Verschlüsselung als Mittel gegen die Überwachung

Selbstverteidigung

Dominique Petersen, Norbert Pohlmann

Nach dem Heartbleed-Desaster ist es nicht damit getan, alle Web-Passwörter und Zertifikate zu ändern, denn Webbrowser können selbst dann auf unsichere Weise kommunizieren, wenn das grüne Zertifikatssymbol erscheint und der Server keine SSL-Lücke mehr aufweist.

Die noch vor Jahresfrist angenommene Vertrauenswürdigkeit des Internets hatte sich schon vor der OpenSSL-Katastrophe als Illusion erwiesen. Seit 2013 ist allgemein bekannt, dass die Geheimdienste praktisch den gesamten Internet-Datenverkehr ausspionieren – vornehmlich in Gestalt der „Five Eyes“: Die Hauptakteure dieser Allianz sind die National Security Agency (NSA) der USA und das britische Government Communications Headquarters (GCHQ), in zweiter Linie gesellen sich Australiens Defence Signals Directorate (DSD), das Communications Security Establishment Canada (CSEC) und Neuseelands Government Communications Security Bureau (GCSB) hinzu.

Wer der Five-Eyes-Allianz möglichst wenig preisgeben will, sollte auf Datensparsamkeit, weniger Cloud- und damit lokalere Dienste sowie Verschlüsselung achten. Datensparsamkeit bedeutet zum Beispiel, dass nicht jedes Handy-Foto samt exakter Geolokalisierung und Personenmarkierung direkt auf Servern von Facebook landen sollte. Die weit verbreiteten Cloud-Dienste, als Rechenwerke oder einfach nur Dateiserver genutzt, liegen meist in Ländern der Five-Eyes-Allianz oder sind ihnen gesetzlich unterstellt. So kann die NSA Dropbox als amerikanisches Unternehmen gemäß dem Patriot Act verpflichten, ihr alle hochgeladenen Dateien preiszugeben. Gleichzeitig müssen die Firmen jegliche Zusammenarbeit mit der NSA dementieren.

Bei den Verschlüsselungsverfahren ist grob zwischen symmetrischer und asymmetrischer Kryptografie zu unterscheiden, die unterschiedlichen Zwecken dienen.

Die symmetrische Verschlüsselung verwendet zum Ver- und Entschlüsseln denselben Schlüssel. Dieses schnelle Verfahren kann große Datenmengen schützen. Da die Schlüssel geheimgehalten oder höchstens zwischen bekannten Kommunikationspartnern ausgetauscht werden, eignen sich symmetrische Verfahren gut zur Festplatten- oder Archivverschlüsselung. TLS/SSL verwendet ebenfalls symmetrische Kryptografie, beispielsweise beim Übertragen von Web-Inhalten per HTTPS. Allerdings muss hier vorab der Schlüsselaustausch sicher erfolgen (mittels asymmetrischer Verschlüsselung und Schlüsselaustausch, siehe unten). Für TLS/SSL sind diverse symmetrische Verschlüsselungsverfahren in Gebrauch, die sich hinsichtlich ihrer Aktualität und Sicherheit voneinander unterscheiden (siehe Tabelle „Gängige symmetrische Kryptoverfahren“).

Kryptografie zur Selbstverteidigung

Der Algorithmus RC4 spielt in vielen Implementierungen immer noch eine große Rolle, obwohl er etliche Schwachstellen aufweist. Namhafte Kryptologen nehmen an, dass beispielsweise die NSA

RC4-verschlüsselte Datenströme live im Klartext mitlesen kann. Der Experte Bruce Schneier schätzt, dass die NSA Kryptoverfahren mit einer Komplexität von 2^{80} brechen kann. Von RC4 (Komplexität nur 2^{15}) ist also dringend abzuraten.

Eine besondere Bedeutung kommt dem Advanced Encryption Standard (AES) zu. Von belgischen Kryptologen 1998 im Rahmen eines Wettbewerbs unter dem Namen Rijndael eingereicht, hat ihn das US-amerikanische National Institute of Standards and Technology (NIST) als Norm definiert, die weltweit zum Einsatz kommt. Zurzeit gilt AES als sicher. Sogar die US-Regierung verschlüsselt vertrauliche Dokumente mit AES 256. Weitere Indizien für die Sicherheit liefern Zeitungsberichte, dass die NSA im Utah Data Center vor allem AES brechen will. Das wird lange dauern, aber dank der unvorstellbar großen Kapazität lassen sich die verschlüsselten Daten beliebig lange speichern und eines Tages vielleicht entschlüsseln. Engagierten Anwendern bleibt nur, Perfect Forward Secrecy zu verwenden (siehe unten).

