IT-Sicherheit bereits durch sichere Hardware statt durch Anti-Viren-Software (gegenwärtige Situation)

Ziel

Ein Hacker-Angriff auf die Firma RSA hat 2011 den Anlass gegeben, einen Weg zu finden, derartige Angriffe zukünftig zu verhindern. Die Lösung ist die in Europa und den USA bereits patentierte Sicherheits-Hardware-Architektur „Schad-Software-sichere Datenverarbeitungssysteme“, die im Folgenden S^3DVS abgekürzt wird.

Für deren Vermarktung wird Unterstützung gesucht, z.B. durch Investoren, Forschungs- und Entwicklungspartner, Multiplikatoren, Vertriebspartner.

Was ist S^3DVS?

S^3DVS ist eine skalierbare Hardware-Architektur für digital arbeitende programmierbare Geräte. Sie bewirkt, dass entsprechend gebaute Geräte technisch nicht mehr im Stande sind, Schad-Software überhaupt auszuführen. Durch ihre Skalierbarkeit ist sie sowohl für Großrechner in Multiprozessortechnik geeignet, wie auch für Einzelplatzgeräte (Desktops, Laptops, Tablets) und tragbare Geräte (Wearables, z.B. Smartphones und Smartwatches).

Wo liegt der große Vorteil eines Hardware-basierten Ansatzes?

Die Ursache für die Angreifbarkeit von Datenverarbeitungssystemen durch Schad-Software liegt bei praktisch allen heute verwendeten programmierbaren Geräten in der Hardware-Architektur begründet. Deren Prinzip hat sich seit den vierziger Jahren des vorigen Jahrhunderts nicht wesentlich geändert. Von den Marktteilnehmern wird diese Tatsache aus verschiedenen Gründen gern verschwiegen:

- Von den Hardware-Herstellern, weil sie durch eine disruptive Änderung
    - den Markterfolg ihrer bisherigen Produktion,
    - den externen Software-Support und
    - etablierte Bindungen zu ihren Zulieferern gefährden.
- Von den Software-Herstellern, weil sie einen einträglichen Zweig – Antiviren-Software und dergleichen – in Frage stellen müssten.
- Von beiden, weil die gewinnbringende gegenseitige Abhängigkeit gefährdet werden könnte.

Zu diesen Gründen kommt die Erkenntnis, dass Neuerungen die bestehende Produkte, Verfahren oder Strukturen in nennenswertem Umfang ändern, nur selten mit geringem Aufwand oder schnell umzusetzen sind.

Wie entsteht der große Gewinn an Sicherheit?

Dieser entsteht durch zwei Maßnahmen:

1. Die Sortierung der Daten in Kategorien. In herkömmlichen Hardware-Architekturen sind Daten aller Kategorien ungeordnet nebeneinander abgespeichert. (Diese Unordnung wird von Hackern genutzt, um Instruktionen als Daten einzuschleusen und dann ausführen zu lassen. Instruktionen in diesem Sinne sind Anweisungen an Prozessoren, bestimmte Manipulationen an Daten oder Programmen vorzunehmen.)
2. Die einzelnen Datenkategorien werden in eigenen, voneinander unabhängigen Speichereinheiten abgelegt. Hierdurch wird erreicht, dass als Daten eingegebene Speicherinhalte nicht als Instruktionen für die Prozessoren missbraucht werden können.

Wovor schützen S^{3}DVS konkret?

S^{3}DVS sind eine wirksame Maßnahme gegen jegliche Software, die in ein System über folgend genannte Vektoren eingebracht wird:

- Wechselspeicher,
- Internet-Seiten,
- E-Mail-Anhänge,
- infizierte Dateien,
- Netzwerke,
- Daten-Schnittstellen

Dadurch schützen S^{3}DVS zum Beispiel vor den folgend aufgeführten Arten von Schad-Software, einschließlich der von ihnen möglicherweise installierten Programme. Dieser Schutz ist unabhängig von Alter und „Einschleppzeitpunkt“ der Schad-Software, und wirkt somit auch gegen alle zukünftige Versionen von:

- Viren,
- Trojaner
- Würmer,
- Ransomware,
- Rootkits,
- Bootkits

Dabei spielt es keine Rolle, ob die angreifende Schad-Software als Dateiinhalt, datei-los („file-less“) oder kodiert ist. Selbst steganografisch kodierte Schad-Software wird sicher geblockt.

