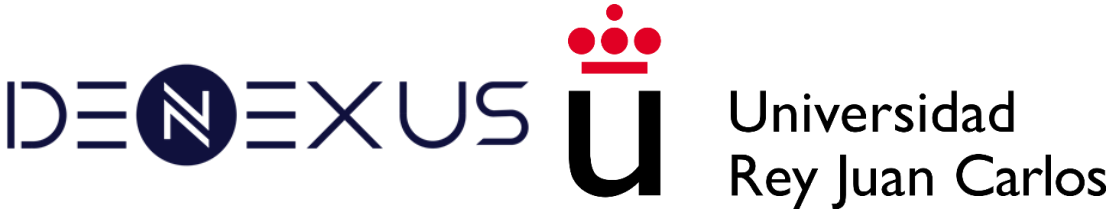
Convocatoria Colaboración Público-Privada CPP2021-009025

DICYME:

Dynamic Industrial CYberrisk Modelling based on Evidence

Iniciativa Conjunta de:



ENTREGABLE 3.3:

Prototipo final del sistema de visualización y toma de decisiones

Coordinadores:

Romy R. Ravines (DeNexus Tech)
Isaac Martín de Diego (Universidad Rey Juan Carlos)
Alberto Fernández Isabel (Universidad Rey Juan Carlos)

Contenido

1 INTRODUCCIÓN Y OBJETIVOS

Este documento resume los pasos realizados en el desarrollo de la Actividad 3 (Sistema de Visualización y toma de decisiones), con el resultado del prototipo final de interfaz gráfica y herramienta de soporte, desarrollado iterando sobre la base establecida en los prototipos primero (entregable E3.1) y segundo (entregable E3.2).

Al igual que en anteriores iteraciones, este sistema de visualización se vale de los datos extraídos gracias a los módulos de extracción automática de datos desarrollados en la Actividad 1 (Sistema de Recogida de evidencias) y a los resultados de los módulos de medida/estimación de probabilidad e impacto de la Actividad 2 (Módulos de medida/estimación de probabilidad e impacto).

En las siguientes secciones se explica:

- Cómo se ha construido la nueva navegación de la aplicación para guiar al usuario en el proceso desde la adquisición de los datos hasta la cuantificación del ciberriesgo.
- Las mejoras que se han introducido en las funcionalidades presentes en el segundo prototipo (entregable E3.2).
- Las nuevas funcionalidades y visualizaciones que se han añadido con los resultados del proyecto para facilitar la toma de decisiones.
- Los datos y código que se encuentran disponibles en este entregable (*deriskGroup / DICYME Project · GitLab*) y cómo pueden emplearse.

2 CAMBIOS Y MEJORAS DEL SISTEMA DE VISUALIZACIÓN

La aplicación descrita a lo largo de este entregable corresponde con la versión v1.0.0, cuyo desarrollo concluyó en julio de 2025. Las siguientes secciones describen los principales cambios y nuevas funcionalidades incluidas en el sistema.

2.1 Conjunto de datos de IDS

Haciendo uso del conjunto de datos desarrollado en el entregable E1.1, se han desarrollado los gráficos y visualizaciones mostrados en la *Ilustración 1. Pestaña de datos de IDS*. En esta se muestran cuatro cajas con valores totales calculados sobre el conjunto de datos, como las instalaciones o *facilities* de las que se disponen datos, redes, etc. Estos sirven para situar al usuario y ubicarle respecto del conjunto de datos a analizar. Seguidamente, se construye una gran serie temporal con todos los instantes de los que hay datos para los diferentes indicadores. Los datos representados varían en función de las entradas que el usuario selecciona en la barra izquierda, que se modifican dinámicamente según la selección del valor previo. Por ejemplo, si el usuario selecciona el alcance "*Site*", se muestra la lista de indicadores que agregan los datos en base a las instalaciones escogidas. Si opta por "*Nivel Purdue*", el usuario tendrá que seleccionar el

indicador y los diferentes niveles que desea visualizar, en este caso entre {1, 2, 3, 3.5, 4, 5}.

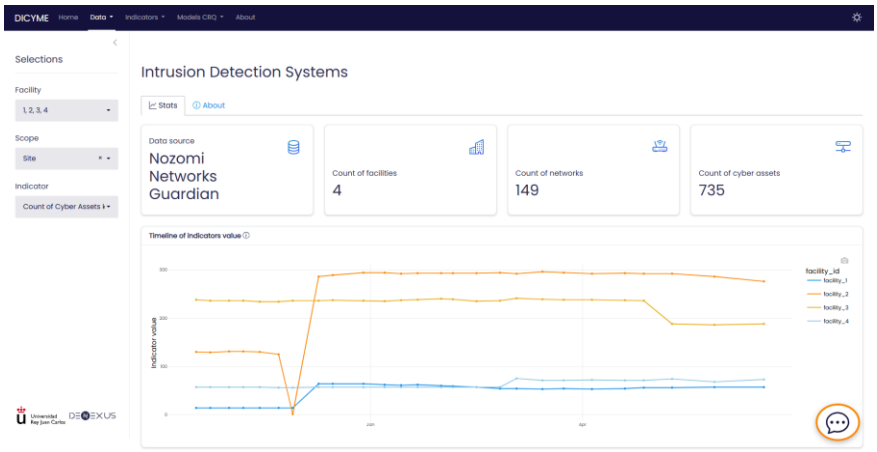


Ilustración 1. Pestaña de datos de IDS

2.2 Conjunto de datos de ciber incidentes

Dentro de esa pestaña se ha incluido un nuevo selector dentro de la agrupación de inputs ya existentes que permite escoger dos tipos de conjuntos de datos que se mostrarán en los gráficos de incidentes. Los datasets que se pueden seleccionar serían el general utilizado previamente y el nuevo conjunto con datos actualizados hasta mayo de 2025 que se ha utilizado para el sistema de combinación de víctimas potenciales.

El nuevo dataset cuenta con incidentes procedentes de tres bases de datos, ICSStrive, EuRepoC y TI Safe, filtradas para únicamente incluir incidentes que afectan a infraestructuras críticas e industriales. Como en el caso del conjunto de datos original, con estos nuevos datos se pueden generar tanto el gráfico de la serie temporal de ciber incidentes como el gráfico de barras con el que se diferencian las distintas variables (véase la Ilustración *2. Visualización del nuevo conjunto de datos de ciber incidentes*).

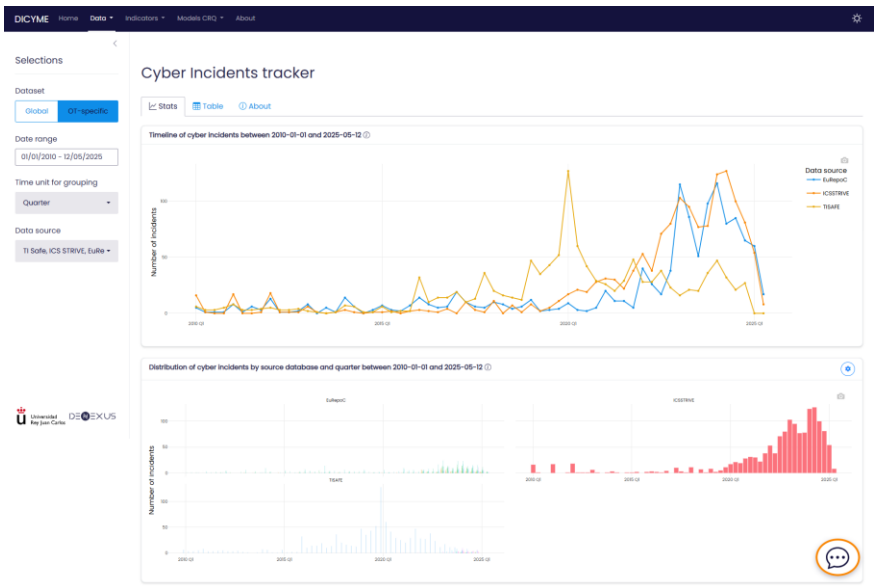


Ilustración 2. Visualización del nuevo conjunto de datos de ciber incidentes

2.3 Indicador de atractivo

Este modelo, cuyo último prototipo se aborda en el entregable E2.3, se ha actualizado en este prototipo final del sistema de visualización y soporte a la toma de decisiones. Además, se ha incluido una nueva pestaña en la que, dada una víctima potencial, definida por los inputs del modelo, el usuario puede calcular en tiempo real el indicador. Como muestra la *Ilustración 3. Cálculo del atractivo para víctimas potenciales*, escogiendo un país, industria, facturación, etc. se muestra el valor de Atractivo, situando los parámetros respecto de los valores del conjunto de datos de perfil de víctimas. Esto permite al usuario ubicar a la entidad potencial respecto del dataset empleado en el entrenamiento del modelo del indicador.

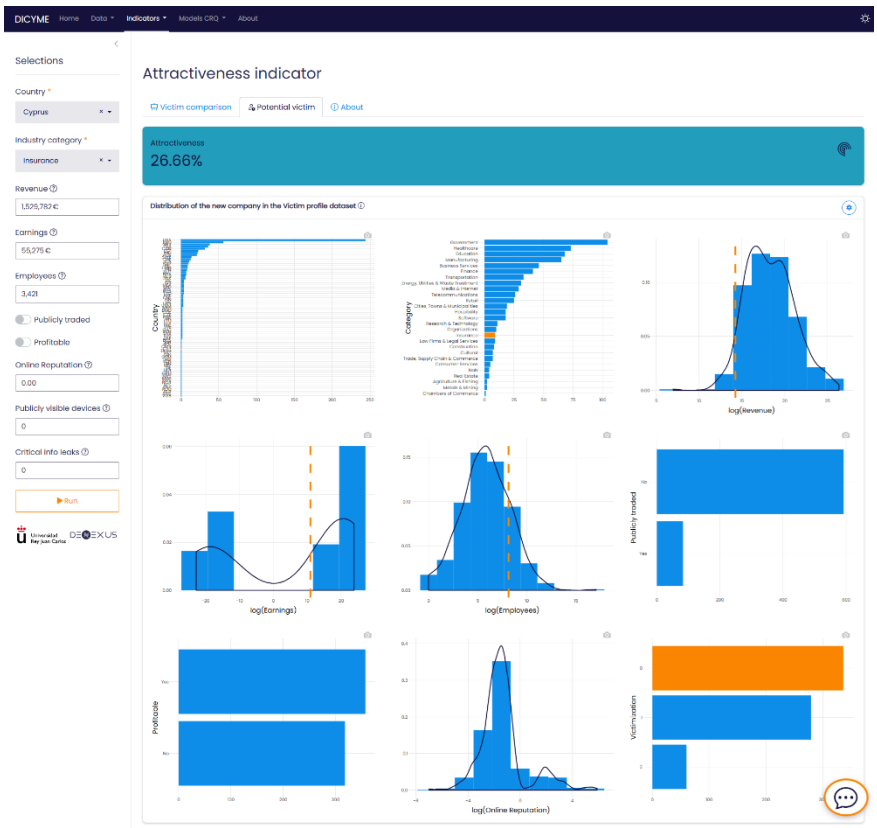


Ilustración 3. Cálculo del atractivo para víctimas potenciales

2.4 Indicador de actores de amenazas

Al igual que para el indicador de atractivo, en este apartado se ha añadido una pestaña que permite ejecutar en tiempo real el indicador para una entidad potencial, definida por su país e industria. El sistema calcula para todos los actores de amenazas de los que se disponen datos el indicador, pudiendo filtrar entre los activos y los latentes o durmientes. Seguidamente, se muestran diferentes aspectos: la cantidad de actores analizados, la distribución del valor del indicador para ellos, el valor medio en cada cuartil, así como los 20 actores con valor del indicador más alto en un gráfico circular que aporta información de los indicadores parciales (valores de objetivo y actividad, incluidos en el cálculo final del indicador).

Por otro lado, se ha añadido una nueva pestaña que permite comparar la distribución del indicador para diferentes entidades del conjunto de datos de perfil de víctimas (véase la *Ilustración 4. Comparación del indicador de actores de amenazas en varias entidades*). Estas puntuaciones, calculadas también en tiempo real, permiten visualizar las similitudes o diferencias entre las distribuciones según las entidades seleccionadas.

Ilustración 4. Comparación del indicador de actores de amenazas en varias entidades

2.5 CVE2TTPs

Como novedad en esta sección se han incluido dos nuevas pestañas que permiten mostrar información importante tanto para las vulnerabilidades (CVEs) como para las tácticas. La primera de ellas, llamada *CVE pairings*, permite mostrar información relativa al CVE como puede ser la descripción, tipo de vulnerabilidad valor de criticidad CVSS v2 y v3, así como las técnicas de MITRE relacionadas con ese CVE por el modelo (véase la *Ilustración 5. Emparejamiento de CVEs*). Los diferentes CVEs pueden ser seleccionados mediante una caja de texto, donde primera se deberá seleccionar el año, que filtrará las opciones disponibles en el siguiente desplegable, donde se escoge el CVE en específico. Este flujo se ha diseñado porque la aplicación no es capaz de asumir la larga lista de técnicas que se genera si no se discrimina por año. De esta manera, se guía al usuario en la búsqueda de la vulnerabilidad que desee analizar, simplificando los conjuntos de datos con los que se trabaja internamente.

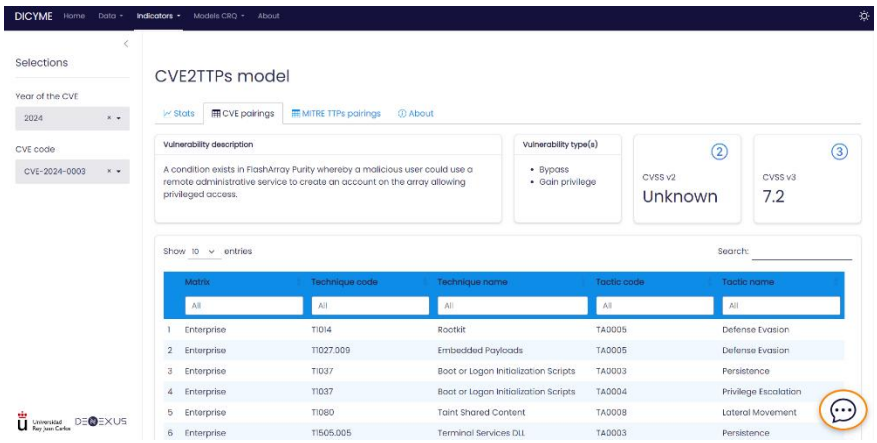


Ilustración 5. Emparejamiento de CVEs

De igual manera, se ha incluido un emparejamiento para las diferentes técnicas del conjunto de datos. En esta pestaña se muestra información acerca de una técnica de MITRE que el usuario podrá seleccionar entre todas las presentes tanto en la matriz *Enterprise* como *ICS*. Para poder seleccionar la técnica, el usuario debe ir filtrando la información mediante la selección de los diferentes desplegables, para reducir el conjunto de datos con el que se trabaja internamente, al igual que en el anterior caso de uso. Primeramente, se seleccionará el tipo de matriz de MITRE, que servirá para

filtrará la información del próximo selector, que corresponde con el listado de todas las tácticas de la misma. Una vez seleccionada la táctica, se podrá seleccionar entre todas las técnicas asociadas.

Una ves escogida una técnica, se muetra tanto el nombre como la descripción de la misma. Por otro lado, en la sección inferior se representa en formato tabla los diferentes CVEs que están asignados a esa técnica, mostrando información relevante de la vulnerabilidad (véase *Ilustración 6. Emparejamiento de técnicas*). Puede darse el caso de que alguna técnica no tenga asociado ningún CVE por el modelo CVE2TTPs, en cuyo caso la tabla se presenta vacía, y el listado de técnicas del selector muestra un mensaje informativo.

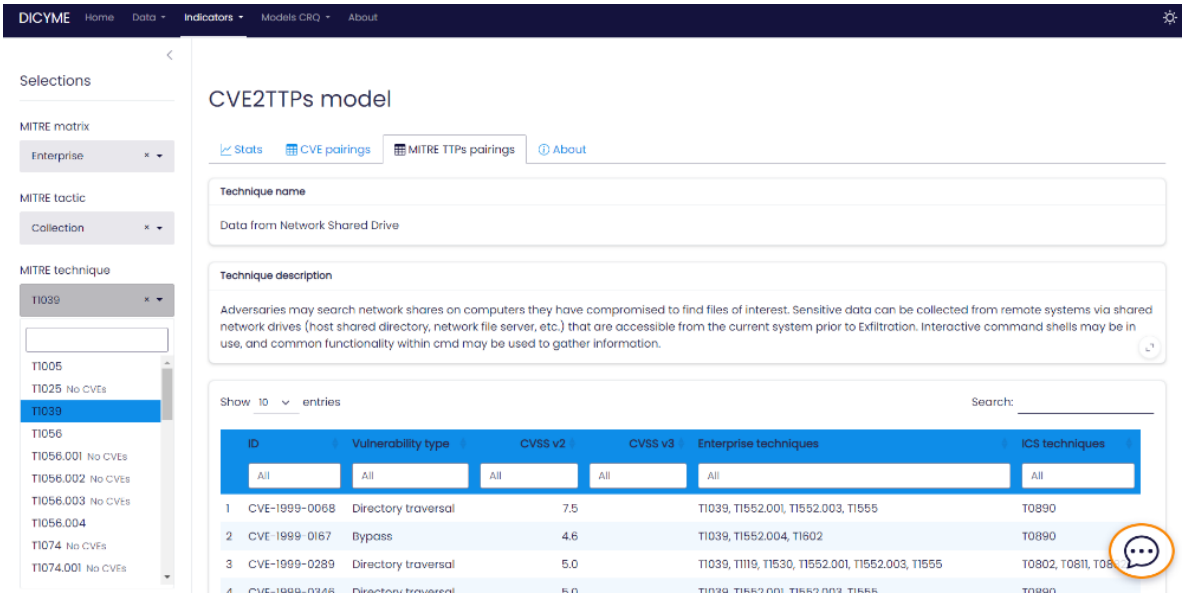


Ilustración 6. Emparejamiento de técnicas

2.6 Modelo de cuantificación del ciberriesgo

Adicionalmente, se ha rediseñado todo el módulo de cuantificación del ciberriesgo añadiendo diferentes mejoras, desde la presentación y navegación de los datos, hasta nuevas funcionalidades como la posibilidad de modificar los valores intermedios de la simulación, descargar un informe completo con los resultados, o interactuar con un *chatbot* inteligente que ayuda al usuario con los resultados.

