Modulare Analyse praxisrelevanter Sicherheitsprotokolle in UC-Modellen*

Max Tuengerthal

ecsec GmbH, Michelau
max.tuengerthal@ecsec.de

Abstract: Sicherheitsprotokolle, wie SSL/TLS, sind aus unserem Alltag nicht mehr wegzudenken und wir verlassen uns stark auf ihre Sicherheit. Da immer wieder ernstzunehmende Angriffe auf diese Protokolle gefunden werden, ist ihre umfassende Sicherheitsanalyse unabdingbar. Die Komplexität dieser Protokolle und damit der Sicherheitsbeweise stellt jedoch eine große Herausforderung dar. Eine Möglichkeit, dieser Komplexität zu begegnen, ist eine *modulare Sicherheitsanalyse* in sogenannten "universal composability" (UC) Modellen, bei der das zu analysierende Protokoll in Komponenten zerlegt wird, die separat betrachtet werden können. Dieser erfolgversprechender Ansatz wurde bisher allerdings nur selten verwendet, um praxisrelevante Protokolle umfassend und im Detail zu analysieren. Dies hat verschiedene Gründe, auf die wir in dieser Arbeit eingehen werden.

Der Hauptbeitrag dieser Arbeit ist es, Methoden und Techniken zur modularen Sicherheitsanalyse in UC-Modellen bereitzustellen. Es wird damit eine solide Grundlage für eine umfassende Sicherheitsanalyse praxisrelevanter Sicherheitsprotokolle gelegt. Zudem wird der entwickelte Ansatz auf in der Praxis weit verbreitete Sicherheitsprotokolle angewendet, um dessen Nutzen zu demonstrieren.

1 Einleitung

Sicherheitsprotokolle, wie SSL/TLS, IEEE 802.11i (WPA/WPA2), SSH, IPsec, DNSSEC, Kerberos und viele mehr, sind aus unserem Alltag nicht mehr wegzudenken und wir verlassen uns stark auf ihre Sicherheit. Sie werden z. B. für den sicheren Schlüsselaustausch und die sichere Kommunikation in unsicheren Netzwerken, wie dem Internet, genutzt.

Der Entwuf von Sicherheitsprotokollen ist äußerst komplex und fehleranfällig, wie viele Schwachstellen und Angriffe auf aktuelle Sicherheitsprotokolle immer wieder belegen. Die genutzten Angriffsvektoren sind dabei vielfältig: Es gibt Angriffe auf die verwendeten kryptographischen Primitive (Verschlüsselung, Digitale Signaturen, Zertifikate, Nachrichtenauthentifizierungscodes, etc.), wie z. B. den berühmten Bleichenbacher Angriff [Ble98] auf SSL, der eine Schwachstelle in der RSA-PKCS#1-Verschlüsselung ausnutzt, den Angriff auf den CBC-Modus der Verschlüsselung SSL/TLS 1.0 [Bar06], die bekannten Angriffe

*Englischer Titel der Dissertation: "Analysis of Real-World Security Protocols in a Universal Composability Framework" [Tue13].

auf WEP (siehe z. B. [TWP07]) und Angriffe von Paterson et al. auf manche in SSH, IPsec und TLS verwendeten Verschlüsselungsmodi (siehe z. B. [PY06, APW09, DP10, AP13]). Darüber hinaus gibt es Angriffe auf das Protokoll an sich, also auf die logische Struktur des Protokolls. Zum Beispiel Angriffe auf Kerberos [CJS+08], eine Schwachstelle (die sogenannte *renegotiation vulnerability*) bei SSL/TLS [RD09] und Angriffe auf die Authentisierung im von Google verwendeten SAML Single Sign-On Protokoll [ACC+08] sowie im von Mozilla entwickelten Single Sign-On Protokoll *Persona* [FKS14].

Die Bedeutung von Sicherheitsprotokollen auf der einen und die zahlreichen Angriffe auf der anderen Seite zeigen deutlich, dass es unverzichtbar ist, Sicherheitsprotokolle systematisch und umfassend zu analysieren. Für die Protokollanalyse gibt es im Wesentlichen zwei Ansätze: den sogenannten *symbolischen* (oder *formalen*) Ansatz und den *kryptographischen* (oder *komplexitätstheoretischen*) Ansatz. Der kryptographische Ansatz kann weiter unterteilt werden in die Analyse in *Spiel-basierten* und *Simulations-basierten* (oder *"universal composability" (UC)*) Modellen. In jedem dieser Ansätze ist die Grundidee, dass das Sicherheitsprotokoll seine Sicherheitseigenschaft in folgendem Szenario erfüllen muss:

- Das Netzwerk wird komplett oder zum Teil von Angreifern kontrolliert.
- Protokollteilnehmer können mit den Angreifern kollaborieren und vom Protokoll abweichen.
- Mehrere Protokollsitzungen mit den gleichen oder verschiedenen Protokollteilnehmern können parallel ablaufen.

