Seguridad del dato en sistemas de Big Data

Pedro González Martínez

Área de Análisis de datos

Nombre Tutor/a de TF
Rafael Garcia Tomas
Profesor/a responsable de la asignatura
Andreu Pere Isern Deyà

Fecha Entrega
13/06/2023

Universitat Oberta
de Catalunya

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Seguridad en Sistemas de Big Data*
Nombre del autor:	*Pedro González Martínez*
Nombre del consultor/a:	*Rafael Garcia Tomas*
Nombre del PRA:	*Andreu Pere Isern Deyà*
Fecha de entrega (mm/aaaa):	*06/2023*
Titulación o programa:	Máster en Ciberseguridad y Privacidad
Área del Trabajo Final:	*Área de Análisis de datos*
Idioma del trabajo:	*Castellano*
Palabras clave	*Seguridad, Big, Data.*

Resumen del Trabajo

La analítica de datos y el almacenamiento masivo de los mismos con el objeto de sustraer información clave para el negocio está a la orden del día. Se calcula que alrededor de un 60% de las compañías del IBEX35 y en torno a un 22% del resto de empresas españolas usan tecnologías basadas en big data para analizar la ingente cantidad de datos que ofrece el ecosistema para optimizar la toma de decisiones, fundamentalmente, con un carácter estratégico.

En este nuevo paradigma, la seguridad de los entornos de big data se ha convertido, del mismo modo, en una de las grandes preocupaciones de los directores de seguridad y CISOs de las grandes compañías. A pesar del gran rédito que se puede conseguir con los entornos de analítica de datos y big data, los expertos en ciberseguridad también alertan del enorme impacto que pudiera suponer el robo, pérdida, filtrado e indisponibilidad de los datos recopilados, almacenados y procesados para las empresas que apoyan sus decisiones estratégicas y de negocio en el análisis de estos grandes volúmenes de datos.

Este trabajo fin de máster tiene por objetivo analizar los entornos big data y de analítica de datos actuales, identificar los riesgos para la seguridad de la información que existen, evaluarlos e identificar y diseñar medidas de seguridad que permitan mantener estos grandes volúmenes de datos protegidos durante todo su ciclo de vida y disponibles para una operativa continuada y sin interrupciones del entorno de analítica en aras de sacar el máximo rendimiento a los mismos.

Abstract

Data analytics and their massive storage in order to extract key information for the business is the order of the day. It is estimated that around 60% of the IBEX35 companies and around 22% of the rest of Spanish companies use technologies based on big data to analyze the huge amount of data offered by the ecosystem to optimize decision-making, fundamentally with a strategic nature.
In this new paradigm, the security of big data environments has likewise become one of the great concerns of security directors and CISOs of large companies. Despite the great revenue that can be achieved with data analytics and big data environments, cybersecurity experts also warn of the enormous impact that the theft, loss, filtering, and unavailability of the data collected, stored, and processed for companies could have. companies that support their strategic and business decisions in the analysis of these large volumes of data.
This master's thesis aims to analyze the current big data and data analytics environments, identify the risks to information security that exist, evaluate them and identify and design security measures that allow these large volumes of data to be kept protected during throughout their life cycle and available for continuous and uninterrupted operation of the analytics environment in order to get the most out of them.

Índice

1. Introducción 1
   1.1. Contexto y justificación del Trabajo 2
   1.2. Objetivos del Trabajo 2
   1.3. Impacto en sostenibilidad, ético-social y de diversidad 3
   1.4. Enfoque y método seguido 3
   1.5. Planificación del Trabajo 4
   1.6. Breve sumario de productos obtenidos 6
   1.7. Breve descripción de los otros capítulos de la memoria 6
2. Materiales y métodos 8
3. Resultados 9
   1. Plataformas y Riesgos comunes 9
   1.1 Plataformas 9
   1.2 Riesgos Comunes 10
   1.3 Medidas de seguridad 13
   2. Fase Ingesta 15
   2.1 Riesgos 15
   2.2 Medidas de seguridad 19
   3. Fase Procesado 23
   3.1 Riesgos 23
   3.2 Medidas de Seguridad 24
   3.3 Calidad del dato 27
   4. Fase Almacenamiento 29
   4.1 Riesgos 30
   4.2 Medidas de seguridad 32
   5. Fase Uso 36
   5.1 Compromiso privacidad/pérdida de información 38
   5.2 Riesgos 40
   5.3 Medidas de seguridad 42
   5.4 Modelos de privacidad 50
   6. Fase Eliminación 53
   6.1 Derecho de supresión 53
   6.2 Riesgos 54
   6.3 Medidas de seguridad 56
4. Conclusiones y trabajos futuros 58
5. Glosario 61
6. Bibliografía 62
7. Anexos 66
   Anexo 1: Kerberos 66
   Anexo 2: LDAP 67
   Anexo 3: ACL 68
   Anexo 4: RBAC 68
   Anexo 5: AES 69
   Anexo 6: SSL/TLS 69
   Anexo 7: DoD 5220.2-M 70

Lista de figuras

Figura 1: Ciclo de vida del dato. 4
Figura 2: Esquema ataque Man-in-the-middle. 18
Figura 3: Vault Tokenization. 47
Figura 4: Protocolo Kerberos. 67

1. Introducción

En la era de la información, las empresas generan y acumulan grandes cantidades de datos que, si son analizados y tratados de manera correcta, puede proporcionar grandes resultados e incluso ventajas competitivas.
Big Data es el término utilizado para describir el proceso de tratamiento y análisis de grandes volúmenes de datos, que pueden ser de diversas fuentes (sensores, redes sociales, sistemas transaccionales, aplicaciones móviles...) y naturalezas (textos, imágenes, vídeos...).
La capacidad de procesar grandes cantidades de datos en el menor tiempo posible es crucial para la competencia entre empresas ya que puede proporcionar información muy valiosa sobre patrones, tendencias, preferencias de los clientes, etc. El Big Data puede ayudar a las empresas a extraer información útil para el negocio y a hacerlo de manera más rápida y eficiente, reduciendo costes.
El Big Data es utilizado en una gran variedad de áreas y sectores, su elevada capacidad de procesado ofrece infinitas posibilidades ya que permite extraer información útil a partir de datos muy variados y con propósitos diversos.
Algunos de los usos más comunes incluyen:

- **Análisis de datos de clientes:** Las empresas utilizan Big Data para analizar los datos de sus clientes y obtener información valiosa sobre sus preferencias y comportamientos (canciones más escuchadas, productos más comprados, productos más descartados del carrito...). Esto permite ofrecer una experiencia personalizada y más adaptada al usuario.
- **Investigación científica:** Las entidades científicas utilizan Big Data para analizar grandes conjuntos de datos y obtener información valiosa sobre diversos temas como salud, clima, consumo energético., astronomía, física de partículas... Algunos ejemplos son: en astronomía, los grandes telescopios modernos generan volúmenes muy grandes de datos al observar galaxias lejanas, el análisis de estos datos ha permitido descubrir nuevos planetas, estrellas y galaxias; en física de partículas, los experimentos realizados en los aceleradores generan una gran cantidad de información que debe ser procesada, su análisis ha llevado al descubrimiento de nuevas partículas y la validación de teorías físicas.
- **Optimización de procesos empresariales:** El Big Data permite analizar los procesos empresariales para mejorar la eficiencia y la productividad. Una posibilidad es rastrear todos los estados y fases por las que pasa un producto desde su creación hasta la distribución y anotar los tiempos de cada fase, de esta manera podrían localizarse los cuellos de botella o analizar si hay épocas del año más críticas (subida de la producción en navidad, falta de personal en agosto por turnos de vacaciones...). Así la empresa podría tomar acciones correctivas para mejorar la productividad.
- **Análisis de riesgos de seguridad:** El Big Data se utiliza en el análisis de riesgos y la seguridad para identificar posibles amenazas. Los analistas pueden utilizarlo para identificar patrones sospechosos o tendencias que puedan desenvocar en una amenaza potencial. También permite el análsis de una gran cantidad de tráfico de datos en tiempo real y una

monitorización constante que puede detectar anomalías (incremento extraño del volumen de datos, incremento de peticiones de conexión sospechosas...).

- **Análsis de redes sociales:** Las empresas utilizan Big Data para analizar las redes sociales y obtener información sobre la opinión de clientes, sus necesidades, gustos y preferencias. Esto permite a las empresas elaborar perfiles y adaptar sus estrategias de marketing para mejorar la conexión con el cliente. Sería posible desde analizar los rangos de edades de 'likes' que tiene un anuncio publicitado en una red social desde estudiar el tiempo medio en que los usuarios se quedan observando el anuncio antes de pasarlo...

1.1. Contexto y justificación del Trabajo

Inicialmente, solo los gobiernos y grandes organizaciones tenían los medios suficientes para implantar sistemas de Big Data, no obstante, la llegada del Cloud y las tecnologías de virtualización han democratizado su alcance permitiendo el acceso al resto de empresas con soluciones más asequible.
El auge de las tecnologías de la información y la irrupción de la telefonía inteligente en la vida cotidiana han ocasionado un cambio de paradigma respecto a la seguridad y privacidad de los datos convirtiéndolos en una de las mayores preocupaciones de los directores de seguridad y CISOs de las grandes compañías.
La gran cantidad de datos generados, procesados y almacenados se ha convertido en un objetivo lucrativo para hackers. No es extraño la aparición de noticias sobre fugas de información de usuarios de aplicaciones pertenecientes a grandes compañías debido a un ataque informático.
Este es el motivo del que surge la necesidad del trabajo de fin de máster que tiene el objetivo de analizar los entornos de Big Data actuales, identificar los riesgos que existen para la seguridad de la información, evaluarlos y diseñar o escoger medidas para proteger los datos durante todo su ciclo de vida.
Al tratarse de una tecnología relativamente joven y en crecimiento no existe un conocimiento generalizado a la hora de securizar este tipo de entornos. Por ello este proyecto pretende divulgar los posibles riesgos que entraña la ingesta, almacenamiento y procesado de datos en sistemas de Big Data, medidas de seguridad y buenas prácticas para protegerlo.

1.2. Objetivos del Trabajo

El trabajo se centrará en alcanzar los siguientes objetivos:

- Certificar la seguridad de los datos y fiabilidad de los orígenes durante la ingesta de información.
- Asegurar el procesado de datos.
- Fijar medidas de almacenamiento seguro del dato.
- Realizar un uso ético y seguro de los datos almacenados.
- Proteger la privacidad de los datos en todo el ciclo de vida.
- Garantizar el borrado seguro de datos.

1.3. Impacto en sostenibilidad, ético-social y de diversidad

El 5 de junio se cumplirán diez años desde que las revelaciones de Edward Snowden sobre las prácticas de vigilancia por parte de agencias de inteligencia comenzaron a publicarse en la prensa.
Vivimos un momento histórico crucial en cuanto a la privacidad en Internet, es una realidad que ya está moldeando nuestro comportamiento digital y teniendo consecuencias graves.
Con un poco de esfuerzo, casi cualquier individuo, empresa o gobierno puede saber hoy más sobre nosotros que lo que ninguna agencia de inteligencia había podido averiguar sobre los individuos en el pasado debido a la cantidad de datos esparcidos por el ecosistema sin protección alguna.
Es importante recordar que la estructura de Internet que permite violaciones de privacidad es altamente lucrativa ya que nuestros datos están en venta.
El mal uso de la alta capacidad de procesado y análisis de Big Data y la falta de privacidad puede tener un impacto negativo en la sociedad. La falta de privacidad puede desembocar en una violación de los derechos humanos. Por ejemplo, la divulgación de información personal, financiera, médica o política puede desencadenar en acaso y discriminación por raza, género, orientación social...
Otro impacto negativo es la manipulación y la propagación de desinformación. La manipulación de los datos recogidos puede llevar a la mala toma de decisiones, afectando a la economía, política y sociedad. La propagación de desinformación puede tener grandes efectos adversos en la sociedad y la salud pública, como en el caso de la pandemia por COVID-19, donde la difusión de información sesgada o falsa ha llevado a la confusión sobre las decisiones tomadas por los gobiernos, desconfianza en las medidas sanitarias o desconocimiento de las restricciones del momento.
Este proyecto tendría un impacto positivo directo en la seguridad y privacidad de las personas y sus derechos humanos.
Mediante la protección de los datos en todo su ciclo de vida se mitigaría el daño que podría sufrir una persona si se produce una fuga de información sensible. Se eliminaría la posibilidad de propagar bulos e injurias sobre los datos de cualquier individuo y se evitarían persecuciones, acoso y discriminación sobre cualquier sujeto.

1.4. Enfoque y método seguido

Este proyecto está enfocado en el análisis y recopilación de las diferentes medidas de seguridad que permitan proteger las dimensiones de seguridad de estos grandes volúmenes de datos en todo su ciclo de vida: Ingesta, Procesado, Almacenamiento, Uso, Eliminación.
Esta estrategia permite definir un alcance del proyecto claro y delimitado estableciendo el inicio de las responsabilidades en la obtención de los datos - desde los diferentes orígenes posibles – y el fin en la eliminación.
Por otra parte, pone al dato en el foco del proyecto dotando al trabajo de la especialización y detalle que requiere.

Figura 1: Ciclo de vida del dato.

1.5. Planificación del Trabajo

Para el desarrollo del proyecto se considerarán las diferentes fases del ciclo de vida del dato como las tareas a planificar y desarrollar.
Para estandarizar la estructura de cada una de las fases se enfocarán de la siguiente manera:

Plataformas y riesgos comunes
Inicialmente se dará una visión sobre las diferentes plataformas de Big Data que ofrece el panorama tecnológico actual.
Haciendo un análisis a priori y a gran escala es posible destacar que habrá riesgos comunes en todas las fases, principalmente aquellos asociados al acceso. Este apartado inicial pretende recoger todos aquellos que puedan aparecer de manera transversal en las diferentes fases del ciclo de vida del dato para posteriormente analizar los específicos de cada etapa.

Ingesta

Introducción: Breve explicación del alcance de la fase de ingesta de datos de los diferentes orígenes.

Riesgos: Se analizarán riesgos más específicos como por ejemplo los relacionados con orígenes de datos maliciosos.

Medidas de Seguridad: Métodos para corroborar que los orígenes de datos son fiables.

Procesado

Introducción: Desarrollo sobre la fase de procesado de datos.

Riesgos: Aunque no sea el objetivo principal de este proyecto es necesario mencionar la importancia de la calidad del dato durante el procesado ya que este aspecto puede influir de manera directa en los formatos de los atributos y por consiguiente en la seguridad de los cifrados.

Medidas de seguridad: Medidas para que el procesado de datos sea seguro.

Almacenamiento

Introducción: Una vez procesados los datos son almacenados.

Riesgos: Unos mecanismos de seguridad no lo suficientemente robustos pueden provocar el acceso a la información almacenada y su fuga, modificación o eliminado.

Medidas de seguridad: El cifrado de datos en reposo mantendrá los datos ilegibles. El control de acceso basado en roles asegurará que los usuarios solo puedan realizar aquellas acciones para las que están autorizados.

Uso

Introducción: En los sistemas de Big Data se puede hacer un uso diverso y variado de los datos almacenados.

Riesgos: El riesgo de re-identificación de los datos personales puede suponer la pérdida del anonimato.

Medidas de seguridad: Será necesario implantar medidas que aseguren la privacidad y anonimidad de los datos.

Eliminación

Introducción: Se trata de la fase final del ciclo de vida. Cuando los datos dejan de ser necesarios deben ser eliminados de manera segura.

Riesgos: Si el proceso de borrado no es seguro un atacante podría acceder a los datos que todavía se encuentran en memoria.

Medidas de seguridad: Uso de sistemas de borrado seguro que eliminan o sobrescriben la información del file system.

Conclusiones
Para finalizar será necesario reflexionar sobre el proceso desarrollado, el alcance de objetivos y la planificación.

Trabajos futuros
Por último, se hará una propuesta de posibles trabajos futuros para desarrollar en una segunda fase del proyecto.

Cada una de las fases se repartirá entre las diferentes entregas:

- Plan de trabajo: 14/03/2023
- Entrega de seguimiento 1: 11/04/2023
- Entrega de seguimiento 2: 09/05/2023

- Memoria final: 13/06/2023
- Presentación vídeo: 20/06/2023
- Defensa del TFM: 30/06/2023

Que serán considerados los hitos principales y la piedra angular para estimar los tiempos y esfuerzos requeridos.
Se planificará una fecha de entrega con 5-7 días de antelación a la formal con tal de evitar posibles retrasos.

Diagrama_Gantt.xlsx

1.6. Breve sumario de productos obtenidos

A lo largo del desarrollo del proyecto y con el fin de cumplir con las entregas se obtendrán los siguientes productos:

- **Plan de trabajo:** Se trata de una versión inicial del trabajo en el que se define el alcance del proyecto, los objetivos que se pretenden alcanzar, las tareas para cumplirlos y una planificación para cumplir con las fechas de entrega.
- **Memoria_v2:** Después del plan de trabajo se desarrollarán las fases iniciales del ciclo de vida del dato. Abordando los riesgos y medidas a implementar para asegurar su seguridad.
- **Memoria_v3:** Una versión parcial que incluirá las fases intermedias del ciclo de vida del dato.
- **Memoria final:** La entrega final contendrá toda la información anterior completando el ciclo de vida del dato. Se incluirán las conclusiones analizando de manera crítica los resultados obtenidos respecto a los objetivos iniciales y posibles trabajos futuros.
- **Presentación en diapositivas:** A partir de la memoria final se creará una presentación mediante diapositivas resumiendo el contenido del trabajo de manera visual, clara y concisa.
- **Presentación en vídeo:** En base a la presentación en diapositivas el autor grabará un vídeo explicando el contenido del proyecto.

1.7. Breve descripción de los otros capítulos de la memoria

El grueso del trabajo dedicará un capítulo a cada fase del ciclo de vida del dato:

Plataformas y riesgos comunes
Haciendo un análisis genalizado se puede observar que hay riesgos que pueden amenazar a todas las fases de manera transversal, sobretodo aquellos relacionados con el acceso. En este capítulo se analizarán las diferentes plataformas de big data y los riesgos comunes a todas las fases.

Ingesta

En las infraestructuras Big Data, se recolecta información de una variedad de fuentes con un elevado grado de heterogeneidad.

Un desafío de seguridad clave en este tipo de proceso de recopilación de datos es la validación de la fuente para asegurar que se trata de una entrada fiable y no maliciosa.

Procesado

La gran variedad de datos ingestados dificulta el etiquetado de la información y aumenta el riesgo de que un atacante re-identifique a una víctima navegando sobre las diferentes fuentes de información. Durante el procesado debe asegurarse que se preserva la privacidad en todo el modelo.

Almacenamiento

Los sistemas Big Data almacenan un gran volumen de datos los cuales pueden contener información sensible. Será necesario establecer medidas de almacenado que garanticen su seguridad y no permitan el acceso ilegítimo a potenciales atacantes.

Uso

Durante la fase de uso, una gran cantidad de datos son accedidos, revisados, analizados y modificados. Debe asegurarse que los usuarios solo puedan realizar aquellas acciones para las que están autorizados. Por otra parte, debe quedar constancia en el sistema de las acciones realizadas sobre los datos.

Eliminación

Al final del ciclo de vida se eliminarán los datos que han dejado de ser útiles.

El método de destrucción normalmente depende de la sensibilidad de los datos. No obstante, se trata de un proceso crítico para preservar la privacidad.

Un reto para la gobernanza del dato es probar que el eliminado del dato se ha realizado correctamente.

2. Materiales y métodos

La metodología a seguir para el desarrollo de este proyecto seguirá las siguientes fases:

Plan de trabajo
En esta fase inicial se definirá el contexto y alcance del proyecto para fijar unos objetivos y la estructura de la memoria. Se realizará una planificación de tareas teniendo en cuenta las fases críticas y los retrasos posibles.

Documentación
Una vez conocido el alcance dará comienzo la fase de documentación para recopilar aquella información que pueda ser de ayuda para el desarrollo del proyecto.

Redactado
De manera paralela al avance del proyecto, con el fin de mejorar la eficacia en el desarrollo de la memoria, deberán redactarse todas las tareas y aspectos relevantes que aseguren el alcance de los objetivos iniciales.

Revisión
Periódicamente se realizarán revisiones del texto redactado con el fin de corregir errores, mejorar la claridad en la exposición de ideas y añadir aspectos olvidados.

Maquetación
Una vez finalizado el redactado será necesario revisar la estructura y composición antes de la entrega de la memoria final.

3. Resultados

Detallad en este apartado los resultados obtenidos utilizando la metodología descrita en el apartado anterior.

1. Plataformas y Riesgos comunes

Con el paso del tiempo, el incremento del flujo de datos de numerosas fuentes y la llegada del cloud y las tecnologías de virtualización han ido apareciendo diferentes plataformas de trabajo y analítica de Big Data.
Una plataforma de Big Data utiliza una combinación de herramientas software y hardware de administración de datos para almacenar conjuntos de datos, generalmente en la nube. [1]
El aumento de la demanda y utilidad de estos sistemas en el ámbito estratégico de una gran cantidad de empresas de todos los sectores ha provocado incluso que muchas herramientas enfocadas al BI, Data Warehouse y analítica de datos tradicional dieran el salto al mundo del Big Data.
Por ejemplo, todas las nuevas plataformas audiovisuales que han surgido en la última década usan estos sistemas para aplicarles algoritmos de Machine Learning y sugerir los vídeos o canciones que más pueden gustar al consumidor en función de sus visualizaciones anteriores.

