On the Suitability of Using SGX for Secure Key Storage in the Cloud

Brorsson, Joakim; Nikbakht Bideh, Pegah; Nilsson, Alexander; Hell, Martin

Published in:
Lecture Notes in Computer Science

DOI:
10.1007/978-3-030-58986-8_3

2020

On the Suitability of Using SGX for Secure Key Storage in the Cloud*

Joakim Brorsson[1,2,4](✉), Pegah Nikbakht Bideh[1], Alexander Nilsson[1,3], and Martin Hell[1]

[1] Lund University, Department of Electrical and Information Technology, Sweden
joakim.brorsson@eit.lth.se
[2] Combitech AB, Sweden
[3] Advenica AB, Sweden
[4] Hyker Security AB, Sweden

Abstract. This paper addresses the need for secure storage in virtualized services running in the cloud. To this purpose, we evaluate the security properties of Intel's Software Guard Extensions (SGX) technology, which provides hardware protection for general applications, for securing virtual Hardware Security Modules (vHSM). In order for the analysis to be comparable with analyses of physical HSMs, the evaluation proceeds from the FIPS 140–3 standard, the successor to FIPS 140–2, which is commonly used to assess security properties of HSMs.
We first make an evaluation using the FIPS 140–3 standard as is. After noting that the standard is designed for a stand-alone system rather than a virtual system, we propose a supplementary threat model, which can consider threats from different actors separately. This model allows for different levels of trust in actors with different capabilities and can thus be used to assess which parts of FIPS 140–3 that should be considered for a specific attacker.
Using FIPS 140–3 in combination with the threat model, we find that SGX enclaves provides sufficient protection against a large part of the potential actors in the cloud. Thus, depending on the threat model, SGX can be a helpful tool for providing secure storage for virtualized services.

1 Introduction

Secret keys used in cryptographic operations need to be safeguarded. If they are leaked to an attacker, many defense mechanisms are rendered ineffective. In systems with high security demands, Hardware Security Modules (HSMs) are often used for secure storage and key management. An HSM can securely isolate sensitive data from other parts of a system using *trusted hardware*, providing guarantees for integrity, confidentiality and isolation of cryptographic keys.

However, using HSMs comes with several drawbacks. First, they are expensive, since they contain specially designed hardware and require maintenance by

* This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

trained personnel. Second, they are purpose built, with a fixed limited set of operations. Third, they are not easily virtualizable, i.e. incorporating them into a virtual cloud environment without making custom configurations is difficult.

Recently, many applications and services have migrated from on-premise solutions to pure or hybrid cloud environments. The main motivation is that the cloud can provide more functionality and practically infinite computing resources without delay. It is possible to build applications and services which can automatically scale up and down with system needs. The infrastructure is handled by the cloud provider, and application owners only pay for what is used.

This points towards a mismatch between using cloud services and using HSMs for data protection. The cloud provides virtualized services on shared infrastructure, while HSMs are physical, single-tenant units. Being purpose built, an HSM can also not easily be continuously updated with better features. This mismatch fuels a need for alternative solutions for secure storage of secret keys in the cloud.

Intel Software Guard Extensions (SGX) [4] can be considered as an alternative technology for secure key storage in a cloud environment. SGX, like HSMs, leverages hardware protection mechanisms to isolate data.Since SGX is not purpose built like HSMs, the technology allows for running general purpose code within a hardware protected *enclave*. Systems based on the technology can therefore be easily virtualizable and continuously updated.

While the cost and functionality benefits that can be gained from virtualizing HSMs are clear, the security properties of such systems are not yet fully investigated. We therefore wish to evaluate the effectiveness of the security measures of SGX for protecting sensitive data in a cloud environment, where there are multiple actors with different levels of trust and capabilities, ranging from restricted application users to system administrators with physical access to servers.

The rest of this paper is organized as follows: We present related work in Sect. 2. Then, the background including details on HSMs, Intel SGX and FIPS 140–3 is given in Sect. 3. An evaluation of SGX for key storage purposes in a cloud environment using FIPS 140–3 is given in Sect. 4. In Sect. 5 we provide a new model for using FIPS 140–3 that consider attacker capabilities and intents. Then, in Sect. 6 we use the new model in combination with FIPS 140–3 and get a more nuanced conclusion. Finally, conclusions are given in Sect. 7.

1.1 Contribution

This paper provides two main contributions. First, we use the Federal Information Processing Standard FIPS 140–3 [13], designed for evaluating the security of cryptographic modules, to evaluate to what exent SGX technology can provide *secure storage*. For each applicable requirement in the standard, we classify the fulfillment of the requirement for SGX based secure storage modules.

