



Install and Upgrade Patch Deployment Guide 2020.3.0 FP6

Version: 2020.3.0

Patch Deployment Guide FP6

The purpose of this document is to guide the users for applying patches on AppViewX v2020.3.0 FP6.

Copyright AppViewX, Inc.

Copyright © 2020 AppViewX, Inc. All Rights Reserved.

This document may not be copied, disclosed, transferred, or modified without the prior written consent of AppViewX, Inc. While all content is believed to be correct at the time of publication, it is provided as general-purpose information. The content is subject to change without notice and is provided “as is” and with no expressed or implied warranties whatsoever, including, but not limited to, a warranty for accuracy made by AppViewX. The software described in this document is provided under written license only, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. Unauthorized use of software or its documentation can result in civil damages and criminal prosecution.

Trademarks

The trademarks, logos, and service marks displayed in this manual are the property of AppViewX or other third parties. Users are not permitted to use these marks without the prior written consent of AppViewX or such third party which may own the mark.

External Reference Links

This product includes software developed by the CentOS Project (www.centos.org).

This product includes software developed by Red Hat, Inc. (www.redhat.com).

This product includes software developed by VMware, Inc. (www.vmware.com).

All other trademarks mentioned in this document are the property of their respective owners.

Contact Information

AppViewX, Inc.

222 Broadway, FL 19

New York, NY 10038

Email: info@appviewx.com

Web: www.appviewx.com

Contents

Patch Deployment Guide FP6	ii
Preface.....	5
Revision History.....	5
About this Guide	5
Audience.....	5
Text Conventions.....	5
Chapter 1. Expectations.....	6
.....	6
Chapter 2. Part 1 - Plugins and Addons Upgrade.....	7
Chapter 3. Part 2 - Kubernetes Infra Upgrade.....	12
Chapter 4. Debugging Information for Part 1.....	13
Chapter 5. Debugging Information for Part 2.....	14
Chapter 6. Frequently Asked Questions (FAQ).....	15
Chapter 7. More Information.....	16
Documentation Feedback.....	16
Requesting Technical Support.....	16
Self-Help Online Tools and Resources.....	16

Preface

Revision History

Revision	Description	Date
v1.0	Patch deployment guide for Appviewx 2020.3.0 FP6	September, 2021
v1.1	Updated the “Debugging Information for Part 1” section	September, 2021

About this Guide

The purpose of this document is to guide the users for applying patches on AppViewX v2020.3.0 FP6.

Audience

This document is intended for internal users and customers of Appviewx to support patch deployment activities.

Text Conventions

The following text conventions are used in this document:

Convention	Description
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>codeblock</code>	Indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Chapter 1: Expectations

Here are some expectations during the FP6 patch deployment.

- Kube infra upgrade
- The FP6 addons file size is 4.32 GB (Plan the download and file transfer accordingly, in accordance with the increased file size.
- Expect the patch deployment process to consume more time.

This patch deployment process involves two parts:

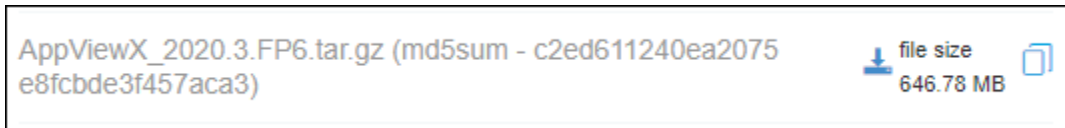
- Part 1: Plugins and Addons Upgrade
- Part 2: Kubernetes Infra Upgrade
- Expect the patch deployment process to consume more time.

Chapter 2: Part 1 - Plugins and Addons Upgrade

Follow the steps below to add the plugins and addons.

