universität wien

AUSZUG DER DIPLOMARBEIT / EXCERPT OF THE DIPLOMA THESIS

Titel der Diplomarbeit / Title of the Diploma Thesis

„Kekse ohne Salz schmecken nicht“
Ein innovatives Unterrichtskonzept zur Vermittlung von Security im Webdatenbereich

verfasst von / submitted by

Simon Marik

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Magister der Naturwissenschaften (Mag.rer.nat.)

Wien, 2017 / Vienna, 2017

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 190 884 313
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Lehramtsstudium UniStG UF Informatik und Informatikmanagement UniStG - 10/2003 UF Geschichte, Sozialkunde und Polit. Bildg. UniStG - 10/2008
Betreut von / Supervisor:	ao. Univ.-Prof. Dipl.-Ing. Dr. Renate Motschnig
Mitbetreut von / Co-Supervisor:	Ass.-Prof. Mag. Dr. Christian Cenker

Standards in der Webentwicklung

0.0.1 Cookies

Dadurch, dass HTTP ein zustandsloses Protokoll ist, geraten Daten spätestens dann in Vergessenheit, sobald die Website an den Client gesendet und die Verbindung wieder getrennt wurde. Im Jahr 1994 implementierte Netscape im hauseigenen Browser schlussendlich das Cookie, welches nur von jener Webseite gelesen werden konnte, von der es auch geschrieben wurde. Somit stellte es eine sichere Art dar, um Informationen dauerhaft über Seiten hinweg zu speichern, obwohl es anfänglich eher einem schlechten Ruf unterlag, da der Host darüber feststellen kann, wie oft eine BenutzerIn die jeweilige Webseite aufruft und was dort gemacht wird. Viele Menschen hatten plötzlich große Sorge um ihre Privatsphäre innerhalb des Internets, wodurch unter anderem auch zahlreiche Gerüchte entstanden. Beispielsweise wurde Cookies nachgesagt, dass sie jede Information auf der Festplatte lesen könnten, wodurch viele Zeitschriften und Blogs auf diesen Zug aufsprangen und sogar zur Anwendungsdeaktivierung von Cookies im Webbrowser rieten. Mittlerweile hat sich die vorerst angespannte Situation wieder beruhigt und Cookies werden im Allgemeinen von der Mehrheit akzeptiert.[1]
Prinzipiell stellen Cookies lediglich reine Textinformationen dar, welche auf Aufforderung eines Webservers durch den Internetbrowser auf der Festplatte des Clients abgespeichert werden.[2] Wenn die gleiche Webseite oder eine Seite der gleichen Domäne erneut aufgerufen wird, so schickt der Browser die im Cookie enthaltenen Textinformationen an den Webserver zurück. Der Webserver selbst ist aber lediglich in der Lage, ihm bereits bekannte Informationen im Cookie abzuspeichern, sodass auch nur solche Informationen vom Webbrowser an ihn zurückgesandt werden. Die oben genannte Einschränkung kann aber auch soweit technisch umgangen werden, dass andere Domänen ebenfalls Gebrauch von einem abgespeicherten Cookie machen können. In diesem Fall spricht man allerdings von Cookies von Drittanbietern *(Abb. 2.3.7)*.[3] Selbstverständlich besteht auch die Möglichkeit, in aktuellen Webbrowsern wie Safari, Chrome, Firefox oder Opera, die Speicherung von Cookies komplett zu deaktivieren. Diese Option kann aber unter Umständen dazu führen, dass der ordnungsgemäße Aufbau einer Seite, sowie ein reibungsloses Surfen an sich,

[1] vgl. *Hudson* (2005), S.181.
[2] s. *Cookie Central*: The Cookie Concept, (Stand: 04.10.2016).
[3] vgl. *Peyton* (2002), S.53 f.

nicht mehr gewährleistet werden kann.[4] Grundsätzlich kann man also sagen, dass es sich bei Cookies im weiteren Sinne um eine Art Cachespeicher handelt, durch den eine bestimmte Anzahl von Informationen gespeichert wird, wodurch eine BenutzerIn vom Webserver wiedererkannt werden kann.[5]

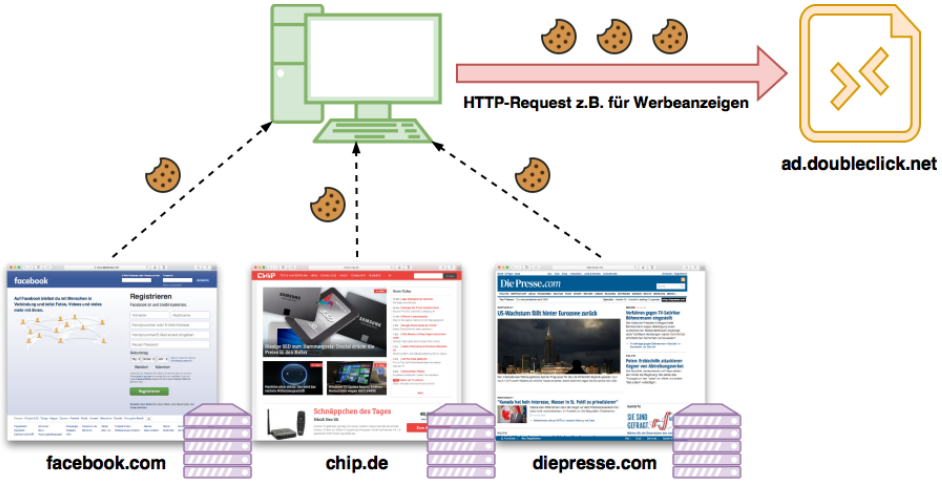

Abbildung 0.0.1: Abfrage von Cookieinformationen durch Drittanbieter anderer Domänen

