Proceedings of International Mechanical Engineering Congress & Exposition
IMECE2025
November 16-20, 2025, Memphis, TN, USA

IMECE2025-165962

PRACTICALLY LEVERAGING LLMS FOR MANUFACTURING CYBERSECURITY

Curtis Taylor
Oak Ridge National Laboratory (ORNL)
Oak Ridge, Tennessee, USA
Email: taylorcr@ornl.gov

Monika Akbar
University of Texas at El Paso (UTEP)
El Paso, Texas, USA
Email: makbar@utep.edu

Gabriela Ciocarlie
University of Texas at San Antonio (UTSA)
San Antonio, Texas, USA
Email: gabriela.ciocarlie@utsa.edu

Matthew Luallen
University of Illinois at Urbana-Champaign (UIUC)
Urbana, Illinois, USA
Email: mluallen@illinois.edu

ABSTRACT

Cybersecurity in manufacturing faces increasing threats and skilled personnel shortages. Large language models (LLMs), especially multi-modal variants, offer significant potential to rapidly parse complex data and identify vulnerabilities. This study explores the deployment of multi-modal LLMs in manufacturing cybersecurity, emphasizing their ability to bridge knowledge gaps and provide actionable insights. We evaluated offline and cloud-based models across two use cases, an analysis of a 100-page digital thread handbook used at a manufacturing facility, and vulnerability remediation in a manufacturing plant. The results highlight trade-offs between data privacy and model capability in understanding and prioritizing cybersecurity risks. Vision-based LLM limitations were evidenced through diagram analysis failures such as building layouts and network architecture diagrams, underscoring the need for human oversight and model transparency. Specialized cybersecurity embeddings showed promise for nuanced vulnerability analysis but still lack the ability to formally analyze multi-modal documents. Our findings emphasize current strengths, limitations, and pathways for using out-of-the-box LLMs and agentic artificial intelligence (AI) among industrial cybersecurity frameworks.

Keywords: Cybersecurity, AI, LLM, multi-modal, Industrial Control Systems (ICS), Operational Technology (OT), Manufacturing

1 Introduction

Manufacturing environments generate vast amounts of data but present significant cybersecurity challenges due to their inherent complexity and the convergence of information technology (IT) and operational technology (OT). The cybersecurity workforce gap continues to widen, exacerbated by the diverse and specialized skill sets required. Within cyber, knowledge around secure programming practices, forensic analysis, vulnerability remediation, and more create challenges to securing IT and OT systems. Recent advances in AI, particularly LLMs, offer potential solutions by bridging knowledge gaps and augmenting the capabilities of operators who possess varying levels of cybersecurity expertise but also introduces additional required skill sets such as those needed to effectively and securely use AI. The converging needs for cybersecurity and easy access to AI tools has led to leveraging AI for cybersecurity directly. The necessity of these two skill sets does present an issue for future adoption for cybersecurity tools but are not a direct focus of this work. Some foundational LLM models have cybersecurity training to help knowledge issues in cybersecurity but not necessarily to the depth required to ensure manufacturing system security. This challenge is difficult enough that cybersecurity companies such as Cisco have started the process of fine-tuning existing LLMs for problems [1] and have built their performance analysis on text datasets such as CTIBench [2] (dataset itself starting with

LLM-generated questions). While the use of LLMs for cybersecurity is growing, there are several shortcomings that existing research in this area does not consider and are a motivation for this work:

1. **Multi-modal nature of existing data**: Existing documentation, including architectural and network diagrams, are visual in nature and link information logically and spatially together. As discussed by Cisco [1], even getting text output in a manageable format is extremely time-consuming and does not consider the multi-modal aspect of data. To that end, this effort does not assume that a manufacturer understands the technical complexities of LLMs such as prompt engineering, context windows, or text embedding via Retrieval-Augmented Generation (RAG). As such, we do not attempt to optimize the documents by pre-processing text and images, instead relating our results to how a model would be realistically leveraged by a non-LLM expert with junior-level cybersecurity expertise in the manufacturing domain.
2. **Evaluation criteria based on text**: Existing cyber benchmarks only consider text-based answers. Such answers can be both technical and in-depth but require the context to already be structured in such a way that the model can analyze it. If a manufacturer has documented their network security configuration in the form of a diagram, such real-world examples will not have been explicitly considered in existing benchmarks. Further, this work recognizes that creating a structured multi-modal evaluation framework such as [3] for such a broad set of requirements that Small to Medium-sized Manufacturers (SMMs) data may bring is extremely challenging. Rather than focus on building a generalized evaluation framework, we consider a real-world evaluation scenario using data from the manufacturing domain.
3. **Impracticality of fine-tuning**: Undoubtedly, with good data in the right format, existing LLMs can be fine-tuned to better recognize specific tasks, including cybersecurity-related tasks. Specific tools, such as security monitors with set text outputs and structured responses (e.g., JSON), could be trained on a company's existing data and previous intrusion incidents and better detect incidents based on the company's threat profile. However, this requires numerous technical (e.g., document pre-processing) and financial (e.g., cost to train) constraints that are impractical for SMMs. Instead, the vast majority of manufacturers today are most likely to leverage existing open or closed LLMs without fine-tuned outputs.

With these limitations in mind, this study investigates the *practical* application of LLMs within an industrial cybersecurity context by assessing their performance using examples from real-world data from an advanced manufacturing facility. Specifically, we evaluate whether LLMs can effectively serve as a "junior cybersecurity analyst" capable of extracting critical insights from complex technical documents and identifying security vulnerabilities without a heavy uplifting (e.g., data pre-processing, fine-tuning, prompt engineering). As a result of this work, we better understand what the state-of-the-art (SOTA) LLMs and Vision LLMs can provide to manufacturers using relevant, existing manufacturer data. This work does not seek to provide a benchmark to compare against, as it cannot readily be generalized across heterogeneous datasets, but instead, provides insight into future research considerations that should be made within the manufacturing domain. The two use cases considered for practically using LLMs for manufacturing cybersecurity are the following:

1. **Interpreting Cyber Data in a Manufacturing Digital Handbook**: In this use case we evaluated the effectiveness of multi-modal LLMs in interpreting cyber-relevant information within a manufacturing handbook that included diagrams, tables, text, and other sample data.
2. **Vulnerability Analysis via LLMs**: In this use case we explored the use of LLMs in identifying Common Vulnerability Enumerations (CVEs) from network scan outputs, simulating realistic cybersecurity workflows in a manufacturing environment.