1.1 Plataformas

A continuación, se van a listar algunas de las más importantes:

Google Cloud
Ofrece una gran cantidad de herramientas de administración de Big Data. BigQuery almacena petabytes de datos en un formato amigable y fácil de consultar. Dataflow analiza flujos de datos en paralelo y con Google Locker Studio los clientes pueden realizar cuadros de mando e informes a partir de una gran variedad de gráficos y visualizaciones.

Amazon Web Services
Es una plataforma de Amazon basada en la nube que ofrece herramientas de analítica muy variadas para cubrir todo el ciclo de vida del dato. Kinesis Firehose se encarga de extraer datos en tiempo real. No requiere ninguna administración y es posible configurar el encriptado de los datos. Redshift es una de las herramientas más famosas de data warehousing para Big Data debido a su velocidad. Amazon DynamoDB es una base de datos NoSQL que puede manejar una gran cantidad de datos y consultas por segundo.

Snowflake
Es una herramienta de data warehouse usada para almacenamiento, procesado y análisis que permite su despliegue en plataformas de nube pública (AWS, Google Cloud, Azure…).

Cloudera
Es una plataforma de datos en la nube híbrida basada en Apache Hadoop, Cloudera puede manejar grandes cantidades de datos de diversas naturalezas: pertenecientes a máquinas, texto, imágenes etc de manera segura. Permite manejar petabytes de datos de manera fácil y rápida.

Oracle Cloud
Permite migrar una gran variedad de datos a servidores en la nube. Oracle ofrece diferentes servicios de interés: Oracle Autonomous Database proporciona una base de datos con un gran rendimiento. Sin necesidad de instalar ningún software adicional maneja el aprovisionamiento de la base de datos, las copias de seguridad, aplicación de parches y actualizaciones… Se trata de un servicio completamente elástico según las necesidades. Incluye además un servicio de Machine Learning en diversos lenguajes: Python, R, SQL. OCI Data Flow es un servicio basado en Apache Spark que permite realizar tareas de procesamiento en conjuntos de datos extremadamente grandes. Los desarrolladores pueden utilizar Spark Streaming para desarrollar ETL en la nube sobre flujos de datos en tiempo real.

MongoDB
Esta plataforma permite el almacenado de datos en la nube como documentos JSON flexibles ya que se pueden organizar de muchas maneras, incluso anidados uno dentro de otro. Es herramienta pensada para desarrolladores de aplicaciones que permite la búsqueda sencilla de cadenas de texto y etiquetas geográficas. MongoDB Atlas ofrece todo lo necesario para implantar un sistema de Big Data. Desde una base de datos en la nube, hasta un sistema de analítica de datos y seguridad end-to-end.

1.2 Riesgos Comunes

El ciclo de vida del dato está divido en las siguientes etapas: **Ingesta**, **Procesado**, **Almacenamiento**, **Uso** y **Eliminación**. De manera inherente al alcance de cada una de estas fases tendremos asociados riesgos específicos, que se tratarán con más detalle en los próximos apartados, no obstante, es posible destacar algunos que pueden afectar de manera transversal a todas ellas:

Manipulación de los registros de actividad (log)
Un atacante que ha conseguido acceder al sistema de manera ilegítima no debería ser capaz de poder manipular los registros de actividad, que dejan traza de todo lo que sucede en el sistema, aunque escale privilegios. Los logs pueden ser un activo fundamental a la hora de reconstruir los sucesos durante el análisis forense posterior a un ataque. La manipulación o supresión de las marcas de tiempo registradas puede imposibilitar la reconstrucción de los hechos. Sólo aquellos usuarios con un grado máximo de privilegios, como administradores, debe tener permisos de escritura sobre estos registros.

Manipulación de la configuración

Prácticamente todos los componentes de un sistema de Big Data o no depende de su configuración. Todo lo relativo a este aspecto es potestad del administrador: privilegios de acceso, flujos de actividades, registro de actividad, encaminamiento, excepciones de seguridad…

La manipulación de la configuración puede tener muchas consecuencias:

- Reducción del límite máximo de recursos de memoria usados en MapReduce (lo que podría afectar drásticamente en el rendimiento del sistema).
- Modificación del host y puertos del sistema.
- Habilitar o deshabilitar el control de acceso.
- Modificar la ruta de almacenamiento de logs.

Suplantación de identidad

Cuando un atacante consigue hacerse pasar por un usuario autorizado, disfruta de los privilegios de este para sus fines propios.

Cuanto más privilegio tenga el usuario suplantado mayor será la brecha en la seguridad.

Con la finalidad de reducir este riesgo han surgido los nuevos mecanismos de autenticación multifactor que añaden una capa adicional de seguridad más allá de la simple posesión de un usuario y una contraseña como puede ser un código vía SMS o datos biométricos.

Abuso de privilegios de acceso

Cada usuario disfruta de un nivel de privilegios para un determinado propósito. Cuando un usuario abusa de su nivel de privilegios para realizar tareas que no son de su competencia puede traer consecuencias muy graves para la seguridad del sistema.

Un control de acceso basado en roles correctamente implementado asegurará que los usuarios solo pueden realizar aquellas acciones para las cuales fueron dados de alta en el sistema.

Modificación/Destrucción de información

Un atacante con acceso al sistema y un grado elevado de privilegios podría eliminar o modificar información de manera de manera intencionada. Como se ha comentado anteriormente una gran cantidad de decisiones empresariales estratégicas son tomadas en base a los datos obtenidos, procesados y analizados. La pérdida o falta de integridad de la información puede inducir a una empresa a una toma de decisión equivocada con consecuencias catastróficas:

- Pérdida de confianza de los clientes.
- Pérdidas económicas.
- Pérdida de prestigio.
- Pérdida de una posición de privilegio frente a los competidores.

Esta amenaza hace patente la importancia de seguir el principio de mínimos privilegios el cual pretende asignar a los usuarios los mínimos permisos necesarios para desempeñar sus tareas.

Ingeniería Social

La ingeniería social es un tipo de ataque que utiliza la manipulación psicológica con el objetivo de conseguir que los usuarios revelen información confidencial o realicen cualquier tipo de acción que pueda beneficiar al ciberdelincuente. Se basa en la premisa de que es más fácil manera a las personas que a las máquinas e implica el uso de técnicas de persuasión, engaño o manipulación para obtener información o acceso no autorizado. [43]

La ingeniería social explota vulnerabilidades humanas para manipular a las personas como:

- **Respeto a la autoridad:** Como normal general, la gran parte de los ciudadanos respeta a la autoridad como sus superiores. Este tipo de ataque utiliza ese respeto a las autoridades como las Fuerzas y Cuerpos de Seguridad del estado para aprovecharse de él.
- **Urgencia:** También es muy frecuente instar a las victimas a realizar una acción de manera urgente con tal de resolver un incidente para provocar una reacción rápida sin pensar.
- **Voluntad de ayudar:** Los ciberdelincuentes pueden hacerse pasar por falsos empleados de la empresa para aprovecharse de la voluntad de ayudar con la que se cuenta en muchos entornos laborables. Otra variante, es hacerse pasar por un técnico de informática para instalar herramientas de acceso remoto no autorizado.
- **Temor a perder un servicio:** Es una técnica típicamente utilizada en campañas de phishing. Bajo el pretexto de existir repetidos accesos no autorizados, cambio de políticas de seguridad o cualquier otro engaño, los ciberdelincuentes fuerzan a la víctima a acceder a una web fraudulenta donde roban información confidencial.
- **Ofertas gratuitas:** En este tipo de engaño se ofrece un producto o servicio de manera gratuita a cambio de información privada. Es común en redes sociales, aplicaciones de mensajería y las webs emergentes que aparecen al navegar por sitios poco legítimos.

Para obtener credenciales de inicio de sesión o información personal, permitiendo a los atacantes acceder a sistemas protegidos mediante diversos métodos:

- **Phishing:** Utiliza correos electrónicos, comunicaciones o mensajes de texto (smishing) para engañar a las personas para que compartan información confidencial como contraseñas, números de tarjeta de crédito…
- **Ingeniería social en línea:** A partir de un perfil falso en red sociales los atacantes pueden enviar mensajes directos para intentar engañar a las personas o convencerlas de hacer clic en enlaces maliciosos.
- **Ingeniería social en persona:** También pueden hacerse pasar por técnicos informáticos o administradores de sistema para obtener acceso a los sistemas de los empleados.
- **Phishing telefónico:** Los delincuentes pueden llamar por teléfono y hacerse pasar por agentes de soporte técnico de una compañía telefónica o similar con el fin de engañarlos para que proporcionen información personal o accedan a sus sistemas.

Una vez dentro, los ciberdelincuentes pueden robar información confidencial, instalar código malicioso, ascender de privilegios o moverse hacia otros servidores de manera lateral.
La ingeniería social se ha convertido en una de las principales causas de ataque de ciberseguridad de los últimos años ya que es la más efectiva para evitar todas las medidas de seguridad y defensas desplegadas. La solución más efectiva es realizar formaciones para concienciar a los empleados del riesgo de sus acciones y la importancia de las buenas prácticas.

1.3 Medidas de seguridad

Todos los riesgos analizados en el apartado anterior y que pueden afectar a todas las fases del ciclo de vida del dato tienen un denominador común, el control de acceso.
Los sistemas de control de acceso se encuentran entre los componentes de seguridad de red más críticos. Es más probable que estos controles se vean comprometidos debido a la mala configuración de las políticas de seguridad de acceso que por fallos en los protocolos o primitivas criptográficas.
Los problemas de acceso se vuelven más graves a medida que los sistemas de software son más complejos, como los sistemas de procesamiento de Big Data, que se implementan para administrar una gran cantidad de información confidencial y sensible.
Estos sistemas se encargan de que solo los usuarios legítimos consigan acceder a los datos.
El proceso de acceso se puede reducir a dos fases fundamentales:

Autenticación
La autenticación es el proceso de demostrar que se es quien se dice ser. Esto se logra mediante la verificación de la identidad de un usuario o dispositivo.
Para implementar este proceso la mayoría de las plataformas de Big Data utilizan:

- Kerberos (Anexo 1)
- LDAP (Anexo 2).

Autorización
Es el acto de conceder a una parte autenticada permiso para realizar alguna acción. Especifica a qué datos se puede acceder y qué se puede hacer con ellos. Se puede implementar mediante:

- Listas de control de acceso, ACL (Anexo 3)
- Control de acceso basado en roles, RBAC (Anexo 4).

Mínimos privilegios
Más que una medida de seguridad se trata de una buena práctica que puede proteger el sistema de manera notoria en caso de que un usuario se vea comprometido.
Este principio básico de seguridad consiste en mantener los privilegios de los usuarios al mínimo, es decir, evitar que los usuarios y grupos tengan más permisos de los estrictamente necesarios.

Para los usuarios generales deben utilizarse grupos que tengan privilegios limitados en lugar de roles de administrador. De esta manera se evitará que los usuarios tengan acceso a servicios e información que no necesitan para desarrollar sus labores.
Esto proporciona una capa adicional de seguridad ya que si un usuario se ve comprometido no podría realizar acciones con un impacto notorio.
Por este mismo motivo se insta a no utilizar ningún tipo de usuario con permisos de administrador.

Formación
Con la llegada del smartphone y su arraigo en la vida cotidiana han aumentado de manera considerable los fraudes y estafas cibernéticos a partir de ingeniería social hasta convertirse en una de las principales amenazas de la década.
De la mano de esta evolución, los entornos de trabajo se han ido digitalizando cada vez más con el tiempo mediante el correo electrónico, aplicaciones en la nube, pedidos online, dispositivos móviles… Exponiendo a los empleados a un nuevo ecosistema.
El crecimiento de las nuevas tecnologías en las empresas hace indispensable la concienciación sobre los riesgos asociados a ellas. Es necesario que los empleados conozcan y apliquen buenas prácticas en el uso de todo tipo de dispositivos y soluciones tecnológicas por lo que se les debe proporcionar la formación en ciberseguridad adecuada a su puesto para prevenir incidentes.
También se pretende buscar el compromiso total por parte de los empleados y la dirección, que debe ser consciente de que la formación debe ser una actividad continua para que tenga el efecto deseado y se adapte a las nuevas tecnologías. [44]
Algunos de los puntos claves a tratar en este tipo de formaciones son:

- Procedimientos y controles de seguridad básicos.
- Conocimiento y cumplimiento de la normativa y legislación vigente.
- Seguridad en el puesto de trabajo, aplicaciones permitidas, uso correcto de recursos, protección de datos, etc.
- Concienciar a los empleados sobre los peligros de la ingeniería social.
- Responsabilidad personal por acción u omisión.
- Realizar simulaciones de ataques de ingeniería social para ayudar a responder adecuadamente.

El objetivo principal de estos cursos es garantizar que los empleados conocen, entienden y cumplen las normas, medidas de protección y buenas prácticas advirtiéndoles de los riesgos que puede entrañar el mal uso de dispositivos y servicios tecnológicos.

2. Fase Ingesta

En la fase inicial del ciclo de vida el dato puede ser recolectado de un gran número de fuentes y dispositivos con diferentes formatos:

- Estructurado.
- Semiestructurado.
- No estructurado.

Y con una gran variedad de propósitos:

- Mantenimiento.
- Medidas de rendimiento.
- Elaboración de perfiles de consumidor y patrones de comportamiento.
- Estrategias de Marketing.
- Monitoreo de actividades.

Hace referencia a una de las disciplinas de Big Data que más ha cambiado en los últimos años debido a que los datos son generados en grandes volúmenes y provenientes de dispositivos distribuidos por todo el mundo que transmiten los datos generados por las redes sociales, plataformas digitales, datos de geolocalización, entre otros muchos.
Para proteger el dato a lo largo de su ciclo de vida es fundamental garantizar la seguridad desde la primera etapa.
Para encontrar medidas de seguridad será necesario plantear las siguientes preguntas:

- ¿De qué manera puede asegurarse que el origen de datos es legítimo?
- ¿Cómo puede saberse que los datos recibidos no han sido modificados?
- ¿Qué se puede hacer para asegurar confidencialidad de los datos en movimiento?

2.1 Riesgos

Dejando de lado todos los riesgos que pueden afectar de manera común a todas las fases del ciclo de vida la ingesta de datos es susceptible a diversos riesgos que serán tratados a continuación:

Fugas de datos
Se considera fuga de datos a la pérdida de confidencialidad de forma que información privilegiada sea accedida por personal no autorizado. [42]
El origen de las amenazas puede ser tanto:

- **Interno:** Ocasionado por propios empleados ya sea de manera intencionada o no.
- **Externo:** Los posibles orígenes externos abarcan desde organizaciones criminales, activistas, ciberdelincuentes solitarios…

Las causas de los principales casos de fuga de información pueden clasificarse en dos grupos:

- **Organizativas:**
  - Debido a la **falta de conocimiento** y formación por parte de los empleados, que deben hacer un uso responsable de los distintos servicios y disponer de ciertas nociones de ciberseguridad.
  - **Ausencia de políticas de seguridad**, procedimientos o pautas para los trabajadores en el ámbito de la ciberseguridad que indiquen claramente a los usuarios qué actividades entrañan un riesgo mayor o menor para que se produzca una fuga de información.
  - No existencia de **acuerdos de confidencialidad** que limiten de manera legítima actividades a los usuarios que puedan entrañar un uso malintencionado de la información.
- **Técnicas:**
  - Sistemas sin protección contra **código malicioso** capaz de mantenerse oculto en el sistema mientras recoge y envía información que puede ser publicada o difundida.
  - Sistemas de autenticación poco seguros. El **acceso no autorizado** es una de las causas más extendidas. En la mayoría de los casos podrían evitarse si los sistemas y aplicaciones estuvieran actualizados.
  - La generalización de **servicios en la nube** puede producir una falsa percepción de seguridad cuando en realidad no es así. Es muy importante prestar atención a las configuraciones por defecto y a las herramientas de seguridad que ofrecen los proveedores de servicios. Un sistema Big Data implantado en cloud debe ser igualmente precavido con la seguridad que un sistema on premise. Relegar responsabilidades al proveedor es una política extremadamente arriesgada y peligrosa.

Una fuga de información puede causar un importante daño a la imagen de la empresa y hiriendo su reputación y mermando la confianza de los clientes. Comprender las consecuencias es esencial y necesario para la adecuada gestión de incidentes y la implementación de medidas preventivas. No obstante, conocer el impacto que tendría este ataque depende de muchos factores:

- **Tipo de organización:** Si se trata de una entidad pública el ataque puede llevar de manera inherente un daño político. En el caso de una entidad privada, las sanciones económicas y pérdida de confianza pueden tener consecuencias muy significativas sobre el negocio y la actividad.
- **Tipo de información:** Si se trata de información confidencial y crítica o si la divulgación no tendría ningún impacto relevante en la empresa.
- **Datos personales:** La difusión de un dato que identifique o puede ayudar a reidentificar a una persona puede conllevar graves sanciones a la organización y daños irreparables a los titulares de los datos, mientras

que si se tratan de datos de mantenimiento, monitorización o técnicos el daño sería menor.

El Instituto Nacional de Ciberseguridad propone un plan de gestión de incidentes de este tipo basado en 6 fases:

- **Fase inicial:** En la mayoría de las ocasiones el incidente no es detectado hasta que la filtración de datos se hace pública, por este motivo, una rápida actuación en los momentos cercanos a la detección es vital. Una vez se conoce el incidente es necesario alertar internamente de la situación para activar el protocolo de actuación correspondiente.
- **Fase de lanzamiento:** Una vez detectado es necesario iniciar el protocolo interno para analizar la situación y coordinar las primeras acciones.
- **Fase de auditoría:** A continuación, es necesario recopilar toda la información posible sobre el incidente. Determinar la cantidad de información difundida (número de registros, volumen de datos, etc.), conocer el tipo de datos ha sido afectada y establecer la causa de la filtración.
- **Fase de evaluación:** Con toda la información recopilada es necesario evaluar el impacto y las posibles consecuencias. Deben realizarse las primeras tareas para cortar la filtración y mitigarla, contactar con los afectados en caso de que fuera necesario, determinar las consecuencias económicas, planificar el contacto con las entidades de seguridad correspondientes y la comunicación pública a los medios de la situación.
- **Fase de mitigación:** El primer paso es reducir la brecha y evitar futuras fugas de información. A continuación, debe minimizarse la difusión, en especial si ha sido publicada en Internet, contactando con los sitios y solicitando su retirada, sobre todo si se trata de información protegida por la LOPD.
- **Fase de seguimiento:** Una vez aplicadas las acciones correctivas será necesario evaluar el resultado y efectividad en relación con las consecuencias y su impacto. Durante esta fase se iniciará la estabilización del sistema.

Cada organización es diferente y será necesario buscar el equilibrio entre el compromiso que suponen la complejidad, coste y riesgo en relación con la implementación de las medidas de seguridad.

Man-in-the-middle

Un ataque MITM es una técnica de hacking en la cual el atacante se sitúa entre dos extremos de comunicación (la fuente de origen de la ingesta y el sistema Big Data destino en este caso) para interceptar, manipular o espiar la información que se transmite:

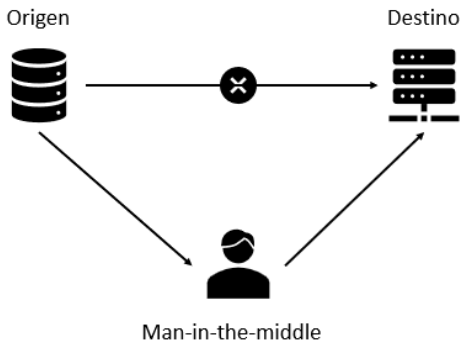

Figura 2: Esquema ataque Man-in-the-middle.

El atacante puede utilizar herramientas de escucha de red (sniffer) para capturar, descifrar - si el protocolo no es lo suficientemente robusto -, leer e incluso modificar los datos en tránsito antes de llegar al destino. En estos casos la confidencialidad e integridad se verían gravemente afectadas.

En un sistema de Big Data, que maneja grandes volúmenes de datos muchos de los cuales son confidenciales, un ataque MITM podría interceptar contraseñas, números de cuenta, números de tarjeta de crédito, importes, ubicaciones…

Por ejemplo, si un atacante intercepta los datos de una transacción financiera en un sistema de Big Data, podría cambiar el número de cuenta destinatario para desviar todos los fondos a una cuenta deseada y modificar los importes.

Por otra parte, si no se aplican las medidas necesarias en las plataformas de pago podría accederse al número de tarjeta y códigos de seguridad de las transacciones o compras online.

Sufrir un ataque de este tipo puede traer consecuencias financieras graves para la empresa a nivel económico, de reputación o a modo de sanciones regulatorias.

Si no existe ninguna medida para que los datos viajen de manera segura la confidencialidad e integridad podrían verse altamente afectadas con el simple acceso a la comunicación.