Als WebentwicklerIn kann man sowohl auf Cookies als auch auf Sessions problemlos zugreifen. Es ist schlussendlich mit beiden Techniken möglich, Daten einer Seite über mehrere andere hinweg zu speichern, wobei diese auch einige Unterschiede aufweisen. Cookies können nämlich – ganz im Unterschied zu Sessions – auf eine lange Lebensdauer eingestellt werden, wodurch Daten innerhalb eines Cookies über Monate bzw. Jahre gespeichert werden können. Da Cookiedaten clientseitig gespeichert werden, ist die Speicherung sensibler Daten allerdings nur dann sinnvoll, wenn z.B. in einem Cluster mehrerer Webserver gearbeitet wird. Sessions wären hier nicht nützlich, da nach der ersten fertig abgearbeiteten Anfrage eines Webservers die Informationen im Cluster schlichtweg nicht mehr verfügbar wären. Außerdem weisen viele moderne Webbrowser eine Speicherplatzbegrenzung bei der Speicherung von Cookies auf, um schädliche Internetseiten davor zu hindern, immens viel Speicherplatz in Form von Cookies auf der Festplatte zu verschwenden. Schlussendlich beruht die Entscheidung, ob man in seinem Webprojekt nun Cookies oder doch lieber Sessions verwenden sollte darauf, ob die Daten noch am nächsten Tag für die BenutzerIn zur Verfügung stehen sollen. Falls dem so sein sollte, kommt man um Cookies nicht herum. Falls sensible Daten gespeichert werden sollen, ist es empfehlenswert, diese in einer gesicherten Datenbank abzulegen und lediglich eine Referenz zur ID im Cookie zu speichern. Dies deshalb, da Cookies dadurch, dass sie am Computer gespeichert werden, von BenutzerInnen sehr leicht bearbeitet werden können. So könnte man beispielsweise die Benutzer-ID, welche zur automatischen Anmeldung bei einer Webseite

[4] vgl. *Christl* (2014), S.21.
[5] vgl. *Voss* (2003), S.229.

benötigt wird, umschreiben und somit in den Account der jeweils geänderten Benutzer-ID einsteigen. Genau deshalb werden heutzutage auch oft Sicherheitstoken innerhalb solcher Login-Systeme als eine Art zusätzliche Hackerbarriere implementiert, welche sich aus verschiedenen Benutzerinformationen zusammensetzen und mittels der Benutzer-ID und sonstigen Daten einen eindeutig feststellbaren Hashtag ergeben.[6]

Verlagert man nun die gesamte Thematik zurück zu PHP, muss unbedingt erwähnt werden, dass der Aufruf der Methode `setcookie()` immer vor dem HTML-Formular stehen muss, da HTTP alle Headerinformationen vor dem `<body>`-Tag sendet. In den Kopfzeilen werden Informationen wie beispielsweise der Webservertyp (z.B. Apache), die Seitengröße in Bytes sowie andere wichtige Daten übertragen. Cookies gehören ebenfalls zu diesen Headerinformationen, was heißt, dass nach jedem `setcookie()` der Webserver eine Zeile in den Header schreibt. Die `setcookie`-Funktion kann dabei drei Hauptparameter annehmen, nämlich den Cookienamen, den Wert sowie das Datum wann das Cookie wieder verfallen soll *(Abb. 2.3.8)*.[7] Im einfachen Fall wird in einem Cookie genau ein Wert gespeichert (max. 4 Kilobyte). PHP bietet aber auch die Möglichkeit, eine Arrayvariable als Cookie abzuspeichern, wodurch es ermöglicht wird, die Werte in einer Schleife abzuarbeiten. Für den Fall, dass noch komplexere Daten in einem Cookie gespeichert werden sollen, gibt es die Möglichkeit, den Variableninhalt mittels der Methode `serialize()` zu serialisieren, indem eine Zeichenkette aus der komplexen Variable erstellt wird.[8]

```
1  /* Erstellung eines Cookies namens "thesis-cookie" */
2  /* Ablaufdatum: aktuelle Zeit + 5000 Tage */
3  setcookie('thesis-cookie', $data, time() + (86400 * 5000), "/");
4
5  /* Entfernung des Cookies "thesis-cookie" */
6  /* Inhalt wird gelöscht / Ablaufdatum: aktuelle Zeit */
7  setcookie('thesis-cookie', '', time() - (86400 * 5000), "/");
```

Abbildung 0.0.2: Minimalbeispiel zur Erstellung und Entfernung eines Cookies

Der größte Nachteil von Cookies liegt sicherlich darin, dass man sich einfach nicht auf sie verlassen kann. Es lässt sich nämlich nicht mit Gewissheit sagen, ob ein Browser auch wirklich ein Cookie nach einer Sitzung gelöscht hat, oder ob sie eine BenutzerIn aufgrund von Sicherheitsbedenken eigenhändig entfernte bzw. ob sie sogar deaktiviert wurden. Darum ist es generell keine gute Idee, gerade bei sicherheitskritischen Bereichen wie z.B. Anmeldesystemen, Cookies zu verwenden, außer man rüstet sein Login-System mit einem Sicherheitstoken aus. Ein weiterer großer Nachteil von Cookies ist, dass man sie nur vor