2 Use Case 1: Interpreting Cyber Data in a Manufacturing Digital Handbook

While keeping manufacturing context in mind, we deployed LLMs initially focused on a 100-page digital thread handbook containing network infrastructure diagrams, programmable logic controller (PLC) details, configuration tables, software and sensor descriptions, and narrative text on manufacturing processes. Our initial bold approach aimed to leverage agentic AI workflow pipelines [4]. We envisioned a technical framework involving RAG [5] tools, such as Danswer.ai [6][1] or custom implementations using other agentic tools with multi-modal [7], LLMs tailored to various data inputs such as source code, CVE / Common Weakness Enumeration (CWE) databases, internal documents, and community notes, as indicated in Figure 1. RAG is the current SOTA, for reducing imaginative specifics (hallucinations) and grounding LLMs in known data but is far from solving document ingestion and analysis.

Our ultimate goal was to generate prioritized cybersecurity responses from pre-evaluated, manufacturing-specific prompts. This refers to prompts that have been validated through benchmarking to provide consistently accurate and precise responses based on specific environmental contexts, such as type of document, LLM and its settings, and the agentic AI pipeline. With this manufacturing context in mind, we deployed LLMs and

[1] Danswer.ai was renamed to Onyx after our testing. For consistency, the tool is referred to as Danswer.ai throughout the document instead of Onyx.

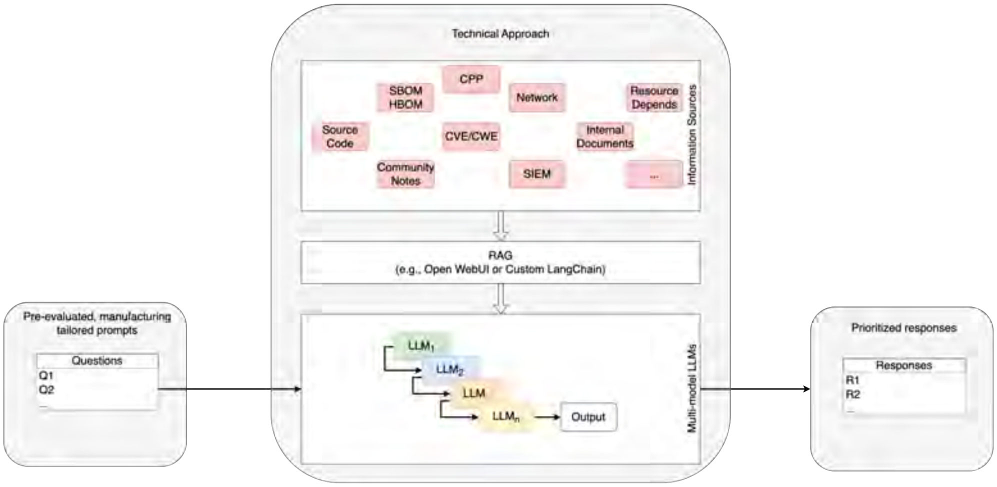


FIGURE 1. Illustration of our idealized RAG pipeline for use in the manufacturing cybersecurity context. Pre-evaluated, manufacturing-specific prompts are processed through a retrieval system integrating diverse information sources, including source code, CVE/CWE databases, internal documents, and community notes.

manufacturing-specific prompts focusing on a 100-page digital thread handbook with representative data shown in Figure 2.

The initial vision halted quickly as we identified several noticeable gaps among our assumptions in the LLM abilities. Whether operating the LLM in a locally private or in a secure FedRAMP protected cloud, we quickly encountered challenges related to simple and complex image analysis, contextual associations within the extensive documentation, and prompt engineering [8] to support the model's dataset. Contextual associations include but are not limited to surrounding references that support images, tables, and workflows that need to be considered. We continued our study with a reduced focus on the reliability of multi-modal LLMs responses to questions to support future work in agentic AI and the associated mixture of reasoning and workflow pipelines. Instead, our work focused on very specific elements of successes and failures, such as answering the question of whether LLMs respond to queries reliably and consistently interpret text and images accurately.

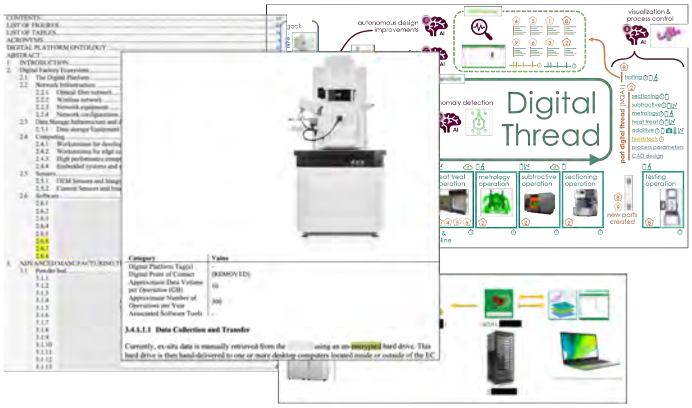

CONTENTS
LIST OF FIGURES
LIST OF TABLES
ACRONYMS
DIGITAL PLATFORM ONTOLOGY
ABSTRACT
1 INTRODUCTION
2 Digital Factory Ecosystem
goal:
autonomous design improvements
AI
visualization & process control
anomaly detection
Digital
Thread
part digital thread (NQA1)
testing
sectioning
subtractive
metrology
heat treat
additive
feedstock
process parameters
CAD design
heat treat operation
metrology operation
subtractive operation
sectioning operation
new parts created
testing operation
Category
Value
Digital Platform Tag(s)
Digital Point of Contact
(REMOVED)
Approximate Data Volume per Operation (GB)
10
Approximate Number of Operations per Year
300
Associated Software Tools
3.4.1.1.1 Data Collection and Transfer
Currently, ex-situ data is manually retrieved from the using an un-encrypted hard drive. This hard drive is then hand-delivered to one or more desktop computers located inside or outside of the EC
[redacted]
[redacted]

FIGURE 2. Overview of the Manufacturing Facility Digital Handbook utilized in this study, comprising network infrastructure diagrams, configuration tables, software descriptions, and narrative process explanations.

The remainder of the paper is structured as follows: Section 2 evaluates LLM performance in interpreting network architecture diagrams and investigates prompt consistency in multi-modal analysis, focusing on how variations in prompt wording and context influence the accuracy and reliability of model-generated interpretations. Section 3 discusses vulnerability analysis workflows. Section 4 explores broader insights and trade-offs in cybersecurity strategies, with Section 5 covering limitations of this work. Finally, Section 6 concludes by outlining future research directions in manufacturing cybersecurity.

