

# Analyse, Design und Einsatz kryptographischer Primitive<sup>1</sup>

Christian Forler<sup>2</sup>

**Zusammenfassung.** Diese Arbeit gibt eine Übersicht über die Dissertation von Herrn Christian Forler, welche sich im Themenkomplex der symmetrischen Kryptographie bewegt. Es wird dabei auf die folgenden drei Resultate eingegangen, sowie deren Einfluss auf die Forschung in diesem Gebiet vorgestellt: (i) eine Robustheitsanalyse von bestehenden Verfahren zur authentisierten Verschlüsselung, (ii) MCOE – das erste robuste Verfahren zur authentisierten Verschlüsselung und (iii) CATENA – ein modernes Verfahren zur Generierung von Passworthashes.

## 1 Einführung

Im Rahmen von Herrn Forlers Dissertation [Fo15] wurden vier Verfahren entworfen, um praktische Probleme in der Anwendung kryptographischer Systeme zu lösen. Aus Platzgründen fokussiert die vorliegende Zusammenfassung lediglich auf zwei dieser Verfahren: MCOE – ein robustes Verfahren zur authentisierten Verschlüsselung; und CATENA – eine Passwort-Hashfunktion mit dem Ziel einer signifikanten Steigerung der Sicherheit von Passwörtern. Die anderen beiden Verfahren (COFFE [Fo14] – ein weiteres Verfahren zur authentisierten Verschlüsselung und Twister <sub>$\pi$</sub>  [F110] – eine dezidierte kryptographische Hashfunktion) werden nicht betrachtet.

## 2 MCOE: Ein Verfahren zur authentisierten Verschlüsselung

Herr Forler setzt sich in seiner Dissertation mit der Robustheit von beweisbar sicheren Verfahren zur authentisierten Verschlüsselung (AE-Verfahren) auseinander. Insbesondere liegt der Fokus auf sogenannten On-line-AE-Verfahren (OAE-Verfahren), welche es ermöglichen, Datenströme unbekannter Länge ohne spürbare Latenz zu verschlüsseln, d.h. AE-Verfahren bei denen eine Nachricht in Blöcke aufgeteilt wird, und zur Berechnung des  $i$ -ten Chiffretextblocks nur die ersten  $i$  Klartextblöcke gelesen werden müssen.

OAE-Verfahren erlauben den Aufbau eines *sicheren Kanals* zwischen Sender und Empfänger, welcher nicht nur die Vertraulichkeit der übertragenen Daten, sondern auch deren Integrität schützt. Dies verhindert zum einen, dass ein Angreifer etwas über den Inhalt der übertragenen Daten erfährt und zum anderen, dass es für einen Angreifer nicht möglich ist, diese unentdeckt zu manipulieren.

Herr Forler hat im Rahmen seiner Promotion die Robustheit von existierenden OAE-Verfahren analysiert. Dabei hat er sich intensiv mit zwei sogenannten Missbrauchsfällen (*Misuse*) auseinandergesetzt: (i) die Wiederverwendung einer Nonce (*Nonce Misuse*), und (ii) die Freigabe unverifizierter Klartexte (*Decryption Misuse*).

<sup>1</sup> Englischer Titel der Dissertation: “Analysis, Design & Applications of Cryptographic Building Blocks”

<sup>2</sup> Hochschule Schmalkalden, c.forler@hs-sm.de

## 2.1 Der Missbrauch von Verfahren zur authentisierten Verschlüsselung

**Wiederverwendung einer Nonce (*Nonce Misuse*).** Ein Verfahren zur Verschlüsselung von Daten kann im kryptographischen Sinne nur sicher sein, falls es entweder probabilistisch ist, oder einen Zustand hat. Goldwasser und Micali haben bereits 1984 in einer ihrer Veröffentlichungen diese Aussage formal bewiesen [GM84]. Der Einsatz von probabilistischen Verfahren gestaltet sich jedoch in der Praxis als äußerst schwierig, da hierfür kryptographisch sichere Zufallszahlen generiert werden müssen. Aus diesem Grund ist die kryptographische Community dem Vorschlag von Rogaway gefolgt, sich auf zustandsbehaftete AE-Verfahren zu konzentrieren [Ro02, Ro04b]. Bei dem Zustand eines AE-Verfahrens handelt es sich um einen weiteren Eingabevektor, der neben dem geheimen Schlüssel und dem Klartext vom Benutzer übergeben werden muss. Damit liegt das Unterbinden von der Mehrfachverwendung eines Zustands – unter dem gleichen Schlüssel – in dem Verantwortungsbereich des Nutzers. Eine Bürde, dieser sich viele Nutzer nicht einmal bewusst sind.

