El módulo **TFM Email Date Coherence** sirve para detectar **manipulaciones temporales** en correos electrónicos mediante el análisis forense de sus cabeceras y metadatos. Se centra en verificar la **integridad cronológica** del mensaje desde que se envía hasta que se almacena en el sistema.

Su función principal es analizar la coherencia entre tres puntos clave: la fecha declarada en la cabecera Date:, la fecha del último servidor de tránsito registrada en la cadena Received:, y los metadatos de tiempo del archivo .eml (creación/modificación).

El código extrae primero el *epoch* de la cabecera Date: y localiza el *epoch* del último *hop* válido en la cadena Received:. Luego, realiza comprobaciones clave:

1. **Date: Posterior a Received:** Alerta si la fecha declarada (Date:) es significativamente posterior al último servidor que procesó el correo. Esto sugiere que la cabecera Date: fue alterada para falsificar la hora de envío.
2. **Archivo EML Posterior a Received:** Advierte si la fecha de modificación del archivo es muy posterior al último Received:, indicando una posible manipulación posterior al tránsito.
3. **Date: Ausente/Inválida:** Penaliza la falta o el formato incorrecto de la cabecera Date:.

El módulo asigna una **puntuación de severidad** a los hallazgos y genera un informe detallado en Autopsy. Este análisis ayuda al forense a **identificar correos electrónicos sospechosos de alteración** o que pasaron por servidores mal configurados, siendo crucial en procesos de **adveración**.

**Detección de Inconsistencias**

El corazón del módulo reside en la función analyze\_date\_coherence, que compara estos tres valores en busca de desfases sospechosos:

1. **Date: posterior al Último Received (Alerta):** Detecta si la fecha declarada del correo es **posterior** a la fecha en que el último servidor de tránsito la registró. Una diferencia de más de 5 minutos (300 segundos) se marca como una anomalía grave.
2. **Metadato Archivo posterior al Último Received (Alerta):** Señala si la fecha de modificación/creación del archivo .eml en el disco es significativamente **posterior** a la fecha en que el servidor terminó de procesar el correo (más de 7 días), lo que sugiere una **manipulación del archivo EML** después de su entrega.
3. **Date: posterior al Metadato Archivo (Info):** Identifica cuando la fecha declarada es significativamente **posterior** a la fecha del archivo .eml (más de 7 días). Esto puede indicar que la cabecera Date: fue alterada y el archivo original fue copiado, o que el archivo EML ha estado guardado por un largo tiempo.

El módulo **Email Date Coherence** verifica la **consistencia temporal** de correos electrónicos comparando la cabecera **Date:** declarada por el cliente emisor con la **línea temporal real** inferida de la cadena **Received:** y, como referencia secundaria, con los **metadatos del archivo .eml** en el contenedor forense.  
Primero extrae Date y la convierte a **epoch UTC**; después parsea todas las cabeceras Received y determina el **instante más reciente** en el que un MTA procesó el mensaje (último hop efectivo).  
Si la Date aparece **posterior** al último Received por encima de un **umbral corto** (p. ej. 5 min), se marca **alerta**: es incoherente que el mensaje “se haya recibido” antes de “haber sido fechado”.  
El módulo también compara la Date con los **tiempos del archivo .eml** (creación/modificación). Diferencias **grandes** (p. ej. >7 días) se anotan como indicios de **reexportación**, **copia tardía** o posible **manipulación** del contenedor, con **penalización suave** porque son efectos habituales en la cadena de custodia.  
Todos los hallazgos se resumen en un **Analysis Result** con **desglose legible** (offsets, hops implicados) y una **puntuación** que cuantifica la severidad, facilitando priorización y reportabilidad.  
En el contexto del TFM, este módulo **apoya la adveración** al aportar una **prueba técnica trazable** de que la **cronología** interna del correo es **verosímil** o presenta **inconsistencias**; además, se integra con el resto de validaciones (cabeceras inconsistentes, hashes, etc.) para construir un **criterio pericial** robusto y reproducible.

## Módulo de Coherencia Temporal de Correos Electrónicos (tfm\_email\_date\_coherence)

El módulo de ingesta **tfm\_email\_date\_coherence** constituye un pilar fundamental en la etapa de **Adveración (Autenticación)** del análisis forense de correos electrónicos. Su función principal es inspeccionar y cuantificar el grado de fiabilidad de los *timestamps* del correo electrónico, contrastándolos entre sí y con los metadatos del archivo contenedor.

La manipulación temporal de la cabecera Date: o de los *logs* internos es una técnica habitual de falsificación de pruebas. Este módulo aborda este riesgo mediante un análisis multifactorial de la coherencia cronológica.

