TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Informatik Institut für Systemarchitektur, Professur für Privacy and Security

Gutachten zur Schutzwirkung des datr-Cookies

Sebastian Rehms, Stefan Köpsell und Florian Tschorsch

31. Januar 2024

1 Ausgangspunkt und Fragestellung

Das Gutachten widmet sich der Fragestellung, inwiefern das sogenannte datr-Cookie für die Sicherheit der Facebook-Webseite relevant ist. Dabei geht es unter anderem darum zu untersuchen, ob das datr-Cookie für die Sicherheit hilfreich bzw. notwendig ist. Dies betrifft dabei auch die Bewertung von Aussagen, ob das datr-Cookie *unerlässlich* bzw. *unbedingt erforderlich (inklusive Speicherzeit)* für die Diensteerbringung ist bzw. ob das datr-Cookie ein *unverzichtbares Element* von dem Stand der Technik entsprechenden Sicherheitsmechanismen zum Schutz der Facebook-Webseite ist. Ferner wird darauf eingegangen, ob (1) das *Erstellen von Fake- und Spam Profilen verhindert* wird, (2) das *Diebstahlrisiko von Nutzerkonten reduziert* wird (3) es *Inhalte von Nutzern vor Diebstahl schützt* und (4) ob *Denial-of-Service-Angriffe verhindert* werden. Abschließend wird noch betrachtet, ob eine *längere Speicherzeit erforderlich ist, um zuvor durch den Nutzer verwendete Browser identifizieren* zu können.

2 Technischer Hintergrund

Das Übertragen von Webseiten und anderen Web-basierten Inhalten erfolgt mittels des Hypertext Transfer Protokolls (HTTP). Das HTTP-Protokoll basiert auf einem zustandslosen Anfrage-Antwort Prinzip. Die anfragende Instanz (Client) stellt Anfragen an einen Endpunkt (Server) für einen Teil einer Webseite. Aus Sicht des Servers lassen sich auf der Anwendungsschicht Anfragen desselben Clients nicht zusammenführen oder sich von Anfragen anderer Clients unterscheiden, d. h. es lässt sich aus mehreren Anfragen keine virtuelle Sitzung eines Clients konstruieren, die eine bestimmte Historie oder einen bestimmten Zustand des Interaktionsverlaufs desselben Clients mit dem Server repräsentiert. Die im Rahmen der Kommunikation auftretenden Client-IP-Adressen eignen sich nur bedingt als Distinktionsmerkmal, da mehrere Clients dieselbe IP-Adresse nutzen können. Der Server ist zur Sitzungsverwaltung insofern darauf angewiesen, dass der Client in seinen Anfragen Informationen über eine Sitzung mitschickt. Das kann ein eindeutiger Identifikator für die Sitzung sein, sodass im Server der Zustand der Sitzung vorgehalten und mit der Anfrage zusammengeführt werden kann. Alternativ kann der Client Informationen über den Zustand der Sitzung bei seinen Anfragen mitschicken. Eine etablierte Technik für beide Ansätze sind sogenannte HTTP-Cookies (kurz: Cookies).

Cookies sind Informationen, die in Webbrowsern[1] abgelegt und von ihnen verwaltet werden. Sie haben immer einen Namen und einen zugeordneten beliebigen Textwert Dieser Textwert ist auf eine maxi-

[1] Genauer: Cookies werden in Instanzen von Webbrowsern, etwa in Form von Profilen abgelegt, wobei die am weitesten verbreitete Nutzung immer dasselbe Profil nutzt. Der in gängigen Browser verfügbare „Private Modus" dient unter anderem genau dem Zweck, neue und isolierte Instanzen von Webbrowsern einfach zugänglich zu machen.

male Länge beschränkt. Optional können weitere Daten beigefügt werden, etwa eine Lebensdauer und eine zugehörige Domain.

Ein Webserver hat verschiedene Möglichkeiten, ein Cookie „zu setzen", genauer: Er kann anfordern, dass ein Webbrowser ein Cookie mit gewünschten Werten lokal erzeugt und vorhält. Prinzipiell können in einem Webbrowser Cookies auch ohne Zutun des Servers mit beliebigen Werten erstellt, verändert und gelöscht werden, da sie ausschließlich lokal durch den Webbrowser verwaltet werden.

