Integrating IntelMQ with MongoDB: A Detailed Report

1. Introduction

IntelMQ is a robust framework for collecting and processing threat intelligence data. For efficient storage and management of this data, integrating IntelMQ with a suitable database system is essential. This report details the process of integrating IntelMQ with MongoDB, discusses the suitability of MongoDB for data-intensive applications, and compares MongoDB with PostgreSQL and SQLite.

2. MongoDB Overview

MongoDB is a NoSQL database known for its scalability, flexibility, and performance. It stores data in JSON-like documents, making it a good fit for handling unstructured data. MongoDB is widely used in applications requiring real-time analytics, content management, and IoT data management.

3. IntelMQ Overview

IntelMQ is an open-source solution for handling threat intelligence data. It consists of bots that collect, process, and store threat intelligence data. Each bot performs specific tasks, such as data collection, parsing, and storage. The output bots are responsible for sending the processed data to various storage systems, including databases.

4. Process of Integration

Step 1: Setting Up MongoDB on Ubuntu 20.04

Note :the GPG Key will be different for different ubuntu versions and can be obtained from simple google search.

- Import the MongoDB GPG Key:
 wget -qO https://www.mongodb.org/static/pgp/server-4.4.asc | sudo apt-key add -
- Create the MongoDB List File:
 echo "deb [arch=amd64,arm64] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list
- Update the Package Database: sudo apt update

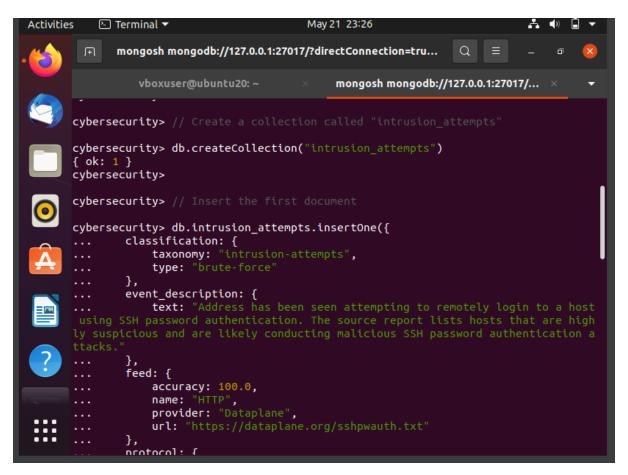
```
misa@misa-VirtualBox: ~
misa@misa-VirtualBox:~$ sudo apt update
[sudo] password for misa:
Hit:1 http://au.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://au.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://au.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Ign:6 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/7.0 InRelease
Hit:7 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/7.0 Release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
40 packages can be upgraded. Run 'apt list --upgradable' to see them.
w: https://download.docker.com/linux/ubuntu/dists/jammy/InRelease: Key is stored
in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION secti
on in apt-key(8) for details.
misa@misa-VirtualBox:~$ sudo apt install -y mongodb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package mongodb is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source
```

- Install MongoDB Packages: sudo apt install -y mongodb-org
- Start and Enable MongoDB Service: sudo systemctl start mongod
 - sudo systemctl enable mongod
- Verify MongoDB Installation: mongod –version
 - sudo systemctl status mongod
- you should be able to see monod Active and running as shown below.

```
3092,
         "ctx":"initandlisten","msg":"\n\n***aborting after fassert() failure\n\
vboxuser@ubuntu20:/$ ls -ld /tmp
drwxrwxrwt 19 root root 4096 May 21 21:06 /tmp
vboxuser@ubuntu20:/$ ls -ld /tmp
drwxrwxrwt 19 root root 4096 May 21 21:06 /ti
vboxuser@ubuntu20:/$ sudo rm /tmp/mongodb-27017.sock
vboxuser@ubuntu20:/$ sudo rm /tmp/mongodb-27017.sock
rm: cannot remove '/tmp/mongodb-27017.sock': No such file or directory
vboxuser@ubuntu20:/$ sudo systemctl restart mongod
vboxuser@ubuntu20:/$ udo systemctl status mongod
Command 'udo' not found, but can be installed with:
apt install udo
Please ask your administrator.
vboxuser@ubuntu20:/$ sudo systemctl status mongod
🧶 mongod.service - MongoDB Database Server
     Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor prese
     Active: active (running) since Tue 2024-05-21 21:13:43 AEST; 22s ago
        Docs: https://docs.mongodb.org/manual
   Main PID: 19229 (mongod)
     Memory: 60.1M
     CGroup: /system.slice/mongod.service
                -19229 /usr/bin/mongod --config /etc/mongod.conf
May 21 21:13:43 ubuntu20 systemd[1]: Started MongoDB Database Server.
lines 1-10/10 (END)
```