### 4.3.1. Utilidad en la Adveración Forense

La adveración de un correo electrónico busca determinar si el mensaje es auténtico, si no ha sido alterado y si sus metadatos reflejan fielmente su historial de tránsito. El módulo contribuye a este objetivo de tres maneras esenciales:

1. **Detección de Falsificación de Origen (Date:):** La cabecera Date: es la más fácil de falsificar, ya que es generada por el cliente de correo del remitente. Al contrastarla con las marcas de tiempo de los servidores (Received:), el módulo puede detectar desajustes significativos, indicando que la fecha de envío declarada fue alterada manualmente.
2. **Identificación de Manipulación Post-Recepción:** La comparación de los *timestamps* del correo (Date: y Received:) con la fecha de Modificación/Creación del archivo EML en el disco (f\_date) revela si el archivo fue alterado significativamente después de ser recibido o guardado. Un f\_date muy posterior al último Received: sugiere que el contenido o los metadatos del archivo fueron manipulados.
3. **Evaluación de la Integridad del Tránsito (Received:):** Al analizar la cadena completa de Received:, el módulo verifica la coherencia interna de los *logs* de tránsito de los servidores. Los hallazgos de **orden cronológico inverso** (un servidor registra la recepción antes que el servidor anterior el envío) o la **ausencia de continuidad entre saltos** (BY del salto $n+1$ no coincide con el FROM del salto $n$) son indicadores de **saltos omitidos o cabeceras falsificadas**.

### 4.3.2. Funcionamiento del Módulo

El proceso del módulo se estructura en las siguientes fases, apoyándose en las utilidades definidas en el *core* (tfm\_email\_core.py):

#### A. Obtención y Normalización de Datos

1. **Metadatos de Archivo:** Se extraen la fecha de Creación (f.getCrtime()) y Modificación (f.getMtime()) del archivo EML. Se toma el valor más reciente o fiable (f\_date) como referencia de la manipulación en el sistema operativo.
2. **Cabecera Date::** La función **get\_date\_header\_epoch()** parsea la cabecera Date: declarada por el remitente, convirtiéndola a **formato Epoch** (segundos desde el 1 de enero de 1970 UTC) para facilitar el cálculo. Adicionalmente, se detecta la anomalía de **múltiples cabeceras Date:**, un fuerte indicador de reencapsulado o falsificación torpe.
3. **Cadena Received::** Se utiliza **parse\_received\_chain()** para iterar sobre todas las cabeceras Received:. Por cada salto (hop), se extraen el emisor (from), el receptor (by), las direcciones IP y, crucialmente, la **marca de tiempo del servidor**, también convertida a Epoch.

#### B. Análisis de Coherencia (analyze\_date\_coherence)

Esta función compara las diferentes fuentes de tiempo y genera un diccionario de hallazgos (issues):

1. **Date: vs. Último Received::** Se calcula la diferencia entre la hora declarada (date\_epoch) y el registro de tiempo más reciente del servidor (last\_recv\_epoch). Una discrepancia superior a un umbral (e.g., 5 minutos) genera una alerta.
2. **Date: vs. Tiempo de Ingesta:** Se contrasta el Date: con la hora actual de procesamiento en Autopsy (ingest\_time). Si el Date: se proyecta más de 24 horas en el futuro, se dispara una alerta de **¡SEVERIDAD ALTA!**, ya que indica que el reloj del remitente estaba mal ajustado o que se ha falsificado una prueba futura.
3. **Date: vs. Metadatos de Archivo (f\_date):** Si el Date: es considerablemente posterior al f\_date (e.g., más de 7 días), sugiere que el mensaje fue modificado o creado con posterioridad.
4. **f\_date vs. Último Received::** Si la fecha del archivo EML es muy posterior al último registro del servidor (last\_recv\_epoch), se genera una alerta por **manipulación del archivo local** (e.g., se guardó y se modificó meses después).

#### C. Asignación de Puntuación y Reporte

La función **summarize\_date\_coherence\_findings()** consolida los hallazgos y asigna una **puntuación de severidad** (score) para clasificar la anomalía:

* Se asignan puntos a cada tipo de incongruencia (e.g., 80 puntos para un Date: en el futuro, 45 puntos para múltiples Date:, 50 puntos para desfase Date/Received).
* La puntuación total determina la importancia forense del hallazgo.
* Los resultados son publicados en el *Blackboard* de Autopsy como un *Interesing File Hit* (TSK\_INTERESTING\_FILE\_HIT), incluyendo una descripción detallada de las incongruencias detectadas y las fechas contrastadas (incluyendo las cabeceras alternativas como contexto).

Este enfoque sistemático permite al analista forense identificar rápidamente los correos electrónicos con inconsistencias temporales, focalizando la investigación en aquellos mensajes con mayor probabilidad de haber sido objeto de **adulteración temporal**.