Security Assessment of an LBP16-Protocol-Based Computer Numerical Control Machine

Yahya Forihat[1], Curtis Taylor[2], Rima Asmar Awad[2], and Irfan Ahmed[1(✉)]

[1] Virginia Commonwealth University, Richmond, VA, USA
iahmed3@vcu.edu
[2] Oak Ridge National Laboratory, Oak Ridge, TN, USA

Abstract. Subtractive manufacturing systems, specifically, computer numerical control machines, have revolutionized the manufacturing industry. Computer numerical control machining is the preferred method for producing finished parts due to its efficiency, speed and suitability for high-volume production. Securing computer numerical control machines is a priority. Compromises or disruptions of these machines can result in significant downtime, loss of productivity and financial loss.

This study examines the vulnerabilities and risks associated with computer numerical control machines, in particular, systems utilizing the LBP16 protocol for controller-machine communications. The study reveals that an adversary can execute cyber-physical attacks such as sabotage and denial of service. The potential security threats emphasize the importance of implementing robust security measures to mitigate the cyber risks to computer numerical control machines.

Keywords: Subtractive Manufacturing · Computer Numerical Control Machine · Cyber-Physical Attacks · Sabotage · Denial of Service

1 Introduction

Subtractive manufacturing creates parts by selectively removing material from workpieces through milling, lathing and drilling techniques. It is versatile in that it accommodates diverse materials and design requirements, produces superior surface finishes and provides cost-effective solutions for high-volume production. The integration of computer numerical control (CNC) technology, an exemplar of subtractive manufacturing, has dramatically improved the precision, consistency, repeatability, efficiency and reliability of machining processes. In fact, CNC machines are indispensable in intelligent manufacturing, especially in precision-critical sectors such as aerospace, healthcare, automotive and defense [15].

CNC has played a pivotal role in shaping new industrial models that adhere to the Industry 4.0 framework. A significant aspect of this progress is the ability of CNC machines to connect to networks. While networking enhances productivity, it also introduces security and privacy risks [3].

Published by Springer Nature Switzerland AG 2025
J. Staggs and S. Shenoi (Eds.): ICCIP 2024, IFIP AICT 725, pp. 65–84, 2025.
https://doi.org/10.1007/978-3-031-81888-2_4

Cyber attacks that target physical systems are on the rise [10]. A prominent example is Stuxnet, which caused significant damage to uranium hexafluoride centrifuges in Iranian nuclear facilities [2,6,12,13,24,25]. In the past, manufacturing systems were isolated, shielding them from remote attacks. However, the ubiquitous deployment of cyber manufacturing systems has made the manufacturing industry highly vulnerable to cyber-physical accidents and attacks [17]. Many cyber-physical attacks are executed by insiders such as disgruntled employees with authorized access to computers and networks [5].

Modern manufacturing relies on a complex mix of digital and physical technologies, including control systems, machining equipment (lathes and mills), assembly tools (pneumatics, welders and robots), post-processing machinery (kilns and paint lines) and inspection devices (digital cameras and measuring machines) [11]. CNC manufacturing systems are encountering industrial viruses and network attacks that have brought cyber security to the forefront [7].

While a large body of research has focused on cyber security vulnerabilities in additive manufacturing systems [1,4,8,9,14,18,19,22,23], studies of subtractive manufacturing systems are notable for their absence. The work described in this chapter fills the gap by shedding light on CNC machine vulnerabilities and providing insights and solutions. In particular, it presents an analysis of vulnerabilities in CNC machines that utilize the LBP16 protocol for controller-machine communications. The vulnerabilities render the systems susceptible to cyber-physical attacks such as sabotage and denial of service. The results, which have been communicated to the affected vendors, are discussed in this chapter along with mitigation strategies that enhance CNC machine security.

2 Background and Related Work

This section provides an overview of CNC manufacturing and research related to manufacturing system security.

2.1 CNC Manufacturing

In CNC manufacturing, computer software directs the movement of machinery and tools in a factory setting. This enables manufacturers to enhance efficiency by reducing production time, minimizing waste and eliminating potential human errors. Workpiece material is gradually removed by controlled machining until the finished part with the desired shape and dimensions is produced.

Figure 1 provides an overview of the CNC machining process. It begins with the creation of a 2D or 3D computer-aided design (CAD) model that specifies the part shape and dimensions. The CAD model is input to computer-aided manufacturing (CAM) software that creates the tool paths and establishes the machining parameters. The CAM software exports the plans as G-code to a CNC machine controller that dispatches detailed instructions to the CNC machine. The CNC machine follows the instructions to precisely cut, drill and mill the material, producing the final product with high precision and consistency.

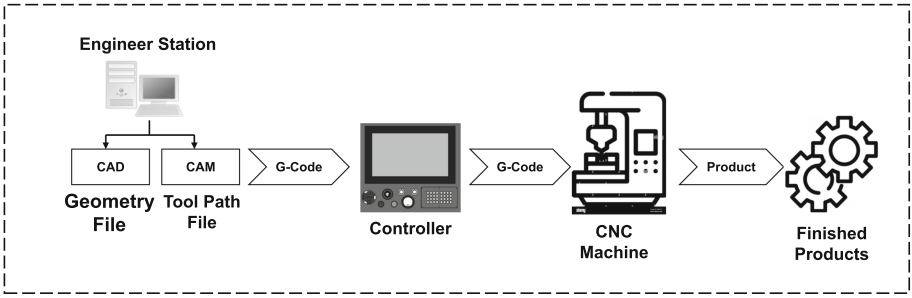


Fig. 1. CNC machining process

Commonly-used CAD software includes AutoCAD, SolidWorks and Fusion 360. Popular CAM software includes Mastercam, Fusion 360 CAM and Siemens NX CAM. CAM software generates tool paths and machining instructions based on the input CAD model. The software also enables the optimization of cutting strategies, selection of appropriate tools and simulations of machining processes to identify potential problems. The tool paths and machining instructions produced by CAM software are typically specified in G-code, a program that guides CNC machine operation. The G-code commands and instructions are processed by a CNC machine to dictate its movements, speeds, tooling operations and other parameters. Each line of code represents a specific action or function, such as moving the machine to a particular location, controlling the tooling operation or activating an auxiliary function. G-code enables the precise, automated control of CNC machines, allowing for the creation of complex parts accurately and efficiently.

