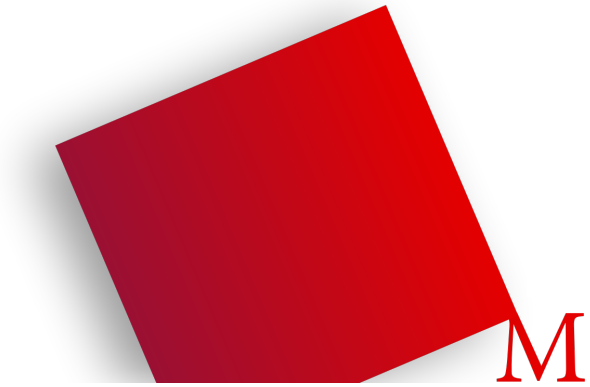
Fachbereich 07 Informatik/Mathematik



Praktikum Datenschutz und Datensicherheit
Sommersemester 2016

Prof. Dr. Rainer W. Gerling
Heidi Schuster

Man-in-the-Middle-Angriffe

Fabian Uhlmann

Diana Irmscher

24. Juli 2016

Zusammenfassung

Im Rahmen des Studiums Bachelor Informatik absolvieren wir (Fabian Uhlmann und Diana Irmscher) die zusätzliche Ausbildung zum betrieblichen Datenschutz an der Hochschule München.
Das Thema Datenschutz und IT-Sicherheit ist in den letzten Jahren immer mehr in den Vordergrund getreten. Medien berichten fast täglich über diverse Angriffe wie z. B. auf den Bundestag im Mai 2015 oder aktuell über den Krypto-Trojaner Locky.
Wir haben das Thema "Man-in-the-Middle" gewählt, weil es sich dabei um Angriffsszenarien handelt, die jeden (vernetzten) Nutzer jederzeit treffen können.
Das Thema ist in zwei Unterthemen aufgeteilt. Herr Uhlmann wird darauf eingehen, wie man die Sicherheit eines Systems mit einem MITM-Angriff sehr effizient aushebeln kann. Frau Irmscher beschäftigt sich mit dem gezielten Angriff in einem gesnifften Netzwerken.

München, 24. Juli 2016

Inhaltsverzeichnis

1 Man in the Middle im Web

1.1 Aufgabenstellung

Dell und Lenovo haben demonstriert, dass man mit Man-in-the-Middle Angriffen die Sicherheit eines Systems sehr effizient aushebeln kann. Wie funktioniert ein derartiger Angriff und was kann man tun, um sich zu schützen.

1.2 Bedeutung einer Man-in-the-Middle Attacke

Man in the Middle (MITM) ist ein Angriffsszenario, bei der ein unberechtigter Dritter versucht, in eine zwischen zwei Kommunikationspartnern geführte, sichere (verschlüsselte) oder auch unsichere (unverschlüsselte) Kommunikation einzudringen. Ziel des Angreifers ist es, die zu übertragenden, vertraulichen Informationen unbemerkt mitzulesen und/oder zu manipulieren. Um unerkannt im Hintergrund an die Daten beziehungsweise Informationen zu gelangen, täuscht der Angreifer vor, der jeweilige andere Kommunikationspartner zu sein. Ein MITM-Angriff kann auf den verschiedenen Ebenen des ISO/OSI Schichten-Modells [1, vgl.], z. B. auf Anwendungsebene (HTTP/HTTPs) oder Netzwerkebene (IP) stattfinden. Quelle: [2, vgl.]

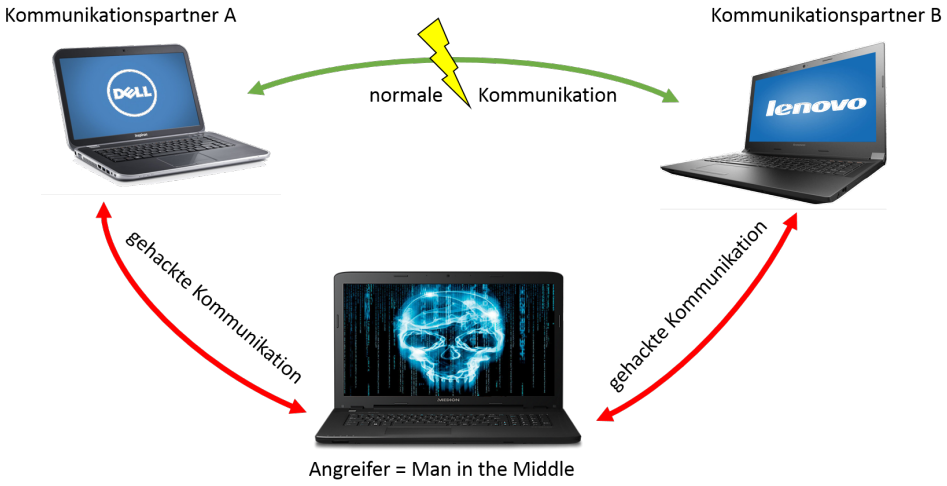


Abbildung 1: Man in the Middle Angriff [3]

1.3 Zusammenhang HTTP(s) und Zertifikate

Typischerweise fragt ein Nutzer über seinen PC (Client) eine Internetseite über das unsichere Hypertext Transfer Protocol (HTTP) an. Dabei werden alle über den Aufrufer benötigten Informationen, wie z. B. Benutzername und Passwort, im Klartext an den Webserver übermittelt. Auch die Rückantwort vom Server zum Client wird ebenfalls unverschlüsselt übertragen. Für einen sicheren Datenaustausch sollte ein Anwender, falls dies angeboten wird, die Internetseite per Hypertext Transfer Protocol Secure (HTTPs) aufrufen. [4, vgl.]

Seriöse Online-Banking-Webseiten unterstützen beispielsweise nur noch HTTPs-Aufrufe. Dabei wird dem Aufrufer durch das Kürzel HTTPs in der Adressleiste signalisiert, dass

es sich hierbei um eine sichere Verbindung handelt. Genauer gesagt bedeutet es, dass eine HTTP-Kommunikation über SSL (Secure Sockets Layer) / TLS (Transport Layer Security) verschlüsselt abläuft. Bei jeder HTTPs-Kommunikation muss sich der Webserver, auf dem die Internetseite gehostet wird, gegenüber des Webseitenaufrufers, meist ein Client, authentifizieren. Für den Client besteht hierbei kein Muss. Im Normalfall wird in der Praxis auf die Client-seitige Authentifizierung verzichtet. Die Authentifizierung des Webservers gegenüber des Clients erfolgt durch ein für den Webserver ausgestelltes Zertifikat[1]. [6, vgl.] Dieses enthält unter anderem den öffentlichen Schlüssel (Public Key), einen eindeutigen Fingerabdruck und Angaben über den Zertifikatsinhaber. [7, vgl.] Ein Zertifikat verbindet somit eindeutig einen Inhaber mit einem öffentlichen Schlüssel. Mit dem Public Key kann der Client verschiedene Daten, z. B. einen Pre-Shared Key, verschlüsselt zum Webserver schicken. Anhand des Fingerabdrucks, welcher auch als digitale Signatur des Zertifikats bezeichnet wird, überprüft der Client vor der Datenübermittlung, ob er mit dem richtigen Webserver kommuniziert. Der Fingerabdruck wird durch einen Hash-Algorithmus wie z. B. SHA-2 erzeugt. Bei der Erzeugung gehen diverse Informationen wie z. B. Zertifikatsaussteller, öffentlicher Schlüssel und Identifizierungsdaten über den Webserver mit ein. [8, vgl.] Wenn das Zertifikat von einer Zertifizierungsstelle (Certification Authority = CA) ausgestellt wurde, deren eigenes Zertifikat (= Root-Zertifikat oder Wurzelzertifikat) bereits im Browser installiert ist, dann wird dem ausgestellten Zertifikat automatisch vertraut. In den bekanntesten Webbrowsern wie z. B. Firefox, Chrome oder dem Internet Explorer sind bereits viele Root-Zertifikate von weltweit verschiedenen Zertifizierungsstellen vorinstalliert.

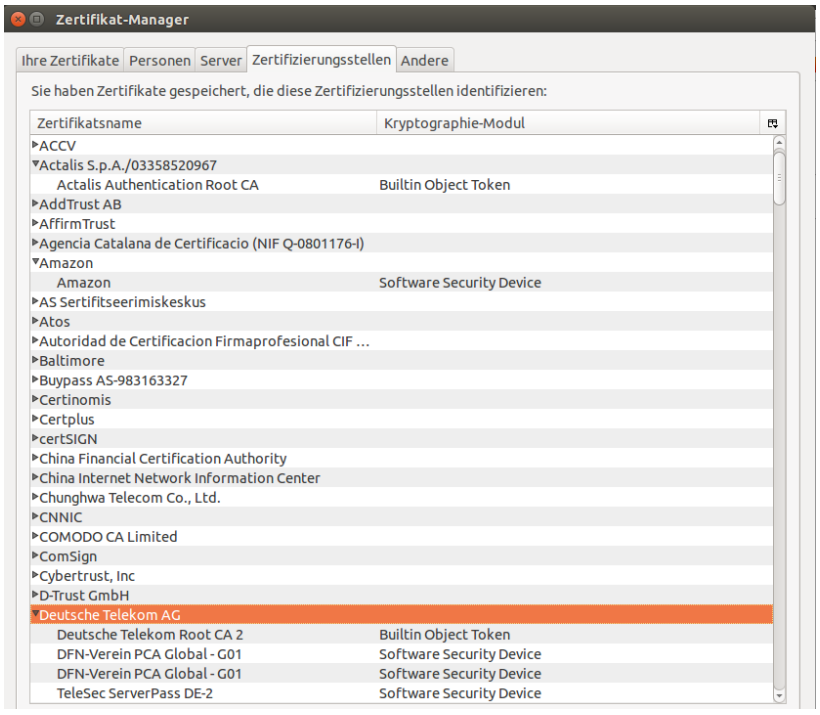


Abbildung 2: Ausschnitt vorinstallierter Zertifizierungsstellen und deren ausgestellten Zertifikate im Firefox Browser

[1]Bei SSL Verbindungen werden in der Regel digitale Zertifikate nach dem ITU-T X.509 Standard verwendet [5, vgl.]

