Winter Ready 2019 Technical Lab
Level: 300
Contacts: omidm, dhgoel

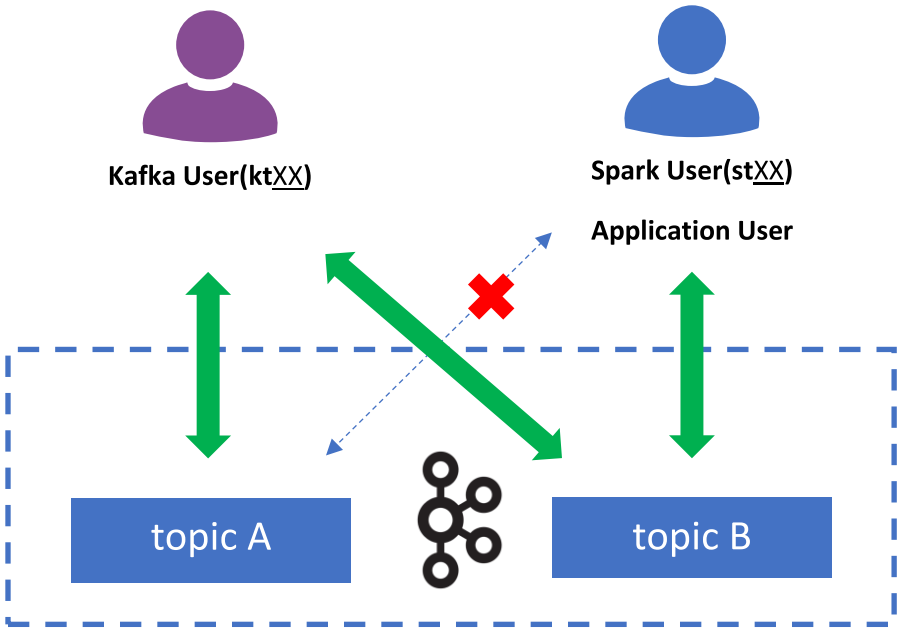
Team XX ==

Kafka User Instructions

Overview:

This Lab walks through the setup of a typical streaming data pattern in Azure. Data is ingested in Apache Kafka on HDInsight into specific topics. This data in Kafka is then read by a Spark streaming job. What makes this pattern more relevant to enterprises is that both clusters have Enterprise Security Package enabled: hence access to Kafka topics can be gated by Apache Ranger policies.

Imagine a company where …



In this document, you will go through the setup of the Kafka cluster, including VNET and security. You will access the Ranger portal for the cluster to setup custom security policies for topics. Then you will create 2 topics, topicA and topicB and send test data to those topics.

Your Spark counterpart also has a document outlining steps for them. You may have to collaborate to get the streaming job running end to end.

Your counterpart will consume from either topic but will succeed or fail based on the policies you set for them.

Winter Ready 2019 Technical Lab
Level: 300
Contacts: omidm, dhgoel

Team XX ==

Kafka User Instructions

Pre-requisites:

- AAD-DS is already setup. The lab instructor is the AAD admin and can provide access when needed. Think of the instructor as the AD team which is typically different from the big data team in enterprises. The AD team has already created your domain credentials which you will use later in this lab.
- An ESP Spark cluster deployed in a different VNET: Your teammate in the lab will be creating and preparing this VNET and cluster. When needed you can coordinate with him.

Common Info you may need:

- AAD-DS VNET: **AD-VNET**
- AAD-DS Resource Group: **AADDS-RG**
- AAD-DS domain name: **HDIESP.onmicrosoft.com**
- AAD-DS DNS Servers IPs: **10.0.0.5,10.0.0.4**
- WASB account name: **ready2019wasb**
- WASB RG: **Ready2019RG**
- Managed Identity: **Ready2019MI** (has access to AAD-DS, AKV key)
- MSI Resource Group: **Ready2019RG**
- Resource Group: Team**XX**RG
- Login to the Azure Portal with the following domain credentials
  - Credential: Domain UPN: kt**XX**@HDIESP.onmicrosoft.com , Pass: Ask Instructors.
- Replace **XX** in the rest of this doc with you team number at the top of this page
- Kafka_Disk_Encryption_Key_URL: https://ready2019akv.vault.azure.net/keys/Team01KafkaKey/751546d0694e494682c50f56360d9722

Instructions:
Login with your domain username in the common section above to portal.azure.com
We will use ARM templates to deploy resources needed in this Lab. However, portal instructions are also provided for the curious.

