THE DEVIL'S IN THE DETAILS

Neil Roseman *outlines the new popularity of AI applications and the vulnerabilities within them*

Picture credit: AdobeStock

Let's start with a basic and unfortunate truth about security: innovation brings risks. It seems like the whole world has been seized with AI fever over the last year. It's already revolutionising a variety of fields – enabling greater productivity and profound new use cases. According to many analysts, we haven't seen anything yet. Since the release of ChatGPT, businesses have been scrambling to find new uses for generative AIs and turn these services to their greater profit and productivity. It's exciting. Yet, it's also presenting challenges throughout the IT ecosystem.

Much is currently being made about the ways generative AI applications can assist security, help with threat hunting and aid developers in writing more secure code. The irony here is that AI applications – whether they assist security or not–- are in themselves potential vectors for attack. The large majority of these LLMS and AI applications – such as ChatGPT – are web and API-based and from that point of view, just as vulnerable to attack as other web-based applications. This isn't an abstract threat – there have been multiple recorded attacks on AI applications. Not even ChatGPT, the most public face of AI, is safe. In May 2023, the popular application was attacked through a vulnerability in the Redis Open-Source library, which allowed users to see the chat history of other users. In November that year, ChatGPT underwent a DDoS attack, causing major outages across the application and its API for two hours.

To make matters worse, LLMs commonly sit in front of sensitive proprietary data and connect to other vulnerable, internally facing applications, making an AI application a potential vector for grievous attack.

There are few better examples of technology hype than artificial intelligence. The luminaries of the tech industry are constantly saying that this technology will either usher in a new golden age or bring about the apocalypse. The enthusiasm to take advantage of AI's possibilities has led to the lightning-fast development of AI applications, which are then eagerly deployed and otherwise consumed by businesses and consumers alike.

Hype, unfortunately, creates perverse incentives. What is currently happening with application development, will likely also happen with Artificial Intelligence applications.

Software developers are under unprecedented pressure to produce. The sheer demand for new services, features and applications has put a big strain on the software development process, which should be a meticulous and careful process. This has created a number of damaging effects, but in principle it does two things. First, it forces developers to work faster, leaving them liable to make more mistakes and unknowingly introduce more basic code vulnerabilities into the development process. Second, it adds pressure to the code review process in which those vulnerabilities would otherwise be found and fixed. The AI application development process is likely undergoing that same pressure.

The security possibilities for Chat GPT are exciting, but also present challenges throughout the IT ecosystem

A group called huntr runs the world's first AI/ML bug bounty programme and collects data from over 15,000 security researchers, finding vulnerabilities in AI applications. Its April 2024 report reveals that 48 vulnerabilities had been discovered in that month alone – the large majority being registered High or Critical Severity – representing a 220 percent growth in the vulnerabilities first reported in November 2023.

The large majority of these vulnerabilities were rated either critical or severe and stemmed from basic mistakes in development. There were a large amount of basic web application vulnerabilities which Marcello Salvati, a senior threat researcher with huntr, later told press: "These types of vulnerabilities are rarely seen in the majority of web applications these days because of the prevalence of secure coding practices and web frameworks with 'built-in' security guardrails." The fact that vulnerabilities are now springing up which are now rare in normal application development, should give us pause. Salvati added that their reemergence shows that in the development of AI/ML tools, security may be an afterthought and that these models may not have learned some of the most basic lessons that regular application developers have over the last decade.

Many of these vulnerabilities stem from the broader ecosystem in which these AIs exist. That is to say, they become vulnerable or expose data when they connect with other services. Generative AI plugins, for example, allow businesses to integrate the functionality of the generative AI service with third-party services like Google Drive. Yet those interactions have been shown to raise potential vulnerabilities and vectors for attack by exploiting OAuth authentication. Researchers have found that the moment those plugins request permission to transmit user data provides an opportunity to attackers to redirect those users to malicious URLs, download their own malicious plugins and gain access to ChatGPT user accounts and all the associated data. Research has also found the inverse too – in which attackers can use those malicious plugins to gain access to the account that is being connected to, such as their GitHub account.

