
Erweiterte Benutzerführung für den Umgang mit sicheren Verbindungen in Browsern

Martin Stopczynski, FB 20, Informatik (Bachelor of Science)
Bachelorarbeit

Technische Universität Darmstadt
Fachbereich Informatik
Fachgebiet Kryptographie und Computeralgebra

Prof. Dr. Johannes Buchmann

Betreuer: Dr. Alexander Wiesmaier

27. Mai 2009



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Zusammenfassung

Die vorliegende Bachelorarbeit beschäftigt sich mit dem Umgang sowie der Benutzerführung von gesicherten Verbindungen und Zertifikaten in Browsern.

Beginnend mit einer begrifflichen Einführung in das Thema und einer Beschreibung wichtiger verwendeter kryptographischer Verfahren, folgt anschließend eine Analyse der aktuellen Situation. Hier wird anhand verschiedener Szenarien untersucht, wie die Benutzerführung in den derzeit populären Browsern bei gültigen, sowie aus verschiedenen Gründen ungültigen Zertifikaten, abläuft. Dabei wird ein besonderer Augenmerk auf die Schwachstellen gelegt, welche die einzelnen Browser aufweisen. Anhand von Beispielen werden die resultierenden Probleme herausgearbeitet und Verbesserungsvorschläge präsentiert.

In der konzeptuellen Entwicklung weiterer Lösungsansätze werden darauffolgend verschiedene Methoden gezeigt, wie das Sicherheitsbewusstsein des Benutzers und der Umgang mit gesicherten Verbindungen verbessert werden kann.

Der Fokus der Arbeit richtet sich dabei auf die Analyse sicherheitsrelevanter Kryptographieaspekte und die Einstufung der daraus resultierender Qualität der Verbindung. Dazu gehört beispielsweise die Untersuchung des Hash-Algorithmus, Signaturverfahrens sowie des Verschlüsselungsstandards der gesicherten Verbindung.

Anhand einer grafischen Darstellung soll die Einstufung des Sicherheitszustands einer geschützten Verbindung symbolisiert und dem Benutzer die Güte der Verbindung aufzeigen werden.

Inhaltsverzeichnis

Zusammenfassung	1
Inhaltsverzeichnis	2
Abbildungsverzeichnis	4
1. Einleitung	7
1.1. Motivation	7
1.2. Problemstellung und Zielsetzung	7
2. Grundlagen	9
2.1. World Wide Web	9
2.2. Web-Browser	9
2.3. HTTP	10
2.4. Schutzziele von Informationssicherheit	10
2.5. HTTPS	11
2.6. SSL/TLS	12
2.7. SSL Handshake-Protokoll	14
2.8. Public Key Infrastructure	15
2.9. Zertifikate	15
2.9.1. Wurzelzertifikate	16
2.9.2. Zertifizierungsstelle und Registrierungsstelle	18
2.9.3. Extended Validation SSL Zertifikat	18
2.10. Digitale Signaturen	18
2.10.1. RSA	19
2.10.2. Digital Signature Algorithm	20
2.11. Hashfunktionen	20
2.11.1. MD5	21
2.11.2. RIPEMD-160	21
2.11.3. SHA-1	21
2.12. Verschlüsselung	22
2.12.1. RC4	22
2.12.2. Data Encryption Standard	22
2.12.3. 23	
2.12.3. Advanced Encryption Standard	23
3. Aktuelle Situation	24
3.1. Microsoft Internet Explorer 6.0	25
3.1.1. Analyse gesicherter Verbindungen mit gültigem Zertifikat	25
3.1.2. Analyse gesicherter Verbindungen mit ungültigem Zertifikat	27
3.2. Microsoft Internet Explorer 7.0	29
3.2.1. Analyse gesicherter Verbindungen mit gültigem Zertifikat	29
3.2.2. Analyse gesicherter Verbindungen mit ungültigem Zertifikat	30

3.3.	Microsoft Internet Explorer 8.0	32
3.4.	Mozilla Firefox 3.0.10	32
3.4.1.	Analyse gesicherter Verbindungen mit gültigem Zertifikat	32
3.4.2.	Analyse gesicherter Verbindungen mit ungültigem Zertifikat	35
3.5.	Mozilla Firefox 3.5 Beta	41
3.6.	Safari 4 Beta	42
3.6.1.	Analyse gesicherter Verbindungen mit gültigem Zertifikat	42
3.6.2.	Analyse gesicherter Verbindungen mit ungültigem Zertifikat	43
3.7.	Übersicht der Benutzbarkeit der Browser	45
4.	Konzept zur Verbesserung der Benutzerführung	46
4.1.	Adresszeile einfärben	47
4.1.1.	Gesicherte Verbindung mit gültigem Zertifikat und vorinstalliertem Wurzelzertifikat	48
4.1.2.	Gesicherte Verbindung mit gültigem Zertifikat ohne vorinstalliertem Wurzelzertifikat	48
4.1.3.	Gesicherte Verbindung mit gültigem EV-SSL Zertifikat und Wurzelzertifikat	49
4.1.4.	Gesicherte Verbindung mit ungültigem Zertifikat	49
4.1.5.	Realisierung	49
4.2.	Passwortfelder einfärben	49
4.3.	Public-Key Verifizierung durch unabhängige Notar-Server	50
4.4.	Sicherheits-Add-on: Sec-Rank (FF Mock-up)	52
4.4.1.	Art des Zertifikats	54
4.4.2.	Klassifizierung der Sicherheit	55
4.5.	Error-Code-Page im FF bei nicht vorhandenem Wurzelzertifikat	57
4.6.	Erweiterte Hilfe im Zertifikats-Manager	57
5.	Erweiterungsmöglichkeiten	59
5.1.	Website Rating	59
5.2.	Zentrale Zertifikatsdatenbank	61
6.	Fazit	62
	Anhang	63
	Literaturverzeichnis	63

Abbildungsverzeichnis

Abbildung 1: Client-Server Kommunikation [10]	9
Abbildung 2: SSL im verkürztem OSI -Modell [12]	12
Abbildung 3: HTTPS Verbindungsaufbau – Modell [18]	13
Abbildung 4: Zertifikatshierarchie der TU Darmstadt Web-Seite	17
Abbildung 5: Warnhinweis des Firefox Browsers, falls das Zertifikat unbekannt ist	17
Abbildung 6: Genereller Ablauf signierter Kommunikation [5]	19
Abbildung 7: Browser Marktanteile März 2009 [16]	24
Abbildung 8: IE6 - Allgemeinen Zertifikatsdetails	26
Abbildung 9: IE6 - Zertifikatsdetails	26
Abbildung 10: IE6 - Zertifizierungspfad bei gültigem Zertifikat	27
Abbildung 11: IE6 - Sicherheitshinweise, wenn dem Zertifikat des Servers nicht vertraut wird	27
Abbildung 12: IE6 - Zertifizierungspfad bei unbekanntem Wurzelzertifikat	27
Abbildung 13: IE6 - Zertifikatsstelle unbekannt	27
Abbildung 14: IE6 - Scheinbar gültiges Zertifikat	28
Abbildung 15: IE6 - Zertifikat abgelaufen (Allg. Informationen links / Zertifizierungspfad rechts)	28
Abbildung 16: IE7 - Adresszeile mit neuem Sicherheitsschloss-Symbol	29
Abbildung 17: IE7 - Zertifikats Kurzinfo-Fenster – Links mit EV-SSL, rechts ohne	29
Abbildung 18: IE7 - Erweiterte Zertifikatsfehler Informationen	30
Abbildung 19: IE7 - Fehlermeldung bei HTTPS-Seiten mit unbekanntem Wurzelzertifikat	31
Abbildung 20: IE7 - Domainname aus dem Zertifikat weicht von der aufgerufenen URL ab	31
Abbildung 21: IE7 - Das Zertifikat ist abgelaufen	32
Abbildung 22: FF2 - Adresszeile einer gesicherten Verbindung	32
Abbildung 23: FF3 - Statuszeile im Browserfuß	32
Abbildung 24: FF3 - Seiten- und Sicherheitsinformationen	33
Abbildung 25: FF3 - Detaillierte Seiten- sowie Sicherheitsinformationen	33
Abbildung 26: FF3 - Allg. Zertifikatsinformationen (links) sowie Zertifizierungspfad (rechts)	34
Abbildung 27: FF3 - Extended Validation, Instant Website ID	35
Abbildung 28: FF3 - Allg. Seiteninformationen einer gesicherten Verbindung mit EV-SSL Validation	35
Abbildung 29: FF3 - Fehlerseite, Unbekanntes Wurzelzertifikat	36
Abbildung 30: FF3 - Instand Website ID – Unknown Issuer	36
Abbildung 31: FF3 - Seiten- und Sicherheitsinformationen	37
Abbildung 32: FF3 - Allgemeine Seiteninformationen	37
Abbildung 33: FF3 - Zertifikatsinformationen herunterladen	38
Abbildung 34: FF3 - Zertifikat Status	38
Abbildung 35: FF3 - (Rechts) Allg. Zertifikatsinformationen, (Links) Zertifizierungspfad	38
Abbildung 36: FF3 - Instant Website ID – Ausnahme wurde hinzugefügt	39
Abbildung 37: FF3 - Seiten- und Sicherheitsinformationen	39
Abbildung 38: FF3 - Ungültiges Sicherheitszertifikat (Bad Cert Domain)	40
Abbildung 39: FF3 - Bad Cert Domain Warnung	40
Abbildung 40: FF3 - Abgelaufenes Zertifikat	40
Abbildung 41: FF3 - Untrusted Issuer	41
Abbildung 42: FF3.5 - Warnmeldung bei unbekanntem Wurzelzertifikat	41
Abbildung 43: Safari 4 - Sicherheitsschloss Symbol bei gesicherten Verbindungen	42
Abbildung 44: Safari 4 - Extended SSL Validierung	42
Abbildung 45: Safari 4 - Zertifikatsübersicht	43
Abbildung 46: Safari 4 - Unkown Issuer	43
Abbildung 47: Safari 4 - Bad Cert Domain	44
Abbildung 48: Safari 4 - Bad Cert Domain, Trusted CA	44

Abbildung 49: Safari 4 - Untrusted Issuer	45
Abbildung 50: Studie zur Internetsicherheit [20]	46
Abbildung 51: Umfrage zu den Ursachen einer Sicherheitswarnung [20]	47
Abbildung 52: Adresszeile wird hell grün eingefärbt	48
Abbildung 53: Adresszeile wird gelb eingefärbt	48
Abbildung 54: Adresszeile wird grün eingefärbt	49
Abbildung 55: Adresszeile wird rot eingefärbt	49
Abbildung 56: Passwortfelder auf Webseiten einfärben	50
Abbildung 57: Funktionsweise von Perspectives	51
Abbildung 58: Statuszeilenanzeige über die Dauer der Konsistenz des Public-Keys	51
Abbildung 59: Kontrollgrafik der Konsistenz des Public-Key, aufgezeichnet von den Notar-Servern	52
Abbildung 60: Add-on Sec-Rank in der Browserleiste des FF	52
Abbildung 61: „Sec Rank“ - Bewertungsskala	53
Abbildung 62: „Sec-Rank“ – Erweiterte Anzeige der Klassifizierung der Sicherheitsmerkmale	53
Abbildung 63: Skala der Zertifikatseigenschaften	54
Abbildung 64: Klassifizierung des Hash-Algorithmus	56
Abbildung 65: Klassifizierung des Signaturverfahrens	56
Abbildung 66: Klassifizierung der Verschlüsselung	57
Abbildung 67: Error Code Page Umgehung	57
Abbildung 68: Erweiterte Hilfe im Zertifikats Manager des FF	58
Abbildung 69: WOT Bewertungssystem	59
Abbildung 70: WOT Bewertungskriterien	60

Ehrenwörtliche Erklärung

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig angefertigt habe. Sämtliche aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und noch nicht veröffentlicht.

Darmstadt, den 27. Mai 2009

1. Einleitung

Diese Arbeit ist in sechs Kapitel unterteilt und gliedert sich wie folgt. Kapitel 1 „Einleitung“ präsentiert die Motivation dieser Bachelorarbeit sowie die Beschreibung der Problemstellung und der daraus resultierenden Zielsetzung zur Verbesserung der aktuellen Situation. In Kapitel 2 „Grundlagen“ erfolgt die Einführung in das Themengebiet mit einer Erläuterung der Grundlagen wichtiger Begriffe. Anschließend folgt in Kapitel 3 „Aktuelle Situation“ die Analyse der jetzigen Schwachstellen der Browser. Kapitel 4 „Konzept zur Verbesserung der Benutzerführung“ führt Ansätze auf, wie der aktuelle Umgang mit gesicherten Verbindungen in der Benutzerführung verbessert werden kann. In Kapitel 5 „Erweiterungen“ werden zusätzlich zwei mögliche Erweiterungen zum vorgestellten Konzept erläutert. Das Kapitel 6 „Fazit“ schließt mit einer Zusammenfassung der Analyse und der erarbeiteten Konzeptvorschläge ab.

1.1. Motivation

Die technische Weiterentwicklung in unserer Zeit und die daraus resultierende Vereinfachung der Umgebung, bewirkt einen selbstverständlichen Umgang mit den Technologien, so dass die Menschen sich über die Funktionalität keine Gedanken mehr machen.

Das Benutzen von Computern sowie des Internets ist für die heutige Generation natürlich geworden, so dass über dessen Anwendung nicht mehr nachgedacht wird. Der Gebrauch des Internets verhält sich vergleichbar wie das Tippen auf der Tastatur als so selbstverständlich, dass die Menschen nicht mehr darüber nachdenken, was die Tastatur oder das Internet ist und wie sie funktionieren.

Erst durch auftretende Schwierigkeiten, wie einer Fehlfunktion der Tastatur, oder einer Warnmeldung nach dem Aufrufen einer gesicherten Internetverbindung, wird dem Nutzer der Umgang dieser Technik bewusst. Die Problematik die dabei auftritt ist, dass durch das natürliche Benutzen des Internets, die Menschen heutzutage mit Warn- oder Fehlermeldungen nur selten etwas anfangen können.

Wird beispielsweise die Internetseite einer Bank zum Online-Banking aufgerufen, geht der Benutzer automatisch davon aus, dass er sich auf einer sicheren Verbindung befindet. In der Regel wird diese Tatsache durch das Einblenden eines „Schloss-Symbols“ im Browserfenster signalisiert. Über die Sicherheitskriterien oder die Funktionsweise machen sich die Benutzer in diesem Fall keine Gedanken.

Erst wenn zum Beispiel die gesicherten Seiten einer Universität aufgerufen werden und der Browser mit einer Warnmeldung den Zutritt zu der Seite unterbricht, weil kein entsprechendes Root-Zertifikat bekannt ist, wird eine Reaktion vom Benutzer verlangt. Durch das fehlende Wissen über diese Technologie, versucht der Benutzer, meist durch Ausprobieren oder Anwendung vorher ausgeführter Aktionen, den Zugang zu der Seite zu erlangen, ohne darüber nachzudenken, wieso diese Warnmeldung aufgetreten ist. Verstärkt wird die Unklarheit über die Bedeutung der Warnmeldung durch die Tatsache, dass zu wenig verständliche Informationen dem Benutzer bereitgestellt werden.

Der Umgang mit gesicherten Verbindungen sollte jedoch das Sicherheitsbewusstsein des Nutzers wecken sowie über den Umgang aufklären.

Durch die Problematik, dass die derzeit eingesetzten Browser diesbezüglich ungenügende Informationen bieten, entstand unter der Leitung von Prof. Dr. J. Buchmann die Idee dieser Bachelorarbeit mit der Aufgabe, die Benutzerführung mit dem Umgang von gesicherten Verbindungen in Browsern zu verbessern.

1.2. Problemstellung und Zielsetzung

Der Browser erkennt gesicherte Verbindungen automatisch und überprüft die Gültigkeit anhand eines digitalen Zertifikats, also ob die Sicherheitskriterien erfüllt werden. Der vollständige technische Ablauf wird im Kapitel 2 „Grundlagen“ erläutert.

In der Regel ruft der Browser Internetseiten mit gesicherten sowie ungesicherte Verbindungen gleichermaßen auf. Der Benutzer bemerkt in diesem Fall keinen Unterschied, bis auf ein „Schloss-Symbol“, welches in den gängigen Browsers bei gesicherten Verbindungen zusätzlich eingeblendet wird.

Problematisch sind Szenarien, in denen die Sicherheitskriterien einer gesicherten Verbindung nicht erfüllt werden und der Browser den Aufruf dieser Internetseite unterbricht. Dies kann mehrere Ursachen haben, wie beispielsweise einen Fehler im Zertifikat oder das nicht Vorhandensein eines übergeordneten Verifizierungszertifikats.

Um die Internetseite dennoch aufzurufen, fordert der Browser den Benutzer dazu auf, sich mit dem Fehler auseinanderzusetzen und über ein weiteres Vorgehen zu entscheiden. Hier liegt derzeit die Schwachstelle der Browser, den Benutzer in angemessener Form über den aufgetretenen Fehler zu informieren sowie einen Status dieser gesicherten Verbindung zu präsentieren.

Diese Informationen sind es allerdings, auf dessen Grundlage hin der Benutzer die Entscheidung treffen sollte, ob er der Internetseite vertraut und somit den Zugriff fortführen möchte.

Der Mangel an einer verständlichen Aufbereitung der sicherheitsrelevanten Informationen für den Benutzer stellt somit das Hauptproblem mit dem Umgang von gesicherten Verbindungen dar.

Ziel dieser Arbeit ist es, die verschiedenen Sicherheitsaspekte von gesicherten Verbindungen herauszuarbeiten und in einer für den Benutzer verständlichen Form aufzuzeigen. Durch Visualisierung sowie Erläuterung kritischer Sicherheitsmerkmale, soll dem Benutzer geholfen werden zu erkennen, wie sicher oder unsicher die Verbindung einzustufen ist.