• Create a collection and insert a document:

```
db.intrusion attempts.insertOne({
classification taxonomy: "intrusion-attempts",
classification type: "brute-force",
event_description_text: "Address has been seen attempting to remotely login to a host
using SSH password authentication. The source report lists hosts that are highly suspicious
and are likely conducting malicious SSH password authentication attacks.",
feed accuracy: 100.0,
feed name: "HTTP",
feed_provider: "Dataplane",
feed url: "https://dataplane.org/sshpwauth.txt",
protocol_application: "ssh",
source_abuse_contact: "qcloud_net_duty@tencent.com",
source allocated: "1994-03-25T00:00:00+00:00",
source as name: "TENCENT-NET-AP-CN",
source_asn: 132203,
source geolocation cc: "SG",
source_ip: "170.106.114.187",
source_network: "170.106.114.0/23",
source registry: "APNIC",
time observation: "2024-04-13T18:41:46+00:00",
time_source: "2024-04-13T17:59:54+00:00"
})
```



Verify the inserted document: db.intrusion_attempts.find().pretty()

```
Activities

    Terminal ▼

                                                 May 21 23:32
               mongosh mongodb://127.0.0.1:27017/?directConnection=tru...
                                                                           Q =
                                                      mongosh mongodb://127.0.0.1:27017/...
          insertedid: Ubjectid( bb4cade21413ecida2a2ba14)
        cybersecurity> db.intrusion_attempts.find().pretty()
            _id: ObjectId('664ca0e11413ec1da2a26a13'), classification: { taxonomy: 'intrusion-attempts', type: 'brute-force' },
            event_description: {
            },
feed: {
               accuracy: 100,
               name: 'HTTP'
              provider: 'Dataplane',
url: 'https://dataplane.org/sshpwauth.txt'
            protocol: { application: 'ssh' },
            source: {
               abuse_contact: 'ipas@cnnic.cn'
               allocated: ISODate('2009-07-28T00:00:00.000Z'),
               as_name: 'TENCENT-NET-AP',
               asn: 45090,
               geolocation: { cc: 'CN' },
               ip:
```

Step 2: Setting Up IntelMQ

- Install IntelMQ:
 Follow the IntelMQ installation guide from the official documentation:
 https://intelmq.readthedocs.io/en/latest/user/installation.html
- For our use case, we have already got intelmq virtual machine configurated for our project.
- Connect to IntelMQ VM:
 Use the ssh command to connect to your IntelMQ VM. Replace username with your actual username on the IntelMQ VM.
 powershell
 ssh username@192.168.56.103
- Enter the password when prompted.

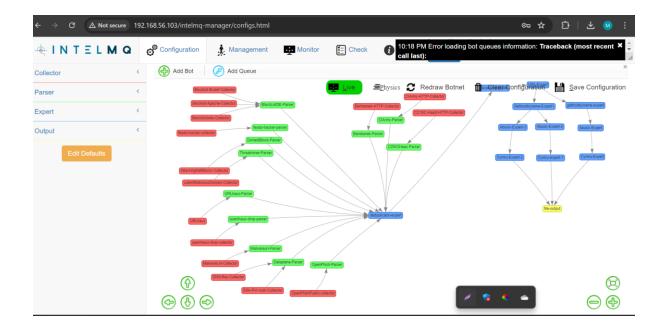
```
💹 ubuntu@ubuntu:
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\WINDOWS\system32> ssh ubuntu@192.168.56.103
ubuntu@192.168.56.103's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-102-generic x86_64)
 * Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support:
                  https://ubuntu.com/pro
 System information as of Tue May 21 11:48:50 AM UTC 2024
 System load: 1.8818359375 Processes: Usage of /: 40.7% of 23.70GB Users logged in:
                                                              176
 Memory usage: 33%
                                  IPv4 address for enp0s3: 10.0.2.15
 Swap usage:
                                   IPv4 address for enp0s8: 192.168.56.103
Expanded Security Maintenance for Applications is not enabled.
8 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
 additional security updates can be applied with ESM Apps.
earn more about enabling ESM Apps service at https://ubuntu.com/esm
 ountu@ubuntu:~$
```