In der Kryptographie wird ein Zustand, der sich unter dem gleichen Schlüssel nicht wiederholen darf, als Nonce (**N**umber **u**sed **o**nce) bezeichnet. Der Sicherheitsbeweis eines zustandsbehafteten AE-Verfahrens geht immer davon aus, dass sich ein Zustand (Nonce) niemals wiederholt. Eine Wiederverwendung einer Nonce führt daher unweigerlich zur Invalidierung des Sicherheitsbeweises. Dies hat zur Folge, dass in diesem Fall keine Sicherheit mehr garantiert wird, da das Verfahren außerhalb seiner Spezifikation betrieben wird.

In der Praxis kommt es durch den Mangel an Sicherheitsbewusstsein immer wieder zu einem fehlerhaften Umgang mit Nonces, einem sogenannten *Nonce Misuse*. Selbst etablierte Softwareprodukte wie Microsoft Word oder Excel sind nicht gegen diese Art von missbräuchlicher Nutzung gefeit [Wu05].

**Freigabe unverifizierter Klartext(teile) (*Decryption Misuse*).** Kryptographen gehen bei der Entschlüsselung eines Chiffretextes implizit davon aus, dass diese korrekt und ohne die Freigabe unverifizierter Klartexte abläuft. Das heißt, ein Klartext wird erst nach erfolgreicher Integritätsprüfung an den Benutzer übergeben. Aus Gründen der Performanz sind die Integritätsprüfung und die Entschlüsselung miteinander verbunden, d.h. erst bei der Entschlüsselung des finalen Chiffretextblockes wird die Integrität des gesamten Chiffretextes und somit auch die Integrität des Klartextes überprüft. Bei einer ordnungsgemäßen Entschlüsselung wird der unverifizierte Klartext daher bis zur Freigabe im Adressraum des Entschlüsselungsprozesses gepuffert.

Oftmals haben kryptographische Bibliotheken, wie beispielsweise OpenSSL, Programmierschnittstellen (APIs), die Anwendungsprogrammierer zur Freigabe unverifizierter Teile eines Klartextes *ermuntern*. Es werden APIs bereitgestellt, bei denen Chiffretexte peu à peu entschlüsselt werden können. Im Anschluss an die Entschlüsselung ist es dann noch möglich, die Integrität des Chiffretextes zu überprüfen. Von solchen APIs ist abzuraten. Weiterhin leitet die Anleitung zur vermeintlich korrekten Entschlüsselung von

OAE-Verfahren auf dem offiziellen OpenSSL-Wiki den Leser zur Freigabe unverifizierter Klartexte an<sup>3</sup>. Das Beispiel liefert selbst bei einer gescheiterten Integritätsprüfung einen manipulierten Klartext zurück. Der Anwender bekommt lediglich durch den Rückgabewert der Entschlüsselungsfunktion mit, ob die Validierung des Chiffretextes erfolgreich war oder nicht. Ein solches vorgehen ist fahrlässig und führt früher oder später unvermeidlich dazu, dass ein naiver Benutzer manipulierte Klartexte für authentisch erachtet, weshalb nicht-authentische Klartexte unverzüglich gelöscht werden sollten. Nur in begründeten Ausnahmen ist von einem solchen Vorgehen abzuweichen.

Philipp Heckel hat 2014 in seinem Blog einen Fehler in der `CipherOutputStream` im Java JDK 1.7 aufgedeckt<sup>4</sup>. Bei der Verwendung dieser Klasse mit einem OAE-Verfahren wurde eine gescheiterte Integritätsprüfung ignoriert. Somit war es nicht möglich, die Manipulation des Chiffretextes zu erkennen und entsprechend zu reagieren.

Mit einer Einschränkung oder Minimierung des Missbrauchspotentials ist bei den üblichen kryptographischen Bibliotheken in absehbarer Zukunft nicht zu rechnen.

Bei der vorzeitigen Freigabe von Klartexten (*Decryption Misuse*) handelt es sich nicht immer um einen Softwarefehler. Es gibt Umgebungen, die sich durch einen hohen Durchsatz, niedrige Latenz und lange Nachrichten auszeichnen. In solch einem Umfeld ist das Zurückhalten des Klartextes bis zur erfolgreichen Integritätsprüfung nicht praktikabel bzw. gar nicht möglich. Ein Beispiel hierfür sind OTNs (Optical Transport Networks) [IT09], bei denen mit einem Durchsatz von bis zu 100 Gbps und einer Latenz von nur wenigen Taktzyklen Netzwerkpakete bis 64 kB verarbeitet werden. Bei einer ordnungsgemäßen Entschlüsselung würde die vorgeschriebene Latenzzeit überschritten werden. Für solche Einsatzgebiete werden robuste OAE-Verfahren benötigt, die selbst unter solch widrigen Umständen noch ein akzeptables Maß an Sicherheit garantieren.

