

第 7 章 CHAPTER 7 SaaS Copilot 設計的 UX 最佳實踐 UX Best Practices for SaaS Copilot Design

在本章中，我們將使用 Microsoft Security Copilot 作為主要範例，回顧對於嘗試為其產品設計功能齊全且有用的 Copilot 的任何人至關重要的 UX 設計最佳做法。

In this chapter, we will review the UX design best practices critical for anyone attempting to design a functional and helpful Copilot for their product, using Microsoft Security Copilot as a primary example.

便條

NOTE

本章中所有螢幕擷取畫面的影像來源（除非另有說明）是 Microsoft Mechanics 的「Microsoft 安全性副駕駛的工作原理」（1）。我強烈建議您觀看整個視頻。

The image source for all screenshots in this chapter (unless indicated otherwise) is “ How Microsoft Security Copilot Works ” by Microsoft Mechanics (1). I strongly recommend you watch the entire video.

Microsoft Security Copilot（MSC）具有許多 SaaS 副駕駛通用的功能，以及一些我認為對成功至關重要的新功能。以下是一些關鍵要點和最佳實踐。

Microsoft Security Copilot (MSC) has many features common to SaaS copilots, plus a few new features that I consider critical for success. Here are some key takeaways and best practices.

任務越重要，需要的房地產就越多

The More Important the Task, the More Real Estate Is Required

您需要專用的整頁 Copilot 嗎？這取決於您希望 Copilot 幫助客戶完成的任務範圍。擁有單獨的經驗並不是成功的必要條件。接下來，我們將研究 Copilot 的三種樣式：側面板、大疊加層和專用頁面。每一種都有略有不同的用途。

Do you need a dedicated full-page Copilot? It depends on the scope of the task you want the Copilot to help your customers with. Having a separate experience is not a requirement for success.

Next, we ' ll examine three styles of Copilot: a side panel, a large overlay, and a dedicated page. Each one serves a slightly different purpose.

側邊面板

Side Panel

圖 7.1 顯示 MSC 如何與另一個 Microsoft 安全性產品整合。MSC 是作為附加元件實現的，透過特定頁面內的按鈕啟動。因此，會呼叫 MSC 來協助使用者處理此特定頁面。我們研究中最一致的回饋之一是，實現此類 Copilot 的最佳方法是將頁面的其餘部分移至左側（例如，不要使用側面板作為模糊頁面內容的彈出視窗）。這是因為這種類型的 Copilot 旨在直接與頁面上的資訊互動。如果您使用 Copilot 遮蓋關鍵頁面內容，則查找互動會很尷尬。側邊面板 Copilot 的另一個關鍵建議是保留在本機頁面上下文中，不要嘗試做太多或嘗試處理全域請求。您隨時可以為此目的彈出整頁專用的全域 Copilot 體驗。請注意此 Copilot 實例中的所有當地語系化頁面層級資訊。

Figure 7.1 shows how MSC is integrated with another Microsoft security product. MSC is implemented as an add-on, activated with a button inside a specific page. Thus, MSC is invoked to help users with this particular page. One of the most consistent pieces of feedback from our research is that the best way to implement this type of a Copilot is by moving the rest of the page over to the left (e.g., not using the side panel as a pop-up that obscures the page content). This is because this type of Copilot is meant to directly interact with the information on the page. If you obscure critical page content with the Copilot, the lookup interaction will be awkward. Another key recommendation for a side panel Copilot is to remain within the local page context and not to try to do too much or attempt to handle global requests. You can always pop out a full-page dedicated global Copilot experience for that purpose. Note all the localized page-level information in this instance of the Copilot.

Microsoft Entra

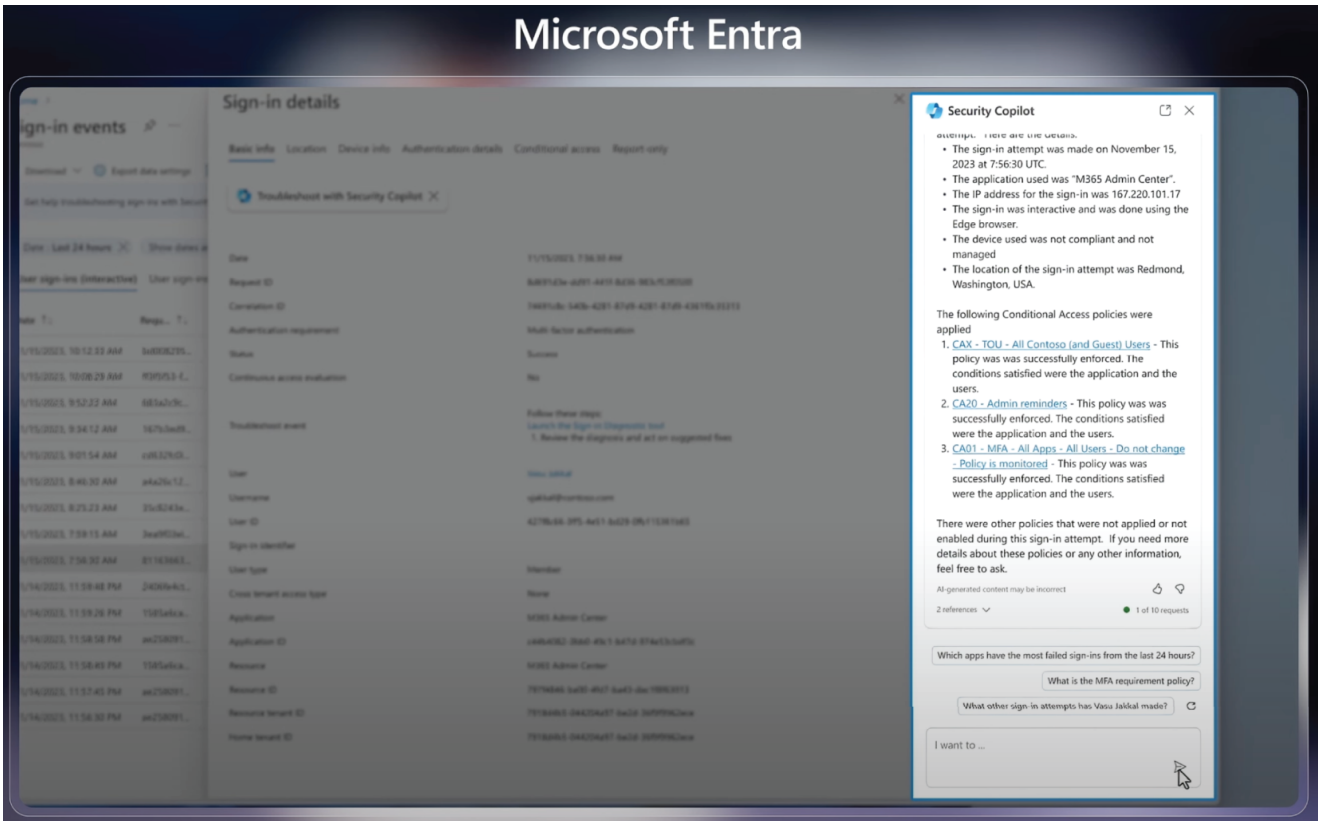


圖 7.1 實作為側邊面板的 Microsoft Security Copilot

Figure 7.1 Microsoft Security Copilot implemented as a side panel

大疊加層

Large Overlay

在圖 7.2 中，Copilot 面板相當大。它的大小適當，可透過自訂儀表板協助回答更一般的問題，該儀表板提供大量背景和支援資訊，以協助執行複雜的分析並處理更深入的問題。此範例來自整合到 QuickSight 中的 Amazon Q Copilot。根據我們的研究，大型覆蓋實作幾乎總是最不理想的選項，因為較大的面板必然會遮蓋父頁面上的大部分資訊。這使得與父頁面的任何互動都變得非常尷尬，在許多情況下，會導致 Copilot 面板的多次昂貴且冗長的內容重新載入，因為使用者被迫關閉面板才能查看父頁面中模糊的資訊。如果您必須對更廣泛的主題進行深入分析以及大量工具和建議，則幾乎普遍首選整頁專用的 Copilot 體驗。

In Figure 7.2, the Copilot panel is quite large. It is sized appropriately to help answer more general questions with a custom dashboard that provides a great deal of background and supporting information to help perform complex analysis and handle questions of greater depth. This example comes from Amazon Q Copilot integrated into QuickSight. Based on our research, a large overlay

implementation is almost always the least desirable option because a larger panel, by necessity, obscures much of the information on the parent page. This makes any interaction with the parent page extremely awkward, in many cases leading to multiple expensive and lengthy content reloads of the Copilot panel, because the users are forced to close the panel to see the obscured information in the parent page. In cases where you must show in-depth analysis on a broader topic and lots of tools and suggestions, a full-page dedicated Copilot experience is almost universally preferred.

