Verfahren zur quantensicheren Signierung und Verifikation digitaler Zahlungstransaktionen unter Verwendung von QRNG-basierten Entropiequellen und post-quanten-kryptografischer Signaturtechnik für POS-Terminals, Karten- und Mobile-Payment-Systeme

Onovera™ TX-Framework

Kapitel 1. Einleitung und Zielsetzung

In der heutigen digitalen Infrastruktur basieren nahezu alle elektronischen Transaktionen – insbesondere im Zahlungsverkehr – auf kryptographischen Verfahren, die auf mathematischen Problemen wie der Faktorisierung großer Primzahlen (RSA) oder diskreten Logarithmen (ECDSA) beruhen. Diese Verfahren gelten im Kontext klassischer Rechenleistung als sicher und sind über Jahrzehnte hinweg als Standard implementiert worden.

Mit dem fortschreitenden Entwicklungsstand von Quantencomputern droht jedoch ein fundamentaler Umbruch. Algorithmen wie Shor und Grover versprechen, die grundlegenden Sicherheitsannahmen hinter aktuellen kryptographischen Verfahren aufzulösen. Bereits jetzt wird in wissenschaftlichen und industriellen Kreisen vom sogenannten "Harvest-Now-Decrypt-Later" -Szenario gesprochen: Angreifer speichern heute verschlüsselte Daten – einschließlich Transaktionspakete – mit dem Ziel, diese zu einem späteren Zeitpunkt mithilfe von Quantencomputern entschlüsseln zu können.

Diese Bedrohung macht es notwendig, schon jetzt Strukturen zu entwickeln, die künftigen Angriffen standhalten – selbst dann, wenn Quantencomputer noch nicht flächendeckend einsatzfähig sind. Besonders im Zahlungsverkehr, bei dem täglich Milliarden von Transaktionen durchgeführt werden, ist ein frühzeitiger Umstieg auf post-quanten-sichere Verfahren (PQC) nicht nur sicherheitsrelevant, sondern wirtschaftlich und regulatorisch essenziell.

Ziel der Erfindung

Die hier beschriebene Erfindung, vorläufig bezeichnet als Onovera TX-Framework -Verfahren, verfolgt das Ziel, ein robustes, skalierbares und zukunftsfähiges Verfahren zur Signatur und Verifikation von digitalen Zahlungstransaktionen bereitzustellen. Es vereint verschiedene Technologien, die einzeln bekannt sein mögen, aber in dieser spezifischen und neuartigen Kombination sowie mit dem Ziel der quantensicheren Absicherung von Zahlungsvorgängen bisher nicht patentiert oder implementiert wurden.

Konkret basiert das Verfahren auf drei Säulen:

1. Verwendung eines echten quantenphysikalischen Zufallszahlengenerators (QRNG) zur Erzeugung nicht deterministischer Schlüsselmaterialien oder Transaktionswerte.

2. Kombination dieser Werte mit transaktionsspezifischen Parametern (z. B. Betrag, Zeit, Empfängerkennung, u.a) mittels eines kollisionsresistente Verfahren (z. B. Hashing wie SHA3, MACs oder andere methoden).
3. Digitale Signatur des resultierenden Integritäts Wertes durch einen post-quantenresistenten Signaturalgorithmus (z. B. CRYSTALS-Dilithium) und anschließende Validierung durch eine autorisierte Stelle (Bank, Zahlungsdienstleister, Clearingstelle etc.).

Warum jetzt handeln?

Der proaktive Umstieg auf ein quantensicheres Signaturverfahren ist keine akademische Übung, sondern eine realistische Vorbereitung auf die nahende Transformation der IT-Sicherheitslandschaft. Ein frühzeitiger Einsatz dieses Verfahrens bietet folgende Vorteile:

Langzeitschutz für gespeicherte Transaktionsdaten (auch gegen spätere Angriffe)
Vermeidung aufwendiger Migrationen, wenn der Quantenbruch tatsächlich erfolgt
Regulatorische Compliance: Vorbereitung auf kommende gesetzliche Anforderungen
Technologische Führerschaft für Anbieter, die ein zukunftsfähiges Zahlungssystem anbieten wollen

Ziel dieses Patents ist es daher, ein Verfahren zu schützen, das die Transaktionssicherheit auch in einer post-quanten-Kryptographie-Welt gewährleistet und sich gleichzeitig modular in bestehende Zahlungssysteme integrieren lässt.