## 2.2 Analyse und Ergebnis der Robustheitsuntersuchung

In seiner Dissertation hat Herr Forler bis dato alle gängigen OAE-Verfahren auf Robustheit gegenüber *Nonce Misuse* und *Decryption Misuse* untersucht. Die Ergebnisse der Untersuchung sind in Tabelle 1 zu finden. Es ist darauf hinzuweisen, dass diese Ergebnisse keineswegs die existierenden Sicherheitsbeweise der untersuchten Verfahren invalidieren. Die Betrachtungen fanden in einem *Misuse*-Szenario statt, in welchem über die Sicherheit dieser Verfahren zuvor keine Aussage getätigt wurde. Bei sachgerechtem Einsatz bieten also alle untersuchten Verfahren ein hohes Maß an Sicherheit.

<sup>3</sup> [https://wiki.openssl.org/index.php/EVP\\_Authenticated\\_Encryption\\_and\\_Decryption](https://wiki.openssl.org/index.php/EVP_Authenticated_Encryption_and_Decryption), Januar 2016

<sup>4</sup> <https://blog.heckel.xyz/2014/03/01/cipherinputstream-for-aead-modes-is-broken-in-jdk7-gcm/>, Januar 2016

Verfahren		Nonce Misuse		Decryption Misuse
		Vertraulichkeit	Integrität	Vertraulichkeit
CCFB	[Lu05]	$O(1)$	$O(1)$	$O(1)$
CHM	[Iw06]	$O(1)$	$O(1)$	$O(1)$
COPA	[An13]	N/A	N/A	$O(1)$
CWC	[KVV04]	$O(1)$	$O(1)$	$O(1)$
EAX	[BRW04]	$O(1)$	$O(1)$	$O(1)$
GCM	[MV04]	$O(1)$	$O(1)$	$O(1)$
IACBC	[Ju08]	$O(1)$	$O(1)$	$O(1)$
IAPM	[Ju08]	$O(1)$	$O(1)$	$O(1)$
OCB1	[Ro01]	$O(1)$	$O(1)$	$O(1)$
OCB2	[Ro04a]	$O(1)$	$O(1)$	$O(1)$
OCB3	[KR11]	$O(1)$	$O(1)$	$O(1)$
RPC	[BKY01]	$O(1)$	$O(1)$	$O(1)$
TAE	[LRW11]	$O(1)$	$O(1)$	$O(1)$
XCBC-XOR	[GD01]	$O(2^{n/4})$	$O(1)$	$O(1)$

Tab. 1: Maximum aus Speicher und Zeitkomplexität von Angriffen, welche die Vertraulichkeit und Integrität gebräuchlicher OAE-Verfahren massiv verletzen. Alle Angriffe haben eine Erfolgswahrscheinlichkeit von 1.

## 2.3 MCOE

Im Rahmen seiner Tätigkeit als wissenschaftlicher Mitarbeiter an der Bauhaus-Universität Weimar, war Herr Forler maßgeblich an der Entwicklung von MCOE beteiligt, welches auf der FSE 2012 in Washington, DC präsentiert wurde [FFL12]. Hierbei handelt es sich um das erste robuste OAE-Verfahren, welches bei fehlerhaftem Umgang mit einer Nonce noch ein moderates Maß an Sicherheit garantierte, d.h. der Schutz der Integrität bleibt vollständig erhalten, während die Vertraulichkeit nicht mehr vollständig gewährleistet werden kann. Bei der Wiederverwendung einer Nonce kann ein Angreifer feststellen, ob zwei Klartexte einen gemeinsamen Präfix haben und wie lang dieser ist. Auf der CRYPTO 2015 haben Hoang et. al darauf hingewiesen, dass diese Eigenschaft zur vollkommenen Verlust der Vertraulichkeit führen kann, falls einem Angreifer (i) bereits die Entschlüsselung des Präfixes vorliegt und (ii) dieser in der Lage ist, beliebige Klartexte mit Hilfe eines Orakels zu verschlüsseln [Ho15].

Die ursprünglich in [FFL12] präsentierte Variante von MCOE ist nicht sicher gegen *Decryption Misuse*. In seiner Dissertation präsentiert Herr Forler eine minimale Modifikation von MCOE, die ohne nennenswerte Auswirkungen auf die Performanz Sicherheit gegen *Nonce Misuse* ermöglicht.