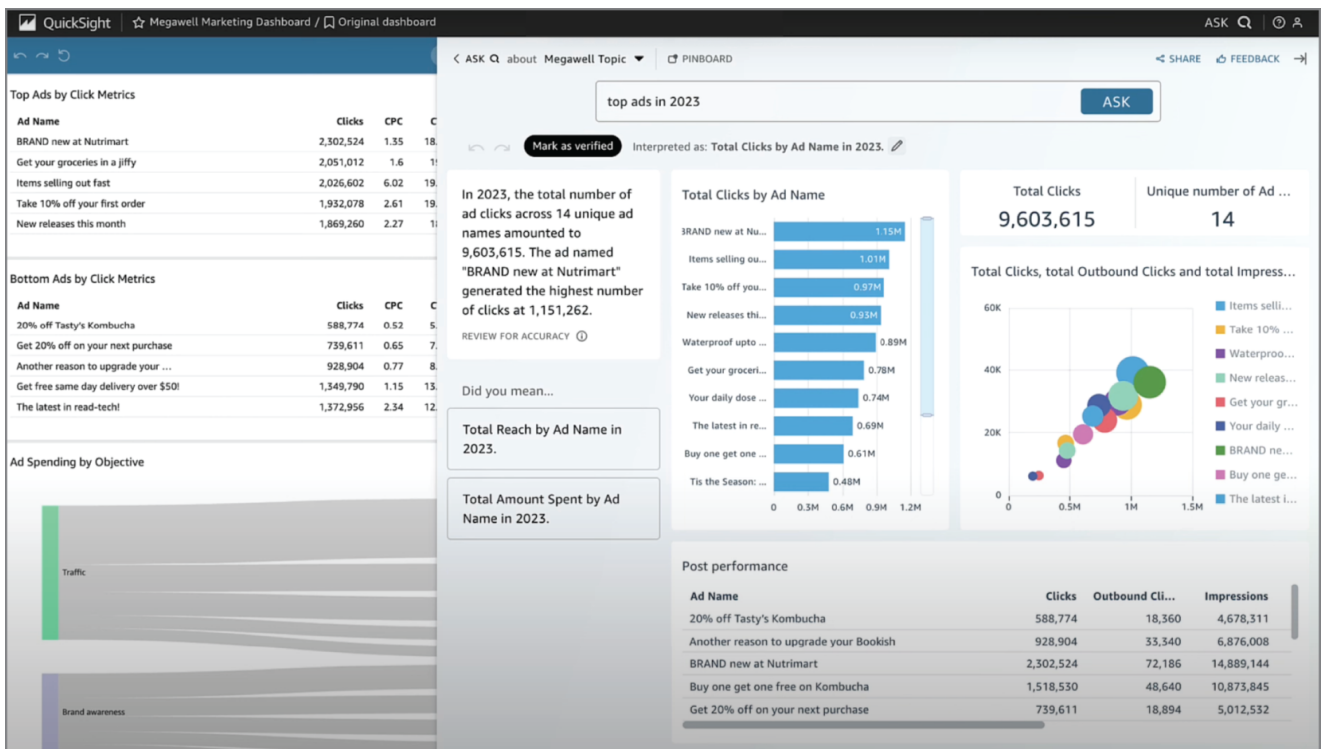


圖 7.2 Amazon Q Copilot 實作為大型覆蓋層

Figure 7.2 Amazon Q Copilot implemented as a large overlay

來源：YouTube 上的 Amazon QuickSight (2)

Source: Amazon QuickSight on YouTube (2)

整頁

Full Page

這個精彩的例子也來自 MSC，但在這個例子中，Copilot 以自訂整頁 UI 呈現，具有所有“花里胡哨”，包括完整的資料整合。它是最靈活但重量級的 Copilot UI 設計模式。就像 Sumo Copilot（在上一章中所述）一樣，此版本的 MSC 可以處理任何層級

的任務，包括提供最深入的分析層級，以及隨之而來的所有控制項和建議。整頁 Copilot 是 AI 優先的替代體驗平台，用於與您的系統互動。當然，需要注意的是，其他地方可用的產品的許多功能必須整合到這種體驗中，這在構建和維護方面既昂貴又耗時。與側邊欄 Copilot 相比，此整頁版本（見圖 7.3）需要與任何其他產品功能類似的專用產品導航。

This brilliant example is also from MSC, but in this instance, the Copilot is rendered in a custom full-page UI, with all the “bells and whistles,” including full data integration. It is the most flexible yet heavyweight Copilot UI design pattern. Just as in the case of the Sumo Copilot (described in the previous chapter), this version of MSC can handle any level of task, including providing the deepest level of analysis with all of the controls and suggestions that go with that. A full-page Copilot is an AI-first alternative experience platform for interacting with your system. The caveat is, of course, that many of the functions of the product available elsewhere will have to be integrated into this experience, which is expensive and time consuming to build and maintain. In contrast to the side-bar Copilots, this full-page version (see Figure 7.3) requires dedicated product navigation similar to any other product feature.

雖然設計師有很多選擇，但任務越嚴肅，Copilot 需要的螢幕空間就越多。最終，您的 Copilot 所需區域的大小和複雜性將取決於您希望它為客戶做什麼。

While designers have many options, the more serious the task, the more screen real estate Copilot will require. Ultimately, the size and complexity of the area you will need for your Copilot will depend on what you want it to do for your customers.

