Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Eidgenössisches Departement für Verteidigung, Bevölkerungsschutz und Sport VBS

Bundesamt für Cybersicherheit BACS
Analyse und Prävention

5. Mai 2025 (revidiert)

Technologiebetrachtung

Confidential Computing

1 Einführung

Die Sicherheit informationsverarbeitender Prozesse in IT-Systemen stellt insbesondere vor dem Hintergrund einer zunehmenden Nutzung von «Cloud Computing» und entsprechenden Dienstleistungen ein grosses Problem dar. Während die Speicherung und Übertragung von Daten mit Hilfe von kryptografischen Verfahren gut gesichert werden kann, stellt die sichere Datenverarbeitung[1] immer noch eine der zentralen Herausforderungen dar: Wie können Daten verarbeitet werden, ohne dass man die die Verarbeitung steuernde Software (Code) und die von einem Cloud-Anbieter zur Verfügung gestellte Ausführungsumgebung effektiv kontrollieren kann? In der Vergangenheit sind dazu verschiedene Lösungsansätze entwickelt und teilweise auch umgesetzt worden.

Im Rahmen dieser Technologiebetrachtung wird mit «Confidential Computing» einer dieser Lösungsansätze kurz vorgestellt und diskutiert, d. h. es wird aufgezeigt, worum es geht, wie der Ansatz im Prinzip funktioniert, was die Vor- und Nachteile sind, welche konkreten Angebote es auf dem Markt gibt und wie die Zukunftsaussichten einzuschätzen sind.

2 Problemstellung

Wie einleitend erwähnt, stellt die sichere Verarbeitung von Daten in einem IT-System eine grosse Herausforderung dar. Dabei müssen sowohl die Integrität der Umgebung, in der die Anwendung ausgeführt wird (d. h. die Ausführungsumgebung), als auch die Vertraulichkeit der Daten während ihrer Verarbeitung gewährleistet sein. Gemäss Abbildung 1 umfasst die Ausführungsumgebung nicht nur die Anwendung selbst (inklusive Daten und Code), die typischerweise in einer virtuellen Maschine (VM) ausgeführt wird, sondern auch das Gast-Betriebssystem der VM, den Hypervisor, das Host-Betriebssystem und die Hardware.[2] All diese Komponenten müssen vor einer Kompromittierung und Exfiltration von Daten geschützt sein,

[1] Man spricht in diesem Zusammenhang auch etwa von «Data in Use» (gegenüber «Data at Rest» für gespeicherte und «Data in Transit» für übertragene Daten).

[2] Wenn auf den Einsatz einer VM und damit auf eine möglichst weitgehende Virtualisierung verzichtet wird, vereinfacht sich die Ausgangslage und die Anwendung kann dann direkt auf der Hardware und dem Host-Betriebssystem betrieben werden.

damit die Sicherheit der Anwendung nicht unterwandert werden kann. Entsprechend sind die in Abbildung 1 rot eingefärbten Komponenten für die Sicherheit der Datenverarbeitung kritisch und müssen als vertrauenswürdig[3] angenommen werden.

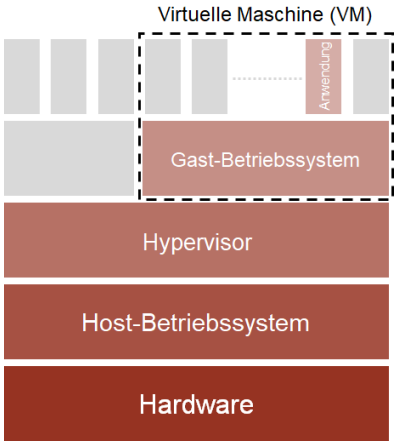

Abbildung 1: Verarbeitung von Daten in einem IT-System

Wenn eine Anwendung nicht lokal («On Premises») ausgeführt wird, sondern stattdessen die Dienstleistungen eines Cloud-Anbieters in Anspruch genommen werden, dann muss der Datenverarbeiter nicht nur auf die Vertrauenswürdigkeit seiner eigenen IT-Systeme zählen können, sondern auch auf die Vertrauenswürdigkeit dieses Anbieters und dessen IT-Systeme. Diese Situation ist in Abbildung 2 schematisch dargestellt. Der Datenverarbeiter sendet Daten und Code zum Cloud-Anbieter, der in einer von ihm bereitgestellten Ausführungsumgebung den Code auf die Daten anwendet und dem Datenverarbeiter am Schluss das Resultat (auf eine gesicherte Art und Weise) zurückgibt. Dabei hat der Datenverarbeiter in der Regel keine Möglichkeit, die Ausführungsumgebung und deren Sicherheit direkt zu kontrollieren, d. h. er ist hier auf das Versprechen des Cloud-Anbieters und entsprechende Bescheinigungen und Zertifikate angewiesen.