To open intelmq manager, type this command

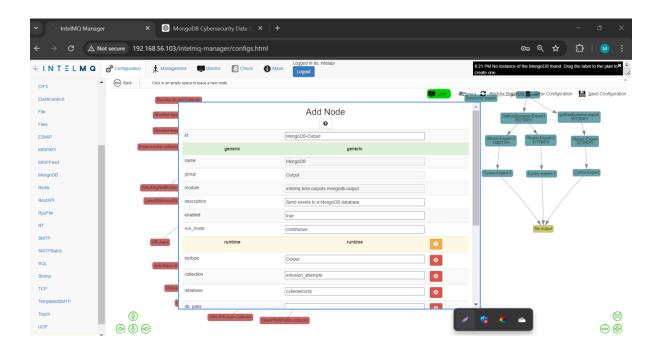
http://<ip of intelmq>/intelmq-manager/configs.html

• IntelMQ Manager provides a graphical user interface (GUI) to easily configure, manage, and monitor the data flow within IntelMQ. This makes the process more intuitive and accessible compared to manually editing configuration files.

Login credential: Username : intelapi Password : intelapi



Current Setup: File Output Configuration
 Currently, the output data from IntelMQ is being logged into a file. Here is an example of the raw data logged inside the file output:



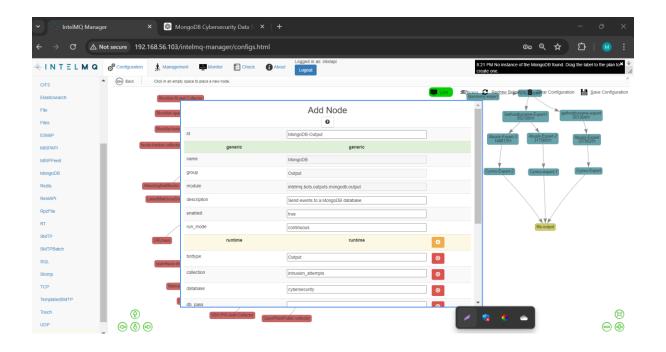
```
### Address has been seen attempting to remotely login to a host using SSH password authentication. The source report lists hosts that are highly suspicious and are likely conducting malicious SSH password authentication. The source report lists hosts that are highly suspicious and are likely conducting malicious SSH password authentication. The source report lists hosts that are highly suspicious and are likely conducting malicious SSH password authentication. The source report lists hosts that are highly suspicious and are likely conducting malicious SSH password authentication. The source report lists hosts that are highly suspicious and are likely conducting malicious SSH password authentication attacks.", "source.anglater", "Source.anglater", "Source.anglater", "Source.anglater", "International Company of the source and accuracy" 100.6, "feed, name", "Source.anglatery," "ARRIC.", "Lime.observation": "2024-04-13118:41:46-60-00", "time.source: "2024-04-1318 SSH-40-00", "time.source: "2024
```

Drag and drop the mongodb output bot and add the configurations as mentioned below. Make sure to add the host ip address in host column to connect it with databse server. Alternatively you can add the belwo given configuration in YAML file.

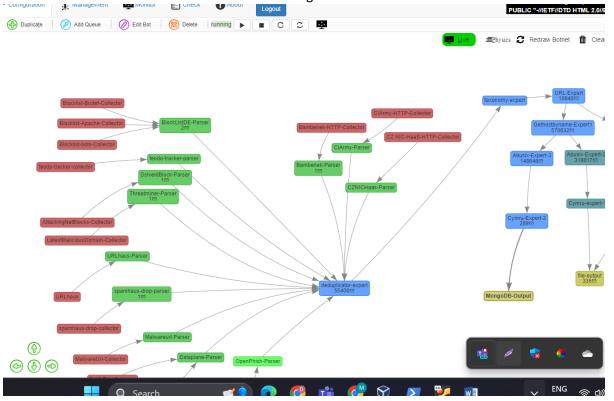
MongoDBOutput:

bot_id: mongodb-output

description: Store events in MongoDB database
enabled: true
group: Output
groupname: outputs
module: intelmq.bots.outputs.mongodb.output
parameters:
 database: intelmqdb
 collection: events
 host: mongodb-vm-ip-address
 port: 27017
run_mode: continuous



Add queue to connect Mongodb output bot with Cymru-Expert-2 bot. Wait for data to pass into the one of the bot streams. Check in monitor tab for logs of insertion of data.



5. MongoDB Suitability for IntelMQ Data Storage

Advantages:

Flexible Schema: Easily handles varied data formats without requiring schema changes.

