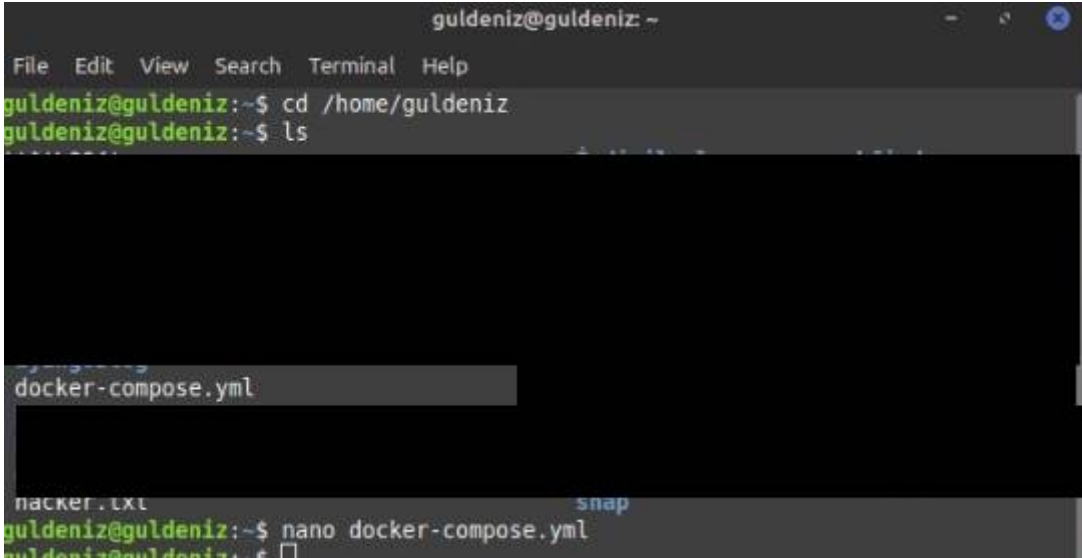


Docker ile Kurulum

Docker compose dosyasını kullanarak TheHive ve Cortex'i çalıştırın:



```
guldeniz@guldeniz: ~  
File Edit View Search Terminal Help  
guldeniz@guldeniz:~$ cd /home/guldeniz  
guldeniz@guldeniz:~$ ls  
...  
docker-compose.yml  
...  
guldeniz@guldeniz:~$ nano docker-compose.yml  
guldeniz@guldeniz:~$
```

Compose dosyasına aşağıdaki kodu ekleyin.

```
version: "3"  
services:  
  thehive:  
    image: strangebee/thehive:5.2  
    depends_on:  
      - cassandra  
      - elasticsearch  
      - minio  
      - cortex  
    mem_limit: 1500m  
    ports:  
      - "9000:9000"  
    environment:  
      - JVM_OPTS="-Xms1024M -Xmx1024M"  
    command:  
      - --secret  
      - "mySecretForTheHive"  
      - "--cql-hostnames"  
      - "cassandra"
```

- "--index-backend"
- "elasticsearch"
- "--es-hostnames"
- "elasticsearch"
- "--s3-endpoint"
- "http://minio:9000"
- "--s3-access-key"
- "minioadmin"
- "--s3-secret-key"
- "minioadmin"
- "--s3-bucket"
- "thehive"
- "--s3-use-path-access-style"
- "--cortex-hostnames"
- "cortex"
- "--cortex-keys"

put cortex api key once cortex is bootstrapped

- "<cortex_api_key>"

cassandra:

image: 'cassandra:4'

mem_limit: 1600m

ports:

- "9042:9042"

environment:

- MAX_HEAP_SIZE=1024M
- HEAP_NEWSIZE=1024M
- CASSANDRA_CLUSTER_NAME=TheHive

volumes:

- cassandradata:/var/lib/cassandra

restart: on-failure

elasticsearch:

image: docker.elastic.co/elasticsearch/elasticsearch:7.17.12

mem_limit: 1500m

ports:

- "9200:9200"

environment:

- discovery.type=single-node
- xpack.security.enabled=false

volumes:

- elasticsearchdata:/usr/share/elasticsearch/data

minio:

image: quay.io/minio/minio

mem_limit: 512m

command: ["minio", "server", "/data", "--console-address", ":9090"]

environment:

- MINIO_ROOT_USER=minioadmin
- MINIO_ROOT_PASSWORD=minioadmin

ports:

- "9090:9090"

volumes:

- "miniodata:/data"

cortex:

image: thehiveproject/cortex:3.1.7

depends_on:

- elasticsearch

environment:

- job_directory=/tmp/cortex-jobs

volumes:

- /var/run/docker.sock:/var/run/docker.sock
- /tmp/cortex-jobs:/tmp/cortex-jobs

ports:

- "9001:9001"

volumes:

miniodata:

cassandradata:

elasticsearchdata:

```

guiludmz1:var/lib/docker# docker-compose up
Starting docker_minio_1 ... done
Starting docker_elasticsearch_1 ... done
Starting docker_cassandra_1 ... done
Starting docker_cortex_1 ... done
Starting docker_thelive_1 ... done
Attaching to docker_minio_1, docker_elasticsearch_1, docker_cassandra_1, docker_cortex_1, docker_thelive_1
[0.000] 64 bit Server VM warning: Option UseCompressedOops has been deprecated in version 9.0 and will likely be removed in a future release.
cassandra_1 [0.005] [warning] [gc.erpo] NewSize was set larger than initial heap size, will use initial heap size.
cassandra_1 [0.005] [warning] [gc.erpo] MaxNewSize (1048576k) is equal to or greater than the entire heap (1048576k). A new max generation size of 1048512k will be used.
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/columns/Serializer.deserializeLargeSubset(Lorg/apache/cassandra/io/util/ByteArrayInputPlus;Lorg/apache/cassandra/db/columns;I)Lorg/apache/cassandra/db/columns;
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/columns/Serializer.serializeLargeSubset(Ljava/util/Collection;Lorg/apache/cassandra/db/columns;I)Lorg/apache/cassandra/io/util/ByteArrayOutputPlus;I
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/columns/Serializer.serializeLargeSubset(Ljava/util/Collection;Lorg/apache/cassandra/db/columns;I)
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/commitlog/AbstractCommitLogSegmentManager.advanceAllocatingFrom(Lorg/apache/cassandra/db/commitlog/CommitLogSegment;I
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/transform/BaseIterator.tryGetMoreContents(I)
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/transform/StoppingTransformation.stop()V
cassandra_1 CompileCommand: dontinline org/apache/cassandra/db/transform/StoppingTransformation.stopPartition(I)
cassandra_1 CompileCommand: dontinline org/apache/cassandra/io/util/BufferedDataOutputStreamPlus.doFlush(I)I
cassandra_1 CompileCommand: dontinline org/apache/cassandra/io/util/BufferedDataOutputStreamPlus.writeSlow(I)I
cassandra_1 CompileCommand: dontinline org/apache/cassandra/io/util/RebufferingInputStream.readPrimitiveSlowly(I)
cassandra_1 CompileCommand: dontinline org/apache/cassandra/util/VMStabilityInspector.forHeapDump(Ljava/lang/OutOfMemoryError;I)
cassandra_1 CompileCommand: inline org/apache/cassandra/db/rows/UnfilteredSerializer.serializeRowBody(Lorg/apache/cassandra/db/rows/Row;Lorg/apache/cassandra/db/rows/SerializationHelper;Lorg/apache/cassandra/io/util/ByteArrayOutputPlus;I)V
cassandra_1 CompileCommand: inline org/apache/cassandra/io/util/Memory.checkBounds(I)I
cassandra_1 CompileCommand: inline org/apache/cassandra/io/util/SafeMemory.checkBounds(I)V
cassandra_1 CompileCommand: inline org/apache/cassandra/io/util/TrackedDataInputPlus.checkCanRead(I)I
cassandra_1 CompileCommand: inline org/apache/cassandra/net/FramedDecoderV1t8BHeader.decode(Ljava/util/Collection;Lorg/apache/cassandra/net/ShareableBytes;I)V
cassandra_1 CompileCommand: inline org/apache/cassandra/service/reads/repair/RowWriterMergeListener.applyToPartition(Ljava/util/function/Consumer;I)
cassandra_1 CompileCommand: inline org/apache/cassandra/util/AsymmetricOrdering.selectBoundary(Lorg/apache/cassandra/util/AsymmetricOrdering;O)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/AsymmetricOrdering.strictnessOf(Lorg/apache/cassandra/util/AsymmetricOrdering;O)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/BloomFilter.indexes(Lorg/apache/cassandra/util/Filter;Lorg/apache/cassandra/util/FilterKey;I)
cassandra_1 CompileCommand: inline org/apache/cassandra/util/BloomFilter.setIndexes(I)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/ByteBufferUtil.compare(Ljava/nio/ByteBuffer;I)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/ByteBufferUtil.compare(Ljava/nio/ByteBuffer;I)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/ByteBufferUtil.compareUnsigned(Ljava/nio/ByteBuffer;Ljava/nio/ByteBuffer;I)
cassandra_1 CompileCommand: inline org/apache/cassandra/util/FastByteOperations.unsafeAofOperations.compareTo(Ljava/lang/Object;J)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/FastByteOperations.unsafeAofOperations.compareTo(Ljava/lang/Object;J)I
cassandra_1 CompileCommand: inline org/apache/cassandra/util/FastByteOperations.unsafeAofOperations.compare(Ljava/nio/ByteBuffer;Ljava/nio/ByteBuffer;I)
cassandra_1 CompileCommand: inline org/apache/cassandra/util/FastByteOperations.unsafeAofOperations.compare(Ljava/nio/ByteBuffer;Ljava/nio/ByteBuffer;I)
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/WaitableAsyncWaitable.await(Ljava/util/concurrent/atomic/AtomicReferenceFieldUpdater;Ljava/util/function/Predicate;Lorg/apache/cassandra/util/concurrent/Awaitable;Lorg/apache/cassandra/util/concurrent/Waitable;
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/WaitableAsyncWaitable.awaitUntil(Ljava/util/concurrent/atomic/AtomicReferenceFieldUpdater;Ljava/util/function/Predicate;Lorg/apache/cassandra/util/concurrent/Waitable;J)Z
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/WaitableAsyncWaitable.register(Ljava/util/concurrent/atomic/AtomicReferenceFieldUpdater;Ljava/util/function/Predicate;Lorg/apache/cassandra/util/concurrent/Awaitable;Lorg/apache/cassandra/util/concurrent/WaitQueue;Signal;
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/WaitableAsyncWaitable.signalAll(Ljava/util/concurrent/atomic/AtomicReferenceFieldUpdater;Lorg/apache/cassandra/util/concurrent/Awaitable;I)V
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/IntrusiveStack.push(Ljava/util/concurrent/atomic/AtomicReferenceFieldUpdater;Ljava/lang/Object;Lorg/apache/cassandra/util/concurrent/IntrusiveStack;Lorg/apache/cassandra/util/concurrent/IntrusiveStack;
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/IntrusiveStack.push(Ljava/util/function/Function;Lorg/apache/cassandra/util/concurrent/IntrusiveStack;Setter;Ljava/lang/Object;Lorg/apache/cassandra/util/concurrent/IntrusiveStack;Lorg/apache/cassandra/util/concurrent/IntrusiveStack;
cassandra_1 CompileCommand: inline org/apache/cassandra/util/concurrent/ListenerList.push(Ljava/util/concurrent/atomic/AtomicReferenceFieldUpdater;Ljava/lang/Object;Lorg/apache/cassandra/util/concurrent/ListenerList;

