

(12) **Österreichische Patentanmeldung**

(21) Anmeldenummer: A 51002/2022 (51) Int. Cl.: **G06F 21/00** (2006.01)
(22) Anmeldetag: 23.12.2022 **G06F 21/12** (2013.01)
(43) Veröffentlicht am: 15.07.2024 **G06F 11/00** (2006.01)
G06F 11/10 (2006.01)

(56) Entgegenhaltungen: WO 2020247993 A1 EP 1622022 A1 WO 2005071543 A2 Schuldt Fabian et al. "Effiziente systematische Testgenerierung für Fahrassistenzsysteme in virtuellen Umgebungen" 10/07/2013 abgerufen im Internet am 15.11.2023 unter dem Link URL:<http://www.digibib.tu-bs.de/?docid=00052570> Papadimitratos P. et al., "Secure vehicular communication systems: design and architecture," in IEEE Communications Magazine, vol. 46, no. 11, November 2008, doi: 10.1109/MCOM.2008.4689252.	(71) Patentanmelder: AVL List GmbH 8020 Graz (AT) (72) Erfinder: Marksteiner Stefan Dipl.-Ing. 8076 Vasoldsberg (AT) Priller Peter Dipl.-Ing. 8111 Gratwein-Straßengel (AT) Wolf Markus MSc 8451 Heimschuh (AT) (74) Vertreter: Patentanwälte Pinter & Weiss OG 1040 Wien (AT)
---	---

(54) **Verfahren zum Adaptieren von Testfällen für eine Sicherheitsüberprüfung**

- (57) Die Erfindung betrifft ein computerimplementiertes Verfahren zum Adaptieren von Testfällen (TC1) für eine Sicherheitsüberprüfung eines zu testenden funktionalen Systems einer Mobilitätsanwendung, insbesondere aus dem Automotive- und/oder Aeronautik-Bereich, mittels eines Testsystems sowie das Testsystem und ein zugehöriges Computerprogrammprodukt. Dem Testsystem werden dabei Testfälle (TC1) zur Verfügung gestellt, welche bereits für Sicherheitsüberprüfungen von funktionalen Systemen, welche in oder für Mobilitätsanwendungen zum Einsatz kommen, angewendet wurden und/oder für derartige Sicherheitsüberprüfungen anwendbar sind. Dabei werden folgende Schritte ausgeführt:

- Unterteilen der Testfälle (TC1) in Testmodule (T11, ..., T14) (101);
- Abstrahieren der Testmodule (T11, ..., T14), wobei in den Testmodulen (T11, ..., T14) enthaltene, systemspezifische Informationen, Daten und Parameter der funktionalen Systeme, auf welche die jeweiligen Testfälle (TC1) angewendet wurden, in einem jeweiligen Testmodul (T11, ..., T14) durch abstrakte Platzhaltervariablen ersetzt werden (102);
- Erstellen von abstrakten Testszenarien (TS1, TS2, TS3) für die Sicherheitsprüfung des zu testenden funktionalen Systems, wobei für ein Testszenario (TS1, TS2, TS3) die abstrahierte Testmodule (a1, ..., a5) kombiniert werden (103);
- Ableiten von Testfällen (TC21, TC22, TC23) für das zu testende funktionale System aus den abstrakten Testszenarien (TS1, TS2, TS3), wobei die jeweiligen abstrakten Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) eines jeweiligen Testszenarios (TS1, TS2, TS3) mit entsprechenden systemspezifischen Informationen, Daten und Parametern des zu testenden funktionalen Systems belegt werden (104); und
- Anwenden der aus den Testszenarien (TS1, TS2, TS3) abgeleiteten Testfälle (TC21, TC22, TC23) auf das zu testende funktionale System (105).

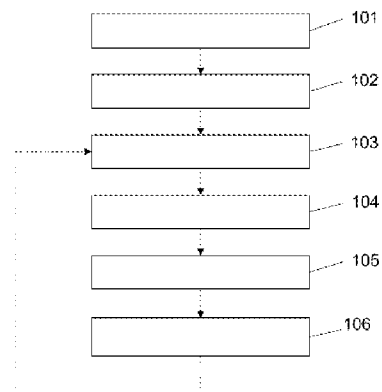


Fig. 1

Zusammenfassung

Die Erfindung betrifft ein computerimplementiertes Verfahren zum Adaptieren von Testfällen (TC1) für eine Sicherheitsüberprüfung eines zu testenden funktionalen Systems einer Mobilitätsanwendung, insbesondere aus dem Automotive- und/oder Aeronautik-Bereich, mittels eines Testsystems sowie das Testsystem und ein zugehöriges Computerprogrammprodukt. Dem Testsystem werden dabei Testfälle (TC1) zur Verfügung gestellt, welche bereits für Sicherheitsüberprüfungen von funktionalen Systemen, welche in oder für Mobilitätsanwendungen zum Einsatz kommen, angewendet wurden und/oder für derartige Sicherheitsüberprüfungen anwendbar sind. Dabei werden folgende Schritte ausgeführt:

- Unterteilen der Testfälle (TC1) in Testmodule (T11, ..., T14) (101);
- Abstrahieren der Testmodule (T11, ..., T14), wobei in den Testmodulen (T11, ..., T14) enthaltene, systemspezifische Informationen, Daten und Parameter der funktionalen Systeme, auf welche die jeweiligen Testfälle (TC1) angewendet wurden, in einem jeweiligen Testmodul (T11, ..., T14) durch abstrakte Platzhaltervariablen ersetzt werden (102);
- Erstellen von abstrakten Testszenarien (TS1, TS2, TS3) für die Sicherheitsprüfung des zu testenden funktionalen Systems, wobei für ein Testszenario (TS1, TS2, TS3) die abstrahierte Testmodule (a1, ..., a5) kombiniert werden (103);
- Ableiten von Testfällen (TC21, TC22, TC23) für das zu testende funktionale System aus den abstrakten Testszenarien (TS1, TS2, TS3), wobei die jeweiligen abstrakten Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) eines jeweiligen Testszenarios (TS1, TS2, TS3) mit entsprechenden systemspezifischen Informationen, Daten und Parametern des zu testenden funktionalen Systems belegt werden (104); und
- Anwenden der aus den Testszenarien (TS1, TS2, TS3) abgeleiteten Testfälle (TC21, TC22, TC23) auf das zu testende funktionale System (105).

Fig. 1

Verfahren zum Adaptieren von Testfällen für eine Sicherheitsüberprüfung

Technisches Gebiet

Die gegenständliche Erfindung betrifft ein computerimplementiertes Verfahren zum Adaptieren von Testfällen für eine Sicherheitsüberprüfung eines zu testenden funktionalen Systems einer Mobilitätsanwendung, insbesondere aus dem Automotive- und/oder Aeronautik-Bereich, mittels eines Testsystems. Dem Testsystem werden dazu Testfälle zur Verfügung gestellt, welche für Sichereinheitsüberprüfungen von funktionalen Systemen, welche in oder für Mobilitätsanwendungen zum Einsatz kommen, angewendet wurden. Weiterhin bezieht sich die gegenständliche Erfindung auf ein zugehöriges Testsystem und ein zugehöriges Computerprogrammprodukt.

Stand der Technik

Viele moderne technische Geräte, insbesondere Fahrzeuge, weisen ab einem gewissen Komplexitätsgrad eine Vielzahl an eingebetteten und untereinander vernetzten Komponenten auf, die jeweils mit eigenen Prozessoren oder Microcontrollern versehen sind. Die gegenständliche Offenbarung ist jedoch nicht nur auf Fahrzeuge im eigentlichen Sinn beschränkt, sondern kann auch in Verbindung mit anderen technischen Einheiten und Systemen, welche eine Vielzahl untereinander vernetzter Komponenten aufweisen und vor allem im Automotive- und/oder Aeronautik-Bereich in Mobilitätsanwendungen – d.h. z.B. in Fahrzeugen oder in deren Umfeld – zum Einsatz kommen, verwendet werden. Im Zusammenhang mit der gegenständlichen Offenbarung werden diese allgemein unter dem Begriff „funktionales System“ zusammengefasst. Das bedeutet, ein derartiges funktionales System kann beispielsweise ein Fahrzeug selbst (z.B. Kraftfahrzeug, Lkw, Luftfahrzeuge, etc.) oder ein komplexes System sein, welches in einem Fahrzeug oder in Verbindung mit einem Fahrzeug zur Realisierung von Mobilitätsanwendungen zum Einsatz kommt.

Zu Beispielen derartiger funktionaler Systeme zählen neben den Fahrzeugen selbst unter anderem Kombinationen aus unterschiedlichen externen Einheiten/Systemen mit Fahrzeugen oder fahrzeuginternen Einheiten und Systemen, Kombinationen aus fahrzeuginternen Systemen und Einheiten sowie Kombinationen aus mehreren Fahrzeugen bzw. Einheiten/Systemen mehrerer Fahrzeuge. So können beispielsweise bestimmte Aufgaben von einer autonomen Computereinheit (z.B. Tablet, Smartphone, etc.) ausgeführt werden, welche über eine Schnittstelle mit einer Einheit oder einem System in einem (autonomen) Fahrzeug (z.B. Infotainmentsystem, GPS, Navigationssystem, etc.) kommuniziert.

Bei Mobilitätsanwendungen, wie z.B. dem autonomen Fahren (AD) oder der so genannten Vehicle-to-X-Kommunikation können Fahrzeuge oder fahrzeuginterne Systeme mit Infrastruktureinheiten (z.B. Ampeln, Schrankenanlagen, etc.) beispielsweise über Funk Daten oder Informationen austauschen. Weiterhin können bei Mobilitätsanwendungen, wie z.B.

- 5 Fahrassistenzsystemen oder so genannte Advanced Driving Assistance Systems (kurz: ADAS) beispielsweise innerhalb eines Fahrzeugs Einheiten und Systeme (z.B. Sensoren, etc.) mit anderen Einheiten und Systemen (z.B. Bremssystem, Motor, etc.) über Schnittstellen kommunizieren. Andererseits können auch mehrere Fahrzeuge untereinander beispielsweise über Funk kommunizieren, um gemeinsam Aufgaben auszuführen – wie etwa im
- 10 Sinne einer virtuellen Deichsel aneinandergeschnitten zu werden, oder um Mobilitätsanwendungen, wie z.B. Vehicle-to-Vehicle-Kommunikation zum Austausch von Informationen und Daten zwischen den Fahrzeugen, zu realisieren. Weiterhin kann beispielsweise ein Fahrzeug, wie z.B. ein Spezialfahrzeug im Baubereich, in der Landwirtschaft, ein Einsatzfahrzeug, etc. mit einem oder mehreren On-Board-Geräten verbunden sein. Derart kombinierte
- 15 Einheiten oder Systeme werden im Sinne der gegenständlichen Offenbarung als funktionales System angesehen, welches in einer Mobilitätsanwendung (z.B. Fahrzeug, autonomes Fahren, ADAS, Vehicle-to-Infrastructure- oder Vehicle-to-Vehicle-Anwendungen, etc.) zum Einsatz kommt bzw. für deren Realisierung verwendet wird.

- Üblicherweise verfügen die Komponenten eines solchen funktionalen Systems über eigene
- 20 Speichereinheiten und Kommunikationsschnittstellen und bilden daher jeweils ein eigenes Computersystem. Solche Komponenten werden auch als „Embedded Systems – ES“ bezeichnet und verrichten – weitgehend unsichtbar – ihre Aufgaben innerhalb des funktionalen Systems. Im Fall eines komplexen Gesamtsystems (z.B. Fahrzeug, Flugzeug, etc.) wird meist eine Vielzahl von ansonsten autonomen eingebetteten Systemen vernetzt. Komponenten
- 25 solcher funktionalen Systeme können auch so genannte „Cyber-Physical Systems – CPS“ sein. Unter einem Cyber-Physical System wird üblicherweise ein Verbund von softwaretechnischen Einheiten mit mechanischen und elektronischen Teilen verstanden, welche über eine Dateninfrastruktur (z.B. Internet, Bussystem, etc.) kommunizieren, wobei ein Cyber-Physical System beispielsweise durch Vernetzung von eingebetteten Systemen über ein
- 30 drahtgebundenes und/oder drahtloses Kommunikationsnetz gebildet und z.B. in Mobilitätsanwendungen (z.B. vernetzten Sicherheitssystemen, vernetzten Fahrassistenzsystemen, etc.) eingesetzt werden kann.

