NCC Group

Secure Device Manufacturing: Supply Chain Security Resilience

Table of Contents

- Table of Contents
- Executive Summary
- Introduction
- Supply Chain Overview
  - Factory Process and Ownership
  - Data Flow
  - Shadow Supply Chain
- Attacker Tools and Techniques
  - Chip-Off
  - Leaked Software/Tools/Schematics/Data
  - Third Party Tools
  - Open Source Research
    - Jailbreaking Community
  - Stolen Network Access
  - Vulnerabilities and Exploits
  - Common Components
- Benefits of Parasitic Manufacturing
  - Competitive Cost Pressure
  - Access to Untapped Markets
  - Out-of-Warranty Repair
  - Market Research
- Legal Issues
  - Prosecution
  - Employee Rights
  - Contractual Obligations
- Mitigations
  - Secure Product Design
    - Cryptography
    - Debug and Manufacturing Interfaces
    - Software Exploit Mitigations
    - No Shared Secrets
    - Identifier Signing
    - Prevent Flash Replacement
    - Code Signing, Secure Boot, and Certificate Pinning
  - Manufacturing and Repair Process
    - Identifier Prediction
    - End-to-End Encryption
    - Physical Security

- Secure Scrap
- Factory Desktop and Network Hardening
- Privilege and Role Separation
- Protective Monitoring for Abusive Patterns
  - Data Sources
  - Abusive Patterns
    - Mismatch in Number of Units Ordered/Shipped/Activated
    - Activation Data
    - Identifier Cloning
    - Same Device in Multiple Locations
    - Identifier Guessing
    - Sequential Identifiers
    - Unit Theft
    - Component Theft
    - RMA of Unknown Units
    - Identifier Mismatch
    - Unknown Ethernet MAC Address/MAC Prefix
    - Multiple MAC/IP Address Combinations Per Day
    - Factory Credentials Used From Incorrect Site
    - Unknown IP Addresses
    - Operations Performed From Wrong Station Type
    - Out of Range Identifiers
    - High Volume Stations
    - Activity During Quiet Times
    - Station-to-Station Communication
    - TCP TTL Out of Range
    - Unknown Operating System Platform
    - High Number of IMEIs Per SIM Card
    - Use of Cloned SIM
    - Obsolete Devices in New Production
    - Missing Scrap
  - Contractual Damages and Reverse Billing
  - Legal Disclaimers on Software
  - Stolen Devices, GSMA Blacklist, Kill Switch Legislation, etc
- Investigation Costs
- Summary
- NCC Group Manufacturing Security Services
- Acknowledgements
- Acronyms

Executive Summary

Today the production of hardware devices involves multiple suppliers at various stages of the production and support life-cycle. There is no electronics manufacturer who manufactures every single component of a device in their own factory. As such, and has been demonstrated, these hardware and manufacturing supply chains introduce risk that threat actors could gain an opportunity to defraud, steal, or otherwise undermine the security of the produced electronic devices.

For these reasons it is important that hardware brand owners understand this risk and take appropriate steps to mitigate them. The appropriate level of mitigation that any organization should take will depend on the expected level of security of the device, the sensitivity of the data that it will process, and the capabilities of the expected threat actors.

Introduction

This whitepaper is primarily concerned with the following questions:

- How can I build a secure product that my customers can trust when I do not trust my factory?
- How do I limit the number of counterfeit devices in the marketplace?
- Can the grey market be of any benefit to my company?

Shadow supply chain, parasitic manufacturing, grey markets; these are all terms used to describe undesirable third party activities related to the production of electronic devices. Some searching will reveal many public examples[1] [2] [3] [4] [5] [6]. There are no doubt many many examples that go unreported because of the brand damage that might follow. These issues are serious enough that the US government, through its DARPA research arm, has created a related project called the DARPA Trust initiative[7] [8]. Furthermore there are several standards that may apply to you depending on your industry namely AS 6081, AS 5553, AC 00-56, and NIST 800-164.

The reason these problems exist is because you have a successful product that people want to purchase. Unfortunately, there will always be some market need that you are failing to fulfill, no matter how small. This unfulfilled need is where the grey market exists, and if large enough, it

[1] http://www.cbc.ca/news/politics/story/2013/01/09/f-vp-weston-hercules-counterfeit-chinese-parts.html
[2] http://www.telegraph.co.uk/news/newstopics/politics/lawandorder/3173346/Chip-and-pin-scam-has-netted-millions-from-British-shoppers.html
[3] http://www.abovetopsecret.com/forum/thread350381/pg1
[4] http://www.clonedinchina.com/2010/06/blackberry-9800-got-cloned.html
[5] http://www.eetimes.com/document.asp?doc_id=1261714
[6] http://zeptobars.ru/en/read/FTDI-FT232RL-real-vs-fake-supereal
[7] http://www.darpa.mil/Our_Work/MTO/Programs/Trusted_Integrated_Circuits_%28TRUST%29.aspx
[8] http://www.bunniestudios.com/blog/?p=2037

will flourish. The need might be simple price competition, out-of-warranty service, an untapped market segment, or some unintended application of your product that you have failed to satisfy. A good market research team can identify many of these before the grey market does, and there is often a business decision to be made as to whether and when you choose to satisfy the need. If you decide not to, and the need is great enough, then there will be real money to be made by someone else should they find the means to do so. A budding grey market related to your product is a sign of success, not of failure. As the old adage goes: Imitation is the sincerest of flattery. You should look at these problems in positive light, you are doing the right things, and you are going to be successful. As your success grows, so does the parasitic manufacturing that will grow in your shadow. This parasitic manufacturing needs to be accounted for as a cost of doing business. The more successful the product, the higher the cost of keeping the parasitic manufacturing at bay. It is unlikely that you will find a silver bullet solution that will stop it completely, and there are some distinct benefits that can be achieved by allowing it to exist in a controlled and monitored fashion. Of course, if left to fester unchecked, it can be the complete undoing of your business as it was for companies like Nortel. In the case of Nortel their software intellectual property was stolen and functionally similar hardware was produced by a third party which was in part powered by this software. Clearly a balance needs to be struck, and collecting data is the first step to finding the right balance for your business.

This whitepaper will cover a number of interrelated topics as outlined in Figure 1. The intended audience is anyone working in those areas with an interest in the security of their products. It should be especially of interest to anyone looking to expand their supply chain into the so called “BRIIC countries” (Brazil, Russia, India, Indonesia, and China) where intellectual property laws are somewhat weaker.

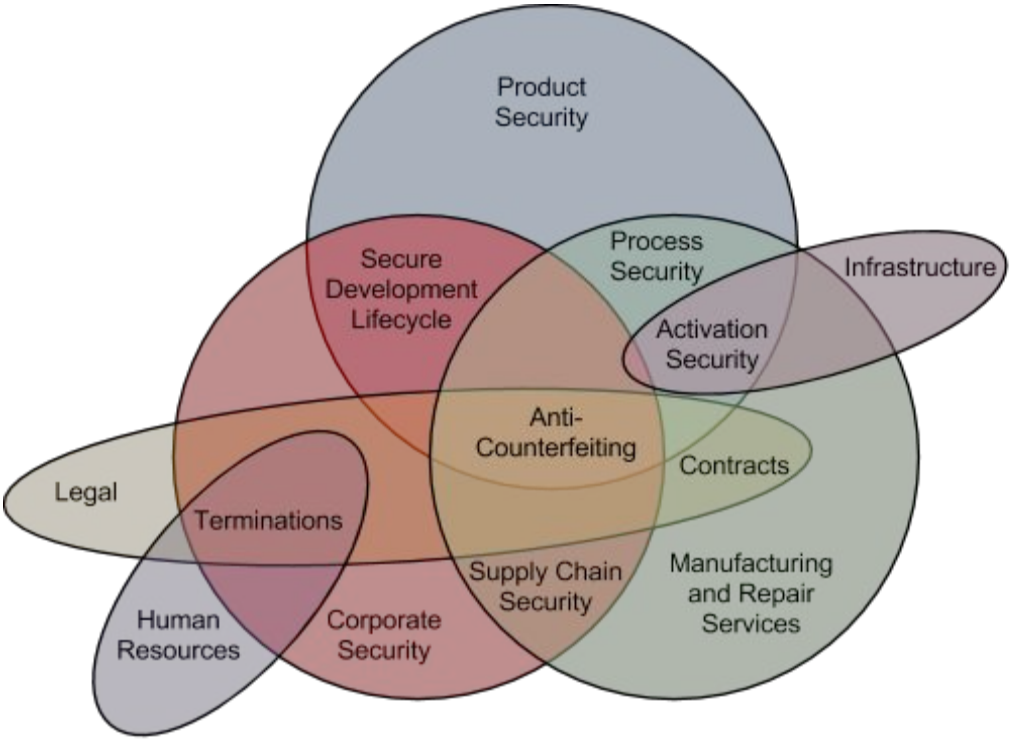


Figure 1: Disciplines related to grey market activity

Because of the diverse skill sets required to tackle these issues, it is quite common for companies to have an immature reactive approach to these concerns. Bringing the right team members to the table to build proactive monitoring and response processes, procedures, and systems is a challenge, but the cost of a reactive incident response can be much greater.