Die aktuelle Entwicklung hat gezeigt, dass S^{3}DVS auch Fehler verhindern, die durch Ausnutzung von Schwachstellen der Prozessorfunktion „speculative execution“ ausgelöst werden können, und Anfang 2018 unter den Namen

- Spectre und
- Meltdown

bekannt wurden und vielfach große finanzielle Schäden verursacht haben. Weil die Entwicklung neuer Prozessoren, die die heutige Schwachstelle dann nicht mehr haben, kostspielig und zeitaufwändig ist, könnten S^{3}DVS als Alternative dazu dienen, weil die betroffenen, bisher gebauten Prozessoren in der Architektur von S^{3}DVS weiterhin fehlerfrei und risikolos funktionieren.

Was leisten S^{3}DVS außerdem?

Durch ihre Architektur ermöglichen S^{3}DVS die Entwicklung von Geräten, die für spezielle Aufgaben besser geeignet sind, als ihre zurzeit verfügbaren Vorläufer. Hier sollen stellvertretend zwei Gruppen von Geräten erwähnt werden:

- Geräte, die dem kontrollierten Austausch von Daten zwischen Netzwerken mit unterschiedlichen
  - Zugangsattributen,
  - (Sicherheits-) Einstufungen oder
  - Verwendungen dienen.
- Zähler, deren Stände nicht manipulierbar sind (z.B. Kilometerzähler von Fahrzeugen)

Solche Anforderungen sind im Rahmen der Digitalisierung und des „Internet of Things“ mit Sicherheit zu erwarten. Mit herkömmlicher Technik ist, bedingt durch ihre inhärent fehlende Sicherheit, eine garantierbare Erfüllung dieser Anforderungen unmöglich.

Darüber hinaus bieten S^{3}DVS gegenüber ihren konventionellen Vorläufern weitere Vorteile:

- Geringere Software-Last durch den Wegfall der Anti-Viren-Software, dadurch
  - Schnellere Reaktionszeiten
  - Kürzere Wartezeiten
  - Geringerer Speicherbedarf
  - Weniger Personalbedarf im IT-Bereich
- Keine Notwendigkeit, Anti-Viren-Software
  - Zu erwerben und
  - Zu aktualisieren
- Zeitersparnis nach Hacker-Angriffen wegen Wegfall des Zeitaufwandes für
  - Software-Neu-Installation und
  - Beseitigung von Schäden.

Kann die Software von S^{3}DVS „Over the Air“ aktualisiert werden?

Mit „Over the Air Updates“ bezeichnet man die heute verbreitete Methode der Aktualisierung von Software:

- Sie erfolgt über das Internet (daher der Name).
- Umfang und Zeitpunkt werden vom Software-Hersteller bestimmt.
- Treffen den Nutzer häufig zur Unzeit.
- Führen in großen Projekten oder Teams zur Nutzung unterschiedlicher Software-Stände.

Die Frage der Überschrift muss in zwei Schritten beantwortet werden:

1. Ja, es ist möglich, neue Software-Versionen nach diesem Verfahren zu laden.
2. Es ist nicht möglich, neu heruntergeladene Software-Stände direkt zu nutzen. Um das zu tun ist es erforderlich, dass die neu heruntergeladene Software mittels eines geeigneten Mediums zunächst als Daten exportiert wird um danach als Software (über eine andere Schnittstelle!) wieder eingebracht wird. Diese Maßnahme gibt dem Systemnutzer die Möglichkeit, den Zeitpunkt für Software-Aktualisierungen selbst zu bestimmen, z.B. an Erfordernissen eines Projektes.
   Zwischen Export und Re-Import ist es möglich, eine Prüfung durchzuführen, die verifizieren soll, ob der Download den Erwartungen hinsichtlich Sicherheit und Funktionalität entspricht.
   Eine solche Qualitätsprüfung ist bei herkömmlichen Geräten bisher gar nicht vorgesehen!

Wie schützen S³DVS Ihre IT-Investitionen und senken IT-Kosten deutlich?

Die wachsende Anzahl von Angriffen mit Schad-Software hat dazu geführt, dass Anti-Virus-Software und dergleichen immer mehr Ressourcen beanspruchen, und zwar sowohl hinsichtlich

- Software (Anschaffung, Lizenzerwerb und Aktualisierungen)
- Hardware-Beschaffung und
- Personaleinsatz.

Insbesondere führt Anti-Viren-Software-Einsatz mit ihrem stetig steigenden Ressourcenbedarf dazu, dass Hardwarekomponenten entweder nach immer kürzeren Einsatzzeiten ausgetauscht werden müssen, oder mit großen Reservekapazitäten eingekauft und vorgehalten werden.

Weil S³DVS ohne laufenden Software-Bedarf Sicherheit schaffen, wird die Nutzungsdauer der Systeme länger, und nicht mehr von Angreifern mitbestimmt.