Dieses Szenario erfasst die in vielen Netzwerken, z. B. dem Internet, existierenden Bedrohungen und gilt als Standard in der Informationssicherheit und Kryptographie. Aufgrund dieses starken, aber realistischen, Angreifermodells ist der Entwurf und die Analyse von kryptographischen Protokollen sehr komplex und fehleranfällig. Praxisrelevante Sicherheitsprotokolle sind meist sehr umfangreich, da sie eine Vielzahl kryptographischer Primitive verwenden und aus verschiedenen Unterprotokollen bestehen, die miteinander interagieren. Ihre umfassende Sicherheitsanalyse stellt daher eine große Herausforderung dar.

Um die Komplexität der Analyse in den Griff zu bekommen, scheint eine *modulare* Sicherheitsanalyse unumgänglich. Dabei wird das zu analysierende Protokoll in Komponenten zerlegt, die separat analysiert werden können.

Ein erfolgversprechender Ansatz für die modulare Sicherheitsanalyse ist die Verwendung der genannten UC-Modelle [Can01, PW01]. Der Hauptbeitrag dieser Arbeit besteht darin, Methoden und Techniken zur modularen Sicherheitsanalyse in UC-Modellen bereitzustellen und damit eine Grundlage für eine umfassende Sicherheitsanalyse praxisrelevanter Protokolle zu legen.

Zwar gibt es auch bei Spiel-basierten Modellen, z. B. Bellare-Rogaway Modellen [BR93], Ansätze zur modularen Analyse, z. B. [HY08, MSW10, BFWW11], der Grad an Modularität, der dort bisher erreicht werden konnte, ist aber deutlich geringer als derjenige, der nötig erscheint und in dieser Arbeit angestrebt und erreicht wird. Auf symbolische Modelle gehen wir an dieser Stelle nicht ein und verweisen stattdessen auf Abschnitt 3.3.

Überblick. In Abschnitt 2 stellen wir den Stand der modularen Analyse in UC-Modellen zu Beginn des Dissertationsvorhabens vor. Die Hauptergebnisse der Doktorarbeit präsentieren wir in Abschnitt 3. In Abschnitt 4 fassen wir die Arbeit zusammen und diskutieren offene Fragen und mögliche zukünftige Arbeiten.

2 Modulare Sicherheitsanalyse in UC-Modellen

In UC-Modellen wird die Sicherheit von Protokollen mit Hilfe von sogenannten idealen Protokollen (auch ideale Funktionalitäten genannt) definiert. Die Grundidee ist, dass man ein Protokoll $\mathcal{P}$ *sicher* bzgl. einer idealen Funktionalität $\mathcal{F}$ nennt (man sagt auch: $\mathcal{P}$ *realisiert* $\mathcal{F}$), wenn es zu jedem Angriff auf $\mathcal{P}$ einen Angriff auf $\mathcal{F}$ gibt, so dass eine Umgebung nicht zwischen diesen beiden Angriffen unterscheiden kann: Da $\mathcal{F}$ per Definition keinen wirklichen Angriff zulässt, kann es auch auf $\mathcal{P}$ keine Angriffe geben. Protokolle, Angreifer und Umgebungen werden dabei als probabilistische polynomzeit-beschränkte Algorithmen (Turing Maschinen) modelliert.