LinuxCNC is an open-source, real-time operating system engineered to serve as a CNC controller. It is renowned for converting a standard personal computer into a highly adaptable CNC controller that maintains precise timing control over low-level hardware such as stepper motors. While it is not a complete tool for CNC machining, LinuxCNC offers considerable flexibility, ranging from preparing G-code instructions using CAM software to selecting appropriate stepper drivers and managing everything in between.

CNC machines employ G-code instructions for machining operations and applications. A CNC mill employs computer control to cut materials. It utilizes G-code instructions to precisely control the movements of the mill spindle in various directions and patterns. Following the instructions, a CNC mill can accurately shape and cut materials according to the programmed specifications. This level of automation and precision has contributed to the wide use of CNC mills in prototyping and manufacturing.

A CNC lathe secures a workpiece on a spindle and rotates it while various cutting tools are applied to shape the material. The movements and operations of the lathe, including spindle rotation, tool positioning, feed rates and depths of cuts, are controlled using G-code.

A CNC router is designed for cutting, shaping and engraving wood, plastic and foam. It utilizes different cutting tools and is used for precision woodworking, sign-making and 3D carving.

A CNC plasma cutter employs a high-velocity ionized gas (plasma) jet to cut through electrically-conductive materials. CNC plasma cutters are commonly used in metal fabrication and industrial applications.

A CNC laser cutter uses a laser beam to precisely cut or engrave materials such as metal, wood, acrylic and fabric. CNC laser cutters are renowned for their accuracy and speed.

2.2 Related Work

Despite their importance in modern manufacturing processes, CNC machines have not been the subject of much security research. The literature primarily concentrates on the broader aspects of industrial control system security, with only a few studies delving into the vulnerabilities and threats associated with CNC machines.

DeSmit et al. [10] have introduced a novel cyber-physical vulnerability assessment method for manufacturing systems. Their two-step method begins by creating intersection maps that represent the diverse processes in a manufacturing setting; potential vulnerabilities are identified by examining how different entities interact. Next, decision tree analysis is employed to evaluate the impact of each exposure at each intersection node. DeSmit and colleagues demonstrated the utility of the assessment method for handling the growing complexity and frequency of attacks targeting manufacturing systems.

Tu et al. [20] have proposed a CNC machine security framework based on Trusted Computing 3.0 technology. The framework automatically recognizes self-instructions and non-self-instructions to protect a CNC/industrial control system (ICS) network from interference. They also introduced a trusted communications monitoring and control scheme that supports real-time monitoring, control and encryption verification of CNC-ICS instructions.

Wu et al. [21] have focused on detecting cyber-physical attacks in computerized manufacturing systems (CMSs) using physical data and machine-learning approaches. They developed and integrated machine learning algorithms to identify and mitigate potential attacks in two manufacturing processes, 3D printing and CNC milling. A vision system was used as the physical data source for 3D-printing and the method for detecting malicious defects yielded 96.1% accuracy. Acoustic signals were used in the case of CNC milling and the random forest algorithm achieved 91.1% accuracy in identifying cyber-physical attacks during manufacturing operations.

Chen et al. [7] have proposed a security protection scheme that incorporates mechanisms such as data encryption, identity authentication, digital signatures, access control, secure communications and key management. Their scheme focuses on protecting CNC system terminal equipment, safeguarding data and securing the network infrastructure.

Li et al. [15] have explored innovative methods for detecting cyber attacks on cyber-physical manufacturing systems in real time. Their MPI-CNC prototype system developed for FANUC CNC machine tools automatically analyzes numerical control programs to detect anomalies by extracting machining process invariants and key parameter rules. The system effectively identifies cyber attack behaviors by comparing real-time data with predefined rules, ensuring that the normal operations of CNC systems are not disrupted. MPI-CNC was rigorously evaluated in several real-world machining scenarios. It displayed superior adaptability and accuracy at detecting attacks compared with other state-of-the-art methods.

In a study related to the work described in this chapter, Balduzzi et al. [3] investigated the security and privacy aspects of CNC machines. They conducted in-depth empirical analyses of modern CNC controllers from several vendors and identified their vulnerabilities. The threat modeling work identified five attack classes: compromise, damage, denial of service, hijacking and theft. The research results have been communicated to the affected vendors along with proposed mitigation strategies.

The research described in this chapter analyzes the communications between controllers and CNC machines. This is significant because a key threat involves adversaries leveraging network access to intercept traffic and gain insights into the communications, enabling them to orchestrate sophisticated attacks. The research has uncovered security weaknesses in CNC machines that employ the LBP16 communications protocol.

3 LBP16 Protocol Attack Methodology

This work focuses on attacks on the LBP16 protocol used by CNC machines for controller-machine communications. This section describes the LBP16 protocol, attack methodology and adversary model.

3.1 LBP16 Protocol

LBP16 is a simple remote register access protocol designed for swift and efficient interactions with hardware registers over Ethernet [16]. The protocol plays a pivotal role in real-time motion control applications. LBP16 can access eight distinct memory spaces, enabling it to handle a broad range of data. Each memory space is linked to an exclusive read-only info area. The first entry in the info area is a cookie, which is used to confirm that access is proper and authorized; this enhances system security.

LBP16 Command Structure. LBP16 commands have a standard structure that enables precise communications with hardware registers. Table 1 shows the LBP16 protocol command structure. Each command is 16 bits long and has several crucial bits.

