



(10) **DE 11 2020 003 666 T5** 2022.06.15

(12) **Veröffentlichung**

der internationalen Anmeldung mit der  
(87) Veröffentlichungs-Nr.: **WO 2021/048708**  
in der deutschen Übersetzung (Art. III § 8 Abs. 2  
IntPatÜbkG)  
(21) Deutsches Aktenzeichen: **11 2020 003 666.9**  
(86) PCT-Aktenzeichen: **PCT/IB2020/058194**  
(86) PCT-Anmeldetag: **03.09.2020**  
(87) PCT-Veröffentlichungstag: **18.03.2021**  
(43) Veröffentlichungstag der PCT Anmeldung  
in deutscher Übersetzung: **15.06.2022**

(51) Int Cl.: **G06F 21/60** (2013.01)  
**G06F 21/78** (2013.01)

(30) Unionspriorität:  
**62/900,045**      **13.09.2019**      **US**  
**16/808,174**      **03.03.2020**      **US**  
(71) Anmelder:  
**International Business Machines Corporation,**  
**Armonk, NY, US**

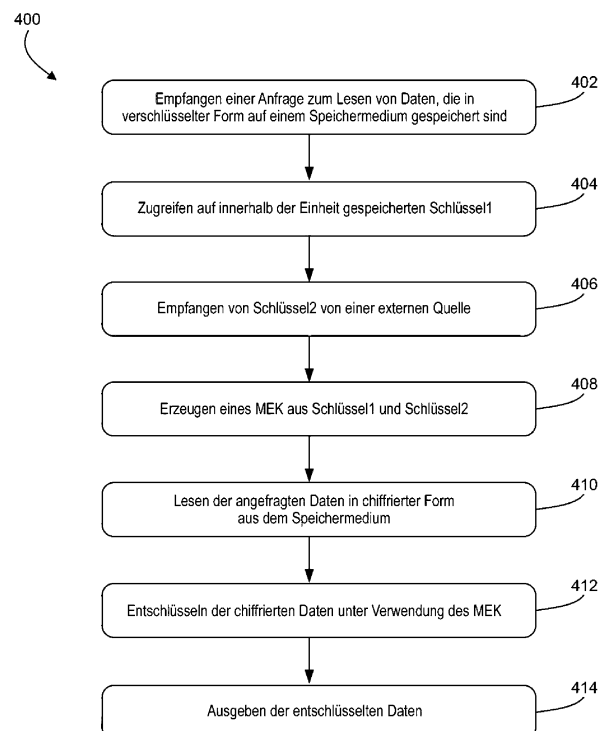
(74) Vertreter:  
**Richardt Patentanwälte PartG mbB, 65185**  
**Wiesbaden, DE**  
(72) Erfinder:  
**Jaquette, Glen, Tucson, AZ, US**

Prüfungsantrag gemäß § 44 PatG ist gestellt.

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.**

(54) Bezeichnung: **KRYPTO-LÖSCHUNG DURCH INTERNE UND/ODER EXTERNE MASSNAHMEN**

(57) Zusammenfassung: Durch eine Einheit implementierte Methodik zur Ermöglichung und/oder Durchführung von Krypto-Löschung durch interne und externe Maßnahmen. Gemäß einem veranschaulichenden Aspekt wird eine Anfrage zum Lesen von Daten an einer Einheit empfangen, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Daten auf dem Speichermedium in verschlüsselter Form gespeichert sind. Gemäß einem Ansatz wird auf einen innerhalb der Einheit gespeicherten ersten Schlüssel zugegriffen. Gemäß einem anderen Ansatz wird ein auf und/oder mit dem Speichermedium gespeicherter erster Schlüssel abgerufen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die verschlüsselte Form der Daten wird aus dem Speichermedium gelesen. Die verschlüsselte Form der Daten wird unter Verwendung des Medienchiffrierschlüssels entschlüsselt. Die entschlüsselten Daten werden ausgegeben. Zudem wird eine Methodik zum Schreiben verschlüsselter Daten vorgestellt.



**Beschreibung**

## TECHNISCHES GEBIET

**[0001]** Die vorliegende Erfindung betrifft Krypto-Löschung und insbesondere Techniken und Systeme zum Ermöglichen von Krypto-Löschung durch interne und/oder externe Maßnahmen.

## HINTERGRUND

**[0002]** Die Bezeichnung „Krypto-Löschung“ bezieht sich allgemein darauf, einen Zugriff auf den zum Entschlüsseln von Daten benötigten Chiffrierschlüssel auf irgendeine Weise zu sperren. Dies kann z.B. durch Löschen aller Kopien des Chiffrierschlüssels, Entfernen eines Teils des Chiffrierschlüssels, Sperren eines Zugriffs auf einen Unterschlüssel, der zum Erzeugen oder Entsperren des Chiffrierschlüssels erforderlich ist, usw. erfolgen. Durch dauerhaftes Sperren des Chiffrierschlüssels können die mit diesem Chiffrierschlüssel verschlüsselten Daten nicht entschlüsselt werden; die verschlüsselten Daten werden effektiv unentschlüsselbar gemacht.

**[0003]** Es wurde festgestellt, dass die heutigen Rechenzentren ein großes Problem damit haben, dass sie nicht in der Lage sind, alle ausgefallenen oder stillgelegten selbstverschlüsselnden Laufwerke (Self-Encrypting Drives, SEDs) einer endgültigen Krypto-Löschung zu unterziehen. So ist es beispielsweise möglich, dass ein SED so ausfällt, dass es nicht mehr mit dem System kommunizieren kann, in dem es sich befindet. In einem solchen ausgefallenen Zustand kann das SED weder einen Krypto-Löschbefehl empfangen noch auf einen solchen Befehl mit einem Status antworten, der angibt, dass der Krypto-Löschbefehl erfolgreich ausgeführt wurde. Für ein Rechenzentrum, das nicht riskieren will, dass Daten aus dem Halbleiterspeicher (SSD, solid-state drive) oder dem Festplattenlaufwerk (HDD, hard disk drive) des ausgefallenen SED, die keiner Krypto-Löschung unterzogen werden konnten, forensisch wiederhergestellt werden können, ist dies ein Problem. Heute greifen diese Rechenzentren in der Regel auf die physische Zerstörung dieser Laufwerke zurück, um eine forensische Wiederherstellung zu verhindern. Der Benutzer kann diese Laufwerke nicht mit gutem Gewissen an den Hersteller zurückgeben oder wiederverwenden, da er befürchten muss, Daten preiszugeben.

**[0004]** Dabei ist zu beachten, dass der Medienchiffrierschlüssel (MEK, Media Encryption Key), der zur Ver- und Entschlüsselung von Daten auf einem SED verwendet wird, in modernen SEDs nicht im Klartext gespeichert wird. Der MEK ist stattdessen kryptografisch verpackt, z.B. selbst verschlüsselt oder anderweitig verschleiert. Es gibt Überlegungen, dass dies auf lange Sicht nicht stark genug sein könnte und

dass die Schlüsselverpackungstechnik in absehbarer Zeit gebrochen werden könnte (z.B. durch Quantencomputer).

**[0005]** Unabhängig vom Ausmaß des Problems oder von der Frage, ob die berechtigte Sorge besteht, dass der verpackte Schlüssel entschlüsselt oder entziffert werden könnte, kann es sehr wohl sein, dass es Einrichtungen gibt, die sich für nichtflüchtige Speicherung und Krypto-Löschung des MEK nicht auf das Laufwerk verlassen oder ihm voll vertrauen wollen. Ein solcher Benutzer könnte es vorziehen, den MEK nach jedem Einschaltvorgang an das Laufwerk zu senden, anstatt ihn auf dem SED zu speichern, um sicherzustellen, dass der MEK zerstört und der gesamte mit ihm erstellte chiffrierte Text einer Krypto-Löschung unterzogen werden kann. Damit hat der Nutzer die Kontrolle über den MEK und muss sich nicht darum sorgen, dass das Laufwerk so ausfällt, dass es keine Krypto-Löschung mehr durchführen kann. Da der MEK nur außerhalb des Laufwerks nichtflüchtig gespeichert wird, kann er vom Benutzer zerstört werden, unabhängig davon, wie das SED ausgefallen ist. Ein Bruch des Schlüsselspeichers des Nutzers, in dem der MEK aufbewahrt wird, oder ein Abhören von Datenaustausch dieses MEK mit dem SED (und ein Brechen diesbezüglicher Sicherheitsvorkehrungen) macht den MEK jedoch anfällig für Erfassung oder Abruf.

**[0006]** Eine Möglichkeit zum Bereitstellen der Fähigkeit zur externen Krypto-Löschung eines SED ist die Verwendung des von LTO-4 (der ersten verschlüsselungsfähigen LTO-Bandlaufwerksgeneration) verwendeten Modells für direkte Schlüsselübergabe. Es ist zu beachten, dass die Art und Weise, wie die Schlüssel außerhalb der verschlüsselungsfähigen Einheit gehandhabt werden, darüber entscheidet, ob diese einer Krypto-Löschung unterzogen werden können oder nicht. Wenn die Schlüsselverwaltung einer Einrichtung ordnungsgemäß durchgeführt wird, sollte es jedoch möglich sein, alle Kopien des MEK, die mit einem ausgefallenen SED-Laufwerk verbunden sind, einer Krypto-Löschung zu unterziehen. In der Regel werden jedoch Kopien des Schlüssels angefertigt und an verteilten Orten aufbewahrt, um sicherzustellen, dass der Schlüssel zum Entschlüsseln von Daten verfügbar bleibt, denn wenn keine Kopie des Schlüssels mehr vorhanden ist, können die mit diesem Schlüssel verschlüsselten Daten, d.h. der chiffrierte Text, nicht mehr entschlüsselt werden, so dass sie effektiv unzugänglich oder „krypto-gelöscht“ sind. Eine sichere Schlüsselverwaltung wird also erheblich schwieriger, wenn immer mehr Kopien des Schlüssels angefertigt werden.

**[0007]** Nach LTO-4 hat sich die SED-Technologie auf Konformität mit den Spezifikationen der Trusted Computing Group (TCG) standardisiert, wie z.B. die

Security Subsystem Classes (SSC) der TCG Storage, einschließlich (zunächst) Enterprise und (in jüngerer Zeit) Opal. Beide SSCs unterstützen Krypto-Löschung auf unterschiedlichste Weise. So gibt es zum Beispiel im Fall von Opal mindestens vier verschiedene Möglichkeiten, eine Krypto-Löschung auszulösen. Keines der beiden TCG-SSC unterstützt jedoch Krypto-Löschung durch eine externe, vom SED selbst unabhängige Stelle, d.h. wenn das SED eine vom SSC angegebene Methode (Befehl) nicht berücksichtigen kann. Der MEK wird immer im SED gespeichert, normalerweise in kryptografisch verpackter Form. Wenn jemand auf irgendeine Weise herausfinden kann, wie man den verpackten MEK aufricht, kann diese Person den gesamten zugehörigen chiffrierten Text wiederherstellen, der in einem ausgefallenen SED gespeichert ist.

## KURZDARSTELLUNG

**[0008]** Gemäß einem Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Lesen von Daten, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind. Es wird auf einen ersten innerhalb der Einheit gespeicherten Schlüssel zugegriffen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die verschlüsselte Form der Daten wird aus dem Speichermedium gelesen. Die verschlüsselte Form der Daten wird unter Verwendung des Medienchiffrierschlüssels entschlüsselt. Die entschlüsselten Daten werden ausgegeben. Mit diesem Verfahren können lokal verschlüsselte Daten durch sowohl interne als auch externe Maßnahmen einer Krypto-Löschung unterzogen werden.

**[0009]** Gemäß einem weiteren Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Lesen von Daten, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind. Ein auf und/oder mit dem Speichermedium gespeicherter erster Schlüssel wird abgerufen. Ein zweiter Schlüssel wird von einer externen Quelle abgerufen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die verschlüsselte Form der Daten wird aus dem Speichermedium gelesen. Die verschlüsselte Form der Daten wird unter Verwendung des Medienchiffrierschlüssels entschlüsselt. Die entschlüsselten Daten werden ausgegeben. Mit diesem Verfahren können lokal ver-

schlüsselte Daten durch sowohl interne als auch externe Maßnahmen einer Krypto-Löschung unterzogen werden. Bei der Verwendung von Wechselspeichermedien wird der erste Schlüssel mit dem Medium transportiert, aber die Daten bleiben sicher, bis der zweite Schlüssel bereitgestellt wird.

**[0010]** Gemäß einem weiteren Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Schreiben von Daten auf das Speichermedium in verschlüsselter Form. Es wird auf einen ersten innerhalb der Einheit gespeicherten Schlüssel zugegriffen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die angefragten Daten werden unter Verwendung des Medienchiffrierschlüssels in verschlüsselter Form auf das Speichermedium geschrieben. Mit diesem Verfahren können die geschriebenen verschlüsselten Daten durch sowohl interne als auch externe Maßnahmen einer Krypto-Löschung unterzogen werden.

**[0011]** Gemäß einem weiteren Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Schreiben von Daten auf das Speichermedium in verschlüsselter Form. Ein auf und/oder mit dem Speichermedium gespeicherter erster Schlüssel wird abgerufen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die angefragten Daten werden unter Verwendung des Medienchiffrierschlüssels in verschlüsselter Form auf das Speichermedium geschrieben. Mit diesem Verfahren können die geschriebenen verschlüsselten Daten durch sowohl interne als auch externe Maßnahmen einer Krypto-Löschung unterzogen werden.

**[0012]** In verschiedenen Ansätzen ist die Einheit, die eines der jeweiligen Verfahren durchführt, bevorzugt so konfiguriert, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet. Da der erste Schlüssel die Einheit nicht verlassen kann, werden durch eine Löschung des ersten Schlüssels aus der Einheit die verschlüsselten Daten auf dem Gerät effektiv einer Krypto-Löschung unterzogen.

**[0013]** Verschiedene vorliegend beschriebene Ansätze sind auf viele Arten von Speichermedien anwendbar, darunter nichtflüchtiger Speicher und Magnetband.

**[0014]** Weitere Aspekte und Ansätze der vorliegenden Erfindung werden aus der nachfolgenden ausführlichen Beschreibung ersichtlich, welche in Zusammenschau mit den Zeichnungen beispielhaft die Prinzipien der Erfindung veranschaulicht.

#### Figurenliste

**Fig. 1** ist ein Schaubild einer Netzwerkarchitektur gemäß einem Aspekt der vorliegenden Erfindung.

**Fig. 2** ist ein Schaubild einer repräsentativen Hardwareumgebung, die gemäß einem Aspekt der vorliegenden Erfindung zu den Servern und/oder Clients aus **Fig. 1** gehören kann.

**Fig. 3** ist ein Schaubild eines mehrstufigen Datenspeichersystems gemäß einem Aspekt der vorliegenden Erfindung.

**Fig. 4** ist ein Ablaufplan eines Verfahrens gemäß einem Aspekt der vorliegenden Erfindung.

**Fig. 5** ist ein Ablaufplan eines Verfahrens gemäß einem Aspekt der vorliegenden Erfindung.

**Fig. 6** ist ein Ablaufplan eines Verfahrens gemäß einem Aspekt der vorliegenden Erfindung.

**Fig. 7** ist ein Ablaufplan eines Verfahrens gemäß einem Aspekt der vorliegenden Erfindung.

**Fig. 8** ist ein Schaubild, das den derzeitigen Stand der Technik zum Stilllegen und Sichern von Standardspeicherprodukten zeigt, die ein SED verwenden.

**Fig. 9** ist ein Schaubild, das veranschaulichende Prozeduren zum Stilllegen und Sichern von Speicherprodukten, die ein SED verwenden, gemäß einem Aspekt der vorliegenden Erfindung zeigt.

**Fig. 10** ist ein Schaubild, das veranschaulichende Prozeduren zum Stilllegen und Sichern von Speicherprodukten, die ein SED verwenden, gemäß einem Aspekt der vorliegenden Erfindung zeigt.

#### AUSFÜHRLICHE BESCHREIBUNG

**[0015]** Die nachfolgende Beschreibung erfolgt zum Zwecke der Veranschaulichung der Grundgedanken der vorliegenden Erfindung und soll nicht die vorliegend beanspruchten erfindungsgemäßen Konzepte einschränken. Ferner können vorliegend beschriebene jeweilige Merkmale in Kombination mit anderen beschriebenen Merkmalen in jeder der verschiedenen möglichen Kombinationen und Permutationen verwendet werden.

**[0016]** Soweit nicht vorliegend ausdrücklich anders angegeben, sind alle Bezeichnungen so breit wie möglich auszulegen, einschließlich aus der Spezifikation implizierter Bedeutungen sowie Bedeutungen, die von einem Fachmann verstanden werden und/oder die in Wörterbüchern, Abhandlungen usw. definiert sind.

**[0017]** Zudem ist zu beachten, dass die in der Spezifikation und den sich anschließenden Ansprüchen verwendeten Singularformen „ein/e/“ und „der/die/-das“ die Möglichkeit mehrerer Bezugsobjekte mit einschließen, sofern nicht anders angegeben. Ferner ist zu beachten, dass die Ausdrücke „weist auf“ und/oder „aufweisend“, soweit in dieser Spezifikation verwendet, das Vorliegen genannter Merkmale, Zahlen, Schritte, Vorgänge, Elemente und/oder Komponenten angibt, jedoch nicht das Vorliegen oder Hinzufügen eines oder mehrerer weiterer Merkmale, Zahlen, Schritte, Vorgänge, Elemente, Komponenten und/oder Gruppen davon ausschließt.

**[0018]** Die vorliegende Beschreibung offenbart mehrere bevorzugte Ansätze von Systemen, Verfahrenen und Computerprogrammprodukten, um zu ermöglichen, lokal verschlüsselte Daten durch sowohl interne als auch externe Maßnahmen einer Krypto-Löschung zu unterziehen. Gemäß einem Ansatz wird ein Medienchiffrierschlüssel (Media Encryption Key, MEK), der zur Ver-/Entschlüsselung von Daten verwendet wird, in zwei Teile aufgeteilt - ein Schlüsselanteil (Schlüssel1) wird zusammen mit den Daten gespeichert, und der andere Schlüsselanteil (Schlüssel2) wird extern verwaltet. Gemäß einem Ansatz wird der Schlüssel1, der in nichtflüchtiger Form vorliegt, nur zusammen mit den Daten gespeichert, wahlweise in verpackter Form (z.B. verschlüsselt oder anderweitig geschützt). Schlüssel2, der in nichtflüchtiger Form gespeichert sein kann, wird nur außerhalb der Einheit/des Produkts gespeichert, auf der/dem sich die Daten befinden, so dass Schlüssel2 der Einheit, die die Daten liest, zugeführt werden muss, um Verschlüsselung und/oder Entschlüsselung zu ermöglichen. Der MEK kann nur berechnet werden, wenn die Einheit sowohl über Schlüssel1 als auch über Schlüssel2 verfügt.

**[0019]** Gemäß einem allgemeinen Ansatz werden die Daten auf einem selbstverschlüsselnden Laufwerk (Self-Encrypting Drive, SED) gespeichert. Schlüssel1 wird intern im SED verwaltet und kann im SED in Rohform oder in verpackter Form gespeichert sein. Schlüssel2 wird dem SED mindestens einmal nach einem Einschaltvorgang und/oder einem anderen Kaltstart bereitgestellt, damit der MEK intern im SED berechnet werden kann.