Kapitel 2. Technischer Hintergrund und Stand der Technik

2.1. Der aktuelle kryptografische Standard im Zahlungsverkehr

Der heutige digitale Zahlungsverkehr – etwa mittels EC-Karten, Kreditkarten oder Mobile Payment – verwendet asymmetrische und symmetrische Verschlüsselungsverfahren, die auf klassischer Mathematik basieren. Dazu zählen unter anderem:

RSA (Rivest–Shamir–Adleman),
ECC (Elliptic Curve Cryptography),
SHA-2/SHA-3 Hashing für Integritätsprüfungen,
sowie Protokolle wie TLS zur sicheren Übertragung.

Diese Verfahren gelten gegenwärtig als sicher – allerdings nur gegenüber klassischen Computern. Die Sicherheit basiert auf Problemen wie der Faktorisierung großer Zahlen oder dem diskreten Logarithmus, die für klassische Rechner nur mit extrem hohem Aufwand lösbar sind.

2.2. Bedrohung durch Quantencomputer

Quantencomputer stellen für diese Verfahren ein ernstes Risiko dar:

Shor’s Algorithmus kann RSA- und ECC-Schlüssel in polynomieller Zeit brechen.
Grover’s Algorithmus halbiert die effektive Stärke von Hash-Funktionen.
Speicherung von heutigen verschlüsselten Daten („Harvest now, decrypt later“) ermöglicht künftiges rückwirkendes Knacken dieser Informationen, sobald leistungsstarke Quantencomputer verfügbar sind.

Zahlreiche Institutionen (z. B. NIST, BSI, ETSI) arbeiten daher bereits an der Standardisierung sogenannter Post-Quantum Cryptography (PQC) – Algorithmen, die auch gegen Angriffe durch Quantencomputer beständig sind.

2.3. Relevante Technologien im Stand der Technik

Es gibt bereits Patente und Forschungsvorhaben zu einzelnen der folgenden Technologien:

QRNG (Quantum Random Number Generators): Physikalische Module, die auf Quantenphänomenen basieren (z. B. Photoneninterferenzen), um echte Zufallszahlen zu generieren.
PQC-Signaturalgorithmen wie CRYSTALS-Dilithium, Falcon, SPHINCS+, die nachweislich gegen Quantenangriffe resistent sind.
Hash-basierte Transaktionsverschlüsselung: SHA3 ist ein aktueller Standard für Integritätsnachweise.

Allerdings existiert nach tiefgehender Recherche kein Patent oder kommerzielles System, das diese drei Kerntechnologien in der hier angestrebten Verfahrenskombination zur Absicherung von Zahlungstransaktionen nutzt oder geschützt hat. Bestehende Schutzrechte adressieren einzelne Aspekte (z. B. Blockchain, QRNG-basierte Authentifizierung), jedoch nicht in dieser Struktur und nicht mit dem spezifischen Fokus auf Zahlungsverkehr.

2.4. Differenzierung der Erfindung

Der Kernunterschied der vorliegenden Erfindung liegt in der:

spezifischen Verwendung einer mittels Quanten-Zufallszahlengenerierung (QRNG) erzeugten, einzigartigen Zufallskomponente zur transaktionsindividuellen Absicherung; der verknüpfenden Einbindung dieser Komponente mit Zahlungsparametern wie Betrag, Zeitstempel, Händler-ID und andere, in eine eindeutig identifizierende Transaktionssignaturbasis (M); sowie der anschließenden digitalen Signierung dieses Integritätswertes mittels postquantenkryptografischer Verfahren (PQC) und der Validierung durch ein autorisiertes Prüfsystem, etwa eine Bank oder einen Zahlungsdienstleister.

Dieses Verfahren schafft eine durchgängig geschützte Transaktionskette – einschließlich Absicherung gegen potenzielle Angriffe durch Quantencomputer – und ist modular aufgebaut, interoperabel und für bestehende Zahlungssysteme adaptierbar.