Das Standardverhalten sieht vor, dass bei allen Anfragen an einen Server alle Cookies, die für die zugehörige Domain gesetzt sind, mitgeschickt werden. Insbesondere muss es sich hierbei nicht immer um denselben Server handeln, der das Setzen des Cookies veranlasst hat – die Domain des Servers und die Domainangabe im Cookie ist hier entscheidend.

Die Löschung von Cookies nach Ablauf der gewünschten Lebenszeit obliegt ebenso dem Webbrowser. Das aktuelle Standardverhalten von etablierten Webbrowsern (etwa Google Chrome) erlaubt eine maximale Lebensdauer von 400 Tagen (ca. 13 Monate). Der Server kann sich die Cookies merken, deren Setzung er von einem Client erbeten hat, und dann einen Abgleich mit den Cookies vornehmen, die ein Client an den Server schickt, um die Sitzung zuzuordnen bzw. den Zustand zwischen Client und Server zu synchronisieren.

Die von einem Server ausgelieferte Webseite kann Inhalte von Servern anderer Domains anfordern, d. h. von sogenannten „dritten Parteien". Hierbei werden dann unter normalem Verhalten des Webbrowsers Anfragen an andere Domains gestellt. Liegen für diese Domains Cookies im Browser vor, so werden diese an die entsprechenden Server übertragen. In der Konsequenz kann also der Server der dritten Partei den Zustand einer Sitzung über Webseiten, die nicht durch diesen Server ausgeliefert werden, hinweg abfragen.

Basierend auf der Funktionsweise der Cookies lassen sich drei wesentliche Anwendungsfälle identifizieren:

- das Etablieren von Sitzungen (Sessions), welche mehrere Web-Anfragen zusam-menfassen
- das Personalisieren von Webseiten, beispielsweise das Speichern von Webseiten spezifischen Einstellungen
- das Tracking von Nutzenden, d. h. die Erstellung von Nutzungs- und Nutzerprofilen (auch über unterschiedliche Webseiten hinweg) beispielsweise im Rahmen der personalisieren Werbung

3 Technische Darstellung des datr-Cookies

Die Darstellungen in diesem Abschnitt beziehen sich nur auf das von einer externen Position und damit auf das lediglich als regulärer externer Teilnehmer des Systems beobachtbaren Verhalten (Analyse- bzw. Beobachtungszeitpunkt: Januar 2024). Ins-besondere Informationen über interne technische Abläufe seitens Facebook bzw. eine entsprechende technische Dokumentation der Abläufe und der zugehörigen Motivation waren für dieses Gutachten nicht vorhanden. An dieser Stelle sei auch auf die Untersuchungen einer Arbeitsgruppe der KU-Leuven verwiesen[2], deren Untersuchungen bezüglich des Cookie-Verhaltens von Facebook sich über den Zeitraum 2015–2022 erstrecken. Die dortigen Darstellungen zum Zeitpunkt 2022 decken sich mit den unseren in weiten Teilen, insbe-sondere in allen, die hier tiefer betrachtet werden.

Der Wert des datr-Cookies ist ein 24 Zeichen langer Text, der scheinbar aus zufälligen Symbolen aus dem klein- und großgeschriebenen Alphabet, Ziffern sowie dem Minuszeichen und dem Unterstrich zusammengestellt wird (z. B.: `5CW_ZQ7mqK6nhYDRI-YXHnh6`). Wie oben bereits erwähnt liegen allerdings keine konkreten Informationen vor, wie genau der Inhalt des datr-Cookies generiert wird und welche Bedeutung der Werte ggf. hat.

Das Cookie wird mit einer Lebensdauer von 13 Monaten für die Domain `.facebook.com` und damit auch für alle Subdomains gesetzt. Es wird durch das Attribut `Same-Site=None` auch in Browsingkontexten, in denen Facebook die Rolle einer dritten Partei annimmt, bei Anfragen an Facebook mitgesendet.

[2] Dimova, Yana, et al. „Tracking the Evolution of Cookie-based Tracking on Facebook." Proceedings of the 21st Workshop on Privacy in the Electronic Society. 2022.