SIGNIFICANTLY MORE SECURITY VULNERABILITIES WERE FOUND IN AI-GENERATED CODE

Vulnerabilities in those AI applications can be seen as simple vectors into the systems and data those AI are in contact with, but it's also important to note that AI and ML models are also part of much larger supply chains, some of which they are the final link in and others in which they are a key point of production.

AI models are commonly welded together from a variety of pre-fabricated and open source components. Not only can those in themselves introduce vulnerabilities into the final AI application, but attackers will actively try to insert malicious code and vulnerabilities into the supply chain so that they can be exploited later.

One study from North Carolina State University found, for example, that the Deep Neural Networks (DNNs) widely used in AI models were replete with vulnerabilities that would allow models to be maliciously manipulated. One of the authors of the study, Tianfu Wu, later noted that: "Attackers can take advantage of these vulnerabilities to force the AI to interpret the data to be whatever they want. This is incredibly important because if an AI system is not robust against these sorts of attacks, you don't want to put the system into practical use – particularly for applications that can affect human lives."

The same is true for the other component parts of an AI application too. In late 2023, a critical vulnerability was discovered in the TorchServe machine learning frameworks – maintained by Amazon and Meta – which could permit attackers to access an AI model's proprietary data and then put their own malicious models in production. Huntr, the aforementioned bug bounty platform, commonly finds critical and severe vulnerabilities within exactly these kinds of open source frameworks.

It doesn't just stop there. The AI supply chain is such a complex thing that there are multiple potential points of failure. Training data, for example, is one of the most fundamental resources for an AI model – it's the sample data that trains them what to do. There are all kinds of risks when it comes to using that data; it could be of poor quality, inaccurate or replete with biases, which will get fed into that model. That can quite easily be a perfectly innocent mistake or oversight, but increasingly it's a malicious security risk too. In fact, OWASP considers malicious poisoning of training data to be one of the top risks for LLMs and AIs.

FORCING DEVELOPERS TO WORK FASTER LEAVES THEM LIABLE TO MAKE MORE MISTAKES

However we can't just see AI applications as the end point of one supply chain, we need to see it as the beginning of another too. Software developers are increasingly using AI to assist in their jobs. Generative AIs have indeed become particularly useful to many who are under ever greater pressure to produce as quickly as possible. There are already many generative AI applications and plugins which help with this task including GitHub's Copilot and Amazon's CodeWhisperer.