It should be noted that outright product forgeries are out of scope here. No technical solutions are possible for outright forgeries, and those are best covered by the import laws of the countries in which you choose to operate. Only those counterfeits that touch your supply chain or operational infrastructure in some way are in scope. For these, we can collect data and implement technical countermeasures to limit the problem.

Supply Chain Overview

Factory Process and Ownership

Figure 2 shows a typical factory process. Physical components are received by a factory from a variety of vendors, soldered to a board, tested, calibrated, assembled into housings, tested again, configured for final shipment, and finally shipped to a customer for field activation.

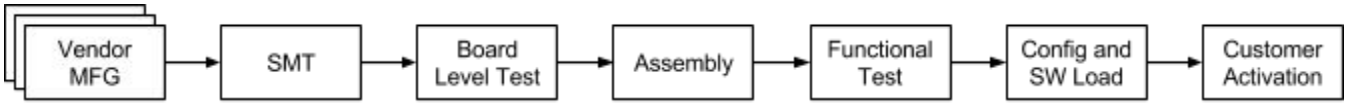


Figure 2: Typical Factory Process

This outlined process is intended to be generic, and any individual step may be broken down into further steps, or combined into larger steps. The same basic process will no doubt apply to any consumer electronic device. To further complicate matters, the manufacturing steps are most likely to occur at a contract manufacturer (CM) or outsource device manufacturer (ODM) partner. Due to cost pressures, this scenario applies to all but the largest and most vertically integrated companies. Indeed, it frequently happens that each step is performed by a different CM/ODM partner. This makes ownership of security issues difficult to assign.

Data Flow

Figure 3 shows the data flow associated with a typical factory process. As physical devices move through the various processing steps, metadata such as logs and test results are collected and fed upstream. Eventually, this metadata, either whole or in aggregate, makes its way to the OEM's data systems. Ensuring the right level of detail in this metadata is critical for detecting security issues in the supply chain.

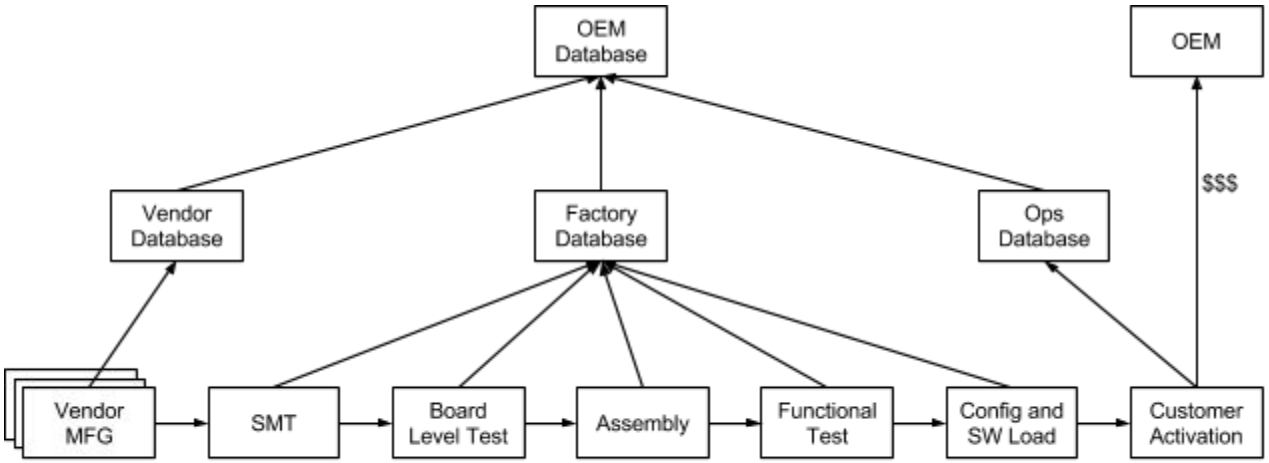


Figure 3: Typical Factory Dataflow

Shadow Supply Chain

Figure 4 shows a parallel supply chain. It is common for a counterfeiting operation to acquire components wherever they can be sourced. This includes used devices, stolen devices, scrapped devices and components, scrapped prototype devices and components from either the manufacturing line or the vendors (or vendors suppliers), surplus devices from sales channels, illicitly imported devices, and alternate (often lower quality) vendors. If you ever take

a tour of the Shenzhen market, you will be astounded at the vast selection of components that can be purchased in both high quality and high volume[9].

