Keywords: security, encryption, tampering, secure, microcontroller

APPLICATION NOTE 5937

FUNDAMENTALS OF ELECTRONIC SECURITY: TAMPERING WITH THE EASY TARGETS

By: Ben Smith, Software Manager

Abstract: More and more frequently, computer-based systems store valuable data and manage the flow of valuable commodities. If an opponent can gain control of the computer that touches this valuable data, they can access your private information, steal your money, or fraudulently access goods and services. In this article we look at tampering and what we can do about it.

A similar version of this article appeared October 5, 2014 on *Embedded*.

Introduction

In 1964 International Business Machines (IBM®) announced its System/360. It was by no means the first computer, but it was one of the most popular, with thousands delivered between 1965 and 1978. It was considered state of the art when introduced, with medium-range systems sporting 128KB to 512KB of memory and a throughput of about 0.3MIPS.

The system was large—multiple cabinets contained auxiliary storage, communications equipment, and peripheral components—and required specialized power and cooling. These machines were tended by a highly trained cadre of computer operators, and unless you had a good reason to be there, you did not easily gain access to the “machine room.” These rooms had the very best security: multiple physical locks and humorless men guarding the door.

These machines managed millions of financial transactions, making them a ripe target for criminals wishing to tap that flow of money. But tampering with these machines was virtually impossible. All transactions were secure because the machines themselves were physically secure within glass-walled machine rooms.

Fast-forward fifty years. The smallest bit of personal technology now has computing horsepower and communication capabilities that dwarf the power in that historic IBM machine room. And today this computing power is not just in the most obvious candidates, cell phones and personal computers. Contemporary televisions, sales (POS) terminals, utility meters and thermostats, portable medical devices, and smart kilowatt-hour meters all contain computers and communication facilities. Many of these products directly or indirectly touch a flow of money; all manipulate private, personal data. And did you notice? There is not a security lock or an armed guard in sight.

And that makes them all amazingly attractive targets.

More and more frequently, computer-based systems store valuable data—think about the medical records stored on systems belonging to your health care provider, or your credit card numbers in transit from a retailer to your bank, or the value remaining on a gift card. The systems also manage the flow of valuable commodities—think now about your smart electricity meter or the cable box that manages access to programming. In each of these cases, if an opponent can gain control of the computer that touches this valuable data, they can access your private information, steal your money, or fraudulently access goods and services.

But even if the smart device stores no valuable information and does not control the flow of valuable resources, there is still a concern: the value of the internal intellectual property (IP) that runs the device. If an attacker can reverse engineer the internal programs, they can produce a competing product without expending the money for development. It is important to protect your IP base to avoid giving your competitors an unfair edge.

While remote cyber attacks seem to receive much of the press coverage, many of the most effective attacks use a much simpler method: tampering. Recall the examples above. Someone might be tempted to use an unauthorized authentication card with the cable box to receive free movies. Another might tamper with the electricity meter so that it underreports energy usage and lowers a monthly bill. Yet another might tamper with a credit card reader so that it reveals private card numbers and data. All of these attacks involve someone with physical access to the device using relatively low-tech means to circumvent the security measures embedded in the device.

In this article we look at tampering, not just what we can do about it, but when it makes sense to do nothing.

Resistance Is Not Futile

At a minimum, devices that purport to be secure should be *tamper resistant*. That is, the designer and manufacturer of the device should take at least minimal steps to deter the curious and the casual hacker. These steps include virtual barricades in the hardware, including using nonstandard fasteners, plastic or metal welds in construction, or glue in assembly. This means, of course, that servicing the device can also become more complicated, but remember that the focus here is on security.

But ultimately it does not matter how difficult you make it to pry open a product. a determined opponent will find a way in. When that happens, there are four possible responses to a tamper attack, all directly related to the value of a secure device and its protected data.

- **Destroy the device**. This may be the best and most straightforward option, particularly if the device is inexpensive but the data it contains has great value. For example, if a credit card terminal detects that its case is being opened, it may rapidly destroy any secret information inside, including the cryptographic keys that decrypt its operating software. Then, when next turned on, it will not be able to function because its encrypted code store is useless without access to the keys required to decrypt it. Any device that destroys its own ability to function when it senses a tamper event is about as close to being *tamper proof* as it can be.

To "repair" the damage, one must replace the device, but presumably at a relatively modest cost compared to the recovery cost if sensitive material had been lost.

- **Send a notification**. If a device is connected to a network, a message is launched to a supervisory computer on the network at the first sign of a security breach. The supervisory computer then notes the device's identity and removes it from the list of active devices. This kind of device is called *tamper evident*: it cannot prevent a tamper event, but it can certainly make a network manager aware of the tampering.
- **Activate a physical indicator**. If a device requires physical interaction with a person to do its job, an automatic indicator can alert the user that the device is no longer trustworthy. For example, there are tamper-evident seals on medical supplies that provide inexpensive but effective security. If broken, they alert the user (i.e., the medical professional) that the device's integrity has been compromised and that it should be discarded.
- **Do nothing**. It may seem strange sometimes to allow outsiders access to our secret information. In fact, in the right circumstances not everything has to be locked down tightly. If device's value is low and if the consequences of losing control of its data are minimal, the simplest reaction may simply be to do nothing. Absent a financial incentive to tamper, attacks against low-value targets often stem from curiosity or accidental damage, and do not warrant recording or action.

So Many Vectors, So Little Time

So far, we have discussed physical tamper events with a secure device, literally opening it to extract useful information or modifying it to cause a malfunction. But there are more devious ways to breach security. One can tamper with a device without even touching it. How?

Most electronic devices are sensitive to environmental factors, including temperature, humidity, power conditions, or electrical or magnetic fields. Expose an electronic device to environmental conditions beyond its rated limits, and it will likely misbehave—and possibly in ways that benefit the attacker.