2.6.1 Nueva navegación

La primera y principal es el rediseño de la navegación, debiendo el usuario seleccionar de la barra de navegación entre entidades del *dataset* (conjunto de datos de perfil de víctimas) o víctimas potenciales (véase la *Ilustración 7. Desplegable del módulo de CRQ en la barra de navegación*). El flujo posterior, así como la UI, es muy similar en ambas situaciones para que al usuario le resulte familiar. Tan sólo varía la manera en que se introducen los datos y parámetros.

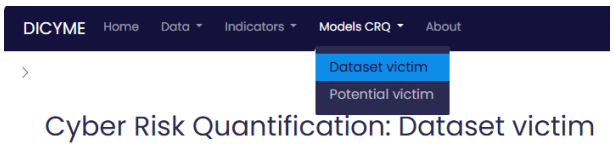


Ilustración 7. Desplegable del módulo de CRQ en la barra de navegación

La primera pantalla que se le presenta al usuario tras haber escogido un tipo de víctima es la de parámetros y entradas. En ambas situaciones, los campos de entrada de datos están relacionados con los homólogos de los apartados de indicadores. Es decir, si en el indicador de actores de amenaza el usuario escoge una empresa del *dataset*, en esta pestaña le aparecerá ya seleccionada dicha entidad. De igual modo, si introduce parámetros de una víctima potencial en el cálculo del Atractivo, por ejemplo, los campos tendrán el mismo valor en este módulo.

En el caso de las empresas del conjunto de datos (véase la *Ilustración 8. Parámetros del módulo de CRQ para una entidad del conjunto de datos*), cuando el usuario escoge una en el panel superior izquierdo, se muestran los valores presentes en el *dataset*, para que el usuario tenga conocimiento de estos. Para las empresas potenciales, tendrá que rellenar cada uno de ellos. En el panel superior derecho, en ambos casos el usuario tendrá que subir un fichero de texto plano que contenga los identificadores de las vulnerabilidades (códigos CVE) que sufre la entidad. Estos datos no se encuentran en ningún conjunto de DICYME dada su complejidad para conseguirlos por cauces lícitos de fuentes abiertas, es decir, sin necesidad de ejecutar escáneres de vulnerabilidades previa autorización de las entidades. En los paneles inferiores, se deben introducir variables de la simulación: cuántas se desea realizar y con qué semilla (un valor numérico usado para asegurar la reproducibilidad de las operaciones aleatorias); así como valores constantes de aseguradoras empleados en los cálculos: el coste medio de una hora de investigación forense, el coste medio de los equipos que puedan verse dañados, así como el valor estadístico de una vida humana. Estos cinco valores, en caso de modificarse, se actualizan automáticamente entre la pestaña de entidades del *dataset* y de víctimas potenciales, de manera que el usuario no tenga que estar actualizándolas constantemente si desea que los valores no sean los propuestos por defecto.

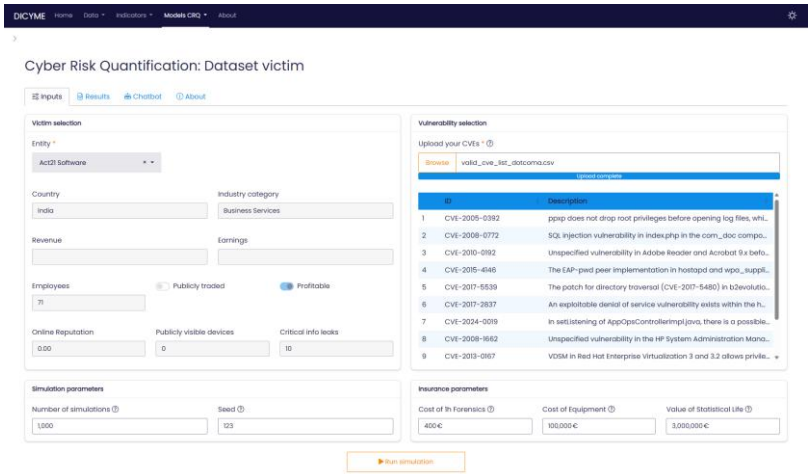


Ilustración 8. Parámetros del módulo de CRQ para una entidad del conjunto de datos

2.6.2 Presentación de resultados

Respecto a la presentación de los resultados, se han incluido las dos principales gráficas empleadas en este tipo de simulaciones: la distribución de eventos de pérdidas en los N años simulados, así como la curva de pérdidas esperadas, que representa la probabilidad de sufrir una pérdida mayor o igual que el valor del eje X. Se ha mantenido el árbol del ciberriesgo, combinando con los valores medios obtenidos en los cálculos intermedios. Todos los valores se obtienen siguiendo el modelo de CRQ detallado en los entregables E4.2 y E4.3. Para aumentar más aún la interactividad del usuario, los parámetros parciales que definen la frecuencia de eventos de pérdidas son modificables todos ellos, de manera que el usuario puede recalcular los resultados simulando qué pasaría si los valores fueran diferentes. Esto sucede para *Baseline*, *Incidents* y *Attractiveness (ATR2ATK)*, que definen la frecuencia de eventos de amenaza, por un lado, y para el indicador de actores de amenazas, la puntuación de técnicas de MITRE ATT&CK, y el perfil de seguridad, que definen la probabilidad de que dichos eventos sean exitosos (véase la *Ilustración 9. Resultados del módulo de CRQ para una entidad del conjunto de datos*). Este proceso no se ha realizado para las pérdidas, pues dependen directamente de entradas del usuario: vulnerabilidades y constantes de seguros, y se puede realizar en la funcionalidad explicada a continuación.

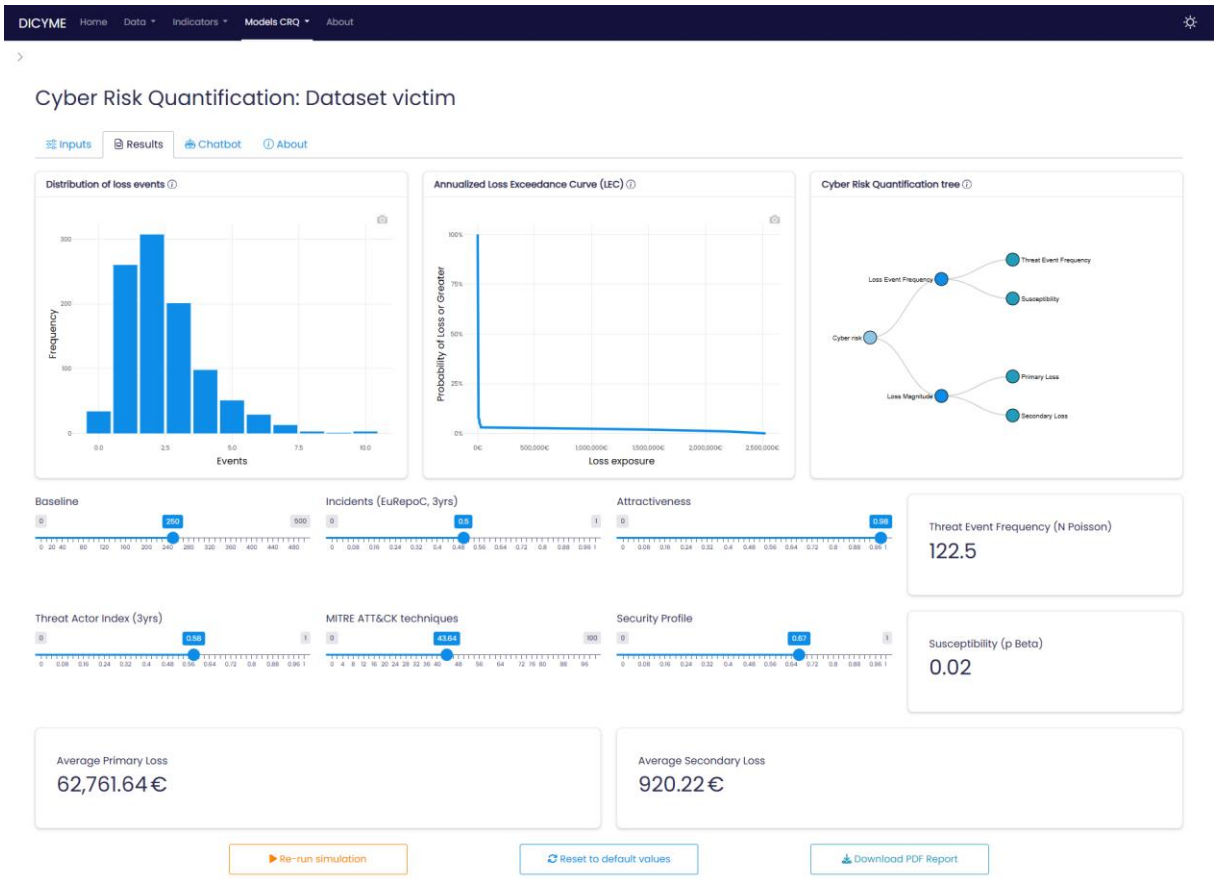


Ilustración 9. Resultados del módulo de CRQ para una entidad del conjunto de datos

2.6.3 Chatbot

Como se describe en la sección de novedades *General*, este módulo tiene un asistente IA mejorado respecto al presente en el resto de las pestañas (introducido en el entregable E4.3), que únicamente proporciona información en base a unas indicaciones predefinidas que el usuario no puede modificar. En este caso, el usuario puede interactuar con el asistente inteligente, pudiéndole solicitar acciones, información adicional sobre los resultados, etc.

2.6.4 Informe descargable generado en tiempo real

Para completar el soporte a la toma de decisiones y ayudar al usuario y responsables de las instalaciones a tomar acciones y distribuir información sobre la cuantificación realizada en la aplicación DICYME, se ha mejorado considerablemente el informe PDF descargable, que está disponible para los dos tipos de víctimas. Incluye un resumen de todos los parámetros del usuario, valores obtenidos de *datasets* del proyecto, así como todas las puntuaciones intermedias y finales calculadas. También contiene los gráficos de distribución de eventos de pérdidas y curva de pérdidas esperadas, así como la representación del árbol para mejor comprensión de los lectores del informe. Finalmente, se incluye un apéndice con detalles sobre el modelo de cuantificación del ciberriesgo, de manera que el lector conozca cómo se llega a los resultados mostrados.

El

ANEXO A. INFORME DE RESULTADOS DEL CIBERRIESGO contiene detalles sobre el informe generado para una entidad escogida aleatoriamente.

El informe se genera empleando Quarto, con una plantilla adaptada para los dos casos de uso: empresas del conjunto de datos de perfil de víctimas y víctimas potenciales definidas por el usuario. A partir de esta plantilla, se automatiza la generación del informe PDF combinando datos parametrizados, cálculos intermedios y salidas gráficas derivadas del modelo DICYME. La plantilla está escrita en R Markdown extendido y estructurada para que, al compilarse, incluya tanto los datos de entrada como los resultados obtenidos mediante simulaciones estocásticas basadas en FAIR.

Además, el diseño modular de la plantilla permite modificar fácilmente el formato de salida, por ejemplo, pasando de PDF a **HTML** o **Word** con simples ajustes en la cabecera del documento (format:), lo cual habilita su uso en plataformas web o entornos colaborativos.

Entre las funcionalidades implementadas, destaca la incorporación automática de:

- Tablas con los parámetros completos de la entidad analizada.
- Resultados estadísticos de los escenarios simulados.
- Gráficos explicativos como el histograma de eventos, la curva de pérdida esperada (LEC) y el árbol CRQ interactivo como imagen (captura de pantalla).
- Un apéndice técnico que documenta detalladamente el modelo de cuantificación del riesgo, facilitando la trazabilidad y validación de los resultados, así como la comprensión de todo el proceso de cuantificación y los resultados producidos.

2.7 Agente IA

Como se introduce en el entregable E4.3, se ha incluido en la aplicación un agente IA que es capaz procesar y generar respuestas con el objetivo de explicar los datos y visualizaciones de todas las pestañas dentro de la aplicación. Para poder realizar peticiones al modelo desde la aplicación de DICYME se ha construido un API en Python usando la librería de Flask. En ella se han desarrollado una serie de *endpoints* a los cuales, dependiendo de la pestaña en la que se encuentre el usuario, se llamará para procesas la información. Dentro de estos *endpoints* se filtrará la información de manera diferente, y también podrán consultar información procedente de la base de datos cuando no sea factible introducirla en el *payload* de la petición web. Una vez obtenidos y filtrados todos los datos se realizará la consulta al modelo LLM, con un *prompt* específico en función de los datos que se estén procesando (según el *endpoint)*. La tarea que realiza el LLM en todos los casos de uso es explicar los parámetros y resultados, su significado, así como patrones detectados o tendencias de los datos o métricas obtenidas.

Durante el proceso de implementación de la capa de explicabilidad de los datos mediante LLMs, se evaluaron distintos modelos hasta seleccionar aquel que ofrecía el mejor rendimiento (relación entre el tiempo dedicado y precisión en las respuestas) para las tareas específicas de la aplicación. En un inicio se empezó utilizando el modelo Llama3.1 pero para ciertos escenarios de la aplicación el modelo generaba respuestas

generalistas y presentaba casos de alucinación. Para mitigarlo se optó por utilizar el modelo de Deepseek r1, donde las respuestas obtenidas en cada pantalla de la aplicación mejoraron considerablemente. Este modelo permitió incorporar más contexto, identificar patrones y tendencias con mayor precisión y ofrecer respuestas más específicas.

Con el objetivo de garantizar respuestas coherentes y precisas, en algunas secciones de la aplicación se ha optado por fragmentar las consultas en aquellos casos en los cuales se manejan grandes volúmenes de datos. Con un simple *prompt* acompañado con grandes volúmenes de datos como contexto, el modelo LLM no es capaz de dar una respuesta coherente ya que este tipo de modelos tienen cierta limitación de tokens de contexto, donde si se sobrepasa dicho límite el modelo dará respuestas incoherentes. Para ello se ha aplicado dicha función de fragmentación, donde se construirán varios *prompts* con los cuales en cada uno de ellos se le aplicará el contexto de los datos fragmentados. Cada *prompt* fragmentado será evaluado por el modelo y las diferentes respuestas se unirán mediante una nueva petición al modelo LLM. Este modelo se encargará exclusivamente de resumir, mostrar los aspectos más fundamentales de los *prompts* y dar una respuesta estructurada. Las pantallas en las que se manejan un gran volumen de datos que implementa dicha *query* fragmentada son las pestañas de *Potential victim* de atractivo y *Stats* en CVE2TTPs.

Para finalizar, esta nueva API implementada se ha desplegado en producción como un nuevo contenedor Docker, totalmente independiente al contenedor de la aplicación principal de DICYME, pero con comunicación directa entre ellas.

Una vez se han explicado aspectos generales y especificaciones de la implementación del modelo LLM para explicar los datos del sistema, se pasará a mostrar cada una de las salidas de cada pantalla de la aplicación. Se mostrará tanto el *prompt* utilizado como la salida producida por el LLM para explicar los datos de cada una de las pestañas de la aplicación. Tanto los *prompts* utilizados como las respuestas obtenidas de cada pestaña, se muestran en el *ANEXO B. PROMPTS Y RESPUESTAS DEL ASISTENTE IA*.

2.7.1 *Chatbot* del módulo de cuantificación del ciber riesgo (CRQ)

En el módulo de cuantificación CRQ no se ha añadido el botón de asistente IA detallado previamente, que en otras pestañas proporciona una explicación en base a un *prompt* predefinido, porque se ha ido un paso más allá: se ha añadido un chat conversacional con el cual los usuarios pueden interactuar, escribiéndole preguntas o solicitándole que realice acciones con los resultados de la simulación de CRQ. Este puede realizar tres tareas principales:

- Responder preguntas generales: el asistente es capaz de responder preguntas generales relacionadas con el sistema de análisis de ciber riesgo, incluidos aspectos como los parámetros utilizados, su significado y su impacto en la simulación. Si el usuario plantea una consulta ajena a la sección de CRQ o al funcionamiento de la propia aplicación, el asistente proporcionará una respuesta orientativa que redirija al tipo de preguntas que sí es capaz de responder.
- Realizar recomendaciones: el asistente está diseñado también para poder realizar un análisis de la simulación y realizar una recomendación final con el que se pueda incrementar o disminuir los parámetros de dicha simulación. Al modelo

se le ha introducido previamente con contexto para que pueda realizar el análisis y dar dichas recomendaciones. El contexto está compuesto por los parámetros de la simulación introducidos por el usuario junto con una serie de simulaciones previamente ejecutadas, donde sobre estas simulaciones se han ido modificando los parámetros para así obtener diferentes resultados. Con este contexto el asistente es capaz de poder mostrar cómo cambia el resultado final si se ajustan los parámetros.