You can find the required templates on GitHub, use the "Deploy to Azure" when needed:
https://github.com/tylerfox/HDInsight-ESP-Kafka-Spark

Template Steps

Step 1 – Create a VNET & Peer to AD VNET

1. Go to GitHub link above and click on "template_vnet.json" Deploy to Azure button or click here:

2. Resource Group: Team**XX**RG
3. VNET name: KT**XX**VNET

Winter Ready 2019 Technical Lab
Level: 300
Contacts: omidm, dhgoel

Team XX ==

Kafka User Instructions

4. Your IP range depends on your team number (XX). Let Y = (2 *XX), then your VNET IP range is 10.Y.0.0/16 and Subnet IP range is 10.Y.0.0/24. For e.g., if your team number is 03, VNET IP range would be 10.6.0.0/16.
5. AADDS_DNS_Server_IPs: from common info (10.0.0.5,10.0.0.4)
6. AADDS_Resource_Group_Name: from common info (AADDS-RG)
7. AD_VNET_NAME: from common info (AD-VNET)
8. Hit "Purchase" to start the deployment.
9. This deployment creates a VNET/Subnet with the specified IP range, peers it to the AAD-DS VNET and sets custom DNS server settings.

Step 2 – Create an HDInsight Kafka Cluster

1. Go to GitHub link above and click on "template_kafka.json" Deploy to Azure button:

2. Resource Group: Team**XX**RG
3. Name: KT**XX**Cluster
4. Admin_SSH_username: sshuser
5. Cluster_Password: Pick a password and don't forget it
6. Storage Account Name: from common info (ready2019wasb)
7. Storage Resource Group: from common info (Ready2019RG)
8. Kafka VNET name: from previous section (KT**XX**VNET)
9. AAD-DS resource group: from common info (AADDS-RG)
10. AAD-DS domain name: **HDIESP.onmicrosoft.com**
11. Cluster Admin AAD account: kt**XX**@HDIESP.onmicrosoft.com
12. Cluster access group name: Team**XX**AADGroup
13. Managed Identity RG and name: from common info
14. Kafka_Disk_Encryption_Key_URL: from common info

Step 3 – Peer to Spark VNET:

1. Go to GitHub link above and click on "template_peer.json" Deploy to Azure button.

2. Before you proceed check with your teammate or look in your resource group to make sure that Spark VNET with the name ST**XX**VNET is already created.
3. My VNET: KT**XX**VNET
4. Remote VNET: ST**XX**VNET

Important: For the purposes of this lab, we have pre-created a cluster for you. For the remaining steps you can switch over to using this cluster. The resource group for your cluster is TeamXXARG, and the

Winter Ready 2019 Technical Lab
Level: 300
Contacts: omidm, dhgoel

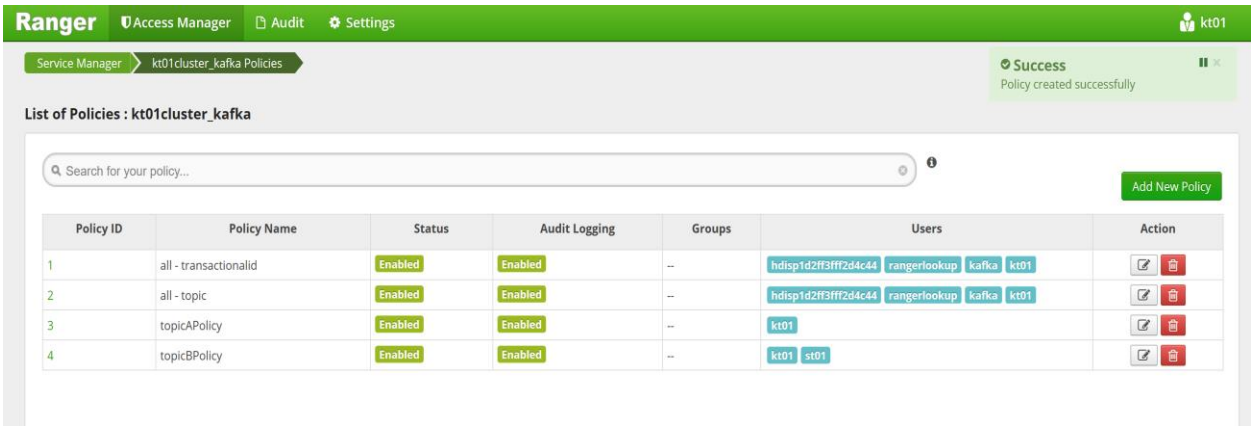
Team XX ==

Kafka User Instructions

name of your cluster is KTXXACluster (instead of KTXXCluster that you just created), with XX being your team number. The steps below use the pre-created cluster name. If you want to use the cluster you just created, remove the 'A' in the cluster name.

Step 4 – Creating Policies with Ranger for Kafka topics

1. Go to https://kt**XXA**cluster.azurehdinsight.net/Ranger/ - (Note: the trailing '/' at the end of the URL is required). Login with:
    - User: kt**XX**
    - Password: Provided by Instructors
2. Create policies to give Kafka User (kt**XX**) access to topicA
    - On the Ranger "Access Manager" Dashboard, click on the cluster name under Kafka
    - Click on "Add New Policy" on the top right
        - PolicyName: topicApolicy
        - Topic: topicA
        - Allow Conditions:
            - Select User: kt**XX**
            - Permissions: Consume, Create, Delete, Describe, Publish
    - Click on "Add"
3. Similarly, create a policy giving both kt**XX** and st**XX** access to topicB



NOTE: For the following shell commands, you can use whatever SSH client you prefer (Azure CloudShell, PuTTy, MobaXterm etc.)