- Die Komplexität solcher funktionalen Systeme, vor allem im Automotive-Bereich und/oder Aeronautik-Bereich, wird damit schnell unüberschaubar hoch. Beispielsweise weist ein Fahrzeug
- 35 üblicherweise dutzende Steuergeräte auf, auf denen zusammen Software mit mehreren 10 Millionen Zeilen Programmcode ausgeführt wird. Als Beispiel für

Kommunikationsschnittstellen können alleine die verschiedenen drahtlosen Verbindungsprotokolle, wie z.B. Mobilfunkprotokolle (z.B. 5G, LTE, etc.), Bluetooth, WirelessLAN, RFID, Vehicle-to-X-Schnittstellen, etc., angeführt werden, die teilweise gleichzeitig im Einsatz sind.

Aufgrund der Komplexität und der vielfältigen Kommunikationsschnittstellen entsteht bei solchen funktionalen Systemen eine große Angriffsfläche („Attack Surface“) für Cyber-Angriffe, wie z.B. so genanntes Denial-of-Service (kurz: DoS)-Attacken/Flooding. Weiterhin besteht auch die Gefahr, dass Angriffe nicht nur über die eigentlichen Kommunikationsschnittstellen, sondern beispielsweise auch über Sensoren (z.B. LIDAR, Radarsystem, etc.) erfolgen, wobei derartige Angriffe z.B. mittels so genannter Fake-Signale erfolgen können. Im Prinzip wird jedes Ereignis, welches von außen oder innen auf das funktionale System, mit dem Ziel das funktionale System in unzulässiger Weise zu beeinflussen und/oder Daten des funktionalen Systems in unzulässiger Weise an Dritte weiterzuleiten, als Cyber-Angriff betrachtet werden.

Aufgrund der physikalischen Fähigkeiten von Fahrzeugen (Masse und Geschwindigkeit, damit hohe kinetische Energie, direkte Interaktion mit potentiell vielen Menschen), der großen Anzahl (Fahrzeugflotten) und der Entwicklung hin zu autonomen Fahrzeugen, können solche Angriffe ein enormes Gefährdungspotential entwickeln. Dies trifft nicht nur auf die große Anzahl an Fahrzeuge, wie etwa Kraftfahrzeuge oder Lkws zu, sondern betrifft auch schienengebundene Fahrzeuge, Wasser- und vor allem Luftfahrzeuge. Daher ist es wichtig, in Mobilitätsanwendungen eingesetzte funktionale Systeme – vor allem im Automotive- und Aeronautik-Bereich – gegen Cyber-Angriffe und damit missbräuchliche Beeinflussung abzusichern, damit derartige Angriffe auf das jeweilige System möglichst keine Wirkung haben bzw. möglichst wirkungslos bleiben.

Um funktionale Systeme gegen Cyber-Angriffe abzusichern und resilient zu machen, gilt es, Schwachstellen bereits möglichst früh im Entwicklungsprozess zu erkennen und möglichst zu verhindern. In der Konzeptphase kann dies durch entsprechende Architekturmaßnahmen geschehen, wie sie zum Beispiel in der Veröffentlichung „Secure Vehicular Communication-Systems: Design and Architecture“ von Papadimitratos, P., Buttyan, L., Holczer, T., Schoch, E., Freudiger, J., Raya, M., Ma, Z., Kargl, F., Kung, A., & Hubaux, J.-P., 2008, IEEE Communications Magazine, 46(11), 100-109, beschrieben sind. Während der Umsetzung des Konzepts kann dies z.B. durch entsprechende Entwicklungsprozesse, wie etwa Best Practice, Source Code Review, etc. geschehen.

Nach einem vollständigen Zusammenbau und einer Integration der Komponenten zum funktionalen System ergibt sich in jedem Fall der Bedarf für eine Sicherheitsüberprüfung des funktionalen Systems, um möglichst alle relevanten Schwachstellen finden zu können – vor allem jene, die sich erst durch die Kombination der Komponenten – wie z.B. eingebettete und/oder Cyber-Physical Systeme – ergeben. Die Sicherheitsüberprüfung soll es Herstellern

(„Original Equipment Manufacturer“ – OEM), staatlichen Stellen (z.B. Zulassungsbehörden), und anderen Interessensvertretern (z.B. Verbraucherverbände, kommerzielle Flottenbetreiber, etc.) ermöglichen, das konkrete Risiko eines funktionalen Systems einer Mobilitätsanwendung, wie z.B. eines Fahrzeugs, etc. in Bezug auf Cyber-Angriffe bewerten zu können.

- 5 Dies kann bei der Weiterentwicklung bzw. Verbesserung der Fahrzeuge bzw. des funktionalen Systems, für die Abnahme oder auch Zertifizierung und ähnliche Aufgaben zur Anwendung kommen. Die Sicherheitsüberprüfung soll insbesondere in den funktionalen Systemen (z.B. Fahrzeug, etc.) vorhandene, aber zunächst meist unbekannte Schwachstellen und potentielle Angriffspunkte für Cyber-Attacken aufdecken. Um Schwachstellen und potentielle
- 10 Angriffspunkte (d.h. zunächst unerkannte Wege, auf welchen ein Cyber-Angriff auf das zu testende System durchgeführt werden können) aufzudecken, werden bei einer Sicherheitsüberprüfung eines funktionalen Systems verschiedene Angriffe (z.B. in Form von Testvektoren) auf das zu testende funktionale System definiert und im Rahmen der Sicherheitsüberprüfung dann als systemspezifische Testfälle auf das zu testende funktionale System angewendet.
- 15

- Beispielsweise werden Sicherheitsüberprüfungen sowohl während der Entwicklung eines neuen Fahrzeugs beim OEM, bei der Markteinführung/Typisierung/Homologisierung, als auch später kontinuierlich während der gesamten Lebenszeit des Fahrzeugs bzw. des jeweiligen funktionalen Systems benötigt. Die Notwendigkeit wiederholter Sicherheitsüberprüfungen auf Systemtest-Ebene und damit eine laufende Überprüfung der Cybersicherheit ergibt sich aufgrund änderbarer Konfigurationen, laufender Updates von (Teilen der) Softwarekomponenten, geänderter Umgebungsbedingungen (beispielsweise Vehicle-to-Infrastructure – V2I, Vehicle-to-Vehicle – V2V: Änderungen bei Interaktionspartnern), neu bekannt gewordener Testvektoren (auch aus eigener Sicherheitsforschung), etc. Die Sicherheitsüberprüfung
- 20
- 25 kann dabei neben dem Hersteller, beispielsweise auch durch (Flotten)Betreiber, Zulassungsbehörden und/oder spezialisierte Drittfirmen durchgeführt werden.

- Allerdings ist es aufgrund sehr restriktiver Schutz- und Geheimhaltungspolitik von Herstellern, vor allem im Automotive-Bereich, aber auch im Aeronautik-Bereich, und/oder aufgrund einer individualisierten und/oder herstellerspezifischen Ausgestaltung der funktionalen Systeme für Mobilitätsanwendungen die Regel, dass für funktionale Systeme keine durchgehend standardisierten Hardware- und Softwaresysteme eingesetzt werden. Es ist daher schwierig, für Sicherheitsüberprüfungen allgemein anwendbare Testfälle zu erstellen, die geeignet sind, Aussagen über die Sicherheit bzw. Cyber-Security von funktionalen Systemen mit gleicher Funktionalität z.B. unterschiedlicher Hersteller zu treffen. Selbst Updates von Komponenten
- 30
- 35 eines funktionalen Systems oder ein Update eines funktionalen Systems können dazu führen, dass neue Testfälle für die Sicherheitsüberprüfung erstellt oder zumindest bestehende

Testfälle angepasst werden müssen. D.h. die auf Sicherheit bzw. Cyber-Security zu überprüfenden funktionalen Systeme sind – selbst bei gleicher Funktionalität – vielfach heterogen und größtenteils herstellerspezifisch, sodass die Testfälle für Sicherheitsüberprüfung daher für jedes zu testende funktionale System systemspezifisch und individuell neu erstellt, zumindest adaptiert werden müssen. Gegebenenfalls können Schwachstellen sowie mögliche Angriffspunkte zwar gleichartig sein, allerdings können bereits bestehende Testfälle aufgrund der individuellen Ausgestaltung des jeweils zu testenden funktionalen Systems nicht direkt von einem bereits getesteten funktionalen System auf ein noch zu testendes funktionales System übertragen werden. D.h., die Testfälle für die Sicherheitsüberprüfung müssen an das jeweils zu testenden funktionalen System individuell und häufig auch manuell angepasst werden. Damit sind Sicherheitsüberprüfungen von funktionalen Systemen, wie z.B. Fahrzeugen, Teil- oder Subsystemen von Fahrzeugen, welche in Mobilitätsanwendungen verwendet werden, mit einem großen Zeit- und Ressourcenaufwand verbunden.

Aus der Schrift US 2005/0160322 A1 ist beispielsweise ein Verfahren sowie ein System bekannt, durch welches eine spezifisches Automatisierungstestskript zum Erkennen von Fehlern in einer Anwendung in eine abstrakte Testfalldarstellung umgewandelt wird. Die abstrakte Testfalldarstellung wird dann in einer Datenbank zur Wiederverwendung abgespeichert. Dabei werden für die abstrakte Testfalldarstellung ein oder mehrere Anwendungszustände, externe Interaktionssequenzen und Eingangsdaten gespeichert und die abstrakte Testdarstellung kann mit Informationen aus einem Anwendungs-Metadaten-Depot ergänzt werden, um diese zum Testen der Anwendung auch in anderen Testumgebungen oder auf einer anderen Testplattformen verwenden werden kann. Damit können zwar Testfälle zum Testen einer Anwendung abstrahiert, allerdings können die abstrakten Testfalldarstellung nur zum Testen derselben Anwendung in einer anderen Testumgebung bzw. auf einer anderen Testplattform genutzt werden.

Darstellung der Erfindung

Der Erfindung liegt daher die Aufgabe zugrunde, ein Verfahren anzugeben, durch welches Testfälle, welche bei Sicherheitsüberprüfungen zum Testen von funktionalen Systemen bereits verwendet wurden und/oder für derartige Sicherheitsüberprüfungen verwendbar sind, für Sicherheitsüberprüfungen weiterer, noch zu testenden funktionaler Systeme auf einfache, ressourcensparende und automatisierte Weise angepasst und wiederverwendet werden können.

Diese und weitere Aufgaben werden durch ein Verfahren gemäß dem unabhängigen Anspruch gelöst. Vorteilhafte Ausführungsformen der vorliegenden Erfindung sind in den abhängigen Ansprüchen beschrieben.