- Recalcular simulación y comparativa con resultados previos: alternativamente el modelo es capaz de realizar una tercera tarea que consiste en recalcular la simulación a partir de una serie de parámetros que pueda especificar el usuario en el *prompt*. Esta simulación solo se ejecutará si el usuario ha seleccionado en la interfaz la opción de '*Ejecutar simulación*'. La simulación no se ejecuta al completo, solo se recalcula el valor de *susceptibility* y *threat event frequency*, mientras que los valores de pérdidas primarias y secundarias se mantienen con el mismo valor que el de la simulación anterior puesto que dependen de los CVEs. El modelo primeramente extraerá del *prompt* los parámetros de la simulación, donde si el usuario no especifica el valor de todos ellos el modelo usará los parámetros de la simulación previa. Con los nuevos parámetros se ejecutará la simulación y finalmente el asistente realizará un análisis y una comparativa con los resultados previos.

La arquitectura ha sido construida empleando la librería de LangGraph que permite diseñar flujos de trabajo complejos de agentes y herramientas (*tools*) mediante un enfoque basado en grafos. El resultado es una arquitectura multi-agente (véase la ***Error! Reference source not found.***), en la que cada uno de los cuatro agentes está diseñado para encargarse de una tarea específica:

- *Coordinator*: se encarga de analizar el *prompt* introducido por el usuario y determinar cuál debe ser la siguiente etapa de ejecución dentro del flujo establecido. En función del contenido y naturaleza del mensaje, este agente decide si se debe derivar la iteración hacia el agente Recommender, Responder o Simulator. Su función es puramente de enrutamiento, donde su decisión está basada en la interpretación del tema principal del *prompt*. Debido a su propósito específico, no se requiere acceso al contexto ni al uso de herramientas externas, ya que su única responsabilidad es identificar la intención del usuario.
- *Responder*: este agente es uno de los tres agentes principales, encargado de atender preguntas relacionadas con cuestiones que referencian al proyecto de DICYME, más concretamente a cuestiones que hablen acerca del módulo de Cyber Risk Quantification. A partir del *prompt* del usuario y el acceso al contexto asociado a los parámetros de CRQ, como el atractivo o el actor principal score entre otros, el agente genera respuestas informativas y relevantes. Su capacidad de respuesta está limitada a una serie de consultas vinculadas al sistema de CRQ como puede ser interpretación y significado de tanto los parámetros utilizados como los resultados obtenidos de la simulación, métodos empleados para ejecutar las simulaciones o consultas generales relacionadas con el módulo. Si el usuario ingresa un *prompt* que no se relaciona con las cuestiones previamente expuestas, el agente mostrará un mensaje indicando que la consulta es inválida

en ese contexto, proporcionando una lista de temáticas sobre las que se puede consultar información.

- *Recommender*: agente encargado de realizar una recomendación final en función del prompt del usuario que se le proporciona. En función de lo que el usuario indique en el prompt acerca de los valores de los parámetros, el agente recomendador dará una recomendación sobre posibles ajustes que se puedan realizar sobre los parámetros de entrada de la simulación para obtener valores más bajos de riesgo. El agente previamente contará con un cierto contexto sobre posibles ajustes que se pueden realizar para mejorar el riesgo, evitando así que el agente de respuestas generales y poco específicas sobre la simulación.
- *Simulator*: este agente solo se ejecutará si desde la interfaz se selecciona la opción de *'Simulate'* dentro del chat. Primeramente, el agente a partir del *prompt* del usuario extraerá los valores de los parámetros de la simulación, donde si en el *prompt* el usuario no especifica el valor de todos ellos se asignarán el valor a esos parámetros, los valores de la simulación previa. Este agente ejecutará su *tool* asignada que se encarga de realizar la simulación a partir de los parámetros previamente obtenidos. Una vez se ha reejecutado la simulación, el modelo analizará y comparará los resultados obtenidos con los resultados sobre los que se compara, es decir, los resultados de la simulación previa.

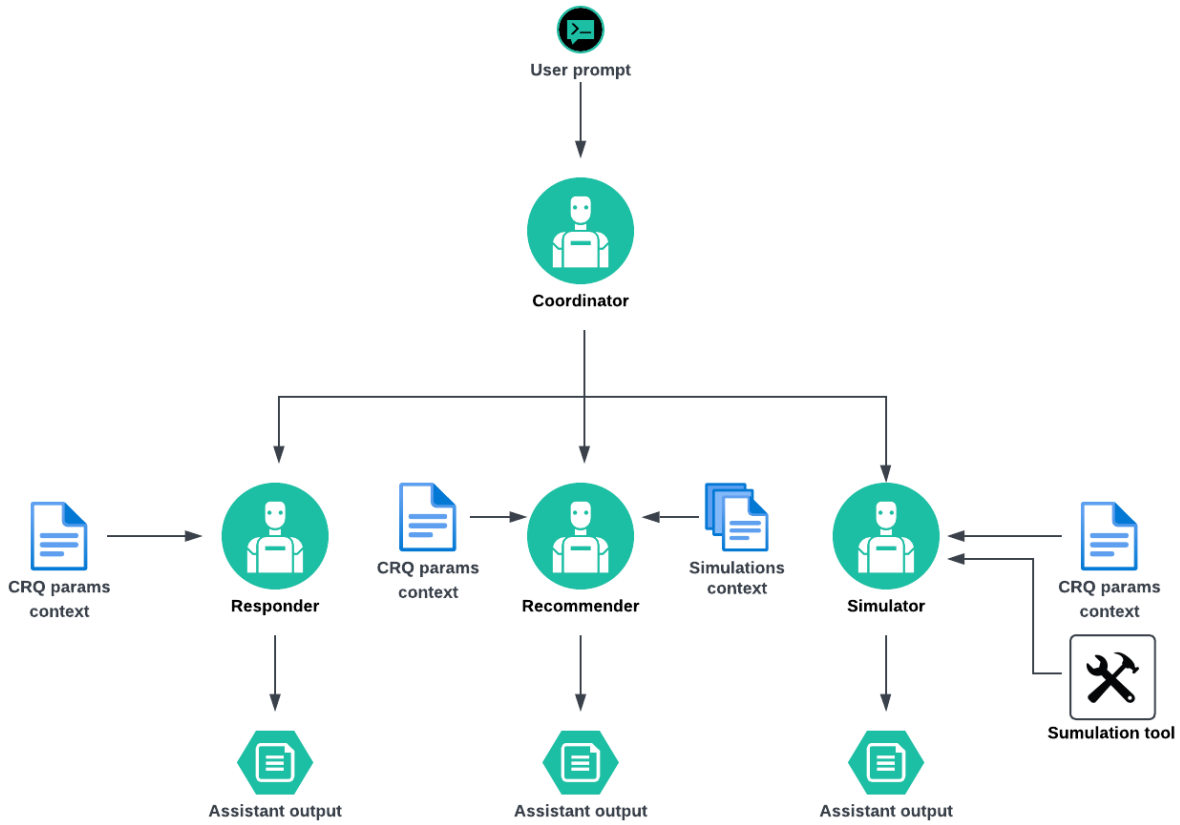


Ilustración 10. Diagrama de arquitectura del chatbot

Cada agente utiliza el modelo Llama 4 Scout, que es la versión más reciente de Llama hasta la fecha de finalización del entregable. Se ha utilizado este modelo en concreto

debido a la configuración de la implementación del asistente, ya que al haberse implementado una arquitectura de multi-agentes con LangGraph el modelo Deepseek no puede utilizarse con esta librería. Por ello se ha utilizado esta versión de Llama más actualizada que cuenta con mayor capacidad de procesamiento, está más actualizado en cuanto a contexto y es más robusto a la hora de generar las respuestas, aunque emplea un mayor tiempo en contestar.

La ***Error! Reference source not found.*** muestra la interfaz a través de la cual el usuario puede interactuar con el modelo. Esta contiene una entrada de texto a través del cual el usuario puede interactuar con el asistente, así como un botón de enviar. En la sección superior derecha aparece un botón con el texto "Simular" ("*Simulate*") que habilita o deshabilita la capacidad de simulación. Si se activa esta opción, se fuerza al *chatbot* a realizar nuevamente la simulación con los parámetros que se indiquen en la entrada del usuario.

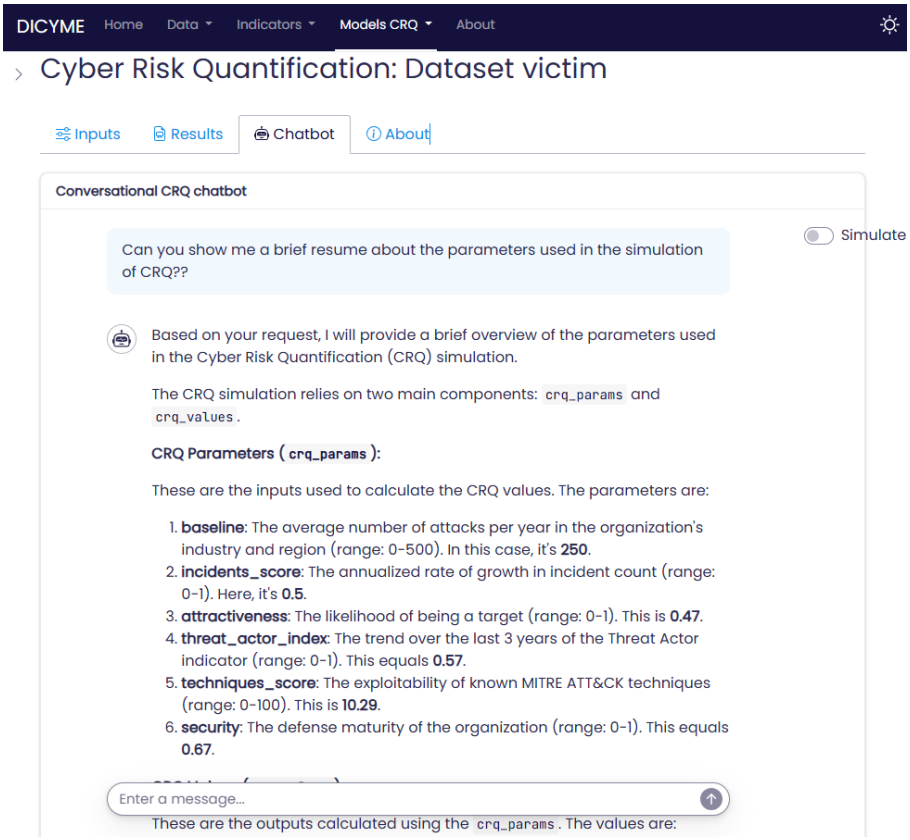


Ilustración 11. Pestaña del chatbot recomendador

2.8 General

Para mejorar la experiencia de usuario y proporcionar una mayor transparencia sobre los procesos en ejecución, se han incorporado una serie de pop-ups que incluyen una barra de progreso que indica el estado de las tareas que se están ejecutando en segundo plano. Estas ventanas emergentes se muestran durante la carga y procesamiento de los conjuntos de datos, que son procesos que ocupan varios segundos, así como en las simulaciones. De esta manera, el usuario está informado del motivo por el cual la aplicación está ocupada y tarda en generar nuevas visualizaciones., donde

primeramente se realiza la petición a la base de datos para obtener los. El pop-up (véase la *Ilustración 12. Ventana emergente de progreso de carga de datos*) se posiciona en la parte inferior derecha, e incluye un mensaje informativo del estado de los procesos en ejecución.

Además, para reducir los tiempos en la carga y procesado de datos, se ha implementado un mecanismo de caché, que almacena en un fichero en memoria de la aplicación (`.rds`) los datos manipulados y preparados para su uso en la aplicación durante un tiempo límite configurable, para que pueda modificarse según la periodicidad con que se actualizan los datos en la base de datos. De esta manera, únicamente se descargan los datos en el primer acceso a la aplicación tras su despliegue y cumplido dicho límite, reduciendo considerablemente el tiempo que debe esperar el usuario cuando se emplean conjuntos de datos de grandes dimensiones.

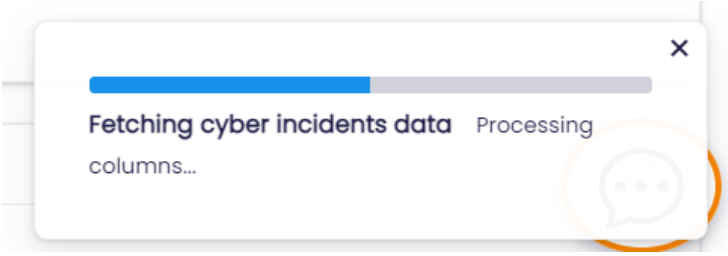


Ilustración 12. Ventana emergente de progreso de carga de datos

3 DATOS Y CÓDIGO

En cuanto al desarrollo y despliegue no se ha modificado el flujo de trabajo definido desde el comienzo y detallado en el entregable E3.1. La aplicación es accesible en la URL `https://gondor.etsii.urjc.es:3866/` y se actualiza en intervalos cortos de desarrollo, generalmente de dos semanas, introduciendo las nuevas funcionalidades que se van implementando, así como generando el paquete de R con el código fuente para cada versión.

En el enlace deriskGroup / DICyME Project · GitLab, dentro del directorio *E.3.3 Final prototype of a graphical interface and decision support tool*, se puede encontrar el directorio `dicymeviz-1.0.0/` que contiene el código de la aplicación en formato paquete de R. El archivo README.md detalla la estructura y modo de ejecución de la aplicación.

Caben destacar algunos elementos del código, que se mantienen respecto a los entregables E3.1 y E3.2:

- `renv.lock` contiene las versiones de todos los paquetes de R empleados, así como sus dependencias y versiones de estas.
- `R/` contiene el código R con las diferentes funciones que componen la aplicación.
- `.github/workflows/` contiene el código con las automatizaciones del despliegue y análisis estático.
- DESCRIPTION contiene la información del paquete, versión, autores, etc.

Aunque las versiones de todas las dependencias aparecen especificadas en el archivo renv.lock y se pueden restaurar de manera sencilla en cualquier entorno de desarrollo, cabe destacar las siguientes:

- R 4.4.1
- Shiny 1.9.1
- bslib 0.9.0.9000 (actualizado)
- shinychat 0.2.0 (nuevo paquete para el asistente conversacional)

Asimismo, el servicio de API de Python requiere de una serie de dependencias para su correcta ejecución. Las dependencias pueden ser instaladas a partir del archivo requierements.txt del proyecto, incluyendo las siguientes:

- DateTime 5.5
- Flask 3.1.0
- numpy 2.2.3
- ollama 0.4.7
- pandas 2.2.3
- pymongo 4.11.1
- python-dotenv 1.0.1
- sshtunnel 0.4.0
- cheroot 10.0.1
- requests 2.32.3

Adicionalmente, la API requiere una serie de variables de entorno, que incluyen las credenciales de la base de datos de MongoDB y las credenciales para establecer el túnel SSH a la aplicación. El archivo que incluye todas estas variables debe incluirse en la carpeta llamada config, donde el archivo debe de llamarse .env. El proceso completo se describe, junto a la arquitectura completa del sistema, en el entregable E3.4.

4 CONCLUSIONES Y SIGUIENTES PASOS

En este punto del proyecto se ha desarrollado un último prototipo de la aplicación web del proyecto. Tras el segundo prototipo, en este se ha mejorado todo el módulo de cuantificación del ciberriesgo y soporte a la toma de decisiones, añadiendo funcionalidades como el asistente conversacional, la posibilidad de poder modificar los parámetros intermedios de las simulaciones, y el informe final. Todas estas buscan contribuir a una mejor y más informada toma de decisiones por parte de los responsables, tanto aseguradoras como gestores de las infraestructuras, personal de ciberseguridad, etc.

Además, se ha integrado la ejecución en tiempo real de todos los modelos en los que era abordable dado el alcance e infraestructura necesaria y disponible, y se ha

desarrollado la funcionalidad transversal de poder obtener resultados para una víctima potencial, pudiendo obtener sus datos y la cuantificación del ciber riesgo final. Asimismo, se ha mejorado considerablemente la experiencia de usuario, facilitándole información en los procesos largos, manteniendo una interfaz similar y reconocible para simplificar la navegación, haciendo que los parámetros de entrada se mantengan entre diferentes pestañas para que el usuario no deba introducirlos o modificarlos repetidas veces, etc. Con estos avances se consigue una aplicación mucho más usable, intuitiva y sencilla de usar por personal tanto técnico como menos avanzado, logrando alcanzar a un público muy amplio: responsables y trabajadores de las infraestructuras, personal de aseguradoras, especialistas de ciberseguridad y ciberriesgo, estudiantes, etc.