Verwendet der Webserver ein selbst ausgestelltes (selbst signiertes) Zertifikat zur Authentifizierung, dann wird beim Verbindungsaufbau dem Nutzer eine Warnung angezeigt. Er kann anschließend selbst entscheiden ob er dem Zertifikat vertraut oder nicht. Außerdem entscheidet er über die Dauerhaftigkeit des Vertrauens. Dabei muss er sich zwischen der Möglichkeit *nur einmalig*, d. h. nur für diese gerade offene Verbindungssession, oder *für immer* entscheiden.

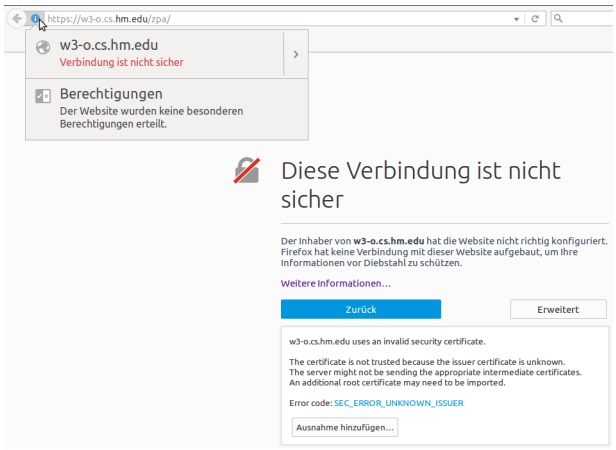


Abbildung 3: Warnung bei unbekanntem / nicht vertrauenswürdigem Zertifikat

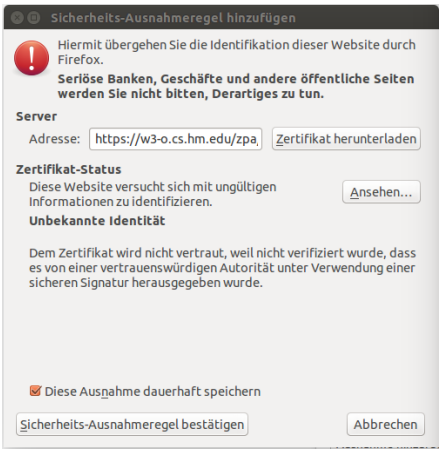


Abbildung 4: Ausnahmeregel für ein unbekanntes / bislang nicht vertrauenswürdiges Zertifikat

Ist letzteres der Fall, dann wird das Zertifikat fest im Browser installiert. Es wird dann bei den schon vorinstallierten Zertifikaten mit abgelegt. Diese dauerhafte Ausnahme hat zu Folge, dass der Benutzer beim Verbindungsaufbau zu der zugehörigen Webseite vom Webbrowser nicht mehr gewarnt wird.

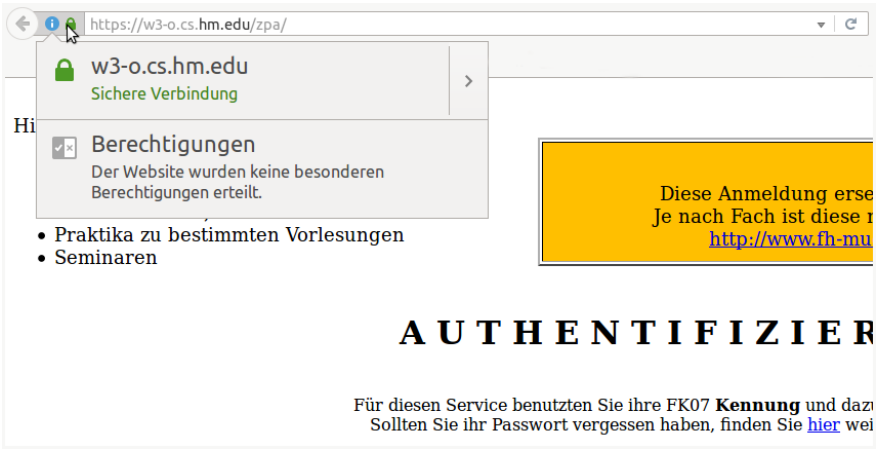


Abbildung 5: Keine Warnung nach dauerhafter Ausnahme / fester Installation des Zertifikats im Browser

1.4 Sicherheitslücken bei bekannten Computer Herstellern

Man sollte davon ausgehen können, dass die großen PC-Hersteller selbst am besten wissen müssten, wie hart der Markt in der Computerbranche umkämpft ist. Neben einer hohen Hardware-Qualität ist Kundenvertrauen immens wichtig. Zwei der weltweit bekanntesten und erfolgreichsten Computerhersteller [9, vgl.] haben den Faktor *Vertrauen* nicht hinreichend erfüllt. Denn sowohl Lenovo als auch DELL haben das Vertrauen ihrer Kunden stark missbraucht.

1.4.1 Lenovo mit potentiellem Risiko

Durch eine bereits vorinstallierte Software der Firma Superfish hat Lenovo versehentlich eine gravierende Sicherheitslücke auf einigen ihrer vertriebenen Notebookmodelle eingebaut. Dadurch wurde der Kunde einer zusätzlichen Gefahr eines erleichterten Hackerangriffes ausgesetzt.

Die Internetseite `www.golem.de` beschreibt die Idee und den Ablauf von den benutzerspezifischen Werbeeinblendungungen auf den Lenovo-Rechnern wie folgt:

> „Grundsätzlich ist die Idee von Superfish, dass das Programm Bilder auf Webseiten durchsucht und anhand von Algorithmen versucht zu erkennen, was sich darauf befindet. Auf Basis dessen werden dem Nutzer passende Shopping-Angebote als Werbebanner angezeigt. Geradezu zynisch wirkt die Beschreibung des Lenovo-Angestellten im Forum: Die Funktion diene dazu, Nutzern zu helfen, visuell Angebote für Produkte zu finden, bei denen sie Schwierigkeiten haben, sie mittels einer textbasierten Suchmaschine zu finden.“[10]

Für die Umsetzung der Idee reichte es nicht, dass nur ein Programm von Superfish auf den Lenovo-Rechnern installiert wird. Zusätzlich benötigte Superfish unbedingt ein selbst signiertes Root-/Wurzel-Zertifikat auf dem System. Ohne dieses Zertifikat wäre es nicht möglich gewesen, auch in verschlüsselten Internetverbindungen (HTTPs-Verbindungen) den Suchalgorithmus anzuwenden, um anschließend personalisierte Werbung einzublenden. Mit dem eigenem Root-Zertifikat war Superfish in der Lage, für jede Verbindung, die zu einer HTTPs-Webseite aufgebaut wurde, ein eigenes, gefälschtes Zertifikat dynamisch zur Laufzeit zu erzeugen. Dieses wurde automatisch anerkannt, da bereits dem zugehörigen Wurzelzertifikat vertraut wurde. Weil Superfish sein Root-Zertifikat direkt im Windows-Zertifikatsspeicher installierte, wurde dieses nicht extra überprüft. Denn allen Zertifikaten, die sich im Windows-Zertifikatsspeicher befinden, wird automatisch das Vertrauen geschenkt. Der Anwender bekam dadurch nicht mit, dass keine direkte verschlüsselte Kommunikation zu dem eigentlichen Server, der die Webseite hostet, aufgebaut wurde.

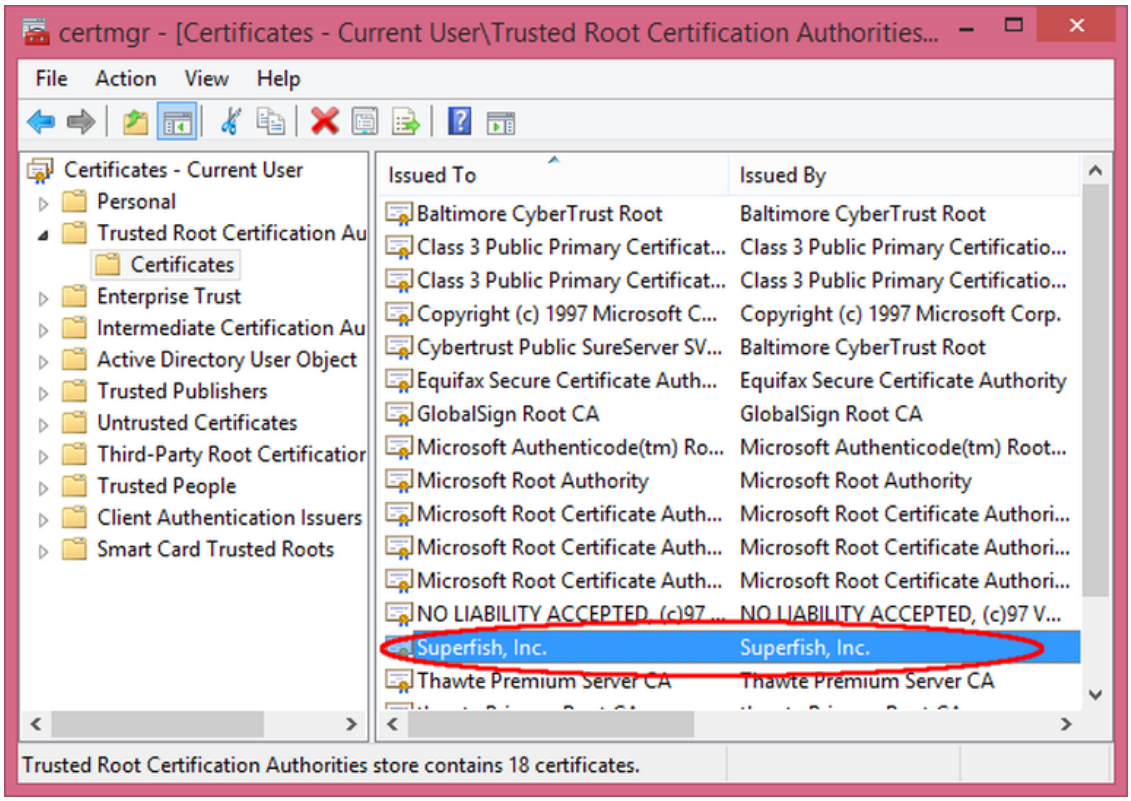


Abbildung 6: Superfish Root-Zertifikat im Windowszertifikatsspeicher, Quelle: [11]