DDoS

Si un atacante consigue suplantar la identidad de una fuente de datos legítima o hacerse pasar por una podría alterar el flujo de los datos de tal manera que provoque una caída del servicio. En este caso, la indisponibilidad de la información del sistema podría ocasionar pérdidas irreparables, ya que los datos son un activo fundamental para el desarrollo de las actividad y toma de decisiones estratégicas de muchas empresas, si no se han implantado las salvaguardas necesarias o si no existe un plan de contingencia para minimizar los daños una cuando suceda el ataque.

Un ataque de denegación de servicio puede producirse de diversas maneras:

- **Saturación de ancho de banda:** El atacante envía grandes cantidades de tráfico de red al objetivo con la intención de agotar el ancho de banda disponible sobrecargando los servidores y dejándolos fuera de servicio.
- **Saturación de recursos:** En este caso, el atacante puede utilizar una red de bots para enviar solicitudes de carga de página al sitio web o aplicación objetivo para que los servidores no puedan responder a las solicitudes legítimas de los usuarios debido a la sobrecarga.
- **Ataque de vulnerabilidad:** El atacante aprovecha una vulnerabilidad para causar un fallo en el sistema.

2.2 Medidas de seguridad

Para mitigar los riesgos se pueden implementar las siguientes medidas de seguridad:

Minimización de datos
La minimización de datos es la práctica de limitar la cantidad de datos recopilados y almacenados en el sistema a aquellos que son estrictamente necesarios para el fin del tratamiento.
Esta medida permite reducir el riesgo de exposición, la superficie a anonimizar, el riesgo de re-identificación y el impacto en caso de brecha de seguridad.
El principio de minimización está recogido en el Reglamento General de Protección de Datos:

"Los datos personales serán:
Adecuados, pertinentes y limitados a lo necesario en relación con los fines para los que son tratados («minimización de datos»)." [41]

Junto con otras prácticas y medidas de seguridad puede ser de gran ayuda a la hora de proteger los datos.

Cifrado
La encriptación es una de las medidas más eficaces y comúnmente utilizadas para proteger los datos en tránsito. En los sistemas de Big Data el encriptado puede aplicarse para proteger los datos durante la transmisión a través de la red (de cualquiera de las fuentes origen al sistema destino) o entre nodos. De esta manera un atacante que tratara de capturar los datos no conseguiría interpretarlos ni extraer información útil.
Proteger la claves es tan importante como proteger los datos en movimiento. Por ello, de la mano del encriptado será necesario implementar una solución de administración de claves para asegurar que están debidamente protegidas.
Normalmente, las plataformas de Big Data más comunes utilizan AES-256 (Anexo 5) y certificados SSL/TLS (Anexo 6) para encriptar los datos en movimiento.
Los certificados SSL/TLS además de cifrar los datos aseguran la legitimidad de la fuente de datos desde la que se realiza la ingesta.
Además, la gran mayoría ofrece soluciones nativas para proporcionar una solución estándar y lograr una implementación más sencilla:

- **Kafka**

Kafka permite a los clientes el uso de SSL para encriptar el tráfico de datos. La solución está desactivada por defecto, pero puede usarse si se requiere.[4]

- **AWS Key Management Service (KMS)**

Es un servicio de gestión de claves de cifrado propocionado por Amazon Web Services que permite controlar el acceso a las claves de cifrado utilizadas de manera sencilla. Es compatible con diversos algoritmos de cifrado además de AES-256. A partir del servicio de administración de identidades y roles de Amazon (AWS IAM) un usuario puede crear, eliminar, modificar o usar las claves para encriptar o desencriptar.[5]

- **AWS Certificate Manager (ACM)**

Es un servicio de Amazon que permite aprovisionar, administrar y desplegar certificados SSL/TLS públicos y privados que pueden ser usados para el cifrado de los datos en tránsito de aplicaciones y servicios que se ejecutan en la nube de AWS.
De esta manera es posible asegurar la comunicación entre los recursos conectados en las redes privadas como servidores, dispositivos IoT (sensores y demás) y aplicaciones.[6]

- **Azure**

Azure permite añadir certificados digitales para proteger el tráfico de datos entre servidores o bases de datos mediante SSL/TLS. Para ello es posible habilitar SSL/TLS en las conexiones a base de datos (en caso de que el sistema Big Data se nutra de una base de datos). [7]
Azure SQL Database, SQL Managed Instance y Azure Synapse exigen el cifrado SSL/TLS en todas las conexiones para garantizar que los datos estén cifrados entre cliente y servidor asegurando la confidencialidad de los datos frente a una posible escucha.[8]
No obstante, también será necesario configurar el cliente origen para que utilice SSL/TLS.

- **Google Cloud Key Management Service**

Es un servicio de Google Cloud Platform que permite crear, importar y administrar claves y realizar operaciones criptográficas en un único servicio en la nube.
Se trata de una solución equivalente a AWS Key Management Service.[9]

- **Cloud SQL**

Se trata de un servicio totalmente gestionado de bases de datos relacionales de MySQL, PostgreSQL y SQL Server, con una gran variedad de colecciones, extensiones y configuraciones posibles.
El servicio ofrece encriptado de datos en tránsito y conectividad privada. Google encripta y autentica todos los datos en tránsito en una o más capas de red cuando los datos se transmiten fuera de los límites físicos no controlados por Google. Según la conexión que se realice, Google aplica las protecciones predeterminadas a los datos en movimiento, asegurándolos mediante TLS, por ejemplo.[10]

IDS/IPS

Un IDS/IPS es un sistema mixto encargado de monitorizar la red en busca de actividades sospechosas, detectar amenazas, enviar alertas e incluso bloquear el tráfico malicioso.

El IDS (Intrusion Detection System) se encarga de monitorizar el tráfico para detectar accesos no autorizados a un servidor o red. Cuando detectan una actividad sospechosa emiten una alerta al administrador del sistema que deberán tomar las medidas oportunas. Estos sistemas detectan pero no mitigan la intrusión. Su actuación es reactiva.

El IPS (Intrusion Prevention System) es un software utilizado para prevenir intrusiones. Estos sistemas realizan un análisis en tiempo real de las conexiones y protocolos para identificar si se está produciendo o se va a producir un ataque según patrones o comportamientos sospechosos.

El principal hándicap de esta medida, que basa su funcionamiento en el análisis del flujo, en entornos de Big Data es que el volumen del tráfico es considerablemente mayor al de los sistemas de Datawarehouse o BI tradicionales por lo que se requiere una capacidad superior. [11]

Existen algunas soluciones específicas de este tipo para evitar ataques DDoS:

- **Azure DDoS Protection**

  Se trata de un servicio específico contra ataques DDoS ofrecido por Azure. Ofrece monitorización 24/7 del tráfico analizando indicadores de ataque de denegación de servicio. Permite configurar un sistema de alertar y acceder al equipo DDoS Rapid Response en caso de ataque. [12]

- **AWS Shield**

  Es el servicio de protección contra DDoS equivalente de Amazon. [13]

Hashing

Un aspecto crítico en el proceso de ingesta es asegurar la integridad de los datos, que estos no han sido modificados en el tránsito de la fuente al sistema Big Data por un atacante.

El hashing es una técnica que permite calcular un valor numérico finito y con longitud determinada para un conjunto de datos. Esta medida puede ser utilizada para comprobar que los datos no han sido alterados ya que la modificación de un solo bit del flujo de datos en movimiento tendría como resultado un valor hash distinto al original.

Es posible garantizar la integridad de los datos a partir de un protocolo de este tipo:

- La fuente de datos calculará el valor hash de los datos originales antes de la transmisión mediante algún algoritmo conocido como SHA-256.
- Los datos serán transmitidos desde el origen al sistema Big Data.
- Se calculará el valor hash sobre los datos insertados mediante el mismo algoritmo que en el punto inicial.
- Se compararán los dos hashes, el calculado antes de la transmisión y una vez recibidos, si los valores coinciden se puede considerar que los datos no han sido modificados en el tránsito y se pueden procesar sin

problemas, de lo contrario, debe considerarse que los datos han sido modificados y deberán descartarse o realizar un nuevo envío.
- Una vez finalizado el proceso de verificación de la integridad de los datos, se almacenará la información relevante en una tabla de auditoría con una marca de tiempo de la recepción, origen…

Uno de los inconvenientes o dificultades de este método es que los sistemas de Big Data pueden recoger un gran volumen de datos en tiempo real lo que dificultaría la selección del set de datos sobre los que aplicar el hash. Una posible solución es establecer ventanas temporales predeterminadas para considerar un set de datos tanto en origen como destino y aplicar la función hash de manera periódica y automatizada.

Monitorización

Una medida muy relacionada con los sistemas de detección de intrusos es la monitorización. Existen varias opciones para detectar anomalías en tiempo real en la ingesta de datos:

- **Definición de métricas:** Es importante definir métricas que den una idea del rendimiento del sistema durante la ingesta, el volumen de datos obtenidos por segundo, número de errores por registro…
- **Monitoreo en tiempo real:** Realizar una monitorización de la ingesta de datos en tiempo real permite detectar problemas de manera temprana y por lo tanto una reacción más rápida ante ellos. Apache Kafka cuenta con una opción de este tipo muy útil para analizar grandes volúmenes de datos de forma instantánea y generar alertas en caso de detectar amenazas.
- **Configuración de alertas:** Permiten notificar a los administradores del sistema en caso de detectar problemas en la ingesta de datos, errores en la fuente, caída del servicio, desviaciones sospechosas en los valores de las métricas definidas en el punto inicial…

Una práctica común es el uso de una tabla de auditoría para mantener el registro de actividad histórica. También puede contener información de las métricas definidas y los posibles errores durante el proceso de ingesta.
Existen dos soluciones de monitorización utilizadas por diversas plataformas de Big Data:

- **Apache Ambari:** Herramienta que permite gestionar y monitorizar los clústeres de Apache Hadoop de una manera más sencilla. Proporciona un cuadro de mando para mostrar las métricas relevantes y la salud del sistema de una manera visual y amigable. Utilizado por Hortonworks Data Platform y Microsoft Azure HDInsight. [15]
- **Nagios:** Ofrece un software de gestión de logs y monitorización centralizado. Permite visualizar los datos de log en tiempo real y analizar los problemas. Es posible crear alertas personalizadas basadas en consultas y umbrales importantes para el cliente. Además, ofrece la posibilidad de crear cuadros de mando flexibles y totalmente personalizados. IBM BigInsights utiliza este servicio. [16]

3. Fase Procesado

Los sistemas Big Data se nutren de una gran cantidad de orígenes de datos que pueden ser de naturalezas totalmente distintas no solo a nivel estructural (un origen puede contener datos estructurados, otro semi-estructurados, otro desestructurados…) sino también de información (datos de facturación, datos de clientes, comentarios de usuarios…) o incluso de formato (números de teléfono con distintos formatos, tarjetas de crédito, códigos postales…). Esta elevada heterogeneidad de los orígenes obliga a los sistemas de Big Data a realizar un procesado para dar un formato estándar a los datos en bruto y dotarlos de sentido convirtiéndolos en información útil para la organización.
El procesado de datos incluye la limpieza, transformación y agregación de datos, así como la aplicación de algoritmos para extraer información útil. Este proceso también puede incluir la eliminación de duplicados, corrección de errores y normalización de datos para asegurar la calidad (aspecto muy importante para con la seguridad que se analizará más adelante) y consistencia de los datos.
Esta etapa puede ser altamente compleja debido al gran volumen de datos y a su diversidad.

3.1 Riesgos

La fase de procesado de datos en Big Data implica una serie de riesgos que deben ser abordados para garantizar las diferentes dimensiones de seguridad del dato (integridad, confidencialidad y disponibilidad). A continuación, se presentan algunos de los riesgos más comunes durante la fase de procesado de datos, dejando de lado aquellos que se hayan comentado anteriormente (fuga de información, ataque de denegación de servicio…):

Ataque de manipulación de datos
Un ataque de manipulación de datos es aquel en el que el atacante intenta modificar los datos que se encuentran en un sistema de Big Data para obtener algún beneficio o causar daños a la organización.
Aunque este ataque puede llevarse a cabo en varias fases del ciclo de vida la etapa de procesado es especialmente vulnerable.
Un atacante con acceso al sistema puede modificar los datos de varias formas:

- **Modificación de valores:** El atacante puede modificar los valores de los datos ingestados o en tránsito si no se han aplicado correctamente las medidas nombradas en el apartado anterior. De esta manera podría infligirse un daño considerable en el procesado y análisis de los resultados con consecuencias directas en las decisiones de la organización.
- **Eliminación de datos:** Consiste en la eliminación de datos por parte de un atacante. De la misma manera que la modificación, la eliminación puede provocar que el análisis y los algoritmos de aprendizaje se apliquen sobre muestras de datos no lo suficientemente representativas

proporcionando soluciones sesgadas y poco realistas, pudiendo significar pérdidas cuantiosas para las empresas.

- **Adición de datos falsos:** Mediante la adición de datos falsos el atacante podría influir en los análisis y conclusiones extraídas. Si el atacante tiene el suficiente conocimiento del negocio y de los datos que trata la organización podría superar los algoritmos de detección de valores atípicos que descartan aquellos valores que salen de unos ciertos límites estadísticos.
- **Corrupción de datos:** Los datos son alterados para que sean ilegibles o inutilizables. De esta manera, la organización no podría realizar aplicar ninguna acción sobre los datos, en caso de no constar sistemas de recuperación de seguridad, obstaculizando considerablemente las tomas de decisiones empresariales.

Vulnerabilidades del sistema

Los sistemas de procesado de datos pueden tener vulnerabilidades que los atacantes pueden aprovechar para comprometer la seguridad del sistema. El código que realiza el procesado puede contener vulnerabilidades a causa de malas prácticas que dejen puertas traseras o información útil para el atacante (nombre de servidores, nombre de usuarios de base de datos…).

Es importante realizar análisis y revisiones del sistema periódicas para corregir y evitar la explotación de vulnerabilidades.

Además, es altamente recomendable seguir las buenas prácticas en el desarrollo de código para evitar este tipo de amenazas.

Las grandes plataformas de sistemas Big Data ofrecen guías de buenas prácticas para el procesado de datos.

3.2 Medidas de Seguridad

Existen varias medidas para mitigar los riesgos mencionados anteriormente.

Validación de esquema

Las bases de datos relacionales tienen esquema, lo cual asegura que los datos contenidos tienen el formato esperado. Pero muchos formatos modernos como JSON, son no-relacionales y no imponen esquemas. JSON permite asociar diferentes tipos de datos con la misma etiqueta en diferentes filas.

Por ejemplo: El atributo LAST_NAME puede tener valores como “Hola” y como 59.38 o incluso no aparecer.

Los datos que no cumplen con el formato esperado son un gran problema para las operaciones posteriores.

Aquí es donde gana importancia la validación de esquema. Se trata de una técnica utilizada en el procesado para comprobar si los datos recibidos cumplen con un formato específico definido en un esquema. El esquema define las reglas y restricciones de cómo debe estar estructurados los datos y qué formato deben tener.

En este método, el proceso de validación analiza los datos que han sido ingresados en el sistema y que han sido procesados y los compara con el esquema definido. Si los datos no cumplen con el esquema, son rechazados. Es posible activar notificaciones a los usuarios, administradores para alertar en estos casos.

La validación de esquema ayuda a prevenir errores en los datos y garantizar su calidad, de esta manera se eliminarían los datos modificados con carácter malicioso por un atacante o aquellos que hayan podido quedar corruptos por su intervención.
Existen diferentes herramientas para implementar la validación de esquema, a continuación, se citan algunas de ellas:

- **Apache Avro:** Es una herramienta de serialización de datos que incluye un esquema definido en JSON. Avro proporciona un mecanismo de validación incorporado para asegurarse de que los datos cumplen con el esquema definido. Si el esquema fuera modificado por un atacante intentando alterar el comportamiento, o forzar un error, del procesado Avro lo detectaría. Esta herramienta es muy popular en sistemas de streaming como Kafka debido a su rendimiento. [18]
- **JSON Schema:** Es una herramienta para definir y validar esquemas JSON. Permite definir restricciones de validación detalladas para cada propiedad en un objeto JSON. Es posible definir un objeto JSON genérico que describe el formato que deberán tener todos los documentos con los nombres de los campos permitidos, obligatorios y su tipo de dato asociado, además de poder añadir valores máximos y mínimos (Por ejemplo, atributos como LATITUDE no pueden contener valores menores a -90 ni mayores a 90 ya que no existen valores menores ni mayores a esos de latitud terrestre). JSON Schema se ha convertido en un método robusto de verificación debido a su sencilla aplicación e integración con Python. Únicamente es necesario importar la librería, declarar el esquema y el JSON a comparar y la función *validate* comparará ambos objetos validando su formato. [19]
- **Apache Nifi:** Es una plataforma de procesamiento de datos que incluye varias bibliotecas de validación de esquema (ValidateCsv, ValidateJson, ValidateXml…). Estos controles permiten la validación de datos según un esquema definido antes de que se procesen los datos en la plataforma. Similar a JSON Schema, con esta herramienta sería posible descartar aquellos documentos con datos alterados de manera maliciosa y que no cumplen las especificaciones de formato impuestas por el esquema definido. [20]
- **Pydantic:** Es una biblioteca de validación de datos para Python que permite definir esquemas y validar datos de entrada contra este. Pydantic es compatible con varios formatos de datos como JSON, YAML y TOML. Tiene un uso similar a la aplicación de JSON Schema en Python. Es posible crear una clase con el modelo base, que se usará como esquema, importar el archivo a comparar y utilizar las funciones de la librería *pydantic.*[21]

Detección de valores atípicos (outliers)

La detección de valores atípicos durante el procesado de datos es una etapa vital para garantizar la calidad de los datos, estrechamente relacionado con la seguridad como se verá en el siguiente apartado, y la precisión de los análisis y conclusiones realizadas a posteriori.
A partir de un buen proceso de detección de valores atípicos el sistema sería capaz de descartar aquellos valores que se encuentren fuera de unos rangos considerados aceptables. De esta manera, todos aquellos valores insertados o

modificados por un atacante para alterar los análisis de los datos y el aprendizaje podrían ser detectados mediante métodos estadísticos y descartados sin afectar a los datos legítimos.
Existen diferentes métodos para detectar valores outliers y garantizar la precisión de los datos. Algunas de las técnicas utilizadas son:

- **Método del rango intercuartil (IQR**): Utiliza los cuartiles Q1 y Q3 para calcular el rango intercuartil (IQR), que es la diferencia entre Q3 y Q1, y lo utiliza para identificar los valores fuera de los límites superiores e inferiores. Los valores que están por debajo o por encima de los límites se consideran outliers:

  $$\text{Outlier} < Q1 - 1{,}5 \cdot IQR$$
  $$\text{Outlier} > Q1 + 1{,}5 \cdot IQR$$

- **Método de la desviación estándar:** Este método utiliza la media ($\bar{x}$) y la desviación estándar (σ) para identificar los valores que están alejados significativamente de la media:

  $$\text{Outlier} < \bar{x} - 3 \cdot \sigma$$
  $$\text{Outlier} > \bar{x} + 3 \cdot \sigma$$

- **Detección de anomalías:** Existen técnicas de aprendizaje automático para detectar valores atípicos mediante Machine Learning.

No obstante, es necesario destacar que la detección de valores outliers depende del dominio de los datos e implica un conocimiento funcional de su naturaleza. Antes de aplicar las técnicas mencionadas anteriormente es necesario asegurarse de qué manera aplicarán sobre los datos objetivo y si el tratamiento será correcto.

Las plataformas de Big Data más conocidas constan de servicios para detectar valores atípicos, algunas de las herramientas incluyen:

- **AWS SageMaker**: Proporciona una serie de algoritmos de aprendizaje automático y detección de valores atípicos. [22]
- **GCP BigQuery ML:** Ofrece una función para la detección de valores atípicos basada en desviación estándar. [23]
- **Apache Spark MLlib:** Se trata de un conjunto de algoritmos creados para la detección de valores atípicos basados en el método del rango intercuartil. [24]
- **Hadoop Mahout:** Es una librería de aprendizaje automático que ofrece Hadoop con algoritmos para la detección de outliers. [25]y

3.3 Calidad del dato

Más allá de los riesgos y medidas de seguridad es necesario destacar la importancia del papel que juega la calidad del dato en su seguridad. Aunque a priori pueden parecer dos temas independientes ambos guardan relación.
Una gran cantidad de sistemas de Big Data y Data Warehouse utilizan cifrados de tipo **FPE (Format-Preserving Encryption)** para proteger determinados atributos de especial sensibilidad y utilidad. FPE es una técnica de cifrado que permite encriptar los datos manteniendo su formato original, es decir, el algoritmo a partir de un texto en claro produce una cadena cifrada de la misma longitud y formato a la original. []
Este proceso puede verse afectado significativamente, hasta el punto de hacer imposible su aplicación, si la calidad de los datos no es la esperada. Algunos de los factores de formato que pueden afectar y deben cumplir todos los datos son:

- **Longitud de la cadena:** La longitud del texto original es de gran importancia en el cifrado FPE. Además de que esta tiene una estrecha correlación con la seguridad del cifrado, todos los datos del atributo deberán tener una misma longitud definida para poder aplicarlo. Por ejemplo, si se intentan cifrar los números de teléfono de un sistema que se alimenta de diversos orígenes y contiene datos del tipo:

  - TELEFONO_1: 660985674
  - TELEFONO_2: +34660985674
  - TELEFONO_3: 0034660985674

  La implementación sería inviable sin una estandarización o procesado previo, ya que los números deberían tener la misma longitud.