**[0020]** Gemäß einem anderen allgemeinen Ansatz werden die Daten auf einem Speichermedium gespeichert, beispielsweise auf einem Magnetband,

einem nichtflüchtigen Direktzugriffsspeicher (NVRAM, beispielsweise Halbleiterspeicher, Festplatte usw.) usw. Schlüssell wird auf und/oder mit dem Speichermedium gespeichert und kann in verpackter Form gespeichert sein. Schlüssel2 wird dem Laufwerk, Computer usw. zur Verfügung gestellt, damit das Laufwerk, der Computer usw. den MEK berechnen kann.

**[0021]** Gemäß einem beispielhaften Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Lesen von Daten, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind. Es wird auf einen ersten innerhalb der Einheit gespeicherten Schlüssel zugegriffen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die verschlüsselte Form der Daten wird aus dem Speichermedium gelesen. Die verschlüsselte Form der Daten wird unter Verwendung des Medienchiffrierschlüssels entschlüsselt. Die entschlüsselten Daten werden ausgegeben.

**[0022]** Gemäß einem weiteren beispielhaften Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Lesen von Daten, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind. Ein auf und/oder mit dem Speichermedium gespeicherter erster Schlüssel wird abgerufen. Ein zweiter Schlüssel wird von einer externen Quelle abgerufen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die verschlüsselte Form der Daten wird aus dem Speichermedium gelesen. Die verschlüsselte Form der Daten wird unter Verwendung des Medienchiffrierschlüssels entschlüsselt. Die entschlüsselten Daten werden ausgegeben.

**[0023]** Gemäß einem weiteren beispielhaften Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Schreiben von Daten auf das Speichermedium in verschlüsselter Form. Es wird auf einen ersten innerhalb der Einheit gespeicherten Schlüssel zugegriffen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die angefragten Daten werden unter Verwendung

des Medienchiffrierschlüssels in verschlüsselter Form auf das Speichermedium geschrieben.

**[0024]** Gemäß einem weiteren beispielhaften Aspekt der vorliegenden Erfindung umfasst ein durch eine Einheit implementiertes Verfahren an einer zum Durchführen von Datenoperationen an einem Speichermedium konfigurierten Einheit erfolgreiches Empfangen einer Anfrage zum Schreiben von Daten auf das Speichermedium in verschlüsselter Form. Ein auf und/oder mit dem Speichermedium gespeicherter erster Schlüssel wird abgerufen. Ein zweiter Schlüssel wird von einer externen Quelle empfangen. Unter Verwendung des ersten und des zweiten Schlüssels wird ein Medienchiffrierschlüssel erzeugt. Die angefragten Daten werden unter Verwendung des Medienchiffrierschlüssels in verschlüsselter Form auf das Speichermedium geschrieben.

Veranschaulichende Datenverarbeitungsumgebung

**[0025]** Fig. 1 veranschaulicht eine Architektur 100 gemäß einem Ansatz. Wie in Fig. 1 gezeigt ist, sind mehrere entfernte Netzwerke 102 vorgesehen, darunter ein erstes entferntes Netzwerk 104 und ein zweites entferntes Netzwerk 106. Ein Gateway 101 kann zwischen die entfernten Netzwerke 102 und ein nahegelegenes Netzwerk 108 geschaltet sein. Im Zusammenhang mit der vorliegenden Architektur 100 können die Netzwerke 104, 106 jede beliebige Form annehmen, darunter, ohne jedoch hierauf eingeschränkt zu sein, ein lokales Netz (LAN), ein Weitverkehrsnetz (WAN) wie das Internet, ein öffentliches Telefonnetz (PSTN, public switched telephone network), ein internes Telefonnetz usw.

**[0026]** Im Einsatz dient das Gateway 101 als Zugangspunkt von den entfernten Netzwerken 102 in das nahegelegene Netzwerk 108. Somit kann das Gateway 101 als Router fungieren, der in der Lage ist, ein bestimmtes Datenpaket, das am Gateway 101 ankommt, weiterzuleiten, und als Switch, der den tatsächlichen Pfad in und aus dem Gateway 101 für ein bestimmtes Paket bereitstellt.

**[0027]** Darüber ist mindestens ein Datenserver 114 umfasst, der mit dem nahegelegenen Netzwerk 108 verbunden und von den entfernten Netzwerken 102 aus über das Gateway 101 zugänglich ist. Es sollte beachtet werden, dass der/die Datenserver 114 jede Art von Datenverarbeitungseinheit/Groupware enthalten können. Mit jedem Datenserver 114 ist eine Mehrzahl von Benutzereinheiten 116 verbunden. Die Benutzereinheiten 116 können auch direkt über eines der Netzwerke 104, 106, 108 verbunden sein. Zu solchen Benutzereinheiten 116 können ein Desktop-Computer, ein Laptop-Computer, ein tragbarer Computer, ein Drucker oder eine beliebige andere Art von Logik zählen. Es sei darauf hingewiesen, dass eine Benutzereinheit 111 gemäß einem

Ansatz auch direkt mit einem der Netzwerke verbunden sein kann.

**[0028]** Ein Peripheriegerät 120 oder eine Reihe von Peripheriegeräten 120, z.B. Faxgeräte, Drucker, vernetzte und/oder lokale Datenspeichereinheiten oder-systeme usw., können mit einem oder mehreren der Netzwerke 104, 106, 108 verbunden sein. Es ist zu beachten, dass Datenbanken und/oder zusätzliche Komponenten mit jeder Art von Netzwerkelement, das mit den Netzwerken 104, 106, 108 verbunden ist, verwendet oder in diese integriert werden können. Im Zusammenhang mit der vorliegenden Beschreibung kann sich ein Netzwerkelement auf jede Komponente eines Netzwerkes beziehen.

**[0029]** Gemäß einigen Ansätzen können die vorliegend beschriebenen Verfahren und Systeme mit und/oder auf virtuellen Systemen und/oder Systemen implementiert werden, die ein oder mehrere andere Systeme emulieren, beispielsweise ein UNIX®-System, das eine IBM® z/OS®-Umgebung emuliert, ein UNIX®-System, das virtuell eine Microsoft® Windows®-Umgebung hostet, ein Microsoft® Windows®-System, das eine IBM® z/OS®-Umgebung emuliert, usw. Diese Virtualisierung und/oder Emulation kann gemäß einigen Ansätzen durch Verwendung von VMware®-Software verbessert werden.

**[0030]** Gemäß weiteren Ansätzen können ein oder mehrere Netzwerke 104, 106, 108 einen Cluster aus Systemen darstellen, die gemeinhin als „Cloud“ bezeichnet werden. Beim Cloud Computing werden gemeinsam genutzte Ressourcen wie Verarbeitungsleistung, Peripheriegeräte, Software, Daten, Server usw. jedem System in der Cloud auf Abruf zur Verfügung gestellt, wodurch Zugriff und Verteilung von Diensten über viele Datenverarbeitungssysteme hinweg ermöglicht werden. Beim Cloud Computing besteht in der Regel eine Internetverbindung zwischen den in der Cloud betriebenen Systemen, es können aber auch andere Methoden zum Verbinden der Systeme verwendet werden.

**[0031]** Fig. 2 zeigt eine repräsentative Hardwareumgebung, die zu einer Benutzereinheit 116 und/oder einem Server 114 aus Fig. 1 gehört, gemäß einem Ansatz. Die Figur veranschaulicht eine typische Hardwarekonfiguration einer Workstation mit einer Zentralverarbeitungseinheit 210, wie beispielsweise einem Mikroprozessor, und einer Reihe anderer Einheiten, die über einen Systembus 212 miteinander verbunden sind.

**[0032]** Die in Fig. 2 gezeigte Workstation umfasst einen Direktzugriffsspeicher (RAM, Random Access Memory) 214, einen Nur-Lese-Speicher (ROM, Read Only Memory) 216, einen Eingabe/Ausgabe- (E/A-) Adapter 218 zum Verbinden von Peripherieeinheiten

wie Plattenspeichereinheiten 220 mit dem Bus 212, einen Benutzerschnittstellenadapter 222 zum Verbinden einer Tastatur 224, einer Maus 226, eines Lautsprechers 228, eines Mikrofons 232 und/oder anderer Benutzerschnittstelleneinheiten wie ein berührungsempfindlicher Bildschirm und eine Digitalkamera (nicht gezeigt) mit dem Bus 212, einen Datenübertragungsadapter 234 zum Verbinden der Workstation mit einem Datenübertragungsnetzwerk 235 (z.B. einem Datenverarbeitungsnetzwerk) und einen Anzeigeadapter 236 zum Verbinden des Buses 212 mit einer Anzeigeeinheit 238.

**[0033]** Auf der Workstation kann sich ein Betriebssystem wie das Microsoft Windows®-Betriebssystem (OS, operating system), ein macOS®, ein UNIX® OS usw. befinden. Es versteht sich, dass ein bevorzugter Ansatz auch auf anderen Plattformen und Betriebssystemen als den genannten implementiert werden kann. Ein bevorzugter Ansatz kann unter Verwendung von eXtensible Markup Language (XML), C und/oder C++ oder anderen Programmiersprachen in Verbindung mit einer objektorientierten Programmiermethodik geschrieben sein. Objektorientierte Programmierung (OOP), die zunehmend zur Entwicklung komplexer Anwendungen eingesetzt wird, kann ebenfalls verwendet werden.

**[0034]** In Fig. 3 ist ein Speichersystem 300 gemäß einem Ansatz gezeigt. Es ist zu beachten, dass einige der in Fig. 3 gezeigten Elemente gemäß verschiedenen Aspekten als Hardware und/oder Software implementiert werden können. Das Speichersystem 300 kann einen Speichersystemmanager 312 zum Datenaustausch mit einer Mehrzahl von Medien und/oder Laufwerken auf mindestens einer höheren Speicherstufe 302 und mindestens einer niedrigeren Speicherstufe 306 enthalten. Die höhere(n) Speicherstufe(n) 302 kann/können bevorzugt ein oder mehrere Wahlzugriffs- und/oder Direktzugriffsmedien 304 enthalten, wie z.B. Festplatten in Festplattenlaufwerken (HDDs), nichtflüchtigen Speicher (NVM, nonvolatile memory), Halbleiterspeicher in Halbleiterdatenträgern (SSDs, solid state drives), Flashspeicher, SSD-Arrays, Flashspeicher-Arrays usw. und/oder andere vorliegend erwähnte oder im Fachgebiet bekannte. Die untere(n) Speicherstufe (n) 306 enthält/enhalten bevorzugt ein oder mehrere Speichermedien 308 mit geringerer Leistung, darunter Medien mit sequenziellem Zugriff, wie z.B. Magnetband in Bandlaufwerken und/oder optische Medien, langsamer zugreifende HDDs, langsamer zugreifende SSDs usw. und/oder andere vorliegend erwähnte oder im Fachgebiet bekannte Medien. Eine oder mehrere zusätzliche Speicherstufen 316 können eine beliebige Kombination von Speichermedien enthalten, die von einem Entwickler des Systems 300 gewünscht wird. Zudem kann jede der höheren Speicherstufen 302 und/oder der unteren Speicher-

stufen 306 eine beliebige Kombination aus Speichereinheiten und/oder Speichermedien enthalten.

**[0035]** Der Speichersystemmanager 312 kann mit den Laufwerken und/oder Speichermedien 304, 308 auf der (den) höheren Speicherstufe(n) 302 und der (den) niedrigeren Speicherstufen (n) 306 über ein Netzwerk 310, z.B. ein Storage Area Network (SAN), wie in **Fig. 3** gezeigt, oder einem anderen geeigneten Netzwerktyp Daten austauschen. Der Speichersystemmanager 312 kann zudem mit einem oder mehreren Host-Systemen (nicht dargestellt) über eine Host-Schnittstelle 314 Daten austauschen, die gegebenenfalls ein Teil des Speichersystemmanagers 312 sein kann. Der Speichersystemmanager 312 und/oder jede andere Komponente des Speichersystems 300 kann in Hardware und/oder Software implementiert sein und kann einen Prozessor (nicht dargestellt) zur Ausführung von Befehlen eines im Fachgebiet bekannten Typs verwenden, wie z.B. eine Zentralverarbeitungseinheit (CPU), ein frei programmierbares Gate-Array (FPGA), eine anwendungsspezifische integrierte Schaltung (ASIC), usw. Natürlich kann jede beliebige Anordnung eines Speichersystems verwendet werden, wie für einen Fachmann beim Lesen der vorliegenden Beschreibung offensichtlich ist.

**[0036]** Gemäß weiteren Ansätzen kann das Speichersystem 300 eine beliebige Anzahl von Datenspeicherstufen umfassen und innerhalb jeder Speicherstufe die gleichen oder unterschiedliche Speichermedien enthalten. Beispielsweise kann jede Datenspeicherstufe denselben Typ von Speichermedien enthalten, wie HDDs, SSDs, Medien mit sequenziellem Zugriff (Band in Bandlaufwerken, optische Platte in optischen Laufwerken usw.), Direktzugriffsmedien (CD-ROM, DVD-ROM usw.) oder eine beliebige Kombination von Medienspeichertypen. In einer solchen Konfiguration kann eine höhere Speicherstufe 302 mehrheitlich SSD-Speichermedien zur Speicherung von Daten in einer leistungsfähigeren Speicherumgebung enthalten, während die übrigen Speicherstufen, einschließlich der unteren Speicherstufe 306 und der zusätzlichen Speicherstufen 316, eine beliebige Kombination von SSDs, HDDs, Bandlaufwerken usw. zur Speicherung von Daten in einer leistungsschwächeren Speicherumgebung enthalten können. Auf diese Weise können Daten, auf die häufiger zugegriffen wird, Daten mit höherer Priorität, Daten, auf die schneller zugegriffen werden muss, usw. auf der höheren Speicherstufe 302 gespeichert werden, während Daten, die keines dieser Attribute aufweisen, auf den zusätzlichen Speicherstufen 316, einschließlich der unteren Speicherstufe 306, gespeichert werden können. Natürlich kann ein Fachmann beim Lesen der vorliegenden Beschreibungen viele andere Kombinationen von Speichermedientypen entwickeln, um sie

gemäß den vorliegend gezeigten Ansätzen und Aspekten in verschiedene Speicherschemata zu implementieren.

**[0037]** Gemäß einigen Ansätzen kann das Speichersystem (z.B. 300) Logik enthalten, die so konfiguriert ist, dass sie eine Anfrage zum Öffnen eines Datensatzes empfängt, Logik, die so konfiguriert ist, dass sie ermittelt, ob der angefragte Datensatz auf einer niedrigeren Speicherstufe 306 eines gestuften Datenspeichersystems 300 in mehreren zugehörigen Teilen gespeichert ist, Logik, die so konfiguriert ist, dass sie jeden zugehörigen Teil des angefragten Datensatzes auf eine höhere Speicherstufe 302 des gestuften Datenspeichersystems 300 verschiebt, und Logik, die so konfiguriert ist, dass sie den angefragten Datensatz auf der höheren Speicherstufe 302 des gestuften Datenspeichersystems 300 aus den zugehörigen Teilen zusammensetzt.

**[0038]** Natürlich kann diese Logik gemäß verschiedenen Ansätzen als Verfahren auf einer beliebigen Einheit und/oder einem beliebigen System oder als Computerprogrammprodukt implementiert werden.

Erfindungsgemäße Verfahren zum Ermöglichen von Krypto-Löschung

**[0039]** Die nachstehende Beschreibung offenbart mehrere bevorzugte Ansätze von Systemen, Verfahrenen und Computerprogrammprodukten, um zu ermöglichen, lokal verschlüsselte Daten durch sowohl interne als auch externe Maßnahmen einer Krypto-Löschung zu unterziehen. Gemäß einem Ansatz wird ein MEK, der zur Ver-/Entschlüsselung von Daten verwendet wird, in zwei Teile aufgeteilt - ein Schlüsselanteil (Schlüssel1) wird zusammen mit den Daten gespeichert (z.B. im SED, auf einem Band usw.), während der andere Schlüsselanteil (Schlüssel2) extern verwaltet wird. Gemäß einem Ansatz wird der Schlüssel1, der in nichtflüchtiger Form vorliegt, nur zusammen mit den Daten gespeichert, bevorzugt in verpackter Form. Schlüssel2, der in nichtflüchtiger Form vorliegen kann, wird nur außerhalb der Einheit/des Produkts gespeichert, auf der/dem sich die Daten befinden, so dass Schlüssel2 der Einheit, die die Daten liest/schreibt, zugeführt werden muss, um Verschlüsselung und/oder Entschlüsselung zu ermöglichen. Der MEK kann nur berechnet werden, wenn die Einheit sowohl über Schlüssel1 als auch über Schlüssel2 verfügt.

**[0040]** Gemäß einem allgemeinen Aspekt sind die Daten in einem SED gespeichert. Schlüssel1 wird intern im SED verwaltet und kann im SED in Rohform oder bevorzugt in verpackter Form gespeichert sein. Schlüssel2 wird dem SED mindestens einmal nach einem Einschaltvorgang und/oder einem Kaltstart des SED zur Verfügung gestellt, damit der MEK berechnet werden kann.

**[0041]** Gemäß einem weiteren allgemeinen Ansatz sind die Daten auf einem Speichermedium wie einem Magnetband, in einem NVRAM usw. gespeichert. Schlüssell ist auf und/oder mit dem Speichermedium gespeichert und kann in verpackter Form gespeichert sein. Beispielsweise kann Schlüssell auf einem Teil eines Magnetbandes, in einem Bandkassettenspeicher oder beidem gespeichert sein. Auf diese Weise ist Schlüssell zusammen mit dem Magnetband tragbar. In ähnlicher Weise kann bei einer NVRAM-Einheit Schlüssell im NVRAM, in einem separaten Speicher usw. gespeichert sein. Schlüssell2 wird dem Bandlaufwerk, Computer usw. zur Verfügung gestellt, damit das Laufwerk, der Computer usw. den MEK berechnen kann.

#### Krypto-Löschung im SED

**[0042]** Wenngleich ein Großteil der nachfolgenden Beschreibung in Verbindung mit einer beispielhaften Implementierung mit einem SED dargestellt wird, geschieht dies lediglich beispielhaft und einzig, um dem Leser hilfsweise einen Kontext bereitzustellen. Dementsprechend sind die nachstehend vorgestellten Konzepte und Lehren ebenso auf Implementierungen mit Speichermedien wie Magnetbändern, Speicherkarten usw. anwendbar.

**[0043]** In **Fig. 4** ist ein Ablaufplan eines Verfahrens 400 gemäß einem Ansatz gezeigt. Gemäß der vorliegenden Erfindung kann gemäß verschiedenen Ansätzen das Verfahren 400 unter anderem in jeder der in den anderen vorliegend beschriebenen Figuren abgebildeten Umgebungen durchgeführt werden. Im Verfahren 400 können natürlich mehr oder weniger Arbeitsschritte als die konkret in **Fig. 4** beschriebenen enthalten sein, wie ein Fachmann bei Lektüre der vorliegenden Beschreibungen versteht.