Erfindungsgemäß erfolgt die Lösung der Aufgabe durch ein computerimplementiertes Verfahren zum Adaptieren von Testfällen für eine Sicherheitsüberprüfung eines funktionalen Systems, welche in einer Mobilitätsanwendung zur Anwendung kommt, mittels eines Testsystems. Dem Testsystem werden dabei die Testfälle, welcher bereits bei Sicherheitsüberprüfungen auf in Mobilitätsanwendungen eingesetzten funktionalen Systemen angewendet wurden und/oder für derartige Sicherheitsüberprüfungen anwendbar sind, zur Verfügung gestellt. Dabei werden folgende Schritte ausgeführt:

- Unterteilen der Testfälle in Testmodule;
- Abstrahieren der Testmodule, wobei in den Testmodulen enthaltene, systemspezifische Informationen, Daten und Parameter der funktionalen Systeme, auf welche die jeweiligen Testfälle angewendet wurden, in einem jeweiligen Testmodul durch abstrakte Platzhaltervariablen ersetzt werden;
- Erstellen von abstrakten Testszenarien für die Sicherheitsprüfung des zu testenden funktionalen Systems, wobei für ein Testszenario die abstrahierte Testmodule kombiniert werden;
- Ableiten von Testfällen für das zu testende funktionale System aus den abstrakten Testszenarien, wobei die jeweiligen abstrakten Platzhaltervariablen in den abstrahierten Testmodulen eines jeweiligen Testszenarios mit entsprechenden systemspezifischen Informationen, Daten und Parametern des zu testenden funktionalen Systems belegt werden; und
- Anwenden der aus den Testszenarien abgeleiteten Testfälle auf das zu testende funktionale System.

Der Hauptaspekt der vorgeschlagenen Lösung besteht darin, dass auf einfache und ressourcensparende Weise bereits bekannte oder bereits für Sicherheitsüberprüfungen von funktionalen Systemen in Mobilitätsanwendungen verwendete und/oder verwendbare Testfälle generalisiert und für Sicherheitsüberprüfungen von neu zu testenden funktionalen Systemen nutzbar gemacht werden können, wobei die Anpassung an das jeweils neu zu testende funktionale System automatisiert durchgeführt wird. Durch das erfindungsgemäße Verfahren werden bereits verwendete Testfälle automatisiert für Sicherheitsüberprüfungen neu zu testender funktionaler Systeme „recycelt“. Das bedeutet, es muss nicht für jedes neu zu testende funktionale System neue Testszenarien und konkrete Testfälle entwickelt werden, sondern es kann auf bereits bei Sicherheitsprüfungen auf anderen funktionalen Systemen angewendete und/oder dafür anwendbare Testfälle zurückgegriffen werden und diese dann automatisiert für die Sicherheitsprüfung des neu zu testenden funktionalen Systems automatisiert angepasst werden. Dadurch werden bei der Sicherheitsüberprüfung Ressourcen und Zeit gespart.

Weiterhin ist es vorteilhaft, wenn nach Anwenden der aus den Testszenarien abgeleiteten Testfälle auf das zu testende funktionale System für jeden Testfall eine Rückmeldung erstellt wird. Die Rückmeldung wird dann an das Testsystem übermittelt und im Testsystem mit dem jeweiligen Testszenario verknüpft, aus welchem der jeweilige Testfall abgeleitet wurde.

- 5 Dadurch können sehr leicht beispielsweise besonders gut funktionierende Testszenarien oder mangelhaft funktionierende Testszenarien identifiziert werden. Durch die Rückmeldung kann auch für jedes Testszenario hinterlegt werden, für welches funktionale System dieses bereits angewendet wurde und wie hilfreich es beispielsweise war, sicherheitsrelevante Schwachpunkte aufzudecken. Damit können bei nachfolgenden Sicherheitsüberprüfungen
- 10 desselben oder eines ähnlichen funktionalen Systems rasch funktionierende Testszenarien gefunden bzw. mangelhaft funktionierende Testszenarien vermieden werden.

- Es ist auch günstig, wenn beim Abstrahieren die Testmodule nach erforderlichen Parametern durchsucht werden, welche für eine Durchführung des jeweiligen Testmoduls vom zu testenden funktionalen System oder zumindest einem der vorhergehenden Testmodule zur Verfügung zu stellen sind. Diese erforderlichen Parameter können in einer Parameterliste im Test-
- 15 system hinterlegt werden, wobei die jeweilige Parameterliste mit dem aus dem jeweiligen Testmodul abstrahierten Testmodul verknüpft wird. Weiterhin können in den Parameterlisten auch Relationen zwischen einzelnen Parametern abgespeichert werden, um diese dann beim Erstellen von Testszenarien nutzen zu können. Anhand der Parameterlisten können
- 20 außerdem beim Erstellen von Testszenarien rasch beispielsweise nicht sinnvolle, nicht funktionsfähige und/oder nicht zielführende Testszenarien identifiziert und eliminiert werden, wenn z.B. aufgrund einer Reihenfolge und/oder Kombination von Testmodulen, etc. erforderliche Parameter z.B. nicht verfügbar sind. Weiterhin können die Parameterlisten dazu genutzt werden, entsprechende, weitere Schritte in das Testszenario zu inkludieren, um diese
- 25 Parameter – insbesondere bei der Anwendung des aus dem jeweiligen Testszenario abgeleiteten Testfall – zu ermitteln. Die Erstellung und Durchführung dieser Schritte, wie z.B. Ermitteln des Parameters oder Parameterwerts aus dem zu testenden funktionalen System, etc., kann beispielsweise automatisiert erfolgen.

- Zweckmäßigerweise wird ein erstelltes, abstraktes Testszenario nach Platzhaltervariablen
- 30 von erforderlichen Parametern durchsucht, welche für eine Durchführung des aus dem Testszenario abgeleiteten Testfalls vom zu testenden funktionalen System und/oder von zumindest einem der vorhergehenden Testmodule dem jeweiligen Testmodul bereitzustellen sind.

- Idealerweise werden zum Auffinden der Platzhaltervariablen der erforderlichen Parameter im jeweiligen Testszenario die mit den jeweiligen abstrahierten Testmodulen des jeweiligen
- 35 Testszenarios verknüpften Parameterlisten herangezogen. Auf diese Weise können die

Platzhaltervariablen und gegebenenfalls auch Relationen zwischen diesen rasch und auf einfache Weise erkannt und genutzt werden.

Weiterhin ist es von Vorteil, wenn die abstrahierten Testmodule und/oder abstrakte Testszenarien und/oder Teile von abstrakten Testszenarien im Testsystem, insbesondere in einer Datenbank, hinterlegt werden. Damit können diese für nachfolgende Sicherheitsüberprüfungen von zu testenden funktionalen Systemen wiederverwendet werden. Einerseits können dann Testfälle rascher an neu zu testende funktionale Systeme angepasst werden, andererseits wird dadurch eine Testmodule- und -szenarien-Datenbank aufgebaut und damit das erfindungsgemäße Verfahren weiter beschleunigt. Dabei kann es auch günstig sein, wenn vor einem Hinterlegen des jeweiligen abstrahierten Testmoduls geprüft wird, ob das jeweilige abstrahierte Testmodul bereits im Testsystem, insbesondere in der Datenbank, hinterlegt ist. Damit wird sehr leicht verhindert, dass abstrahierte Testmodule mehrfach abgespeichert werden.

Idealerweise wird im Testsystem zu jedem abstrakten Testszenario eine Information hinterlegt, bei welchem funktionalen System das jeweilige Testszenario angewendet werden kann. Damit können für Sicherheitsüberprüfungen zu testender funktionaler Systeme rasch und mit geringem Aufwand passende Testszenarien gefunden werden.

Eine zweckmäßige Ausgestaltung des Verfahrens sieht vor, dass die systemspezifischen Informationen, Daten und Parameter des zu testenden funktionalen Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen der jeweiligen Testszenarien vom Testsystem, insbesondere von der Datenbank, in welchem Erfahrungswerten aus vorhergehenden Sicherheitsüberprüfungen eines selben oder ähnlichen funktionalen Systems hinterlegt sind, und/oder von einem Systemhersteller zur Verfügung gestellt werden und/oder mittels maschinellern Lernens aus Erfahrungswerten vorhergehender Sicherheitsüberprüfungen ermittelt werden. Als systemspezifische Informationen, Daten und Parameter des zu testenden funktionalen Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen der jeweiligen Testszenarien können allerdings auch Werte verwendet werden, welche aus einer Analyse des jeweils zu testenden funktionalen Systems ermittelt werden.

Die oben angeführte Aufgabe wird weiterhin auch durch ein Testsystem sowie ein Computerprodukt gelöst. Das Testsystem weist zumindest eine Computereinheit auf, welche zur Ausführung des erfindungsgemäßen Verfahrens eingerichtet ist. Zweckmäßigerweise kann das Testsystem auch eine Speichereinheit, wie z.B. eine Datenbank, umfassen oder mit einer Speichereinheit, wie z.B. einer Datenbank, verbunden sein. In diese Speichereinheit ist idealerweise dazu eingerichtet, die abstrahierten Testmodule, die abstrakten Testszenarien und/oder Teile von abstrakten Testszenarien sowie Erfahrungswerte aus vorhergehenden

Sicherheitsüberprüfungen eines selben oder ähnlichen funktionalen Systems abzuspeichern. Das Testsystem kann auch mit einem System für maschinelles Lernen verbunden sein oder ein solche aufweisen, mit welchen die systemspezifischen Informationen, Daten und Parameter des zu testenden funktionalen Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen der jeweiligen Testszenarien aus Erfahrungswerten, etc. ermittelt werden können.

Das Computerprogrammprodukt ist in vorteilhafter Weise in das Testsystem – beispielsweise in die zumindest eine Computereinheit ladbar, und umfasst Befehl, welche bei einer Ausführung des Computerprogrammprodukts im Testsystem das Computerprogrammprodukt dazu veranlassen, das erfindungsgemäße Verfahren auszuführen.

Kurzbeschreibung der Figuren

Die gegenständliche Erfindung wird nachfolgend unter Bezugnahme auf die Figuren 1 bis 2c näher erläutert, die beispielhaft, schematisch und nicht einschränkend vorteilhafte Ausgestaltungen der Erfindung zeigen. Dabei zeigt

Fig. 1 einen Ablauf des computerimplementierten Verfahrens zum Adaptieren von Testfällen einer Sicherheitsprüfung eines zu testenden funktionalen Systems

Fig. 2a einen Abstrahierungsschritt des erfindungsgemäßen Verfahrens anhand eines konkreten Ausführungsbeispiels

Fig. 2b einen Syntheseschritt des erfindungsgemäßen Verfahrens anhand des konkreten Ausführungsbeispiels; und

Fig. 2c einen Konkretisierungsschritt des erfindungsgemäßen Verfahrens anhand des konkreten Ausführungsbeispiels.

Ausführung der Erfindung

Figur 1 zeigt einen beispielhaften Ablauf eines computerimplementierten Verfahrens, mit welchem Testfälle TC1, die bereits bei Sicherheitsüberprüfungen von funktionalen Systemen, durch welche beispielsweise Mobilitätsanwendungen (z.B. Kraftfahrzeuge, etc.) realisiert werden und/oder welche in Mobilitätsanwendungen (z.B. Autonomes Fahren, Vehicle-to-Vehicle-Kommunikation, Vehicle-to-X-Kommunikation, etc.) zum Einsatz kommen, angewendet wurden, für eine Sicherheitsprüfung eines zu testenden funktionalen Systems (z.B. Fahrzeug, Fahrzeugsystem, etc.) automatisiert adaptiert werden können. Als zu testendes funktionales System kann dabei ein neu entwickeltes funktionales System, eine Weiterentwicklung

und/oder Verbesserung eines bestehenden funktionalen Systems oder auch ein funktionales System, bei welchen z.B. einzelne Komponenten (z.B. Hardwareeinheiten, Softwareeinheiten, etc.) getauscht oder upgedatet wurden, betrachtet werden.

Das Verfahren kann auf einem zugehörigen Testsystem ausgeführt werden, wobei das Testsystem zumindest eine Computereinheit aufweist. Das Testsystem stellt ein aktives System dar, das die Sicherheitsüberprüfung eines zu testenden Systems gemäß einem Computerprogrammprodukt durchführt, welches z.B. in einen internen Speicher des Testsystem geladen und z.B. mittels der zumindest einen Computereinheit bzw. eines Prozessors durchführt werden kann. Dazu umfasst das Computerprogrammprodukt Befehle, welche bei einer Ausführung durch das Testsystem das Computerprogrammprodukt zur Durchführung der Schritte des erfindungsgemäßen Verfahrens veranlassen.