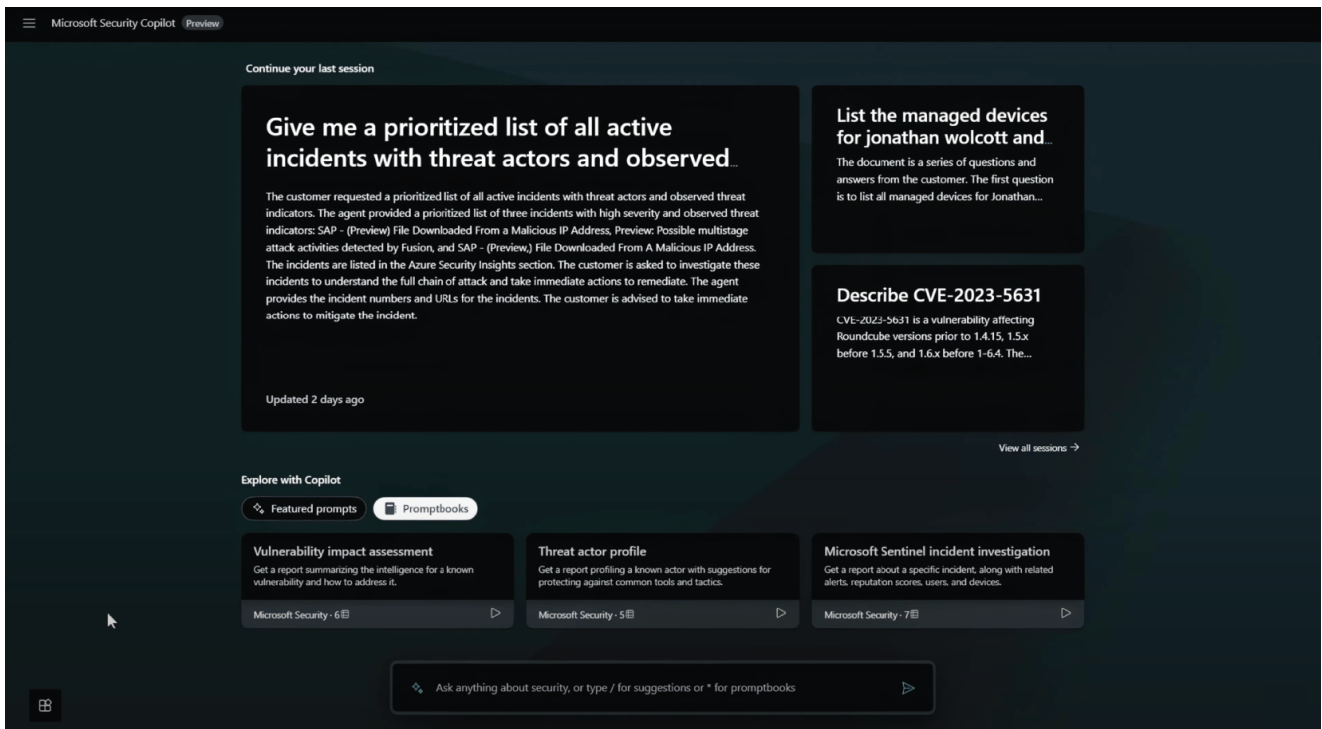


圖 7.3 整頁實作的 Microsoft Security Copilot

Figure 7.3 Microsoft Security Copilot implemented in a full page

SaaS Copilot 是有狀態的

SaaS Copilot Is Stateful

Microsoft Bing Copilot 沒有長期記憶，所以它有點像鴝鳥——每次見到你，都是在遇到一個新朋友！相較之下，像 MSC 這樣的 SaaS Copilot 強調狀態很重要。作為使用者，您可以與 MSC 進行多個重疊的對話，隨著 Copilot 不斷為您的會話添加更多上下文和資訊，從上次中斷的地方繼續。維護狀態是 SaaS Copilot 的基本功能。如果您希望使用者能夠進行更深入的多階段對話和深入分析，他們需要能夠根據需要進出工作階段並執行其他工作。

The Microsoft Bing Copilot has no long-term memory, so it is a bit like an ostrich—every time it meets you, it ' s meeting a new person! In contrast, a SaaS Copilot like MSC makes it a big point to be stateful. As a user, you can have multiple overlapping conversations with MSC, picking up where you left off as the Copilot continuously adds more context and information to your session. Maintaining state is an essential feature of a SaaS Copilot. If you want your users to be able to have deeper, multistage conversations and in-depth analysis, they need to be able to dip in and out of their session and perform

other tasks, as needed.

專門微調的 ChatGPT 模型

Specialized Fine-Tuned ChatGPT Model

MSC 的另一個重要功能是根據最新數據訓練的 AI 模型。這令人印象深刻。與股票 ChatGPT 相比，MSC 的表現要好得多。在不久的將來，微調、檢索增強生成（RAG）和其他在自訂內容上「訓練」LLM 的方法對於 SaaS Copilots 和 AI 代理來說尤其重要。（有關 RAG 的簡單實際範例，請參閱第 14 章中的側邊欄「為 AI 災難做好準備：作為 AI 提示的速記 UX 設計符號」。有關 AI 代理用戶體驗的更多信息，請參閱第 15 章。

Another essential feature of the MSC is the AI model trained on up-to-date data. It is impressive. Compared with the stock ChatGPT, MSC performs considerably better. Fine-tuning, retrieval-augmented generation (RAG), and other methods of “training” LLMs on custom content will be especially critical for SaaS Copilots and AI Agents in the near future. (For a simple practical example of RAG, see the sidebar “Getting Ready for AI-pocalypse: Shorthand UX Design Notation as AI Prompt” in Chapter 14. See Chapter 15 for more on UX of AI Agents.)

圖 7.4 顯示了根據最新公司資料訓練的自訂 MSC 模型與庫存 ChatGPT 實例之間的回應等級差異。

Figure 7.4 shows the difference in response level between a custom MSC model trained on the latest company data and a stock ChatGPT instance.

MSC 模型可以存取最新資訊（股票 ChatGPT 沒有）。MSC 還可以存取包含 IP 位址的專用資料庫，這些資料庫進一步豐富了補充資訊，例如 IP 位址是否屬於惡意組織。

The MSC model has access to up-to-date information (that the stock ChatGPT does not have). MSC also has access to specialized databases containing IP addresses further enriched with supplemental information, such as whether the IP address belongs to a malicious organization.

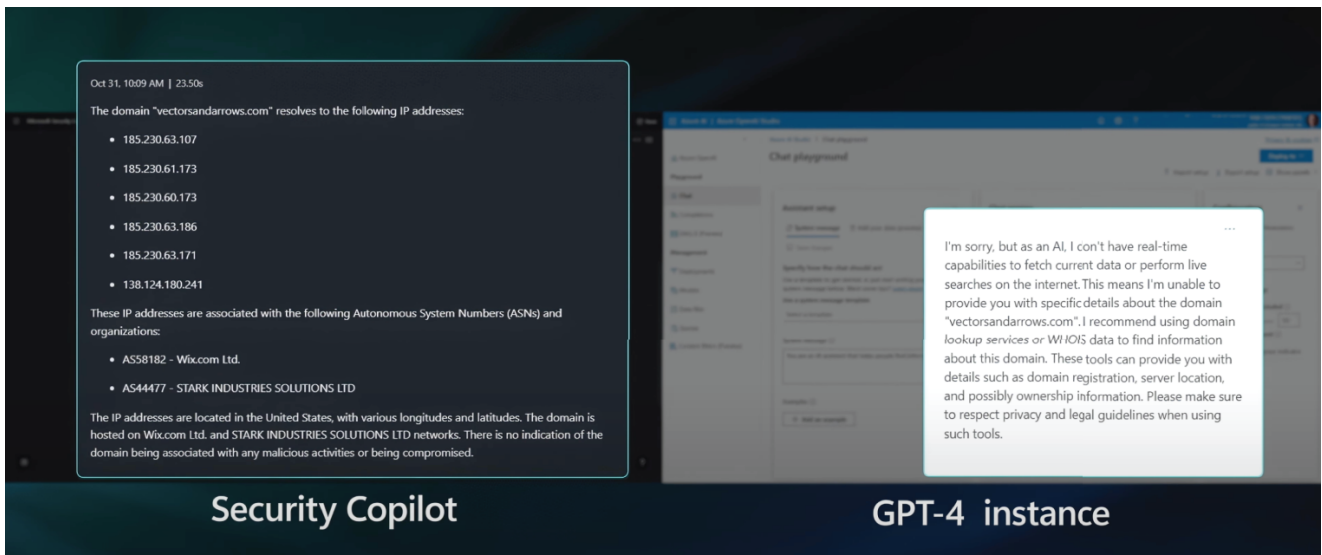


圖 7.4 經過訓練的 MSC 與未經訓練的 ChatGPT

Figure 7.4 Trained MSC vs. untrained ChatGPT

插件：整合持續學習您的特定系統

Plug-Ins: Integrated Continuous Learning About Your Specific System

除了訓練模型之外，MSC 的創建者還能夠通過他們所謂的“插件”添加來自多個系統的實時數據源。這些是專門配置的自訂資料摘要，可近乎即時地將特定系統的相關資料饋送至 LLM。

In addition to training the model, the creators of MSC have been able to add live data feeds from multiple systems via what they call “plug-ins.” These are custom data feeds specially configured to feed data about your specific system into the LLM in near real time.

這是一個徹底的遊戲規則改變者，因為庫存法學碩士通常具有僅延伸到過去某個時間的靜態數據，不會進一步延伸。例如，GPT-4 只能在 2023 年 9 月之前存取資訊 (3)。該模型沒有關於該日期之後發生的任何事情的資訊。

This is a complete game changer because stock LLMs typically feature static data that extends only to some time in the past and no further. For example, GPT-4 only has access to information until September 2023 (3). The model has no information about anything that happened after that date.