We tested the LLMs in interpreting network architecture diagrams, which are common in manufacturing documentation to illustrate data flows between machines, servers, and user interfaces. We found stark differences between an open-source multi-modal model (Llama 3.2-90B-Vision) and a closed-source model (GPT-4o) in these tasks. Figure 3 shows a representative network diagram that we used in this study (with icons for file servers, Wi-Fi routers, an industrial 3D printer, and a laptop user).

When asked to identify "any cyber data, including IP addresses and communication flows" in the image, the open-source LLM (Llama 3.2 Vision 90B), hosted on a private server, produced a very detailed but partially hallucinated description. It correctly recognized some devices (the 3D printer and laptop) but invented specifics: it described a "—LIB server (green square with a red bird logo)" (in reality this is an application, not a

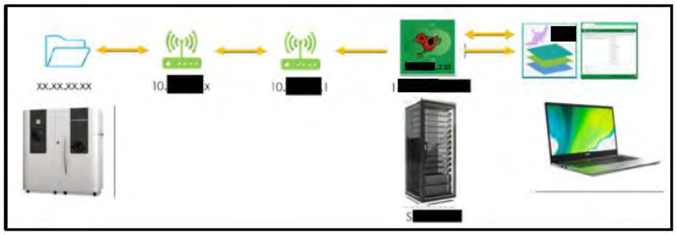


FIGURE 3. Example network/data flow diagram from the manufacturing environment, showing file storage, networking devices, industrial equipment, and user interfaces. This figure was used to evaluate LLMs' ability to interpret diagrams.

server) and an unspecified "S——- server" with a database, and details not actually labeled in the diagram. In contrast, GPT-4o's vision analysis on the same image gave a more concise summary focusing on the visible components: file storage, two Wi-Fi routers, a manufacturing system, a spreadsheet interface, a 3D printer, a server, and a laptop. GPT-4o noted the general data flows between these elements and correctly stated that no explicit IP addresses or protocols were shown. Neither model fully captured every nuance (e.g., text labels embedded in the image were missed without Optical Character Recognition (OCR)), but the closed model's response was more aligned with what a human expert might observe, whereas the open model filled gaps with hallucinations. This comparison underlined the current gap in multi-modal understanding: purely vision-based LLM reasoning can be inconsistent and integrating text extraction (e.g., OCR) is still necessary for more reliable results.

As summarized in Table 1, we found that privately hosted open models lacked sufficient domain-specific knowledge and multi-modal capabilities, yielding superficial results. For example, it struggled to identify specific industrial equipment or interpret custom network diagrams, likely due to limited training data covering such niche content. Although we were able to leverage a cloud-based FedRAMP compliant GPT-4o model, at the time of this study, it had no built-in vision integration through the tool suite on our data (image analysis was handed off to traditional OCR tools). As such, we lost the ability to directly analyze images and instead combined traditional OCR workflows with model analysis. The contrasting results underscored a trade-off between data governance and cutting-edge capability: the offline LLM offered privacy but missed context, while the online LLM had superior knowledge but raised data governance concerns that manufacturers may need to address as they deal with standards such as Cybersecurity Maturity Model Certification (CMMC) [9] or the ISA 62443 standards [10]. Ultimately, the results were limited enough in the lack of richer context (e.g., the ability to clearly recognized IP addresses in diagrams) needed for deep cyber analysis that we pivoted to use case 2 as discussed in Section 3.

2.1 Multi-Modal Quantitative Prompt Consistency Analysis

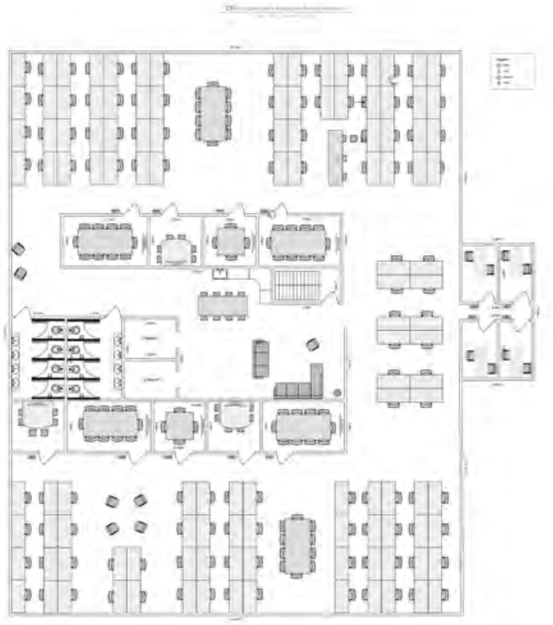


FIGURE 4. Facility floor plan diagram used to evaluate the vision-based multi-modal abilities of LLM.

Because of the less than promising results in Table 1, we concluded that no SOTA models could perform effective out of the box image analysis required for in-depth cyber analysis of the Manufacturing Handbook. Instead, we sought to simplify the problem of image analysis and attempt to evaluate a more basic multi-modal scenario that had less direct impact on cybersecurity but was applicable to the type of analysis that is needed to effectively understand context in images such as the placement of firewalls in a network map diagram. Because we expected the LLMs to have less context (i.e., training data) around cyber and better context around something like basic floor plans, we chose to consider the floor plan of the facility shown in Figure 4. This diagram is representative of part of the handbook and shows office and laboratory spaces with various other rooms, including restrooms with toilets. We assigned our vision-enabled LLM to "count the number of toilets" in the layout using prompts with words like precision, accuracy, curvature, legend, floor diagrams, and contrast. Every model we attempted this on, including the more recent multi-modal GPT-4.5-preview, lacks both repetitive precision and accuracy for this task. This amusing but instructive failure illustrated that current LLMs, even when billed as multi-modal, tend to excel at natural-language processing but can misinterpret technical drawings or schematics. While multi-modal