Gängige symmetrische Kryptoverfahren

Symmetrische Verschlüsselung	Anzahl Bits	Aufwand zum Brechen	Sicherheit
DES	56	2^{39}	niedrig
RC4	128	2^{15}	extrem niedrig
Triple-DES	168	2^{112}	mittel
AES	128/192/256	$2^{126,1}/2^{169,7}/2^{254,4}$	hoch/sehr hoch/maximal
Camellia	128/192/256	$2^{128}/2^{192}/2^{256}$	hoch/sehr hoch/maximal

Stand: Sommer 2014

Asymmetrische Verschlüsselung

Verfahren	Anzahl Bits	Aufwand zum Brechen	Sicherheit
DSA (nur signieren)	1024	Wegen SHA-1 nur 2^{61}	niedrig
RSA	1024/2048/4096/8192	$2^{70,4}$/ $2^{94,6}/2^{126,3}/2^{167,8}$	niedrig/mittel/hoch/sehr hoch
ECC (NIST)	192/224/256/384/521	$2^{96}/2^{112}/2^{128}/2^{192}/2^{260,5}$	mittel bis maximal, wenn nicht manipuliert
ECC (Curve25519)	256	2^{128}	hoch

Stand: Sommer 2014

Wie der japanische Algorithmus Camellia zeigt, kann ein Verfahren ohne den Segen der NIST einen hohen Bekanntheitsgrad erreichen. Camellia gehört seit Jahren zur berühmt-berüchtigten Open-Source-Bibliothek OpenSSL und zum Firefox seit Version 3. Heute ist Camellia sogar die Präferenz im Firefox. Die asymmetrischen Verschlüsselungsverfahren basieren auf je einem öffentlichen (Public Key) und einem geheimen Schlüssel (Private Key), die jeweils dem Ver- und dem Entschlüsseln dienen. Der öffentliche Schlüssel verschlüsselt personalisiert die Daten, denn nur mit dem korrespondierenden privaten Schlüssel lassen sich die Daten korrekt entschlüsseln. Mit dem Verfahren lassen sich in umgekehrter Richtung auch Nachrichten unterschreiben, genauer gesagt eine kryptografische Prüfsumme (Hash-Wert) der zu signierenden Daten.

Angreifer können, wenn alles richtig programmiert ist, nur mit extrem hohem Aufwand den geheimen Schlüssel aus dem öffentlich bekannten ableiten. Daher sollte ein Private/Public-Key-Paar zurzeit 2048 Bit aufweisen, besser 4096 Bit.

Asymmetrische Verfahren eignen sich wegen des hohen Rechenaufwands nicht für große Datenmengen. Praktische Anwendungen kombinieren daher die Vorteile der asymmetrischen und symmetrischen Methoden miteinander, indem sie die Nutzdaten mit einem temporär generierten Sitzungsschlüssel symmetrisch und diesen wiederum asymmetrisch verschlüsseln.

Die Tabelle „Asymmetrische Verschlüsselung“ zeigt derzeit gebräuchliche asymmetrische Verfahren. Der Digital Signature Algorithm (DSA) findet nur zum Erstellen und Verifizieren von Signaturen Verwendung. Da er intern SHA-1 verwendet, gilt damit auch DSA als unsicher. In der Vergangenheit kam praktisch nur RSA (nach den Anfangsbuchstaben der Erfinder Rivest, Shamir und Adleman) als asymmetrisches Verfahren zum Einsatz. Dessen Sicherheit hängt vor allem von der Schlüssellänge ab. Das Verfahren basiert auf der Schwierigkeit, große Zahlen in ihre Primfaktoren zu zerlegen. Zurzeit empfiehlt das Bundesamt für Sicherheit in der Informationstechnik (BSI) mindestens 2048 Bit bei RSA. Wer sichergehen will, sollte auch hier mindestens 4096 Bit einsetzen. Falls es die NSA schafft, zum Faktorisieren großer Zahlen Quantencomputer einzusetzen, ist das RSA-Verfahren grundlegend neu zu bewerten [1].