[6] vgl. *Hudson* (2005), S.181 f.
[7] vgl. *ebd.*, S.182 f.
[8] vgl. *Kofler & Öggl* (2010), S.171.

dem Ausliefern eines Seiteninhalts erstellen bzw. bearbeiten kann. Diese Einschränkung ist innerhalb der Spezifikation von HTTP geregelt und ist nicht PHP bedingt. Des Weiteren werden Cookies nach dem Aufruf der Methode `setcookie()` lediglich erstellt. Erst nach dem Laden der nächsten Seite wird das Cookie auch im Webbrowser als aktiv angezeigt, was ein häufig auftauchender „Fehler“ in so manchem Forum ist.[9]

0.0.1.1 Rechtliche Aspekte

Jedes europäische Mitgliedsland und jeder Drittstaat regelt den rechtlichen Umgang mit Cookies auf seine eigene Art und Weise. So unterliegen Cookies beispielsweise innerhalb Deutschlands dem TMG (d.h. Telemediengesetz), sowie dem BDSG (d.h. Bundesdatenschutzgesetz).[10] In Österreich werden die Rechtsbestimmungen für Cookies und darüber hinaus wiederum nur im DSG geregelt. Allerdings gibt es auch eine flächendeckende EU-Richtlinie, welche im sogenannten TKG (d.h. Telekommunikationsgesetz) geregelt ist und somit alle EU-Mitgliedsstaaten betrifft. Eigentlich könnte man fast vor lauter Gesetzesbestimmungen und Verhaltensrichtlinien meinen, dass die viele Bürokratie hier den eigentlichen Sinn verfehlt hätte, aber gerade heute, in einer Zeit in der eine Unzahl an Cookies auf den Benutzerrechnern abgespeichert werden, sind diese wichtiger denn je.

Prinzipiell wird in jedem Gesetzestext – wie auch im DSG – zwischen Cookies mit personenbezogenem Inhalt (z.B. Namen), indirekt personenbezogenem Inhalt (z.B. Benutzer-IDs oder auch IP-Adressen) und nicht personenbezogenem Inhalt (z.B. multilinguale Sprachunterstützungsdaten) unterschieden.[11] Aus datenschutzrechtlicher Sicht kann sich allerdings nur dann ein Problem ergeben, wenn Cookies zumindest indirekt personenbezogene Daten zum Inhalt haben, bei denen ein besonderer Schutz zur Geheimhaltung gegeben werden soll. Demnach fallen Cookies, welche nicht personenbezogene Daten speichern und vor dem Setzen akzeptiert wurden, auch nicht unter das DSG. Obwohl Cookies auf den ersten Blick nicht wirklich viel über BenutzerInnen preisgeben, verbergen sich in den Buchstaben- und Zahlenkombinationen oft wichtige und vor allem sensible Benutzerdaten. Der Webseitenbetreiber ist durch das Abrufen solcher Cookies aber dennoch nur in der Lage, einen reinen Maschinenbezug dazu herzustellen. Nach §4 (Abs. 1) DSG sind personenbezogene Daten nämlich nur jene, in denen Angaben über einen Betroffenen gemacht werden, dessen Identität bestimmt oder bestimmbar ist.[12] Das bedeutet, dass die AnwenderInnen über den bloßen Maschinenbezug hinauskommen und einen direkten Personenbezug herstellen müssen, bevor das DSG beim Setzen und Auslesen solcher Cookies greifen kann. Indirekt personenbezogene Daten innerhalb von Cookies entstehen dann, wenn der Ersteller der Cookies die Identität des Betroffenen

[9] vgl. *Kofler & Öggl* (2010), S.171 f.
[10] s. *Meine-Cookies*: Rechtliche Aspekte, (Stand: 05.10.2016).
[11] s. *ebd.*
[12] s. *Rechtsinformationssystem*: Gesamte Rechtsvorschrift für Datenschutzgesetz 2000, (Stand: 05.10.2016).

nicht bestimmen kann, da z.B. die Verwendung von Cookies im Webbrowser durch die BenutzerIn deaktiviert wurde. Für einen „Anderen“, welcher im DSG beispielsweise als Systemadministrator auftritt, würden diese Daten dennoch wieder direkt personenbezogene Daten darstellen, da er dadurch trotzdem in der Lage wäre, die Identität des Betroffenen offen zu legen.[13] Damit man im österreichischen Paragraphendschungel allerdings nicht die Orientierung verliert, werden in der nachfolgenden Tabelle *(Tab. 2.3.1)* die wichtigsten Paragraphen und Absätze zum DSG[14] aufgelistet und näher erläutert.

