# Supply Chain Threats Against Integrated Circuits

## White Paper

Matthew Areno, PhD

July 2020

# LEGAL NOTICE

# Contents

# 1. Introduction

Computing systems today face an unprecedented number of attacks that begin even before the system is ever turned on by the end user. The supply chain used to generate these systems has been the target of a variety of different attacks and the topic of an array of research papers by respected scholars. Understanding supply-chain attacks against the Integrated Circuit (IC) components that make up computing systems requires an assessment of all potential attack vectors throughout the lifecycle of the IC and the computing system itself.

A variety of attacks might be leveraged at each stage of the IC lifecycle via one or multiple attack vectors. Some attack vectors are unique to specific stages of the lifecycle, while others might be universal across all stages. Equally, mitigations against a specific attack or attack vector at one stage might be insufficient or inappropriate for other stages. Further complicating supply-chain protections is the fact that the level of access to IC components or computing systems can vary across the entire lifecycle. Ultimately, because a computing system is typically composed of multiple components from different manufacturers, each with its own level of scrutiny in relation to supply-chain attacks, ensuring the integrity of a computing system across all stage of its lifecycle is extremely challenging.

The purpose of this paper is to identify all known supply-chain related attacks against IC components and computing systems and show at which stages of the supply-chain lifecycle these attacks might be leveraged. It should be understood that the list of attacks discussed in this paper is limited to known attacks either from published articles or academic papers. Additionally, it is expected that the list of attacks will continue to grow as new attacks and attack vectors are discovered.

This paper consists of four sections beyond the Introduction. Section 2 provides a listing of the different stages of component lifecycle as viewed for the purposes of this paper, along with accompanying definitions for each stage. Section 3 details the known supply-chain attacks, identifies which of the supply-chain lifecycle stages the attacks target, and then provides an associated definition of each attack. Section 4 includes examples of each attack discussed in prior sections (please recognize that some links might change without notice). Finally, Section 5 presents a brief conclusion of these attacks and recommendations for future efforts in this area.

# 2. Component Lifecycle

For the purposes of this document, the supply-chain lifecycle for ICs, components, or computing systems (collectively, "component") is divided into seven stages. Each of these stages represents a different aspect of the development of the component and might be performed entirely by a single entity or represent work by multiple entities. The seven stages of the component lifecycle are:

1. Conceptual
2. Design
3. Integration
4. Fabrication
5. Testing
6. Provisioning
7. Deployment

It should be understood that these phases are not necessarily sequential in nature, as execution of one phase might overlap execution of another. However, it is assumed that no phase is capable of being completed prior to the completion of any previous phase(s). It should also be understood that the lifecycle presented here is reiterative, for instance, occurring once for the IC, then for the component, and then for the system.

## 2.1. Conceptual Stage

This stage is the birthplace for all components. It is here that the initial thoughts and ideas for the purpose, use, and functionality of the component are collected. This stage is specifically meant to address the "what" and "why" questions regarding the component. This work often includes people from a variety of organizations within a business and is meant to address an existing need or gap in capabilities. Output is traditionally a collection of diagrams and requirements identifying exactly what is needed and why it addresses the need or gap.

## 2.2. Design Stage

In the Design Stage, the concept created in the Conceptual Stage takes form. This stage only considers in-house development related to the component and includes use of Intellectual Property (IP), Computer-Aided Design (CAD) tools, third-party software, and hardware prototyping. Final decisions are made to ensure the resulting design meets all requirements identified in the Conceptual Stage and performs the needed function.

## 2.3. Integration Stage

The Integration Stage specifically relates to incorporating third-party hardware and/or software into the overall design of the component. As such, it is likely that this stage will begin prior to the completion of the Design Stage. Third-party vendors often provide proven solutions that are easily integrated into existing designs to provide a needed capability, without requiring the component designer to "re-create the wheel".

## 2.4. Fabrication Stage

Once an initial "draft" of an IC is completed, the Fabrication Stage is kicked off to begin creation of components that can be used to prove it achieves the desired functionality. This stage includes development of all ICs, as well as creation of Printed Circuit Boards (PCBs) and other assemblies on which the ICs are included. It includes all preliminary and completed versions of the components.

## 2.5. Testing Stage

Testing is conducted on all components once preliminary fabrication is completed. This stage will typically continue until all fabrication is completed and be the last stage before components are ready for provisioning.