ECC, eine relativ junge Disziplin asymmetrischer Kryptografie, basiert grob gesagt auf Operationen mit Punkte-Paaren auf bestimmten elliptischen Kurven [2]. Rein mathematisch ist die ECC-Kryptografie bei gleicher Anzahl Bits deutlich sicherer als RSA. Allerdings enthalten gängige Krypto-Bibliotheken die ECC-Funktionen noch nicht lange, sodass hier noch praktische Verbreitungsprobleme auftreten.

Im Zuge der Snowden-Enthüllungen wurde leider bekannt, dass die NSA den ECC-Standardisierungsprozess beim NIST zu ihren Gunsten beeinflusst hat. Namhafte Kryptologen nehmen daher an, dass die NSA manche ECC-NIST-Verfahren – und zwar die mit einem „r“, konkret „secp{256,384,521}r1“ – deutlich leichter brechen kann als angenommen. Hintergrund ist die gezielte Vorgabe bestimmter Startwerte („Seeds“) für die Verschlüsselung. In diesen Fällen gilt ECC – anders als bei pseudozufälligen Seeds – als nicht vertrauenswürdig. Eine Abwandlung von ECC in der Form „secp{256,384,521}k1“ gilt hingegen noch als sicher, da die Seeds pseudozufällig sind. Von der modernen ECC-Variante Curve25519, die nicht von der NIST stammt, sind bisher keine Manipulationen bekannt.

Schlüsselaustauschprotokolle

Kommunikationspartner, die symmetrische Verschlüsselungsverfahren verwenden wollen, ohne einander je begegnet zu sein, müssen sich auf einen gemeinsamen geheimen Schlüssel einigen, und das auf möglichst sichere Weise – selbst über das unsichere Internet. Diese Aufgabe erfüllen Schlüsselaustauschprotokolle wie Diffie-Hellman (DH), auch in einigen Anwendungsfällen als Perfect Forward Secrecy bekannt. Beide Parteien einigen sich darüber auf einen gemeinsamen geheimen Sitzungsschlüssel. Aktuelle Kryptobibliotheken enthalten Implementierungen von Diffie-Hellman mittels elliptischer Kurven (ECDH), allerdings mangelt es noch an deren Verbreitung.

Schlüsselaustauschprotokolle sind vor dem Hintergrund der Snowden-Enthüllungen von besonderer Bedeutung. Behörden, die Zugriff auf private Schlüssel benötigen, zwingen die beteiligten Dienstleister notfalls zur Herausgabe. Ein prominentes Beispiel: der ehemalige sichere Maildienst Lavabit, den auch Edward Snowden benutzte. Dessen Betreiber Ladar Levison gehörte zu den wenigen, die sich öffentlich gegen die Spio-

nageaktivitäten der NSA wehrten. Als er per Gesetz gezwungen wurde, die privaten Schlüssel an den Geheimdienst abzugeben, und somit die Sicherheit seiner Nutzer nicht mehr gewährleisten konnte, stellte er den Betrieb ein und warnte medienwirksam vor der Nutzung von Diensten amerikanischer Firmen. Wenn sogar die verhältnismäßig kleine Firma Lavabit schon unter Generalverdacht stand und sich im Visier der NSA befand, stellt sich die Frage, wie viele andere Unternehmen die USA bereits zur Schlüsselherausgabe gezwungen hat. Als Lehre bleibt für Anwender, den Einsatz ausländischer Dienste genau zu prüfen.

Kryptografische Hash-Funktionen reduzieren eine beliebige Menge von Input-Daten auf eine Zahl bestimmter Länge, einen sogenannten Fingerabdruck, der zum Beispiel 160 Bit umfasst. Als Einwegfunktionen bieten sie keine praktikable Möglichkeit, aus diesem Fingerabdruck auf die ursprünglichen Daten zu schließen. Außerdem muss eine Hash-Funktion weitgehend kollisionsresistent sein: Zwar gibt es theoretisch unendlich viele Input-Werte, die zu demselben Fingerabdruck führen. Die Hash-Werte zweier einander stark ähnelnder Eingaben müssen jedoch möglichst unterschiedlich sein.

