Experimental Setup for Grid Control Device Software Updates in Supply Chain Cyber-Security

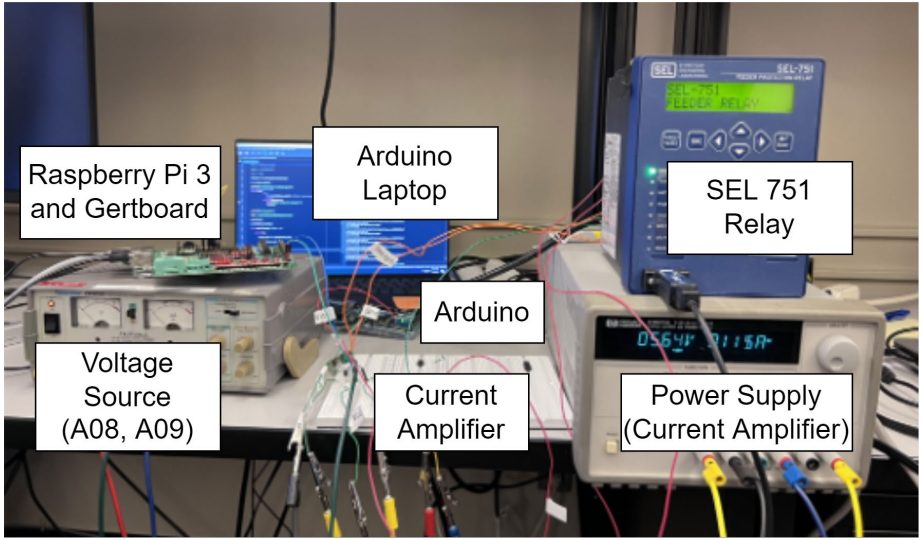
By Joseph Keller, Shuva Paul, Santiago Grijalva and Vincent Mooney

The Research Project

Supply chain attacks use less secure third-party software to infiltrate their target.

Our research is attempting to create defense techniques against this approach.

To test our strategies, need a laboratory setup.



SEL 751 Relay

Overcurrent Protection Relay
Serial and Ethernet Ports
1A Nominal Current

OSIsoft

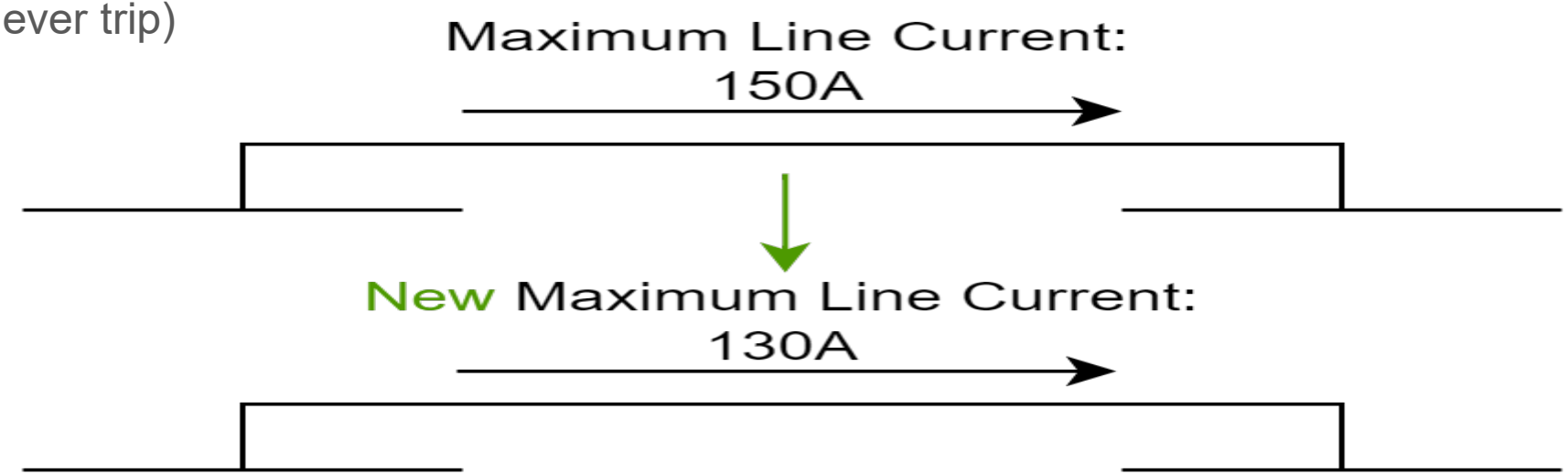
- Communication and data management software
  - Works with DNP3, MODBUS, and IPSec
- Simulates Power system control center
  - Plant Information (PI) system is used