## 2.6. Provisioning Stage

During the Provisioning Stage, components are loaded with both standard and sensitive data needed by the component to perform its required function. Standard data represent the collection of any information that is generic across components or component families, which data might include freely available or open-source information or code. Sensitive data are anything for which the disclosure of their value would result in the compromise of IP, security, or user information.

## 2.7. Deployment Stage

The final stage is the deployment of the products and includes both the delivery of products from the original manufacturer's facilities to OEMs as well as the delivery from OEMs to end customers.

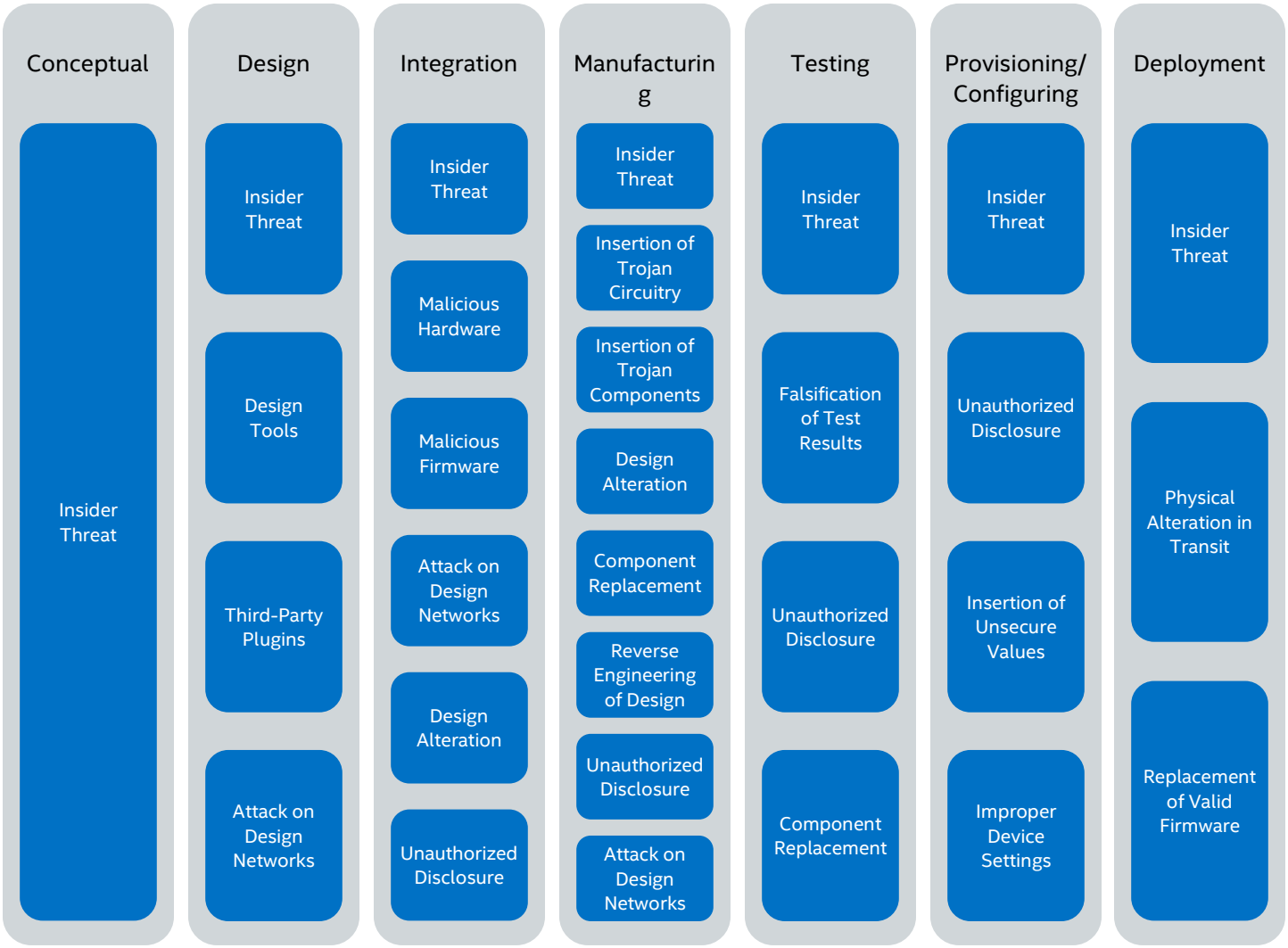


*Figure 1 - Supply-Chain Attacks by Lifecycle Stage*

## 3. Supply-Chain Attacks

A variety of supply-chain attacks exists that might be leveraged at one or more stages of the component lifecycle. Some of these attacks directly target the component itself, whereas others might utilize attacks against corporate elements (such as development networks) to indirectly attack the component. Additionally, each attack might manifest itself in different lays based on which stage the attack is targeting.