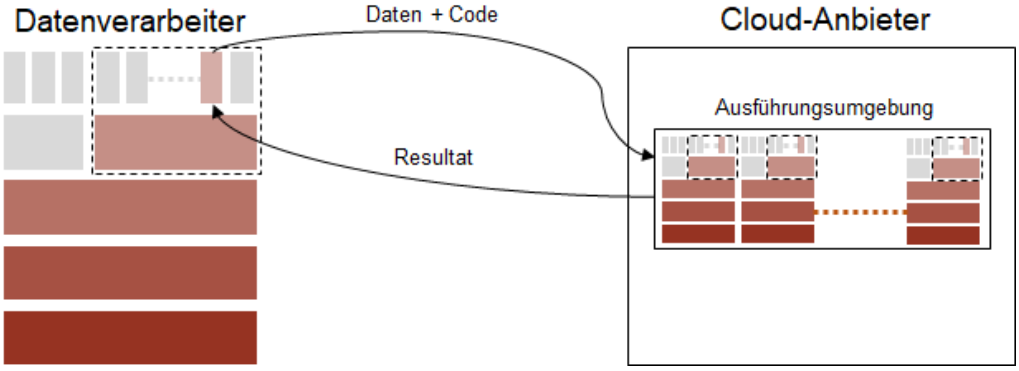

Abbildung 2: Die Ausführung einer Anwendung im Rahmen von «Cloud Computing»

[3] In der IT-Sicherheit müssen die (System-)Komponenten, deren Sicherheit man weder beweisen noch verifizieren kann, als vertrauenswürdig angenommen werden. Sie stellen in ihrer Gesamtheit die Trusted Computing Base (TCB) dar, die so klein wie möglich gehalten werden muss.

Die dieser Situation zugrundeliegende Problemstellung wird in der Literatur auch etwa als «Secure Remote Computation» umschrieben: Wie kann auf einer fremdbetriebenen und nicht direkt kontrollierbaren IT-Infrastruktur eine vertrauenswürdige Ausführungsumgebung bereitgestellt und die Exfiltration von Daten wirksam verhindert werden, so dass insgesamt eine «sichere» Datenverarbeitung resultiert? Diese Frage stellt die Umkehrung der Problemstellung dar, die sich mit dem Schutz einer vertrauenswürdigen IT-Infrastruktur vor potenziell bösartigem Code (z. B. Java-Applets) befasst und mit Sandboxing wenigstens teilweise gelöst werden kann. Beim Sandboxing wird die Ausführungsumgebung minimal gehalten und so weit wie möglich vom Rest der IT-Infrastruktur abgeschottet (isoliert), damit potenziell bösartiger Code diese nicht beeinflussen oder kompromittieren kann. Sandboxing ist für das «Secure Remote Computation» keine Option, weil es eben nicht die IT-Infrastruktur ist, die geschützt werden muss, sondern die Daten und der Code. Für das «Secure Remote Computation»-Problem sind andere und meist auch komplexere Lösungsansätze erforderlich.

3 Lösungsansätze

Langfristig wird die Lösung des «Secure Remote Computation»-Problems wohl in voll homomorphen Verschlüsselungsverfahren liegen. Mit solchen Verfahren kann der Datenverarbeiter dem Cloud-Anbieter seine Daten so in verschlüsselter Form zur Verfügung stellen, dass dieser die Daten zwar verarbeiten aber nicht entschlüsseln kann, d. h. es ist dann sichergestellt, dass der Cloud-Anbieter die Daten nicht im Klartext sieht. Am Ende der Datenverarbeitung wird das Resultat in immer noch verschlüsselter Form dem Datenverarbeiter zurückgegeben, so dass dieser das Resultat entschlüsseln kann. Damit lässt sich eine Software-mässig realisierbare Lösung für das «Secure Remote Computation»-Problem konstruieren. Leider sind die heute bekannten und zur Verfügung stehenden voll homomorphen Verschlüsselungsverfahren noch zu wenig effizient, als dass sie für allgemeine Datenverarbeitungsaufgaben genutzt werden könnten.