Aufgrund dieser Sicherheitsdefinition können übergeordnete Protokollkomponenten auf Basis untergeordneter idealisierter Komponenten (idealer Funktionalitäten) entwickelt und analysiert werden. Kompositionstheoreme erlauben dann die Ersetzung der idealen Funktionalitäten durch ihre Realisierung, so dass ein Protokoll ohne idealisierte Komponenten entsteht. Anstatt also die Sicherheit eines komplexen Protokolls ohne idealisierte Komponenten direkt zu beweisen, können die untergeordneten Komponenten in Isolation und das komplexe Protokoll auf Basis der idealisierten Komponenten analysiert werden. Da die übergeordneten Komponenten selbst ideale Funktionalitäten realisieren, können sie als untergeordnete Komponenten in immer komplexeren Protokollen fungieren. Diese Vorgehensweise wird in den Abbildungen 1 bis 4 am Beispiel des SSL/TLS Protokolls veranschaulicht: Zunächst beweist man, das kryptographische Primitive, wie solche zur Verschlüsselung und für digitale Signaturen, geeignete ideale Funktionalitäten realisieren (Abbildung 2). Man kann dann diese idealen Funktionalitäten verwenden, um zu zeigen, dass das Handshake-Protokoll von SSL/TLS ein sicheres Schlüsselaustauschprotokoll ist, d. h. eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{KE}}$ realisiert (siehe Abbildung 3). Schließlich kann man $\mathcal{F}_{\mathrm{KE}}$ verwenden, um zu zeigen, dass SSL/TLS einen sicheren Kommunikationskanal realisiert, also eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{SC}}$ (siehe Abbildung 4). Durch die Kompositionstheoreme können nun alle idealen Funktionalitäten durch ihre Realisierung ersetzt werden, was dann, falls die einzelnen Beweisschritte jeweils erfolgreich waren, insgesamt zeigen würde, dass das SSL/TLS Protokoll (ohne idealisierte Komponenten) ein sicheres Kommunikationsprotokoll ist (Abbildung 1).

Darüber hinaus erlauben es die Kompositionstheoreme, wie z. B. Canettis Kompositionstheorem im UC-Modell [Can01] oder die Kompositionstheoreme im IITM-Modell von Küsters [Küs06], und Kompositionstheoreme mit *gemeinsamem Zustand* (*joint state*), z. B. das von Canetti und Rabin [CR03], eine Protokollsitzung in Isolation zu betrachten und aus deren Sicherheit, die Sicherheit beliebig vieler, paralleler Protokollsitzungen zu folgern.

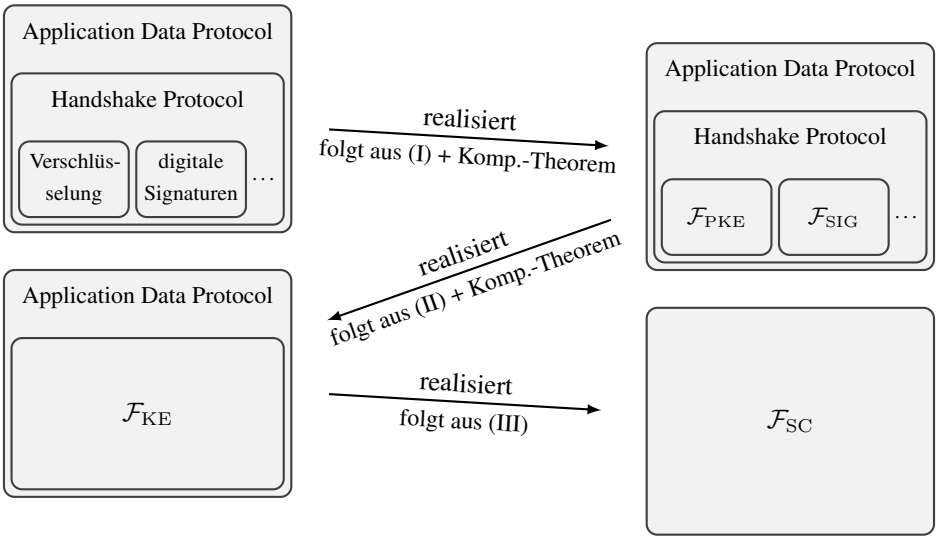

Abbildung 1: Modulare Protokollanalyse in UC-Modellen am Beispiel von SSL/TLS. Die Sicherheit von SSL/TLS folgt aus den Beweisschritten I, II und III (siehe Abbildung 2, 3 bzw. 4) und mit Hilfe von Kompositionstheoremen und der Transitivität der Realisierungs-Relation.

Abbildung 2: Beweisschritt I. Es muss gezeigt werden, dass die im Protokoll verwendeten kryptographischen Primitive geeignete ideale Funktionalitäten realisieren.

Abbildung 3: Beweisschritt II. Es muss gezeigt werden, dass das TLS Handshake Protocol (mit idealisierten kryptographischen Primitiven) eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{KE}}$ realisiert.