Paragraphentitel:	regelt folgende Absätze:
§**6 Grundsätzliche Verwendung von Daten**	(Abs. 1) Was Daten lediglich dürfen (Abs. 2) Wo der Auftraggeber die Verantwortung trägt (Abs. 3) Vertretungsregelung ausländischer Auftraggeber (Abs. 4) Verwendung von Daten nach Treu und Glauben
§**7 Zulässigkeit der Verwendung von Daten**	(Abs. 1) Wann Daten lediglich verarbeitet werden dürfen (Abs. 2) Wann Daten übermittelt werden dürfen (Abs. 3) Grundsätze des §6 werden vorausgesetzt
§**8 Schutzwürdige Geheimhaltungsinteressen bei nicht-sensiblen Daten**	(Abs. 1) Wann diese bei nicht-sensiblen Daten nicht verletzt sind (Abs. 2) Keine Verletzung bei publiken Daten oder indirekt personenbezogenen Daten (Abs. 3) Wann diese besonders nicht verletzt sind (Abs. 4) Verwendung von Daten strafbarer Handlungen oder Unterlassungen
§**9 Schutzwürdige Geheimhaltungsinteressen bei sensiblen Daten**	(Abs. 1) Wann diese bei sensiblen Daten nicht verletzt sind
§**12 Genehmigungsfreie Übermittlung und Überlassung von Daten in das Ausland**	(Abs. 1) Bei Datenverkehr mit Vertragsstaaten des Europäischen Wirtschaftsraumes (Abs. 2) Bei Datenverkehr mit Empfängern in Drittstaaten mit angemessenem Datenschutz (Abs. 3) Wo der Datenverkehr ins Ausland noch genehmigungsfrei ist (Abs. 4) Wo der Datenverkehr ins Ausland sonst noch genehmigungsfrei ist (Abs. 5) Rechtmäßigkeit der Datenanwendung des §7 wird vorausgesetzt
§**13 Genehmigungspflichtige Übermittlung und Überlassung von Daten ins Ausland**	(Abs. 1) Wann eine Genehmigung der Datenschutzbehörde einzuholen ist (Abs. 2) Wann die Genehmigung der Datenschutzbehörde zu erteilen ist (Abs. 3) Wann die Genehmigung bei meldepflichtigen Anwendungen zu erteilen ist (Abs. 4) Wie inländische Dienstnehmer eine Genehmigung beantragen können (Abs. 5) Übermittlung von Daten an ausländische Vertretungsbehörden (Abs. 6) Wann der Bundeskanzler zur Einholung einer Genehmigung verpflichtet ist
§**24 Informationspflicht des Auftraggebers**	(Abs. 1) Welche Informationspflichten Auftraggeber einer Datenanwendung haben (Abs. 2) Verwendung von Daten nach Treu und Glauben (Abs. 3) Wann Informationen entfallen dürfen, wenn Betroffene nicht befragt werden (Abs. 4) Keine Informationspflicht bei Anwendungen, die nicht meldepflichtig sind
§**25 Pflicht zur Offenlegung der Identität des Auftraggebers**	(Abs. 1) Wann der Auftraggeber seine Identität in geeigneter Weise offenzulegen hat (Abs. 2) Wann die Identität der betroffenen Person und des Auftraggebers anzugeben ist

Tabelle 0.0.1: Wichtige Paragraphen zum DSG über den Schutz personenbezogener Daten

Wie man sieht, darf der Eingriff in das Grundrecht des Datenschutzes durch verschiedenste Datenübermittlungen nur im erforderlichen Ausmaß und mit besonderer Sorgfalt geschehen, wobei stets auf die Einhaltung der Grundsätze in §6 geachtet werden muss. So ist es zum Glück auch kein Problem mehr, Daten über Staatsgrenzen hinaus zu transportieren, welche durch eine globalisierte Welt und einem internationalen Handel auch dafür

[13] vgl. *Christl* (2014), S.101 ff.
[14] s. *Rechtsinformationssystem*: Gesamte Rechtsvorschrift für Datenschutzgesetz 2000, (Stand: 05.10.2016).

prädestiniert waren. Nichts desto trotz sind die Seitenanbieter dazu angehalten, Informationen an die BenutzerInnen auszuhändigen, ob und inwiefern Cookies auf deren Webseite zum Einsatz kommen und welche Daten diese umfassen. Die Verwendung von Cookies ist jedoch nur solange zulässig, bis die vertraglichen Verpflichtungen beider Seiten erfüllt wurden. Von diesem Gesichtspunkt her ist eine Verwendung der Daten innerhalb solcher Cookies kaum geregelt. In der Praxis passiert es nämlich sehr häufig, dass jene Daten, die eigentlich nach Vertragsende gelöscht werden sollten, nach wie vor auf den Servern der Webseitenanbieter gespeichert bleiben.[15] Als Beispiel lässt sich hier vergleichsweise ein Onlinebestellvorgang bei Amazon nennen. Dort werden Cookies verwendet um beispielsweise den Vertragsabschluss (d.h. Warenkorberstellung, Zahlungsoptionen, Lieferungsart, usw.) zu ermöglichen bzw. zu erleichtern. In diesem Fall wird das Setzen von Cookies bis zum Abschicken der Bestellung als zulässige Handlung gestattet.[16] Aber auch Lernportale, welche beispielsweise zur Übermittlung von Lehrinhalten und der Kommunikation eines geschlossenen Nutzerkreises dienen, kommen heutzutage ohne Cookies einfach nicht mehr aus und unterliegen somit ebenfalls dem DSG. So sind aber auch öffentlich gespeicherte Profilinformationen genauso wie Profilbilder innerhalb solcher Lernplattformen wichtige Streitthemen, welche neben dem Einsatz von Cookies nochmals explizit im TKG der Europäischen Union geregelt wurden.[17]
Aufgrund einiger Richtlinien der EU sowie der Datenschutzrichtlinie für elektronische Kommunikation, wurde in Österreich das TKG im Jahre 2003 erlassen. Obwohl Cookies in dieser Richtlinie gar nicht namentlich genannt werden, können dennoch einige Parallelen dazu gezogen werden. So werden beispielsweise unter Abschnitt 12 „Kommunikationsgeheimnis, Datenschutz“ im §95 die Datensicherheitsmaßnahmen geregelt, welche Betreiber eines öffentlichen Kommunikationsdienstes zu folgenden drei Punkten verpflichten:

- Sicherstellung, dass nur ermächtigte Personen für rechtlich zulässige Zwecke Zugang zu personenbezogenen Daten erhalten
- Schutz der gespeicherten oder übermittelten personenbezogenen Daten vor unbeabsichtigter oder unrechtmäßiger Zerstörung, unbeabsichtigtem Verlust oder unbeabsichtigter Veränderung und unbefugter oder unrechtmäßiger Speicherung oder Verarbeitung, unbefugtem oder unberechtigtem Zugang oder unbefugter oder unrechtmäßiger Weitergabe
- Umsetzung eines Sicherheitskonzepts für die Verarbeitung personenbezogener Daten[18]

„Diese Regelung ist jedoch nicht nur alleine auf Cookies anzuwenden, sondern auch auf alle anderen Webinstrumente, welche Daten auf einem Computer speichern wie z.B.

[15] vgl. *Christl* (2014), S.106 ff.
[16] vgl. *Jahnel* (2001), S.88.
[17] vgl. *Horn* (2015), S.76 f.
[18] s. *Rechtsinformationssystem*: Gesamte Rechtsvorschrift für Telekommunikationsgesetz 2003, (Stand: 05.10.2016).

Web-Bugs oder Spyware.“[19]

So führt einem die unklare Formulierung bezüglich der „Speicherung von Informationen“ zu der Frage, ob diese Regelung nun für sämtliche Cookies oder lediglich für solche mit indirektem Personenbezug gilt.[20] Würde nämlich die Informationspflicht auch bei nicht personenbezogenen Daten bestehen, wäre das ein Widerspruch zu den Regelungen im DSG, da diese Art der Daten nach den Bestimmungen nach nicht schutzwürdig wären.[21] Nichts desto trotz sind nach wie vor manche EU-Mitgliedsstaaten wie z.B. Deutschland bei der Umsetzung der europäischen Cookie-Richtlinie sehr zaghaft, wodurch nationale Interessen nach wie vor über globale gestellt werden, was sich in naher Zukunft auch höchstwahrscheinlich kaum ändern wird. Der wohl größte Kritikpunkt dieser Rechtsbestimmung liegt aber dennoch in der Unverständlichkeit mancher Begriffsregelungen wie z.B. „Opt-In“ und „Opt-Out“. In diesen wird nämlich geregelt, ob eine BenutzerIn seine Zustimmung („Opt-In“) bzw. die Ablehnung eines solchen Services („Opt-Out“) bekannt geben muss. Obwohl sich die meisten Länder, darunter auch Österreich, für die „Opt-In“ Methode entschieden haben, genügen in anderen europäischen Ländern wie Finnland oder Portugal einfache „Opt-Out“ Lösungen.[22]

[19] *Jahnel* (2004), S.339.
[20] vgl. *Jahnel* (2003), S.109.
[21] vgl. *Christl* (2014), S.113.
[22] s. *Artworkz*: Die Cookie-Richtlinie der EU, (Stand: 05.10.2016).

0.0.1.1.1 Security

Zu Beginn des Internets spielte unter WebentwicklerInnen oder gar bei BenutzerInnen die Websicherheit kaum eine Rolle, da aufgrund einfacher statischer Websites wie z.B. durch HTML- bzw. Textdokumente, keine potentielle Angriffsfläche gegeben war, um komplexe Anweisungen an den Webserver oder an Subsysteme zu schicken. Zudem kam hinzu, dass anfangs das Internet hauptsächlich von Forschern benutzt wurde, welche sich untereinander alle bestens kannten und somit einen ordentlichen Umgang mit dem neuen Tool pflegten. Allerdings entwickelte sich die Websecurity nach dem Aufkommen größerer Webapplikationen und der damit einhergehenden Kommerzialisierung in den 1990er Jahren zu einem wichtigen Thema innerhalb der IT-Branche. Durch die frische Sensibilisierung in diesem Segment wurde mit der Zeit selbst den meisten Unternehmen bewusst, dass ihre Webanwendungen durch zahlreiche Firewalls oder verschlüsselte Kommunikationskanäle lediglich auf der Transportebene, aber nicht auf der Applikationsebene gesichert waren. Somit kam es auch rund um das Millennium zu zahlreichen Zwischenfällen und Hackerangriffen, wodurch ein allgemeines Umdenken stattfand. Den Menschen wurde allmählich klar, dass Gefahren selbst im Web drohen können und diese oft einen geringen Aufwand für Angreifer darstellen aber einen horrenden Schaden verursachen können.[23] Wenn man bedenkt, dass es nach wie vor eine Unzahl an Webseiten gibt, die entweder mit versteckten HTML-Formularfeldern, welche Informationen in Form von XML-Dateien oder ähnlichem clientseitig speichern oder sensiblen Benutzerdaten innerhalb von URLs bzw. Textdokumenten bearbeiten, kann dieser Paradigmenwechsel auch gar nicht schnell genug von statten gehen.[24]