1. Log in to the [release portal](#) and download the FP6 patch files

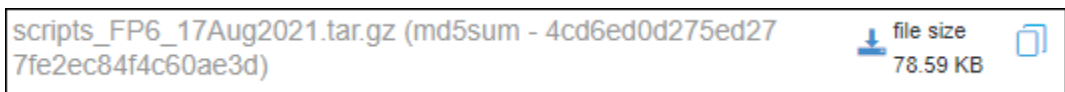
a. Download **AppViewX_2020.3.0_FP6.tar.gz** from the [release portal](#)



b. Download **appviewx_addons_2020.3.0_FP6.tar.gz** from the [release portal](#)



c. Download **scripts_FP6_17Aug2021.tar.gz** from the [release portal](#)



2. Move all the downloaded files to the node where the installation is initiated.

3. Open the terminal window with valid credentials.

4. To know the status of the pods, execute the command,

```
kubectl get pods -A
```

If a pod is in a state other than *“Running”* or the two containers associated with the pod (0/2 or 1/2) are not up and running, take note of it.

5. If the previous fixed patch from AppViewX had already been applied, ignore this step or else follow the two steps below before proceeding with the patch upgrade.

a. Run the following command on any one node

```
helm uninstall avx-platform-webavx-platform-gatewayavx-platform-amc
```

b. Run the following commands on all the nodes

```
cd <installation_path>/plugins
```

```
sudo rm -rf avx_platform_web avx_platform_gateway avx_platform_amc
```

6. Copy the downloaded files to the AppViewX server where the AppViewX installer is available.

7. Untar the scripts using the command

```
tar -xzf scripts.tar.gz
```

8. Check and replace the **hpa_conf.json.template** file that is present in the scripts directory.

```
hpa_conf.json.template  hpa_conf.json.template  mergeconf.  
insight_install.sh     mongo_back  
installation_logs.txt  mongo-back  
install_bcp.sh         mongodb_se
```



Note: If you have a **hpa_conf.json** file in the scripts directory ignore the below steps and move on to Step 9.

- a. Copy the content from **hpa_conf.json.template** file to **hpa_conf.json** by executing the command:


```
cp hpa_conf.json.template hpa_conf.json
```

- b. If there are customized values that need to be set for the keys in the **hpa_conf.json** file, configure them accordingly.

```

1 {
2   "deploymentfiles": {
3     "avx_platform_queue": {
4       "xms": "1g",
5       "xmx": "3g"
6     },
7     "avx_vendors": {
8       "xms": "1g",
9       "xmx": "2g"
10    },
11    "avx_subsystems": {
12      "xms": "1g",
13      "xmx": "3g"
14    }
15  },
16  "autoscalereplica": [
17    "avx_vendors",
18    "avx_subsystems"
19  ],
20  "hpafiles": {
21    "avx_subsystems_sync": {
22      "cputhreshold": "200",
23      "maxreplica": "3"
24    },
25    "avx_platform_core": {
26      "cputhreshold": "200",
27      "maxreplica": "3"
28    },
29    "avx_vendors": {}
30  },
31  },
32  "plugins_sync_memory_with_xmx": [
33    "avx_vendors",
34    "avx_subsystems_sync",
35  ]
36 }

```

NORMAL  hpa_conf.json
"hpa_conf.json" 39L, 805C

c. By default, if the customer does not create the **hpa_conf.json** file before patch, the file will be created using the **hpa_conf.json.template** file.

9. Take backup of the **plugins_install.sh** file in **<INSTALLER_PATH>/appviewx_kubernetes/scripts** using the command:

```
cd <INSTALLER_PATH>/appviewx_kubernetes/scripts mv plugins_install.sh plugins_install.sh.bak
```

10. Copy the files from scripts directory to the **<INSTALLER_PATH>/appviewx_kubernetes/scripts** directory using the command:

```
cp -r <Download_Directory>/scripts/* <INSTALLER_PATH>/appviewx_kubernetes/scripts/.
```

11. Execute the script file **apply_patch.sh**. On execution it will request the following:

- Absolute file path for plugins and add-on tar files

Atleast one input must be given to proceed with the patch process, or both inputs can be given at the same time also. Both examples are shown below:

```
./apply_patch.sh
```

