Nastel XRay

Installation Guide

Version 10

Document Number: XRIG15.005

Document Title: Nastel XRay Installation Guide
Document Release Date: February 2021
Nastel Document Number: XRIG15.005

Published by:
Research & Development
Nastel Technologies, Inc.
88 Sunnyside Blvd, Suite 101
Plainview, NY 11803

Acknowledgements: The following terms are trademarks of Nastel Technologies Corporation in the United States or other countries or both: TransactionWorks, M6 AutoPilot, AutoPilot/IT, AutoPilot/Enterprise, M6 for WMQ, AutoPilot/WMQ, M6 Web Server, M6 Web Console, AutoPilot/WEB, MQControl, MQControl Express, AutoPilot/Transaction Analyzer, AutoPilot/WAS, AutoPilot/Transaction Monitor, AutoPilot/OS Monitor, Nastel Navigator.

The following terms are trademarks of the IBM Corporation in the United States or other countries or both: IBM, MQ, MQSeries, WebSphere, WebSphere MQ WIN-OS/2, AS/400, OS/2, DB2, AND AIX, z/OS.

The following terms are trademarks of Hewlett-Packard in the United States or other countries or both: OpenView, HP-UX.

Compaq, the Compaq logo, AlphaServer, Compaq Insight Manager, CDA, DEC, DECnet, TruCluster, ULTRIX, and VAX Registered in U.S. Patent and Trademark Office. Alpha and Tru64 are trademarks of Compaq Information Technologies Group, L.P in the United States and other countries.

SNMPc, SNMPc, WorkGroup, and SNMPc Enterprise are trademarks of Castle Rock Computing in the United States or other countries, or both.

Sun, Sun Microsystems, the Sun logo, iForce, Java, Netra, N1, Solaris, Sun Fire, Sun Ray, SunSpectrum, Sun StorEdge, SunTone, The Network is the Computer, all trademarks and logos that contain Sun, Solaris, or Java, and certain other trademarks and logos are trademarks or registered trademarks of Oracle Corporation and/or its affiliates.

InstallAnywhere is a registered trademark of ZeroG Software in the United States or other countries, or both.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/). The Jakarta Project" and "Tomcat" and the associated logos are registered trademarks of the Apache Software Foundation.

Intel, Pentium and Intel486 are trademarks or registered trademarks of Intel Corporation in the United States, or other countries, or both.

Microsoft, Windows, Windows NT, Windows XP, .NET, .NET Framework and the Windows logos are registered trademarks of the Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

"Linux" and the Linux Logos are registered trademarks of Linus Torvalds, the original author of the Linux kernel. All other titles, applications, products, and so forth are copyrighted and/or trademarked by their respective authors.

SCO CUSA, SCO Doctor, SCO Doctor for Networks, SCO Doctor Lite, SCO Global Access, SCO MPX, SCO MultiView, SCO Nihongo OpenServer, SCO OK, the SCO OK logo, SCO OpenServer, SCO Open Server, SCO Portfolio, SCO POS System, SCO ToolWare, and The World Never Stops are trademarks or registered trademarks of Caldera International, Inc. in the U.S.A. and other countries, All Rights Reserved.

Oracle® is a registered trademark of Oracle Corporation and/or its affiliates.

TIBCO is a registered trademark of TIBCO Software Inc. in the United States and/or other countries.

Other company, product, and service names, may be trademarks or service marks of others.

Contents

Chapter 1: Introduction

Nastel XRay is designed with a flexible, modular, clustered, component architecture and may be deployed in various configurations to one or more Virtual Machine Images, based on user requirements. Nastel provides the following two standard configuration packages for Nastel XRay:

- **Standalone Appliance Configuration:** A single node standalone appliance suitable for Proofs of Concepts, and development environments.
- **Multi-Node Cluster Configuration:** A multi-node small cluster, which provides the performance, fault tolerance and scalability needed for on-premise production systems.

These packages are provided by Nastel as Linux TAR files, which can be deployed to any properly configured Linux environment.

1.1 How this Guide is Organized

1.2 History of this Document

Table 1-1. Document History

Release Date	Document Number	Version	Summary
May 2019	XRIG15.001	15	Initial release
June 2019	XRIG15.002	15	Miscellaneous updates in sections 3.1.2, 3.2.2 and 4.2.
July 2019	XRIG15.003	15	Added descriptions to Table 3-1.
September 2019	XRIG15.004	15	Created Chapter 2 and moved sections 1.2 – 1.5 to this new chapter. Updated proceeding chapter and section numbers accordingly. Minor updates to sections 4.1 and 4.2. Updates to #1 in section 4.2.3. Updates in #1b and #3 in section 4.2.4.
February 2021	XRIG15.005	15	In section 4.1.4, add information on using the correct ZooKeeper folder. Update copyright year. Update "jKoolAdmin" to "Administrator." Update section 2.1 – added compatible web browsers. Updates in the following sections: 3.2 (#2), 4.1.2 (#5), command in 5.1.1, Table 5-2, section 4.1 (#4), Table 5-1. Rename config.sh to setup.sh.

1.3 User Feedback

Nastel encourages all users and administrators to submit comments, suggestions, corrections, and recommendations for improvement for all documentation. Please send your comments via e-mail to: *support@nastel.com*. You will receive a response, along with status of any proposed change, update, or correction.

1.4 Release Notes

See README files located in the following location:

```
$APIN_HOME/misc/docs
```

1.5 Intended Audience

This guide is intended for systems administrators and operating engineers responsible for the installation and administration of the Nastel XRay environment.

1.6 Technical Support

Use one of the following methods for technical support:

- **Call:** 800-963-9822 ext. 1
  If you are calling from outside the United States: 001-516-801-2100
- **Email:** support@nastel.com
- **Resource center:** *https://customers.nastel.com*
- **Automated support system:** http://support.nastel.com/ (user ID and password required)

Chapter 2: Overview

2.1 Nastel XRay Component Architecture

A Nastel XRay system consists of the following components, which are installed and preconfigured for the two configurations discussed in this document.

- ZooKeeper 3.XX
- Database cluster
  - For 1.0 and earlier, SolrCloud 6.5.X
  - For 1.1, SolrCloud 6.6.X
- ActiveMQ 5.11+
- Kafka 1.0.0
- Storm 1.1.x
- AutoPilot M6 SU27+
  - Domain Server
  - CEP Server
    - Nastel XRay Gateway expert
    - Nastel XRay Service expert
    - Nastel XRay DB Writer expert
    - Nastel XRay Cold Storage expert
    - Nastel XRay Trigger expert
    - Nastel XRay Metrics expert
  - Web Server (Tomcat 7.0.56+)
    - Nastel XRay/APInsight web UI
    - Nastel XRay web REST Admin module
- A compatible web browser: Google Chrome v27, Internet Explorer v11, Microsoft Edge v13, Mozilla Firefox v20, Safari v7, Opera v23

2.2 Standalone Appliance Configuration

The Nastel XRay Standalone Appliance consists of a single on-premise standalone node to be run on a single server. It is configured with all Nastel XRay components listed above (*Section 2.1*), which are pre-configured to work together. A default license is provided with a default Nastel XRay organization, repository, teams and configured users. The Nastel XRay Appliance is ready to run once deployed and configured.

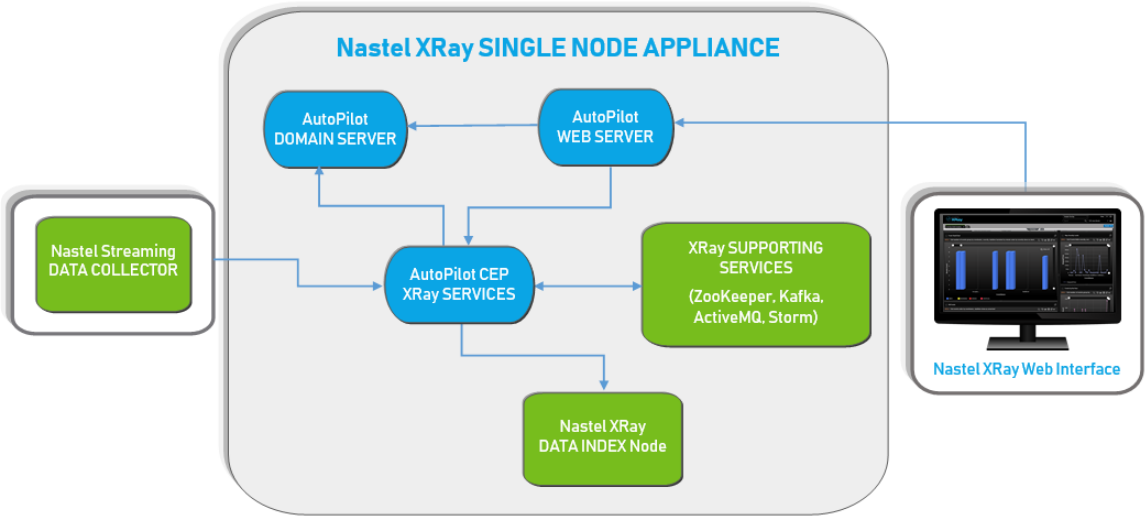

Figure 2.2-A. Nastel XRay Single Node Appliance

2.3 Multi-Node Cluster Configuration

The Multi-Node Cluster system supports a 4 node Solr Database Cluster, to be deployed on separate virtual machines. All other Nastel XRay components and services are intended to remain on the Data Compute Node. Like the Standalone Appliance, all components are provided in Linux TAR file format, configured with all Nastel XRay components listed above (*Section 2.1*), which are pre-configured to work together after basic setup.

The Multi-Node Cluster system does not provide a preconfigured default configuration, a license file will need to be obtained from Nastel Support. Steps to configure the Solr Database cluster, the Nastel XRay organization, repositories, team and users, are provided later in this document.

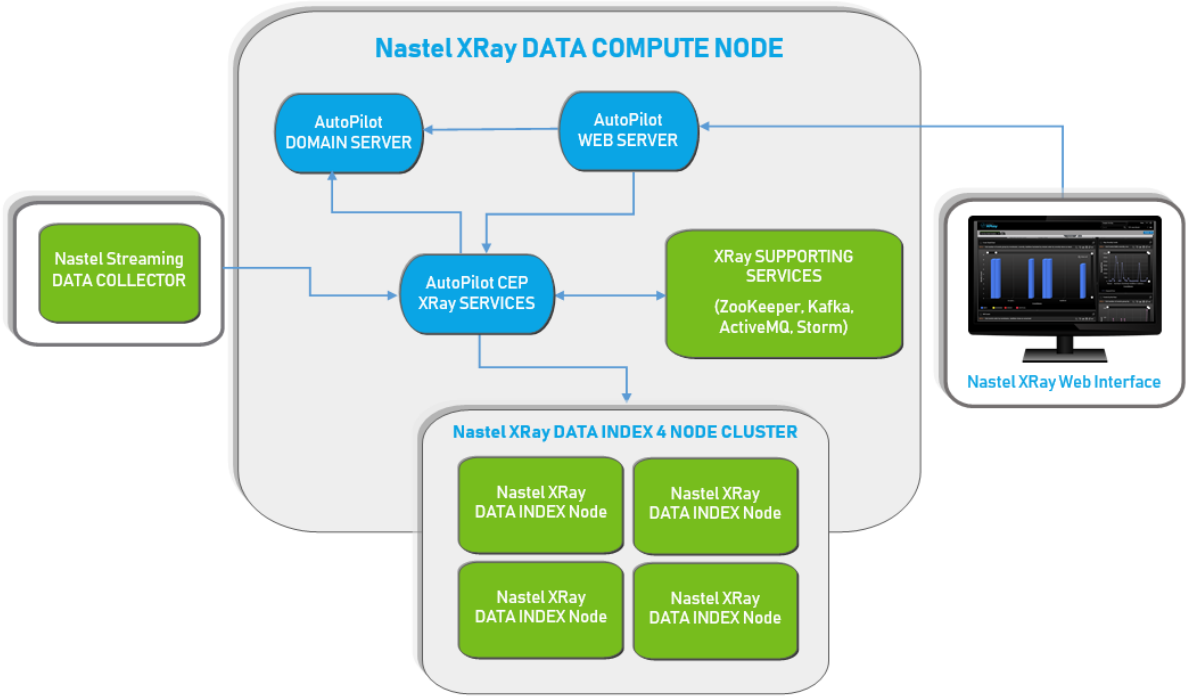

Figure 2.3-A. Nastel XRay Data Compute Node

2.4 Other Configurations

Nastel XRay is fully scalable. Due to its flexible, modular, clustered, component architecture, the system may be deployed in many configurations across multiple Virtual Machines Images, based on specific user requirements. Contact your Nastel Sales Representative or Nastel Professional Services for assistance with designing a Nastel XRay cluster to support your unique requirements.

Chapter 3: Prerequisites

The following prerequisites must be in place before installation.

3.1 System Requirements

3.1.1 Standalone Appliance Requirements

- **Operating System:** Linux 64 bit OS, CentOS 7 or equivalent of Red Hat or Fedora
- **CPU:** Minimum 2 CPUs (4 virtual CPUs)
- **Memory:** Minimum 32 GB RAM (48 GB RAM recommended)
- **File System:** Minimum 500 GB free space, Solid-State Disc (SSD) preferred
- **Recommended Install folder:** `/opt/nastel`

3.1.2 Multi-Node Cluster Requirements

For a production system deployment, please contact your Nastel Sales representative, Nastel Support, or Nastel Professional Services to properly size the Nastel XRay nodes for your environment. The specifications provided below are minimum requirements for a cluster intended to support approximately 5 GB per day of streaming data, with a retention period of 14 days.

Nastel XRay Data Compute Node (qty 1) :

- **Operating System:** Linux 64 bit OS, CentOS 7 or equivalent of RedHat or Fedora
- **CPU:** Minimum 8 virtual CPUs
- **Memory:** Minimum 32 GB RAM
- **File System:** Minimum 100 GB free space, SSD preferred
- **Recommended Install folder:** `/opt/nastel`

Nastel XRay Data Index Nodes (qty 4) :

- **Operating System:** Linux 64 bit OS, CentOS 7 or equivalent of RedHat or Fedora
- **CPU:** Minimum 4 virtual CPUs
- **Memory:** Minimum 8 GB RAM
- **File System:** Minimum 500 GB free space, SSD preferred
- **Recommended Install folder:** `/opt/nastel`

3.2 Disable Swap

Swap must be turned off within the operating system on all virtual machine servers running Nastel XRay Compute Node or Nastel XRay Data Index Node. To disable swap, perform the following:

1) Identify configured swap devices and files with the following command:

```
cat /proc/swaps
```

2) Turn off all swap devices and files with the following command:

```
swapoff -a
```

3) Remove any matching references found in `/etc/fstab`.

3.3 Ulimits Requirements

The required minimum ulimits (ulimit –a) for virtual machine servers running Nastel XRay Compute Node or Nastel XRay Data Index Node are listed below. These must be set before starting Nastel XRay for the first time. If they are not set, root level access is required to set them.

```
core file size (blocks, -c) 0
data seg size (kbytes, -d) unlimited
scheduling priority (-e) 0
file size (blocks, -f) unlimited
pending signals (-i) 1031597
max locked memory (kbytes, -l) 64
max memory size (kbytes, -m) unlimited
open files (-n) 32768
pipe size (512 bytes, -p) 8
POSIX message queues (bytes, -q) 819200
real-time priority (-r) 0
stack size (kbytes, -s) 8192
cpu time (seconds, -t) unlimited
max user processes (-u) 1031597
virtual memory (kbytes, -v) unlimited
file locks (-x) unlimited
```

3.3.1 Ulimit Configuration Files

The system reads ulimits from the `/etc/security/limits.conf` configuration file by default. Individual `*.conf` files are then read from the `/etc/security/limits.d/` directory. The files are parsed one after another in the order of "C" locale. The effect of the individual files is the same as if all files were concatenated together in the order of parsing. If a configuration file is explicitly specified with a module option, then the files in the `/etc/security/limits.d/` directory will not be parsed.

Within the two locations where ulimits can be configured, update the following configuration files based on your operating system.

CentOS

```
/etc/security/limits.conf
/etc/security/limits.d/90-nproc.conf
/etc/security/limits.d/20-nproc.conf
```

RHEL 5, RHEL 6, RHEL 7

```
/etc/security/limits.conf
/etc/security/limits.d/90-nproc.conf
```

3.3.2 Soft / Hard Limits

Two types of values are used to define limits: Soft and Hard. A Soft limit can be adjusted up to the Hard limit. A Hard limit can only be lessened and is the maximum resource limit a user may have.

Using the command, `ulimit -n`, the user will be presented with the Soft limit. If the `/etc/security/limits.conf` file has a hard value set, it will not be presented by default.