**[0044]** Jeder der Schritte des Verfahrens 400 kann durch jede geeignete Komponente der Betriebsumgebung durchgeführt werden. Beispielsweise kann in verschiedenen Ansätzen das Verfahren 400 teilweise oder ganz durch einen Computer, ein Laufwerk oder eine andere Einheit mit einem oder mehreren darin befindlichen Prozessoren durchgeführt werden. Der Prozessor, z.B. Verarbeitungsschaltung(en), Chip(s) und/oder Modul(e), die in Hardware und/oder Software implementiert sind und bevorzugt über mindestens eine Hardware-Komponente verfügen, kann in jeder Einheit genutzt werden, um einen oder mehrere Schritte des Verfahrens 400 durchzuführen. Zu veranschaulichenden Prozessoren zählen, ohne jedoch hierauf eingeschränkt zu sein, eine zentrale Verarbeitungseinheit (CPU), eine anwendungsspezifische integrierte Schaltung (ASIC), eine vor Ort programmierbare Gatter-Anordnung (FPGA) usw., Kombinationen aus diesen oder irgendeine

andere geeignete Datenverarbeitungseinheit, die im Fachgebiet bekannt ist.

**[0045]** Wie in **Fig. 4** gezeigt, kann das Verfahren 400 mit dem Arbeitsschritt 402 beginnen, bei dem eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen wie beispielsweise Lesen und/oder Schreiben von Daten von dem/auf das Speichermedium durchführt, eine Anfrage zum Lesen von Daten empfängt, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind. Die Anfrage kann einfach die Daten anfragen oder zusätzliche Informationen wie den Schlüssell2 enthalten (siehe unten).

**[0046]** In Arbeitsschritt 404 wird z.B. von einem Speicher der Einheit, von einer ASIC usw. aus auf einen in der Einheit gespeicherten ersten Schlüssel (Schlüssell) zugegriffen. Wie vorstehend erwähnt, ist gemäß einigen Ansätzen der erste Schlüssel in Roh-(unverschlüsselter) Form in der Einheit gespeichert. Gemäß anderen Ansätzen ist der erste Schlüssel in verpackter Form in der Einheit gespeichert (z.B. in verschlüsselter Form, passwortgeschützter Form, verschleierte Form usw.). Informationen wie ein weiterer Schlüssel, ein Kennwort usw. zum Entpacken des ersten Schlüssels können empfangen und zum Entpacken des ersten Schlüssels verwendet werden.

**[0047]** Gemäß bevorzugten Aspekten ist die Einheit so konfiguriert, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet. Eine in der Einheit befindliche Steuereinheit kann jedoch auf den ersten Schlüssel innerhalb der Einheit zugreifen.

**[0048]** In Arbeitsschritt 406 wird ein zweiter Schlüssel (Schlüssell2) von einer externen Quelle empfangen. Bei der externen Quelle kann es sich um eine beliebige externe Quelle handeln. Beispielsweise kann es sich bei der externen Quelle um den Anfrager der Daten, einen Schlüsselspeicher, einen Schlüsselserver oder eine Schlüsseleingabe durch einen Benutzer handeln (z.B. über eine Tastatur, durch Einstecken eines Flashspeichers usw.).

**[0049]** In Arbeitsschritt 408 wird unter Verwendung des ersten und des zweiten Schlüssels ein Medienchiffrierschlüssel (MEK) erzeugt. Es kann eine beliebige bekannte Methode verwendet werden, um aus dem ersten und dem zweiten Schlüssel einen MEK zu erstellen, z.B. durch XOR-Verknüpfung des ersten und des zweiten Schlüssels.

**[0050]** In Arbeitsschritt 410 wird die verschlüsselte Form (d.h. der chiffrierte Text) der angefragten Daten aus dem Speichermedium gelesen, z.B. unter Verwendung herkömmlicher Methoden. Das Speichermedium kann beliebiger vorliegend offener Art sein, z.B. Magnetband, Magnetplatte,

NVRAM usw. Dementsprechend kann es sich bei der Einheit um eine beliebige Art von Datenspeichereinheit handeln, z.B. Bandlaufwerk, SSD, HDD, verschlüsselungsfähige USB-Datenträger, NVRAM-Module, usw.

**[0051]** In Arbeitsschritt 412 wird der chiffrierte Text der angefragten Daten unter Verwendung des MEK entschlüsselt, um die Daten in unverschlüsselter (d.h. Klartext-) Form zu erhalten. Die Daten können in chiffrierter Form in einen Puffer kopiert und dann entschlüsselt werden, sie können während des Lesens „on the fly“ entschlüsselt werden (z.B. wenn die Arbeitsschritte 410 und 412 gleichzeitig ausgeführt werden) usw.

**[0052]** In Arbeitsschritt 414 werden die entschlüsselten Daten ausgegeben, z.B. an einen Anfrager der Daten über die Hostschnittstelle usw.

**[0053]** Wenn zweite Daten von der Einheit auf dem Speichermedium gespeichert werden sollen, werden die zweiten Daten empfangen. Zudem kann von einer zweiten externen Quelle ein dritter Schlüssel empfangen werden. Aus dem ersten und dem dritten Schlüssel kann zudem ein zweiter Medienchiffrierschlüssel erzeugt werden. Die Daten werden unter Verwendung des zweiten Medienchiffrierschlüssels verschlüsselt, und die verschlüsselten Daten werden auf das Speichermedium geschrieben. Es ist zu beachten, dass es sich bei der zweiten externen Quelle um dieselbe wie die vorstehend genannte Quelle oder um eine andere Quelle handeln kann. Ebenso kann sich der dritte Schlüssel vom zweiten Schlüssel unterscheiden oder mit diesem identisch sein. Wenn der dritte Schlüssel mit dem zweiten identisch ist, muss das Laufwerk ihn normalerweise nicht erneut empfangen.

**[0054]** Nachstehend werden Aspekte der verschiedenen Arbeitsschritte aus **Fig. 4** beschrieben. Diese Aspekte werden lediglich beispielhaft dargestellt und sind nicht als Einschränkung gedacht. Darüber hinaus können diese Aspekte entsprechend der Vielzahl möglicher Ansätze der vorliegenden Erfindung in beliebiger Weise kombiniert werden.

**[0055]** Ein MEK kann aus zwei separaten Schlüsseln unter Verwendung jeder Methode erstellt werden, die im Fachgebiet bekannt ist und/oder die für einen Fachmann beim Lesen der vorliegenden Beschreibung ersichtlich wäre. Eine Möglichkeit, einen MEK-Schlüssel zu erstellen, für dessen Berechnung sowohl Schlüssell als auch Schlüssel2 erforderlich sind, besteht beispielsweise darin, dass Schlüssell und Schlüssel2 zwei unabhängig voneinander erzeugte Zufallszahlen sind und dann eine Berechnung durchgeführt wird, für die sowohl Schlüssell als auch Schlüssel2 erforderlich sind, z.B. die XOR-Verknüpfung oder Verkettung dieser

beiden Werte, um den MEK zu berechnen. Der MEK kann alternativ unter Verwendung eines Schlüsselaufteilungsalgorithmus (z.B. nach Shamir, Blakely (euklidisch) usw.) in zwei (oder mehr) Schlüsselanteile „aufgeteilt“ werden, z.B. Schlüssell und Schlüssel2.

**[0056]** Schlüssel1 kann aus jeder denkbaren Quelle stammen. Schlüssel1 wird bevorzugt innerhalb des SED erstellt, wobei dies der sicherste Ansatz ist, da Kopien von Schlüssell niemals außerhalb des SED vorhanden sein müssen, es sei denn, sie werden absichtlich von dort kopiert.

**[0057]** Gemäß anderen Ansätzen kann Schlüssell außerhalb des SED erstellt und dem SED zugeführt werden. Zum Beispiel kann Schlüssel1 während der Herstellung in die Einheit programmiert werden, kann während eines Schreibens der Daten in die Einheit programmiert werden usw.

**[0058]** Gemäß weiteren Ansätzen, die zusätzliche Merkmale und Arbeitsschritte eines beliebigen anderen vorliegend beschriebenen Ansatzes enthalten können, können Schlüssell und/oder Schlüssel2 in Verbindung mit einem Schreiben der Daten berechnet werden. Beispielsweise kann bei einem Schreibvorgang, bei dem ein neuer MEK verwendet werden kann, der MEK erzeugt werden (z.B. durch Zufallszahlengenerierung oder eine andere bekannte Methode), und die Schlüssell- und Schlüssel2-Schlüsselanteile können dann aus dem MEK unter Verwendung eines bekannten Schlüsselaufteilungsalgorithmus wie dem von Shamir, Blakely (euklidisch) usw. berechnet werden. Die MEK-Erzeugung und die Berechnung der Schlüsselanteile werden bevorzugt von der Einheit durchgeführt. Anschließend kann Schlüssell in der Einheit verbleiben und Schlüssel2 von dort exportiert werden, um extern gespeichert zu werden. In verschiedenen Aspekten wird diese MEK-Erzeugung präemptiv (d.h. bevor der Schreibbefehl empfangen wird, so dass sie sich nicht auf die für die Verarbeitung des Schreibvorgangs erforderliche Zeit auswirkt) oder reaktiv (z.B. in Reaktion auf den Schreibbefehl, wobei in diesem Fall Schlüssel 2 bevorzugt zur externen Speicherung aus der Einheit exportiert wird, bevor der Befehl zum Abschließen des Schreibvorgangs erfolgt) durchgeführt.

**[0059]** In einem Aspekt kann Schlüssel1 für einen bestimmten Datensatz spezifisch sein, so dass die Einheit möglicherweise mehrere eindeutige Schlüssell speichert, die jeweils einem eindeutigen Datensatz zugeordnet sind.

**[0060]** Bevorzugt ist das SED so konfiguriert, dass es keine externe Sichtbarkeit von Schlüssell oder eine Übertragung oder ein Kopieren von Schlüssell aus dem SED heraus zulässt.

**[0061]** Schlüssel2 kann intern im SED erstellt werden (wobei es sich um das bevorzugte Verfahren handelt, wenn ein Schlüsselaufteilungsalgorithmus verwendet wird, um einen Schlüsselanteil als Schlüssel2 zu erstellen) oder außerhalb des SED usw.

**[0062]** Gemäß einem veranschaulichenden Ansatz ist Schlüssel1 der erste Schlüsselanteil des MEK und ist eine erste Zufallszahl mit derselben Länge (in Bits) wie der MEK. Schlüssel2 ist der zweite Schlüsselanteil des MEK und ist eine zweite Zufallszahl mit derselben Länge wie der MEK. Schlüssell wird mit Schlüssel2 in einer vordefinierten Weise verarbeitet, um einen resultierenden MEK zu erzeugen. Zum Beispiel kann Schlüssell mit Schlüssel2 XOR-verknüpft (oder dergleichen) werden, um den MEK zu erzeugen. Alternativ können anstatt der XOR-Verknüpfung (oder dergleichen) zur Berechnung des MEK auch Standardverfahren zur Schlüsselableitung (z. B. durch Hashing oder Verschlüsselung) verwendet werden. Jede Form der Schlüsselableitung, die eine Verarbeitung sowohl von Schlüssell als auch von Schlüssel2 zur Berechnung des MEK erfordert, ist potentiell akzeptabel. Wenn Schlüsselaufteilung verwendet wurde, um Schlüssell und Schlüssel2 aus dem MEK zu erzeugen, wäre die Berechnung, die verwendet wird, um den MEK aus Schlüssell und Schlüssel2 zu erzeugen, die (zur Aufteilung) komplementäre oder inverse Berechnung.

**[0063]** Es ist zu beachten, dass bei dieser Methode der aufgeteilten Schlüssel die einzige Möglichkeit, den MEK zu berechnen, darin besteht, sowohl Schlüssell als auch Schlüssel2 zu haben, wenn man nicht über den MEK verfügt. Im Zusammenhang mit M-von-N-Schlüsselaufteilung im Allgemeinen könnte man dies als eine Implementierung von 2 von 2 Schlüsselanteilen (nämlich Schlüssell und Schlüssel2) betrachten, die erforderlich sind, um den (MEK-) Schlüssel wiederherzustellen. Eine alternative, wenn auch schlechtere Lösung wäre, M auf 2 zu setzen und eine Anzahl N von Schlüsselanteilen zu erzeugen, die größer als 2 ist, aber alle anderen Schlüsselanteile bis auf zwei zu verwerfen oder zu zerstören und somit nur die verbleibenden zwei dieser Schlüsselanteile zu verwenden (was die Schlüsselaufteilung effektiv auf 2 von 2 reduziert, aber die Komplexität der Kombinationsberechnung erhöht).

**[0064]** Ein bevorzugter Ansatz zur Implementierung eines MEK mit aufgeteilten Schlüsseln ist die Erweiterung von TCG Opal (oder einer anderen TCG-SSC) so, dass dieses Konzept der zwei Schlüsselanteile unterstützt wird, so dass ein Schlüsselanteil (Schlüssell) intern im SED und der andere Schlüsselanteil (Schlüssel2) außerhalb des SED verwaltet wird. Jeder, der mit der Komplexität der Opal-SSC von TCG vertraut ist, wird verstehen, dass es sehr viele Möglichkeiten gibt, dieses Konzept zu implementieren, und dementsprechend die Erfindung

nicht auf die vorliegend vorgestellte beispielhafte Beschreibung eingeschränkt ist.

**[0065]** In einem veranschaulichenden Beispiel wird nur die verpackte Form von Schlüssell im nichtflüchtigen Speicher des SED gespeichert. Schlüssel2 wird nur außerhalb des SED in nichtflüchtiger Form gespeichert und muss daher mindestens einmal nach einem Einschaltvorgang und/oder einem Kaltstart an das SED gesendet (oder von diesem abgerufen) werden, damit der MEK berechnet werden kann. Der MEK kann nur berechnet werden, wenn das SED sowohl über Schlüssell als auch über Schlüssel2 verfügt. Somit muss Schlüssel2 dem SED in irgendeiner Form zugeführt (oder von ihm abgerufen) werden. Dies kann je nach den verschiedenen Aspekten auf viele Arten erfolgen. Eine Möglichkeit besteht darin, dass das SED einen KMIP- (Key Management Interoperability Protocol) Client unterstützt und Schlüssel2 über einen sicheren Kanal von einem externen Schlüsselverwalter anfragt und empfängt, wie er im Fachgebiet bekannt ist. Alternativ kann Schlüssel2 dem SED in ähnlicher Weise wie Persönliche Identifikationsnummern (PINs) zugeführt werden, z.B. in Klartextform durch einen sicheren Tunnel (z.B. TLS) usw.

**[0066]** Schlüssel1 wird bevorzugt in verpackter Form im SED gespeichert und kann erst dann entpackt werden, wenn der Verpackungsschlüssel (der von PINs abhängen kann, die dem SED zur Authentifizierung verschiedener Benutzer für verschiedene vom SED unterstützte Rollen zugeführt werden) bereitgestellt wird. Dementsprechend werden alle Teile des Verpackungsschlüssels, die von außerhalb des SED bereitgestellt werden, dem SED zugeführt, damit Schlüssell entpackt werden kann. Sobald dem SED alle für die Berechnung des MEK erforderlichen Informationen (einschließlich Schlüssel2) zugeführt wurden oder es darauf Zugriff hat, berechnet das SED den MEK und ist dann in der Lage, vorhandenen chiffrierten Text zu entschlüsseln, um den resultierenden Klartext zu erzeugen (z.B. um auf einen Host Read zu reagieren), oder neu empfangene Kundendaten in Klartextform in chiffrierten Text zu verschlüsseln (z.B. um einen Host Write zu berücksichtigen).

**[0067]** Gemäß einem Ansatz ist Schlüssel2 die PIN, die einer Rolle eines administrativen Sicherheitsanbieters (im Folgenden „AdminSP“, Administrative Security Provider) zugeordnet ist, die aktiviert wird, sobald sich eine autorisierte Instanz wie der Laufwerkseigentümer mit der Systemidentifikationsnummer (SID) oder einer anderen PIN, die einer Rolle zugeordnet ist, die über die entsprechende Berechtigung verfügt, authentifiziert hat.

**[0068]** Die PIN von Schlüssel2 kann geändert werden. Gemäß einem Ansatz würde die PIN von

Schlüssel2 mit einem Standardwert beginnen, z.B. mit einem vordefinierten Wert wie z.B. nur Nullen (0x000...). Eine denkbare Implementierung würde es erlauben, die PIN von Schlüssel2 zu ändern, allerdings nur bevor die PIN der ersten administrativen Berechtigung des AdminSP (z.B. AdminSP.Admin1) individualisiert wird, um versehentliche Krypto-Löschung von Daten, die durch den MEK verschlüsselt wurden, durch unbeabsichtigtes Ändern der PIN von Schlüssel2 zu verhindern. Alternativ kann die PIN von Schlüssel2 jederzeit geändert werden, um eine andere Möglichkeit zu bieten, die Krypto-Löschung aufzurufen.

**[0069]** Wenn Schlüssel2 auf eine neu generierte Zufallszahl gesetzt wird, könnte es scheinen, als ob die PIN von Schlüssel2 logischerweise außerhalb des SED erzeugt würde (z.B. durch den Benutzer und seine externe Schlüsselverwaltungsinfrastruktur), obwohl diese Generierung in Wirklichkeit einfach durch den Aufruf der Random-Methode des SED erfolgt sein könnte (ein Befehl, den das SED unterstützen muss (es ist ein nicht authentifizierter Dienst), wenn es die Opal-Spezifikation von TCG erfüllt). Als Antwort auf die Random-Methode berechnet das SED eine neue Zufallszahl unter Verwendung einer beliebigen akzeptablen Methode und sendet (liefert) diese an den Host zurück.

**[0070]** Eine alternative Implementierung sieht vor, dass das SED in einigen oder allen Fällen die Quelle für die Erzeugung von Schlüssel2 ist. In diesem Fall kann der Schlüssel2 aus dem SED gelesen werden. Idealerweise folgt diese Implementierung dem vorstehend vorgeschlagenen Weg, d.h. sie ermöglicht die Umsetzung mit bestehenden TCG-Opal-Verfahren (Befehlen). Ein Ansatz verwendet beispielsweise die Random-Methode in einem SED, um Schlüssel2 zu erzeugen. Dann wird die PIN des AdminSP.Schlüssel2 vom Host auf denselben Zufallswert personalisiert, den er gerade über Random aus dem SED gelesen hat. Es ist zu beachten, dass der Host durch das Ändern der PIN von Schlüssel2 gegenüber ihrem Standardwert die Zuständigkeit für die Speicherung dieser PIN übernommen hat. Zu diesem Zweck speichert das SED nur die zuletzt über Random erzeugte Zufallszahl im flüchtigen Speicher und bestimmt diese zuletzt erzeugte Zufallszahl als Kandidaten-Schlüssel2. Wenn die PIN des AdminSP.Schlüssel2 auf denselben Wert gesetzt ist, erlaubt das SED die Änderung der Schlüssel2.PIN auf diesen neuen Wert. Wenn der Host versucht, die AdminSP.Schlüssel2-PIN auf einen anderen Wert als die letzte von Random vorgegebene Zufallszahl zu setzen, kann das SED wahlweise die Annahme dieser PIN-Änderung verweigern und den entsprechenden Fehlercode ausgeben, der angibt, dass der Befehl fehlgeschlagen ist. In diesem Fall kann das SED den Standardwert der PIN von Schlüssel2 (d.h. nur Nullen = 0x000...) bei-

behalten, was das SED im Wesentlichen auf denselben Standardbetrieb zurücksetzt (d.h. wie ein SED, das TCGs Opal entspricht, heute funktionieren würde).

