

Hands-on Activity

We've discovered that an attacker gained access to a Programmable Logic Controller (PLC) by exploiting its password. Originally, the PLC password was set to "cool" by the engineer. However, the attacker changed it and reprogrammed the PLC. As a result, the conveyor belt in the chocolate factory stopped working correctly. It could no longer sort chocolates with nuts from those without which a serious problem for customers with nut allergies. If not fixed, the company could be sued!

In this activity, you'll act as a cyber detective. You'll examine the communication between the attacker and the PLC and uncover the new password the attacker set.

How Machines Talk

Machines communicate over a network using IP addresses (like digital names for devices). This traffic can be recorded using tools like **Wireshark**, which lets us see who talked to whom and what they said.

- The **PLC's IP** is 192.168.10.1
- The **attacker's IP** is 192.168.10.112

How Passwords Are Hidden

Instead of saving passwords directly, systems use **hashing**, a one-way process that turns passwords into scrambled strings. This keeps them safe even if someone breaks into the system.

There are different types of hashes, like **MD5**, **SHA-1**, and **SHA-256**. The attacker used **SHA-256** to hash the new password. Your mission is to dig through the network communication and figure out what password was set.

Are you ready to find it?

Investigation Steps

Let's walk through how to uncover the password the attacker set on the PLC. You'll be using a tool that shows the messages (called **packets**) exchanged between the attacker and the PLC.

Key Terms to Know

Packet: Think of it as a digital envelope. It carries data (like a command or a message) from one device to another.

Byte: A group of 8 bits — like a small piece of information. It can represent a number, a letter, or part of a password.

Hash: A scrambled version of a password. It's not the password itself, but it's how passwords are usually stored securely.

XOR: A type of puzzle where you mix two things and get a new result. You can use an online tool to do this.

Step 1: Download the File

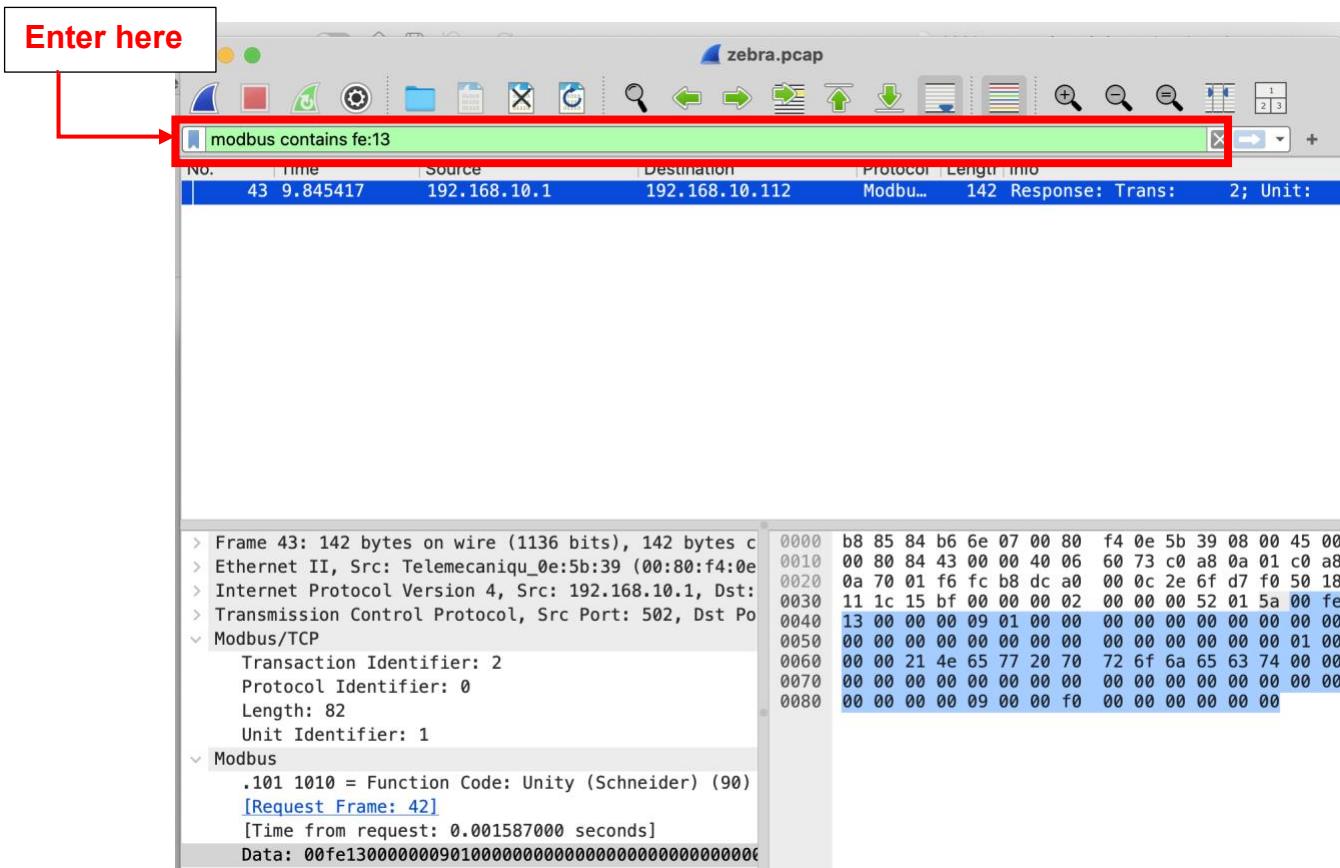
Download zebra.pcap to your computer. This file contains recorded messages between the attacker and the PLC.