Second, we consider a more flexible model, incorporating both different levels of trust and expected capabilities of other cloud actors. Using this model, we evaluate system security capabilities, using FIPS 140–3 as the basis for requirement fulfillment. This model and analysis can thus be used to better understand real world security capabilities of a system using SGX for secure storage.

2 Related Work

The general area of securing services running in the cloud is surveyed in e.g. [3,10,20]. These surveys discuss aspects of cloud security addressed in this paper, such as trust, multi-tenancy and malicious insiders. While [10] takes an attack focused approach and suggests TPM based solutions for trusted computing, [20] and [3] suggests SGX as an emerging solution, and roots their analysis in the capabilities and roles for different actors in a cloud environment, making them a suitable starting point for our analysis. It should be noted that SGX was a very new technology at the time these surveys were published.

The use of SGX for protecting containerized applications is proposed in SCONE and PANOPLY [1,21]. Similarly to this paper, the work explores SGX as a technique for creating trusted remote entities. While SCONE and PANOPLY give methods for creating containerized SGX enclave services, neither of the works consider side-channel or physical attacks as within scope for their security models, and can therefore not provide a full attacker model evaluation.

There has also been various suggestions for securing more specific cloud services and distributed use-cases with the technology [12,16,19]. Although these suggestions are dependent on the security properties of SGX, none of them present an attack model which considers the full capabilities of an attacker such as exploiting side-channels and having physical access. This is unfortunate since the security properties of SGX depends on the capabilities of an attacker.

A categorization of security properties of SGX in context of known attacks is given in [14], which surveys and categorizes the attacks on SGX and concludes that while there might come more attacks in the future, it is possible to mitigate the known attacks. Lindell [11] surveys attacks on SGX and concludes that the technology is not suitable as a protection mechanism or as an HSM replacement. In this paper, we instead argue that the suitability of SGX for data protection is not black or white, but instead use-case and attack model dependent, i.e. that SGX is not a suitable solution for all use-cases, but perfectly adequate for many.

3 Background

3.1 Hardware Security Modules

HSMs are hardware units which provide cryptographic functionality. They have physical protection measures, such as tamper proof coatings and separated input paths. Strong physical security measures, combined with strong security requirements, provides a base for trusting them not to leak or mishandle sensitive data. Further, the security of HSMs is often certified by a third party. One of the most common certifications to aim for is the FIPS 140–x standards [13]. Examples of functions supported by HSMs are key generation, key storage and random number generation. These functions are typically wrapped by a software API, which is responsible for giving access to the cryptographic functions.

While HSMs can give strong, certifiable, security guarantees, they are usually expensive, do not easily allow for multi-tenant usage, and can not be virtualized.

3.2 Software Guard Extensions

SGX is a set of CPU extensions that provide integrity and confidentiality for computations running on Intel processors. SGX provides isolated execution environments called *enclaves* to run code and operate on sensitive data, protected from the outside software environment.

While isolated execution environments can also be obtained using virtualization, such isolation techniques rely on concepts such as address translation, which puts full trust in the hypervisor or OS-kernel to control what physical memory is accessed when reading or writing to a virtual address. In a cloud scenario, this means that the cloud provider is fully trusted. The SGX threat model eliminates this trust by not trusting anything outside the software enclave, including the kernel, hypervisor and BIOS. On the physical side, no trust is placed on any component outside the CPU die (such as RAM modules).

However, physical attacks against the CPU module itself are not considered in Intel's threat model, nor are so-called side-channel attacks [4]. SGX has recently been subject to several attacks compromising the hardware isolation of data and code, most of these being side-channel based attacks. Although the SGX security model [4] does not specify side-channels stemming from application code as in-scope, some of these attacks are based on side-channels present in the micro architecture of the processor [6,22,23] and we must consider them. Side-channels originating from an improper sofware implementation is not in-scope, however.

SGX further provides functionality for *remote attestation*, where a user can convince herself that the enclave is running on genuine unmodified hardware, and that the software inside the enclave is indeed the expected one. This functionality is advantageous when running on shared hardware, as is our scenario. One can note that remote attestation is also available through use of a Trusted Platform Module (TPM). A TPM, however, has the drawback that the OS-kernel must be included in the Trusted Computing Base (TCB) for the measurement.

3.3 FIPS 140–3

FIPS 140–3 [13], is a standard issued by the American National Institute of Standards and Technology (NIST). It supersedes FIPS 140–2 and came into effect in September 2019. The standard specifies security requirements for systems protecting sensitive information, which has made it a common standard for evaluating the security of HSMs. The new FIPS 140–3 standard is more detailed on side channel attacks compared to the older version.