So existierten anfänglich nicht einmal ausreichend verbindliche Sicherheitsstandards innerhalb des WWW. Anstatt dessen wurden sogenannte RFC (d.h. „Request for Comment“) oder BCP (d.h. „Best Current Practice“) Hinweise ausgeteilt, welche in gewisser Weise eine Vorbildwirkung für spätere Internetstandards (STD) hatten. Erst im Jahr 2001 wurde durch die Gründung des OWASP (d.h. „Open Web Application Security Project“) im Detail über die Sicherheit von Webapplikationen diskutiert, welches bis heute eine der entscheidenden Säulen im Bereich der Websicherheit darstellt. Dieser Diskurs war auch bereits zwingend notwendig geworden, da manch einer zwischenzeitlich schon glaubte, beispielsweise durch sogenannte „Web-Application-Firewalls“, welche ähnlich wie Netzwerkfirewalls funktionieren, Sicherheitslücken innerhalb des Webs schließen zu können, was sich später natürlich als absoluter Irrglaube herausstellte. Die neuesten Tendenzen in Sachen Websecurity gehen heutzutage viel mehr in Richtung sogenannter „Bug-Bounty-Programme“, in denen Sicherheitsexperten bezahlt Jagd auf Sicherheitslücken machen, wodurch sich Unternehmen auch im weiteren Sinne erhoffen,

[23] vgl. *Schäfers* (2016), S.28 f.
[24] vgl. *Splaine* (2002), S.105.

den boomenden Hackerschwarzmarkt damit eindämmen zu können.[25] Bezieht man nun die gesamte Thematik der Sicherheitsstandards auf Cookies, so fällt einem auf, dass hier ebenfalls ein viel zu geringes Angebot existiert. Die meisten Verhaltensregeln, wie beispielsweise das Setzen oder das Löschen von Cookies, sind allesamt in der SOP (d.h. „Same-Origin-Policy“) geregelt, welche einen eigenständigen Standard für Cookies darstellt.[26]

Um jene Informationen, welche innerhalb von Cookies gespeichert werden, weitgehend zu standardisieren, hat das W3C eine eigene Spezifikation, namens „P3P 1.0“ (d.h. „Platform for Privacy Preferences“) herausgegeben, welche sich bis heute als aktueller Standard im Umgang mit Cookies behaupten kann. Vor allem bei der Speicherung direkt personenbezogener Daten greift der Standard in jedem Belangen und sollte von jeder WebentwicklerIn genauestens beachtet werden, da z.B. für die Verwendung von Third-party Cookies eigene P3P-Informationen innerhalb des Webprojekts zur Verfügung gestellt werden müssen.[27]

0.0.1.2 Probleme mit Cookies

Obwohl Cookies eigentlich der Erleichterung im Umgang mit dem Internet dienen sollten, können diese einem auch ganz einfach zum Verhängnis werden. Die meisten Probleme damit sind jedoch sicherlich auf einen eher leichtsinnigen Benutzerumgang zurückzuführen, obwohl es auch andere Mittel und Wege für potentielle Angriffe gibt. Ursprünglich wurde angedacht, Cookies überhaupt nur für die Dauer der aktuellen Verbindung am Nutzerrechner zu speichern und danach sofort wieder zu löschen. Allerdings verfolgte die Praxis einen komplett eigensinnigen Weg und so kam es dazu, dass Cookies auf den Computern überhaupt nicht mehr von Webseitenbetreibern gelöscht wurden. Ganz im Gegensatz zum Ursprungsgedanken wurden sogar im weiteren Verlauf die Zugriffsrechte der Cookies ausgedehnt, sodass es Drittanbietern ermöglicht wurde, ebenfalls Gebrauch von diesen Cookies zu machen.[28] Die wohl größte Gefahr dieses neuen Usus steckt allerdings nicht an den erweiterten Zugriffsrechten für Dritte, sondern viel eher in der fremden Auswertung, wodurch Personen im schlimmsten Fall identifiziert und auf deren Vorlieben bzw. Interessen Rückschlüsse gezogen werden können. Dies ist auch jener Grund, weshalb wir tagtäglich mit personalisierter Werbung innerhalb von E-Mails oder gar Banneranzeigen im Web konfrontiert sind.[29] Die so gesammelten Datenprofile werden in der Praxis auch häufig verkauft bzw. vermietet, wofür sich bereits ein florierender Markt entwickelt hat.[30]

[25] vgl. *Schäfers* (2016), S.30.
[26] vgl. *ebd.*, S.43.
[27] vgl. *Kofler & Öggl* (2010), S.172.
[28] vgl. *Brandl & Mayer-Schönberger* (1999), S.368.
[29] vgl. *Jahnel* (2001), S.87.
[30] vgl. *Christl* (2014), S.100.