Weiterhin kam erschwerend hinzu, dass das Programm von Superfish für die Man-in-the-Middle Attacken ein schwaches Zertifikat nutzte. Das Root-Zertifikat verwendet für die digitale Signatur einen SHA-1 Hash-Alsgorithmus und für die asymmetrische Verschlüsselung ein 1024 Bit RSA-Verschlüsselungsverfahren. [12, vgl.] Sowohl der SHA-1 Hash-Algorithmus als auch das RSA-Verschlüsselungsverfahren mit einer Schlüssellänge von 1024 Bit sind bereits erfolgreich geknackt worden und daher als unsicher einzustufen. [13, 14, vgl.] Doch noch fataler als der Einsatz des unsicheren Hash-Algorithmus und des zu schwachen Verschlüsselungsfahrens, ist die Leichtigkeit der Entschlüsselung des privaten, geheimen Schlüssels. Robert Graham beschreibt in seinem Blog auf Errata Security eindrucksvoll, wie mit simplen Mitteln der Private Key exportiert und anschließend mit einer einfachen Wörterbuch-Attacke entschlüsselt werden konnte. [15, vgl.] Mittels des privaten Schlüssels kann jeder Angreifer, genau wie das Superfish-Programm, eigene, durch das Root-Zertifikat signierte, Zertifikate erzeugen und diese für bösartige Verwendungen nutzen. Somit bekommen die Anwender nicht mit, dass sie z. B. auf einer gefälschten Webseite surfen und ihre Daten ausspioniert oder manipuliert werden. Neben Zertifikaten für Webseiten, ist ein Hacker auch in der Lage, Zertifikate für kriminelle Software (z. B. Malware) zu erstellen und diese dem Nutzer als gutartig erscheinen zu lassen.

1.4.2 DELL mit ähnlichen Fehlern

Der Computerhersteller DELL leistete sich einen ähnlich gravierenden Sicherheitsfehler wie sein Konkurrent Lenovo zuvor. Genau wie Lenovo hat DELL auch selbst signierte Root-Zertifikate auf einigen seiner Laptops installiert. Bei dem US-amerikanischen Hersteller sind

es sogar zwei Root-Zertifikate. Sowohl das eDellRoot, als auch das DSDTestProvider Zertifikat wurden, genau wie das Superfish-Zertifikat, im Windows-Zertifikatsspeicher abgelegt. Beim Aufruf der allgemeinen Eigenschaften des eDellRoot-Zertifikats wird sogar ein Hinweis angezeigt, dass ein passender Private Key vorhanden ist. Joe Nord wendet in seinem Online-Blog die gleichen Vorgehensweisen zum Export und zur Entschlüsselung des DELL Private Keys an, wie Robert Graham beim Superfish Private Key. [16, vgl.]

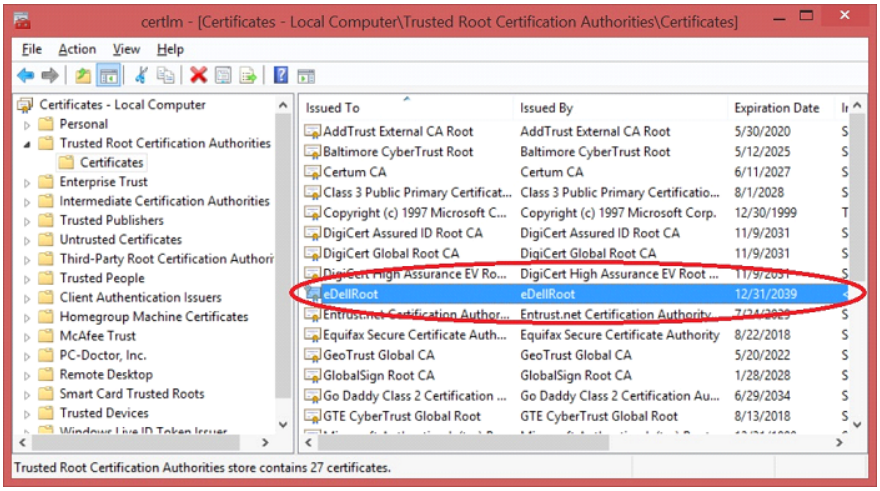


Abbildung 7: Ausschnitt aus Windowszertifikatsspeicher mit installiertem eDELLRoot Zertifikat, Quelle: [17]

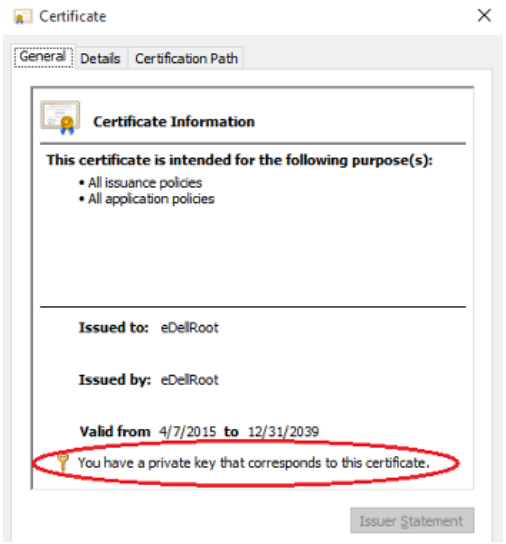


Abbildung 8: Eigenschaftsansicht von eDELLRoot Zertifikat mit Hinweis auf Private Key, Quelle: [16]

Angreifer mit dem privaten DELL Schlüssel haben die gleichen Angriffsmöglichkeiten auf infizierte DELL Geräte, wie Angreifer mit dem Superfish Private Key auf infizierte Lenovo Geräte. Beide DELL Zertifikate wurden durch Software installiert, das eDellRoot Zertifikat

mit dem Dell Foundation Services Programm und das DSDTestProvider Zertifikat mit der Dell System Detect Software. Sie sollen, laut DELL, für einfacheren Support dienen. DELL hat auf seiner Webseite folgende Stellungnahme publiziert:

> „[...] Das Zertifikat eDellRoot wurde zusammen mit unserer Anwendung Dell Foundation Services installiert und wird zur Unterstützung einer besseren, schnelleren und einfacheren Support-Erfahrung für unsere Kunden verwendet. Das Zertifikat ist keine Schadsoftware oder Adware. Es war ursprünglich dazu gedacht, die Service-Tag-Nummer des Systems an den Onlinesupport von Dell zu übermitteln, damit wir schnell das Computermodell identifizieren und unseren Kunden so einen unkomplizierteren und schnelleren Service bieten können. Das Zertifikat wird nicht zum Sammeln persönlicher Kundendaten verwendet. Bitte beachten Sie, dass sich das Zertifikat nicht von selbst neu installiert, nachdem es mit dem empfohlenen Dell Prozess ordnungsgemäß entfernt wurde.
> Wenn wir eDellRoot kennen, können wir uns auf alle unsere Anwendungen konzentrieren, die auf Dell PCs geladen werden. Wir können bestätigen, dass keine weiteren Root-Zertifikate auf dem werkseitig installierten Image installiert wurden. Wir haben jedoch herausgefunden, dass die Anwendung Dell System Detect und das dazugehörige Root-Zertifikat DSDTestProvider ähnliche Eigenschaften hat wie eDellRoot. Im Fall von Dell System Detect lädt der Kunde die Software proaktiv herunter, um mit der Webseite vom Dell Support zu interagieren. Dadurch können wir eine bessere und individuell abgestimmte Support-Erfahrung anbieten. Wie eDellRoot auch wurde das fragliche Support-Zertifikat entwickelt, um unseren Kunden einen schnelleren und einfacheren Support zu bieten. Die Vorteile sind jedoch auf die Kunden beschränkt, die die Funktion zur Produkterkennung auf unserer Support-Webseite zwischen dem 20. Oktober und dem 24. November 2015 genutzt haben. Die Anwendung wurde von der Dell Support-Webseite sofort entfernt und es steht ab sofort eine neue Anwendung ohne das Zertifikat zur Verfügung. Wir unterstützen proaktiv ein Software-Update, um das Problem anzugehen und haben im Folgenden Anweisungen zum Entfernen des Zertifikats bereitgestellt.“[17]

Mit dieser Aussage hat DELL offiziell bestätigt, eigene Zertifikate auf Nutzer-Endsystemen installiert zu haben. Das eDellRoot wurde augenscheinlich bereits vor Auslieferung auf den Geräten installiert und das DSDTestProvider erst nachträglich durch Installation und Verwendung des DELL Support-Programms. Durch das DELL-Statement fällt auf, dass das eDellRoot Zertifikat sich eventuell erneut installieren kann, wenn es nicht mit spezieller DELL Software bzw. geeignetem DELL Prozess richtig deinstalliert wurde.

> „[...] Bitte beachten Sie, dass sich das Zertifikat nicht von selbst neu installiert, nachdem es mit dem empfohlenen Dell Prozess ordnungsgemäß entfernt wurde.“[17]

Die Ursachen für eine eventuelle Neuinstallation des DELL Zertifikats sind im nachfolgenden Kapitel 1.5 Schutzmöglichkeiten beschrieben.