Enter the absolute path of Addon tar package downloaded from the AppViewX Release Portal (Press enter to leave it blank): /home/appviewx/installer/appviewx_kubernetes/scripts/appviewx_addons_2020.3.0.7.tar.gz

Enter the absolute path of Plugins tar package downloaded from the AppViewX Release Portal (Press enter to leave it blank):

```
./apply_patch.sh
```

Enter the absolute path of Addon tar package downloaded from the AppViewX Release Portal (Press enter to leave it blank): /home/appviewx/installer/appviewx_kubernetes/scripts/appviewx_addons_2020.3.0.7.tar.gz

Enter the absolute path of Plugins tar package downloaded from the AppViewX Release Portal (Press enter to leave it blank): /home/appviewx/installer/appviewx_kubernetes /scripts/AppViewX_2020.3.0_Latest_Plugins.tar.gz

12. By default, the patch script works at interactive mode and the following questions will be asked during the process:
- Verify the list of enabled plugins and their respective data center. After the verification, provide the appropriate input to continue the deployment or exit the process.

```
ENABLED_PLUGINS
```

```
_____
```

```
appviewx_dependencies
```

```
_____
```

```
ENABLED PLUGINS AND NAMESPACE
```

```
....
```

```
Do you wish to continue (Yes/No)? Yes
```

- MongoDB and Vault backup can be taken before deploying the newer version for rollback, when prompted.

```
Do you wish to take DB backup(Yes/No)? Yes
```

- Old Existing DB Backups can be cleaned.

```
Do you wish to remove Existing DB backup (Yes/No) - Default (No): ? No
```

- Old Existing plugin backup can be cleaned up.

```
Do you wish to remove Existing plugin backup (Yes/No) - Default (No): ? No
```

- e. In case of any failure during the patch deployment, an automated rollback can be initiated and done within the script

```
Do you wish to roll back the patch process(Yes/No)? Yes
```

- 13. After successful deployment, the following message will be displayed along with the manual restore commands for rollback. **Kindly wait for a few minutes for the backend process to complete**

```
Apply complete! Resources: 104 added, 0 changed, 0 destroyed.
```

```
Patch Process Completed and Plugins are Upgraded.
```

```
Use the following commands to restore:
```

```
Restore Plugins:
```

```
1. rm -rf ../yaml/appviewx_plugins && mv
```

```
/opt/opt1/appviewx_kubernetes/scripts/backup_20210208-021556/appviewx_plugins ../yaml/ Restore Database:
```

Chapter 3: Part 2 - Kubernetes Infra Upgrade

1. Check whether all the pods are up and running. All the pods must be in 2/2 running state. Verify by executing the command:

```
kubectl get pods -A
```

2. Navigate to `<INSTALLER_PATH>/appviewx_kubernetes/scripts/infra_upgrade` directory.
3. Execute the command below

```
./upgrade.sh
```

During execution of the above command, users will be prompted to enter certain input values, follow the instructions below:

- a. Confirm if some pods are not in running state.
 - i. If it is a Prometheus pod, type “yes” to ignore and continue with the upgrade.
 - ii. If some other pods are not in running state, fix those issues before continuing the upgrade.
 - b. Enter valid Appviewx Sudo user password of all the nodes when prompted.
 - c. If ELK is enabled, enter the elastic user password, if requested.
4. After the patch upgrade is done, execute the command:

```
kubectl get pods -A
```

- a. If the upgrade is successful, all the pods in the custom data centers should be up and running.
 - b. If a pod is not in the running state but its predecessor (derived from the age of the pod) is up and running, then the latest plugin has failed to deploy. Reach out to AppViewX's support team.
5. To ensure that the Kubernetes upgraded to v1.22.6 successfully, execute the command,

```
kubectl get nodes
```

STATUS	ROLES	AGE	VERSION
Ready	control-plane,master	5h41m	v1.20.7
Ready	control-plane,master	5h39m	v1.20.7
Ready	<none>	5h39m	v1.20.7
Ready	<none>	5h39m	v1.20.7
Ready	control-plane,master	5h39m	v1.20.7
Ready	<none>	5h39m	v1.20.7
Ready	<none>	5h39m	v1.20.7