Die Ideen aus [FFL12] wurden von zahlreichen internationalen Forschergruppen aufgegriffen, um ihrerseits eigene robuste OAE Verfahren zu entwickeln [An13, Ab14, An14a, An14b, DN14]. Inzwischen hat sich das Konzept der Robustheit etabliert; beispielswei-

se trägt der laufende Wettbewerb CAESAR<sup>5</sup> (Competition for Authenticated Encryption: Security, Applicability, and Robustness), bei dem ein neuer Standard zur authentisierten Verschlüsselung gesucht wird, bereits das Wort *Robustheit* im Namen.

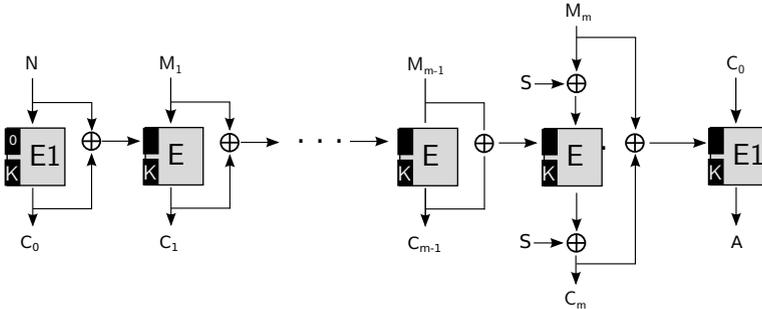


Abb. 1: Illustration der Verschlüsselungsoperation von MCOE.

Die grundlegende generische Struktur von MCOE (siehe Abbildung 1) ist gegeben durch TC3 [RZ11] – ein Verschlüsselungsverfahren basierend auf einer *tweakable Blockchiffre*. Bei einer *tweakable Blockchiffre*  $\tilde{E} : \{0, 1\}^k \times \{0, 1\}^\tau \times \{0, 1\}^n \rightarrow \{0, 1\}^n$  handelt es sich um eine Blockchiffre mit einem zusätzlichen öffentlichen  $\tau$ -Bit-Parameter  $T$ , dem Tweak. Für jedes Tupel  $(K, T) \in \{0, 1\}^k \times \{0, 1\}^\tau$ , welches aus einem Schlüssel  $K$  und einem Tweak  $T$  besteht, gilt:  $\tilde{E}_K(T, \cdot)$  ist eine schlüsselabhängige  $n$ -Bit-Permutation, welche sich effizient aus einer regulären Blockchiffre, wie beispielsweise dem AES [DR00], konstruieren lässt [LRW11]. Der Ver- und Entschlüsselungsprozess von MCOE wird im Folgenden näher erläutert.

**Verschlüsselung.** Die Verarbeitung eines Tupels  $(N, M)$ , bestehend aus einer  $n$ -Bit-Nonce  $N$  und einer Nachricht  $M = M_1, \dots, M_m$  funktioniert wie folgt. Als erstes wird die Nonce  $M_0 = N$  mittels  $\tilde{E}_K$  unter dem Tweak  $T = 0^n$  (Folge von  $n$  Nullen) zu dem Chiffretextblock  $C_0$  verschlüsselt. Anschließend werden die einzelnen Nachrichtenblöcke, mit Ausnahme des letzten Blockes verschlüsselt, wobei gilt:  $C_i = \tilde{E}_K(C_{i-1} \oplus M_{i-1}, M_i)$ . Der letzte Nachrichtenblock  $M_m$  wird dann wie folgt verarbeitet:  $C_m = \tilde{E}_K(C_{m-1} \oplus M_{m-1}, M_m \oplus S) \oplus S$  (siehe Abbildung 1). Bei  $S = \tilde{E}_K(1^n, |M_m|)$  handelt es sich um einen Schlüsselstrom, der aus der Bitlänge des letzten Nachrichtenblockes ( $|M_m|$ ) zusammen mit dem Tweak  $1^n$  (Folge von  $n$  Einsen) berechnet wird. Nach der Verarbeitung des finalen Nachrichtenblockes erfolgt die Generierung des Authentisierungstags  $A$ , wobei gilt:  $A = \tilde{E}_K(C_m \oplus M_m, C_0)$ . Der Chiffretext  $C$  ergibt sich aus der sukzessiven Konkatenation der Werte  $C_1$  bis  $C_m$ , d.h.  $C = C_1, \dots, C_m$ . Um die Entschlüsselung und Verifizierung des Chiffrextes zu gewährleisten, wird  $C$  zusammen mit  $N$  und  $A$  übertragen.