Table 1. LBP16 protocol command structure

Bit(s)	Description
0 to 6	`N`: Transfer count (number of transfers)
7	`I`: Increment address (1 = yes, 0 = no)
8, 9	`S`: Transfer element size specifier (00b = 8 bits, 01b = 16 bits, 10b = 32 bits, 11b = 64 bits)
10, 11, 12	`M`: Memory space specifier (000b through 111b)
13	`C`: Memory space indicator (0 = memory, 1 = info area)
14	`A`: Includes address (1 = yes, 0 = no)
15	`W`: Write bit (1 = write, 0 = read)

Info Area. The info area is an essential component of each of the eight possible memory spaces [16]. Three entries in the info area play key roles:

- **COOKIE (Address 0):** This first entry serves as a verification mechanism, ensuring proper and authorized access to the memory space. The cookie is represented as `0X5A0N`, where `N` corresponds to the address space (ranging from 0 to 7).
- **MEMSIZES (Address 2):** This entry provides information about the memory sizes in the associated memory space.
- **MEMRANGES (Address 4):** This entry holds data related to the memory ranges of the associated memory space.

Each MEMSIZES and MEMRANGES entry is characterized by specific bits that denote whether the memory space is writeable, the memory space type (register, memory, EEPROM or flash) and the allowed data access size.

Table 2 shows the hexadecimal command structure of the LBP16 protocol info area. `NN` indicates the count/increment field whereas `LL` and `HH` refer to the lower and higher bytes of the address.

Table 3 shows the reading and writing commands for different LBP16 memory spaces. The same notation is used to clarify the structure and functionality of the commands.

3.2 Attack Methodology

A vulnerability was identified in the LBP16 protocol, a crucial part of the communications process. While the protocol effectively transmits data between a CNC controller and a CNC machine, it has inadequate mechanisms for authenticating entities that attempt to access and control CNC machines. An adversary exploiting the vulnerability can access the control system of the CNC machine, alter the operational parameters or send harmful commands directly to the machine. The negative impacts include the production of defective parts and physical damage to the CNC machine.

Table 2. LBP16 command structure

Info Memory Space	Command	Description
Ispace 0: Hostmot2	NN61LLHH	Read with specified address
	NN21	Read without specified address
Ispace 1: Ethernet Chip	NN65LLHH	Read with specified address
	NN25	Read without specified address
Ispace 2: Ethernet EEPROM	NN69LLHH	Read with specified address
	NN29	Read without specified address
Ispace 3: FPGA Flash EEPROM	NN6DLLHH	Read with specified address
	NN2D	Read without specified address
Ispace 6: LBP16 R/W	NN79LLHH	Read with specified address
	NN39	Read without specified address
Ispace 7: LBP16 R/O	NN7DLLHH	Read with specified address
	NN3D	Read without specified address

Table 3. Command structures for memory spaces

Memory Space	Reading Command	Writing Command
Space 0: HostMot2	NN42LLHH	NNC2LLHH
	NN02	NN82
Space 1: Ethernet Chip	NN45LLHH	NNC5LLHH
	NN05	NN85
Space 2: Ethernet EEPROM	NN49LLHH	NNC9LLHH
		D9 Enable EEPROM area writes
	NN09	NN89
Space 3: FPGA Flash EEPROM	NN4ELLHH	NNCELLHH
	NN0E	NN8E
Space 4: LBP Timer/Utility Area	NN51LLHH	NND1LLHH
	NN11	NN91
Space 6: LBP Status/Control Area	NN59LLHH	NND9LLHH
	NN19	NN99
Space 7: LBP Read-Only Area	NN5DLLHH	Not applicable
	NN1D	

Several actions were executed based on the identified vulnerability to simulate and analyze potential attacks on CNC machines utilizing the LBP16 protocol:

– The initial step involved a comprehensive analysis of the communications protocol between the CNC machine controller and CNC machine. This included a review of the official documentation as well as protocol reverse engineering,

which provided a thorough understanding of the communications dynamics and functionality.
- A parser was developed to facilitate the analysis. The tool dissected network dumps, enabling the accurate association of commands with their memory locations. This step helped plan the attack vectors and understand how commands could be manipulated.
- Next, the key registers and memory locations used by the controller to read and write data were identified and mapped. This established clear links between memory locations and the corresponding commands issued by the controller, providing the foundation for the simulated attacks.
- With a clear understanding of the vulnerability, tools were designed and implemented for its exploitation. The tools accessed and modified device memory, enabling the simulation of controlled attacks on CNC machines to assess their effectiveness and impacts.

3.3 Adversary Model

CNC machines are typically integrated into manufacturing networks [3]. Traditionally, these networks were isolated. However, in contemporary manufacturing facilities, CNC machines interface with external servers to facilitate remote programming and process monitoring. Also, CNC machines are often located in corporate networks and are accessible via industrial gateways or mobile networks. Mobile network operators provide CNC machines with Internet connectivity and industrial gateways bridge operational technology and information technology networks [3]. Balduzzi et al. [3] report that interviews with domain experts corroborate these assertions.

Figure 2 shows the adversary model adopted in this research. To a certain extent, the model is inspired by the work od Balduzzi et al. [3]. The adversary is modeled as follows:

- A remote adversary with access to the operational technology network poses a threat. The adversary could be an insider with direct access to the operational technology network housing the CNC machine. Alternatively, the adversary could be an external entity that exploits an enterprise network that lacks segmentation or is improperly configured, exposing the CNC machine to attacks.
- A remote adversary with access to the information technology network to which the CNC machine is connected is a potential threat. The adversary can pivot from the information technology network to access the CNC machine, potentially exploiting misconfigurations or vulnerabilities in the industrial gateway that bridges the information technology and operational technology networks.
- A remote adversary that establishes communications with the machine operator presents a threat. In this scenario, the attacker could employ social engineering techniques to gain access to the CNC machine.
- Insiders, employees as well as contractors, with authorized access to the CNC machine also pose threats.