The scope of the standard is defined by a a set of security requirement areas, such as *authentication* and *physical security*. A *security level* is given as a measure of the quality of a security mechanisms. Depending on to what degree a system satisfies the *security requirements*, a system can be certified to *security level* 1–4.

FIPS 140–3 is not self contained, but consists of a series of other standards referencing each other. The relations between these can be seen in Fig. 1. FIPS 140–3 mandates the use of ISO 19790 [7] and ISO 24759 [9]. These standards specify,

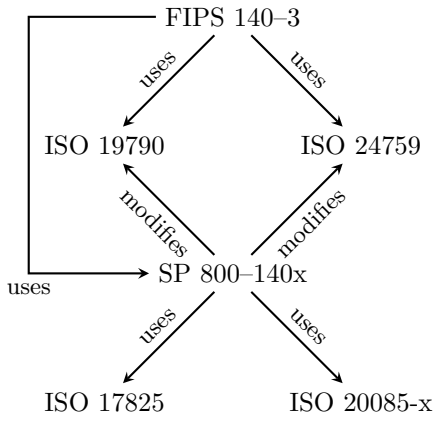

Fig. 1. Relationships between standards.

respectively, security requirements and test requirements for cryptographic modules. Not being fully satisfied with these ISO standards as-is, FIPS 140–3 also references a series of soon to be released NIST Special Publications, SP 800–140 A-F [15], which modify ISO 19790 and ISO 24759. They are currently available as drafts. These Special Publications in turn uses two other ISO standards as the basis for the modifications. These are ISO 17825 [8], which describes testing methods for side channel attack prevention mechanisms, and the soon to be released ISO 20085–1 and ISO 20085–2, which define test tool requirements.

To get FIPS 140–3 certified for a specific level, the system must comply with the requirements of that level as well as the requirements for lower levels.

Security **Level 1** is the basic security level with requirements analogue to those of standard production-grade systems.

Security **Level 2** adds physical security requirements through measures such as hard coatings, which will leave detectable traces of physical attacks, called tamper-evidence. Level 2 further requires role-based operator authentication.

In addition to tamper-evidence, **Level 3** requires *high* probability of tamper-detection and response, i.e. physical security that not only leaves evidence of abuse, but acts to prevent it. Further, identity-based authentication is required for operators, physical or logical separation is required for I/O ports for secret data, the system needs to protect sensitive data against compromise following environmental factors such as varying power supply or temperature, and it needs to provide protection against non-invasive attacks, i.e. side channel attacks.

Finally, **Level 4** requires *very high* probability of tamper detection with immediate and uninterruptible response, regardless if the system is connected to power or not. Further, multi-factor authentication for operators is required, the protection against environmental factors must be explicitly specified and tested, and there are higher testing requirements for mitigations of non-invasive attacks.

4 Evaluation Using FIPS 140–3

We use the following method to use FIPS 140–3 for determining (1) the security of a cryptographic module when applied to a certain scenario or use case, and (2) how the security depends on a given threat model.

1. Define the scenario.
2. Identify the FIPS 140–3 requirements that need particular attention for the given scenario, and evaluate the attainable security level.
3. Define the threat model in terms of attacker capabilities and intents.
4. Evaluate the identified FIPS 140–3 requirements in relation to the threat model and determine the security level attainable for each requirement.

Both step 2 and step 4 in this process will require interpretations of the FIPS 140–3 standard, taking both security and test requirements into account. Since we are interested in the attainable level, we also need to rely on assumptions on algorithms, protocols and implementations. Thus, it is important to clearly document these interpretations and assumptions in order to be transparent when determining the security level. In the following, we demonstrate how this methodology is used to evaluate the attainable security for using SGX for key storage in the cloud.

4.1 Scenario and Threat Model

This paper investigates whether there is a virtualizable alternative to HSMs, which can provide similar security features. Therefore, the evaluation considers the concrete scenario of a virtual HSM, hosted on a public cloud, implemented as an SGX enclave, and used for secure key storage. In choosing this scenario, we imitate a real public cloud system, and introduce the full set of present actors. The actors present in this scenario are then: the cloud provider and its personnel, the application owner and users, and owners and users of other co-hosted services on the same cloud.