Vom Testsystem werden entsprechende Testfälle TC1, TC21, ..., TC23 durchführt, um beispielsweise Schwachstellen des zu testenden funktionalen Systems aufzudecken. Dazu kann das Testsystem beispielsweise Schnittstellen und Umgebung (z.B. Prüfstand bzw. Testbed für Mobilitätsanwendungen, etc.) zur Verfügung stellen und gemeinsam mit den zu testenden funktionalen Systemen eine Testumgebung bilden, in welcher dann das erfindungsgemäße Verfahren durchgeführt wird. In der Testumgebung bzw. für das Testsystem sind Testfälle TC1, welcher bereits bei Sicherheitsüberprüfungen anderer funktionaler Systeme angewendet wurden, verfügbar. D.h. das Testsystem kennt Testfälle TC1, welche bereits durchgeführten Sicherheitsüberprüfungen von in Mobilitätsanwendungen verwendeten funktionalen System angewendet wurden und/oder für derartige Sicherheitsüberprüfungen verwendbar sind. Ein Testfall TC1, TC21, ..., TC23 ist dabei ein auf das jeweils zu testende, funktionale System angepasstes Testszenario TS1, TS2, TS3 und kann beispielsweise auf dem Testsystem und dem zu testenden funktionalen System im Rahmen der Sicherheitsüberprüfung ausgeführt werden. Als Testszenario TS1, TS2, TS3 wird im Kontext dieser Offenbarung eine abstrakte Beschreibung eines Testfalls TC1, TC21, ..., TC23 gesehen, welche definiert, was mit welchen Mitteln getestet werden soll, und in welchen einzelne Testszenario-Phasen beschrieben sein können. Testszenarien TS1, TS2, TS3 für funktionale Systeme können beispielsweise von Sicherheitsanalysen und/oder aus Sicherheitsanforderungen gewonnen werden.

Um einen konkreten Testfall TC1, welcher bereits mittels des Testsystems bei einem funktionalen System angewendet wurden, für ein weiteres funktionales System, welches aktuell getestet werden soll, zu adaptieren, werden vom Testsystem die in der Folge beschriebenen und in Figur 1 beispielhaft dargestellten Schritte durchgeführt.

Dazu werden in einem Modularisierungsschritt 101 bereits zum Testen von funktionalen Systemen verwendete Testfälle TC1 vom Testsystem analysiert und in einzelne Testmodule T11, T12, T13, T14 unterteilt bzw. zerlegt. Ein Testmodul T11, T12, T13, T14 kann beispielsweise ein Testskript sein, welches bei Ausführung einen gezielten Angriff auf das funktionale System darstellt. Gegebenenfalls kann das Testmodul T11, T12, T13, T14 auch nur aus einem oder mehreren Befehlen bestehen.

In einem folgenden Abstrahierungsschritt 102 werden dann die aus den Testfällen TC1 ermittelten Testmodule T11, T12, T13, T14 abstrahiert. Dabei werden in den Testmodulen T11, T12, T13, T14 jeweils enthaltene, systemspezifische Informationen, Daten und Parameter des funktionalen Systems, auf welches der jeweilige Testfall TC1 angewendet wurde, durch abstrakte Platzhaltervariablen ersetzt. Damit werden im Abstrahierungsschritt 102 aus den konkreten Testmodulen T11, T12, T13, T14, welche für einen jeweiligen Testfall TC1 systemspezifisch angepasst sind, abstrahierte Testmodule a1, a2, a3, a4. Ein abstrahiertes Testmodul a1, a2, a3, a4 stellt damit einen system-agnostischen Testbaustein dar, welcher eine einzelne Phase eines Testszenarios TS1, TS2, TS3 beschreibt und agnostisch in Bezug auf ein spezielles funktionales System ist.

Die abstrahierten Testmodule a1, a2, a3, a4 können beispielsweise im Testsystem, z.B. in einer zugehörigen Datenbank, hinterlegt werden. Die Datenbank kann dazu z.B. in einer internen Speichereinheit des Testsystems oder in einer Speichereinheit, welche mit dem Testsystem verbunden ist, angeordnet sein. Dabei kann vor einem Hinterlegen des jeweiligen abstrahierten Testmodule a1, a2, a3, a4 geprüft werden, ob dieses Testmodul a1, a2, a3, a4 bereits im Testsystem bzw. in der zugehörigen Datenbank gespeichert ist. Um die abstrahierten Testmodule a1, a2, a3, a4 in einer adäquaten Form im Testsystem bzw. in der Datenbank speichern zu können, wird für die abstrahierten Testmodule a1, a2, a3, a4 eine entsprechende Beschreibungssprache – wie z.B. eine so genannte Domain Specific Language oder kurz: DSL – verwendet. Eine Domain Specific Language ist eine formale Sprache, welche für Interaktionen zwischen z.B. Menschen und digital arbeitenden Einheiten (z.B. Computer) für ein bestimmtes Problemfeld – der so genannten Domäne bzw. Domain – entworfen und implementiert wird.

Weiterhin können im Abstrahierungsschritt 102 die Testmodule T11, T12, T13, T14 noch vor dem Abstrahieren nach erforderlichen Parametern durchsucht werden. Erforderliche Parameter sind beispielsweise Parameter, welcher für eine sinnvolle Ausführung des jeweiligen Testmodules T11, T12, T13, T14 unbedingt erforderlich sind. Diese Parameter können beispielsweise vom zu testenden funktionalen System oder von einem der im jeweiligen Testfall TC1 vorhergehenden Testmodule T11, T12, T13, T14 zur Verfügung gestellt werden. Sie können aber auch aus einer externen Quelle, wie z.B. einer Datenbank, maschinenlesbaren

Spezifikation, etc. stammt oder als manueller Input eingegeben werden. Solche Parameter wären z.B. Netzwerkadressen eines Testziels, eine Identifikation und/oder eine konkrete Mapping-Adresse, etc., welche beim Aussenden eines bestimmten Befehls über eine Komponente oder an eine Komponente des zu testenden funktionalen Systems für die Ausführung des Testmodul T11, T12, T13, T14 vorhanden sein müssen, oder Konfigurationsdaten. Diese Parameter definieren beispielsweise Eigenschaften (in einem Schritt) des jeweiligen Testmoduls T11, T12, T13, T14 – wie z.B. zu welcher Netzwerkadresse oder zu welcher Portnummer eine bestimmte Nachricht gesendet werden soll oder wie lange diese Nachricht sein soll.

Erforderliche Parameter können in jedem Testmodul T11, T12, T13, T14 vorkommen und für das jeweilige Testmodul T11, T12, T13, T14 spezifisch sein. Weiterhin kann es aber auch erforderliche Parameter geben, welche zwangsläufig für alle Testmodule T11, T12, T13, T14 z.B. eines Testfalls TC1 bzw. auch weitere Testfälle TC21, TC22, TC23 für das jeweilige funktionale System gleich sind, wie z.B. Portnummer. Derartige Parameter können beispielsweise einmal für alle Testmodule T11, T12, T13, T14 eines Testfalls TC1 als „globale“ Parameter definiert werden.

Die ermittelten, erforderlichen Parameter des jeweiligen Testmoduls T11, T12, T13, T14 können dann beispielsweise in einer Parameterliste zusammengefasst werden. Zusätzlich zu den ermittelten, erforderlichen Parametern des jeweiligen Testmoduls T11, T12, T13, T14 können in der Parameterliste auch Relationen zwischen einzelnen, erforderlichen Parametern abgespeichert werden, wie z.B. das einzelne Parameter gleich Werte aufweisen müssen oder sich auch einer Kombination anderer Parameter ergeben. Die Parameterlisten können dann mit dem aus dem jeweiligen Testmodul T11, T12, T13, T14 generierten, abstrahierten Testmodul a1, a2, a3, a4 verknüpft werden. Dazu kann z.B. die jeweilige Parameterliste im Testsystem bzw. in der zugehörigen Datenbank mit dem jeweiligen abstrahierten Testmodul a1, a2, a3, a4 abgespeichert werden.

In der Folge werden dann in einem Syntheseschritt 103 durch Kombination der abstrahierten Testmodule a1, a2, a3, a4 abstrakte Testszenarien TS1, TS2, TS3 für die Sicherheitsüberprüfung des jeweils zu testenden funktionalen Systems erstellt. Dabei können abstrahierte Testmodule a1, a2, a3, a4, a5, welche aus verschiedenen – beispielsweise bereits angewendeten und/oder anwendbaren – Testfällen TC1 generiert wurden, zu neuen Testszenarien TS1, TS2, TS3 für die Sicherheitsüberprüfung des zu testenden funktionalen Systems kombiniert werden. Auch die erstellten Testszenarien TS1, TS2, TS3 oder zumindest Teil der erstellten Testszenarien TS1, TS2, TS3, TS4 können im Testsystem bzw. in der zugehörigen Datenbank beispielsweise für eine Wiederverwendung bei ähnlichen oder denselben Testanforderungen abgespeichert werden. Dazu kann auch für die Testszenarien TS1, TS2, TS3

wie für die abstrahierten Testmodule a1, a2, a3, a4, a5 die entsprechende Beschreibungssprache (z.B. Domain Specific Language) verwendet werden.

Weiterhin kann im Syntheseschritt 103 jedes neu erstellte Testszenario TS1, TS2, TS3 nach den erforderlichen Parametern durchsucht werden, welche bei der Durchführung eines aus dem Testszenario TS1, TS2, TS3 abgeleiteten Testfall TC21, TC22, TC23 einem Testmodul T21, T22, T23, T24, T25 vom zu testenden funktionalen System und/oder zumindest einem der vorhergehenden Testmodule T21, T22, T23, T24, T25 zu ermitteln bzw. bereitzustellen sind. Auf diese Weise können beispielsweise bereits im Syntheseschritt 103 nicht zielführende, nicht funktionsfähige und/oder nicht sinnvolle Testszenarien TS1, TS2, TS3 (z.B. aufgrund der Reihenfolge der abstrahierten Testmodule a1, a2, a3, a4, a5) erkannt und eliminiert werden. Um diese erforderlichen Parameter im jeweiligen Testszenario TS1, TS2, TS3 rascher erkennen und auffinden zu können, können beispielsweise die Parameterlisten herangezogen werden, welche im Abstrahierungsschritt 102 erstellt wurden. Es werden dazu die Parameterlisten herangezogen, welche mit den jeweils im Testszenario TS1, TS2, TS3 verwendeten, abstrahierten Testmodulen a1, a2, a3, a4, a5 verknüpft sind.

In einem folgenden Konkretisierungsschritt 104 werden dann aus den abstrakten Testszenarien TS1, TS2, TS3 konkrete Testfälle TC21, TC22, TC23 für die Sicherheitsüberprüfung des zu testenden funktionalen Systems abgeleitet. Dazu werden die jeweiligen abstrakten Platzhaltervariablen in den abstrahierten Testmodulen a1, a2, a3, a4, a5 der Testszenarien TS1, TS2, TS3 mit Information, Daten und Parametern des zu testenden funktionalen Systems belegt. Die systemspezifischen Informationen, Daten und Parameter des aktuell zu testenden Systems können dazu beispielsweise von einem Systemhersteller oder von einem Auftraggeber der durchzuführenden Sicherheitsüberprüfung zur Verfügung gestellt werden. Alternativ oder zusätzlich, können dazu auch Erfahrungswerte aus vorangegangenen Sicherheitsüberprüfungen desselben oder eines ähnlichen funktionalen Systems herangezogen werden. Diese Erfahrungswerte können z.B. ebenfalls im Testsystem bzw. in der zugehörigen Datenbank hinterlegt sein. Alternativ oder zusätzlich, können die systemspezifischen Informationen, Daten und Parameter zumindest teilweise anstatt aus der Datenbank auch von einem Machine-Learning-System stammen, welche die Informationen, Daten und Parameter auf Basis von Erfahrungswerten generiert. Weiterhin können systemspezifische Informationen, Daten und Parameter des zu testenden Systems, mit welchen die abstrakten Platzhaltervariablen in den abstrahierten Testmodulen a1, a2, a3, a4, a5 der Testszenarien TS1, TS2, TS3 belegt werden können, auch aus einer Analyse des aktuell zu testenden funktionalen Systems (z.B. Scans am System, Portscan, etc.) oder aus verfügbaren Herstellerdaten ermittelt werden.