S³DVS können bei ihrer Anschaffung nach den prognostizierten operationellen Anforderungen konfiguriert werden; die weiterhin zunehmende Bedrohung durch Schad-Software bedarf keiner Berücksichtigung. Die bisher für Anti-Viren-Software vorgehaltenen Ressourcen, insbesondere Personal und Software, werden nicht mehr benötigt.

Sind S³DVS ökologisch?

Das hängt ganz davon ab, mit welchen Techniken S³DVS umgesetzt werden. Weil es sich um eine Architektur handelt, sind ökologische Gesichtspunkte erst bei der Planung konkreter Produkte zu beachten. Unter diesem Aspekt werden an deren Hersteller keine Fragen gestellt, die nicht auch bei Produkten zu beantworten sind, die nach herkömmlichen Architekturen gebaut werden.

Wie können S³DVS konstruiert werden?

Weil es sich bei S³DVS um eine Architektur handelt, ist die Antwort auf diese Frage abhängig vom Integrationsgrad der betrachteten Ausgangsarchitektur:

- Diskrete Bauelemente: Diese finden sich nur noch in relativ alten Geräten oder bei Einzellösungen; bei ihnen ist die Anpassung an S³DVS möglich. Voraussetzung für den erforderlichen Umbau ist die technische Möglichkeit, Verbindungen (Daten- und Signalleitungen) zu lösen und an anderer Stelle neu herzustellen.
- Geräte mit Prozessoren und separaten Speicher-Bausteinen: In diesen Fällen reicht prinzipiell die Neugestaltung der Leiterplatte.
- System-on-a-chip (SOC): Diese hohe Integrationsstufe ist bei tragbaren Geräten (z.B.: Smartphones, Wearables) zu finden, und bei industriellen Anwendungen, bei denen räumliche oder andere Anforderungen eine solche Lösung verlangen. Eine Anpassung an die Architektur von S³DVS verlangt bei diesen Geräten einen Neuentwurf der Halbleiterstrukturen auf dem Chip.

Soll ein existierendes Gerät entsprechend der Architektur von S³DVS nachgebaut werden, ist, verglichen mit dem Vorläufermodell, die Verbindung zwischen den Prozessoren und den Speichern aufzubrechen. Diese Verbindung ist unter Berücksichtigung der erforderlichen Anzahl unabhängiger Speicherbereiche neu herzustellen. Dabei ist darauf zu achten, dass jedem Speicherbereich individuell die Zugriffsattribute zugeordnet sind, die der in ihm abgelegten Datenkategorie entsprechen.

Welche Alternativen gibt es zu S³DVS?

Zurzeit ist keine Lösung zu Sicherheitsfragen in der IT bekannt, die - wie S³DVS - auf Änderungen der Hardware-Architektur aufsetzt.

Folglich besteht die Konkurrenz zu S³DVS aus

- „Trusted Platform Modules“:
  Diese relativ junge Technologie benötigt einen zusätzlichen Chip mit kryptografischen Informationen, und beruht auf einer Authentifizierung, die beim Initialisieren des Systems angewandt wird, um die Vertrauenswürdigkeit der Software zu verifizieren. Gegenüber S³DVS hat sie folgende Nachteile:
  - Es kann nur eine Software, bzw. ein Softwarehersteller authentifiziert werden.
  - Es besteht eine dauernde Abhängigkeit von diesem Softwarehersteller.
  - Die Authentifizierung erfolgt an Hand einer – wenn auch langen – Folge von Bits und kann prinzipiell mit den Methoden überwunden werden, die auch zum Hacken von Passworten benutzt werden.
  - Sie bietet keinen Schutz vor Malware, die Schwachstellen ausnutzen wie
    - Stack-Overflows,
    - Spectre oder
    - Meltdown.
  - Dem Nutzer werden Steuer- und Kontrollmöglichkeiten genommen, weil jegliche Software-Konfiguration nur im Rahmen der extern bereitgestellten Programmteile möglich ist.
  - Verschiedenen Organisationen – z.B. das Bundesamt für Sicherheit in der Informationstechnik – stehen dem Verfahren skeptisch gegenüber.
- Antiviren-Software und ähnlichen Produkten, deren Nachteile hinlänglich bekannt sind:
  - Sie wirken nur gegen ihnen bekannte Angreifer,
  - erkennen kodierte Schad-Software nicht,
  - sind wirkungslos gegenüber zukünftigen Angreifern und
  - erfordern häufige Aktualisierungen.
- Anweisungen an Nutzer, mit folgenden Schwachpunkten:
  - Wiederholte Schulungen, Unterweisungen und Ermahnungen sind erforderlich, insbesondere nach Erkennen neuer Angriffsformen,
  - Nachlässigkeit, insbesondere bei Routinearbeiten und
  - Anfälligkeit gegenüber
    - Neugier,
    - Täuschungen,
    - „Social Engineering“ und
    - „Phishing“.