2.5. Definition der zentralen Notationen (für die folgenden Kapitel)

Zur besseren Lesbarkeit der folgenden Kapitel werden im Folgenden zentrale Variablen und Ausdrücke definiert, die das beschriebene Verfahren standardisiert abbilden:

- B = Betrag der Transaktion
- t = Zeitstempel der Transaktion
- H = Händlerkennung oder Empfänger-ID
- k_q = durch Quantum Random Number Generator (QRNG) erzeugter, echt zufälliger Wert
- M = Integritätswert, der aus einer kryptografischen Kombination der Transaktionsparameter (B, t, H) mit dem Zufallswert k_q erzeugt wird.
  Beispielhaft dargestellt durch:
  M = Hash(B, t, H, k_q)
  Alternativ kann M auch unter Verwendung anderer kollisionsresistenter Verfahren erzeugt werden, etwa durch MAC-Funktionen, symmetrische Verschlüsselung oder vergleichbare Techniken.
- σ = Digitale Signatur des Werts M mittels eines postquantenkryptografischen Verfahrens (PQC), z. B.
  σ = PQCsign(M)

  Hinweis: Die Verwendung eines Hash-Algorithmus zur Bildung von M dient lediglich der Veranschaulichung. Das Verfahren ist nicht auf Hashing beschränkt, sondern schließt jede kollisionsresistente Form der Kombination von k_q und Transaktionsdaten ein.

Diese Notationen dienen im weiteren Verlauf als Grundlage zur Beschreibung der technischen Abläufe und Schutzansprüche.

Kapitel 3. Erfindungskern und Ablaufbeschreibung des Verfahrens

3.1. Überblick des Verfahrens

Die vorliegende Erfindung beschreibt ein neuartiges Verfahren zur quantensicheren Absicherung digitaler Zahlungstransaktionen, das auf der Kombination dreier zentraler Sicherheitsbausteine beruht:

1. **Echt zufällige Transaktionswerte** aus einer Quanten-Zufallsquelle (QRNG), die für jede Transaktion einen nicht deterministischen Schlüssel k_q erzeugen,

2. **Kryptografische Verknüpfung** dieses k_q mit wesentlichen Transaktionsparametern (wie Betrag, Zeitstempel, Händler-ID), um einen Integritätswert M zu erzeugen – mittels kollisionsresistenter Verfahren wie Hashing, MAC oder symmetrischer Verschlüsselung,
3. **Digitale Signatur** dieses Wertes M unter Verwendung eines Post-Quantum-Kryptografieverfahrens (PQC), z. B. Falcon oder CRYSTALS-Dilithium, zur Sicherstellung der Authentizität und Integrität.

Ziel des Verfahrens ist es, jede Transaktion eindeutig, nachträglich nicht manipulierbar und resistent gegenüber zukünftigen quantenbasierten Angriffen zu gestalten – sowohl im lokalen als auch im verteilten Einsatzkontext.

3.2. Technischer Ablauf (Detailbeschreibung)

Schritt 1: Initialisierung der Transaktion

Wenn ein Kunde eine Zahlung an einem Terminal (z. B. POS in einem Supermarkt) mit einer Karte, einem mobilen Gerät oder einem anderen Zahlungsmedium vornimmt, werden folgende Daten systemseitig initialisiert:

Betrag der Transaktion (B)
Zeitpunkt der Transaktion (t)
Händler- oder Empfängerkennung (H)
ggf. weitere Parameter (Transaktions-ID, Geräte-ID etc.)

Diese Daten alleine wären ohne zusätzliche Absicherung durch Quantenmechanismen potenziell langfristig angreifbar – insbesondere, wenn sie klassisch verschlüsselt oder gespeichert werden.

Schritt 2: Erzeugung einer echten Zufallszahl k_q mittels QRNG

Ein QRNG-Modul (lokal oder cloudbasiert über die Bank) erzeugt nun in Echtzeit eine quantengenerierte Zufallszahl k_q, welche:

für jede Transaktion einzigartig ist,
nicht vorhersagbar ist (kein deterministischer Algorithmus),
und nicht rekonstruierbar ist (selbst vom QRNG nicht reproduzierbar).

> Beispiel:
> k_q = 10111001001110100010100100110101001001 1... (256–512 Bit)

Die vorliegende Erfindung verwendet zur Absicherung jeder Transaktion eine durch einen Quanten-Zufallszahlengenerator erzeugte Bitfolge k_q mit einer empfohlenen Länge von mindestens 256 Bit, idealerweise 512 Bit. Die Darstellung kann dabei in binärer oder hexadezimaler Notation erfolgen. Diese Bitfolge dient als hochgradig entropiereiche Eingabe zur Berechnung eines transaktionsspezifischen Integritätswerts (M), der in der Folge mittels Post-Quantum-Kryptografie (PQC) signiert wird.