To view Soft limits, use the following command:

```
# ulimit -n -S
```


To view Hard limits, use the following command:

```
# ulimit -n -H
```

3.3.3 Setting User Process Limits

The allowed number of processes must be limited for each Nastel user. By default, the rules are read from the `/etc/security/limits.conf` file. To limit the number of processes for each Nastel user, modify this file using the following commands. Replace <*username*> with the username that will run XRay Processes.

Limit the maximum number of open files. We recommend the following minimum limits for both nofile and nproc settings.

```
# cat /etc/security/limits.conf | grep nofile | grep -v ^#
<username> soft nofile 32768
<username> hard nofile 32768
```

Limit the maximum number of processes. We recommend the following minimum limits for both nofile and nproc settings.

```
# cat /etc/security/limits.conf | grep nproc | grep -v ^#
<username> soft nproc 1031597
<username> hard nproc 1031597
```

3.4 Linux Firewall

By default, CentOS and other Linux distributions have an active firewall that block ports which are needed to connect to Nastel XRay services remotely. The following ports should be opened, as these are the ports Nastel XRay utilizes:

- ZooKeeper: 2181
- Solr: 8983
- ActiveMQ: 8161
- Storm: 8088
- AutoPilot Domain Server: 2323, 3000
- AutoPilot XRay CEP: 3005
- AutoPilot XRay UI CEP : 3010
- Nastel XRay: 8080
- Nastel XRay Gateway: 6580

The following commands can be used to check the firewall status, temporarily stop/disable the firewall, or re-enable/re-start the default firewall on CentOS 7.

```
>systemctl stop firewalld
>systemctl disable firewalld
>systemctl enable firewalld
```


3.5 Linux ID

Create the Linux user ID (UID) and group which will be used to start all Nastel XRay services.

The default user and group are **nastel:nastel**, this can be changed to any value. The UID and group must own all files and directories under $APIN_HOME.

3.6 Installation Path

Determine the installation path prior to deployment. The default assumed is `/opt/nastel`. However, any location can be used

Chapter 4: Installation

4.1 Nastel XRay Standalone Appliance

Below are the installation and startup steps for a Nastel XRay Standalone Appliance configuration.

4.1.1 Step 1: Confirm Minimum Server Requirements

Confirm that the server meets the minimum requirements specified within *Sections* 3.1, 3.2 and 3.3 prior to performing the next steps.

4.1.2 Step 2: Extract the Install File

Upload and extract the install file by performing the following:

1. Log in as the Nastel user (or other user that will run all Nastel XRay services).
2. Create a Temp directory within the user's Home directory: `/home/nastel/temp`
3. Upload **Xray_x.x.xx_UIv.xx_GMvx.x.tar.Z** (**x.x.x** represents the version number) and place it in the temp directory created. Be sure that the file is owned by the user and group **nastel:nastel**. Do this with the following command:

```
>chown nastel:nastel Xray_x.x.xx_UIv.xx_GMvx.x.tar.Z
```

4. Create the Nastel directory within the `/opt` directory. Be sure that the Nastel user has write permissions for this directory.
5. Extract **Xray_x.x.xx_UIv.xx_GMvx.x.tar.Z** into the `/opt` directory by using the following commands:

```
>cd /opt
>tar –xzvf /home/nastel/temp/Xray_x.x.xx_UIv.xx_GMvx.x.tar.Z
```

6. Change to the folder: InstallationPath/sbin/.

   IMPORTANT: As of version 1.3, XRay requires an SQL database server (either PostgreSQL or MySQL) to run the job scheduler service.

   This XRay Deployment Pack includes the PostgreSQL database server. In the file "installed_services.conf", the service PostgreSQL is enabled by default:

   pgsql=yes

   This is required for the Xray services to start automatically.

4.1.3 Step 3: Set Environment Variables

Open the **apin_env.sh** file and modify the following environment variables based on your configuration:

- APIN_HOME: Set this to the XRay installation path, for example: APIN_HOME=/opt/Nastel
- ZK_HOST: Set this to the ZooKeeper server location. Include the port number, for example: ZK_HOST=localhost:2181

4.1.4 Step 4: First-time Solr Initialization Setup

For database cluster versions 1.0 and earlier, SolrCloud 6.5.X is required. For database cluster version 1.1, SolrCloud 6.6.X is required. Solr 6.X requires Java V8. Other XRay components can run with higher Java versions.

Ensure ZooKeeper and Solr are up and running before running the Solr configuration script. The XRay Solr configuration script is **setup.sh** and is located in `$APIN_HOME/sbin`. Run **setup.sh**. as follows and follow the onscreen instructions:

```
run ./setup.sh all
```

Confirm Correct ZooKeeper Folder is Being Used: Nastel XRay Solr scripts and the Solr ZooKeeper configuration must use the same ZooKeeper folder (zkchroot, to use ZooKeeper terminology). If not, when the Solr collection configurations are loaded, Solr will not pick up the changes.

To verify the correct folder is being used, compare the values being used by the Nastel XRay Solr script with the value in the Solr configuration. The script will output a line similar to the following:

```
Using Zookeeper at: localhost:2181
```

The value being used by Solr is defined in **<solr-home>/bin/solr.in.sh**, as follows:

```
ZK_HOST="localhost:2181"
```

These two values need to match. If there is a ZooKeeper zkchroot specification (starting with '/' after port number), e.g. ZK_HOST="localhost:2181/solr", then you will have to change one of them so that they either both do not have any zkchroot specification, or they both have the same one. To change it in Solr configuration, simply add or remove the "/solr" from ZK_HOST. By default, Nastel XRay Solr scripts do not use this, add the following to a command line to have them use it:

```
-zr /solr
```

4.1.5 Step 5: Start Nastel XRay

Run all Nastel startup scripts located in `$APIN_HOME/sbin` to start Nastel XRay. For usage and to start Nastel XRay please run: `./start.sh all`

4.1.6 Optional: User Environment Setup

Optional environment variable configuration: Please note that to run the standalone appliance as delivered in the package, it is not necessary to set the environment variables at the user or system level, however in some cases, such as deployment of a single server development environment, it may be desirable to set them.

For the Nastel user or other user that will run the Nastel services, add the following lines to their **.bash_profile**:

```
export APIN_HOME=<path_to_XRay_filesystem>
export APIN_LOGS=$APIN_HOME/AutoPilotM6/logs
```



```
export APM6_HOME=$APIN_HOME/AutoPilotM6
export JAVA_HOME=$APIN_HOME/java/current
export SOLR_HOME=$APIN_HOME/solr/current
export KFKA_HOME=$APIN_HOME/kafka/current
export
PATH=$PATH:$APIN_HOME/sbin/:$JAVA_HOME/bin:$SOLR_HOME/bin:$KFKA_HOME/bin
```

NOTE

Only the environment variable $APIN_HOME requires the full path to the installation directory. All other environment variables will be built from this variable.

Optionally, the user's **.bash_profile** can be set up to run the **apin_env.sh** script edited for the target environment as per *Section 4.1.2*.

NOTE

If needed, the Standalone Appliance database can be built from scratch. See *Section 4.2.5, Initialize the Database*, for more information

4.1.7 License File

The Nastel XRay Standalone Appliance comes with a default license file, however, it may be necessary to update this with a license file more suitable to a given customer use case. See the below table for the quota specifications of the default license.

Table 4-1. Standalone License Quotas

Quota	Limit	Description
DataPoints	500,000	Defines the total number of data points (total number of Activities, Events, and Snapshots) that can be stored in the data store at any one time (based on Retention).
MaxMsgSize	1KB	Defines the maximum number of bytes that is stored in the Message field of Events (generally represents the payload of the data involved in the Event).
MaxOrganizations:	2	The maximum number of Organizations that can be defined in the entire system (has no effect for Default or organization-specific license).
MaxPropValueRollup	100	During the stitching process of grouping related Events/Activities into a single Activity, we merge the custom properties (Properties field) of all the child Events and SubActivities up to the root-level Activity. This limit controls the number of such properties that are stored in the root-level Activity. If the total property count would exceed this limit, the additional properties are not rolled up. Which properties are rolled up and which are not is indeterminate.

MaxRepositories	2	The maximum number of Repositories that can be defined in the entire system (for Master License) or in a specific organization (for Default or organization-specific license).
MaxTeams	2	The maximum number of Teams that can be defined in the entire system (for Master License) or in a specific organization (for Default or organization-specific license)
MaxTokens	2	The maximum number of Access Tokens that can be defined in the entire system (for Master License) or in a specific organization (for Default or organization-specific license).
MaxUsers	5	The maximum number of Users that can be defined in the entire system (for Master License) or in a specific organization (for Default or organization-specific license).
RateLimitBytes	50KB	Defines the maximum streaming rate, in bytes per second, which data can be sent to the system. If data comes in at a higher rate, the defined OveragePolicy will be applied to the connection.
RateLimitCount	50	Defines the maximum streaming rate, in messages per second, which data can be sent to the system. If data comes in at a higher rate, the defined OveragePolicy will be applied to the connection.
StreamBytesPerDay	5GB	Total number of bytes that can be streamed in per calendar day. This is computed based on the total length of the streamed JSON message.
StreamMsgsPerDay	1,000,000	Total number of individual messages that can be streamed in per calendar day.
OveragePolicy	Throttle	Defines what action is taken when the streaming rate exceeds either RateLimitBytes or RateLimitCount: • THROTTLE – the connection is throttled so that the processing rate on the connection is the minimum of RateLimitBytes and RateLimitCount • DROP – messages are dropped until the streaming rate slows down to the limits defined by RateLimitBytes and RateLimitCount • ALLOW – no action is taken and the streaming is allowed to continue at the current rate
Retention	5 Days	Defines the length of time, in seconds, that data is kept. When the Retention time expires, the data is deleted from the database.

4.2 Nastel XRay Multi-Node Cluster

Below are the installation and startup steps for a Nastel XRay Multi-Node Small Cluster, with one Nastel XRay Data Compute Node and four Nastel XRay Data Index Nodes.

4.2.1 Step 1: Install the Multi-Node Cluster

For the Data Compute Node:

1. Log in as the Nastel user (or other user that will run all Nastel XRay services).
2. Create a Temp directory within the user's Home directory: `/home/nastel/temp`
3. Upload **XRay_DCN_x.x.x.tar.gz** and place it in the Temp directory created. Be sure that the file is owned by the user and group **nastel:nastel**. Do this by using the following command:

```
>chown nastel:nastel XRay_DCN_x.x.x.tar.gz
```

4. Create the Nastel directory within the `/opt` directory. Be sure that the Nastel user has write permissions for this directory.
5. Extract **XRay_DCN_x.x.x.tar.gz** into the `/opt` directory. Use the following commands:

```
>cd /opt
>tar -xzvf /home/nastel/temp/XRay_DCN_x.x.x.tar.gz
```

For the Data Index Nodes:

1. Log in as the Nastel user (or other user that will run all Nastel XRay services).
2. Create a Temp directory within the user's Home directory: `/home/nastel/temp`
3. Upload **XRay_DIN_x.x.x.tar.gz** and place it in the temp directory created. Be sure that the file is owned by the user and group **nastel:nastel**. Use the following command:

```
>chown nastel:nastel XRay_DIN_x.x.x.tar.gz
```

4. Create the Nastel directory within the `/opt` directory. Be sure that the Nastel user has write permissions for this directory.
5. Extract **XRay_DIN_x.x.x.tar.gz** into the `/opt` directory. Use the following commands:

```
>cd /opt
>tar -xzvf /home/nastel/temp/XRay_DIN_x.x.x.tar.gz
```

6. Repeat these steps for each of the four **DIN** nodes.

4.2.2 Step 2: Set Environment Variables

For the Multi-Node Small Cluster the environment variables should be set on each of the nodes in the cluster.

For the user that will run the Nastel XRay process, the following environment variables must be created. All scripts and components are configured to use these.

For the **DCN** node set the following environment variables:

a. APIN_HOME
b. APIN_LOGS
c. APM6_HOME
d. JAVA_HOME
e. KFKA_HOME

For each of the four **DIN** nodes, set the following environment variables:

a. APIN_HOME
b. JAVA_HOME

c. SOLR_HOME

For the Nastel user or other user that will run the Nastel services, add the following lines to the user's **.bash_profile**:

```
export APIN_HOME=<path_to_XRay_filesystem>
export APIN_LOGS=$APIN_HOME/AutoPilotM6/logs
export APM6_HOME=$APIN_HOME/AutoPilotM6
export JAVA_HOME=$APIN_HOME/java/current
export SOLR_HOME=$APIN_HOME/solr/current
export KFKA_HOME=$APIN_HOME/kafka/current
export PATH=$PATH:$APIN_HOME/sbin/:$JAVA_HOME/bin:$KFKA_HOME/bin
```

NOTE

Please note that only the environment variable $APIN_HOME requires the full path to the installation directory. All other environment variables will be built from this variable.

Optionally, the user's **.bash_profile** can be configured to run the **apin_env.sh** script edited for the target environment as per *Section 4.1.2*.

4.2.3 Step 3: Start the Solr Cluster

Before continuing, the below files must be edited for proper connectivity between the DIN Nodes and the DCN.

NOTE

In order to complete the additional configuration described in this section, you will need to know the Admin User Name, Organization Name, Team Name, Repository Name, and Token Name for your environment. Contact Nastel for additional information.

1. If you have not set the environment variable ZK_HOST within the **apin_env.sh** file, set it now (please see *Section 4.1.2*). Edit the **solr.in.sh** file located in `$APIN_HOME/solr/current/bin`. Update the following entry with the IP address of your ZooKeeper host (the DCN Node), and save the file.

```
ZK_HOST="<zookeeperIP>:2181"
```

NOTE

<zookeeperIP> must be set to the IP address of the DCN, since that is where ZooKeeper will be running. This step must be completed on all 4 DIN nodes before they can be started.

2. Start ZooKeeper on the DCN node. Run the **xray_zoo_start.sh.** script located in `$APIN_HOME/sbin/`. This will start ZooKeeper on the DCN so that the Solr Cluster can be configured (Solr configuration must be complete before you can continue). Execute the following commands:

```
>cd /$APIN_HOME/sbin/
>./xray_zoo_start.sh
```


3. Start Solr on each of the four DIN nodes:
    a. Run the **xray_solr_start.sh** script located in `$APIN_HOME/sbin/` to start Solr on each of the four Solr nodes. Do this by executing the following commands on each of the four nodes:

```
>cd $APIN_HOME/sbin/
>./xray_solr_start.sh
```

4.2.4 Step 4: Confirm Solr Cluster is Running

1. Validate ZooKeeper is started on DIN:
    a. Confirm ZooKeeper is running by issuing the following command.

```
>ps -ef | grep zookeeper
```

This command returns information about the running ZooKeeper process:

```
[nastel@localhost bin]$ ps -ef | grep zookeeper
nastel   12771 49795  0 17:31 pts/0    00:00:00 grep --
color=auto zookeeper
nastel   51497     1  0 Feb13 pts/0    00:01:56
/datafs/apps/xray/nastel/java/current/bin/java -
Dzookeeper.log.dir=. -Dzookeeper.root.logger=INFO,CONSOLE -cp
/datafs/apps/xray/nastel/zookeeper/current/bin/../build/classes:
/datafs/apps/xray/nastel/zookeeper/current/bin/../build/lib/*.ja
r:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib/slf4j-
log4j12-
1.7.25.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib
/slf4j-api-
1.7.25.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib
/netty-
3.10.6.Final.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/
../lib/log4j-
1.2.17.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib
/jline-
0.9.94.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib
/audience-annotations-
0.5.0.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../zook
eeper-
3.4.13.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../src
/java/lib/*.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/.
./conf: -Dcom.sun.management.jmxremote -
Dcom.sun.management.jmxremote.local.only=false
org.apache.zookeeper.server.quorum.QuorumPeerMain
/datafs/apps/xray/nastel/zookeeper/current/bin/../conf/zoo.cfg
```

    b. Confirm that the ZooKeeper PID (Process ID) was created in the proper location by running the following command. You should see a file named **zookeeper_server.pid** with a date and timestamp corresponding to when the start command was issued.

```
ls -l  $APIN_HOME/zookeeper/current/dataDir
```



```
>cat $APIN_HOME/zookeeper/current/dataDir/zookeeper_server.pid
7331[nastel@ip-172-31-76-185 bin]$
```

Where 7331 is the Zookeeper PID.