## Kryptographie

**[0071]** Es ist zu beachten, dass eine Standardmethode für ein SED zur Authentifizierung einer PIN darin besteht, sie zu hashen und mit dem Hash der PIN zu vergleichen, der bei der erstmaligen Festlegung berechnet (und seitdem nichtflüchtig im SED gespeichert) wurde. In der vorstehenden Diskussion wurde die PIN von AdminSP.Schlüssel2 als Schlüssel2 verwendet. Es ist zu beachten, dass diese spezielle PIN insofern anders als andere behandelt werden sollte, als kein Hash dieser PIN im SED gespeichert werden sollte, da dies irgendwann in der Zukunft unsicher werden könnte (d.h. wenn der Hash in der Zukunft gebrochen wird, z.B. durch Quantencomputer). Zumindest würde der Hash eine Möglichkeit bieten, Brute-Force-Rateversuche zu testen. Es gibt also triftige Gründe dafür, sich nicht einfach auf einen Hash der PIN als Mittel zur Authentifizierung der PIN des AdminSP.Schlüssel2 zu verlassen. Dementsprechend kann die PIN von Schlüssel2 nicht unabhängig von der Validierung des MEK selbst authentifiziert werden, weshalb die Rolle von Schlüssel2 vorzugsweise keine andere Berechtigung hat, als ihr eigenes Passwort zu ändern. Damit entfällt jede direkte, unmittelbare Notwendigkeit, diese PIN zu authentifizieren.

**[0072]** Schließlich werden Schlüssel1 und Schlüssel2 XOR-verknüpft, kombiniert oder dergleichen, um den MEK zu erstellen.

**[0073]** Um zu überprüfen, ob der resultierende MEK korrekt ist (z.B. im Gegensatz zu einer Abweichung von einem Bit, was der Fall wäre, wenn Schlüssel2 um ein Bit abweiche), kann ein Schlüsselsignaturkonzept verwendet werden. Das Schlüsselsignaturkonzept umfasst die Verwendung des Schlüssels zur Verschlüsselung eines bekannten Wertes, und der resultierende chiffrierte Text wird dann als Signatur dieses Schlüssels gespeichert.

**[0074]** Ein bevorzugter Ansatz zur Überprüfung der Korrektheit des resultierenden MEK besteht darin, Schlüsselverwaltung intern im SED so durchzuführen, dass die Verfügbarkeit von Schlüssel2 eine Voraussetzung für ein erfolgreiches Entpacken von Schlüssel1 ist. Beispielsweise kann ein Schlüsselstrom durch Schlüsselableitung von Schlüssel2 erzeugt werden, und dieser Schlüsselstrom wird mit der verpackten Schlüsselstruktur XOR-verknüpft oder dergleichen, bevor er in nichtflüchtiger Form in der Einheit gespeichert wird. Die 8-Byte-Prüfsumme, die in den Advanced Encryption Standard (AES) Key Wrap eingebaut ist, dient dann dazu, sowohl Schlüs-

sel2 als auch den anderen Verpackungsschlüssel zu validieren, der zum Verpacken von Schlüssel2 verwendet wurde. Dementsprechend wird bevorzugt eine Schlüsselableitung auf Schlüssel2 angewendet, um genügend abgeleitetes Schlüsselmaterial zu erzeugen, um die gesamte verpackte Schlüsselstruktur zu umfassen. Die XOR-Verknüpfung dieses Schlüsselstroms mit der verpackten Schlüsselstruktur (WKS, wrapped key structure) erzeugt eine neue geschützte verpackte Schlüsselstruktur (PWKS, protected wrapped key structure), die anstelle der ursprünglichen WKS gespeichert wird. Zu diesem Zeitpunkt hätte jedes Bit in der PWKS eine 50 %ige Wahrscheinlichkeit, richtig oder falsch zu sein, und jede spekulative „Korrektur“ könnte sich (bei ausreichender konvergenter Korrektur) im Wesentlichen für irgendeine der sehr zahlreichen Möglichkeiten der Schlüsselpackstruktur entscheiden. Dementsprechend ist die XOR-Verknüpfung eines Schlüsselstroms mit der verpackten Schlüsselstruktur selbst praktisch eine zweite Verpackung, was einen weiteren Ansatz ermöglicht, bei dem es sich bei den Daten innerhalb der doppelten Verpackung um den MEK handelt und die Kombination durch ein Durchführen beider Entpackungen erfolgt.

**[0075]** Es ist zu beachten, dass die PIN von Schlüssel2 in einigen der vorgenannten Ansätze zu einem so genannten kryptographisch sensiblen Parameter (CSP, Cryptographically Sensitive Parameter) geworden ist. Wenn eine Möglichkeit besteht, dass die zwischen dem SED und dem Host übertragenen Daten aufgezeichnet werden könnten, sollte der Kanal zwischen ihnen mit einer Form von Encryption of Data in Flight (EDiF) wie Internet Protocol Security (IPsec), Fibre Channel-Security Protocol (FC-SP) oder Transport Layer Security (TLS) geschützt werden. In einigen Rechenzentren besteht keine Besorgnis darüber, dass Daten in dieser Umgebung von und zu einem SED geschickt werden; die Besorgnis bezieht sich nur darauf, was mit dem SED geschieht, nachdem es die geschützte Umgebung verlassen hat, weshalb eine absolute Sicherheit der Krypto-Löschung wünschenswert ist.

**[0076]** Es ist zu beachten, dass es im Falle des XTS-Verschlüsselungsmodus (z.B. XTS-AES-256) zwei verschlüsselungsbezogene Schlüssel gibt, einen Chiffrierschlüssel und einen separaten Tweak-Schlüssel. Gemäß einigen Ansätzen können beide Schlüssel (durch Schlüsselableitung) aus einem einzigen Stammschlüssel erzeugt werden. Dementsprechend kann ein 256-Bit-MEK bereitgestellt werden, der zur Erzeugung der beiden für XTS-AES-256 erforderlichen 256-Bit-Schlüssel (durch Schlüsselableitung) verwendet werden kann.

**[0077]** Zu den Optionen für Krypto-Löschung in SEDs, die verschiedene Aspekte der vorliegenden

Erfindung umgesetzt haben, zählen eine oder mehrere der folgenden:

1. Möchte ein Benutzer ein betriebsfähiges SED einer Krypto-Löschung unterziehen, so ruft er einfach eine der vielen verschiedenen Möglichkeiten auf, die vorliegend vorgeschlagen werden. Die verpackten Schlüsselstrukturen werden überschrieben, und Schlüssel2 ist nun nicht mehr wiederherstellbar. In diesem Szenario besteht keine Notwendigkeit, auch Schlüssel2 zu löschen - der MEK kann nicht wiederhergestellt werden, da Schlüssel2 nicht mehr vorhanden ist.

2. Wenn ein Benutzer an einem SED eine Krypto-Löschung durchführen möchte (möglicherweise, weil es sich um ein nicht betriebsfähiges SED handelt, das nicht auf Befehle reagiert, oder um ein fehlendes SED), aber aus irgendeinem Grund „1.“ nicht durchführen kann, hat der Benutzer nun eine weitere Möglichkeit - er kann stattdessen den Schlüssel2 löschen, den er unabhängig vom SED kontrolliert. In diesem Szenario ist es nicht notwendig, alle verpackten Versionen von Schlüssel2 zu löschen; der MEK ist nicht wiederherstellbar, da Schlüssel2 nicht mehr vorhanden ist. Selbst wenn es jemandem irgendwann in der Zukunft gelingen sollte, die verpackten Schlüsselstrukturen eines ausgefallenen SED aufzubrechen und sich so Zugang zu Schlüssel1 zu verschaffen, wäre die ganze Arbeit umsonst, da der MEK weiterhin nicht verfügbar ist. Hierfür würde der gelöschte Schlüssel2 benötigt. Dementsprechend bestünde weiterhin die Situation, dass der einzige gangbare Weg zum Zugriff auf Kundendaten darin bestünde, den Verschlüsselungsalgorithmus (z.B. XTS-AES-256), der den chiffrierten Text der Benutzerdaten schützt, an sich zu brechen.

3. Wenn ein Benutzer besonders auf die Sicherheit der Krypto-Löschung bedacht ist, kann er sich dafür entscheiden, sowohl Schlüssel1 als auch Schlüssel2 zu löschen. Dieses Szenario ist am wahrscheinlichsten, wenn die Befürchtung besteht, dass Schlüssel2 aufgezeichnet wurde (z.B. auf dem Weg zum SED, auch wenn er durch einen sicheren Kanal geschützt war, dessen Sicherheit (vielleicht später) gebrochen wurde). Andernfalls sollte es jedoch aufgrund der Verschlüsselung keinen Grund für eine Löschung von Schlüssel1 und Schlüssel2 geben. Ein allgemeiner Ansatz, der relativ robust ist (z.B. gegenüber dem Ausfall des SED), besteht darin, immer zu versuchen, sowohl Schlüssel1 als auch Schlüssel2 zu löschen, sich aber damit zufrieden zu geben, wenn nachgewiesen wird, dass zumindest eine der beiden Löschungen erfolgreich war.

## Krypto-Löschung für Magnetbänder und andere tragbare Speicher

**[0078]** Wie bereits erwähnt, werden gemäß einigen Ansätzen verschlüsselte Daten auf einem nichtflüchtigen Speichermedium wie einem magnetischen Aufzeichnungsmedium (z.B. Band, Festplatte) oder einem Halbleiterspeicher (z.B. NAND-Flash, NVRAM usw.) gespeichert. Auch in diesem Fall kann jeder/s der vorstehend vorgestellten Arbeitsschritte, Konzepte usw. in den vorliegenden Ansätzen verwendet werden.

**[0079]** In **Fig. 5** ist ein Ablaufplan eines Verfahrens 500 gemäß einem Ansatz gezeigt. Gemäß der vorliegenden Erfindung kann gemäß verschiedenen Ansätzen das Verfahren 500 unter anderem in jeder der in den anderen vorliegend beschriebenen Figuren abgebildeten Umgebungen durchgeführt werden. Im Verfahren 500 können natürlich mehr oder weniger Arbeitsschritte als die konkret in **Fig. 5** beschriebenen enthalten sein, wie ein Fachmann bei Lektüre der vorliegenden Beschreibungen versteht.

**[0080]** Jeder der Schritte des Verfahrens 500 kann durch jede geeignete Komponente der Betriebsumgebung durchgeführt werden. Beispielsweise kann in verschiedenen Ansätzen das Verfahren 500 teilweise oder ganz durch einen Computer, ein Laufwerk oder eine andere Einheit mit einem oder mehreren darin befindlichen Prozessoren durchgeführt werden. Der Prozessor, z.B. Verarbeitungsschaltung(en), Chip(s) und/oder Modul(e), die in Hardware und/oder Software implementiert sind und bevorzugt über mindestens eine Hardware-Komponente verfügen, kann in jeder Einheit genutzt werden, um einen oder mehrere Schritte des Verfahrens 500 durchzuführen. Zu veranschaulichenden Prozessoren zählen, ohne jedoch hierauf eingeschränkt zu sein, eine zentrale Verarbeitungseinheit (CPU), eine anwendungsspezifische integrierte Schaltung (ASIC), eine vor Ort programmierbare Gatter-Anordnung (FPGA) usw., Kombinationen aus diesen oder irgendeine andere geeignete Datenverarbeitungseinheit, die im Fachgebiet bekannt ist.

**[0081]** Wie in **Fig. 5** gezeigt, kann das Verfahren 500 mit dem Arbeitsschritt 502 beginnen, bei dem eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen wie beispielsweise Lesen und/oder Schreiben von Daten von dem/auf das Speichermedium durchführt, eine Anfrage zum Lesen von Daten empfängt, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind. Die empfangene Anfrage kann einfach die Daten anfragen oder zusätzliche Informationen wie den Schlüssel2 enthalten (siehe unten).

**[0082]** In Arbeitsschritt 504 wird ein erster Schlüssel (Schlüssel1), der auf und/oder mit dem Speichermedium gespeichert ist, abgerufen. Diese Prozedur kann Abrufen des ersten Schlüssels von dem Medium, auf dem die Daten gespeichert sind, von einem Speicher, der physisch mit dem Medium verbunden ist, wie beispielsweise einem Kassettenspeicher, und dergleichen umfassen.

**[0083]** Das Speichermedium kann beliebiger vorliegend offenbarer Art sein, z.B. Magnetband, Magnetplatte, NVRAM usw. Dementsprechend kann es sich bei der Einheit um eine beliebige Art von Datenspeichereinheit handeln, z.B. Bandlaufwerk, SSD, HDD usw.

**[0084]** Wie vorstehend erwähnt, ist gemäß einigen Ansätzen der erste Schlüssel auf und/oder mit dem Speichermedium in unverschlüsselter Form gespeichert. Gemäß anderen Ansätzen ist der erste Schlüssel auf und/oder mit dem Speichermedium in verpackter Form gespeichert (z.B. verschleierte Form, verschlüsselte Form, passwortgeschützte Form usw.). Informationen wie ein weiterer Schlüssel oder eine PIN, ein Kennwort usw. zum Entpacken des ersten Schlüssels können empfangen und zum Entpacken des ersten Schlüssels verwendet werden.

**[0085]** Gemäß bevorzugten Aspekten ist die Einheit so konfiguriert, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet. Eine innerhalb der Einheit befindliche Steuereinheit kann jedoch auf den ersten Schlüssel von dem Speichermedium oder einer anderen mit dem Speichermedium verbundenen Quelle aus zugreifen.

**[0086]** In Arbeitsschritt 506 wird ein zweiter Schlüssel (Schlüssel2) von einer externen Quelle empfangen. Bei der externen Quelle kann es sich um eine beliebige externe Quelle handeln. Beispielsweise kann es sich bei der externen Quelle um den Anfrager der Daten, einen Schlüsselspeicher, einen Schlüsselserver oder eine Schlüsseleingabe durch einen Benutzer handeln (z.B. über eine Tastatur usw.).

**[0087]** In Arbeitsschritt 508 wird unter Verwendung des ersten und des zweiten Schlüssels ein Medienchiffrierschlüssel (MEK) erzeugt. Es kann eine beliebige bekannte Methode verwendet werden, um aus dem ersten und dem zweiten Schlüssel einen MEK zu erzeugen.

**[0088]** In Arbeitsschritt 510 werden die angefragten Daten in chiffrierter Form aus dem Speichermedium gelesen, z.B. unter Verwendung herkömmlicher Methoden.

**[0089]** In Arbeitsschritt 512 wird der chiffrierte Text der angefragten Daten unter Verwendung des MEK

entschlüsselt. Die Daten können in einen Puffer kopiert und dann entschlüsselt werden, sie können während des Lesens „on the fly“ entschlüsselt werden (z.B. werden die Arbeitsschritte 510 und 512 gleichzeitig ausgeführt) usw.

**[0090]** In Arbeitsschritt 514 werden die entschlüsselten Daten ausgegeben, z.B. an den Anfrager der Daten usw.

**[0091]** Wenn zweite Daten von der Einheit auf dem Speichermedium gespeichert werden sollen, werden die zweiten Daten empfangen. Ein dritter Schlüssel kann von einer zweiten externen Quelle empfangen werden. Ist dies der Fall, wird unter Verwendung des ersten und dritten Schlüssels ein zweiter Medienchiffrierschlüssel erzeugt. Die Daten werden unter Verwendung des zweiten Medienchiffrierschlüssels verschlüsselt, und die verschlüsselten Daten werden auf das Speichermedium geschrieben. Es ist zu beachten, dass es sich deshalb bei der zweiten externen Quelle um dieselbe wie die vorstehend genannte Quelle oder um eine andere Quelle handeln kann. Ebenso kann sich der dritte Schlüssel, sofern ein solcher empfangen wird, vom zweiten Schlüssel unterscheiden oder mit diesem identisch sein.

**[0092]** Nachstehend werden Aspekte der verschiedenen Arbeitsschritte aus **Fig. 5** beschrieben. Diese Aspekte werden lediglich beispielhaft dargestellt und sind nicht als Einschränkung gedacht. Darüber hinaus können diese Aspekte entsprechend der Vielzahl möglicher Ansätze der vorliegenden Erfindung in beliebiger Weise kombiniert werden. Das Verfahren 500 kann beispielsweise ähnliche Merkmale besitzen oder beinhalten wie die vorstehend in Bezug auf **Fig. 4** dargestellten, z.B. Schlüsselerzeugung usw.

**[0093]** Schlüssel1 ist mit dem Speichermedium gespeichert, z.B. auf dem Medium selbst und/oder mit dem Speichermedium, z.B. in einem separaten Speicher. Schlüssell ist bevorzugt in verpackter Form gespeichert. Beispielsweise kann Schlüssell auf einem Teil eines Magnetbandes, in einem Bandkassettenspeicher oder beidem gespeichert sein. Auf diese Weise ist Schlüssell zusammen mit dem Magnetband tragbar. Bei einer NVRAM-Einheit wie einem Speicherstick, einer Speicherkarte usw. kann Schlüssell im NVRAM der Einheit, in einem separaten Speicher der Einheit usw. gespeichert sein.

**[0094]** Schlüssel1 wird bevorzugt intern in der Einheit (z.B. Laufwerk, Computer usw.) erstellt, die auf dem Speichermedium arbeitet. Die Einheit ist bevorzugt so konfiguriert, dass sie keine externe Sichtbarkeit von Schlüssell oder ein Kopieren von Schlüssel1 nach außerhalb der Einheit zulässt, mit Ausnahme des mit ihr verbundenen Speichermediums und/oder

Speichers. Bevorzugt überträgt die Einheit den Schlüssel1 auf das Speichermedium und/oder den mit ihr verbundenen Speicher und löscht den Schlüssel1 aus der Einheit. Gemäß einigen Ansätzen, beispielsweise denen, die mit auswechselbaren Medieneinheiten verbunden sind, befindet sich der Schlüssel1 somit nur auf dem Speichermedium.

**[0095]** Bei einer Einheit, bei der das Speichermedium in die Einheit selbst integriert ist (z.B. ein selbst-verschlüsselndes (SED-) HDD- oder SSD-Laufwerk, ein NVRAM-Modul usw.), kann der Schlüssel1 mit der Einheit gespeichert werden. In einem (möglicherweise hermetisch) versiegelten SED-Gehäuse kann der Schlüssel1 beispielsweise an einer beliebigen Stelle innerhalb dieser typischerweise rechteckigen Begrenzung aufbewahrt werden. Bei einem SSD handelt es sich bei dem nichtflüchtigen Hauptspeicher (in dem die Benutzerdaten gespeichert werden) häufig um einen NAND-Flashspeicher. Schlüssell (möglicherweise in verpackter Form) kann dort ebenfalls gespeichert werden. Bei anderen Ansätzen kann Schlüssel1 stattdessen in einem separaten (von den Benutzerdaten getrennten) nichtflüchtigen Speicher innerhalb des Gehäuses aufbewahrt werden, z.B. in einem anderen NAND-Flash- oder NOR-Flash-Chip, in einem magnetischen Direktzugriffsspeicher (MRAM), in einem Spin-Transfer-Torque-Direktzugriffsspeicher (STT-RAM), in einem ferroelektrischen Direktzugriffsspeicher (FeRAM), in einem Phasenwechselspeicher (PCM, phase-change memory), in einem resistiven Direktzugriffsspeicher (RRAM), in anderen Formen von NVRAM (was weniger technologiespezifisch ist als die vorhergehenden Bezeichnungen) usw. Es ist zu beachten, dass Schlüssel1 auch in einer Form von ROM gespeichert werden kann, die gelöscht werden kann, wie z.B. EEPROM. Gemäß verschiedenen Ansätzen kann Schlüssel1 somit in einem anderen Speichertyp gespeichert werden als das Speichermedium, auf dem die Benutzerdaten gespeichert sind.