We used a system model as illustrated in Fig. 2 to evaluate our selected scenario explained above. We base this model on the findings in [20], which also discusses threat models for virtualized systems in a cloud environment. Note that the SGX protection mechanisms reside in the hardware layer, so that while an appliction runs in the guest layer, the protection mechanisms are rooted in hardware. The FIPS 140–3 security requirements, as they are written. Therefore, our threat model does not differentiate between actors in this evaluation. We consider all actors potentially malicious and thus bring all FIPS 140-3 related security measures into scope. An extended model which differentiates between actors will be discussed later in Sect. 5.2.

4.2 Identifying FIPS 140–3 Requirements

FIPS 140–3 includes a lengthy set of *security requirements*, specifying things such as roles of operators, the finite state model of the system, etc. For the sake

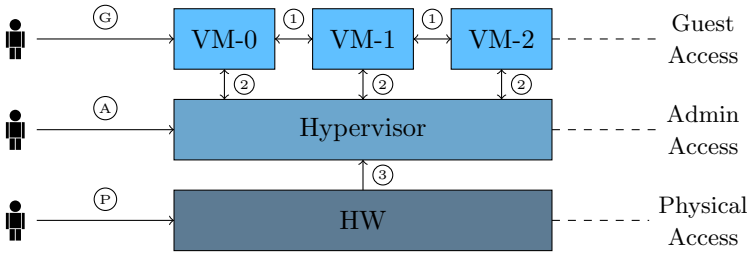

Fig. 2. Cloud System Model.

of brevity, we here limit ourselves to discussing the requirements which are of interest when comparing HSMs and SGX based systems for key storage.

Based on our interpretations while analyzing the requirements, we find the following sections for which the outcome can differ when applied to HSMs or SGX based solutions.

- **FIPS 140–3 section 7.3: Cryptographic Module Ports and Interfaces** These requirements handle separation of I/O ports.
- **FIPS 140–3 section 7.7: Physical Security** The Physical Security requirements address tamper-proofing, -evidence and -detection.
- **FIPS 140–3 section 7.8: Non-invasive Security** This section concerns mitigating passive side-channel attacks.
- **FIPS 140–3 section 7.9: Sensitive Security Parameter Management** These are requirements on input and output of sensitive data.
- **FIPS 140–3 section 7.12: Mitigation of Other Attacks** These requirements concern resistance towards attacks not specified in the standard. Depending on the type of attack, there might be differences in implementation.

We go through the selected requirement areas individually and discuss how an SGX based system can comply with the requirements. The standard also makes heavy use of *test requirements* which mandate how the security requirements are meant to be evaluated. These give great insight into how the security requirements are to be interpreted. In the following sections, these are only referenced implicitly. In our evaluation, for brevity, we focus on only security requirements.

4.3 Evaluating FIPS 140–3 requirements

Cryptographic Module Ports and Interfaces: Level 1–2 requires logically distinct interfaces, which can be implemented as e.g. an API, and restriction of information flow to points identified as entry and exit. Different required interfaces are specified, e.g. control input interface for function calls. There are also requirements for the specification of all interfaces and of all input and output data paths. For achieving level 3 there is a requirement to implement a *trusted channel* for protected input and output of sensitive data with physical or logical

separation from other channels. This channel further needs to have identity-based authentication. Level 4 additionally requires multi-factor authentication for using a trusted channel.

Conclusion Since the requirements allow for logical separation of interfaces, SGX is well documented, and authentication is implementation independent, we see no problem in implementing security up to level 4 using SGX. Note that attacks on trusted channels such as TLS is outside the scope of our threat model.

Physical Security: According to ISO 19790, software based cryptographic modules are not subject to requirements for physical security. The maximum allowed level for software based systems is level 2. For a hardware based system it is required for level 1 that there are production grade components. For level 2 physical tampering should leave evidence on an opaque enclosure. It is possible to reach level 3 if there is *environmental failure testing*, where fluctuations in environmental factors such as varying power supply or temperature is tested to not have any impact on the security. Level 3 also specifies that zeroization of sensitive data should occur in case of tampering. For level 4 the system must additionally provide *environmental failure protection* which includes active monitoring and immediate response to environmental changes. Additionally, protection against fault induction must be documented for this level.

Conclusion Since Intel CPUs include glued-on heatsinks, which are difficult to remove, we conclude that level 2 ought to be attainable.

Depending on the requirements on the coating and fault injection, which are not clearly stated, SGX based modules will probably not live up to level 3. Regarding environmental failure testing and protection Intel SGX will not be able to reach level 3 or 4.

Non-invasive Security: For security level 1 & 2, it is required to document mitigation techniques used for non-invasive attacks, otherwise known as passive side-channel attacks. ISO-19790 does not in itself specify any list of approved non-invasive attack mitigation test metrics.