2. Validate that Solr is running on all four DIN nodes:

   a. Confirm Solr is running by issuing the following command. This command returns information about the running Solr process.

```
>ps -ef |grep solr
```

This command returns information about the running Solr process:

```
[nastel@localhost bin]$ ps -ef |grep solr
nastel   12717 49795  0 17:29 pts/0    00:00:00 grep --
color=auto solr
nastel   53383     1  0 Feb13 pts/0    00:06:41
/datafs/apps/xray/nastel/java/current/bin/java -server -Xms4000m
-Xmx4000m -XX:NewRatio=3 -XX:SurvivorRatio=4 -
XX:TargetSurvivorRatio=90 -XX:MaxTenuringThreshold=8 -
XX:+UseConcMarkSweepGC -XX:+UseParNewGC -XX:ConcGCThreads=4 -
XX:ParallelGCThreads=4 -XX:+CMSScavengeBeforeRemark -
XX:PretenureSizeThreshold=64m -XX:+UseCMSInitiatingOccupancyOnly
-XX:CMSInitiatingOccupancyFraction=50 -
XX:CMSMaxAbortablePrecleanTime=6000 -
XX:+CMSParallelRemarkEnabled -XX:+ParallelRefProcEnabled -XX:-
OmitStackTraceInFastThrow -verbose:gc -XX:+PrintHeapAtGC -
XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+PrintGCTimeStamps
-XX:+PrintTenuringDistribution -
XX:+PrintGCApplicationStoppedTime -
Xloggc:/datafs/apps/xray/nastel/solr/logs/solr_gc.log -
XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=9 -
XX:GCLogFileSize=20M -DzkClientTimeout=15000 -
DzkHost=localhost:2181/solr -
Dsolr.log.dir=/datafs/apps/xray/nastel/solr/logs -
Djetty.port=8983 -DSTOP.PORT=7983 -DSTOP.KEY=solrrocks -
Duser.timezone=UTC -
Djetty.home=/datafs/apps/xray/nastel/solr/current/server -
Dsolr.solr.home=/datafs/apps/xray/nastel/solr/solrData/data -
Dsolr.install.dir=/datafs/apps/xray/nastel/solr/current -
Dlog4j.configuration=file:/datafs/apps/xray/nastel/solr/solrData
/log4j.properties -Xss256k -Dsolr.log.muteconsole -
XX:OnOutOfMemoryError=/datafs/apps/xray/nastel/solr/current/bin/
oom_solr.sh 8983 /datafs/apps/xray/nastel/solr/logs -jar
start.jar --module=http
```

   b. Check the Solr logs by running the following commands. The log files should show normal startup of Solr.

```
>cd $APIN_HOME/solr/logs
>cat solr.log
```


3. Confirm Solr is properly connected to ZooKeeper, and that ZooKeeper shows all Solr nodes active. Do this by running the following command from DIN Node 1 to validate ZooKeeper and Solr.

```
>cd $APIN_HOME/solr/current/bin
>./solr zk ls/solr/live_nodes -z <zookeeperIP>:2181
```

This command returns a list of 4 active Solr Nodes, for example:

```
Connecting to zookeeper at 172.18.140.25:2181 ...
Getting listing for zookeeper node /solr/live_nodes from
zookeeper at 172.18.140.25:2181 recurse: false
172.18.140.22:8983_solr
172.18.140.24:8983_solr
172.18.140.21:8983_solr
172.18.140.23:8983_solr
```

4. Validate Solr from the UI is accessible on all nodes from a web browser. Launch a web browser and navigate to the Solr URL for each of the four Solr nodes.

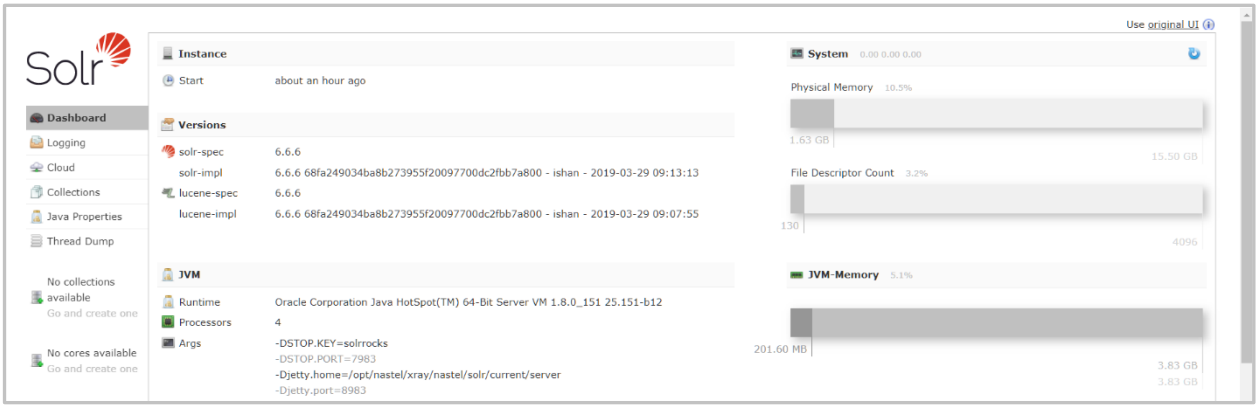

Figure 4.2.4-A. Validate Solr

4.2.5 Step 5: Initialize the Database

The process of initializing the database is performed once as part of the Multi-Node Cluster installation process. These steps can be used to rebuild the database from scratch, if needed, on both the Standalone Appliance or the Multi-Node Cluster.

4.2.5.1 Prepare the Scripts

Complete the steps in this section from DIN Node 1.

1. Update the below lines of the **create-cores.sh** script located in:
`/opt/Nastel/misc/jkool-dbapi/current/schemas`
Please note, ***current*** is a soft link to the latest version of the dbapi directory. These changes only need to be performed on one instance of the DIN. It will need to be the host where you will run the **jkool-dbapi** scripts for Solr database configuration and maintenance. This DIN should be designated as DIN 1 of 4.

```
SOLR_HOME=$APIN_HOME/solr/current
```



```
ZKHOST=<zookeeperIP>
```

<*zookeeperIP*> must be set to the IP address of the DCN, since that is where ZooKeeper will be running.

```
ZKPORT=2181
```

The default port for ZKPORT is 2181, but needs to be updated to ensure that the Solr files are located in the dedicated path within the ZooKeeper hierarchy.

The **create-cores.sh** script is preconfigured to create a configuration optimized for the recommended 4 node DIN Solr cluster. If you are using this script to create DB tables for a different configuration, such as a single DIN node, or possibly 6 or 8 DIN nodes, contact Nastel support for assistance in updating the **create-cores.sh** script to suit your desired configuration.

```
[nastel@ip-172-31-76-185 current]$ pwd
/opt/nastel/xray/nastel/misc/tools/jkool-dbapi/current
[nastel@ip-172-31-76-185 current]$ cd ../../
[nastel@ip-172-31-76-185 tools]$ ls -l
total 8
drwxrwxr-x. 3 nastel nastel 4096 May  2  2017 jesl
drwxrwxr-x. 4 nastel nastel 4096 Jul  3 18:48 jkool-dbapi
[nastel@ip-172-31-76-185 tools]$ cd jkool-dbapi/
[nastel@ip-172-31-76-185 jkool-dbapi]$ ls -l
total 8
lrwxrwxrwx. 1 nastel nastel   23 Jun 28 21:11 current -> jkool-dbapi-solr-1.1.24
drwxrwxr-x. 6 nastel nastel 4096 Jun 28 23:48 jkool-dbapi-0.15.21
drwxr-xr-x. 8 nastel nastel 4096 Jul  8 22:17 jkool-dbapi-solr-1.1.24
[nastel@ip-172-31-76-185 jkool-dbapi]$ cd current/
[nastel@ip-172-31-76-185 current]$ ls -l
total 24
drwxr-xr-x.  3 nastel nastel 4096 Jul  1 16:59 bin
drwxr-xr-x.  2 nastel nastel 4096 Jun 28 21:10 config
drwxrwxr-x.  2 nastel nastel 4096 Jun 28 21:10 lib
drwxrwxr-x.  2 nastel nastel 4096 Jul  1 17:19 log4j
drwxr-xr-x. 32 nastel nastel 4096 Jul  8 19:48 schemas
drwxr-xr-x.  3 nastel nastel 4096 Jul  8 20:39 scripts
```

Figure 4.2.5.1-A. Script Location

2. The **jkool-cmd.sh** script is located in:
   `$APIN_HOME/misc/tools/jkool-dbapi/current/bin`

4.2.5.2 Execute the Scripts to Create Solr Cores

1. Create the database tables in SolrCloud (requires curl) by running the **create-cores.sh** script. Run the script by using the following command:

```
>cd $APIN_HOME/misc/tools/jkool-dbapi/current/scripts/solr
>./create-cores.sh
```

   Successful completion of this command will show all Solr cores created and Solr configurations uploaded.

2. Perform the following to validate that the database tables have been created. Wait approximately one minute before performing the below to allow the collections to be

generated.

a. Enter the following command from the command line on DIN Node 1:

```
>cd $APIN_HOME/solr/current/bin
>./solr zk ls /solr/collections -z <zookeeperIP>:2181
```

This command returns a list of the collections that were created by running the **create-cores.sh** script. The figure below displays the list of collections that should be returned. Repeat this step if you do not get the same collections returned.

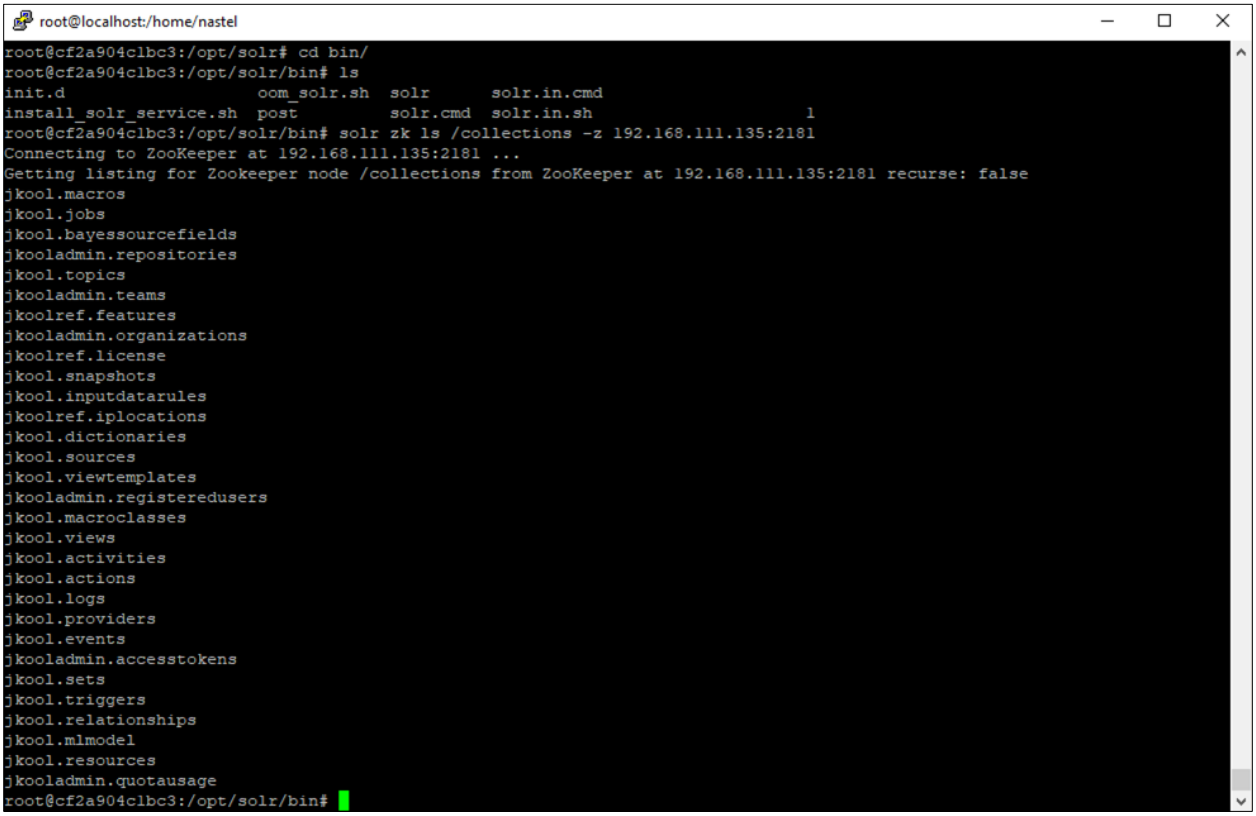

Figure 4.2.5.2-A. Returned Collections

b. Launch a web browser and navigate to the Solr URL for each of the four Solr nodes. See *Section 6.2* below in this document for more information on validating Solr. You should see a list of the cores and collections across all 4 nodes, with 4 shards and 2 replicas for each collection.

3. Run **admin.jkql**:

```
bin/jkool-cmd.sh -run -f:scripts/admin.jkql
-C:http://172.31.76.185:8983 -U:Administrator -P:admin
```

4.2.5.3 Prepare Initial Database

1. The Nastel XRay Standalone Appliance comes with a default license file, however it may be necessary to update this with a license file more suitable to a given customer use case. For the Nastel XRay Multi-Node Small Cluster, a license file must be obtained from Nastel prior to

completing the installation and configuration. At a minimum, Solr must be running when loading the license file.

Place the Nastel XRay license file obtained from Nastel into the following directory: `$APIN_HOME/misc/tools/jkool-dbapi/current/scripts`

2. Make copies of the following script files located in:
   `$APIN_HOME/misc/tools/jkool-dbapi/current/scripts`
   a. Copy **admin.jkql** to **admin.jkql.bak**
   b. Copy **xray-admin.jkql** to **xray-admin.jkql.bak**
3. Edit **admin.jkql** as follows:

```
Create User 'Admin' Password='admin', Active=true;
```

NOTE The password for the admin user can be changed later for security purposes.

4. Edit **xray-admin.jkql** as follows. Please note the addition of the last line. This must be added prior to running the script to create the first organization, which is the default. In most cases only one organization is needed, but if additional organizations are added, this setting will need to be commented out or removed.

```
Signin 'Admin' using 'admin';
Create Organization '<yourorgname>' Owner='U:Admin';
Create Team '<yourteamname>' Orgname='<yourorgname>', Owner='U:Admin';
Create Repository '<yourreponame>' Orgname='<yourorgname>',
Owner='U:Admin';
Create Token '<reponame>@<orgname>' Orgname='<yourorgname>',
Reponame='<yourreponame>';
Alter Organization 'OrgName' Properties+=('IsDefaultOrganization'=true);
```

NOTE <*yourorgname*> is the name of your organization, noted in your license.
<*orgname*> can be anything, but should be relevant to your organization name and installation.

5. Save the files after editing.

4.2.5.4 Load the License File

1. To load the license, use the command '**jkool-cmd.sh -loadlic'** provided with the XRay_dbapi package located on Solr Node 01 in the '$APIN_HOME/misc/tools/jkool-dbapi/current' directory.
2. Load the license by running the following commands:

```
>cd $APIN_HOME/misc/tools/jkool-dbapi/current
run
>bin/jkool-cmd.sh -loadlic -f:<path_to_lic-file> -C:<solr-url>
-U: Administrator -P:<pwd>
```



```
Example:

>bin/jkool-cmd.sh -loadlic -f:$APIN_HOME/misc/tools/jkool-
dbapi/current/scripts/<newlicense.lic> -C:http://localhost:8983
-U: Administrator -P:admin
```

3. View the license by running the command, `get license`, from the Nastel XRay UI or Query Browser.

   Results in the Nastel XRay UI should be similar to the following figure.

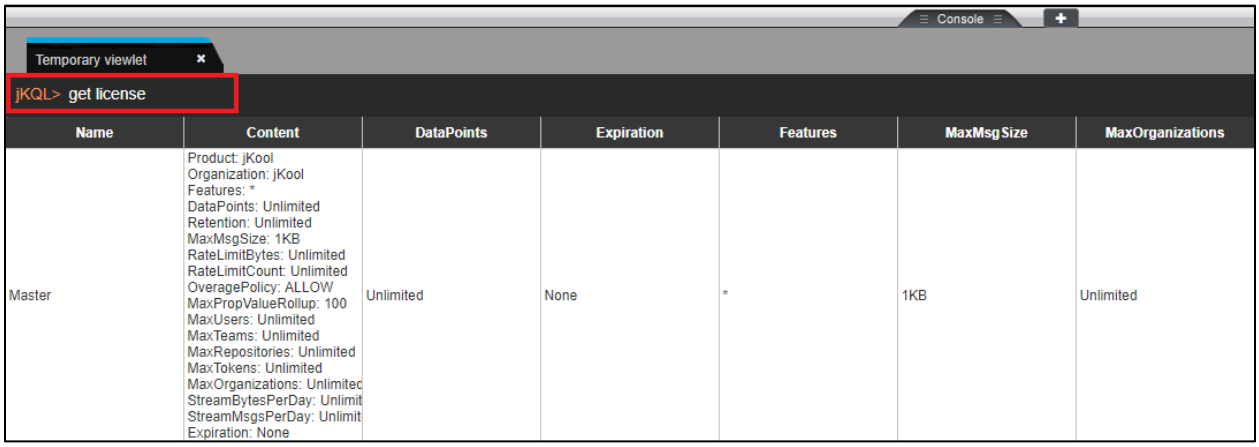

Figure 4.2.5.4-A. Nastel XRay UI – Get License Command

Results in the Query Browser should be similar to the following figure.