1.5 Schutzmöglichkeiten

Bedauerlicherweise haben Betroffene nicht allzu viele Möglichkeiten, sich gegen eben beschriebene Sicherheitslücken zu schützen.
Eine Option wäre der Kauf von neuen Systemen ohne vorinstalliertem Betriebssystem. Wenn der Anwender selbst ein Betriebssystem nach dem Kauf installiert, kann er mit großer Sicherheit davon ausgehen, dass sich keine ungewollte, vorinstallierte Software und/oder kein selbst signiertes Zertifikat auf dem neu angeschafften Gerät befindet. Jedoch schützt diese Option nicht vor solchen Zertifikaten, die sich erst nachträglich installieren. Ein Beispiel ist das DSDTestProvider Zertifikat, das durch das DELL Support-Programm Dell System Detect auf das Endsystem kommt. [17, vgl.]
Eine andere Variante des Schutzes ist die regelmäßige Kontrolle der auf dem System installierten Zertifikate. Durch diese Maßnahme können auch nachträglich installierte Zertifikate entdeckt werden. Zugegeben, bei der heutigen Masse an vorhandenen, vertrauenswürdigen Zertifizierungsstellen sowie zugehörigen Zertifikaten ist es nicht leicht, den Überblick zu behalten. Erschwerend kommt hinzu, dass viele Programme, z. B. Webbrowser wie Chrome oder Firefox, eigene Zertifikatsspeicher besitzen. Dabei kann nicht immer davon ausgegangen werden, dass diese Speicher sich auf den Zertifikatsspeicher des Betriebssystems beziehen und die identischen Zertifikate beinhalten.
Um sicherzustellen, dass mit dem richtigen Server kommuniziert wird und folglich keine Verbindung zu einer gefälschten Webseite besteht, sollte man das Zertifikat prüfen, welches für die aktuelle Kommunikation verwendet wird. Die meisten Browser zeigen neben der HTTPs-Adresse ein Schlosssymbol. Über dieses Symbol werden die Eigenschaften des aktuell für diese verschlüsselte Verbindung verwendeten Zertifikats erreicht.

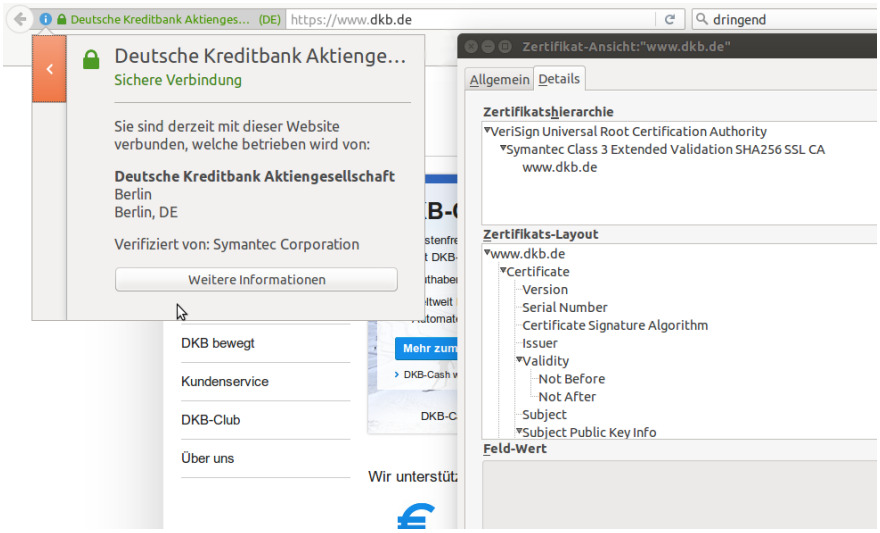


Abbildung 9: Aufruf der DKB Webseite. Details des Zertifikats, welches für die Verbindung verwendet wird

Verschiedene Webseiten, wie z. B. `https://globalsign.ssllabs.com`, stellen einen Dienst zur Verfügung, mit dem es möglich ist gewünschte Webseiten zu überprüfen. Als Ergebnis zeigen sie unter anderem an, welches originale Zertifikat diese Webseite wirklich verwendet. Mittels dieser Information und der Zertifikats-Eigenschaften durch den Browser ist jeder

Internetnutzer in der Lage zu vergleichen, ob die verschlüsselte Verbindung das richtige Zertifikat nutzt. Handelt es sich nicht um das korrekte, originale Zertifikat und gab es zusätzlich keine Fehlermeldung/Warnung, so muss davon ausgegangen werden, dass keine direkte Verbindung zu der gewünschten Webseite bzw. Webserver besteht. Solche Fälle deuten stark darauf hin, dass die verschlüsselte Verbindung durch eine MITM-Attacke aufgebrochen wurde und die Daten eventuell mitgelesen oder sogar manipuliert werden.
Nach Aufdeckung einer solchen Attacke muss das gefälschte Zertifikat im System (Zertifikatsspeichern) gesucht und anschließend gelöscht werden. Dabei ist zu beachten, dass es nicht immer ausreicht nur das entdeckte Zertifikat selbst zu löschen. Da Zertifikate auf der Basis einer Public Key Infrastruktur [7, vgl.] erstellt sind, können noch weitere Zertifikate für solch eine Attacke verantwortlich sein. Es sollten, neben dem aufgespürten, gefälschten Zertifikat, zusätzlich alle anderen Zertifikate, die sich in der hierarchischen Kette befinden, überprüft und ggf. entfernt werden. Im Lenovo-Beispiel konnten mittels des gehackten, selbst signierten Root-Zertifikats weitere Zertifikate erstellt werden. Mit jedem dieser Zertifikate war anschließend je eine separate MITM-Attacke möglich. Das Sicherheitsproblem war mit der Eliminierung des für die MITM Attacke verwendeten Zertifikats noch nicht gelöst. Erst nach erfolgreichem Entfernen des selbst signierten Root-Zertifikats konnten keine weiteren gefälschten Zertifikate erzeugt werden.
Beim DELL-Vorfall schrieb Liam Tung auf der ZDNET-Webseite:

> „[...] das einfache Entfernen des eDELLRoot-Zertifikats aus dem Administrator und persönlichen Zertifikatsspeicher ist nicht genug, um den Nutzer zu schützen. Einige Nutzer haben in der Tat berichtet, dass das Zertifikat nach einem Neustart wieder aufgetaucht ist.“[18]

Ursache für die Neuinstallation ist das DELL-Programm „Dell Foundation Services“, welches dieses Zertifikat verwendet. Erst mit der Deinstallation des Programms bzw. eines Plug-ins des Programms und der manuellen Löschung des DELL Zertifikats ist das selbst signierte Zertifikat dauerhaft vom System erfolgreich entfernt und die Sicherheitslücke geschlossen worden. Liam Tung schrieb zur korrekten Entfernung:

> „Um es dauerhaft zu entfernen und um zu verhindern, dass es sich erneut installiert, müssen Nutzer das eDELL Plugin entfernen.“[18]

Für die detaillierte Information um welches Plug-in es sich genau handelt und worauf besonders zu achten ist, nutzt er die Ergebnisse von den Duo Security Forschern Darren Kemp, Michail Davidov und Kyle Lady.

> „‘Dies kann vollbracht werden mit Hilfe der Löschung des Dell.Foundation.Agent.Plugins.eDell.dll Moduls vom System. Geschieht dies nicht, so kann es weiterhin zur Aussetzung dieser Sicherheitslücke kommen.‘, sagte Duo Security.
> ‘Beachte, immer wenn sie ein Werksreset auf ihrem DELL System durchführen, wird dieses Zertifikat und das eDell Plugin auf dem System wieder hergestellt und sie müssen es erneut manuell entfernen‘ [...]“[18]

Neben all für einen Angriff verantwortlichen Zertifikaten dürfen installierte und verwendete Programme nicht außer acht gelassen werden. Die beiden Beispiele haben gezeigt, dass

es nicht ausreicht nur die Zertifikatsspeicher auf ungewöhnliche Eintragungen zu durchsuchen, sondern ebenfalls die Programmliste des Systems auf unerwartete oder ggf. unnötig installierte Programme zu kontrollieren.

1.6 Zwischenfazit

Beide Systemhersteller nutzten ihre Position in der Marktwirtschaft und das Vertrauen, welches ihnen die Kunden entgegenbringen, schamlos aus. Sie ließen dem Käufer im Glauben, dass er ein solides Gerät mit sicherer Software gekauft hätte. Doch ohne selbständige Gegenmaßnahmen des Kunden war dieser, mittels der von Lenovo und DELL selbst signierten Zertifikate, potentiellen Angreifern hilflos ausgesetzt. Die Käufer müssen auf der einen Seite für die Zukunft hoffen, dass die Gerätehersteller aus ihren Fehlern gelernt haben und wieder sichere Systeme herstellen bzw. verkaufen. Auf der anderen Seite kann und sollte jeder selbständig Kontrollen und Überprüfungen durchführen. Außerdem ist es ratsam, sich ein wenig mit den Systemen zu befassen mit denen man täglichen Umgang hat und wichtige Tätigkeiten, wie z. B. Online Banking, durchführt. Darunter zählt auch das regelmäßige Abrufen von Informationen über Sicherheitsneuigkeiten in bekannten Sicherheitsforen, z. B. `https://www.heise.de/security`

Gemäß dem Sprichwort „Vertrauen ist gut, Kontrolle ist besser.“, sollte jeder Benutzer, mit den verfügbaren Möglichkeiten, die verwendeten Systeme und Dienste immer wieder überprüfen.

2 Man-in-the-Middle-Angriffe im geswitchten Netz

Aufgabenstellung

Es gibt Man-in-the-Middle-Angriffe nicht nur gegen SSL/TLS Verbindungen sondern auch gegen "normale"Netzwerkverbindungen. Sie werden von Angreifern eingesetzt, um in einem geswitchten Netz zu sniffen.

2.1 Switch vs. Hub

Zu Beginn sei erst einmal beschrieben, wie ein Hub und ein Switch funktionieren, denn dies ist relevant für die Betrachtung der möglichen Angriffsszenarien in einem geswitchten Netzwerk.