To best understand how the attacks listed in this section might be used against components, it is important to identify when each attack might occur. Figure 1 provides this mapping of each attack to one or more of the seven previously identified lifecycle stages. Figure 1 should not be interpreted to say that any of the listed attacks does not apply to any other stage or that Figure 1 captures all the possible attacks. Rather, this is a listing of all types of attacks the authors of this paper were able to identify.

## 3.1. Insider Threat

Insider Threat is a constant issue that exists in all stage of the supply-chain lifecycle. Capabilities of the insider vary from one stage to another, but the threat remains. At any stage, an insider with access could potentially modify or add functionality, exfiltrate sensitive information or design files, insert or swap out components, or any number of other possible attacks that could compromise the security of the overall product.

## 3.2. Design Tools

The design tools are critical for the development of ICs. However, a malicious attacker could conceivably inject trojan circuitry into design tools used that appears to allow for a proper functional unit, but when that unit receives a unique input, the unit changes its functionality. Further, the design tools could contain trojans themselves that, upon installation on corporate systems, begin exfiltrating sensitive design files to an external server.

## 3.3. Third-party Plugins

Design tools often support integration with third-party plugins, such as from outside hardware vendors or even from original manufacturers. Many of these modules are not signed or authenticated and could be contain malicious elements. Additionally, installation of these modules could be subject to man-in-the-middle (MITM) attacks during transit over the network because they are not sufficiently or properly verified before installation.

## 3.4. Attack on Design Networks

In cases where the network used by design systems is not a closed network, attack can be launched against these networks from external adversaries in the hopes of obtaining or modifying sensitive design files. Such attacks could be supported by a malicious insider threat that installs malware or a virus on critical security systems, allowing for easier access and attacks against target platforms.

## 3.5. Malicious Hardware

Malicious hardware is a component that provides all the required functionality and meets all design requirements but has additional or hidden functionality that performs some malicious action continuously or only when activated by a specific input. It is statistically impossible in most cases to exhaustively test a component with all possible input vectors.

## 3.6. Malicious Firmware

Many hardware elements require their own embedded software, referred to as firmware, in order to properly function. In a similar manner to malicious hardware, the firmware might also include additional or hidden functionality. Further, much of this firmware is stored in non-volatile memory and is not integrity-protected or authenticated prior to its execution. This means the firmware could be malicious from the start or could be altered by a number of different actors throughout the duration of the supply-chain lifecycles.

## 3.7. Design Alteration

Alteration of the design might occur during a number of different stages by various actors. A supplier might suggest a seemingly innocuous change that actually provides the ability to induce an undefined state of execution in the system. An integrator might make a covert change to the design files provided as part of an integration effort.

## 3.8. Unauthorized Disclosure

Most of the design files used in the creation of IC components is only shared under the strictest of Non-Disclosure Agreements (NDAs). However, any violations of these NDAs, either intentional or unintentional, might or might not be reported by the violating party. This expands the insider threat potential beyond just the manufacturer to include every supplier or vendor given sensitive information about the manufacturer's products.

## 3.9. Insertion of Trojan Circuitry

Trojan circuitry might be inserted during one of several different phases. Beyond the trojan circuitry discussed as part of malicious hardware, other circuitry modification could be made directly to the PCB on which the manufacturer's components are installed. The result could be the full disclosure of sensitive information or control or part of a multi-step process needed to activate the trojan circuitry.

## 3.10. Insertion of Trojan Component

A trojan component is a device that has been made to look and function identically to a valid component but has some level of additional functionality that performs a malicious action when triggered. These devices might be easily distinguishable when powered on due to power draw or throughput deviations from the norm, yet they are easily inserted during stages when only visual validation is performed.

## 3.11. Component Replacement

If an attacker is unable to directly impact the original component or authorized, third-party components, the attacker might resort to attacking the supply chain of authorized, third-party partners or fabrication facilities. This could result in the mixture of malicious and benign components being provided to partners responsible for populating PCBs. Depending on the complexity of the component, this could also be done by someone with physical access in a relatively short period of time after the product has been produced.