Es stellt an und für sich ein relativ kontroverses Problem dar, dass man Cookies als eine Art Verfolgungstool missbrauchen kann. Dabei kann jeder erdenkliche Schritt, den BenutzerInnen innerhalb des Internets setzen, mitgeloggt werden und somit oft sehr pikante Details von BenutzerInnen offenlegen. So nutzen diese Informationen auch gezielt Unternehmen wie beispielsweise DoubleClick, Globaltrack oder ADSmart, welche Werbebanner auf diversesten Seiten hosten und unter Synonymen wie „Advertising-Rings“ bzw. „Tracking-Networks“[31] im Internet zu finden sind. Dabei kommt die Ring-Bezeichnung nicht von irgendwo, da das Geschäftsmodell einem einzigen Kreislauf ähnelt. Die Kunden solcher Werberinge platzieren einen <img> Tag auf ihrer HTML-Seite, welcher auf eine URL des Werberingwebservers verweist. Sieht ein Webbrowser nun ein solches <img> Element, kontaktiert er den Webserver des jeweiligen Werbeanbieters, um die Bildressource zu erhalten. Wenn dieser Werbebanner das erste Mal heruntergeladen wurde, wird ein Cookie am Computer abgespeichert, welches eine zufällig generierte ID enthält. Ab diesem Zeitpunkt wird dann jedes Mal beim Aufrufen einer Seite auf der solche Werbebanner zum Einsatz kommen, dieses Cookie an den Webserver der Werbeagentur geschickt. Dadurch wird den Werbeanbietern ermöglicht, personalisierte Benutzerprofile zu erstellen, in denen genau festgelegt ist, wie sich das Surfverhalten bzw. einzelne Interessen einer BenutzerIn definieren *(Abb. 2.3.7)*. Aus diesem Grund verwenden moderne Webapplikationen auch meist kodierte bzw. verschlüsselte Cookies, um Dateneinsicht oder Datenraub etwas Einhalt gebieten zu können. Allerdings werden dabei oft nur herkömmliche Verschlüsselungsmethoden ohne jeglicher Authentifizierungsmöglichkeit angewendet, welche sehr leicht zu dechiffrieren sind und somit wieder keinen besonders effektiven Schutz darstellen.[32]

Selbstverständlich stellt das bei weitem noch nicht alle Cookie bezogenen Problemstellungen dar. Wenn man bedenkt, dass Cookies auch oft gerne dazu verwendet werden, um sensible Benutzerdaten wie beispielsweise Benutzernamen oder Passwörter zu speichern, muss eine gewisse Sicherheitsbarriere wie z.B. ein Salted-Hash-Verfahren eingebaut werden, um stets die korrekte Authentifizierung einer BenutzerIn gewährleisten zu können. Fehlt diese gänzlich, wäre jeder Angreifer dazu in der Lage, die Identität einer anderen BenutzerIn zu übernehmen und mittels eines nachgebildeten Cookies in dessen Account oder dergleichen einzusteigen.[33] Diese Art des Angriffs wird im Fachjargon als „Cookie-Replay-Attack“ bezeichnet, da sie im Wesentlichen die Nachahmung eines fremden Benutzercookies durch einen Angreifer beschreibt. Bei einem solchen Angriff wird die zuvor ausspionierte Session-ID (durch „Sniffing“ oder Ausprobieren) verwendet, um sich gegenüber der Webanwendung ausweisen zu können. Sofern man es schafft eine gültige Session-ID zu replizieren, kann man mit dieser Information der Anwendung vorgaukeln,

[31] vgl. *Knuckles & Yuen* (2005), S.502.
[32] vgl. *Khu-smith & Mitchell* (2002), S.134 f.
[33] vgl. *ebd.*, S.133.

eine andere BenutzerIn zu sein und somit auf den kompletten Funktionalitätsumfang zugreifen. Dabei spielen natürlich lange Verfallszeiten von Cookies den Angreifern in die Hände, da beispielsweise nach dem Absturz eines Computers oder dergleichen, Sessions nach wie vor gültig und aktiv sein können. Eine andere Angriffsmethode wäre noch die sogenannte „Cookie-Steal-Attack“, in welcher der Angreifer durch Versenden einer schadhaften URL inkl. implementiertem Skript ein Cookie des Opfers kopieren und sich ganz bequem zurücksenden lassen kann *(Abb. 2.3.9)*. Natürlich kann man auch ganz ohne Cookies im WWW angegriffen werden, wenn man z.B. das sogenannte „Session-Hijacking“ bedenkt. Hier wird nämlich ganz im Gegensatz zur „Cookie-Replay-Attack“, die Session-ID bereits während der Applikationsnutzung ausspioniert und nicht erst im Nachhinein, wodurch ganz unabhängig von etwaigen schlecht implementierten Cookie-Verfahren die aktuelle Nutzersitzung recht komfortabel übernommen werden kann.[34]