Diese Zufallszahl bildet den dynamischen Sicherheitsanker der Transaktion. Sie wird nicht dauerhaft gespeichert oder offengelegt, sondern ausschließlich für die kryptografische Verknüpfung mit den Transaktionsparametern genutzt.

Hinweis: Zur besseren Verständlichkeit wird in den folgenden Kapiteln exemplarisch ein **Hash-Verfahren** zur Bildung des Integritätswerts *M* verwendet. Alternativ kommen jedoch auch andere kollisionsresistente oder kryptografisch abgesicherte Verfahren in Betracht (z. B. MAC, symmetrische Verschlüsselung oder deterministische Bindungsverfahren), wie in den Patentansprüchen offen gelassen.

Schritt 3: Bildung eines Integrität-Werts M

Es erfolgt eine kollisionsresistente Verknüpfung der Transaktionsparameter mit der Zufallszahl, in unseren Beispiel wird Hash benutzt:

M = *Hash*(B, t, H, k_q)

Als Hashfunktion kann z. B. SHA3-512 oder ein anderer NIST-zugelassener Algorithmus verwendet werden. Dieser Wert M ist eine eindeutige, irreversible Darstellung der Transaktion inklusive der quantenbasierten Zufallszahl.

> Beispielhafte Hash-Funktion:

> M = *SHA3-512* (50.00, 2025:14:37:52, *Händler\EDEKA*_008, k_q)

Dieser Wert wird anschließend signiert – nicht verschlüsselt – und übertragen.

Schritt 4: Digitale Signatur durch Post-Quantum Cryptography (PQC)

Sobald der Hashwert M erstellt wurde – also die Kombination der zentralen Transaktionsdaten wie Betrag (B), Zeitstempel (t), Händlerkennung (H) und einer quantenzufällig erzeugten Zahl (k_q) – muss sichergestellt werden, dass dieser Hashwert nicht nur vertraulich ist, sondern auch authentisch und unverändert bleibt. Um das zu garantieren, wird M mit Hilfe eines quantenresistenten Signaturverfahrens signiert, das zur Klasse der Post-Quantum Cryptography (PQC) gehört.

Dabei entsteht eine Signatur σ, die wie folgt definiert ist:

$$\sigma = \text{PQCsign}(M)$$

Diese Signatur erfüllt drei entscheidende Aufgaben:
(1) Authentizität – nur eine autorisierte Einheit mit dem passenden privaten Schlüssel kann diese Signatur erzeugen.
(2) Integrität – jede Veränderung an den Transaktionsdaten, selbst an einem Bit, macht die Signatur ungültig.
(3) Zukunftssicherheit – auch wenn leistungsstarke Quantencomputer verfügbar werden, bleibt die Signatur durch das zugrundeliegende PQC-Verfahren unknackbar, sofern der private und öffentliche Schlüssel geheim bleibt.

Zum Einsatz kommen hierbei anerkannte und standardisierte PQC-Verfahren, darunter insbesondere:

CRYSTALS-Dilithium (vom NIST offiziell als PQC-Standard empfohlen),
Falcon,
SPHINCS+,
oder vergleichbare Algorithmen mit nachgewiesener Resistenz gegen Quantenangriffe.

Diese Algorithmen basieren nicht wie klassische Systeme auf Primfaktorzerlegung oder elliptischen Kurven, sondern auf fundamentalen mathematischen Problemen, etwa Gitterproblemen oder Hash-basierten Konstruktionen, die selbst für Quantencomputer als praktisch unlösbar gelten.

Ein wichtiger Bestandteil dieser Signatur ist das verwendete Schlüsselpaar, bestehend aus einem privaten und einem öffentlichen Schlüssel:

Der private Schlüssel dient zur Erzeugung der Signatur σ und muss stets geheim bleiben. Er kann entweder lokal gespeichert sein, etwa auf einem sicheren Chip (Smartcard, TPM-Modul, Handy-Secure-Element), oder auf einem abgesicherten Server (z. B. bei der Bank) über eine verschlüsselte API-Schnittstelle angesprochen werden.
Der öffentliche Schlüssel wird hingegen von der Bank oder einer autorisierten Prüfinstanz verwendet, um die Echtheit der Signatur zu überprüfen.