2. Grundlagen

In diesem Kapitel werden die Grundlagen einer sicheren Kommunikation des Web-Browsers mit dem World Wide Web (WWW) erläutert. Dies umfasst den Einsatz der Datenübertragungsprotokolle, die Funktionsweise der Kryptographieverfahren und der eingesetzten Verschlüsselungsalgorithmen.

2.1. World Wide Web

Das World Wide Web ist ein über das Internet abrufbares Hypertext-System und integriert bestehende Internetdienste durch einheitliche Adressierung sowie Bedienung. Man unterscheidet zwischen WWW-Objekten, -Server und -Browser.

WWW-Objekte können Dokumente oder beliebige Datenbestände wie Datenbanken sein. Weiterhin sind Web-Seiten spezielle WWW-Objekte, die in der Hypertext Markup Language (HTML) geschrieben und mittels des Hypertext Transfer Protocols (HTTP – Kapitel 2.3) übertragen werden. Zum Laden und Anzeigen von Web-Seiten wird ein Web-Browser benötigt, um die Daten vom Web-Server zu laden und auf dem Bildschirm anzuzeigen [6].

2.2. Web-Browser

Web-Browser, wie der Microsoft Internet Explorer oder Mozilla Firefox stellen Dienste zum Navigieren im WWW, zur Darstellung von Web-Seiten sowie Interaktion mit dem Web-Server zur Verfügung.

Mittels eines Browsers können WWW-Objekte von einem Server abgerufen, oder auch interaktive Benutzereingaben an den Server weitergeleitet werden [6].

Die Kommunikation zwischen einem lokalen Rechner (Client) und einem Web-Server geschieht über das Internet. Der Verbindungsaufbau wird vom Client über ein lokales Programm, dem Web-Browser initiiert. Der Browser verarbeitet dabei die Anfragen, die ein Benutzer an einen Web-Server im Internet stellt (siehe Abb. 1). Dies können Webseiten, Datenbankzugriffe oder auch Dateien sein.

Die Aufgabe des Browser ist dabei die Darstellung der Web-Seite sowie die Verarbeitung der empfangenen Daten. Betrachtet man das verkürzte OSI-(Open Systems Interconnection Reference Model) Schichtenmodell (Abb. 2), welches als Designgrundlage für Kommunikationsprotokolle dient, befindet sich der Web-Browser in der obersten Schicht, also der Anwendungsschicht und ist für die Funktionsaufrufe zuständig. Der eigentliche Datenübertragungs- und Verarbeitungsprozess wird in den darunterliegenden Schichten abgewickelt.



Abbildung 1: Client-Server Kommunikation [10]

Die Beziehung zwischen lokalem Rechner (Benutzer/Client) und Server wird als Client-Server-Infrastruktur bezeichnet und beschreibt die Kommunikation der involvierten Rechner.

Der Client stellt eine Anfrage zur Benutzung eines bestimmten Dienstes an den Server, der wiederum die Anfrage entgegennimmt, auswertet und den Client durch das Bereitstellen des gewünschten Dienstes

bedient [10]. Der Web-Browser verarbeitet somit die empfangenen Daten und übernimmt das Verschieken der Anfragen des Benutzers an den Server.

Die Kommunikation geschieht dabei über das HTTP-Protokoll und wird im folgenden Unterkapitel beschrieben.

2.3. HTTP

Das Hypertext Transfer Protocol (HTTP) ist ein standardisiertes Verfahren, mit dem Web-Browser und Web-Server miteinander kommunizieren.

Seine Aufgabe besteht hauptsächlich darin, Webseiten und andere Daten oder Dateien aus dem Internet in einen Web-Browser zu übertragen. Eine HTTP-Transaktion, bei der beispielsweise eine Datei geladen werden soll, folgt dabei immer dem gleichen Schema.

Als erstes initiiert der Client eine Verbindung zum Server. Dies geschieht meist mithilfe des TCP/IP-Protokolls. Ist die Verbindung zustande gekommen, beginnt der Datenaustausch. Der Client schickt dazu eine Anfrage (Request) an den Server, in der er die entsprechende Datei anfordert. Der Server verarbeitet diese Anfrage und schickt eine Antwort (Response) zurück. HTTP ist so ausgelegt, dass beide Teilnehmer die Verbindung möglichst solange aufrecht erhalten, bis eine Transaktion abgeschlossen ist. Sollte die Verbindung zwischenzeitlich abbrechen, muss sie erneut aufgebaut werden und das Verfahren beginnt von neuem.

HTTP lässt allgemein nur kurzzeitige Verbindungen zu. Hat ein Server seine Antwort abgeschickt, beendet er die Verbindung zumeist auch gleich wieder. Dadurch kann der Server genügend freie Verbindungen zu anderen Clients bearbeiten und verhindern, dass eine Verbindung zu nicht mehr existierenden Clients aufrechterhalten werden muss [27].

2.4. Schutzziele von Informationssicherheit

Informationen beziehungsweise Daten sind zu schützende Güter informationssicherer Systeme. Der Zugriff auf diese ist zu beschränken und zu kontrollieren, so dass nur dazu autorisierten Subjekten ein Zugriff gewährt wird. Da die Daten über HTTP-Verbindungen im Klartext übertragen werden, müssen für eine erweiterte Sicherheit bestimmte anzustrebende Schutzziele definiert werden. Diese Schutzziele sind im einzelnen [6]:

Integrität: Ein System gewährleistet die Integrität von Daten, wenn es nicht möglich ist, die zu schützenden Daten unautorisiert und unbemerkt zu manipulieren – alle Änderungen sind damit nachvollziehbar.

Authentizität: Unter der Authentizität wird die Echtheit und Glaubwürdigkeit eines Objekts beziehungsweise Subjekts verstanden, die anhand einer eindeutigen Identität überprüfbar ist.

Vertraulichkeit: Ein System gewährleistet die Vertraulichkeit von Informationen, wenn keine unautorisierte Informationsgewinnung möglich ist.

Verbindlichkeit: Ein System gewährleistet die Verbindlichkeit beziehungsweise die Zuordenbarkeit einer Menge von Aktionen oder Daten, wenn es möglich ist, dass einem Subjekt nachgewiesen werden kann diese Aktionen getätigt beziehungsweise Daten angefordert zu haben.

Verfügbarkeit: Ein System gewährleistet die Verfügbarkeit von Daten oder Diensten, wenn authentifizierte und autorisierte Subjekte den Zugriff auf Daten/Dienste innerhalb eines vereinbarten Zeitrahmens erhalten.

2.5. HTTPS

Bei dem HTTPS-Protokoll (Hypertext Transfer Protocol over Secure Socket Layer) handelt es sich um das „klassische“ HTTP-Protokoll mit zusätzlichen Sicherheitsfunktionen. Alle diese Sicherheitsfunktionen werden als SSL/TLS (Secure Socket Layer/Transport Layer Security) bezeichnet und werden im Abschnitt 5.6 genauer beschrieben. Sollen jedoch beide Protokolle zusammengefasst betrachtet werden, wird üblicherweise ein „S“ für „Secure“ dem Protokoll der Anwendungsschicht angehängt (zum Beispiel HTTPS).

Genau wie HTTP wird auch das HTTPS-Protokoll von Browsern verwendet um Inhalte im Internet zu laden und Informationen zu übertragen, zum Beispiel Formulare oder Login-Daten. Der Unterschied zum HTTP-Protokoll liegt in der Verschlüsselung und Authentifizierung der Kommunikation zwischen Web-Server und Client im World Wide Web. Andernfalls werden die übertragenen Daten im Klartext übermittelt und wären somit leicht für Dritte einsehbar. Dies will man besonders bei der Übertragung sensibler Daten wie Login- oder Bankdaten vermeiden.

Die Authentifizierung dient dazu, dass sich jede Seite (Client/Server) der Identität des Verbindungspartners vergewissern können. Dies soll beispielsweise dem Problem von Phishing-Angriffen entgegenwirken, bei dem sich eine gefälschte Webseite als eine andere, wie beispielsweise eine vertrauenswürdige Bankseite, ausgibt.

Die mit HTTP verbundenen Risiken werden mit HTTPS verhindert. Dies ist auf drei zusätzliche Eigenschaften zurückzuführen, die auf der SSL/TLS-Verschlüsselung basieren und nachfolgende Schutzziele erfüllen:

Verschlüsselung: Dank der Verschlüsselung können über das Netz übertragene Informationen nicht von Personen gelesen werden, die nicht über die verwendeten geheimen Schlüssel verfügen.

Integrität: Diese Eigenschaft bietet die Sicherheit, dass die über den sicheren Kanal übertragenen Informationen nicht verändert wurden, weder durch einen technischen Fehler (Hardware oder Software) noch durch die gewollte Handlung eines Dritten (beabsichtigte Modifikation).

Authentifizierung: Die Integrität bietet keine Gewähr hinsichtlich der Quelle der empfangenen Informationen. Mit der Eigenschaft der Authentifizierung wird sichergestellt, dass die empfangene Nachricht tatsächlich von einem Rechner stammt, welcher den privaten Schlüssel besitzt, der dem öffentlichen Schlüssel entspricht, mit dem die Kommunikation eingeleitet wurde. In diesem Fall kann sich der HTTPS-Client darauf verlassen, dass er mit dem richtigen Server kommuniziert. Um die Authentizität des Clients gegenüber dem Server nachzuweisen, muss der Client entsprechend seinen öffentlichen Schlüssel bereitstellen.

Eine Nachweisbarkeit ist mit HTTPS nicht gegeben. Nur die beim Aufbau der SSL/TLS-Sitzung ausgetauschten Daten sind signiert. Bei den übrigen ausgetauschten Daten ist dies nicht der Fall, hier besteht ein Sicherheitsrisiko [4].

2.6. SSL/TLS

Das SSL (Secure Socket Layer)-Protokoll wurde ursprünglich von der Firma Netscape für dessen Web-Browser entwickelt. Die letzte Version von SSL 3.0 wurde für die Entwicklung und Standardisierung von TLS (Transport Layer Security) Version 1 erweitert [6].

TLS und SSL sind hybride Verschlüsselungsprotokolle zur Datenübertragung im Internet. Eine hybride Verschlüsselung bezeichnet dabei die Kombination von symmetrischen und asymmetrischen Kryptoverfahren.

TLS 1.0, 1.1 und 1.2 sind die standardisierten Weiterentwicklungen von SSL 3.0 (TLS 1.0 steht neu für SSL 3.1). SSL wird also nun unter dem Namen TLS weiterentwickelt. Hier wird die Abkürzung SSL für beide Bezeichnungen verwendet [26].

Die grundsätzliche Funktion von SSL ist folgende: Es vermittelt den kryptographischen Algorithmus und die Sitzungsschlüssel für beide Kommunikationspartner und erstellt einen verschlüsselten Tunnel in dem andere Protokolle ihre Daten (wie HTML) transportieren können. Zusätzlich kann SSL die Authentifizierung beider Seiten durch Zertifikate sicherstellen.

Das SSL Protokoll besteht aus folgenden vier Subprotokollen, auf die hier nur kurz eingegangen wird [12]:

- SSL Handshake Protocol
- SSL Change Cipher Spec Protocol
- SSL Alert Protocol
- SSL Record Layer

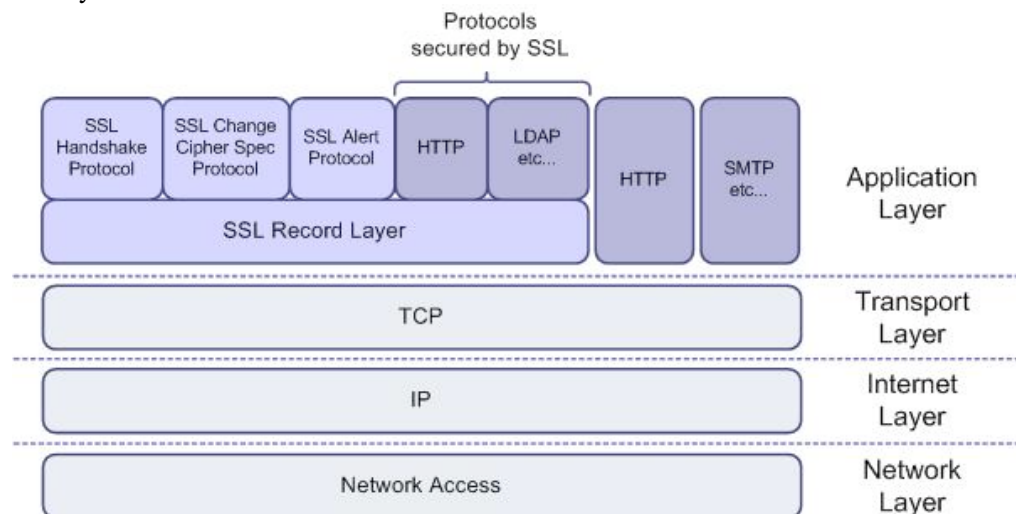


Abbildung 2: SSL im verkürzten OSI -Modell [12]

Im verkürzten OSI-Modell (Abb. 2) ist SSL oberhalb der Transportschicht (TCP) und unter den Anwendungsprotokollen wie HTTP angesiedelt. Deswegen wird dies in den Spezifikationen auch häufig als „HTTP over SSL“ bezeichnet. Dieses Design hat den Vorteil, dass SSL unabhängig von dem verwendeten Applikationsprotokoll eine sichere Punkt-zu-Punkt Verbindung herstellen kann [26].

Der Prozess eine neue SSL Verbindung aufzubauen, beginnt mit dem Austausch der Verschlüsselungsparameter und dem optionalen Authentifizieren des Servers, welches mit dem Handshake-Protokoll durchgeführt wird. Wenn das Handshake-Protokoll erfolgreich war und beide Seiten sich auf einen Verschlüsselungsalgorithmus sowie Schlüssel einigen konnten, können die Daten durch den verschlüsselten Tunnel übertragen werden.

Abbildung 3 zeigt den generellen Ablauf eines Verbindungsaufbaus mit einem HTTPS Server.

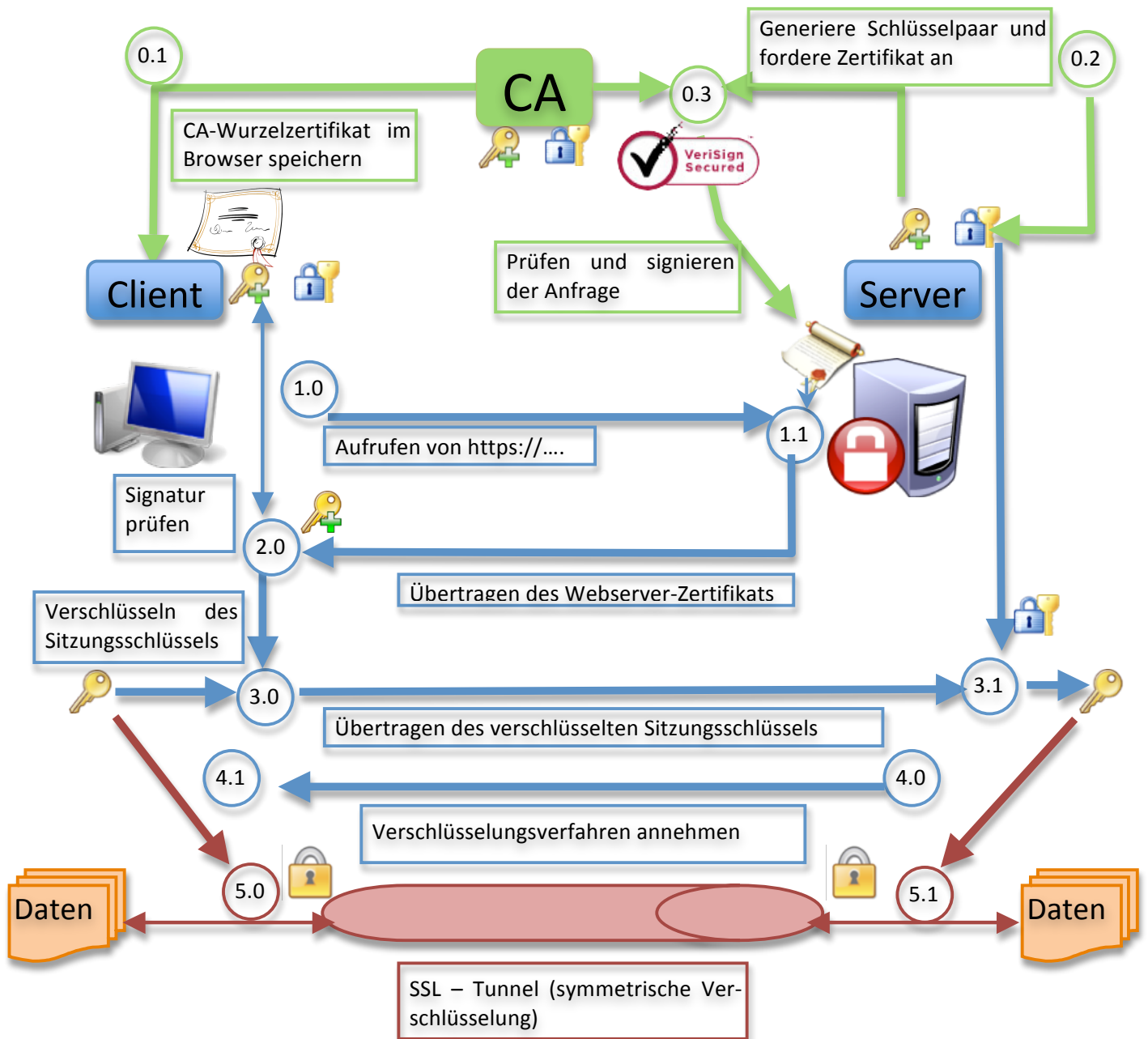


Abbildung 3: HTTPS Verbindungsaufbau – Modell [18]

Die Funktionsweise von SSL ist in Abbildung 3 mit blauen und roten Pfeilen dargestellt. Das Handshake Verfahren ist in blau abgebildet und wird im nachfolgenden Unterkapitel beschrieben. Die grün dargestellten Abläufe beschreiben die Erstellung der Zertifikate, auf die in den darauffolgenden Abschnitten eingegangen wird. Der rot eingefärbte Abschnitt zeigt den Aufbau des SSL-Tunnels und der verschlüsselten Verbindung.