Das finale datr-Cookie wird außerdem als `HttpOnly` markiert, das heißt, dass es nicht durch im Browser im Rahmen der Anzeige der Webseite ausgeführte Programme (JavaScript) ausgelesen werden kann. Es wird weiter als „Secure“ markiert, womit der Webbrowser angewiesen wird, das Cookie nicht über ungesicherte Verbindungen zu senden, sondern lediglich über verschlüsselte (TLS-gesicherte) Verbindungen.

Das Setzen des datr-Cookie erfolgt auf eine spezielle Art und Weise. Konkret erfolgt dies mit Hilfe eines Programms (JavaScript), welches beim Anzeigen der Facebook-Hauptseite im Browser ausgeführt wird. Dieses Programm sorgt dabei dafür, dass das datr-Cookie erst durch die Interaktion mit dem Cookie-Consent-Banner (kurz: Cookie-Banner) überhaupt im Browser gesetzt wird. Direkt nach der Interaktion mit dem Cookie-Banner wird dabei für jede der beiden möglichen Nutzerentscheidungen („Optionale Cookies ablehnen“ bzw. „Alle Cookies erlauben“) *immer* das datr-Cookie gesetzt. Ohne nachfolgende Interaktionen mit der Facebook-Webseite macht die Interaktion mit dem Cookie-Banner also aus technischer Außensicht zunächst keinen Unterschied bezüglich der danach vorhandenen Cookies. Die Cookie-Banner-Entscheidung führt insofern erst nach weiteren Interaktionen mit der Webseite zu (technischen) Auswirkungen.

Da das datr-Cookie unabhängig von der Consent-Entscheidung immer gesetzt wird, besteht für Facebook auf Grund des im datr-Cookie hinterlegten Wertes die Möglichkeit der eindeutigen Verknüpfung von Interaktionen des Nutzers mit der Facebook-Webseite sowie anderen Webseiten, welche Facebook-Inhalte einbinden.

Anfragen, in denen Facebook als dritte Partei fungiert, führen auf Grund der Konstruktion nur zum Mitsenden des datr-Cookies, wenn vorher auf der Facebookseite mit dem Cookie-Banner interagiert wurde.

Die folgenden Beobachtungen sind zum prinzipiellen Verständnis des Gutachtens nicht nötig, können aber hilfreich bei der Bewertung sein.

- Der Wert des datr-Cookies ist bereits im JavaScript-Quellcode der zuerst ausgelieferten Seite von Facebook nach einer regulären Anfrage enthalten, wird jedoch an dieser Stelle noch nicht im Browser gesetzt. Erst durch die Interaktion mit dem Cookie-Banner wird beim Klicken auf einen der beiden Buttons („Optionale Cookies ablehnen“ oder „Alle Cookies erlauben“) in jedem Fall ein Cookie im Browser lokal durch das JavaScript-Programm gesetzt. Dieses Cookie hat den Namen `_js_datr` und enthält den im JavaScript-Programm hinterlegten Wert. Die nach Interaktion mit dem Cookie-Banner vom Server gesendet Antwort beinhaltet dann einen Header, der das eigentliche datr-Cookie mit demselben Wert setzt – das nur kurzfristig gesetzte `_js_datr`-Cookie wird daraufhin entfernt. Ein reines Laden der Seite ohne weitere Interaktion führt insofern nicht zu im Browser gespeicherten Cookies – erst durch die Interaktion mit dem Cookie-Banner wird das datr-Cookie als einziges nicht-optionales Cookie gesetzt.
- Interagiert man mit der Webseite von einer US-Amerikanischen IP-Adresse aus, wird kein Consent-Banner ausgeliefert. Das `_js_datr`-Cookie wird direkt nach dem Aufruf der Webseite gesetzt und nach jeglicher weiteren Interaktion mit der Webseite in den datr-Cookie umgewandelt.
- Bemerkenswert ist auch, dass beim Schrittweisen Durchgehen des Ablaufs der Cookie-Banner nicht erscheint und somit weder das `_js_datr`-Cookie noch das eigentliche datr-Cookie im Browser verfügbar werden. Ein darauffolgender Webseiten-Abruf führt zu einer Fehlermeldung des Servers.