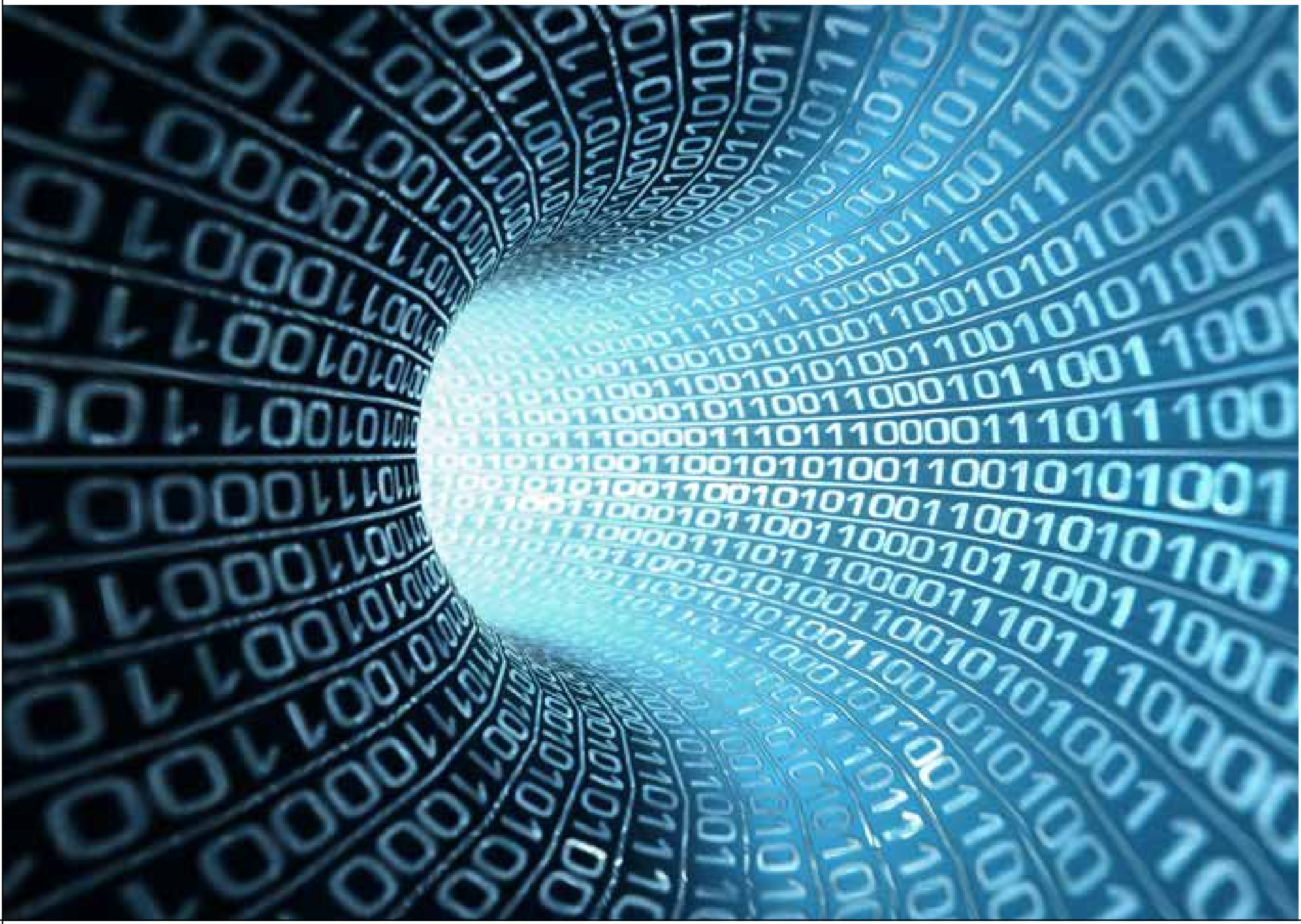
Yet they appear to have been given a great degree of trust in writing those applications, which may not be entirely justified considering the common and repeated code errors that generative AIs spit out. According to security researcher Natan Nahorai, generative AIs often give advice that leads to insecure decisions or generates code with vulnerabilities.

One study from Stanford University found that code written by developers who used AI tools was less secure than those who didn't. In fact, significantly more vulnerabilities were found in the AI-generated code. The study adds that there's a sort of Dunning-Kruger when it comes to AI. Researchers found that using AI gave developers a false sense of security when developing software, imagining that the code they generated with AI assistance to be more secure, when, in fact, it was less. The paper concluded: "We observed that participants who had access to the AI assistant were more likely to introduce security vulnerabilities for the majority of programming tasks, yet were also more likely to rate their insecure answers as secure compared to those in our control group."

Innovation is a near-unstoppable force and it can't be resisted without significant competitive drawbacks. AI models are sparking a revolution and for understandable reasons businesses want to use that functionality to their own benefit. That said, AI must be approached with caution – there is a galaxy of complications involved in using and building AI applications, which can introduce serious risk and mitigate the overall benefits of AI. Anyone who wants to take advantage of those benefits has to adopt those potential threats into their risk analyses before charging headfirst into this nascent field ●

Neil Roseman is CEO of Invicti Security and Advisory Partner at Summit Partners, with over 20 years of experience in building high-scale software and web services for consumers and the enterprise.

The AI supply chain is such a complex thing that there are multiple potential points of failure



Picture credit: iStock