**[0096]** Gemäß anderen Ansätzen wird der Schlüssel1 außerhalb der Einheit erstellt und der Einheit zugeführt und in dieser gespeichert.

**[0097]** Schlüssel2 kann innerhalb der Einheit oder außerhalb der Einheit oder durch gemeinsame Teilschritte von innerhalb und außerhalb der Einheit erstellt werden.

**[0098]** Schlüssel2 wird dem flüchtigen Speicher (z.B. SRAM) im Bandlaufwerk, Computer usw. zugeführt und/oder darin gespeichert, damit das Laufwerk, der Computer usw. den MEK berechnen kann. Gemäß einem Ansatz kann Schlüssel2 jedes Mal, wenn er benötigt wird, in das Bandlaufwerk, den Computer usw. eingegeben werden. Gemäß einem anderen Ansatz kann Schlüssel2, nachdem er der

Einheit zugeführt wurde, intern in der Einheit in einem flüchtigen Speicher auf dem Bandlaufwerk, Computer usw. gespeichert werden, wobei in diesem Fall Schlüssel2 bevorzugt verpackt wird.

**[0099]** Es sollte beachtet werden, dass verschiedene Ansätze vorliegend mit einer breiten Palette von Speichermedien implementiert werden können, darunter zum Beispiel NVRAM-Technologien wie NAND-Flashspeicher, NOR-Flashspeicher, Phasenwechselspeicher (PCM), magnetoresistiver RAM (MRAM) und resistiver RAM (RRAM). Um einen Kontext zu liefern und lediglich als Hilfestellung für den Leser können verschiedene Ansätze unter Bezugnahme auf einen nichtflüchtigen Speichertyp beschrieben werden. Hierbei handelt es sich lediglich um Beispiele, die nicht als Einschränkung der in den Ansprüchen definierten Erfindung angesehen werden sollten.

**[0100]** Gemäß einem beispielhaften Ansatz wird Schlüssel1 in nichtflüchtiger Form nur in verpackter Form mit dem Speichermedium gespeichert. Und Schlüssel2 kann zwar vorübergehend in einem flüchtigen Speicher abgelegt werden, wird aber in nichtflüchtiger Form nur außerhalb des Speichermediums gespeichert.

**[0101]** Das Laufwerk, der Computer usw., das bzw. der auf dem Speichermedium arbeitet, ruft den Schlüssel1 aus dem Speichermedium ab und empfängt oder erhält den Schlüssel2, um die Berechnung des MEK zu ermöglichen.

**[0102]** Schlüssel2, der bevorzugt nicht lokal auf dem Laufwerk, Computer usw. gespeichert ist, kann dem Laufwerk, Computer usw. in irgendeiner Form zugeführt werden. Hierfür gibt es viele Möglichkeiten, je nach den verschiedenen Aspekten, einschließlich der im vorherigen Abschnitt aufgeführten Ansätze. Eine Möglichkeit besteht beispielsweise darin, dass das Laufwerk, der Computer usw. einen KMIP-Client unterstützt und den Schlüssel2 über einen sicheren Kanal von einem externen KMIP-Server (Schlüsselmanager) empfängt, wie er im Fachgebiet bekannt ist.

**[0103]** Schlüssel2 wird gemäß einigen Ansätzen lokal auf dem Laufwerk, Computer usw. gespeichert, bevorzugt in verpackter Form. Ein Schlüssel zum Entpacken von Schlüssel2 kann von einem Benutzer, von einem Schlüsselmanager usw. empfangen werden.

**[0104]** Schlüssel1 ist bevorzugt mit dem Speichermedium in verpackter Form gespeichert und kann entpackt werden, sobald das Laufwerk, der Computer usw. den Verpackungsschlüssel erhält (der eine persönliche Identifikationsnummer (PIN) sein oder von einer solchen abhängen kann, die dem Lauf-

werk, Computer usw. zugeführt wird, um verschiedene Benutzer für verschiedene Rollen zu authentifizieren, die das Laufwerk, der Computer usw. unterstützt). Dementsprechend werden alle Teile des Verpackungsschlüssels, die von außerhalb des Laufwerks, Computers usw. bereitgestellt werden, dem Laufwerk, Computer usw. zugeführt, damit Schlüssel1 entpackt werden kann. Sobald das Laufwerk, der Computer usw. alle erforderlichen Informationen (einschließlich Schlüssel2) gesammelt hat, die nötig sind, um den MEK zu berechnen, berechnet das Laufwerk, der Computer usw. den MEK und ist dann in der Lage, vorhandenen chiffrierten Text zu entschlüsseln, um den resultierenden Klartext zu erzeugen (z.B. um auf einen Host Read zu reagieren), oder neue Kundendaten in Klartextform in chiffrierten Text zu verschlüsseln (z.B. um einen Host Write zu berücksichtigen).

**[0105]** Gemäß einem Ansatz ist der Schlüssel2 eine PIN, wie im vorherigen Abschnitt beschrieben.

**[0106]** Eine alternative Implementierung sieht vor, dass in einigen oder allen Fällen das Laufwerk, der Computer usw. die Quelle für die Erzeugung von Schlüssel2 ist. In diesem Fall kann der Schlüssel2 zunächst aus dem Speicher des Laufwerks, des Computers usw. gelesen werden.

**[0107]** Bei Ansätzen, die mit einem Bandlaufwerk implementiert werden, ist es bevorzugt, dass der Schlüssel1 vom Bandmedium und der Schlüssel2 von einer Schnittstelle kommt und dass die Berechnung des MEK intern im Bandlaufwerk erfolgt und dieser das Bandlaufwerk nie verlässt.

Schreiben mit SED mit aktivierter Krypto-Löschung

**[0108]** Wenngleich ein Großteil der nachfolgenden Beschreibung in Verbindung mit einer beispielhaften Implementierung mit einem SED dargestellt wird, geschieht dies lediglich beispielhaft und einzig, um dem Leser hilfsweise einen Kontext bereitzustellen. Dementsprechend sind die nachstehend vorgestellten Konzepte und Lehren ebenso auf Implementierungen mit Speichermedien wie Magnetbändern, Speicherkarten usw. anwendbar.

**[0109]** In **Fig. 6** ist ein Ablaufplan eines Verfahrens 600 gemäß einem Ansatz gezeigt. Gemäß der vorliegenden Erfindung kann gemäß verschiedenen Ansätzen das Verfahren 600 unter anderem in jeder der in den anderen vorliegend beschriebenen Figuren abgebildeten Umgebungen durchgeführt werden. Beispielsweise kann das Verfahren 600 ähnliche Merkmale aufweisen oder beinhalten wie die vorstehend in Bezug auf **Fig. 4** dargestellten, und in einigen Aspekten kann eine Einheit in der Lage sein, die Arbeitsschritte aus **Fig. 4** und **Fig. 6** durchzuführen. Im Verfahren 600 können natürlich mehr oder

weniger Arbeitsschritte als die konkret in **Fig. 6** beschriebenen enthalten sein, wie ein Fachmann bei Lektüre der vorliegenden Beschreibungen versteht.

**[0110]** Jeder der Schritte des Verfahrens 600 kann durch jede geeignete Komponente der Betriebsumgebung durchgeführt werden. Beispielsweise kann in verschiedenen Ansätzen das Verfahren 600 teilweise oder ganz durch einen Computer, ein Laufwerk oder eine andere Einheit mit einem oder mehreren darin befindlichen Prozessoren durchgeführt werden. Der Prozessor, z.B. Verarbeitungsschaltung(en), Chip(s) und/oder Modul(e), die in Hardware und/oder Software implementiert sind und bevorzugt über mindestens eine Hardware-Komponente verfügen, kann in jeder Einheit genutzt werden, um einen oder mehrere Schritte des Verfahrens 600 durchzuführen. Zu veranschaulichenden Prozessoren zählen, ohne jedoch hierauf eingeschränkt zu sein, eine zentrale Verarbeitungseinheit (CPU), eine anwendungsspezifische integrierte Schaltung (ASIC), eine vor Ort programmierbare Gatter-Anordnung (FPGA) usw., Kombinationen aus diesen oder irgendeine andere geeignete Datenverarbeitungseinheit, die im Fachgebiet bekannt ist.

**[0111]** Wie in **Fig. 6** gezeigt, kann das Verfahren 600 mit dem Arbeitsschritt 602 beginnen, bei dem eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen wie beispielsweise Lesen und/oder Schreiben von Daten von dem/auf das Speichermedium durchführt, eine Anfrage zum Schreiben von Daten in verschlüsselter Form auf das Speichermedium empfangen kann. Die Anfrage kann einfach das Schreiben der Daten erbiten oder zusätzliche Informationen wie den Schlüssel2 enthalten (siehe unten).

**[0112]** In Arbeitsschritt 604 wird z.B. von einem Speicher der Einheit, von einer ASIC usw. aus auf einen in der Einheit gespeicherten ersten Schlüssel (Schlüssel1) zugegriffen. Wie vorstehend erwähnt, ist gemäß einigen Ansätzen der erste Schlüssel in Roh-(unverschlüsselter) Form in der Einheit gespeichert. Gemäß anderen Ansätzen ist der erste Schlüssel in der Einheit in verpackter Form gespeichert (z.B. verschleierte Form, verschlüsselte Form, passwortgeschützte Form usw.). Informationen wie ein weiterer Schlüssel, ein Kennwort usw. zum Entpacken des ersten Schlüssels können empfangen und zum Entpacken des ersten Schlüssels verwendet werden.

**[0113]** Gemäß bevorzugten Aspekten ist die Einheit so konfiguriert, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet. Eine in der Einheit befindliche Steuereinheit kann jedoch auf den ersten Schlüssel innerhalb der Einheit zugreifen.

**[0114]** In Arbeitsschritt 606 wird ein zweiter Schlüssel (Schlüssel2) von einer externen Quelle empfangen. Bei der externen Quelle kann es sich um eine beliebige externe Quelle handeln. Beispielsweise kann es sich bei der externen Quelle um den Anfrager der Daten, einen Schlüsselspeicher, einen Schlüsselserver, eine Schlüsseleingabe durch einen Benutzer z.B. über eine Tastatur usw. handeln.

**[0115]** In Arbeitsschritt 608 wird unter Verwendung des ersten und des zweiten Schlüssels ein Medienchiffrierschlüssel (MEK) erzeugt. Es kann eine beliebige bekannte Methode verwendet werden, um aus dem ersten und dem zweiten Schlüssel einen MEK zu erstellen, z.B. durch XOR-Verknüpfung des ersten und des zweiten Schlüssels.

**[0116]** In Arbeitsschritt 610 wird die verschlüsselte Form (d.h. der chiffrierte Text) der angefragten Daten unter Verwendung des Medienchiffrierschlüssels auf das Speichermedium geschrieben, z.B. mit herkömmlichen Methoden. Das Speichermedium kann beliebiger vorliegend offenbarer Art sein, z.B. Magnetband, Magnetplatte, NVRAM usw. Dementsprechend kann es sich bei der Einheit um eine beliebige Art von Datenspeichereinheit handeln, z.B. Bandlaufwerk, SSD, HDD, verschlüsselungsfähige USB-Datenträger, NVRAM-Module, usw.

**[0117]** Aspekte der verschiedenen Arbeitsschritte und Komponenten aus **Fig. 6** ähneln bevorzugt denen, die vorstehend im Abschnitt, der **Fig. 4** beschreibt, beschrieben wurden, können aber auch Merkmale aus jedem anderen vorliegenden Abschnitt enthalten. Diese Aspekte werden lediglich beispielhaft dargestellt und sind nicht als Einschränkung gedacht. Darüber hinaus können diese Aspekte entsprechend der Vielzahl möglicher Ansätze der vorliegenden Erfindung in beliebiger Weise kombiniert werden.

Schreiben auf Magnetband und andere tragbare Speicher

**[0118]** Wie bereits erwähnt, werden gemäß einigen Ansätzen verschlüsselte Daten auf einem nichtflüchtigen Speichermedium wie einem magnetischen Aufzeichnungsmedium (z.B. Band, Festplatte) oder einem Halbleiterspeicher (z.B. NAND-Flash, NVRAM usw.) gespeichert. Auch in diesem Fall kann jeder/s der vorstehend vorgestellten Arbeitsschritte, Konzepte usw. in den vorliegenden Ansätzen verwendet werden.

**[0119]** In **Fig. 7** ist ein Ablaufplan eines Verfahrens 700 gemäß einem Ansatz gezeigt. Gemäß der vorliegenden Erfindung kann gemäß verschiedenen Ansätzen das Verfahren 700 unter anderem in jeder der in den anderen vorliegend beschriebenen Figuren abgebildeten Umgebungen durchgeführt wer-

den. Beispielsweise kann das Verfahren 700 ähnliche Merkmale aufweisen oder beinhalten wie die vorstehend in Bezug auf **Fig. 4** und **Fig. 5** dargestellten, und gemäß einigen Aspekten kann eine Einheit in der Lage sein, die Arbeitsschritte aus **Fig. 5** und **Fig. 7** durchzuführen. Im Verfahren 700 können natürlich mehr oder weniger Arbeitsschritte als die konkret in **Fig. 7** beschriebenen enthalten sein, wie ein Fachmann bei Lektüre der vorliegenden Beschreibungen versteht.

**[0120]** Jeder der Schritte des Verfahrens 700 kann durch jede geeignete Komponente der Betriebsumgebung durchgeführt werden. Beispielsweise kann in verschiedenen Ansätzen das Verfahren 700 teilweise oder ganz durch einen Computer, ein Laufwerk oder eine andere Einheit mit einem oder mehreren darin befindlichen Prozessoren durchgeführt werden. Der Prozessor, z.B. Verarbeitungsschaltung(en), Chip(s) und/oder Modul(e), die in Hardware und/oder Software implementiert sind und bevorzugt über mindestens eine Hardware-Komponente verfügen, kann in jeder Einheit genutzt werden, um einen oder mehrere Schritte des Verfahrens 700 durchzuführen. Zu veranschaulichenden Prozessoren zählen, ohne jedoch hierauf eingeschränkt zu sein, eine zentrale Verarbeitungseinheit (CPU), eine anwendungsspezifische integrierte Schaltung (ASIC), eine vor Ort programmierbare Gatter-Anordnung (FPGA) usw., Kombinationen aus diesen oder irgendeine andere geeignete Datenverarbeitungseinheit, die im Fachgebiet bekannt ist.

**[0121]** Wie in **Fig. 7** gezeigt, kann das Verfahren 700 mit dem Arbeitsschritt 702 beginnen, bei dem eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen wie beispielsweise Lesen und/oder Schreiben von Daten von dem/auf das Speichermedium durchführt, eine Anfrage zum Schreiben von Daten in verschlüsselter Form auf das Speichermedium empfangen kann. Die empfangene Anfrage kann einfach das Schreiben der Daten erbitten oder zusätzliche Informationen wie den Schlüssel2 enthalten (siehe unten).

**[0122]** In Arbeitsschritt 704 wird ein erster Schlüssel (Schlüssel1), der auf und/oder mit dem Speichermedium gespeichert ist, abgerufen. Diese Prozedur kann Abrufen des ersten Schlüssels von dem Medium, auf dem die Daten gespeichert sind, von einem Speicher, der physisch mit dem Medium verbunden ist, wie beispielsweise einem Kassettenspeicher, und dergleichen umfassen.

**[0123]** Das Speichermedium kann beliebiger vorliegend offenbarer Art sein, z.B. Magnetband, Magnetplatte, NVRAM usw. Dementsprechend kann es sich bei der Einheit um eine beliebige Art von Datenspeichereinheit handeln, z.B. Bandlaufwerk, SSD, HDD,

verschlüsselungsfähige USB-Datenträger, NVRAM-Module, usw.

**[0124]** Wie vorstehend erwähnt, ist gemäß einigen Ansätzen der erste Schlüssel auf und/oder mit dem Speichermedium in unverschlüsselter Form gespeichert. Gemäß anderen Ansätzen ist der erste Schlüssel auf und/oder mit dem Speichermedium in verpackter Form gespeichert (z.B. verschleierte Form, verschlüsselte Form, passwortgeschützte Form usw.). Informationen wie ein weiterer Schlüssel oder eine PIN, ein Kennwort usw. zum Entpacken des ersten Schlüssels können empfangen und zum Entpacken des ersten Schlüssels verwendet werden.

**[0125]** Gemäß bevorzugten Aspekten ist die Einheit so konfiguriert, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet. Eine innerhalb der Einheit befindliche Steuereinheit kann jedoch auf den ersten Schlüssel von dem Speichermedium oder einer anderen mit dem Speichermedium verbundenen Quelle aus zugreifen.

**[0126]** In Arbeitsschritt 706 wird ein zweiter Schlüssel (Schlüssel2) von einer externen Quelle empfangen. Bei der externen Quelle kann es sich um eine beliebige externe Quelle handeln. Beispielsweise kann es sich bei der externen Quelle um den Anfrager der Daten, einen Schlüsselspeicher, einen Schlüsselserver oder eine Schlüsseleingabe durch einen Benutzer handeln (z.B. über eine Tastatur usw.).

**[0127]** In Arbeitsschritt 708 wird unter Verwendung des ersten und des zweiten Schlüssels ein Medienchiffrierschlüssel (MEK) erzeugt. Es kann eine beliebige bekannte Methode verwendet werden, um aus dem ersten und dem zweiten Schlüssel einen MEK zu erzeugen.

**[0128]** In Arbeitsschritt 710 werden die angefragten Daten unter Verwendung des Medienchiffrierschlüssels in chiffrierter Form auf das Speichermedium geschrieben, z.B. mit herkömmlichen Methoden.

**[0129]** Aspekte der verschiedenen Arbeitsschritte und Komponenten aus **Fig. 7** ähneln bevorzugt denen, die vorstehend im Abschnitt, der **Fig. 5** beschreibt, beschrieben wurden, können aber auch Merkmale aus jedem anderen vorliegenden Abschnitt enthalten. Diese Aspekte werden lediglich beispielhaft dargestellt und sind nicht als Einschränkung gedacht. Darüber hinaus können diese Aspekte entsprechend der Vielzahl möglicher Ansätze der vorliegenden Erfindung in beliebiger Weise kombiniert werden.

**[0130]** Gemäß verschiedenen Ansätzen enthalten Systeme eine Einheit, die so konfiguriert ist, dass sie Datenoperationen an einem Speichermedium

durchführt, wobei die Einheit über einen Prozessor und mit dem Prozessor integrierte, durch den Prozessor ausführbare oder mit dem Prozessor integrierte und durch den Prozessor ausführbare Logik verfügt, wobei die Logik so konfiguriert ist, dass sie die Einheit veranlasst, einige oder alle der vorstehenden Arbeitsschritte, z.B. die Arbeitsschritte aus **Fig. 4** bis **Fig. 7**, durchzuführen.

**[0131]** Gemäß verschiedenen Ansätzen enthalten Computerprogrammprodukte zum Ermöglichen und/oder Durchführen von Krypto-Löschung ein durch einen Computer lesbares Speichermedium mit darauf enthaltenen Programmanweisungen, wobei die Programmanweisungen von einer Einheit ausführbar sind, die so konfiguriert ist, dass sie einige oder alle der vorstehenden Arbeitsschritte, z.B. die Arbeitsschritte aus **Fig. 4** bis **Fig. 7**, durchführt.