2.7. SSL Handshake-Protokoll

Das Handshake-Protokoll kann in vier Phasen eingeteilt werden und beginnt wie in Abbildung 3 zu sehen mit dem Punkt 1.0 [6,8].

Phase	Beschreibung
Phase 1:	<p>Mit dem Aufrufen einer HTTPS-Seite (Abb. 3 – Punkt 1.0) baut der Web-Browser eine Verbindung zum Server auf, indem er eine „ClientHello-Nachricht“ absendet. Mit dieser Nachricht werden bereits bestimmte Informationen ausgetauscht, die in späteren Schritten zur Berechnung der gemeinsamen Geheimnisse erforderlich sind. Dazu gehört:</p> <ul style="list-style-type: none">• die höchste vom Client unterstützte SSL-Protokoll-Version,• ein 32-Bit Zeitstempel,• eine 32 Byte lange Zufallszahl (die später verwendet wird, um das Pre-Master Secret zu bilden),• einen Sitzungsidentifikator,• und eine Prioritätsliste mit denjenigen kryptographischen und Kompressionsverfahren, die der Client unterstützt (Cipher Suite). Dazu zählt beispielsweise RC4-128 Bit, MD5-MAC und andere, auf die später eingegangen wird.
Phase 2:	<p>Der Server antwortet seinerseits mit einer „ServerHello-Nachricht“, die ebenfalls einen Zeitstempel, eine Zufallszahl sowie einen gültigen Sitzungsschlüssel enthält (Abb. 3 – 1.1). Weiterhin wählt der Server aus der Liste der kryptographischen Verfahren des Clients das Verfahren aus, welches er unterstützt und die höchste Priorität besitzt.</p> <p>Soll der Server authentifiziert werden, sendet er in diesem Schritt sein X.509 Zertifikat mit. Falls er kein Zertifikat besitzt, schickt der Server einen temporären öffentlichen RSA-Schlüssel. Ebenso kann er in diesem Schritt um eine Authentifikation des Clients anfordern.</p>
Phase 3:	<p>Falls ein Client-Zertifikat gefordert wurde, sendet der Client sein signiertes Zertifikat und überprüft die Signatur des Server-Zertifikats anhand des gespeicherten Wurzelzertifikats (Abb. 3 – 2.0).</p> <p>Wird stattdessen ein RSA-Schlüssel verwendet, so verschlüsselt der Client sein generiertes Pre-Master-Secret mit dem öffentlichen Schlüssel des Servers (Abb. 3 – 3.0), sodass nur der Server mit seinem privaten Schlüssel es entschlüsseln kann (3.1).</p> <p>Mit dem Pre-Master-Secret und den in den „Hello-Nachrichten“ ausgetauschten Zufallszahlen, wird anschließend das 48-Byte Master-Secret (Sitzungsschlüssel) berechnet. Dieses wird mit einer Change-Cipher-Spec Nachricht an den Server übermittelt (Abb. 3 – 3.0).</p>
Phase 4:	<p>Mit der Change-Cipher-Spec Nachricht zeigt auch der Server an, dass er ab jetzt die ausgetauschten kryptographischen Verfahren verwendet (Abb. 3 – 4.0). Mit der Client/Server-Finish Nachricht signalisieren beide die erfolgreiche Beendigung ihres jeweiligen Anteils am Protokoll – danach beginnt die Datenübertragung über den SSL Tunnel (Abb. 3 – 5.0).</p>

2.8. Public Key Infrastructure

Bevor eine sichere Kommunikation möglich ist, müssen die Voraussetzungen für eine Vertrauensbeziehung zwischen den Akteuren (Client/Server) geschaffen werden. Ein solches Vertrauensmodell wird als "Public Key Infrastructure" (PKI) bezeichnet. Eine PKI dient dabei der Erzeugung, Prüfung und Verwaltung von Zertifikaten. Im Bereich der asymmetrischen Kryptosysteme hat sich für die Gesamtheit der Komponenten, die hierfür benötigt werden, der Begriff PKI eingebürgert. Abbildung 3 gibt einen Überblick über die Komponenten einer PKI [4,6].

Die zentrale Idee asymmetrischer Verfahren besteht darin, dass jeder Kommunikationspartner ein Schlüsselpaar bestehend aus seinem privaten (geheimen) Schlüssel und einem öffentlichen bekanntzugebenden Schlüssel besitzt. So wird ein Klartext vom Sender mit dem öffentlichen Schlüssel des Empfängers verschlüsselt und der Empfänger entschlüsselt den Kryptotext mit seinem privaten Schlüssel. Damit muss kein geheimer Schlüssel, wie bei einem symmetrischen Kryptosystem, über eine unsichere Datenleitung ausgetauscht werden.

Da die Verschlüsselung unter Verwendung des öffentlichen Schlüssels des Partners erfolgt, muss die Authentizität dieses Schlüssels gewährleistet sein, damit ein Angreifer nicht seinen eigenen Schlüssel als den öffentlichen Schlüssel eines anderen ausgeben und ein Opfer dazu verleiten kann, sensible Daten damit zu verschlüsseln. Zur Sicherstellung und Kontrolle der Schlüsselauthentizität werden Zertifikate eingesetzt.

Bei einem typischen PKI-Infrastrukturtyp verwaltet eine Zertifizierungsstelle (Abb. 3 – CA) die verschiedenen privaten und öffentlichen Schlüsselsätze, die den Akteuren zugewiesen werden (Abb. 3 – 0.3). Die öffentlichen Schlüssel werden zusammen mit der Identifikation des Besitzers in Dokumente eingebunden, die von der Zertifizierungsstelle signiert und als Zertifikate bezeichnet werden. Jeder Akteur hat die Möglichkeit zu prüfen, ob die Signatur eines Zertifikats mit dem öffentlichen Schlüssel der Zertifizierungsstelle, die die Signatur erstellt hat, übereinstimmt (Abb. 3 – 0.1). So kann zum Beispiel ein Client die Echtheit des von einem Web-Server gesendeten Zertifikats prüfen, solange der Client das erforderliche Wurzelzertifikat (siehe Abschnitt 5.8.1) besitzt. Gleiches gilt für die umgekehrte Richtung: Ein Web-Server, der das Zertifikat eines Client empfängt, kann dessen Echtheit feststellen, indem er die Signatur der Zertifizierungsstelle überprüft [4,6].

2.9. Zertifikate

Digitale Zertifikate sind im eigentlichen Sinne Dateien, welche eine digitale Bescheinigung über die Zuordnung des öffentlichen Schlüssels zu einer Identität – dies kann eine Person, eine Organisation oder ein IT-System sein – ausstellen. Die Struktur und der Aufbau derzeit eingesetzter Zertifikate wird durch den X.509 Standard festgelegt und wird nachfolgend genauer beschrieben [6].

Grundsätzlich enthält ein Zertifikat die Angaben zum Eigentümer des Zertifikats, den Aussteller, Start- und Ablaufdatum der Gültigkeit, die verwendeten kryptographischen Algorithmen sowie für welchen Verwendungszweck es benutzt werden darf.

Durch Zertifikate werden die Schutzziele Vertraulichkeit, Authentizität und Integrität von IT-Sicherheit gesichert.

Die von HTTPS verwendeten Zertifikate basieren auf dem Standard X.509 Version 3. Ein solches Zertifikat enthält die folgenden Informationen [6]:

Inhalt	Erläuterung
Versionsnummer	Beschreibt das verwendete X.509 Zertifikatsformat / Version
Seriennummer	Eindeutiger Identifikator
Signatur Information	Verwendete Algorithmen und Parameter
Zertifikatsaussteller	Name der ausstellenden Instanz
Gültigkeitsdauer	Angabe des Zeitintervalls in dem das Zertifikat gültig ist
Benutzername	Eindeutiger Name des Benutzers/Besitzers
Schlüsselinformationen	Öffentlicher Schlüssel und Signieralgorithmus
Eindeutiger Identifikator	In Version v2, v3
Erweiterungen	In Version v2, v3 (Geltungsbereich, Verwendungszweck)

Damit ein von HTTPS verwendetes Zertifikat als gültig betrachtet werden kann, müssen verschiedene Kriterien erfüllt sein [1,4]:

- die URL des in dem Zertifikat eingetragenen Web-Servers muss mit der tatsächlich aufgerufenen URL übereinstimmen,
- das Tagesdatum muss zwischen dem Datum, ab dem das Zertifikat gültig ist, und dem Datum, an dem es seine Gültigkeit verliert, liegen,
- das Zertifikat darf nicht auf einer Zertifikatssperrliste (CRL – Abschnitt 5.8.2) stehen,
- die Signatur des Ausstellers muss gültig sein. Hierfür überprüft der Web-Browser das Zertifikat des Ausstellers - siehe Abbildung 3 Punkt 2.0.

Zusammenfassend werden Zertifikate benötigt um die Echtheit der Identität des Servers/Clients, mit dem kommuniziert wird, nachzuweisen. Ähnlich wie bei der Vorlage eines Ausweises, übernimmt in diesem Fall das Zertifikat die Funktion des Ausweises.

Damit der Browser das Zertifikat eines bestimmten Web-Servers prüfen kann, benötigt er das Zertifikat der nächsthöheren Zertifizierungsstelle, beziehungsweise der höchsten Zertifizierungsstelle (die sogenannten Root-CA). Dieses Verfahren wird als Zertifizierungshierarchie bezeichnet, an dessen oberem Ende das Wurzelzertifikat (siehe Abschnitt 5.8.1) der Root-CA steht.

Eine Zertifizierungshierarchie beschreibt dabei eine Kette von Zertifikaten, die immer einem nächsthöheren Zertifikat bis hin zum Wurzelzertifikat vertrauen (Abb. 4). Dies bedeutet, dass ein Zertifikat eines Webservers wiederum anhand eines übergeordneten Herausgebers des Zertifikates überprüfbar ist. So sind alle Zertifikate wiederum von dessen übergeordnetem bis hin zur "Spitze" – dem Wurzelzertifikat, welchem a priori vertraut werden muss – überprüfbar, um die Gültigkeit eines Zertifikats zu verifizieren. Dies ist vergleichbar mit einem Personalausweis, der von der Behörde einer Stadt ausgestellt wird, die wiederum gegenüber dem Land und dann dem Staat als vertrauenswürdig angesehen wird [24].

2.9.1. Wurzelzertifikate

Web-Browser sind üblicherweise mit den Zertifikaten der gängigsten Aussteller (Zertifizierungsstellen - CA) ausgestattet (Abb. 3 – 01.). Diese Zertifikate werden auch Wurzelzertifikate (Root-Zertifikat oder Stammzertifikat) genannt und dienen dazu die Gültigkeit aller darunterliegender Zertifikate zu validieren.

Da nicht alle Zertifikate vorinstalliert werden können, müssen in einigen Fällen diese Zertifikate manuell nachinstalliert werden. So zum Beispiel fehlt die "Root CA" der Technischen Universität Darmstadt.

Wird eine sichere Webseite beispielsweise einer Bank aufgerufen, deren Zertifikat gegenüber einer im Browser eingetragenen "Root CA" verifiziert werden kann, akzeptiert der Browser diese Webseite als legitim.

Wird hingegen beispielsweise die sichere Webseite "https://www.tu-darmstadt.de/" aufgerufen, so versucht der Browser das Zertifikat der Domain "www.tu-darmstadt.de" mit einem bereits vom Benutzer akzeptierten oder installiertem Zertifikat zu vergleichen. Findet der Browser dieses nicht, so schaut er nach, ob er von der ausstellenden Zertifizierungsstelle für die Adresse "www.tu-darmstadt.de" - in diesem Falle die "TUD CA G01" - das Zertifikat besitzt. Ist dies auch nicht der Fall, so geht er weiter nach "oben" zum DFN-Verein PCA Global-G01 und letztendlich zur Root-CA der Deutschen Telekom (siehe Abb. 4) [19].



Abbildung 4: Zertifikathierarchie der TU Darmstadt Web-Seite

Findet er keines der Zertifikate, so zeigt der Browser einen Warnhinweis (Abb. 5).

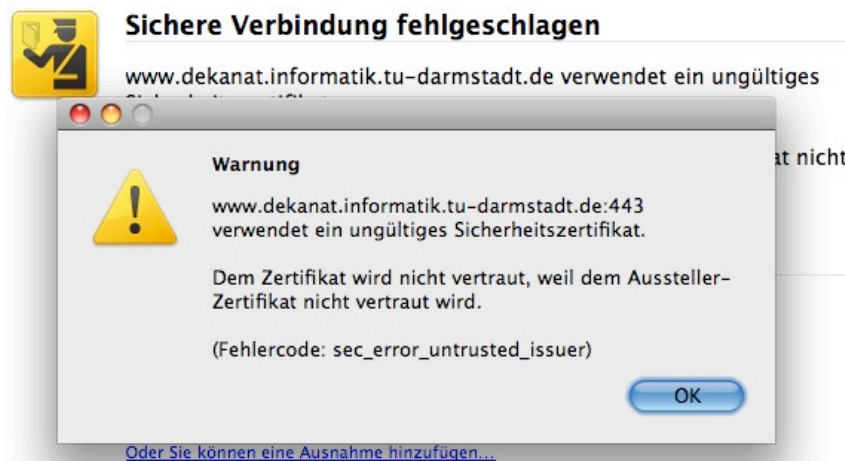


Abbildung 5: Warnhinweis des Firefox Browsers, falls das Zertifikat unbekannt ist

Der Warnhinweis in Abbildung 5 bedeutet lediglich, dass der Web-Browser das Zertifikat nicht kennt und der Benutzer sich sicher sein sollte, dass er die richtige Seite aufgerufen hat.

Der Nutzer kann nun das Zertifikat entweder temporär oder permanent akzeptieren. Wer sicher sein will, dass er es wirklich mit dem richtigen Server zu tun hat, muss sich das Zertifikat eigenständig in seinen Browser installieren und die Zertifikatsdetails selbst überprüfen.

Sobald das Zertifikat einmal installiert wurde, prüft der Browser bei jedem Zugriff auf der Gegenseite (Server), ob diese noch gültig und richtig ist. Sollte jemand versuchen die Adresse des Web-Servers zu fälschen oder einen anderen Server "unterzuschieben", würde der Browser dies anhand der bei ihm lokal abgelegten Zertifikate erkennen und einen entsprechenden Warnhinweis anzeigen.

Wird das Zertifikat nicht akzeptiert oder für nicht vertrauenswürdig gehalten, wird der Vorgang vom Browser unterbrochen und die Webseite nicht angezeigt. Beispiele dazu werden in Kapitel 3 beschrieben.

2.9.2. Zertifizierungsstelle und Registrierungsstelle

Eine Zertifizierungsstelle (CA – Certification Authority) bietet Dienste zur Ausstellung von Zertifikaten. Wie in Abbildung 3 im Punkt 0.3 zu sehen ist, signiert eine CA eine Zertifikatsanfrage – und damit den öffentlichen Schlüssel des Antragsstellers - mit dem privaten Schlüssel der CA. Dadurch versichert die CA die Echtheit des öffentlichen Schlüssels und die Identität des Antragsstellers. Zusätzlich ist die CA dafür zuständig, eine Zertifikatssperrliste von ungültigen Zertifikaten (Certificate Revocation List – CRL) zu führen. Diese Liste wird bei jeder Gültigkeitsüberprüfung eines Zertifikats abgefragt.

Die Identifikation erfolgt durch eine Registrierungsstelle (RA – Registration Authority).

Eine Registrierungsstelle stellt eine direkte Verbindung zwischen einem Objekt (beispielsweise einer Person oder einem Server) und einem für sie ausgestelltem Zertifikat her. Dazu nimmt sie Daten, die dieses Objekt klassifizieren, auf. Bei Zertifikaten, die auf Personen ausgestellt werden, ist zum Beispiel die Prüfung eines amtlichen Dokuments mit Lichtbild möglich um sicherzugehen, dass das Zertifikat dieser Person zugeordnet werden kann.

Die Registrierungsstelle sorgt weiterhin dafür, dass die Informationen des Zertifikates mit den Informationen des Objektes, dem dieses Zertifikat gehört, übereinstimmen [24].

2.9.3. Extended Validation SSL Zertifikat

Extended Validation SSL Zertifikate (EV-SSL) sind X.509 SSL-Zertifikate, welche an strengere Vergabekriterien gebunden sind. Dies bezieht sich vor allem auf die detaillierte Überprüfung des Antragsstellers durch die CA. Welches wiederum dem Anwender mehr Sicherheit vor beispielsweise Phishing-Angriffen geben soll.

Die Vergabekriterien sind in den Guidelines for Extended Validation Certificates spezifiziert. Diese Richtlinien werden vom CA/Browser Forum herausgegeben, einem freiwilligen Zusammenschluss von führender CA-Institutionen und Browser-Herstellern [22].

Das Ziel der EV-SSL Zertifikate ist es dem Anwender auf dem ersten Blick zu zeigen, dass es sich um eine sichere Verbindung handelt. Dies wird in den gängigen Browsern durch eine grüne Einfärbung der Adresszeile realisiert (Abb. 44) – der Safari ab Version 3.2 zeigt beispielsweise auf der rechten Seite der Adresszeile den Organisationsnamen in grün an.

Standardmäßig wird EV-SSL im Firefox ab Version 3, im Internet Explorer 7 und im Opera 9.5 unterstützt.