2.1.1 Hub

Ein Hub stellt eine Verbindungskomponente in einem Netzwerk dar. Er ist ein Knotenpunkt zwischen einzelnen Netzwerkteilnehmern, z.B. Rechnern und weiteren Subnetzen. Er arbeitet ausschließlich auf der Schicht 1 des ISO-/OSI-Schichtenmodell. Im Gegensatz zum Switch arbeitet ein Hub nicht zielorientiert (siehe unten), sondern sendet einfach alle Bits (Einheit in Schicht 1) an alle Teilnehmer, welche an den Hub angeschlossen sind, vergleichbar mit einem Broadcast. Aufgrund dieser Eigenschaft ist es möglich, den Datenverkehr mitzulesen und zu analysieren, wenn man an diesem Hub angeschlossen ist.

Ein Hub wird über eine Stern-Topologie realisiert. Logisch betrachtet funktioniert ein Hub wie bei einer Bustopologie, da jeder der Teilnehmer den Datenfluss lesen kann. Die Teilnehmer befinden sich deshalb auch in einer gemeinsamen Kollisionsdomäne. Auf Kollisionen wird hier allerdings nicht weiter eingegangen, es geht hier um die Sichtbarkeit der Daten unter den Netzwerkteilnehmern.

Ein Hub wurde hauptsächlich noch aus Kostengründen in Netzwerken eingesetzt, seit Switches auf den Markt gekommen sind. Allerdings wurde der Hub vom Switch mittlerweile fast vollständig verdrängt, da Switches mit der Zeit günstiger geworden sind.[19, vgl.]:

2.1.2 Switch

Ein Switch arbeitet in der 2. Schicht des ISO-/OSI-Referenzmodells, genannt Sicherungsschicht. Aufgaben dieser Schicht sind [20, vgl.]:

- Rahmenorientierte Datenübertragung auf einer Teilverbindung
- Übertragungseinheit: Frame

Bei Switches kann man zwischen zwei Typen unterscheiden, dem einfachen Switch und solchen, die zusätzlich noch die Daten auf der 3. Schicht, der Netzwerkschicht, betrachten.

Ein einfacher Switch leitet Pakete mit Hilfe der MAC-Adressen von Quelle und Ziel weiter. Dafür verwendet ein Switch die Switch-Tabelle, in welcher die Zieladresse mit dem entsprechenden Port vermerkt ist.

```
▼ Ethernet II, Src: LcfcHefe_41:be:1e (28:[illegible]), Dst: IntelCor_1d:94:81 (68:[illegible])
  ▼ Destination: IntelCor_1d:94:81 (68:[illegible])
      Address: IntelCor_1d:94:81 (68:[illegible])
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  ▼ Source: LcfcHefe_41:be:1e (28:[illegible])
      Address: LcfcHefe_41:be:1e (28:[illegible])
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IP (0x0800)
```

Abbildung 10: Aufbau des Headers in Schicht 2, Auszug aus WireShrak (MAC-Adressen entfernt)

Die Frames werden hier dem Ziel, einem Netzwerkteilnehmer, zugestellt, welcher im Header des Frames mit der MAC-Adresse vermerkt ist. Das ist der wesentliche Unterschied zum Hub, welcher die empfangenen Daten an alle Teilnehmer gesendet hat.

Ein Layer-3-Switch oder Multilayerswitch verfügt über mehr Funktionen als ein einfacher Switch. So können zum Beispiel Steuer- und Überwachungsfunktonen genutzt werden, wie IP-Filterung oder Routing. Im Prinzip verletzen diese Funktionalität das ISO-/OSI-Schichtenmodell, allerdings wird die Weiterleitung von Pakete auf der Hardware realisiert, was sehr viel schneller geht als über einen Router.

Der Einsatz eines Switches bringt im Verglich zu den Vorgängern Hub und Bridge einige Vorteile mit sich, hier sei im Besonderen darauf hingewiesen, dass der Switch die Datensicherheit erhöht, denn die Daten werden nur den Netzwerkteilnehmer gesendet, an welchen die Informationen adressiert sind (wenn die MAC-Adresse bekannt ist).

2.1.3 Funktion der Switch-Tabelle

Die Switch-Tabelle ermöglicht den wesentlichen Punkt in der Funktionsweise eines Switch: dass die Daten nur noch zu einem bestimmten Empfänger gelangen. [20, Kapitel 5]

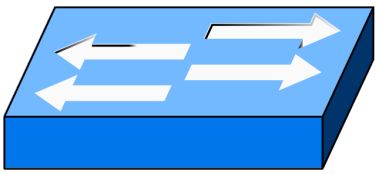


Abbildung 11: Switch [21]

Ein Switch verfügt über Ein- und Ausgänge, die sogenannten Ports. Diese können unabhängig von einander senden und empfangen. Damit keine Frames verloren gehen, verfügt der Switch über einen Datenpuffer. An den Ein- und Ausgängen sind die einzelnen Netzwerkteilnehmer angeschlossen. In der Switch-Tabelle ist für jeden Port die MAC-Adresse des angeschlossenen Teilnehmers hinterlegt.

Adresse	Interface	Zeit
62-FE-F7-11-89-A3	1	9:01
7C-BA-B2-B4-91-10	3	9:12
...	...	...

Abbildung 12: Switch-Tabelle mit eingetragenen MAC-Adressen und den Ports, hier Interfaces genannt [20, Kapitel 5, Abbildung Folie 39]

Somit kann entschieden werden, an welchen Port der eingegangene Frame gesendet wird. Es gibt unterschiedliche Arbeitsweisen, nach welchen ein Switch funktioniert. Store-and-Forward wird von jedem Switch umgsetzt, deshalb wird hier auch nur diese betrachtet.

Store-and-Forward ist von den Switch-Methoden die sicherste, allerdings auch die langsamste. Wird vom Switch ein Frame empfangen, werden die folgenden Schritt bearbeitet.

1. Schritt
   - Switch empfängt gesamtes Frame (store)
   - berechnet Prüfsumme (CRC)
   - wenn Prüfsumme nicht mit CRC-Wert im Frame übereinstimmt, wird das Frame verworfen (keine fehlerhaften Daten im lokalen Netzwerk)
2. Schritt
   - überprüfen, ob Quell-Adresse in Switch-Tabelle vorhanden ist
   - wenn nicht, Quell-MAC-Adresse zusammen mit Port in der Tabelle ergänzen
   - wenn schon vorhanden, lediglich Aging-Timer aktualisieren
3. Schritt
   - Zieladresse des Frames mit Eintrag in der Switch-Tabelle vergleichen
   - ist Zieladresse vorhanden, wird das Frame an den Empfänger weitergeleitet
   - ist die Zieladresse (noch) nicht vorhanden, wird das Frame an alle Ports weitergeleitet

In einem IPv4-Netzwerk wird der Eintrag in der Switch-Tabelle meist schon von dem ARP-Request (Address Resolution Protocol) mit entsprechendem Port hinterlegt. Dabei wird zunächst aus der ARP-Anfrage der Empfänger herausgelesen und zugeordnet. Ist dieser schon in der Tabelle hinterlegt, brauch da nichts weiter vorgenommen werden. Aus der ARP-Response erhält man folglich den Empfänger, dieser wird in die Switch-Tabelle eingetragen. Das ARP-Protokoll wird genutzt, um Netzwerkteilnehmer untereinander bekannt zu machen, z.B. Rechner mit Router. Die Switch-Tabelle füllt sich demnach automatisch und braucht nicht zu konfiguriert zu werden.

2.1.4 Was passiert, wenn die Switch-Tabelle überläuft

Ist die Tabelle gefüllt mit Einträgen und kann keinen neuen Eintrag mehr aufnehmen (die Tabelle könnte zu klein sein), werden alle Frames, die nicht zugeordnet werden können, an alle Ports gesendet werden. Das würde die Netzwerklast drastisch heben.

2.2 Was ist ein geswitchtes Netz?

Ein geswitchtes Netzwerk zeichnet aus, dass die Auslastung des Netzwerke durch den Datenfluss der Netzwerkteilnehmer stark reduziert wird, da ein Frame nur an die Ziel-Adresse gesendet wird, wenn sie bereits in der Switch-Tabelle hinterlegt ist.

Somit bringt der Einsatz eines Switches auch mehr Sicherheit, wenn es darum geht, wer im Netzwerk hängt und wer welche Daten erhalten soll.

Ein Switch kann Netzwerkteilnehmer auch in Gruppen aufspalten und hier unterscheiden, an welche Gruppe das Frame zugestellt werden soll, das Verhalten ist wie bei zwei getrennten LAN. Die Netzwerkteilnehmer sehen dann nur die Frames in ihrer Gruppe.

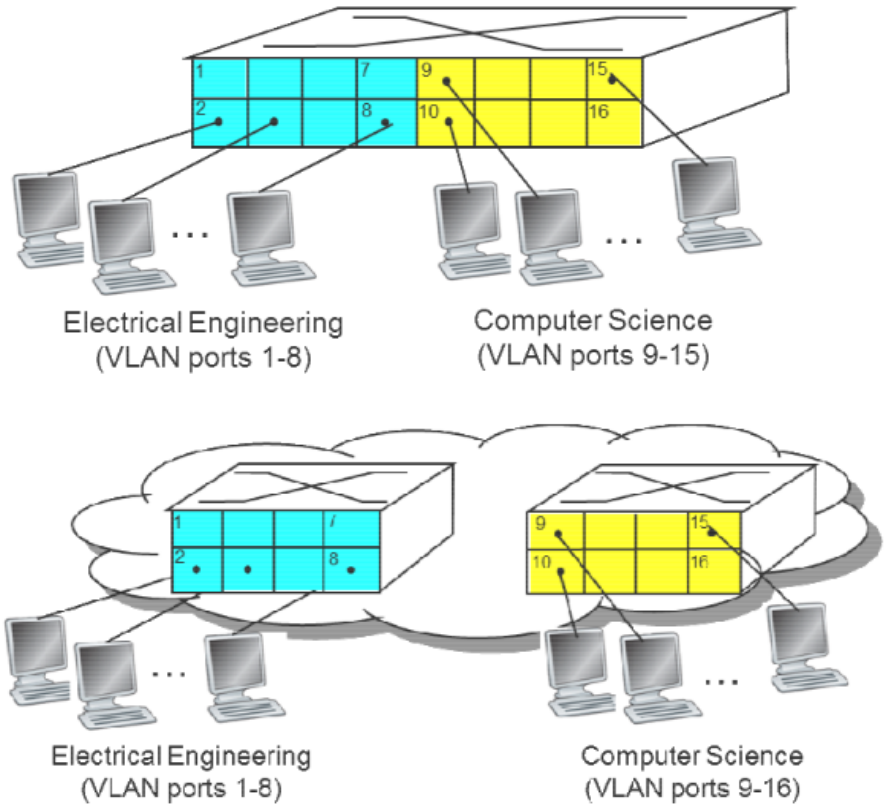


Abbildung 13: Zusammenfassung der Ports eines Switches zu einzelnen Gruppen [20, Kapitel 5, Abbildung Folie 41]

Jedes Netzwerk verfügt heutzutage über mindestens einen Switch, da mehrere Netzwerkteilnehmer so besser verwaltet werden können. Früher hat diese Aufgabe ein Hub oder eine Bridge übernommen, bis diese vom Switch abgelöst wurden.