**Entschlüsselung.** Die Verarbeitung eines Nonce-Chiffretext-Authentisierungstag-Tupels  $(N, C, T)$  beginnt mit der erneuten Berechnung von  $C_0$ . Anschließend werden alle Chiffretextblöcke, bis auf den letzten ( $C_m$ ), entschlüsselt; es gilt:  $M_i = \tilde{E}_K^{-1}(C_{i-1} \oplus M_{i-1}, C_i)$ ,

<sup>5</sup> <http://competitions.cr.yp.to/caesar.html>, Januar 2016

wobei es sich bei  $\tilde{E}_K^{-1}$  um das Inverse (Entschlüsselungsoperation) von  $\tilde{E}_K$  handelt. Anschließend wird der letzte Nachrichtenblock  $M_m$  rekonstruiert; es gilt:  $M_m = \tilde{E}_K^{-1}(C_{m-1} \oplus M_{m-1}, C_m \oplus S) \oplus S$ . Zum Schluss wird getestet, ob die Gleichung  $\tilde{E}_K(C_m \oplus M_m, C_0) = A$  erfüllt ist. Falls ja, ist der Chiffretext authentisch, ansonsten fand bei der Übertragung ein Fehler oder eine Manipulation statt.

**Anmerkungen.** Bei dem vorgestellten Verfahren handelt es sich um eine vereinfachte Variante von MCOE für Nachrichten, deren Länge ein Vielfaches der Blockgröße  $n$  ist und bei der keine Zusatzinformationen verarbeitet werden. Das Original erlaubt die Verarbeitung von Nonces und Nachrichten beliebiger Länge [FFL12]. Dies ermöglicht die Authentisierung von Metadaten, wie beispielsweise den Nachrichtenkopf (Header) eines Netzwerkpaketes, indem diese als (Teil einer) Nonce deklariert werden. Bei der Verarbeitung einer beliebig langen Nachricht wird eine Technik namens *Tagsplitting* angewandt. Dabei wird der Authentisierungstag  $A$  auf zwei Blöcke aufgespalten. Der letzte Chiffretextblock wird mit dem ersten Teil von  $A$  *aufgefüllt* und der letzte Teil von  $A$  wird wie gehabt generiert und anschließend auf die entsprechende Bitlänge gekürzt. Dieses Verfahren wurde von Herrn Forler im Rahmen von MCOE mitentwickelt. Weiterhin war er auch maßgeblich an den Sicherheitsbeweisen von MCOE beteiligt. Die vollständige Spezifikation von MCOE und alle dazugehörigen Sicherheitsbeweise finden sich in seiner Dissertation wieder [Fo15].

### 3 CATENA: Ein Passworthashverfahren

Ein weiterer Aspekt von Herr Forlers Dissertation ist die Konstruktion und der Einsatz kryptographischer Hashfunktionen. Im Fokus liegen hier Verfahren zur Generierung von Passworthashes<sup>6</sup>, welche auf einem Rechnersystem zum Zweck der Benutzerauthentisierung verwendet werden.

Seit den 60er Jahren ist es üblich, dass sich ein Benutzer gegenüber einem Rechner durch die Kenntniss eines Geheimnisses in Form eines Passwortes authentisiert. Wilkes hat bereits 1968 darauf hingewiesen, dass die Speicherung von Passwörtern im Klartext unsicher ist [Wi68], da jeder Benutzer potentiell lesenden Zugriff auf den Passwortspeicher hat. Aus diesem Grund wird, seit der Entwicklung von UNIX, auf einem Rechner nur noch der Hash eines Passwortes gespeichert, welcher mit einer kryptographisch sicheren Passworthashfunktion (PHF) generiert wurde [MT79]. In der Regel wird ein Passworthash nicht nur aus dem vertraulichen Passwort, sondern zusätzlich auch aus einer (*öffentlich einsehbaren*) Zufallszahl (Salt) abgeleitet. Die Generierung eines Codebuches – bei der zu jedem Passwort der zugehörige Hash gespeichert wird – wird damit unpraktikabel. Ein Angreifer müsste für jeden Wert eines  $s$ -bit-Salts ein eigenes Codebuch generieren. Beispielsweise werden bereits für die Speicherung der Codebücher eines 80-Bit-Salts mehrere Yottabytes (eine Billion Terabytes) benötigt.

---

<sup>6</sup> Den *Hash* eines Passwortes kann man als dessen Fingerabdruck interpretieren.