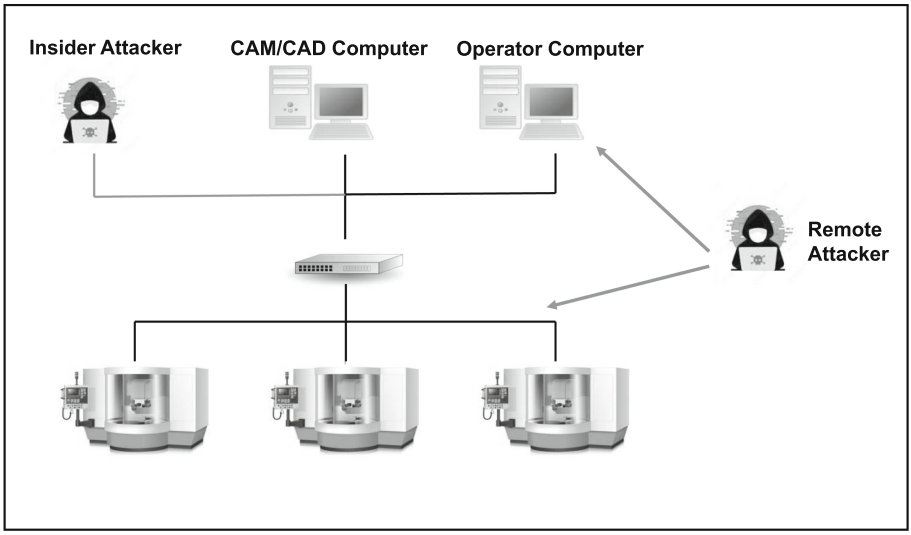


Fig. 2. Adversary model

4 CNC Machine Attack Case Study

This section describes the CNC machine case study and the attack implementation.

4.1 CNC Machine Case Study

For reasons of sensitivity, the CNC machine used in the case study is referred to as Machine X. The CNC machine, which employs the LBP16 protocol for controller-machine communications, incorporates a touchscreen controller, keyboard, mouse, jog shuttle for convenient control and a Wi-Fi module for connectivity.

The controller used in the case study, referred to as Controller Y, is built on LinuxCNC. Controller Y offers an intuitive, user-friendly graphical interface that simplifies CNC machine operation. The LinuxCNC capabilities are augmented with features such as conversational programming, which enables operators to generate G-code directly on the controller instead of being restricted to using pre-written code.

Attacks on Machine X were launched from an Ubuntu 22.04 virtual machine configured to communicate with the CNC machine via Ethernet. In line with the adversary model described above, Machine X was connected to a local network using an industrial switch. This setup mirrors a common industrial configuration where CNC machines in an operational technology network are linked to an information technology network using industrial switches.

Table 4. LBP16 protocol attacks and impacts

Attack Class	Attack	Impact
Sabotage	Fuzzing with manipulated input	Causes damage to tools, spindle and stepper motor
Denial of Service	Hostmot2 cookie overwriting	Causes the controller to display an interface error, preventing the machine from starting
	Card name overwriting	Causes the controller to display an interface error, preventing the machine from start-up
	IP address overwriting	Disrupts machine network connectivity
	Flash sector erasing	Causes the machine to halt and become unresponsive

The adversary in the model has network access and is assumed to have adequate knowledge of the LBP16 protocol that enables it to send potentially malicious commands to the CNC machine. Specifically, the adversary can launch targeted attacks that exploit protocol-specific vulnerabilities, such as sending malformed or unauthorized commands, testing the resilience of the CNC machine under simulated threat conditions.

Table 4 provides details about the five attacks developed to target the LBP16 protocol vulnerability described in Sect. 3.2. The attacks are classified into two categories based on their impacts on CNC machine functionality – sabotage and denial of service.

4.2 CNC Machine Attack Implementation

This section describes the implementation of the CNC machine attacks, including the memory mapping analysis required to develop the sabotage and denial-of-service attacks.

Memory Mapping Analysis. In order to elucidate the command structure and memory interactions of the LBP16 protocol, a parser was developed to deconstruct the packet payloads extracted from network dumps. The analysis revealed that each payload comprises a sequence of regularly-transmitted protocol commands. The commands initially write to specific memory locations followed by read operations, which continue cyclically. The systematic analysis provided a deep understanding of the protocol's operational dynamics related to network communications.

It was observed that during idle periods the protocol overwrites specific memory locations with zeros. These locations are subsequently populated with hexadecimal values upon executing a G-code file. G-code commands were executed in order to monitor and analyze the resulting network traffic. This enabled the correlation of registers and their values. For example, when a G-code instruction was issued to reposition the spindle, analysis of the network traffic enabled the identification of the memory locations that were overwritten with the corresponding x-, y- and z- coordinate values.

These efforts enabled the mapping of the critical memory locations used by protocol to overwrite memory. However, the mapping could vary depending on the CNC machine model. In any case, the analysis provided a precise understanding of how the protocol interacts with the machine's operational parameters.

Sabotage Attack. The first attack category involved the sabotage of tools, spindles and stepper motors. Fuzzing techniques that evaluate protocol implementations were employed to identify flaws that could be exploited to gain control of the system. Fuzzing involves generating and transmitting malformed packets to a device under test and evaluating the impacts.

A fuzzer for testing CNC machine responses was created to manipulate command inputs extracted from captured a network dump utilizing the parser. Specifically, the protocol values in memory location `0x0598` that stores the x-, y- and z- coordinate values were manipulated using a byte-flipping technique. For example, an original packet `84c2980598a2ffff9c6c000003fbffff00000000` was changed to `84c2980598a2ffff9c6c000003fbff0000ff0000` and stored in the memory location.

Two tests were conducted to assess the influence of altered data on Machine X. The first test was performed when the machine was idle and the second test was conducted during model creation. During the first test, the CNC machine emitted a startling sound because the fuzzed value caused the stepper motor to drive the spindle beyond its operational limit. Simultaneously, an error message appeared on the Controller Y screen indicating a machine malfunction and suggesting a review of the power supply and cable connections (Fig. 3). Despite the warning, the spindle persisted in its bidirectional movements and attempted to exceed its constraints even after the emergency stop button was activated.