Step 2: Filter the First Authentication Request

Open the file in Wireshark (the tool that shows network traffic).

In the search bar, type: “**modbus contains fe:13**”

This filters for the packet (message) where the attacker starts the authentication. You should see only one packet appear.



Step 3: Find the First Value (m_1).

Inside this packet, look for the 1-byte value (7th position counting from backwards) This value is called m1.

0000	00	0c	29	dd	d6	56	00	50	56	e7	6f	fb	08	00	45	00
0010	00	80	a5	8e	00	00	80	06	26	17	c0	a8	0a	01	c0	a8
0020	e3	80	01	f6	d2	5e	6e	20	39	b2	63	12	c7	af	50	18
0030	fa	f0	8e	0c	00	00	00	02	00	00	00	52	01	5a	00	fe
0040	13	00	00	00	09	01	00	00	00	00	00	00	00	00	00	00
0050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	00
0060	00	00	21	4e	65	77	20	70	72	6f	6a	65	63	74	00	00
0070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0080	00	00	00	00	09	00	00	8f	00	00	00	00	00	00	00	00

Step 4: Find the Second Value (m2)

In the filter bar, type: “**modbus contains 6d:05**”

You'll see many packets. This is because the attacker kept trying to write the password followed by authentication until it worked. Since all packets contain the same password hash, pick the **first one**. In this packet, find the **1 byte** right after 6d:05. This is **m2**.

modbus contains 6d:05

Enter here

NO.	Time	Source	Destination	Protocol	Length	Info
20	58.307370	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 4; Un.
26	58.310778	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 6; Un.
32	58.314777	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 8; Un.
38	58.318797	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 10; Un.
44	58.322693	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 12; Un.
50	58.326768	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 14; Un.
56	58.330787	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 16; Un.
62	58.334768	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 18; Un.
68	58.338649	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 20; Un.
74	58.342694	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 22; Un.
80	58.346730	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 24; Un.
86	58.350779	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 26; Un.
92	58.354744	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 28; Un.
98	58.358617	192.168.227.128	192.168.10.1	Modbu...	98	Query: Trans: 30; Un.

```
> Frame 20: 98 bytes on wire (784 bits), 98 bytes capt
> Ethernet II, Src: VMware_dd:d6:56 (00:0c:29:dd:d6:56)
> Internet Protocol Version 4, Src: 192.168.227.128, D
> Transmission Control Protocol, Src Port: 53854, Dst
  Modbus/TCP
    Transaction Identifier: 4
    Protocol Identifier: 0
    Length: 38
    Unit Identifier: 1
  Modbus
    .101 1010 = Function Code: Unity (Schneider) (90)
    Data: 6a6d05773bb8bd39592375e3c7327291d960be91aa22
```

0000	00	50	56	e7	6f	fb	00	0c	29	dd	d6	56	08	00	45	00
0010	00	54	13	1b	40	00	40	06	b8	b6	c0	a8	e3	80	c0	a8
0020	0a	01	d2	5e	01	f6	63	12	c7	df	6e	20	3a	14	50	18
0030	fa	83	6f	19	00	00	00	04	00	00	00	26	01	5a	6a	6d
0040	05	77	3b	b8	bd	39	59	23	75	e3	c7	32	72	91	d9	60
0050	be	91	aa	22	56	ff	[12]	0e	e8	b3	b4	7f	15	c3	ad	4e
0060	57	e3														

M2

Step 5: Get the Encrypted Password Hash

Now that you have m2, look at the bytes that come **right after m2**. These bytes are the **encrypted version of the password's hash**.

0000	00	50	56	e7	6f	fb	00	0c	29	dd	d6	56	08	00	45	00
0010	00	54	13	1b	40	00	40	06	b8	b6	c0	a8	e3	80	c0	a8
0020	0a	01	d2	5e	01	f6	63	12	c7	df	6e	20	3a	14	50	18
0030	fa	83	6f	19	00	00	00	04	00	00	00	26	01	5a	6a	6d
0040	05	77	3b	b8	bd	39	59	23	75	e3	c7	32	72	91	d9	60
0050	be	91	aa	22	56	ff	[12]	0e	e8	b3	b4	7f	15	c3	ad	4e
0060	57	e3														

Encrypted Password hash

Step 6: Decrypt the Password Hash using XOR

Now we need to “unmix” the hash by applying XOR to each byte.

Use this online tool: [XOR calculator](#)

Remember, m_1 and m_2 are each 1 byte long. First, calculate $m_1 \text{ XOR } m_2$ — this gives you a single byte. Then, to decrypt the password hash, XOR each of the 32 bytes in the encrypted password hash with this single byte value. This will give you the actual SHA-256 hash of the password.

If $m_1 = f_0$

$$m_2 = 77,$$

Then $m1 \text{ XOR } m2 = 87$

To determine the SHA-256 hash

You'll enter:

The result will be the original **SHA-256 hash** of the password.

Step 7: Crack the Password

Now that you have the SHA-256 hash, try to find out what password produces this hash. You can use online [hash cracking tools](#).

Questions

- What password did the attacker set?
 - Is it still “cool”, or did they change it?
 - What would have happened if this change hadn’t been discovered?