Scenario

A distribution line is having its thermal limits changed to accomidate hotter weather

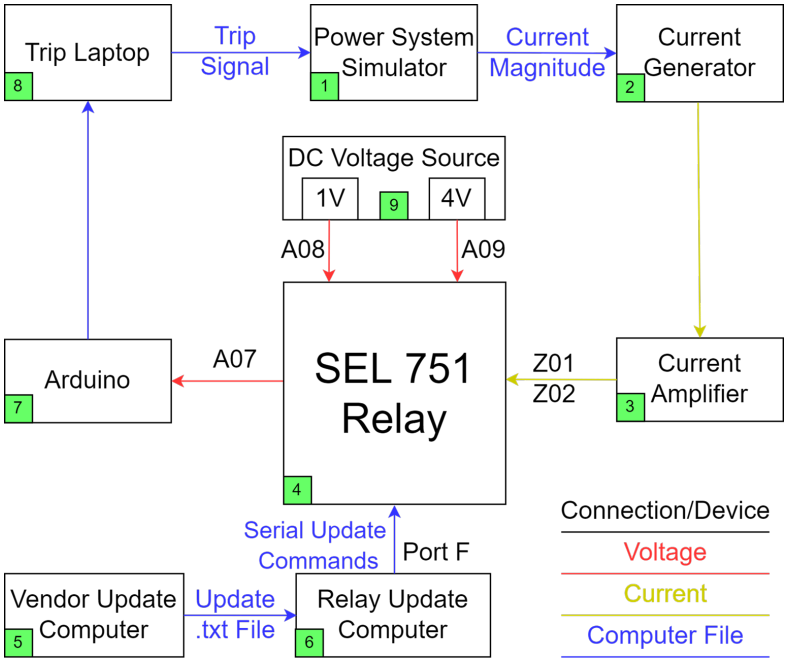
Change the current limit from 150A to 130A

An attacker will either try to lower the limit further (false trip) or increase the limit (never trip)



Laboratory Setup

9 main components are utilized to create the testing environment



Power System Simulator

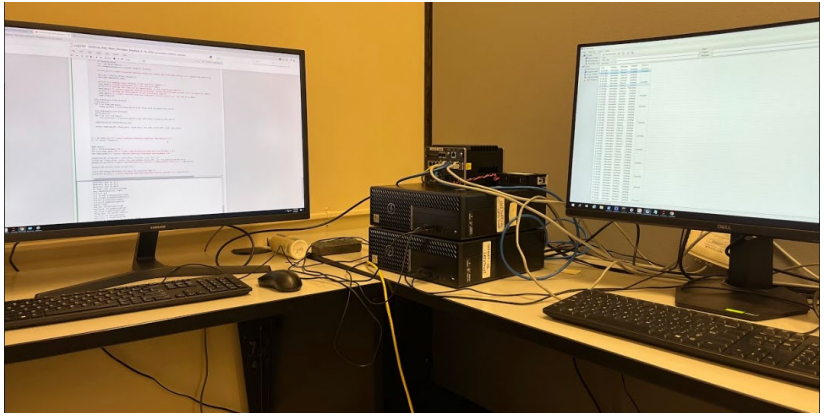
Developed our own Power System Simulator

Written in Python

Allows for changes to loads, generation, and opening/closing of lines

The first process to be completed during the loop

- Sends current values to the current generator
- Receives trip signal from the trip laptop