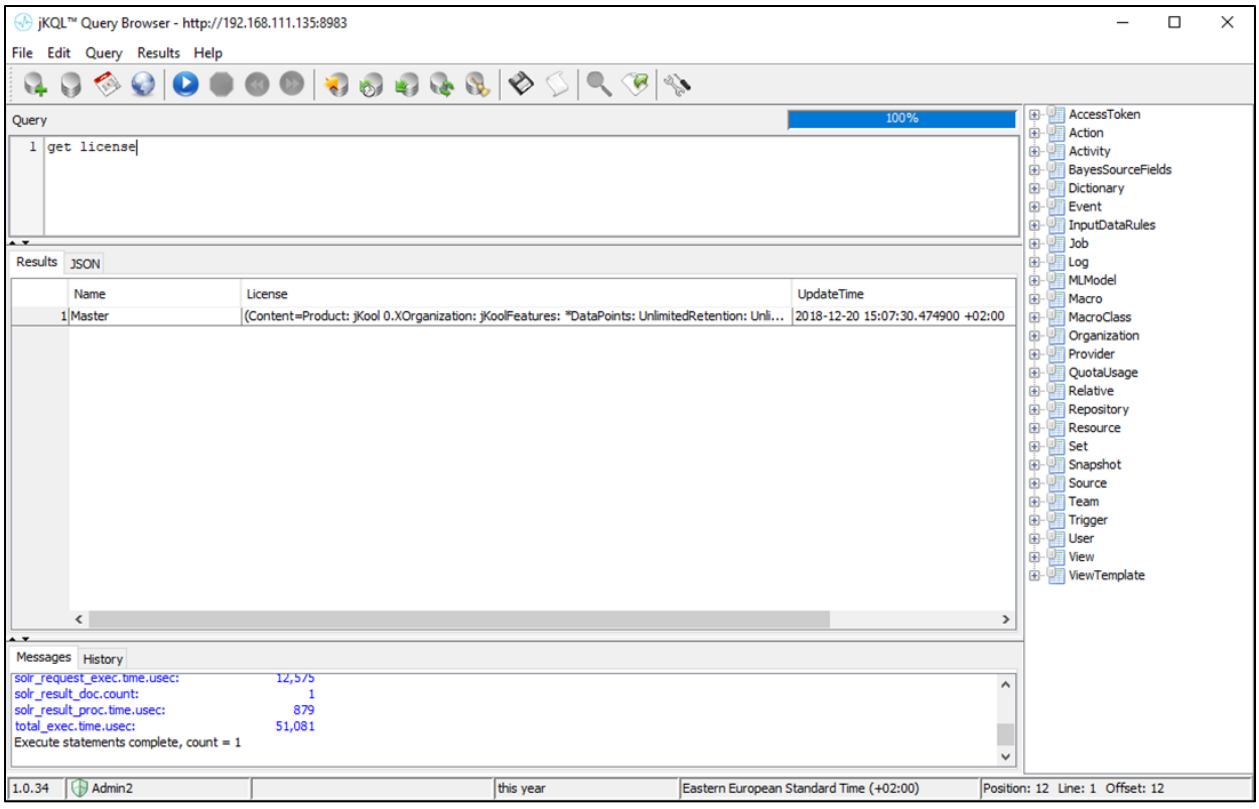

Figure 4.2.5.4-B. Query Browser – Get License Command

4.2.5.5 Load Feature Set

1. Use the **jkool-cmd.sh** script to load the defined feature set:

```
>cd $APIN_HOME/misc/tools/jkool-dbapi/current/
Run
>bin/jkool-cmd.sh solr -load -f:$APIN_HOME/misc/tools/jkool-
dbapi/current/scripts/features.csv -C:http://<solrip>:8983 -
S:http://<solrip>:8983 -U: Administrator -P:admin

Example:
> bin/jkool-cmd.sh -load -f:$APIN_HOME/misc/tools/jkool-
dbapi/current/scripts/features.csv -C:http://localhost:8983
-U: Administrator -P:admin
```

4.2.5.6 Populate Initial Database

1. Use the **jkool-cmd.sh** script to create the Nastel XRay administrator user as follows:

```
>cd $APIN_HOME/misc/tools/jkool-dbapi/current/bin
>./jkool-cmd.sh solr -run -f:$APIN_HOME/misc/tools/jkool-
dbapi/current/scripts/admin.jkql -C:http://<solrip>:8983 -
S:http://<solrip>:8983 -U: Administrator -P:admin
```

To confirm the administrator user was successfully created, within Solr select **jkooladmin.registeredusers** from the drop-down list on the left side of the screen. A screen similar to the following will display.

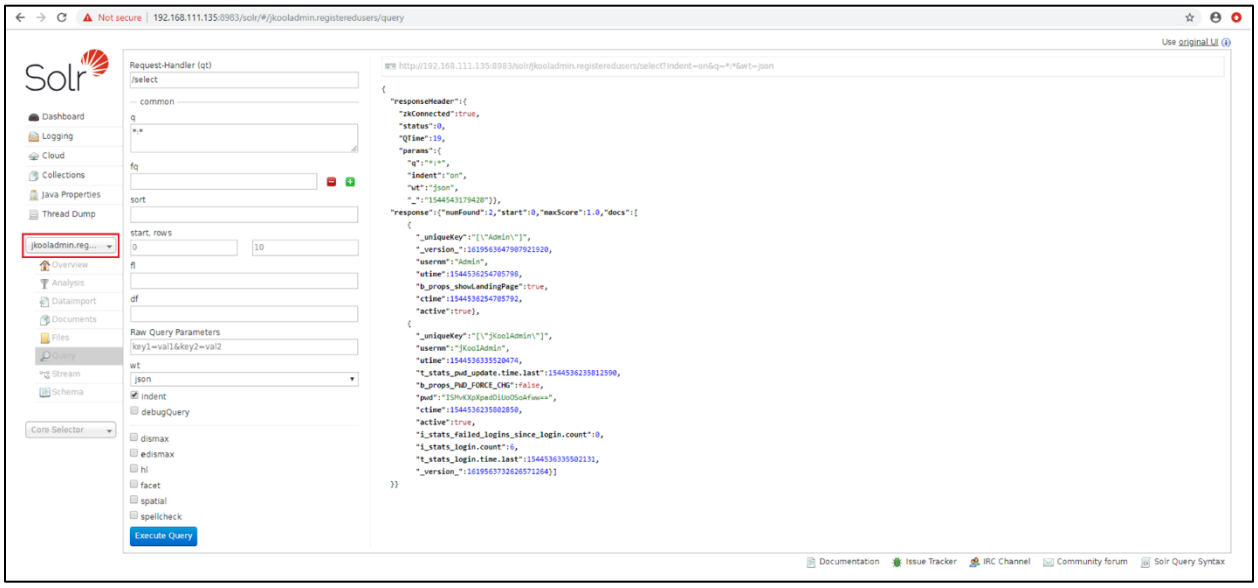

Figure 4.2.5.6-A. Solr – Confirm Creation of Admin User

2. Use the **jkool-cmd.sh** script to create a default organization, repository, team and token, as follows:

```
>cd $APIN_HOME/misc/tools/jkool-dbapi/current/bin
>./jkool-cmd.sh solr -run -f:$APIN_HOME/misc/tools/jkool-
```



```
dbapi/current/scripts/xray-admin.jkql -C:http://<solrip>:8983 -
S:http://<solrip>:8983 -U:Admin -P:admin
```

To confirm the default organization was successfully created, within Solr select **jkooladmin.organizations** from the drop-down list on the left side of the screen. A screen similar to the following will display.

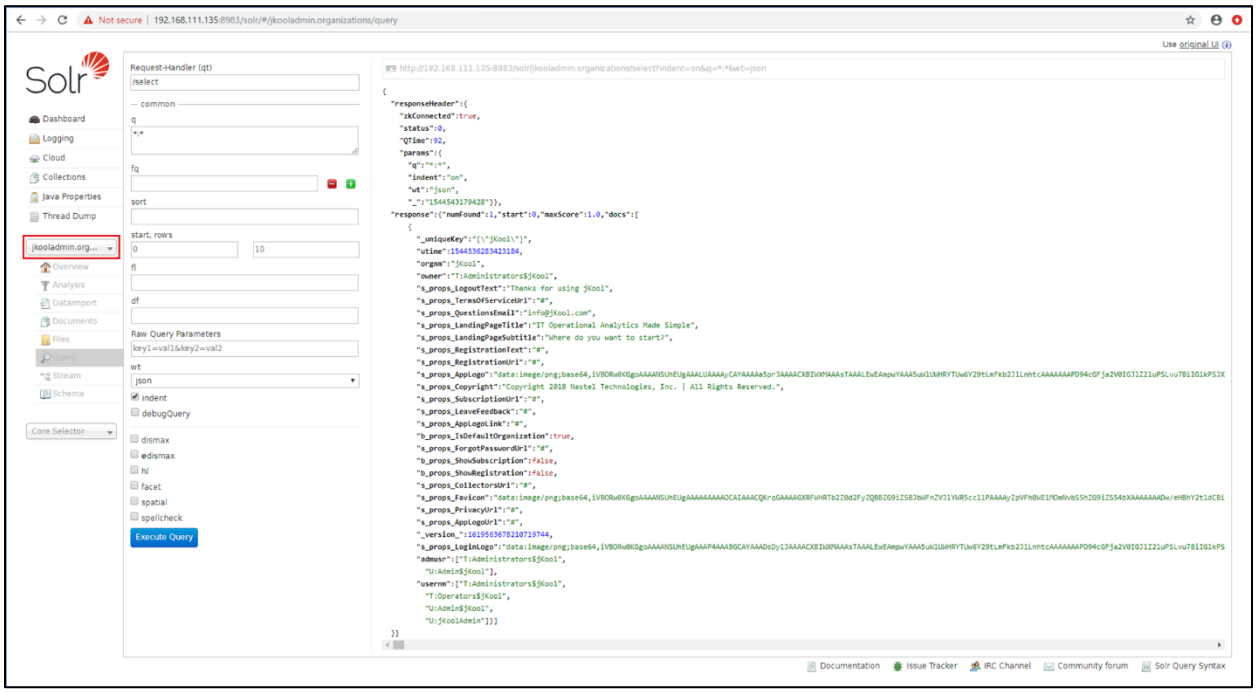

Figure 4.2.5.6-B. Solr – Confirm Creation of Organization

To confirm the default repository was successfully created, within Solr select **jkooladmin.repositories** from the drop-down list on the left side of the screen. A screen similar to the following will display.

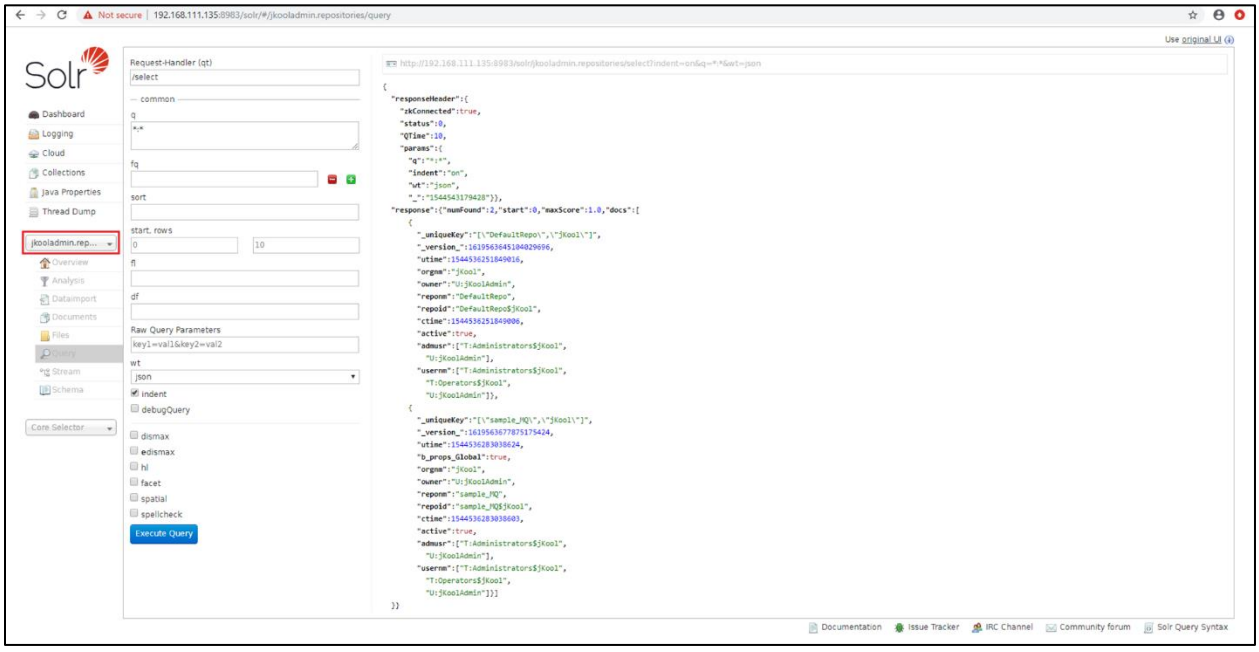

Figure 4.2.5.6-C. Solr – Confirm Creation of Repository

To confirm the default team was successfully created, within Solr select **jkooladmin.teams** from the drop-down list on the left side of the screen. A screen similar to the following will display.

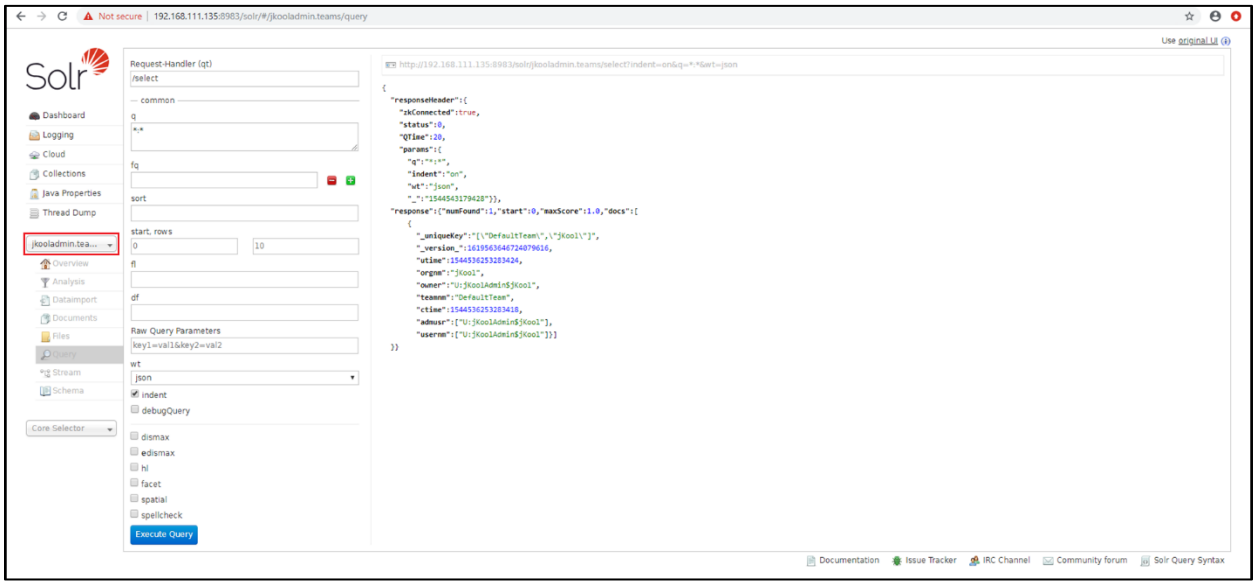

Figure 4.2.5.6-D. Solr – Confirm Creation of Team

To confirm the default token was successfully created, within Solr select **jkooladmin.accesstokens** from the drop-down list on the left side of the screen. A screen similar to the following will display.