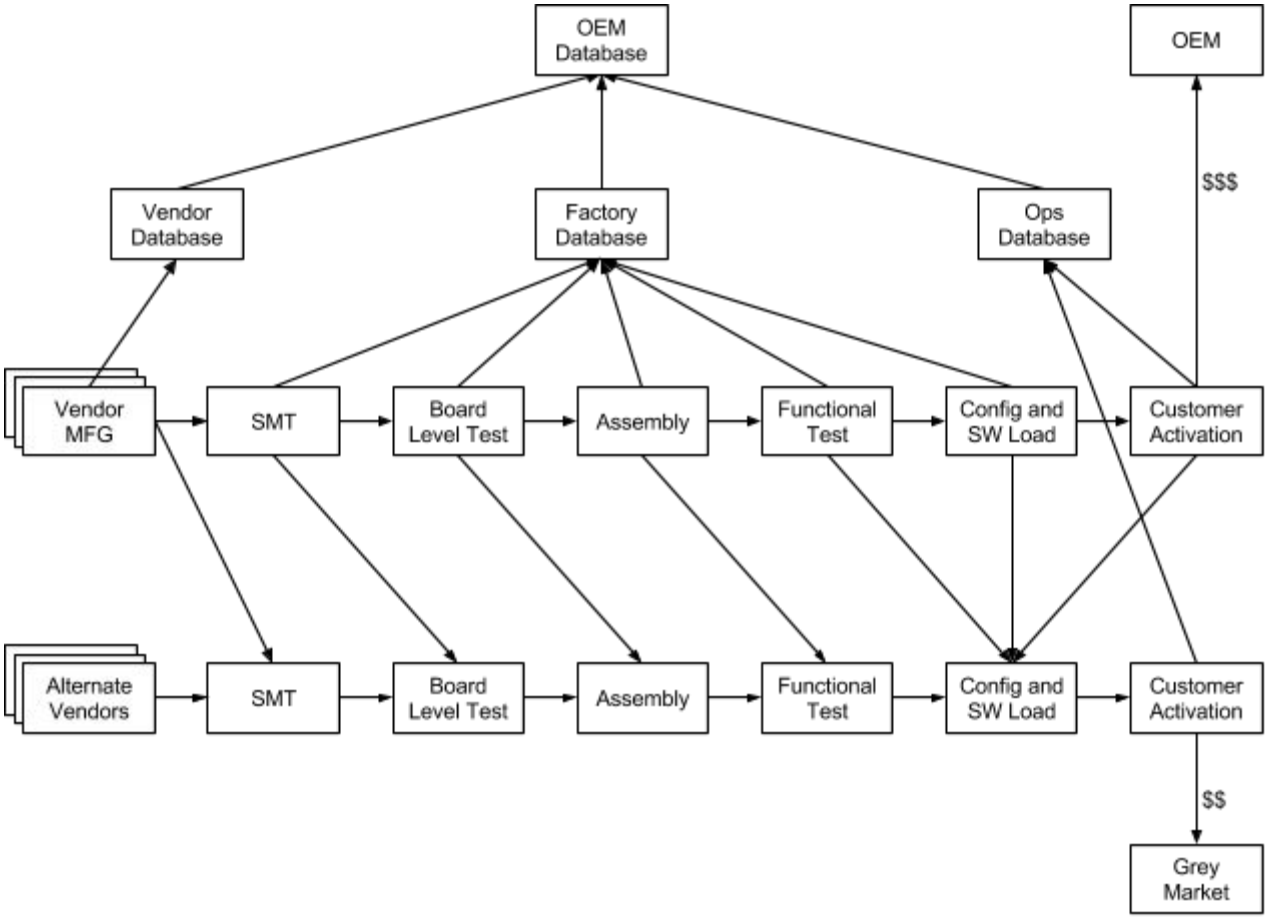


Figure 4: Grey Market Supply Chain

Again, all of this exists for one reason: you have a successful product, and there is money to be made by mimicry. It is even possible, and indeed common, for portions of this shadow supply chain to be in the same factory and use the same materials and stations as the legitimate products before being diverted. This scenario is often referred to as the "Sunday run" scenario, in which an operator in a factory will over-produce, but under-report, thus allowing a grey market profit at the expense of the OEM. As mentioned in the introduction, we are concerned primarily about shadow supply chains that have some form of interaction with the OEM's systems. It is often possible to construct a completely independent process to produce counterfeit devices that in no way touch the legitimate supply chain and once activated in the customer's hands, do not touch the operational infrastructure of the OEM. These types of counterfeits are out of scope here. If these devices do not touch your systems, then they are truly beyond the control of the technical measures you may put in place. Fortunately, this type of counterfeit device is

[9] http://www.bunniestudios.com/blog/?p=147

the most expensive to produce, and therefore the least common. These are best handled by border import controls and anti-counterfeiting laws. Such laws are common around the world, however they are known to be weak specifically in the BRIIC countries. Even still, less than 5% of counterfeit imported goods are caught at the border[10].

Attacker Tools and Techniques

Chip-Off

In all but the smallest volumes, it is very economical to remove a flash memory, reprogram its contents, and re-solder it to the board. This technique is generally referred to as “chip-off”. Many firmware developers still incorrectly see physical attacks as expensive, difficult, and unlikely. The reality is that this type of attack against consumer device software has been cheap, easy, and commonplace since at least 2004, and probably much earlier. This operation can take as little as ten minutes for a skilled repair worker, and if the original part can be reused, the cost is negligible. On smartphones, this attack has been used for all sorts of purposes, including replacing the device identifiers (such as IMEI), disabling the subsidy lock, and enabling OEM debug features to increase attacker privileges. Care should be taken when designing any security-related feature to account for this style of attack. Sensitive data should be encrypted or signed as appropriate, so that modification is not possible in the hands of an attacker. From a grey market supply chain perspective, this is often the easiest means to refurbish and resell second-hand devices. These second-hand devices can easily be sourced in bulk from surpluses of obsolete devices in the sales channel, or from less legitimate aggregations of stolen devices.

Leaked Software/Tools/Schematics/Data

In general you should expect that any software you provide to your factory, third party or not, will eventually be leaked to the Internet. There are many vendors (often of questionable legal standing) selling collections of such tools for aftermarket repair of all makes and models of consumer devices such as smartphones. These have been widely available since at least 2007. It is often surprising how much of this is freely available if you know where to look. The components we are talking about here include:

- device bootloaders
- device firmware (operating system and applications)
- device factory software
- factory tools such as flashing tools and test station software
- any factory tools to access backend databases, permission systems, and web services
- schematics, component data sheets, and other hardware information
- device unlock codes (both bootloader lock and subsidy lock)

[10] http://stopthedrugwar.org/chronicle/2011/dec/05/ny_times_dea_launders_mexico_dru

Third Party Tools

There are many tools available that are created by third parties. Some of these may be based on your own leaked tools[11], but many are created from scratch[12], available from your vendors[13], or even commercial products created with or without your help[14]. These tools may consist of hardware, software, firmware, or any combination thereof. As an example, so called "unlocking cables" are very common in the smartphone industry[15]. Any firmware portion of legitimate tools that runs on your product needs to be locked down as part of your device design. Code signing is your primary defense. Beyond that, these tools will rely on exploits for their privileges.

Open Source Research

Web searching can reveal much publicly-available information about your product, whether you like it or not. There are websites that provide freely downloadable schematics[16], companies providing complete product tear-down and reverse-engineering services[17], companies selling tools to exploit your devices[18], and scholarly researchers just doing their thing[19]. This is only the stuff on the surface web. There are vast riches of data that can be had in darker corners of the Internet where the counterfeiters are active. When doing defensive research it helps to speak Russian and Chinese. These groups of attackers are not isolated. They belong to a global community of aficionados that range from the curious, to resellers, to convicted organised crime members. Interaction among them is primarily online and anonymous, through marketplaces and other information exchange sites. It is these loosely-organised groups that are your adversary, and not some traditional hierarchically-organised company.

Jailbreaking Community

The jailbreaking community deserves special mention. They are primarily concerned with gaining privileged access to devices for personal use. These intelligent folks will routinely go to great (often heroic) efforts to defeat the access controls that manufacturers build into their products. This is a very niche use, and does not generally pose a threat. That said, they tend to make all of their work public, and once in the public sphere, it can be used by other actors with less innocent goals. Monitoring the jailbreaking forums for vulnerabilities and exploits is key to keeping your devices patched. Some manufacturers choose to grant privileged access to local users through controlled methods[20], which avoids encouraging these communities to

---

[11] https://chimeratool.com/
[12] http://www.smart-clip.com/
[13] http://www.getdriver.com/category/other/tools
[14] http://www.cellebrite.com/Mobile-Forensics/Products/ufed-touch
[15] http://www.fonefunshop.co.uk/Unlocking/unlocking.htm
[16] http://service-schematics.ru/catalog/apple/
[17] http://www.techinsights.com/
[18] http://www.cellebrite.com/Mobile-forensics
[19] http://natashenka.ca
[20] http://www.motorola.com/unlockbootloader

develop the more dangerous multipurpose exploits[21]. You will need to decide if this fits with your business model.

Stolen Network Access

Because many parts of a secure supply chain rely directly or indirectly on real-time network communication with the OEM, getting access to this network is a priority for the attackers. This can be accomplished through any number of methods, but the simplest and therefore most common are through malware infection of a factory computer, through the use of rogue network devices such as USB cellular modems, or some combination of the two. Factory workers have been offered several times their hourly pay just to make sure this equipment is installed and working. This type of attack was common at least as early as 2010. The insider attack possibilities here are innumerable. Great care should be taken to limit the access these employees, subcontractors, and partners have.

Vulnerabilities and Exploits

Vulnerability research and exploit development is a complex, time-consuming, and therefore expensive task. Many markets exist for the sale and purchase of software exploits, and if there is sufficient money to be made, then counterfeiters will gladly pay for these when there are no easier options. Exploiting a firmware vulnerability is a common technique for counterfeiters, at least as early as 2011, to gain privileged access to devices in order to reconfigure them for resale. Special care needs to be taken when designing debug features and access controls, as these are priority targets for such exploits, since they can so easily be co-opted to elevate privileges. This problem is only exacerbated by the component monoculture that exists in many consumer electronics. A solid and secure software development process, combined with a robust set of exploit mitigations, is your best defense. Unfortunately, there are still many embedded systems that lack the hardware capabilities and compiler-level defences required for even the most basic protections. You should consider this in your early design choices because fixing it later is exponentially more difficult.

Common Components

Component monoculture caused by the success of a dominant industry vendor can lead to easier attacks on your devices. Even if you do a remarkable job with the security design of your product, if you rely on components or software from a vendor that also provides to another OEM who does not do such a great job, then you may find yourself vulnerable. It is common for attackers to find a vulnerability on one device, and then use that as a development platform to create exploits useable on a wider range of products all sharing a common vendor.

[21] http://www.tgdaily.com/security-features/50953-drive-by-ios-jailbreak-sparks-security-concerns

Benefits of Parasitic Manufacturing

As previously mentioned, these issues are not necessarily all negative. There can be some benefit if handled correctly.