Current Generator and Amplifier

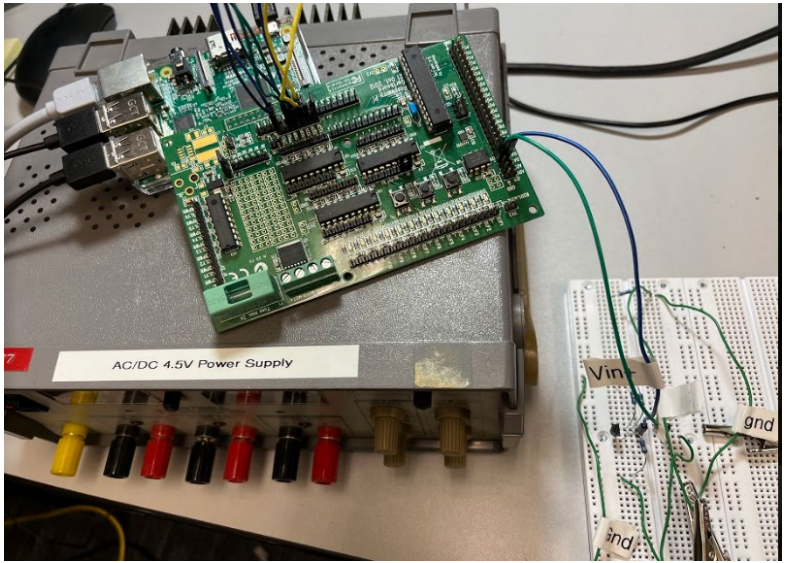
Current generator created using a Raspberry Pi 3 and a Gertboard

Raspberry Pi 3 acts as a computer, receiving current value from the power system simulator

Gertboard acts as a I/O device, producing the signal

Current amplifier created by an NPN, resistors, and a DC voltage supply

Scales the current generator up to 170mA.



Software Update Methodology

Two aspects

- A text file that contains serial commands for the relay
- Rust code that sends the serial commands to the relay

Vendor computer sends the text file to the update computer

Update computer uses the Rust code to update the relay

Table I: Summary of serial command text file for changing the 50P1P value

Sequence	Code	Execution
1	ACC	Gain access level one
2	OTTER	Default password for level one
3	2AC	Gain access level two
4	TAIL	Default password for level two
5	SET 50 P1P	Target the 50P1P value for change
6	0.13	Change the 50P1P value to 0.13A
7	...	Keep the other settings the same
8	Y	Confirm changes
9	Y	Begin update
10	STA R	Restart the relay
11	Y	Confirm restarting the relay

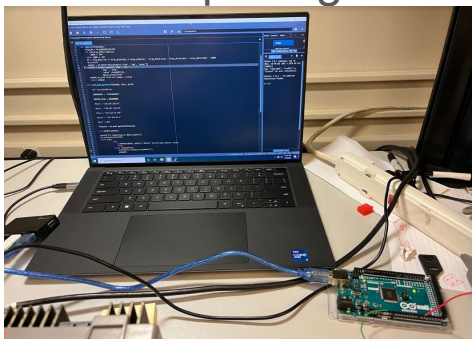
Table II: Summary of serial command text file for changing the active settings group.

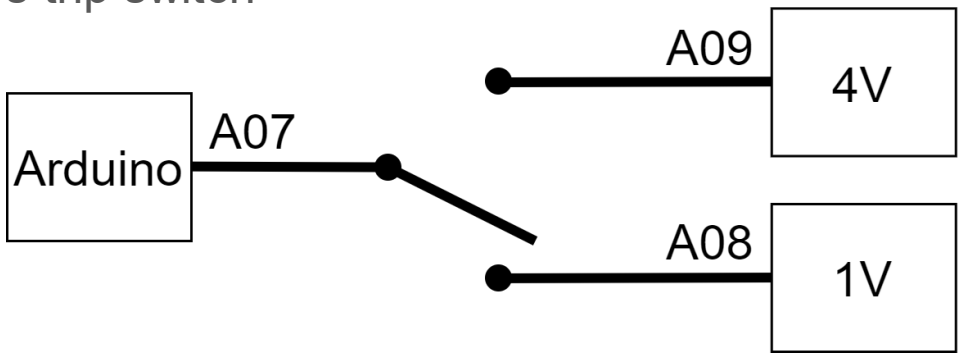
Sequence	Code	Execution
1	ACC	Gain access level one
2	OTTER	Default password for first level
3	2AC	Gain access level two
4	TAIL	Default password for second level
5	Group n	Change active settings group to group n
6	Y	Confirm and begin update
7	STA R	Restart the relay
8	Y	Confirm restarting the relay