Wichtig im Zusammenhang mit der Funktionsweise des datr-Cookies ist die Tatsache, dass ohne ein gültiges datr-Cookie, d. h. ein datr-Cookie, welches zuvor von Facebook gesetzt wurde, keine (über die Startseite hinausgehende) sinnvolle Interaktion mit der Facebook-Webseite möglich ist. Löscht man beispielsweise im Browser manuell das datr-Cookie und versucht sich direkt über die Login-Seite bei Facebook anzumelden, so erscheint lediglich eine Fehlermeldung (siehe Abbildung 1), welche auf generelle technische Probleme verweist (und sich insbesondere von der Fehlermeldung unterscheidet für den Fall, das Nutzername oder Passwort falsch sind). Das gleiche Verhalten lässt sich auch beobachten, wenn man den Wert des datr-Cookies ändert. Dies lässt darauf schließen, dass durch den Browser an Facebook übermittelte datr-Cookies auf Gültigkeit überprüft werden.

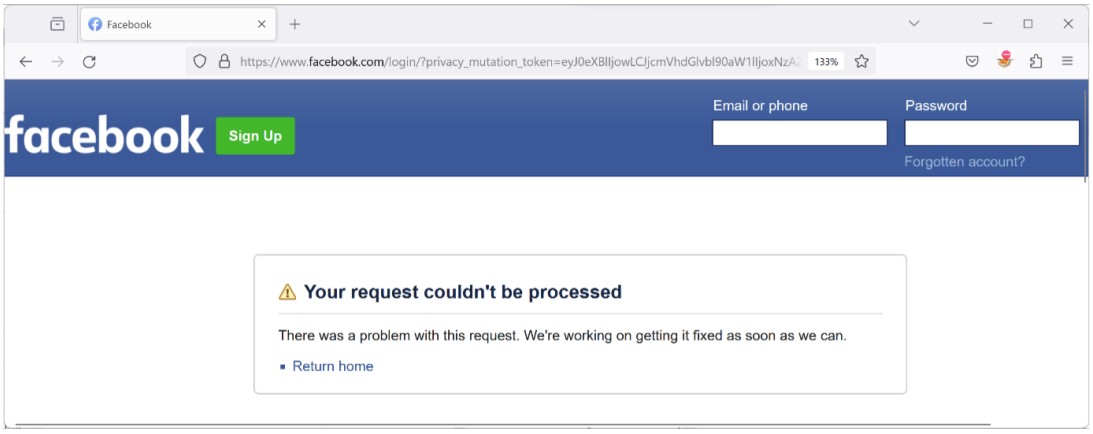

Abbildung 1: Fehlermeldung beim Facebook-Login ohne gültigen datr-Cookie.

Angemerkt sei, dass es möglich ist, die Lebensdauer des datr-Cookies im Browser manuell beliebig zu verkürzen, da die Lebensdauer nur lokal gespeichert aber nicht beim Versenden des datr-Cookies mit übertragen wird.

4 Einschätzung zur Schutzwirkung des datr-Cookies

Die Darstellungen und Einschätzungen in diesem Abschnitt basieren auf dem von außen erkennbaren Verhalten des Gesamtsystems sowie daraus abgeleiteten Interpretationen. Der Fokus liegt hier auf einer Beurteilung des Beitrags des datr-Cookies zur Sicherheit der Facebook-Webseite auf Basis des beobachteten Verhaltens.

Zunächst ist allgemein festzustellen, dass sich IT-Sicherheitsrisiken aus Bedrohungen er-geben, wobei erfolgreiche Angriffe die Realisierung von Bedrohungen unter Ausnutzung von Schwachstellen sind. Es ist insofern zu betrachten, inwiefern das datr-Cookie einen Beitrag in der Verhinderung oder Reduktion der Erfolgschance von Angriffen erreicht und somit der Realisierung einer Bedrohung vorgebeugt wird. Eine wesentliche Kategorie von Angriffen sind die bereits im Rahmen der Fragestellung erwähnten Denial-of-Service-Angriffe (DoS-Angriffe), deren Ziel darin besteht, die Verfügbarkeit eines Dienstes (hier: der Facebook-Webseite) zu beeinträchtigen.