Der Sicherheitszuwachs im Vergleich zu einem normalen SSL-Zertifikat, ist die genauere Prüfung der Identität des Antragsstellers sowie der höhere Preis, welcher den Erwerb des Zertifikats erschwert.

2.10. Digitale Signaturen

Eine digitale Signatur ist ein kryptographisches Verfahren, bei dem zu beliebigen Daten eine Zahl, die digitale Signatur, berechnet wird. Diese Unterschrift soll dabei folgende Schutzziele erfüllen:

Fälschungssicherheit: Niemand kann die Unterschrift nachahmen.

- Keine Wiederverwendbarkeit:** Die Unterschrift kann nicht von einem Dokument auf ein anderes Dokument kopiert werden.
- Unveränderbarkeit:** Nach der Unterzeichnung kann das Dokument nicht mehr geändert werden, beziehungsweise eine Änderung ist erkennbar.
- Zugehörigkeit:** Der Unterzeichner kann später nicht behaupten, dass er das Dokument nicht unterschrieben hat.

Digitale Signaturen basieren auf asymmetrischen Kryptosystemen und verwenden folglich ein Schlüsselpaar, das aus einem privaten (geheimen) und einem öffentlichen (nicht geheimen) Schlüssel besteht. Mit dem privaten Schlüssel werden die Daten signiert, der öffentliche Schlüssel dient anschließend zum Verifizieren der Signatur.

In der Regel werden digitale Signaturen nicht direkt auf die Nachricht angewendet, sondern auf deren Hashwert. Das ermöglicht eine Beschleunigung der Signiervorgangs und verhindert bestimmte Angriffe. Die am häufigsten verwendeten Signaturverfahren sind RSA und DSA, welche in den darauffolgenden Abschnitten erläutert werden.

Abbildung 6 zeigt den grundsätzlichen Ablauf einer Kommunikation mit signierten Nachrichten.

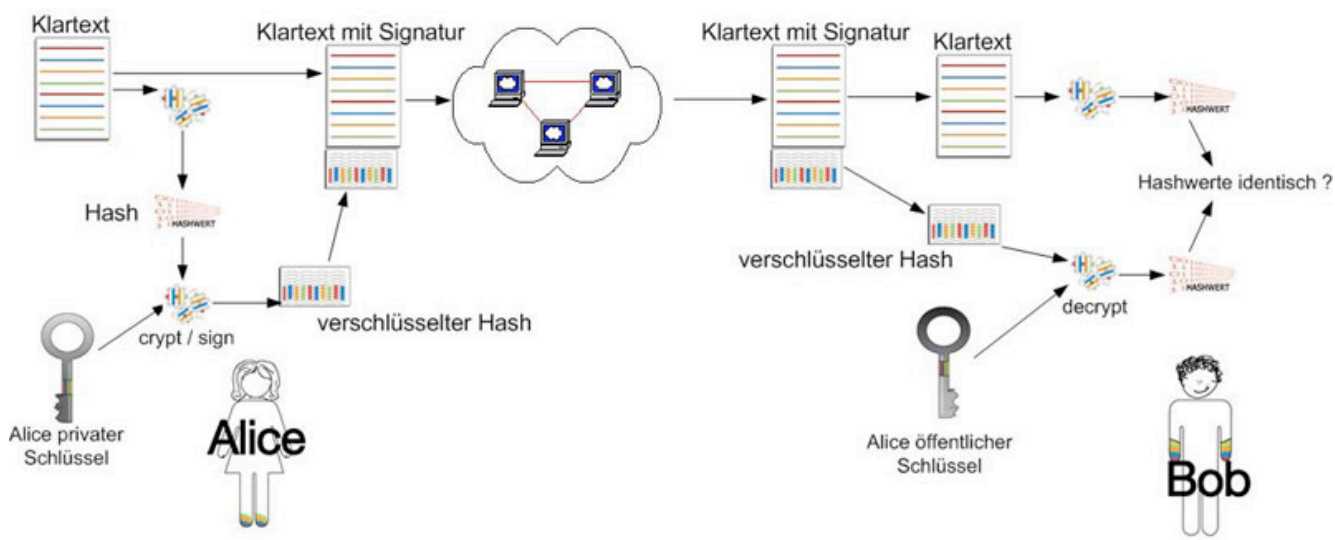


Abbildung 6: Genereller Ablauf signierter Kommunikation [5]

2.10.1. RSA

RSA ist ein asymmetrisches Kryptosystem, das sowohl zur Verschlüsselung, als auch zur digitalen Signatur verwendet werden kann. Es verwendet ein Schlüsselpaar bestehend aus einem privaten Schlüssel, der zum Entschlüsseln oder Signieren von Daten verwendet wird und einem öffentlichen Schlüssel, mit dem man verschlüsselt oder Signaturen prüft.

RSA wurde 1977 von Ronald L. Rivest, Adi Shamir und Leonard Adleman am MIT entwickelt. Die Sicherheit dieses Verfahrens beruht auf dem mathematischen Problem große Zahlen zu faktorisieren.

2.10.2. Digital Signature Algorithm

Der Digital Signature Algorithm (DSA) ist ein Standard der US-Regierung für Digitale Signaturen. Er wurde vom National Institute of Standards and Technology (NIST) im August 1991 für die Verwendung in deren Digital Signature Standard (DSS) empfohlen.

Im Gegensatz zu RSA eignet sich DSA nur zum signieren und nicht zum Verschlüsseln. Die Sicherheit dieses Algorithmus beruht wie RSA auf dem Problem diskreter Logarithmen.

2.11. Hashfunktionen

Sichere Hashfunktionen bilden wichtige Bestandteile heutiger Verschlüsselungsinfrastrukturen. Mittels kryptographisch sicherer Hashfunktionen werden eindeutige, so genannte digitale Fingerabdrücke von Datenobjekten berechnet und zusammen mit dem Objekt versandt, beziehungsweise gespeichert. Damit ist es dem Empfänger oder Objektnutzer möglich, anhand des Fingerabdrucks (Fingerprint) die Integrität oder auch Authentizität des Objekts zu überprüfen [6].

Hashfunktionen werden außerdem in anderen Teilgebieten der Informatik eingesetzt, um Zugriff auf Objekte effizient zu realisieren beziehungsweise ungeordnete Objektnamen zu verwalten.

Eine Hashfunktion definiert einen endlichen Bildbereich, der häufig als Adressbereich oder Ausgabemenge bezeichnet wird. Der Adressbereich ist in der Regel erheblich kleiner als der Urbildbereich (Eingabemenge). Eine Hashfunktion ist eine nicht injektive Abbildung (deterministische Funktion), die jedes Objekt der Eingabemenge mit beliebiger Länge, auf einen Hash mit fester Länge abbildet.

Anwendungsbereiche für Hashfunktionen sind unter anderem schnelle Such-, Sortier- und Zugriffsverfahren im Datenbankbereich oder Adressierungsbereich in Betriebssystemen [6]. Die Geschwindigkeitssteigerung wird durch die Komprimierung der Eingabedaten erreicht: Unterschiedliche Eingaben beliebiger Länge erhalten gleiche Ausgabewerte (Hashwerte) fester Länge. Ein Beispiel ist die Abbildung von Namen mit gleichem Anfangsbuchstaben in einem Telefonbuch.

Da Hashfunktion nicht injektiv sind und ihr Bildbereich meist signifikant kleiner ist, als die abzubilden- den Eingabemenge, können Kollisionen auftreten. Das heißt, dass zwei unterschiedliche Objekte u_1 , u_2 der Eingabemenge auf den gleichen Hashwert abgebildet werden, $h(u_1) = h(u_2)$, $u_1 \neq u_2$.

In den genannten Anwendungsbereichen ist dies allein ein Verwaltungsproblem, das man mit verschiedenen Techniken zur Kollisionsauflösung wie mehrfaches Hashing löst.

Soll jedoch mit einer Hashfunktion ein Wert berechnet werden, der ein Objekt eindeutig charakterisiert und damit die Überprüfung der Integrität von Daten ermöglichen, so muss das Auftreten von Kollisionen vermieden werden [6].

Anders ausgedrückt, während es bei Such- beziehungsweise Sortierverfahren unumgänglich ist, von Hashwerten auf die Eingabe zu schließen, müssen im Kontext von sicheren System sogenannte Einweg-Hashfunktionen eingesetzt werden, wo genau dies unmöglich ist.

Eine Einweg-Hashfunktion besitzt dabei folgende Eigenschaften:

1. Zu einem gegebenen Ausgabewert $h(u_1) = x$ ist es praktisch unmöglich einen Eingabewert u_1 zu finden
2. Es ist praktisch unmöglich für einen gegebenen Wert u_1 ein davon verschiedenes u_2 zu finden, der denselben Hashwert $h(u_1) = h(u_2) = x$ ergibt. Diese Eigenschaft wird auch schwache Kollisionsresistenz genannt.

Für Kollisionsresistente Funktionen wird zusätzlich eine weitere Eigenschaft definiert:

3. Es ist praktisch unmöglich zwei verschiedene Eingabewerte u_1 und u_2 zu finden, die denselben Hashwert ergeben. Diese Eigenschaft wird als starke Kollisionsresistenz bezeichnet.

Beim heutigen Stand der Technik werden Hashfunktionen mit Hashwerten der Länge $k=160$ Bit als hinreichend sicher angesehen. Für eine höhere Sicherheit sollten aber mittlerweile Bitlängen von $k=256$ verwendet werden.

In der heutigen Zeit werden vor allem Hashfunktionen verwendet, bei denen die Kompressionsfunktion speziell für die Erzeugung von Hashwerten konstruiert wurde. Im Rahmen von PKI werden Hashfunktionen beispielsweise dazu verwendet, um einen Hashwert (Fingerprint) über ein Zertifikat zu bilden. Die gebräuchlichsten Algorithmen hierfür sind MD5, RIPEMD-160 und SHA-1, welche im folgenden beschrieben werden.

2.11.1. MD5

Die MD-Verfahren (Message Digest) wurden 1990 von R. Rivest mit dem MD4 eingeführt und mit dem MD5 als eine Weiterentwicklung von MD4 fortgesetzt. Der MD5 liefert einen 128-Bit Hashwert und arbeitet mit 512-Bit Eingabeblocken, welche wiederum in sechzehn 32-Bit Worte aufgeteilt werden [6]. Schon 1996 wurde eine erste Kollisionsfunktion für MD5 gefunden und 2004 für den gesamten Algorithmus [23]. Damit gilt MD5 als nicht Kollisionsresistent und somit als unsicher. MD5 wird heutzutage dennoch sehr häufig eingesetzt, obwohl es nicht mehr als Kollisionsresistent gilt und damit als unsicher eingestuft wurde [11]. Der Einsatz von MD5 wird zum Teil dadurch begründet, dass MD5 historisch gewachsen ist und mit Salt-(Streu) Werten eine Kollisionsgefahr reduziert werden kann.

2.11.2. RIPEMD-160

RIPEMD (Race Integrity Primitives Evaluation Message Digest) wurde für das RIPE-Projekt der EU entwickelt und ist eine Variante von MD4 mit einer Hashwertlänge von ursprünglich 128 Bit. Mittlerweile wurde die Ausgabe des Algorithmus auf 160 Bit erweitert und gleicht hinsichtlich seiner Stärke und Geschwindigkeit dem SHA-1 Algorithmus. Um RIPEMD-160 im Vergleich zu MD4 gegen kryptoanalytische Angriffe resistenter zu machen, wurden die Rotationen und die Reihenfolge der Nachrichtenwörter modifiziert [3].

2.11.3. SHA-1

Der Secure Hash Algorithm, kurz SHA, ist Bestandteil des Secure Hash Standards und wurde 1993 vom amerikanischen National Institute of Standards and Technology (NIST) als Federal Information Processing Standard (FIPS) veröffentlicht. Der SHA erzeugt 160-Bit Hashwerte und verwendet eine Blocklänge von 512 Bit. Durch Hinzufügen einer zusätzlichen 1-Bit Rotation wurde 1995 der SHA-1 veröffentlicht und gilt bis 2010 noch als sichere Hashfunktion [2].

2002 wurden vom NIST drei weitere Varianten des Algorithmus veröffentlicht, die größere Hashwerte erzeugen. Es handelt sich dabei um den SHA-256, SHA-384 und SHA-512, welche auf längere Sicht weit aus mehr Sicherheit bieten [9].

Ende 2008 hat das NIST eine Ausschreibung zum Nachfolger SHA-3 ausgerufen, dessen endgültige Wahl für 2012 vorgesehen ist und den Sicherheitsgrad weiter anheben soll [25].

2.12. Verschlüsselung

Unter Verschlüsselung versteht man den Vorgang einen Klartext mit Hilfe eines Kryptosystems in einen Chiffretext zu transformieren (verschlüsseln) und entsprechend wieder zurück zuführen (entschlüsseln). Das wesentliche Ziel beim Einsatz kryptographischer Verfahren besteht in der Geheimhaltung der in der Nachricht codierten Informationen gegenüber Dritten (Angreifern).

Die eingesetzten Schlüssel zur Ver- und Entschlüsselung können dabei gleich (symmetrisch) oder verschieden (asymmetrisch) sein.

In einem symmetrischen System müssen zwei Kommunikationspartner einen gemeinsamen, geheimen Schlüssel verwenden. Dies bedeutet, dass sie sich vorab über diesen gemeinsamen Schlüssel verständigen müssen. Die Sicherheit einer Kommunikation mittels symmetrischer Verfahren hängt damit nicht nur von der kryptographischen Stärke der verwendeten Verfahren ab, sondern auch von der sicheren Aufbewahrung und Verwendung der geheimen Schlüssel durch die Kommunikationspartner sowie von der sicheren initialen Übermittlung der verwendeten gemeinsamen Schlüssel [6]. Zu den häufig eingesetzten und derzeit sicheren symmetrischen Verschlüsselungsverfahren zählt der Advanced Encryption Standard (AES), auf den später eingegangen wird.

Die Kernidee asymmetrischer Verfahren besteht darin, dass jeder Kommunikationspartner ein Schlüssel-paar bestehend aus einem persönlichen, geheimen Schlüssel und einem öffentlich bekannt zu gebenden Schlüssel besitzt. Der Vorteil beim Einsatz asymmetrischer Schlüssel ist somit, dass die geheimen Schlüssel nicht zuvor über sichere Wege ausgetauscht werden müssen. Zu den bekanntesten asymmetrischen Systemen gehört das RSA-Verfahren.

2.12.1. RC4

RC4 (Ron's Code 4) wurde 1987 von R. Rivest entwickelt und ist eine Stromchiffre, welche beispielsweise in SSL, HTTPS, WEP und WPA eingesetzt wird. Dabei verarbeitet der Algorithmus anders als eine Blockchiffrierung nicht ganze Blöcke, sondern immer ein Bit oder ein Byte. Bei einer RC4 Verschlüsselung wird der Klartext Byte für Byte per XOR mit der zuvor generierten Pseudozufallszahl verknüpft. Zur Erzeugung der Zufallszahlen wird ein geheimer Schlüssel verwendet.

Die Sicherheit eines solchen Verfahrens ist nur solange gewährleistet, wie sich die Zufallsfolge nicht wiederholt. Aus diesem Grund darf der Schlüssel zur Erzeugung der Zufallszahlenfolge nur einmal verwendet werden.

Nach Einstufung des Instituts für Internet-Sicherheit (if-is) gilt eine RC4 128-Bit Verschlüsselung schon heute als Unsicher [11,13].

2.12.2. Data Encryption Standard

Eines der weit verbreiteten symmetrischen Verschlüsselungsalgorithmen ist der Data Encryption Standard (DES). Der DES-Algorithmus wurde 1976 von der US-Regierung zum offiziellen Standard erklärt und wird noch heute eingesetzt. Aufgrund der effektiven 56-Bit Schlüssellänge, gilt der Algorithmus jedoch heute als nicht ausreichend sicher [11]. Durch Mehrfache Anwendung des DES Algorithmus auf einen Datenblock, kann die Schlüssellänge vergrößert werden. So bietet die derzeit eingesetzte dreifache Ausführung mehr Sicherheit und wird als Triple-DES oder 3DES bezeichnet.

2.12.3. Advanced Encryption Standard

Der Advanced Encryption Standard (AES) ist eine symmetrische Blockchiffre mit fester Blocklänge von 128 Bit und variabler Schlüssellänge von 128, 192 oder 256 Bit. AES, auch bekannt als „Rijndael-Algorithmus“, ist der Nachfolger des Data Encryption Standard (DES) und wurde im Oktober 2000 vom NIST unter dem Titel AES als Standard bekanntgegeben.

Jeder Klartextblock wird bei dem Algorithmus in mehrere Runden mit einer sich wiederholenden Abfolge von Funktionen bearbeitet.

Der Einsatzbereich von AES findet sich in SSH, WAP2 (Wi-Fi Protocol Access – Wireless LAN), Win.ZIP (Komprimierungssoftware) sowie der E-Mail-Push Kommunikation von BlackBerries [6].

Wirklich erfolgreiche Angriffe auf die algebraische Struktur, die nicht nur eine theoretische Bedeutung haben, sind bislang nicht bekannt.

3. Aktuelle Situation

In diesem Kapitel soll die Benutzerführung für den Umgang mit gesicherten Verbindungen in den derzeit am häufigsten genutzten Browsern verdeutlicht werden.

Analysiert wird hierbei die Visualisierung sowie der Umgang mit Zertifikaten auf HTTPS Seiten in den verschiedenen Browsern. Herausgearbeitet werden besonders die Schwächen der jeweiligen Darstellungen.

Die Auswahl der untersuchten Browser stützt sich auf die Popularität der Browser weltweit. Microsofts Internet Explorer (MSIE) führt die Statistik der meist genutzten Web-Browser an (65%), an zweiter Stelle kommt Mozillas Firefox (21%), welcher in kürzester Zeit dem Browsergiganten Microsoft viele Marktanteile streitig gemacht hat. Speziell der MSIE wird in den gegenwärtig drei verschiedenen eingesetzten Versionen untersucht. Dadurch soll auf die Veränderungen in den letzten Jahren eingegangen werden.