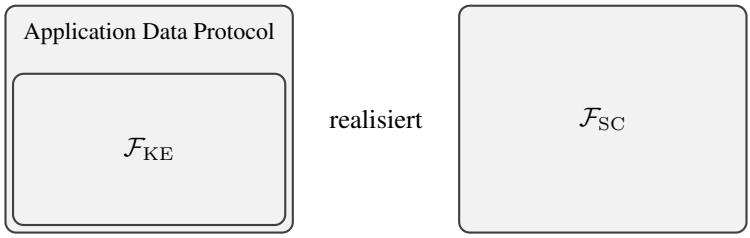

Abbildung 4: Beweisschritt III. Es muss gezeigt werden, dass das TLS Application Data Protocol (mit idealisiertem Schlüsselaustausch) eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{SC}}$ realisiert.

Dieser vielversprechende Ansatz wird in der Kryptographie häufig genutzt, um (neue) Protokolle modular zu entwerfen (siehe [Can06] für einen Überblick). Allerdings wurde er bisher nur selten zur Analyse existierender, praxisrelevanter Sicherheitsprotokollen verwendet. Die nennenswertesten Arbeiten hierzu sind die von Gajek et al. [GMP+08] und Backes et al. [BCJ+06]. Gajek et al. [GMP+08] benutzten Canettis UC-Modell, um den kryptographischen Kern von TLS zu analysieren. Allerdings analysierten sie lediglich eine stark modifizierte Variante von TLS. Backes et al. [BCJ+06] analysierten Kerberos, betrachteten aber auch nur eine idealisierte Variante des Orginalprotokolls und benutzten ein unrealistisches Angreifermodell.

Es gab verschiedene Hindernisse für die Verwendung des ansonsten vielversprechenden UC-Ansatzes für die modulare Analyse praxisrelevanter Sicherheitsprotokolle.

Ein Hindernis war das Fehlen geeigneter idealer Funktionalitäten grundlegender kryptographischer Primitive. Obwohl ideale Funktionalitäten für viele kryptographische Aufgaben, wie asymmetrische Verschlüsselung, digitale Signaturen, Authentifizierung und viele andere Aufgaben in UC-Modellen formuliert wurden, zusammen mit ihren Realisierungen, existierte überraschenderweise keine solche Funktionalität für die symmetrische Verschlüsselung und viele andere grundlegende symmetrische kryptographische Verfahren.

Ein weiteres entscheidendes Hindernis war, dass alle bisher vorgestellten Kompositionstheoreme annehmen, dass Parteien, welche an einer gemeinsamen Protokollsitzung teilnehmen, eine im Vorfeld ausgehandelte Sitzungskennung (SID) verwenden. Zwar ist das Verwenden einer solchen SID ein gutes Konstruktionsprinzip, aber existierende Protokolle, insbesondere in der Praxis verwendete, benutzen solche SIDs typischerweise nicht; zumindest nicht explizit oder in der Art und Weise wie von den Theoremen gefordert. Daher können diese Theoreme nicht für die modulare Analyse solcher Protokolle verwendet werden.

3 Hauptbeiträge

Das Hauptziel der Doktorarbeit war es, Techniken, Hilfsmittel und Kompositionstheoreme zur Verfügung zu stellen, welche eine hochmodulare Protokollanalyse in UC-Modellen erlauben, ohne deren Genauigkeit einzuschränken. Dazu mussten insbesondere die oben beschriebenen Hindernisse genau verstanden und überwunden werden.

3.1 Eine ideale Funktionalität für symmetrische kryptographische Primitive

Wie erwähnt, bilden ideale Funktionalitäten die Grundlage für die modulare Protokollanalyse in UC-Modellen. Ideale Funktionalitäten für kryptographische Primitive stellen übergeordneten Protokollen die Funktionalität der Primitive (z. B. Ver- und Entschlüsselung) bereit und garantieren Sicherheit auf „syntaktische“ Art und Weise, z. B. werden Chiffretexte unabhängig von den zu verschlüsselnden Klartexten generiert. Dies vereinfacht die Sicherheitsanalyse, da man auf die syntaktisch garantierten Eigenschaften der idealen Funktionalitäten zurückgreifen kann.

Es existierten bereits viele Funktionalitäten für asymmetrische Primitive, aber keine geeigneten Funktionalitäten für symmetrische Verfahren, insbesondere symmetrische Verschlüsselung. Im Vergleich zu einer idealen Funktionalität für asymmetrischen Verschlüsselung, gibt es verschiedene Schwierigkeiten beim Entwurf einer Funktionalität für symmetrische Verschlüsselung. Bei asymmetrischer Verschlüsselung ist es vernünftig anzunehmen, dass der private Schlüssel niemals die Funktionalität verlässt. Dadurch ist es relativ einfach, geeignete Sicherheitsgarantien zu formulieren. Symmetrische Schlüssel hingegen müssen typischerweise zwischen Protokollteilnehmern ausgetauscht werden, wie z. B. bei Kerberos.