相比之下，MSC 可以回答有關幾分鐘前發生的事件的問題（見圖 7.5）。

In contrast, MSC can answer questions about the incident that occurred just minutes ago (see Figure 7.5).

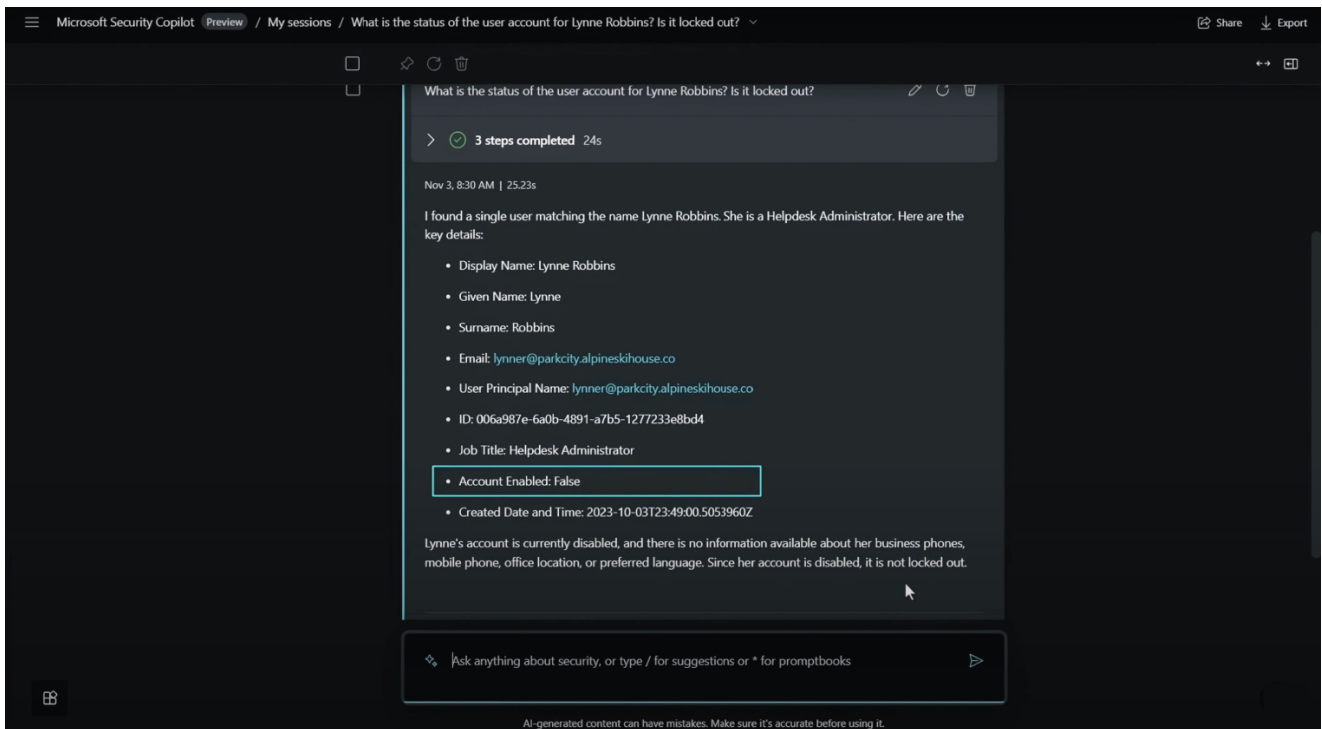


圖 7.5 經過專門訓練的 MSC Copilot 可以回答近乎即時的問題

Figure 7.5 Specially trained MSC Copilot can answer near real-time questions

MSC 可以實現這一點，這要歸功於插件提供的實時數據或為模型提供最新信息的數據源（見圖 7.6）。

MSC can accomplish this thanks to the real-time data supplied by the plug-ins, or data sources that feed the model up-to-date information (see Figure 7.6).