Respecto a la arquitectura, se ha conseguido desarrollar un sistema completo *end-to-end* que muestra al usuario todos los resultados del proyecto, desde el análisis de los conjuntos de datos hasta su transformación indicadores útiles y su uso en la cuantificación del ciberriesgo. Basada en microservicios y con una arquitectura modular, este último prototipo pone de manifiesto lo sencillo que es añadir componentes al sistema (en este caso, la API para la interacción con el LLM) sin modificar el resto de la aplicación.

El sistema DICYME está alcanzando sus últimas etapas de desarrollo, dado el fin próximo del proyecto. En esta etapa, los equipos se encuentran realizando pruebas y analizando en detalle los resultados del sistema, para poder terminar de afinar la precisión de los resultados, fallos en la experiencia de usuario, etc. De manera paralela, los equipos trabajarán para poder traspasar a DeNexus todo el conocimiento relativo a la aplicación, infraestructura, y demás componentes de manera que puedan sacar todo el beneficio posible al desarrollo, de manera conjunta a su línea de negocio y explotación de resultados.

ANEXO A. INFORME DE RESULTADOS DEL CIBERRIESGO

A continuación, se reproduce el informe generado para la entidad Act21 Software, que forma parte del conjunto de datos de perfil de víctimas de DICYME.

DICYME Cyber Risk Quantification tool

Wednesday, July 16, 2025

This report presents a cyber risk quantification assessment for **Act21 Software**. The aim is to estimate potential cyber losses using stochastic simulations based on FAIR principles and DICYME-specific indicators.

1 Cyber Risk Quantification simulation

Given the defined CRQ model, the inputs and results are presented hereunder.

1.1 Entity parameters

These are all the parameters available for the entity, either because they are in DICYME datasets or because they have been entered by the user. They are used into DICYME models and statistical distributions and have a direct impact on results. The available data for **Act21 Software** are shown in Table 1.

Table 1: Entity parameters

Parameter	Value
Country	India
Industry category	Business Services
Revenue	NA
Earnings	NA
Employees	71
Publicly traded	No
Profitable	Yes
Online reputation	0
Publicly visible devices	0
Critical info leaks	10

Parameter	Value
Vulnerabilities	CVE-2005-0392, CVE-2008-0772, CVE-2008-1662, CVE-2008-4358, CVE-2010-0192, CVE-2012-0181, CVE-2013-0167, CVE-2015-4146, CVE-2016-8483, CVE-2017-2837, CVE-2017-5539, CVE-2018-18250, CVE-2019-4387, CVE-2020-2208, CVE-2021-22182, CVE-2021-37343, CVE-2022-4762, CVE-2022-45611, CVE-2022-46324, CVE-2024-0009, CVE-2024-0019, CVE-2024-0044, CVE-2024-0050, CVE-2024-0054, CVE-2024-0076, CVE-2024-20046

1.2 Simulation parameters

Table 2 contains the simulation parameters that have been introduced by the user. They customize the Cyber Risk Quantification results, modifying the number of simulations that are run, the seed used for reproducibility, and other economic constants.

Table 2: Simulation parameters

Parameter	Value
Number of simulations	1,000
Simulation seed	123
Cost of 1h forensics	400 €
Cost of equipment	100,000 €
Value of statistical life	3,000,000 €

1.3 Simulation results

Given this layout, the following average results have been obtained after **1000** simulations:

- **Estimated Cyber Risk**: 156,021 € expected/year. This is the expected annual financial loss due to cyber incidents, combining both frequency and severity.
  - **Loss Event Frequency**: 2.45 events/year. This value represents how many successful cyber incidents are expected per year.
    - **Threat Event Frequency**: 122.5 events/year. The expected number of attack attempts per year based on threat intelligence, incident trends, and firm characteristics.

    - **Susceptibility**: 2% success. The probability that an attack attempt succeeds, depending on threat actor activity, technical exposure, and security maturity.

- **Loss Magnitude**: 63,682 € per event. This represents the average financial impact of a successful incident.
    - **Primary Loss**: 62,762 €. Direct costs such as operational disruption, asset damage, or extortion payments.
    - **Secondary Loss**: 920 €. Indirect costs including forensic investigations, reputational damage, or regulatory penalties.

2 Distribution insights

The following plots illustrate the distribution of simulated cyber events and their corresponding loss magnitudes. These help to visualize the risk landscape in probabilistic terms.

2.1 Loss event frequency

Figure 1 shows the distribution of the number of successful cyber incidents observed across the **1000** simulation runs. This reflects the expected frequency of cyber loss events over time, taking into account both the likelihood of attacks and the organization's susceptibility. A right-skewed distribution indicates that while most years may have a low number of incidents, there is a non-negligible probability of experiencing multiple events in a single year. This insight is critical for capacity planning and business continuity strategies.

2.2 Loss magnitude

Figure 2 presents the Loss Exceedance Curve (LEC), which visualizes the annualized probability of incurring a loss equal to or greater than a given amount. The X-axis shows the potential loss values, and the Y-axis indicates the probability (%) that losses will exceed those values in any given year. This chart provides a comprehensive view of the financial exposure faced by the organization and supports informed decision-making about risk appetite, insurance needs, and cybersecurity investments. The curve also highlights extreme but plausible loss scenarios, which are often overlooked in average-based assessments.

These insights are intended to support prioritization of cybersecurity investments and mitigation strategies. This assessment provides a quantitative foundation, while final decisions should integrate qualitative expert judgment and broader business context.

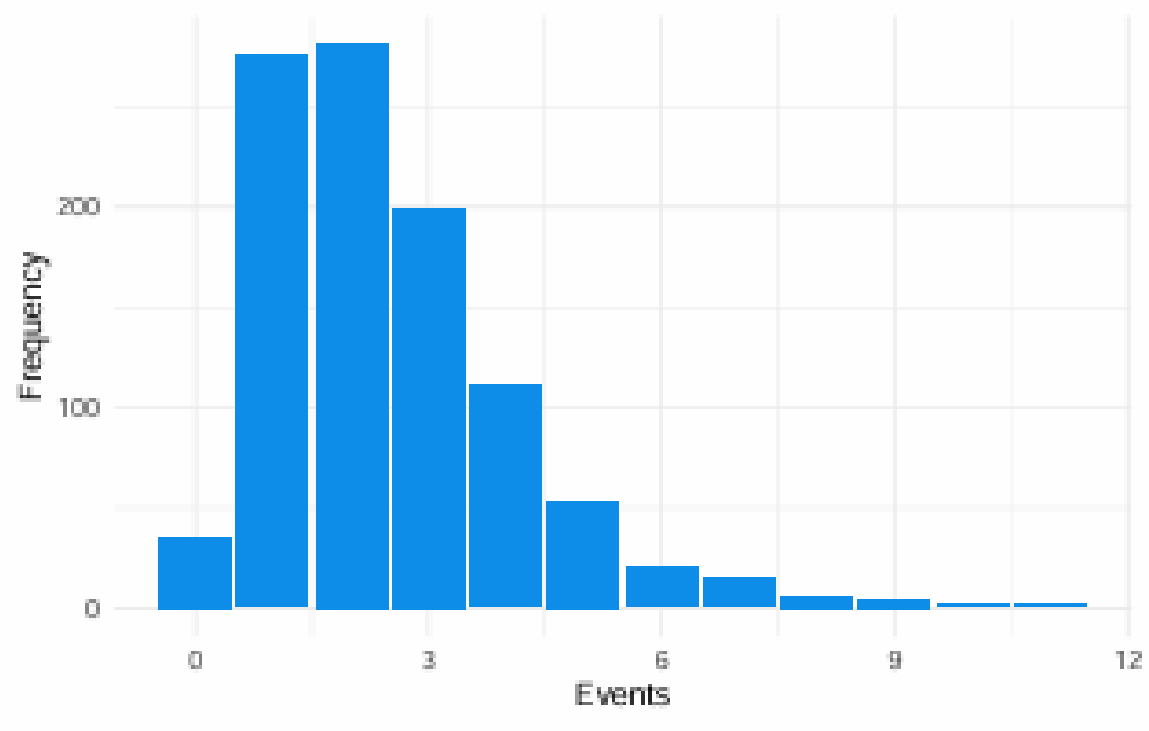


Figure 1: Distribution of loss events

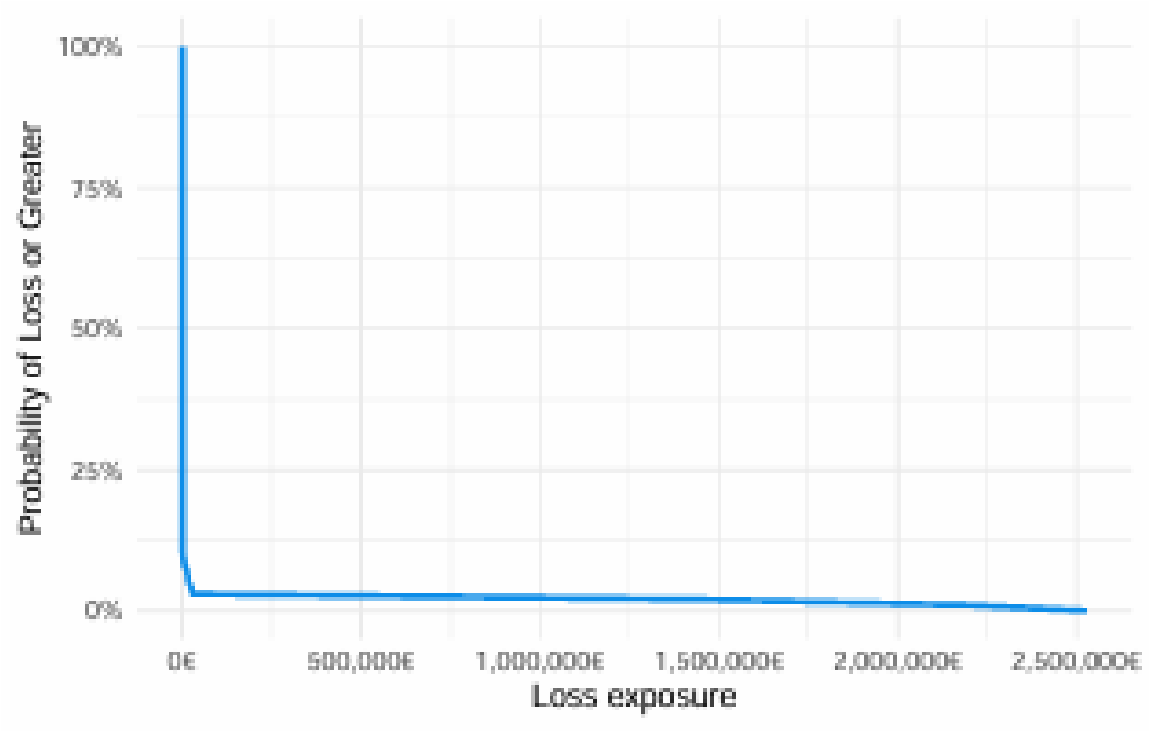


Figure 2: Annualized Loss Exceedance Curve (LEC)

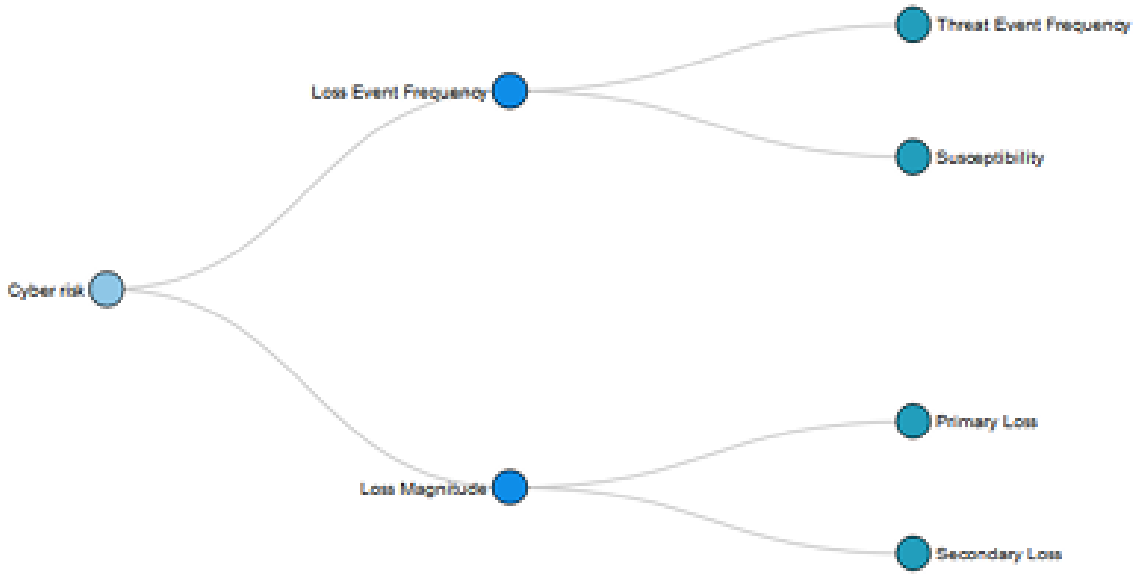


Figure 3: CRQ decomposition tree

3 Cyber Risk Quantification model

Cyber risk is defined as the combination of how often cyber incidents occur and how much they cost when they do. In quantitative terms:

$$\text{Risk} = \text{Loss Event Frequency (LEF)} \times \text{Loss Magnitude (LM)}$$

This metric represents the expected annual financial loss due to cyber incidents and is estimated through stochastic Monte Carlo simulations. These simulations generate realistic outcomes for each component of the model:

- **Loss Event Frequency (LEF)**: the number of successful cyber incidents expected per year.
- **Loss Magnitude (LM)**: the average economic impact per successful incident, including direct and indirect losses.

Together, these yield the final estimate of cyber risk. To make this structure more intuitive, each part is decomposed, obtaining a tree layout (see Figure 3).

3.1 Loss Event Frequency (LEF)

LEF represents the expected number of successful cyber incidents per year. It is defined as:

$$\text{LEF} = \text{TEF} \times \text{Susceptibility}$$

The TEF quantifies how many cyberattack attempts are expected against the organization per year. It is made up of the following components:

- **Baseline**: The average number of attacks per year in the organization's industry and region, based on public third-party reports.
- **Incident rate**: Annualized rate of growth in incident count, computed from historical records.
- **Attractiveness**: The possession of features or the exhibition of behaviours in entities that raise interest for potential adversaries. Thus, the more significant the attractiveness value is, the greater the proneness of an entity to be attacked. It's made up of:
  - Basal attractiveness: relevance of the entity in the world, given its static firmographics.
  - Online reputation: the opinion of individuals and the reach of the entity in social media and social networks.
  - Victimisation: the interest that the entity arouses for potential attackers given its technical online visibility.

Susceptibility models the probability that an attack attempt will succeed, resulting in a successful incident. Its components are:

- **Threat Actor index**: Analysis of the trend over the last 3 years of the DICYME Threat Actor indicator, which is a metric that provides an overview of the threat actor regarding the target country and industry. Considering public data from the Electronic Transactions Development Agency (ETDA), different partial scores are computed regarding the activity of the actor, its capacity and the target.
- **Exposure**: Quantifies the exploitability of known MITRE ATT&CK techniques based on submitted CVEs. It is obtained from the CVE2TTPs model, which is a Machine Learning model that relates Common Vulnerabilities and Exposures (CVEs) to Tactics, Techniques and Procedures (TTPs) from the MITRE ATT&CK framework. From the uploaded entity CVEs, the model selects all that allow an ICS matrix technique. It then computes a weighted index based on CVSS severity and tactic relevance, assigning a higher weight to those vulnerabilities that allow a technique from the Impact tactic.
- **Security profile**: Defense maturity of the organization, indicating ability to detect and contain attacks.

3.2 Loss Magnitude (LM)

Loss Magnitude (LM) represents the financial cost of a successful cyber incident. It is composed of the simulated value of **primary** and **secondary** losses:

- Primary Losses: direct financial costs, such as operational disruption or equipment damage.
- Secondary Losses: indirect costs, such as forensic investigations, fines, or reputational damage.

The process begins with the **list of CVEs** uploaded by the user. Using the CVE2TTPs model, the system determines which **MITRE ATT&CK techniques** from the *Impact* tactic are potentially applicable.

For each technique identified, the system simulates:

- One of the applicable **primary loss types**, which are *Business Interruption*, *Equipment Damage*, *Extortion and Human Damage*.
- All related **secondary loss types**, which are *Forensic Investigation*, *Reputational Damage* and *Penalties*.

Each simulation iteration selects a valid impact technique, samples the corresponding loss values, and computes the total loss magnitude as:

$$\mathrm{LM}_k = \text{Primary Loss}_k + \sum \text{Secondary Losses}_k$$

ANEXO B. *PROMPTS* Y RESPUESTAS DEL ASISTENTE IA

4.1.1 Ciber incidentes

En esta pestaña se han procesado los datos para agruparlos mediante cuartiles si desde la aplicación se agrupan por día, semana o mes para evitar que el modelo tenga problemas de contexto. En caso de que el usuario agrupe por año entonces se agrupará también por año para dar la respuesta ya que hay menos datos y no hay problemas de contexto. Para obtener los datos, desde la aplicación se pasan los valores de los selectores y con esos parámetros desde el servicio de Python se consultan los incidentes y se procesan los datos.