Competitive Cost Pressure

The most obvious effect of a grey market is price undercutting. It is hard to compete against a product that is exactly the same as yours, but cheaper due to a grey (or even black) market supply chain. The underground economy acts as a sort of pressure relief valve of the regular economy[22]. The underground economy can help to dictate the upper bound at which you can set your product price. As your prices go up, so too does the size of your grey market, as more and more people are willing to take risks necessary to capture the price differential. Equivalently, you can help limit the size of the grey market simply by keeping your prices reasonable. Price competition is unquestionably good for consumers.

Access to Untapped Markets

Setting up a global sales channel is massively expensive. If your product is highly desirable, you may find your products making their way to countries where you have no physical presence. This can be of particular benefit in countries with strict import regulations or tariff structures. In such situations, the grey market can be of great benefit both in gaining an initial toehold, and in gauging the demand in those markets. There is clear business intelligence to be gathered by monitoring the volume of devices activating in countries where you do not actually sell them. Once the demand is known, stemming the flow of smuggled devices can be as simple as opening a few-well placed stores.

Out-of-Warranty Repair

There comes a point in the lifecycle of a product where the cost of supporting it will outweigh the profit it generates. If your product is a good one, with a long life, then secondary markets for it may thrive long after this point. Global repair services are expensive, and if you choose to discontinue support for a product, then you will need to make a choice: pay someone else to support it ($), replace the in-market devices ($), or allow the grey market provide these support services.

Even devices that have limited or no secondary market can be a concern as they are cannibalised by the grey market supply chain for components. It is quite common for OEMs to iterate designs, reusing the same components for years. Old devices can thus be a good source of components when counterfeiting a newer model.

Market Research

The business intelligence that can be gleaned from the grey market activities is invaluable. It will show which markets you should expand or contract, help highlight the lifetime of your

[22] http://books.google.ca/books?hl=en&lr=&id=CmbKloygryEC

products, demonstrate novel use cases and market segments, and identify the insane prices that early adopters are willing to pay to get the latest device models[23].

Legal Issues

Prosecution

Cyber-crime and counterfeiting laws are often subjective, new, or untested, so prosecuting perpetrators may not always be successful. Assigning individual blame when the perpetrator(s) may be a well-organised and compartmentalised criminal gang can be a difficult and fruitless exercise. Even when prosecution is a clear option, it may be a never-ending, and thus expensive, game of whack-a-mole, as new players step up to fill the void left by the previous counterfeiter. Furthermore, legal cases in the courts become a matter of public record, and depending on the circumstances could have a public relations and brand impact. Because of these concerns, we are led to recommend technical solutions rather than a legal course of action in all but the most egregious cases.

Employee Rights

Many mitigations essentially boil down to some form of monitoring. It is worth noting that many countries have privacy and employee rights laws that cover either your employees or those of your manufacturing partners. This may or may not impact your ability to implement some of the mitigations mentioned below. Working carefully with your legal team can help navigate these murky waters.

Contractual Obligations

Attack patterns show up in the data analysis only if you have all the correct data. Data sharing can become a contentious issue if not planned for in advance. In the case of smartphones for example, data from the network carriers, OEMs, component vendors, manufacturing partners, repair partners, GSMA, cloud service providers, and others may all be required to be correlated before you can easily confirm a pattern as malicious. In too many cases these entities will not be able, or not be permitted by their pre-existing contracts or by SEC fair disclosure regulations, to share their internal databases. This subject is best broached as part of the initial contract discussions, since dealing with it as a reactive measure is likely to be much more difficult.

Mitigations

As has been mentioned already in this document, there is much that can be done about the issues being discussed. A well rounded and secure product design, manufacturing process, monitoring system, and incident response process are all fundamental requirements of this strategy. What follows are some specific recommendations about how to counter real-world observed attack patterns.

[23] http://www.cnn.com/2015/01/13/asia/china-iphone-smuggler/

Secure Product Design

Designing secure products that are not easily modified is not only critical to your customer's trust in you. Secure product design is critical to preventing your devices from being used in the grey market supply chain as well. There are many things you can do to make sure you have a securely designed product.

Cryptography

Proper use of strong cryptography is essential to protect data both at rest and in transit. Avoid weak cryptographic algorithms. When choosing an algorithm look for algorithms without known weaknesses[24] [25]. When choosing a key size, anything less than 128 bit symmetric key equivalency should be considered weak[26]. This means at minimum you should be selecting algorithms like AES128 (symmetric), SHA256 (hash), and ECC256/RSA4096 (asymmetric). At no time should you consider rolling your own algorithm. Similarly, the implementation of even the standard algorithms can be a daunting challenge prone to subtle failures. Instead, consider the many well maintained and supported cryptographic libraries that are available under a variety of licensing schemes.

The algorithms are only part of the picture. The use of cryptographically sound entropy sources is mandatory for almost all crypto-systems. Failure to do so can yield disastrous results[27] [28]. Key management can also be a complex task. Safely generating, storing, and using keys on a device that is potentially in the hands of an attacker can be a challenge. Careful design and implementation review by an expert is a wise idea.

Debug and Manufacturing Interfaces

As previously mentioned, debug interfaces are a common vector of attack. These exist on nearly all devices so that developers can debug issues with both prototypes and field returns. Less commonly known, but a favourite among counterfeiters is the factory test and configuration interfaces. All of these interfaces are intended only for the OEM development, manufacturing, or repair teams. Though they offer a high degree of privilege within a system, they need not do so without authentication. Both user and OEM authentication should be required before allowing anyone to use these interfaces. Furthermore these interfaces should be role separated from each other. Developers will need more privileged access than factory workers. Barring a strong authentication mechanism (with an audit trail!), these interfaces should be completely disabled. Both software debug interfaces and hardware level debug interfaces like JTAG and UARTs must be scrutinized.

---

[24] https://eprint.iacr.org/
[25] http://safecurves.cr.yp.to/
[26] http://www.keylength.com/en/
[27] https://eprint.iacr.org/2012/064.pdf
[28] http://www.loyalty.org/~schoen/rsa/

Software Exploit Mitigations

If your devices contain any software or firmware, then you really need to consider implementing software exploit mitigations. Bugs will happen, and unless you have a robust set of mitigations in place, an exploit for that bug will soon follow. The list of common mitigations evolves, but at minimum you should consider including DEP, ASLR, heap and stack protections, privilege separation, and a host of others. Care should be taken during the hardware component selection to provide components that can support these mitigations.

No Shared Secrets

An attacker who invests the time and energy to defeat the security of a single device should not be rewarded with the ability to attack other devices at no additional cost. Embedding the same secret key in all devices is a terrible practice that occurs too often[29] [30] [31]. Shared secret keys are suitable only when you trust the eventual owner of the device not to dig into the hardware to extract these secrets. For consumer electronics, there is simply no place secure enough to support this model[32]. If secrets must be embedded, then consider the use of public key crypto-systems. The secrets can still be extracted, but now the attacker's costs will not scale so economically.

Identifier Signing

Every cell phone contains a unique International Mobile Equipment Identity number (IMEI) that allows the carrier to track and identify devices. While this number is supposed to be difficult to modify, and there are laws against this practice in many countries, these numbers are normally modifiable by special OEM tools for several good business reasons. To circumvent the OEM restrictions on modifying the IMEI, it became common around 2004/2005 for attackers to use a "chip-off" technique to replace the IMEI in the device's flash memory. In response to this attack, many OEMs began digitally signing the device identifiers as part of the manufacturing process. This not only prevents unauthorised users from changing the IMEI or moving it to a different device, but has the nice side effect of tying the manufacturing process to the OEM's network, which further raises the bar that attackers need to jump over. In addition to the IMEI, a phone may contain any number of other unique identifiers programmed during manufacturing: MEID, WLAN MAC, BT MAC, BB PIN, NFC ID, OEM serial number, SoC serial number, etc. The above discussion regarding the IMEI should be considered to apply equally to any other device identifier. For best results, the digital signature needs to cover ALL of the identifiers you want to

---

[29] http://www.contextis.com/resources/blog/hacking-internet-connected-light-bulbs/

[30] http://www.atmel.com/Images/doc2589.pdf

[31] https://hackaday.io/project/956-chipwhisperer-security-research/log/10108-aes-256-is-not-enough-breaking-a-bootloader

[32] https://www.blackhat.com/presentations/bh-dc-10/Tarnovsky_Chris/BlackHat-DC-2010-Tarnovsky-DASP-slides.pdf

protect, AND something that is baked into the hardware, such as a hardware serial number. Care should be taken to use a hardware serial number that is programmed into the silicon at the vendor at an early part of their process. It is common for smartphone processors to contain user-programmable fuses, and these may contain spaces designated for programming of serial numbers as well as other generic OEM-defined fuses. Since these are intended for OEM programming, it will often be possible to acquire unprogrammed parts, and this lets attackers defeat the identifier signing in some limited circumstances. Clever use of these fuses can be leveraged by the OEM to prevent the flash memory from being replaced by unauthorised users, and this may help limit the "chip-off" style of attacks.