For security level 3 & 4, ISO 17825 should be used for evaluating the protection against non-invasive attacks. The standard specifies 3 main attack classes: Timing, Power-Consumption and Electro-magnetic emission.

Conclusion Regarding non-invasive attacks (i.e. passive side-channel attacks) using *timing* information, it is possible to use standardized cryptographic libraries which are designed to be constant time. Note that for the SGX scenario, special care must be taken in order to also be secure against *cache*-timing and other microarchitectural timing-based attacks. Using an up-to-date and well vetted industry standard cryptographic library which includes these kind of attacks in its threat model should be sufficient to enable an SGX based system to pass the level 4 security requirements considering only timing based attacks.

For electromagnetic emission and power consumption based analysis, Intel SGX — which is implemented inside regular Intel manufactured x86 CPUs — is not designed to protect against such attacks. Such attacks have been found, see for example [5,2]. For this reason we would not expect SGX to be resistant against these classes of attacks. However some work have been proposed which

would allow for software based mitigations against these types of attacks [18]. It remains an open research question whether or not such mitigations would be effective in our proposed scenario. For these reasons, we are hesitant to offer a clear conclusion of what FIPS 140–3 security level the certification process would ultimately end up with. We would like to point out however that it would not be unreasonable to argue that level 1 & 2 are passable since those requirements are only for documentation about reasonable protections, while 3 & 4 require vetting and extensive testing according to ISO 17825.

Sensitive Security Parameter Management: For obtaining security level 1, ISO 19790 specifies that input and output of sensitive data must be done only through well defined and documented channels. See ISO 19790 section 7.9.5. for more details. It also requires role-based authentication and that it is possible for a module operator to zeroize (securely erase) sensitive data independently of the system operation.

To obtain level 2, the system must additionally perform zeroization of temporary or unprotected sensitive data if and when the data is no longer needed. Level 3 additionally specifies that input and output of sensitive data must be done either through a trusted channel or by using encryption. Further, an identity-based authentication method is required for level 3.

In level 4, it is also required that the system uses multi-factor identity-based operator authentication for inputting or outputting of data. Further, the system must be able to zeroize the above mentioned data without being interrupted and the process must be immediate.

Conclusion Since it is possible for Intel SGX based systems to zeroize by overwriting all sensitive data, not otherwise physically or logically protected by the system, we judge that level 3 is obtainable in regard to the zeroization requirements. Level 4 is not obtainable in situations where the process can be interrupted and prevented from executing.

Mitigation of Other Attacks: For level 1, 2 & 3, the system under evaluation must provide documentation on mechanisms protecting against specific attacks not found elsewhere in the FIPS 140–3 standard. The documentation must include an enumeration of all possible known attacks against the implementation.

For level 4, the documentation additionally must include the methods used for testing the effectiveness of the mitigation techniques. One notable consequence of these requirements is that if no attacks are known then level 4 is reachable.

Conclusion Because of the known attacks, SGX based systems will have to provide mitigations and documentations about these attacks. Since mitigations[5] exist for all attacks known to the authors (just to mention a few: [6,22,23]), we consider level 3 to be obtainable for these requirements.

It is unclear whether or not level 4 is obtainable in practice, since many of the attacks and their mitigations are difficult to practically implement and test. The level 4 security requirements listed in ISO 17825 are too vague for us to

[5] For some cache-based attacks the mitigation must be implemented in the software implementation since the manufacturer (Intel) has not considered such attacks as in-scope for SGX's threat model.

Table 1. Summery of level fulfillment according to FIPS 140–3 requirements

Requirement areas	Level 1	Level 2	Level 3	Level 4
Cryptographic Module Ports and Interfaces				✓
Physical Security		✓		
Non-invasive Security		✓		
Sensitive Security Parameter Management			✓	
Mitigation of Other Attacks			✓	

evaluate without performing a full FIPS 140–3 certification. Since ISO 17825 is used for evaluating the protections against non-invasive attacks, we theorize that it is possible to use it for evaluating the more invasive side-channel attacks as well. This would be reasonable since both attack classes depend on the same type of biases (timing, or otherwise).

4.4 FIPS 140–3 Evaluation Results

Building on the intermittent conclusions above we can summarize the level of fulfillment for the different requirement areas as illustrated in Table 1. It should be noted that we here summarize our most optimistic views and we refer back the respective sub-sections in Sect. 4 for further discussions.

We believe that an Intel SGX based system might be able to reach level 1 & 2 due to meeting the requirements of level 1 & 2 for all requirment areas. However level 3 & 4 are not obtainable since the requirements are not fulfillable for all requirement areas.