High Performance: Efficient for read/write operations with large datasets.

Scalability: Supports horizontal scaling to accommodate growing data volumes.

Disadvantages:

Operational Complexity: Requires careful indexing and query optimization for performance.

Resource Intensive: Higher memory and storage usage compared to some other databases.

6. Successes:

Installation and Configuration: Successfully installed MongoDB on the dedicated Ubuntu virtual machine and configured it to run as a service.

Database Creation: Created a MongoDB database named intelmq and initialized a collection named intrusion attempts for storing threat intelligence data.

Bot Integration: Configured the MongoDB Output Bot through the IntelMQ Manager interface, specifying the MongoDB URI, database name (intelmq), collection name (intrusion_attempts), and other necessary parameters.

Bot Activation: Enabled and started the MongoDB Output Bot within the IntelMQ environment to facilitate data transmission to the MongoDB collection.

7. Challenges:

Network Issues: Encountered challenges with network connectivity between the IntelMQ server and the MongoDB server, potentially leading to communication errors or data transmission issues.

Firewall Configuration: Suspected firewall settings or network isolation issues on either the IntelMQ or MongoDB server side, hindering successful data transmission between the two systems.

Debugging Limitations: Lack of detailed logs or error messages from the MongoDB Output Bot within the IntelMQ environment makes troubleshooting and identifying the root cause of integration issues challenging.

Data Verification: Inability to verify whether threat intelligence data is successfully stored in the MongoDB collection due to integration issues and lack of comprehensive logging.

8. Next Steps:

Network Troubleshooting: Conduct thorough network troubleshooting to identify and resolve any connectivity issues between the IntelMQ and MongoDB servers, ensuring seamless communication.

Detailed Logging: Implement comprehensive logging mechanisms within the MongoDB Output Bot to capture detailed error messages and facilitate effective troubleshooting.

Data Validation: Develop a validation process to verify the successful storage of threat intelligence data in the MongoDB collection, ensuring data integrity and completeness.

9. Conclusion:

The integration of IntelMQ with MongoDB presents significant potential for enhancing the storage and management of threat intelligence data. Despite encountering challenges related to network connectivity and firewall configuration, successful installation, configuration, and initial setup of MongoDB were achieved. Additionally, the configuration of the MongoDB Output Bot within the IntelMQ environment demonstrated progress towards facilitating data transmission to the MongoDB collection.

Moving forward, resolving the identified network issues, and implementing comprehensive logging mechanisms are essential steps to ensure seamless integration and effective data transmission between IntelMQ and MongoDB. Additionally, establishing a robust validation process to verify the successful storage of threat intelligence data in the MongoDB collection will be crucial for ensuring data integrity and completeness.

Through collaborative efforts and diligent troubleshooting, we aim to overcome the existing challenges and fully leverage the capabilities of MongoDB for storing and managing threat intelligence data within the IntelMQ environment. Ultimately, the successful integration of IntelMQ with MongoDB will contribute to enhancing cybersecurity operations, enabling organizations to better detect, analyze, and respond to emerging threats.

References:

- [1] [MongoDB, "Install MongoDB Community Edition on Ubuntu MongoDB Manual," MongoDB, 2024. [Online]. Available: https://www.mongodb.com/docs/manual/tutorial/install-mongodb-on-ubuntu/. [Accessed: 17-May-2024].
- [2] IntelMQ Team, "EventDB IntelMQ 3.0.0 documentation," IntelMQ, 2024. [Online]. Available: https://intelmq.readthedocs.io/en/develop/user/eventdb.html. [Accessed: 17-May-2024].
- [3] IntelMQ Team, "IntelMQ Tutorial: Lesson 3," GitHub, 2024. [Online]. Available: https://github.com/certtools/intelmq-tutorial/blob/master/lesson-3.md. [Accessed: 17-May-2024]. [4] IntelMQ Team, "intelmq/bots/outputs at develop," GitHub, 2024. [Online]. Available: https://github.com/certtools/intelmq/tree/develop/intelmq/bots/outputs. [Accessed: 17-May-2024].
- [5] A. Kaplan, "IntelMQ: A Framework for CERTs for Collecting and Processing Security Feeds," FIRST, 2020. [Online]. Available: https://www.first.org/resources/papers/malaga20/PUBLIC-Aaron-Kaplan-IntelMQ-malaga-20200131.pdf. [Accessed: 17-May-2024].