Im Folgenden werden zwei Klassen bezüglich der Umsetzung von Angriffen unterschieden, welche weiter unten gesondert diskutiert werden:

1. Angriffe mit niedriger Frequenz: Hiermit sind Angriffe gemeint, welche nicht auf einer hohen Anzahl an Anfragen oder auf einem hohen Durchsatz basieren, d. h., dass hier wenige gezielte Interaktionen mit der Webseite passieren, wie sie etwa durch einen Menschen in dieser Frequenz und Umfang durchgeführt werden können.
2. Angriffe mit hoher Frequenz: Hiermit sind Angriffe gemeint, welche in irgendeiner Weise darauf angewiesen sind, möglichst viele bzw. umfangreiche (Datenvolumen) Anfragen bezüglich irgendeiner logischen Schicht oder Schnittstelle zu stellen. Zur Umsetzung derartiger Angriffe sind üblicherweise automatisierte Systeme nötig, welche eine hohe Anzahl an Operationen möglicherweise über einen längeren Zeitraum hinweg durchführen können.

Zunächst sei nochmal darauf hingewiesen, dass ein Nicht-Vorhandensein des datr-Cookies oder Verändern seines Wertes zu einer Fehlermeldung führt (siehe Abschnitt 3). Nur beim Mitsenden des korrekten Wertes, kann etwa ein regulärer Login-Versuch (unabhängig vom Erfolg des Logins) durchgeführt werden. Dadurch wird technisch sichergestellt, dass eine Interaktion mit der Facebook-Webseite (im Sinne eines üblichen Nutzerverhaltens) nur nach einer Interaktion mit dem Cookie-Banner möglich ist. Insofern impliziert das Vorhandensein des datr-Cookies eine zuvor durchgeführte Interaktion mit dem Cookie-Banner.

Beim Verändern des Cookie-Textwertes kann theoretisch ein anderer gültiger Wert eingesetzt werden. Zunächst ist dies möglich, wenn bereits ein weiterer gültiger Wert bekannt ist, etwa durch das vorherige Initialisieren einer Sitzung. Andererseits könnte versucht werden, einen gültigen Wert zu erraten. Die Wahrscheinlichkeit hierfür ist aufgrund der Länge (und unter der Annahme von zufällig generierten Werten) verschwindend gering. Ein Angreifer kann ein gültiges datr-Cookie insofern praktisch nicht raten.[3]

4.1 Angriffe mit niedriger Frequenz

Nachfolgend wird argumentiert, dass das datr-Cookie generell keinen relevanten Schutz bezüglich Angriffen mit niedriger Frequenz bietet. Ein Beispiel für einen potenziellen Angriff mit niedriger Frequenz ist das gezielte Ausnutzen einer Schwachstelle in der Implementierung der Webseite beispielsweise mit dem Ziel, sich in die Server von Facebook „einzuhaken".

Es lassen sich zwei Fälle von niederfrequenten Angriffen unterscheiden. Zunächst jegliche Angriffe, bei denen das datr-Cookie prinzipiell nicht vorhanden sein muss. Dieser Fall ist für einen Angreifer leicht zu erreichen, da er das datr-Cookie einfach nicht mitzusenden braucht, da Cookies immer lokal im Browser verwaltet werden. Insbesondere braucht es hierfür kein tieferes technisches Verständnis, da schon der weit verbreitete „Privat"- oder „Inkognito"-Modus von Browsern genau dies leistet: Cookies werden nicht in neue Sitzung mit überführt. Das datr-Cookie ist hier keine Ausnahme.

Bei der zweiten Gruppe von Angriffen wird ein gültiges datr-Cookie benötigt. Die jeweilige Sitzung des Angreifers wird hierbei durch das datr-Cookies eindeutig identifiziert und der Angriff kann theoretisch durch diese Identifikation einer durchgängigen Sitzung zugeordnet werden. Schutzmaßnahmen auf Sitzungsebene sind dann auch an diese Identifikation gebunden. Da der Angreifer schwer gültige datr-Cookies erraten kann, ist er auf eine Abfragefolge, die effektiv der Interaktion mit dem Cookie-Banner entspricht, angewiesen, um einen vom Server akzeptierten datr-Cookie zu erhalten. Die sehr lange Lebensdauer des datr-Cookies erlaubt eine Identifikation von Sitzungen über lange Zeiträume und damit theoretisch dann auch Schutzmaßnahmen über diese langen Zeiträume hinweg.