Nach dem Konkretisierungsschritt 104 werden die aus den abstrakten Testszenarien TS1, TS2, TS3 abgeleiteten, konkreten Testfälle TC21, TC22, TC23 in einem Anwendungsschritt 105 auf das zu testende funktionale System angewendet. Im Anwendungsschritt 105 kann beispielsweise während der laufenden Sicherheitsüberprüfung dokumentiert werden, welche konkreten Testfälle TC21, TC22, TC23 beispielweise besonders gut zum Aufdecken von Schwachstellen geeignet sind, welche Testfälle TC21, TC22, TC23 beispielweise das beste Ergebnis liefern, welche Testfälle TC21, TC22, TC23 beispielweise keine besonders guten Ergebnisse liefern, welche Testfälle TC21, TC22, TC23 beispielweise auf dem zu testenden funktionalen System nicht oder nur fehlerhaft funktionieren, etc.

Nach dem Anwendungsschritt 105 können dann während der Durchführung der Sicherheitsüberprüfung gesammelten Informationen zu den jeweiligen Testfällen TC21, TC22, TC23 in einem Rückmeldungsschritt 106 an das Testsystem übermittelt werden. Dazu kann beispielsweise zu jedem Testfall TC21, TC22, TC23 eine Rückmeldung erstellt werden, in welcher beispielsweise Informationen zur Durchführung des jeweiligen Testfalls TC21, TC22, TC23 enthalten sind. Die Rückmeldungen können dann an das Testsystem übermittelt und dort z.B. mit den Testszenarien TS1, TS2, TS3 verknüpft werden, aus welchen die Testfälle TC21, TC22, TC23 abgeleitet wurden. Durch die Rückmeldungen können dann besonders gut funktionierende Testfälle TC21, TC22, TC23 oder nicht oder nur fehlerhaft funktionierende Testfälle TC21, TC22, TC23 identifiziert werden. Weiterhin kann zu jedem abstrakten Testszenario TS1, TS2, TS3 eine Information hinterlegt werden, bei welchem funktionalen System das jeweilige Testszenario TS1, TS2, TS3 bereits angewendet wurde. Damit können bei einer neuerlichen Sicherheitsüberprüfung desselben oder eines ähnlichen funktionalen Systems im Testsystem bereits erprobte und als erfolgreich bzw. gut funktionierende Testszenarien TS1, TS2, TS3 im Syntheseschritt 103 identifiziert und im Konkretisierungsschritt 104 wieder zum Generieren von Testfällen TC21, TC22, TC23 herangezogen werden. Weiterhin können auf diese Weise sehr einfach nicht funktionierende oder für bestimmte funktionale System nicht geeignete oder sinnlose Testszenarien TS1, TS2, TS3 (z.B. aufgrund der Testmodul-Reihenfolge, aufgrund von erforderlichen Parametern auf einem funktionalen System nicht ausführbar, etc.) aus dem Testsystem bzw. der zugehörigen Datenbank ausgeschieden werden. Das Testsystem kann dadurch laufend verbessert werden und aus bereits durchgeführten Sicherheitsüberprüfungen – insbesondere Konkretisierungsschritten 104 – lernen.

In den Figuren 2a bis 2c werden der Abstrahierungsschritt 102, der Syntheseschritt 103 und der Konkretisierungsschritt 104 des Verfahrens anhand eines Ausführungsbeispiels detaillierter beschrieben. Im Ausführungsbeispiel soll z.B. eine Sicherheitsprüfung für ein konkretes Fahrzeug, für welches z.B. nach einer Neuentwicklung einer oder mehrerer

Komponenten oder nach einem Update einer oder mehrerer Komponenten eine Sicherheitsüberprüfung durchgeführt werden. Dabei soll z.B. Schwachstellen bzw. Angriffspunkte für Cyber-Attacken auf dieses Fahrzeug ausfindig gemacht werden.

5 In Figur 2a ist beispielhaft der Abstrahierungsschritt 102 dargestellt. Dabei wird beispielsweise von einem beispielhaften konkreten Testfall TC1 ausgegangen, welcher bereits z.B. auf ein ähnliches funktionales System (z.B. Fahrzeug eines anderen Herstellers, etc.) oder auf dasselbe funktionale System (z.B. Fahrzeug vor einem Komponententausch oder -update) angewendet wurde. Der Testfall TC1 stellt beispielsweise einen konkreten Angriffsvektor auf ein funktionales System dar, welches bereits einer Sicherheitsüberprüfung unterzogen wurde. Mit diesem konkreten Angriffsvektor wurde beispielsweise über einen Angriff auf
10 eine externe Schnittstelle (z.B. Bluetooth, Funk, etc.) des funktionalen Systems versucht, bis zur Motorsteuerung des funktionalen Systems bzw. Fahrzeugs zu gelangen.

Der beispielhafte Testfall TC1 kann z.B. im Modularisierungsschritt 101 in vier Testmodule T11, T12, T13, T14 unterteilt werden. Dabei entspricht z.B. jedes Testmodul T11, T12, T13,
15 T14 einem konkreten Angriff auf eine Komponente oder auf ein Subsystem des getesteten funktionalen Systems bzw. Fahrzeugs. Das erste Testmodul T11 kann beispielsweise einen Angriff auf eine externe Schnittstelle des Fahrzeugs darstellen, mit welchen z.B. existierende externe Schnittstellen (z.B. drahtlose Kommunikationsschnittstelle, Sensoren, etc.) ausfindig gemacht werden. Durch ein zweites Testmodul T12 wird versucht, nach einem erfolgreichen
20 Angriff mittels des ersten Testmoduls T11 z.B. auf das Infotainment-System des Fahrzeugs zu gelangen. Bei erfolgreicher Durchführung des zweiten Testmoduls T12 versucht ein drittes Testmodul T13 beispielsweise das fahrzeuginterne Netzwerk des Fahrzeugs anzugreifen bzw. zu nutzen. Ist das dritte Testmodul T13 damit erfolgreich, so wird beispielsweise in einem vierten Testmodul T14 versucht, die Motorsteuerung des Fahrzeugs zu übernehmen.

25 Im Abstrahierungsschritt 102 werden nun die konkreten Testmodule T11, T12, T13, T14, in welche der konkrete Testfall TC1 unterteilt wurde, abstrahiert. D.h., es werden im Testsystem aus den Testmodulen T11, T12, T13, T14 alle enthaltenen, systemspezifischen Informationen, Daten und Parameter des mit dem Testfall TC1 getesteten Fahrzeugs entfernt und durch abstrakte Platzhaltervariablen ersetzt. Aus den Testmodulen T11, T12, T13, T14 werden damit abstrahierte Testmodule a1, a2, a3, a4, welche die jeweiligen Angriffe abstrakt be-
30 schreiben. So kann beispielsweise ein aus dem ersten Testmodul T11 abgeleitetes abstrahiertes Testmodul a1 die Beschreibung enthalten, nach einer drahtlosen Schnittstelle eines (beliebigen) funktionalen Systems zu suchen und mit dieser eine Verbindung herzustellen. Ein aus dem zweiten Testmodul T12 abgeleitetes, zweites abstrahiertes Testmodul a2 kann
35 z.B. die Beschreibung enthalten, nach erfolgreicher Verbindung mit der drahtlosen Schnittstelle eine Verbindung mit einer Komponente oder einem Subsystem des funktionalen

Systems herzustellen, welches bzw. welche über die gefundene Schnittstelle erreichbar ist. In einem von dritten Testmodul T13 abgeleiteten, dritten abstrahierten Testmodul a3 wird z.B. eine Nutzung eines internen Netzwerks des funktionalen Systems abstrakt beschrieben. Von einem vierten abstrahierten Testmodul a4 kann z.B. eine Übernahme einer Steuerung des funktionalen Systems abstrakt beschrieben werden. Die abstrahierten Testmodule a1, a2, a3, a4 können dann z.B. im Testsystem bzw. in der zugehörigen Datenbank hinterlegt werden, sofern diese abstrahierten Testmodule a1, a2, a3, a4 nicht bereits im Testsystem bzw. in der zugehörigen Datenbank gespeichert sind.

Weiterhin können die Testmodule T11, T12, T13, T14 noch vor dem Abstrahieren auf erforderliche Parameter durchsucht werden und entsprechend testmodulspezifische Parameterlisten erstellt werden. Die jeweilige testmodulspezifische Parameterliste kann mit dem jeweiligen abstrahierten Testmodul a1, a2, a3, a4 verknüpft und mit diesem im Testsystem bzw. in der zugehörigen Datenbank abgelegt werden.

In Figur 2b ist der Syntheseschritt 103 des Verfahrens anhand des Ausführungsbeispiels wie z.B. einer Sicherheitsüberprüfung eines aktuell zu testenden Fahrzeugs dargestellt. Dabei werden die abstrahierten Testmodule a1, a2, a3, a4, a5 zu abstrakten Testszenarien TS1, TS2, TS3 kombiniert, wobei jedes Testszenario TS1, TS2, TS3 eine abstrakte Beschreibung eines möglichen Angriffsvektors darstellt. Zum Erstellen der Testszenarien TS1, TS2, TS3 können beispielsweise die im Abstrahierungsschritt 102 aus den konkreten Testfall TC1 abgeleiteten, abstrahierten Testmodule a1, a2, a3, a4 sowie weitere, abstrahierte Testmodule a5, welche aus Testfällen TC1 weiterer, bereits durchgeführter Sicherheitsüberprüfungen abstrahiert wurden, verwendet werden. Weiterhin können dafür auch abstrahierte Testmodule a5 herangezogen werden, welche bereits im Testsystem bzw. in der zugehörigen Datenbank gespeichert sind.

Im in Figur 2b dargestellten Syntheseschritt 103 werden beispielsweise die im Abstrahierungsschritt 102, welcher in Figur 2a dargestellt ist, generierten, abstrahierten Testmodule a1, a2, a3, a4 zu einem ersten Testszenario TS1 kombiniert. In einem weiteren, zweiten Testszenario TS2 werden z.B. das zweite, das dritte und das vierte abstrahierte Testmodul a2, a3, a4 aus dem konkreten Testfall TC1 mit einem fünften abstrahierten Testmodul a5 (z.B. einer abstrakten Beschreibung, dass eine Verbindung zu einer systeminternen Komponente (z.B. Gateway) aufgebaut und/oder diese verwendet werden soll) kombiniert, wobei das fünfte abstrahierte Testmodul a5 beispielsweise zwischen dem dritten und vierten abstrahierten Testmodule a3, a4 eingefügt wird. In einem weiteren, dritten Testszenario TS3 werden ebenfalls das zweite, das dritte und das vierte abstrahierte Testmodul a2, a3, a4 mit dem fünften abstrahierten Testmodul a5 kombiniert, wobei beim dritten Testszenario das

fünfte abstrahierte Testmodul a5 beispielsweise zwischen dem zweiten und dem dritten abstrahierten Testmodul a2, a3 angeordnet wird.