5 A Threat Based Security Model

5.1 Motivation

Secure key storage in the cloud comes with a threat model which contrasts with on-premise solutions in that it introduces new actors (enumerated in Sect. 4.1) to the system.

To evaluate the security of a system, the evaluation requirements must be clear. We need an evaluation model that takes the system use case into account and can be used to decide what security capabilities are required from a system. The FIPS 140–3 standard is good for evaluating the *capabilities* of a security system. However, we find that it needs complementing for evaluating a system with different levels of trust in different actors, which we argue is crucial for cloud use cases. In order to compare the security of SGX-based key storage to other methods, such as HSMs, we need a complementary security evaluation framework to be used with FIPS 140–3.

Table 2. The attack model

	Honest	Curious	Malicious
Guest Access	G+H	G+C	G+M
Admin Access	A+H	A+C	A+M
Physical Access	P+H	P+C	P+M

We suggest a model designed to address security in a cloud environment. It is based on a model where the threat from an actor is defined as a combination of *trust* in the actor in combination with its *capabilities* to perform an attack. The model can be used to complement FIPS 140–3, by allowing the security classifications to take the perspective of what threat model a service can handle, in contrast to only measuring inherent system capabilities directly. Thus, the suggested threat model allows for varying degrees of trust in different actors.

5.2 Model Description

Based on the motivation above, we define a two-dimensional attack model, taking into account that actors in a cloud scenario can have different *capabilities* and different levels of *trust* from the application designer. The attack model is given in Table 2. The *capabilities* are based on access levels and are defined as follows.

- **Guest Access (G).** A remote attacker with admin or user level access to a specific virtual service.
- **Admin Access (A).** An attacker with hypervisor-level remote administrative access to the physical machine hosting the VM.
- **Physical Access (P).** An attacker with physical access to the physical machine hosting the VM.

The second dimension specifies the *trust* that the application designer has in the actor in the form of expected maliciousness or honesty. Trust is divided into three categories.

- **Honest (H).** The actor is expected to abide by any rules as agreed on with the application designer. This includes the expectation that this actor will not attempt any bypass of technical protection mechanisms.
- **Curious (C).** The actor is expected to try to circumvent rules and technical protection mechanism by mounting non-invasive or passive attacks.
- **Malicious (M).** The actor is expected to attempt all existing attacks.

In Fig. 2, the capabilities corresponds to the three horizontal layers. An actor has to interact with the system through the corresponding interface (Ⓖ, Ⓐ or Ⓟ). Further there are attack paths (①, ② and ③) to consider, where an attacker can traverse between different entities on the same access level (①),

and jump between access levels (② and ③). For example, co-hosted services is an actor which has to be considered in the cloud. In [17], it is deemed feasible to get co-hosted with a cloud service of your choice, allowing attacks through ① or ② on neighbouring applications. Another attack vector, not present in on-premise systems, is a malicious service provider, i.e. a hostile cloud, which can access virtual services by an attack through e.g. ②. For further discussions of capabilities and possible attacks, we refer to [20].

By combining capabilities and trust, security requirements can be defined with more flexibility, allowing us to target more specific use-cases. An example of this is in the scenario where you trust the cloud provider to be *honest* but consider co-hosted VMs to be potentially *malicious*. In this scenario, requirements would be different compared to if you had to abide by the lowest trust level (malicious) for all actors.

The security model discussed here considers the confidentiality and integrity of data. The requirements for availability of systems are considered out of scope, which is consistent with the attack models for both Intel SGX and FIPS 140–3.

6 Mapping our Security Model to FIPS 140–3

Using FIPS 140–3 for evaluating the suitability of a system does not take the specific scenario into consideration. Thus, in this section, based on our earlier evaluation of SGX using FIPS 140–3 in Sect. 4, we adapt the scope of FIPS 140–3 to a cloud scenario by taking the threat model into account.

6.1 Methodology

In order to map our threat model to FIPS 140–3, for each category in Table 2, we take the the threat model into consideration when evaluating the relevance and fulfillment of a requirement. For example, if an actor is honest, the full set of requirements are out of scope, and if an actor does not have physical capabilities, requirements for physical security and non-invasive attacks based on power trace and EM are out of scope. From this we get a security level for each requirement category for each actor. This method is analogous to how FIPS 140–3 excludes its own requirements in physical security for pure software based cryptographic modules which implicitly renders the Operating System a trusted actor.

6.2 Mapping the Threat Model

The result of mapping our threat model onto the scenario can be seen in Table 3. The possible security level for each actor is based on the conclusions in the relevant requirement category in Sect. 4.

For *Cryptographic Module Ports and Interfaces*, an SGX based implementation is capable of reaching level 4 for all actors considered.

