Die TLS Wiederverhandlungs–problematik

(TLS renegotiation issue)

Thomas Gebhardt <thomas.gebhardt@tu-berlin.de> und
Thomas Hildmann <thomas.hildmann@tu-berlin.de>
tubIT - TU Berlin

9. November 2009

Kurzfassung

Im TLS-Protokoll existiert eine schwerwiegende Sicherheitslücke, die spätestens mit dem Erscheinen von verschiedenen Artikeln in der letzten Woche allgemein bekannt sind. Unsicherheit besteht zumeist in der Einschätzung der Schwachstelle und den damit verbundenen Maßnahmen. Dieser Bericht fasst die Informationen unterschiedlicher Quellen in deutscher Sprache zusammen.

Einleitung

Das TLS-Protokoll (Transport Layer Security) ist die standardisierte Weiterentwicklung des SSL-Protokolls (Secure Socket Layer). Aktuell ist die in RFC5246 spezifizierte Version TLS 1.2. Es ermöglicht eine verschlüsselte Datenübertragung und zertfikatbasierte Authentisierung von Clients und Servern. Viele Anwendungsprotokolle, die nicht selbst über geeignete Sicherheitsmechanismen verfügen, werden mit Hilfe von TLS-Verbindungen gesichert. Beispiele hierfür sind: HTTPS, IMAPS, POP3S, LDAPS, etc.

Problembeschreibung

Das TLS-Protokoll erlaubt es, innerhalb einer bestehenden Verbindung die Eigenschaften der Verbindung neu zu verhandeln (renegotiation). Mit Hilfe der Neuverhandlung sind Anwendungsszenarien realisierbar, bei denen ein Client zunächst eine gesicherte Verbindung zum Server aufbaut, bei der die Authentizität des Servers geprüft wird. Eine Authentisierung des Clients kann mittels Neuverhandlung später erfolgen, wenn dies die Anwendung verlangt. Ebenso können kryptographische Parameter wie Schlüssellängen oder verwendete Algorithmen neu verhandelt werden.

TLS enthält bezüglich der Wiederverhandlung einen Entwurfsfehler, der es einem potentiellen Angreifer erlaubt die Wiederverhandlung zu manipulieren und eigene Daten in die Kommunikation zwischen Client und Server einzufügen. Dies verletzt fundamentale Annahmen über die Authentizität von Daten die von den meisten Anwendungen und Protokollen auf Anwendungsebene beim Einsatz von TLS gemacht werden. Daten werden so fälschlich als integer und der Absender als authentisiert betrachtet.

Das Einbringen von Daten durch einen Angreifer ist nur während der Wiederverhandlung möglich. Die Verschlüsselung zwischen Client und Server wird nicht gebrochen, ein Mitlesen der gesamten Kommunikation durch den Angreifer ist nicht unmittelbar möglich.

Es existieren mindestens drei bekannte Angriffszenarien in denen ein Angreifer auf das Auftreten einer TLS-Wiederverhandlung wartet oder diese selbst auslöst.

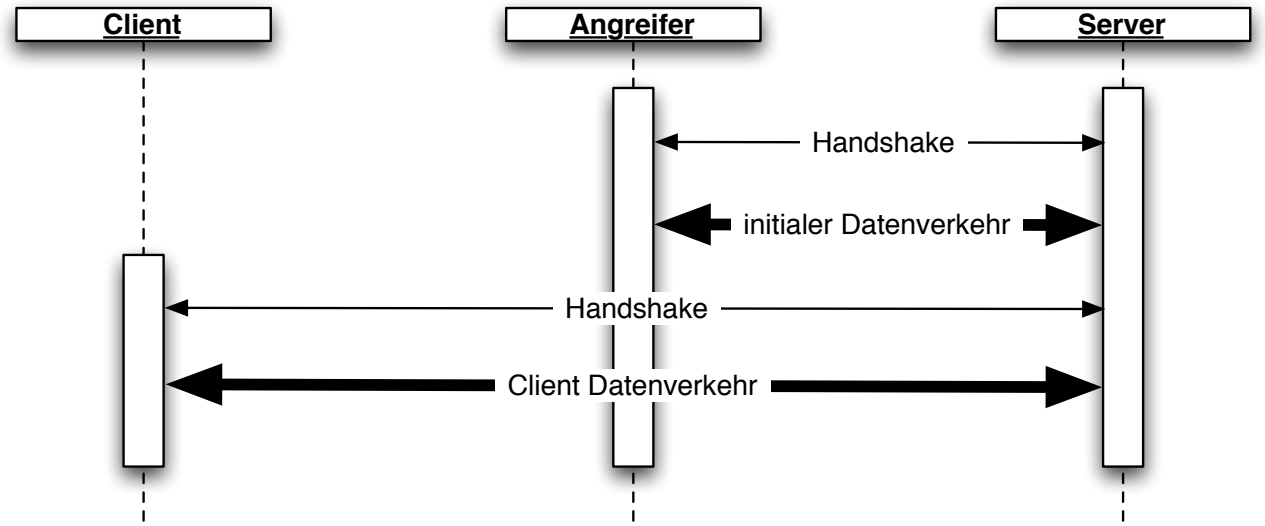
Wie diese Authentisierungslücke ("authentication gap") zum Missbrauch genutzt werden kann, muss für jedes Protokoll auf Anwendungsebene analysiert werden. Für HTTP(S) existieren bereits konkrete Szenarien.