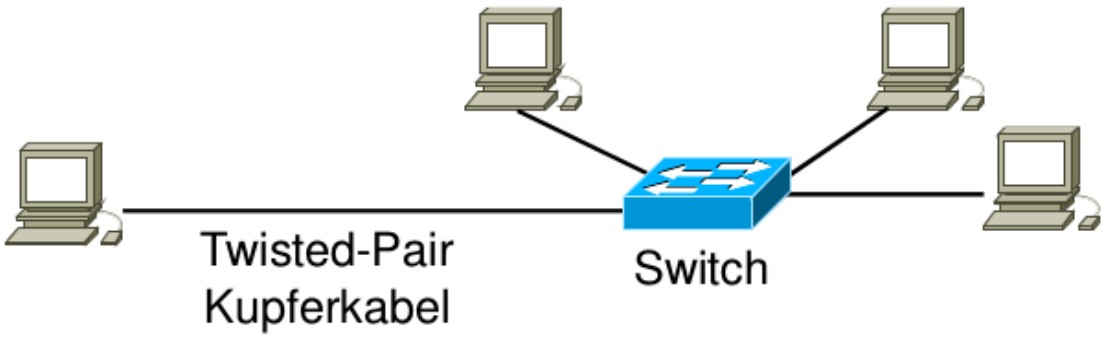


Abbildung 14: Geswitchtes Netzwerk [20, Kapitel 5, Abbildung Folie 33]

2.2.1 Mehrere Switches in einem Netzwerk

Mehrere Switches in einem Netzwerk sind natürlich möglich, allerdings kann man diese nicht einfach so miteinander verbinden, sonst entsteht eine Schleife. Ist eine Schleife in einem Netzwerk gelegt, bricht der gesamte Datenverkehr zusammen.

Für die Umsetzung mehrerer Switches in einem Netz müssen ein paar Vorkehrungen getroffen werden, unter anderem spezielle Kabel, um eben diese Schleife zu verhindern.

2.3 Angriffsmöglichkeiten in einem geswitchten Netzwerk

2.3.1 MAC-Flooding

Der Speicher der Switch-Tabelle ist begrenzt. MAC-Flooding macht sich das zunutze. Der Switch wird mit gefälschten MAC-Adresse überhäuft, bis der Speicher der Tabelle voll ist. Dann verhält sich der Switch wie ein Hub, was auch Ziel des Angriffes ist.

Einige Switches haben Schutzmaßnahmen gegen diesen Angriff, diese sind allerdings eher in den höheren Preiskategorien zu finden. Zum Beispiel kann für einen Port eine Liste mit zugelassenen MAC-Adressen angelegt werden. Frames von nicht bekannten MAC-Adressen werden verworfen, bzw. der betroffene Port wird gesperrt.

2.3.2 MAC-Spoofing

Eine weitere Möglichkeit, als Dritter den Datenverkehr mitzulesen, nennt sich MAC-Spoofing. Hierbei wird die Quell-Adresse mit der MAC-Adresse des Angreifers ersetzt. Die Respons des Empfängers des Frames wird an den Angreifer gesendet. Somit können die Daten vom Angreifer mitgelesen werden, ohne dass der eigentliche Sender und auch der Empfänger etwas davon mitbekommen. Der Angreifer wird allerdings in die Switch-Tabelle eingetragen. Diesen Umstand kann man nutzen und den Angriff umgehen, indem man wie oben schon erwähnt eine Liste mit erlaubten MAC-Adresse für den jeweiligen Post erstellt.

2.3.3 ARP-Spoofing

Das ist ein interessanter Angriff, denn er nutzt eine Schwäche des ARP-Protokolls aus. Dazu wird kurz erklärt, wofür das ARP-Protokoll da ist und wie es funktioniert.

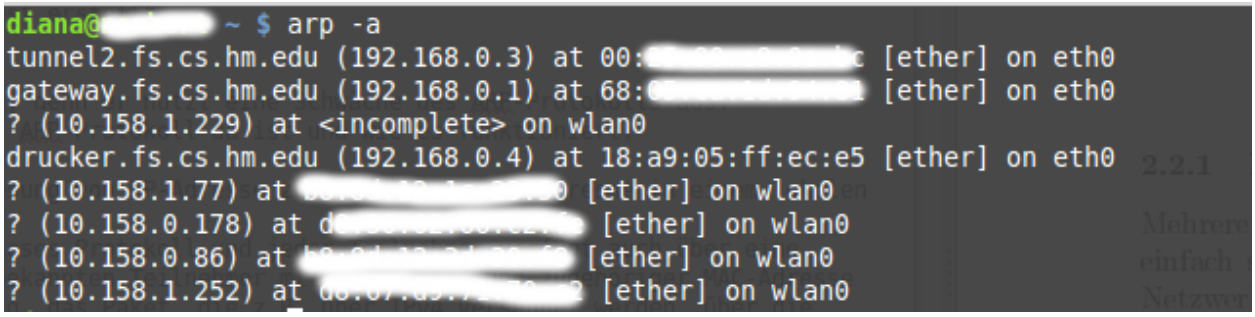


Abbildung 15: Auszug aus einer ARP-Tabelle, MAC-Adressen entfernt

Das ARP-Protokoll dient zur Auflösung von IP-Adresse zu zugehöriger MAC-Adresse in einem lokalen Netzwerk. Jeder Netzwerkteilnehmer kennt dieses Protokoll und jeder Teilnehmer verfügt auch über eine ARP-Tabelle, in welcher die ihm bekannten Teilnehmer mit IP-Adresse und zugehöriger MAC-Adresse hinterlegt sind. Das ist notwendig, um die Pakete, die z.B. über IPv4 versendet werden, über die einzelnen Komponenten im Netzwerk weiterzuleiten, d.h.: versendet mein Rechner eine Anfrage nach außen, z.B. ein Aufruf der Seite www.hm.edu, so steht in Header der Schicht 3 die IP-Adresse des geforderten Servers. Auf Schicht 2 werden immer MAC-Adressen benötigt, um zum nächstmöglichen Gerät zu senden. In meinem Heimnetzwerk wäre das z.B. ein Router, denn es wurde ja eine externe IP-Adresse aufgerufen, kein intern bekanntes Geräte wie z.B. ein Drucker. Der Rechner kennt den Router bereits, und weiß, dass er diesen wählen muss, wenn eine externe IP-Adresse angefragt ist. Somit wählt er die MAC-Adresse des Routers und sendet die Frames an diese Adresse.

Damit sich alle Rechner in einem Netzwerk kennen, wird in regelmäßigen Abständen eine Broadcast-Nachricht, ein sogenannter ARP-Request, gesendet, um zu ermitteln, welche MAC-Adresse zu einer IP-Adresse gehört und diese auch aktuell zu halten, damit keine Daten an Geräte gesendet werden, die sich eventuell gar nicht mehr im Netzwerk befinden. Somit kennen sich alle Geräte in einem internen Netzwerk.
Das Protokoll hat zwei Schwäche, die von einem Angreifer ausgenutzt werden können.[22, vgl.]

1. Jede ARP-Response wird ausgewertet, auch wenn es gar keinen unmittelbaren Request gab

2. Die Identität des Netzwerkteilnehmer wird nicht überprüft, es wird einfach angenommen, dass das Gerät korrekt funktioniert

2.4 Theoretisches Beispiel eines Angriffs mit ARP-Spoofing

In einem Netzwerk befinden sich folgende Teilnehmer

- Fabian
- Diana
- X

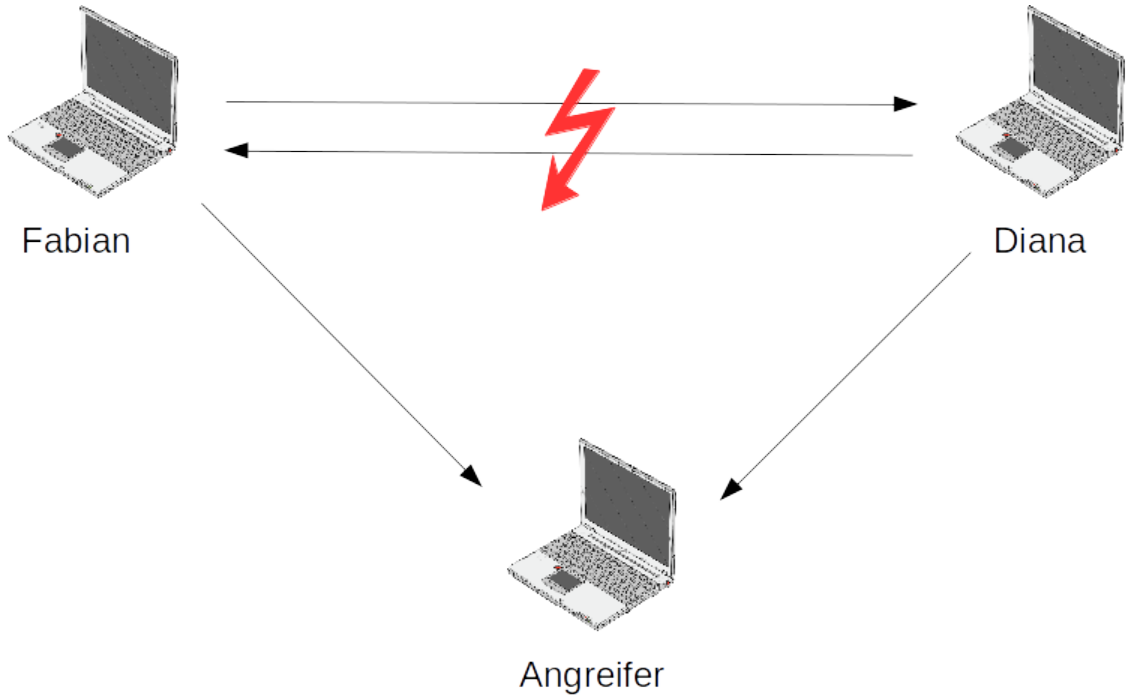


Abbildung 16: Darstellung der Kommunikation nach erfolgreicher Manipulation der ARP-Tabellen [23]

