STUDY OF INTRUSION DETECTION SYSTEMS BASED ON OPEN SOURCE BIG DATA ANALYSIS FOR THE EARLY DETECTION OF COMPUTER INCIDENTS

ANTONIO JOSE LOBO FIGUEROA
Agenda

- Problem
- Goals
- Concepts – Big data
- Incident factors in the implementation of an intrusion detection system based on open source big data analysis.
- Conclusions
PROBLEM

• The security operations centers (SOC) or the places in charge of guaranteeing the IT security of the institutions present problems to detect in a timely manner attacks or malicious activities.

Some problems:

➢ False alarms or false positives,

➢ The difficulties in the correlation of events due to the large amount of data.
Is it possible to monitor and correlate security events from different heterogeneous sources to detect new attack patterns using Big Data analysis techniques?
MAIN GOAL

Conduct a study of intrusion detection systems based on open source Big Data analysis that help in the detection of incidents of a cybernetic nature in the security operations centers (SOC).
Specific Goals

Describe the process of “Big Data analysis” in the detection of computer attacks.

Analyze open source Big Data systems for the analysis of computer security events.

Analyze the incident factors in the implementation of an intrusion detection system based on open source Big Data analysis for the early detection of computer incidents for security operations centers.
BIG DATA - Big Data Characteristics

- **Variety:**
  - Structured
  - Semi-structured
  - Hybrid

- **Velocity:**
  - Flows
  - Nearby Time
  - Real time

- **Volume:**
  - Petabytes
  - Records
  - Transactions
Big Data Infrastructure - Hadoop Ecosystem

- **Ambari**
  - is aimed at making Hadoop management simpler

- **HDFS**
  - It is a distributed file system

- **YARN**
  - Provides resource and data management in Hadoop

- **SPARK**
  - It is a fast and general engine for large-scale data processing

- **HBASE**
  - Distributed NoSQL database

- **HIVE**
  - High level language similar to SQL

- **PIG**
  - High level language to generate tasks

- **Zookeeper**
  - Distributed coordination service
Evolution of tools “systems” for intrusion detection

IDS  SIEM  Big Data Analysis
Challenges for the analysis of Big Data

- The source of the data.
- Privacy
- Secure Big Data storage
- Man-machine interaction
Systems for the analysis of big data in open code for the detection of computer incidents

- The Hunter
- PACKETPIG
- Hogzilla-IDS
- OPENSOC
- APACHE METRON
- CHUKWA
Incident factors in the implementation of an intrusion detection system based on open source big data analysis
Implementation of an intrusion detection system based on big data

• Apache metron was selected to install it in the Datacenter of the University of Santander, UDES.
Installation of Apache Metron - Apache Ambari

- Apache Ambari was used to install the Apache Metron system, which allows to install, manage and monitor in a simpler way through a graphic interface all the elements of the Hadoop cluster.
Installation of Apache Metron - Apache Ambari

• Summary of Apache Metron cluster nodes
Apache Metron - Kibana

• Apache Metron dashboard

Metron’s default dashboard, which is designed to validate Metron’s end-to-end operation with its default sensor set.
Apache Metron - Kibana

• Location of events (Kibana dashboard)

Location of the flows: Apache Metron has the function of locating (based on the IP Address) on a map the events recorded by the different sensors.
Problems in the implementation of apache metron

After installing the Apache Metron using Ambari, the elasticsearch package does not raise its service correctly, which creates a problem in the Kibana service and therefore in the Apache Metron.
Problems in the implementation of apache metron

The HDFS file system of the Apache Metron is 100% full, which causes the system to crash and the Kibana dashboard to stop working. Additionally, when the HDFS service fails, other services such as YARN and Kafka fail.
Problems in the implementation of apache metron

The Apache Metron system presents instability since cluster nodes fail for no apparent reason, increasing the level of maintenance of the platform since it has to review the variables of the packages (components) and adjust values without having documentation that helps support this activity.
Comparativo entre Apache Metron vs. OSSIM y Security Onion

<table>
<thead>
<tr>
<th>Sistema</th>
<th>Hardware</th>
<th>Instalación</th>
<th>Configuración</th>
<th>Mantenimiento</th>
<th>Documentación</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Metron</td>
<td>03 servidores (mínimo)</td>
<td>Complicada 480 minutos</td>
<td>Complicada 480 Minutos</td>
<td>Muy Complicado</td>
<td>Muy Poca No existen mucha documentación para ambientes de producción. Esta más enfocado a temas de investigación y prueba.</td>
</tr>
<tr>
<td></td>
<td>• Numero de Cores: 8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Memoria RAM: 16GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Disco duro: 1Tera</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OSSIM</td>
<td>01 servidor</td>
<td>Muy Fácil 30 minutos</td>
<td>Fácil 60 minutos</td>
<td>Normal</td>
<td>Bastante Existen muchos foros y blogs donde se registran problemas o preguntas comunes. Además tiene manuales actualizados.</td>
</tr>
<tr>
<td></td>
<td>• Numero de core: 8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Memoria RAM: 8GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Disco duro: 500GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security Onion</td>
<td>01 servidor</td>
<td>Fácil 60 minutos</td>
<td>Normal 120 minutos</td>
<td>Complicado</td>
<td>Normal Existen foros y blogs donde se registran problemas o preguntas comunes.</td>
</tr>
<tr>
<td></td>
<td>• Numero de core: 8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Memoria RAM: 8GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Disco duro: 500GB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CONCLUSIONS

• After reviewing and analyzing everything that has been done and researched, there are sufficient arguments to consider that the open source Big Data analysis platforms for the early detection of computer incidents, specifically the Apache Metron platform, is not ready to be implemented in production environments.

• Too many hardware resources are needed for its operation.

• Maintenance is a problem since the components of the platform are unconfigured for no apparent reason. Finally, there is not a community to support in case of problems on the platform.
CONCLUSIONS II

• The path of using Big Data analysis in the area of computer security is very promising, but unfortunately it is still in the research phase with many factors that make its deployment as a platform in a production environment very complicated and unfeasible.
¿Questions?
Thanks!!!!