SSH (secure shell) ist ein Anwendungsprotokoll, das aufgrund seines Protokolldesigns nicht betroffen ist. SSH führt Sessioninformationen mit, mit deren Hilfe manipulierte Neuverhandlungen erkannt werden. Die Verbindung wird in diesem Fall mit einem Fehler beendet.

Das speichern von Schlüsseln und Zertifikaten auf Chipkarten bietet keinen Schutz gegen die Lücke, da es sich um eine Schwachstelle im Protokoll handelt.

Man-in-the-Middle Angriffsszenario (MitM)

Ausgangspunkt für alle Angriffsszenarien ist eine Man-in-the-Middle-Attacke, bei der sich ein Angreifer in die Kommunikation zwischen Client und Server einschaltet.



Quelle: EKR, http://www.educatedguesswork.org/

Der Angreifer stellt zuerst eine Verbindung zum TLS Server her. Der Angreifer kann beliebige Anfragen an den Server stellen. Der Datenverkehr ist verschlüsselt (durch dicke Doppelpfeile dargestellt). Wenn der Angreifer bereit ist, entführt er eine unverschlüsselte Verbindung des Clients. Faktisch spielt es keine Rolle, welche Verbindung zuerst aufgebaut wird. Der Datenverkehr zwischen Client und Server wird vom Angreifer, der als Proxy agiert, weitergeleitet. Es existieren nun verschiedene Möglichkeiten eine Wiederverhandlung auszulösen. Lässt der Angreifer diesen erneuten Handshake vom Client ausführen, erkennt dieser gar nicht, dass er eine Wiederverhandlung durchführt und kann denn Angriff somit auch nicht erkennen. Der Server wiederum übernimmt nach dem Handshake Teile der ursprünglich vom Angreifer gesendeten Anfragen in die gesicherte Verbindung. Er erkennt nicht, das die Anfragen vor und nach dem Handshake aus zwei verschiedenen Quellen stammen. Die weitere Kommunikation zwischen Client und Server kann der Angreifer zwar weder sehen noch manipulieren, aber es ist ihm gelungen diese mit einem von ihm frei wählbaren Präfix zu versehen (plaintext injection). Handelt es sich beim Anwendungsprotokoll um HTTP(S) könnte der Angreifer den HTTP-Request des Clients durch einen eigenen ersetzen, dabei aber die HTTP-Header des Clients (inklusive seiner Cookies oder sonstige Authentisierungsinformationen) intakt lassen.

Die Attacke ist nicht trivial, insbesondere die Einzelheiten der Verbindungsbehandlung und erfordert eine Analyse der Kommunikation auf Anwendungsebene um tatsächlich Schaden anzurichten. Einen langfristig hinreichenden Schutz stellt diese Tatsache jedoch nicht dar.

Gefährdungen

Folgende Gefahren ergeben sich aus dem beschriebenen Problem.

Server

Server sind nach derzeitigem Stand der Dinge nicht direkt bedroht. Der Protokollfehler bietet keine konkrete Möglichkeit Server auf Betriebsystemebene zu kompromittieren.

Daten

Die Integrität und Authentizität von Daten, die über TLS-gesicherte Verbindungen gesendet werden, ist generell gefährdet.

Diese Tatsache impliziert eine Schwächung der Nichtabstreitbarkeit von Transaktionen. Im Streitfall können Benutzer in Bezug auf die TLS-Schwachstelle abstreiten, Daten erfasst oder Transaktionen ausgelöst zu haben.