Prevent Flash Replacement

As previously mentioned, "chip-off" techniques are a common means of modifying the memory contents of a device. Most modern flash memory devices have a write protection feature that can be used to permanently prevent portions of the memory from being altered. This write protection, combined with a processor serial number, and one-time-programmable fuses can be used in a number of ways to prevent the contents of the flash from being replaced in a chip-off style attack, and to prevent the flash chip from being replaced entirely with a brand new chip.

Code Signing, Secure Boot, and Certificate Pinning

Creating a counterfeit product such as a smartphone from scratch is a difficult task. Cloning someone else's product is certainly easier. Largely the types of counterfeiting seen revolve around hardware, and the software components are often reused from the OEM without modification. This is the one exception to the earlier exemption we made about outright forgeries, where you may have some small measure of technical control. To limit the behavioural differences between your legitimately produced products and the counterfeit devices, it is important that the software you provide has a chain of security checks. Each component of the device firmware should validate a digital signature on any subsequent component (often referred to as Secure Boot), AND on any previously executing component. This bidirectional signature checking forces attackers who wish to reuse your software to do a lot of extra work to defeat any software-based security mechanisms you may have implemented. It also helps ensure that any connection points to the OEM data systems or operational infrastructure remain intact, which will help for detection of supply chain abuses. The public key certificates used for these validation check should not be easily modifiable. This is often accomplished by burning the public key, a root certificate, or a cryptographic hash of it into one-time-programmable hardware fuses.

Manufacturing and Repair Process

When the attackers are in the factory, it is not enough to have a secure product design. You must also have a secure production and repair system. Careful thought needs to be taken when planning out a new manufacturing or repair facility, especially when outsourcing to a CM

or ODM partner. It is much more difficult to fix things after all of the contracts have been signed than it is to get things right up front.

Identifier Prediction

Simply making up random device identifiers is not always a valid strategy for attackers. Network operators can easily identify and block devices with invalid identities. Finding valid combinations of device identifiers is therefore a goal of counterfeiters and there are several strategies for this:

1. reuse identifiers from legitimate devices that are old, broken, or obsolete
2. steal lists of valid identifiers from the OEM or factory
3. correctly guess what identifiers will be used in the OEM factory

#1 certainly happens, but is costly as it supposes a supply of physical devices from which to "borrow" the identities. #2 is the simplest approach, but involves access to leaked data. #3 is usually first reported by carriers when they try to activate a new device only to discover that the IMEI (or other identifier) is already active on the network in question. This type of collision can be costly to resolve, especially if there is insufficient information to know which device is legitimate, and which is not. Moreover, even if the phone is counterfeit, the disconnection of any paying customer is not something the carriers are happy about. This type of issue can largely be solved by simply randomising the identifier values used during manufacturing. If the attackers cannot predict the next unused values, then they cannot steal them before you use them. Furthermore, your operational infrastructure, and that of the carriers, should be configured to whitelist only device identities that were actually manufactured and shipped. Anything outside this valid set is clearly from a counterfeiter, and can therefore be blocked.

End-to-End Encryption

It needs to be understood that there are attackers in your factories monitoring the data path between the device being manufactured and the back-end data systems that support the manufacturing processes. The overwhelming evidence[33][34] suggests that this is true for all consumer electronics manufacturers, even yours. Because of this, all data should be encrypted by default in a way that prevents such snooping. This of course needs to be balanced with the business needs of the factories, as some of the data being sent back and forth is needed to support their operation. For example, both the third party CM/ODM partner, and the OEM may need to access the device identifiers for tracking purposes. This does not mean that plaintext identifiers need to be sent from the OEM to the factory, and then from the factory to the device. A better model would be to send them encrypted to the device, and have the device decrypt and report back to the factory stations, only the relevant details. This ensures that the factory cannot modify data in transit. Similar end-to-end encryption should be applied to any logs that are sent from the device to the OEM or factory.

[33] https://www.fireeye.com/content/dam/fireeye-www/global/en/customers/pdfs/fireeye-global-mfg-casestudy.pdf

[34] http://www.verizonenterprise.com/DBIR/2015/

Physical Security

Too often overlooked in information security circles, physical security is very important. Security guards, metal detectors, security cameras, employee background checks, locked doors and cabinets, even bolting machines to desks, are all valuable things to implement in your manufacturing environment. Be sure not to overlook your internal mailroom as it may pose a very convenient route to exfiltrate both data and hardware from the premises and bypass other physical security measures. Be sensible with your physical security measures; do not merely implement them to fit the letter of some bloated standard[35], implement them where they make sense to actually solve a problem. Be redundant, and implement monitoring, especially where humans are involved. Security guards[36] and system administrators[37] are weak links. While they make sense in most cases, keep in mind that they are human: they're corruptible, underpaid, undertrained, indifferent, outsourced (and hence uninvested in what they protect), and often involved in any existing problem. Above all, test your security measures. Hire a third party investigator or auditor to actively probe for weaknesses. Do this on both a regular and surprise basis, to ensure that there are no lapses.

Secure Scrap

Securing the process by which scrap is discarded is vitally important in eliminating a key source of grey market components. Additionally, it can be difficult to convince a senior leader or executive that they're losing anything more than scrap. Explaining the process by which a $0 stolen scrap board turns into a $1000 loss is not always easy. There are many e-scrap shredding companies that can be hired for on-site destruction of materials before sending them off for recycling. Before this happens, care should be taken to record the identifiers of any devices being scrapped, so that you can be certain they do not show up unexpectedly in your systems.

Factory Desktop and Network Hardening

Malware installed on factory computer systems is a very real threat. These systems, even though they may be unconnected from the Internet, should be hardened as much as possible to prevent infections that might cause a breach. We will not cover this in depth here, but all the standard methods should apply. Some minimal suggestions include:

1. Using an operating system that implements modern exploit mitigations (DEP/ASLR/Privilege Separation, etc).
2. Limiting unwarranted network connectivity such as outbound connections to the Internet and station-to-station communication. Use both station firewalls and network-based defenses.
3. Regular auditing and whitelisting of software installed on stations

---

[35] http://www.praxiom.com/iso-17799-7.htm
[36] http://www.diversifiedriskmanagement.com/articles/security-guard-crimes/
[37] http://en.wikipedia.org/wiki/Edward_Snowden

Privilege and Role Separation

Depending on the scale of your operation, you may wish to implement multiple accounts per manufacturing site. Ideally per-employee user accounts could be implemented however we recognise that this may be impractical due to the level of automation involved in the factory. A reasonable middle ground would be to have one account per location, per role or station type. Each account can be assigned only the permissions required of the role it satisfies. This makes malicious accesses easy to flag in the logs. Your Mexican factory should not be building device models that were only scheduled to be built in your Indonesian factory, and your calibration stations should not be attempting to perform operations that are only supposed to be performed by your functional test stations.

Protective Monitoring for Abusive Patterns

Closing the loop, by monitoring for abuse, is often one of the best defences. Detecting abuses early before they are a big problem will greatly reduce the cleanup costs of an incident. In most cases, there are visible signs of development activity by the attackers long before they reach the point of counterfeiting in large scales. In this phase, they are often sloppy and are much easier to detect. Catching them early has advantages however, there is a balance here, as showing your hand too soon might cause them to hide their tracks better and make them harder to detect next time. Finding the needle in a haystack is much easier if you have done the necessary prep work to create nice orderly haystacks. Understanding your processes, and designing them in a clear fashion, should be the number one priority.