- **Caracteres especiales y formatos:** FPE tiene en cuenta los caracteres especiales y formatos de la cadena original, como espacios, comas, guiones… Por tanto, si un atributo que desea cifrarse presenta algún carácter especial en el texto en claro deberán contenerlo todos los datos del sistema para poder implementarlo. Por ejemplo, en caso de tener almacenados números de cuenta con los siguientes formatos:

  - NUMERO_1: 1111 2222 3333 4444
  - NUMERO_2: 1111-2222-3333-4444
  - NUMERO_3: 1111222233334444

  Como se observa, además de la longitud también varían los caracteres especiales, por ello no existiría un formato que el algoritmo pueda preservar una vez cifrado el texto.

Es aquí donde reside la importancia de la calidad del dato y, por lo tanto, de la fase de procesado, en la seguridad.
Los sistemas de Big Data pueden obtener datos del mismo tipo de un gran conjunto de orígenes.
Por ejemplo, el equipo de análisis de datos de una farmacéutica internacional tiene la necesidad de implementar un sistema de Big Data central. Este se

alimentará de los sistemas de todas sus plantas de producción (repartidas por el mundo) y recogerá información sobre los fármacos producidos e ingredientes utilizados (lo que puede ser altamente confidencial). El proceso de implementación de medidas de cifrado y seguridad puede ser altamente complejo e incluso inasequible si previamente no se procesan los datos en crudo para normalizarlos y darles un formato estándar. El equipo deberá tratar los identificadores de producto, identificadores de proveedores, cantidades de ingredientes… Para que sigan un mismo patrón.

En la actualidad, existen muchas empresas que deciden aplicar medidas de seguridad a posteriori, como por ejemplo cifrados, sobre sus sistemas de Big Data puestos en producción - en lugar de haber seguido una metodología de seguridad del dato desde el diseño –. Es habitual en la práctica que durante el análisis del proyecto se encuentren con que la calidad del dato no es la esperada y previo a la implementación de medidas sea necesario realizar un procesado de datos para estandarizar el formato.

Por este motivo, la calidad del dato es uno de los aspectos críticos para las consultorías o empresas tecnológicas enfocadas en aplicar medidas de seguridad a los sistemas de sus clientes.

En la actualidad, han surgido técnicas con un potencial muy elevado que pueden aplicarse en la mejora de la calidad del dato:

- **Machine Learning:** Sus propiedades inherentes hacen que sea una herramienta muy útil para mejorar la calidad del dato ya que es capaz de:
  - **Aprendizaje supervisado:** El machine learning supervisado es capaz de clasificar y etiquetar datos a partir del entrenamiento de modelos de clasificación que permitan clasificar datos correctos e incorrectos y etiquetarlos para su eliminación o corrección.
  - **Detección anomalías:** Es posible utilizarlo para la detección de valores atípicos y anomalías a partir de la identificación de patrones en los datos que sugieren que determinados valores son atípicos o errores.
  - **Imputación de datos faltantes:** Un modelo de regresión de machine learning sería capaz de predecir valores faltantes a partir de términos estadísticos.
- **Inteligencia Artificial:** El surgimiento de esta tecnología tan potente tiene una gran cantidad de aplicaciones todavía inexploradas. No obstante, permite realizar algunas tareas de interés para mejorar la calidad de la información:
  - **Procesamiento de lenguaje natural (NLP):** La Inteligencia Artificial puede comprender y analizar el lenguaje natural en los datos. Los algoritmos de lenguaje natural pueden identificar sinónimos y términos relacionados, lo que haría más sencillo encontrar patrones y tendencias.
  - **Minería de datos:** La IA puede utilizarse para analizar grandes volúmenes de datos y encontrar tendencias ocultas. De esta manera es posible identificar errores y anomalías.

4. Fase Almacenamiento

Una vez que los datos recolectados han sido procesados para normalizar su formato y conseguir el nivel de calidad necesario tiene lugar el almacenamiento para un posterior análisis de resultados.
Esta fase es crítica por el gran volumen de datos que manejan los sistemas de Big Data. Para solventar la falta de espacio y la escalabilidad los datos se suelen almacenar en un sistema distribuido como Hadoop HDFS, bases de datos NoSQL o en plataformas Cloud especialmente preparados para ello.
En los sistemas distribuidos, los datos son almacenados en bloques y distribuidos en varios nodos del clúster de almacenamiento. Cada nodo realiza copias de seguridad de sus datos para garantizar la disponibilidad en caso de fallo o catástrofe.
Un clúster de Big Data es un conjunto de servidores interconectados llamados nodos que trabajan juntos para almacenar y procesar de manera óptima cantidades elevadas de datos.
En general, el volumen de datos de los sistemas Big Data tiende a crecer debido a que no dejan de recolectarse datos nuevos y suelen conservarse los datos históricos. Aunque también existe la eliminación, que se comentará más adelante, la recolección de datos suele ser mayor que la eliminación lo que incrementa de manera irremediable el volumen con el paso del tiempo.
El crecimiento del volumen de datos puede ser un desafío para los sistemas de Big Data, es aquí donde reside la importancia del almacenamiento distribuido, es relativamente sencillo escalarlo de manera horizontal, que consiste en incrementar la capacidad del sistema añadiendo más nodos al clúster en lugar de incrementar la capacidad de almacenamiento y procesado de una sola máquina.
Algunas de las soluciones de almacenamiento para Big Data son las siguientes:

- **Hadoop Distributed File System (HDFS):** Es la solución de almacenamiento distribuido de Hadoop diseñada para manejar grandes cantidades de datos en nodos de un clúster. [27]
- **Amazon Simple Storage Service (S3):** Es un servicio de almacenamiento en nube proporcionado por AWS. S3 permite el almacenamiento de una gran variedad de tipos de objeto. La principal ventaja es la fácil integración con el resto de las herramientas del proveedor. [28]
- **Azure Blob Storage:** Azure ofrece esta herramienta de almacenamiento de objetos en la nube. Permite guardar y acceder a un gran volumen de datos no estructurados como imágenes, vídeos, archivos de audio, copias de seguridad… Permite la integración con el resto de los servicios de Microsoft Azure. [29]
- **Google Cloud Storage:** Es la solución de almacenamiento equivalente de Google Cloud. Está pensado para implementarse junto con Google BigQuery. [30]
- **MongoDB:** Se trata de un servicio de gestión de bases de datos NoSQL. Trabaja con colecciones y documentos de tipo JSON. Una de las principales ventajas de este tipo de base de datos es su flexibilidad, escalabilidad y capacidad de ingesta.

4.1 Riesgos

La fase de almacenamiento puede presentar varios riesgos que deben tratarse para garantizar la confidencialidad, integridad y disponibilidad de los datos almacenados.
A continuación, se incluyen algunos riesgos más comunes que pueden afectar al almacenamiento:

Pérdida de datos
Existen diversas formas que pueden ocasionar la pérdida o la corrupción de los datos almacenados en los sistemas de Big Data:

- **Fallos en el hardware:** Fallos en los discos duros, memoria RAM, procesadores, sistemas de refrigeración de los centros de procesado de datos (CPD) donde se encuentran los servidores pueden provocar la pérdida o la corrupción de los datos almacenamos.
- **Errores humanos:** Los errores cometidos por los desarrolladores o administradores del sistema, que suelen manejar usuarios con un elevado rango de privilegios, puede provocar la pérdida o corrupción de datos. El administrador, de manera accidental, podría borrar un archivo o tabla o mediante una consulta SQL introducida incorrectamente, que tenga un comportamiento inesperado, modificar o eliminar datos importantes.
- **Ataques informáticos:** Un ataque informático, como un programa maligno o virus, que aproveche una vulnerabilidad de los procesos o de la infraestructura del sistema Big Data puede borrar o dañar la información almacenada.
- **Desastres naturales:** Un terremoto, una inundación, un incendio o una tormenta pueden provocar la destrucción de los servidores físicos alojados en un centro de procesado de datos y la pérdida de la información si no se aplican las medidas necesarias para proteger el entorno de los agentes climatológicos.

Robo/Secuestro de datos
El ransomware es un tipo de malware, que bloquea la información almacenada en un sistema mediante cifrado, impidiendo su acceso y amenazando con destruirla o publicarla si las víctimas no acceden a pagar un rescate. [31]
En la actualidad, es cada vez más común la tendencia a amenazar a las víctimas con la fuga de información confidencial, si no cumplen con las peticiones, al ámbito público (internet).
Una vez que los datos han sido cifrados, los atacantes piden un rescate a través de una ventana emergente, mensaje o similar para advertir a la víctima de que la única forma de recuperar los datos y evitar su publicación es pagar el secuestro en un límite de tiempo.
Los sistemas Big Data pueden contener datos sensibles de diversos tipos: biológicos o genéticos, políticos, religiosos, relativos a la orientación sexual, de origen racial, etc. Que si son divulgados afectarían gravemente a la organización exponiéndola a daños reputacionales y multas por incumplimiento del GDPR, por no hablar del daño causado a los titulares de los datos.

Este tipo de malware está en constante evolución por parte de los ciberdelincuentes para explotar nuevas vulnerabilidades y propagarse por diversas vías como:

- **Servidores desactualizados:** Los sistemas y herramientas software están en constante actualización para mejorar su funcionamiento y corregir las brechas o vulnerabilidades de seguridad posibles. Un sistema desactualizado puede presentar graves vulnerabilidades y puertas traseras conocidas y publicadas que podrían ser usadas como vía de acceso al ransomware.
- **Equipos IoT vulnerables:** Muchos sistemas industriales y sensores conectados a internet no tienen las medidas básicas de seguridad. Hoy en día es cada vez más común encontrar sistemas de climatización, maquinaria, sensores… conectados a redes corporativas o Internet. Si estos dispositivos no cuentan con las medidas de seguridad requeridas pueden suponer una brecha en la seguridad.
- **Ingeniería social:** Aunque pueda sorprender es la vía más frecuente y fácil para el ciberdelincuente. Mediante el envío de un correo falso (phishing) con un enlace o un archivo que simula ser legítimo (desde una imagen, hasta un calendario, carpeta comprimida…) y que en realidad se trata de un malware que al descargarse infecta el equipo. También pueden aparecer en redes sociales y mensajería instantánea o SMS.
- **Aprovechar servicios expuestos a Internet:** Como puede ser el escritorio remoto, muchos de estos permiten abrir una puerta a un posible ataque debido a su mala o débil configuración comprometiendo el servidor de destino al que se pretende conectar. Los atacantes suelen usar credenciales de usuario típicas de los sistemas operativos como: administrator, admin, user o ssm-user. También utilizan ataques de fuerza bruta que prueban múltiples credenciales de forma automatizada. En caso de que la política de seguridad no sea lo suficientemente fuerte y no haya un bloqueo de usuario tras varios intentos fallidos de acceso los atacantes podrían llegar a acceder por prueba y error. Una vez dentro pasarán a instalar el software malicioso. [32]

Error en la configuración

Las bases de datos NoSQL son usadas típicamente en los sistemas de Big Data ya que permiten manejar grandes cantidades de datos no estructurados o semiestructurados de manera flexible. A diferencia de las bases de datos relacionales tradicionales, las NoSQL no requieren de un esquema fijo lo que las hace más adecuadas para este tipo de soluciones.

Aunque este tipo de bases de datos presenten ventajas respecto a las relacionales por sus características tienen un historial mucho más corto lo que hace que sus soluciones de seguridad puedan ser menos robustas y, al ser menos conocidas por desarrolladores y administradores, puedan presentar más vulnerabilidades a causa de errores de configuración debido a la falta de experiencia acumulada.

Las bases de datos no relacionales son relativamente nuevas y menos conocidas, lo que puede llevar a usar configuraciones por defecto, que no sean todo lo seguras que deberían, o implementaciones inadecuadas de las medidas de seguridad.

Algunas de las vulnerabilidades debidas a errores de configuración son:

- **Autenticación y autorización débiles:** Las bases de datos NoSQL, como MongoDB, pueden no tener activada la autenticación y autorización de manera predeterminada durante su creación, lo que las hace completamente vulnerables a cualquier atacante.
- **Acceso público:** Es habitual que bases de datos NoSQL se encuentren alojadas en servidores públicos o expuestas en internet sin las medidas de seguridad adecuadas. Esto las hace vulnerables a cualquier atacante que tenga intención de acceder.
- **Falta de cifrado:** Las bases de datos NoSQL pueden no tener habilitado el cifrado de manera predeterminada, aunque tanto MongoDB como Cassandra lo ofrecen. Esto expondría datos sensibles a los potenciales atacantes.
- **Inyección de código**: Al tratarse de un tipo de base de datos con un conocimiento menos extendido las buenas prácticas para evitar la inyección de código están menos estandarizadas respecto a las bases de datos relacionales, muchos lenguajes de programación web ofrecen librerías y funcionalidades para realizar las consultas SQL a servidores de datos de manera segura. La falta de conocimiento puede desembocar en un diseño vulnerable a inyección de código.

4.2 Medidas de seguridad

Muchos sistemas de Big Data basan su almacenamiento en bases de datos no relacionales ya que sus propiedades inherentes las hace más adecuadas para trabajar con grandes volúmenes de datos, no obstante, son menos robustas que las bases de datos relacionales debido a su madurez.
Por este motivo, en los sistemas de Big Data, es necesario asegurar el almacenamiento con herramientas de protección de datos.

Copias de Seguridad
Las copias de seguridad es una medida crítica para proteger los datos almacenados en el sistema Big Data.
Los datos almacenados suelen ser críticos para el desarrollo de las actividades de muchas empresas, una pérdida de datos puede tener un impacto económico irreversible. Las copias de seguridad permiten recuperar los datos en caso de pérdida o corrupción. Las copias se realizan de manera periódica y automatizada y se almacenan en ubicaciones seguras.
Es posible distinguir distintos tipos de copias de seguridad:

- **Completa:** Se conoce como copia total o full dump. Se hace una copia del sistema completo.
- **Incremental:** Se guardan los ficheros que se han modificado desde la última copia de seguridad.
- **Diferencial:** Este tipo realiza una copia de todos los ficheros que se han modificado desde la última copia total.
- **Selectiva:** Es posible realizar copias solo de unos ficheros seleccionados. [40]

La principal diferencia entre la copia incremental y la diferencial se encuentra en el proceso de recuperación. En la diferencial, solo es necesario recuperar la última copia total y la última diferencial, mientras que, con la incremental, es necesario recuperar día tras día desde la copia total.
El proceso está formado por diversas etapas:

- **Política:** La política deberá definir al detalle la estrategia a seguir a la hora de hacer las copias de seguridad para asegurar el correcto funcionamiento y recuperación. Para ello, será necesario analizar el sistema y la variabilidad de la información:

  - Cada cuanto varía la información.
  - Cuando se realizan los procesos de carga.

  En este punto debe definirse la frecuencia y periodicidad con que se va a realizar el proceso y el tipo de copia. No siempre es posible, por tema de recursos, o necesario, realizar copias totales, Una posibilidad es realizar una copia total el último día de la semana (domingo) y el resto de los días realizar copias diferenciales.
- **Configuración:** El servicio, herramienta de copias de seguridad o procesos deben ser configurados para especificar las rutas en las que se almacenarán los backups, la periodicidad de ejecución, alertas y todo lo necesario para cumplir con la política definida.
- **Ejecución**: Después de la configuración deberá ejecutar el proceso de manera manual y automatizada, en caso de que exista la planificación, para asegurar su correcto funcionamiento.
- **Almacenamiento:** Las copias de seguridad son almacenadas en una ruta accesible, para poder ser recuperadas en caso de necesidad, y segura, para evitar posibles ataques. Algunos ransomware son capaces de cifrar las copias de seguridad inclusive si no están protegidas debidamente, inutilizándolas y eliminando la posibilidad de restaurar el sistema al punto anterior al ataque.
- **Pruebas de recuperación:** Deben realizarse pruebas periódicas para asegurar que los datos se pueden recuperar correctamente en caso de desastre.

Por lo general, los proveedores de servicios en la nube ofrecen soluciones sencillas de copias de seguridad y redundancia geográfica para evitar la pérdida de datos en caso de desastre.

Actualizaciones
Los sistemas y herramientas software están en constante actualización para mejorar su funcionamiento y seguridad.
Con internet, cuando una vulnerabilidad es descubierta en alguna herramienta, funcionalidad o plataforma de trabajo se hace pública rápidamente. Los proveedores trabajan con celeridad para aplicar parches y actualizar el sistema, es aquí donde radica la gran importancia de mantener los sistemas actualizados, reducir al máximo posible los agujeros o puertas que puedan ser utilizados por los atacantes en amenazas emergentes.

Por otra parte, las empresas proveedoras de herramientas de Big Data, trabajan constantemente en la investigación y desarrollo de nuevas medidas de seguridad como algoritmos de cifrado o sistemas de autenticación.
Los sistemas de Big Data están sujetos a un gran números de estándares de seguridad, como por ejemplo PCI-DSS, HIPAA y GDPR. Al mantener los sistemas actualizados, es más probable que cumplan con los estándares de seguridad requeridos por estas regulaciones.
En definitiva, un sistema actualizado será más robusto frente a amenazas que uno obsoleto. Las actualizaciones son esenciales para minimizar riesgos de seguridad.

Cifrado en reposo
El cifrado de datos en reposo es una técnica que se utiliza para proteger los datos almacenados en dispositivos físicos o en la nube. El proceso consiste en un algoritmo que transforma los datos en claro a un texto cifrado ilegible para todos aquellos que no posean la clave de descifrado.
Existen diferentes formar de cifrar los datos almacenados en reposo en sistemas de Big Data:

- **Cifrado a nivel de archivo:** Es posible cifrar cada archivo de datos individualmente mediante herramientas de cifrado, la configuración de la base de datos o la configuración del sistema de archivos.
- **Cifrado a nivel de disco:** El disco o partición en el que se encuentran los datos puede ser almacenado configurando el sistema de ficheros o con herramientas de cifrado de disco.
- **Cifrado a nivel de base de datos:** Algunas bases de datos como Cassandra, Hbase y MongoDB ofrecen la opción de cifrar los datos en reposo. Cuando se configura este método la base de datos cifra y descifra los datos automáticamente cuando se accede a ellos.
- **Cifrado a nivel de columna:** En caso de que no sea necesario cifrar una tabla completa, es posible hacerlo solo con aquellas columnas que contengan datos sensibles como números de tarjeta de crédito, DNI…

A continuación, se muestran algunas soluciones de almacenamiento que proporcionan cifrado de los datos en reposo:

- **Amazon S3:** Ofrece la posibilidad de cifrar datos en reposo mediante el servicio Amazon S3 Server-Side Encryption que proporciona diferentes opciones de cifrado. [36]
- **Hadoop:** Proporciona una solución de cifrado mediante HDFS Transparent Encryption. Esta herramienta ofrece un cifrado end-to-end una vez configurado. Todos los datos son descifrados de manera transparente durante su acceso y cifrados cuando son escritos en los directorios de HDFS. Se puede considerar un cifrado intermedio entre nivel de base de datos y de archivo que ofrece un gran rendimiento para aplicaciones Hadoop. [37]
- **Google Cloud Storage:** Tiene la posibilidad de cifrar datos en reposo mediante Google Cloud Storage Encryption. Google usa varias capas de encriptación para proteger los datos [38]:
    - Sistemas de archivos distribuidos

  - Base de datos y almacenamiento de ficheros
  - Encriptación de dispositivos de almacenamiento en los centros de datos de producción de Google.
- **Azure Storage Service Encryption:** Azure Storage proporciona cifrado automático del lado del servidor. Los datos son cifrados y descifrados de manera transparente mediante AES-256 de manera similar al cifrado que utiliza BitLocker en Windows. Los datos se cifran por defecto y no puede deshabilitarse la opción por seguridad. [39]

No obstante, es importante comentar que existe un compromiso entre rendimiento y el cifrado de los datos en reposo. El cifrado de datos en reposo es importante para proteger los datos almacenados en sistemas de Big Data pero puede tener un impacto en el rendimiento.
El cifrado puede incrementar la latencia minimizando el rendimiento durante la lectura y escritura en el sistema. A menudo, el cifrado implica un consumo importante de recursos de computación, lo que puede aumentar los requisitos de recursos de computación, en este punto aflora la utilidad de cloud en infraestructuras Big Data, su elevado grado de escalabilidad.
Para minimizar el impacto en el rendimiento es recomendable el uso de técnicas de cifrado eficientes y valorar qué datos se cifrarán, ya que no todos los datos pueden requerir el mismo nivel de protección.
En la práctica, muchas organizaciones deciden aplicar - en la actualidad, a través de empresas tecnológicas - medidas de seguridad como el cifrado en sus sistemas de Big Data productivos y en funcionamiento. Antes de implementarlo, es importante analizar qué datos requieren ser cifrados y el impacto que tendría en el rendimiento ya que muchas entidades trabajan a diario explotando los datos del sistema y una bajada en la eficiencia puede convertirlo en una herramienta inútil para determinadas labores que requieran inmediatez.
El rendimiento es un punto clave en la utilidad de Big Data, para muchos clientes es el aspecto principal, si se ve afectado, pueden mostrarse reticentes a la hora de aplicar las medidas requeridas, llegados a este punto hay que buscar el equilibrio necesario entre rendimiento y seguridad del sistema.