Da sich die asymmetrischen Verfahren aus Durchsatzgründen nicht für größere Datenmengen eignen, sind Hash-Funktionen von zentraler Bedeutung für den Signiervorgang. Der besteht im Verschlüsseln des überschaubaren Fingerabdrucks mit dem Private Key – sodass jeder Empfänger mithilfe des Public Key verifizieren kann, dass die Nachricht tatsächlich vom Absender stammt und nicht manipuliert wurde.

Auch bei den Hash-Funktionen gibt es große Unterschiede in puncto Sicherheit (siehe Tabelle „Kryptografische Hash-Funktionen“). Den Klassiker MD5 kann aktuelle PC-Hardware innerhalb weniger Minuten oder gar Sekunden brechen. Das gewährleistet die Integrität einer Nachricht bei weitem nicht mehr.

Integritätsprüfung mit Hash-Funktionen

Nach wie vor im Einsatz ist das mittlerweile als unsicher geltende SHA-1. Anwender sollten auf die SHA-2- und RIPE-MD-Familien ausweichen, die eine mittlere bis sehr hohe Sicherheit bieten. Als sich das Ende von SHA-1 abzeichnete, rief das NIST vor einigen Jahren einen weiteren Wettbewerb aus, um den nächsten Secure Hash Algorithm SHA-3 Ende 2012 bekannt zu geben: „Keccak“, an dem wieder belgische Kryptologen beteiligt waren.

Aber auch hier kristallisiert sich eine mögliche Beeinflussung seitens der NSA heraus: Ursprünglich sollten SHA-3-Fingerabdrücke bis zu 512 Bit lang sein. Als die NSA-Affäre gerade so richtig ins Rollen kam, gab das NIST im August 2013 eigene Änderungen bekannt – angeblich aus Performance-Gründen. So sollte es zum einen nur noch die Hash-Längen 128 und 256 Bit geben, was das Brechen drastisch vereinfacht, zum anderen Änderungen am Algorithmus selbst. Diese Meldung veranlasste viele führende Kryptologen, der Sicherheit von SHA-3 zu misstrauen und davor zu warnen, dass das NIST eigentlich sichere kryptografische Funktionen systematisch unterminierte. Nach einer Welle der Empörung in der wissenschaftlichen Welt warf das NIST im November 2013 das Ruder jedoch herum und ließ die ursprüngliche kryptografische Stärke von SHA-3 zu.

Jedes Verschlüsselungsverfahren benötigt Zufallszahlen zum Erzeugen von Schlüsseln. Die kryptografische Stärke basiert vor allem auf der Entropie und damit der Qualität des (Pseudo-)Zufalls. Kennt ein Angreifer etwa das Intervall möglicher „Zufallszahlen“, erleichtert ihm das eine Entschlüsselung ungemein. Im November 2013 musste ein NIST-Vertreter einräumen, dass die Behörde einen von der NSA entwickelten, absichtlich schwachen Pseudo-Zufallsgenerator standardisiert und sogar explizit zur Verwendung empfohlen hat. Die NSA dürfte somit Verschlüsselungen, die darauf basieren, deutlich leichter knacken können als öffentlich bekannt.

Aber nicht nur die Standards bergen Risiken. Immer wieder gibt es Meldungen über fehlerhafte Implementierungen in Open-Source-Werkzeugen, die möglicherweise von Geheimdiensten eingeschleust wurden. So lieferte neben dem Heartbleed-Bug auch Java und damit das Android-Betriebssystem keine ausreichende Entropie bei der Schlüsselgenerierung.

Manipulation von Zufallszahlen

Immerhin bieten Open-Source-Produkte eine Chance, dass solche Schwächen eines Tages auffallen. In proprietärer Software sind Fehler oder absichtlich eingebaute Schwächen weitaus schwerer zu finden. Besonders gravierend ist die Implementierung von Hintertüren in Software, die eigentlich mehr Sicherheit ermöglichen soll. So hat die amerikanische Firma RSA bereits 2004 in ihre Software BSAFE den manipulierten Zufallsgenerator Dual_EC_DRBG als Standard eingebaut und dafür angeblich sogar zehn Millionen Dollar von der NSA erhalten. Der nach außen sicherere Algorithmus ermöglichte also der NSA ein leichteres Entschlüsseln von Kommunikationskanälen und beispielsweise E-Mails. Der schwache Zufallsgenerator Dual_EC_DRBG sollte im Auftrag eines anonymen Spenders auch in die beliebte Open-Source-Variante OpenSSL Einzug finden, funktionierte jedoch aufgrund eines Implementierungsfehlers zum Glück gar nicht erst.