Ein wesentlicher Beitrag der Doktorarbeit war die Entwicklung einer idealen Funktionalität für symmetrische Verschlüsselung [KT11b]. Diese Funktionalität unterstützt außerdem Verfahren zum Ableiten von Schlüsseln, asymmetrische Verschlüsselung und Message Authentication Codes (MAC). Wir haben bewiesen, dass diese Funktionalität mit gängigen Annahmen an übliche kryptographische Verfahren realisiert werden kann.

Diese Funktionalität, welche mit anderen idealen Funktionalitäten, insbesondere solchen, die für reale Protokolle relevant sind (z. B. für digitale Signaturen), kombiniert werden kann, hat ein weites Anwendungsfeld und bietet eine gute Basis für präzise und modulare kryptographische Analyse von praxisrelevanten Sicherheitsprotokollen. Sie vereinfacht, wie erwähnt, aufgrund ihrer syntaktischen Sicherheitsgarantien derartige Analysen sehr.

3.2 Kriterium für Schlüsselaustauschprotokolle und eine Fallstudie an WPA2

Um den Nutzen unserer neuen idealen Funktionalität für die Analyse von praxisrelevanten Sicherheitsprotokollen zu demonstrieren, entwickelten wir ein Kriterium für Schlüsselaustauschprotokolle [KT11b]. Wir zeigten, dass ein Protokoll, welches unser Kriterium erfüllt, sicher im Sinne von UC-Modellen ist, d. h. eine ideale Funktionalität für Schlüsselaustausch realisiert. Da dieses Kriterium auf der neuen Funktionalität basiert, erfordert es lediglich informationstheoretische oder gar rein syntaktische Argumente, anstatt aufwendiger Reduktionsargumente, um das Kriterium zu zeigen.

Als Fallstudie nutzten wir unsere Methode, um zwei zentrale Protokolle des IEEE Standards 802.11i (WPA2),[1] nämlich das 4-Way Handshake (4WHS) Protokoll und das CCM Protokoll, zu analysieren und deren Sicherheit zu beweisen. Dies stellt die erste fundierte kryptographische Analyse dieser Protokolle dar. Diese Fallstudie ist auch deshalb erwähnenswert, weil wir in der Lage waren, das Protokoll sehr präzise zu modellieren. Zum Beispiel konnten die verwendeten Nachrichtenformate exakt nachempfunden werden.

3.3 Kryptographische Korrektheit von Analysen in symbolischen Modellen

Formale Analysen von Sicherheitsprotokollen basierend auf symbolischen Modellen, auch Dolev-Yao-Modelle [DY83] genannt, werden sehr erfolgreich eingesetzt, um Schwachstel-

[1] Dieses Protokoll wird typischerweise in WLAN-Netzen für die sichere Kommunkation zwischen Computer (Laptop/Smartphone/Tablet) und Access Point verwendet.

len in existierenden Protokollen zu finden und Sicherheit mit automatischen Methoden zu beweisen. Eine dabei wichtige Frage ist, ob diese Art der formalen Analyse Sicherheitsgarantien im Sinne der modernen Kryptographie bietet. Initiiert durch die grundlegende Arbeit von Abadi und Rogaway [AR00] wurde diese Frage untersucht und viele positive Ergebnisse zeigen diese sogenannte kryptographische Korrektheit formaler Analysen. Für den Fall von aktiven Angreifern und Protokollen, welche symmetrische Verschlüsselung verwenden, blieb dies allerdings eine Herausforderung.

In dieser Arbeit stellten wir das erste allgemeine Ergebnis zur kryptographischen Korrektheit formaler Analysen für Schlüsselaustauschprotokolle, die symmetrische Verschlüsselung verwenden, vor [KT09]. Dazu entwickelten wir ein symbolisches, automatisch verifizierbares Kriterium und zeigten, dass Schlüsselaustauschprotokolle, welche dieses Kriterium erfüllen, sicher im Sinne von UC-Modellen sind. Unser Ergebnis gilt unter gängigen kryptographischen Annahmen. Im Beweis dieses Ergebnisses nutzten wir unsere neue ideale Funktionalität zur Abstraktion von symmetrischer Verschlüsselung und wir verwendeten Kompositionstheoreme, um Sicherheit für mehrere Protokollsitzungen aus der Sicherheit einer einzelnen Protokollsitzung zu folgern. Dies demonstriert den Nutzen unserer idealen Funktionalität auf dem Gebiet der kryptographischen Korrektheit formaler Analyse.

