The Tao of Hardware,

The Te of Implants

5 examples of simple and easy hardware implants

Joe FitzPatrick joefitz@securinghardware.com

Abstract

Embedded, IOT, and ICS devices tend to be things we can pick up, see, and touch. They're designed for nontechnical users who think of them as immutable hardware devices. Even software security experts, at some point, consider hardware attacks out of scope. Thankfully, even though a handful of hardware manufacturers are making some basic efforts to harden devices, there's still plenty of cheap and easy ways to subvert hardware. The leaked ANT catalog validated that these cheap hardware attacks are worthwhile. The projects of the NSA Playset have explored what's possible in terms of cheap and easy DIY hardware implants, so I've continued to apply those same techniques to more embedded devices and industrial control systems. I'll show off a handful of simple hardware implants that can 1) Blindly escalate privilege using JTAG 2) Patch kernels via direct memory access on an embedded device without JTAG 3) Enable wireless control of the inputs and outputs of an off-the-shelf PLC 4) Hot-plug a malicious expansion module onto another PLC without even taking the system offline and 5) Subvert a system via a malicious display adapter. Some of these are new applications of previously published implants - others are brand new.

I'll conclude with some potential design decisions that could reduce vulnerability to implants, as well as ways of protecting existing hardware systems from tampering.

Introduction

In Chinese Philosophy, Tao refers to the absolute principle underlying the universe. It combines yin and yang and behavior in balance with natural order. By extension, the Tao of Hardware refers to the fact that hardware was and is the underlying bedrock upon witch all the order of the world of computing depends upon.

Likewise, Te refers to virtue and inner power. This is relevant to implants, because regardless of their size, complexity, or cost, they harness inner strength that can dramatically affect the operation of a normal hardware, and by extension, software system.

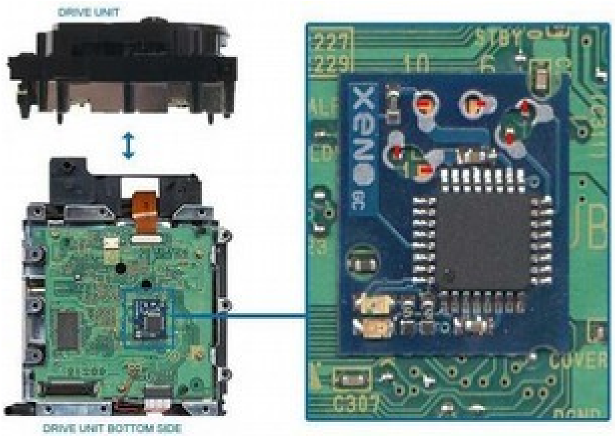
Lao-Tze, the purported author of the Tao Te Ching stated:

‘Do the difficult things while they are simple and do the great things while they are small”

Fundamentally, this is the inner strength of hardware implants. Given a complex mixed system of hardware and software, even one incorporating best practices of both hardware and software security, oftentimes the simplest hardware implant is able to take advantage of some small implicit trust made by the system, and extend that to undermine the entire system.

Background

Before 2013, public knowledge of ‘malicious’ hardware implants was generally limited to game console modchips. By tinkering or



tampering with a couple wires, these modchips are able to undermine the hardware and software security of game consoles to enable home-brew, cross-region, backed up, and pirated games. While the hardware aspect - which often means installation requires soldering to a relatively expensive piece of consumer electronics – limits adoption of hardware modchips, it is often the hardware mod chips that enable further research and understanding of systems that eventually make soft-mods possible.

On December 30, 2013, Der Spiegel published “Catalog Advertises NSA Toolbox”, highlighting leaked pages from a purported catalog of hardware and software implants and spy toys. Media may have been impressed by the ‘magic’ some of the hardware implants were capable of, but motivated hackers worked together to publish the NSA Playset, a suite of open-hardware and open-software re-implementations of many tools mentioned in the leaked catalog.