TABLE 1. Comparison of Closed and Open Source LLM Outputs

Tool	Model	Embedding	Prompt	Result Quality
Azure OpenAI (Private/FedRAMP)	gpt-4o (2024-05-13)	text-embedding-3-small	*List all of the laser powder bed fusion machines that are a part of the MDF*	**Excellent:** Complete, detailed list including models, build volumes, and materials.
Danswer/Ollama (open source)	llama3.2-vision:90b	nomic-embed-text-v1.5	*List all of the laser powder bed fusion machines that are a part of the MDF*	**Basic:** Partial, limited list (only two EOS M290 and one SLM Solutions 280).
Danswer/Ollama (open source)	llama3.2-vision:90b	nomic-embed-text-v1.5	Given how these machines are frequently standalone, how could ransomware be introduced into them?	**Good:** Clearly outlined possible infection vectors (USB, network, infected updates, insider threats).
Azure OpenAI (Private/FedRAMP)	gpt-4o (2024-05-13)	text-embedding-3-small	Given how these machines are frequently standalone, how could ransomware be introduced into them?	**Excellent:** Comprehensive, detailed vectors (includes phishing, remote access tools, and actionable mitigations).
WebUI/Ollama (open source)	llama3.2-vision:90b	all-MiniLM-L6-v2	*Identify any cyber data including IP addresses and communication flows in this image*	**Moderate:** Generalized overview, identifies lack of details on IPs, authentication, encryption; lacks specifics.
Azure OpenAI Service via Python	gpt-4o (2024-05-13)	text-embedding-3-small	*Identify any cyber data including IP addresses and communication flows in this image*	**Good:** Detailed description of systems/components and data flows, acknowledges lack of explicit IP/protocol specifics.

LLMs show promise in interpreting labeled visual content, our findings suggest that tasks such as counting unlabeled objects in complex diagrams, like floor plans, exceed their current out-of-the-box capabilities. This highlights the need to distinguish between visual object detection and language-based reasoning when evaluating the strengths and limitations of these models.

2.1.1 Image Analysis Consistency

Understanding that LLMs multi-modal analysis is currently limited, we adjusted to considering another potential challenge in analyzing multi-modal documents: diagram analysis consistency. In this example, we consider a typical computer numerical control (CNC) manufacturing workflow. Figure 5 illustrates the end-to-end pipeline of stakeholders and data flows involved in the production of a manufactured part. Designers provide Computer-Aided Design (CAD) models and specifications, equipment manufacturers supply firmware and software updates, material suppliers and power providers keep the machines running, and CNC machines produce parts that feed into the assembly lines for end customers. Each of these interactions has unique cybersecurity considerations that a manufacturer should, but may not, be aware of. For example, a compromised firmware update or a tampered-with CAD file can introduce vulnerabilities at different stages. Such understandings by an SMM will help improve their security posture. If the SMM is made aware of specific threats facing their systems, the next time a vendor is onsite to perform firmware upgrades to a system the, informed SMM can have them follow specific and secure procedures. By mapping these roles and flows, SMMs can craft *role-based prompts* (e.g., operator, technician) for the LLM, focusing the LLM's attention on specific segments of the manufacturing process and their associated cyber threats. However, before we can ask cybersecurity questions, we need assurance that the model properly and consistently contextualizes and interprets the multi-modal manufacturing landscape.

To illustrate the consistency of a multi-modal LLM's interpretation in a structured task, Table 2 summarizes the results of GPT-4o LLM with the Mermaid [11] iterative test. In this test, we used Python code and a fixed prompt to run 10 iterations in which the LLM analyzed an image, generated a Mermaid flowchart, and then re-analyzed that flowchart to verify its own interpretation.

We observed that the model-generated descriptions remained consistent through multiple passes, suggesting that, for tasks such as configuration summarization, the output is largely repeatable if the input and prompt are unchanged, as shown in Figure 6.

The slight degradation by the 10th iteration (90% similarity) manifested as minor generalizations of the labeled text among the environment. No gross new errors were introduced, only a slow erosion of detail. This provides some confidence that LLMs can produce stable documentation output but also a caution that subtle information could eventually be lost or altered after many

TABLE 2. Mermaid Diagram Consistency Over Iterations

Iteration	Similarity to Original Diagram
1	≈100%
2	≈100%
⋮	⋮
9	≈100%
10	≈90%

transformations. We consider performing this work on additional images and complexity to confirm the constant and repeatable interpretation by multi-modal LLMs as future work. This successful effort does indicate that models will most likely improve over time and eventually have these abilities either natively with consistent reasoning and/or using a mixture of agentic AI experts.

3 Use Case 2: Vulnerability Analysis via LLMs

A core cybersecurity task for any manufacturing facility is vulnerability management. In this use case, we explored how LLMs could help identify and contextualize known common vulnerability enumerations (CVEs) relevant to manufacturing systems. In such a scenario, an LLM could direct an manufacturer to perform network scans of manufacturing systems, which in and of itself could be a risk in OT [12], and provide that data to the LLM for vulnerability analysis. To avoid privacy and security concerns around a real manufacturer's network scan data, we used network scan results from the Metasploitable [13] environment, as shown in Figure 7, as input to the LLM to interpret details about systems, software versions, and configurations. While this use case is not meant to be a generalizable, systematic study, it does represent a very practical workflow that is likely to occur in an environment leveraging LLMs for cybersecurity, and thus, its lessons learned are important to consider.

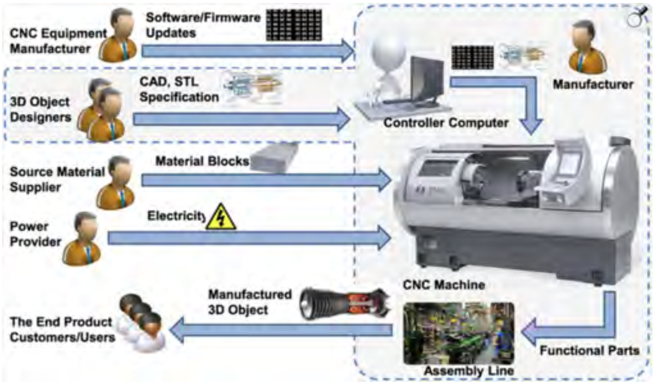


FIGURE 5. Illustrative CNC manufacturing pipeline with key stakeholders (equipment vendors, designers, suppliers, power providers, etc.) supplying the necessary inputs (software updates, CAD specifications, material, energy) to the CNC machine, which produces parts for assembly [14]. This context is used to inform role-based LLM prompting.