Welche Maßnahmen erfordert S³DVS?

Jede neue Hardware-Architektur, die an programmierbaren Geräten angewandt wird, erfordert Maßnahmen, um die zugehörige Software an die neuen Gegebenheiten anzupassen. Im Falle der Umstellung eines realisierten Systems auf S³DVS ist keine Änderung der programmierten Logik erforderlich, sondern eine Behandlung der Software nach folgender Liste:

- Ersetzen von selbstmodifizierendem Code durch statischen Code,

- Ersetzen von on-line zu kompilierendem Source-Code durch entsprechenden Object-Code,
- Sortieren aller Daten des Systems nach festgelegte Kategorien.

Ferner ist es erforderlich, Nutz-Software über eine eigene Schnittstelle vorzusehen, die physisch von den Datenschnittstellen verschieden ist.

Wie entstand die Idee zu S^{3}DVS?

Im Frühjahr 2011 erfolgte ein Hacker-Angriff auf die Firma RSA, bei dem es Hackern gelang, sicherheitsrelevante Informationen des Unternehmens zu entwenden, auf denen das Geschäftsmodell von RSA beruht.

Der damalige international aufgestellte Arbeitgeber des Entwicklers nutzte die RSA-Sicherheitstechnik um sein Virtual Private Network (VPN) zu sichern. Durch den Hacker-Angriff war die Kommunikation über das VPN nicht mehr möglich. Mehrere Wochen Rückgriff auf herkömmliche Kommunikationstechniken hat die tägliche Arbeit massiv stark beeinträchtigt.

Die technischen Ursachen für den Erfolg dieses Angriffes wurden detailliert analysiert und in der Hardware-Architektur gefunden. Bei vielen anderen diesbezüglich untersuchten Systemen wurde die exakt gleiche Schwachstelle ebenfalls festgestellt. Auf der Basis dieser Ergebnisse wurden die erkannten Schwachstellen ausgemerzt, und unter Erhalt der technischen Notwendigkeiten programmierbarer Geräte das Prinzip von S^{3}DVS konzipiert und bis zur Patentreife entwickelt.

Kontaktperson:

Friedhelm Becker

von-Thünen-Straße 65A

26434 Waddewarden

Tel.: 04461 5911

Mobil: 0152 0582 7500

E-Mail: friedhelm.becker@dcb-becker.de

Homepage: http://dcb-becker.de

Social Media: https://www.xing.com/profile/Friedhelm_Becker7

Friedhelm Becker ist Freier Erfinder im Sinne des Arbeitnehmererfindungsgesetzes. Er wurde 1952 geboren. Nach erfolgreich abgeschlossenem Chemiestudium hat er drei Jahre in einem Labor für Baustoffprüfung gearbeitet und war anschließend acht Jahre bei der Bundeswehr. Er ist Marineoffizier im Ruhestand. Nach dem Ausscheiden aus dem militärischen Dienst war er bei namhaften Firmen auf den Sektoren Rechnerbau (Univac, Sperry, UNISYS) sowie Luft- und Raumfahrttechnik (Lockheed-Martin) in verschiedenen Verwendungen tätig. Von 1974 bis 2017 arbeitete er auf dem Sektor rechnergestützte Sensor-Effektor-Integration.

Im Laufe seiner beruflichen Tätigkeiten hat er Qualifikationen in

- Werkstoffprüfung (Gesteine, Erdölprodukte, bituminöse Baustoffe),
- Entwicklung von Werkstoffen, die vorgegebene Eigenschaften haben (wasserdichte Auskleidung des Elbe-Seitenkanal-Troges),
- Statistik,
- Programmierung (speziell von Real-Time-Systems),
- Lineare Optimierung,
- Digitaltechnik,
- Rechner-Hardware,
- Betriebssystemen,
- Echtzeitsystemen,
- Systemdesign,
- Fehleranalyse (in Hardware, in Software, zwischen Systemkomponenten)
- Qualitätssicherung (auf Software- und System-Ebene),
- Ausbildungstechniken und
- Projektleitung

erworben und intensiv angewendet.

Insbesondere die Kenntnisse über Digitaltechnik, Rechner-Hardware, Betriebssysteme und Qualitätssicherung, sowie die Kenntnis analytischer Verfahren haben dazu beigetragen, die hier vorgeschlagene Sicherheitslösung zu erarbeiten und bis zur Patentreife zu entwickeln.