Rotación de claves
La rotación de claves es un proceso que implica la modificación periódica de la clave de cifrado utilizada para proteger los datos en reposo. Este proceso ayuda a que las claves no se vean comprometidas por atacantes o puedan volverse débiles debido a los avances tecnológicos.
El funcionamiento es el siguiente:

- Se genera una nueva clave a partir de un algoritmo de generación de claves seguro.
- Se descifran los datos almacenados en el sistema con la clave antigua.
- Se cifran los datos almacenados en el sistema de Big Data con la nueva clave.
- Se actualiza la información de clave del sistema para que los datos almacenados futuros sean cifrados con la nueva clave.
- Se elimina la clave antigua.

El proceso de rotación de claves puede ser complejo en los sistemas de Big Data. Por lo que es necesario planificar el proceso cuidadosamente si se hace de manera manual y asegurarse de que se aplican las medidas adecuadas para proteger la clave y datos.
Las plataformas de Big Data presentan servicios de gestión de contraseñas que permiten la rotación de clave automática:

- **AWS Key Management Service:** Además de la gestión de contraseñas, este servicio proporcionado por Amazon permite habilitar la rotación de claves automática. [33]
- **Azure Key Vault:** Este servicio de Microsoft puede ser configurado para generar automáticamente una nueva versión de clave con una frecuencia especificada. La recomendación es girar la clave de cifrado al menos cada dos años para cumplir con las buenas prácticas. [34]
- **Google Key Management Service:** Esta solución de administración de claves de Google permite y recomienda la rotación de claves de manera periódica y automática como medida de seguridad para la encriptación simétrica. [35]

5. Fase Uso

Después del almacenamiento los datos se utilizan para generar información útil y, en muchos casos, tomar decisiones empresariales, estratégicas y operativas a partir de ellos.
El objetivo principal de esta etapa es aportar utilidad a los usuarios en sus actividades laborales dándoles un valor añadido.
Durante esta fase pueden realizarse diversas actividades entre las que se incluyen:

- **Análisis de datos:** Después del procesado, una vez que los datos han sido normalizados y dotados de un formato estándar para asegurar un nivel alto de calidad, es necesario realizar un análisis para generar información valiosa. Para ello, los usuarios pueden utilizar diversas técnicas:
  - Análisis estadístico para establecer patrones e indicadores de interés.
  - Minería de datos para extraer un conocimiento a partir de los datos almacenados.
  - Aprendizaje automático sobre el modelo de datos para extraer tendencias y realizar predicciones.
- **Visualización de datos:** Una vez que se ha realizado el análisis es importante presentar los datos de una manera clara y amigable para sacar el máximo valor posible a la información. La visualización puede ayudar a detectar errores, tendencias y patrones que tengan como consecuencia la mejora de la productividad, eficiencia o supresión de cuellos de botella.
- **Toma de decisiones:** Finalmente, la información obtenida en el análisis y la visualización es utilizada para tomar decisiones que afecten al negocio. Estas decisiones estarán cimentadas en la información extraída y en la calidad del análisis y pueden suponer:

  - Identificación de oportunidades de negocio.

- Optimización de procesos.
- Identificación de riesgos.
- Mejora de la eficiencia en las operaciones.

Pero también puede tener consecuencias negativas si el análisis no es correcto o si los datos tratados no son legítimos:

- Pérdida de oportunidades laborales.
- Desventaja frente a la competencia.
- Pérdida de reputación y confianza.

De manera simplificada, podemos generalizar tres usos distintos de un sistema de Big Data:

- **Transaccional:** Los sistemas transaccionales de Big Data son utilizados para procesar grandes cantidades de datos transaccionales en tiempo real. Suelen ser datos generados a través de operaciones diaria de una organización como: transacciones financieras, actualización de estado de registros, compras, ventas, registros de llamadas o de clientes, etc. En este uso, es muy importante asegurar la integridad de los datos y los requisitos ACID: debe asegurarse, mediante algoritmos de procesado de transacciones atómicas principalmente, que se mantiene la consistencia de los datos.
- **Data Warehouse:** Un sistema Big Data también puede ser utilizado como data warehouse, de la misma manera que los sistemas de base de datos relacionales tradicionales, para almacenar grandes cantidades de datos históricos y actuales. Permite a las organizaciones realizar análisis de datos con una estructura ordenada, coherente y consistente que facilita la interpretación de los datos, visualización y toma de decisiones. Para implementar un sistema de data warehouse se requiere una gran capacidad de almacenamiento y procesado. En este aspecto, big data tiene la ventaja de que es mucho más sencillo escalar un sistema de manera horizontal que vertical, como realizan las bases de datos tradicionales.
- **Machine Learning:** Los sistemas de Big Data para machine learning se utilizan para analizar grandes cantidades de datos y encontrar patrones, correlaciones y tendencias. Mediante algoritmos y modelos de aprendizaje automático son capaces de establecer patrones y realizar predicciones a partir de los datos. Para ello es necesario procesar una gran cantidad de datos en tiempo real. Se requiere de un conjunto diverso de herramientas de aprendizaje automático, visualización y análisis para interpretar los resultados.

5.1 Compromiso privacidad/pérdida de información

Durante la fase de uso es necesario aplicar medidas de seguridad para proteger los datos. Estas medidas pueden suponer la alteración o distorsión de los datos para reducir el riesgo de revelación de información en caso de pérdida o fuga. No obstante, al alterar los datos, estamos perdiendo información original y puede que utilidad.
Para tratar de entenderlo se muestra el siguiente ejemplo: Una empresa realiza el análisis de los datos de los pacientes diagnosticados de COVID-19 para establecer patrones y que el Ministerio de Sanidad pueda tomar las medidas acorde. Si para asegurar los datos y proteger la privacidad de los infectados se elimina la ubicación o residencia de estos pacientes será imposible realizar un análisis de la evolución de los contagios por zona geográfica o establecer "zonas calientes" con gran repunte de contagios.
De la misma manera, si se eliminara la edad de los pacientes ingresados en la UCI, o si se establecieran únicamente dos grandes rangos [0-50] y [51-100] para evitar reidentificaciones de pacientes, no sería posible conocer si hay algún parte de la población más vulnerable que la otra.
Uno de los principales problemas o compromisos que ofrece la seguridad de los datos es conseguir un buen balance entre privacidad y utilidad. Para ello, es necesario entender tres conceptos:

Riesgo de revelación
Es el riesgo de que los datos proporcionen información sensible o privada. El objetivo es minimizar al máximo el riesgo de revelación. (Apartado 5.2)

Pérdida de información
Es la cantidad de información que se pierde al aplicar las medidas de protección. Determina el error que se puede estar cometiendo al utilizar los datos protegidos en lugar de los originales en el análisis estadístico o aprendizaje automático. Es posible cuantificarla mediante el cálculo de diversos indicadores. Se pretende minimizar la pérdida de información.
Existen diversos indicadores para cuantificar la pérdida de información en caso de atributos numéricos. Dados dos conjuntos de microdatos A y B, uno con los datos originales y otro con los datos protegidos, con N registros y M variables, es posible considerar cada conjunto como una matriz y definir las siguientes funciones:

- **Mean Squared Error (MSE):** Mide la media de la diferencia elemento a elemento al cuadrado. Una característica de esta medida es que da más peso a los valores atípicos (outliers) al elevar al cuadrado la diferencia.

$$MSE(A,B) = \frac{1}{NM}\sum_{ij}(a_{ij} - b_{ij})^2$$

- **Mean Absolute Error (MAE):** Mide la media de la diferencia elemento a elemento en valor absoluto.

$$MAE(A,B) = \frac{1}{NM}\sum_{ij}|a_{ij} - b_{ij}|$$

- **Mean Relative Error (MRE):** Mide la diferencia elemento a elemento en valor relativo respecto al elemento original.

$$MRE(A,B) = \frac{1}{NM}\sum_{ij}\frac{|a_{ij} - b_{ij}|}{|a_{ij}|}$$

Este tipo de medidas suelen ser una buena estimación genérica sobre la pérdida de información. Pero pueden no ser útiles o precisas para usos específicos de los datos que vayan más allá del análisis estadístico.

Utilidad

Indica lo útiles que son los datos para ser analizados y extraerles información de valor. Se puede considerar inversamente proporcional a la pérdida de información. En este caso se busca maximizar la utilidad.

No es complejo observar que las medidas anteriores están muy relacionadas. Si se disminuye el riesgo de revelación se consigue más privacidad pero también aumenta la pérdida de información, y por consiguiente, disminuye la utilidad.

El objetivo es encontrar un buen balance entre estas medidas. Un buen método de protección es aquel que consigue un equilibrio entre riesgo de revelación y utilidad.

A pesar de ello en la práctica no es sencillo encontrarlo ni cuantificar las medidas anteriores. Por tanto, se puede afirmar que en la seguridad y privacidad de datos no existe la solución perfecta. Siempre es necesario jugar con el balance entre riesgo y pérdida de información.

Las medidas de seguridad pueden influir en distintos aspectos relevantes relativos a la fase de uso. Aunque todo depende del caso de uso concreto y del tipo de explotación que se haga del sistema, aunque las posibilidades son muy grandes, es posible destacar algunos ejemplos generales interesantes:

- En el caso del cifrado dinámico los datos se cifran y descifran según sea necesario durante las consultas. Cada usuario tiene su clave de cifrado y es necesario comprobar sus privilegios para cifrar y descifrar lo que sea necesario. Este cifrado y descifrado constante de cada usuario y a cada consulta al sistema puede tener un grave impacto en el rendimiento, especialmente en sistemas de Big Data con grandes volúmenes de datos y elevadas tasas de transmisión. Un sistema con un rendimiento muy bajo puede considerarse prácticamente inútil en la actualidad.
- En cuanto al persistente los datos se cifran y descifran en todo momento. En función de la consulta y el uso el rendimiento puede ser mejor o peor que el dinámico. No obstante, su aplicación también puede impactar a la hora de relacionar tablas en caso de sistemas data warehouse ya que no es posible enlazar campos en claro con campos cifrados, por ello, antes de aplicar cualquier medida sobre un atributo es necesario conocer a fondo el modelo de datos, analizar el alcance y valorar el riesgo. Por otra

parte, según el algoritmo de cifrado es posible perder funcionalidad a la hora de agregar datos (contabilizarlos o sumarlos).
- Otra opción frecuente es el uso de tablas temporales sobre las que se aplican medidas de seguridad a los datos que se desea proteger para que sean explotadas por los usuarios. El gran inconveniente de esta solución es que al crear una nueva tabla a partir de la original se está duplicando la superficie de ataque y disminuyendo por tanto la seguridad. Una posible solución es aplicar las medidas en una vista de manera que el contenido se crea y destruye dinámicamente cuando se consulta.

Aspectos como el rendimiento, enlaces entre tablas y la agregación llevan a muchas organizaciones a no aplicar medidas sobre sus sistemas productivos por el impacto negativo que estas pudieran tener en la explotación diaria del sistema. Muchas empresas relativas a la banca y seguros deciden asumir riesgos de seguridad para que la funcionalidad no se vea afectada.
En cada situación será necesario analizar a fondo el caso de uso para conocer los riesgos, implicaciones e impactos que podría tener la implantación de medidas en la explotación del sistema para los usuarios.

5.2 Riesgos

El uso de los datos de un sistema de Big Data puede ser muy amplio y variado. En muchas ocasiones puede surgir la necesidad de hacer públicos datos que contengan información privada muy útil para diversos campos como por ejemplo: estudios económicos, de población, sanitarios… También es cada vez más común por las administraciones públicas realizar campañas de transparencia que publican sus datos. Sin embargo, pueden contener información privada de muchos tipos relativa a los individuos o entidades a los que pertenecen que no se deben revelar por seguridad.
No obstante, no solo será necesario proteger los datos en caso de publicación, tambien es necesario aplicarles medidas de seguridad y privacidad a los datos no publicados ya que la fuga de determinados datos sensibles, por parte de atacantes externos o empleados, podría causar graves daños a los titulares.
Existen diversos riesgos asociados a la fase de uso que pueden amenazar la seguridad, privacidad de los datos. A continuación se muestran algunos de los más relevantes:

Riesgo de revelación
La revelación de información privada se produce cuando se puede obtener conocimiento nuevo sobre un individuo o registro del conjunto de datos.
El objetivo de cualquier atacante es obtener la mayor cantidad de información a partir de los datos sensibles. Además, este puede tener conocimiento previo que al compararlo con los datos, publicados o no, le ayude a sacar conclusiones o inferencias sobre los titulares.
No todos los atributos de un conjunto de datos o sistema presentan el mismo riesgo de revelación. Se pueden clasificar de la siguiente manera:

- **Identificadores:** Son aquellos atributos que identifican de manera unívoca y por sí solos a un individuo. Número de seguridad social, DNI…

- **Cuasi-identificadores:** No identifican por sí solos a un individuo salvo que sean combinados entre ellos o con información extra del atacante. Edad, código postal, fecha de nacimiento, profesión…
- **Confidenciales:** Atributos sensibles para el individuo Por ejemplo: enfermedad, salario, afiliación política…
- **No confidenciales:** No contienen información sensible sobre el individuo. [45]

.

En la práctica, no siempre es posible clasificar los atributos en un solo grupo de manera sencilla.

Es importante remarcar que no es suficiente con proteger únicamente los atributos identificadores. Aunque pueda parecer que unos datos sin identificadores son anónimos, no lo son. Un atacante puede llegar a obtener información sobre un individuo a partir de atributos cuasi-identificadores si son combinados de manera inteligente.

Un estudió determinó que es posible identificar de forma única el 87,1% de la población de Estados Unidos mediante tres sencillos cuasi-identificadores combinados: sexo, fecha de nacimiento y código postal. [46]

Es posible distinguir entre dos tipos de revelación:

- **Revelación de atributo:** Se produce cuando un atacante puede adquirir información nueva respecto al valor de un atributo de un individuo o registro. Puede tratarse del valor completo o del aumento de la precisión de la información anterior. Por ejemplo, conocer una edad exacta a partir de un intervalo.
- **Revelación de identidad:** Se produce cuando el atacante puede reidentificar de forma inequívoca el registro que corresponde a un individuo o titular. Esta particularidad en la revelación está relacionada con el riesgo de reidentificación.

Será necesario analizar a fondo cada escenario para determinar qué atributos son útiles y sobre cuales es necesario aplicar medidas de seguridad.

Riesgo re-identificación

El riesgo de reidentificación se refiere a la posibilidad de que un individuo sea identificado a partir de unos datos anonimizados. Aunque se apliquen medidas para preservar la privacidad de los datos, existe la posibilidad de que el proceso de anonimización sea revertido y asociar los datos con el individuo titular. Este riesgo puede tener graves implicaciones legales y éticas a causa de su gran impacto en el derecho a la intimidad.

Existen diferentes técnicas con las que se puede llevar a cabo la reidentificación de datos anonimizados. A continuación se muestran algunos de los principales:

- **Enlace de datos:** Si el atacante tiene acceso a múltiples conjuntos de datos con información similar, sería posible identificar a un individuo mediante el enlace y la combinación de los datos anonimizados con los conjuntos a los que tiene acceso. Por ejemplo, un conjunto de datos anonimizado que contiene información demográfica como edad y género, y otro conjunto también anonimizado con información sobre transacciones

financieras y sus importes. Un atacante podría enlazar los datos a partir de las características comunes como la edad y si se encontrara una combinación única inferir la identidad de la persona.

- **Datos auxiliares:** Aunque nos sorprenda, casi siempre es posible encontrar algo de información sobre un individuo que está disponible públicamente en internet o redes sociales. Un ciberdelincuente podría utilizar esa información pública para vincularla con los datos anonimizados e identificar individuos. Por ejemplo, combinando información disponible en Twitter o LinkedIn sería posible asociar los datos a personas específicas.
- **Ataques de correlación:** Implican comparar datos anonimizados con conjuntos de datos de referencia o bases de datos externas para identificar patrones que permitan la reidentificación.
- **Reversión de la generalización:** Aunque los algoritmos de generalización se aplican para proteger la privacidad, un atacante podría llegar a vulnerarla si tiene información extra. Por ejemplo, un conjunto de datos anonimizado que contiene información de ingresos anuales agrupados por rangos: “menos de 10.000€”, “10.000€-20.000€”, etc. Si el atacante tiene información relativa al sector o puesto de trabajo de un individuo puede estimar los ingresos anuales y llegar a identificarlo.

Aunque se ha observado que puede producirse reidentificación si hay revelación de algún atributo identificador, como DNI o similar, también se puede realizar a partir de los valores de los atributos cuasi-identificadores.
La unicidad de valores cuasi-identificadores es un problema más común de lo que se pueda pensar. En un conjunto de microdatos en el que aparecen cuasi-identificadores con valores únicos puede ser sencillo llegar a reidentificar al titular si el atacante tiene algo de información extra.
No obstante, el riesgo de reidentificación es difícil de calcular ya que no es posible cuantificar la cantidad de conocimiento que puede tener un atacante sobre algún individuo en concreto. Aun así, es posible aplicar medidas de seguridad que garanticen algunas propiedades en los conjuntos reduciendo riesgos.

5.3 Medidas de seguridad

Existe una gran cantidad de medidas para proteger la privacidad de los datos durante la fase de uso. Es posible destacar dos tipos de métodos de enmascaramiento:

- **Métodos perturbativos:** Modifican los datos originales para ocultar la información sensible mediante su distorsión. Este tipo de métodos tienen el inconveniente de que introduce errores en los datos, es decir, en ellos hay información incorrecta. Se consideran perturbativos los siguientes: ruido aditivo, ruido multiplicativo, rank swapping, microagregación, etc.
- **Métodos no perturbativos:** No alteran o modifican los datos introduciendo información errónea. En su lugar, lo que hacen es reducir la precisión, el detalle o incluso eliminar algunos valores. No introducen errores en el sentido de que la información de los datos protegidos no es

incorrecta aunque sí puede tener menos detalle. Es posible destacar: la generalización, supresión, etc. [47]

A continuación se desarrollan algunas de las medidas más relevantes:

Cifrado

El cifrado puede aplicarse de diversas maneras a lo largo del ciclo de vida del dato para proteger distintos aspectos de la información. En este apartado se considera el cifrado para proteger la confidencialidad y privacidad durante la fase de uso.

En general, existen dos tipos de cifrado:

- **Cifrado dinámico:** Los datos son cifrados y descifrados según la necesidad y privilegios del usuario durante la consulta o transmisión. El cifrado se aplica de manera temporal, es decir, los datos en reposo no permanecen encriptados. Aunque depende de la implementación, el proceso de cifrado, desde la consulta por parte de un cliente de analítica o similar a una base de datos hasta la recepción, podría tener la siguiente estructura:
    1. Envío de la consulta: El cliente envía la consulta al sistema de Big Data para acceder a los datos.
    2. Procesado de la consulta: El sistema recibe la consulta y la procesa para determinar qué datos pretenden ser accedidos.
    3. Evaluación de privilegios: Se evalúan los privilegios y roles asociados con el usuario para verificar si puede ver los datos en claro o si deben ser cifrados.
    4. Decisión de cifrado: En función de los privilegios, se decide si se cifran o no los datos consultados.
    5. Acceso a la clave de cifrado: El sistema accede a la clave de cifrado correspondiente. Puede ser generada dinámicamente o recuperada de una ubicación segura. Para mejorar el rendimiento, en los casos en que se realizan muchas consultas que requieren de cifrados y descifrados constantes, algunos sistemas almacenan las claves en caché para agilizar el proceso, no obstante, esto puede ser altamente peligroso porque podrían llegar a ser accedidas por un atacante.
    6. Cifrado de datos (si corresponde): Utilizando la clave, los datos necesarios para satisfacer la consulta son cifrados si hace falta.
    7. Envío de datos: Los datos son enviados al cliente.
    8. Recepción de los datos: El cliente recibe los datos, ya sea en claro o cifrados.
    9. Descifrado (si corresponde): Si los datos se enviaron cifrados, el cliente utilizará la clave para descifrarlos.

  La principal ventaja de este tipo es que permite una manipulación más eficiente de los datos ya que no es necesario cifrarlos y descifrarlos constantemente durante su almacenamiento. En oposición, la mayor limitación es que los datos en reposo pueden estar en claro y ser visibles si se produce un acceso no autorizado. Un aspecto clave en el cifrado dinámico es que es necesario anticipar los casos de usos que se va a dar al sistema para aplicar las restricciones pertinentes sobre cada rol tabla o

vista, lo cual es imposible por distintos aspectos. Por ejemplo, podría restringirse un campo pero que a la hora de ser seleccionado se cambie el nombre con un alias y se muestre la información en claro.