Die Entscheidung, ob die Signatur lokal oder serverbasiert erzeugt wird, liegt beim jeweiligen Anbieter bzw. Integrator des Systems. Beide Optionen – lokal und API-gestützt – sind technisch machbar und können je nach Infrastruktur und Sicherheitsanforderung flexibel eingesetzt werden. Wichtig ist lediglich, dass der private Schlüssel zu keinem Zeitpunkt öffentlich zugänglich ist.

Die Anwendung von PQC innerhalb dieses Systems ist kein zusätzliches Merkmal, sondern eine notwendige Weiterentwicklung klassischer Signaturmethoden im Zeitalter bevorstehender Quantenangriffe. Da bereits heute Daten von Angreifern gespeichert und später mit Quantenrechnern entschlüsselt werden könnten (sog. „Harvest Now, Decrypt Later“-Angriffe), ist die Verwendung quantenresistenter Signaturen essenziell für die Langzeitsicherheit von Zahlungstransaktionen.

Zusammengefasst schafft dieser Schritt die Grundlage für ein vollständig überprüfbares, nicht fälschbares und quantenresistentes Transaktionssystem – ohne dabei auf konkrete Anbieter oder einzelne Algorithmen beschränkt zu sein.

Schritt 5: Verifikation der Transaktion durch die Bank

Nach der digitalen Signatur des Transaktionswertes M (siehe Schritt 4) wird der Transaktionsvorgang zur Überprüfung an die Bank oder einen autorisierten Zahlungsdienstleister übermittelt.

Da es sich hierbei um ein quantensicheres Verfahren handelt, setzt die Verifikation auf zwei Säulen:
Die Integritätsprüfung mittels der digitalen Signatur σ, und die Rekonstruktion des Hash-Werts M aus bekannten Parametern.

Im Gegensatz zu herkömmlichen Verfahren, bei denen verschlüsselte Inhalte oder statische Schlüssel ausgetauscht werden, basiert dieses Modell auf einer vorher synchronisierten, deterministisch reproduzierbaren Transaktionsstruktur.

Ablauf:

Die Bank besitzt bereits alle zur Verifikation notwendigen Elemente, und zwar:

den öffentlichen Schlüssel zur Überprüfung der Signatur (zuvor hinterlegt, z. B. bankintern oder über zertifizierte Schlüsselinfrastruktur),
den zufällig generierten quantensicheren Wert k_q , der im Vorfeld durch ein gemeinsames QRNG-Modul erzeugt oder über eine sichere API zwischen Zahlungseinheit und Bank geteilt wurde,
sowie die Transaktionsparameter, also z. B. Betrag “B”, Zeitpunkt “t”, Händler-ID “H” und ggf. weitere operationale Metadaten, die automatisch im Transaktionsprozess von der Zahlungseinheit übermittelt werden.

Da sowohl k_q als auch alle Transaktionsdetails bekannt sind, kann die Bank nun selbstständig den Hash-Wert M rekonstruieren:

$M^* = \text{Hash}(B \mid t \mid H \mid k_q)$

Diese Hashfunktion entspricht der auf der Zahlungseinheit verwendeten (z. B. SHA3-256 oder SHA3-512) und ist im System vorab definiert bzw. standardisiert.

Signaturprüfung:

Anschließend wird die empfangene digitale Signatur σ geprüft – und zwar anhand des öffentlichen Schlüssels und des lokal erzeugten Hash-Wertes M:

PQCverify (σ, M) = TRUE

Ist die Verifikation erfolgreich, so bedeutet das:

Der Hash-Wert M wurde nicht verändert,
Die Transaktionsdaten sind unverändert und authentisch,
Die Signatur stammt vom autorisierten Zahlungssystem mit gültigem Private Key,
Die Transaktion kann sicher freigegeben werden.

Im Fehlerfall (z. B. Manipulation, falsch übermittelter Hash, nicht reproduzierbarer k_q, ungültige Signatur) wird der Vorgang automatisch blockiert und protokolliert.

Warum ist dieses Modell sicher?