Step 5 – Download the jq utility on the cluster

1. SSH into the cluster as sshuser to install jq utility
    - ssh sshuser@KT**XXA**Cluster-ssh.azurehdinsight.net
        - Login with the Cluster_Password you specified in the template when creating cluster
    - sudo apt -y install jq
2. You can log out

Step 6 – Creating Topics on Kafka cluster

1. SSH to the Kafka cluster using the domain username:
    - ssh kt**XX**@KT**XXA**Cluster-ssh.azurehdinsight.net
        - Login with the domain password provided by instructors

Winter Ready 2019 Technical Lab
Level: 300
Contacts: omidm, dhgoel

Team XX ==

Kafka User Instructions

2. Download the kafka-producer-consumer-esp.jar file from Apache Kafka domain-joined producer consumer examples:
    - wget https://github.com/Azure-Samples/hdinsight-kafka-java-get-started/raw/master/Prebuilt-Jars/kafka-producer-consumer-esp.jar
3. Run the following commands to create topics
    - export USER=kt**XX**@HDIESP.onmicrosoft.com
    - export CLUSTERNAME=kt**xxa**cluster
        - NOTE: clustername is in lowercase (numbers are ok)!!
    - export KAFKABROKERS=`curl -sS -u $USER -G https://$CLUSTERNAME.azurehdinsight.net/api/v1/clusters/$CLUSTERNAME/services/KAFKA/components/KAFKA_BROKER | jq -r '["\(.host_components[].HostRoles.host_name):9092"] | join(",")' | cut -d',' -f1,2`;
        - You will have to provide the Kafka user domain password provided by instructors
    - echo $KAFKABROKERS
        - It should look something like below:

```
kt01@hn0-kt01cl:~$ echo $KAFKABROKERS
wn0-kt01cl.hdiesp.onmicrosoft.com:9092,wn1-kt01cl.hdiesp.onmicrosoft.com:9092
kt01@hn0-kt01cl:~$
```

    - java -jar -Djava.security.auth.login.config=/usr/hdp/current/kafka-broker/config/kafka_client_jaas.conf kafka-producer-consumer-esp.jar create **topicA** $KAFKABROKERS
    - java -jar -Djava.security.auth.login.config=/usr/hdp/current/kafka-broker/config/kafka_client_jaas.conf kafka-producer-consumer-esp.jar create **topicB** $KAFKABROKERS

Step 7 – Send sample data to Kafka topic

1. Continue previous session as domain User A (kt**XX**) into Kafka cluster
2. Send data to topicA and topicB
    - java -jar -Djava.security.auth.login.config=/usr/hdp/current/kafka-broker/config/kafka_client_jaas.conf kafka-producer-consumer-esp.jar producer **topicA** $KAFKABROKERS
    - java -jar -Djava.security.auth.login.config=/usr/hdp/current/kafka-broker/config/kafka_client_jaas.conf kafka-producer-consumer-esp.jar producer **topicB** $KAFKABROKERS

You can work with your teammate to consume from these topics in the spark cluster. You can repeat sending data to these topics as needed and see the results on the spark cluster.

Winter Ready 2019 Technical Lab
Level: 300
Contacts: omidm, dhgoel

Team XX ==

Kafka User Instructions

Appendix

Portal Steps for VNET and cluster setup

Step 1 – Create a VNET & Peer

1a. Create a VNET

- VNET name: KT**XX**VNET
- Resource Group: TeamXXRG
- Location: East US 2
- All other settings are default

1b. Peer VNETs and set custom DNS server

- Create Peering in KT**XX**VNET
    1. Name: Peer2ADVNET
    2. Virtual network: Select the AAD-DS VNET from common info
    3. Configuration: "Enabled" for "Allow virtual network access"
    4. Configuration: Select "Allow forwarded Traffic"
- Create Peering in AD-VNET
    1. Name: Peer2KT**XX**VNET
    2. Virtual network: KT**XX**VNET
    3. Configuration: "Enabled" for "Allow virtual network access"
    4. Configuration: Select "Allow forwarded Traffic"
- Confirm that the Peering status says "Connected"
- Similarly, Peer to ST**XX**VNET (User B will create the ST**XX**VNET in their instructions)
- Set custom DNS server - provide IP address of AADDS
    1. In KT**XX**VNET, select DNS Servers from the navigation blade
    2. Select Custom and provide from common info

Step 2 - Create an HDInsight Kafka Cluster

- Name: KT**XX**cluster
- Cluster Type: Kafka
- Version: Kafka 1.1.0 on HDI 3.6
- Remember the sshuser and admin password that you choose.
- Resource Group: Team**XX**RG
- Location: East US 2
- Enterprise Security Package: Select "Enabled"
- Admin user: kt**XX**
- Cluster users: kt**XX** and st**XX**; OR you can specify AAD Group: team**XX**grp
- Virtual Network: KT**XX**VNET
- Identity MSI: **Ready2019MI**
- Storage Azure Storage: ready2019wasb
- Choose 3 worker nodes and 1 disk per node with the default option for VM sizes.