In the second test conducted during model design, the controller engine halted unexpectedly (i.e., the spindle switched off), causing the machine to become unresponsive. The controller displayed an identical error message as in the first test, but the spindle continued to move beyond its limit. Maintaining this attack on the CNC machine would damage critical components such as the machine tool and stepper motor. Recognizing and mitigating such attacks promptly are crucial to avoid significant equipment damage and associated operational disruptions. Equipment damage is a concern because CNC machine costs range from a few thousand to millions of U.S. dollars [3]. Even if the damage is restricted to tools or specific machine components, delays due to replacement and

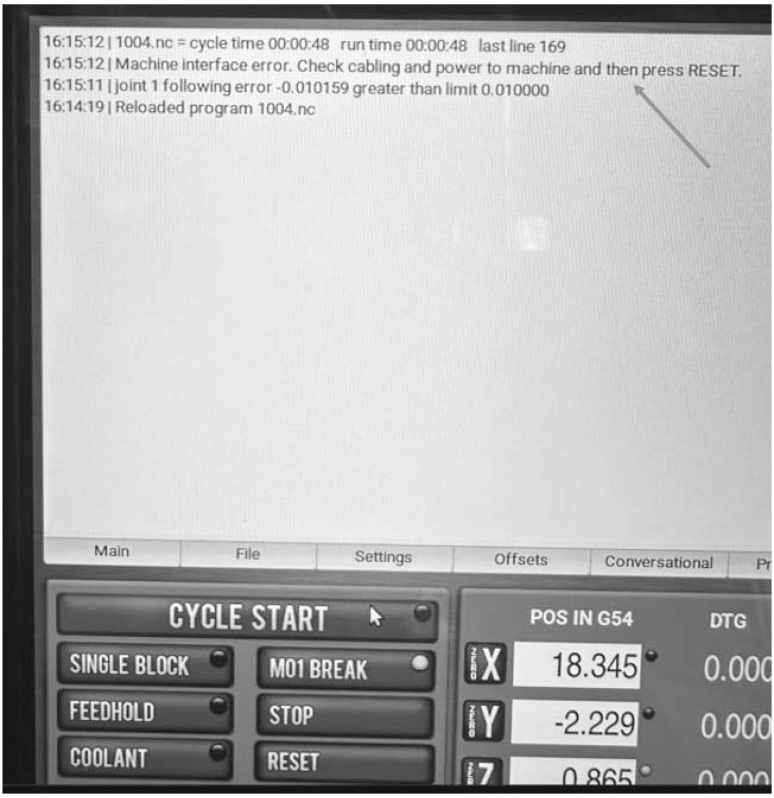


Fig. 3. Controller error message

maintenance could result in production line stoppage and significant economic losses.

Denial-of-Service Attacks. Four distinct denial-of-service attacks were developed: Hostmot2 cookie overwriting, card name overwriting, IP address overwriting and flash sector erasing:

– **Hostmot2 Cookie Overwriting Attack:** This denial-of-service attack targets the Hostmot2 cookie located in the Hostmot2 memory space [16]. The cookie value should not be manipulated or overwritten because the protocol uses it for authentication to confirm authorized access. Figure 4 shows that the initial LBP16 protocol command reads the cookie to confirm proper access – `01420001` is the command used to read the cookie and `fecaaa55` denotes the cookie itself. The objective of the Hostmot2 cookie overwriting attack is to corrupt the authentication process that relies on the integrity of the Hostmot2 cookie, undermining CNC machine security.

```
0000  02 80 e1 11 74 0c 68 ed  a4 5d 95 26 08 00 45 00   ····t·h· ·]·&··E·
0010  00 20 c7 18 40 00 40 11  4b 8e 0a 0a 0a 09 0a 0a   ·  ··@·@· K·····
0020  0a 0a bf 18 6a 2d 00 0c  ad 26 01 42 00 01 00 00   ····j-·· ·&·B····
0030  00 00 00 00 00 00 00 00  00 00 00 00              ···· ···· ····
```

(a) Initial command requesting the Hostmot2 cookie.

```
0000  68 ed a4 5d 95 26 02 80  e1 11 74 0c 08 00 45 00   h··]·&·· ··t···E·
0010  00 20 00 01 00 00 ff 11  93 a5 0a 0a 0a 0a 0a 0a   · ···· ·· ·····
0020  0a 09 6a 2d bf 18 00 0c  05 49 fe ca aa 55 00 00   ··j-···· ·I···U··
0030  00 00 00 00 00 00 00 00  00 00 00 00              ···· ···· ····
```

(b) Hostmot2 cookie.