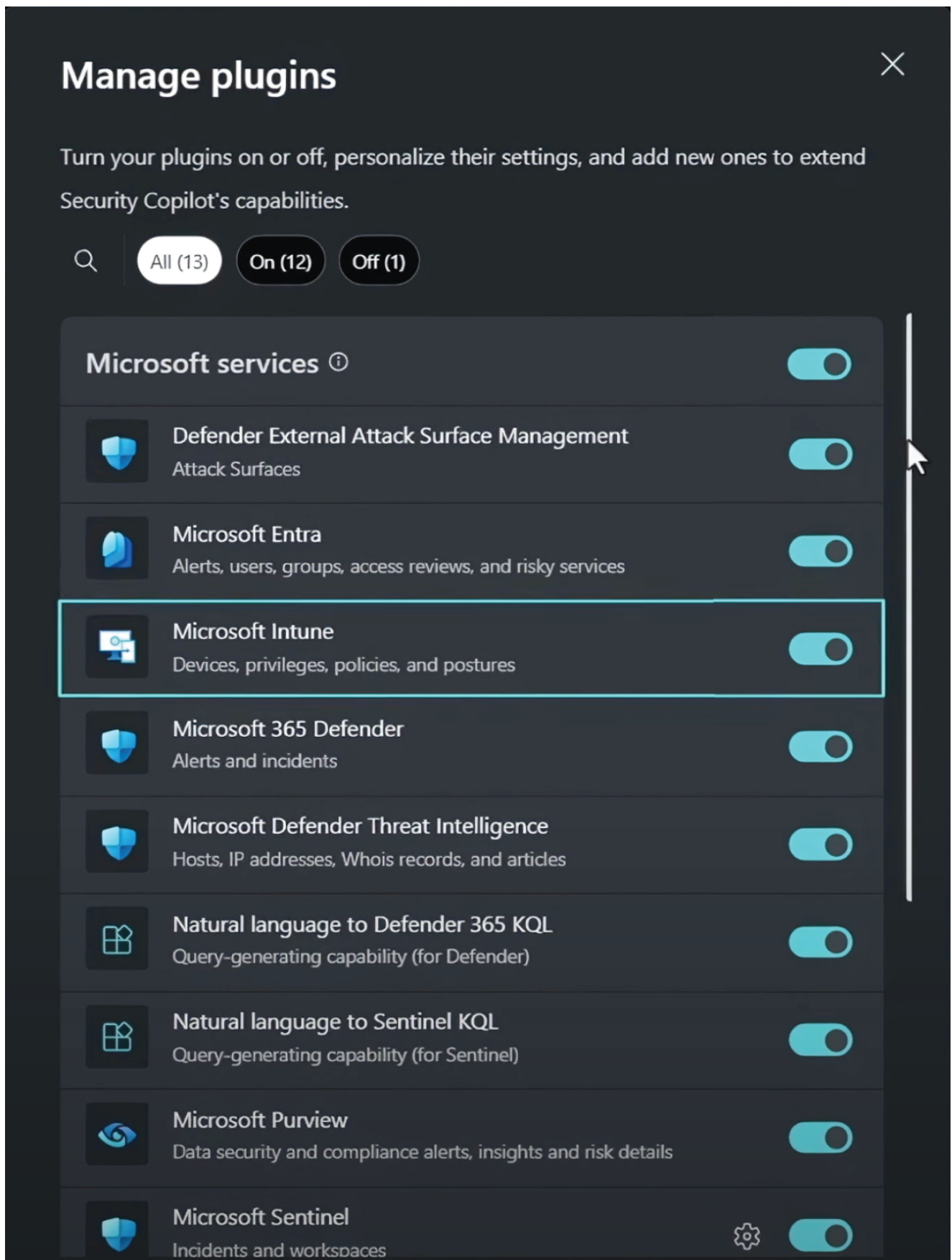


圖 7.6 MSC 可用的外部資料來源（外掛程式）清單

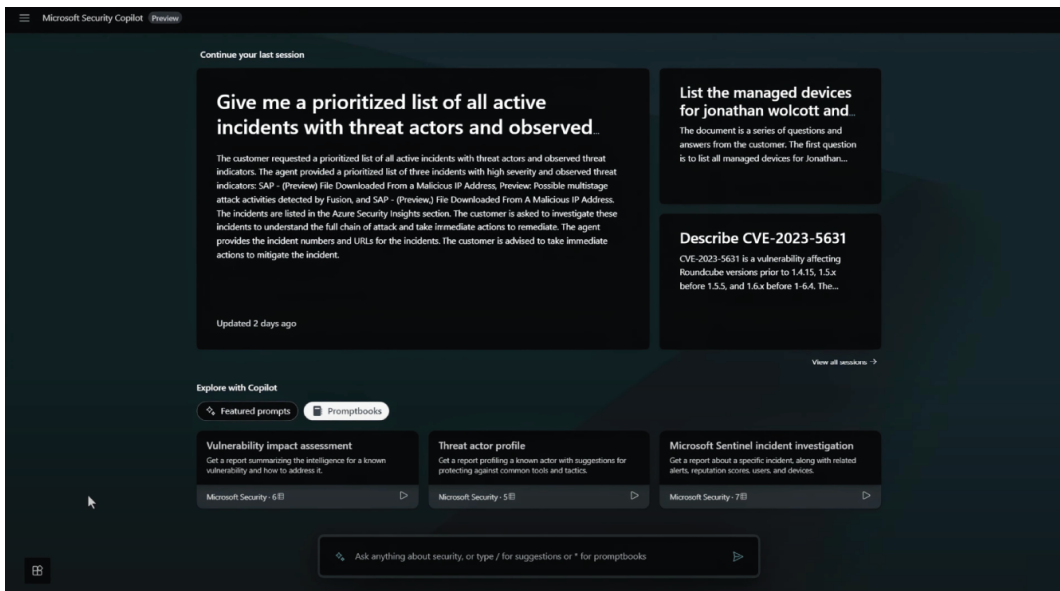
Figure 7.6 A list of external data sources (plug-ins) available to MSC

AI的IA很簡單，專注於聊天

The IA of the AI Is Straightforward, Focused on Chat

這款人工智慧（AI）產品的資訊架構（IA）相對簡單（圖 7.7）：一個登陸頁面，然後是「我的會話」頁面（類似於 ChatGPT 的「歷史記錄」窗格），然後最終是對特定問題的回答或基於聊天的調查會話。無論這個特定 UI 中包含的任何「魔法」都在各個會話中：人類和 AI 之間的交互，以及終端節點頁面（聊天）構成了 Copilot 交互的焦點。（請參閱第 12 章「人工智慧優先應用程式的現代資訊架構」中有關人工智慧優先資訊架構的詳細討論。

The information architecture (IA) for this artificial intelligence (AI) product is relatively straightforward (Figure 7.7): a landing page, followed by a My Sessions page (similar to ChatGPT 's History pane), and then ultimately followed by an answer to the specific question or chat-based investigation session. Whatever “ magic ” is contained in this particular UI is in the individual sessions: the interactions between the human and AI, and that terminal node page (chat) forms the focus of the Copilot interaction. (See a detailed discussion on AI-first information architecture in Chapter 12, “ Modern Information Architecture for AI-First Applications. ”)



My Sessions/ History

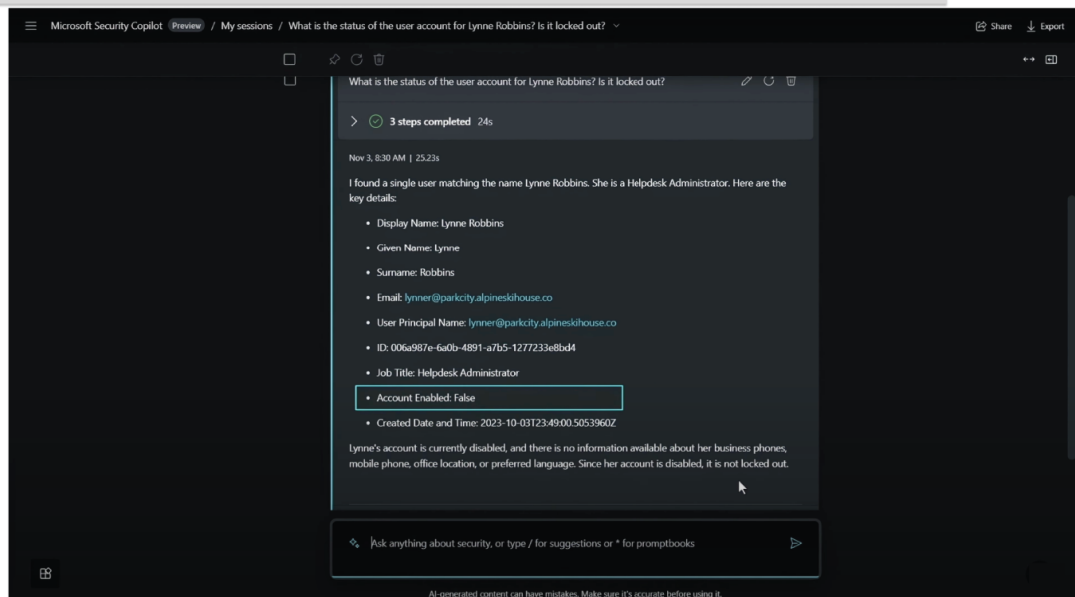


圖 7.7 典型 Copilot 的資訊架構

Figure 7.7 The information architecture of a typical Copilot

Promptbooks : 無需扭曲椒鹽捲餅來編寫提示

Promptbooks: No Need to Twist into Pretzels to Write Prompts

提示工程顯然已經佔領了互聯網。就像牧師主動提供新鮮老鼠內臟占卜一樣，互聯網上充斥著顧問，他們只需支付適度的費用，就能幫助你寫出創建下一個價值數十億美元的帝國的提示，競選總統，或者寫一篇成功的大學論文，讓你進入斯坦福大學。

Prompt engineering has clearly taken over the Internet. Like priests offering to provide divination from fresh rat entrails, the Internet is brimming with consultants who, for a modest fee, will help you write a prompt to create your next billion-dollar empire, run for president, or write a winning college essay that will get you into Stanford.

就像我們在上一章中討論的 Sumo Logic Copilot 建議功能一樣，MSC 提供了一個受歡迎的替代方案：幫助您編寫提示以幫助您與 AI 互動的 AI。這類似於戰術手冊的概念，即基於特定場景的預定戰術（例如，“49 人隊需要 2020 年超級碗的戰術手冊，但手邊沒有一本，所以他們輸給了堪薩斯城酋長隊”）。MSC 使用 Promptbooks 來救援，這是常見調查的預製食譜（見圖 7.8）。這些都推薦在主頁上。這特別方便，因為在安全事件響應中，就像在足球場上一樣，情況壓力極大，而且時間往往非常短。無論提示顧問如何試圖推銷他們的智慧，都沒有時間在現實世界中編寫複雜的多頁 LLM 提示怪物。

Just like Sumo Logic Copilot suggestions feature we discussed in the previous chapter, MSC offers a welcome alternative: the AI that helps you write the prompts to help you interact with AI. This is similar to the concept of playbooks, that is, the predetermined plays based on specific scenarios (as in, “the 49ers team needed a playbook for the 2020 Super Bowl and did not have one handy, so they lost to the Kansas City Chiefs”). The MSC comes to the rescue with Promptbooks, premade recipes for common investigations (see Figure 7.8). These are recommended right on the homepage. That is especially handy because in security incident responses, as on the football field, the situation is extremely stressful, and time is often exceedingly short. No matter how much prompt consultants try to sell their smarts, there is no time to write complex multipage LLM prompt monstrosities in the real world.

便條

NOTE

MSC 和 Sumo Copilot 示範的最佳實務是以自然語言提供特定的簡短查詢，以可預測且可重複的方式與自訂資料互動，從而沿著預先確定的工作路徑產生有價值的資訊。這是 SaaS Copilot 最佳實踐的本質。

The best practice demonstrated by the MSC and the Sumo Copilot is to provide specific, short queries in a natural language that interact with the custom data in predictable and repeatable ways that yield valuable information along a predetermined work path. This is the essence of SaaS Copilot best practice.