Während Hintertüren oder Fehler in Software mit mehr oder weniger Aufwand zu entdecken sind, bleiben sie in Hardware praktisch unauffindbar. Die NSA könnte beispielsweise die beiden großen CPU-Hersteller Intel und AMD zwingen, die Befehlssätze in ihren Produkten anzupassen. Auch bei sicherheitsrelevanten Produkten wie Smartcards oder Trusted Platform Modules (TPM) bleibt normalen Anwendern nur, zu vertrauen: Sie haben keine Chance, schlechte Zufallszahlengeneratoren zu erkennen.

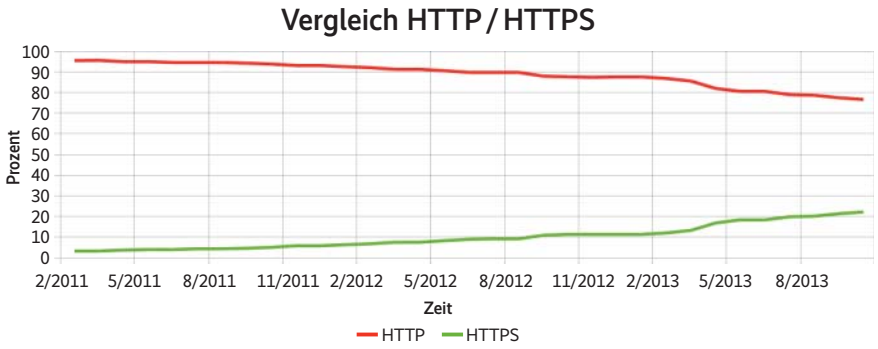
Der Kryptostandard fürs Web schlechthin ist Transport Layer Security (TLS), eine Weiterentwicklung des Secure Sockets Layer (SSL) 3.0 von Netscape. Bis heute ist der Begriff SSL gebräuchlich, selbst wenn er eigentlich TLS bezeichnet. Beide gehören zu den hybriden Verschlüsselungsverfahren, sie kombinieren also symmetrische und asymmetrische Algorithmen, um deren Vorteile zu verei-

Kryptografische Hash-Funktionen

Verfahren	Anzahl Bits	Aufwand zum Brechen	Sicherheit
MD5	128	2^{18}	extrem niedrig
SHA-1	160	2^{61}	niedrig
RIPE-MD	160/256/320	$2^{80}/2^{128}/2^{160}$	niedrig/mittel/hoch
SHA-2-Familie	224/256/384/512	$2^{112}/2^{128}/2^{192}/2^{256}$	mittel/hoch/sehr hoch/maximal
SHA-3-Familie	224/256/384/512	$2^{112}/2^{128}/2^{192}/2^{256}$	mittel/hoch/sehr hoch/maximal

Stand: Sommer 2014

nen. Der Verbindungsaufbau findet via Public-Key-Verfahren statt (optional abgesichert per Schlüsselaustauschprotokoll), die spätere Datenübermittlung mittels symmetrischer Verschlüsselung. Ein Hash-Wert beweist die Integrität der übermittelten Daten. Die 4er Tupel bestehen aus optionalem Schlüsselaustauschprotokoll, Public/Private-Key-Verfahren und dem Hash-Algorithmus. TLS/SSL nennt eine solche Kombination auch Cipher Suite, etwa DHE_RSA_WITH_AES_256 _CBC_SHA. Je nach TLS- oder SSL-Version stehen unterschiedliche Cipher Suites bereit, die sich hinsichtlich ihrer Sicherheit erheblich voneinander unterscheiden.



Der Anteil geschützter Webkommunikation steigt seit Jahren langsam. Ein Snowden-Effekt ist nicht sichtbar (Abb. 1).

Sicherheitsvorgaben für TLS/SSL

Welche Verschlüsselungen ein Anwender tatsächlich nutzt, kann er beeinflussen, indem er in den Browser-Einstellungen diejenigen Cipher Suites markiert, denen er vertraut und unter denen der Server auswählen soll. Als symmetrische Variante kommen AES und Camellia infrage, bei den Hash-Funktionen RIPE-MD und SHA-2/3 mit mindestens 256 Bit. Als asymmetrische Variante empfiehlt sich RSA mit mindestens 2048 Bit. Bietet der Server keine davon an, gibt es keine verschlüsselte Verbindung. Viele große Websites bieten derzeit kaum anerkannt sichere Varianten an. Es müssen also nicht nur die Browser, sondern auch die Webserver sicher genug konfiguriert sein.