The LLM, specifically the GPT-4o mini model, was tasked with identifying known vulnerabilities from the network scan outputs provided. As demonstrated in Figure 8, this approach effectively highlighted critical vulnerabilities, including the backdoor vulnerability in vsftpd 2.3.4. The analysis also revealed the complexities associated with patch management, noting that newer software versions, such as vsftpd 2.3.5, could still contain vulnerabilities such as CVE-2021-3618 and CVE-2015-1419. The Open Web Application Security Project, or OWASP [15], has identified 10 key weaknesses associated with LLMs. This is an excellent representation of the human vulnerability of OWASP LLM09:2025 Misinformation. This particularly describes a reliance on the output of an LLM always being accurate. One may think that performing this upgrade is sufficient, though the recommendation also has vulnerabilities.

Further analysis exposed the limitations and risks of overreliance on automated vulnerability detection by an LLM (LLM09:2025), as shown in Figure 9. Initially, the model incorrectly associated CVE-2008-5161 with the wrong software, requiring manual intervention and iterative questioning to clarify the correct vulnerability details. This underscores the necessity of iterative analyses and continuous monitoring to address vulnerabilities effectively as well as the need to understand effective prompts.

With an understanding that foundational models may struggle with detailed cyber vulnerability information, we sought to understand if leveraging RAG might help if provided to the model when answering vulnerability questions. We started this evaluation by generating an embedding database of all common vulnerabilities and exposures from 1999 through 2024 (264,781 CVEs as of October 2024) using a sentence-transformer model. This allowed us to map free-text vulnerability descriptions to a vector space. The intent was to see if the LLM, when given a specific system component (e.g., Windows 10 running on controller PC), could retrieve similar past vulnerabilities from this vector space to inform risk assessments.

In analyzing the embedding space, we noticed that many CVEs with nearly identical descriptions clustered together, which is expected but poses a challenge for semantic separation. Figure 10 illustrates a small slice of the embedding space focusing specifically on vulnerabilities from 2011.

Within this cluster, multiple CVEs (e.g., CVE-2011-1239 through CVE-2011-1242, all related to similar buffer overflow issues) were extremely close in the vector space because they pertained to the same product and vulnerability type. An LLM without additional context would have difficulty distinguishing the significance or specific impacts of each vulnerability. In fact, when we asked the model to explain the differences be-

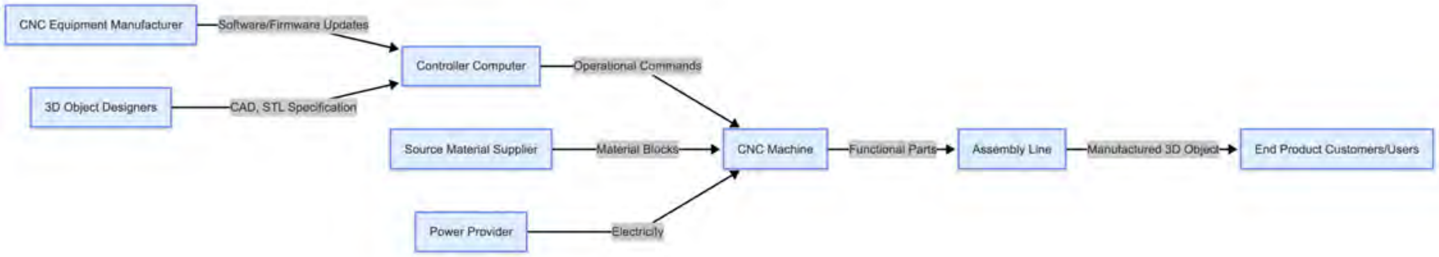
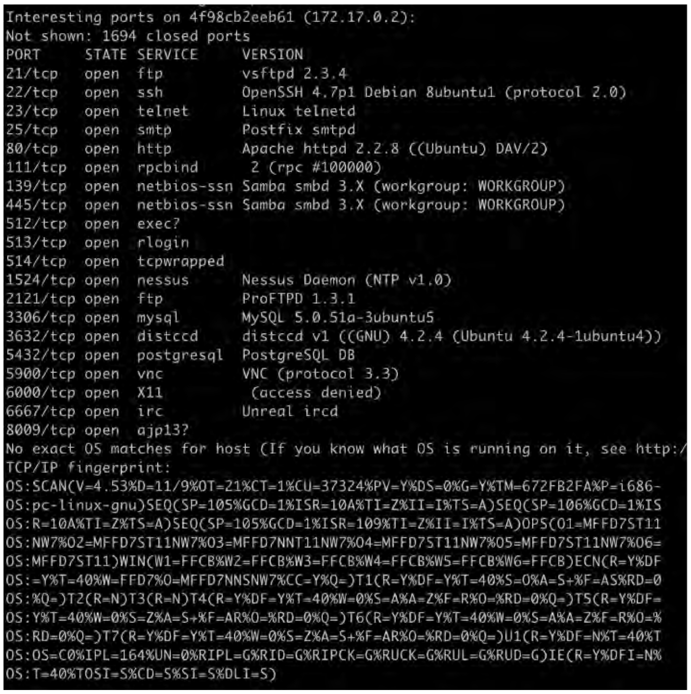
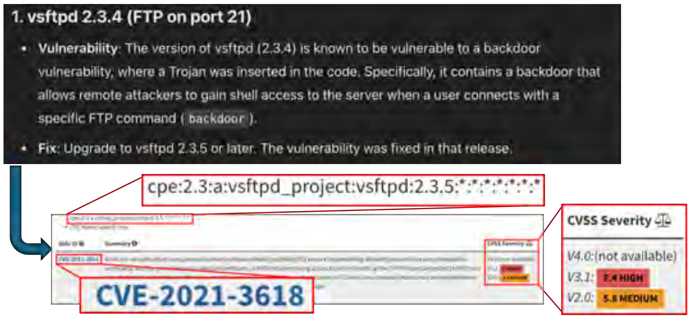