3.4 Kompositionstheoreme ohne vorherbestimmte Sitzungskennungen

Um dem zweiten in Abschnitt 2 angesprochenen Hindernis zu begegnen, stellten wir ein universelles Kompositionstheorem und eines mit gemeinsamem Zustand vor, welche keine vorherbestimmten Sitzungskennung (SID) annehmen oder verwenden. Der gemeinsame Zustand im Kompositionstheorem ist unsere neue ideale Funktionalität (siehe Abschnitt 3.1). Diese Kompositionstheoreme erlauben es, die Sicherheit von vielen parallelen Protokollsitzungen aus der Sicherheit einer einzelnen Sitzung zu folgern.

Kompositionstheoreme mit gemeinsamem Zustand sind geeignet, um Protokolle zu analysieren, bei denen verschiedenen Sitzungen einen gemeinsamen Zustand haben. Dies ist meist der Fall für grundlegende Schlüsselaustausch- oder Authentifizierungsprotokolle, da diese meist Langzeit-Schlüssel (asymmetrische oder symmetrische Schlüssel, die über viele Protokollsitzungen hinweg verwendet werden) verwenden. Die bisher existierenden Kompositionstheoreme mit gemeinsamem Zustand basierten auf vorherbestimmten SIDs. Diese SIDs wurden genutzt, um den gemeinsamen Zustand quasi wieder aufzuheben. Zum Beispiel dadurch, dass nicht der eigentliche Klartext, sondern Klartext + SID verschlüsselt wird. Der Chiffretext kann dann nicht mehr in eine andere Sitzung eingeschleust werden, da dort auffallen würde, dass er die falsche SID enthält. Das Problem ist, dass viele Protokolle (z. B. SSL/TLS und das 4WHS von WPA2) dies nicht so explizit machen. Dadurch können diese Kompositionstheoreme meist nicht verwendet werden.

Ein entscheidender Beitrag unserer Arbeit war es, ein geeignetes Kriterium (*implizite Disjunktheit* genannt) zu entwickeln, welches diese Quasi-Trennung des Zustandes charakterisiert [KT11a]. Im Wesentlichen drückt das Kriterium aus, dass eine Nachricht, die in

einer Protokollsitzung verschlüsselt (bzw. signiert) wird, nicht in einer anderen Protokollsitzung erfolgreich entschlüsselt (bzw. verifiziert) werden kann.

Praxisrelevante Protokolle erfüllen typischerweise implizite Disjunktheit und dies ist meist leicht zu zeigen (z. B. ist oft nur ein kleiner Teil der verwendeten kryptographischen Primitive zu betrachten), wie wir in Fallstudien zu den Sicherheitsprotokollen SSL/TLS, IEEE 802.11i (WPA2), SSH, IPsec und EAP-PSK gezeigt haben. All diese Protokolle sind implizit disjunkt und daher ist unser Kompositionstheorem mit gemeinsamem Zustand auf sie anwendbar. Um nun also ihre Sicherheit zu beweisen, genügt es, eine einzelne Protokollsitzung in Isolation zu betrachten. Dies vereinfacht den Sicherheitsbeweis stark und erlaubt es nun leichter, die Protokolle im Detail zu analysieren.

4 Zusammenfassung und Ausblick

In dieser Arbeit wurde erläutert, dass es unverzichtbar ist, komplexe, praxisrelevante Sicherheitsprotokolle im Detail zu analysieren. Die existierenden Methoden und Techniken, insbesondere grundlegende ideale Funktionalitäten und Kompositionstheoreme, reichten allerdings für eine solche Analyse nicht aus.

Mit der Entwicklung vielseitig einsetzbarer, grundlegender idealer Funktionalitäten für kryptographische Primitive und mit der Entwicklung geeigneter Kompositionstheoreme liefert diese Arbeit eine solide Grundlage für eine umfassende modulare Sicherheitsanalyse praxisrelevanter Sicherheitsprotokolle. Die Anwendbarkeit und der Nutzen der in dieser Arbeit entwickelten Methoden und Techniken wurde in mehreren Fallstudien an praxisrelevanten Protokollen (SSL/TLS, IEEE 802.11i (WPA2), SSH, IPsec und EAP-PSK) demonstriert. Eine hochmodulare Analyse, wie z. B. in Abbildung 1 beschrieben, ist nun auch für praxisrelevante Sicherheitsprotokolle möglich.