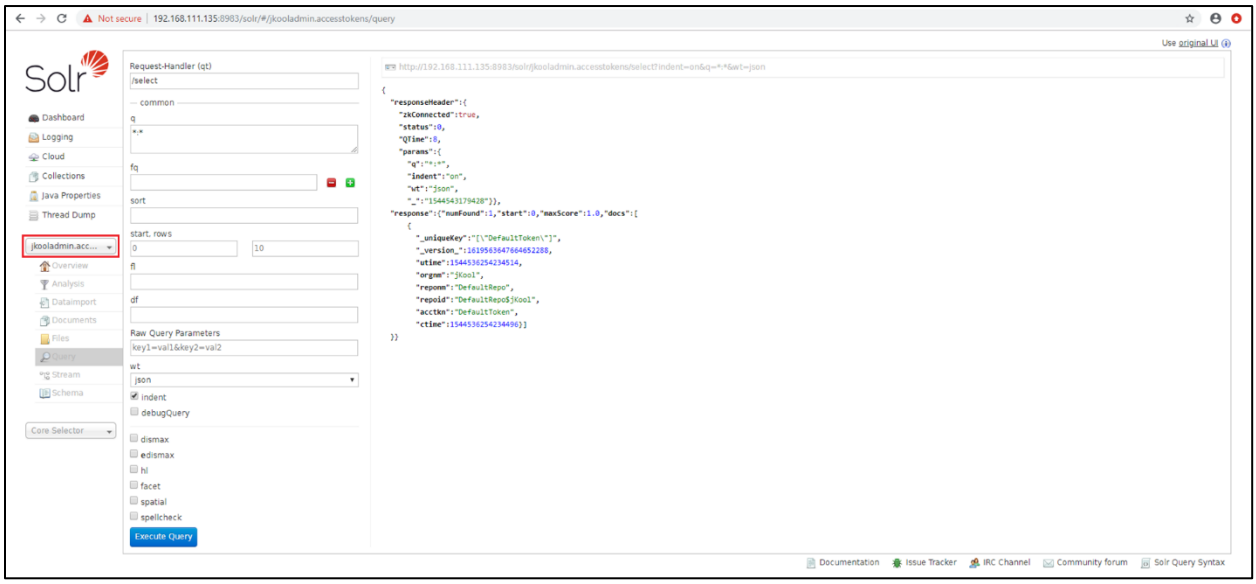

Figure 4.2.5.6-E. Solr – Confirm Creation of Token

Chapter 5: Startup

5.1 Nastel XRay Standalone Appliance

The system's components can be started using an interactive script, individual commands *or* component scripts. Each method is explained below.

5.1.1 Startup with an Interactive Script

An interactive script is provided to start the complete system. Default configurations for the components will be used. Run the script by using the following commands. The script includes time delays, so please be sure to wait for the prompts to proceed.

```
> cd $APIN_HOME/sbin
> ./start.sh all
```

The component startup actions will appear within the prompts as follows:

- Confirm no processes are running
- Start ZooKeeper
- Check ZooKeeper
- Start Solr Cloud – this step may take some time
- Check Solr Cloud
- Start Kafka
- Check Kafka
- Start Active MQ
- Check Active MQ
- Start Storm components
  - Storm Nimbus
  - Storm Supervisor
  - Storm UI
- Check Storm components
- Start AutoPilot components
  - Domain Server
  - CEP Server
  - Web Server
- Check AutoPilot components
- Check all components
- Start Storm Topology

5.1.2 Startup with Individual Scripts or Commands

The table below is a quick reference for starting the individual components in the correct order with either the startup scripts provided, or from the command line.


NOTE

Run **start.sh** without parameters to display an explanation of options.

Table 5-1. Component Commands Quick Reference

Component	Scripts
Master Scripts Directory	`$APIN_HOME/sbin/`
Check Running Nastel XRay Processes	`>cd $APIN_HOME/sbin/` `>./show.sh`
Start All Processes	`>cd $APIN_HOME/sbin` `$ ./start.sh all`
Start ZooKeeper	`>cd $APIN_HOME/sbin/` `>./start.sh zoo`
ZooKeeper Manual Start	`>cd $APIN_HOME/zookeeper/current` `>nohup bin/zkServer.sh start &`
Check ZooKeeper Process	`>ps –ef |grep zookeeper`
Start Solr	`>cd $APIN_HOME/sbin/` `>./start.sh solr`
Solr Node Manual Start	`>cd $APIN_HOME/solr/current` `>bin/solr start`
Check Solr Process	`>ps –ef | grep solr`

Start Kafka
Kafka Manual Start
	`>cd $APIN_HOME/sbin/`
	`>./start.sh kafsrv`
	`>cd $APIN_HOME/kafka/current`
	`>nohup bin/kafka-server-start.sh config/server.properties > kafka.out 2>&1 &`
	`>ps –ef | grep kafka`

Start ActiveMQ	`>cd $APIN_HOME/sbin/` `>./start.sh mq`
ActiveMQ Manual Start	`>cd $APIN_HOME/actmq/current` `>nohup bin/activemq start &`
Check ActiveMQ Process	`>ps –ef | grep activemq`
Start Storm	`>cd $APIN_HOME/sbin/` `>./start.sh storm`
Storm Manual Start	`>cd $APIN_HOME/storm/current`
Nimbus	`>nohup bin/storm nimbus &`
Check Process	`>ps –ef | grep storm`
Supervisor	`>nohup bin/storm supervisor &`
Check Process	`>ps –ef | grep storm`
Storm UI	`>nohup bin/storm ui &`

Check Process	>ps –ef | grep storm
Start APM6 All	>cd $APIN_HOME/sbin/ >./start.sh ap
APM6 Domain Manual Start	>cd $APIN_HOME/AutoPilotM6/naming >nohup ./ATPNAMES &
Check APM6 Domain Process	>ps –ef | grep ATPNAMES
APM6 CEP Manual Start	>cd $APIN_HOME/AutoPilotM6/localhost >nohup ./ATPNODE &
Check APM6 CEP Process	>ps –ef | grep ATPNODE
APM6 WEB Manual Start	>cd $APIN_HOME/sbin/ >./start.sh web >cd $APIN_HOME/AutoPilotM6/apache-tomcat7 >bin/startup.sh

5.1.3 Manual Startup with Individual Commands

This section reviews the manual startup process where the user will change directories and start each component from its built-in start command. This is different from using the individual start scripts which allow the user to start components individually from a single location.

From a command line, start each of the components in the order listed below.

ZooKeeper

ZooKeeper must be running for Solr, Kafka and Storm. ZooKeeper is configured with **chroot** set to **`/solr`** which separates Solr records from Storm and Kafka records.

```
>cd $APIN_HOME/zookeeper/current/bin
>./zkServer.sh start &
>ps – ef | grep zookeeper
```

ZooKeeper data files can be found in: `>cd $APIN_HOME/zookeeper/zoo_data`

Solr Cloud

Solr will start in Cloud mode with two nodes on the local host. It uses the same ZooKeeper instance as Storm. The Solr cloud nodes can be configured on other hosts.

```
>cd $APIN_HOME/solr/current
>bin/solr start   (this starts node1 of the Solr cluster, on port 8983)
>ps –ef | grep solr
```


Kafka

Kafka MUST be started before starting AutoPilot M6. Use the following to start Kafka:

```
>cd $APIN_HOME/kafka/current/bin
>nohup ./kafka-server-start.sh config/server.properties >kafka.out 2>&1 &
>ps -ef | grep kafka
```

Kafka data logs can be found in:
`>cd $APIN_HOME/kafka/kafka-data-logs`

ActiveMQ

ActiveMQ must be started before starting AutoPilot M6/XRay. Use the following to start ActiveMQ:

```
>cd $APIN_HOME/actmq/current
>nohup bin/activemq start &
>ps –ef | grep activemq
```

Storm – Real-time Cluster

A single-node Storm cluster is configured as the real-time engine which processes subscriptions and triggers. Storm uses ZooKeeper and the central repository of definitions that all Storm nodes require. ZooKeeper and Storm must be running before starting the Nastel XRay Subscription and Trigger Storm topologies.

```
>cd $APIN_HOME/storm/current
>nohup bin/storm nimbus &
>nohup bin/storm supervisor &
>nohup bin/storm ui &
>ps –ef | grep storm
```

NOTE

You should see listings for the three storm components started.

AutoPilot M6 / Nastel XRay

Domain Server

```
>cd $APIN_HOME/AutoPilotM6/naming
>nohup ./ATPNAMES &
>ps -ef |grep ATPNAMES
```

CEP Server

```
>cd $APIN_HOME/AutoPilotM6/apache-tomcat
>bin/startup.sh
>ps –ef |grep Catalina
```


Web Server

```
>cd $APIN_HOME/AutoPilotM6/localhost
>nohup ./ATPNODE &
>ps -ef | grep ATPNODE
```

Nastel XRay Subscription and Trigger Topologies

```
>cd $APIN_HOME/AutoPilotM6/jkool/scripts
>./start-storm-topology.sh <STORM_HOME>
* <STORM_HOME> = '$APIN_HOME/storm/current'
```

5.2 Nastel XRay Multi-Node Small Cluster

5.2.1 Prepare the Compute Node for the Multi-Node Small Cluster

Before starting the multi-node small cluster for the first time, the compute node must be configured to communicate with the 4 nodes Solr cluster.

Edit the **global.properties** file located in `$APIN_HOME/AutoPilotM6`. Update the following entries with the IP addresses of your Solr hosts (the DIN Nodes), and save the file.

```
property jkool.db.url = http://<DIN_Node01>:8983,
http://<DIN_Node02>:8983, http://<DIN_Node03>:8983,
http://<DIN_Node04>:8983
property jkool.solr.url = http://<DIN_Node01>:8983, http://<DIN_Node02>:8983,
http://<DIN_Node03>:8983,
http://<DIN_Node04>:8983
```

5.2.2 Start Sequence and Location of Scripts for the Multi-Node Small Cluster

Start Sequence

In order for Nastel XRay to start and run properly, the components should always be started, and validated in the proper sequence. In the case of a multi-node cluster the 4 Solr nodes are all running on separate servers, which must all be started after ZooKeeper and before any of the other components.

- Start ZooKeeper on DCN
- Validate ZooKeeper
- Start Solr Cloud Node1 – DIN01
- Start Solr Cloud Node2 – DIN02
- Start Solr Cloud Node3 – DIN03
- Start Solr Cloud Node4 – DIN04
- Validate Solr Cloud Cluster
- Start Kafka – DCN
- Check Kafka
- Start Active MQ – DCN

- Check Active MQ – DCN
- Start Storm components – DCN (Optional)
  - Storm Nimbus
  - Storm Supervisor
  - Storm UI
- Check Storm components
- Start AutoPilot components - DCN
  - Domain Server
  - CEP Server
  - Web Server
- Check AutoPilot components
- Check all components
- Start Storm Topology – DCN (Optional)

Location of Scripts

Script files for the Multi-Node Small Cluster components are located in the following directory on each node in the cluster $APIN_HOME/sbin/.

- On the DCN scripts are provided to start all components, though you will not need to run the Solr start scripts from the DCN in this configuration.
- On each of the 4 DIN Nodes, a script file is provided to start the Solr node on that server.

5.2.3 First Start of Multi-Node Small Cluster

If this is the first time starting the Multi-Node Small Cluster, then ZooKeeper and Solr may already be running after following the steps detailed in *Section 4.2 Nastel XRay Multi-Node Cluster*. If ZooKeeper and the Solr cluster are already running, run the startup procedure begging with starting Kafka.

5.2.4 Startup with Individual Scripts or Commands

The table below is a quick reference for starting the individual components in the correct order with either the startup scripts provided, or from the command line.

NOTE

Run **start.sh** without parameters to display an explanation of options.

Table 5-2. Component Commands Quick Reference

Component	Scripts
Master Scripts Directory	`$APIN_HOME/sbin/`
Check Running Nastel XRay Processes	`>cd $APIN_HOME/sbin/` `>./show.sh`
Start All Processes	`>cd $APIN_HOME/sbin` `>./start.sh all`
Start ZooKeeper	`>cd $APIN_HOME/sbin/` `>./start.sh zoo`

ZooKeeper Manual Start	>cd $APIN_HOME/zookeeper/current >nohup bin/zkServer.sh start &
Check ZooKeeper Process	>ps –ef |grep zookeeper
Start Solr (Each of the 4 Solr nodes needs to be started separately on the DIN nodes)	>cd $APIN_HOME/sbin/ >./start.sh solr
Solr Node Manual Start	>cd $APIN_HOME/solr/current >bin/solr start
Check Solr Process	>ps –ef | grep solr
Start Kafka Kafka Manual Start	>cd $APIN_HOME/sbin/ >./start.sh kafsrv >cd $APIN_HOME/kafka/current >nohup bin/kafka-server-start.sh config/server.properties > kafka.out 2>&1 & >ps –ef | grep kafka
Start ActiveMQ	>cd $APIN_HOME/sbin/ >./start.sh mq
ActiveMQ Manual Start:	>cd $APIN_HOME/actmq/current >nohup bin/activemq start &
Check ActiveMQ Process:	>ps –ef | grep activemq
Start Storm	>cd $APIN_HOME/sbin/ >./start.sh storm
Storm Manual Start	>cd $APIN_HOME/storm/current
Nimbus	>nohup bin/storm nimbus &
Check Process	>ps –ef | grep storm
Supervisor	>nohup bin/storm supervisor &
Check Process	>ps –ef | grep storm
Storm UI	>nohup bin/storm ui &
Check Process	>ps –ef | grep storm
Start APM6 All	>cd $APIN_HOME/sbin/ >./start.sh ap
APM6 Domain Manual Start	>cd $APIN_HOME/AutoPilotM6/naming >nohup ./ATPNAMES &
Check APM6 Domain Process	>ps –ef | grep ATPNAMES
APM6 CEP Manual Start	>cd $APIN_HOME/AutoPilotM6/localhost >nohup ./ATPNODE &

Check APM6 CEP Process	`>ps –ef | grep ATPNODE`

APM6 WEB Manual Start	`>cd $APIN_HOME/sbin/`
	`>./start.sh web`
	`>cd $APIN_HOME/AutoPilotM6/apache-tomcat7`
	`>bin/startup.sh`

Chapter 6: System Validation

To complete system validation, you must either have your server running with a desktop environment such as GNOME, or have X Windows with X11 forwarding set up and a tool such as Putty, Cygwin or MobaXterm.

Validation can also be done using a local web browser and a local terminal (Linux) or command prompt (Windows) version, or Nastel AutoPilot Enterprise Manager and a local installation of the Nastel XRay Query Browser package.

If validation is unsuccessful, please refer to *Chapter 8, Troubleshooting*, or try starting the individual components again.

If you are running a Linux desktop or remote X-Windows environment, you can access all components with the host name **localhost**. If you are using remote components you will need to know the IP address of your system. **localhost** will be used in the examples below.