Data Sources

For proper monitoring, data should be acquired from all available sources, both internal and external. While it is common for much of the interesting data to be logged and recorded, it is much less common for all of the relevant data to be accessible from the same system. This is especially true if your systems have grown organically over the course of many years. And of course none of the data is of any value if nobody is actually looking at the data and searching for anomalies. Correlating data between various unrelated parts of your business will be key to identifying many of these anomalies. Once all the data is accessible (a non-trivial task), automated reporting systems can do much of the heavy lifting. Some of the data sources that have proven to be of value in the past include:

- Open source forums (forum.gsmhosting.com, xda-developers.com, hackaday.com, google alerts, etc). Some of these sites are known to have administrators that actively search their access logs for IP addresses belonging to OEMs and law enforcement and will delete threads if they think it is being monitored. Access only from “clean” IP addresses (consider using Tor[38]), and keep local records of anything you may want to reference again.

[38] https://www.torproject.org/

- The “dark web” including private forums and underground markets. Some amount of social engineering work will be required here to gain access and infiltrate these groups. Building credibility can be difficult. This is sometimes referred to as the “meat space” problem. Seek out the threats, learn and understand them, plot and chart their nature and trajectory, and learn about the human beings behind the threat. When that group, malware, or criminal entity attacks your supply chain, you'll at least know what you're up against.
- Internal databases: sales data, manufacturing data, activation data, diagnostic logs and reports, DHCP logs, signing server logs, permission access control logs, etc. It helps immensely if all of this data is available to a single system from which queries can be run. Setting up remote database views, or replication to an analytics or security operations centre would be ideal. Queries can be large, especially when looking at historical data, so impacting production systems is a real possibility unless they are in an offline system. That said, care should be taken to ensure timely data replication so that near-real-time alerts can be generated effectively. If you do not have an analytics system of your own, Google’s BigQuery[39] is a convenient cloud-based alternative. Data can be quickly uploaded and mined for a nominal fee.
- Relationships with other OEMs, particularly those that share vendors. Because many device vulnerabilities will be discovered on another, less security-focussed OEM product before your own, it helps immensely to have close ties to the product security team at other companies. Conferences, summits, Twitter, etc are all useful avenues to build this personal network. The global security community is not terribly large, so keeping your ear to the ground does not need to be an insurmountable task.

Abusive Patterns

Odds are that you are already collecting sufficient information to detect most of the abusive patterns listed below. Collecting additional information to catch the rest should be made a priority. It is important to note, that all of the patterns listed below have been used in the past to spot actual abuse in the field. These are not merely theoretical.

Mismatch in Number of Units Ordered/Shipped/Activated

The number of units ordered from the factory, built by the factory, shipped by the factory, and activated by the customer should always be closely related. Accounting for scrap and unsold units, these numbers should be roughly in decreasing order. If there are large discrepancies in these numbers, then it could indicate an alternate counterfeit source of devices leading to customer activation, or devices or parts being stolen from the factory. It can sometimes be difficult to collect sufficient data to detect this type of abuse. CM/ODM partners often keep their factory yields a closely-guarded secret, because it is closely tied to their prices and profit margins. The business agreements between OEM and CM/ODM partner need to be negotiated with this detail in mind.

[39] https://cloud.google.com/bigquery/

Activation Data

When a device first registers or otherwise connects to the operational infrastructure, certain information is reported. Likely this will include a device identifier (or identifiers), a device model number, software version, IP address/location/carrier, and similar information. All of this information should be checked for sanity. Some suggested checks include:

1. ensure that the device identifier was not reused from an older device model,
2. ensure that the device identifier was not reused from a different OEM's device,
3. ensure that the software version matches a valid range for the reported model,
4. in the case of multiple device identifiers, ensure that they were all assigned to the same device during manufacturing
5. ensure that the location makes sense for the customer to whom the device was shipped (this one is fuzzier, but can help locate stolen shipments)

Identifier Cloning

Cloned devices may show up in either the activation data as mentioned above, or in the data collected from incoming RMA devices. In either case, you will be looking to see if each identifier matches the device's manufacturing records, and if there are any other active devices with the same identity.

Same Device in Multiple Locations

There is no physical way that a device can be in two locations at once. If you see this in your activation logs, then it is obvious that a device has been cloned. This is similar to the fraud detection logic used by credit card companies, and indeed a great many of the techniques used by the credit card industry can apply here.

Identifier Guessing

Even if you have not properly randomised the device identifiers during assignment, there should be considerable entropy added to the system throughout the sales channel. The end result is that the operational infrastructure should not see large numbers of devices activating with sequential device identifiers. If you see this pattern, then it is highly likely that someone is attempting to brute-force and use your infrastructure as an oracle to help them identify valid device credentials. Furthermore, unless the aggressors have implemented rate limiting, this type of activity is likely to be in much higher volume than typical activation volumes, and so will likely be easy to spot.

Sequential Identifiers

As mentioned above, devices activating on your operational infrastructure for the first time will normally do so in a somewhat random order. The one possible exception is if someone in the sales channel is manually testing each device in a particular shipment to screen for some specific defect (this should hopefully be a rare occurrence). If you see large volumes of devices activating in sequential order, it is much more likely that someone is using automation to

brute-force the activation process. The failed activation attempts are of particular interest here, so make sure those are recorded in the logs and not merely ignored.

Unit Theft

In certain countries where crime is high and manufacturing is cheap (Mexico and Brazil are prime examples) it may be unfortunately common for completed product to be stolen directly from the factory, or during shipment. To combat this, proper tracking of all devices throughout their lifetime is essential. Any devices known to be stolen should immediately be blocked from the operational infrastructure (through things like the GSMA's IMEI blacklist), and alerts set up in case they appear in a repair facility to be laundered. Smartphones report a great deal of information that can be used (especially in aggregate) to help solve the where and who questions regarding a large volume theft.

Component Theft

Theft of components from your factory inventory, or during shipments, is a very real possibility. Detecting this in high volumes should be straightforward. In lower volumes it can be harder without component-level tracking. Component-level tracking is vitally important for any security-sensitive components. In the case of a smartphone this means at minimum the main application processor and flash memory.

RMA of Unknown Units

Units being returned for repair or refurbishment should be screened immediately to confirm that they are in fact legitimately-produced devices with a valid manufacturing history. Further, check to see if they are on your list of known stolen devices. This prevents your return process from unknowingly laundering stolen units. While the monitoring for this should be relatively straightforward to implement, the response process is much more difficult in practice. Commonly, customer returns are handled through the sales channels, where access to the data systems is either not desirable for security reasons or not possible because of contractual obligations.

Identifier Mismatch

Devices are often manufactured with a number of different identifiers (IMEI, MAC, serial number, model number, etc). This is recorded in your manufacturing data systems. In any place where a device will report more than one of these, it is important that the identifier combination be checked to confirm that they were indeed assigned to the same device. Any device with a mismatched identifier set is likely to have undergone some form of unsanctioned after-market modification.

Unknown Ethernet MAC Address/MAC Prefix

The DHCP logs of your corporate network and your manufacturing network (possibly at a CM/ODM partner) are invaluable sources of information. Typically you will have a limited set of hardware deployed in these networks, and this is recognisable from the Ethernet MAC address

recorded in the DHCP server logs. Any new piece of hardware should be correlated to a valid system replacement, or manufacturing line expansion. Furthermore, the MAC prefix indicates the Ethernet equipment vendor, and any unknown or new brands of hardware showing up on your network should be treated with suspicion. While it is relatively trivial for an attacker to perform MAC spoofing to hide their tracks, they often neglect this countermeasure out of incompetence or laziness. This is especially apparent during the early developmental stages of an attack.

Multiple MAC/IP Address Combinations Per Day

MAC spoofing, while subtle, will still show up in your DHCP logs. This will present itself as multiple host names with the same MAC address. While this situation can sometimes happen under normal situations (during a DHCP lease expiry for example), if you see repeated log entries in the same DHCP lease time period with multiple host names and the same MAC address, then it is likely you have MAC spoofing going on. If this is the case, you might want to review other traffic from those machines to see if anything else looks suspicious.