Obwohl sich unsere Anwendungen auf praxisrelevante Protokolle konzentrieren, sind unsere Theoreme und Techniken über diesen Anwendungsfall hinaus von großem Interesse für die Informationssicherheit und Kryptographie, da sie dazu beitragen, die Grenzen des Möglichen bei modularer Analysen auszuloten und zu erweitern.

Unsere Arbeit bietet eine sehr fundierte Basis für zukünftige Forschung. Im Rahmen der Arbeit wurde das Protokoll IEEE 802.11i (WPA2) einer genaueren Analyse unterzogen. Für andere praxisrelevante Protokolle wurde demonstriert, dass die entwickelten Methoden und Techniken prinzipiell anwendbar sind. Eine eingehende (modulare) Analyse dieser Protokolle ist deshalb eine natürliche Fortsetzung der hier begonnenen Arbeit. Dazu ist es auch sinnvoll, den Satz der hier bereits entwickelten idealen Funktionalitäten zu erweitern und die vorgestellten Kompositionstheoreme noch zu verallgemeinern. Diese Arbeiten sind bereits Gegenstand aktueller Forschungsprojekte.

Die hier erzielten Resultate werden auch bereits außerhalb der klassischen Kryptographie zur Analyse von sicherheitskritischen Softwaresystemen direkt auf der Ebene der Programmiersprachen verwendet. Dazu werden die in diesen Systemen benutzten kryptographischen Primitive durch ideale Funktionalitäten ersetzt. Diese so idealisierten Systeme werden dann mit Werkzeugen zur Programmanalyse verifiziert (siehe, z. B., [FKS11, KTG12]).

Literatur

[ACC+08] A. Armando, R. Carbone, L. Compagna, J. Cuéllar und M. L. Tobarra. Formal Analysis of SAML 2.0 Web Browser Single Sign-on: Breaking the SAML-based Single Sign-on for Google Apps. In *FMSE 2008*, Seiten 1–10. ACM, 2008.

[AP13] N. J. AlFardan und K. G. Paterson. Lucky Thirteen: Breaking the TLS and DTLS Record Protocols. In *S&P 2013*, Seiten 526–540. IEEE Computer Society, 2013.

[APW09] M. R. Albrecht, K. G. Paterson und G. J. Watson. Plaintext Recovery Attacks against SSH. In *S&P 2009*, Seiten 16–26. IEEE Computer Society, 2009.

[AR00] M. Abadi und P. Rogaway. Reconciling Two Views of Cryptography (The Computational Soundness of Formal Encryption). In *IFIPTCS 2000*, Jgg. 1872 *LNCS*, Seiten 3–22. Springer, 2000.

[Bar06] G. V. Bard. A Challenging but Feasible Blockwise-Adaptive Chosen-Plaintext Attack on SSL. Bericht 2006/136, Cryptology ePrint Archive, 2006. Online verfügbar: `http://eprint.iacr.org/2006/136`.

[BCJ+06] M. Backes, I. Cervesato, A. D. Jaggard, A. Scedrov und J.-K. Tsay. Cryptographically Sound Security Proofs for Basic and Public-Key Kerberos. In *ESORICS 2006*, Jgg. 4189 *LNCS*, Seiten 362–383. Springer, 2006.

[BFWW11] C. Brzuska, M. Fischlin, B. Warinschi und S. C. Williams. Composability of Bellare-Rogaway Key Exchange Protocol. In *CCS 2011*, Seiten 51–62. ACM, 2011.

[Ble98] D. Bleichenbacher. Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1. In *CRYPTO 1998*, Jgg. 1462 *LNCS*, Seiten 1–12. Springer, 1998.

[BR93] M. Bellare und P. Rogaway. Entity Authentication and Key Distribution. In *CRYPTO 1993*, Jgg. 773 *LNCS*, Seiten 232–249. Springer, 1993.

[Can01] R. Canetti. Universally Composable Security: A New Paradigm for Cryptographic Protocols. In *FOCS 2001*, Seiten 136–145. IEEE Computer Society, 2001.

[Can06] R. Canetti. Security and Composition of Cryptographic Protocols: A Tutorial. Bericht 2006/465, Cryptology ePrint Archive, 2006. `http://eprint.iacr.org/`.

[CJS+08] I. Cervesato, A. D. Jaggard, A. Scedrov, J.-K. Tsay und C. Walstad. Breaking and Fixing Public-key Kerberos. *Inf. Comput.*, 206(2-4):402–424, 2008.