Benutzer

Für die Benutzer besteht allgemein die Gefahr, dass für sie schädliche Transaktionen unter Ihrer Identität und ohne Ihre Kenntnis ausgelöst werden. Voraussetzung ist Nutzung des Dienstes durch den Benutzer und eine geeignete Schwäche auf Anwendungsebene, die vom Angreifer ausgenutzt werden kann.

Maßnahmen

Mittelfristig muss das Protokoll geeignet nachgebessert werden. Zur Zeit finden bereits Aktivitäten in Richtung Aktualisierung des Standards statt. Die Implementierungen sind dann entsprechend durch die Hersteller anzupassen.

Kurzfristig bietet das Abschalten der Wiederverhandlungsfunktion einen adäquaten Schutz vor den skizzierten Angriffen. Leider bieten die gängigen SSL-Implementierungen keine Konfigurationsoptionen zur Deaktivierung dieser Funktion. Daher ist zu hoffen, dass alle Distributoren zeitnahe aktualisierte Bibliotheken zur Verfügung stellen. Seit letzter Woche steht OpenSSL zur Verfügung. Damit steht allen Administratoren mit OpenSSL-basierten Systemen die Möglichkeit offen, auf diese Version zurückzugreifen.

An dieser Stelle darf nicht versäumt werden, auch Netzwerkkomponenten zu aktualisieren, die TLS/SSL zur Verfügung stellen, wie z.B. Router, die SSL-Verbindungen terminieren. Reagieren hier die Hersteller nicht adäquat zügig, bleibt nur die Rekonfiguration des Netzwerkes mit Terminierung der SSL-Verbindungen auf einem aktualisierten SSL-Server mit deaktivierter Wiederverhandlung.

Wie zügig die Aktualisierungen durchzuführen sind, hängt vom jeweiligem Umfeld ab. Zur Zeit existiert bereits ein Exploit, der demonstriert, wie die Sicherheitslücke ausgenutzt werden kann. Da gerade die Ausnutzung der Sicherheitslücke mit clientseitig ausgelöster Wiederverhandlung nicht trivial ist, kann als Sofortmaßnahme zunächst sichergestellt werden, dass alle serverseitigen Wiederverhandlungen unterbunden werden. Auf eine zügige Aktualisierung mindestens im Sinne der Abschaltung der Wiederverhandlung sollte jedoch in jedem Fall gedrängt werden!

Fazit

Es handelt sich um einen schwerwiegenden Sicherheitsfehler, der die meisten SSL-Implementierungen betrifft. Die Erforschung des Problems befindet sich erst am Anfang. Es kann davon ausgegangen werden, dass in kurzer Zeit verschiedene Exploits für unterschiedliche Protokolle oberhalb von SSL (SMTPS, IMAPS, LDAPS, etc.) erscheinen werden. Die zur Zeit bekannten Angriffsszenarien sind komplex, schwer generalisierbar und automatisierbar. Es kann jedoch nicht davon ausgegangen werden, dass dieser Zustand lange anhalten wird. Aus diesem Grund muss unbedingt gehandelt werden. Zeit dafür ist jedoch noch vorhanden. Es ist jedoch nicht möglich vorherzusagen, wie viel Zeit zur Verfügung steht. Daher muss sofort überlegt gehandelt werden.

Quellen

E. Rescorla: "Understanding the TLS Renegotiation Attack", http://www.educatedguesswork.org/2009/11/understanding_the_tls_renegoti.html, 5. November 2009

M. Ray, S. Dispensa: "Renegotiating TLS", http://extendedsubset.com/?p=8, 4. November 2009

M.Rex: "MITM attack on delayed TLS-client auth through renegotiation", http://www.ietf.org/mail-archive/web/tls/current/msg03928.html, 4. November 2009

B.Laurie: "Another Protocol Bites The Dust", http://www.links.org/?p=780, 5. November 2009

Cox, et al. (OpenSSL Projekt Team): "Announcement: OpenSSL version 0.9.8l", http://www.openssl.org/news/announce.html, November 2009

"CVE-2009-3555 TLS: MITM attacks via session renegotiation", https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2009-3555, 5. November 2009

D. Miller, "SSH is not vulnerable to the SSL/TLS MITM attack", http://djm.net.au/2009/11/6/ssh-is-not-vulnerable-to-the-ssl-tls-mitm-attack, 6. Novermber 2009

"Major vulnerability in SSL authentication", http://www.net-security.org/secworld.php?id=8477, 5. November 2009