Bei einer klassischen PHF  $F$  handelt es sich um die iterative Ausführung einer kryptographischen Hashfunktion  $H$ , d.h.  $F(\text{pwd}, \text{salt}) = H(H(\dots(H(\text{pwd}, \text{salt})\dots)))$ , wobei  $\text{pwd}$  das geheime Passwort darstellt. Die üblichen Anforderungen an eine kryptographische Hashfunktion haben zur Folge, dass sie im Regelfall äußerst effizient und mit wenig Speicher ( $< 4$  KB) berechnet werden kann. Diese Eigenschaft macht Hashfunktionen extrem anfällig gegenüber hoch-parallelisierbaren Angriffen. Moderne Graphical Processing Units (GPUs) mit hunderten von Kernen [Co12], wie sie in handelsüblichen Grafikkarten verbaut werden, ist es daher möglich, aus einem Passworthash, ein (schwaches oder moderates) Passwort in absehbarer Zeit wieder zu *rekonstruieren*. Beispielsweise kann das *Password Recovery Tool hashcat*<sup>7</sup>, mit Hilfe einer AMD Radeon™ HD 7970 Grafikkarte, pro Sekunde ungefähr 17 Milliarden NTLM<sup>8</sup>-Passwortkandidaten durchprobieren. Von 2013 bis 2015 fand die *Password Hashing Competition* (PHC) statt, um die Forschung in diesem Teilbereich der Kryptographie zu intensivieren. Ziel für die folgenden Jahre ist die internationale Etablierung des Wettbewerbsesiegers Argon2 [BDK15] als Standard für das zum Hashen von Passwörtern.

### 3.1 CATENA

Bei dem PHC-Finalisten CATENA handelt es sich um eine moderne und speicherintensive PHF, welche auch neuen Anforderungen für Sicherheit und Funktionalität, wie intensiven Speicherverbrauch und nutzerunabhängige Aktualisierungen des Passworthashes, gerecht wird. CATENA wurde von Herrn Forler auf der ASIACRYPT 2014 in Taiwan vorgestellt [FLW14].

CATENA basiert grundsätzlich auf zwei Primitiven: (i) einer kryptographischen Hashfunktion  $H : \{0, 1\}^* \rightarrow \{0, 1\}^n$  und (ii) einer speicherintensiven Hashfunktion  $F_\lambda : \{0, 1\}^\alpha \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ , welche effizient mit  $G = 2^s$  Speicherzellen berechnet werden kann. Die wichtigste Anforderung an  $F_\lambda$  ist die sogenannte  $\lambda$ -Memory-Hardness, d.h. der Time-Memory Tradeoff für diese Funktion ist  $O(S^\lambda)$  für  $S \ll G$ . Daher lässt sich  $F_\lambda$  schon für kleine Werte  $\lambda$  mit wenigen Speicherzellen nicht mehr effizient berechnen. Dies hat zur Folge, dass auf einer GPU nur noch wenige Passwortkandidaten pro Sekunde getestet werden können. Eine mögliche Instantiierung für  $F_\lambda$  ist ein Stapel aus  $\lambda$  leicht modifizierten G-Superkonzentratoren. In seiner Dissertation hat Herr Forler eine Instantiierung mit einem modifizierten *Doublebutterfly*-Graph, welcher aus zwei zusammengesetzten FFT-Graphen besteht [Br14], vorgestellt. Dabei beschreibt jeder Knoten des Graphen den unter anderem aus seinem direkten Vorgänger berechneten Hashwert dar.

Für zwei Instanzen von  $H$  und  $F_\lambda$  funktioniert CATENA wie folgt. Zuerst wird aus einem Tweak  $T$ , dem Passwort  $P$ , und einem Salt  $S$  der initiale Verkettungswert  $X$  berechnet. Der Tweak aus den folgenden fünf Komponenten zusammen: (i) einem Anwendungszweck (Byte-kodiert), (ii) den Wert  $\lambda$  (Byte-kodiert), (iii) die Blockgröße  $n$  (16-Bit-kodiert), (iv) die Länge des Salts (32-Bit-kodiert), (v) der Hash von Metadaten, wie beispielsweise der Hostname oder die Benutzer-ID (n-bit-kodiert). Im Anschluss wird der Verkettungswert

<sup>7</sup> <https://hashcat.net/oclhashcat/>, Januar 2016

<sup>8</sup> Eine PHF-basiertes Verfahren von Microsoft, welches in Windows zum Einsatz kommt.

---

**Algorithm 1** CATENA

---

```
1:  $X \leftarrow H(T \parallel P \parallel S)$ 
2: for  $c = 1, \dots, g$  do
3:    $X \leftarrow F_\lambda(c, X)$ 
4:    $X \leftarrow H(c \parallel X)$ 
5: end for
6: return  $x$ 
```

---

für jede Ganzzahl  $c$  in dem Intervall  $[1, g]$  durch die Funktion  $F_\lambda$  und  $H$  aktualisiert (siehe Algorithmus 1). Zum Schluss wird der Passworthash  $X$  zurückgegeben.