```

Oturum açma sayfasını görmek için [<http://localhost:9000>] adresinden bağlanın.

Kimlik bilgileri :

Login	admin@thehive.local
Password	secret

MinIO'da "thehive" kovalarını oluşturmamayı unutmayın. Varsayılan kimlik bilgilerini ve gizli dizileri değiştirmeniz önemle tavsiye edilir.

Kendi Yapılandırma Dosyanızı Kullanarak :

Giriş noktası argümanları, bir uygulama.conf dosyasının konteynırda oluşturulması için kullanılır.

Özel bir yapılandırma dosyası da sağlanabilir:

thehive:

image: strangebee/thehive:<version>

depends on:

- cassandra
- elasticsearch
- minio
- cortex

mem limit: 1500m

ports:

- "9000:9000"

```
environment:
  - JVM_OPTS="-Xms1024M -Xmx1024M"
volumes:
  - <host_conf_folder>:/data/conf
command:
  - --no-config
  - --config-file
  - /data/conf/application.conf
```

...

```
docker run --rm -p 9000:9000 -v <host_conf_folder>:/data/conf strangebee/thehive:<version> --no-config --
config-file /data/conf/application.conf
```

<host_conf_folder> klasörünün bir application.conf dosyası içermesi gerekmektedir.

--no-config, giriş noktasına herhangi bir yapılandırma dosyası oluşturulmamasını söylemek için kullanılır. Aksi takdirde, giriş noktası varsayılan bir yapılandırma oluşturacak ve bu dosya sizin dosyanızla birleştirilecektir.

Komut Satırı Argümanları Kullanarak

Veri depolama için Cassandra ve Elasticsearch ile TheHive'ı çalıştırmanızı ve dosya depolama için minio'yu kullanmanızı öneririz. Örnek olarak, örneklerinizin ana bilgisayar adlarını argümanlar aracılığıyla iletebilirsiniz:

```
docker run --rm -p 9000:9000 strangebee/thehive:<version> \
  --secret <secret>
  --cql-hostnames <cqlhost1>,<cqlhost2>,...
  --cql-username <cqlusername>
  --cql-password <cqlusername>
  --index-backend elasticsearch
  --es-hostnames <eshost1>,<eshost2>,...
  --s3-endpoint <minio_endpoint>
  --s3-access-key <minio_access_key>
  --s3-secret-key <minio_secret_key>
```

Bu, docker konteynerinizi harici cassandra ve elasticsearch düğümlerine bağlayacaktır. Veri dosyaları minio üzerinde saklanacaktır. Konteyner, TheHive'ı 9000 numaralı bağlantı noktasında gösterir.