For *Physical Security*, the only actors with the required capabilities are the physical actors (P+C and P+M). For these actors, only the malicious (P+M)

has a possible intent to mount such an attack since it requires an active intrusion. As a result, level 2 can be achieved for (P+M). No other actors are relevant here.

For the *Non-invasive Security*, we split the timing related requirements from the ones related to EM and power emissions. For timing attacks, we find that level 4 is reachable for SGX based systems. For EM/power however, it is concluded that only level 2 is reachable. This is because the EM/power analysis attacks uses physical access. Therefore, here, we can consider the attack channel as not applicable for all actors except those with physical access. For both (P+C) and (P+M), the attacks are feasible since they are passive attacks.

In *Sensitive Security Parameter Management*, level 4 requires zeroization of data being uninterruptible. Considering different capabilities of actors, we can conclude that actors with guest level capabilities, e.g. co-hosted services, can not mount such attacks and can therefore achieve level 4. Further, curious actors are not expected to mount such invasive attacks. The system is therefore given level 4 with regards to those actors. Level 3 is obtainable for all remaining actors.

Finally, we consider *Mitigation of Other Attacks*. Curious actors are not expected to perform invasive or active attacks. By this reasoning level 4 is, by our estimate, achievable for all curious actors. There exists a small number of recently published attacks (for example, [22,23]) that ought to be achievable[6] from inside a malicious co-hosted VM (G+M), luckily they appear to have been mitigated by Intel microcode updates. Still, by the reasoning given in Sect. 4.3 ("Mitigation of other attacks") we put level 3 as the maximum achievable level that can be obtained. For the remaining actors and capabilities (A+M, P+M), we reuse the same reasoning to achieve level 3 since there is such a broad range of attacks possible here. We would like to point out, however, that different conclusions might be drawn depending on the level of effort one put into the testing of the proposed mitigation that exist for the corpus of known attacks. Level 4 is not out of the question, especially not for the G+M combination which is affected by a far smaller number of attacks.

7 Conclusions

In this paper we have used FIPS 140–3 to evaluate the suitability of SGX as a method for keeping sensitive data secure in a cloud environment. While the standard is well suited to assess the capabilities of a system executing in an isolated environment, it is not sufficient to provide the nuanced judgment needed in a cloud scenario with actors of different levels of trust and capabilities.

When using our security model to complement FIPS 140–3, we reached a more nuanced view, where the scenario and threat model is taken into account when evaluating the security of a system.

[6] These attacks can leak secrets from VM boundaries and even SGX boundaries, what is not so clear however is whether or not the combination of the two technologies would be a significant hinderance for the attacker. We have here elected to use the most pessimistic interpretation.

Table 3. Mapping FIPS 140–3 to our attack model

		G+C	G+M	A+C	A+M	P+C	P+M
Cryptographic Module Ports and Interfaces		4	4	4	4	4	4
Physical Security		N/A	N/A	N/A	N/A	N/A	2
Non-invasive Security for:	Timing	4	4	4	4	4	4
	Power/EM	N/A	N/A	N/A	N/A	2	2
Sensitive Security Parameter Management		4	4	4	3	4	3
Mitigation of Other Attacks		4	3	4	3	4	3

Based on considerations in this paper, for systems with a threat model including a curious or malicious cloud provider attempting to illegally obtain sensitive data from the customer, SGX is deemed not suitable. The main reasons for this is due to our model and FIPS 140–3 classification of EM and power analysis as non-invasive attacks. These reasons led to the conclusion that an SGX-based system could not live up to FIPS 140–3 level 3 for Power/EM analysis in Non-invasive Security for actors with physical access, which is the lowest level requiring tested protections against these attacks. The same reasoning is applicable to the Physical Security requirements, where level 3 is the lowest level with active data zeroization requirements. If you consider those kind of attacks to be outside the scope of your threat-model then SGX would be a suitable choice of protection mechanism.

Thus HSMs can better protect against adversaries with physical access. However for all the other actors, including curious and malicious co-hosted services and attackers inside the same VM, SGX is deemed to successfully protect sensitive data from those actors.