Die gegenwärtigen Marktanteile gliedern sich folgendermaßen auf:

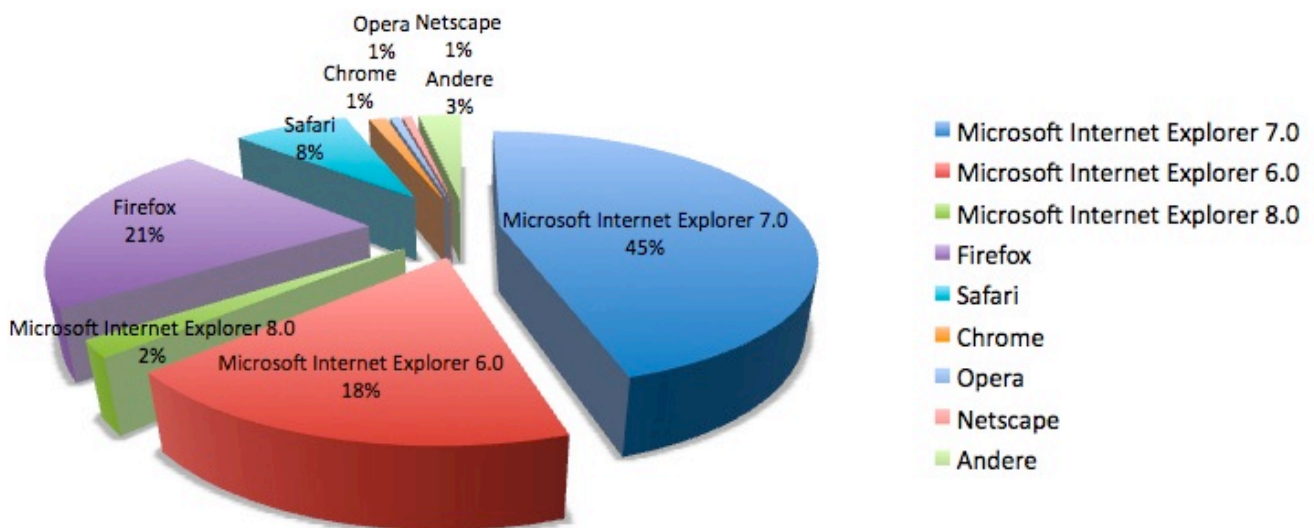


Abbildung 7: Browser Marktanteile März 2009 [16]

Diese Arbeit konzentriert sich auf die Analyse der folgenden am häufig genutzten Browser:

- Microsoft – Internet Explorer (Version 6.0, 7.0, 8.0)
- Mozilla – Firefox (Version 3.0.10, 3.5 Beta)
- Apple – Safari (Version 4 Beta)

In den folgenden Unterkapiteln werden verschiedene Szenarien untersucht, wie die Benutzerführung in den ausgewählten Browsern bei gültigen, sowie aus verschiedenen Gründen ungültigen Zertifikaten, abläuft.

Die hier untersuchten Szenarien für ungültige Zertifikate ergeben sich aus den nachstehenden Fällen:

- Not Trusted CA - Unkown Issuer: Wurzelzertifikat ist unbekannt
- Domain mismatch - Bad Cert Domain: Domaineintrag im Zertifikat stimmt mit URL nicht überein
- Certificate expired - Untrusted Issuer: Zertifikat ist abgelaufen

Kombinationen dieser Fälle oder andere Fehlerarten ergeben keine grundlegend veränderte Benutzerführung und wurden aus diesem Grund nicht weiter betrachtet. Eine Liste möglichen Fehler mit Zertifikaten für den Firefox kann unter [14] eingesehen werden.

Bevor die aufgeführten Szenarien in den verschiedenen Browsern untersucht werden, wird eine kurze Erläuterung der drei unterschiedlichen Szenarien gegeben:

Unkown Issuer: Das Wurzelzertifikat ist unbekannt. Das Zertifikat wurde von einer CA oder dem Betreiber selbst signiert, deren Wurzelzertifikat nicht vertraut wird, da sich das Wurzelzertifikat nicht in der Zertifikatsliste des Browsers befindet.

Eine solche HTTPS Verbindung wird zwar verschlüsselt und ist somit vor Angreifern sicher, jedoch identifiziert das Zertifikat nicht eindeutig den Empfänger/Server und gilt aus Sicherheitsgründen als nicht vertrauenswürdig.

Bad Cert Domain: Diese Fehlermeldung weist darauf hin, dass das empfangene Zertifikat für eine andere URL ausgestellt wurde. Wie oben erläutert, werden die Informationen zwar ebenso verschlüsselt, doch ist der Server (Empfänger) dieser Daten nicht derjenige, welcher im Zertifikat eingetragen ist.

Eine mögliche Ursache dieses Problem könnte jedoch sein, dass das Zertifikat für einen anderen Bereich der selben Seite gilt. Zum Beispiel ist dies der Fall, wenn folgende Seite aufgerufen wird: <https://beispiel.de>, das Zertifikat aber für <https://www.beispiel.de> vorgesehen ist.

Untrusted Issuer: Dieser Fehler tritt auf, wenn der Browser erkennt, dass das ausgestellte Zertifikat nicht mehr gültig ist und das Gültigkeitsdatum überschritten wurde. Des Weiteren taucht dieser Fehler auf, wenn das Zertifikat revoked und seine Gültigkeit zurückgenommen wurde. Das Zertifikat gilt aus diesen Gründen als nicht mehr sicher und nicht vertrauenswürdig.


3.1. Microsoft Internet Explorer 6.0

Microsofts Web-Browser Internet Explorer 6.0 (MSIE 6) erschien 2001 und wurde Bestandteil der Standardinstallation des Betriebssystems Microsoft Windows XP. Die Integration eines Browsers in das weltweit verbreitete Betriebssystem, brachte Microsoft seit Jahren Marktanteile von über 80%. Erstmals seit 2009 sind die Marktanteile des MSIE unter die 70% Marke gesunken.

Obwohl im Jahr 2006 die neuere Version MSIE 7 erschienen ist, nutzen etwa 18% der Benutzer weiterhin die alte Version des Browsers.

Aus diesem Grund ist es noch heute interessant die Funktionsweise dieses mittlerweile veralteten Browsers zu analysieren.

3.1.1. Analyse gesicherter Verbindungen mit gültigem Zertifikat

Wird beim MSIE 6 eine Verbindung zu einer HTTPS-Seite mit gültigen Zertifikat aufgebaut, erscheint in der Statuszeile des Browsers ein kleines gelbes Schloss-Symbol  - dieses symbolisiert, dass es sich um eine gesicherte Verbindung handelt. Per Doppelklick auf dieses Schloss können weitere Informationen zum eingesetzten Zertifikat dieser Seite angezeigt werden.

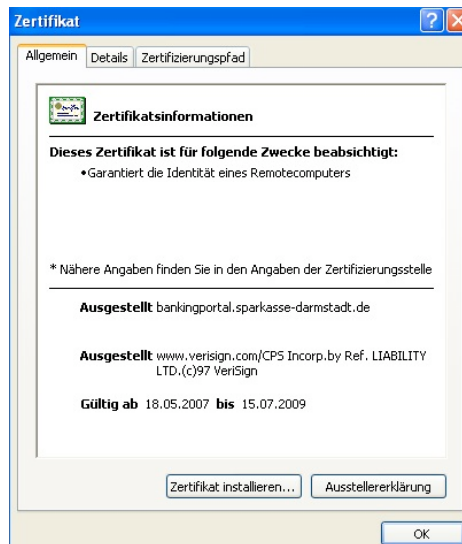


Abbildung 8: IE6 - Allgemeinen Zertifikatsdetails

Abbildung 8 zeigt die allgemeinen Zertifikatsinformationen wie sie im MSIE 6 dargestellt werden. Zu sehen ist der Namen des Besitzers, das Ausstellungsunternehmen und die Gültigkeit dieses Zertifikats. Unter dem Reiter „Details“ können weitere Informationen, wie beispielsweise der Fingerprint, Verschlüsselungsalgorithmus oder das Verschlüsselungsverfahren des öffentlichen Schlüssels angezeigt werden (Abb. 9).

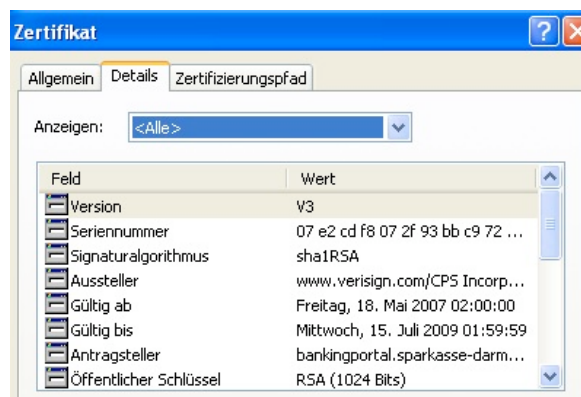


Abbildung 9: IE6 - Zertifikatsdetails

Der Nutzer erhält somit Details zum Zertifikat selbst, kann aber ohne kryptographisches Hintergrundwissen keine Aussage darüber machen, wie sicher das Zertifikat wirklich ist. Beispielsweise könnte der Verschlüsselungsalgorithmus veraltet oder zu schwach sein und eine leichte Angriffsmöglichkeit bieten. Hier erkennt man deutlich eine Verbesserungsmöglichkeit mit einer Klassifizierung einer Sicherheitsstufe des Zertifikats, die im Kapitel Konzept aufgegriffen wird.

Unter dem Reiter „Zertifizierungspfad“ wird der Pfad des Zertifikats bis zu seinem Wurzelzertifikat angezeigt (Abb. 10).

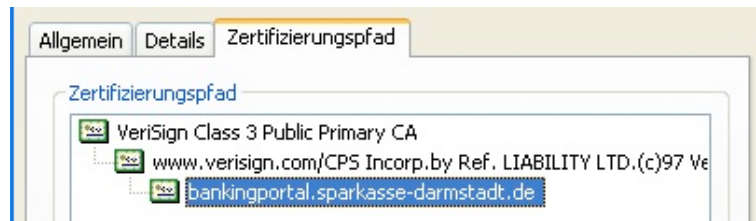


Abbildung 10: IE6 - Zertifizierungspfad bei gültigem Zertifikat

3.1.2. Analyse gesicherter Verbindungen mit ungültigem Zertifikat

Szenario 1 – Unknown Issuer: Wird eine HTTPS Verbindung zu einer Seite aufgebaut, deren Zertifikat nicht den Gültigkeitsrichtlinien entspricht, wird ein Pop-Up Fenster mit einer Fehlermeldung der entsprechend Richtlinienverletzung angezeigt, siehe Abbildung 11.

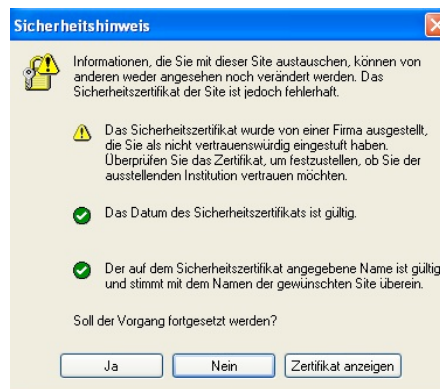


Abbildung 11: IE6 - Sicherheitshinweise, wenn dem Zertifikat des Servers nicht vertraut wird

Der in Abbildung 11 dargestellte Fall kann das Zertifikat nicht bis zu einem vertrautem Wurzelzertifikat zurückverfolgt werden. Entsprechend zeigt der Zertifizierungspfad in Abbildung 12 nur einen Eintrag.

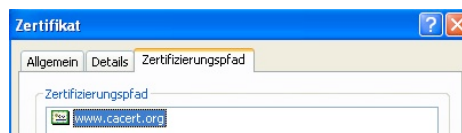


Abbildung 12: IE6 - Zertifizierungspfad bei unbekanntem Wurzelzertifikat

Der Benutzer kann nun entscheiden, ob er diese Webseite trotzdem aufrufen möchte, oder ob er aufgrund der Fehlermeldung den Vorgang abbricht. Wird die Option „Zertifikat anzeigen“ gewählt, kann der Nutzer sich das vermeintlich unsichere Zertifikat anschauen.

In den allgemeinen Zertifikatsinformationen wird dem Benutzer mit einem roten „X-Icon“ angezeigt, dass das Zertifikat nicht verifiziert werden konnte (Abb. 13).

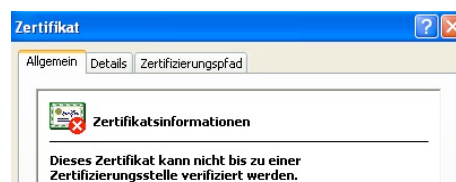
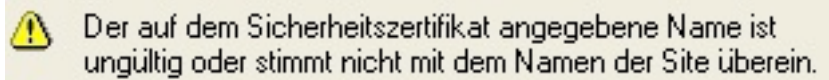


Abbildung 13: IE6 - Zertifikatsstelle unbekannt

Der Warnmeldung des Pop-Up Fensters (Abb. 11) und der Sicherheitshinweis in den Zertifikatsdetails (Abb. 12), sind die einzigen Anhaltspunkte für Benutzer zur Deutung eines Fehlers dieses Zertifikats. Weitere Informationen zur Problematik des Fehlers kann der Benutzer an dieser Stelle nicht erhalten um den Fehler genauer zu ergründen. Eine Verbesserung der mangelnden Fehlerquellenanalyse ist ebenso ein Teil dieser Arbeit und wird im Kapitel 4 behandelt.

Szenario 2 – Bad Cert Domain: Wird eine HTTPS-Seite aufgerufen, die ein Zertifikat zurücksendet, in dem die URL des Zertifikatsbesitzer nicht mit der aufgerufenen URL übereinstimmt, gibt der IE 6 folgende Fehlermeldung in dem Pop-up Fenster zurück:



Werden die Zertifikatsdetails dieser Seite aufgerufen, wird dem Benutzer nicht verdeutlicht, an welcher Stelle sich der Fehler befindet. Erfüllt das Zertifikat somit alle anderen Kriterien, erscheint das Zertifikat unter den allgemein Zertifikatsdetails sowie im Zertifizierungspfad, wie in Abbildung 15 gezeigt, als „korrekt“ und somit gültig, obwohl sich der Name der aufgerufenen URL von dem ausgestellten Namen im Zertifikat unterscheidet.

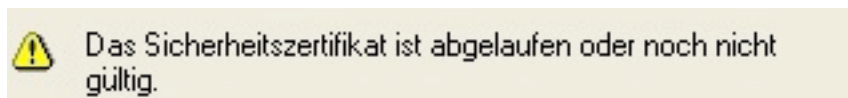


Abbildung 14: IE6 - Scheinbar gültiges Zertifikat

Weil das Zertifikats an sich korrekt ist, nur nicht zu der aufgerufenen URL passt, kann der Benutzer hier fälschlicherweise zu dem Schluss kommen, dass er sich auf einer sicheren Seite befindet und die Pop-up Fehlermeldung anscheinend nicht korrekt interpretiert hat.

Dieses Szenario bedarf einer Verbesserung und muss dem Benutzer deutlich anzeigen, dass das Zertifikat an sich gültig ist, jedoch nicht als korrekt für die angeforderte Seite ist und das Sicherheitsrisiko somit vergleichbar mit dem Fall, dass kein Zertifikat vorgelegt wurde.

Szenario 3 – Untrusted Issuer: Eine HTTPS-Seite mit abgelaufenem Zertifikat wird mit folgender Warnmeldung angezeigt:



Werden die Zertifikatsdetails geöffnet, kann der Benutzer durch die Fehlermeldung und dem roten „X-Icon“ erkennen, das es sich um ein abgelaufenes Zertifikat handelt (Abb. 15).

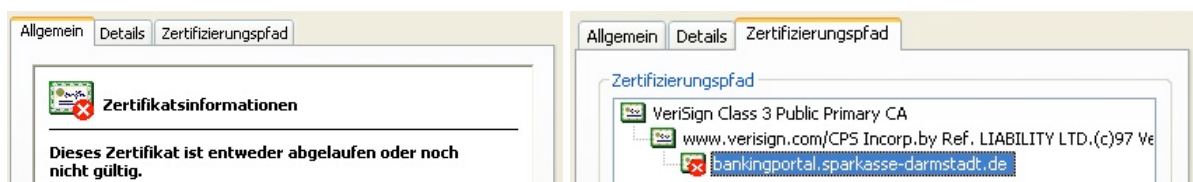


Abbildung 15: IE6 - Zertifikat abgelaufen (Allg. Informationen links / Zertifizierungspfad rechts)

Kombinationen der verschiedenen Szenarien werden auf die gleiche Weise behandelt und zeigen die entsprechenden Fehlermeldungen an.

3.2. Microsoft Internet Explorer 7.0

Wie bereits beschrieben, erschien der MSIE 7 im Jahr 2006 und brachte einige Neuerungen im Umgang mit gesicherten Verbindungen. Dazu gehört die Anzeige der EV-SSL Zertifikate, welche die Adresszeile grün einfärben, sowie eine rot gefärbte Adresszeile bei nicht vertrauenswürdigen Zertifikaten. Zugleich wanderte das Schloss-Symbol in die Adresszeile, mit dem sich der Benutzer durch einen Klick Informationen zu der aktuellen HTTPS-Seite anschauen kann. Außerdem wurde das Pop-Up Fenster, welches eine Warnung über nicht vertraute Zertifikate anzeigte, durch eine Fehlerseite direkt im Browserfenster ersetzt (Abb. 19).