Welche dieser Kombinationen tatsächlich im Einsatz sind, lässt sich an zentralen Datenverkehrsknoten messen, etwa mit der Sensorik des Internet-Analyse-Systems (IAS) vom Gelsenkirchener Institut für Internet-Sicherheit – if(is). Das IAS zählt anonymisierte Protokoll-Header-Informationen für seine Statistiken. Es identifiziert Kommunikationsparameter der OSI-Schichten 2 (Sicherungsschicht) bis 7 (Anwendungsschicht), etwa ein gesetztes TCP-Syn-Flag beim Verbindungsaufbau, die verwendete TLS/SSL-Verschlüsselung oder einen speziellen Browser-Typ, aber keine personenbeziehbaren Informationen

wie IP-Adressen oder gar die Nutzdaten (Payload) eines Pakets. Bereits 2005 bestätigte der damalige Bundesdatenschützer Peter Schaar die Unbedenklichkeit des IAS.

HTTPS auch nach dem Patch oft unsicher

Derzeit unterscheidet das IAS über 3,1 Millionen Kommunikationsparameter im Datenstrom und speichert sie in einer Datenbank für Analysen oder Prognosen. Für repräsentative Zahlen aus Deutschland kamen Rohdaten vom zentralen Internet-Knoten DE-CIX zur Auswertung.

Während Anfang 2011 nur etwa 5 % des gesamten Web-Datenverkehrs verschlüsselt waren (HTTPS), stieg der Anteil bis Anfang 2013 auf 12 % (Abb. 1). Kurz vor den Snowden-Enthüllungen im Juni 2013 waren fast 20 % des Web-Verkehrs verschlüsselt, derzeit sind es rund 23 %. Der Trend zu mehr HTTPS begann jedoch lange vor der NSA-Affäre und deren Bekanntwerden hatte keinen erkennbaren Einfluss auf den Trend.

Derzeit arbeitet die IETF an dem Nachfolger HTTP 2.0, zu dem sowohl Google als auch Microsoft jeweils einen Vorschlag eingereicht haben. Googles SPDY (sprich „speedy“) soll sogar bei jeder HTTP-Kommunikation zwingend TLS verwenden, jedoch ermöglicht der aktuelle Draft von HTTP 2.0 der IETF weiterhin unverschlüsselte Verbindungen, die als abwärtskompatible HTTP-1.x-Kanäle gelten. Standard soll bei der Nachfolgerversion aber HTTPS sein, wenn im Browser nichts anderes ausgewählt ist.

HTTPS respektive SSL/TLS wurden vor Heartbleed oft als grundsätzlich geeigneter Schutz propagiert. Das hängt jedoch in erheblichem Maße davon ab, welche Verschlüsselungsarten – die Cipher Suites – zum Einsatz kommen (Abb. 2).

Ende 2013 fiel gut die Hälfte (rund 53 %) der HTTPS-Verbindungen dank AES/Camellia 256 Bit in die Kategorie „sehr sicher“. Als noch sicher genug gelten weitere knapp 4 % (AES/Camellia 128 Bit). Während die mittlere Qualität (3DES 168 Bit) mit nicht einmal einem Promille so gut wie keine Rolle mehr spielt, sind über 43 Prozent der Verbindungen unzureichend verschlüsselt.

Als wäre das nicht schlimm genug, gibt es – wenn auch wenige – HTTPS-Verbindungen, etwa mit DES 40 Bit, RC2 et cetera, die praktisch keinen Schutz mehr bieten. Besonders tückisch sind HTTPS-Sitzungen, die Klartext übertragen: Obwohl die Adresszeile ein Schlosssymbol anzeigt und dem Nutzer ein Mindestmaß an Sicherheit suggeriert, gehen die Daten unverschlüsselt durchs Internet. Derzeit betrifft das glücklicherweise nur jede 200 000. Verbindung.