最後的思考

Final Thoughts

SaaS 是一項嚴肅的工作，而 SaaS Copilots 現在是一項嚴肅的業務。若要成為成功的 Copilot 設計師，您必須弄清楚 Copilot 將處理哪些工作，並為您的功能規劃適當的螢幕空間量：側邊列、大面板或專用體驗。接下來，您必須研究要將哪些其他資料來源引入體驗，並找出您將使用哪些自訂資料集來訓練 Copilot LLM。（回想一下，第 4 章「數位學生 - 系統實體元件的數位表示」中討論的數位學生練習對於此目的來說是最方便的！我建議保持資訊架構簡單，首選模式是會話/歷史記錄清單作為中心輻射設計中的中心樞紐。不過，Copilot 的用途和資訊架構可能會有所不同。最後，必須考慮 Copilot 將向使用者提供哪些「提示手冊」（例如，預製提示、建議和自動完成）和 UI 控制項，以協助他們繼續流程（更多內容請參閱第 9 章「LLM 設計模式」）。

SaaS is serious work, and SaaS Copilots are now a serious business. To be successful as a Copilot designer, you need to figure out what tasks the Copilot will be handling and map out the appropriate amount of screen real estate for your feature: side bar, large panel, or dedicated experience. Next you have to research which additional data sources to bring into the experience and figure out what custom data set you will use to train the Copilot LLM. (Recall that the digital twin exercise discussed in Chapter 4, “ Digital Twin—Digital Representation of the Physical Components of Your System, ” is most handy for this purpose!) I recommend keeping the information architecture simple, with a go-to pattern being the list of sessions/history as a central hub in a hub-and-spokes design. However, your Copilot ’ s purpose and information architecture may differ. Finally, it ’ s essential to consider what “ promptbooks ” (e.g., premade prompts, suggestions, and auto-complete) and UI controls your Copilot will be providing to the users to help them continue with their flow (more on this in Chapter 9, “ LLM Design Patterns ”).

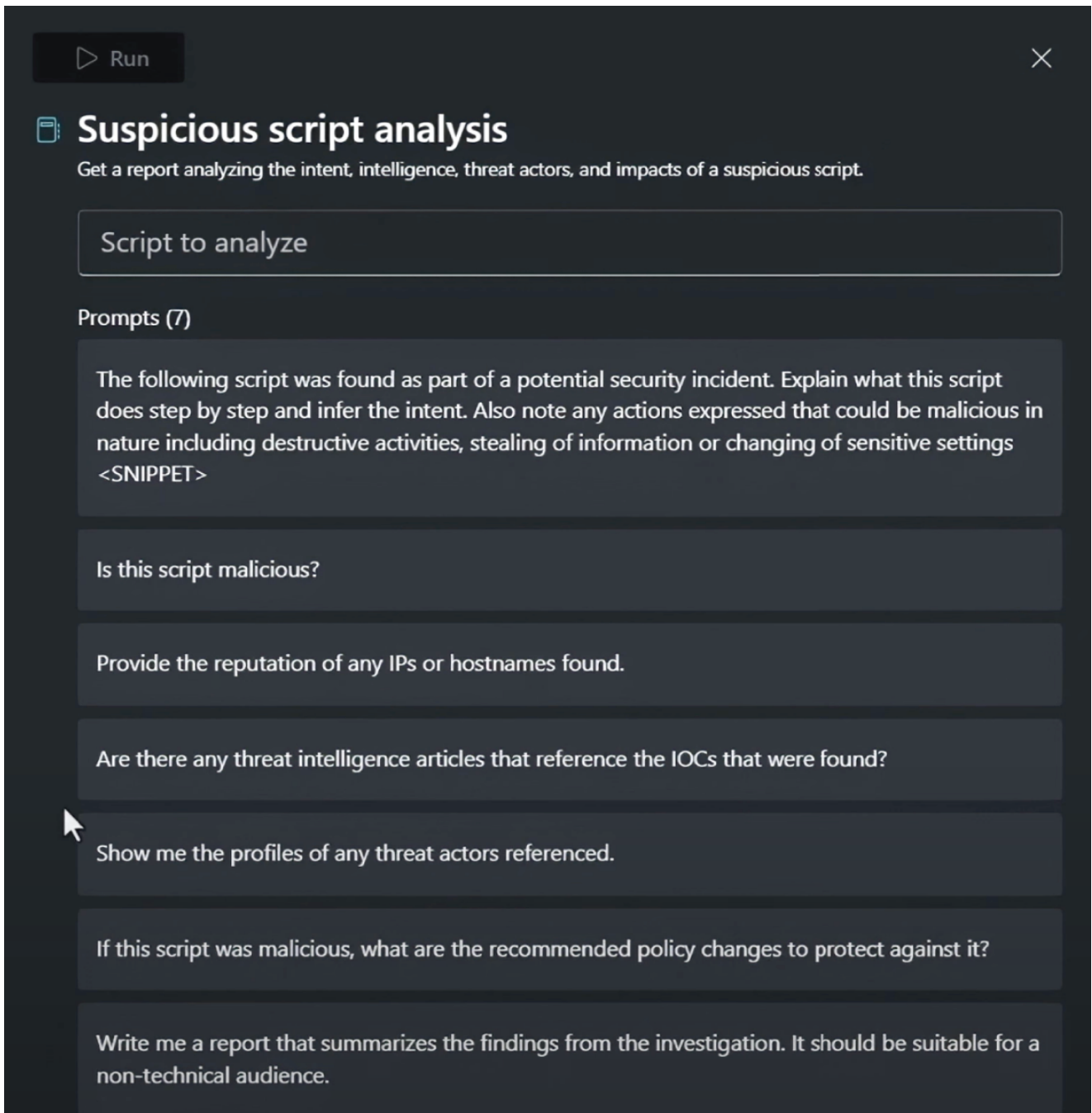


圖 7.8 MSC 提示手冊為安全調查提供了簡單的起點

Figure 7.8 MSC Promptbooks provide easy starting points for a security investigation

設計練習：建立您自己的行動副駕駛

Design Exercise: Create Your Own Mobile Copilot

在您的第一個實際 UI 設計練習中，您將為自己的使用案例設計行動 Copilot UI：

In your first actual UI design exercise, you will be designing a mobile Copilot UI for your own use case:

- 以 MSC 和 Sumo Copilot 為例，集思廣益並勾勒出一些想法，以使用標準 Copilot 「聊天」功能來驅動流程。您的新 Copilot 將是移動的，因此請考慮使用移動設備屏幕提供的全部空間。（當然，如果你願意，你也可以把它做成一個模態或部分面板——畢竟，它是你的設計！ Using the MSC and Sumo Copilot as examples, brainstorm and sketch some ideas for your experience using the standard Copilot “ chat ” features to drive the flow. Your new Copilot will be mobile, so consider using the entire real estate afforded by the mobile device screen. (You can, of course, make it a modal or a partial panel as well, if you like—it is, after all, your design!)
- 根據您的使用案例，請考慮任何其他資料來源，這些資料來源有助於合併到訓練模型的體驗中，並為其提供即時態勢感知。 Depending on your use case, consider any additional data sources that will be useful to incorporate into the experience to train the model and give it real-time situational awareness.
- 考慮您設計的資訊架構。您的 Copilot 需要有狀態嗎？您需要記住並處理多個人類與人工智慧的對話嗎？如果是這樣，使用者將如何存取先前的對話？您可以使用「歷程記錄」頁面作為簡單軸輻式 IA 模型的中心嗎？ Consider your design ’ s information architecture. Does your Copilot need to be stateful? Do you need to remember and handle multiple human – AI conversations? If so, how would the user access previous conversations? Can you use the History page as a hub for a simple hub-and-spokes IA model?
- 考慮您的使用者可能想要看到哪些提示書。 Consider what promptbooks your users might want to see.
- 您的 Copilot 應該提供哪些螢幕 UI 控制項給使用者，以協助他們處理流程？ What onscreen UI controls should your Copilot provide to your users to help them with their flow?