6.1 Validate ZooKeeper

Enter the following in a command line:

```
>ps –ef | grep zookeeper
>echo stat | nc 127.0.0.1 2181
```

Results should be similar to the following:

```
[nastel@localhost bin]$ ps -ef | grep zookeeper
nastel   12771 49795  0 17:31 pts/0    00:00:00 grep --color=auto zookeeper
nastel   51497     1  0 Feb13 pts/0    00:01:56
/datafs/apps/xray/nastel/java/current/bin/java -Dzookeeper.log.dir=. -
Dzookeeper.root.logger=INFO,CONSOLE -cp
/datafs/apps/xray/nastel/zookeeper/current/bin/../build/classes:/datafs/apps
/xray/nastel/zookeeper/current/bin/../build/lib/*.jar:/datafs/apps/xray/nast
el/zookeeper/current/bin/../lib/slf4j-log4j12-
1.7.25.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib/slf4j-api-
1.7.25.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib/netty-
3.10.6.Final.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib/log4j
-1.2.17.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib/jline-
0.9.94.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../lib/audience-
annotations-
0.5.0.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../zookeeper-
3.4.13.jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../src/java/lib/*.
jar:/datafs/apps/xray/nastel/zookeeper/current/bin/../conf: -
Dcom.sun.management.jmxremote -
Dcom.sun.management.jmxremote.local.only=false
org.apache.zookeeper.server.quorum.QuorumPeerMain
/datafs/apps/xray/nastel/zookeeper/current/bin/../conf/zoo.cfg
---------------------------------------------------------------
[nastel@localhost bin]$ echo stat | nc 127.0.0.1 2181
Zookeeper version: 3.4.13-2d71af4dbe22557fda74f9a9b4309b15a7487f03, built on
06/29/2018 04:05 GMT
```



```
Clients:
 /0:0:0:0:0:0:0:1:34920[1](queued=0,recved=11401,sent=11402)
 /0:0:0:0:0:0:0:1:36574[1](queued=0,recved=8973,sent=8973)
 /127.0.0.1:38178[0](queued=0,recved=1,sent=0)

Latency min/avg/max: 0/0/201
Received: 83830
Sent: 86175
Connections: 3
Outstanding: 0
Zxid: 0xd423
Mode: standalone
Node count: 2504
```

6.2 Validate Solr

Before validating Solr, please confirm that the data was uploaded to Solr properly as described in *Section 4.2.5*. Enter the following in a command line to validate Solr:

```
>ps –ef | grep solr
```

Results similar to the following screenshot should appear:

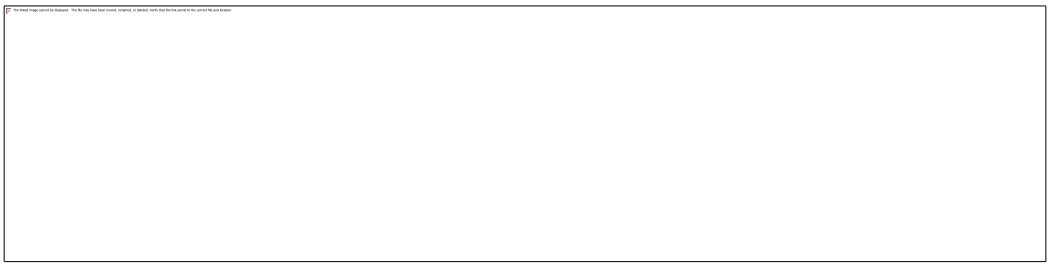

Figure 6.2-A. Validate Solr

To access the Solr UI from a web browser, use the following address:
http://<*serverip*>:8983/solr

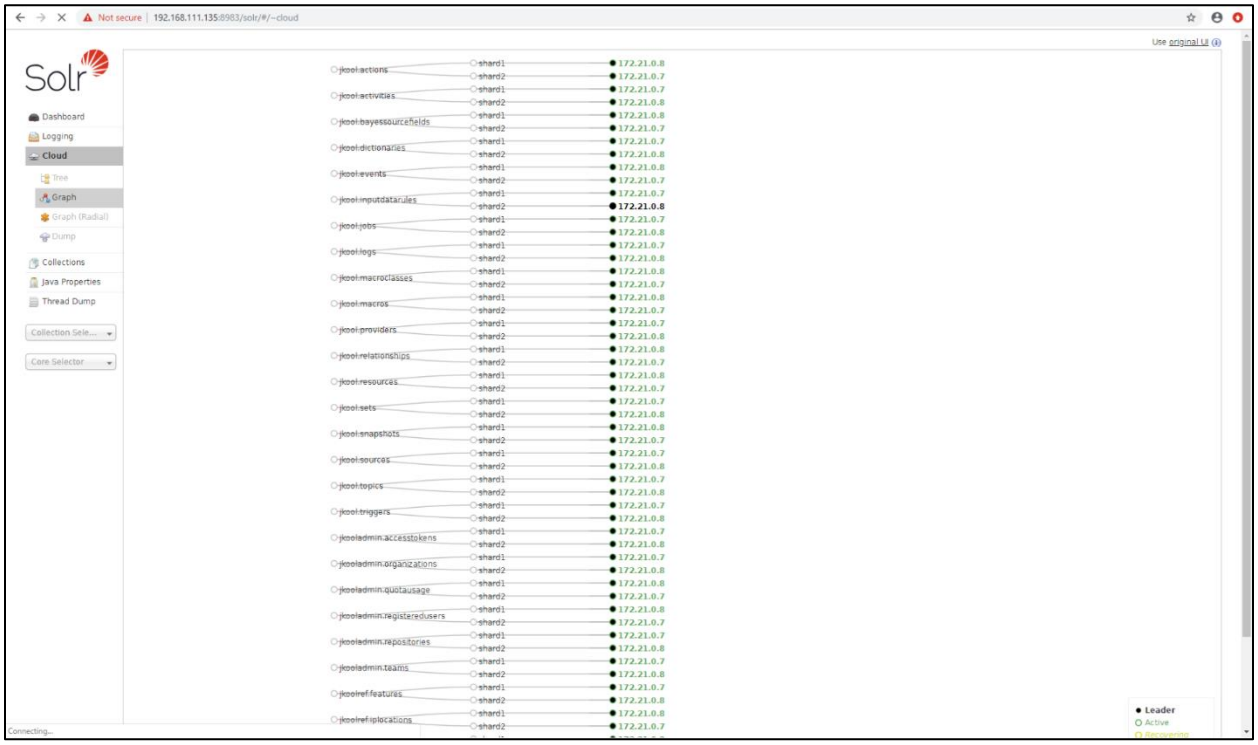

Figure 6.2-B. Solr

6.3 Validate Kafka

Enter the following in a command line to validate Kafka:

```
>ps -ef | grep kafka
> echo dump | nc localhost 2181 | grep brokers
```

Results similar to the following screenshot should appear:

Figure 6.3-A. Validate Kafka

The following command must be run on a ZooKeeper node to get the list of Kafka brokers from ZooKeeper.

```
> echo dump | nc localhost 2181 | grep brokers
```

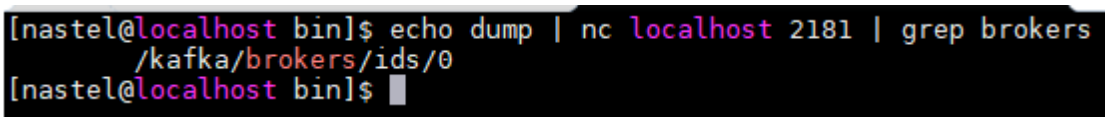

Figure 6.3-B. Get List of Kafka Brokers from ZooKeeper

6.4 Validate ActiveMQ

Confirm that there are queues connected to Nastel XRay within ActiveMQ. To check this, launch ActiveMQ and go to **Queues** (located within the navigation menu). There should be three default queues.

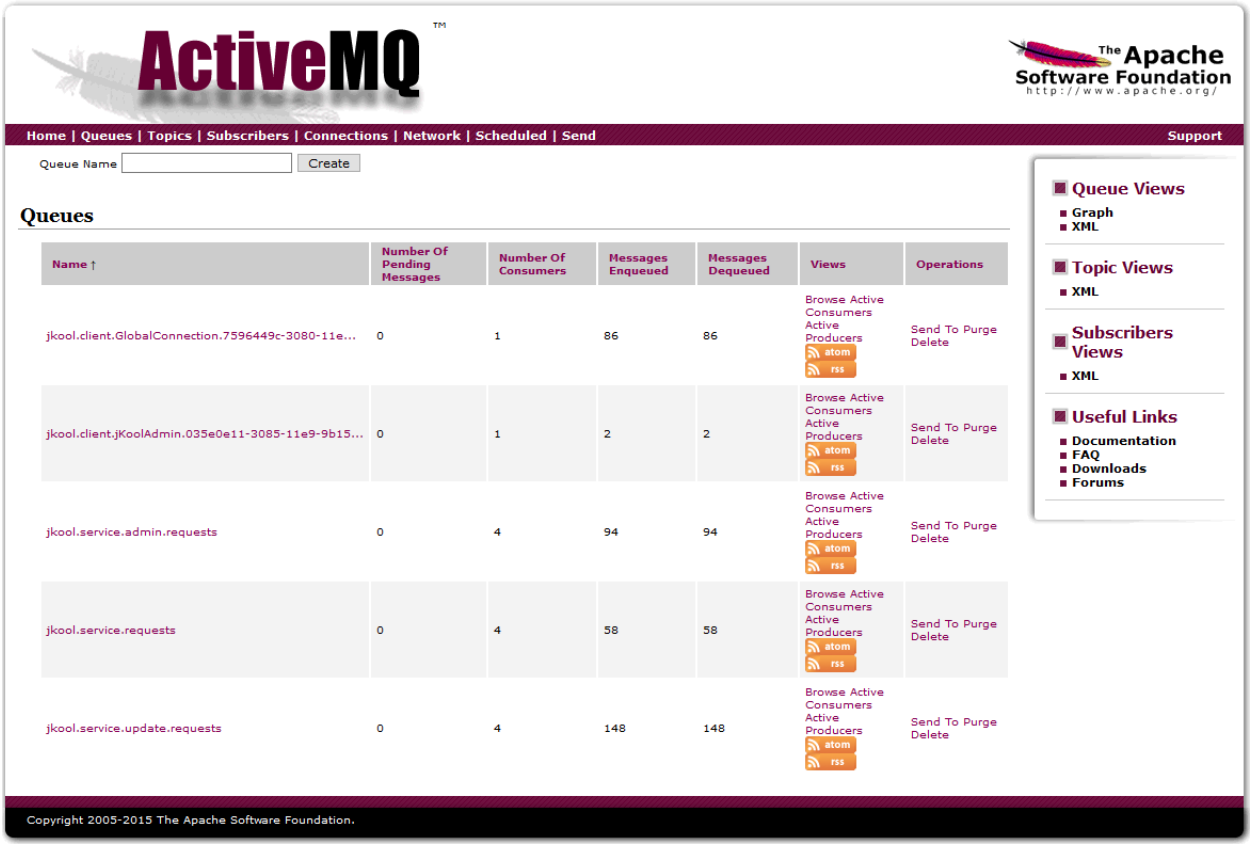

Figure 6.4-A. Default Queues

Enter the following in a command line:

```
UI Via Browser
Userid:admin
Pwd:admin
```

To access Active MQ from a web browser, use the following address:

http://<serverip>:8161/admin

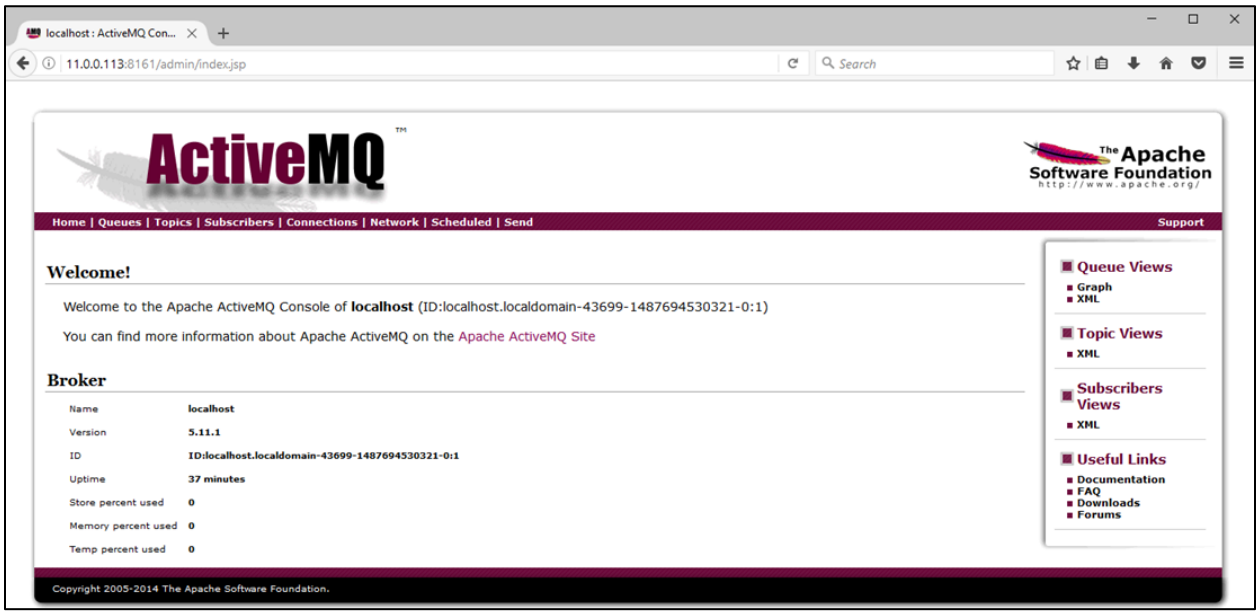

Figure 6.4-B. ActiveMQ

6.5 Validate Storm

Confirm that the status of the topology is Active. Also, the topology should have existing workers and executors. If the number of workers and/or executors is zero, the topologies may not function properly.

Enter the following in a command line:

```
>ps –ef | grep storm
```

To access the Storm UI from a web browser, use the following address:
http://localhost:8088

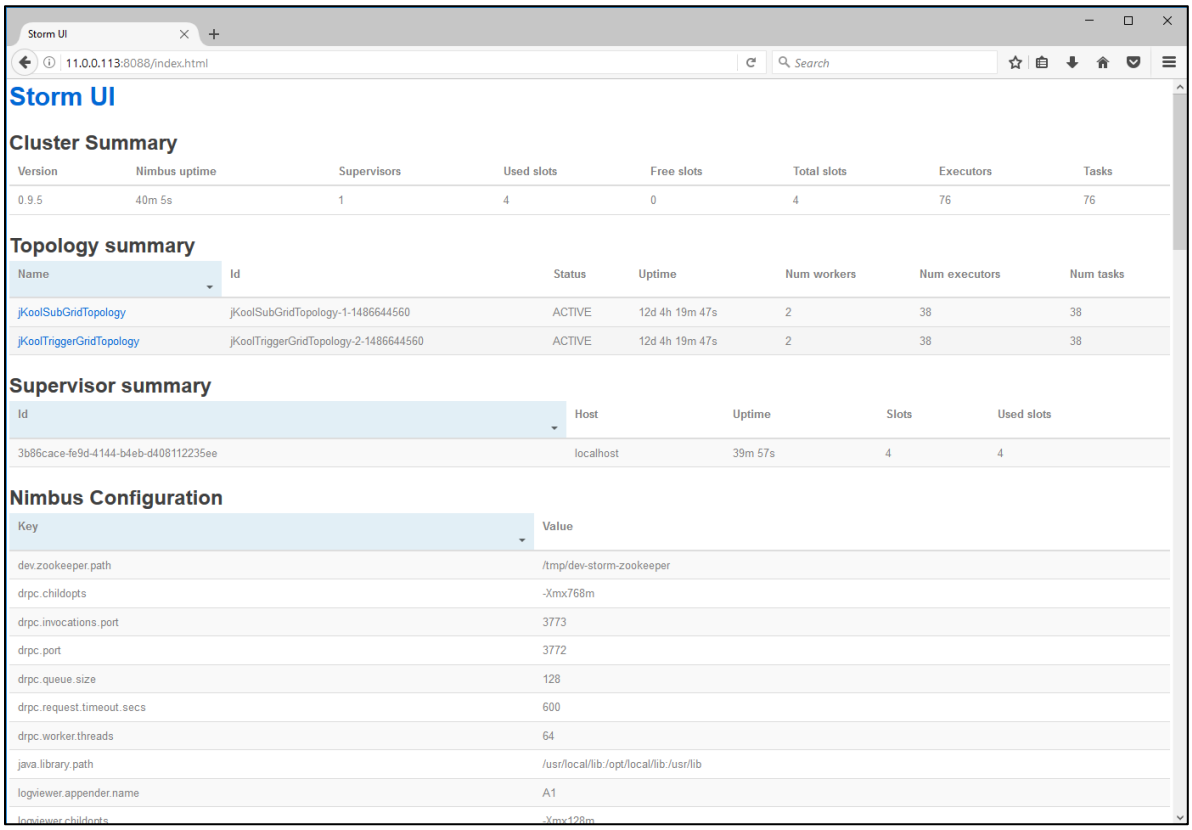

Figure 6.5-A. Storm

6.6 Validate AutoPilot

Enter the following in a command line:

```
>ps -ef | grep ATPNODE
>ps -ef | grep ATPNAMES
>ps -ef | grep catalina
```

Results should be similar to the following:

```
[nastel@localhost bin]$ ps -ef |grep ATPNODE
nastel    5135     1 35 12:41 pts/0    01:50:49 /usr/lib/jvm/java-1.8.0-
openjdk-1.8.0.171-7.b10.el7.x86_64/jre/bin/java -server -Xmx4g -Xms4g -
XX:+UseG1GC -Xloggc:gc2g.log -XX:+PrintGCDetails -XX:+PrintGCDateStamps -
XX:+PrintGCTimeStamps -XX:MaxGCPauseMillis=500 -XX:ParallelGCThreads=8 -
XX:ConcGCThreads=4 -XX:InitiatingHeapOccupancyPercent=70 com.zerog.lax.LAX
/datafs/apps/xray/nastel/AutoPilotM6/localhost/./ATPNODE.lax
/tmp/env.properties.5135
nastel   13250 49795  0 17:54 pts/0    00:00:00 grep --color=auto ATPNODE
------------------------------------------------------------------
[nastel@localhost bin]$ ps -ef |grep ATPNAMES
nastel    4995     1  0 12:41 pts/0    00:01:41 /usr/lib/jvm/java-1.8.0-
openjdk-1.8.0.171-7.b10.el7.x86_64/jre/bin/java -server -Xmx1024m -Xms1024m -
XX:+UseG1GC -Djdk.security.defaultKeySize=DSA:1024 com.zerog.lax.LAX
```



```
/datafs/apps/xray/nastel/AutoPilotM6/naming/./ATPNAMES.lax
/tmp/env.properties.4995

nastel   13322 49795  0 17:55 pts/0    00:00:00 grep --color=auto ATPNAMES

------------------------------------------------------------

[nastel@localhost bin]$ ps -ef |grep catalina

nastel    6046     1  1 12:45 pts/0    00:04:41
/datafs/apps/xray/nastel/java/current/bin/java -
Djava.util.logging.config.file=/datafs/apps/xray/nastel/AutoPilotM6/apache-
tomcat/conf/logging.properties -
Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -
Djkool.stream.url=http://localhost:6580 -Djkclient.message.expiry.msec=0 -
Xms1g -Xmx2g -Dautopilot.home=/datafs/apps/xray/nastel/AutoPilotM6 -
Djava.awt.headless=true -
Djava.endorsed.dirs=/datafs/apps/xray/nastel/AutoPilotM6/apache-
tomcat/endorsed -classpath /datafs/apps/xray/nastel/AutoPilotM6/apache-
tomcat/bin/bootstrap.jar:/datafs/apps/xray/nastel/AutoPilotM6/apache-
tomcat/bin/tomcat-juli.jar -
Dcatalina.base=/datafs/apps/xray/nastel/AutoPilotM6/apache-tomcat -
Dcatalina.home=/datafs/apps/xray/nastel/AutoPilotM6/apache-tomcat -
Djava.io.tmpdir=/datafs/apps/xray/nastel/AutoPilotM6/apache-tomcat/temp
org.apache.catalina.startup.Bootstrap start

nastel   13337 49795  0 17:56 pts/0    00:00:00 grep --color=auto catalina
```