SSL 3.0 sollte inzwischen tabu sein. Das Bundesamt für Sicherheit in der Informationstechnik empfiehlt sogar die konsequente Nutzung von TLS 1.2. Während nur noch rund 1,3 % aller HTTPS-Pakete mittels SSL-3.0-Varianten verschlüsselt sind, entfällt der Löwenanteil mit etwa 65 % auf TLS 1.0, also bestenfalls RC4 und 3DES. Auf TLS 1.1 entfallen derzeit 9,2 %, auf TLS 1.2 immerhin schon 24,5 %. Gerade die verwendeten SSL/TLS-Versionen zeigen, inwieweit aktuelle Verfahren im Einsatz sind.

Anwender stehen ebenso in der Pflicht wie Webserver-Betreiber, diesen Missstand zu beheben und so schnell wie möglich nur noch sichere HTTPS-Verbindungen zuzulassen. Dazu gehört es, Client- und Host-Software sowie deren Betriebssysteme aktuell zu halten – sonst bleiben sicherheitskritische Updates eines Tages aus. Beispielsweise beherrscht der Internet Explorer 8, bei Windows XP standardmäßig an Bord, maximal RC4 und 3DES, mit RC4/MD5 als noch unsichererer Standardeinstellung. Da Ende 2013 immer noch fast ein Drittel aller Web-Anwender Windows XP nutzten, ergibt sich ein bedrohliches Szenario.

Leider garantiert die Aktualität der Software allein keine sicheren Verbindungen. Zusätzlich gilt es in Browsern und Webservern unsichere Cipher Suites zu deaktivieren und alle kryptografischen Bibliotheken aktuell zu halten, insbesondere aufseiten der Server und nicht nur wegen Heartbleed. Sonst kann es passieren, dass Webseiten nicht mehr per HTTPS erreichbar sind, nämlich wenn Browser und Server sich nicht auf eine Cipher Suite einigen können. Sowohl die Client-Administratoren als auch Inhaltsanbieter können ihre Browser- und Webserverkonfiguration online prüfen, etwa bei SSLLabs (www.ssllabs.com).

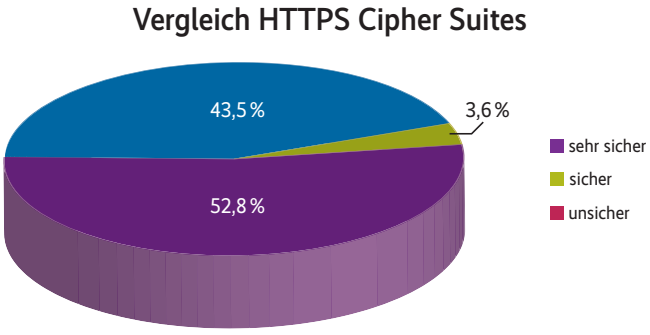
Fazit und Ausblick

Der allumfassenden Überwachung können durchschnittliche Anwender nur mit sorgfältiger Datensparsamkeit, konsequenter Nutzung lokaler statt Cloud-Dienste sowie sicherer Verschlüsselung aller gespeicherten und übertragenen Daten entgehen. Echte Sicherheit vor den Geheimdiensten der Five-Eyes-Allianz ist allerdings auch dann nicht gegeben, da diese einfach zu viel überwachen und die nationalen Gesetze ihnen beliebige Freiheiten einräumen, etwa die Zwangsherausgabe privater Kryptoschlüssel und den Einbau von Hintertüren. (un/ur)

Literatur

[1] Klaus Schmeh, Thomas Zeggel; Verschlüsselung: Quantencomputer und Post-Quanten-Kryptografie; *iX* 3/2014, S. 104

[2] Christine Priplata, Colin Stahlke; Kryptisch elliptisch; Public-Key-Verfahren mit ECC; *iX* 9/2001, S. 109



Immerhin über die Hälfte des HTTPS-Datenverkehrs findet auf anerkannt sichere Weise statt (Abb. 2).

Dominique Petersen
ist wissenschaftlicher Mitarbeiter am Institut für Internet-Sicherheit der Westfälischen Hochschule Gelsenkirchen und seit Januar 2007 Leiter des Forschungsbereichs Internet-Frühwarnsysteme.

Prof. Norbert Pohlmann
ist geschäftsführender Direktor des Instituts für Internet-Sicherheit und Professor für Verteilte Systeme und Informationssicherheit an der Westfälischen Hochschule Gelsenkirchen und Leiter des Master-Studiengangs Internet-Sicherheit.