Derartige Schutzmaßnahmen gegen niederfrequente Angriffe, bei denen ein gültiges datr-Cookie vorhanden sein muss, lassen sich allerdings sehr leicht für einen Angreifer umgehen, da er jederzeit ohne weiteres durch reguläre Interaktion mit dem Cookie-Banner ein neues datr-Cookie vom Server erfragen kann , sofern noch kein gültiges datr-Cookie vorhanden ist. Auch hier ist wieder das Löschen eines vorher vom Server angefragten datr-Cookies durch die technische Funktionsweise von Cookies trivial, wodurch immer ein neues gültiges datr-Cookie erlangt werden kann. Der dadurch entstehende Aufwand ist annähernd irrelevant, da hier angenommen wurde, dass es sich um niedrige Frequenzen der Interaktion handelt. Es sei auch darauf hingewiesen, dass jeder Schritt automatisierbar ist – d. h. ein manuelles Durchführen von Angriffen auch bei geringer Frequenz ist nicht nötig. Etablierte Testtechniken in der Webentwicklung sind ubiquitär verfügbar und genau für solche Fälle leicht anpassbar.

4.2 Angriffe mit hoher Frequenz

Nachfolgend wird argumentiert, dass der Beitrag des datr-Cookies zur Sicherheit in Fällen von Angriffen mit hoher Frequenz gering, jedoch nicht vernachlässigbar ist. Jedoch lässt sich hier auch argumentieren, dass die lange Lebensdauer des datr-Cookies die Sicherheit sogar schmälern kann.

Angriffe, die in irgendeiner Form auf hohe Frequenzen setzen, können als Opfer zum einen die Nutzer der Plattform selbst haben, etwa indem Passwörter von Nutzern durch Ausprobieren erraten werden sollen oder aber den Betreiber der Server, etwa indem durch hohe Last auf den Diensten deren Verfügbarkeit eingeschränkt werden soll (sogenannte Denial-of-Service Angriffe).

[3] Abschätzung einer groben oberen Schranke für die Wahrscheinlichkeit des Erratens einen gültigen datr-Cookie zu erraten: Der datr-Cookie-Textwert hat 24 Stellen bei einer Alphabetgröße von $26 \cdot 2 + 10 + 2 = 64$ (26 Klein- und 26 Großbuchstaben, 10 Ziffern sowie Minuszeichen und Unterstrich). Dies ergibt $64^{24} \approx 2,3 \cdot 10^{43}$ verschiedene Textwerte. Es wird ferner angenommen, dass nicht mehr als 10^{15} gültige datr-Cookies bei hinterlegt sind (jeder Mensch (10^{10}) interagiert mit der Facebook-Webseite mit verschiedenen Geräten/Browsern; insgesamt 10^5 verschiedene Sessions). Die Wahrscheinlichkeit einen gültigen Wert zu erraten ist demnach $p < 10^{28}$.

Zunächst lässt sich das Verhalten des datr-Cookies als ein logischer Verbindungsaufbau zwischen Server und Client auf Anwendungsschicht interpretieren. Damit der Client akzeptierte Anfragen an den Server schicken kann, muss er zunächst das datr-Cookie durch die Interaktion mit dem Cookie-Banner abrufen. Zielt der Angreifer auf eine hohe Anzahl an Anfragen ab, kann die notwendige Interaktion mit dem Cookie-Banner durchaus den Aufwand erhöhen. Dies liegt daran, dass für jede potentiell begrenzte Menge an Anfragen ein neues, gültiges datr-Cookie notwendig ist. Die zugehörige Interaktion mit dem Cookie-Banner erhöht den Aufwand auf Seiten des Angreifers entsprechend. Zu berücksichtigen ist hier allerdings, dass auf Grund der verwendeten Protokolle (TLS und TCP) dem Angreifer bereits Aufwand für Etablierung der Verbindungen zum Server entsteht.

Die allermeisten etablierten Schutzmechanismen gegen hohe Last setzen typischerweise unterhalb der Anwendungsschicht auf Transport- und Netzwerkschicht an. Im Gegensatz zum datr-Cookie können diese Netzwerk-basierten Maßnahmen, wie sie oft von Content-Distribution-Netzwerken (CDNs) angeboten werden, als Industriestandard angesehen werden. Sie wirken besonders effektiv, da sie kaum von Nutzern (und auch Angreifern) beeinflusst werden können.