Fig. 4. LBP16 protocol request for the Hostmot2 cookie

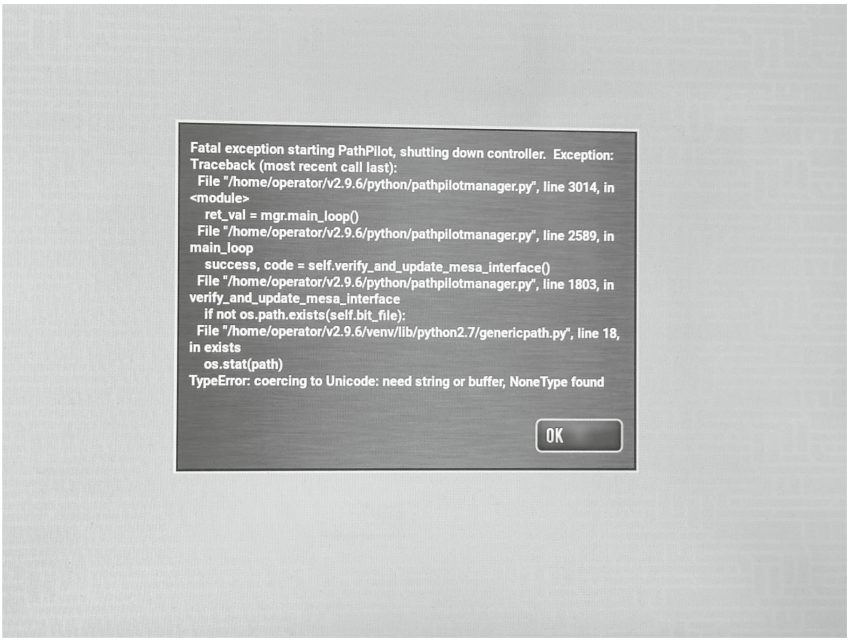
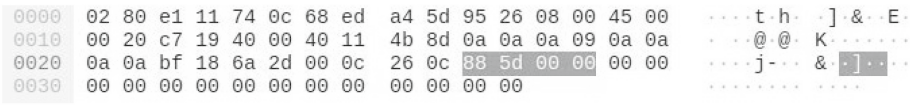
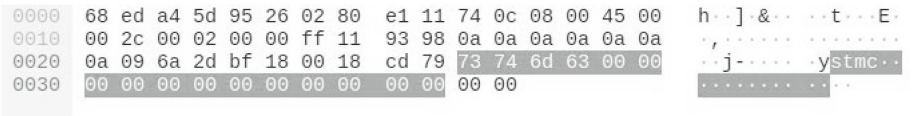


Fig. 5. Fatal exception error displayed by the controller

The Hostmot2 cookie overwriting attack was executed by issuing an overwrite memory command `01c20001fecc55aa` in which `fecc55aa` denotes the manipulated cookie value. The memory that holds the cookie is supposed to be read-only memory, but the attack successfully overwrote the cookie.
After Controller Y proceeded to read and process the manipulated cookie value, a fatal exception error shown in Fig. 5 was displayed. The error indicated the controller's inability to correctly interpret the manipulated cookie value. The attack terminated the connection between the controller and



(a) Command requesting the card name.



(b) Card name.

Fig. 6. LBP16 protocol request for the card name

the CNC machine, inducing a denial-of-service condition that halted CNC machine operation.

- **Card Name Overwriting Attack:** This attack targets the card name stored in the device memory. The card name, located in the LBP16 protocol read-only memory space, is crucial for device identification. According to the protocol sequence in Fig. 6, the command to read the card name is executed immediately after the Hostmot2 cookie is read. Specifically, the command `885d0000` is used to read the card name `73746d63`. The objective of the card name overwriting attack is to replace the card name with a corrupt value.
  The card name overwriting attack was executed by issuing an overwrite memory command `88dd0000feccaa55` that replaces the card name `73746d63` with the cookie value `feccaa55`. The memory that holds the card name is supposed to be read-only memory, but the attack successfully overwrote the value. When Controller Y proceeded to read and process the manipulated card name, the fatal exception error shown in Fig. 5 was displayed. The attack induced a denial-of-service condition that halted CNC machine operation.
- **IP Address Overwriting Attack:** This attack targets the network configuration of the CNC machine by altering its IP address, which is hard-coded in Ethernet EEPROM memory space. The IP address is vital for network communications and identity verification.
  The IP address overwriting attack was executed by setting the value of the write-enable register `EEPROMEna` to `5A02` to enable writing or erasing. Importantly, the `EEPROMEna` value and the command to replace the IP address `82c92000(190a0a0a)` must be sent together in a single packet. Accordingly, the packet `01D91A00025A82c92000190a0a0a` is sent to change the IP address from `0a0a0a0a` (hard-coded in memory) to the new IP address `190a0a0a`.
  The IP address overwriting attack disrupted the network connection between Controller Y and the CNC machine, resulting in denial of service. As shown in Fig. 7, Controller Y continually attempts to establish communications with the CNC machine using its IP address, but the efforts are unsuccessful. The attack is feasible because the LBP16 protocol, which supports IP address updates, lacks a robust authentication mechanism.