#### Beispielhafte Implementierungen

**[0132]** Ein beispielhafter Prozess zum Schutz einer verschlüsselungsfähigen Speichereinheit, beispielsweise einer selbstverschlüsselnden Einheit (Self-Encrypting Device, SED), umfasst gemäß einem Ansatz die folgenden Arbeitsschritte:

1. Erzeugen eines Schlüssels1 und Speichern dieses Schlüssels1 innerhalb der Speichereinheit auf eine sichere Weise innerhalb der Speichereinheit. Normalerweise würde Schlüssel1 niemals in Klartext im nichtflüchtigen Speicher gespeichert werden.
2. Erzeugen eines Schlüssel2 und Speichern dieses Schlüssel2 extern auf der Speichereinheit, und:
  - A. Wenn Schlüssel2 intern auf der Speichereinheit erstellt wurde, Abrufen von Schlüssel2 aus der Speichereinheit
  - B. Wenn Schlüssel2 außerhalb der Speichereinheit erstellt wurde, Schlüssel2 bei Bedarf der Speichereinheit zuführen
3. Die Speichereinheit verwendet Schlüssel1 und Schlüssel2, um einen Medienchiffrierschlüssel (MEK) zu berechnen
4. Der MEK wird direkt oder indirekt verwendet, um Daten zu verschlüsseln, um chiffrierten Text zu erzeugen, und um chiffrierten Text zu entschlüsseln, um Klartext zu erzeugen.

**[0133]** Gemäß einem Ansatz umfasst ein veranschaulichender Prozess nach einem Stromausfall oder einem Kaltstart, um den Zugriff auf eine verschlüsselungsfähige Speichereinheit wie beispielsweise eine selbstverschlüsselnde Einheit (SED) wiederherzustellen, die folgenden Arbeitsschritte:

1. Schlüssel2 wird der Speichereinheit zugeführt
2. Zugreifen auf Schlüssel1 innerhalb der Speichereinheit, was kryptografisches Entpacken von Schlüssel1 beinhalten kann, wofür Schlüssel2 sowie ein oder mehrere Zugangsschlüssel (z.B. PIN) erforderlich sein können
3. Die Speichereinheit verwendet Schlüssel1 und Schlüssel2, um einen Medienchiffrierschlüssel (MEK) zu berechnen
4. Der MEK wird direkt oder indirekt verwendet, um Daten zu verschlüsseln, um chiffrierten Text zu erzeugen, und um chiffrierten Text zu entschlüsseln, um Klartext zu erzeugen.

Prozess zur Stilllegung einer Einheit oder zur Sicherung einer Gestohlenen Einheit

**[0134]** Ein veranschaulichender Prozess, der durchgeführt werden kann, wenn eine Speichereinheit stillgelegt werden soll, z.B. weil ihre Hintergrundfehler-rate zu groß geworden ist (was z.B. durch zu viele Schreibzyklen geschehen kann) oder vielleicht, weil es „ausgefallen“ ist und eine Operation nicht erfolgreich abgeschlossen hat), umfasst gemäß einem Ansatz die folgenden Arbeitsschritte:

1. Senden eines Befehls an die Speichereinheit, der diese anweist, ihren Schlüssel1 einer Krypto-Löschung zu unterziehen
2. Wurde die Krypto-Löschung von Schlüssel1 erfolgreich abgeschlossen? Wenn:
  - Ja, dann war die Krypto-Löschung der unter diesem MEK verschlüsselten Daten erfolgreich
  - in diesem Fall muss Schlüssel2 nicht zerstört werden, auch wenn dies ebenfalls möglich ist
  - Nein, dann kann nicht darauf vertraut werden, dass eine Krypto-Löschung stattgefunden hat:
  - Zerstören von Schlüssel2 außerhalb der Speichereinheit und damit Krypto-Löschung der Speichereinheit durch externe Mittel
  - Ausschalten oder Kaltstarten der Speichereinheit, um sicherzustellen, dass die Speichereinheit kein flüchtiges Abbild eines Schlüssels speichert.

**[0135]** Es ist zu beachten, dass eine Löschung von Schlüsseln auch in umgekehrter Reihenfolge erfolgen kann, wobei Schlüssel2 zuerst zerstört wird, bevorzugt mit einem anschließenden Versuch, Schlüssel1 zu löschen. Ein veranschaulichender Prozess umfasst gemäß einem Ansatz die folgenden Arbeitsschritte:

1. Zerstören von Schlüssel2 außerhalb der Speichereinheit und damit Krypto-Löschung der Speichereinheit durch externe Mittel

2. Senden eines Befehls an die Speichereinheit, der diese anweist, ihren Schlüssel1 einer Krypto-Löschung zu unterziehen

3. Wurde die Krypto-Löschung von Schlüssel1 erfolgreich abgeschlossen? Wenn:

- Ja, dann wurde an den unter diesem MEK verschlüsselten Daten durch die Zerstörung von Schlüssel1 und Schlüssel2 erfolgreich Krypto-Löschung durchgeführt
- Ausschalten oder Kaltstarten der Speichereinheit, um sicherzustellen, dass die Speichereinheit kein flüchtiges Abbild eines Schlüssels speichert
- Nein, aber Schlüssel2 außerhalb der Speichereinheit wurde zerstört, so dass die Krypto-Löschung der Speichereinheit mit externen Mitteln bewirkt wird.

**[0136]** Wird eine Speichereinheit gestohlen oder wurde keine Krypto-Löschung durchgeführt, bevor sie aus dem Speichersystem entfernt wurde, und geht dann verloren, z.B. auf dem Transportweg zu einem anderen Ort, so muss, da eine Krypto-Löschung der Daten durch Zerstörung von Schlüssel1 nicht mehr möglich ist, weil man keinen physischen Zugang zu der Speichereinheit hat, stattdessen Schlüssel2 zerstört werden, was auch ohne physischen Zugang zu der Speichereinheit möglich ist.

**[0137]** Fig. 8 zeigt ein Beispiel dafür, was derzeit ohne die Erfindung erreicht werden kann, während Fig. 9 und Fig. 10 Beispiele dafür zeigen, was mit einem oder mehreren Ansätzen der Erfindung operativ erreicht werden kann. Die Schaubilder in Fig. 8 bis Fig. 10 sind selbsterklärend und werden nur als Beispiel dargestellt.

**[0138]** Bei der vorliegenden Erfindung kann es sich um ein System, ein Verfahren und/oder ein Computerprogrammprodukt jedes möglichen technisch detaillierten Integrationsgrads handeln. Das Computerprogrammprodukt kann (ein) durch einen Computer lesbare(s) Speichermedium (oder -medien) umfassen, auf dem/denen durch einen Computer lesbare Programmanweisungen gespeichert ist/sind, um einen Prozessor dazu zu veranlassen, Aspekte der vorliegenden Erfindung auszuführen.

**[0139]** Bei dem durch einen Computer lesbaren Speichermedium kann es sich um eine physische Einheit handeln, die Anweisungen zur Verwendung durch eine Einheit zur Ausführung von Anweisungen behalten und speichern kann. Bei dem durch einen Computer lesbaren Speichermedium kann es sich

zum Beispiel um eine elektronische Speichereinheit, eine magnetische Speichereinheit, eine optische Speichereinheit, eine elektromagnetische Speichereinheit, eine Halbleiterspeichereinheit oder jede geeignete Kombination daraus handeln, ohne auf diese beschränkt zu sein. Zu einer nicht erschöpfenden Liste spezifischerer Beispiele des durch einen Computer lesbaren Speichermediums gehören die Folgenden: eine tragbare Computerdiskette, eine Festplatte, ein Direktzugriffsspeicher (RAM), ein Nur-Lese-Speicher (ROM), ein löschbarer programmierbarer Nur-Lese-Speicher (EPROM bzw. Flash-Speicher), ein statischer Direktzugriffsspeicher (SRAM), ein tragbarer Kompaktspeicherplatte-Nur-Lese-Speicher (CD-ROM), eine DVD (digital versatile disc), ein Speicher-Stick, eine Diskette, eine mechanisch codierte Einheit wie zum Beispiel Lochkarten oder gehobene Strukturen in einer Rille, auf denen Anweisungen gespeichert sind, und jede geeignete Kombination daraus. Ein durch einen Computer lesbares Speichermedium soll in der Verwendung hierin nicht als flüchtige Signale an sich aufgefasst werden, wie zum Beispiel Funkwellen oder andere sich frei ausbreitende elektromagnetische Wellen, elektromagnetische Wellen, die sich durch einen Wellenleiter oder ein anderes Übertragungsmedium ausbreiten (z.B. ein Lichtwellenleiterkabel durchlaufende Lichtimpulse) oder durch einen Draht übertragene elektrische Signale.

**[0140]** Hierin beschriebene, durch einen Computer lesbare Programmanweisungen können von einem durch einen Computer lesbaren Speichermedium auf jeweilige Datenverarbeitungs-/Verarbeitungseinheiten oder über ein Netzwerk wie zum Beispiel das Internet, ein lokales Netzwerk, ein Weitverkehrsnetz und/oder ein drahtloses Netzwerk auf einen externen Computer oder eine externe Speichereinheit heruntergeladen werden. Das Netzwerk kann Kupferübertragungskabel, Lichtwellenübertragungsleiter, drahtlose Übertragung, Leitwegrechner, Firewalls, Vermittlungseinheiten, Gateway-Computer und/oder Edge-Server aufweisen. Eine Netzwerkadapterkarte oder Netzwerkschnittstelle in jeder Datenverarbeitungs-/Verarbeitungseinheit empfängt durch einen Computer lesbare Programmanweisungen aus dem Netzwerk und leitet die durch einen Computer lesbaren Programmanweisungen zur Speicherung in einem durch einen Computer lesbaren Speichermedium innerhalb der entsprechenden Datenverarbeitungs-/Verarbeitungseinheit weiter.

**[0141]** Bei durch einen Computer lesbaren Programmanweisungen zum Ausführen von Arbeitsschritten der vorliegenden Erfindung kann es sich um Assembler-Anweisungen, ISA-Anweisungen (Instruction-Set-Architecture), Maschinenanweisungen, maschinenabhängige Anweisungen, Mikrocode, Firmware-Anweisungen, zustandssetzende Daten oder entweder Quellcode oder Objektcode

handeln, die in einer beliebigen Kombination aus einer oder mehreren Programmiersprachen geschrieben werden, darunter objektorientierte Programmiersprachen wie Smalltalk, C++ o.ä. sowie herkömmliche prozedurale Programmiersprachen wie die Programmiersprache „C“ oder ähnliche Programmiersprachen. Die durch einen Computer lesbaren Programmanweisungen können vollständig auf dem Computer des Benutzers, teilweise auf dem Computer des Benutzers, als eigenständiges Software-Paket, teilweise auf dem Computer des Benutzers und teilweise auf einem fernen Computer oder vollständig auf dem fernen Computer oder Server ausgeführt werden. In letzterem Fall kann der entfernt angeordnete Computer mit dem Computer des Benutzers durch eine beliebige Art Netzwerk verbunden sein, darunter ein lokales Netzwerk (LAN) oder ein Weitverkehrsnetz (WAN), oder die Verbindung kann mit einem externen Computer hergestellt werden (zum Beispiel über das Internet unter Verwendung eines Internet-Diensteanbieters). In einigen Ausführungsformen können elektronische Schaltungen, darunter zum Beispiel programmierbare Logikschaltungen, vor Ort programmierbare Gatter-Anordnungen (FPGA, field programmable gate arrays) oder programmierbare Logikanordnungen (PLA, programmable logic arrays) die durch einen Computer lesbaren Programmanweisungen ausführen, indem sie Zustandsinformationen der durch einen Computer lesbaren Programmanweisungen nutzen, um die elektronischen Schaltungen zu personalisieren, um Aspekte der vorliegenden Erfindung durchzuführen.

**[0142]** Aspekte der vorliegenden Erfindung sind hierin unter Bezugnahme auf Ablaufpläne und/oder Blockschaltbilder bzw. Schaubilder von Verfahren, Vorrichtungen (Systemen) und Computerprogrammprodukten gemäß Ausführungsformen der Erfindung beschrieben. Es wird darauf hingewiesen, dass jeder Block der Ablaufpläne und/oder der Blockschaltbilder sowie Kombinationen von Blöcken in den Ablaufplänen und/oder den Blockschaltbildern mittels durch einen Computer lesbarer Programmanweisungen ausgeführt werden können.

**[0143]** Diese durch einen Computer lesbaren Programmanweisungen können einem Prozessor eines Computers oder einer anderen programmierbaren Datenverarbeitungsvorrichtung bereitgestellt werden, um eine Maschine zu erzeugen, so dass die über den Prozessor des Computers bzw. der anderen programmierbaren Datenverarbeitungsvorrichtung ausgeführten Anweisungen ein Mittel zur Umsetzung der in dem Block bzw. den Blöcken der Ablaufpläne und/oder der Blockschaltbilder festgelegten Funktionen/Schritte erzeugen. Diese durch einen Computer lesbaren Programmanweisungen können auch auf einem durch einen Computer lesbaren Speichermedium gespeichert sein, das einen Computer, eine programmierbare Datenverarbei-

tungsvorrichtung und/oder andere Einheiten so steuern kann, dass sie auf eine bestimmte Art funktionieren, so dass das durch einen Computer lesbare Speichermedium, auf dem Anweisungen gespeichert sind, ein Herstellungsprodukt aufweist, darunter Anweisungen, welche Aspekte der/des in dem Block bzw. den Blöcken des Ablaufplans und/oder der Blockschaltbilder angegebenen Funktion/Schritts umsetzen.

**[0144]** Die durch einen Computer lesbaren Programmanweisungen können auch auf einen Computer, eine andere programmierbare Datenverarbeitungsvorrichtung oder eine andere Einheit geladen werden, um das Ausführen einer Reihe von Prozessschritten auf dem Computer bzw. der anderen programmierbaren Vorrichtung oder anderen Einheit zu verursachen, um einen auf einem Computer ausgeführten Prozess zu erzeugen, so dass die auf dem Computer, einer anderen programmierbaren Vorrichtung oder einer anderen Einheit ausgeführten Anweisungen die in dem Block bzw. den Blöcken der Ablaufpläne und/oder der Blockschaltbilder bzw. Schaubilder festgelegten Funktionen/Schritte umsetzen.

**[0145]** Die Ablaufpläne und die Blockschaltbilder bzw. Schaubilder in den Figuren veranschaulichen die Architektur, die Funktionalität und den Betrieb möglicher Ausführungen von Systemen, Verfahren und Computerprogrammprodukten gemäß verschiedenen Ausführungsformen der vorliegenden Erfindung. In diesem Zusammenhang kann jeder Block in den Ablaufplänen oder Blockschaltbildern bzw. Schaubildern ein Modul, ein Segment oder einen Teil von Anweisungen darstellen, die eine oder mehrere ausführbare Anweisungen zur Ausführung der bestimmten logischen Funktion(en) aufweisen. In einigen alternativen Ausführungen können die in den Blöcken angegebenen Funktionen in einer anderen Reihenfolge als in den Figuren gezeigt stattfinden. So können beispielsweise zwei nacheinander gezeigte Blöcke in Wirklichkeit in einem Schritt, gleichzeitig, im Wesentlichen gleichzeitig, teilweise oder vollständig zeitlich überlappend ausgeführt werden, oder die Blöcke können je nach der betreffenden Funktionalität auch in umgekehrter Reihenfolge ausgeführt werden. Es ist ferner anzumerken, dass jeder Block der Blockschaltbilder und/oder der Ablaufpläne sowie Kombinationen aus Blöcken in den Blockschaltbildern und/oder den Ablaufplänen durch spezielle auf Hardware beruhende Systeme umgesetzt werden können, welche die festgelegten Funktionen oder Schritte durchführen, oder Kombinationen aus Spezial-Hardware und Computeranweisungen ausführen.

**[0146]** Darüber hinaus kann ein System gemäß verschiedenen Ausführungsformen einen Prozessor und Logik enthalten, die mit dem Prozessor integriert

und/oder durch diesen ausführbar ist, wobei die Logik so konfiguriert ist, dass sie einen oder mehrere der vorliegend angegebenen Prozessschritte durchführt. Der Prozessor kann jede vorliegend beschriebene Konfiguration aufweisen, wie z.B. einen diskreten Prozessor oder eine Verarbeitungsschaltung, die viele Komponenten wie Verarbeitungshardware, Speicher, E/A-Schnittstellen usw. umfasst. Mit „integriert in“ ist gemeint, dass der Prozessor über eine in diesen eingebettete Logik als Hardware-Logik verfügt, beispielsweise eine anwendungsspezifische integrierte Schaltung (ASIC), ein FPGA usw. Mit „vom Prozessor ausführbar“ ist gemeint, dass es sich bei der Logik um Hardware-Logik, Software-Logik wie Firmware, einen Teil eines Betriebssystems, einen Teil eines Anwendungsprogramms usw. oder eine Kombination aus Hardware- und Software-Logik handelt, auf die der Prozessor zugreifen kann und die so konfiguriert ist, dass sie bei Ausführung durch den Prozessor den Prozessor veranlasst, eine bestimmte Funktionalität auszuführen. Software-Logik kann auf einem lokalen und/oder entfernt angeordneten Speicher jedes im Fachgebiet bekannten Speichertyps gespeichert sein. Es kann jeder im Fachgebiet bekannte Prozessor verwendet werden, beispielsweise ein Software-Prozessormodul und/oder ein Hardware-Prozessor wie beispielsweise eine ASIC, eine FPGA, eine zentrale Verarbeitungseinheit (CPU), eine integrierte Schaltung (IC), eine Grafikverarbeitungseinheit (GPU) usw.

**[0147]** Es ist klar, dass die verschiedenen Merkmale der vorstehenden Systeme und/oder Methodiken in beliebiger Weise kombiniert werden können, wodurch aus den vorstehend vorgestellten Beschreibungen eine Vielzahl an Kombinationen geschaffen wird.

**[0148]** Ferner ist zu beachten, dass Aspekte der vorliegenden Erfindung in Form eines im Auftrag eines Kunden eingesetzten Dienstes bereitgestellt werden können, um einen Dienst nach Bedarf bereitzustellen.

**[0149]** Die Beschreibungen der verschiedenen Aspekte der vorliegenden Erfindung wurden zu Veranschaulichungszwecken dargelegt, sind jedoch nicht als abschließend oder auf die Ansätze beschränkt zu verstehen. Für den Fachmann sind viele Abwandlungen und Variationen ersichtlich, ohne den Umfang der beschriebenen Aspekte zu verlassen. Die hierin verwendete Terminologie wurde gewählt, um bestmöglich die Grundgedanken der verschiedenen Ansätze, der praktischen Anwendung oder technischen Verbesserung gegenüber den auf dem Markt erhältlichen Technologien zu erklären oder um dem Fachmann das Verständnis der vorliegend offenbarten Ansätze zu ermöglichen.