## 3.12. Reverse Engineering of Design

Any design that can be created can also be reverse engineered. Although an attacker might not be able to gain direct access to the original design files, extraction of the synthesized files might allow for reverse engineering of critical design secrets. Such design secrets could include any cryptographic key material stored directly in the ICs.

## 3.13. Falsification of Test Results

During the test phase, an attacker might modify the results of the testing performed. This could be done to prevent detection of malicious changes made to the component or to mask the presence of trojan circuitry injected into a component.

## 3.14. Insertion of Unsecure Values

Every component that contains cryptographic material or other values designed to uniquely identify it from all others is potentially susceptible to insertion of unsecure values. Rather than assigning legitimate values that pass all associated requirements, an insider or attacker might attempt to modify or replace these values with a weaker version that does not meet these requirements. The result would be devices that have weaker or no security, are misidentified from other components, or are simply not unique enough.

## 3.15. Improper Device Settings

Many modern ICs use electrical fuses (eFuses) as a means of controlling access to specific capabilities or information within the component. If an attacker can modify the default setting or in some way compromise

the eFuse's integrity, the attacker might be able to enable access to sensitive data or functionality that should not be permitted once the component leaves the manufacturer's facility.

### 3.16. Physical Alteration in Transit

Once a product has completed all testing and provisioning steps, an attacker might attempt to make physical alterations to the product during transit. The complexity of such attacks can range from relatively simple to extremely complex, depending on the target component. The purpose of such attacks is similar to that of malicious hardware discussed previously but can be performed after the most extensive tests are conducted.

### 3.17. Replacement of Valid Firmware

Along the same lines as physical alteration in transit, it might also be possible for an attacker to replace valid firmware images with malicious images or to make alterations to the existing firmware. This is especially true for firmware that is not signed or integrity-checked by a trusted element on the component.

### 3.18. Overproduction of Parts

A malicious production entity might make more copies of a given product that the entity reports to the original owner. These parts would not be trackable by the original owner and might be sold to a malicious party. Such a situation can be further complicated if the product contains what should be uniquely identifiable information or key material, in which case the extra part might be modified and potentially inserted in place of the corresponding legitimate part.

### 3.19. Fictitious Recycling

Parts or products consigned to recycling might instead be intercepted by a malicious party and re-inserted into the commodity market. This could be done for malicious purposes or simply for profit. This is especially concerning for ICs that end up in critical systems with a long shelf-life, such as Industrial Control Systems.

## 4. Threat Examples

The table below provides a non-comprehensive collection of known issues, attacks, and discoveries of supply-chain attacks. This includes articles, research publications, public disclosures, and conference proceedings. As such, some links represent proven attacks, whereas others represent the potential for such attacks against a variety of targets. Although many of these attacks could be listed under multiple categories, they should be uniquely listed under only a single category.

**Insider Threat**

- https://www.wired.com/2013/06/nsa-whistleblower-klein/
- https://www.bloomberg.com/news/articles/2014-03-13/target-missed-warnings-in-epic-hack-of-credit-card-data
- https://www.interguardsoftware.com/blog/supply-chain-attacks-insider-threats-that-are-often-ignored/
- https://www.dni.gov/index.php/ncsc-what-we-do/ncsc-insider-threat

**Design Tools**

- https://www.forcepoint.com/blog/x-labs/autocad-malware-computer-aided-theft
- https://www.zdnet.com/article/new-industrial-espionage-campaign-leverages-autocad-based-malware/
- https://knowledge.autodesk.com/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/AutoCAD-Core/files/GUID-9C7E997D-28F8-4605-8583-09606610F26D-htm.html
- https://dwheeler.com/trusting-trust/dissertation/html/wheeler-trusting-trust-ddc.html

**Third-Party Plugins**

- https://blog.renditioninfosec.com/2017/08/software-pluginsextensions-should-be-part-of-your-threat-model/
- https://www.computing.co.uk/ctg/news/3023427/popular-wordpress-plugin-plagued-with-malicious-code
- https://threatpost.com/researchers-show-how-popular-text-editors-can-be-attacked-via-third-party-plugins/130559/
- https://eclypsium.com/2019/07/16/vulnerable-firmware-in-the-supply-chain-of-enterprise-servers/
- https://www.bloomberg.com/news/articles/2014-03-13/target-missed-warnings-in-epic-hack-of-credit-card-data

**Attack on Design Networks**

- https://www.justice.gov/opa/pr/two-chinese-hackers-associated-ministry-state-security-charged-global-computer-intrusion
- https://digitalguardian.com/blog/how-to-secure-intellectual-property
- https://www.willistowerswatson.com/en-US/insights/2018/03/decode-cyber-brief-emerging-cyber-risk-intellectual-property-theft
- https://blog.malwarebytes.com/threat-analysis/malware-threat-analysis/2017/07/all-this-eternalpetya-stuff-makes-me-wannacry/
- https://www.darkreading.com/endpoint/privacy/chinese-apt-backdoor-found-in-ccleaner-supply-chain-attack/d/d-id/1331250?