Trip Signal

Three devices used to send the trip signal

- Arduino MEGA 2560 to read the analog output of terminal A07 in the relay
- Trip laptop to communicate with the Power System Simulator and Arduino
- DC Voltage Source to provide voltage to the A09 and A08 terminals, completing the Relay's trip switch

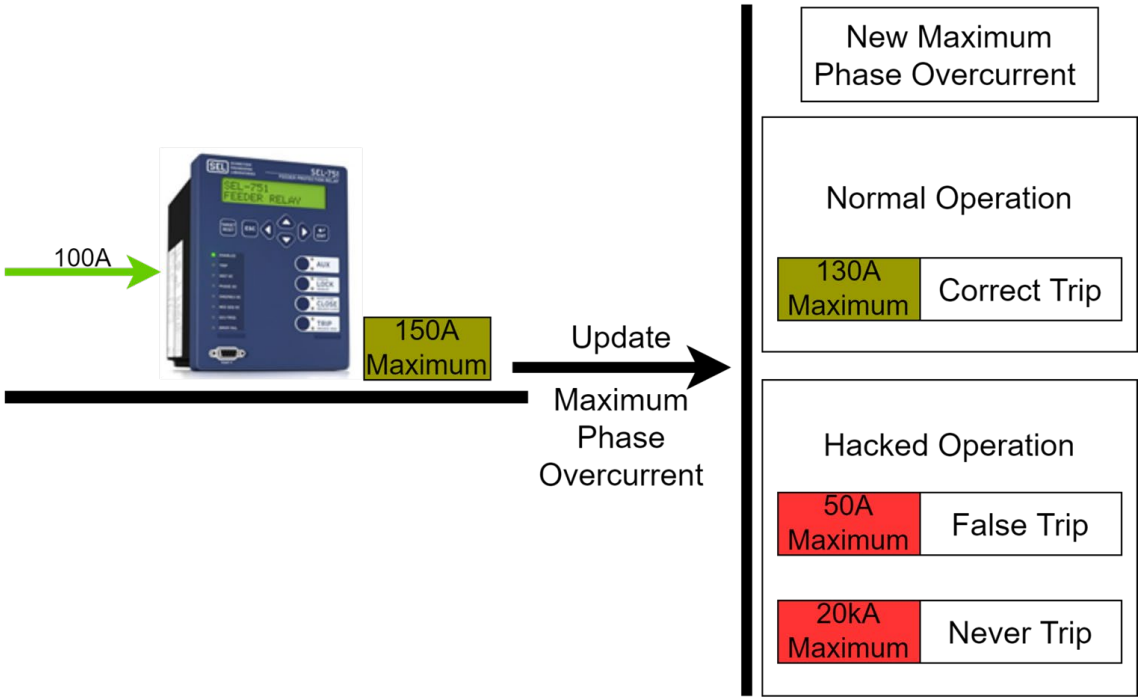




Experimental Use Cases

Three Use Cases

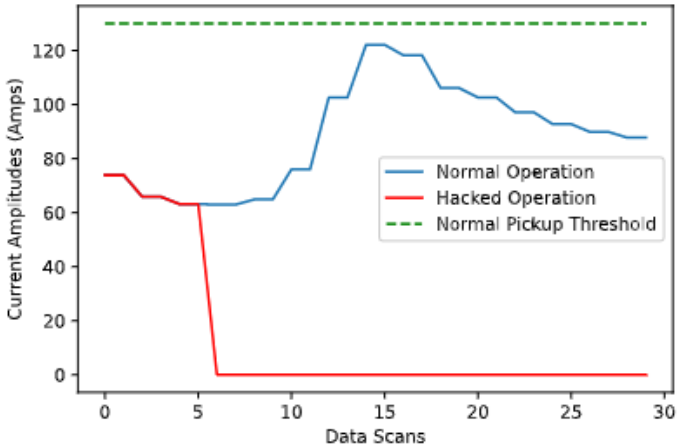
- Normal
  - 150A-130A
- False Trip
  - 150A-50A
- Never Trip
  - 150A-20kA



Experimental Results

False Trip

The simulator trips after the update is applied in the fifth data cycle.



Never Trip

The simulator does not trip during the correct time after the update is applied.