## Patentansprüche

1. Durch eine Einheit implementiertes Verfahren, umfassend:  
Empfangen einer Anfrage zum Lesen von Daten an einer Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind;  
Zugreifen auf einen innerhalb der Einheit gespeicherten ersten Schlüssel;  
Empfangen eines zweiten Schlüssels von einer externen Quelle;  
Erzeugen eines Medienchiffrierschlüssels unter Verwendung des ersten und des zweiten Schlüssels;  
Lesen der verschlüsselten Form der Daten aus dem Speichermedium;  
Entschlüsseln der verschlüsselten Form der Daten unter Verwendung des Medienchiffrierschlüssels;  
und  
Ausgeben der entschlüsselten Daten.
2. Durch eine Einheit implementiertes Verfahren nach Anspruch 1, wobei der erste Schlüssel in der Einheit in Rohform gespeichert wird.
3. Durch eine Einheit implementiertes Verfahren nach Anspruch 1, wobei der erste Schlüssel in der Einheit in verpackter Form gespeichert wird und das Verfahren Empfangen von Informationen zum Entpacken des ersten Schlüssels umfasst.
4. Durch eine Einheit implementiertes Verfahren nach Anspruch 1, wobei die Einheit so konfiguriert ist, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet.
5. Durch eine Einheit implementiertes Verfahren nach Anspruch 1, wobei das Speichermedium einen nichtflüchtigen Speicher enthält.
6. Durch eine Einheit implementiertes Verfahren nach Anspruch 5, wobei der erste Schlüssel in der Einheit in einem nichtflüchtigen Speicher gespeichert wird, bei dem es sich um einen anderen Speichertyp als das Speichermedium handelt.
7. Durch eine Einheit implementiertes Verfahren nach Anspruch 1, wobei es sich bei der Einheit um ein Bandlaufwerk handelt, wobei es sich bei dem Speichermedium um ein magnetisches Aufzeichnungsband handelt.
8. Durch eine Einheit implementiertes Verfahren nach Anspruch 1, umfassend Empfangen zweiter Daten zur Speicherung auf dem Speichermedium, Empfangen eines dritten Schlüssels von einer zweiten externen Quelle, Erzeugen eines zweiten Medienchiffrierschlüssels unter Verwendung des ersten und dritten Schlüssels, Verschlüsseln der

Daten unter Verwendung des zweiten Medienchiffrierschlüssels und Schreiben der verschlüsselten Daten auf das Speichermedium.

9. Computerprogrammprodukt zum Ermöglichen und/oder Durchführen von Krypto-Löschung, wobei das Computerprogrammprodukt ein durch einen Computer lesbares Speichermedium mit darauf enthaltenen Programmanweisungen aufweist, wobei die Programmanweisungen durch eine Einheit ausführbar sind, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, um die Einheit zu veranlassen, das Verfahren nach Anspruch 1 durchzuführen.

10. System, das Folgendes aufweist: eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Einheit einen Prozessor und mit dem Prozessor integrierte, durch den Prozessor ausführbare oder mit dem Prozessor integrierte und durch den Prozessor ausführbare Logik aufweist, wobei die Logik so konfiguriert ist, dass sie die Einheit veranlasst, das Verfahren nach Anspruch 1 durchzuführen.

11. Durch eine Einheit implementiertes Verfahren, umfassend:  
Empfangen einer Anfrage zum Lesen von Daten an einer Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Daten in verschlüsselter Form auf dem Speichermedium gespeichert sind;  
Abrufen eines auf und/oder mit dem Speichermedium gespeicherten ersten Schlüssels;  
Empfangen eines zweiten Schlüssels von einer externen Quelle;  
Erzeugen eines Medienchiffrierschlüssels unter Verwendung des ersten und des zweiten Schlüssels;  
Lesen der verschlüsselten Form der Daten aus dem Speichermedium;  
Entschlüsseln der verschlüsselten Form der Daten unter Verwendung des Medienchiffrierschlüssels;  
und  
Ausgeben der entschlüsselten Daten.

12. Durch eine Einheit implementiertes Verfahren nach Anspruch 11, wobei das Speichermedium einen nichtflüchtigen Speicher enthält.

13. Durch eine Einheit implementiertes Verfahren nach Anspruch 12, wobei der erste Schlüssel in der Einheit in einem nichtflüchtigen Speicher gespeichert wird, bei dem es sich um einen anderen Speichertyp als das Speichermedium handelt.

14. Durch eine Einheit implementiertes Verfahren nach Anspruch 11, wobei es sich bei der Einheit um ein Bandlaufwerk handelt, wobei es sich bei dem

Speichermedium um ein magnetisches Aufzeichnungsband handelt.

15. Durch eine Einheit implementiertes Verfahren nach Anspruch 11, wobei die Einheit so konfiguriert ist, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet.

16. Durch eine Einheit implementiertes Verfahren nach Anspruch 11, wobei der erste Schlüssel auf und/oder mit dem Speichermedium in Rohform gespeichert wird.

17. Durch eine Einheit implementiertes Verfahren nach Anspruch 11, wobei der erste Schlüssel auf und/oder mit dem Speichermedium in verpackter Form gespeichert wird, und das Verfahren Empfangen von Informationen zum Entpacken des ersten Schlüssels umfasst.

18. Durch eine Einheit implementiertes Verfahren nach Anspruch 11, umfassend Empfangen zweiter Daten zur Speicherung auf dem Speichermedium, Empfangen eines dritten Schlüssels von einer zweiten externen Quelle, Erzeugen eines zweiten Medienchiffrierschlüssels unter Verwendung des ersten und dritten Schlüssels, Verschlüsseln der Daten unter Verwendung des zweiten Medienchiffrierschlüssels und Schreiben der verschlüsselten Daten auf das Speichermedium.

19. System, das Folgendes aufweist: eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Einheit einen Prozessor und mit dem Prozessor integrierte, durch den Prozessor ausführbare oder mit dem Prozessor integrierte und durch den Prozessor ausführbare Logik aufweist, wobei die Logik so konfiguriert ist, dass sie das Verfahren nach Anspruch 11 durchführt.

20. Durch eine Einheit implementiertes Verfahren, umfassend:  
Empfangen einer Anfrage zum Schreiben von Daten in verschlüsselter Form auf das Speichermedium an einer Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt;  
Zugreifen auf einen innerhalb der Einheit gespeicherten ersten Schlüssel;  
Empfangen eines zweiten Schlüssels von einer externen Quelle;  
Erzeugen eines Medienchiffrierschlüssels unter Verwendung des ersten und des zweiten Schlüssels;  
und  
Schreiben der angefragten Daten in verschlüsselter Form auf das Speichermedium unter Verwendung des Medienchiffrierschlüssels.

21. Durch eine Einheit implementiertes Verfahren nach Anspruch 20, wobei die Einheit so konfiguriert ist, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet.

22. System, das Folgendes aufweist:  
eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Einheit einen Prozessor und mit dem Prozessor integrierte, durch den Prozessor ausführbare oder mit dem Prozessor integrierte und durch den Prozessor ausführbare Logik aufweist, wobei die Logik so konfiguriert ist, dass sie die Einheit veranlasst, das Verfahren nach Anspruch 20 durchzuführen.

23. Durch eine Einheit implementiertes Verfahren, umfassend:  
Empfangen einer Anfrage zum Schreiben von Daten in verschlüsselter Form auf das Speichermedium an einer Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt;  
Abrufen eines auf und/oder mit dem Speichermedium gespeicherten ersten Schlüssels;  
Empfangen eines zweiten Schlüssels von einer externen Quelle;  
Erzeugen eines Medienchiffrierschlüssels unter Verwendung des ersten und des zweiten Schlüssels;  
und  
Schreiben der angefragten Daten in verschlüsselter Form auf das Speichermedium unter Verwendung des Medienchiffrierschlüssels.

24. Durch eine Einheit implementiertes Verfahren nach Anspruch 23, wobei die Einheit so konfiguriert ist, dass sie eine Übertragung des ersten Schlüssels nach außerhalb der Einheit verbietet.

25. System, das Folgendes aufweist:  
eine Einheit, die so konfiguriert ist, dass sie an einem Speichermedium Datenoperationen durchführt, wobei die Einheit einen Prozessor und mit dem Prozessor integrierte, durch den Prozessor ausführbare oder mit dem Prozessor integrierte und durch den Prozessor ausführbare Logik aufweist, wobei die Logik so konfiguriert ist, dass sie die Einheit veranlasst, das Verfahren nach Anspruch 23 durchzuführen.

Es folgen 10 Seiten Zeichnungen

## Anhängende Zeichnungen

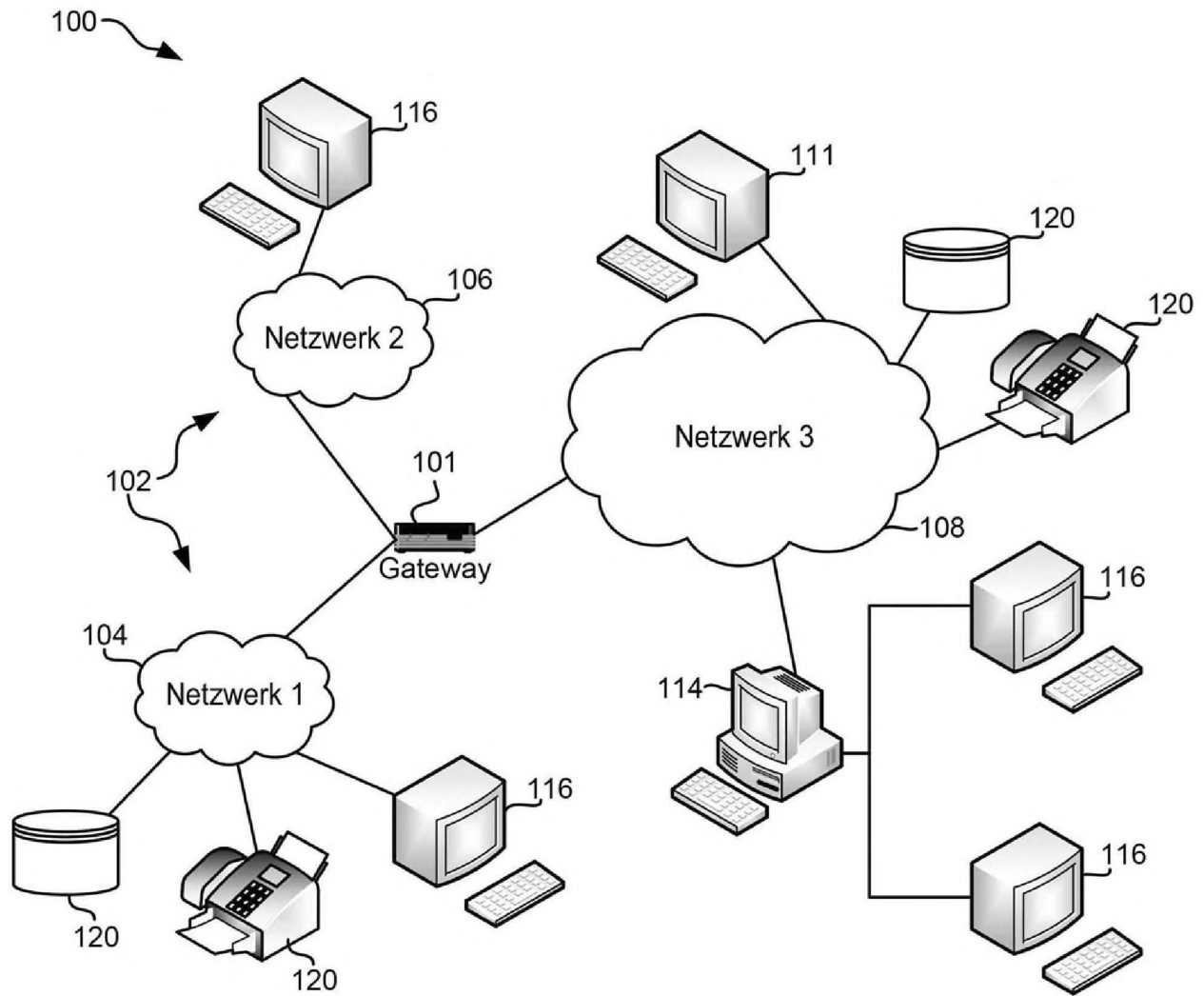
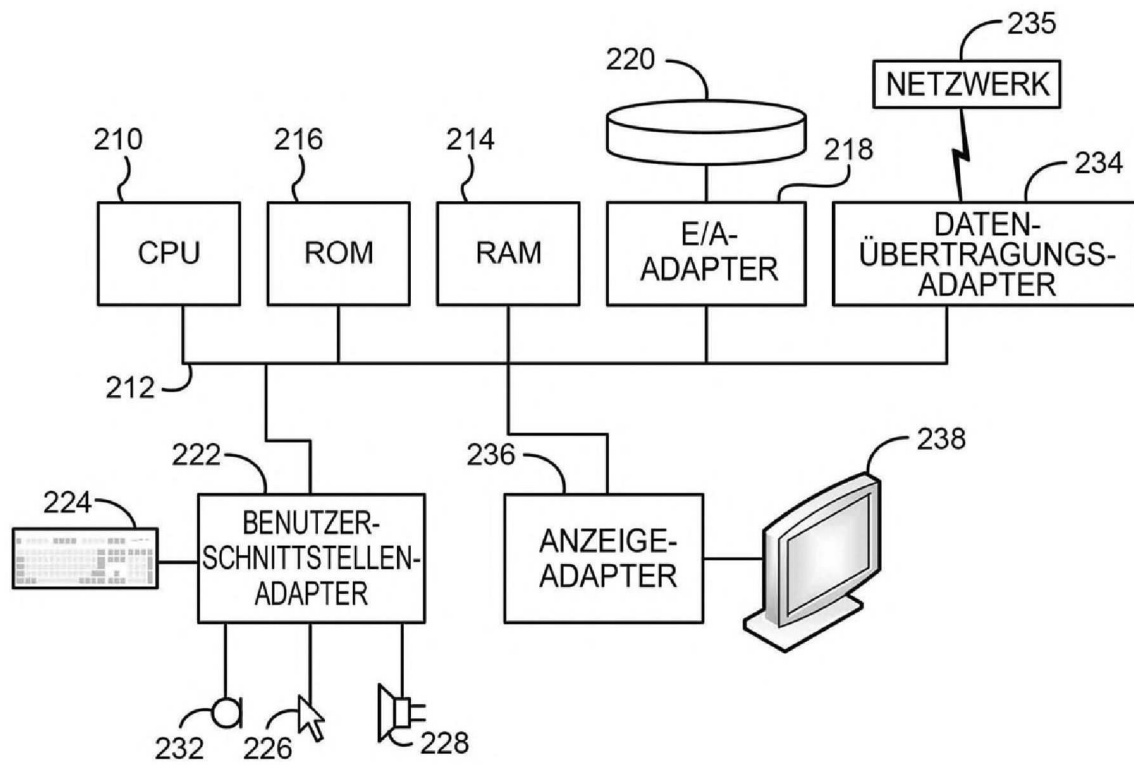
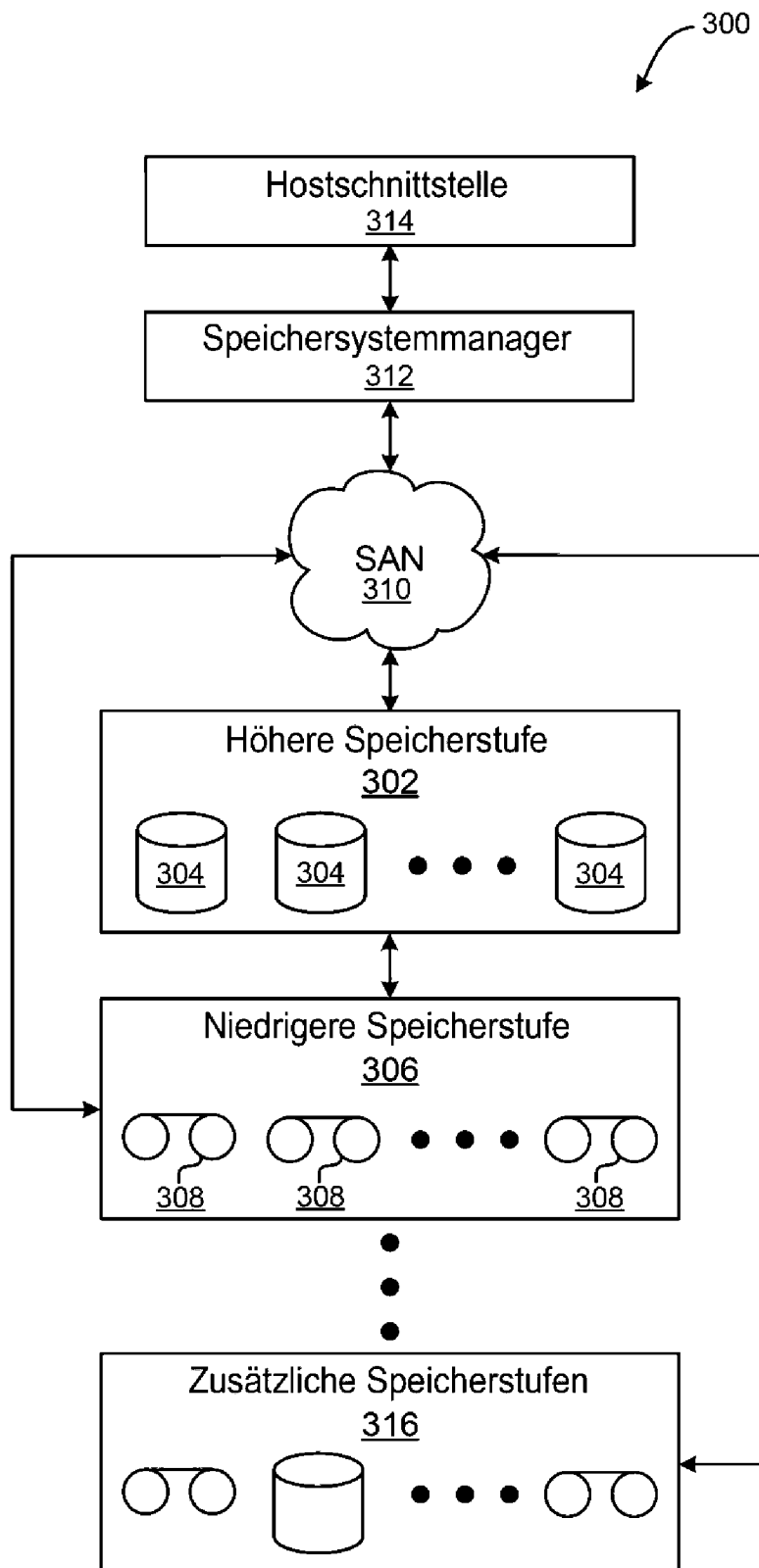