Im Syntheseschritt 103 können damit durch eine beliebige Kombination der abstrahierten Testmodule a1, a2, a3, a4, a5 vom Testsystem abstrakte Testszenarien TS1, TS2, TS3 erstellt werden. Um beispielsweise nicht zielführende oder sinnlose Kombinationen von abstrahierten Testmodulen a1, a2, a3, a4, a5 auszuschließen – wie z.B. einer Kombination, bei welcher z.B. das vierte abstrahierte Testmodule a4 (z.B. eine Übernahme der Steuerung des funktionalen Systems) vor dem ersten Testmodul a1 (z.B. Suche nach externen Schnittstellen des funktionalen Systems als Angriffspunkt) angeordnet wird, können die erstellten, abstrakten Testszenarien z.B. nach erforderlichen Parametern durchsucht werden bzw. die bei den jeweiligen abstrahierten Testmodulen a1, a2, a3, a4, a5 gespeicherten Parameterlisten ausgewertet werden. Weiterhin können zum Erstellen von abstrakten Testszenarien TS1, TS2, TS3 im Testsystem bzw. in der zugehörigen Datenbank verfügbare Vorschläge für Kombinationen von abstrahierten Testmodulen a1, a2, a3, a4, a5 zum Erstellen von Testszenarien TS1, TS2, TS3 genutzt werden. Weiterhin können die erstellten, abstrakten Testszenarien TS1, TS2, TS3 oder zumindest Teil davon im Syntheseschritt 103 im Testsystem bzw. in der zugehörigen Datenbank – beispielsweise als Vorschläge zum Erstellen von Testszenarien TS1, TS2, TS3 in späteren Syntheseschritten 103 – hinterlegt werden.

In Figur 2c ist der Konkretisierungsschritt 104 des Verfahrens anhand des konkreten Ausführungsbeispiels – z.B. der Sicherheitsüberprüfung eines aktuell zu testenden Fahrzeugs – dargestellt. Im Konkretisierungsschritt 104 werden aus den im Syntheseschritt 103 erstellten abstrakten Testszenarien TS1, TS2, TS3 konkrete Testfälle TC21, TC22, TC23 für das aktuell zu testende funktionale System – z.B. ein Fahrzeug mit zumindest einer neu entwickelten oder upgedateten Komponente – abgeleitet. Dabei werden die abstrakten Platzhaltervariablen in den abstrahierten Testmodulen a1, a2, a3, a4, a5 der jeweiligen Testszenarien TS1, TS2, TS3 mit systemspezifischen Informationen, Daten und Parametern des aktuell zu testenden funktionalen Systems bzw. Fahrzeugs belegt. Aus dem ersten Testszenario TS1 wird damit ein konkreter, erster Testfall TC21 für die Sicherheitsüberprüfung des aktuell zu testenden Fahrzeugs. Weiterhin werden damit das zweite Testszenario TS2 bzw. das dritte Testszenario TS3 zu einem konkreten, zweiten Testfall TC22 bzw. zu einem konkreten, dritten Testfall TC23. Die systemspezifischen Informationen, Daten und Parameter können vom Auftraggeber der Sicherheitsüberprüfung oder aus Herstellerdaten des Fahrzeugs stammen. Es können beispielsweise auch Erfahrungswerte aus früheren Sicherheitsüberprüfungen des Fahrzeugs oder ähnlicher Fahrzeuge genutzt werden oder Daten aus Analysen, etc. des aktuell zu testenden Fahrzeugs herangezogen werden.

Die im Konkretisierungsschritt 104 abgeleiteten Testfälle TC21, TC22, TC23 werden dann im Anwendungsschritt 105 für die Sicherheitsüberprüfung des aktuell zu testenden funktionalen Systems bzw. Fahrzeugs verwenden. Die Erkenntnisse und Erfahrungen aus der Anwendung der Testfälle TC21, TC22, TC23 können dann im Rückmeldungsschritt 106 in Form von

5 Rückmeldungen zu den Testfällen TC21, TC22, TC23 an das Testsystem übermittelt werden. Dort können diese mit den entsprechenden Testszenarien TS1, TS2, TS3 verknüpft und hinterlegt werden, um die Rückmeldungen wieder bei nachfolgenden Sicherheitsüberprüfungen von funktionalen Systemen nutzen zu können.

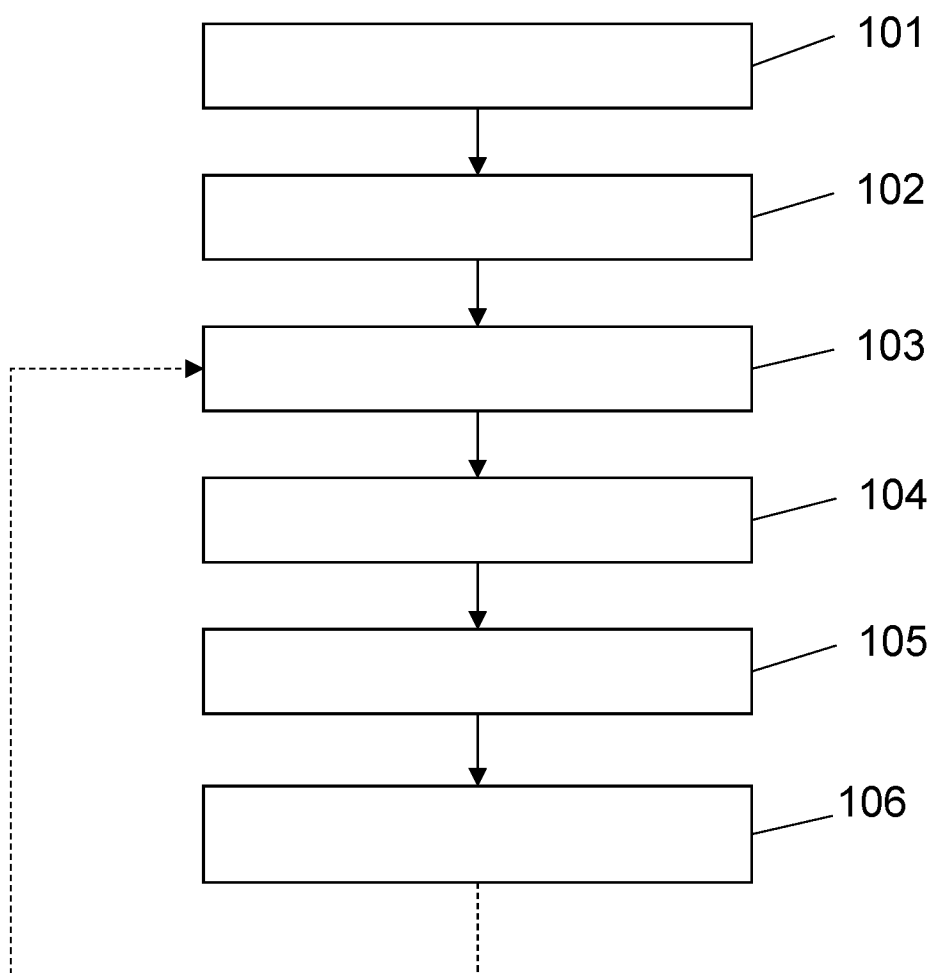
Patentansprüche

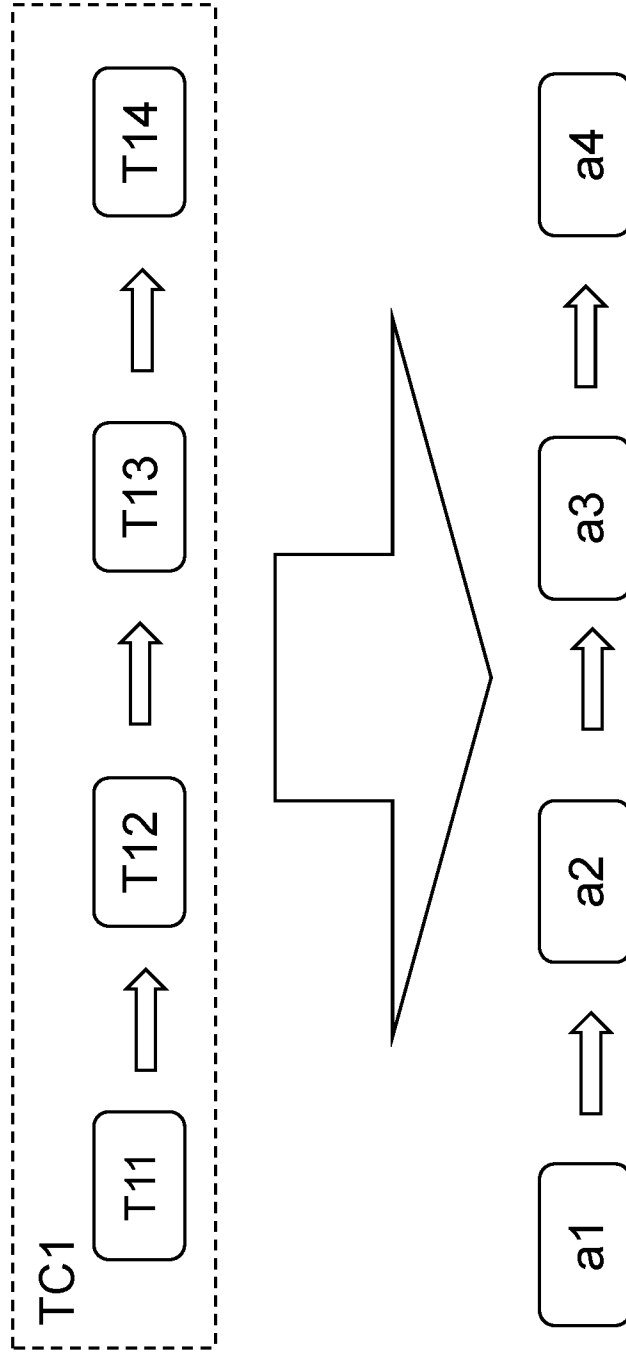
1. Computerimplementiertes Verfahren zum Adaptieren von Testfällen (TC1) für eine Sicherheitsüberprüfung eines zu testenden funktionalen Systems (105) einer Mobilitätsanwendung mittels eines Testsystems, wobei dem Testsystem Testfälle (TC1) zur Verfügung gestellt werden, welche bei Sicherheitsüberprüfungen von in Mobilitätsanwendungen verwendeten funktionalen Systemen angewendet wurden und/oder für derartige Sicherheitsüberprüfungen anwendbar sind, und wobei folgende Schritte ausgeführt werden:
 - Unterteilen der Testfälle (TC1) in Testmodule (T11, ..., T14) (101);
 - Abstrahieren der Testmodule (T11, ..., T14), wobei in den Testmodulen (T11, ..., T14) enthaltene, systemspezifische Informationen, Daten und Parameter der funktionalen Systeme, auf welche die jeweiligen Testfälle (TC1) angewendet wurden, in einem jeweiligen Testmodul (T11, ..., T14) durch abstrakte Platzhaltervariablen ersetzt werden (102);
 - Erstellen von abstrakten Testszenarien (TS1, TS2, TS3) für die Sicherheitsprüfung des zu testenden funktionalen Systems, wobei für ein Testszenario (TS1, TS2, TS3) die abstrahierte Testmodule (a1, ..., a5) kombiniert werden (103);
 - Ableiten von Testfällen (TC21, TC22, TC23) für das zu testende funktionale System aus den abstrakten Testszenarien (TS1, TS2, TS3), wobei die jeweiligen abstrakten Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) eines jeweiligen Testszenarios (TS1, TS2, TS3) mit entsprechenden systemspezifischen Informationen, Daten und Parametern des zu testenden funktionalen Systems belegt werden (104); und
 - Anwenden der aus den Testszenarien (TS1, TS2, TS3) abgeleiteten Testfälle (TC21, TC22, TC23) auf das zu testende funktionale System (105).
2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet, dass** nach Anwenden der aus den Testszenarien (TS1, TS2, TS3) abgeleiteten Testfälle (TC21, TC22, TC23) auf das zu testende funktionale System für jeden Testfall (TC21, TC22, TC23) eine Rückmeldung erstellt wird, **dass** die Rückmeldung an das Testsystem übermittelt wird und im Testsystem mit dem jeweiligen Testszenario (TS1, TS2, TS3) verknüpft wird (106), aus welchem der jeweilige Testfall (TC21, TC22, TC23) abgeleitet wurde.
3. Verfahren nach einem der vorangegangenen Ansprüche, **dadurch gekennzeichnet, dass** beim Abstrahieren die Testmodule (T11, ..., T14) nach erforderlichen Parametern durchsucht werden (102), welche für eine Durchführung des jeweiligen Testmoduls (T11, ..., T14) vom zu testenden funktionalen System (105) oder zumindest einem der vorhergehenden Testmodule (T11, ..., T14) zur Verfügung zu stellen sind, **und dass** die erforderlichen Parameter in einer Parameterliste im Testsystem hinterlegt werden, wobei die