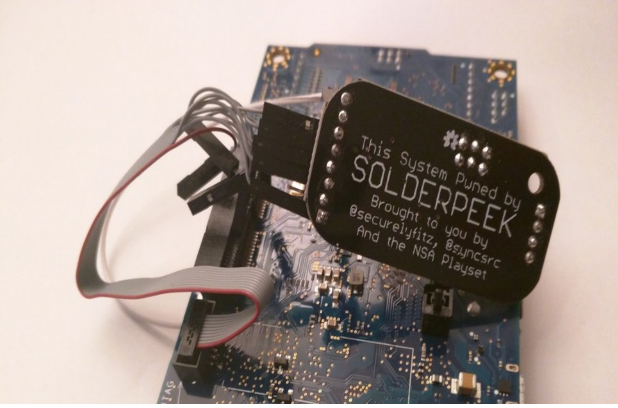
This proved to the world that hardware implants are real, but also reminded us that hardware implants don’t live in a vacuum. The most effective hardware implants exist to install, enable, or empower malicious software, which is then a much more flexible and versatile tool to carry out the implant’s intended purpose.

This paper will review 5 separate uses of implants for malicious tampering with systems. First, an implant with a pre-programmed series of JTAG commands will escalate privilege of a software console with zero feedback from the system. Second, a malicious PCIe device will be used to patch running kernels in an embedded network devices. Moving toward critical infrastructure targets, an inexpensive radio and microcontroller quickly embedded into an off-the-shelf PLC will enable remote wireless control of the PLC’s outputs without any software visibility. Continuing on infrastructure targets, the previous JTAG implant will be embedded into a hot-pluggable expansion module in order to once again toggle outputs of a PLC system without software visibility. Lastly, this paper will explore some possibilities of reprogrammable display adapters that will soon be the only option for display output on many computers.

Blindly Escalating Privilege via JTAG

SOLDERPEEK from the NSA Playset is a reimplementation of FLUXBABBIT, disclosed in the previously mentioned Der Spiegel article. It is a general-purpose microcontroller that can play back a scripted series of JTAG commands to modify the state an execution of a running system. SOLDERPEEK was introduced at DEFCON 24, and was followed with “JTAG to Root, 5 ways” which described a series of different ways that JTAG could be used to manipulate an operating system to escalate privilege.

In this example, we use SOLDERPEEK to deliver a series of JTAG commands that will patch getty on the fly to force authentication of a user when they attempt to login. The target is an Intel Quark CPU, and based on memory analysis of the system, we know where in memory to find getty and exactly what address to patch.



In this case, the implant was based on an ATMega328p – found on many complete Arduino boards for under $5. The payload was prototyped with OpenOCD, and then a log of JTAG transactions was trimmed down to the essential few to get the job done. Installation was rapid because the chosen target board already had a JTAG header populated. Since the design of SOLDERPEEK is quite simple, it is trivial to adapt it to various shapes, sizes, and pinouts. Lasty – SOLDERPEEK speaks JTAG. At the lowest transport level, JTAG is standardized, so SOLDERPEEK can work on x86, ARM, MIPS, and other architectures.

Patching Kernels via DMA on embedded devices

SLOTSCREAMER was another entry in the NSA Playset. While not based directly on an entry in any NSA catalog, similar capabilities are available from forensics tools suppliers. DMA attacks are not new, and PCIe based DMA attacks have been demonstrated over and over again, but SLOTSCREAMER did it with a $15 ASIC instead of a $75 FPGA, meaning that a complete functional board is both compact and inexpensive.

There is a common assumption that PCIe slots only exist in full-fledged desktop computers. The reality is that, in pursuit of I/O performance, even sub-$10 embedded arm SOCs now contain PCIe. Even when PCIe is not used in a specific design, the functionality still exists on the silicon and may be configurable. Fundamentally, PCIe is a specification designed to be easy to work with from a hardware perspective. As long as the wires are connected right, software can do all the rest. Since PCIe devices are almost exclusively I/O devices, it makes sense for them to be granted access to the system in order to transfer data in and out of memory. DMA is nearly the whole point of why PCIe exists, so it’s usually trivial to get DMA working once a PCIe device is connected properly.