1. Asymmetrische Kryptographie: Der private Signierschlüssel verbleibt stets im autorisierten Modul (z. B. Karte, Smartphone, TPM), die Bank nutzt nur den öffentlichen Schlüssel zur Prüfung.

2. Einwegfunktion Hash: Der erzeugte Wert M kann ohne Zugriff auf k_q nicht sinnvoll rekonstruiert werden, selbst wenn alle anderen Daten bekannt sind.

3. Synchronisierte QRNG-Verwendung: Da k_q entweder gemeinsam erzeugt oder sicher geteilt wurde, können beide Seiten identische Hash-Werte erzeugen – dies ermöglicht deterministische Verifikation, aber verhindert externe Rekonstruktion.

4. Keine Übertragung sensitiver Daten: Weder k_q noch M müssen im Klartext übertragen werden – sie entstehen lokal auf beiden Seiten. Nur die Signatur σ selbst verlässt das Gerät.

Schritt 6: Abschluss der Transaktion und Sicherheitsgarantien

Nach erfolgreicher Verifikation der digitalen Signatur σ durch den autorisierten Verifizierer (z. B. die Bank oder ein zertifizierter Zahlungsdienstleister), erfolgt die abschließende Freigabe der Transaktion. Diese Bestätigung kann in Echtzeit an das Zahlungsterminal zurückgemeldet werden und wird im internen System der Bank registriert.

Die Entscheidung basiert auf folgenden, eindeutigen Prüfmerkmalen:

1. Die Transaktionsparameter (B, t, H) stimmen mit den bankseitig bekannten bzw. übermittelten Daten überein.

2. Der daraus (gemeinsam mit k_q) generierte Hashwert M* stimmt bitgenau mit dem Hashwert M überein, der durch die Signatur σ abgesichert wurde.
3. Die Signaturprüfung mit dem öffentlichen Schlüssel verläuft erfolgreich

---

Vertrauliche Verifikation ohne Klartextübertragung

Ein entscheidender Vorteil dieser Architektur liegt darin, dass das Integritäts Wertes M nicht im Klartext entschlüsselt oder übertragen werden muss. Die Bank verfügt über sämtliche benötigten Daten zur Validierung:

Zugriff auf k_q durch interne Erzeugung oder API-Synchronisation,
Kenntnis über B, t, H, da diese Parameter im Rahmen der Transaktionsanfrage vorab übermittelt wurden,
Zugriff auf den öffentlichen Schlüssel, zur Prüfung der Signatur.

Mit diesen Daten wird intern ein Referenzwert M* berechnet, der nicht von außen angreifbar ist, da er nie übertragen wurde.

Wenn M* mit dem übermittelten signierten M übereinstimmt, ist sichergestellt, dass:

keine Manipulation stattgefunden hat,
der Absender autorisiert war,
und der Inhalt korrekt ist – ohne dass M entschlüsselt werden musste.

Transaktionslogging – ohne Datenschutzrisiko

Da die Bank alle Transaktionsparameter (z. B. Betrag, Händlerkennung, Uhrzeit) bereits kennt, kann sie diese nun direkt in ihren Systemen speichern und protokollieren – ohne die Notwendigkeit, M im Klartext zu entschlüsseln. Die Bank hat ja selbst M* rekonstruiert und somit sämtliche Informationen über die Transaktion erhalten.

Das bedeutet:

vollständige Nachvollziehbarkeit der Transaktion intern,
aber keine preisgegebenen sensiblen Daten während der Übertragung – auch nicht für MITM-Angreifer mit Zugang zu Quantencomputern,
sowie nachweisbare Unverfälschbarkeit durch die digitale Signatur.

Verhalten bei Fehler oder Manipulation

Im Falle einer nicht bestandenen Prüfung (z. B. abweichender Hash, ungültige Signatur, fehlende k_q Korrelation) wird die Transaktion automatisch abgelehnt. Die Bank kann einen Fehlerbericht oder Angriffsprotokoll erzeugen und optional forensische Analysemechanismen einleiten.

Zusammenfassung Schritt 6:

Transaktion wird nur nach vollständiger Signaturprüfung freigegeben.
Die Bank kann alle relevanten Daten protokollieren, ohne dass M je entschlüsselt oder preisgegeben wurde.
Die Transaktion bleibt quantenresistent, datenschutzkonform und nachweisbar sicher.