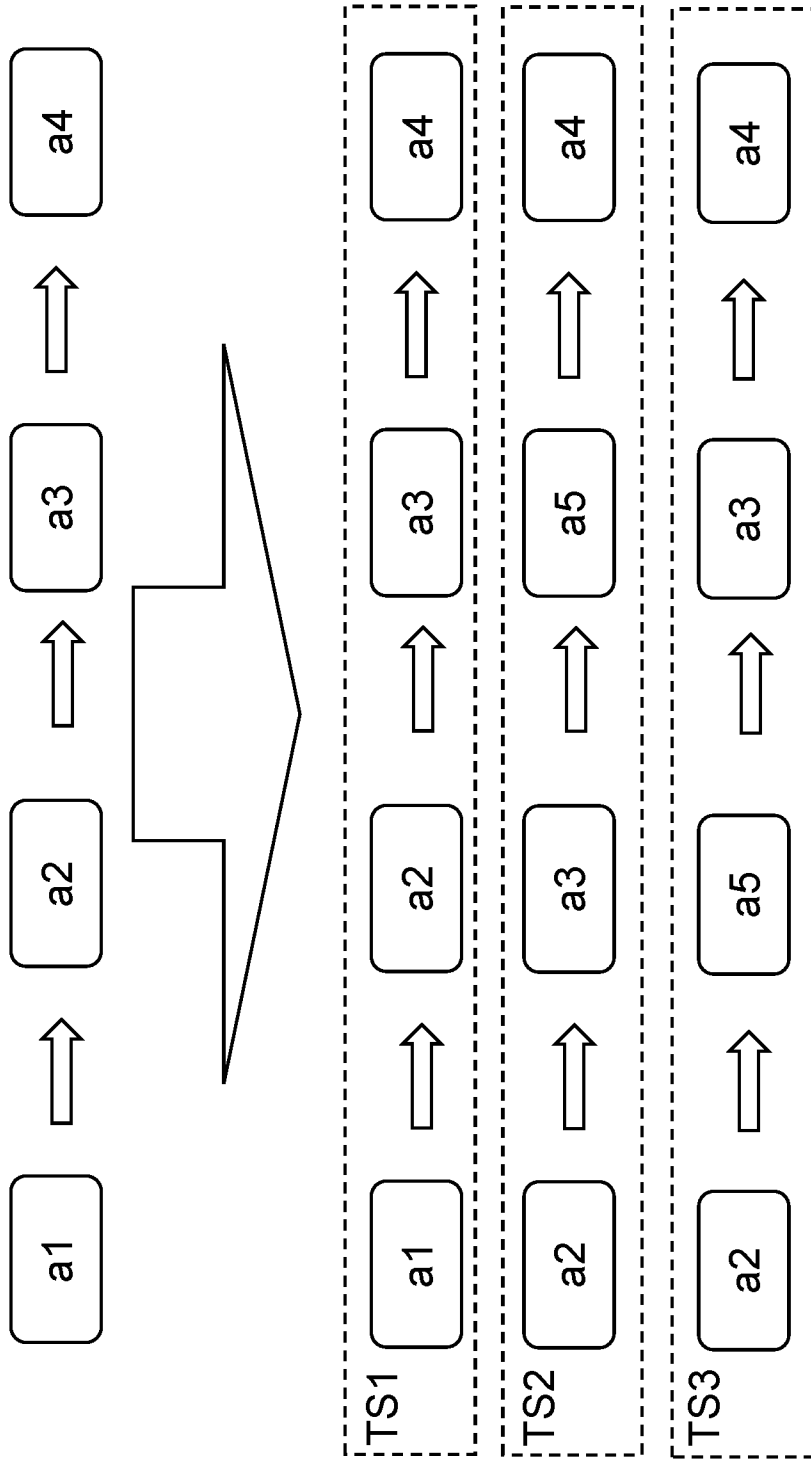
jeweilige Parameterliste mit dem aus dem jeweiligen Testmodul (T11, ..., T14) abstrahierten Testmodul (a1, ..., a5) verknüpft wird.

4. Verfahren nach einem der vorangegangenen Ansprüche, ***dadurch gekennzeichnet, dass*** ein erstelltes, abstraktes Testszenario (TS1, TS2, TS3) nach Platzhaltervariablen von erforderlichen Parametern durchsucht wird (103), welche für eine Durchführung des aus dem Testszenario (TS1, TS2, TS3) abgeleiteten Testfalls (TC21, TC22, TC23) vom zu testenden funktionalen System und/oder von zumindest einem der vorhergehenden Testmodule (T21, ..., T25) dem jeweiligen Testmodul (T21, ..., T25) bereitzustellen sind.
5. Verfahren nach Anspruch 4, ***dadurch gekennzeichnet, dass*** zum Auffinden der Platzhaltervariablen der erforderlichen Parameter im jeweiligen Testszenario (TS1, TS2, TS3) die mit den jeweiligen abstrahierten Testmodulen (a1, ..., a5) des jeweiligen Testszenarios (TS1, TS2, TS3) verknüpften Parameterlisten herangezogen werden (103).
6. Verfahren nach einem der vorangegangenen Ansprüche, ***dadurch gekennzeichnet, dass*** die abstrahierten Testmodule (a1, ..., a5) und/oder abstrakte Testszenarien (TS1, TS2, TS3) und/oder Teile von abstrakten Testszenarien (TS1, TS2, TS3) im Testsystem, insbesondere in einer Datenbank, hinterlegt werden (102, 103).
7. Verfahren nach Anspruch 6, ***dadurch gekennzeichnet, dass*** vor einem Hinterlegen des jeweiligen abstrahierten Testmoduls (a1, ..., a5) geprüft wird (102), ob das jeweilige abstrahierte Testmodul (a1, ..., a5) bereits im Testsystem, insbesondere in der Datenbank, hinterlegt ist.
8. Verfahren nach einem der vorangegangenen Ansprüche, ***dadurch gekennzeichnet, dass*** zu jedem abstrakten Testszenario (TS1, TS2, TS3) im Testsystem eine Information hinterlegt wird (103), bei welchen funktionalen Systemen das jeweilige Testszenario (TS1, TS2, TS3) angewendet worden ist.
9. Verfahren nach einem der vorangegangenen Ansprüche, ***dadurch gekennzeichnet, dass*** die systemspezifischen Informationen, Daten und Parameter des zu testenden funktionalen Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) der jeweiligen Testszenarien (TS1, TS2, TS3) vom Testsystem, insbesondere von der Datenbank, in welchem Erfahrungswerten aus vorhergehenden Sicherheitsüberprüfungen eines selben oder ähnlichen funktionalen Systems hinterlegt sind, und/oder von einem Systemhersteller zur Verfügung gestellt werden und/oder mittels maschinellem Lernens aus Erfahrungswerten vorhergehender Sicherheitsüberprüfungen ermittelt werden (104).

10. Verfahren nach einem der Ansprüche 1 bis 9, ***dadurch gekennzeichnet, dass*** als systemspezifischen Informationen, Daten und Parameter des zu testenden funktionalen Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) der jeweiligen Testszenarien (TS1, TS2, TS3) Werte verwendet werden (104), welche
 5 aus einer Analyse des zu testenden funktionalen Systems ermittelt werden.
11. Testsystem mit zumindest einer Computereinheit, welche dazu eingerichtet ist, das Verfahren nach einem der Ansprüche 1 bis 10 auszuführen.
12. Testsystem nach Anspruch 11, wobei das Testsystem weiterhin zumindest eine Speichereinheit umfasst und/oder mit einer Speichereinheit verbunden ist, welche zum Abspeichern abstrahierter Testmodule (a1, ..., a4, a5), abstrakter Testszenarien (TS1, TS2, TS3) und/oder von Teilen von abstrakten Testszenarien (TS1, TS2, TS3) sowie von Erfahrungswerten aus vorhergehenden Sicherheitsüberprüfungen eines selben oder ähnlichen funktionalen Systems eingerichtet ist.
 10
13. Computerprogrammprodukt, wobei das Computerprogrammprodukt in das Testsystem nach einem der Ansprüche 11 bis 12 geladen werden kann, und wobei das Computerprogrammprodukt Befehle umfasst, welche bei einer Ausführung durch das Testsystem das Computerprogrammprodukt veranlassen, das Verfahren nach einem der Ansprüche 1 bis 10 auszuführen.
 15

**Fig. 1**

**Fig. 2a**

**Fig. 2b**

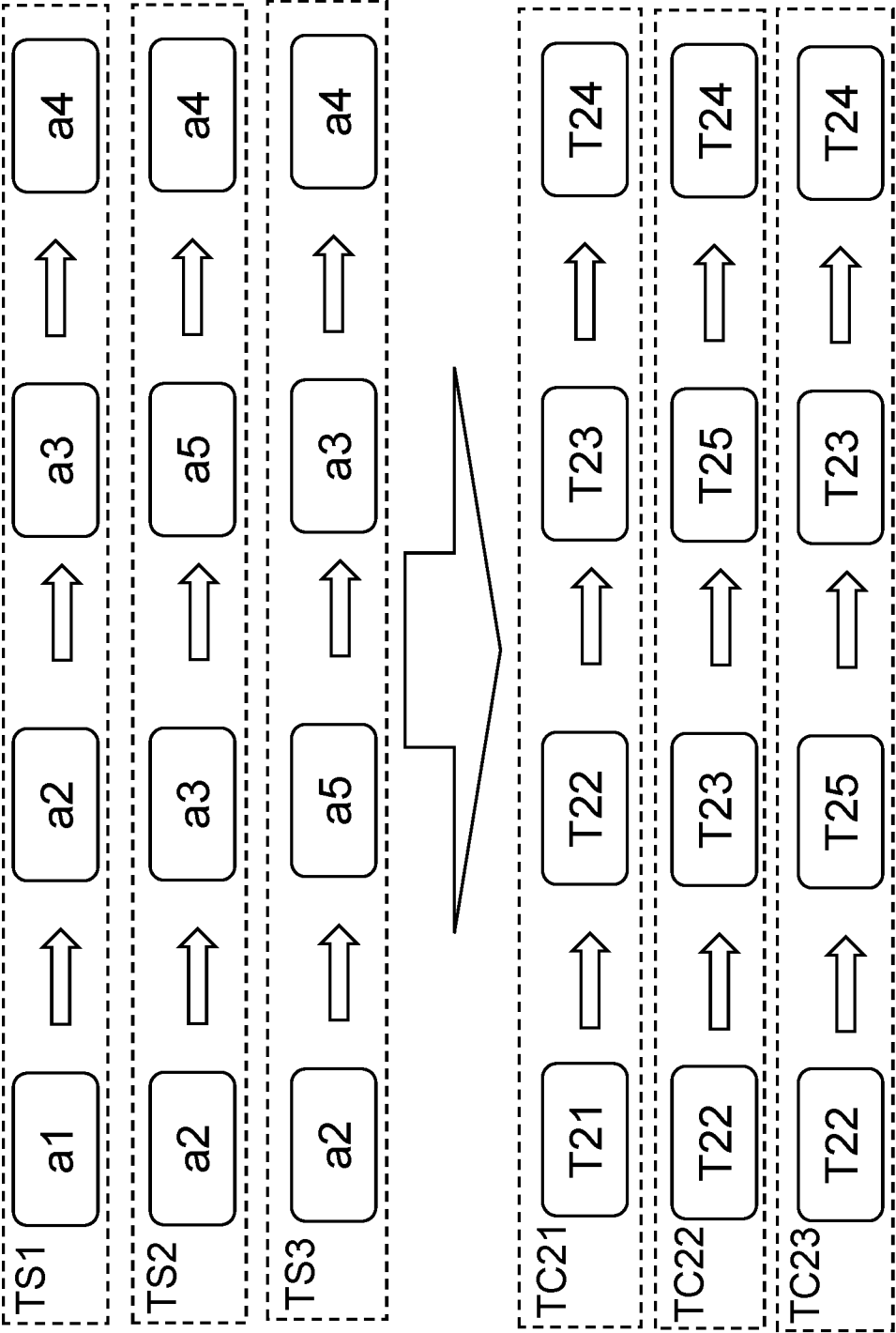


Fig. 2c

Klassifikation des Anmeldungsgegenstands gemäß IPC:

G06F 21/00 (2006.01); G06F 21/12 (2013.01); G06F 11/00 (2006.01); G06F 11/10 (2006.01)

Klassifikation des Anmeldungsgegenstands gemäß CPC:

G06F 21/00 (2018.05); G06F 21/12 (2015.11); G06F 11/00 (2018.05); G06F 11/1004 (2013.01); G06F 11/3684 (2013.01); G06F 2212/1052 (2013.01)

Recherchierter Prüfstoff (Klassifikation):

G06F; H04L

Konsultierte Online-Datenbank:

WPIAP; EPODOC; TXTEN; TXTDE; INSPEC; NPL; Internet

Dieser Recherchenbericht wurde zu den am 23.12.2022 eingereichten Ansprüchen 1-13 erstellt.