Prompt:

```
### **Cyber Incidents Temporal Trends Analysis**

You are an expert cybersecurity data analyst. Your task is to analyze the **temporal trends** of cyber-incidents based on the provided dataset.


### **Dataset Structure:**

The dataset is a table that contains information of a time series, containing the counts of the total number of recorded cyber-incidents for each **cyber incidents repositories** selected. Columns include:

- **date**: The date on which the cyber incidents were collected.

- **Cyber incidents repository Count**: The total number of cyber-incidents recorded for that period for each **Cyber incidents repository**.


### **Additional Data:**

The names of the **cyber incidents repositories** are: {databases}

The time unit for grouping the data is: {time_unit}


### **Analysis Instructions:**

1. **Cyber-Incident Trends Over Time:**
   - Identify **significant increases or decreases** in the number of cyber-incidents.
   - Highlight **spikes, drops, and periods of stability**.
   - Compare trends across **different time granularities** (daily, monthly, quarterly, yearly).
2. **Comparison Across Cyber incidents repositories:**
   - Analyze **differences in reported incidents** among various **cyber incidents repositories**.
   - Identify which **cyber incidents repositories** report the **highest** and **lowest** number of incidents.
3. **Upload Delay Analysis:**
   - Detect **possible delays** in incident reporting by observing sudden peaks or missing data.
   - Compare reporting patterns among **cyber incidents repositories**.
4. **Temporal Patterns:**
   - Examine the **seasonality** of incidents: are there recurring peaks in certain months or quarters?
   - Analyze whether incident counts follow a **long-term upward or downward trend**.


### **Dataset in DataFrame format:**

{}


### **Response Guidelines:**
```

```
- The response must be **a single concise paragraph** (**max 300 words**).
- **IMPORTANT**. Group the data by each **cyber incidents repository** (Columns of the dataset).
- **IMPORTANT**. Analyze the cyber incidents by the date correctly.
- **FOCUS** on the date of the cyber incidents to analyze the time series.
- Analyze **ALL** **cyber incidents repositories** and highlight key patterns.
- **DO NOT** speculate on causes or motivations—focus on observable trends.
- Include **specific numeric examples** from the dataset.
```

Respuesta:

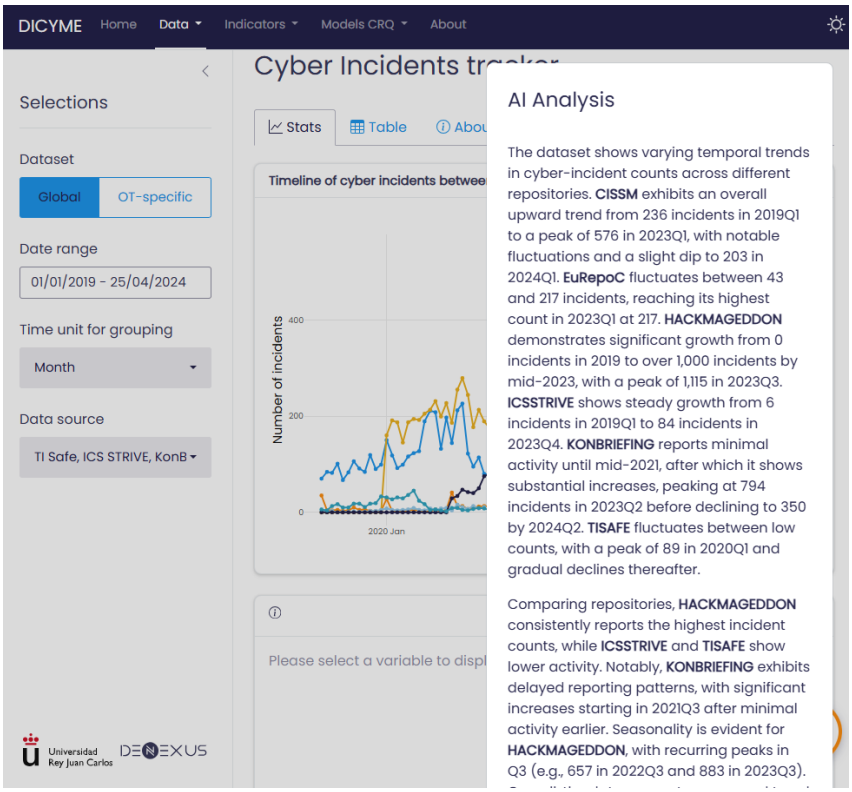


Ilustración 13. Respuesta del agente IA en la pestaña de Cyber Incidents

4.1.2 Victim profile

Se obtienen los datos a través de la base de datos a partir de los parámetros devueltos por la aplicación de R, donde en la parte de Python se realizará el mismo procesamiento que el que se hace en R para visualizar los datos.

Prompt:

```
### **Victim Profile Analysis**

You are an expert cybersecurity data analyst. Your task is to analyze the **incidents** data reported for each country.


The provided dataset is a DataFrame object where each row represents a group of country with the incidents count reported. Each column represents:

- **Country**: The country name

- **Count**: The number of incidents reported in the country


### **Additional Information:**
```

```
The dataset contains aditional information like:
- **Number of victims** of the dataset: {number_victims}
- **Number of non-victims** of the dataset: {number_non_victims}
- **Start date** of the incidents selected: {from_date}
- **End date** of the incidents selected: {to_date}
- **Type of events** selected: {events_type}

### **Analysis Instructions:**
1. **Identify** the countries with the highest number of incidents and compare their counts.
2. **Start** the response with a brief resume of the additional information provided, like the number of countries victims, dates...
2. **Highlight** the countries with the lowest incident counts and compare them to high-incident countries.
3. **Provide a general descriptive comparison** without explaining causation.

### **Dataset in JSON format:**
{}

### **Response Guidelines:**
- The analysis must be **a single concise paragraph** (max 250 words).
- Focus on **ranking and comparing** the most and least affected countries.
- Avoid **explaining causation and motivations** behind the data.
```

Respuesta:

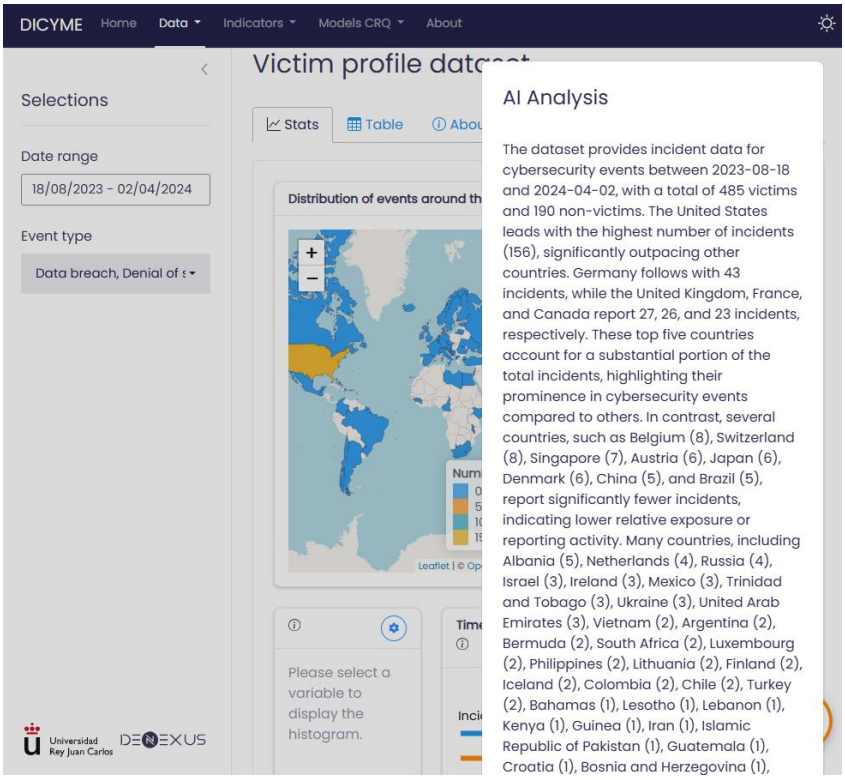


Ilustración 14. Respuesta del agente IA en la pestaña Victim Profile

4.1.3 IDS

En función del tipo de indicador que se seleccione desde la aplicación, en el servidor del LLM se filtrará la información por el alcance seleccionado. Adicionalmente en el *prompt* se le indicará información relevante del significado y valor de cada uno de los parámetros como son los nombres de todos los indicadores, *facility* o alcance.

Prompt:

Cybersecurity Temporal Analysis

You are an expert cybersecurity data analyst. Your task is to analyze **temporal trends** in indicator values, identifying patterns, anomalies, and significant variations.

Dataset Explanation:

The dataset is a list of DataFrames, where each table is the information of each indicator id, filtered by user-selected inputs that contains the following columns:

- {indicator_id}

- indicator_value: The numeric value of the indicator.

- date_uploaded: The date on which the indicator value was taken

Additional Information:

The dataset contains aditional information like:

{scope}

{facility}

{indicator_name}

{indicator_unique}

Analysis Requirements:

1. **Trend Identification**:

- Analyze whether the **indicator values** are increasing, decreasing, or fluctuating over time.

- Identify **patterns, outliers, or sudden shifts** in the values.

2. **Comparative Analysis Across Groups** (if multiple groups exist):

- Compare how different **groups (cyber assets, networks, levels, etc.)** behave over time.

- Highlight differences and correlations between them.

3. **Significant Findings & Insights**:

- Identify **peak periods**, sudden drops, or stable trends.

4. **Brief resume** of the data, mention the indicator_name, indicator_description...

Dataset (DataFrame format):

{}

Response Guidelines:

- The response must be **a single concise paragraph** (max 200 words).

- Begin with a brief **summary of the dataset** (including the indicator name and description).
- Avoid reccomendations and avoid provide conclusions about the data.
- Highlight **specific numeric trends**, patterns, or anomalies.
- If applicable, compare the behavior of different groups over time.

Respuesta:

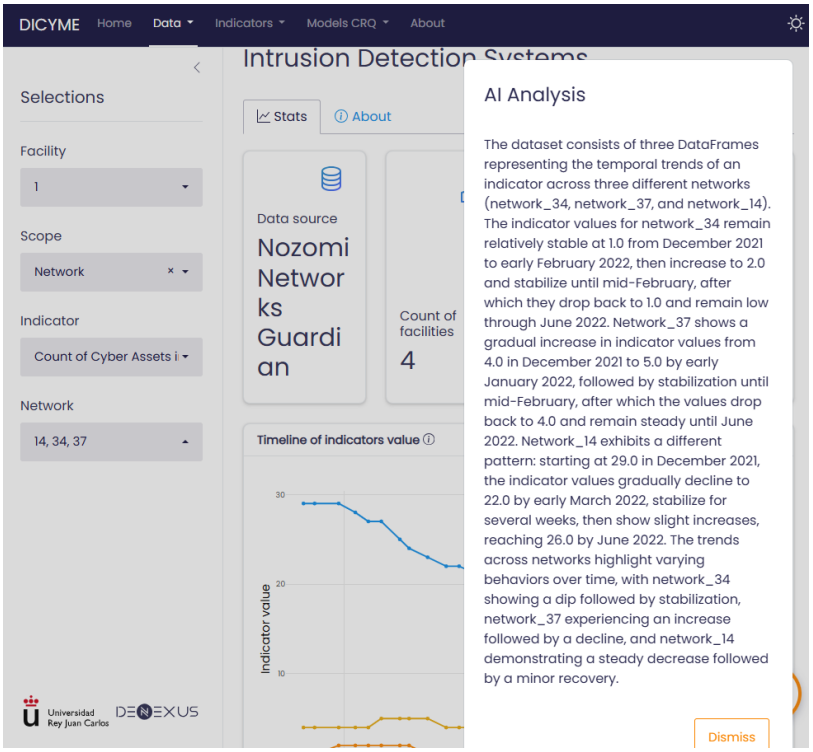


Ilustración 15. Respuesta del agente IA en la pestaña IDS

4.1.4 Threat actors overview

Desde el servidor de Python se obtienen los datos de actores de amenazas de la base de datos de MongoDB y se realiza el mismo filtrado de datos que se produce en la aplicación de R para la visualización.

Prompt:

```
### **Threat Actors Overview Analysis**

You are a cybersecurity threat intelligence analyst. Your task is to analyze the provided threat actor dataset and extract insights based on activity, objectives, target regions, and industries.


### **Data Structure:**

The provided data is a list that contains DataFrames, where each DataFrame represents:

- **objectives_counts**: A DataFrame that contains the counts of the different objectives of a threat actor. Rows:

    -- **objectives**: The type of objetive

    -- **frequency**: The number of each objective

- **target_regions_counts**: A DataFrame that contains the counts of the different regions that a threat actor targets. Rows:

    -- **target_regions**: The name of the region
```

    -- **frequency**: The number of each region

- **target_industries_counts**: A DataFrame that contains the counts of the different industries of a threat actor. Rows:

    -- **target_industries**: The name of each industry

    -- **frequency**: The number of each industry

Additional Information:

The dataset contains aditional information like:

- Amount of threat actors: {amount_threat_actors}

- The extraction dates of the dataset, formatted for readability: {extraction_dates}

- Number of threat actor actives: {threat_actors_actives}

- Number of threat actor dormant: {threat_actors_dormant}

Analysis Instructions:

1. Identify the most and least frequent active categories and describe the distribution.

2. Highlight the primary objectives of threat actors and indicate any dominant trends.

3. Examine the most frequently targeted regions and industries, noting any concentrations.

4. Compare the dataset's overall volume of threat actors to past trends (if applicable).

5. Provide a concise summary of key findings, emphasizing major risk areas.

Dataset in JSON format:

{}

Response Guidelines:

- The response must be **a single concise paragraph** (max 300 words).

- Focus on comparisons, trends, and significant findings.

Respuesta:

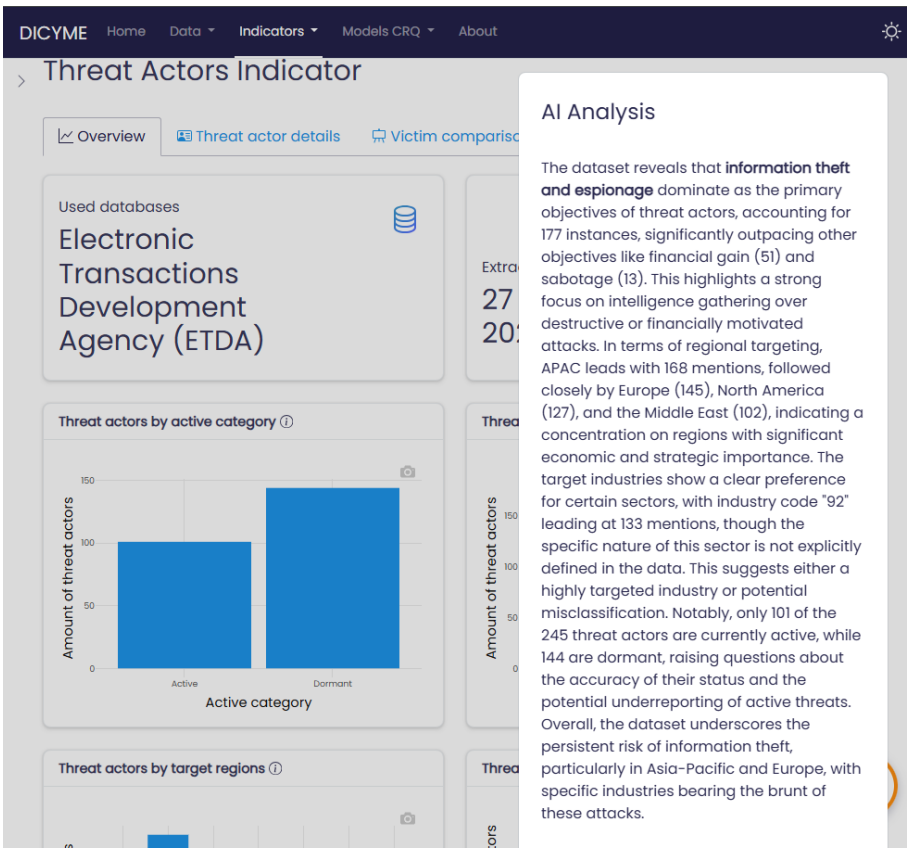


Ilustración 16. Respuesta del agente IA de la pestaña Threat Actors Overview

4.1.5 Threat actors details

Desde la aplicación se enviará una solicitud al servidor de Python, introduciendo como cuerpo de la petición los valores de los selectores usados en dicha pestaña. Los parámetros son usados desde el servidor de Python para filtrar la información tal y como se muestra desde la aplicación, donde con todos estos datos se ingresarán como contexto en el *prompt*.

Prompt:

```
### **Threat Actor Detailed Analysis**

You are a cybersecurity data analyst. Your task is to analyze a dataset containing threat actor details
and their associated risk scores.


### **Data Structure:**

The provided data contain the next information:

- The official name of the actor: {threat_actor_name}

- The mean indicator of the threat actor: {mean_threat_actor}

- The current status of the threat actor: {activity_category}

- The current score of activity of the threat actor: {activity_score}

- The capacity of the threat actor: {capacity_score}

- The date when this data was last updated: {extraction_date}

- The first recorded activity of the actor: {first_activity_date}
```

- The most recent recorded activity of the actor: {last_activity_date}

- Alternative names used by the actor: {threat_actor_aliases}

- The primary goals of the actor: {objectives}

- The countries targeted by this actor: {target_countries}

- The industries affected by this actor: {target_industries}

The provided data also contains an **actor_score** table that provides threat assessment scores based on actor targets. It is in DataFrame format:

-- **Target country**: The country being analyzed.