In this case, the implant was inserted into a Mediatek MT7629 MIPS-based router with a miniPCIe slot. This particular system ships with a linux image that includes built-in drivers for several PCIe-based SATA, USB, and WIFI controllers. By reporting itself to be a SATA controller, SLOTSCREAMER coaxes the system into enabling DMA access. In this demonstration, an attack system controls SLOTSCREAMER via USB, but there is an embedded 8051 core that could handle all of that control.

The key takeaway from this example is that even ‘small’ systems have advanced interfaces. No device is too simple or too complicated to be impervious to implants. In addition, just because a device doesn’t use an interface like PCIe doesn’t mean the interface is truly disabled in a way that it couldn’t be re-enabled from hardware.

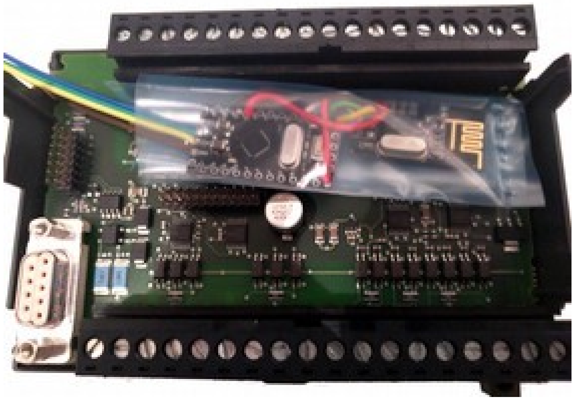
Enable wireless control of an off-the-shielf PLC

The two previous examples could be dismissed as ‘junk hacking’. There’s no need to challenge that assertion, but the reality of junk hacking is that it’s the NDA-free, low-risk playground for honing hardware skills. The differences between a $20 IOT doodad on kickstarter and a $300 PLC are minimal or nonexistent to a skilled hardware engineer.

The selected targets are Siemens S7-200 and S7-1200 Programmable Logic Controllers. These are devices that are used to control valved, dials, machinery and other controls in an industrial setting. Usually they are installed in secured locations – but not always. In addition the definition of ‘secured’ is relative.

Disassembling either device reveals an organized design with a Power PCB, Logic/CPU PCB, and I/O PCB. Off-the-shelf parts are easy to identify and the fundamental functionality is apparent in a matter of seconds to a skilled engineer. Signals to control the industrial equipment are generated by software running on an ARM cpu on the logic PCB, and then travel over a connector to the I/O PCB where they are buffered and drive relays. By intercepting these signals electrically, we can control them far enough down the wire that software has no way of perceiving that they are being manipulated.

As an implant, an off-the-shelf Arduino Pro Micro ($2) is combined with an nRF24L01+ radio module($1). The implant is so cheap, the wire and solder are significant contribution to the overall cost of the implant.



Using sample code available directly from Arduino, the implant listens for a command to set an output to high when it receives a ‘1’, low when it receives a ‘0’ and high-impedance when it receives an ‘x’.

The implant is then placed inside the easily disassembled PLC. Lines for Power, Ground, and Signal are soldered or crimped to specific locations on the I/O PCB. The implant is reassembled and placed into service running a known program – but at any point in time, the implant might receive a command to alter the output of the system.

What’s most surprising about this implant is how cheap it is. Second, how quickly, easily, and undetectably it can be implanted into a target devices. It might be unrealistic to enter a controlled facility to manipulate an installed PLC, but it’s entirely conceivable to interdict shipments or otherwise intercept the supply chain to deliver tampered PLCs. Also – the concept and implant are portable to nearly any type of embedded system.