Kategorie*)	Bezeichnung der Veröffentlichung: Ländercode, Veröffentlichungsnummer, Dokumentart (Anmelder), Veröffentlichungsdatum, Textstelle oder Figur soweit erforderlich	Betreffend Anspruch
X	WO 2020247993 A1 (AVL LIST GMBH) 17. Dezember 2020 (17.12.2020) Siehe besonders Zusammenfassung; Fig. 1-3 und die dazugehörigen Figurenbeschreibungen; Seite 1, Zeile 1 bis Seite 30, Zeile 20; Patentansprüche 1 bis 12	1, 2, 4, 11-13
A		3, 5-10
A	EP 1622022 A1 (SIEMENS AG) 01. Februar 2006 (01.02.2006) Siehe besonders Fig. 1 und die dazugehörige Figurenbeschreibung; Zusammenfassung; Paragraphen 1 bis 41; Patentansprüche 1 bis 20	1-13
A	WO 2005071543 A2 (SYMPHONY SERVICES CORP) 04. August 2005 (04.08.2005) Siehe besonders Zusammenfassung; Fig. 1, 3, 4, 5, 7 und die dazugehörigen Figurenbeschreibungen; Seite 3, Zeile 1 bis Seite 4, Zeile 17; Patentansprüche 1 bis 49	1-13
A	Schuldt Fabian et al. "Effiziente systematische Testgenerierung für Fahrassistenzsysteme in virtuellen Umgebungen" 10/07/2013 abgerufen im Internet am 15.11.2023 unter dem Link URL:<http://www.digibib.tu-bs.de/?docid=00052570> Siehe besonders Seiten 1 bis 20	1, 11-13
A	Papadimitratos P. et al., "Secure vehicular communication systems: design and architecture," in IEEE Communications Magazine, vol. 46, no. 11, November 2008, doi: 10.1109/MCOM.2008.4689252. Siehe besonders Seiten 100 bis 109	1, 11, 13
Datum der Beendigung der Recherche: 15.11.2023		
Seite 1 von 1		Prüfer(in): KÖGL Christian

*) Kategorien der angeführten Dokumente:

- X Veröffentlichung von besonderer Bedeutung: der Anmeldungsgegenstand kann allein aufgrund dieser Druckschrift nicht als neu bzw. auf erfinderischer Tätigkeit beruhend betrachtet werden.
- Y Veröffentlichung von Bedeutung: der Anmeldungsgegenstand kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren weiteren Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist.

- A Veröffentlichung, die den allgemeinen Stand der Technik definiert.
- P Dokument, das von Bedeutung ist (Kategorien X oder Y), jedoch nach dem Prioritätstag der Anmeldung veröffentlicht wurde.
- E Dokument, das von besonderer Bedeutung ist (Kategorie X), aus dem ein „älteres Recht“ hervorgehen könnte (früheres Anmeldedatum, jedoch nachveröffentlicht, Schutz ist in Österreich möglich, würde Neuheit in Frage stellen).
- & Veröffentlichung, die Mitglied der selben Patentfamilie ist.

Neue Patentansprüche

1. Computerimplementiertes Verfahren zum Adaptieren von Testfällen (TC1) für eine Sicherheitsüberprüfung eines zu testenden funktionalen Systems (105) einer Mobilitätsanwendung mittels eines Testsystems, wobei dem Testsystem Testfälle (TC1) zur Verfügung gestellt werden, welche bei Sicherheitsüberprüfungen von in Mobilitätsanwendungen verwendeten funktionalen Systemen angewendet wurden und/oder für derartige Sicherheitsüberprüfungen anwendbar sind, und wobei folgende Schritte ausgeführt werden:
 - Unterteilen der Testfälle (TC1) in Testmodule (T11, ..., T14) (101);
 - Abstrahieren der Testmodule (T11, ..., T14), wobei in den Testmodulen (T11, ..., T14) enthaltene, systemspezifische Informationen, Daten und Parameter der funktionalen Systeme, auf welche die jeweiligen Testfälle (TC1) angewendet wurden, in einem jeweiligen Testmodul (T11, ..., T14) durch abstrakte Platzhaltervariablen ersetzt werden (102);
 - Erstellen von abstrakten Testszenarien (TS1, TS2, TS3) für die Sicherheitsprüfung des zu testenden funktionalen Systems, wobei für ein Testszenario (TS1, TS2, TS3) die abstrahierte Testmodule (a1, ..., a5) kombiniert werden und zu jedem abstrakten Testszenario (TS1, TS2, TS3) im Testsystem eine Information hinterlegt wird (103), bei welchen funktionalen Systemen das jeweilige Testszenario (TS1, TS2, TS3) angewendet worden ist;
 - Ableiten von Testfällen (TC21, TC22, TC23) für das zu testende funktionale System aus den abstrakten Testszenarien (TS1, TS2, TS3), wobei die jeweiligen abstrakten Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) eines jeweiligen Testszenarios (TS1, TS2, TS3) mit entsprechenden systemspezifischen Informationen, Daten und Parametern des zu testenden funktionalen Systems belegt werden (104); und
 - Anwenden der aus den Testszenarien (TS1, TS2, TS3) abgeleiteten Testfälle (TC21, TC22, TC23) auf das zu testende funktionale System (105).
2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet, dass** nach Anwenden der aus den Testszenarien (TS1, TS2, TS3) abgeleiteten Testfälle (TC21, TC22, TC23) auf das zu testende funktionale System für jeden Testfall (TC21, TC22, TC23) eine Rückmeldung erstellt wird, **dass** die Rückmeldung an das Testsystem übermittelt wird und im Testsystem mit dem jeweiligen Testszenario (TS1, TS2, TS3) verknüpft wird (106), aus welchem der jeweilige Testfall (TC21, TC22, TC23) abgeleitet wurde.
3. Verfahren nach einem der Ansprüche 1 bis 2, **dadurch gekennzeichnet, dass** beim Abstrahieren die Testmodule (T11, ..., T14) nach erforderlichen Parametern durchsucht werden (102), welche für eine Durchführung des jeweiligen Testmoduls (T11, ..., T14)

vom zu testenden funktionalen System (105) oder zumindest einem der vorhergehenden Testmodule (T11, ..., T14) zur Verfügung zu stellen sind, **und dass** die erforderlichen Parameter in einer Parameterliste im Testsystem hinterlegt werden, wobei die jeweilige Parameterliste mit dem aus dem jeweiligen Testmodul (T11, ..., T14) abstrahierten Testmodul (a1, ..., a5) verknüpft wird.

4. Verfahren nach einem der Ansprüche 1 bis 3, **dadurch gekennzeichnet, dass** ein erstelltes, abstraktes Testszenario (TS1, TS2, TS3) nach Platzhaltervariablen von erforderlichen Parametern durchsucht wird (103), welche für eine Durchführung des aus dem Testszenario (TS1, TS2, TS3) abgeleiteten Testfalls (TC21, TC22, TC23) vom zu testenden funktionalen System und/oder von zumindest einem der vorhergehenden Testmodule (T21, ..., T25) dem jeweiligen Testmodul (T21, ..., T25) bereitzustellen sind.
5. Verfahren nach Anspruch 4, **dadurch gekennzeichnet, dass** zum Auffinden der Platzhaltervariablen der erforderlichen Parameter im jeweiligen Testszenario (TS1, TS2, TS3) die mit den jeweiligen abstrahierten Testmodulen (a1, ..., a5) des jeweiligen Testszenarios (TS1, TS2, TS3) verknüpften Parameterlisten herangezogen werden (103).
6. Verfahren nach einem der Ansprüche 1 bis 5, **dadurch gekennzeichnet, dass** die abstrahierten Testmodule (a1, ..., a5) und/oder abstrakte Testszenarien (TS1, TS2, TS3) und/oder Teile von abstrakten Testszenarien (TS1, TS2, TS3) im Testsystem, insbesondere in einer Datenbank, hinterlegt werden (102, 103).
7. Verfahren nach Anspruch 6, **dadurch gekennzeichnet, dass** vor einem Hinterlegen des jeweiligen abstrahierten Testmoduls (a1, ..., a5) geprüft wird (102), ob das jeweilige abstrahierte Testmodul (a1, ..., a5) bereits im Testsystem, insbesondere in der Datenbank, hinterlegt ist.
8. Verfahren nach einem der Ansprüche 1 bis 7, **dadurch gekennzeichnet, dass** die systemspezifischen Informationen, Daten und Parameter des zu testenden funktionalen Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen (a1, ..., a5) der jeweiligen Testszenarien (TS1, TS2, TS3) vom Testsystem, insbesondere von der Datenbank, in welchem Erfahrungswerten aus vorhergehenden Sicherheitsüberprüfungen eines selben oder ähnlichen funktionalen Systems hinterlegt sind, und/oder von einem Systemhersteller zur Verfügung gestellt werden und/oder mittels maschinellern Lernens aus Erfahrungswerten vorhergehender Sicherheitsüberprüfungen ermittelt werden (104).
9. Verfahren nach einem der Ansprüche 1 bis 8, **dadurch gekennzeichnet, dass** als systemspezifischen Informationen, Daten und Parameter des zu testenden funktionalen

Systems zum Belegen der Platzhaltervariablen in den abstrahierten Testmodulen ($a_1, \dots, a_5$) der jeweiligen Testszenarien (TS1, TS2, TS3) Werte verwendet werden (104), welche aus einer Analyse des zu testenden funktionalen Systems ermittelt werden.

10. Testsystem mit zumindest einer Computereinheit, welche dazu eingerichtet ist, das Verfahren nach einem der Ansprüche 1 bis 9 auszuführen.
11. Testsystem nach Anspruch 12, wobei das Testsystem weiterhin zumindest eine Speichereinheit umfasst und/oder mit einer Speichereinheit verbunden ist, welche zum Abspeichern abstrahierter Testmodule ($a_1, \dots, a_4, a_5$), abstrakter Testszenarien (TS1, TS2, TS3) und/oder von Teilen von abstrakten Testszenarien (TS1, TS2, TS3) sowie von Erfahrungswerten aus vorhergehenden Sicherheitsüberprüfungen eines selben oder ähnlichen funktionalen Systems eingerichtet ist.
12. Computerprogrammprodukt, wobei das Computerprogrammprodukt in das Testsystem nach einem der Ansprüche 10 bis 11 geladen werden kann, und wobei das Computerprogrammprodukt Befehle umfasst, welche bei einer Ausführung durch das Testsystem das Computerprogrammprodukt veranlassen, das Verfahren nach einem der Ansprüche 1 bis 10 auszuführen.