-- **Target industry**: The industry affected (translated from NAICS codes).

-- **Actor Target Score**: The threat level score assigned to this country-industry pair.

-- **Actor indicator**: A secondary metric to assess risk severity.

Analysis Instructions:

1. **Summarize the actor's details**: Briefly list their name, status, aliases, first and last activity, objectives, and primary target regions. Do NOT explain them in depth, just mention them.

2. **Perform an in-depth analysis of risk scores**:

- Identify which **countries** have the highest and lowest threat scores.

- Identify which **industries** have the highest and lowest threat scores.

- Compare the **Actor Target Score** and **Actor Indicator**, highlighting significant patterns with the actor details data.

- Detect if any country-industry pair has an unusually high or low risk assessment.

3. **Analyze relationships between actor details and scores**:

- Check if threat actors classified as **active** have higher scores than inactive or dormant ones.

- Determine if actors with a history of **longer activity (older first_activity_date)** tend to have higher threat scores.

- Identify whether actors with specific **objectives** (e.g., espionage vs. financial theft) are more likely to target high-risk countries or industries.

- Look for patterns between target **regions and industries** in relation to threat scores.

Dataset of the actor_score in DataFrame format:

{}

Response Guidelines:

- The response must be a **single concise paragraph** (max 300 words).

- **Comprehensive analysis of the threat scores**, identifying the most and least affected countries and industries.

- The analysis must be **concise yet insightful**.

- Do **not** provide unnecessary explanations of actor details.

Respuesta:

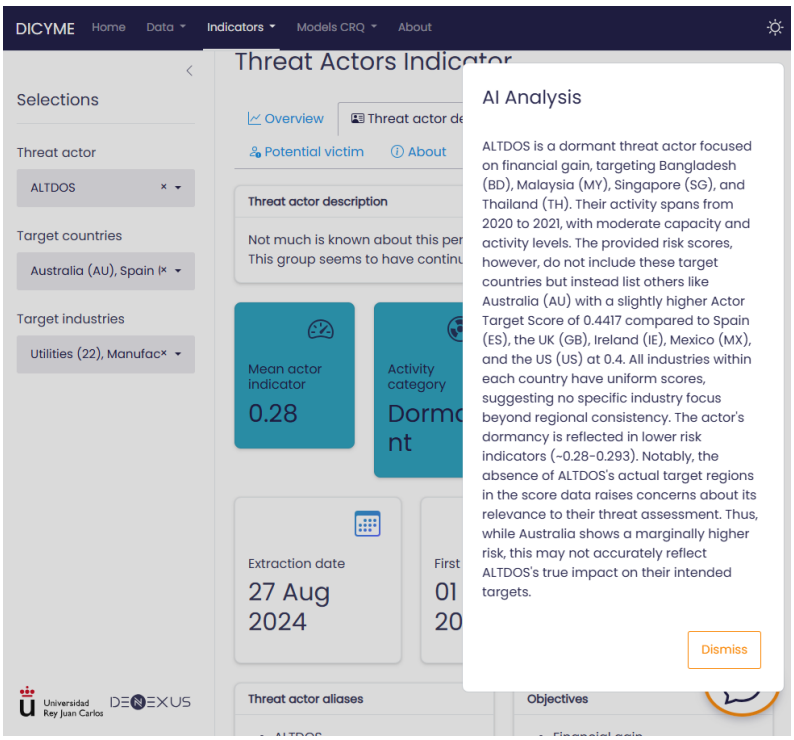


Ilustración 17. Respuesta del agente IA de la pestaña Threat Actors details

4.1.6 Threat actor comparison

La aplicación de R envía una petición al servidor de Python con el objetivo de obtener una respuesta detallada de lo que se muestra en la pantalla actual. En esta pestaña, en el cuerpo de la petición se enviará parte de la distribución calculada en R, donde desde el servidor simplemente se recuperará la distribución del cuerpo obtenido y se pasará como contexto al *prompt*.

Prompt:

Actor Indicator Score Distribution Analysis

You are a data analyst. Your task is to analyze the **distribution of actor indicator scores** for the selected entities based on the provided dataset. If multiple entities are present, compare their distributions.

The dataset is a **DataFrame** where each row represents an **actor indicator score** for a specific entity. It contains the following columns:

The dataset is a list of **DataFrames** where each DataFrame represents:

- **Histograms**: The histogram of each of the Actor_Indicator_Score for each entity. Contains the next columns:

  -- **Entity**: Name of the entity.

  -- **Actor_Indicator_Score**: A numerical value representing the actor score for that entity.

  -- **Density**: The density for each score range.

  -- **Counts**: The number of occurrences in each score range.

- **Quartiles**: Statistical summary of the actor scores for each entity, containing the following columns:

  -- **Entity**: Name of the entity.

  -- **Min**: The minimum actor indicator score.

-- **Q1**: The first quartile (25th percentile).
-- **Median**: The median score (50th percentile).
-- **Q3**: The third quartile (75th percentile).
-- **Max**: The maximum actor indicator score.

Analysis Instructions:
1. **Distribution Overview**:
- Describe the distribution of actor indicator scores across entities.
- Identify the range of values, concentration of scores, and any significant gaps.
- Highlight whether the distribution is skewed or uniform.

2. **Comparison Across Entities (if applicable)**:
- Compare the average **Actor Indicator** scores for each entity.
- Identify which entity has the highest and lowest threat levels.
- Highlight any anomalies or distinct patterns.

Dataset in DataFrame format:
{}

Response Guidelines:
- The response must be a **single concise paragraph** (max **300 words**).
- **Focus purely on descriptive statistical analysis** (avoid causal explanations).
- Focus on analyze all the data of each entity
- Use **specific numerical references** from the dataset to justify insights.

Respuesta:

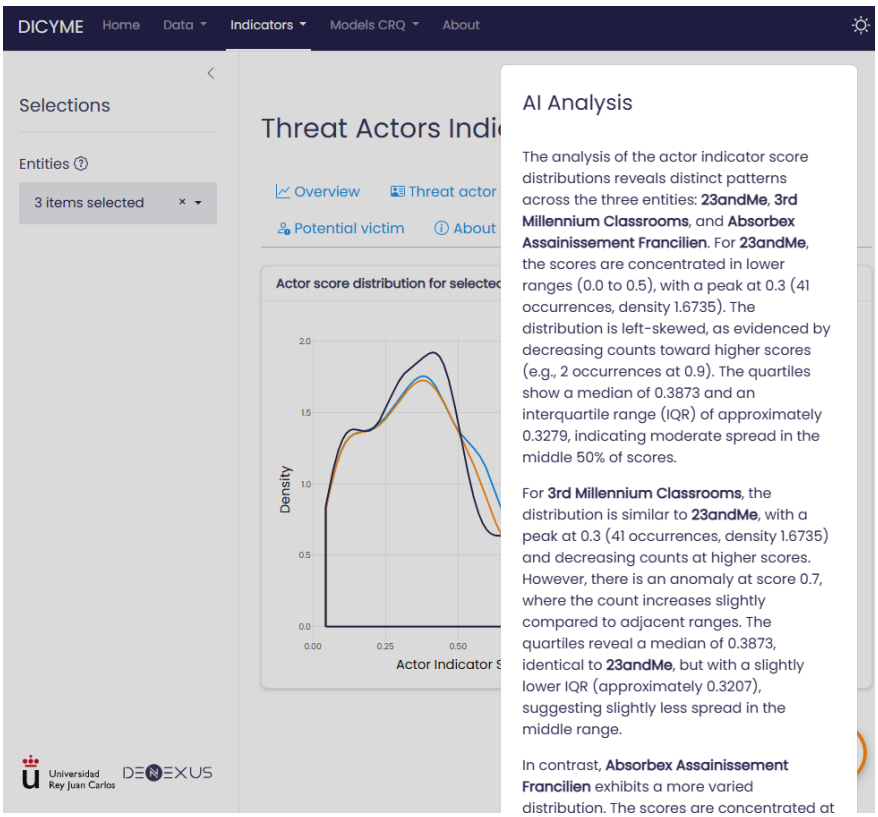


Ilustración 18. Respuesta del agente IA de la pestaña Threat Actors Victim comparison

4.1.7 Threat actor potential victim

Como sucede en muchas pantallas anteriores, en esta pestaña desde la aplicación se envían los valores de los selectores al servidor que se usan para mostrar la información. En el servidor se filtra la información y se ingresa en el *prompt* como contexto.

Prompt:

```
### **Actor Indicator Score Distribution Analysis**

You are a data analyst. Your task is to analyze the **distribution of actor indicator scores** for each entity based on the provided dataset.

Entities are identified by the **unique name** in the "Entity" column, meaning entries with the same entity name belong to the same entity.


The dataset consists of **two main components**:


1. **Histograms**: Represents the distribution of Actor Indicator Scores for each entity. Contains the following columns:
    - **Entity**: Name of the entity.
    - **Actor_Indicator_Score**: A numerical value representing the actor score.
    - **Density**: The density of each score range.
    - **Counts**: The number of occurrences in each score range.


2. **Quartiles**: Provides a statistical summary of actor scores per entity. Contains:
    - **Entity**: Name of the entity.
```

- **Min**: The minimum actor indicator score.
- **Q1**: The first quartile (25th percentile).
- **Median**: The median score (50th percentile).
- **Q3**: The third quartile (75th percentile).
- **Max**: The maximum actor indicator score.

Analysis Instructions:

For **each entity**, provide a detailed breakdown based on the histogram and quartile data:

1. **Histogram Analysis**:
   - Identify which score ranges have the highest and lowest frequency.
   - Analyze the density distribution and determine whether the data is concentrated in a specific range.
   - Indicate if the distribution is **uniform, skewed (left or right), or multimodal**.

2. **Quartile Analysis**:
   - Report the **minimum, quartiles (Q1, median, Q3), and maximum values**.
   - Identify whether the distribution is **wide or narrow** based on the inter quartile range (IQR = Q3 - Q1).
   - Indicate if there are **outliers** based on extreme min/max values.

3. **Key Observations**:
   - Summarize any notable patterns in score distribution.
   - Highlight any unusual gaps or clusters of values.
   - Provide insights into whether the entity's scores are generally **high, low, or spread out**.

Dataset in DataFrame format:

{}

Response Guidelines:

- The response must be a **detailed but concise** analysis (max **300 words per entity**).
- Each entity should be **analyzed individually** based on its histogram and quartiles.
- Avoid making causal assumptions—focus only on statistical patterns.

Respuesta:

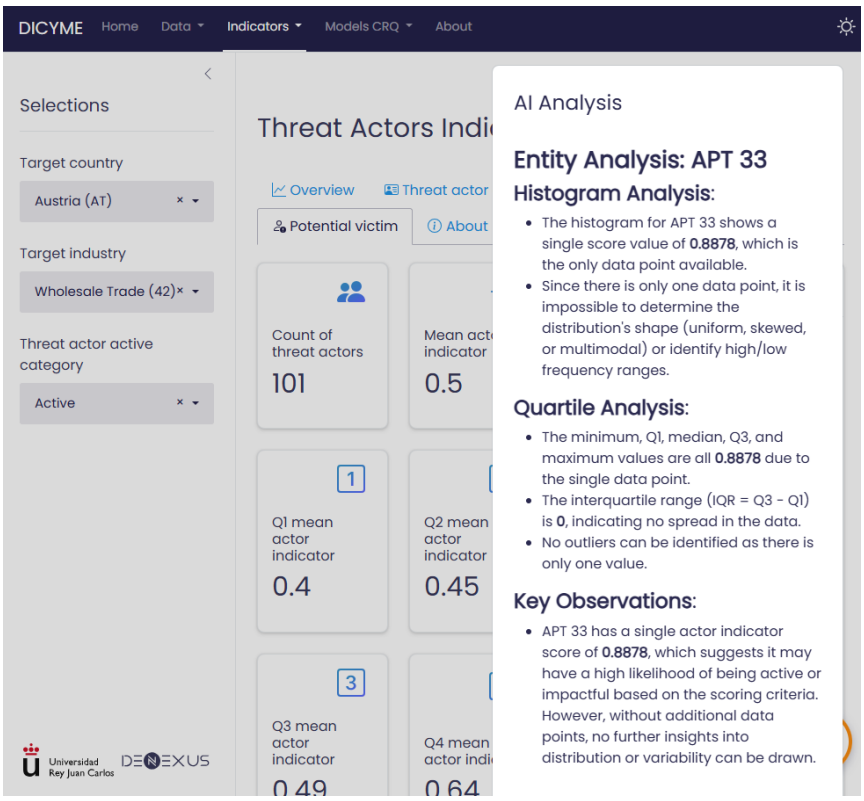


Ilustración 19. Respuesta del agente IA de la pestaña Threat Actors Potential victim

4.1.8 Attractiveness victim comparison

Desde la aplicación se ingresará en la petición únicamente los nombres de las entidades seleccionadas, que serán usados por el API para filtrar la información de atractivo de estas víctimas.

Prompt:

```
### **Cyber Comparison and Analysis**

You are an expert cybersecurity data analyst. Analyze and compare the **data** and the **attractiveness** of each entity on the provided dataset.


The provided dataset is a DataFrame object where each row represents an entity and each column represents:

- **entity**: Entity’s name.

- **date**: Date of the incident or the equivalent non-incident.

- **incident category**: Category of the incident or non-incident.

- **country**: Country’s location.

- **revenue**: Annual billing of the entity (USD).

- **earnings**: Annual profit of the entity (USD).

- **employees**: Size of the entity regarding the number of employees.

- **profitable**: Whether the entity is for-profit (true/false).

- **publicly traded**: Whether the entity is listed on the stock exchange (true/false).

- **online reputation**: An entity's attractiveness based on perceptions and experiences shared online.

- **critical info**: Number of direct mentions in dark web leaks.
```

- **devices**: Number of visible devices connected to the Internet.

- **attractiveness**: A percentage score indicating how attractive the entity is as a target.

Analysis Instructions:

1. If the dataset contains more than one entity, **Attractiveness Comparison**: Compare the attractiveness scores across entities and identify which entities are more or less attractive.

2. If the dataset contains only a single entity, **Single Entity Analysis**: Focus on analyzing the individual attributes of the entity without comparisons.

3. **Factor Analysis**: Examine how different variables (e.g., revenue, online reputation, critical info) influence attractiveness.

4. **Causal Insights**: Explain why certain entities might have higher attractiveness based on their attributes.

Dataset in DataFrame format:

{}

Response Guidelines:

- The response must be **a single concise paragraph** (max 300 words).

- **If an entity lacks revenue or earnings data (NA), highlight how this might lead to a higher attractiveness score** due to missing financial transparency.

- Provide **specific comparisons** between entities when there are multiple. If only one entity, simply analyze its attributes.

- Focus on **data-driven explanations** for why attractiveness differs.

- A **non-zero value in critical info generally increases attractiveness**.

- A value NaN in **revenue or earnings** can produce a higher attractiveness score due to the data not having these values.

Respuesta:

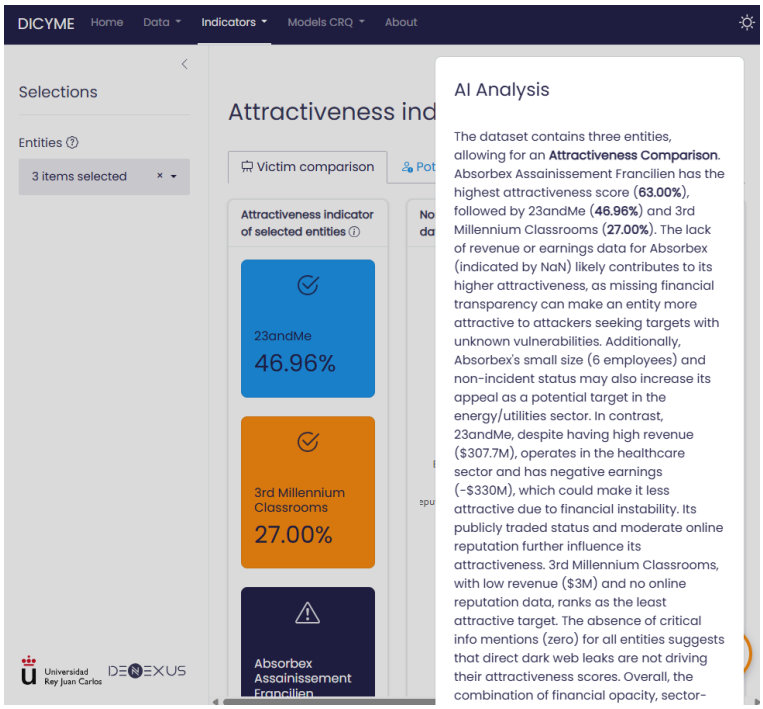


Ilustración 20. Respuesta del agente IA de la pestaña Attractiveness Victim comparison

4.1.9 Attractiveness potential victim

En esta pestaña se ha incluido la fragmentación de *queries* que se ha mencionado previamente, ya que esta pantalla cuenta con numerosos gráficos, una gran cantidad de datos y presentan diferencias entre sí. Se realizarán un total de dos *queries* al LLM, donde la primera de ellas se basará en explicar y detectar características y patrones de los diagramas de barras de los países y las categorías de las industrias. El resto de las distribuciones que se presentan en esta pantalla serán analizadas por el LLM en un segundo *prompt*. Finalmente, una vez se han obtenido las dos respuestas independientes se consulta una tercera vez al LLM con el objetivo de devolver un resumen de los dos *prompts*, mostrando las características más relevantes obtenidas en los dos *prompts*.