- **Cifrado persistente:** Los datos se cifran y permanecen cifrados tanto en reposo como durante el uso. Los datos se mantienen encriptados incluso cuando no están siendo utilizados activamente. El proceso sería bastante similar con la salvedad de que no es necesario realizar ninguna decisión ya que los datos se cifran y descifran siempre.
  La manera más sencilla y extendida de realizar el cifrado persistente es a partir de **UDF (User-Defined Function)**, funciones definidas por el usuario que a partir de unos parámetros de entrada son capaces de aplicar un algoritmo de cifrado sobre el campo o valor deseado. Suponiendo que existe una tabla llamada “Clientes” con columnas como “Nombre”, “Correo electrónico” y “Número de tarjeta” en la cual se desea cifrar de manera persistente el número de tarjeta de crédito para proteger la información confidencial: Inicialmente se definen las funciones ***EncryptCreditCardNumber*** y ***DecryptCreditCardNumber*** para cifrar y descifrar respectivamente con el algoritmo deseado. A continuación, se añade al proceso de carga la función de encriptado para cifrar los datos antes de ser insertados en las tablas de destino.

```
INSERT INTO Clientes (Nombre, CorreoElectronico,
NumeroTarjetaCredito)
VALUES ('Juan Pérez', 'juan@example.com',
EncryptCreditCardNumber ('1234567890123456'))
```

Para consultar los datos, será necesario usar la función de descifrado para mostrar los datos en claro (después de comparar el usuario, privilegios y todo lo necesario):

```
SELECT Nombre, CorreoElectronico,
DecryptCreditCardNumber(NumeroTarjetaCredito) AS
NumeroTarjetaCredito
FROM Clientes
```

Es necesario destacar que esta solución puede implementarse de muchas maneras, otra posibilidad, es usar las UDF para acceder al KMS (Key Management Service) y después a la aplicación de cifrado, la cual contendrá la lógica necesaria para cifrar o descifrar datos en función de distintos criterios como la tabla de destino, los permisos del usuario, etc.
Esta solución tiene la ventaja de que proporciona una mayor protección de datos incluso si se produce un acceso no autorizado. Además protege la confidencialidad de los datos en reposo. Por el contrario, puede tener impacto en el rendimiento ya que los datos, como normal general, deberán ser cifrados y descifrados en cada uso.

Cómo se ha observado en el apartado 5.1, la decisión de aplicar el cifrado sobre un sistema de Big Data puede tener grandes implicaciones en la utilidad. Los algoritmos de cifrado dificultan la agregación de información, el enlace entre tablas y la búsqueda de valores entre otras cosas.

A continuación, se presentan algunos algoritmos de cifrado que preservan algunas propiedades interesantes que facilitan el uso:

- **DTE (Deterministic Encryption):** Es una técnica criptográfica que cifra los datos de manera determinista, es decir, un mismo texto en claro siempre generará el mismo texto cifrado.

  $$\text{Si } a=b \rightarrow E(a)=E(b)$$

  A diferencia de algún cifrado tradicional que cada ejecución podría generar una salida distinta para una misma entrada. Tiene la gran ventaja de facilitar la búsqueda de datos cuando se realizan búsquedas o comparaciones. Además, los datos cifrados podrían indexarse y enlazarse entre tablas en caso de que fuera necesario. Por último, asegura un mayor rendimiento y menor sobrecarga ya que la lógica del algoritmo es más sencilla y no requiere del almacenamiento de ningún vector de inicialización o información adicional para cifrar. Por otra parte, es necesario destacar que no ofrece el mismo nivel de seguridad que el resto de cifrados tradicionales. El DTE se utiliza en escenarios donde se prioriza la búsqueda eficiente y el análisis de datos cifrados.
- **OPE (Order-Preserving Encryption):** Es una técnica utilizada para mantener el orden relativo entre los valores cifrados. El OPE asegura que el orden de los valores cifrados es el mismo que el de los valores en claro.

  $$\text{Si } a>b \rightarrow E(a)>E(b)$$

  Es útil en casos en los que es necesario realizar clasificaciones o comparaciones de datos sin tener que revelar el contenido real. No obstante, tiene el inconveniente de que puede ser vulnerable a filtraciones de información, revelación y reidentificación, si alguien conoce el orden de algún registro podría llegar a identificar al individuo al cual pertenecen los datos o inferir nueva información. Su uso debe evaluarse al detalle en función de los requisitos de seguridad y necesidades de uso.
- **Homomorphic Encryption:** Es una técnica criptográfica avanzada que permite realizar operaciones matemáticas directamente en datos cifrados sin necesidad de descifrarlos previamente. Permite realizar cálculos sobre datos encriptados y obtener resultados equivalentes a los que se obtendrían sobre los datos en claro. Al realizar cálculos directamente sobre los datos cifrados, minimiza el riesgo de revelación de información. No obstante, las operaciones sobre datos cifrados son computacionalmente costosas y requieres una mayor capacidad de procesamiento, lo que puede afectar al rendimiento. Además, pueden existir limitaciones en los tipos de operaciones lo que restringiría la utilidad y capacidad de análisis de los datos. Es necesario tener en cuenta que se trata de una técnica de cifrado compleja y relativamente joven.
- **FPE (Format-Preserving Encryption):** Este algoritmo permite cifrar datos de manera que el formato y estructura del texto en claro se mantengan en el texto cifrado. Es decir, los datos cifrados conservan la misma longitud y caracteres que los originales. La ventaja de esta técnica es que al mantener el formato puede tener menor impacto en el sistema

a la hora de implantarlo, ya que se evitan posibles cambios de adaptación en la longitud o tipología de los campos. En contrapartida, la necesidad de preservar el formato puede limitar las transformaciones que se pueden aplicar a los datos disminuyendo ligeramente la seguridad en comparación con los cifrados tradicionales.

- **TDE (Transparent Data Encryption):** TDE permite cifrar automáticamente los archivos de datos de una base de datos de manera que los datos en disco estén protegidos a partir de una clave maestra de cifrado, generada y gestionada por el sistema de gestión de base de datos (DBMS) o una solución de cifrado externa. Cuando se aplica en Big Data, el proceso es transparente para los usuarios y aplicaciones. Las operaciones de cifrado y descifrado se realizan de forma automática por el DBMS durante la lectura y escritura. Tiene la gran ventaja de que los usuarios no necesitan realizar ninguna acción adicional para cifrar o descifrar datos ya que el TDE se encarga de ello de manera transparente. Es una de las técnicas más extendidas y recomendadas en la actualidad, ya que elude de responsabilidades a las aplicaciones y usuarios, evitando los errores de tipo humano o de programación. Por otra parte, la gestión de claves es muy importante para TDE. Se debe prestar especial cuidado a las claves, así como a las políticas y procedimientos de gestión para evitar la pérdida de datos. [48]

Tokenización

La tokenización, es una técnica de seguridad que se utiliza para proteger los datos sensibles mediante el remplazamiento por valores no sensibles llamados tokens. Esta técnica es típicamente utilizada en pasarelas de pago con tarjeta. Como es sabido, todas las transacciones deben ser lo más seguras posibles debido a que los datos involucrados en este proceso son críticos y altamente vulnerables en caso de ataque. Por este motivo, para que el número de tarjeta no sea legible por parte de los atacantes es reemplazado por un token que no contiene información sensible. [49]

Existen dos opciones para implementar la tokenización de información:

- **Vault Tokenization:** En esta solución, se utiliza un servidor centralizado para generar y gestionar los tokens. Los sistemas de gestión de tokens (Token Management Systems) centralizan el control y la administración de tokens y su mapeo con datos sensibles. A partir de este mapeo se realiza la detokenización para reemplazar la información no sensible a los datos originales. El proceso se puede describir a partir de los siguientes pasos:
    1. La organización envía los datos sensibles a la bóveda.
    2. La bóveda convierte los datos originales en datos no sensibles (tokens) y almacena el mapeo correspondiente en la base de datos.
    3. Los tokens son devueltos a la organización. Que solo almacena esta información.
    4. Cualquier usuario que requiera de los datos originales deberá autenticarse contra el sistema de la organización para acceder al token respectivo.
    5. El usuario envía la información no sensible a la bóveda, la cual convierte en los datos originales a partir del mapeo.

6. El usuario recibe los datos originales.

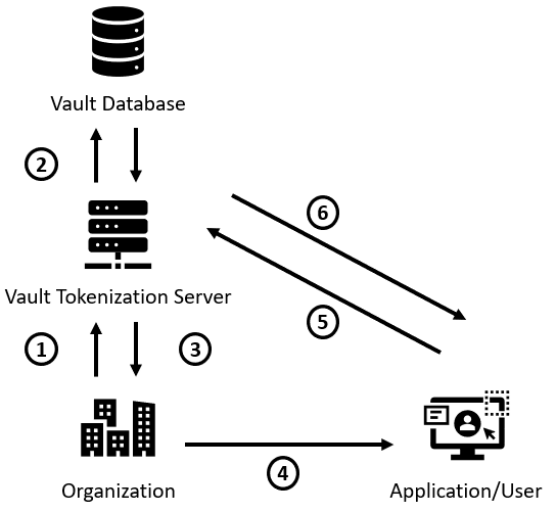

Figura 3: Vault Tokenization.

- **Vaultless Tokenization:** A diferencia de la anterior, la tokenización sin bóveda no requiere de una base de datos donde almacene el mapeo de los tokens con su texto en claro. En lugar de eso, utiliza algoritmos criptográficos para convertir los datos sensibles en tokens. Para recuperar el valor original es suficiente con proporcionar el token y el algoritmo realiza la detokenización.

Hay que tener en cuenta que cada sistema de tokenización puede tener características y enfoques específicos y hay que valorar cada situación y las implicaciones de la implantación.

Hash

Una función hash de tamaño *n* es una función que a partir de una cadena de entrada de un tamaño arbitrario devuelve una cadena fija de tamaño *n,* denominado hash o resumen. Una función hash clásica es determinista y unidireccional, es decir, no puede obtenerse el valor original a partir del valor resumen.

Existen diversos tipos de técnicas de implementación de hashes:

- **Classic Hash:** La función resumen es aplicada directamente sobre el dato original sin agregar ningún valor adicional. Es el enfoque más simple de todos pero también el menos seguro ya que son vulnerables a ataques de fuerza bruta y utilización de tablas de arcoíris.
- **Salt Hash:** Se agrega un valor aleatorio “salt” al dato original antes de aplicar la función hash. Cada dato tiene un salt único, lo que significa que incluso dos datos iguales tendrían dos hashes distintos. En este caso, el riesgo de reidentificación dependerá principalmente de las propiedades de la salt: Una salt de longitud más corta será más vulnerable a ataques de fuerza bruta y por tablas de arcoíris. Con el tiempo, la capacidad de computación y nuevos algoritmos de ruptura de hashes hacen necesario la generación de claves cada vez más largas. [50]

- **Pepper hash:** Se utiliza un valor secreto “pepper” junto al original antes de aplicar la función hash. Se trata de un valor fijo y que debe mantenerse en secreto. A diferencia del salt, el pepper se aplica de la misma manera a todos los datos, por lo que si dos datos son iguales sus hashes también lo serán.

Aunque las funciones hash pueden aportar varias ventajas como: validación de la integridad de los datos, protección de contraseñas, eficiencia en la búsqueda, etc. La anonimización de datos sensibles en la fase de uso es una de las aplicaciones más extendidas. En lugar de utilizar los datos originales, se aplica una función hash para generar un valor único e irreversible que representa al dato original. Esto permite realizar análisis sin tener que revelar información personal sensible.

Supresión
Consiste en la eliminación completa de los datos personales de un conjunto de datos. Por ejemplo, se pueden eliminar nombres, direcciones, número de identificación persona, etc. La gran implicación de esta medida es su impacto con la utilidad de los datos, suprimir atributos puede disminuir el potencial de análisis y la extracción de información interesante de los datos almacenados.
Por otra parte, no es suficiente con eliminar aquellos atributos identificadores ya que según la información de la que disponga un atacante podría llegar a reidentificar a una víctima a partir de cualquier atributo o conjunto de atributos cuasi-identificadores.

Generalización
La generalización consiste en reemplazar un valor por otro más general. La generalización puede ser de diversas manera dependiendo del tipo de atributo que estemos tratando y de la información que tengamos sobre ellos:

- **Atributo categórico:** Si se trata de un atributo categórico es posible hacer uso de jerarquías o agrupación de valores.
- **Atributo numérico:** En caso de atributos numéricos como la edad es posible cuantificarlos en intervalos.

La generalización es un método que se aplica con frecuencia para conseguir k-anonimidad. En la práctica, conseguir una buena generalización es difícil, una generalización óptima es aquella que consigue cumplir un modelo de privacidad deseado (como por ejemplo k-anonimidad) reduciendo al máximo la pérdida de información.

Agregación
Consiste en la agregación de los datos y presentación de los resultados como estadísticas, lo que ayuda a proteger la privacidad de los usuarios. Por ejemplo, se pueden agregar los datos por área geográfica, departamento, rango de edad, etc. Es una de las técnicas más extendidas en sistemas de Data Warehouse. Es habitual que los procesos de carga finalicen en tablas agregadas que, mediante la gestión de roles y privilegios, son las únicas que pueden explotar los usuarios mediante sus herramientas de analítica o visualización. De esta manera se preservaría la privacidad y confidencialidad de los datos.

Ruido

El uso de ruido para perturbar los datos es una práctica muy extendida. Existen diferentes estrategias a la hora de utilizar ruido en privacidad de datos que se pueden aplicar en diversos escenarios. Generalmente, se intenta ajustar las características del ruido de manera que el resultado final preserve algunas propiedades respecto a los datos originales. Los dos casos más comunes son:

- **Ruido aditivo:** Se trata de sumar ruido (ε) a los datos originales X para obtener unos datos protegidos X':

$$X' = X + \varepsilon$$

  Un caso sencillo es utilizar una distribución normal $N(\mu,\sigma^2)$ para modelar el ruido ε. Para proteger la variable V_j utilizamos $N(\mu_j,\sigma_j^2)$ de manera que $\mu_j=0$ y $\sigma_j^2= p \cdot \text{Var}(V_j)$, siendo p una constante que determina el nivel de perturbación. Es decir, el ruido presenta valores alrededor de 0 (ya que la media es 0) y con una varianza proporcional a los valores originales. Se aplica el proceso a cada variable de forma independiente.

- **Ruido multiplicativo:** De manera análoga, en este caso se calcula X' como el producto del ruido por los datos originales:

$$X' = X \cdot \varepsilon$$

Una ventaja del ruido multiplicativo respecto al aditivo es que la perturbación introducida por el ruido (el error en los datos protegidos) es proporcional al valor que se aplica. Una variable con valores pequeños tendrá una perturbación menos que una con valores mayores.

Generación de datos sintéticos

Una técnica totalmente distinta a las anteriores es la generación de datos sintéticos. Con ella, se pretende generar un conjunto de datos artificiales que reemplazan los datos originales. En cierta manera se pueden considerar como datos simulados. Pueden distinguirse dos alternativas:

- **Datos totalmente sintéticos:** Todos los datos protegidos son generados de forma sintética.
- **Datos parcialmente sintéticos:** Se generan de forma sintética algunas variables de algunos registros (los que conllevan un mayor riesgo de revelación).

Los datos se generan de forma aleatoria manteniendo ciertos indicadores estadísticos, modelos o relaciones de los datos originales. Esta técnica suele realizarse mediante el entrenamiento de modelos de datos y aprendizaje automático con sistemas de Machine Learning, aunque la irrupción de la Inteligencia Artificial también podría ser tremendamente útil para mejorar la precisión de los datos sintéticos.

En relación con el riesgo de revelación, los datos totalmente sintéticos se consideran muy seguros. A priori, no es posible la reidentificación dado que todos

los datos son generados y por tanto no hay presencia de originales. A día de hoy, no existen ataques conocidos a este tipo de datos. No obstante, existe una gran dificultad en la práctica de conseguir modelos precisos de determinados tipos de datos.
A diferencia de los datos totalmente sintéticos, los parcialmente sintéticos sí presentan riesgo de reidentificación y revelación de forma similar a los métodos perturbativos o no perturbativos anteriores, ya que sí contienen información original parcial.

5.4 Modelos de privacidad

Medir la privacidad o anonimato de un conjunto de datos no es sencillo y muchas veces depende de factores externos como el conocimiento previo del atacante. Sin embargo, es posible utilizar enfoques o técnicas que aseguren ciertas propiedades de los datos para proteger la información personal y sensible en los sistemas de Big Data. A continuación se destacan algunas de las más importantes:

k-anonimidad
Es un modelo de privacidad muy popular definido de la siguiente manera: Un conjunto de datos X cumple k-anonimidad respecto a los cuasi-identificadores de X, si cualquier combinación de valores cuasi-identificadores aparece como mínimo *k* veces. Es decir, que cada registro sea indistinguible entre al menos *k-1* del resto de registros del conjunto.
El conjunto de registros indistinguibles recibe el nombre de clase de equivalencia y puede entenderse como un conjunto de anonimato.
Como se ha comentado con anterioridad, la unicidad de valores cuasi-identificadores de un conjunto de datos lo hace altamente vulnerable al riesgo de reidentificación. Esto es lo que pretende evitar k-anonimidad al asegurar la existencia de k registros indistinguibles a partir de cualquier combinación de cuasi-identificadores.
Esta propiedad se basa en la idea de que al tener grupos más grandes de individuos con atributos similares es más difícil identificar a una persona en particular dentro del conjunto.
Para lograrla, es posible aplicar medidas como la generalización o supresión de ciertos atributos (comentadas anteriormente) para agrupar individuos en clases o categorías más amplias. De esta manera, se oculta la información sensible y se protege la privacidad de los individuos dentro del conjunto de anonimato.

***l*-Diversidad**
Esta propiedad busca garantizar un nivel de diversidad mínimo en el atributo confidencial de cada clase de equivalencia.
Un conjunto de datos k-anónimo, cumple l-diversidad, si cada clase de equivalencia o conjunto de anonimato contiene al menos l valores bien representados para el atributo confidencial.
En esta definición un punto clave es el concepto bien representado, que puede interpretarse de diversas maneras:

- **Valores distintos:** Se requieren *l* valores distintos del atributo confidencial en cada clase de equivalencia.
- **Entropía:** Se requiere que los valores confidenciales en cada clase de equivalencia presenten:

$$H(Q) \geq \log_2 (l)$$

Donde H(Q) es la entropía de la clase de equivalencia Q y S_Q el conjunto de valores.
La entropía se define como:

$$H(Q) = -\sum_{s \in S_Q} p(Q,s) \log_2 (p(Q,s))$$

Donde *p(Q,s)* es la fracción de registros con valores confidenciales iguales a *s* en *Q.*

_t_-Proximidad
En algunos casos la aplicación de l-diversidad puede resultar difícil o contraproducente. En cierta medida, l-diversidad asume una distribución uniforme de los valores del atributo confidencial. No obstante, en caso de tener un atributo que determina 3 rangos de salario donde el 98% de registros tiene el rango 1, el 1,2% tiene el rango 2 y el 0,8% tiene el rango 3. En este caso la aplicación de la l-diversidad es complicada e incluso podría inducir a un aumento de la revelación de atributo.
La idea de la t-proximidad es hacer que la distribución de los valores confidenciales en una clase de equivalencia sea la misma a la distribución de esos valores en todo el conjunto.
Un conjunto de datos cumple *t*-proximidad si para todas las clases de equivalencia, la distancia entre las distribuciones de valores de los atributos confidenciales en la clase y la distribución del atributo en todo el conjunto de datos es mejor o igual a *t*.

Privacidad diferencial
La privacidad diferencial es un modelo que ha ganado popularidad en los últimos años. En privacidad diferencial se considera que un conjunto de datos es privado, si al añadir, eliminar o modificar un solo registro del conjunto, no afecta al resultado del análisis que se hace sobre esos datos. Para ello, se considera que una consulta a una base de datos debería dar el mismo resultado (o uno similar) aunque se borre o añada un registro.
Considerando dos bases de datos D_1 y D_2 que difieren en un solo registro. Se realiza una consulta *f* a una base de datos *D* y obtiene un resultado *f(D)*. El objetivo es encontrar una función K_f que, sustituyendo a *f,* garantice privacidad diferencial. Es decir K_f es una versión segura de la consulta *f.* Intuitivamente se puede pensar en K_f como f más un ruido aditivo:

$$K_f\,(D) = f(D) + ruido$$

Una función K_f para una consulta f, proporciona ε-privacidad diferencial si para todas las bases de datos D_1 y D_2 que difieren en un único registro:

$$P[K_f(D_1) \in S] \leq e^{\varepsilon} \cdot P[K_f(D_2) \in S]$$

Lo que significa que la probabilidad de obtener un resultado para un conjunto de datos original y para otro alterado es muy similar y queda limitado por el umbral ε. Cuanto menor sea ε más parecidas serán las dos respuestas. Si ε = 0 la privacidad es máxima. [51]
El objetivo de la privacidad diferencial es hacer que las diferencias entre las consultas sean lo menor posible y que a partir de esa diferencia no se pueda inferir información sobre el registro eliminado en el conjunto de datos alterado.
Una de las técnicas más comunes para conseguir la privacidad diferencial consiste en añadir ruido a la respuesta de la consulta.
La manera más típica de definir o modelar es ruido es la conocida como el mecanismo de Laplace, es decir, a partir de variables aleatorias que tienen distribución de probabilidad de Laplace.
La cantidad del ruido agregado está determinado por la privacidad deseada, representada por ε. Cuanto menor sea ε, mayor será la privacidad, pero también se introducirá más ruido.
No obstante, esta técnica tiene una limitación que se puede explicar a partir del teorema de composición secuencial. La composición secuencial determina que dada una función o consulta K_1 que proporciona ε_1-privacidad diferencial, y otra función K_2 que proporciona ε_2-privacidad diferencial. La aplicación de ambas funciones o consultas proporciona $(\varepsilon_1 + \varepsilon_2)$-privacidad diferencial. De manera análoga, si se realiza la misma consulta *t* veces, dado a que cada la respuesta de cada una de ellas son sucesos independientes se obtiene:

$$\varepsilon = t \cdot \varepsilon_1$$

Esto limita de forma clara y precisa el número de consultas que se pueden realizar sobre un conjunto sin que la privacidad se vea en peligro. Se denomina **presupuesto de privacidad**.