Factory Credentials Used From Incorrect Site

If you have properly separated privileges between and within your manufacturing sites, then you should see each user credential performing a limited and predictable set of operations in your logs. Anything outside this pattern should be considered suspicious and investigated further. One common example of this is seeing credentials for one factory site show up in the logs from an IP address range belonging to a different site. On its own this could be a one-time innocent mistake for a CM/ODM partner who has multiple sites. At no time should you legitimately see credentials for one CM/ODM partner being used by a different CM/ODM partner. Limiting user accounts to specific IP ranges is a good defence against this type of abuse.

Unknown IP Addresses

Related to the above section, you should be aware of all IP addresses on your network. Anything out of the ordinary should be flagged. Most factories are behind a firewall and NAT, so it is unlikely that you should ever see an Internet-routable IP address hitting one of your systems. Furthermore, you should definitely flag any IP address listed as a Tor exit node[40], as this would be a clear indicator that someone is hiding something.

Operations Performed From Wrong Station Type

Each factory station is responsible for a specific set of tasks. Any station attempting to perform tasks outside of this normal pattern should be considered suspicious. For example, a station that is responsible for RF calibration should not be seen loading final software on devices. In such a case it is possible that an operator has exceeded their privileges, or malware on the station is allowing a remote party to misuse it.

[40] https://check.torproject.org/cgi-bin/TorBulkExitList.py

Out of Range Identifiers

Device identifiers such as IMEI and MEID are purchased and allocated from a defined pool of numbers. Devices attempting activation from outside this allowable range are clearly unacceptable. Simple screening for this case should be implemented. To further restrict the range, you can limit activations to only identifiers known to have actually been correctly assigned to shipped products. This requires some measure of real-time communication between factory systems and operational infrastructure, however this is quite do-able.

High Volume Stations

Each test station processes a certain number of devices in a given time period. This will be limited by factors such as manual operator speed, number of test ports or USB cables, station performance specs, and duration of test sequence. A typical number might be ten devices every five minutes. If this number is exceeded by a significant margin, then you should investigate further. A reasonable threshold might be 50% or 100% of the normal volume. You will need to look at your actual data to determine what works for your process. These thresholds will need to be specific to each station type, and possibly even differ between factory facilities. In the case of a malware-infected station it would not be unthinkable to see these volumes hundreds or even thousands of times higher than expected. If you find one infected station, there are almost certainly more, possibly just idle in case the primary infection gets detected and cleaned. Do not set your thresholds too high however, or distributing the extra load across multiple infected stations becomes an easy way for the attackers to stay under your radar.

Activity During Quiet Times

While it is possible that some test stations are completely automated, for complex consumer devices such as smartphones, there is still a good deal of manual interaction. Because of this, most factories will experience a number of predictable down times such as during shift changes, lunch breaks, holidays, nights (unless it is a 24 hour facility), and weekends. Any test station activity during these periods should be analysed further. Monitoring illicit activity and identifying periods of inactivity can also help with attribution, by revealing the timezone of the attacker(s) or public holidays of their country of origin.

Station-to-Station Communication

Test stations should be designed to get their operating data from, and report back their results to centralized systems. Any network traffic between stations themselves is most likely not necessary. The test station should have a firewall installed preventing this form of communication, as well any networking equipment in the factory should be configured to block and report on these rogue traffic patterns. Malware propagating through your test stations will rely heavily on this type of lateral movement.

TCP TTL Out of Range

Every TCP packet on the network has a Time-To-Live (TTL) value set. This is decremented on each hop in the network and the packet is dropped when it reaches zero. This is designed to ensure that packets cannot loop endlessly and clog routers. This also happens to be a very useful tool in determining how far a packet has travelled across the network. For a fixed factory environment all routes should be of a very predictable and consistent length. Packets traversing a much longer route are likely to have been routed through a rogue network appliance or a malware-assisted proxy. Analysing your network traffic can be easy with many freely-available tools such as p0f[41] and wireshark[42].

Unknown Operating System Platform

As mentioned in the previous section, p0f is a very useful analysis tool. It excels in profiling network packets to determine the originating operating system. If your factory operates a homogeneous software environment, then this can readily identify anomalies such as laptops, virtual machines, proxies, routers, and bridged networks.

High Number of IMEIs Per SIM Card

Under normal situations each SIM card will activate with only a handful (5-10) of distinct devices (IMEIs) in its lifetime. This is typically seen in the case of a customer upgrading or swapping phones. Less common is the case of small repair shops that perform testing with SIM cards on live networks. In such a case you may see a couple of hundred devices (<400) sharing a given SIM card. If you see a specific SIM card activating with thousands of distinct devices, then it could be something more. In this case you may be looking at a counterfeiting operation where live-air testing is used to determine if the device is configured properly. This technique has been used in the past to brute-force valid device identifiers by using the live infrastructure as an oracle. Other traits of this sort of operation are sequential device identities, and higher than normal activation failure rates.

Use of Cloned SIM

Another possible explanation for large numbers of IMEIs per SIM card is the possibility that the SIM card itself has been cloned a large number of times. SIM cloning is an illegal activity in many countries, so in itself it will indicate that some underground activity is at play.

Obsolete Devices in New Production

It is common for counterfeiters to acquire a supply of obsolete devices because they are cheap and to attempt to refurbish them into a more modern and desired model. The typical production lifetime of a product will be between 4 and 18 months depending on market demand, part supply, and frequency of hardware upgrades. This may vary greatly depending on your specific product. That said, there is always a point where a given device model is no longer in active

[41] http://lcamtuf.coredump.cx/p0f3/
[42] https://www.wireshark.org/

production, and similarly, a point where it is no longer serviced in high volumes in your repair centers. If you are seeing obsolete devices being processed in higher-than-anticipated volumes, you should investigate more closely.
Because older devices and software versions are likely to have many more known security vulnerabilities than the latest and greatest, they may pose a softer target for attackers. This fact has been leveraged by attackers in a variety of ways to make exploit development easier. Some of this activity is visible in the digital paper trail, but the specific details will vary greatly between products.

Missing Scrap

Tracking of scrapped devices, in particular anything security-sensitive (processor, memory, TPM, etc), should be done on a per-unit basis through serial numbers. Scrapped serial numbers should be flagged as scrap in your database, to ensure that they cannot be reused should the physical parts later turn up in the grey market supply chain. Scrap that cannot be accounted for should be investigated.

Contractual Damages and Reverse Billing

While most of the issues around supply chain security stem from physical or network security problems at a CM or ODM partner, most of the financial and reputational penalties will be born by the OEM. This imbalance cannot always be solved by playing the technical cat-and-mouse game between attackers and defenders. If the opportunity exists, building contractual requirements into your business relationships up front can help immensely. An internationally experienced legal team is a big help here in drafting an enforceable master service agreement for the relevant country. For example, you could reverse-bill a virtual good such as a device identifier, certificate, or similar at a modest cost ($10 each?) to the factory at the time it is consumed, and then have it billed back to the OEM at the time the completed unit is shipped. The net exchange of funds for any legitimate product is zero, but any devices that go missing along the way will show up on the books of the CM/ODM partner. Transferring the financial hardship in this way is sure to get them on board with rooting out their security problems. In implementing such a scheme, care needs to be taken to accommodate the normal day-to-day scrap of faulty parts and units. Every device, even scrap, needs to be tracked by serial number rather than merely by weight, and on-site destruction should be implemented. There are many secure scrap services that can help with all of this.

Legal Disclaimers on Software

Preventing the leak of internal development and factory tools is nearly impossible in the long run. When this does happen, there needs to be a clear legal disclaimer upon startup of the tool that indicates the ownership of the tool, and the permitted uses of same. This will help provide some recourse to your legal department should they attempt to go after the illicit users of these tools. While the disclaimer will not actually prevent anyone from using the tool, it does remove

the defence argument that the illicit users simply did not know they were not allowed to use these tools.

Stolen Devices, GSMA Blacklist, Kill Switch Legislation, etc

The GSMA maintains a list of stolen phones to which carriers are expected to subscribe. The idea behind this is that if you report your phone as stolen, it will not be allowed to activate on any carrier network. This should remove the incentive for smartphone theft. In practice however, it requires the paid subscription of every cellular phone carrier worldwide. This type of global cooperation is unlikely to ever happen when there are carriers in third world countries gladly willing to accept monthly payments from customers regardless of where their phone comes from. Furthermore, there are other issues with the GMSA blacklist implementation. From a practical perspective, lookups and billing are on a per device basis, and caching and bulk downloads are not allowed. This means that your repair facility requires a real-time network connection, which is not always easy. From a legal perspective, the per-device query model means you must implicitly share the identity of each repaired device with the GMSA. This could potentially be an SEC fair disclosure violation[43].