Tüm Seçenekler

Docker giriş noktası tarafından desteklenen tüm seçenekleri -h ile alabilirsiniz:

```
docker run --rm strangebee/thehive:<version> -h
```

Available options:

--config-file <file>	Yapılandırma dosyasının yolunu belirtir.
--no-config	TheHive'in gizli diziler ve Elasticsearch ayarları eklemek de dahil olmak üzere kendini yapılandırmaya çalışmasını engeller.
--no-config-secret	Yapılandırmada rastgele oluşturulmuş bir gizli anahtar eklenmesini hariç tutar.
--secret <secret>	Oturumları güvence altına almak için kullanılan gizli anahtar ayarlar.
--show-secret	Oluşturulan gizli anahtar görüntüler.
--no-config-db	Veritabanının otomatik olarak yapılandırılmasını sağlar.
--cql-hostnames <host>,<host>,...	Cassandra örneklerini bulmak için bu ana bilgisayar adlarını çözümler.
--cql-username <username>	Cassandra veritabanı için kullanıcı adını belirtir
--cql-password <password>	Cassandra veritabanı için parolayı belirtir.
--no-cql-wait	Cassandra'nın kullanılabilir olmasını beklemeyi atlar.
--bdb-directory <path>	Cassandra kullanılmıyorsa yerel veritabanının konumunu tanımlar (varsayılan: /data/db).
--index-backend	Dizin için kullanılacak arka ucu belirtir, 'lucene' veya 'elasticsearch' (varsayılan: lucene).
--es-hostnames	Dizin için kullanılan Elasticsearch örneklerini belirtir
--es-index	Kullanılacak Elasticsearch dizin adını belirtir (varsayılan: thehive).
--no-config-storage	Depolamanın otomatik yapılandırılmasını devre dışı bırakır.
--storage-directory <path>	S3 kullanılmıyorsa yerel depolama alanının konumunu belirtir (varsayılan: /data/files.).
--s3-endpoint <endpoint>	AWS S3 için 's3.amazonaws.com' ile S3'ün veya kullanılıyorsa diğer nesne depolamanın uç noktasını belirtir
--s3-region <region>	S3 bölgesini belirtir, MinIO için isteğe bağlıdır.
--s3-bucket <bucket>	Kullanılacak kovanın adını belirtir (varsayılan: thehive), bu kovanın zaten var olması gerekir.
--s3-access-key <key>	S3 erişim anahtarını belirtir (S3 için gereklidir).
--s3-secret-key <key>	S3 gizli anahtarını belirtir (S3 için gereklidir).
--s3-use-path-access-style	MinIO veya başka bir AWS dışı S3 sağlayıcısı kullanılıyorsa bu bayrağı ayarlar, varsayılan olarak sanal ana bilgisayar stili kullanılır.
--no-config-cortex	Cortex yapılandırmasını hariç tutar.
--cortex-proto <proto>	Cortex'e bağlanmak için protokolü tanımlar (varsayılan: http).
--cortex-port <port>	Cortex'e bağlanmak için portu tanımlar (varsayılan: 9001).
--cortex-hostnames <host>,<host>,...	Cortex örneklerini bulmak için bu ana bilgisayar adlarını

çözümler.

--cortex-keys <key>,<key>,... | Cortex anahtarlarını tanımlar.

--kubernetes | Diğer düğümlere katılmak için Kubernetes API'sini kullanır.

• `--kubernetes-pod-label-selector <selector>`: Uygulamayı çalıştıran diğer podları seçmek için kullanılacak seçiciyi belirtir (varsayılan app=thehive).

`--cluster-min-nodes-count <count>` | Bir küme oluşturmak için minimum düğüm sayısını belirtir (varsayılan 1'dir).

`migrate <param> <param> ...` | Geçiş aracını çalıştırır.

`cloner <param> <param> ...` | Klonlama aracını çalıştırır.

Kubernetes'te Kullanım

Bir konteyner, Docker'ın izin verdiği kadar çok daha fazla bellek kullanıyor. Konteyner için daha fazla bellek izni vermek için `mem_limit` parametresini artırabilirsiniz. JVM tabanlı uygulamalar için (TheHive, Cassandra, Elasticsearch gibi) JVM parametrelerini ayarlayarak kullanılacak maksimum heap boyutunu belirleyebilirsiniz. TheHive için `JVM_OPTS` ortam değişkenini kullanabilirsiniz: `JVM_OPTS="-Xms1024M -Xmx1024M"`

Revision #5

Created 8 April 2024 10:38:39 by Güldeniz Akca

Updated 2 May 2024 10:56:08 by Güldeniz Akca