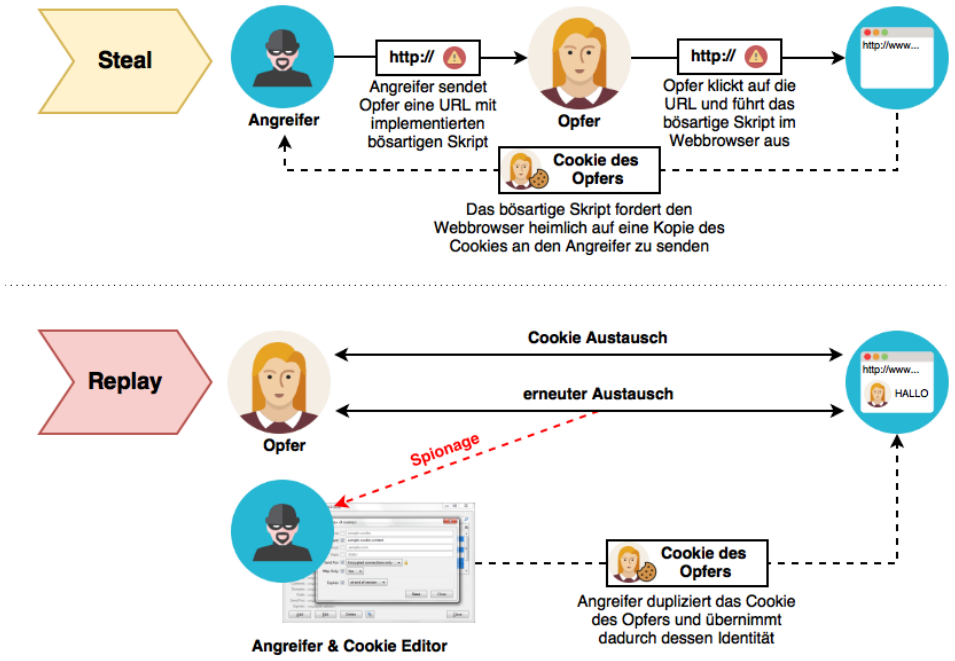

Abbildung 0.0.3: Vorgehensweise von Angreifern bei einer „Cookie-Steal“- bzw. „Cookie-Replay-Attack“

[34] vgl. *Schäfers* (2016), S.92 f.

Literatur- & Quellenverzeichnis

Brandl, Ernst O. & Mayer-Schönberger, Viktor (1999): CPU-IDs, Cookies und Internet-Datenschutz; In: Ecolex, Heft Nr. 5, Manz Verlag, Wien.

Christl, Alexander (2014): Datenschutz im Internet - Cookies, Web-Logs, Location Based Services, eMail, Webbugs, Spyware; 1. Auflage, Disserta Verlag, Hamburg.

Horn, Janine (2015): Rechtliche Aspekte digitaler Medien an Hochschulen; 1. Auflage, Waxmann Verlag, Münster.

Hudson, Paul (2005): PHP in a nutshell; übersetzt von: Speidel, Sigrid & Ulrich; 1. Auflage, O'Reilly Verlag, Köln.

Jahnel, Dietmar (2001): Datenschutz im Internet - Rechtsgrundlagen, Cookies und Web-Logs; In: Ecolex, Heft Nr. 1, Manz Verlag, Wien.

Jahnel, Dietmar (2003): Spamming, Cookies, Web-Logs, LBS und die Datenschutzrichtlinie für elektronische Kommunikation; In: Wirtschaftsrechtliche Blätter, Heft Nr. 3, Manz Verlag, Wien.

Jahnel, Dietmar (2004): Spamming, Cookies, Logfiles und Location Based Services im TKG 2003; In: Österreichische Juristen Zeitung, Heft Nr. 9, Manz Verlag, Wien.

Knuckles, Craig D. & Yuen, David S. (2005): Web Applications - Concepts & Real World Design; 1. Auflage, Wiley Publishing, Michigan.

Kofler, Michael & Öggl, Bernd (2010): PHP 5.3 & MySQL 5.4 - Programmierung, Administration, Praxisprojekte; 1. Auflage, Addison-Wesley Verlag, München.

Khu-smith, Vorapranee & Mitchell, Chris; Kim, Kwangjo (Hrsg.) **(2002):** Information Security and Cryptology - ICISC 2001; 1. Auflage, Springer Verlag, Seoul.

Peyton, Christine (2002): Das neue PC Lexikon für Alle; 1. Auflage, Sybex Verlag, Düsseldorf.

Schäfers, Tim Philipp (2016): Hacking im Web - Denken Sie wie ein Hacker und schließen Sie die Lücken in ihrer Webapplikation, bevor diese zum Einfallstor für Angreifer wird; 1. Auflage, Franzis Verlag, München.

Splaine, Steven (2002): Testing Web Security - Assessing the Security of Web Sites and Applications; 1. Auflage, Wiley Publishing, Indianapolis.

Voss, Andreas (2003): Das große PC & Internet Lexikon; 1. Auflage, Data Becker Verlag, Düsseldorf.

Artworkz: Die Cookie-Richtlinie der EU; siehe online unter: http://www.artworx.at/die-cookie-richtlinie-der-eu-in-oesterreich-ist-opt-in-fuer-personenbezogene-daten-gesetz/ (Stand: 05.10.2016).

Cookie Central: The Cookie Concept; siehe online unter: http://www.cookiecentral.com/c_concept.htm (Stand: 04.10.2016).

Meine-Cookies: Rechtliche Aspekte; siehe online unter: http://www.meine-cookies.org/alles_ueber_cookies/rechtliche_aspekte.html (Stand: 05.10.2016).

Rechtsinformationssystem: Gesamte Rechtsvorschrift für Datenschutzgesetz 2000; siehe online unter: https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=bundesnormen&Gesetzesnummer=10001597 (Stand: 05.10.2016).

Rechtsinformationssystem: Gesamte Rechtsvorschrift für Telekommunikationsgesetz 2003; siehe online unter: https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=Bundesnormen&Gesetzesnummer=20002849 (Stand: 05.10.2016).

Abbildungsverzeichnis

Tabellenverzeichnis