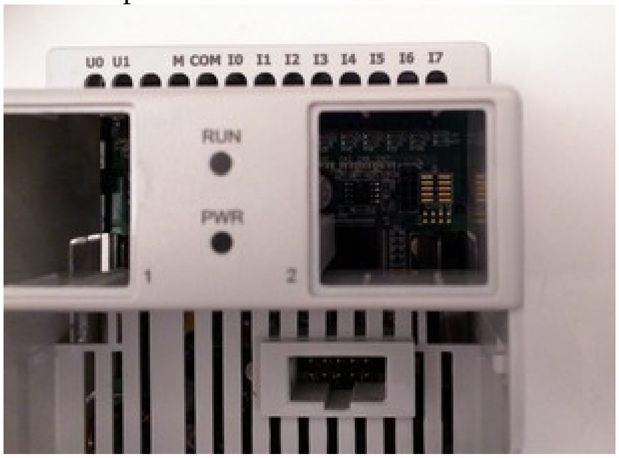
Hot-Plug malicious PLC expanders

While the previous implant required disassembly and invasive manipulation of the target PLC, this implant is an attempt to manipulate a live system. In this case, SOLDERPEEK is re purposed against a Phoenix Contact PLC.

Inspecting the PLC reveals that there is a JTAG header adjacent to the 10-pin female header that the expansion modules typically plug into. USB, Serial, and Ethernet modules are available, but the device ships with blanks in place. Solderpeek is combined with carefully placed pogo pins inside a blank so that when the blank is inserted, it gets power from the expansion header, and touches the JTAG pins of the exposed unpopulated header.

Like many embedded processors, the NXP LPC1765 inside this PLC controls its outputs via memory mapped IO regions. Analysis of the

PCB reveals which PLC outputs are driven by which LPC1765 I/O pins, and the related memory mapped regions are well documented in the chip's datasheet.



In this scenario, the SOLDERPEEK is programmed to delay 5 seconds, then play back a JTAG script that will overwrite the GPIO output to turn on one of the output relays. The code running on the system is unaware that this has happened, and operates like normal. Software control to enable/disable the output works like normal and never knows that there was any manipulation.

Critics of 'junk hacking' consider it unrealistic and too easy when targeting low-end consumer devices, but the reality is that high end industrial devices are not much different. JTAG should be disabled – but it's usually not. Test headers still exist on most production devices. In this case, the implant can be attached to a live, running system in operation with minimal interruption. The moral of the story – if you can get JTAG working, you win.

BadUSB-style display adapters

The last implant is not a custom hardware device, but a re-purposed off-the-shelf device. Malicious USB devices have been talked about for over a decade marked by advanced in lower and lower level attacks without much improvement in user habits. USB type C was recently unveiled as a new, better, more universal connector. Regardless of any new vulnerabilities USB type C might have added, one new risk is how it affects usage and behavior.

For decades, display outputs have remained mostly just outputs. As systems are smaller and more integrated, and as display outputs require higher and higher bandwidths, manufacturers keep combining display outputs with other ports. The combined Thunderbolt/Displayport connectors on nearly all Macs are an example – you cannot plug in a display adapter without using a port that also exposes a much more vulnerable interface (PCIe). The best solution is to carry and use only a trusted known Displayport to VGA/HDMI/Displayport cable and not trust others.

USB type C causes this same problem again, but worse. Every USB type C device needs to support a series of interfaces, implemented in multiple chips, regardless of whether any standard USB functionality is needed. The result is an $80 Apple USB type C to HDMI adapter the only way to get display output form a Macbook – that comes complete with a suite of progammable chips and pads for a JTAG header. At the lower end of the spectrum are clones that have no experience or desire to take hardware precautions.

An inexpensive USB type C to HDMI adapter available for $20 includes two programmable NXP chips for power delivery and to support the USB Billboard Device – the USB device that shows up when you plug a USB type C display adapter into a host that doesn't support display over USB type C. On this particular device, the NXP USB microcontroller has Device Firmware Update enabled. Without opening the case, the micro controller can be reprogrammed to function as a keyboard, mass storage device, or any other USB functionality.

The result is, with USB type C, even basic devices have reached a level of complexity that normal end users can’t fathom or be expected to fathom.

Summary and Conclusions

In summary, the 5 implants show each cost only a few dollars. They can be implemented by anyone with basic electronics skills. Custom malicious hardware implants are no longer exclusive to nation-state actors, and belong in the threat model whenever both supply chain and physical security can not be guaranteed. As shown, it only takes a matter of seconds to insert an implant that can completely undermine the security of a system.