In den nächsten Abschnitten werden die Neuerungen vorgestellt.

3.2.1. Analyse gesicherter Verbindungen mit gültigem Zertifikat

Wird eine gesicherte Verbindung aufgebaut, zeigt der MSIE 7 ein Sicherheitsschloss-Symbol in der Adresszeile (siehe Abb. 16).



Abbildung 16: IE7 - Adresszeile mit neuem Sicherheitsschloss-Symbol

Durch Anklicken des Schlosses erhält der Benutzer allgemeine Informationen zum Zertifikat der HTTPS-Seite (Abb. 17).

Der oberste Eintrag „Websiteidentifizierung“ zeigt dem Benutzer, ob es sich um eine EV-SSL zertifizierte Seite handelt (grünes Häkchen) oder ohne EV-SSL Zertifikat (blaues Fragezeichen). Handelt es sich um eine EV-SSL Zertifikat, werden die zum Betreiber der Seite relevanten Adressinformationen veröffentlicht.

Weiterhin wird die CA angezeigt, welche das Zertifikat signiert hat, gefolgt vom Besitzer des Zertifikats.

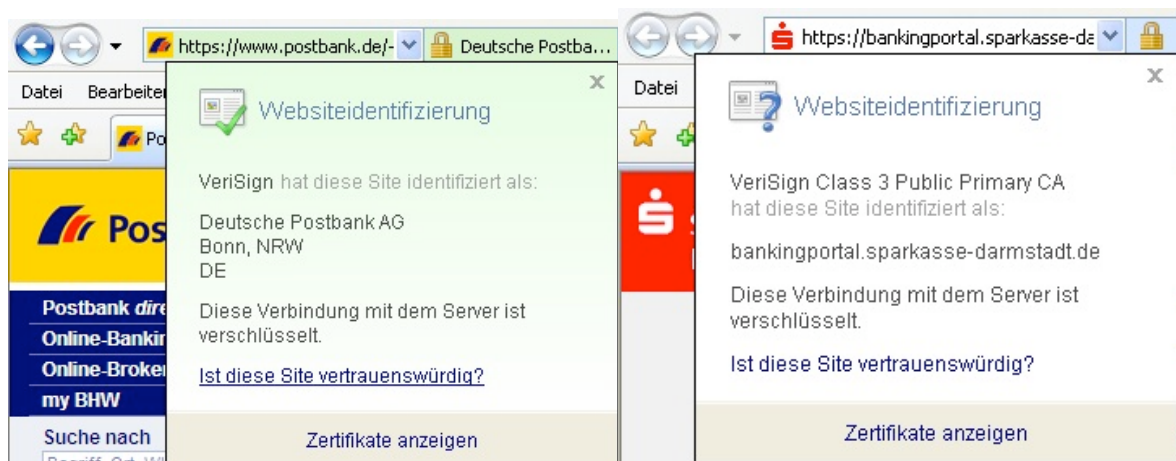


Abbildung 17: IE7 - Zertifikats Kurzinfo-Fenster – Links mit EV-SSL, rechts ohne

Mit dem Link „Ist diese Seite vertrauenswürdig?“ gelangt der Benutzer zu einem Microsoft Windows Hilfe Menü und kann sich allgemeine Informationen zu verschiedenen Sicherheitsfragen, die sich mit dem Thema Zertifikate beschäftigen, durchlesen.

Der Nachteil dieses allgemein gehaltenen Hilfetextes ist, dass nicht auf die entsprechenden Fehler direkt eingegangen wird. Der Benutzer muss sich die Informationen zu den auftretenden Fehlern eigenständig herausuchen. Für unerfahrene Benutzer erschwert dies die Suche nach einer Lösung.

Der unterste Link „Zertifikat anzeigen“, öffnet ein Fenster mit allgemeinen Informationen zum Zertifikat, welches sich in seiner Darstellungsform nicht von der Version 6 unterscheidet.

3.2.2. Analyse gesicherter Verbindungen mit ungültigem Zertifikat

Szenario 1 – Unknown Issuer: Wird eine HTTPS Seite aufgerufen, deren Zertifikat mit keinem Wurzelzertifikat des Browsers verifiziert werden kann, färbt sich die Adresszeile rot und das Schloss-Symbol wird durch ein rotes Schild-Symbol ersetzt. Zusätzlich wird ein Text „Zertifikatsfehler“ angezeigt, der durch einen Klick weitere Informationen zum Fehler bietet (Abb. 18).

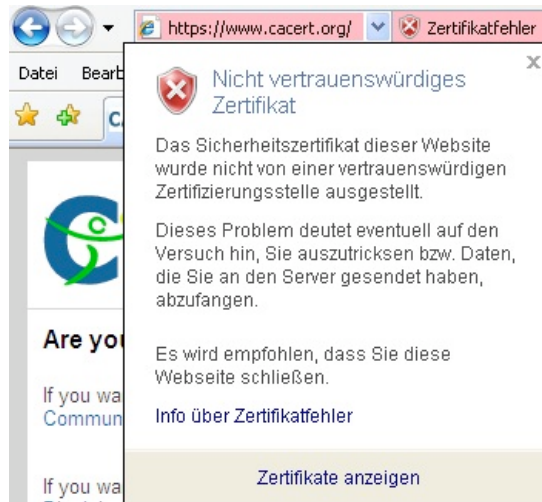


Abbildung 18: IE7 - Erweiterte Zertifikatsfehler Informationen

Das Informationsfenster präsentiert dem Benutzer im oberen Teil des Fensters, welches Problem aufgetreten ist und enthält zudem den Hinweis, dass diese Seite den Benutzer „austricksen“ möchte und deswegen geschlossen werden sollte.

Diese Formulierung ist relativ allgemein gehalten sowie sehr unglücklich ausgedrückt und bietet keine genauen Informationen in wie weit die aktuelle Verbindung den Benutzer „austricksen“ will.

Des Weiteren erhält der Benutzer eine Fehlermeldung im Hauptfenster des Browsers, welche in Abbildung 19 zu sehen ist.



Es besteht ein Problem mit dem Sicherheitszertifikat der Website.

Das Sicherheitszertifikat dieser Website wurde nicht von einer vertrauenswürdigen Zertifizierungsstelle ausgestellt.

Die Sicherheitszertifikatprobleme deuten eventuell auf den Versuch hin, Sie auszutricksen bzw. Daten die Sie an den Server gesendet haben abzufangen.

Es wird empfohlen, dass Sie die Webseite schließen und nicht zu dieser Website wechseln.

 Klicken Sie hier, um diese Webseite zu schließen.

 Laden dieser Website fortsetzen (nicht empfohlen).

 Weitere Informationen

- Wenn Sie zu dieser durch einen Link weitergeleitet wurden, dann überprüfen Sie die Websiteadresse in der Adressleiste, um sicherzustellen, dass dies die erwartete Adresse ist.
- Wenn Sie zu Websites wie <https://example.com> wechseln, versuchen Sie "www" zu der Adresse hinzuzufügen (<https://www.example.com>).
- Geben Sie keine persönlichen Informationen auf der Website an, wenn Sie diesen Fehler ignorieren und den Vorgang fortsetzen.

Weitere Informationen erhalten Sie unter "Zertifikatfehler" in der Internet Explorer-Hilfe.

Abbildung 19: IE7 - Fehlermeldung bei HTTPS-Seiten mit unbekanntem Wurzelzertifikat

Diese Informationen sind ausführlicher, als die des IE6 und können dem Benutzer eine grobe Vorstellung des Problems vermitteln.

Eine klare Einstufung und Beschreibung des Sicherheitsrisikos ist auch hier nicht gegeben und bedarf Verbesserungen, auf welche im Kapitel 4 eingegangen wird.

Szenario 2 – Bad Cert Domain: Die Darstellung dieses Szenarios unterscheidet sich lediglich in der angezeigten Fehlermeldung, weswegen sie nur kurz in Abbildung 20 aufgeführt wird:



Es besteht ein Problem mit dem Sicherheitszertifikat der Website.

Das Sicherheitszertifikat dieser Website wurde für eine andere Adresse der Website ausgestellt.

Abbildung 20: IE7 - Domainname aus dem Zertifikat weicht von der aufgerufenen URL ab

Die Behandlung der verschiedenen Fehler wird auf die gleiche Weise abgewickelt, sodass schwerwiegendere Fehler, wie die Vortäuschung einer anderen Domain (URL), im erklärenden Text untergehen können. Für den Benutzer ergibt sich dadurch eine problematischere Entscheidungsfindung, in wie weit dem Zertifikat vertraut werden kann.

Im Kapitel Konzept soll dazu eine grafisch bessere und visuell einfacher zu verstehende Meldung präsentiert werden.

Szenario 3 – Untrusted Issuer: Ist das Zertifikat abgelaufen, wird dies auf der Fehlerseite in einem kurzen Satz aufgeführt (Abb. 21). Eine ersichtlichere Darstellung wäre in diesem Fall wünschenswerter.



Es besteht ein Problem mit dem Sicherheitszertifikat der Website.

Das Sicherheitszertifikat dieser Website ist entweder abgelaufen oder noch nicht gültig.

Abbildung 21: IE7 - Das Zertifikat ist abgelaufen

3.3. Microsoft Internet Explorer 8.0

Der MSIE 8 wurde 2009 veröffentlicht und bietet eine Reihe neuer Funktionen sowie einen erheblichen Geschwindigkeitszuwachs zur Vorgängerversion. Im Sicherheitsbereich hat sich hingegen nichts Grundlegendes verändert. Der Umgang sowie die Visualisierung der Zertifikate geschieht genauso wie im IE7. Standardmäßig wurde nur die Einfärbung der Adresszeile für Extended SSL Validierung deaktiviert und muss im Menüpunkt „Sicherheit“ durch „Smartfilter aktivieren“ eingeschaltet werden. Diese Standardeinstellung bewirkt einen Rückschritt im Bezug auf das Sicherheitsbewusstsein des Benutzers und der Erkennung von EV-SSL Zertifikaten.

3.4. Mozilla Firefox 3.0.10

Mozilla Firefox ist ein kostenloser Web-Browser der Mozilla-Foundation. Der seit 2002 entwickelte Open-Source-Web-Browser erwies sich schnell als einer der beliebtesten Browser und ist derzeit, nach dem Internet Explorer, der zweithäufigste genutzte Browser.

Die aktuelle Version des Firefox 3.0 (FF3) ist 2008 erschienen und wird ständig verbessert.

3.4.1. Analyse gesicherter Verbindungen mit gültigem Zertifikat

Während die Version 2 des Firefox die Adresszeile gesicherte Verbindungen standardmäßig in hellorange einfärbte (Abb. 22) und ein Sicherheitsschloss-Symbol einblendete, erkennt man in Firefox ab Version 3 eine gesicherte Verbindung nur noch an dem Sicherheitsschloss-Symbol in der Statuszeile im Browser-Fuß (Abb. 23). Zusätzlich können erweiterte Sicherheitsinformationen, ähnlich wie im IE7, zum aktuellen Zertifikat einer HTTPS Seite angezeigt werden, indem man auf das Symbol der aufgerufenen Seiten – genannt „Favicon“ – links von der Adresszeile klickt (Abb. 24).

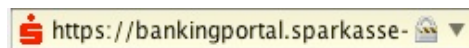


Abbildung 22: FF2 - Adresszeile einer gesicherten Verbindung

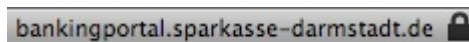


Abbildung 23: FF3 - Statuszeile im Browserfuß

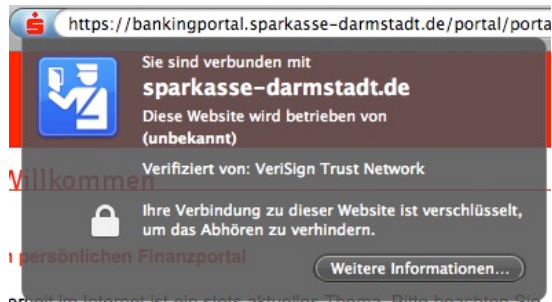


Abbildung 24: FF3 - Seiten- und Sicherheitsinformationen

Das Anzeigefenster, welches sich nach dem klicken auf das Favicon öffnet, wird im Firefox als „Instand Website ID“ bezeichnet. Von diesem Fenster aus, gelangt der Benutzer durch klicken auf „Weitere Informationen“, zu detaillierten Seiten- sowie Sicherheitsinformationen (Abb. 25).



Abbildung 25: FF3 - Detaillierte Seiten- sowie Sicherheitsinformationen

Sehr gut strukturiert ist die Aufteilung in drei Abschnitte: „Website-Identität“, „Datenschutz & Chronik“ sowie „Technische Details“.

Im ersten Abschnitt wird zu der URL auch die CA angegeben, welche das Zertifikat signiert hat. Ist der Betreiber der Seite im Besitz eines EV-SSL Zertifikats, wird dies im Feld „Besitzer“ angegeben. Wenn es sich nicht um eine EV-SSL Zertifikat handelt, zeigt der FF3 die Meldung „Diese Website bietet keine Informationen an, um Ihre Identität zu bestätigen“.

Obwohl ein gültiges Zertifikat vorliegt, kann diese Anzeige für manche Benutzer irreführend sein, da es den Anschein erweckt, dieses Zertifikat sei nicht komplett vertrauenswürdig.

Vor allem da auch ungesicherte Webseiten keine Informationen zum Besitzer bieten.

Der Abschnitt „Datenschutz & Chronik“ hilft dem Benutzer zu erkennen, ob er genau diese Seite schon einmal besucht hat, oder ob er durch einen Phishing-Angriff auf eine andere URL weitergeleitet wurde, die vorgibt dieselbe Seite zu sein. Diese Anzeige ist eine sehr guter Hinweis für den Benutzer, damit er erkennen kann ob er sich auf einer Seite befindet, die er schon einmal Besucht hat und somit der Seite mehr vertrauen schenkt.

Im letzten Abschnitt Technische Details, kann der Nutzer nachlesen, ob und welche Verschlüsselung auf dieser Seite verwendet werden. Diese Funktion ist eine gute Erweiterung, doch können Nutzer ohne Kenntnis über den Verwendeten Algorithmus keine Aussage über die Qualität der Sicherheit sagen. Eine Verbesserung stellt das im Abschnitt 7.4.2 vorgestellte Konzept zur Klassifizierung des Sicherheitsgrades des verwendeten Algorithmus dar.

Klickt der Benutzer auf „Zertifikat anzeigen“ erscheinen genaue Informationen zum Zertifikat dieser Webseite (Abb. 26).

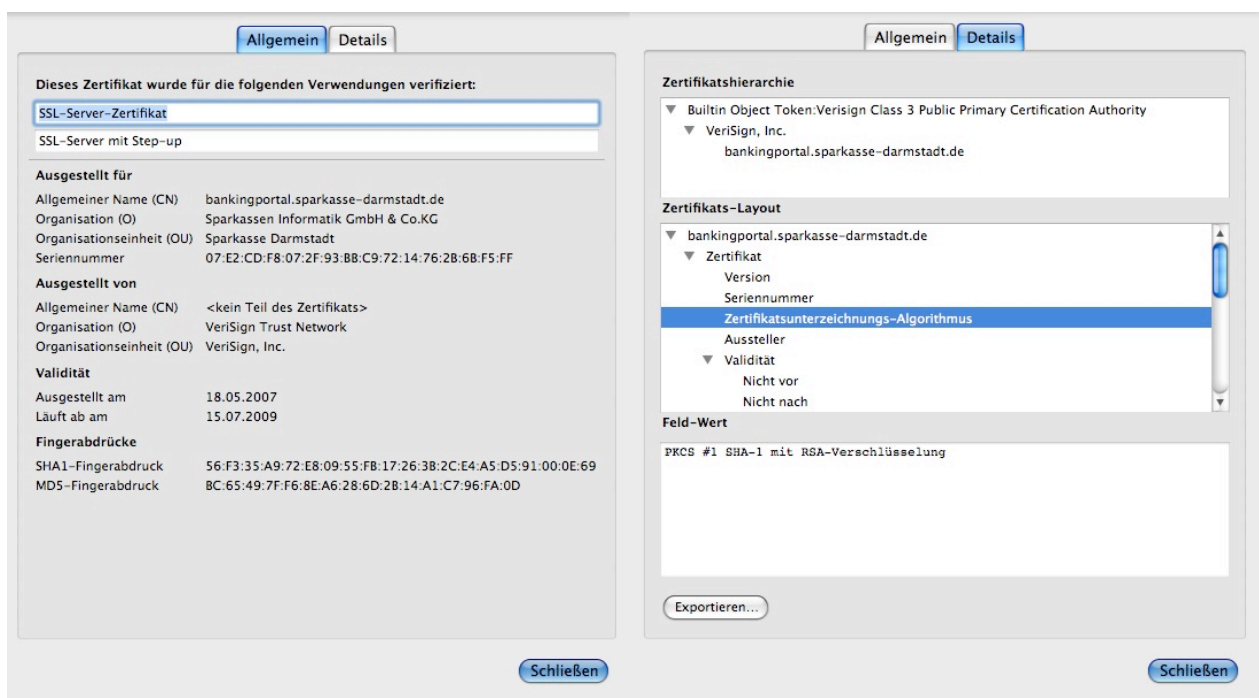


Abbildung 26: FF3 - Allg. Zertifikatsinformationen (links) sowie Zertifizierungspfad (rechts)

In einer strukturierten Übersicht können die wichtigsten Zertifikatsinformationen ausgelesen werden. Die Ansicht sowie die Informationen sind ähnlich denen des IE, jedoch wesentlich einfacher und klarer angeordnet. Auch im FF fehlt eine Erklärung über die Bedeutung der einzelnen Feldwerte, welches durch einen Konzeptuellen Verbesserungsvorschlag im Abschnitt 7.6 thematisiert wird.