- Angreifer Y

Der Angreifer Y möchte den Datenverkehr von Fabian und Diana mitlesen und/oder verändern. Dafür muss er seine eigenen MAC-Adresse in die ARP-Tabelle von Fabian und Diana zu der jeweiligen IP-Adresse der Kommunikationspartner eintragen. Die ARP-Einträge von Fabian und Diana sehen dann wie folgt aus:

ARP-Tabelle von Fabian		
IP-Adresse Diana	:	MAC-Adresse von Y
IP-Adresse X	:	MAC-Adresse von X
IP-Adresse Y	:	MAC-Adresse von Y

ARP-Tabelle von Diana		
IP-Adresse Fabian	:	MAC-Adresse von Y
IP-Adresse X	:	MAC-Adresse von X
IP-Adresse Y	:	MAC-Adresse von Y

Diese gefälschten Einträge in der ARP-Tabelle der Netzwerkteilnehmer kann man mit dem Werkzeug dsniff umsetzten, welche später genauer Beschreiben wird.

Sind die ARP-Tabllen zugunsten des Angreifers verändert, kann dieser nun den Datenverkehr von Fabian und Diana mitlesen, da Fabian die Pakte bzw. Frames für Diana an die MAC-Adresse des Angreifers senden wird, Diana die Frames für Fabian ebenfalls. Die Frames werden über den Switch an den Angreifer verteilt, Fabian und Diana erhalten diese nicht. Dies gilt für geswitchte Netzwerke ebenso wie für Netzwerke, in welchem ein Hub eingesetzt ist. Diana und Fabian würden in diesem Fall zwar die Frames erhalten, diese allerdings verwerfen, da sie nicht für sie adressiert sind.

Zu diesem Zeitpunkt empfängt der Angreifer die Daten, die vermeintlich zwischen Fabian und Diana gesendet werden. Allerdings muss der Angreifer zudem dafür sorgen, dass Fabian und Diana die Frames erhalten, die erwartet werden, denn das Protokoll, dass für die Kommunikation darüber liegt, TCP, würde die Verbindung sonst abbrechen. Es würden keine Daten ausgetauscht werden, da somit schon der 3-Wege-Handshake des TCP-Protokolls scheitert. Der Angreifer muss dafür sorgen, dass die Frames bzw. Pakete an den eigentlichen Empfänger weitergeleitet werden, damit die Verbindung bestehen bleibt. Dafür muss die MAC-Adresse des eigentlichen Empfängers in den Header eingetragen werden, damit der Empfänger die Pakete nicht verwirft. Dabei können die Daten vom Angreifer manipuliert oder analysiert werden. Der Angreifer kann das mit mehreren Möglichkeiten erreichen, z.B. mit IP-Forwarding. Wichtig ist dabei nur, dass die Verzögerung im Rahmen des Timeouts bleibt, da die Verbindung sonst abbrechen könnte.

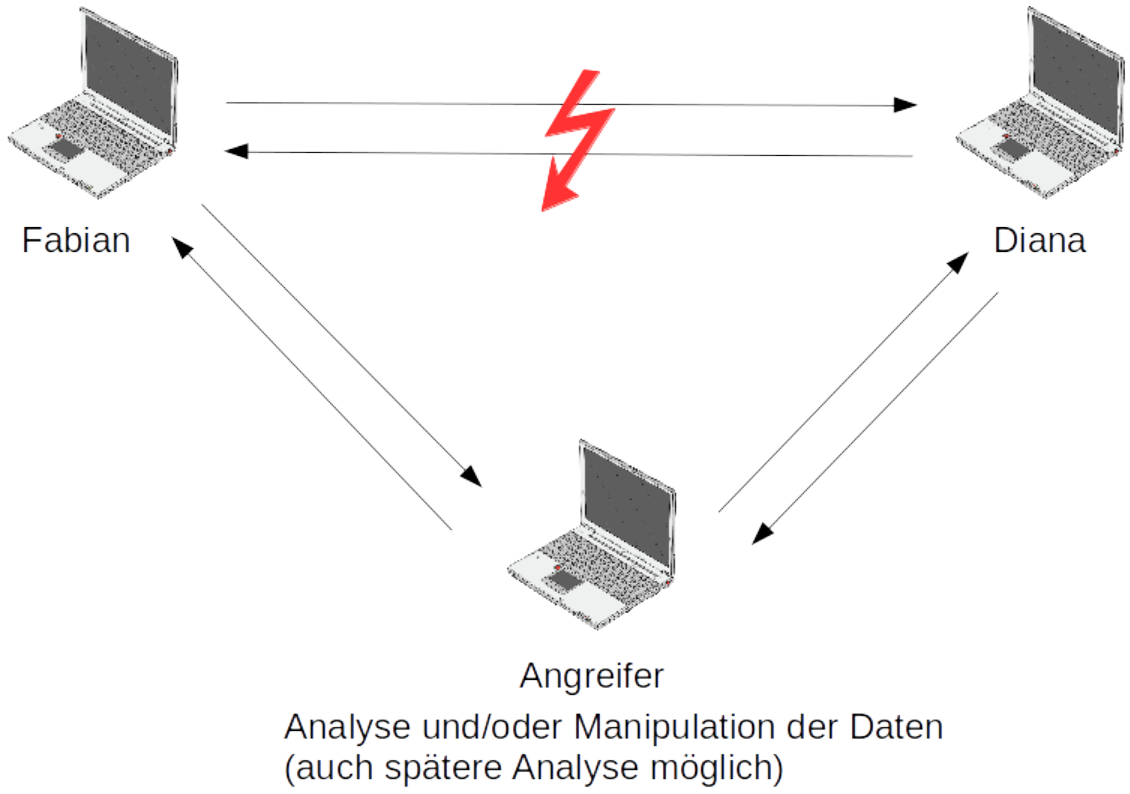


Abbildung 17: Angreifer muss die abgefangenen Frames an den eigentlichen Empfänger weiterleiten, damit Verbindung aufrecht erhalten bleibt [24]

Für die Analyse der abgefangenen Daten gibt es ebenfalls Werkzeuge, welche mit Filtern nach den gewünschten Informationen suchen.

2.5 Werkzeug zur Umsetzung eines ARP-Spoofing-Angriffs

Es gibt mehrere Werkzeuge, die sich gut dafür eigenen, hier wird dsniff näher erläutert [25, vgl.].

dsniff bietet eine Sammlung von Werkzeugen, mit welchen eine Man-in-the-middle-Attakte umgesetzt werden kann [26, vgl.]. Unter anderen ist da der Passwort-Sniffer dsniff enthalten, welcher unverschlüsselte Passwörter im Netzwerk filtert.

arpspoof macht genau das, was oben beschreiben wurde. Dieses Werkzeuge leitet Pakete vom Zielhost, welchen in einem Netzwerk zu einem anderen Host gesendet werden sollen, um, indem die ARP-Antworten (ARP-Replies) vom Angreifer gefälscht werden.[27, vgl.]

Dazu kommen noch spezielle Sniffer wie urlsniff und mailsniff, und einiges mehr. Für den einfachen Zweck und diesem theoretischen Beispiel würde arospoof ausreichen.

Im Anhang ist eine Auflistung aller enthaltener Werkzeuge mit Funktionalität aufgelistet.

2.6 Schutzmaßnahmen und Kontrolle im Netzwerke

Die einzig wirklich sichere Methode zur Vermeidung von ARP-Spoofing in einem Netzwerk ist die manuelle Verwaltung der IP- und zugehörigen MAC-Adressen. Das bringt allerdings einiges an Arbeitsaufwand mit sich, und dieser steigt zudem mit jedem neuen Netzwerkteilnehmer. Gerade in einem Netzwerk mit wechselnden Geräten ist das fast nicht umsetzbar bzw. sehr unpraktisch.

Eine weitere Maßnahme wäre, den Netzwerkverkehr zu überwachen, z.B. am Switch, und auf eine erhöhte Anzahl an ARP-Responses zu achten. Häufen sich diese, könnte man einen Angriff verdächtigen und der Sache auf den Grund gehen.

Auch Arpwatch[28] wäre ein Möglichkeit, um das Netzwerk vor einem solchen Angriff zu schützen. Wenn das Programm gestartet wird, werden alle ARP-Pakete im Netzwerk, welche an einem Rechner ankommen, mit den bereits bekannten MAC-Adressen abgeglichen. Diese Adressen sind in der Datei arp.dat hinterlegt und werden da verwaltet. Taucht eine nicht bekannte Adresse auf, wird ein Eintrag in den syslogs gemacht und eine Nachricht an den Benutzer oder root gesendet. Hat sich die Zuordung von MAC- und IP-Adresse geändert, wird ebenfalls ein Eintrag in den syslogs gemacht. [29, vgl.]