**Malicious Hardware**

- https://www.bloomberg.com/news/articles/2018-10-04/the-big-hack-inside-the-bag-of-tech-tricks-used-by-china-spies
- http://static.usenix.org/events/leet08/tech/full_papers/king/king.pdf

**Malicious Firmware**

- https://www.nytimes.com/2014/01/15/us/nsa-effort-pries-open-computers-not-connected-to-internet.html
- https://www.vice.com/en_us/article/pan9wn/hackers-hijacked-asus-software-updates-to-install-backdoors-on-thousands-of-computers
- https://eclypsium.com/2019/04/23/shadowhammer-and-the-firmware-supply-chain/

**Design Alteration**

- https://www.brookings.edu/wp-content/uploads/2016/06/Villasenor_HW_Security_Nov7.pdf

**Unauthorized Disclosure**

- https://yourstory.com/2014/09/intellectual-property-rights-third-party

**Insertion of Trojan Circuitry**

- https://doi.org/10.1109/MSPEC.2015.7024511
- https://doi.org/10.1109/MDT.2013.2247460
- https://www.defcon.org/images/defcon-16/dc16-presentations/defcon-16-kiamilev.pdf

**Component Replacement**

- https://www.scientificamerican.com/article/the-pentagon-rsquo-s-seek-and-destroy-mission-for-counterfeit-electronics/
- https://www.sae.org/standards/content/as5553/
- http://www.bis.doc.gov/defenseindustrialbaseprograms/osies/defmarketresearchrpts/final_counterfeit_electronics_report.pdf
- http://www.homelandsecuritynewswire.com/fake-chips-china-threaten-us-military-systems

**Reverse Engineering of Design**

- https://www.vice.com/en_us/article/qvmkdd/counterfeit-iphone-x-review-and-teardown
- https://www.fanaticalfuturist.com/2018/06/reverse-engineering-computer-chips-ridiculously-easy/
- https://www.blackhat.com/docs/us-15/materials/us-15-Thomas-Advanced-IC-Reverse-Engineering-Techniques-In-Depth-Analysis-Of-A-Modern-Smart-Card.pdf

**Falsification of Test Results**

- https://embeddedartistry.com/blog/2018/11/5/musings-on-supply-chain-vulnerability-in-light-of-the-big-hack

**Insertion of Unsecure Values**

- https://www.tomshardware.com/news/infineon-tpm-insecure-rsa-keys,35668.html

**Improper Device Settings**

- https://www.wired.com/story/barium-supply-chain-hackers/
- https://www.mitre.org/sites/default/files/publications/supply-chain-attack-framework-14-0228.pdf

| |
|---|
| **Physical Alteration in Transit**<br>• https://www.acq.osd.mil/se/docs/Supply-Chain-WP.pdf |
| **Replacement of Valid Firmware**<br>• https://www.infoworld.com/article/2608141/snowden--the-nsa-planted-backdoors-in-cisco-products.html |

## 5. Conclusions

Attacks against the supply chain represent one of the most critical collection of attacks currently in use against computing systems. These attacks are not always immediately obvious or detectable and might not trigger at a time when the component is directly accessible by a validating party. Such problems make many supply-chain attacks very difficult to mitigate. Many attacks can be mitigated through better internal policies, more expansive monitoring, and greater requirements on integrated logic.

It is critical for all companies involved in manufacturing of computing systems or individual components to work together to improve current approaches and provide better validation of exchanged goods. It is also neccesary for companies to work together on a validation and attestation solution for hardware and firmware that can be conducted prior to integration into a larger system.

An assessment of the complexity of the attacks presented in this document versus the cost of proper mitigations is critical to identifying the simpliest and quickest improvements that can be made to strenghten resiliency to supply-chain threats. The information presented in this document should help support this manner of assessment and allow companies or organizations to identify deficiencies in their existing, supply-chain mitigations and develop a plan to address as many of these deficiencies as possible.