Launch AutoPilot Enterprise Manager. Use the following login information:

User Name: Admin
Password: admin
Domain Server: <*serverip*>:2323

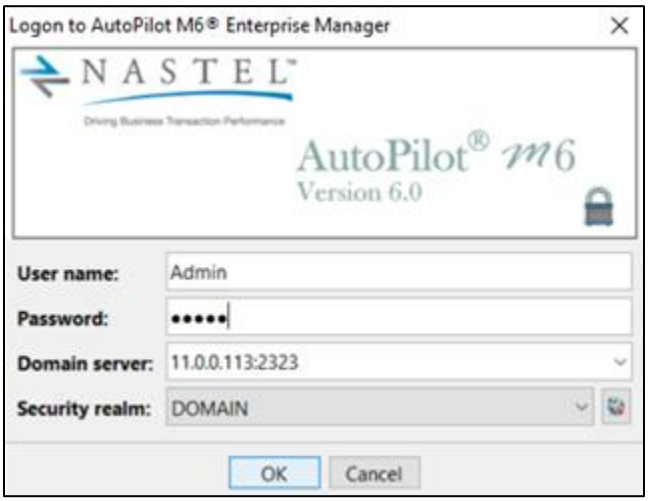

Figure 6.6-A. Log in to AutoPilot M6

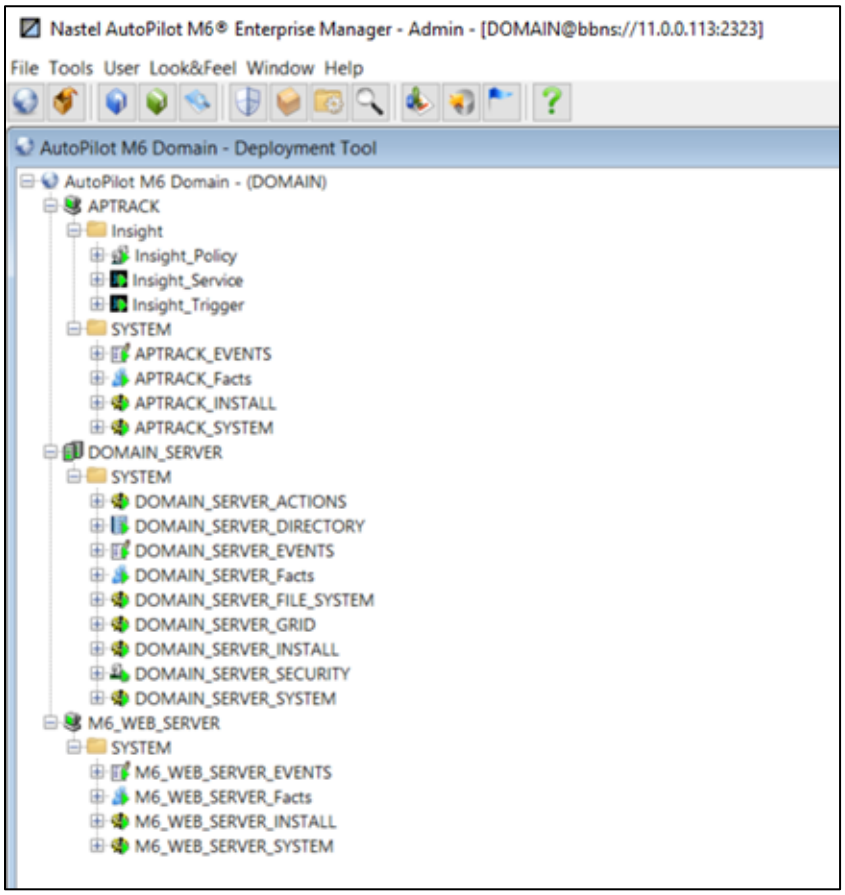

Figure 6.6-B. Log on to AutoPilot

6.7 Validate Nastel XRay

To access the Nastel XRay UI from a web browser, use the following address:
http://<*serverip*>:8080/*yourorgname*/login.jsp

Use the following log in credentials:

User Name: Admin
Password: admin

Figure 6.7-A. Nastel XRay Login

6.7.1 Validating with the Default Repository

If the preconfigured standalone appliance configuration is used, the below screen will appear. By default, the **DefaultRepo** repository will be loaded with a preconfigured dashboard named, **test**.

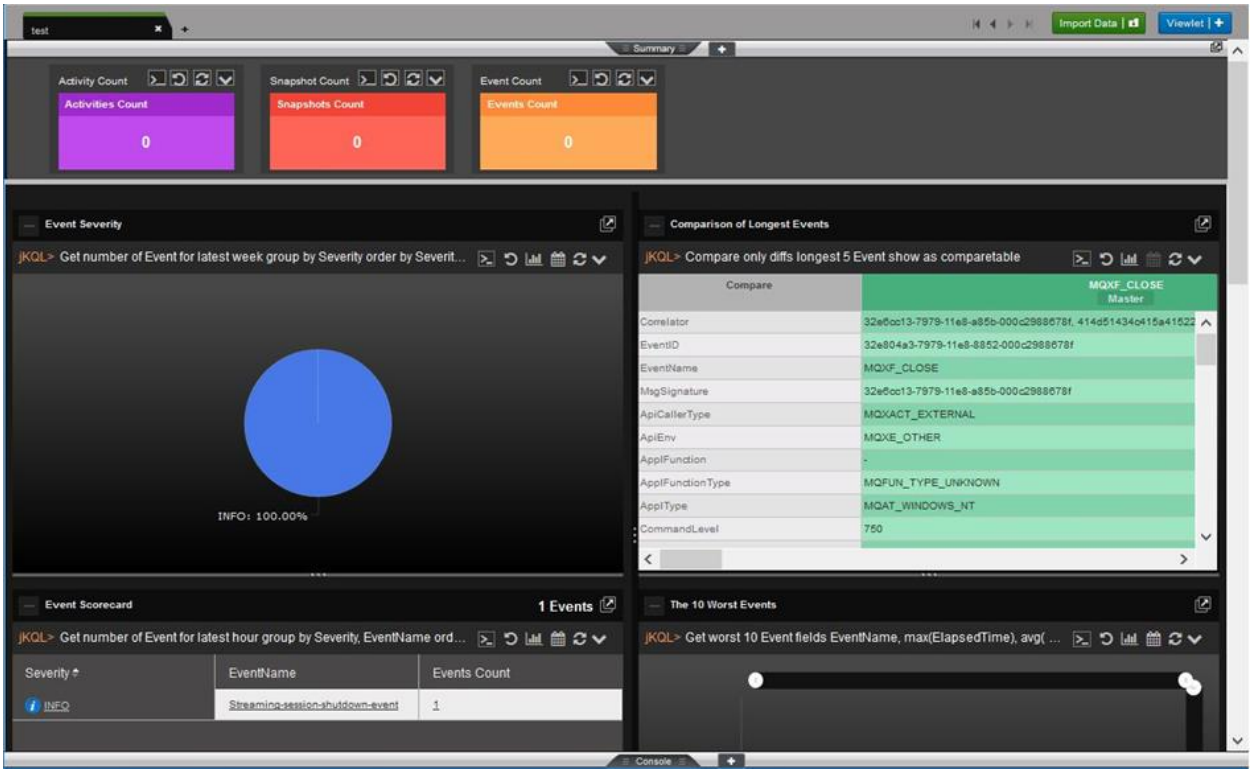

Figure 6.7.1-A. DefaultRepo Repository

6.7.2 Validating with a New Repository

If a new repository has been created per the instructions in *Section 4.2*, the below screen will display. The new repository will be loaded with a prompt to create a new dashboard.

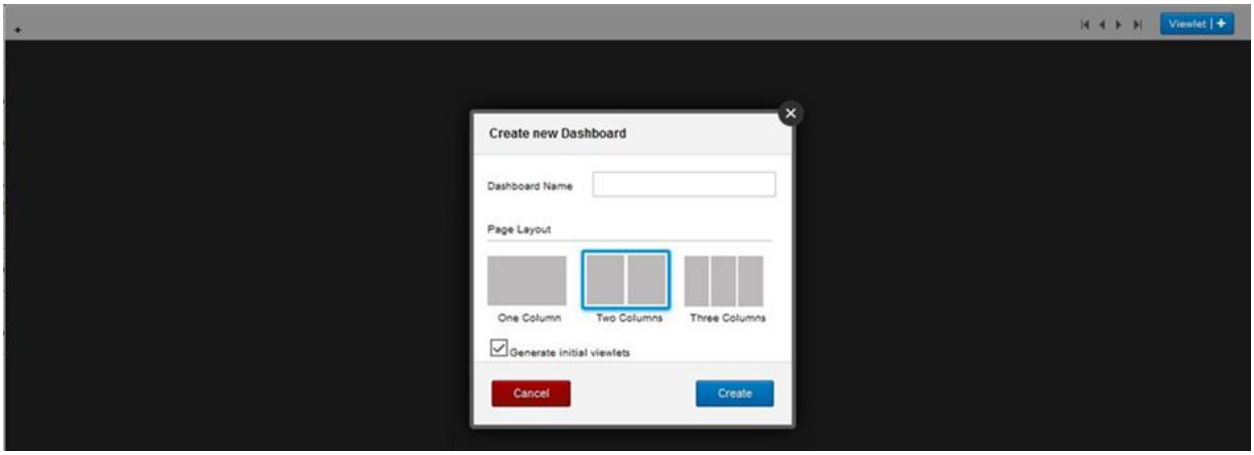

Figure 6.7.2-A. New Repository

On the *Create New Dashboard* prompt, enter a name within the **Dashboard Name** field, select the **Two Columns** layout and uncheck the **Generate Initial Viewlets** checkbox. Click **Create**.

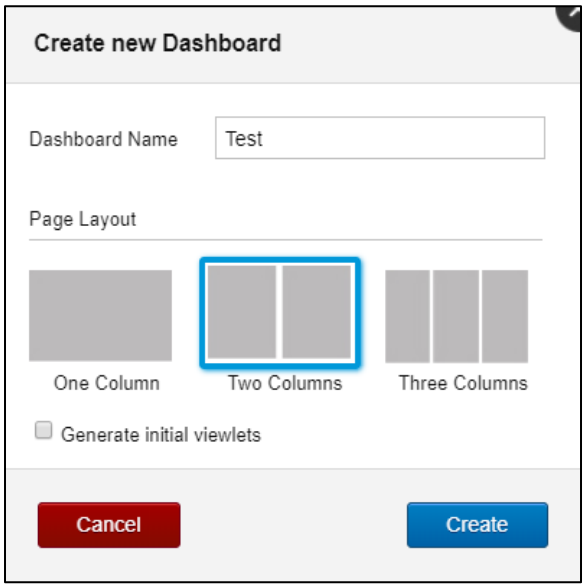

Figure 6.7.2-B. Create New Dashboard

A new dashboard named **Test**, with no viewlets is displayed:

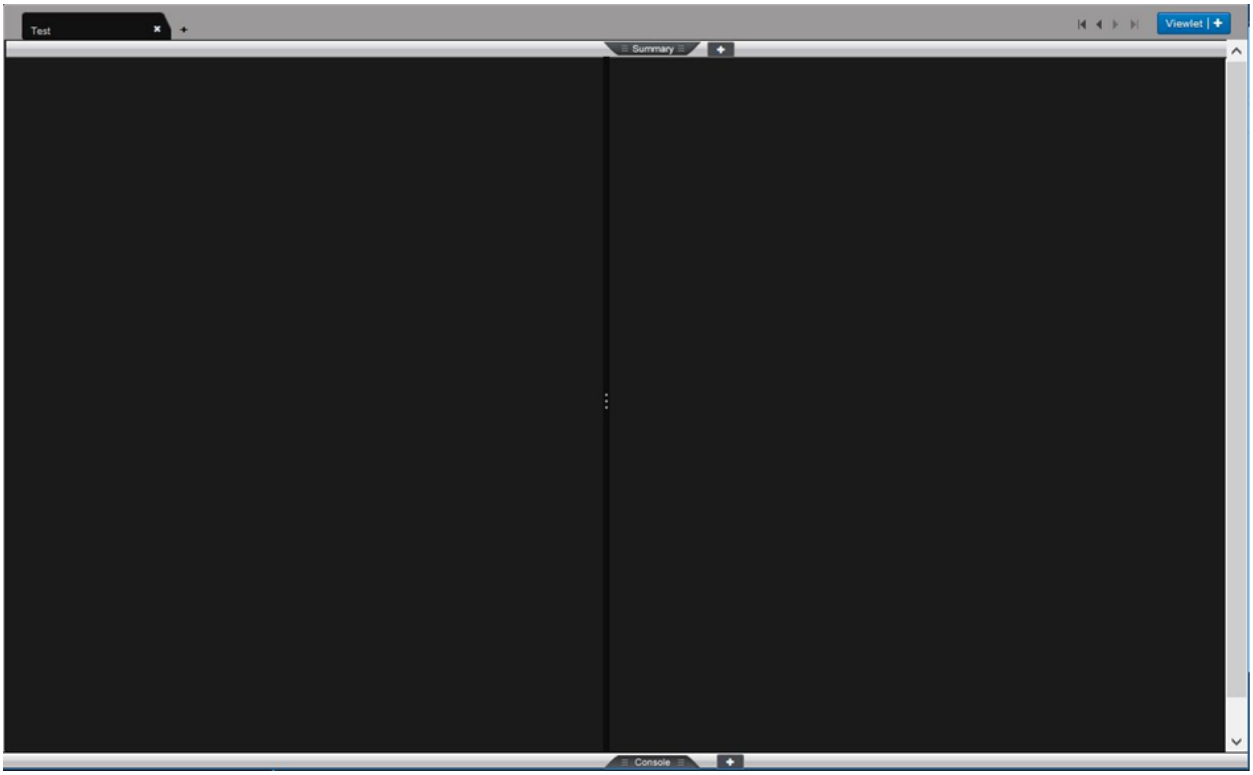

Figure 6.7.2-C. New Dashboard

Right click on the dashboard tab, and select *Save* to save the new dashboard:

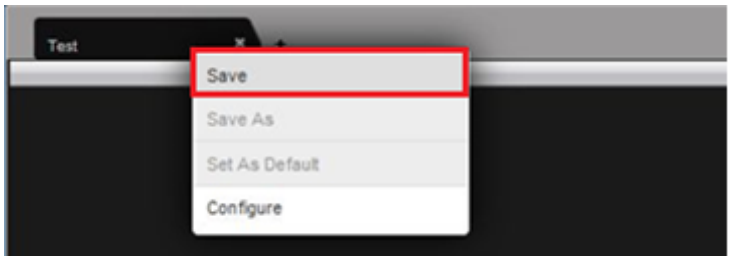

Figure 6.7.2-D. Save New Dashboard

Click **OK** on the following prompt:

Figure 6.7.2-E. Save Confirmation

To create a viewlet, click the blue **Viewlet** button located at the top right of the screen.

Figure 6.7.2-F. Viewlet Button

Select **Create Viewlet with jKQL** and click create:

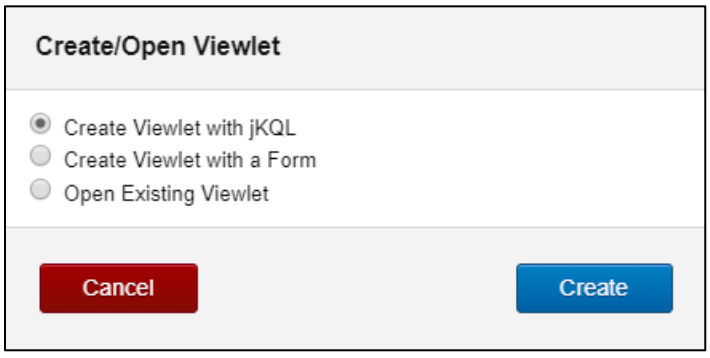

Figure 6.7.2-G. Create / Open Viewlet

Within the *Define Query* field, enter 'Get Events'. The UI will create a default viewlet name. Click **Create**.