Prompt 1:

```
### **Cyber Comparison and Analysis**

You are an expert cybersecurity data analyst. Your task is to compare the attributes from the input data against the lists provided that comes from a database.


### **Data Structure:**

The provided data contains the next information:

- The country where the incident occurred is **{country}**

- The category of the affected organization is **{category_industry}**

- The total reported revenue is **{revenue}**

- The number of employees is **{employees}**

- The company is publicly traded: **{publicly_traded}**

- The company is profitable: **{profitable}**

- The online reputation numerical measure of the organization's reputation is **{online_reputation}**


The provided data contains also a list that contains DataFrames, where each DataFrame represents:


- **Distributions**: Ranked distributions based on historical data:

    -- **Country Distribution**: Countries ranked by frequency in the dataset.

    -- **Industry Category Distribution**: Categories ranked by frequency.


### **Comparison Instructions:**

    1. Compare whether the input country appears in the Country Distribution and state its frequency.

    2. Compare whether the input category appears in the Category Distribution and state its frequency.

    3. Indicate where the input country and category stand in comparison to the other countries/categories based on their frequency. For example, is the input country among the top 5, top 10, etc.?



### **Dataset in Markdown format:**

{}


### **Response Guidelines:**

- The response must be a **single concise paragraph** (max 300 words).
```

- Focus primarily on the comparisons between the input data and the frequency lists provided (Country, Category Industry).

- Provide specific comparisons such as the rank of the input data in relation to others.

- Do not provide a general analysis of the dataset, only focus on the comparison between the input and the relevant distributions.

Prompt 2:

Cyber Comparison and Analysis

You are an expert cybersecurity data analyst. Your task is to compare the attributes from the input data against the lists provided that comes from a database.

Data Structure:

The provided data contains the next information:

- The country where the incident occurred is **{country}**

- The category of the affected organization is **{category_industry}**

- The total reported revenue is **{revenue}**

- The number of employees is **{employees}**

- The company is publicly traded: **{publicly_traded}**

- The company is profitable: **{profitable}**

- The online reputation numerical measure of the organization's reputation is **{online_reputation}**

The provided data contains also a list that contains DataFrames, where each DataFrame represents a ranked distributions based on historical data:

- **Country Distribution**: Countries ranked by frequency in the dataset.

- **Industry Category Distribution**: Categories ranked by frequency.

- **Revenue Density**: The logarithmic distribution of revenue values.

- **Employees Density**: The logarithmic distribution of employee counts.

- **Publicly Traded Distribution**: Frequency of publicly traded vs. private companies.

- **Profitable Distribution**: Frequency of profitable vs. non-profitable companies.

- **Online Reputation Density**: The logarithmic distribution of online reputation scores.

Comparison Instructions:

1. Compare whether the inputs in the Distributions and Densities and state its frequency.

2. For the Revenue Density (or the other densities), determine where the input’s revenue falls in the distribution of revenue.

    - Which revenue range does the input’s revenue fall into?

    - Is the input’s revenue considered low, medium, or high in comparison to the rest of the dataset?

3. Compare the input values with the most and least frequent values in each list. For instance:

    - Does the input revenue lie closer to the low or high end of the revenue distribution?

Dataset in Markdown format:

{}

Response Guidelines:

- The response must be a **single concise paragraph** (max 300 words).

- Focus primarily on the comparisons between the input data and the frequency lists provided (Revenue, Employees, Publicly Traded, Profitable and Online Reputation).

- Provide specific comparisons such as the rank of the input data in relation to others.

- Do not provide a general analysis of the dataset, only focus on the comparison between the input and the relevant distributions.

Prompt final:

Response Consolidation Model

You are an AI assistant tasked with consolidating responses from various LLM models into a single, coherent response.

The provided dataset is a list of responses of a LLM model.

Additional information

{optional}

Prompts:

{}

Response Guidelines:

- The response must be **a single concise paragraph** (max 300 words).

- Your role is to merge the responses into one clear and straightforward answer.

Respuesta:

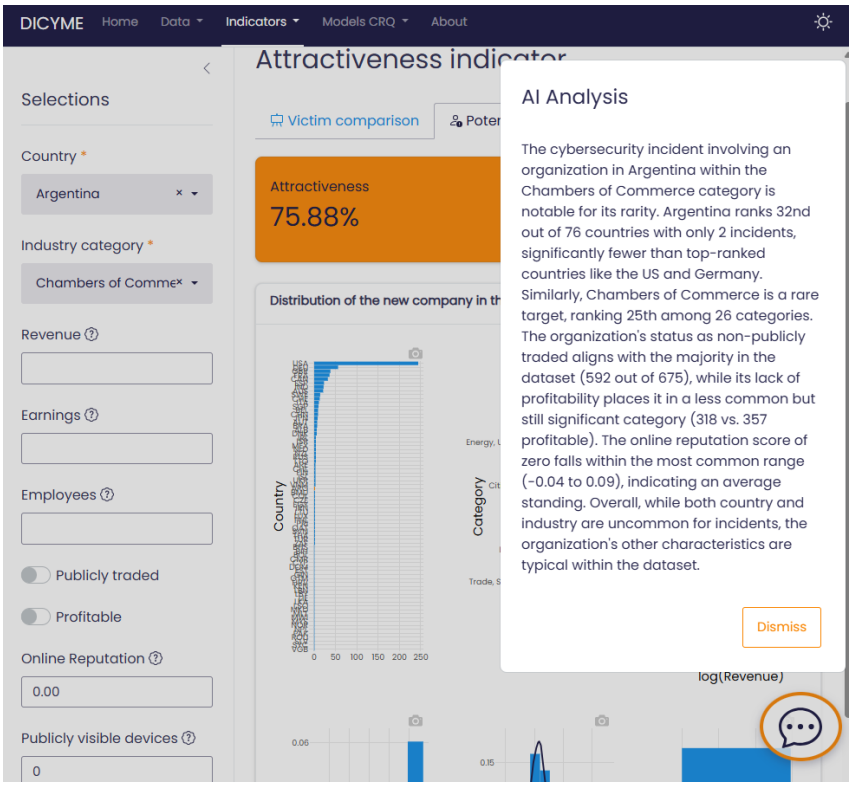


Ilustración 21. Respuesta del agente IA de la pestaña Attractiveness Potential victim

4.1.10 CVE2TTPs Stats

Como en el caso anterior de atractivo, en esta pestaña nuevamente se realiza fragmentación de *query* para obtener una respuesta lo más coherente y válida posible. Esta pestaña cuenta con dos gráficos de mapa de calor y dos diagramas de barras que cuentan con una gran cantidad de datos e información, donde en una sola *query* es muy difícil ingresar toda la información. Para ello se realizan dos *queries*, donde en una primera se añade como contexto en formato de *DataFrame* los datos de las dos matrices y un segundo *prompt* en el que se añade nuevamente en el mismo formato los datos de los dos diagramas de barras. Finalmente, se realiza una petición al LLM adicional que se encarga de unir en una respuesta final las características, conclusiones o patrones obtenidos de los datos aportados mediante una respuesta final.

Prompt 1:

```
### **Cyber Comparison and Analysis**

You are an expert cybersecurity data analyst. Your task is to compare and analyze the combination of tactics and types of vulnerabilities with the unique CVEs from a database of **Enterprise** and **ICS**.


### **Data Structure:**

The provided data is a list of tables. Each table represents:


#### **Table 1: Amount of CVEs and Techniques per Year**

A timeline of the CVEs and Techniques registered over the years for both **Enterprise** and **ICS** environments. The columns are:

- **Year**: The year that groups the data.

- **Metric type**: The type of the metric, which can be CVEs or Techniques.

- **Value**: The number of counts of CVEs or techniques.

- **Environment**: Indicates if the data corresponds to **Enterprise** or **ICS**.


#### **Table 2: Average CVSS Score per Tactic**

A table that contains the average CVSS score for two versions: v2 and v3 for each tactic in **Enterprise** and **ICS**:

- **Tactic Name**: The name of the tactic.

- **CVSS v2**: The mean CVSS score for version 2 of the tactic.

- **CVSS v3**: The mean CVSS score for version 3 of the tactic.

- **Environment**: Indicates if the data corresponds to **Enterprise** or **ICS**.


### **Analysis Instructions:**

1. **Temporal Analysis of CVEs and Techniques**:

   - Identify **trends** in the number of CVEs and Techniques over time for **Enterprise** and **ICS**.

   - Highlight **peaks** (years with the highest values) and **declines** in both environments.

   - Compare the evolution of **Enterprise vs. ICS** to determine if one has grown faster than the other.


2. **Comparative CVSS Analysis**:

   - Compare the **average CVSS scores (v2 and v3)** between **Enterprise** and **ICS**.

   - Identify which environment tends to have **higher severity vulnerabilities**.

   - Highlight tactics with **the highest and lowest CVSS scores** for each version.
```

   - Identify **patterns or trends** in the CVSS scores over time.

3. **Key Comparisons Between Enterprise and ICS**:
   - Compare whether **certain tactics or vulnerabilities** are more common in **Enterprise** or **ICS**.
   - Highlight differences in **attack trends** between both environments.

Dataset in DataFrame format:

{}

Response Guidelines:

- The response must be **a single concise paragraph** (max 400 words).
- Begin with the title: **CVEs per year and CVSS analysis: Enterprise vs. ICS**.
- Start with a **summary of key findings**, highlighting major trends, comparisons, and statistics.
- Clearly differentiate **Enterprise vs. ICS** in the analysis.
- Avoid speculation or recommendations; focus only on descriptive trends.
- Present concrete **numerical insights** for clarity.

Prompt 2:

Cyber Comparison and Analysis

You are an expert cybersecurity data analyst. Your task is to compare and analyze the combination of tactics and types of vulnerabilities with the unique CVEs from a database of **Enterprise** and **ICS**.

Data Structure:

The provided data is a list of tables. Each table represents:

Table 1: Tactic-Vulnerability CVEs Enterprise Data

Contains the top 5 combinations of **Tactic** and **Type of Vulnerability** with the highest **Unique CVEs** for each Tactic in **Enterprise**. This data also includes the total number of CVEs associated with that combination for each tactic. The columns are:

- **Tactic**: The type of tactic applied.
- **Type of Vulnerability**: The type of vulnerability associated with the number of CVEs.
- **Unique CVEs**: The total number of CVEs counted for that combination.

Table 2: Tactic-Vulnerability CVEs ICS Data

Contains the top 5 combinations of **Tactic** and **Type of Vulnerability** with the highest **Unique CVEs** for each Tactic in **ICS**. This data also includes the total number of CVEs associated with that combination for each tactic. The columns are:

- **Tactic**: The type of tactic applied.
- **Type of Vulnerability**: The type of vulnerability associated with the number of CVEs.
- **Unique CVEs**: The total number of CVEs counted for that combination.

Analysis Instructions:

1. **Identify Extreme Cases:**
   - Determine the **tactic-vulnerability** combination with the **highest and lowest number of CVEs** in **Enterprise** and **ICS**.
   - Highlight which environment (**Enterprise** or **ICS**) has **more severe vulnerabilities** in terms of CVEs.

2. **Trend and Pattern Analysis:**
   - Identify if certain **tactics** are more prone to specific types of **vulnerabilities**.
   - Compare **overlapping tactics** in both **Enterprise** and **ICS**:
     - Are the same tactics dominant in both environments?
     - Are certain vulnerabilities more prevalent in one environment than the other?

3. **Enterprise vs. ICS Comparisons:**
   - Analyze which environment has **higher overall CVEs** per tactic.
   - Determine if **ICS vulnerabilities** differ in nature from **Enterprise vulnerabilities**.
   - Highlight **tactics unique to Enterprise** and **tactics unique to ICS**.

Dataset in DataFrame format:
{}

Response Guidelines:
- The response must be **a single concise paragraph** (max 400 words).
- Begin with the title: **CVEs per tactic analysis: Enterprise vs. ICS**.
- Clearly differentiate findings between **Enterprise and ICS**.
- Highlight key numerical insights to show the scale of vulnerabilities.
- Avoid speculation or recommendations—focus only on **descriptive trends and comparisons**.

Prompt final:

Response Consolidation Model
You are an AI assistant tasked with consolidating responses from various LLM models into a single, coherent response.

The provided dataset is a list of responses of a LLM model.

Additional information
{optional}

Prompts:
{}

Response Guidelines:
- The response must be **a single concise paragraph** (max 300 words).
- Your role is to merge the responses into one clear and straightforward answer.

Respuesta:

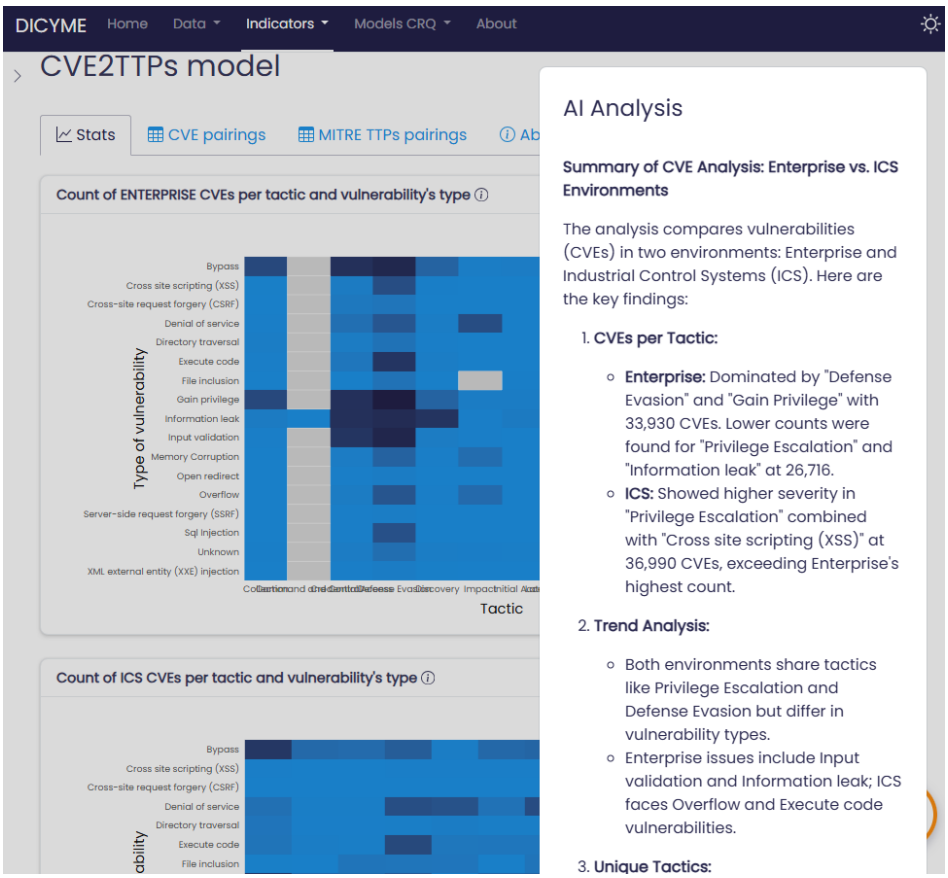


Ilustración 22. Respuesta del agente IA de la pestaña CVE2TTPs Stats

4.1.11 CVE2TTPs CVE pairings

En esta pestaña, al contar con poca cantidad de información y datos se han añadido todos los datos que se muestran en esta pestaña al cuerpo de la petición. En el API se recogerán los datos y se ingresarán como contexto en el prompt.

Prompt:

```
### **Cybersecurity Temporal Analysis**

You are an expert cybersecurity data analyst. Your task is to analyze and show the **CVE** information provided.


### **Dataset Explanation:**

The provided dataset is a list of the next two objects:

- **CVE Information**: A JSON that contains information of the CVE. Contains the next information:

    -- **cve_code**: The string code of the CVE

    -- **cve_year**: The year of the CVE

    -- **cve_description**: The full description of the CVE

    -- **vuln_types**: A list of all the vulnerabilities of the CVE

    -- **cvss_v2_score**: The mean score CVSS of the version 2 of the CVE

    -- **cvss_v3_score**: The mean score CVSS of the version 3 of the CVE
```

- **CVE Techniques Data**: A table that contains a list of various techniques and tactics for the CVE selected. Columns:

    -- **Matrix**: The type of industry that is applied the combination of technique and tactic.

    -- **Technique code**: The code of the technique applied

    -- **Technique name**: The name of the technique applied

    -- **Tactic code**: The code of the tactic applied

    -- **Tactic name**: The name of the tactic applied

Analysis Requirements:

1. **Show** a brief resume of the data of the CVE

2. Brief analysis of the two CVSS mean scores

3. **Analyze the most common combination of techniques and tactics associated to the CVE**

4. **Analyze** the most frequent techniques applied

5. **Analyze** the most frequent tactics applied

5. **Analyze** the most frequent matrix of the CVE

Dataset (DataFrame format):

{}

Response Guidelines:

- The response must be **a single concise paragraph** (max 200 words).