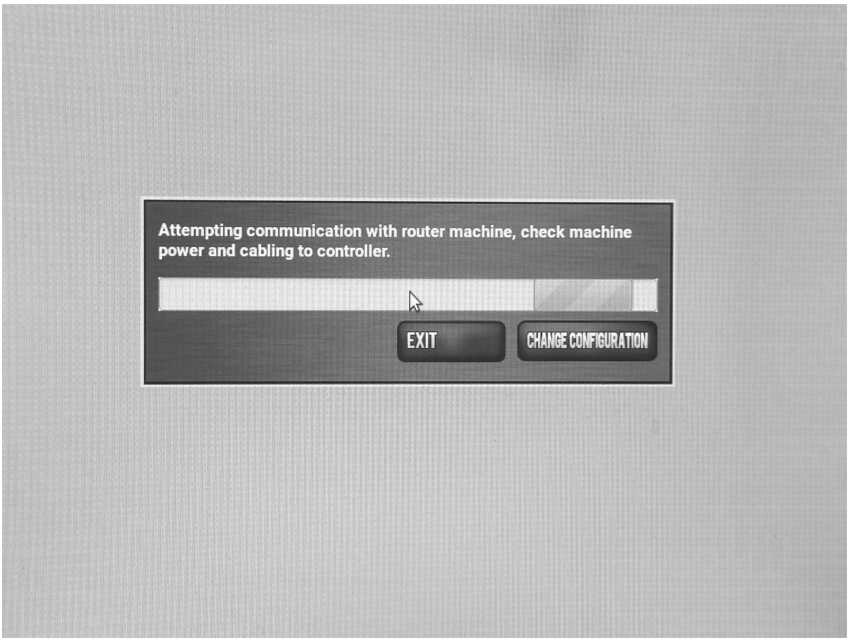


Fig. 7. Communications error displayed by the controller

– **Flash Sector Erasing Attack:** This attack targets the FPGA flash memory of the CNC machine, which holds critical data and configurations necessary for normal operation. Due to the absence of authentication mechanisms in the LBP16 protocol, an adversary who gains network access to the CNC machine can issue LBP16 protocol commands to delete specific sectors of the FPGA flash EEPROM memory space.
Access to the flash EEPROM using the LBP16 protocol is mediated via direct interactions with designated registers [16]. The primary mechanism involves the flash address register `FL_ADDR` and flash data register `FL_DATA`. In order to initiate an operation on the EEPROM, the target address is first written to the `FL_ADDR`. Subsequent read and write operations are executed via `FL_DATA`, facilitating precise control over data manipulation. Additionally, certain operations, especially erasing, require the EEPROM write function to be enabled via the EEPROM write-enable register `EEPROMWEna`.
The flash sector erasing attack requires four commands to be issued, all of which must be sent in a single packet:
  - **Enable EEPROM Write:** The EEPROM write-enable register is set to allow erasing by issuing the command `01D91A00(035A)` where `5A03` is the enabling code.
  - **Specify Target Flash Address:** The target sector address for the erase operation is specified by issuing the command `01CE0000(00000100)`, which points to `0x00010000` in the EEPROM.

- **Execute Sector Erase Command:** Sector erasure is initiated by issuing the command `01CE0C0000000000` that effectively erases the specified address.
- **Synchronize the Host Post-Erase:** The flash address is re-read by issuing the command `014E0000` to ensure host synchronization and verify the erase operation.

The flash sector erasing attack removed crucial network configuration data, including the IP address, resulting in the corruption of the CNC machine firmware. Following the restart of the CNC machine, Controller Y displayed a communications error on its screen, indicating it could not reconnect to the CNC machine (Fig. 7). The erasure led to a significant disruption of CNC machine operations that rendered standard recovery methods ineffective. The only feasible solution is to re-flash the firmware on the CNC machine circuit board, which is not a trivial process.

5 Controller Vulnerability and Case Study Analysis

While the research has primarily focused on LBP16 protocol vulnerabilities, the investigation also identified a vulnerability in the controller. Although it is not directly linked to the LBP16 protocol, the vulnerability underscores the multifaceted nature of security challenges related to CNC machines.

5.1 Controller Vulnerability Analysis

The analysis of the controller revealed a significant security risk stemming from unfettered access to the G-code shared folder. Specifically, the vendor-provided access feature expedites the transfer of G-code files directly from a CAD/CAM system to a CNC machine. While this mechanism is intended to enhance operational efficiency, it inadvertently introduces a significant vulnerability due to the absence of a robust authentication mechanism. The vulnerability can be leveraged by an adversary to gain access to a shared G-code folder, resulting in security breaches such as data theft, data corruption, system hijacking and denial of service.

5.2 G-code Encryption Case Study

The attack is linked to the controller feature that supports network-based sharing of G-code files. An experiment was conducted to replicate the behavior of an adversary that exploits the vulnerability. The experiment involved two steps:

– **Network Connection Establishment:** A Windows 10 virtual machine was employed to emulate a typical CAM/CAD computer system in a manufacturing environment. The virtual machine was connected to the shared G-code drive on the controller, replicating a realistic network scenario in which G-code files are commonly accessed and modified.

Algorithm 1: G-code file encryption.

Input: Path to G-code file $file\_path$.
Output: Encrypted G-code file.

$key \leftarrow$ Generate secure 16-byte encryption key
$IV \leftarrow$ Generate 16-byte initialization vector
$file\_path \leftarrow$ Path to the G-code file in the shared folder
$original\_data \leftarrow$ Read binary data from $file\_path$
$encrypted\_data \leftarrow$ **Encrypt**$(original\_data, key, IV)$
open $file\_path$ **in write mode**
write IV **followed by** $encrypted\_data$ **to** $file\_path$
close $file$
output "Encryption completed for" $file\_path$

– **Attack Script Development and Execution:** A Python script was developed to encrypt the G-code file on the shared drive. Algorithm 1 specifies the G-code file encryption script.

Execution of the attack script encrypted the G-code file, rendering it unusable by the CNC machine. The G-code file encryption illustrates the vulnerability of the shared drive to unfettered access and manipulation due to the lack of authentication. The vulnerability could also be leveraged by an adversary to steal G-code files containing proprietary manufacturing information, potentially resulting in a competitive disadvantage and financial loss. Furthermore, the access enables the adversary to inject malicious instructions into a G-code file, leading to a hijacking attack. The adversary could also alter the operation of CNC machinery, potentially leading to the creation of defective parts that could severely impact safety and brand reputation.

The case study demonstrates the impact of the lack of authentication vulnerability, highlighting how effortlessly critical manufacturing data can be compromised. It also draws attention to the need to implement robust security measures to protect shared network resources in CNC systems and safeguard the confidentiality of proprietary information and the integrity of manufacturing processes.

6 Disclosure and Mitigation

The security analysis of a CNC machine utilizing the LBP16 communications protocol has identified vulnerabilities that threaten the operational integrity and security of CNC machines and controllers. Recognizing the severity of the vulnerabilities, steps were taken to inform the vendor and engage in a collaborative effort to address the problems. Comprehensive reports were provided that included descriptions of the vulnerabilities, the methodologies used to expose and exploit the vulnerabilities, and the potential risks. Additionally, the following mitigation strategies were devised.

- **LBP16 Protocol Vulnerability Mitigation:** It is imperative to ensure that firmware and software of the CNC machine and other components utilizing the LBP16 protocol are updated with the latest security patches. A rigorous patch management process must be implemented to resolve known vulnerabilities. Additionally, the implementation of strong authentication mechanisms is crucial.
- **Controller Vulnerability Mitigation:** It is important to ensure that controller is secured. Robust software updates and patch management processes must be instituted. Also, strong access controls with strong password policies and multi-factor authentication must be implemented to safeguard CNC machines and associated systems.