Natürlich kann man dennoch auch auf Anwendungsschicht eine (zusätzliche) Schutzmaßnahme gegen hohen Anfragelast einsetzen. Durch eine eindeutige Identifikation einer aufgebauten Sitzung auf Anwendungsebene kann diese in ihrem Verhalten analysiert und gegebenenfalls gesondert reguliert/limitiert werden. Es ermöglicht theoretisch auch ein Zusammenführen der Sitzung mit unterliegenden Schichten in komplexen Szenarios. Im einfachsten Fall kann die Sitzung dadurch an verschiedenen Stellen blockiert werden z. B. kann ein Fehlverhalten auf Anwendungsschicht auf IP-Ebene unterbunden werden.

Um die Schutzwirkungen gegen hochfrequente Anfragen bestmöglich zu erreichen, sollte das datr-Cookie eine vergleichsweise kurze Lebensdauer haben. Andernfalls kann ein Angreifer über einen längeren Zeitraum gültige Werte für das datr-Cookie niederschwellig vom Server anfragen und sammeln. Dadurch könnte er sich dann aus einer größeren Menge an vom Server akzeptierten datr-Cookies bedienen, um doch wieder Angriffe mit hoher Frequenz durchzuführen. Die Anzahl an gesammelten datr-Cookies hängt dabei von der Lebensdauer des datr-Cookies ab: je länger diese ist, um so mehr Zeit hat der Angreifer für das Sammeln gültiger datr-Cookies. Insofern ist die vergleichsweise lange Lebensdauer des datr-Cookies für die Verhinderung von Hochfrequenzangriffen kontraproduktiv.

5 Beantwortung der Fragestellungen

5.1 Verhinderung der Erstellung von Fake- und Spam-Profilen

Es erscheint unklar, auf welche Art und Weise das datr-Cookie das Erstellen von Fake- bzw. Spam-Profilen verhindert oder auch nur erschwert. So war es in niedriger Frequenz möglich, mehrere Profile mit zufällig gewählten, realistisch klingenden Namen und nahezu identischer E-Mail-Adresse anzulegen. Bei einer automatisierten hochfrequenten Erstellung von Profilen reduziert sich die Schutzwirkung auf die in Abschnitt 4.2 ausgeführten Überlegungen.

5.2 Reduzierung des Diebstahlrisikos von Nutzerkonten

Es lässt sich nicht feststellen, dass das datr-Cookie den Diebstahl von Nutzerkonten effektiv verhindern kann. So können im Allgemeinen niederfrequente Anmeldeversuche durch das datr-Cookie nicht verhindert werden (siehe Abschnitt 4.1), beispielsweise wenn der Angreifer die Login-Daten des Nutzers bereits kennt. Bestenfalls könnte das datr-Cookie die unberechtigte Verwendung der Nutzerdaten erschweren, um so dass Missbrauchsrisiko zu reduzieren. Allerdings war es möglich, in einem Browser ohne gesetztes datr-Cookie die Login-Daten eines zuvor beispielhaft angelegten Nutzerkontos zu verwenden. Darüber hinaus war auch eine erfolgreiche Anmeldung von einen völlig unabhängigen Rechner möglich. In beiden Fällen erfolgte keine Benachrichtigung per E-Mail über die potenziell verdächtigen Login-Versuche. Hier hätte das datr-Cookie (in dem Fall konkret: das Nicht-Vorhandensein des erwarteten datr-Cookies) eine Schutzwirkung entfalten können. Dies wurde jedoch augenscheinlich nicht genutzt.

Eine andere Variante eines Angriffs bezüglich des Diebstahls von Nutzerkonten besteht darin, Kombinationen aus Nutzername und Passwort durchzuprobieren (sogenannte Brute-Force-Angriffe). Hier könnte das datr-Cookie prinzipiell die Frequenz der Login-Versuche limitieren. Wie in Abschnitt 4.2 allerdings dargelegt, ist eine Limitierung, welche ausschließlich an das datr-Cookie gebunden wird, nur eingeschränkt wirkungsvoll. Stattdessen erscheint es sinnvoller, die Anzahl fehlerhafter Login-Versuche (bevor weitere Versuche blockiert werden) an den Account selber zu binden und die Überprüfung/Ermittlung der Anzahl der Fehlversuche Server-seitig durchzuführen. Dies ist Stand der Technik und erfordert insbesondere kein datr-Cookie.