使用矩形便利貼繪製所有內容。如果您需要靈感，請看以下範例。在完成自己的設計練習之前，不要繼續下一章。

Sketch everything using rectangular sticky notes. If you need inspiration, look at the following example. Do not proceed to the next chapter until you complete your own design exercise.

設計練習範例：生命時鐘副駕駛

Design Exercise Example: Life Clock Copilot

在我們在第 1

部分中探討的故事板中，基本流程非常簡單：用戶用手機拍攝食物的照片，AI 告訴他們食物的類型、數量以及它如何影響用戶的壽命和健康。我們可以輕鬆地利用本章中的 Copilot 設計原則來設計 UI 來執行此任務。在圖 7.9 的草圖線框圖中，使用者開啟一個空白的 Life Clock 應用程式（我們將此設計稱為「Life Copilot」）螢幕，並利用手機的相機拍攝食物並將其上傳到 Copilot AI。然後，Life Copilot 對照片進行分析並打印出食物類型、對份量的猜測，以及對這種特定食物可能對健康產生的長期和短期影響的尖刻固執己見的純文本猜測：

In the storyboard we explored in Part 1, the basic flow is very simple: The user takes a photo of the food with their phone, and the AI tells them the type of food, its quantity, and how it might affect the user's longevity and health. We can easily leverage the Copilot design principles in this chapter to design the UI to perform this task. In the sketched wireframe in Figure 7.9, the user opens up a blank Life Clock app (let's call this design "Life Copilot") screen and utilizes the phone's camera to photograph a food item and upload it to the Copilot AI. The Life Copilot then performs the analysis of the photo and prints out the food type, its guess at the portion size, and a snarky opinionated plain-text speculation on the likely long- and short-term health effects of this particular food:

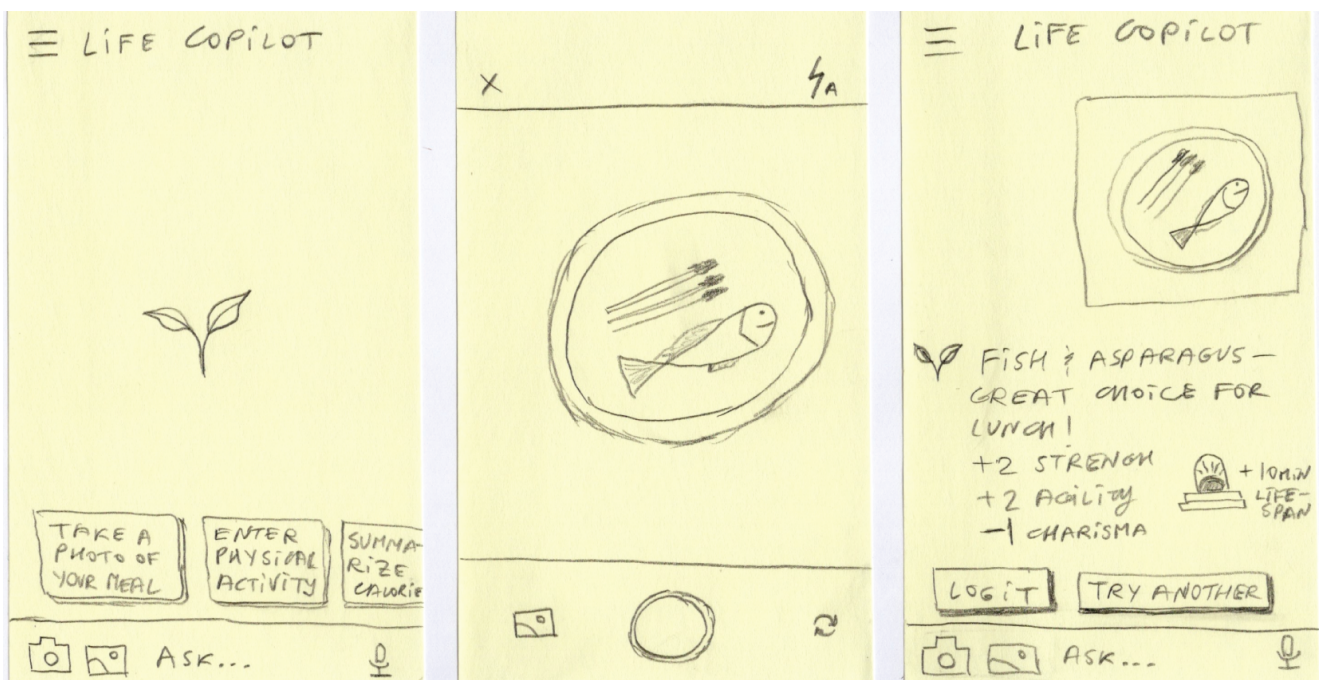


圖 7.9 Life Copilot 應用程式的簡單工作流程

Figure 7.9 A simple workflow for a Life Copilot app

為了執行此類分析，Life Copilot 將需要標記食品的圖像數據庫和複雜的分析引擎，如第 4 章的數字學生練習中所述。

To perform this kind of analysis, the Life Copilot will need the image database of labeled food items and a sophisticated analysis engine, as described in the digital twin exercise in Chapter 4.

有趣的是，在本書出版前不久，我最喜歡的健身追蹤器應用程式 MyNetDiary 中出現了一個非常相似的功能，稱為「AI 膳食掃描」（圖 7.10）。

Interestingly, shortly before the publication of this book, a very similar feature called “AI Meal Scan” became available in my favorite fitness tracker app, MyNetDiary (Figure 7.10).

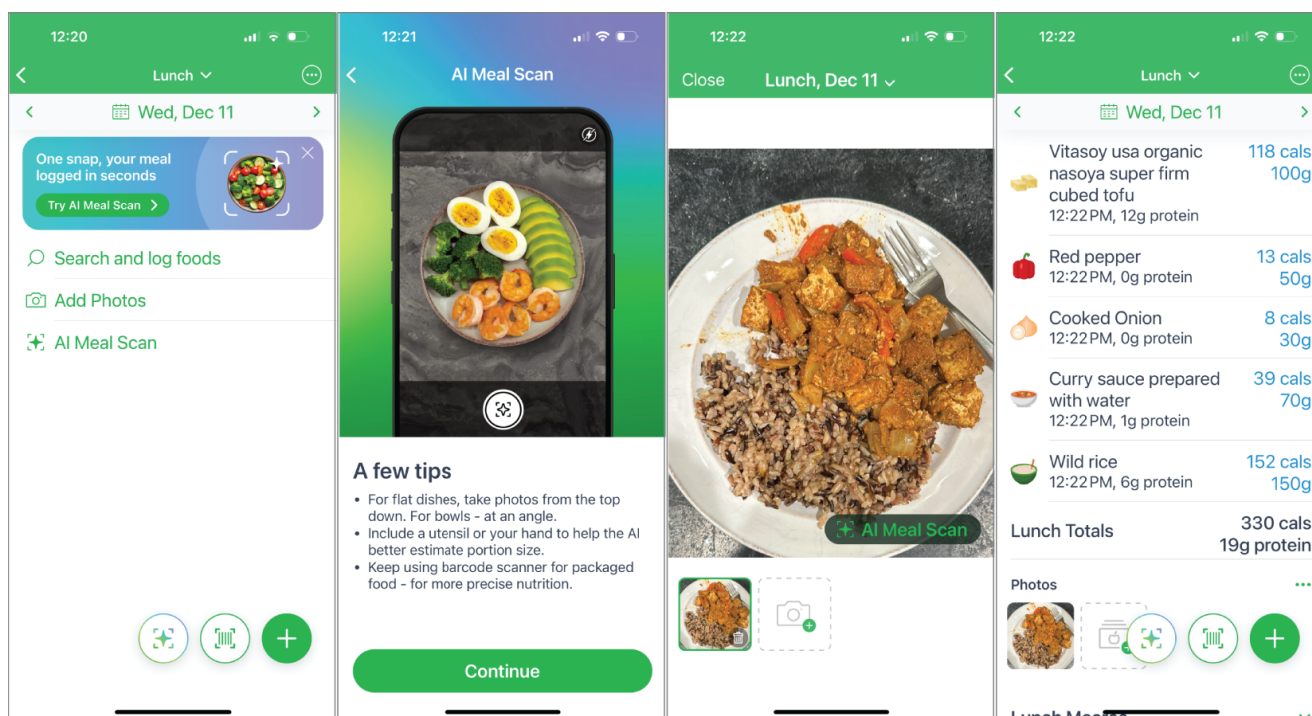


圖 7.10 MyNetDiary 應用程式的 AI 膳食掃描功能

Figure 7.10 AI Meal Scan feature from MyNetDiary app

我們早期的設計非常接近！

Our early design was pretty close!