Como conclusión, es importante destacar la diferencia entre k-anonimidad y privacidad diferencial. K-anonimidad se centra en anonimizar un conjunto de datos antes de su publicación para su posterior análisis. Por otro lado, la privacidad diferencial consiste en realizar consultas sobre unos datos, mediante un análisis predefinido, para que las respuestas no atenten contra la privacidad de los datos. Aunque la privacidad diferencial se pensó para sustituir a la k-anonimidad, esta todavía tiene un papel dominante sobre su sucesora ya que la implementación práctica de la privacidad diferencial puede ser muy compleja o inviable en según qué escenarios. Será necesario analizar el tipo de uso del sistema para escoger la solución que mejor se adapte a las necesidades.

6. Fase Eliminación

En el ciclo de vida del dato, la fase de eliminación se refiere al proceso de eliminar de manera segura y permanente los datos que ya no son necesarios. Esta fase es crucial para el cumplimiento de las regulaciones y normativas, así como para la gestión eficiente del almacenamiento y escalado del sistema.
La fase de eliminación implica varios pasos importantes:

- **Evaluación de la necesidad de la retención:** Antes de eliminar cualquier dato, es necesario analizar las obligaciones legales, en caso de existir, respecto a la retención por un periodo de tiempo específico. Esto puede variar en función del tipo de dato, el sector o la zona geográfica.
- **Definición de políticas de retención:** A partir del análisis anterior será necesario establecer políticas claras y documentadas sobre el periodo de retención de los datos por parte de la organización. Las políticas deben ser acordes a la legalidad y ajustarse a las necesidades de la empresa. La política debe estar respaldada por la gerencia y comunicada de manera correcta a todos los integrantes.
- **Identificación y clasificación de datos a eliminar:** Después de definir las políticas de borrado es necesario identificar y clasificar los datos susceptibles de ser borrados. Esto implica analizar y revisar los sistemas de almacenamiento, bases de datos, repositorios y copias de seguridad para identificar los conjuntos obsoletos o inútiles.
- **Eliminación segura:** Una vez se han identificado los datos que deben ser eliminados deben borrarse de manera segura y permanente a partir de métodos como la sobreescritura de datos, borrado seguro, etc. La reglamentación vigente obliga a las empresas a trabajar de manera proactiva en la seguridad de sus procesos.

6.1 Derecho de supresión

La llegada del Reglamento General de Protección de Datos ha dotado a los ciudadanos de una serie de deberes y derechos que puede ejercer sobre sus datos. Es necesario destacar el derecho al olvido, ya que está estrechamente relacionado con la fase de eliminación, y establece que todas las personas tienen derecho a la supresión de sus datos personales cuando ya no sean necesarios para los fines de recolección o cuando la persona haya retirado su consentimiento:

“1. El interesado tendrá derecho a obtener sin dilación indebida del responsable del tratamiento la supresión de los datos personales que le conciernan, el cual estará obligado a suprimir sin dilación indebida los datos personales cuando concurra alguna de las circunstancias siguientes:

a. los datos personales ya no sean necesarios en relación con los fines para los que fueron recogidos o tratados de otro modo;
b. el interesado retire el consentimiento en que se basa el tratamiento de conformidad con el artículo 6, apartado 1, letra a), o el artículo 9, apartado 2, letra a), y este no se base en otro fundamento jurídico;
c. el interesado se oponga al tratamiento con arreglo al artículo 21, apartado 1, y no prevalezcan otros motivos legítimos para el tratamiento, o el

interesado se oponga al tratamiento con arreglo al artículo 21, apartado 2;

d. *los datos personales hayan sido tratados ilícitamente;*
e. *los datos personales deban suprimirse para el cumplimiento de una obligación legal establecida en el Derecho de la Unión o de los Estados miembros que se aplique al responsable del tratamiento;*
f. *los datos personales se hayan obtenido en relación con la oferta de servicios de la sociedad de la información mencionados en el artículo 8, apartado 1."* [52]

Con base a ese artículo, las organizaciones deberán eliminar los datos de los usuarios si se cumple alguno de los supuestos anteriores. No cumplir con ello puede conllevar sanciones graves.

6.2 Riesgos

Existen varios riesgos asociados a la eliminación de datos. En caso de que no se realice el borrado de manera segura, si los datos fueran recuperados por un atacante, además de las sanciones administrativas y económicas, podría haber una pérdida de confianza por los clientes y la posible ventaja con los competidores del sector.
A continuación se desarrollan algunos de los riesgos principales:

Pérdida de datos irrecuperable
Si se eliminan datos que aún eran necesarios a causa de algún error y los procesos de copia de seguridad no se realizan de manera correcta podría perderse información de manera irremediable.
Pueden darse diversos escenarios que desemboquen en pérdida de datos irreparables:

- **Errores de configuración:** Fallos en la configuración pueden ocasionar la sobreescritura, la eliminación accidental o la corrupción de datos o copias de seguridad. Puede ocurrir si se establecen políticas de retención incorrectas o si se configuran mal las copias de seguridad.
- **Errores humanos:** El borrado accidental de datos o la ejecución de un comando que afecte la integridad de los datos puede significar la pérdida de datos. Estos errores pueden ocurrir durante la administración del sistema de Big Data.
- **Corrupción de datos:** Si los datos se corrompen por errores en los procesos de eliminación puede ser dificil o imposible recuperarlos en su forma original e íntegros.

Fuga de información
La fuga de información puede producirse por distintos orígenes. En este caso, si los datos no son eliminados de manera correcta y segura podrían ser accedidos por un atacante.
La eliminación incompleta de información en Big Data puede darse por varias razones:

- **Eliminación lógica y no física:** Algunos sistemas, eliminan los datos de forma lógica, es decir, únicamente marcan los registros como “eliminados” en lugar de borrarlos físicamente. Otros en cambio, marcan primero los registros para que los procesos de borrado ataquen únicamente los datos señalados. En los casos que solo se realiza borrado lógico, existe la posibilidad que un atacante pueda acceder a información que debería estar borrada.
- **Procesos de eliminación no completados:** Si los procesos de eliminación no se completan adecuadamente debido a errores en el proceso o interrupciones en el sistema podrían quedar datos sin eliminar. De esta manera, el atacante podría acceder a los datos.
- **Copias de seguridad no eliminadas:** Las copias de seguridad deben eliminarse de manera periódica y no guardarse más del tiempo requerido. Según la política de recuperación y copias de seguridad, es posible que las copias realizadas hace semanas dejen de ser útiles. En cada caso habría que analizar pros y contras y asegurarse de que el proceso elimina las copias inservibles.
- **Retención de archivos temporales:** Algunos sistemas generan archivos temporales para mantener la integridad de los datos. Si no se gestionan y eliminan correctamente es posible que puedan ser accedidos por un atacante.

Es importante considerar que la eliminación completa de datos en un sistema de Big Data es un proceso técnico complejo y debe implementarse para garantizar que los datos sean eliminados de forma irreversible.

Fallos en la sincronización de eliminación

La fragmentación es una técnica utilizada en sistemas distribuidos de Big Data en la cual los datos son divididos en fragmentos pequeños llamados particiones o shards que se reparten entre los nodos con el fin de distribuir la carga de trabajo y el almacenamiento, lo que mejora la escalabilidad y el rendimiento. La fragmentación puede realizarse siguiendo dos estrategias:

- **Fragmentación horizontal:** Los datos son divididos en filas o registros y distribuidos entre los nodos. Cada nodo almacenará una parte de los registros completos siguiendo algún criterio: orden alfabético, fecha de carga, etc.
- **Fragmentación vertical:** Los datos se dividen en columnas y se distribuyen. Cada nodo almacenará solo las columnas específicas asignadas.

Aunque este proceso tiene ventajas en el rendimiento, como cualquier solución, es susceptible de errores.

Si la eliminación se realiza de manera incorrecta o incompleta, algunos fragmentos de datos pueden seguir sin borrar en algún nodo. Por ejemplo, en el caso de la fragmentación vertical, un fallo en el proceso o en la sincronización con el resto de los nodos puede dejar columnas sin eliminar en nodos que podrían ser consultados por atacantes. De esta manera, podría producirse revelación de atributo e incluso reidentificación según el tipo de atributo y el conocimiento previo. Por otra parte, algunos sistemas replican la información en

distintos servidores para dotarlo de tolerancia al fallo. En estos casos, será necesario asegurarse de que los procesos de eliminación realizan el borrado de manera consistente.

6.3 Medidas de seguridad

Es necesario aplicar las medidas de seguridad que garanticen el borrado seguro de datos en sistemas de Big Data. En el siguiente apartado se presentan algunas soluciones:

Eliminación física segura
La eliminación física segura implica borrar los datos de manera irreparable de los dispositivos de almacenamiento. Es posible conseguirlo de diversas maneras:

- **Sobreescritura:** Consiste en sobrescribir los datos con información aleatoria para asegurarse de que van a ser irrecuperables por ningún software o atacante. DoD 5220.22-M (Annexo 7) es un estándar de desinfección basado en software muy extendido.
- **Borrado criptográfico:** Otra solución es proteger los datos almacenados mediante técnicas de cifrado y borrar las claves de manera segura. Esto garantiza que aunque los datos sean recuperados no podrán ser utilizados sin la clave. No obstante, seguirán ocupando un espacio en disco.
- **Destrucción física:** En casos extremos, cuando la seguridad de los datos es crítica, se puede optar por la destrucción física de los medios de almacenamiento, como la trituración o quemado de discos duros, para asegurar que la información será irrecuperable.

Monitorización
Es necesario que el sistema monitorice los distintos procesos de borrado seguro de datos y copias de seguridad para tener un control exhaustivo de la actividad. Además, también es posible la elaboración de cuadros de mando que muestren los errores e incluso un sistema automatizado de envío de alertas.
La monitorización ayudaría a detectar si hay algún proceso de eliminación de copias de seguridad o datos históricos que esté fallando para ser solucionado.

Políticas de retención y borrado
Una política de retención y borrado seguro es un conjunto de directrices, buenas prácticas y procesos establecidos por una organización para definir el modo en que los datos serán eliminados al final de su ciclo de vida. Esta regla establece los periodos de retención de los datos y define los criterios de eliminación cuando han dejado de ser útiles.
Las políticas tienen como objetivo disminuir los riesgos de seguridad relativos al proceso y asegurar el cumplimiento normativo.
Una política de eliminación debe contener los siguientes aspectos:

- **Objetivo y alcance:** Inicialmente debe establecerse el propósito de la política y el ámbito de aplicación, indicando qué sistemas y aplicaciones

están dentro del alcance. También se definirá a qué afecta esta política: copias de seguridad, datos históricos, ficheros de datos, ficheros de log de sistema, etc.

- **Categorización datos:** Después es necesario clasificar los datos en categorías según su tipo, sensibilidad y requisitos legales aplicables.
- **Periodos de retención:** Se definirán plazos de retención para las distintas categorías, indicando de manera clara cuánto tiempo se deben mantener los datos antes de ser eliminados.
- **Procedimientos de eliminación:** Se describirán los métodos y medidas técnicas que van a realizarse para alcanzar los objetivos y cumplir con los periodos de retención estipulados. Debe incluir el borrado seguro, destrucción física, eliminación de claves o cualquiera de las medidas que se vayan a implantar.
- **Responsabilidad y roles:** Es necesario asignar responsabilidades claras sobre los diferentes roles. Para que no haya ninguna confusión sobre las funciones de los integrantes del equipo.
- **Comunicación:** Finalmente será necesario realizar una campaña de comunicación para explicar la importancia de la política, y los riesgos relacionados con las malas prácticas, a los empleados para concienciarlos. Además, se proporcionarán las sesiones de capacitación necesarias.

De esta manera, podría evitarse la fuga de información innecesaria ya que se disminuiría la exposición de datos que deberían haber sido eliminados. Por otra parte, se evitarán incumplimientos en las regulaciones y las sanciones pertinentes.
Además, se facilitará la tarea de gestión de almacenamiento y recursos del sistema ya que los datos no solo crecerán.

4. Conclusiones y trabajos futuros

La seguridad de los datos en los sitemas Big Data es de suma importancia en la actualidad. A medida que las organizaciones recopilan y procesan grandes volúmenes de datos, la protección de esa información se vuelve fundamental para garantizar la **confidencialidad, integridad, disponibilidad y privacidad** de los datos.
Una vez finalizado el análisis de la seguridad del dato en sistemas de Big Data es reseñable la necesidad de prestar la atención que requiere cada una de las fases del ciclo de vida del dato, desde el inicio con la ingesta hasta que finalmente es eliminado.
Un sistema es tan seguro como el eslabón más débil. Teniendo en cuenta que cada etapa es susceptible a errores y riesgos, descuidar la segurdidad de cualquiera de ellas penalizaría grávemente el conjunto.
Aunque existe una gran variedad de medidas de seguridad y métodos de implementación, la aplicación de las que se han recopilado a lo largo de este proyecto permitiría disminuir de forma considerable un gran abanico de riesgos, asegurando las diferentes fases del ciclo de vida del dato para alcanzar los objetivos marcados:

- Del objetivo **“Certificar la seguridad de los datos y fiabilidad de los orígenes durante la ingesta de información”** puedo concluir que mediante la aplicación del cifrado en tránsito y el uso de certificados se reducirán en gran parte muchos de los riesgos.
- En cuanto al objetivo de **“Asegurar el procesado de datos”** después de la ingesta me parece importante concluir que aunque existen medidas para detectar los formatos, estructuras o valores anómalos de los datos, un atacante que conociera los órdenes de magnitud de ciertos indicadores y consiguiera enviar datos erróneos al sistema podría falsearlos de manera que no sean detectados en el análisis estadístico de detección de valores atípicos. Por eso mismo, esta fase podría ser especialmente vulnerable a amenazas o errores, ya que en cierta parte confía en que el origen de datos sea legítimo, lo cual no es una buena política.
- Con la ayuda de medidas como el cifrado de datos en reposo (con rotación de claves) y procesos de copia de seguridad sería posible securizar en gran parte el objetivo de **“Fijar medidas de almacenamiento seguro del dato”**. Aún así, debe existir un plan de contingencia para recuperar el sistema en caso de producirse un ataque, como por ejemplo ransomware, de manera inesperada.
- La aplicación de medidas y modelos de privacidad analizados permitirían **“Realizar un uso ético y seguro de los datos almacenados”** como se pretendía al inicio del proyecto, asegurar la privacidad e intimidad de los usuarios. No obstante, no se puede conocer el nivel de conocimiento que un atacante tiene sobre los datos o sobre un usuario en concreto, y aunque se hayan aplicado medidas siempre existe un mayor o menor riesgo de revelación y reidentificación.
- **“Proteger la privacidad de los datos en todo el ciclo de vida”** posiblemente haya sido el objetivo más difícil de alcanzar. Asegurar esta

dimensión de seguridad del dato en ciertas fases del ciclo de vida puede ser altamente complejo. Por ejemplo, la ingesta en sistemas de Big Data puede realizarse de una gran cantidad de orígenes muy heterogéneos que envíen información confidencial de los usuarios en claro (lo cual puede ser potestad de otra organización y limitarnos gravemente). Normalmente, el sistema destino no puede escoger los valores que recibe, es a posteriori, durante el procesado, cuando se aplican medidas de preservación de la privacidad.

- **"Garantizar el borrado seguro de datos"** ha sido un objetivo complejo de alcanzar aun con las medidas que se han aplicado debido a la alta diversidad de los sistemas Big Data y al procesamiento distribuido. Aunque la sincronización de borrado entre nodos funcione correctamente y las políticas de retención sean adecuadas puede haber resquicios en los que se incumpla el derecho al olvido, aunque no debería ser así, a la hora de borrar no solo los datos del sistema si no también de las copias de seguridad que lo contengan. Deberá prestarse especial atención a este aspecto en futuros proyectos.

No obstante, la seguridad no puede tratarse como un fin en sí mismo, ningún sistema es perfectamente seguro, es un proceso que debe estar en constante análisis y **mejora continúa**. Las amenazas y riesgos evolucionan con el tiempo y las medidas de seguridad deben ir de la mano.

Por otra parte, existe el compromiso con otros factores como la utilidad y, principalmente, el presupuesto. En los últimos años, se está incrementando el presupuesto dedicado a la ciberseguridad, ya que antes, muchos ejecutivos no consideraban que fuera a aportar un valor a la organización. Aún así, no todas las empresas gozan del mismo potencial económico y algunas de ellas no puede implementar todas las medidas que les gustaría, con lo cual, deben asumir ciertos riesgos de seguridad y desarrollar planes de contigencia para cuando se materialicen.

Además, se suma al desafío de aplicar medidas de seguridad sobre sistemas Big Data que ya han sido puestos en producción y están siendo usados de manera activa y regular. De por sí, este tipo de sistemas distribuidos suele ser complejo y puede estar siendo explotado por múltiples aplicaciones críticas, por lo que la implementación de cada medida nueva puede tener un gran impacto. Poner en riesgo el funcionamiento puede ser uno de los principales inconvenientes a la hora de aplicar medidas de seguridad. Es por eso que la **privacidad desde el diseño** es esencial a la hora de desarrollar soluciones Big Data. Al integrar la seguridad y privacidad como un factor más a tener en cuenta desde las etapas iniciales del desarrollo adoptando medidas desde el principio puede ahorrar obstáculos y dificultades futuras.

Tambiés es posible concluir la importancia de la **gobernanza** en la seguridad. La gobernanza establece las políticas, procesos, controles, roles y responsabilidades para la buena gestión de los datos. Su correcta aplicación proporciona un marco de trabajo para implementar medidas y supervisar la mejora continua. Además, está estrechamente relacionado con otro aspecto destacado en el grueso del trabajo, la calidad del dato, y su impacto en la seguridad. La gobernanza puede ayudar a mejorar la calidad del dato estandarizando procesos y estableciendo controles sobre los datos.

La buena planificación inicial de las tareas, dividiendo las diferentes fases del ciclo de vida entre las diferentes entregas, ha permitido cumplir con los plazos satisfactóriamente. Además, los *tiempos de backup* definidos han posibilitado la revisión del contenido con el tutor antes de su entrega.
A partir de este proyecto, podrían derivarse distintas líneas futuras de trabajo a desarrollar con un contenido lo suficientemente amplio y de interés:

- **A partir de un sistema de Big Data concreto puesto en producción y en uso:** realizar un análisis de riesgos y medidas de seguridad nativas que ofrece la plataforma en la que está implementando (Hadoop, MongoDB, AWS, Azure...). También sería interesante analizar las implicaciones que pudiera tener la aplicación de medidas sobre el uso y la analítica de los datos.
- Una segunda fase (**fase defensiva**) del proyecto podría centrarse en securizar el sistema implementando de manera práctica las medidas analizadas en el apartado anterior.
- Una tercera fase (**fase ofensiva**) podría consistir en auditar la seguridad del sistema y realizar pruebas de penetración.
- De manera independiente al resto, sería posible implantar un sistema de Big Data teniendo en cuenta la **seguridad y privacidad desde el diseño** como un factor más durante todo el proceso.
- Por otra parte, sería muy interesante realizar un plan de contingencia que se active cuando se da una amenaza desconocida o que ha decidido asumirse por falta de presupuesto para aplicar salvaguardas y recupere el sistema Big Data de la manera más eficiente posible.

5. Glosario

TFM: Trabajo Final de Máster.
CISO (Chief Information Security Officer): Director de seguridad de la información. Desde un rol ejecutivo debe alinear la seguridad de la información con los objetivos de negocio.
BI: Business Intelligence.
AS (Authentication Server): Servidor de autenticación.
LDAP (Lightweight Directory Access Protocol): Protocolo ligero de acceso a directorios.
ACL (Access Control Lists): Listas de control de acceso.
RBAC (Role-Based Access Control): Control de acceso basado en roles.
AES (Advanced Encryption Standard): Estándar de Cifrado Avanzado.
IDS (Intrusion Detection System): Sistema de Detección de Intrusión.
IPS (Intrusion Prevention System): Sistema de Prevención de Intrusión.
CA (Certification Authority): Autoridad de Certificación.
GCP: Google Cloud Platform.
FPE (Format-Preserving Encryption): Cifrado que preserva el formato.
CPD: Centro de procesamiento de datos.
GDPR (General Data Protection Regulation) o RGPD: Reglamento General de Protección de Datos.
UDF (User Defined Functions): Funciones Definidas por el Usuario.
OPE (Order-Preserving Encryption): Cifrado con Preservación del Orden.
TDE (Transparent Data Encryption): Cifrado de Datos Transparente.