Die Struktur von CATENA erlaubt es, bei Erhöhung des Sicherheitsparamters  $g$ , die Passworthashes ohne Nutzerinteraktion zu erneuern. Weiterhin erlaubt es CATENA einen Server zu entlasten, indem der Client alle Operationen bis auf den finalen Aufruf von  $H$  berechnen (Algorithmus 1, Zeile 4 für  $c = g$ ). Der Server muss daher nur noch einmal die effiziente Hashfunktion  $H$  aufrufen, um den Passworthash zu generieren.

In seiner Dissertation hat Herr Forler formal die Sicherheit des Verfahrens, einen Betriebsmodus für die Generierung kryptographischer Schlüssel und konkrete Instanziierungen vorgestellt.

## Literaturverzeichnis

- [Ab14] Abed, Farzaneh; Fluhrer, Scott R.; Forler, Christian; List, Eik; Lucks, Stefan; McGrew, David A.; Wenzel, Jakob: Pipelineable On-line Encryption. In: Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. S. 205–223, 2014.
- [An13] Andreeva, Elena; Bogdanov, Andrey; Luykx, Atul; Mennink, Bart; Tischhauser, Elmar; Yasuda, Kan: Parallelizable and Authenticated Online Ciphers. In: Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I. S. 424–443, 2013.
- [An14a] Andreeva, Elena; Bilgin, Begül; Bogdanov, Andrey; Luykx, Atul; Mennink, Bart; Mousha, Nicky; Yasuda, Kan: APE: Authenticated Permutation-Based Encryption for Lightweight Cryptography. In: Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. S. 168–186, 2014.
- [An14b] Andreeva, Elena; Luykx, Atul; Mennink, Bart; Yasuda, Kan: COBRA: A Parallelizable Authenticated Online Cipher Without Block Cipher Inverse. In: Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. S. 187–204, 2014.
- [BDK15] Biryukov, Alex; Dinu, Daniel; Khovratovich, Dmitry: , Fast and Tradeoff-Resilient Memory-Hard Functions for Cryptocurrencies and Password Hashing. Cryptology ePrint Archive, Report 2015/430, 2015. <http://eprint.iacr.org/>.
- [BKY01] Buonanno, Enrico; Katz, Jonathan; Yung, Moti: Incremental Unforgeable Encryption. In: Fast Software Encryption, 8th International Workshop, FSE 2001 Yokohama, Japan, April 2-4, 2001, Revised Papers. S. 109–124, 2001.
- [Br14] Bradley, William F.: Superconcentration on a Pair of Butterflies. CoRR, abs/1401.7263, 2014.

- [BRW04] Bellare, Mihir; Rogaway, Phillip; Wagner, David: The EAX Mode of Operation. In: Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers. S. 389–407, 2004.
- [Co12] Corporation, Nvidia: , Nvidia GeForce GTX 680 - Technology Overview, 2012.
- [DN14] Datta, Nilanjan; Nandi, Mridul: ELmE: A Misuse Resistant Parallel Authenticated Encryption. In: Information Security and Privacy - 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings. S. 306–321, 2014.
- [DR00] Daemen, Joan; Rijmen, Vincent: Rijndael for AES. In: AES Candidate Conference. S. 343–348, 2000.
- [FFL12] Fleischmann, Ewan; Forler, Christian; Lucks, Stefan: MCOE: A Family of Almost Fool-proof On-Line Authenticated Encryption Schemes. In: Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. S. 196–215, 2012.
- [Fl10] Fleischmann, Ewan; Forler, Christian; Gorski, Michael; Lucks, Stefan: TWISTER<sub>pi</sub> - a framework for secure and fast hash functions. IJACT, 2(1):68–81, 2010.
- [FLW14] Forler, Christian; Lucks, Stefan; Wenzel, Jakob: Memory-Demanding Password Scrambling. In: Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014, Proceedings, Part II. S. 289–305, 2014.
- [Fo14] Forler, Christian; McGrew, David; Lucks, Stefan; Wenzel, Jakob: , COFFE: Ciphertext Output Feedback Faithful Encryption. Cryptology ePrint Archive, Report 2014/1003, 2014. <http://eprint.iacr.org/>.
- [Fo15] Forler, Christian: Analysis, Design & Applications of Cryptographic Building Blocks. Dissertation, Bauhaus-Universität Weimar, 2015.
- [GD01] Gligor, Virgil D.; Donescu, Pompiliu: Fast Encryption and Authentication: XCBC Encryption and XECB Authentication Modes. In: Fast Software Encryption, 8th International Workshop, FSE 2001 Yokohama, Japan, April 2-4, 2001, Revised Papers. S. 92–108, 2001.
- [GM84] Goldwasser, Shafi; Micali, Silvio: Probabilistic Encryption. Journal of Computer and System Sciences, 28(2):270–299, 1984.
- [Ho15] Hoang, Viet Tung; Reyhanitabar, Reza; Rogaway, Phillip; Vizár, Damian: Online Authenticated-Encryption and its Nonce-Reuse Misuse-Resistance. In: Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I. S. 493–517, 2015.
- [IT09] ITU-T: Interfaces for the Optical Transport Network (OTN). Recommendation G.709/Y.1331, International Telecommunication Union, Geneva, December 2009.
- [Iw06] Iwata, Tetsu: New Blockcipher Modes of Operation with Beyond the Birthday Bound Security. In: Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. S. 310–327, 2006.
- [Ju08] Jutla, Charanjit S.: Encryption Modes with Almost Free Message Integrity. J. Cryptology, 21(4):547–578, 2008.
- [KR11] Krovetz, Ted; Rogaway, Phillip: The Software Performance of Authenticated-Encryption Modes. In: Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. S. 306–327, 2011.
- [KVV04] Kohno, Tadayoshi; Viega, John; Whiting, Doug: CWC: A High-Performance Conventional Authenticated Encryption Mode. In: Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers. S. 408–426, 2004.