[CR03] R. Canetti und T. Rabin. Universal Composition with Joint State. In *CRYPTO 2003*, Jgg. 2729 *LNCS*, Seiten 265–281. Springer, 2003.

[DP10] J. P. Degabriele und K. G. Paterson. On the (In)Security of IPsec in MAC-then-Encrypt Configurations. In *CCS 2010*. ACM, 2010.

[DY83] D. Dolev und A. C. Yao. On the Security of Public-Key Protocols. *IEEE Transactions on Information Theory*, 29(2):198–208, 1983.

[FKS11] C. Fournet, M. Kohlweiss und P.-Y. Strub. Modular code-based cryptographic verification. In *CCS 2011*, Seiten 341–350. ACM, 2011.

[FKS14] D. Fett, R. Küsters und G. Schmitz. An Expressive Model for the Web Infrastructure: Definition and Application to the BrowserID SSO System. In *S&P 2014*. IEEE Computer Society, 2014.

[GMP+08] S. Gajek, M. Manulis, O. Pereira, A. Sadeghi und J. Schwenk. Universally Composable Security Analysis of TLS. In *ProvSec 2008*, Jgg. 5324 *LNCS*, Seiten 313–327. Springer, 2008.

[HY08] A. Herzberg und I. Yoffe. The Layered Games Framework for Specifications and Analysis of Security Protocols. In *TCC 2008*, Jgg. 4948 *LNCS*, Seiten 125–141. Springer, 2008.

[KT09] R. Küsters und M. Tuengerthal. Computational Soundness for Key Exchange Protocols with Symmetric Encryption. In *CCS 2009*, Seiten 91–100. ACM, 2009.

[KT11a] R. Küsters und M. Tuengerthal. Composition Theorems Without Pre-Established Session Identifiers. In *CCS 2011*, Seiten 41–50. ACM, 2011.

[KT11b] R. Küsters und M. Tuengerthal. Ideal Key Derivation and Encryption in Simulation-based Security. In *CT-RSA 2011*, Jgg. 6558 *LNCS*, Seiten 161–179. Springer, 2011.

[KTG12] R. Küsters, T. Truderung und J. Graf. A Framework for the Cryptographic Verification of Java-like Programs. In *CSF 2012*, Seiten 198–212. IEEE Computer Society, 2012.

[Küs06] R. Küsters. Simulation-Based Security with Inexhaustible Interactive Turing Machines. In *CSFW-19 2006*, Seiten 309–320. IEEE Computer Society, 2006. Vollständige und überarbeitete Version: `http://eprint.iacr.org/2013/025`.

[MSW10] P. Morrissey, N. P. Smart und B. Warinschi. The TLS Handshake Protocol: A Modular Analysis. *Journal of Cryptology*, 23(2):187–223, 2010.

[PW01] B. Pfitzmann und M. Waidner. A Model for Asynchronous Reactive Systems and its Application to Secure Message Transmission. In *S&P 2001*, Seiten 184–201. IEEE Computer Society, 2001.

[PY06] K. G. Paterson und A. K. L. Yau. Cryptography in Theory and Practice: The Case of Encryption in IPsec. In *EUROCRYPT 2006*, Jgg. 4004 *LNCS*, Seiten 12–29. Springer, 2006.

[RD09] M. Ray und S. Dispensa. Renegotiating TLS. November 2009. Online verfügbar: `http://kryptera.se/Renegotiating%20TLS.pdf`.

[Tue13] M. Tuengerthal. *Analysis of Real-World Security Protocols in a Universal Composability Framework*. Logos Verlag Berlin, 2013. Doktorarbeit.

[TWP07] E. Tews, R.-P. Weinmann und A. Pyshkin. Breaking 104 Bit WEP in Less Than 60 Seconds. In *WISA 2007*, Seiten 188–202, 2007.

Dr. Max Tuengerthal studierte Informatik an der Christian-Albrechts-Universität zu Kiel und machte 2007 seinen Abschluss zum Diplom-Informatiker. Danach arbeitete er als wissenschaftlicher Mitarbeiter in der Foundations of Computer and Network Security Group an der ETH Zürich und am Lehrstuhl für Informationssicherheit und Kryptographie an der Universität Trier und promovierte 2013 an der Universität Trier. Sein Forschungsschwerpunkt war das Design und die Analyse von Sicherheitsprotokollen, insbesondere die Entwicklung von Methoden und Techniken zur modularen Protokollanalyse. Seit 2014 ist Max Tuengerthal Berater und Softwareentwickler bei der ecsec GmbH, Michelau.