7 Conclusions

The investigation of the security vulnerabilities of CNC machines using the LBP16 protocol for controller communications has yielded significant insights into the potential risks to subtractive manufacturing systems. These advanced machines, which are efficient and indispensable to high-volume production, are prone to cyber-physical attacks that include sabotage and denial of service. It is vital to integrate robust cyber security measures in CNC systems. The integration of cyber security measures is not just a technical requirement, but a fundamental aspect that should be instituted in the manufacturing industry's culture.

The views expressed in this chapter do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

Acknowledgement. This research was supported by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy under the Advanced Materials and Manufacturing Technologies Office (AMMTO) Award no. DE EE0009046.

References

1. Ahsan, M., Rais, M., Ahmed, I.: SOK: side channel monitoring for additive manufacturing – bridging the cybersecurity and quality assurance communities. In: Proceedings of the Eighth IEEE European Symposium on Security and Privacy, pp. 1160–1178 (2023)
2. Ayub, A., Yoo, H., Ahmed, I.: Empirical study of PLC authentication protocols in industrial control systems. In: Proceedings of the IEEE Security and Privacy Workshops, pp. 383–397 (2021)
3. Balduzzi, M., Sortino, F., Castello, F., Pierguidi, L.: A security analysis of CNC machines in Industry 4.0. In: Gruss, D., Maggi, F., Fischer, M., Carminati, M. (eds.) Detection of Intrusions and Malware, and Vulnerability Assessment, pp. 132–152. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-35504-2_7

4. Bi, E.: Detection and intrusion of attacks in cyber-physical security for additive manufacturing, Ph.D. Dissertation, Department of Industrial and Systems Engineering, Auburn University, Alabama (2023)
5. Cardenas, A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., Sastry, S.: Challenges for securing cyber physical systems. In: Presented at the Workshop on Future Directions in Cyber-Physical Systems Security (2009)
6. Chen, T., Abu-Nimeh, S.: Lessons from Stuxnet. IEEE Comput. **44**(4), 91–93 (2011)
7. Chen, X., Wang, Z., Yang, S.: Research on information security protection of industrial-internet-oriented CNC system. In: Proceedings of the Sixth IEEE Information Technology and Mechatronics Engineering Conference, pp. 1818–1822 (2022)
8. Chhetri, S.: Novel side-channel attack model for cyber-physical additive manufacturing systems, M.S. Thesis, Department of Electrical and Computer Engineering, University of California, Irvine, Irvine, California (2016)
9. Cultice, T., Thapliyal, H.: Vulnerabilities and attacks on CAN-based 3D printing/additive manufacturing. IEEE Consum. Electron. Mag. **23**(1), 54–61 (2023)
10. DeSmit, Z., Elhabashy, A., Wells, L., Camelio, J.: Cyber-physical vulnerability assessment in manufacturing systems. Procedia Manuf. **5**, 1060–1074 (2016)
11. Elhabashy, A., Wells, L., Camelio, J., William, H.: A cyber-physical attack taxonomy for production systems: a quality control perspective. J. Intell. Manuf. **30**, 2489–2504 (2019)
12. Falliere, N., Murchu, L., Chien, E.: W32.Stuxnet Dossier, Version 1.4, Symantec, Mountain View, California (2011)
13. Fidler, D.: Was Stuxnet an act of war? Decoding a cyberattack. IEEE Secur. Priv. **9**(4), 56–59 (2011)
14. Graves, L., King, W., Carrion, P., Shao, S., Shamsaei, N., Yampolskiy, M.: Sabotaging metal additive manufacturing: powder delivery system manipulation and material-dependent effects. Add. Manuf. **46**, 102029 (2021)
15. Li, Z., et al.: Detecting cyber-attacks against cyber-physical manufacturing system: a machining process invariant approach. IEEE Internet Things J. **11**(10), 17602–17614 (2024)
16. Mesa Electronics, 7I94 Ethernet Anything I/O Manual, Version 1.0, El Sobrante, California (2024). www.mesanet.com/pdf/parallel/7i94tman.pdf
17. Prasad, R., Moon, Y.: Architecture for preventing and detecting cyber attacks in a cyber-manufacturing system. IFAC-PapersOnLine **55**(10), 2246–2251 (2022)
18. Rais, M., Ahsan, M., Ahmed, I.: FRoMEPP: digital forensic readiness framework for a material-extrusion-based 3D printing process. Forensic Sci. Int. Digit. Invest. **44**, 301510 (2023)
19. Rais, M.H., Ahsan, M., Sharma, V., Barua, R., Prins, R., Ahmed, I.: Low-magnitude infill structure manipulation attacks on fused filament fabrication 3D printers. In: Staggs, J., Shenoi, S. (eds.) Critical Infrastructure Protection XVI: 16th IFIP WG 11.10 International Conference, ICCIP 2022, Virtual Event, March 14–15, 2022, Revised Selected Papers, pp. 205–232. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-20137-0_8
20. Tu, S., Liu, G., Lin, Q., Lin, L., Sun, Z.: Security framework based on trusted computing for industrial control systems of CNC machines. Int. J. Perform. Eng. **13**(8) (2017)
21. Wu, M., Song, Z., Moon, Y.: Detecting cyber-physical attacks in cyber manufacturing systems with machine learning methods. J. Intell. Manuf. **30**, 1111–1123 (2019)

22. Yampolskiy, M., et al.: Security of additive manufacturing: attack taxonomy and survey. Addit. Manuf. **21**, 431–457 (2018)
23. Yampolskiy, M., Schutzle, L., Vaidya, U., Yasinsac, A.: Security challenges of additive manufacturing with metals and alloys. In: Rice, M., Shenoi, S. (eds.) ICCIP 2015. IAICT, vol. 466, pp. 169–183. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26567-4_11
24. Zetter, K.: Blockbuster worm aimed for infrastructure, but no proof Iran nukes were target. Wired, 23 September 2010
25. Zubair, N., Ayub, A., Yoo, H., Ahmed, I.: Control logic obfuscation attack in industrial control systems. In: Proceedings of the IEEE International Conference on Cyber Security and Resilience, pp. 227–232 (2022)