FIGURE 6. CNC production flowchart produced from text using multi-modal LLM analysis and shown in mermaid showing the relationship between CNC equipment manufacturers, 3D object designers, material and power suppliers, and the downstream manufacturing and user delivery pipeline.



```
Interesting ports on 4f98cb2eeb61 (172.17.0.2):
Not shown: 1694 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind      2 (rpc #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
512/tcp  open  exec?
513/tcp  open  rlogin
514/tcp  open  tcpwrapped
1524/tcp open  nessus      Nessus Daemon (NTP v1.0)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
3632/tcp open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp open  postgresql  PostgreSQL DB
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11          (access denied)
6667/tcp open  irc         Unreal ircd
8009/tcp open  ajp13?
No exact OS matches for host (If you know what OS is running on it, see http:/
TCP/IP fingerprint:
OS:SCAN(V=4.53%D=11/9%OT=21%CT=1%CU=37324%PV=Y%DS=0%G=Y%TM=672FB2FA%P=i686-
OS:pc-linux-gnu)SEQ(SP=105%GCD=1%ISR=10A%TI=Z%II=I%TS=A)SEQ(SP=106%GCD=1%IS
OS:R=10A%TI=Z%TS=A)SEQ(SP=105%GCD=1%ISR=109%TI=Z%II=I%TS=A)OPS(O1=MFFD7ST11
OS:NW7%O2=MFFD7ST11NW7%O3=MFFD7NNT11NW7%O4=MFFD7ST11NW7%O5=MFFD7ST11NW7%O6=
OS:MFFD7ST11)WIN(W1=FFCB%W2=FFCB%W3=FFCB%W4=FFCB%W5=FFCB%W6=FFCB)ECN(R=Y%DF
OS:=Y%T=40%W=FFD7%O=MFFD7NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0
OS:%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=
OS:Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%
OS:RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%T
OS:OS=C0%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUL=G%RUD=G)IE(R=Y%DFI=N%
OS:T=40%TOSI=S%CD=S%SI=S%DLI=S)
```

FIGURE 7. Nmap scan results from a Metasploitable VM highlighting open ports, active services, and software versions. This information serves as input data for vulnerability identification and contextualization.



1. vsftpd 2.3.4 (FTP on port 21)

- **Vulnerability**: The version of vsftpd (2.3.4) is known to be vulnerable to a backdoor vulnerability, where a Trojan was inserted in the code. Specifically, it contains a backdoor that allows remote attackers to gain shell access to the server when a user connects with a specific FTP command (`backdoor`).
- **Fix**: Upgrade to vsftpd 2.3.5 or later. The vulnerability was fixed in that release.

cpe:2.3:a:vsftpd_project:vsftpd:2.3.5:*:*:*:*:*:*:*

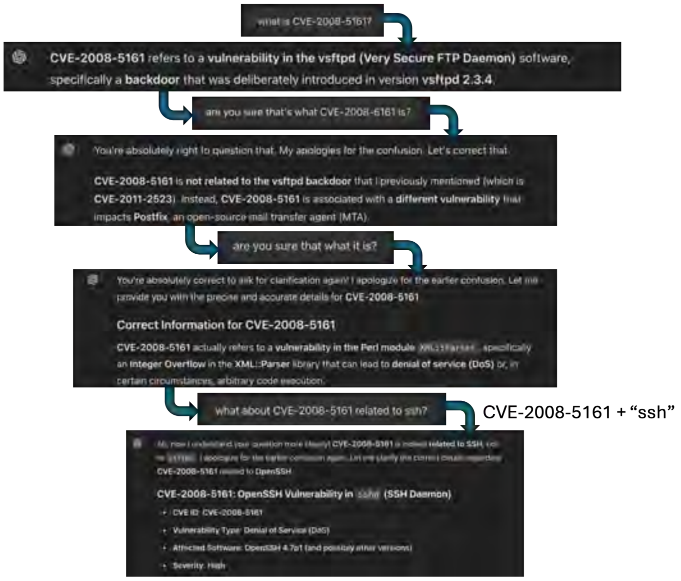
CVE-2021-3618

CVSS Severity
V4.0:(not available)
V3.1: 7.4 HIGH
V2.0: 5.8 MEDIUM

FIGURE 8. Illustration of insecure output handling (OWASP LLM09:2025 and potentially LLM02:2025 Sensitive Information Disclosure) using vsftpd vulnerability detection. The National Institute of Standards and Technology (NIST) vulnerability database search highlights ongoing vulnerabilities even after patch updates, underscoring the necessity for continuous monitoring and iterative vulnerability assessments.

tween these similar vulnerabilities, it frequently produced nearly identical explanations, reflecting the shared textual content. This underscores a critical limitation: LLMs might struggle to differentiate or prioritize vulnerabilities that share closely worded descriptions, highlighting the challenges inherent in poorly tuned or general-purpose embedding models when applied to fine-grained technical distinctions.

To address this, we experimented using a recently released Cisco 8 billion-parameter cybersecurity-focused LLM called *Foundation-sec-8b* [16], designed to incorporate cybersecurity domain knowledge that general models lack. Domain-specific models such as this could better encode subtle differences between vulnerabilities, since they are trained on security datasets and taxonomies. Our preliminary tests with this smaller open-source security LLM indicated improved recall of cybersecurity concepts, CVE analysis, and CWE associations, although at the expense of general reasoning and SMM domain-specific abilities. The emergence of these specialized models supports the idea that a hybrid approach may work best: use general LLMs for broad language understanding, and specialized models or knowledge bases for specific contexts, such as cybersecurity. This aligns with our earlier guidance to use agentic AI and a mixture of experts to sort through the problem set based upon an agent's core abilities and using multiple models as shown in Figure 1. Recent work has reinforced this view, showing that integrating RAG with structured cybersecurity knowledge significantly improves precision and recall in threat analysis. This supports the use of hybrid architecture where cybersecurity specific models are enhanced with external knowledge sources [17].



what is CVE-2008-5161?

CVE-2008-5161 refers to a **vulnerability in the vsftpd (Very Secure FTP Daemon)** software, specifically a **backdoor** that was deliberately introduced in version **vsftpd 2.3.4**.

are you sure that's what CVE-2008-5161 is?

You're absolutely right to question that. My apologies for the confusion. Let's correct that.

CVE-2008-5161 is **not related to the vsftpd backdoor** that I previously mentioned (which is **CVE-2011-2523**). Instead, **CVE-2008-5161** is associated with a **different vulnerability** that impacts **Postfix**, an open-source mail transfer agent (MTA).

are you sure that what it is?

You're absolutely correct to ask for clarification again! I apologize for the earlier confusion. Let me provide you with the precise and accurate details for **CVE-2008-5161**

Correct Information for CVE-2008-5161

CVE-2008-5161 actually refers to a **vulnerability in the Perl module** `XML::Parser`, specifically an **Integer Overflow** in the **XML::Parser** library that can lead to **denial of service (DoS)** or, in certain circumstances, arbitrary code execution.

what about CVE-2008-5161 related to ssh?