Chapter 4: Debugging Information for Part 1

Information on debugging details are mentioned below

1. The **patch_logs.txt** file can be located at `<INSTALLER_PATH>/appviewx_kubernetes/scripts`
2. If deployment fails with the message: *“scp failed: Upload failed”*, run the commands below:

```
chown -R appviewx:appviewx <installation_path>/plugins
```

```
chown -R appviewx:appviewx <installation_path>/logs
```



Note: The `<installation_path>` is mentioned in the `/appview_kubernetes/scripts/appviewx.conf` file as the parameter `INSTALLATION_PATH`.

3. The config server pod must be in “Running” state to deploy the plugin. If the config server pod is not in “Running” state, the script will terminate with the following error message.

```
Please ensure that config server pod is in running state before applying the patch.
```

4. To check the status of pods use the command below. If the plugin upgrade is successful, all the pods will be in the running state.

```
kubectl get pods -n <namespace>
```

5. If the helm install is triggered instead of helm upgrade, the following error message is displayed: *“cannot re-use a name that is still in use”*. This is due to a timeout issue while helm chart check is in progress. Fix the issue by re-triggering the following command:

```
scripts/plugins_install.sh
```

```
(and 20 more similar warnings elsewhere)
Error: Error running command 'helm install --set-string timestamp=2021-08-17T14:59:21Z \
--set-string appviewx.multi=true \
--set common.namespace="{avx}" \
--set appviewx.replicas="2" \
--set appviewx.nodeAffinity="{us,eu}" \
--set appviewx.installation_user=appviewx \
--set appviewx.installation_user_id=1000 \
--set appviewx.appviewx_path=/home/appviewx/appviewx_cluster/ \
avx-platform-web /home/appviewx/appviewx_binaries/appviewx_kubernetes/yaml/appviewx_plugins/avx_platform_web/chart;
: exit status 1. Output: Error: cannot re-use a name that is still in use
```

Chapter 5: Debugging Information for Part 2

Information on debugging details are mentioned below

1. The **patch_logs.txt** file can be located at `<INSTALLER_PATH>/appviewx_kubernetes/scripts/infra_upgrade/.infra_upgrade_$(DATE).log`



Note: This is a hidden file, to view run the command:

```
ls -la
```

2. What if the infra upgrade fails at some point and exits
 - a. Analyze the failure.
 - b. If it is an issue related to etcd timeouts, kube-api server connection refused errors, and health check failures, re-run the script `upgrade.sh`. Once the kube commands respond, it will run from the stage where it failed.
 - c. If the cause of the failure is not identified, do not re-run the upgrade script again as it may take the entire infra to an unrecoverable state.

Chapter 6: Frequently Asked Questions (FAQ)

How is FP6 different from our previous FP's?

In FP6, we plan to upgrade our infra which means we upgrade each and every K8s and third party components to a latest stable version. We also plan to change the container runtime from Docker to Container. Whereas in our previous patches we only patch avx related components.

Chapter 7: More Information

For the latest, most complete information about known and fixed issues with the AppViewX modules, see the latest revision of the release notes.

To access Software Release Notifications for AppViewX Releases, visit our Help center at <https://help.appviewx.com/home>. You need to log in to your AppViewX account. From the Help center, search by the specific release number or navigate to Release Portal and choose the release, for example, v20.3.0.

Documentation Feedback

We request you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to tech-documentation@appviewx.com

If you are preferred to send feedback through e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable).

Requesting Technical Support

Technical product support is available through AppViewX help support center, request to send an email to help@appviewx.com

Self-Help Online Tools and Resources

For quick and easy problem resolution, AppViewX is designed an online self-service portal called the help support center that provides you with the following features:

- Find help support center: <https://help.appviewx.com/home>
- Find product technical documentation: <https://helpcenter.appviewx.com/techdoc/>
- Find solutions and answer questions using our Knowledge Base: <https://internalkb.appviewx.com/knowledge-base>
- Download the latest versions of software: <https://release.appviewx.com>