6. Bibliografía

[1] Mae Rice, Brennan Whitfield, “What is a Data Platform?”, Sep 23, 2022. (Última consulta: 14/03/2023). What Is a Data Platform? Examples of Big Data Platforms | Built In

[2] Esquema Nacional de Seguridad, MAGERIT, Libro II – Catálogo de Elementos, pág. 40-44.

[3] Yazan Alshboul, Raj Kumar Nepali, Yong Wang, “Big Data LifeCycle: Threats and Security Model”, Twenty-first Americas Conference on Information Systems, Puerto Rico, 2015.

[4] Kafka, Documentation, 7.3 Encryption and Authentication using SSL. (Última consulta: 28/03/2023). https://kafka.apache.org/documentation/#security_ssl

[5] AWS, Características de AWS Key Management Service. (Última consulta: 28/03/2023).
https://aws.amazon.com/es/kms/features/#AWS_Service_Integration

[6] AWS, AWS Certificate Manager. (Última consulta: 29/03/2023). https://aws.amazon.com/es/certificate-manager/

[7] Azure, Add and manage SSL/TLS certificates in Azure App Service. (Última consulta: 29/03/2023). https://learn.microsoft.com/en-us/azure/app-service/configure-ssl-certificate?tabs=apex%2Cportal

[8] Azure, An overview of Azure SQL Database and SQL Managed Instance security capabilities. (Última consulta: 29/03/2023). https://learn.microsoft.com/en-us/azure/azure-sql/database/security-overview?view=azuresql

[9] Google Cloud, Documentación de Cloud Key Management Service. (Última consulta: 29/03/2023). https://cloud.google.com/kms/docs?hl=es
[10] Cloud SQL, FAQ, ¿Cómo se administra la encriptación de los datos en tránsito? (Última consulta: 29/03/2023). https://cloud.google.com/sql/faq?hl=es-419#encryption-manage-transit

[11] INCIBE, ¿Qué son y para qué sirven los SIEM, IDS e IPS? (Última consulta: 29/03/2023). https://www.incibe.es/protege-tu-empresa/blog/son-y-sirven-los-siem-ids-e-ips#:~:text=Estos%20sistemas%20llevan%20a%20cabo,red%2C%20implementando%20pol%C3%ADticas%20que%20se

[12] Azure, What is Azure DdoS Protection? (Última consulta: 30/03/2023)
https://learn.microsoft.com/en-us/azure/ddos-protection/ddos-protection-overview

[13] AWS, AWS Shield, (Última consulta: 30/03/2023)
https://aws.amazon.com/es/shield/

[14] UOC, Jordi Herrera Joancomartí, Cristina Perez Solà, Criptografía de clave simétrica, pág.37-44, 2021.

[15] Apache Ambari. (Última consulta: 04/04/2023). https://ambari.apache.org/

[16] Nagios. (Última consulta: 04/04/2023). https://www.nagios.com/products/nagios-log-server/

[17] Cloudflare, ¿Cómo funciona SSL? Certificados SSL y TLS. (Última consulta: 11/04/2023). https://www.cloudflare.com/es-es/learning/ssl/how-does-ssl-work/

[18] Aprender Big Data, Apache Avro: Una introducción sencilla. (Última consulta: 12/04/2023). https://aprenderbigdata.com/apache-avro/

[19] Towards Data Science, JSON Schema: Integrity checking for NoSQL Data, Chuck Connell, 21/01/2022. (última Consulta: 12/04/2023) https://towardsdatascience.com/json-schema-integrity-checking-for-nosql-data-b1255f5ea17d

[20] Apache Nifi, ValidateJson. (Última consulta: 12/04/2023) https://nifi.apache.org/docs/nifi-docs/components/org.apache.nifi/nifi-standard-nar/1.20.0/org.apache.nifi.processors.standard.ValidateJson/index.html

[21] Pydantic, (Última consulta: 12/04/2023). https://docs.pydantic.dev/

[22] AWS, AWS Machine Learning Blog. (Última consulta: 13/04/2023). https://aws.amazon.com/es/blogs/machine-learning/use-the-built-in-amazon-sagemaker-random-cut-forest-algorithm-for-anomaly-detection/

[23] GCP, What's new with BigQuery ML: Unsupervised anomaly detection for time series and non-time series data, 02/07/2021. (Última consulta: 13/04/2023). https://cloud.google.com/blog/products/data-analytics/bigquery-ml-unsupervised-anomaly-detection

[24] Otmane Azeroual, Anastasija Nikiforova,
Apache Spark and MLlib-Based Intrusion Detection System or How the Big Data Technologies Can Secure the Data, 24/01/2022. Pág. 5-6.

[25] Apache Mahout. (Última consulta: 13/04/2023). https://mahout.apache.org/

[26] Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers, Fotmat-Preserving Encryption. Pág. 1.

[27] Hadoop, HDFS Architecture Guide. (19/04/2023). https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html#Introduction

[28] AWS, Características de Amazon S3. (Última consulta: 19/04/2023). https://aws.amazon.com/es/s3/features/

[29] Azure, Azure Blob Storage. (Última consulta: 21/04/2023). https://azure.microsoft.com/es-es/products/storage/blobs/

[30] Google Cloud, Almacenamiento de objetos para empresas de todos los tamaños. (Última consulta: 21/04/2023). https://cloud.google.com/storage?hl=es-419#section-1

[31] INCIBE, Ransomware: Una guía de aproximación para el empresario. 2020. Pág:6-9. (Última consulta: 24/04/2023). https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_ransomware.pdf

[32] INCIBE, ¿Es seguro tu escritorio remoto?. Publicado el 22/08/2019. (Última consulta: 24/04/2023). https://www.incibe.es/protege-tu-empresa/blog/seguro-tu-escritorio-remoto

[33] AWS, Rotación de AWS KMS keys. (Última consulta: 25/04/2023). https://docs.aws.amazon.com/es_es/kms/latest/developerguide/rotate-keys.html

[34] Microsoft, Configure la rotación automática de claves en Azure Key Vault. (Última consulta: 26/04/2023). https://learn.microsoft.com/es-es/azure/key-vault/keys/how-to-configure-key-rotation

[35] Google Cloud Key Management Service, Rotación de claves. (Última consulta: 26/04/2023). https://cloud.google.com/kms/docs/key-rotation?hl=es-419

[36] AWS, Creating an Encrypted File System. (Última consulta: 26/04/2023). https://docs.aws.amazon.com/es_es/whitepapers/latest/efs-encrypted-file-systems/creating-an-encrypted-file-system.html

[37] Apache Hadoop, Transparent Encryption in HDFS. (Última consulta: 28/04/2023). https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/TransparentEncryption.html

[38] Google Cloud, Encriptación en reposo predeterminada. (Última consulta: 28/04/2023). https://cloud.google.com/docs/security/encryption/default-encryption?hl=es-419

[39] Azure, Azure Storage encryption for data at rest. (Última consulta: 28/04/2023). https://learn.microsoft.com/en-us/azure/storage/common/storage-service-encryption

[40] UOC, Miguel Colobran Huguet, Gestión de incidentes de seguridad. Pág. 21-26.

[41] BOE, REGLAMENTO (UE) 2016/679 DEL PARLAMENTO EUROPEO Y DEL CONSEJO de 27 de abril de 2016. Artículo 5.1 . https://www.boe.es/doue/2016/119/L00001-00088.pdf

[42] INCIBE, Cómo gestionar una fuga de información. (Última consulta: 05/05/2023). https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_ciberseguridad_gestion_fuga_informacion_0.pdf

[43] INCIBE, Ingeniería social: técnicas utilizadas por los ciberdelincuentes y cómo protegerse, 05/09/2019. (Última consulta: 05/05/2023). https://www.incibe.es/empresas/blog/ingenieria-social-tecnicas-utilizadas-los-ciberdelincuentes-y-protegerse

[44] INCIBE, Concienciación y formación, Políticas de seguridad para la PYME. (Última consulta: 08/05/2023). https://www.incibe.es/sites/default/files/contenidos/politicas/documentos/concienciacion-y-formacion.pdf

[45] UOC, Navarro-Arribas, G. (2020) Introducción a la privacidad en la publicación de datos. [Recurso de aprendizaje textual]. 1.ª ed. Barcelona: Fundació Universitat Oberta de Catalunya (FUOC). Pág. 17-19.

[46] L. Sweeney (2000). «Simple Demographics Often Identify People Uniquely?». Data Privacy Working Paper 3. Pittsburgh: Carnegie Mellon University. https://bit.ly/3fGYq1J

[47] UOC, Navarro-Arribas, G. (2020) Introducción a la privacidad en la publicación de datos. [Recurso de aprendizaje textual]. 1.ª ed. Barcelona: Fundació Universitat Oberta de Catalunya (FUOC). Pág. 42-58.

[48] ENISA, Privacy by design in big data An overview of privacy enhancing technologies in the era of big data analytics, december 2015. Pág. 38-41.

[49] Encryption Consulting, Types of Tokenization: Vault and Vaultless. (Última consulta 22/05/2023). https://www.encryptionconsulting.com/education-center/types-of-tokenization-vault-and-vaultless/

[50] AEPD, Introducción al hash como técnica de seudonimización de datos personales. Octubre 2019. https://www.aepd.es/es/documento/estudio-hash-anonimidad.pdf

[51] UOC, Navarro-Arribas, G. (2020) Introducción a la privacidad en la publicación de datos. [Recurso de aprendizaje textual]. 1.ª ed. Barcelona: Fundació Universitat Oberta de Catalunya (FUOC). Pág. 20-28.

[52] BOE, REGLAMENTO (UE) 2016/679 DEL PARLAMENTO EUROPEO Y DEL CONSEJO de 27 de abril de 2016. Artículo 17.1 . https://www.boe.es/doue/2016/119/L00001-00088.pdf

[53] Tecnautas, DoD 5220.22-M: Todo lo que necesita saber. (Última consulta 30/05/2023). https://tecnonautas.net/dod-5220-22-m-todo-lo-que-necesita-saber/ .

7. Anexos

Anexo 1: Kerberos

Kerberos es un servicio de autenticación utilizado en redes abiertas o no seguras. Este protocolo sirve para autenticar las solicitudes de servicio entre dos o más host a través de una red no fiable, como Internet. El cifrado criptográfico y un tercero de confianza se utilizan para autenticar las aplicaciones cliente-servidor y verificar la identidad de los usuarios.

a. Agentes

A continuación, se presentan los agentes implicados en el proceso de autenticación:

- **Cliente:** Actúa como representante del usuario e inicia la comunicación y la solicitud de servicio.
- **Servidor host:** Es el servidor que aloja el servicio al que quiere acceder el usuario.
- **Servidor de autenticación (AS):** Realiza la autenticación del cliente. Si se realiza con éxito, el AS emite un ticket para el cliente, el TGT (ticket-granting ticket). Este ticket garantiza a los demás servidores que el cliente está autenticado.
- **Ticket-granting server (TGS):** Es un servidor de aplicación que emite tickets de servicio.
- **Centro de distribución de claves (KDG):** Formado por el servidor de autenticación (AS) y el ticket-granting server (TGS).

b. Proceso

El proceso se puede resumir a partir de los siguientes pasos:

1. El cliente realiza una solicitud cifrada al servidor de autenticación. Cuando el AS la recibe, busca la contraseña en la base de datos de Kerberos mediante el ID de usuario. Si la contraseña es correcta, el AS descifra la solicitud.
2. Una vez se verifica el usuario, el AS emite un ticket-granting ticket (TGT), que se envía de vuelta al cliente.
3. El cliente envía el TGT al ticket-granting server. Junto con el TGT el cliente indica el motivo de acceso al servidor host. El TGS descifra el ticket con la clave secreta que comparten el AS y el TGS.
4. Si el TGT es válido, el TGS emite un ticket de servicio para el cliente.
5. El cliente envía el ticket de servicio al host. El servidor descifra el ticket con la clave secreta que comparten el servidor y el TGB.
6. Si las claves secretas coinciden, el host permite al cliente acceder al servicio.

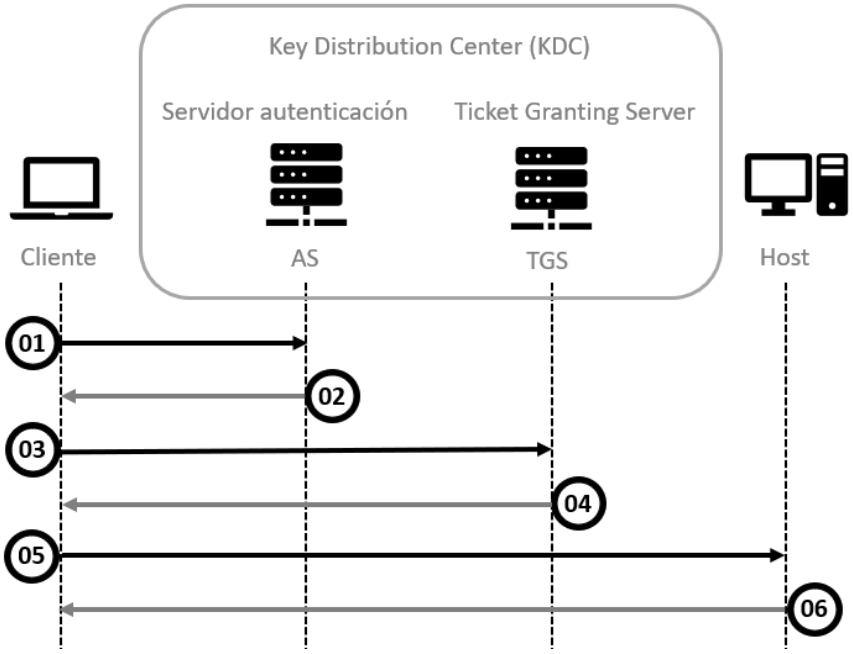

Figura 4: Protocolo Kerberos.

Anexo 2: LDAP

LDAP es un protocolo de aplicación de TCP/IP que permite el acceso a un servicio de directorio para buscar información en un entorno de red.
En el servidor LDAP se encontrarán los nombres de usuario y contraseñas correspondientes para dar acceso al sistema. Además, también podría incluirse otra información como datos de contacto del usuario, ubicación de los recursos de la red local, certificados digitales…
Si montamos un servidor LDAP que de servicio de acceso, se podrían generar inicios de sesión en el sistema operativo sin necesidad de agregar diferentes usuarios al sistema operativo, ya que el acceso lo daría el propio servidor. De esta manera todo está centralizado.
LDAP es un protocolo con arquitectura cliente-servidor y el siguiente funcionamiento:

1. El cliente se conecta al servidor LDAP.
2. Se establece la conexión cliente-servidor y se intercambian datos entre ambos lados para iniciar el acceso si los datos son correctos.

Es importante destacar que un cliente puede realizar dos acciones cuando está conectado al servidor:

- Autenticación: Mecanismo por el que un usuario se identifica frente a un sistema, con sus credenciales, como puede ser nombre de usuario y contraseña.
- Autorización: Mecanismo para conseguir permiso para realizar una acción en el sistema o tener acceso a ciertos recursos de red.

El protocolo LDAP también nos permite intercambiar información entre varios servidores. Por ejemplo, en una red corporativa con varios servidores y unos

datos alojados en uno de ellos y otros en otro. Es posible desde uno de los servidores consultar a otro si dispone de acceso.

Anexo 3: ACL

Una lista de control de acceso es una lista de reglas que especifican que usuarios o sistemas tienen acceso o no a un objeto o sistema. Es una forma de determinar los permisos de acceso apropiados a un determinado objeto.
Las ACL permiten controlar el flujo del tráfico en equipos de redes como rúters y conmutadores. Su principal objetivo es filtrar tráfico: permitir o denegar el tráfico de red de acuerdo con alguna condición.

a. Funcionamiento

Cada lista tiene una o más entradas de control de acceso que consisten en el nombre de un usuario o grupo. El usuario también puede ser un nombre de rol, como programador, administrador… Para cada uno de estos usuarios o grupos los privilegios de acceso se establecen en una cadena denominada máscara de acceso.

b. Tipos

Hay dos tipos básicos de listas:

- **File System ACL:** Gestionan el acceso a archivos y directorios. Establecen los permisos de acceso de los usuarios y sus privilegios una vez han accedido.
- **Networking ACL:** Administran el acceso a la red proporcionando instrucciones a los conmutadores y enrutadores de red que especifican los tipos de tráfico que pueden interactuar con la red. También especifican permisos de usuario una vez dentro de la red. Tienen un funcionamiento similar a un cortafuegos en este contexto.

Anexo 4: RBAC

Para garantizar la seguridad en las grandes organizaciones, las autorizaciones de acceso individuales se definen en la lista de control de acceso (ACL), con el inconveniente de que a medida que crecen los usuarios mayor es el mantenimiento que requiere y es más probable que pueda haber errores al asignar las autorizaciones individuales. Una alternativa flexible y eficiente es el control de acceso basado en roles (RBAC).
El control de acceso basado en roles es un modelo de seguridad en el cual el administrador del sistema asigna un nivel y una categoría de seguridad a cada usuario y objeto en función del rol que cumple. Según el rol asignado el usuario podrá realizar unas u otras acciones. En el contexto del Big Data, el usuario podrá acceder, modificar o eliminar datos dependiendo del rol que tenga.
La aplicación y supervisión del RBAC tiene lugar a través de un sistema de gestión de la identidad (IAM). Para las empresas con un gran número de empleados, este sistema es particularmente útil en las áreas de registros, control y actualización de las identidades y derechos de acceso. La asignación de autorizaciones se denomina provisioning, mientras que la retirada se denomina de-provisioning.

Estos dos términos pueden ser críticos para mantener la seguridad de un sistema. Es necesario llevar un inventario actualizado de las altas y bajas de usuarios para retirarles los privilegios. Los ataques internos o de exempleados descontentos son un factor a tener en cuenta en el análisis de riesgos.

Anexo 5: AES

AES es un algoritmo de cifrado de bloques de texto en claro de 128 bits y longitud de clave variable entre 128, 192 o 256 bits. [12]
Su funcionamiento se basa en una transformación inicial y una serie de iteraciones que varía según la longitud de la clave.
AES trabaja con una representación matricial de bytes (cadena de bits del texto en claro y claves) cómo la siguiente:

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

Todas las funciones que ejecuta AES tienen como entrada y salida una matriz con el mismo formato que la anterior.
Las matrices resultantes de aplicar los diferentes pasos se denominan matrices de estado.
Por cada iteración el algoritmo realiza las siguientes funciones:

a. **AddRoundKey:** Hace la suma XOR de la matriz de estado con la subclave correspondiente dispuesta de manera matricial. Cada iteración utiliza una subclave diferente generada por el sistema.
b. **ByteSub:** A partir de una matriz de estado A se aplica una transformación S para obtener B.
   Las cajas S de AES son una matriz de 256 elementos que se utiliza como tabla de consulta.
c. **ShiftRow:** Desplaza las filas de la matriz de estado a la izquierda tantas veces como el subíndice de la fila: 0 veces la fila 0, 1 vez la fila 1…
d. **MixColumns:** Mezcla las columnas de la matriz de estado a partir de operaciones polinomiales.

Anexo 6: SSL/TLS

SSL/TLS es un protocolo criptográfico que permite la comunicación segura entre dos sistemas (fuente origen de los datos y sistema Big Data destino). Los certificados SSL/TLS proporcionan encriptación de los datos además de autenticación de las partes, punto clave para asegurar la legitimidad del origen de datos. Las comunicaciones TLS incluyen un código de autenticación del mensaje, o MAC, se trata de una firma digital que confirma que la comunicación procede del origen real. De esta manera se evita la suplantación de identidad de una fuente de datos maliciosa y garantiza que los datos no hayan sido alterados en tránsito.

Los certificados son emitidos por organizaciones llamadas Autoridades de Certificación (CA) e instalados en el servidor origen para establecer la comunicación segura y encriptada. El funcionamiento se puede comprender mediante los siguientes puntos [17]:

- Las dos partes abren una conexión segura e intercambian la clave pública.
- Las dos partes generan claves de sesión para encriptar las comunicaciones en cada sesión.
- TLS garantiza que ambas partes realmente son quien dicen ser.
- TLS también garantiza que los datos no hayan sido alterados ya que incluye un código de autenticación de mensajes en las transmisiones.

Anexo 7: DoD 5220.2-M

DoD 5220.2-M es un método de desinfección de datos basado en sotfware utilizado para la destrucción de archivos y datos mediante la sobreescritura de información en discos duros o dispositivos de almacenamiento.
El eso de este estándar permite evitar la recuperación de información del disco duro.
Este método está formado por tres pasos:

- **Paso 1:** Escribe un cero y verifica la escritura.
- **Paso 2:** Escribe un uno y verifica la escritura.
- **Paso 3:** Escribe un carácter aleatorio y verifica la escritura.

Es una de la soluciones de eliminación segura de información más extendidas. [53]