Bis praxistaugliche voll homomorphe Verschlüsselungsverfahren verfügbar sind, kann versucht werden, das «Secure Remote Computation»-Problem mit Hilfe von vertrauenswürdigen Hardware-Komponenten zu lösen. Damit lässt sich die Integrität einer Ausführungsumgebung bis zu einem bestimmten Punkt gewährleisten. Vertrauenswürdige Hardware-Komponenten gibt es in verschiedenen Ausführungen, wie z. B. Smartcards, Sicherheitselemente, Hardware-Sicherheitsmodule und Trusted Platform Modules (TPMs). Letztere sind im Rahmen der «Trusted Computing»-Initiative der Trusted Computing Group (TCG[4]) spezifiziert und breit verbaut worden. So kann auf handelsüblichen IT-Systemen z. B. mit Hilfe eines TPM sichergestellt werden, dass dieses in einen definierten Zustand aufstartet («bootet») und entsprechend nicht von manipulierter Software kompromittiert worden ist. Damit kann die Integrität einer Ausführungsumgebung gesichert werden – wenigstens so lange, bis nicht bekannte Software in das System geladen und ausgeführt wird. Weil dies aber früher oder später geschehen wird, ist der Aufwand, den die Nutzung eines TPM verursacht, gegenüber seinem Nutzen doch relativ hoch.

Eine einfachere und breiter einsetzbare Möglichkeit zur Hardware-Unterstützung ist von einem Industriekonsortium[5] unter der Bezeichnung «Confidential Computing» vorgeschlagen worden. Dabei bietet die Hardware die Möglichkeit, eine sogenannte «Trusted Execution Environment» (TEE) als «sichere Enklave» zu betreiben, in der eine Datenverarbeitung in einem gesicherten Modus stattfinden kann, d. h. die Verarbeitung findet dann nur im Prozessor statt

[4] https://trustedcomputinggroup.org

[5] https://confidentialcomputing.io

und keine Software ausser die die TEE betreibende Anwendung kann auf die TEE zugreifen oder Daten von dort auslesen. Zudem kann die Anwendung sowohl die TEE als auch den Code, der in der TEE ausgeführt werden soll, vor der Einspielung von Daten validieren (man spricht in diesem Zusammenhang von «Remote Attestation»). Damit Zustandsvariablen und Resultate dennoch zwischengespeichert werden können, unterstützt die TEE ein sogenanntes «Sealing», d. h. eine kryptografische Absicherung von Daten in einer dafür vorgesehenen Speicherablage. Die Schlüssel, die für «Remote Attestation» und «Sealing» erforderlich sind, existieren nur in der Hardware bzw. in den entsprechenden TEEs und können von dort nicht bzw. nur mit unverhältnismässig grossem Aufwand ausgelesen werden.

Die Ausführung einer Anwendung unter Zuhilfenahme einer TEE ist in Abbildung 3 dargestellt. Man beachte, dass die einzigen vertrauenswürdigen Komponenten in diesem Szenario die Anwendung selbst und die TEE sind (beide sind in der Abbildung denn auch rot eingefärbt). Insbesondere kann in diesem Szenario der Betreiber des IT-Systems die Daten der Anwendung nicht einsehen – wenigstens solange die Hardware die geforderten Eigenschaften von «Confidential Computing» korrekt umsetzt. Konkret bedeutet das, dass dem Lieferanten der Hardware vertraut werden muss.