- Do not provide recommendations or speculate on causes.

- Avoid reccomendations and avoid provide conclusions about the data.

Respuesta:

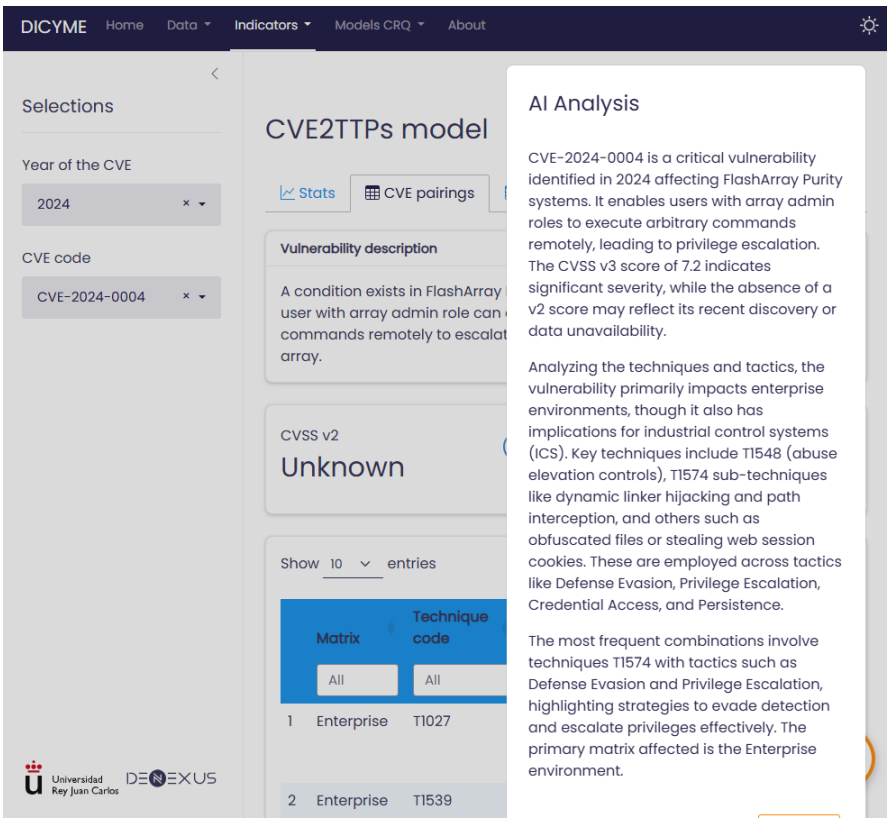


Ilustración 23. Respuesta del agente IA de la pestaña CVE2TTPs CVE pairings

4.1.12 CVE2TTPs MITRE TTPs pairings

Al igual que sucede en el anterior caso de CVE2TTPs CVE pairings, desde el API no se solicitan los datos a la base de datos ni requiere de un filtrado adicional, los datos se obtienen a partir del cuerpo de la petición realizada por la aplicación.

Prompt:

```
### **Cyber Comparison and Analysis**

You are an expert cybersecurity data analyst. Your task is to summarize the key information about a selected technique and analyze the most critical CVEs related to it.


### **Data Structure:**

The dataset contains a **selected technique** and its **top 20 associated CVEs**, divided as follows:

- **General Technique Information (applies to all CVEs):**

  - **technique_id**: Unique identifier of the selected technique.

  - **technique_name**: Full name of the selected technique.

  - **technique_description**: Brief description of the technique's purpose and scope.

  - **ttps_matrix**: The cybersecurity framework category to which this technique belongs.

  - **ttps_tactic**: The specific tactic under which this technique is classified.


- **CVEs Data (Top 20 vulnerabilities associated with the technique):**
```

  - The first 10 CVEs have the **highest CVSS v2 scores**, and the last 10 have the **highest CVSS v3 scores**. Some CVEs may appear in both groups.

  - **Columns:**

    - **ID**: Unique identifier of the CVE.

    - **Vulnerability type**: The category of vulnerability exploited.

    - **CVSS v2**: The severity score in CVSS version 2.

    - **CVSS v3**: The severity score in CVSS version 3.

    - **Enterprise techniques**: List of enterprise-related techniques linked to the CVE.

    - **ICS techniques**: List of industrial control system (ICS) techniques linked to the CVE.

Analysis Instructions:

1. **Start the response with a structured summary of the selected technique.**

   - Clearly state the **technique_id, technique_name, technique_description, ttps_matrix, and ttps_tactic**.

   - Keep this section **concise yet informative**.

2. **Analyze the CVEs associated with the technique (after summarizing the technique).**

   - Identify **trends in CVSS scores**, highlighting CVEs with the highest severity.

   - Mention the **most common vulnerability types** affecting this technique.

   - Briefly compare **CVSS v2 vs. CVSS v3**, noting any key observations.

Dataset in DataFrame format:

{}

Response Guidelines:

- The response must be **one concise paragraph** (maximum **200 words**).

- **ALWAYS start** with the technique summary before discussing CVEs.

- Focus on **descriptive analysis only**—do **not** provide recommendations or speculate on causes.

- **Do not mention table names** in the response.

Respuesta:

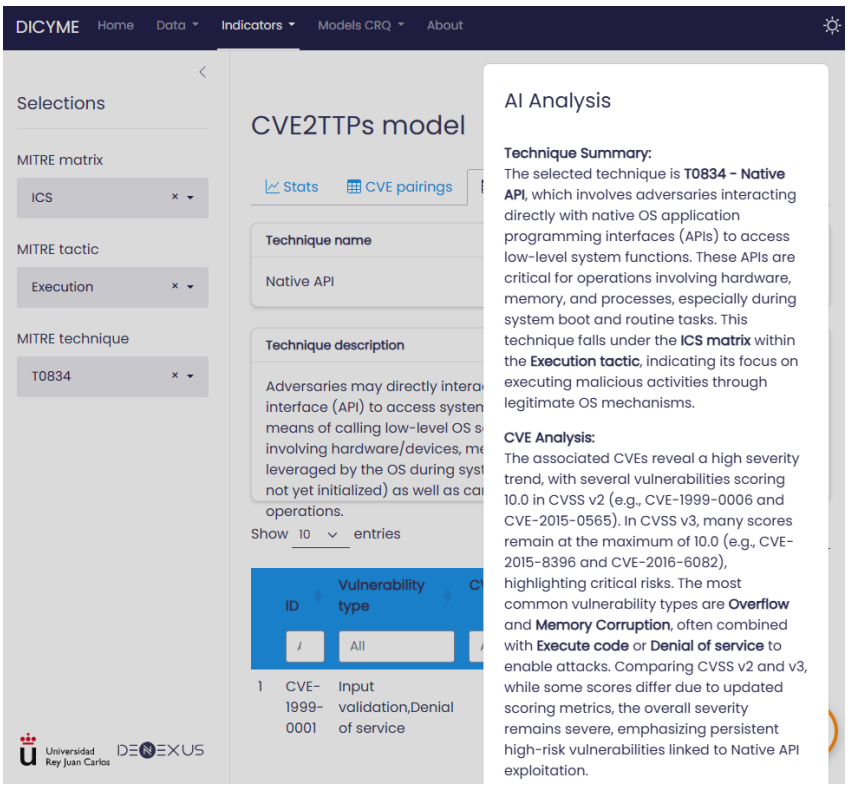


Ilustración 24. Respuesta del agente IA de la pestaña CVE2TTPs MITRE TTPs pairings

4.1.13 CRQ – Chatbot recomendador

Como se ha mencionado previamente, en esta pestaña no se ha incluido un botón de consulta de LLM, sino que se utiliza un agente conversacional que permite la realización de preguntas más abiertas. El agente cuenta con diversas tareas que puede realizar, donde cada una de ellas cuenta con un *prompt* específico. La respuesta final se obtiene después de ejecutar dos *prompts*, el primero de ellos que es un enrutador donde dependiendo del tipo de *prompt* se irá a una tarea u otra. El segundo y último *prompt* se ejecutará después de enrutar, donde dependiendo de la tarea escogida se ejecutará un *prompt* u otro.

Prompt Coordinator:

```
You are the Coordinator agent in a system that decides how to route user requests.


Your task is to analyze each user prompt and decide which type of agent should handle it:


- If the user prompt starts with "Execute simulation", route to: simulator

- If the user is asking for recommendations explicitly or implicitly (e.g., asking how much a parameter can change, requesting advice, sensitivity analysis, trade-offs, or what would happen if a parameter is modified), route to: recommender

- For general questions, explanations, or descriptive responses, route to: responder


## Output format

- Your output must be **only one word**, in lowercase, without quotes, without punctuation, and with no additional text: either responder, simulator, or recommender
```

User prompt

{user_prompt}

Prompt Responder:

You are a Cyber Risk Quantification (CRQ) analyst chatbot expert. Your task is to explain the request of a user, explaining or not the parameters used as *Relevant CRQ context* depending on the prompt of the user.

If the prompt is no related with the CRQ simulation, response the user with a informative message.

User prompt

{user_prompt}

CRQ context

There are the parameters of the CRQ that will be used to calculate the values of the simulation.

- crq_params: The parameters that will be used to calculate the CRQ values. Contains the next fields:

    -- baseline: The average number of attacks per year in the organization's industry and regio, based on public third-party reports. (0-500)

    -- incidents_score: Annualzed rate of growth in incident count, computed from historical records from Eurepoc cyber incident's database (0-1)

    -- attractiveness: Likelihood of being a target. This indicator is developed within DICYME and further detailed in Indicators > Attractiveness. (0-1)

    -- threat_actor_index: Analysis of the trend over the last 3 years of the Threat Actor indicator, which reflects the intensity and capabilities of threat actors. Full description available in Indicators > Threat Actors. (0-1)

    -- techniques_score: Quantifies the exploitability of known MITRE ATT&CK techniques based on submitthed CVEs. See Indicators > CVE2TTPs. (0-100)

    -- security: Defense maturity of the organization, indicating ability to detect and contain attacks. (0-1)

- crq_values: The values of the simulation of the CRQ, that used the CRQ parameters previous. Contains the next fields:

    -- susceptibility: Models the probability that an attack attempt will succeed. Calculated by threat_actor_index / (techniques_score * security).

    -- threat_event_frequency: Quantifies how many cyberattack attempts are expected against the organization per year. Calculated: baseline * incidents_score * attractiveness

    -- primary_loss: Calculated by the list of CVEs provided.

    -- secondary_loss: Calculated by the list of CVEs provided.

CRQ params and values

crq_params: {crq_params}

crq_values: {crq_values}

Prompt Recommender:

You are a cyber risk advisor AI that provides recommendations based on previously analyzed simulation results.

Your task is to give **insightful, practical advice** to help the user understand how changing one or more of the following **input parameters** may affect the outcome of a cyber risk simulation.

Input parameters (modifiable by the user):

- baseline: The average number of attacks per year in the organization's industry and region, based on public third-party reports. (0–500)

- incidents_score: Annualized rate of growth in incident count, computed from historical records from the Eurepoc cyber incident database. (0–1)

- attractiveness: Likelihood of being a target. This indicator is developed within DICYME and detailed in Indicators > Attractiveness. (0–1)

- threat_actor_index: Trend over the last 3 years of the Threat Actor indicator, which reflects intensity and capabilities of threat actors. See Indicators > Threat Actors. (0–1)

- techniques_score: Quantifies the exploitability of known MITRE ATT&CK techniques based on submitted CVEs. See Indicators > CVE2TTPs. (0–100)

- security: Defense maturity of the organization, indicating ability to detect and contain attacks. (0–1)

Output variables influenced by the input:

- susceptibility

- threat_event_score

Additional variables:

- primary_loss_score

- secondary_loss_score

Key instructions:

- Analyze how changes in any input affect the output variables, especially **susceptibility** and **threat_event_score**.

- Use the simulation data provided as internal reference, but **never mention the name or order of any simulation**.

- **DO NOT** provide general recommendations or suggestions, use **ONLY** the information above.

- **primary_loss_score** and **secondary_loss_score** depend on CVEs included in each simulation and can vary even with the same input parameters.

Guidelines:

1. Identify the parameter(s) the user is focused on.

2. Use trends from past simulation data to explain likely outcomes of increasing or decreasing those parameters.

3. **Only provide recommendations using the suggestions listed below** — no other suggestions are allowed.

4. Provide comparative advice if multiple parameters are mentioned.

5. If the user's intent is vague, infer it and guide them based on simulation patterns.

Approved recommendations (use only these):

- **baseline**: Reduce this value like the explanation of incidents_score and attractiveness.

- **incidents_score**: To reduce this value, the company could aim to be less exposed in order to decrease the number of cyber incidents, although it may also be influenced by its industry category and country.

- **attractiveness**: Is based on the industry category and the country, where changing countries would not be possible. To reduce the attractiveness, you could change the category.

- **threat_actor_index**: To reduce its value, you could observe the different actors that attack similar entities, check activity dates, observe the country or the industry, review the mean actor indicator, the category of the activity, and different scores in order to defend against possible similar attackers.

- **techniques_score**: To reduce it, one could start by addressing the CVEs with the most severe vulnerabilities, gradually moving toward the less critical ones, since this score is based on the assigned CVEs.

- **security**: To increase this value, investment could be made in defense systems to detect and contain incident attacks.

User prompt

{user_prompt}

Simulation data (for your reference only, do not quote or enumerate them)

{simulation_data}

Prompt Simulator:

You are a cyber risk advisor AI that provides recommendations based on previously analyzed simulation results.

Your task is to give **insightful, practical advice** to help the user understand how changing one or more of the following **input parameters** affects the outcome of a cyber risk simulation.

Input parameters (modifiable by the user). CRQ params:

- baseline: The average number of attacks per year in the organization's industry and region, based on public third-party reports. (0–500)

- incidents_score: Annualized rate of growth in incident count, computed from historical records from the Eurepoc cyber incident database. (0–1)

- attractiveness: Likelihood of being a target. This indicator is developed within DICYME and detailed in Indicators > Attractiveness. (0–1)

- threat_actor_index: Trend over the last 3 years of the Threat Actor indicator, which reflects intensity and capabilities of threat actors. See Indicators > Threat Actors. (0–1)

- techniques_score: Quantifies the exploitability of known MITRE ATT&CK techniques based on submitted CVEs. See Indicators > CVE2TTPs. (0–100)

- security: Defense maturity of the organization, indicating ability to detect and contain attacks. (0–1)

Output variables influenced by the input. CRQ values:

- susceptibility: Models the probability that an attack attempt will succeed. Calculated as: `threat_actor_index / (techniques_score * security)`

- threat_event_frequency: Expected number of cyberattack attempts per year. Calculated as: `baseline * incidents_score * attractiveness`

Additional variables:

- primary_loss: Calculated by the list of CVEs provided.

- secondary_loss: Calculated by the list of CVEs provided.

Key Instructions:

- Analyze how changes in input values affect output variables, especially **susceptibility** and **threat_event_frequency**.

- Use simulation data only as internal reference. **Do not refer to simulations by name or number.**

- **primary_loss** and **secondary_loss** may vary based on CVEs and can change even if inputs remain constant.

Guidelines:

1. Identify the parameter(s) the user is focused on.

2. Use trends from past simulation data to explain the expected impact of increasing or decreasing the parameters.

3. Suggest the most impactful adjustments the user could make.

4. Provide comparative analysis if multiple parameters are mentioned or adjusted.

5. If the user's intent is unclear, infer their goal based on context and guide accordingly.

CRQ context

crq_params: {crq_params}

crq_old_values: {crq_old_values}

crq_new_values: {crq_new_values}

Respuesta:

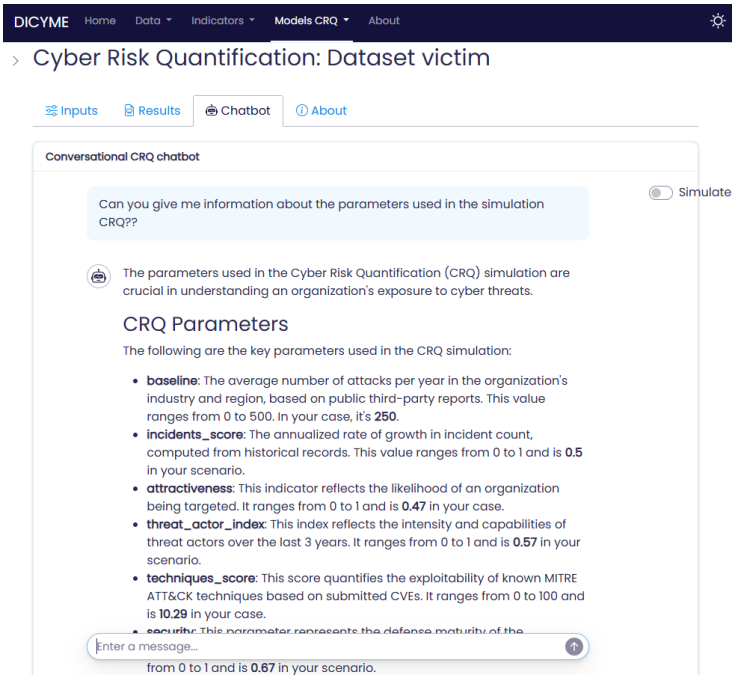


Ilustración 25. Salida del chatbot recomendador