How does this work?

- An attacker discovers that one particular component in a device is temperature sensitive. Then exposing the device to elevated (or depressed) temperature makes it fail. If the device's startup code does not properly anticipate such a failure, it can drop back to a command prompt, thereby giving an opponent root access to the device.
- An attacker discovers that, by placing a strong permanent magnet around the smart energy meter that measures electricity usage at their home or business, the core of the meter's power transformer becomes saturated. When the core is saturated, the transformer can no longer power the metering components, even while the consumer continues to use electricity. Then, just before the meter is to be read, the attacker removes the magnet. The meter begins working again. When the meter reader arrives to record the energy usage, everything looks normal, but the meter has recorded energy usage for only a few days and not the entire month!
- An attacker discovers that by quickly cycling power, a cable box can be made to malfunction. Malfunctioning when performing repeated power on-off cycles is not uncommon because power-on reset (POR) is a relatively complex operation. If some components sense the power-fail condition

while others do not, an unexpected operational state may be forced. If the cable box enters, for example, a factory test mode because of the POR failure, the attacker may be able to read secret information from the device and gain open unlimited access to the video content.

There is a common pattern in each of the above instances of what can be called “indirect” tampering: design mistakes in the original equipment.

In the first case of temperature fluctuation, the design error was in the system response to a component failure. When the temperature moved beyond prescribed limits and the device sensed a failure, the designer had the device drop to a debug prompt—which is exactly what the attacker wanted! In the second case of the utility meter the designer failed to anticipate how large, powerful magnets might disrupt meter operation. In the designer’s experience transformers “just work.” The third case of power cycling is interesting because most engineers tend to internalize the state of the device that they are designing or debugging. They know what kind of settling time is required from the time the power switch is turned off to the time the device is ready to turn on again. Knowing this, they treat their systems much more “nicely” than the average consumer might… much less the way a malicious hacker would.

Design engineers typically think about the best-case scenario for their end application. Their focus is delivering a working product, and quickly. The main concern is, how can I make my product work in the shortest time, at the least cost, and with the most robust features? But the attacker is thinking differently. The attacker’s main concern is, how can I cause the product to malfunction so I can get what I want?

There is a lesson here: every team designing a secure product should have at least one person on board who can think like a criminal!

Plugging the Holes

We must accept a reality in today’s electronic world: your opponent will have physical access to your device and will attempt to take advantage of any weaknesses that they discover. There will be no one to stop them from doing this, so what steps can you take to reduce the threat to strengthen the embedded security?

Here are some practical suggestions.

- **Get realistic**. First, as mentioned above, engineering teams need to change their mindset. In particular, someone on the engineering team needs to think like an attacker, and this thinking needs to extend even to the earliest phases of engineering. Remember that your attacker has the advantage of time and stealth, but you define the playing field. So do not make it easy.
- **Define a security boundary**. That is, draw a box in your design. Anything that must be kept secret (financial data, client information, or proprietary IP belongs in the box. This is the box that your opponent will be trying to breach, and it is what you must protect.
- **Make it hard**. If your opponent wants access so badly, make them work harder. Use proprietary closures, nonstandard hardware, and even glue and welds, if necessary. All this will not deter the most determined attackers, but it will keep casual spectators from taking a peek. Then use electrical countermeasures: switches that sense a case opening or when the circuit board is being removed. Consider a serpentine mesh around, at least, the secure areas of the design and, preferably, the entire interior of the product. Then breaking the serpentine or sensing an opening switch can trigger a response.
- **Anticipate problems**. Is any component sensitive to magnetic fields? Then put a magnetic field sensor in the design. Report when the field exceeds some threshold. Is operating temperature a

potential problem? Then include a temperature sensor and take action when the temperature goes outside a safe range.

- **Let someone know**. Think in advance about how to alert responsible parties if there is a breach. If the device requires a network connection for its basic operation, then sending a message is an easy option. But what if there is no network connection? Lights, sounds, or obvious operational indications are all options. Silently leaking sensitive information is never a good idea.
- **If all else fails, become a brick**. This is, admittedly, not a viable alternative or the optimal solution for many applications. Nonetheless, at some point the best security may be to have the product shut itself down in an unrecoverable way. This is especially true for financial terminals, where an active, robust tamper detection with fast key destruction is a practical reality for these devices.

Conclusion

The easy targets in any secure system are the physical points of interface between the system and the rest of the world. By opening the box, by tapping the communication links, or by exposing the device to environmental extremes, the device may be coaxed into giving up its secrets. Then the attacker can take advantage of the malfunction. The job of the designer is to anticipate these attacks, plan for them, and understand the consequences of a successful breach.

So, we know what security really means. With security targets identified, we seal off a product from a physical attack, at least as best we can. We implement a strategy and tactics in the product to thwart the most determined opponents, What is left? Nothing…unless, of course, your product needs to communicate with the outside world. That's the realm of communications security which we will discuss at length in our next article.

IBM is a registered trademark and registered service mark of International Business Machines Corporation.

Related Parts		
MAX71637	Energy Measurement SoCs	Free Samples
MAXQ1050	DeepCover Secure Microcontroller with USB and Hardware Cryptography	
MAXQ1850	DeepCover Secure Microcontroller with Rapid Zeroization Technology and Cryptography	Free Samples

More Information
For Technical Support: http://www.maximintegrated.com/en/support
For Samples: http://www.maximintegrated.com/en/samples
Other Questions and Comments: http://www.maximintegrated.com/en/contact

Application Note 5937: http://www.maximintegrated.com/en/an5937
APPLICATION NOTE 5937, AN5937, AN 5937, APP5937, Appnote5937, Appnote 5937