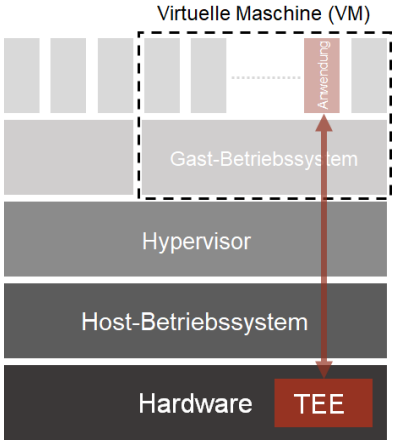

Abbildung 3: Die Ausführung einer Anwendung unter Zuhilfenahme einer TEE

Besonders interessant und erfolgversprechend scheint der Einsatz von «Confidential Computing» in einer Cloud-Umgebung zu sein: Die in Abbildung 2 dargestellte Situation ist um Confidentinal Computing erweitert und in Abbildung 4 schematisch dargestellt. Demnach werden Daten und Code vom Datenverarbeiter so in die TEE des Cloud-Anbieters eingespiesen, dass sie End-zu-End verschlüsselt sind und erst in der TEE entschlüsselt werden. Weil sie auch erst in der TEE ausgeführt werden, hat der Cloud-Anbieter als Betreiber der Ausführungsumgebung bzw. TEE keine Möglichkeit, die Daten einzusehen oder den Code und damit die Datenverarbeitung zu kompromittieren. Die TEE stellt damit eine gesicherte Ausführungsumgebung zur Verfügung, deren Integrität die Anwendung selbst validieren kann. Mit Hilfe von «Remote Attestation» kann sichergestellt werden, dass sowohl die TEE als auch der dort ausgeführte Code authentisch sind und die Datenverarbeitung korrekt und im Sinne des Datenverarbeiters verläuft. Das Resultat wird der Anwendung direkt von der die Datenverarbeitung leistenden TEE zugestellt. Natürlich bleibt die Sicherheit der Datenverarbeitung auf der Seite des Kunden (bzw. des Datenverarbeiters in Abbildung 4) abhängig von allen Komponenten. Auf der Seite des Cloud-Anbieters können aber die Abhängigkeiten und erforderliche Vertrauenswürdigkeit stark reduziert werden. Damit kann sich auch der Cloud-Anbieter aus der Verantwortung nehmen, weil er grundsätzlich keine Möglichkeit hat, Daten einzusehen oder Code zu kompromittieren.

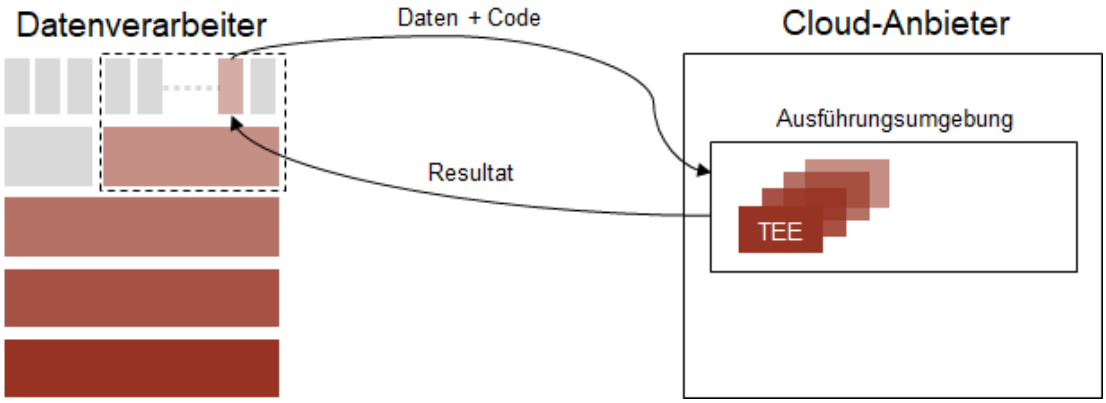

Abbildung 4: Die Ausführung einer Anwendung unter Zuhilfenahme einer TEE in einer Cloud-Umgebung

4 Marktübersicht

Der Begriff «Confidential Computing» bezeichnet den zuletzt beschriebenen Technologie- beziehungswiese Lösungsansatz, der von unterschiedlichen Herstellern von Prozessoren zum Teil auch unterschiedlich umgesetzt und implementiert wird. Grundsätzlich gibt es zwei Ansätze:

- Auf der einen Seite kann ein Hersteller «Confidential Computing» so umsetzen und implementieren, dass einzelne Anwendungen oder sicherheitsrelevante Teile davon (z. B. in einem Container) in einer TEE betrieben werden, während andere Anwendungen und Teile ausserhalb der TEE ausgeführt werden. Die bekanntesten Beispiele für diesen Ansatz sind die Software Guard Extensions (SGX) von Intel [1] und TrustZone von ARM.
- Auf der anderen Seite kann ein Hersteller «Confidential Computing» auch so umsetzen und implementieren, dass ganze VMs (mit allen darin ausgeführten Anwendungen) in einer TEE betrieben werden. Hier sind die bekanntesten Beispiele TDX von Intel [2], Secure Encrypted Virtualization (SEV[6]) von AMD, sowie die Umsetzung von «Confidential Computing» auf spezialisierten Prozessoren von NVIDIA.[7]