References

1. Arnautov, S., Trach, B., Gregor, F., Knauth, T., Martin, A., Priebe, C., Lind, J., Muthukumaran, D., Stillwell, M.L., Goltzsche, D., Eyers, D., Pietzuch, P., Fetzer, C.: SCONE: Secure Linux Containers with Intel SGX. osdi pp. 689–704 (2016)
2. Callan, R., Popovic, N., Daruna, A., Pollmann, E., Zajic, A., Prvulovic, M.: Comparison of electromagnetic side-channel energy available to the attacker from different computer systems. In: IEEE Int. Symp. Electromagn. Compat. vol. 2015-Septm, pp. 219–223. IEEE (2015)
3. Coppolino, L., D'Antonio, S., Mazzeo, G., Romano, L.: Cloud security: Emerging threats and current solutions. Computers & Electrical Engineering **59**, 126–140 (2017)
4. Costan, V., Devadas, S.: Intel sgx explained. IACR Cryptology ePrint Archive **2016**(086), 1–118 (2016)
5. Genkin, D., Pipman, I., Tromer, E.: Get your hands off my laptop: physical side-channel key-extraction attacks on PCs: Extended version. J. Cryptogr. Eng. **5**(2), 95–112 (2015)

6. Huo, T., Meng, X., Wang, W., Hao, C., Zhao, P., Zhai, J., Li, M.: Bluethunder: A 2-level Directional Predictor Based Side-Channel Attack against SGX. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2020(1) (2019)
7. International Organization for Standardization: ISO/IEC 19790:2012: Information technology — security techniques — security requirements for cryptographic modules (2012)
8. International Organization for Standardization: ISO/IEC 17825:2016: Information technology — security techniques — testing methods for the mitigation of non-invasive attack classes against cryptographic modules (2016)
9. International Organization for Standardization: ISO/IEC 24759:2017: Information technology — security techniques — test requirements for cryptographic modules (2017)
10. Khan, M.A.: A survey of security issues for cloud computing. Journal of network and computer applications **71**, 11–29 (2016)
11. Lindell, Y.: The security of intel sgx for key protection and data privacy applications. Tech. rep. (2018), `https://cdn2.hubspot.net/hubfs/1761386/Unbound_Docs_/security-of-intelsgx-key-protection-data-privacy-apps.pdf`
12. Mokhtar, S.B., Boutet, A., Felber, P., Pasin, M., Pires, R., Schiavoni, V.: X-search: revisiting private web search using intel sgx. In: Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference. pp. 198–208 (2017)
13. National Institute of Standards and Technology: Fips 140-3: Security requirements for cryptographic modules (2018)
14. Nilsson, A., Nikbakht Bideh, P., Brorsson, J.: A survey of published attacks on intel sgx. Tech. rep. (2020), `http://lup.lub.lu.se/record/a6d6575f-ac4f-466f-8582-48e1fe48b50c`
15. (NIST), K.S.: SP 800-140F(draft): CMVP approved non-invasive attack mitigation test metrics: CMVP validation authority updates to ISO/IEC 24759:2014(E) (2019)
16. Priebe, C., Vaswani, K., Costa, M.: EnclaveDB: A Secure Database Using SGX. In: Proc. - IEEE Symp. Secur. Priv. vol. 2018-May, pp. 264–278 (2018)
17. Ristenpart, T., Tromer, E., Shacham, H., Savage, S.: Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In: Proceedings of the 16th ACM conference on Computer and communications security. pp. 199–212. ACM (2009)
18. Saab, S., Rohatgi, P., Hampel, C.: Side-Channel Protections for Cryptographic Instruction Set Extensions. IACR Cryptology ePrint Archive **2016**, 700 (2016)
19. Schuster, F., Costa, M., Fournet, C.C., Gkantsidis, C., Peinado, M., Mainar-Ruiz, G., Russinovich, M.: VC3: Trustworthy Data Analytics in the Cloud Using SGX. In: 2015 IEEE Symp. Secur. Priv. vol. 2015-July, pp. 38–54. IEEE (2015)
20. Sgandurra, D., Lupu, E.: Evolution of attacks, threat models, and solutions for virtualized systems. ACM Computing Surveys (CSUR) **48**(3), 1–38 (2016)
21. Shinde, S., Chua, Z.L., Narayanan, V., Saxena, P.: Preventing Your Faults From Telling Your Secrets: Defenses Against Pigeonhole Attacks. arxiv.org (2015)
22. Van Schaik, S., Minkin, M., Kwong, A., Genkin, D., Yarom, Y.: CacheOut: Leaking Data on Intel CPUs via Cache Evictions. cacheoutattack.com p. 16 (2020)
23. Weisse, O., Bulck, J.V., Minkin, M., Genkin, D., Kasikci, B., Piessens, F., Silberstein, M., Strackx, R., Wenisch, T.F., Yarom, Y.: Foreshadow-NG: Breaking the Virtual Memory Abstraction with Transient Out-of-Order Execution. Proc. 27th USENIX Secur. Symp. **0** (2018)