p>If not stated, the last scaled PL is removed. Use this parameter especially when the PL removed from the Cluster corresponds to a node that is not the last scaled node.</p>                                         |

For example:

- In a situation with an onboarding stack with 2 PLs that was previously scaled-out with two additional PL. This initial situation has the following instanced VMs: SC-1, SC-2, PL-3, PL-4, PL-5, PL-6. PL-5 and PL-6 are scaled VMs that corresponds to Scale\_0 and Scale\_1. In a nova list, the output is like this:

|                                      |                                  |        |   |         |
|--------------------------------------|----------------------------------|--------|---|---------|
| e6a56147-c7f0-4907-a1b1-602ccc2642ab | sub23_mini_lsv21_pretest_PL-3    | ACTIVE | - | Running |
| 0a50af05-5e9a-4446-8c84-750e07c05790 | sub23_mini_lsv21_pretest_PL-4    | ACTIVE | - | Running |
| 3b96becf-9294-479d-8d69-9ce6a514e3ec | sub23_mini_lsv21_pretest_SC-1    | ACTIVE | - | Running |
| f8b225bd-e235-44ff-b7e3-d9db9284709a | sub23_mini_lsv21_pretest_SC-2    | ACTIVE | - | Running |
| 10b50afa-7901-4734-8d9a-f568af49ea59 | sub23_mini_lsv21_pretest_Scale_0 | ACTIVE | - | Shutoff |
| a1fec792-d6ca-43c8-be33-1ac5fd43aa1a | sub23_mini_lsv21_pretest_Scale_1 | ACTIVE | - | Running |

- Where Scale\_0 is the one that corresponds to PL-5 previously removed from Cluster that appears as shutoff:

```
atlasadm@atlas:~$ openstack stackupdate sub23_mini_lsv2
1_pretest -t ipw_hot_onboarding.yaml -e <env_file.yaml>
--parameter "number_of_total_scaled_vms=1;
availability_zones_for_scaling=nova:compute-1-12.
domain.tld;list_of_vms_to_scale_in=0" --rollback enabled
```

This is the expected output after stack\_update where there is only one scaled VM Running and ACTIVE left that corresponds to Scaled\_1:

|                                      |                                  |        |   |         |
|--------------------------------------|----------------------------------|--------|---|---------|
| e6a56147-c7f0-4907-a1b1-602ccc2642ab | sub23_mini_lsv21_pretest_PL-3    | ACTIVE | - | Running |
| 0a50af05-5e9a-4446-8c84-750e07c05790 | sub23_mini_lsv21_pretest_PL-4    | ACTIVE | - | Running |
| 3b96becf-9294-479d-8d69-9ce6a514e3ec | sub23_mini_lsv21_pretest_SC-1    | ACTIVE | - | Running |
| f8b225bd-e235-44ff-b7e3-d9db9284709a | sub23_mini_lsv21_pretest_SC-2    | ACTIVE | - | Running |
| a1fec792-d6ca-43c8-be33-1ac5fd43aa1a | sub23_mini_lsv21_pretest_Scale_1 | ACTIVE | - | Running |

- Monitor the status until the stack\_status shows UPDATE\_COMPLETE for your <stack\_name>:

```
atlasadm@atlas:~$ openstack stack list
```



```
atlasadm@atlas:~$openstack resource list <IPW_STACK_NAME>
```

### 2.3.6 Monitoring the Scale-In Progress

1. Log in to ECLI.

```
#ssh <user>@<OAM_MIP> -p <port> -s -t cli
```

2. Navigate to the Scaling Management model information:

```
>ManagedElement=<Node name>,SystemFunctions=1,SysM=1,CrM=1
```

3. Verify that the scaling process has started.

```
(CrM=1)>show -r
```

The following are the example outputs for the scale-in process:

```
(CrM=1)>show -r
CrM=1
 autoRoleAssignment=ENABLED
 ...
 ComputeResourceRole=PL-5
 adminState=SHUTTINGDOWN
 instantiationState=UNINSTANTIATING
 operationalState=ENABLED
 provides="ManagedElement=1, SystemFunctions=1, SysM=1, CrM=1, Role=Default-Role"
 uses="ManagedElement=1, Equipment=1, ComputeResource=PL-5"
 ComputeResourceRole=PL-6
 adminState=UNLOCKED
 instantiationState=INSTANTIATED
 operationalState=ENABLED
 provides="ManagedElement=1, SystemFunctions=1, SysM=1, CrM=1, Role=Default-Role"
 uses="ManagedElement=1, Equipment=1, ComputeResource=PL-6"
 ...
(CrM=1)>show -r
CrM=1
 autoRoleAssignment=ENABLED
 ...
 ComputeResourceRole=PL-5
 adminState=LOCKED
 instantiationState=UNINSTANTIATING
 operationalState=DISABLED
 provides="ManagedElement=1, SystemFunctions=1, SysM=1, CrM=1, Role=Default-Role"
 uses="ManagedElement=1, Equipment=1, ComputeResource=PL-5"
 ComputeResourceRole=PL-6
 adminState=UNLOCKED
 instantiationState=INSTANTIATED
 operationalState=ENABLED
 provides="ManagedElement=1, SystemFunctions=1, SysM=1, CrM=1, Role=Default-Role"
 uses="ManagedElement=1, Equipment=1, ComputeResource=PL-6"
 ...
```

**Note:** This example shows that instantiationState has changed to UNINSTANTIATING for node PL-5. It means that the scale-in has started. The adminState changes first to SHUTTINGDOWN and then to LOCKED and operationalState changes to DISABLED.