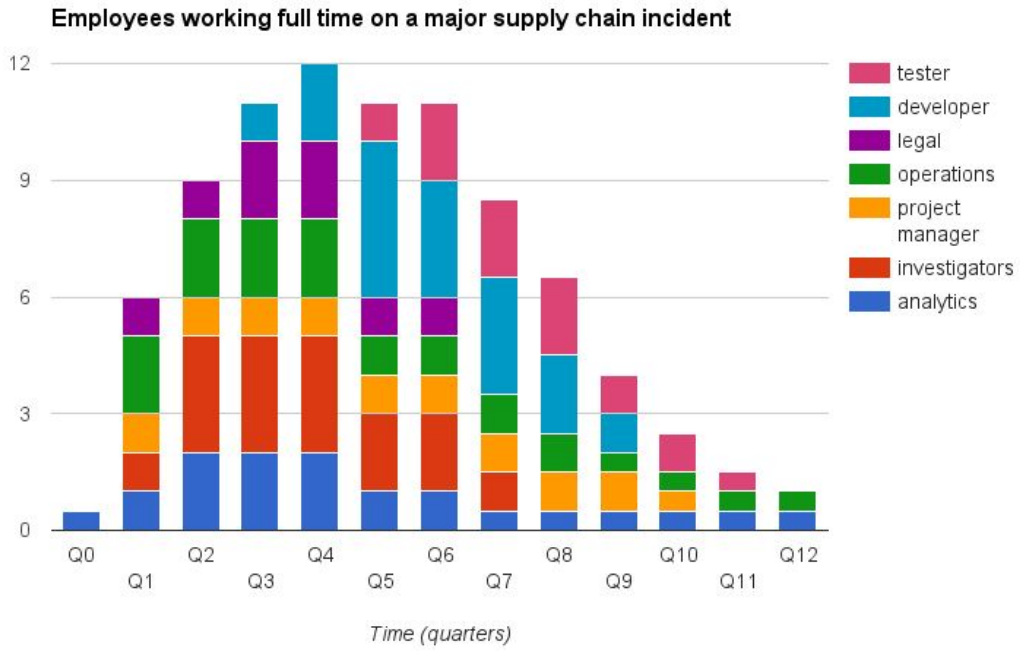
There is much recent discussion, most notably in California[44] and New Mexico, about legislating a so called Kill Switch into new smartphones. This is an attempt to legislate technical solutions to devalue stolen phones, and thereby reduce the amount of smartphone-related thefts for consumers. While the goal is commendable, the legislation as enacted missed the mark. Stolen devices rarely get stolen for individual use. Far more common is for shipping containers full of stolen goods to accumulate and be sent overseas to be torn down and used as components in the grey market supply chains. In such a high-volume model, replacing the flash memory (and thereby disabling the kill switch) is a relatively cheap operation. This attack pattern was common at least as early as 2004 and continues unabated today. Furthermore, while a technically strong Kill Switch implementation could be designed beyond the mere letter of the new laws, it would interfere with the legitimate RMA of smartphones, and so would represent a high cost to OEMs and their repair operations. Kill switch bypass mechanisms are therefore needed by the OEMs, but this poses an obvious danger of unintentionally, or (in the case of rogue network access) intentionally laundering stolen devices through the repair process.

Investigation Costs

A reactive incident response process is far more costly than a proactive one. Below is a chart showing the human cost of investigating a major supply chain incident at an actual smartphone vendor. This does not account for any of the monetary costs involved with dealing with the actual counterfeit devices, and certainly does not include any of the intangible costs such as

[43] https://www.sec.gov/rules/final/33-7881.htm
[44] http://leginfo.legislature.ca.gov/faces/billNavClient.xhtml?bill_id=201320140SB962

irate customers and brand damage. The end result of all the effort was to build a more proactive monitoring system such that Q12 indicates the new steady-state norm. It is believed that this new low-effort process will scale to account for many future such incidents.



Summary

There is no success without risk, and there is considerable risk inherent in a global supply chain. There are a number of challenges to ensure you build a secure product while managing the grey market that grows in the shadow of your success. This is highlighted by the many public breaches, and the reactions of standards bodies, industry groups, and governments. You will need to have a diverse set of skills to tackle these challenges effectively. Through thoughtful product design, careful manufacturing, and a thorough monitoring scheme, you can be sure that you will not have to suffer through an expensive incident cleanup effort.

NCC Group Manufacturing Security Services

NCC Group is a global information assurance specialist providing organisations worldwide with expert escrow, verification, security consulting, website performance, software testing and domain services. We have extensive experience in providing our clients with the knowledge, tools and training needed to detect, investigate, mitigate, and prevent rogue manufacturing activities. We can help in all phases of the product lifecycle, including:

- concept and requirements

- software and hardware architecture and design
- implementation review and penetration testing
- manufacturing and repair service design and implementation

Let us help you manage your risks, so that your customers can be free from doubt. To arrange a follow-up, fill in the contact form located at https://www.nccgroup.trust/.

Acknowledgements

This document was prepared by the author (Rob Wood) but the knowledge contained within is, nevertheless, a direct result of working with many fine folks over the years. The author would like to thank Mike McCallum for the inner workings of the security response process, Matt Welch for delving deep into the underground information markets, John Sheeran and Stuart MacKinnon for helping me understand all the complexities of the aftermarket repair enterprises, Ollie Whitehouse and Andrew Hickey for peer review, and countless others for teaching countless other things.

Acronyms

AES	Advanced Encryption System. A commonly used symmetric cipher.
ASLR	Address space layout randomization. A technique that involves randomisation of the memory space to make exploitation more difficult.
BB PIN	BlackBerry PIN. A network agnostic device identifier specific to the BlackBerry device network.
BRIIC	Brazil, Russia, India, Indonesia, and China. Commonly listed countries with weak intellectual property protection laws.
BT	Bluetooth. A short range wireless networking protocol.
CM	Contract manufacturing. A manufacturing model that involves contracting out some component of the manufacturing process.
ECC	Elliptic Curve Cryptography. A commonly used public key cryptographic algorithm.
DARPA	Defense Advanced Research Projects Agency
DEP	Data execution prevention. A hardware feature of modern processors that prevents the execution of instructions that reside in specially marked memory pages. This is used to make exploitation of software vulnerabilities more difficult.

DHCP	Dynamic host configuration protocol. A protocol used to automatically configure the networking parameters of computers on a computer network.
GSMA	GSM Association. An association of mobile operators and related companies.
IMEI	International Mobile Station Equipment Identity. A unique identifier assigned to each GSM mobile phone.
IP	Internet Protocol.
MAC	Media access control address. A hardware address used at the physical layer of the networking protocol.
MEID	Mobile Equipment Identity. A unique identifier assigned to each CDMA mobile phone.
NFC ID	Near Field Communication. A very short range wireless protocol.
NIST	National Institute of Standards and Technology.
ODM	Outsourced device manufacturing. A manufacturing model that involves outsourcing the complete manufacturing and possibly design of a device.
OEM	Original Equipment Manufacturer.
RMA	Return merchandise authorization. The process by which devices are returned to the OEM for refund, replacement, repair, and recycling.
RSA	Rivest-Shamir-Adleman. A commonly used public key cryptographic algorithm.
SHA	Secure Hashing Algorithm. A one way hashing function used with various crypto-systems.
SIM	Subscriber Identity Module. A small chip inserted into GSM phones to deal with service and billing information.
SMT	Surface mount technology.
SoC	System on Chip. An integrated circuit that incorporates all of the necessary functions into a single chip.
TCP	Transmission control protocol. The core protocol of the internet.
TPM	Trusted Platform Module. A security chip included in many modern

	computers and other electronics.
TTL	Time to Live. In the context of TCP, this indicates the maximum number of hops a packet can take across a computer network.
USB	Universal Serial Bus. A commonly used serial port specification.
WLAN	Wireless local area network. A medium range wireless networking protocol.