Wird eine HTTPS Verbindung mit einem EV-SSL Zertifikat aufgerufen, wird die „Instant-Website-ID“ Leiste grün eingefärbt und der Besitzer dargestellt (Abb. 27).



Abbildung 27: FF3 - Extended Validation, Instant Website ID

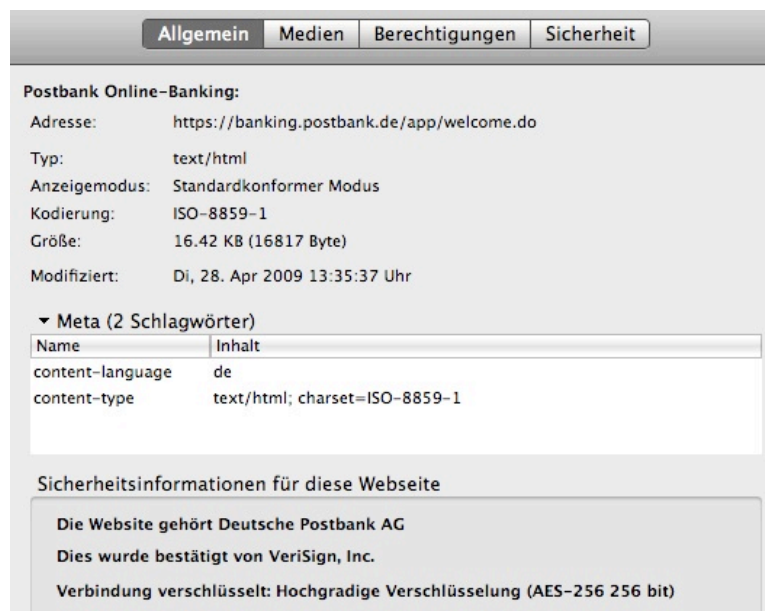


Abbildung 28: FF3 - Allg. Seiteninformationen einer gesicherten Verbindung mit EV-SSL Validation

Im unteren Abschnitt der Abbildung 26 erkennt der Benutzer deutlich, wem dieses Zertifikat ausgestellt wurde und welche CA es verifiziert hat.

3.4.2. Analyse gesicherter Verbindungen mit ungültigem Zertifikat

Szenario 1 – Unkown Issuer: Wird im FF3 eine HTTPS Verbindung mit einer gesicherten Seite aufgerufen, deren Wurzelzertifikat nicht standardmäßig im Browser enthalten ist, wird dem Benutzer ähnlich wie im IE7, eine Fehlerseite im Hauptfenster angezeigt (Abb. 29).

Sehr gut zu erkennen ist hier der aufgetretene Fehler, welcher zusätzlich durch eine verständliche Erklärung dargestellt wird.



Sichere Verbindung fehlgeschlagen

www.cacert.org verwendet ein ungültiges Sicherheitszertifikat.

Dem Zertifikat wird nicht vertraut, weil das Aussteller-Zertifikat unbekannt ist.

(Fehlercode: sec_error_unknown_issuer)

- Das könnte ein Problem mit der Konfiguration des Servers sein, oder jemand will sich als dieser Server ausgeben.
- Wenn Sie mit diesem Server in der Vergangenheit erfolgreich Verbindungen herstellen konnten, ist der Fehler eventuell nur vorübergehend, und Sie können es später nochmals versuchen.

[Oder Sie können eine Ausnahme hinzufügen...](#)

Abbildung 29: FF3 - Fehlerseite, Unbekanntes Wurzelzertifikat

Ein Klick auf das Favicon zeigt, dass der Browser keine Informationen dieser Seite bereithält, da jegliche Verbindung zu dieser Seite vorerst abgebrochen wurde (Abb. 30).

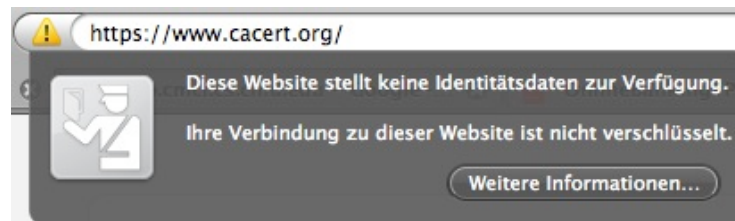


Abbildung 30: FF3 - Instand Website ID – Unknown Issuer

Werden die erweiterten Seiteninformationen über den Button „Weitere Informationen“ aufgerufen, erhält der Benutzer keine Informationen bezüglich einer verschlüsselten Verbindung oder Besitzerinformationen, obwohl eine HTTPS Verbindung versucht wurde aufzubauen (Abb. 31 und 32). Das Vorenthalten dieser Informationen ist störend, kann aber durch ein manuelles Herunterladen des Zertifikats überbrückt werden. Die notwendigen Schritte hierfür werden nachfolgend erläutert.

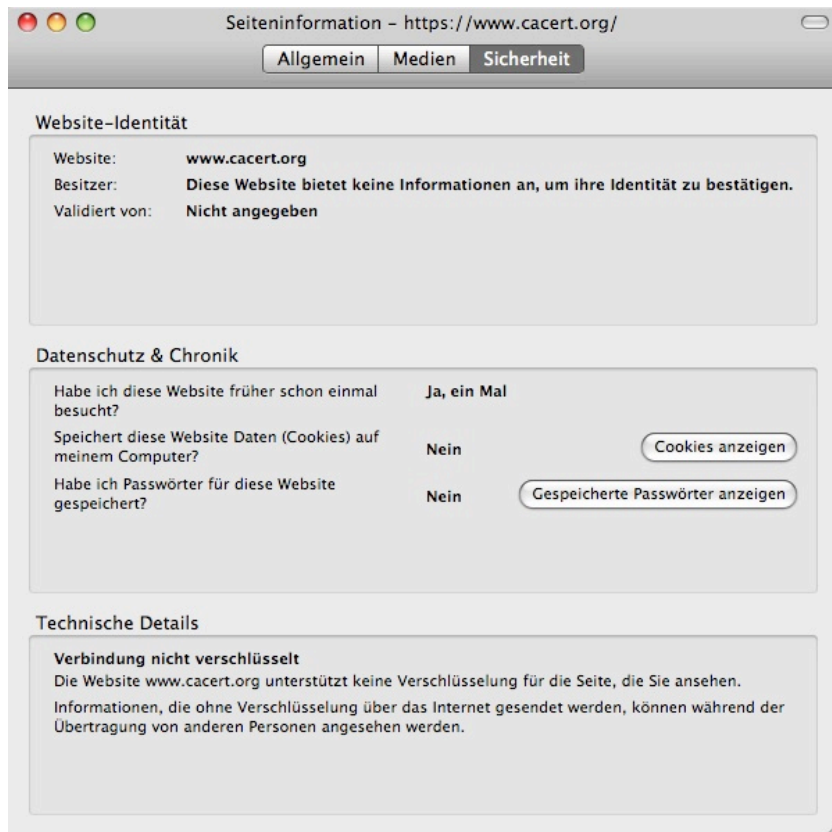


Abbildung 31: FF3 - Seiten- und Sicherheitsinformationen

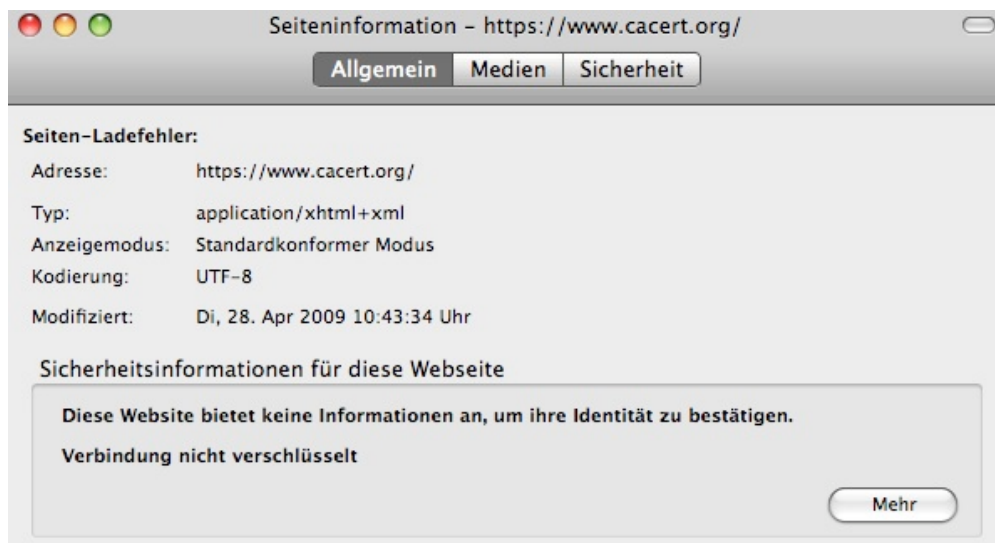


Abbildung 32: FF3 - Allgemeine Seiteninformationen

Um eine HTTPS Seite mit ungültigem Zertifikat zu betreten oder weitere Informationen über die HTTPS Verbindung zu erfahren, muss der Benutzer eine Ausnahme zum anzeigen der Seite einrichten. Dazu folgt er dem Link „Oder Sie können eine Ausnahme hinzufügen...“ auf der Fehlerseite des Browser (Abb. 29). Anschließend muss das Zertifikat heruntergeladen werden (Abb. 33).

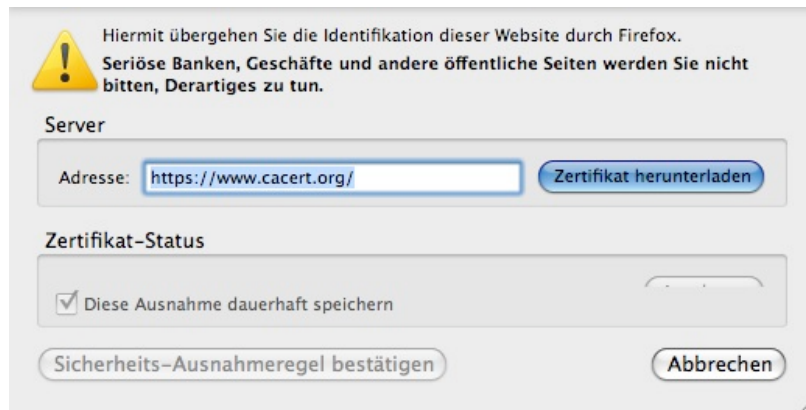


Abbildung 33: FF3 - Zertifikatsinformationen herunterladen

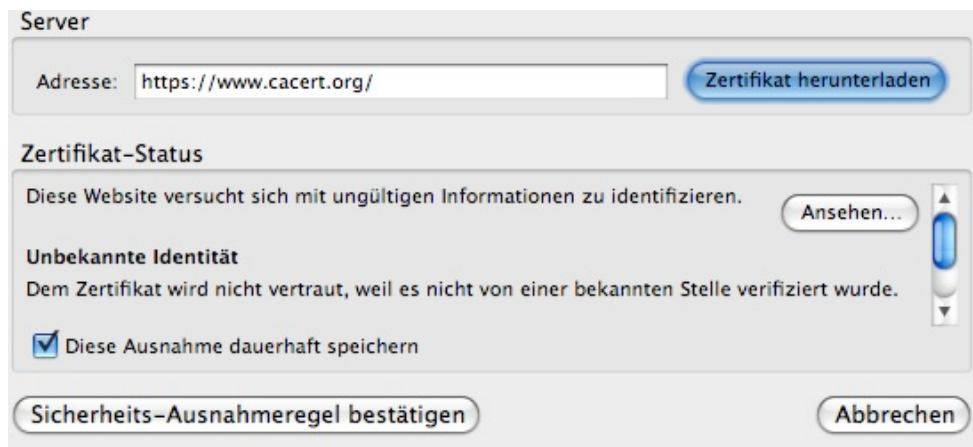


Abbildung 34: FF3 - Zertifikat Status

Wenn das Zertifikat heruntergeladen ist, zeigt FF3 noch einmal die resultierende Fehlermeldung des ungültigen Zertifikats im Feld „Zertifikat-Status“ an (Abb. 34). Daraufhin können die Zertifikatsinformationen eingesehen werden (Abb. 35).

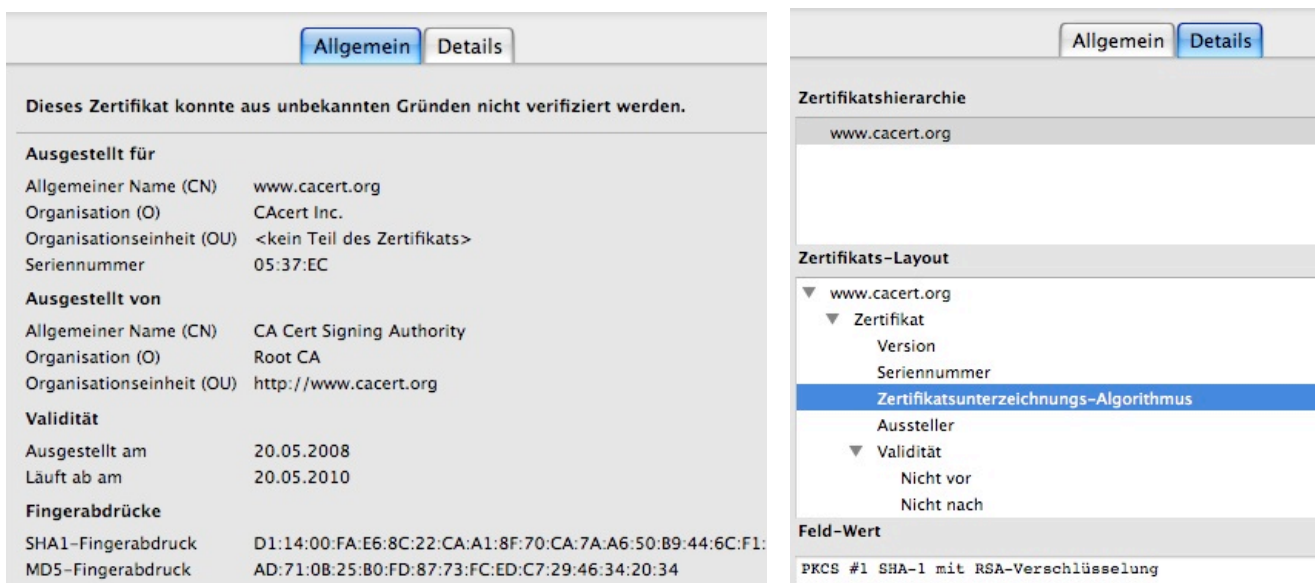


Abbildung 35: FF3 - (Rechts) Allg. Zertifikatsinformationen, (Links) Zertifizierungspfad

Wie in Abbildung 35 (links oben) zu sehen ist, zeigt der FF3 an, dass der Browser das Zertifikat nicht automatisch verifizieren konnte, da das Wurzelzertifikat nicht bekannt ist. Damit verweist FF3 in allen Schritten auf den Fehler des Zertifikats, was dem Benutzer hilft das Problem jederzeit vor Augen zu halten. Einzig der lange Weg zum Installieren des Zertifikats kann als Nachteil gewertet werden.

Andere Anhaltspunkte über die Güte der Verbindung, bleiben dem Benutzer aber auf den ersten Blick verborgen und können nur in den Zertifikatsdetails (Abb. 35 rechts) herausgesucht werden. Eine automatische Überprüfung und Visualisierung könnte dem Benutzer hier die Arbeit sowie die Entscheidung der Seite zu vertrauen oder nicht zu vertrauen, erleichtern.

Wird vom Benutzer eine Ausnahme für diese HTTPS Verbindung eingerichtet, zeigt FF3 die bekannten Informationen zum Status der Verschlüsselung in den Seiteninformationen sowie im Favicon an (Abb. 36).

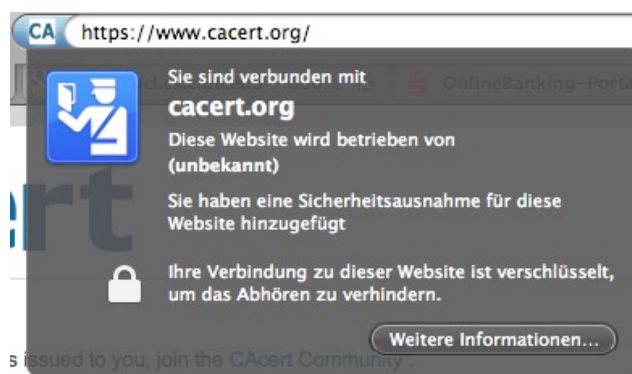


Abbildung 36: FF3 - Instant Website ID – Ausnahme wurde hinzugefügt

Ab diesem Zeitpunkt kann der Benutzer die erweiterten Seiteninformationen nutzen, um Details zur Verschlüsselung der HTTPS Verbindung zu erhalten (Abb. 37).



Abbildung 37: FF3 - Seiten- und Sicherheitsinformationen

Szenario 2 – Bad Cert Domain: Ruft der Benutzer eine gesicherte Webseite auf, die ein dafür ungültiges Zertifikat vorweist, warnt der FF3 entsprechend mit einer Fehlermeldung, dass das Zertifikat für eine andere Seite ausgestellt wurde (Abb. 38).



Abbildung 38: FF3 - Ungültiges Sicherheitszertifikat (Bad Cert Domain)

Wird das Zertifikat unter Ausnahmen heruntergeladen, verweist FF3 erneut auf einen möglichen Betrugsversuch in der Fehlermeldung in Abbildung 39 hin.