4. Continue to monitor the progress.

```
(CrM=1)>show -m ComputeResourceRole -p instantiationState,operationalState
```



The expected result:

```
(CrM=1)>show -m ComputeResourceRole -p instantiationState,operationalState
ComputeResourceRole=PL-3
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-4
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-6
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=SC-1
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=SC-2
 instantiationState=INSTANTIATED
 operationalState=ENABLED
(CrM=1)>
```

This example shows that node PL-5 has disappeared. It means that PL-5 is removed from the cluster.

**Note:** After PL-5 is scaled-in, command `tipc-config -n` will show PL-5 as unknown status, this is a known behavior. When PL-5 is scaled-out again, the status will be updated to correct one.

However, if the scaling process fails, you will receive the following result:

```
(CrM=1)>show -m ComputeResourceRole -p instantiationState,operationalState
ComputeResourceRole=PL-3
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-4
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=PL-5
 instantiationState=UNINSTANTIATION_FAILED
 operationalState=ENABLED
ComputeResourceRole=PL-6
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=SC-1
 instantiationState=INSTANTIATED
 operationalState=ENABLED
ComputeResourceRole=SC-2
 instantiationState=INSTANTIATED
 operationalState=ENABLED
(CrM=1)>
```

This example shows that the value of `instantiationState` is changed to `UNINSTANTIATION_FAILED` for node PL-5. It means that the PL-5 is not removed from the cluster.

## 2.4 Operation of Scale Out/In through VNF-LCM

Scale out/in IPWorks through VNF-LCM, refer to section [Scale VNF in IPWorks VNF Life Cycle Management \(Workflow\) - Small Stack](#).



## 2.5 Scaling Health Check

The health check to be performed before and after a scaling operation is listed in this section. Additionally an entire IPWorks health check can be performed, for more information, refer to [IPWorks Manual Health Check](#).

**Note:** It is not recommended to proceed with the scaling operation if the result of the health check is not successful. For troubleshooting, refer to [IPWorks Troubleshooting Guideline](#).

The health check described below can be executed at once running this script:

```
#ssh root@<SC OAM MIP>
#cd /opt/ipworks/common/scripts
#./ipw_scale_hc.sh
```

Check the output if the needed log file created. Followed the example output:





```

Scaling Health Check started

#####
CHECK ping

PASSED: ping status is OK

#####
CHECK ss7

PASSED: ss7 status is OK

#####
CHECK instantiationState

PASSED: instantiationState status is OK

#####
CHECK cmwstatusnode

PASSED: cmwstatusnode status is OK

#####
CHECK cmwscalingconf

PASSED: cmwscalingconf status is OK

#####
CHECK appl

PASSED: appl status is OK

#####
CHECK servicetype

The IPWorks Service Type support scaling #

PASSED: servicetype status is OK

#####
CHECK evip

PASSED: evip status is OK

HEALTHCHECK: PASSED
Logfile: /cluster/storage/no-backup/ipworks/scaling/scalehc_20170827_233634.log

```

1. Check the state of the following system items at Core Middleware (Core MW) level.

**cmw-status node app csiass comp node sg si siass su**

For example:

```
SC-X:~ # cmw-status node app csiass comp node sg si siass su
safComp=CXP9033814,safSu=PL-3,safSg=NWA,safApp=ERIC-sv.SVDNSSM
OperState=ENABLED(1)
PresenceState=UNINSTANTIATED(1)
```



```

ReadinessState=OUT-OF-SERVICE(1)
safComp=CXP9033814,safSu=PL-4,safSg=NWA,safApp=ERIC-sv.SVDNSSM
OperState=ENABLED(1)
PresenceState=UNINSTANTIATED(1)
ReadinessState=OUT-OF-SERVICE(1)
safComp=CXP9033812,safSu=PL-4,safSg=NWA,safApp=ERIC-sv.SVAAASM
OperState=ENABLED(1)
PresenceState=UNINSTANTIATED(1)
ReadinessState=OUT-OF-SERVICE(1)
...

```

**Note:** All failed items are printed. Make sure the configured services are not printed as the value of safApp.

2. Check that all the SS7 processes are in Running state.

```

echo -e ' procp;\ndisconnect;\nexit' | /opt/sign/EABss7050/bin/
signmcli -own.conf=/cluster/storage/system/config/ss7caf-ana901
37/etc/signmgr.cnf -online=yes

```

For example:

```

SS7 PROCESS STATES
cli> connect;
EXECUTED
cli> procp;
Process State
GEN RP:1 [PL-3] Running
GEN RP:2 [PL-4] Running
GEN RP:3 [PL-5] Running
SCTP FEP:0 [PL-3] Running
SCTP FEP:1 [PL-4] Running
SCTP FEP:2 [PL-5] Running
NMP:0 [PL-3] Running
OAMP:0 [PL-3] Running
LOGD:0 [PL-3] Running
ECM:0 [PL-3] Running
ECM:1 [PL-4] Running
ECM:2 [PL-5] Running
ECSP:0 [PL-3] Running
ECSP:1 [PL-4] Running
ECSP:2 [PL-5] Running
SAFOAM:0 [PL-3] Running
cli> disconnect;
EXECUTED
cli> exit;

```



## 2.6 Creating the Final Backup

Create a backup after the scaling is performed following the same steps as described in Section **Create Initial Backup**, name it AFTER\_SCALE\_PL\_<Number of \_PLS\_after\_Scaling>.





## Reference List

- [1] IPWorks Initial Configuration, 5/1553-AVA 901 33/3
- [2] IPWorks Manual Health Check
- [3] IPWorks Troubleshooting Guideline
- [4] Backup and Restore
- [5] IPWorks Deployment Guide
- [6] IPWorks VNF Life Cycle Management (Workflow) - Small Stack
- [7] IPWorks Deployment Guide, 21/1553-AVA 901 33/3
- [8] BRF-C Management Guide, 9/1553-APR 901 0444/4