按照 MyNetDiary 應用程式的方式簡單地記錄食物與利用智能食物選擇來延長壽命和改善用戶健康之間的一個關鍵區別是建議的質量。對於 Life Copilot 應用程式，我們需要進行一些額外的微調/檢索增強生成，以溫和地引導用戶選擇更健康的食物，而不會聽起來過於迂

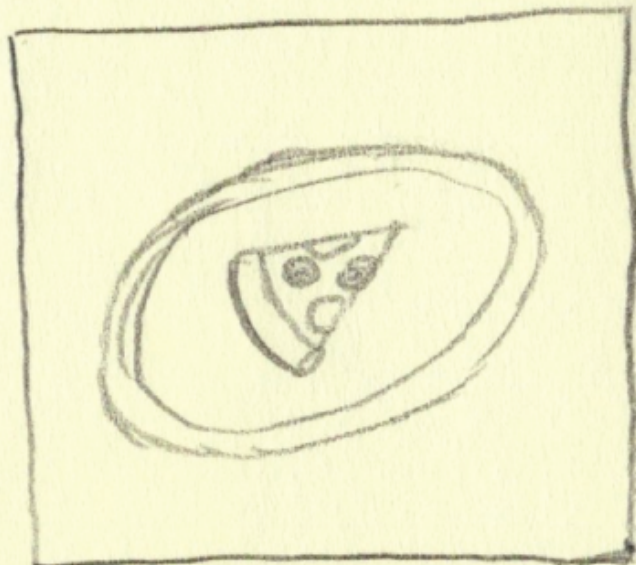
腐或說教——換句話說，我們的法學碩士應該具有富有同情心和支持性（但強硬）的健康教練的個性，他可以建議和指導用戶度過困難的生活方式改變。

One crucial difference between simply logging the food the way MyNetDiary app is doing and leveraging smart food choices to extend lifespan and improve the user ' s health is the quality of advice. For the Life Copilot app, we will need to do some additional fine-tuning/retrieval-augmented generation that would gently guide the user through healthier choices of food without sounding too pedantic or preachy—in other words, our LLM should have the personality of a compassionate and supportive (but tough!) health coach, who can advise and guide the user through difficult lifestyle changes.

我們也希望我們的 Life Copilot 是有狀態的。因此，它可以了解日曆日期和一天中的時間，以記下當天消耗的卡路里、微量營養素和運動的運行總數，並保持每日統計，以便更好地建議用戶如何處理下一餐選擇，例如他們是否“負擔得起”（熱量上）多吃一片意大利辣香腸披薩和甜點，或者應該選擇繞著街區走一圈（見圖 7.11）。

We also would want our Life Copilot to be stateful. Thus it can be aware of the calendar date and time of day to note the running total for the day of the consumed calories, micronutrients, and exercise and keep a running daily tally, so it can better advise the user on what to do with the next meal choice such as if they can “ afford ” (calorically speaking) to have an extra slice of pepperoni pizza and a desert or should instead go for a walk around the block (see Figure 7.11).

☰ LIFE COPILOT

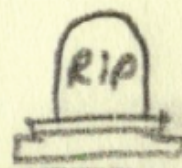


PEPPERONI

✓ PIZZA FOR LUNCH? MAY NOT BE THE BEST IDEA. REMEMBER YOU HAD DOUGHNUTS FOR BREAKFAST ALREADY! LOOK ^{FOR} PIZZA W/ VEGGIE & ROAST CHICKEN OR SHRIMP INSTEAD.

HOWEVER, IF YOU INSIST ON MY ANALYSIS: "PEPPERONI PIZZA":

+1 TO STRENGTH
+1 TO VITALITY
-1 TO AGILITY



-5 MIN
LIFE-
SPAN

LOG IT

TRY ANOTHER

圖7.11 帶有健康教練AI解說的Life Copilot應用程式

Figure 7.11 Life Copilot app with health coach AI commentary

除了尖刻的建議之外，我們還可以添加輸入智慧手錶等穿戴式裝置無法捕捉到的稱重和運動的功能。這可以透過傳統的 HTML 表單來完成，甚至可以更簡單地完成：透過 LLM 聊天，只需要求 Life Copilot 新增此資訊即可。

In addition to snarky advice, we could add the ability to enter weigh-ins and exercise that are not captured by wearable devices such as smartwatches. This can be accomplished through traditional HTML forms or even simpler: through the LLM chat, simply by asking the Life Copilot to add this information.

最後，我們還想為每日對話歷史記錄添加一個簡單的中心輻射型 IA 模型，其中包含每週總計和刪除錯誤條目的能力。我們還可以添加生成精美報告的功能（以及通常的標誌性報告），顯示一段時間內的健康趨勢，並在他們的生活方式選擇對他們的壽命產生負面影響時提醒用戶。我們將在下一章中作為「副駕駛報告」練習的一部分勾勒歷史和報告（並在本書後面的適當時間處理動態警報、預測和週期性以及 DOI 儀表板，因為這些主題中的每一個都值得有自己的章節）。

Finally, we would also want to add a simple hub-and-spoke IA model for the daily conversation history with weekly totals and the ability to delete erroneous entries. We can also add the ability to produce a fancy report (along with the usual signature snark) showing health trends over time and alerting users if their lifestyle choices negatively impact their lifespan. We will sketch the History and the Report as part of the “ Copilot Report ” exercise in the next chapter (and tackle dynamic alerts, forecasting, and periodicity, and DOI dashboards in due time later in the book, as each of these topics deserves its own chapter).

但首先，在我們討論所有其他好東西之前，輪到您為自己的用例繪製 Copilot 草圖了。在完成此練習之前，不要進入下一章（或收集 200 美元）。我保證，這應該只需要幾分鐘，而且時間會花得值！

But first, before we get to all the other goodies, it ' s your turn to sketch a Copilot for your own use case. Do not advance to the next chapter (or collect \$200) until you have completed this exercise. It should only take you a few minutes, and it will be time well spent, I promise!

Do or Do Not. There is no Try.

參考

References

1. 1. Microsoft 機械 (2023)。 Microsoft Security Copilot 的運作方式。 YouTube。 www.youtube.com/watch?v=0lg_derTkaM1. Microsoft Mechanics (2023). How Microsoft Security Copilot works. YouTube. www.youtube.com/watch?v=0lg_derTkaM1
2. 2. 亞馬遜 QuickSight (2023)。 在 QuickSight 中使用 Amazon Q 的生成式 BI。 YouTube。 <https://youtu.be/uBG7IFXV6II?si=1UZkBpNkCXcJO-Wc2>. Amazon QuickSight (2023). Generative BI with Amazon Q in QuickSight. YouTube. <https://youtu.be/uBG7IFXV6II?si=1UZkBpNkCXcJO-Wc2>
3. 3. 巴斯蒂安 , M. (2023)。 ChatGPT 可能已經更新了最新資訊。 解碼器。 <https://the-decoder.com/chatgpt-might-have-been-updated-with-more-recent-information3>. Bastian, M. (2023). ChatGPT might have been updated with more recent information. The Decoder. <https://the-decoder.com/chatgpt-might-have-been-updated-with-more-recent-information>