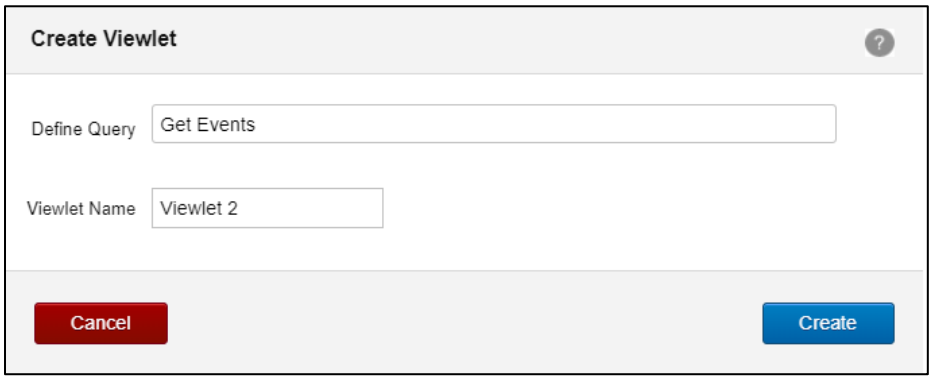

Figure 6.7.2-H. Create Viewlet

With the new viewlet created, open the drop down menu and select *Save Viewlet*.

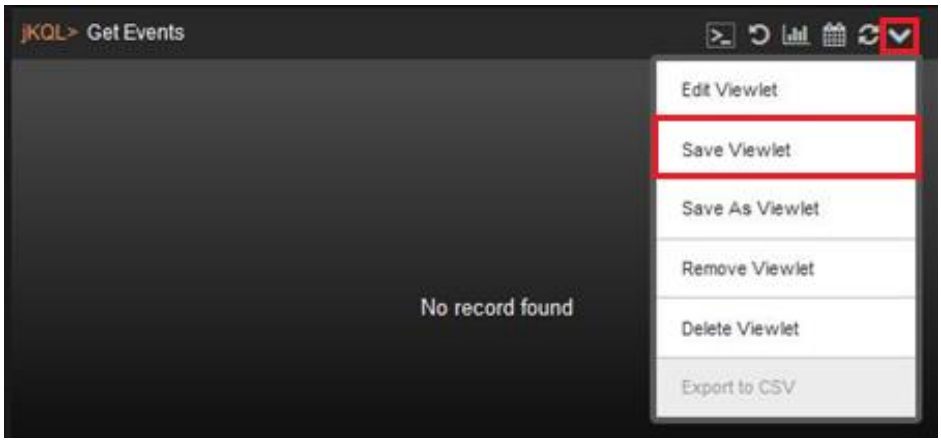

Figure 6.7.2-I. Save Viewlet

Click **OK** on the following prompt to save the viewlet.

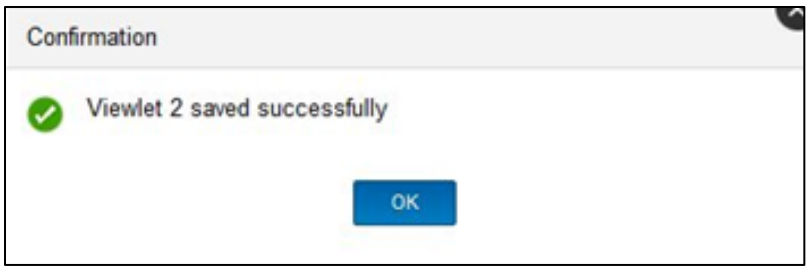

Figure 6.7.2-J. Save Confirmation

Right click on the dashboard tab again, and select *Assign to Teams* to modify permissions of the 'Administrators' team.

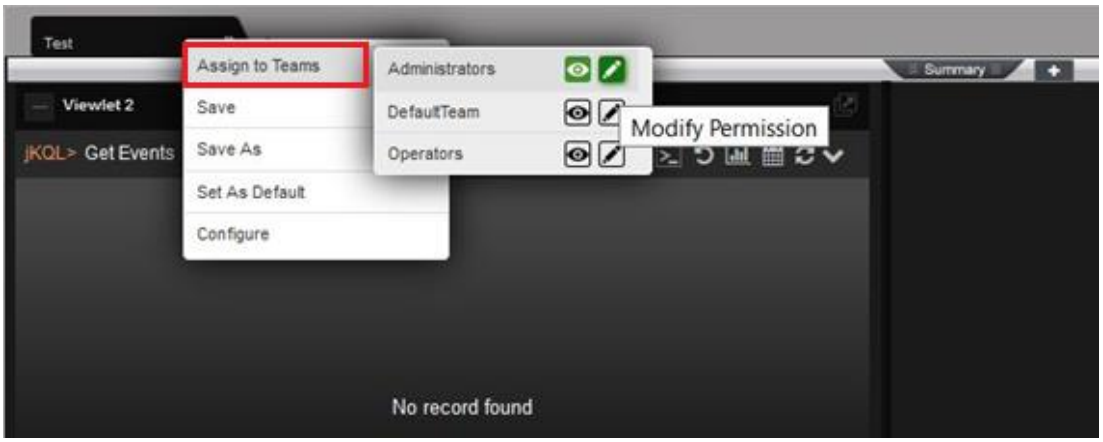

Figure 6.7.2-K. Assign to Teams

6.8 Validate Streaming Data to Nastel XRay

Validate streaming data by running a test stream using the **tnt4j-streams** package located in `$APIN_HOME/misc/tnt4j-streams/current` on the Standalone appliance or the DCN node of the Small Cluster.

First edit the file named **tnt4j.properties** to set two properties if needed, the properties are; '`event.sink.factory.Url`' and '`event.sink.factory.Token`'. These properties are already preset for the standalone appliance version, and may or may not need to be updated for the small cluster configuration.

```
>cd $APIN_HOME/misc/tnt4j-streams/current/config
>vi tnt4j.properties
```

These properties are set as follows for the standalone appliance.

```
event.sink.factory.Url: http://localhost:6580
event.sink.factory.Token: DefaultRepo@Nastel
```

If you have created a new org, repo and token as part of configuring the small cluster, you will need to edit the value of `event.sink.factory.Token` to the value of the token you have created. The value of `event.sink.factory.Url` does not need to be edited as long **asten4j-streams** is being run on the same host as the DCN.

To test streaming, use the single-log sample provided with the package, by running the **run.sh** script from the following location: `$APIN_HOME/misc/tnt4j-streams/current/run/single-log`

```
>cd $APIN_HOME/misc/tnt4j-streams/current/run/single-log
> ./run.sh
```

You should see the following in results with no exceptions or errors:

```
[nastel@localhost single-log]$ ./run.sh

2019-05-12 19:05:17,607 INFO  [main!StreamsAgent] - jKool TNT4J-Streams
v.1.7.0-20190328102040 (JVM v.18.0_151) session starting as standalone
application...|RUNTIME=8853@localhost.localdomain#SERVER=localhost.local
domain#NETADDR=127.0.0.1#DATACENTER=UNKNOWN#GEOADDR=0,0

2019-05-12 19:05:17,688 INFO  [main!StreamsAgent] - Loading TNT4J-
Streams data source configuration from file: tnt-data-source.xml |
```



```
RUNTIME=8853@localhost.localdomain#SERVER=localhost.localdomain#NETADDR=
127.0.0.1#DATACENTER=UNKNOWN#GEOADDR=0,0
..........................................................................................................................
2019-05-12 19:05:20,668 DEBUG
[9:FileLineStream:FileStream!JKCloudActivityOutput]'DefaultParseableInpu
tStreamOutput' built new tracker instance '69465387-750a-11e9-b652-
525400031131' for thread
'9':com.jkoolcloud.tnt4j.TrackingLogger@e1c7f0f{logger:TrackerImpl{jid=7
bcefd4c,name=com.jkoolcloud.tnt4j.streams.outputs.AbstractJKCloudOutput,
keep.context=false,
sink=com.jkoolcloud.jesl.tnt4j.sink.JKCloudEventSink@12a7f0f9{piped.sink
: com.jkoolcloud.tnt4j.sink.impl.slf4j.SLF4JEventSink@753b79e7}{url:
http://localhost:6580, token: xxxxxxxxxxxxxxstel, jk.handle:
com.jkoolcloud.jesl.net.JKClient@18328448}}} |
RUNTIME=8853@localhost.localdomain#SERVER=localhost.localdomain#NETADDR=
127.0.0.1#DATACENTER=UNKNOWN#GEOADDR=0,0
..........................................................................................................................
2019-05-12 19:05:21,688
INFO[9:FileLineStream:FileStream!JKCloudActivityOutput] - Stream session
status (SUCCESS) message sent! |
RUNTIME=8853@localhost.localdomain#SERVER=localhost.localdomain#NETADDR=
127.0.0.1#DATACENTER=UNKNOWN#GEOADDR=0,0
..........................................................................................................................
2019-05-12 19:05:21,690 DEBUG
[9:FileLineStream:FileStream!JKCloudActivityOutput] - Closing stream
output 'DefaultParseableInputStreamOutput' tracker '69465387-750a-11e9-
b652-525400031131': com.jkoolcloud.tnt4j.TrackingLogger@e1c7f0f{logger:
TrackerImpl{jid=7bcefd4c,
name=com.jkoolcloud.tnt4j.streams.outputs.AbstractJKCloudOutput,
keep.context=false,
sink=com.jkoolcloud.jesl.tnt4j.sink.JKCloudEventSink@12a7f0f9{piped.sink
: com.jkoolcloud.tnt4j.sink.impl.slf4j.SLF4JEventSink@753b79e7}{url:
http://localhost:6580, token: xxxxxxxxxxxxxxstel, jk.handle:
com.jkoolcloud.jesl.net.JKClient@18328448}}} |
RUNTIME=8853@localhost.localdomain#SERVER=localhost.localdomain#NETADDR=
127.0.0.1#DATACENTER=UNKNOWN#GEOADDR=0,0
..........................................................................................................................
2019-05-12 19:05:21,756 INFO
[9:FileLineStream:FileStream!FileLineStream] - Stream 'FileStream'
statistics: [activities.total=10, activities.current=10,
activities.skipped=0, activities.filtered=0, activities.lost=0,
bytes.total=978, bytes.streamed=960, time.elapsed=00:00:00.970,
rate.average=10.40aps]|RUNTIME=8853@localhost.localdomain#SERVER=localho
st.localdomain#NETADDR=127.0.0.1#DATACENTER=UNKNOWN#GEOA
```

Now log into the XRay UI, open a temporary viewlet and type in the query "*get events for the past hour*." You should see two events; one Start and one Stop as per the screenshot.

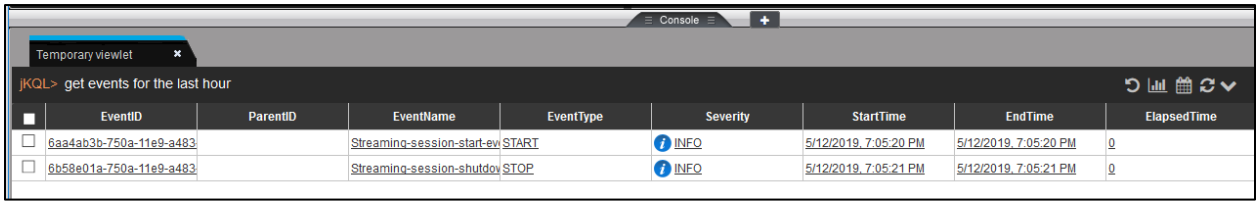

Figure 6.8-A. Start and Stop Events

This Page Intentionally Left Blank

Chapter 7: Component Shutdown

All components are shutdown manually or by doing a clean reboot of the OS. Manual shutdowns are done as follows:

Stop Storm Topologies

```
>cd $APIN_HOME/AutoPilotM6/jkool/scripts
>./stop-storm-topology.sh $APIN_HOME/storm/current
```

Stop AutoPilot

- **Web Server**
  Kill AP WebServer Process (using the shutdown script does not always end the process, so kill must be done).

```
>cd $APIN_HOME/AutoPilotM6/apache-tomcat7
>bin/shutdown.sh
```

- **CEP Server**
  Kill the ATPNODE Process
- **Domain Server**
  Kill the ATPNAMES Process

Stop Storm Components

The following command returns the 5 running Storm processes:

```
>ps –ef |grep storm
```

Kill all Storm processes:

- Storm Nimbus (and spawned child processes)
- Storm Supervisor
- Storm UI

Stop ActiveMQ

```
>cd $APIN_HOME/actmq/current
>./activemq stop
```

Stop Kafka

```
>cd $APIN_HOME/kafka/current
>bin/kafka-server-stop.sh
```


Stop Solr

```
>cd $APIN_HOME/solr/current/bin
>./solr stop –all
>./solr2 stop -all
```

Stop ZooKeeper

```
./zkServer.sh stop
```


Chapter 8: Troubleshooting

A script runs to collect and compress all log files into a **tar.gz** file for Nastel Support. The best place to locate issues is the log files for each component. The log file locations for each component are listed in the table below.

Table 8-1. Log File Locations

Component	Log File Location
Solr	**$APIN_HOME/solr/solr/var/logs**
Kafka	**$APIN_HOME/kafka/current/logs** **$APIN_HOME/kafka-data-logs**
ActiveMQ	**$APIN_HOME/actmq/current/activemq.logcd**
ZooKeeper	**$APIN_HOME/zookeeper/current/bin/zookeeper.[log|out]**
Storm	**$APIN_HOME/storm/apache-storm-0.9.5/logs** **Note**: worker-* files are for the Nastel XRay subscription and trigger topologies. Others are for Storm.
AutoPilot/XRay	**$APIN_HOME/AutoPilotM6/logs/log4j** **$APIN_HOME/AutoPilotM6/apache-tomcat7/logs**

8.1 Installation Errors

The following are potential errors that could appear during installation:

Table 8-2. Installation Errors

Error	Possible Resolution

There are no queues in Active MQ.	Within the **global.properties** file located in `$APIN_HOME/AutoPilotM6`, confirm that the correct value is set for the following property:
	`jkool.service.conn.str=localhost:61616`
	If ActiveMQ is running on a local machine, use the address, `localhost:61616`.
	If ActiveMQ is not on a local machine, use the server address of where ActiveMQ is running:
	`jkool.service.conn.str=<server_address>:61616`

The topology does not have workers.	Repeat the configuration of Storm and upload the topologies. See *Section 5.1.3* and *Section 6.5* for more information.
The expert(s) is/are stopped in AutoPilot.	**Kafka** • Confirm Kafka is running. • Check whether or not there are created topics in Kafka. • If Kafka is not running properly, search for the following within the `node.properties` file:

	jkool.kafka.server=<*kafka_server_address*>:9092 If it is missing, Nastel XRay will search for Kafka within localhost:9092. **AutoPilot** • Try to start the experts in AutoPilot via M6 Enterprise Manager.

8.2 ZooKeeper Error

Issue: An error similar to the following is encountered.

```
2019-06-20 02:32:11,401 [myid:3] - INFO  [main:FileSnap@86] - Reading
snapshot /local/home/venomhq/apinsight/nastel/zookeeper2/zoo_data/version-
2/snapshot.11b500009387
2019-06-20 02:32:11,599 [myid:3] - ERROR [main:Util@214] - Last transaction
was partial.
2019-06-20 02:32:11,601 [myid:3] - ERROR [main:QuorumPeer@692] - Unable to
load database on disk
java.io.IOException: The accepted epoch, 12ab is less than the current
epoch, 12ac
        at
org.apache.zookeeper.server.quorum.QuorumPeer.loadDataBase(QuorumPeer.java:6
89)


2019-06-20 02:32:11,603 [myid:3] - ERROR [main:QuorumPeerMain@92] -
Unexpected exception, exiting abnormally
java.lang.RuntimeException: Unable to run quorum server
        at
org.apache.zookeeper.server.quorum.QuorumPeer.loadDataBase(QuorumPeer.java:6
93)
        .
.
.
Caused by: java.io.IOException: The accepted epoch, 12ab is less than the
current epoch, 12ac
        at
org.apache.zookeeper.server.quorum.QuorumPeer.loadDataBase(QuorumPeer.java:6
89)
        ... 4 more
```

Resolution: Run the following command.

```
cp currentEpoch acceptedEpoch
```

Restart Solr and allow it to recover.

Index