Anhang

Liste aller Werkzeuge in dsniff mit Erklärung

- mitsniffen des gesamten Netzverkehrs

  1. **dsniff**:

     „is a password sniffer which handles FTP, Telnet, SMTP, HTTP, POP, poppass, NNTP, IMAP, SNMP, LDAP, Rlogin, RIP, OSPF, PPTP MS-CHAP, NFS, VRRP, YP/NIS, SOCKS, X11, CVS, IRC, AIM, ICQ, Napster, PostgreSQL, Meeting Maker, Citrix ICA, Symantec pcAnywhere, NAI Sniffer, Microsoft SMB, Oracle SQL*Net, Sybase and Microsoft SQL protocols.“[30]

  2. **filesnarf**:

     „saves files sniffed from NFS traffic in the current working directory.“[31]

  3. **mailsnarf**:

     „outputs e-mail messages sniffed from SMTP and POP traffic in Berkeley mbox format, suitable for offline browsing with your favorite mail reader.“[32]

  4. **msgsnarf**:

     „records selected messages from AOL Instant Messenger, ICQ 2000, IRC, MSN Messenger, or Yahoo Messenger chat sessions.“[33]

  5. **urlsnarf**:

     „outputs all requested URLs sniffed from HTTP traffic in CLF (Common Log Format, used by almost all web servers), suitable for offline post-processing with your favorite web log analysis tool (analog, wwwstat, etc.).“[34]

  6. **webspy**:

     „sends URLs sniffed from a client to your local Netscape browser for display, updated in real-time (as the target surfs, your browser surfs along with them, automagically). Netscape must be running on your local X display ahead of time.“[35]

- Manipulation im Netzverkehr

  1. **arpspoof**:

     „redirects packets from a target host (or all hosts) on the LAN intended for another host on the LAN by forging ARP replies. This is an extremely effective way of sniffing traffic on a switch.“[27].

  2. **dnsspoof**:

     „forges replies to arbitrary DNS address / pointer queries on the LAN. This is useful in bypassing hostname-based access controls, or in implementing a variety of man-in-the-middle attacks. “[36].

3. **macof**:

   „floods the local network with random MAC addresses (causing some switches to fail open in repeating mode, facilitating sniffing).“[37]

- Man-in-the-middle-Attake in SSH/TLS-Verbindungen

  1. **sshmitm**:

     „proxies and sniffs SSH traffic redirected by dnsspoof(8), capturing SSH password logins, and optionally hijacking interactive sessions. Only SSH protocol version 1 is (or ever will be) supported - this program is far too evil already.“[38]

  2. **webmitm**:

     „transparently proxies and sniffs HTTP / HTTPS traffic redirected by dnsspoof(8), capturing most ßecureSSSL-encrypted webmail logins and form submissions.“[39]

Literatur

[1] TELECOMMUNICATION STANDARDIZATION SECTOR OF INTERNATIONAL TELECOMMUNICATION UNION (ITU-T).
X.200 : Information technology - Open Systems Interconnection - Basic Reference Model: The basic model,
01. Juli 1994.
http://www.itu.int/rec/T-REC-X.200-199407-I
(Abrufdatum: 15.07.2016).

[2] Serge Malenkovich.
Was ist eine Man-in-the-Middle-Attacke?,
10. April 2013.
https://blog.kaspersky.de/was-ist-eine-man-in-the-middle-attacke/905/
(Abrufdatum: 15.07.2016).

[3] Bild durch Fabian Uhlmann aus Bildern von nachfolgend aufgeführten URLs zusammengestellt.
http://ecx.images-amazon.com/images/I/81LcrgMpIcL._SL1500_.jpg,
http://i5.walmartimages.com/dfw/dce07b8c-6381/k2-_
aa74fc49-072f-4d83-afdc-4164105621f2.v1.jpg,
https://gigaom.com/wp-content/uploads/sites/1/2013/07/
hacker-cyber-attack-640x522.jpg
(Abrufdatum: 15.07.2016).

[4] R. Fielding and J. Reschke.
Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing. RFC 7230, RFC Editor,
Juni 2014.
http://www.rfc-editor.org/rfc/rfc7230.txt (Abrufdatum: 15.07.2016).

[5] A. Freier, P. Karlton, and P. Kocher.
The Secure Sockets Layer (SSL) Protocol Version 3.0. RFC 6101, RFC Editor,
August 2011.
http://www.rfc-editor.org/rfc/rfc6101.txt (Abrufdatum: 15.07.2016).

[6] E. Rescorla.
HTTP Over TLS. RFC 2818, RFC Editor,
Mai 2000.
http://www.rfc-editor.org/rfc/rfc2818.txt (Abrufdatum: 15.07.2016).

[7] TELECOMMUNICATION STANDARDIZATION SECTOR OF INTERNATIONAL TELECOMMUNICATION UNION (ITU-T).
X.509 : Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks,
14. Oktober 2012.
https://www.itu.int/rec/T-REC-X.509-201210-I/en
(Abrufdatum: 15.07.2016).

[8] Russell Housley, Warwick Ford, Tim Polk, and David Solo.
Internet X.509 Public Key Infrastructure Certificate and CRL Profile. RFC 2459, Januar 1999.
http://www.rfc-editor.org/rfc/rfc2459.txt (Abrufdatum: 15.07.2016).

[9] STAMFORD, Conn.
Gartner Says Worldwide PC Shipments Declined 9.6 Percent in First Quarter of 2016, 11. April 2016.
http://www.gartner.com/newsroom/id/3280626 (Abrufdatum: 15.07.2016).

[10] Hanno Böck.
Lenovo-Laptops durch Superfish-Adware angreifbar,
19. Februar 2015.
http://www.golem.de/news/adware-lenovo-laptops-durch-superfish-adware-angreifbar-
html (Abrufdatum: 15.07.2016).

[11] Screenshot by Robert Graham.
http://www.cnet.com/how-to/lenovo-superfish-adware-uninstall-fix/
(Abrufdatum: 15.07.2016).

[12] Marc Rogers.
Lenovo installs adware on customer laptops and compromises ALL SSL,
19. Februar 2015.
http://marcrogers.org/2015/02/19/lenovo-installs-adware-on-customer-laptops-and-c
(Abrufdatum: 15.07.2016).

[13] Jürgen Schmidt. Kryptoverfahren SHA-1 geknackt, 16. Februar 2005.
http://www.heise.de/newsticker/meldung/Kryptoverfahren-SHA-1-geknackt-135372.
html (Abrufdatum: 15.07.2016).

[14] Andrea Pellegrini, Valeria Bertacco, and Todd Austin. Fault-Based Attack of RSA Authentication, 2010.
http://web.eecs.umich.edu/~valeria/research/publications/DATE10RSA.pdf
(Abrufdatum: 15.07.2016).

[15] Robert Graham.
Extracting the SuperFish certificate,
19. Februar 2015.
http://blog.erratasec.com/2015/02/extracting-superfish-certificate.
html#.V4uWdu22Hfa (Abrufdatum: 15.07.2016).

[16] Joe Nord.
New Dell computer comes with a eDellRoot trusted root certificate,
22. November 2015.
http://joenord.blogspot.de/2015/11/new-dell-computer-comes-with-edellroot.
html (Abrufdatum: 15.07.2016).

[17] DELL.
Informationen zu den Zertifikaten eDellRoot und DSDTestProvider und Anweisungen

zum Entfernen dieser Zertifikate von Ihrem Dell PC,
letzte Modifizierung 20 Juni 2016.
http://www.dell.com/support/article/us/en/19/SLN300321/DE
(Abrufdatum: 15.07.2016).

[18] Liam Tung.
How to remove Dell's 'Superfish 2.0' root certificate - permanently,
24. November 2015.
http://www.zdnet.com/article/how-to-remove-dells-superfish-2-0-root-certificate-p
(Abrufdatum: 15.07.2016).

[19] Wolfgang Riggert.
Rechnernetze - Grundlagen, Ethernet, Internet,
2014.

[20] Prof. Dr. Lars Wischhof. Skript aus Vorlesung "Netzwerke I", Bachelor Informatik,
Hochschule München,
WS 2015/16.
http://www.cs.hm.edu/die_fakultaet/ansprechpartner/professoren/wischof/
index.de.html.

[21] Abbildung eines Switches.
http://cliparts.co/clipart/2346052.

[22] Michael Psarros, Hannes Oberender, Florian Bache.
Angriffe in geswitchten Netzwerken,
2013.
http://ei.ruhr-uni-bochum.de/media/ei/lehrmaterialien/225/
3b10ff4588f51446e1d93dc5f28ab4b88532fe47/AngriffeInGeswitchtenNetzwerken.
pdf.

[23] Abbildung Manipulation der ARP-Tabelle.
http://cliparts.co/clipart/31556.

[24] Notwendingkeit der Weiterleitung der abgefangenen Frames.
http://cliparts.co/clipart/31556.

[25] dsniff - Beschreibung und Download.
https://www.monkey.org/~dugsong/dsniff/.

[26] dsniff - Beschreibung der Werkzeuge.
http://www.ouah.org/dsniffintr.htm.

[27] arpspoof - Erklärung.
http://linux.die.net/man/8/arpspoof.

[28] arpwatch - download.
http://ee.lbl.gov/.

[29] arpwatch - man page.
http://linux.die.net/man/8/arpwatch.

[30] dsniff - Erklärung.
http://linux.die.net/man/8/dsniff.

[31] filesnarf - Erklärung.
http://linux.die.net/man/8/filesnarf.

[32] mailsnarf - Erklärung.
http://linux.die.net/man/8/mailsnarf.

[33] msgsnarf - Erklärung.
http://linux.die.net/man/8/msgsnarf.

[34] urlsnarf - Erklärung.
http://linux.die.net/man/8/urlsnarf.

[35] webspy - Erklärung.
http://linux.die.net/man/8/webspy.

[36] dnssppf - Erklärung.
http://linux.die.net/man/8/dnsspoof.

[37] macof - Erklärung.
http://linux.die.net/man/8/macof.

[38] sshmitm - Erklärung.
http://linux.die.net/man/8/sshmitm.

[39] webmitm - Erklärung.
http://linux.die.net/man/8/webmitm.