Beide Ansätze haben Vor- und Nachteile. Während der erste Ansatz selektiv betrieben werden kann, indem nur kritische Anwendungen oder Teile davon in einer TEE ausgeführt werden müssen, ist der zweite Ansatz pauschal, weil hier ganze VMs in einer TEE ausgeführt werden. Damit ist dieser Ansatz im praktischen Einsatz einfacher, weil Anwendungen dann nicht mehr im Hinblick auf den Betrieb in einer TEE umgeschrieben, sondern nur verschoben werden müssen.

Auf der Basis der heute verfügbaren Prozessoren, die «Confidential Computing» auf die eine oder andere Art umsetzen, bieten auch international auftretende Hyperscaler verschiedene Dienste im Bereich von «Confidential Computing» an. Die folgenden Ausführungen sind nur als Beispiele zu verstehen, und natürlich bieten auch andere Hyperscaler und Cloud-Anbieter entsprechende Dienste an, wie z. B. Alibaba mit «Enclave VM». Die Marktsituation ist in

[6] Gemäss https://www.amd.com/en/developer/sev.html gibt es SEV in verschiedenen Ausprägungen, wie z. B. SEV-ES (SEV Encrypted State) und SEV-SNP (SEV Secure Nested Paging).

[7] https://www.nvidia.com/en-us/data-center/solutions/confidential-computing/

diesem Bereich sehr dynamisch und aufgrund unterschiedlicher Terminologien auch unübersichtlich.

- Microsoft ist breit aufgestellt und bietet in seiner Azure-Cloud viele auf die spezifischen Vor- und Nachteile der verschiedenen Prozessoren ausgerichtete «Confidential Computing»-Dienste an.[8] Insbesondere gibt es auch einen entsprechenden Dienst für den Azure Kubernetes Service (AKS).
- Google verfolgt auch eine diversifizierte Dienststrategie, wobei dem Angebot von «Confidential VMs» eine Schlüsselrolle zukommt. Ähnlich wie Microsoft mit AKS bietet Google im Rahmen seiner Google Kubernetes Engine (GKE) auch «Confidential GKE Nodes» an und unterstützt für hoch performante KI/ML-Anwendungen auch «Confidential Computing» auf den oben erwähnten spezialisierten Prozessoren von NVIDIA.
- Amazon Web Services (AWS) hat auf der Basis ihres selbst entwickelten Nitro Systems, das für die Virtualisierung und den automatisierten Betrieb von Elastic Compute Cloud (EC2) Instanzen genutzt wird, Nitro Enclaves konzipiert, mit denen im Sinne von «Confidential Computing» sichere Enklaven umgesetzt werden können. In einer Nitro Enklave können sicherheitskritische Teile einer EC2-Instanz (wie z. B. ein Schlüsselverwaltungssystem) enkapsuliert und betrieben werden.

Darüber hinaus gibt es im Bereich von «Confidential Computing» auch eine zunehmend grosse Zahl von spezialisierten Anbietern und Dienstleistern, die bei konkreten Fragestellungen und Projekten weiterhelfen können, wie z. B. Decentriq[9] und CYSEC[10] in der Schweiz oder enclaive[11] in Deutschland.

5 Potential und Zukunftsaussichten

Mit Hilfe von «Confidential Computing» und entsprechenden TEEs kann die Verarbeitung von Daten sicherer und im Umfeld von Cloud Computing auch weniger abhängig von der Vertrauenswürdigkeit des oder der Cloud-Anbieter gestaltet werden. Insofern ist der Ansatz zur Lösung des «Secure Remote Computation»-Problem interessant und stellt einen wichtigen Beitrag zur Verbesserung der digitalen Souveränität in einer sich zunehmend auf Cloud Computing ausrichtenden Welt dar. Man kann davon ausgehen, dass (neben Intel SDX und TDX, ARM TrustZone, AMD SEV und NVIDIA) auch noch andere Implementierungen auf dem Markt erscheinen und in den Rechenzentren der Hyperscaler und anderen Cloud-Anbietern verbaut werden. Natürlich gilt das primär für Cloud-Angebote im Sinne von Infrastructure-as-a-Service (IaaS) und Platform-as-aService (PaaS), bzw. etwas weniger für Software-as-a-Service (SaaS). Wenn die Angebote sinnvoll genutzt werden, können sie insbesondere auch Anwendungen abzusichern helfen, bei denen Daten gegenüber den Cloud-Anbietern geschützt werden müssen, wie z. B. bei der Verwaltung kryptografischer Schlüssel oder beim Abgleich von datenschutzrelevanten Daten (ein prominentes Beispiel stellt hier der Abgleich von persönlichen Kontaktdaten im End-zu-End verschlüsselnden Messenger Signal dar).