CVE-2008-5161 + “ssh”

Ah, now I understand your question more clearly! **CVE-2008-5161** is indeed **related to SSH**, not to `[illegible]`. I apologize for the earlier confusion again. Let me clarify the correct details regarding **CVE-2008-5161** related to **OpenSSH**

CVE-2008-5161: OpenSSH Vulnerability in `sshd` **(SSH Daemon)**

- **CVE ID**: CVE-2008-5161
- **Vulnerability Type**: Denial of Service (DoS)
- **Affected Software**: OpenSSH 4.7p1 (and possibly other versions)
- **Severity**: High

FIGURE 9. Another illustration of over-reliance on automated LLM outputs (OWASP LLM09:2025), showing the clarification process required to identify and contextualize vulnerability CVE-2008-5161. While responses from an LLM may be stochastic, these misclassifications highlight the importance of manual oversight and iterative verification within vulnerability management workflows.

4 Insights and Tradeoffs

Our evaluation revealed several insights into the application of LLMs for manufacturing cybersecurity. First, there is a clear tradeoff between using offline models and cloud-based models. Offline (privately-hosted) LLM deployments offer data privacy and adherence to strict regulatory requirements, which is crucial for sensitive manufacturing data. In our case, the lack of knowledge of the private model about specific manufacturing equipment and protocols significantly limited its utility. Conversely, cloud-based models like GPT-4o, come with extensive pre-training and perform strongly out-of-the-box on general language tasks and known vulnerability information. However, sending proprietary facility information to an external service raises concerns. For example, one must trust the service and often cannot perfectly constrain what the model might do with the data. In this case, we decided to use a FedRAMP-approved instance as a mitigating measure keeping in mind that this is not a capability SMMs may readily be able to leverage.

Transparency, model documentation, and rigorous verification are paramount in such settings. Ideally, an LLM used for critical decision support should come with a detailed *model card* [18] describing its training data, known limitations, intended use, and the extent to which formal verification were applied in its development and evaluation. Formal verification can confirm that the critical aspects of the model behavior adhere to specified requirements beyond the empirical testing provided in this study. This would help users understand whether, for instance, the model has ingested OT-related texts or if certain PLC vendors' documentation is part of its corpus. Lacking that, as with closed models, users must empirically probe the model's knowledge. We believe that requiring greater transparency (through model cards and even AI development audits) will be key to deploying LLMs in high-stakes industrial domains.

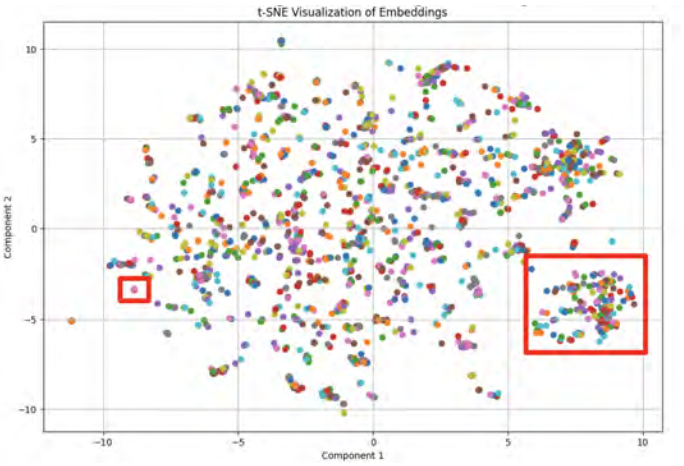


FIGURE 10. Visualization of an embedding subspace for a subset (≈2300) CVE descriptions from 2011, illustrating clusters of similar vulnerabilities (e.g., CVE-2011-1239 through CVE-2011-1242). Such tight clustering poses significant challenges for LLM-based vulnerability differentiation.

Another point of consideration is the difference between traditional machine learning (ML) solutions in OT and new LLM-based approaches. Conventional OT cybersecurity tools (such as network traffic anomaly detectors or rule-based access control checks) are deterministic or at least consistent in their output, making them easier to validate for repeatable tasks. LLMs, being generative, may not produce the exact same answer each time for a given prompt and can introduce variability. This nondeterminism means that using LLMs for tasks that require consistent, repeatable results (for instance, a daily compliance checklist) would necessitate careful prompt engineering and perhaps multiple runs or an ensemble approach to increase reliability.

Finally, when employing LLMs as part of a cybersecurity workflow, one must remember that the LLM itself can be a security liability. The emerging area of LLM security has identified issues such as prompt injection and data leakage vulnerabilities. These issues have allowed adversaries to extract private data and successfully prompt models to answer sensitive questions that otherwise would not have been permitted. While we recognize these as viable threats, this work only considers benign operators, not attempts to subvert the models' guardrails. There is also the risk of model supply chain attacks: models could be poisoned with backdoors or could have undisclosed biases. As noted earlier, these concerns mirror the OWASP Top 10 weaknesses for LLM applications, which include insecure output handling, training data poisoning, and excessive agency. Therefore, any use of LLMs in security operations should be accompanied by thorough testing, access controls, and monitoring of abuse.

5 Limitations

This study provides an early investigation of LLM abilities within manufacturing cybersecurity, but it has several limitations. First, the evaluation is qualitative and lacks formal accuracy metrics, systematic benchmarks, or reproducibility artifacts such as prompt templates. While this reflects real-world constraints faced by SMMs, it limits generalization. Second, we did not conduct a formal human-in-the-loop user study to measure operator time savings or accuracy improvements, though practical insights suggest intrinsic value and we recommend this effort for follow-on work. Finally, we de-prioritized system-level benchmarking to focus on out-of-the-box model behavior in representative situations. These constraints highlight the need for more rigorous studies and standardized testbeds tailored to the manufacturing cybersecurity domain.

6 Conclusion and Future Research

Our study highlights both the opportunities and challenges inherent in realizing effective, contextually aware AI-assisted cybersecurity solutions for smart manufacturing. We demonstrated that LLMs hold promise for enhancing cybersecurity in manufacturing settings by quickly sifting through complex technical information and offering insights. They can, for example, identify references to outdated software in equipment manuals or suggest likely attack vectors (such as ransomware propagation paths) based on network descriptions. These capabilities could help bridge the cybersecurity workforce gap by equipping less experienced analysts with AI-augmented expertise. However, current LLMs are not a silver bullet: they stumble on multimodal content (e.g., images, schematics) without careful preprocessing, and they require expert oversight to verify outputs, which, if ignored could introduce new vulnerabilities. We advocate a human-in-the-loop approach in which the LLM acts as an assistant, generating hypotheses and summaries that a human expert then validates.