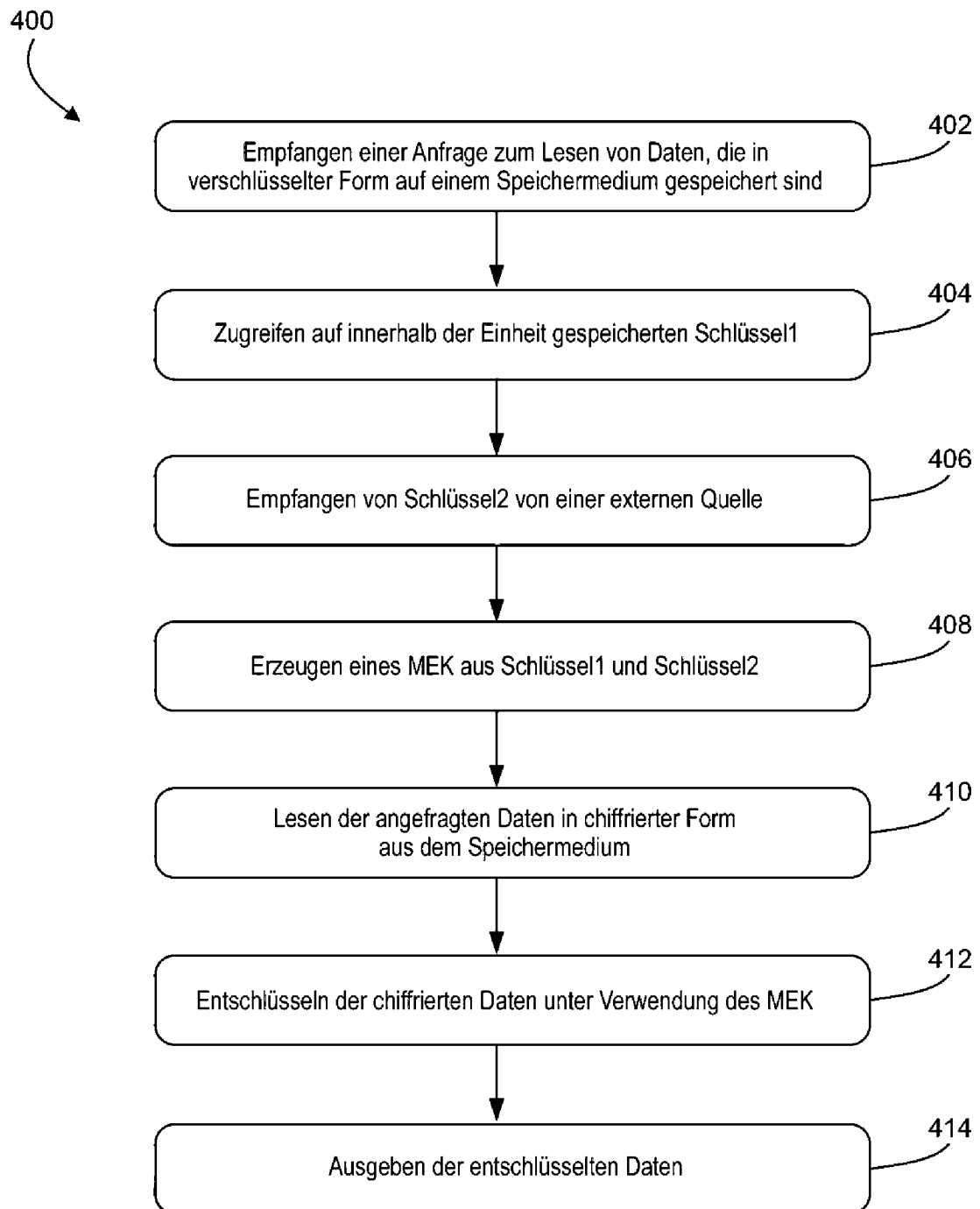
FIG. 1

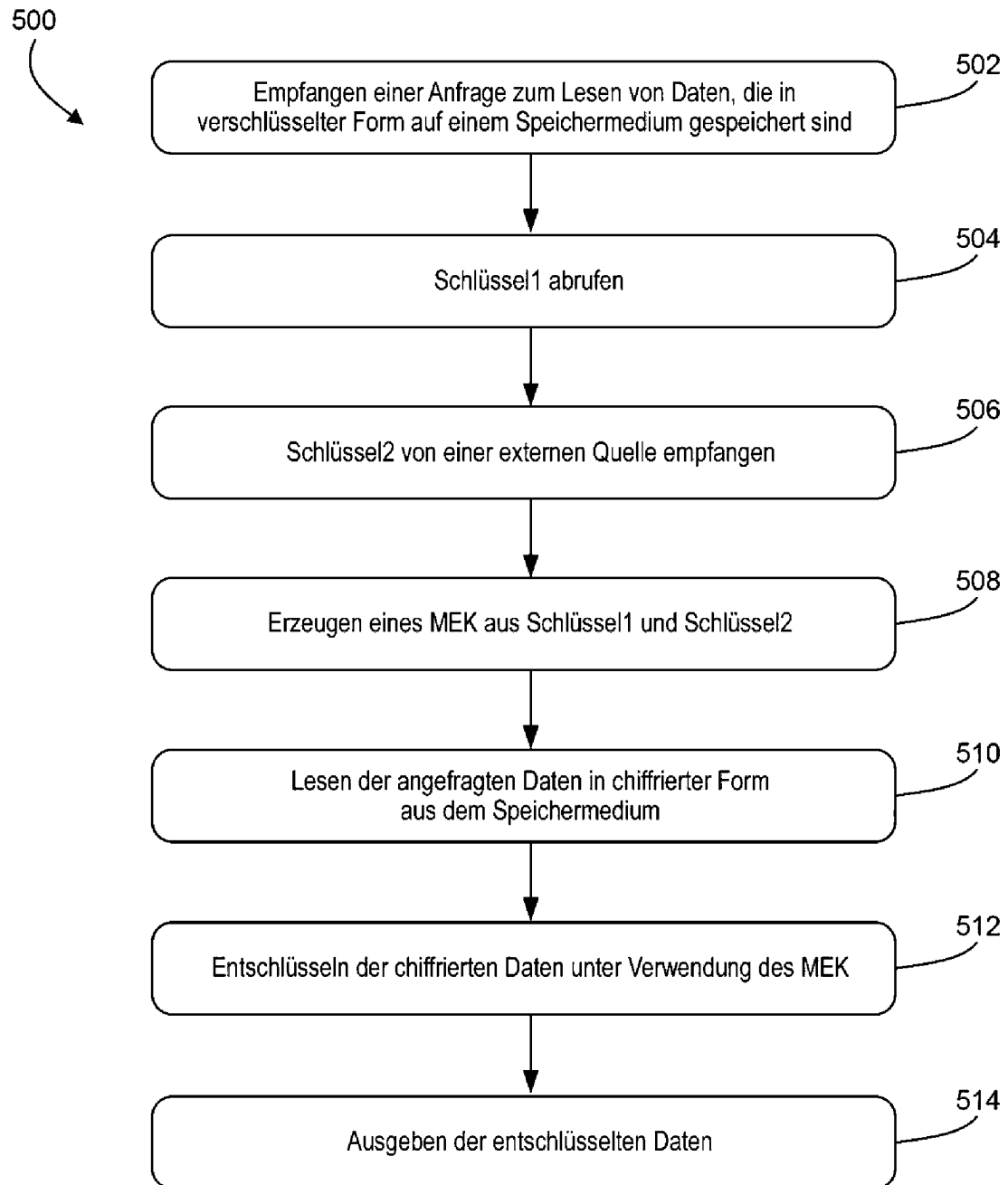


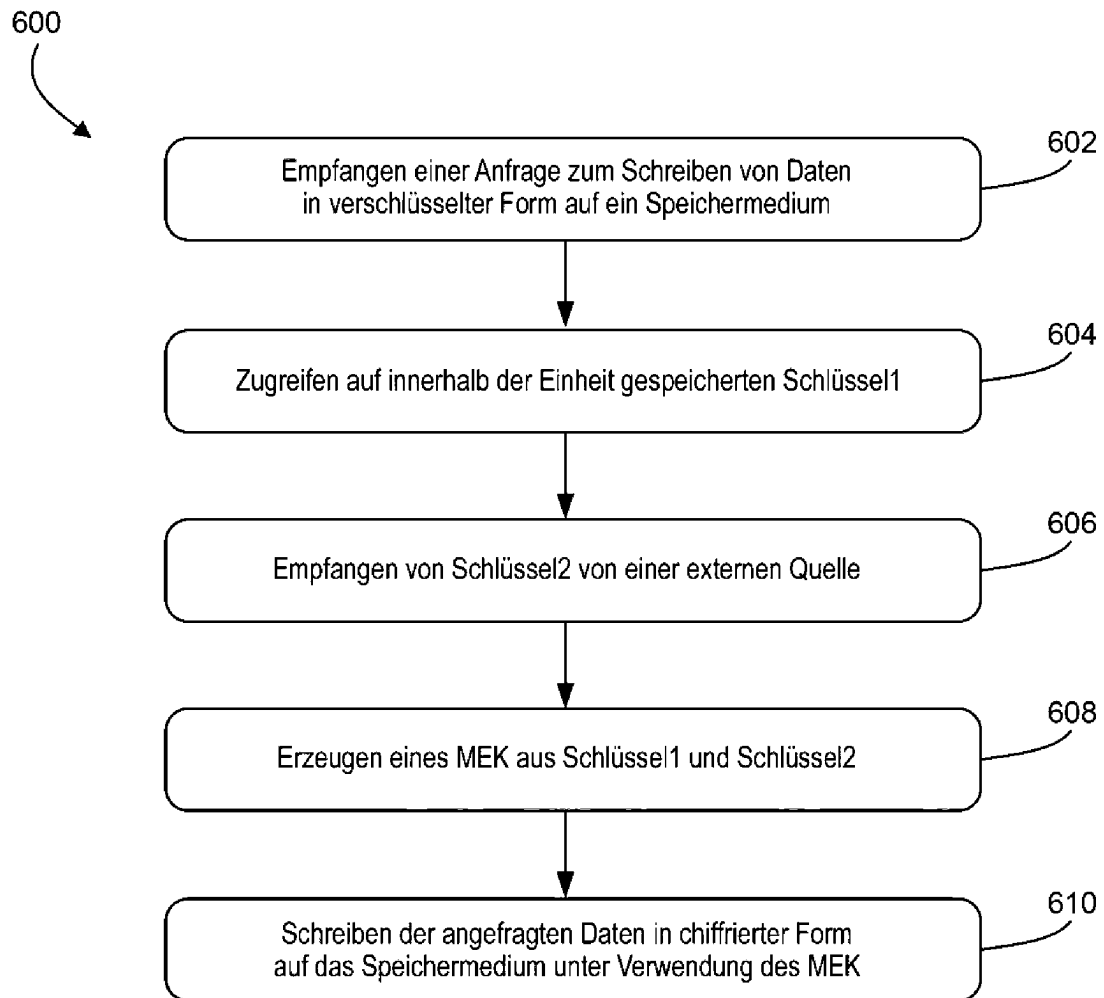
**FIG. 2**

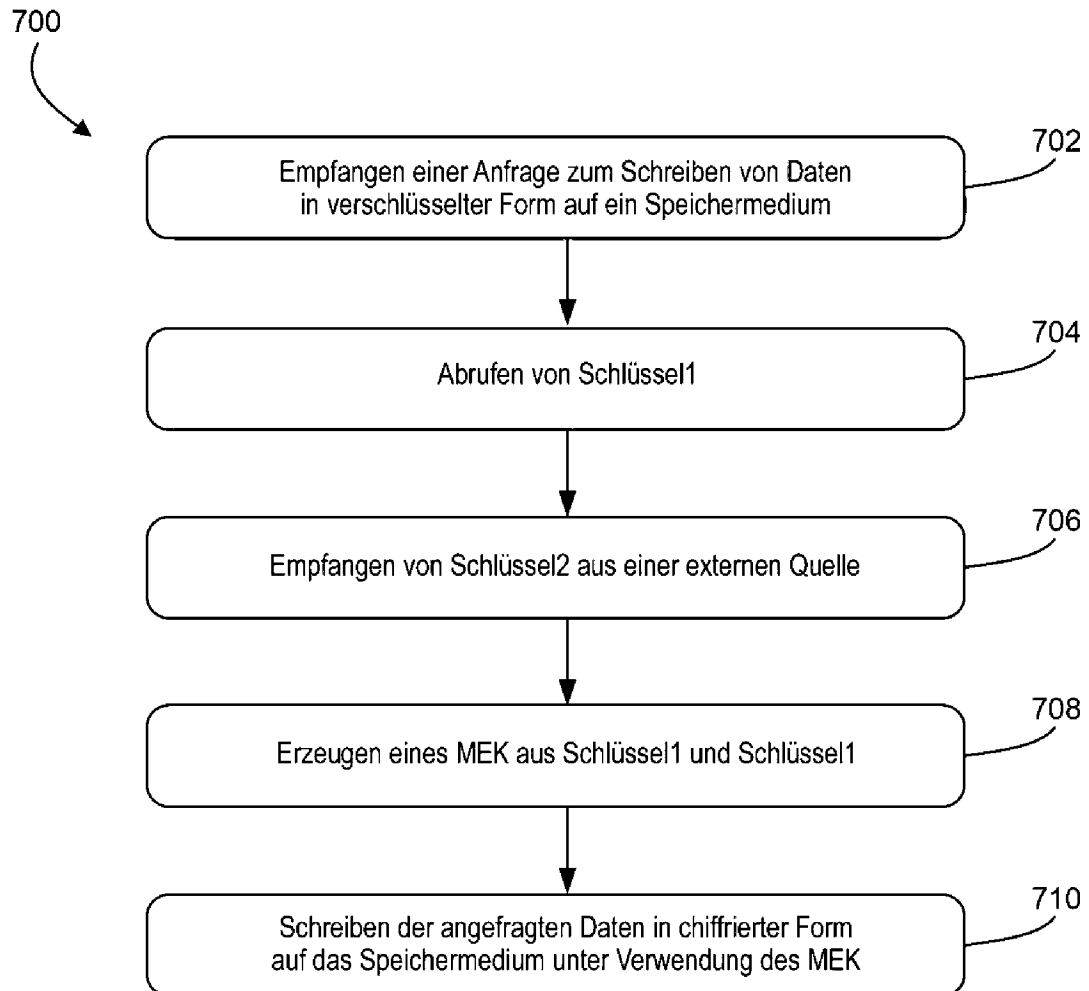


**FIG. 3**

**FIG. 4**

**FIG. 5**

**FIG. 6**

**FIG. 7**

| Art des Problems   | Verschlüsselung auf Systemebene aktiviert? | Was ist als Erstes zu tun?                 | Formatieren erfolgreich? | Nach Formatieren sicher? | Was ist als Zweites zu tun? | Ist das SED anschließend sicher? |
|--------------------|--------------------------------------------|--------------------------------------------|--------------------------|--------------------------|-----------------------------|----------------------------------|
| Gestohlenes SED *3 | Ja                                         | Es kann nichts getan werden                | Nicht anwendbar          | Nicht anwendbar          | Es kann nichts getan werden | Teilweise *1                     |
| Ausgefallenes SED  | Ja                                         | Formatieren einschließlich Krypto-Löschung | Nein *2                  | Teilweise *1             | Es kann nichts getan werden | Teilweise *1                     |
|                    |                                            |                                            | Ja                       | Ja                       | Dem Stillleger übergeben    | Ja *4                            |

**LEGENDE:**

- \*1 Die Sicherheit des SED hängt davon ab, dass der Angreifer niemals Zugriff auf den Verpackungsschlüssel erlangt und niemals in der Lage ist, die zum Verpacken des Schlüssels verwendete Kryptografie zu brechen
- \*2 Formatierung sollte erfolgreich sein, sofern nicht das SED in einer Weise ausgefallen ist, die es daran hindert, Befehle zu empfangen oder mit positivem Status zu antworten
- \*3 Wenn ein ausgefallenes SED nicht formatiert war und auf dem Weg zu einem Stillleger verloren ging (z.B. einem Hersteller oder jemand anderem, der das Medium/Laufwerk stilllegt (löscht oder zerstört)), besteht im Wesentlichen dasselbe Problem wie bei Diebstahl
- \*4 Das SED ist sicher, da der verpackte MEK überschrieben wurde und somit nicht wiederherstellbar ist, unabhängig davon, ob jemand auch Zugriff auf den Verpackungsschlüssel hat

**FIG. 8**

| Art des Problems   | Verschlüsselung auf Systemebene aktiviert? | Was ist als Erstes zu tun?                 | Formatieren erfolgreich? | Nach Formatieren sicher? | Was ist als Zweites zu tun? | Ist das SED anschließend sicher? |
|--------------------|--------------------------------------------|--------------------------------------------|--------------------------|--------------------------|-----------------------------|----------------------------------|
| Gestohlenes SED *3 | Ja                                         | Zum nächsten Schritt gehen                 | Nicht anwendbar          | Nicht anwendbar          | Schlüssel2 zerstören        | Ja *5                            |
| Ausgefallenes SED  | Ja                                         | Formatieren einschließlich Krypto-Löschung | Nein *2                  | Teilweise *1             | Schlüssel2 zerstören        | Ja *5                            |
|                    |                                            |                                            | Ja                       | Ja                       | Dem Stillleger übergeben    | Ja *4                            |

LEGENDE:

- \*1 Die Sicherheit des SED hängt davon ab, dass der Angreifer niemals Zugriff auf den Verpackungsschlüssel erlangt und niemals in der Lage ist, die zum Verpacken des Schlüssels verwendete Kryptografie zu brechen
- \*2 Formatierung sollte erfolgreich sein, sofern nicht das SED in einer Weise ausgefallen ist, die es daran hindert, Befehle zu empfangen oder mit positivem Status zu antworten
- \*3 Wenn ein ausgefallenes SED nicht formatiert war und auf dem Weg zu einem Stillleger verloren ging, besteht im Wesentlichen dasselbe Problem wie bei Diebstahl
- \*4 Das SED ist sicher, da der verpackte MEK überschrieben wurde und somit nicht wiederherstellbar ist, unabhängig davon, ob jemand auch Zugriff auf den Verpackungsschlüssel hat
- \*5 Das SED ist sicher, da Schlüssel2 zerstört wurde

**FIG. 9**

| Art des Problems   | Verschlüsselung auf Systemebene aktiviert? | Was ist als Erstes zu tun? | Was ist als Zweites zu tun?                | Formatieren erfolgreich? | Nach Formatieren sicher? | Ist das SED anschließend sicher? |
|--------------------|--------------------------------------------|----------------------------|--------------------------------------------|--------------------------|--------------------------|----------------------------------|
| Gestohlenes SED *3 | Ja                                         | Schlüssel2 zerstören       | Zum nächsten Schritt gehen                 | Nicht anwendbar          | Nicht anwendbar          | Ja *5                            |
| Ausgefallenes SED  | Ja                                         | Schlüssel2 zerstören       | Formatieren einschließlich Krypto-Löschung | Nein *2                  | Teilweise *1             | Ja *5                            |
|                    |                                            |                            |                                            | Ja                       | Ja                       | Ja *4                            |

**LEGENDE:**

- \*1 Die Sicherheit des SED hängt davon ab, dass der Angreifer niemals Zugriff auf den Verpackungsschlüssel erlangt und niemals in der Lage ist, die zum Verpacken des Schlüssels verwendete Kryptografie zu brechen
- \*2 Formatierung sollte erfolgreich sein, sofern nicht das SED in einer Weise ausgefallen ist, die es daran hindert, Befehle zu empfangen oder mit positivem Status zu antworten
- \*3 Wenn ein ausgefallenes SED nicht formatiert war und auf dem Weg zu einem Stillleger verloren ging, besteht im Wesentlichen dasselbe Problem wie bei Diebstahl
- \*4 Das SED ist sicher, da der verpackte MEK überschrieben wurde und somit nicht wiederherstellbar ist, unabhängig davon, ob jemand auch Zugriff auf den Verpackungsschlüssel hat
- \*5 Das SED ist sicher, da Schlüssel2 zerstört wurde

**FIG. 10**