Trotz seiner Vorteile stellt «Confidential Computing» kein Allheilmittel für die IT-Sicherheit (bei IaaS und PaaS) dar. Auch wenn damit die Abhängigkeit von der Vertrauenswürdigkeit der Cloud-Anbieter sinkt, kommt eine neue Abhängigkeit hinzu: Die Vertrauenswürdigkeit der

[8] Eine entsprechende Übersicht ist z. B. unter https://learn.microsoft.com/en-us/azure/confidential-computing/overview-azure-products verfügbar.

[9] https://www.decentriq.com

[10] https://www.cysec.com

[11] https://www.enclaive.io

Lieferanten der TEEs muss sichergestellt sein. Ein Cloud-Anbieter könnte mit einem Lieferanten kooperieren, um gemeinsam eine Datenverarbeitung in einer TEE zu kompromittieren. Allerdings ist eine solche Kooperation in der Realität wohl nur schwer zu realisieren und noch schwerer geheim zu halten. Letztlich können aber auch Implementierungen von «Confidential Computing» und entsprechenden TEEs angegriffen werden, wobei die relevantesten Angriffe (insbesondere zeitbasierte) Seitenkanäle ausnutzen.[12] Obwohl viele dieser Angriffe rein theoretisch sind und «nur» potentielle Möglichkeiten aufzeigen, wird sich dennoch ein für die IT-Sicherheit übliches Katz-und-Maus-Spiel einstellen, d. h. man wird auch in Zukunft bekannt gewordene Angriffe in Bezug auf Voraussetzungen und Rahmenbedingungen anschauen müssen, um die Auswirkungen auf real existierende Cloud-Lösungen und -Angebote sinnvoll abschätzen zu können. Wie oft im Bereich der IT-Sicherheit sind solche Abschätzungen aufwändig und nur mit einer hohen Fachkompetenz überhaupt möglich. Aus sicherheitstechnischer Sicht spricht aber sicherlich nichts gegen den Einsatz von «Confidential Computing» und entsprechenden TEEs, d. h. entsprechende Lösungen und Angebote sollten für sicherheitskritische Anwendungen so weit als möglich und wirtschaftlich vertretbar genutzt werden.

Abkürzungen

GKE	Google Kubernetes Engine
IaaS	Infrastructure-as-a-Service
IT	Informationstechnologie
KI	Künstliche Intelligenz
ML	Maschinelles Lernen
PaaS	Platform-as-a-Service
SaaS	Software-as-a-Service
SEV	Secure Encrypted Virtualization
SEV-ES	SEV Encrypted State
SEV-SNP	SEV Secure Nested Paging
SGX	Software Guard Extensions
TCB	Trusted Computing Base
TCG	Trusted Computing Group
TEE	Trusted Execution Environment
TPM	Trusted Platform Module
VM	Virtuelle Maschine

Referenzen

[1] V. Costan und S. Devadas, *Intel SGX Explained*, IACR Cryptology ePrint Archive, Vol. 2016, No. 086, https://eprint.iacr.org/2016/086.pdf

[2] Intel, *Intel Trust Domain Extensions*, White Paper, aktualisiert im Februar 2023

[3] A. Muñoz, R. Ríos, R. Román und J. López, *A Survey on the (In)security of Trusted Execution Environments*, Computers & Security, Vol. 129, June 2023, 103180, https://www.sciencedirect.com/science/article/pii/S0167404823000901

[4] A. Nilsson, P. Nikbakht Bideh und J. Brorsson, *A Survey of Published Attacks on Intel SGX*, arXiv: 2006.13598, aktualisiert im Juni 2020, https://arxiv.org/abs/2006.13598

[12] In [3] sind viele bekannte Angriffe gegen verschiedene TEEs zusammengestellt (und in [4] ausschliesslich Angriffe gegen Intel SGX).