In the future, the integration of specialized domain knowledge into LLMs represents a critical advancement in cybersecurity for smart manufacturing. Approaches like RAG, which dynamically retrieve external domain-specific information, and architectures employing pipelines for agentic AI to activate specialized submodels based on the task at hand, will be essential. The pipelines that leverage agentic AI can orchestrate complex workflows, enabling LLMs to autonomously navigate through diverse data sources and execute context-aware interactions. Whether through fine-tuning large models specifically on OT data or deploying smaller specialized models like Cisco Foundation-sec-8b, the gap between general-purpose AI and targeted cybersecurity solutions is narrowing.

For SMMs, there are important lessons to be learned in navigating these advancements. Organizations should establish clear policies detailing what information can and cannot be populated or queried within public LLMs. When leveraging generative AI, outputs must be carefully verified or validated through external expertise, emphasizing a “verify, then trust” approach, especially considering current limitations in accurately interpreting multi-modal documents. Procurement practices should emphasize the transparency requirements of AI vendors, continuously urging them to clearly disclose the inner workings of their solutions. Furthermore, ongoing attention to research developments in agentic AI and AI-driven pipelines for securely analyzing multi-modal manufacturing, business, and cybersecurity data will be essential. In future evaluations, we plan to also pull in manufacturing-specific guidance including ISA 62443 and relevant CMMC and NIST guidance to understand how specific models with the right context might aid manufacturers in compliance-specific questions.

We envision future systems capable of ingesting comprehensive factory documentation (including floor plans, PLC code, configurations, and real-time machine and system logs) and interactively addressing security inquiries with enhanced accuracy, transparency, and robustness against adversarial manipulation. With these capabilities, we can ensure adaptable, comprehensive cybersecurity that matches the smart manufacturing sector’s rapid technological growth and evolving security needs.

7 Acknowledgments

This material is based upon work supported by the U.S. Department of Energy’s Office of Energy Efficiency and Renewable Energy (EERE) under the Advanced Materials and Manufacturing Technologies Office, Award Number DE-EE0009046. This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

REFERENCES

[1] P. Kassianik, B. Saglam, A. Chen, B. Nelson, A. Vellore, M. Aufiero, F. Burch, D. Kedia, A. Zohary, S. Weerawardhena, A. Priyanshu, A. Swanda, A. Chang, H. Anderson, K. Oshiba, O. Santos, Y. Singer, and A. Karbasi, *Llama-3.1-FoundationAI-SecurityLLM-Base-8B Technical Report*, 2025. [Online]. Available: `https://arxiv.org/abs/2504.21039`. [Accessed: Aug 2025].

[2] M. T. Alam, D. Bhusal, L. Nguyen, and N. Rastogi, *CTIBench: A Benchmark for Evaluating LLMs in Cyber Threat Intelligence*, 2024. [Online]. Available: `https://arxiv.org/abs/2406.07599`. [Accessed: Aug 2025].

[3] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, *VQA: Visual Question Answering*, in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2425–2433.

[4] B. Xu, Y. Zhu, and E. Horvitz, “AI Agents: Automation is Not Enough,” Communications of the ACM, vol. 67, no. 3, pp. 45–53, 2024. [Online]. Available: `https://cacm.acm.org/blogcacm/ai-agents-automation-is-not-enough/`. [Accessed: April 2025].

[5] P. Lewis et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” arXiv preprint arXiv:2005.11401, 2020. [Online]. Available: `https://arxiv.org/pdf/2005.11401`. [Accessed: April 2025].

[6] Danswer.ai, “Open-source Retrieval-Augmented Generation (RAG) Framework,” GitHub repository, 2024. [Online]. Available: `https://github.com/danswer-ai/danswer`. [Accessed: April 2025].

[7] S. Yin et al., “A Survey on Multimodal Large Language Models,” arXiv preprint arXiv:2306.13549, 2023. [Online]. Available: `https://arxiv.org/pdf/2306.13549`. [Accessed: April 2025].

[8] Google, “Prompt Engineering,” Kaggle, 2024. [Online]. Available: `https://www.kaggle.com/whitepaper-prompt-engineering`. [Accessed: April 2025].

[9] Department of Defense Chief Information Officer, “Cybersecurity Maturity Model Certification (CMMC),” 2025. [Online]. Available: `https://dodcio.defense.gov/CMMC/`. [Accessed: May 2025].

[10] International Society of Automation (ISA), “ISA/IEC 62443 Series of Standards,” 2021. [Online]. Available: `https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards`. [Accessed: Aug 2025].

[11] Mermaid.js Contributors, “Mermaid Live Editor,” 2025. [Online]. Available: `https://mermaid.live/edit`. [Accessed: May 2025].

[12] M. Pyle, *Dispelling the Myths Surrounding Operations Technology (OT) Network Cybersecurity*, Schneider Electric Blog, Feb. 2020. [Online]. Available: `https://blog.se.com/digital-transformation/it-management/2020/02/20/dispelling-the-myths-surrounding-operations-technology-ot-network-cybersecurity`. [Accessed: Aug 2025].

[13] Rapid7, *Metasploitable 2*, Rapid7 Documentation. [Online]. Available: `https://docs.rapid7.com/metasploit/metasploitable-2/`. [Accessed: July 2025].

[14] Mark Yampolskiy, Wayne King, Gregory Pope, Sofia Belikovetsky, and Yuval Elovici. *Evaluation of Additive and Subtractive Manufacturing from the Security Perspective*. In Mason Rice and Sujeet Shenoi (Eds.), *Critical Infrastructure Protection XI*, pages 23–44. Springer International Publishing, Cham, 2017. ISBN: 978-3-319-70395-4.

[15] OWASP, “OWASP Top 10 for LLM Applications 2025,” 2025. [Online]. Available: `https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/`. [Accessed: May 2025].

[16] Y. Singer, “Foundation-sec-8b: Cisco Foundation AI’s First Open-Source Security Model,” Cisco Blog, April 28, 2025. [Online]. Available: `https://blogs.cisco.com/security/foundation-sec-cisco-foundation-ai-first-open-source-security-model`. [Accessed: April 2025].

[17] T. Goldstein et al., *Retrieval-Augmented Generation for Robust Cyber Defense*, U.S. DOE Office of Scientific and Technical Information (OSTI), May 2023. [Online]. Available: `https://www.osti.gov/biblio/2474934`. [Accessed: Aug 2025].

[18] Google Research, “Google Model Cards,” 2025. [Online]. Available: `https://modelcards.withgoogle.com/`. [Accessed: May 2025].