5.3 Schutz vor Diebstahl von Nutzerinhalten

Es konnte nicht festgestellt werden, dass das datr-Cookie den Diebstahl von Nutzerinhalten verhindert bzw. signifikant erschwert. Anzumerken ist hier, dass davon ausgegangen wird, dass mit „Diebstahl von Nutzerinhalten" das Kopieren und unberechtigte Verwendung von Nutzerinhalten, welche auf Facebookseiten hinterlegt sind, gemeint ist.

Für das automatisierte Abfragen öffentlich verfügbarer Inhalte mit hoher Frequenz reduziert sich die Schutzwirkung abermals auf die in Abschnitt 4.2 ausgeführten Überlegungen.

Für Inhalte, die nur nach erfolgreichem Login abrufbar sind, ist zunächst eine erfolgreiche Anmeldung notwendig. Für die Etablierung der Sitzung ist dabei das `c_user`-Cookie in Kombination mit dem `xs`-Cookie verantwortlich. Das datr-Cookie spielt nur eine untergeordnete Rolle (siehe Abschnitt 3 und Abschnitt 4.1). Auch im Falle eines Zugriffs auf Inhalte nur nach Anmeldung war es jedenfalls in praktischen Untersuchungen möglich, die Inhalte zu abzurufen. Insofern hat das datr-Cookie den Zugriff auf Nutzerinhalte nicht verhindern können.

5.4 Verhinderung von Denial-of-Service-Angriffen

Es konnte eine geringfügige Schutzwirkung im Hinblick auf Denial-of-Service-Angriffe festgestellt werden (siehe Abschnitt 4.2). Die lange Lebensdauer des datr-Cookies ist in diesem Fall allerdings als kontraproduktiv zu betrachten und reduziert die Wirksamkeit.

5.5 Lange Speicherzeit zur Identifizierung von verwendeten Browsern

Diese kann nur sehr bedingt nachvollzogen werden. Wird über einen langen Zeitraum dasselbe datr-Cookie verwendet, so liegt die Vermutung nahe, dass auch derselbe Browser verwendet wurde, da Cookies im Allgemeinen Browser-spezifisch sind. Für den Fall, dass ein nicht-angemeldeter Nutzer im zeitlichen Verlauf unterschiedliche Browser verwendet, bleibt unklar, wie hier die Verkettung stattfindet (und insbesondere welche Rolle das datr-Cookie spielt) so dass es möglich ist, die Browser-Nutzungshistorie eines Nutzers nachzuvollziehen: Unterschiedliche Browser erhalten zunächst unterschiedliche datr-Cookies aus denen nicht offensichtlich nachvollziehbar ist, dass sie zum selben Nutzer gehören. Diese Verknüpfung ist möglich, nachdem sich der Nutzer unter Benutzung verschiedener Browser (mit unterschiedlichen datr-Cookies) mit demselben Nutzer-Account angemeldet hat. Allerdings braucht es in diesem Fall das datr-Cookie nicht, um die Historie zu speichern. Hierfür reicht das `c_user`-Cookie.

Zusammenfassend lässt sich daher feststellen, dass nicht nachvollzogen werden kann, warum das datr-Cookie für die Identifizierung der verwendeten Browser notwendig sein sollte. Insofern kann das Argument der Browser-Historie dann auch nicht als Begründung für die lange Speicherzeit (Lebensdauer) dienen.

5.6 Unerlässlichkeit zur Umsetzung von Sicherheit der Facebookseite

Es kann nicht nachvollzogen werden, warum das datr-Cookie für die Umsetzung der Sicherheit der Facebookseite unerlässlich sein sollte. So konnte bezüglich mancher der gewünschten bzw. vorgebrachten Schutzziele keine starke Wirksamkeit festgestellt werden. In vielen Fällen existieren alternative bzw. sogar effektivere Verfahren, welche nicht auf das datr-Cookie angewiesen sind. Insgesamt erscheint es nur sehr schwer vorstellbar, dass das datr-Cookie tatsächlich unerlässlich für die Gewährleistung der Sicherheit der sein soll.

5.7 Abschließende Bemerkung

Wenn das datr-Cookie tatsächlich einen stärkeren Beitrag zur Sicherheit leistet als hier eingeschätzt, so ist eine diesbezügliche gutachterliche Analyse nicht ohne weitere Offenlegung interner Abläufe und Schutzmechanismen von Facebook möglich.