- [LRW11] Liskov, Moses; Rivest, Ronald L.; Wagner, David: Tweakable Block Ciphers. *Journal of Cryptology*, 24(3):588–613, 2011.
- [Lu05] Lucks, Stefan: Two-Pass Authenticated Encryption Faster Than Generic Composition . In: *Fast Software Encryption: 12th International Workshop, FSE 2005* , Paris, France, February 21-23, 2005, Revised Selected Papers. S. 284–298, 2005.
- [MT79] Morris, Robert; Thompson, Ken: Password Security - A Case History. *Communications of the ACM*, 22(11):594–597, 1979.
- [MV04] McGrew, David A.; Viega, John: The Security and Performance of the Galois/Counter Mode (GCM) of Operation. In: *Progress in Cryptology - INDOCRYPT 2004, 5th International Conference on Cryptology in India, Chennai, India, December 20-22, 2004*, Proceedings. S. 343–355, 2004.
- [Ro01] Rogaway, Phillip; Bellare, Mihir; Black, John; Krovetz, Ted: OCB: a block-cipher mode of operation for efficient authenticated encryption. In: *CCS 2001, Proceedings of the 8th ACM Conference on Computer and Communications Security, Philadelphia, Pennsylvania, USA, November 6-8, 2001*. S. 196–205, 2001.
- [Ro02] Rogaway, Phillip: Authenticated-encryption with associated-data. In: *Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18-22, 2002*. S. 98–107, 2002.
- [Ro04a] Rogaway, Phillip: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In: *Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004*, Proceedings. S. 16–31, 2004.
- [Ro04b] Rogaway, Phillip: Nonce-Based Symmetric Encryption. In: *Fast Software Encryption, 11th International Workshop, FSE 2004* , Delhi, India, February 5-7, 2004, Revised Papers. S. 348–359, 2004.
- [RZ11] Rogaway, Phillip; Zhang, Haibin: Online Ciphers from Tweakable Blockciphers. In: *Topics in Cryptology - CT-RSA 2011 - The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18* , 2011. Proceedings. S. 237–249, 2011.
- [Wi68] Wilkes, M.V.: *Time-Sharing Computer Systems*. MacDonal computer monographs. American Elsevier Publishing Company, 1968.
- [Wu05] Wu, Hongjun: The Misuse of RC4 in Microsoft Word and Excel. *IACR Cryptology ePrint Archive*, 2005:7, 2005.



**Christian Forler** wurde am 5. Mai 1978 in Landau geboren. Nach seinem erfolgreich abgeschlossenen Studium der Wirtschaftsinformatik an der Universität-Mannheim hat er mehrere Jahre bei der Sirrix AG als Systemprogrammierer und Systemarchitekt an Sicherheitslösungen im Umfeld Trusted-Computing und Virtuelle Private Netzwerke (VPN) gearbeitet. Im Oktober 2010 hat Christian Forler seine Promotion an der Bauhaus-Universität Weimar am Lehrstuhl für Mediensicherheit bei Prof. Dr. Stefan Lucks begonnen. Während seiner Promotion hat er sich nicht nur mit der Robustheit von Verfahren zur authentisierten Verschlüsselung beschäftigt, sondern

hat auch an mehreren internationalen Wettbewerben wie der *SHA-3 Competition* oder *CEASAR* teilgenommen. Bei dem Wettbewerb PHC wurde sein Finalist vom Programmkomitee mit einer besondere Anerkennung gewürdigt. 2015 hat er mit seiner Verteidigung seiner Dissertation seine Promotion mit dem Prädikat *summa cum laude* abgeschlossen. Seit 1. Oktober 2015 ist Christian Forler Professor an der Hochschule Schmalkalden.