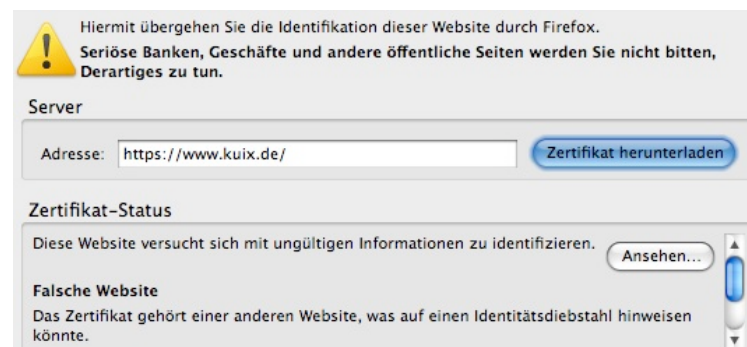


Abbildung 39: FF3 - Bad Cert Domain Warnung

Wird die Ausnahme hinzugefügt, werden keine weiteren Fehler oder Warnungen angezeigt. Weiterhin werden in den Zertifikatseigenschaften keine Informationen über den möglichen Betrugsversuch mit dem fehlerhaften Zertifikat für diese Seite angezeigt. Das Verbergen dieser Informationen nach manueller Installation des Zertifikats ist ein weiterer Nachteil und sorgt dafür, dass dem Benutzer über die Tatsache eines möglichen Fehlers im Zertifikat nicht mehr aufgeklärt wird.

Szenario 3 – Untrusted Issuer: Andere Zertifikatsfehler, wie ein abgelaufenes Zertifikat, werden entsprechend des Fehlers mit einer anderen Fehlerbeschreibung im Hauptfenster des Browser angezeigt (Abb. 40).

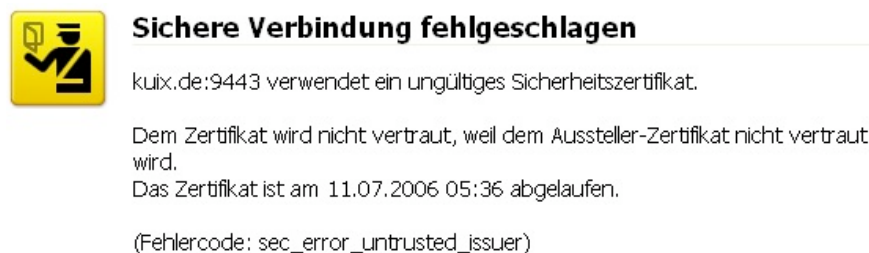


Abbildung 40: FF3 - Abgelaufenes Zertifikat

Vergleichbar mit dem Ablauf in Szenario 1, wird das Problem dargestellt, beziehungsweise durch Einfügen von Ausnahmen beseitigt.

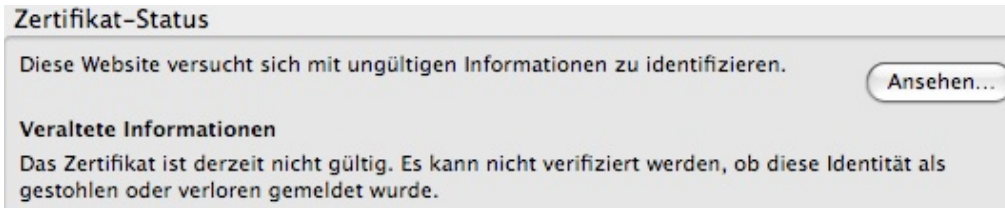


Abbildung 41: FF3 - Untrusted Issuer

3.5. Mozilla Firefox 3.5 Beta

Ein Hauptproblem der Fehlermeldungen bei Zertifikaten, denen nicht vertraut wird, ist der Mangel an Informationen für den Benutzer. Darüber hinaus kommt hinzu, dass die wenigsten Benutzer die Warnmeldungen verstehen.

Diesem Problem versucht die neue Version FF3.1 in einem Ansatz entgegenzuwirken. In der neuen Version sollen die Warnmeldungen auf der Hauptseite des Browsers, die bei gesicherten Verbindungen mit ungültigen Zertifikaten angezeigt werden, überarbeitet werden. Der Nutzer soll durch mehr Informationen über mögliche Gründe und Risiken aufgeklärt werden.

Mit einer frühen Betaversion des FF3.5 soll nun exemplarisch ein Szenario aufgeführt werden. In diesem Beispiel wird eine Seite aufgerufen, deren Wurzelzertifikat nicht verifiziert werden konnte, also vom Browser nicht automatisch akzeptiert wird.

Abbildung 42 zeigt die neue Warnmeldung.

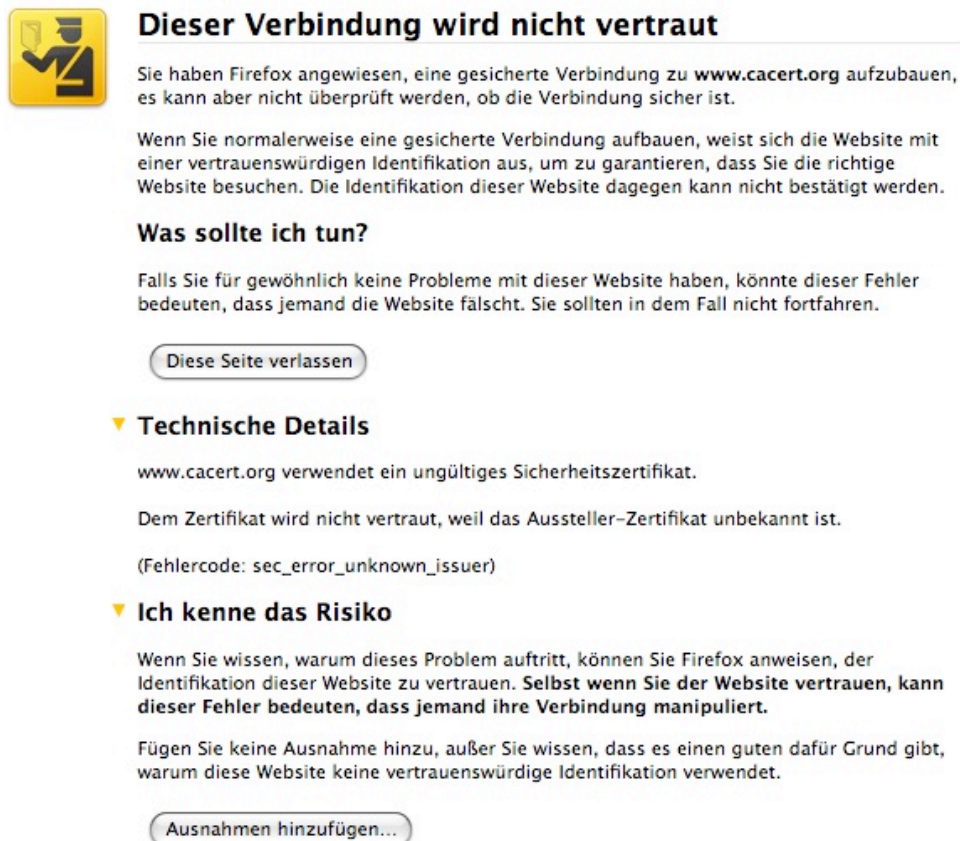


Abbildung 42: FF3.5 - Warnmeldung bei unbekanntem Wurzelzertifikat

Deutlich zu erkennen ist der ausführliche Text mit verschiedenen Erklärungen, wie der Benutzer sich verhalten sollte. Ebenso ändert sich der Titel der Warnmeldung. Während er in der Version 3.0.x noch „Sichere Verbindung fehlgeschlagen“ lautete, wird der Benutzer in Zukunft „Dieser Verbindung wird nicht vertraut“ lesen.

Die neue Formulierung ist verständlicher formuliert und spiegelt zudem den in der Tat gegebenen Sachverhalt wieder. Schritt für Schritt kann der Nutzer auf dieser Seite herauslesen, wie er sich zu verhalten hat. Unter „Was soll ich tun“ wird beispielsweise vorgeschlagen, die Seite zu verlassen, falls auf dieser Seite sonst nie ein Zertifikatsproblem aufgetreten ist.

Anschließend folgen die technischen Details zur aufgetretenen Fehlermeldung, welche aus der FF3 Version übernommen wurden. Zudem wurde der Text vor „Ausnahmen hinzufügen“ überarbeitet und soll über ein mögliches Risiko informieren, wenn diesem Zertifikat vertraut werden sollte.

Weitere Änderungen wurden bisher nicht vorgenommen.

3.6. Safari 4 Beta

Safari ist ähnlich zum Internet Explorer der Standardbrowser der Firma Apple und wird mit dem Betriebssystem Mac OS X ausgeliefert. Seit 2007 ist der Safari Web-Browser auch für Microsoft Windows verfügbar. Die aktuell offizielle Version ist 3.2, welche im Jahr 2008 erschien.

Die hier verwendete Version Safari 4 Beta ist eine stabile öffentliche Beta und soll 2009 den Beta Status verlassen. Durch die erhöhte Geschwindigkeit und die überdurchschnittliche Unterstützung gängiger Internetstandards wurde diese Version für die Analyse gewählt.

3.6.1. Analyse gesicherter Verbindungen mit gültigem Zertifikat

Wird eine HTTPS-Seite mit einem gültigen Zertifikat aufgerufen, sieht der Benutzer in der Titelleiste des Browsers ein kleines graues Schloss-Symbol (Abb. 43).



Abbildung 43: Safari 4 - Sicherheitsschloss Symbol bei gesicherten Verbindungen

Besitzt eine HTTPS Verbindung ein EV-SSL Zertifikat, wird zusätzlich der Besitzernamen in grüner Schrift in der Adresszeile angezeigt (Abb. 44).

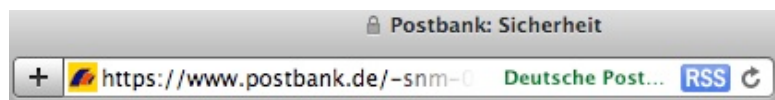


Abbildung 44: Safari 4 - Extended SSL Validierung

Mit einem Klick auf das Schloss-Symbol, erhält der Nutzer alle Informationen zu dem aktuellen Zertifikat der aufgerufenen HTTPS Verbindung (Abb. 45).

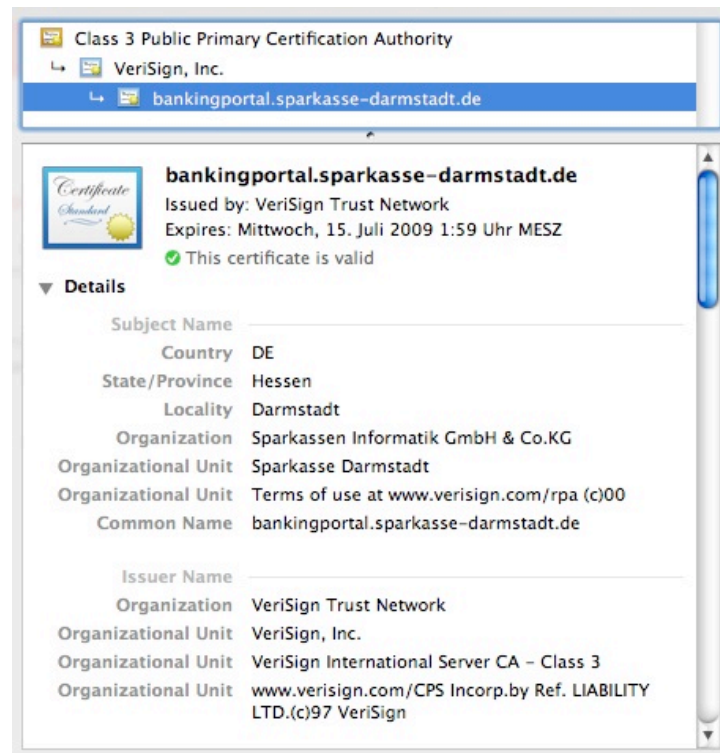


Abbildung 45: Safari 4 - Zertifikatsübersicht

Mit einer sehr übersichtlichen Darstellung präsentiert sich dem Benutzer als erstes der Zertifizierungspfad. Durch Anklicken der übergeordneten CAs kann der Nutzer sich die dazugehörigen Zertifikatsinformationen aufrufen und weiter unten im Detail ansehen.

Wenn das Zertifikat korrekt ist, wird dies durch einen kleinen grünen Haken symbolisiert. Andernfalls erscheint eine Fehlermeldung und ein rotes X-Symboln wird angezeigt. Wie Safari mit den Fehlermeldungen umgeht, wird nachfolgend beschrieben.

3.6.2. Analyse gesicherter Verbindungen mit ungültigem Zertifikat

Szenario 1 – Unknown Issuer: Wird ein Wurzelzertifikat nicht erkannt, zeigt Safari 4 automatisch ein Warnfenster und stoppt vorerst den Verbindungsaufbau. Ein Hinweistext weist den Benutzer darauf hin, dass das Zertifikat von einer unbekanntenen CA verifiziert wurde und der Aufruf der Seite ein Risiko sein könnte. Um weitere Informationen zu erhalten, folgt man dem Button „Show Certificate“ (Abb. 46).

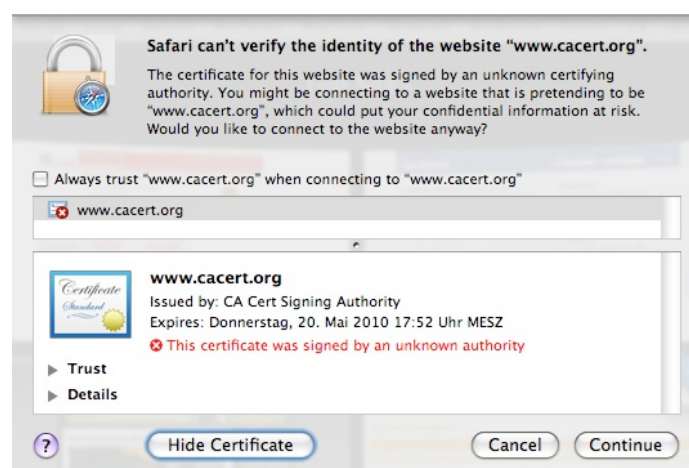


Abbildung 46: Safari 4 - Unkown Issuer

Der Vorteil dieser Ansicht liegt in der Übersichtlichkeit sowie der Leichtigkeit Informationen zu erhalten. So wird nicht nur der Zertifizierungspfad direkt angezeigt, sondern auch die Zertifikatsdetails können mit einem Klick aufgerufen werden (Abb. 46).

Wie in Abbildung 46 zu sehen, steht einmal ganz oben im Einleitungstext, dass dieses Zertifikat von einer unbekanntenen CA signiert wurde, genauso wie in der Mitte des Bildes, wird dies mit einem rot hervorgehobenen Text verdeutlicht. Dadurch erkennt der Benutzer sofort, wo das Problem aufgetreten ist.

Szenario 2 – Bad Cert Domain: Der Unterschied zum oben vorgestellten Szenario ist in diesem Fall nur der Text der Warnmeldung. Erwähnenswert sei aber hier der Umgang mit sowohl unbekanntenen, also vorerst ungültigen, als auch mit gültigen Zertifikaten in einer Zertifizierungskette. Abbildung 47 zeigt das Zertifikat des Besitzer der Seite, welches ungültig ist, da eine andere Domain im Zertifikat definiert ist. Dies wird dadurch mit einem roten X-Symbol und einem Text, dass das Zertifikat ungültig ist, dargestellt.



Abbildung 47: Safari 4 - Bad Cert Domain

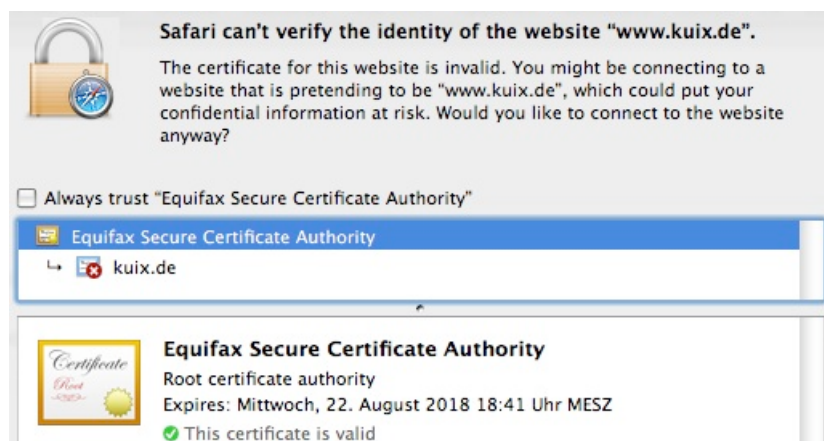


Abbildung 48: Safari 4 - Bad Cert Domain, Trusted CA

Klickt der Benutzer hingegen eine Ebene höher, sieht er, dass das übergeordnete Zertifikat gültig ist (Abb. 48). Dies heißt aber nicht, dass das drunter liegende Zertifikat automatisch auch gültig sein muss. Die Möglichkeit so einfach zwischen den Zertifikaten in dieser Kette hin und her zu schalten, erleichtert die Benutzung des Browsers.

Szenario 3 – Untrusted Issuer: Ist das Zertifikat abgelaufen, ist es damit automatisch ungültig und wird somit als nicht sicher eingestuft.

Mit einem roten X-Symbol wird dies dem Benutzer im Zertifikatspfad symbolisiert (Abb. 49). Eine entsprechende Fehlermeldung wird, wie weiter oben beschrieben, angezeigt.



Abbildung 49: Safari 4 - Untrusted Issuer

3.7. Übersicht der Benutzbarkeit der Browser

Das Ergebnis der Analyse zur Benutzerführung bei gesicherten Verbindungen der untersuchten Browser, soll anhand der folgenden Tabelle wiedergegeben werden.

Analyse / Browser	MSIE6	MSIE7 & 8	FF3	Safari 4
Fehleranalyse	+	++	++	++
Sicherheitsmerkmale	+	+	++	++
Hilfe	+	+	+	+
EV-SSL Erkennung	-	++	++	+
Bedienbarkeit	+	+	++	+++
Struktur/Übersicht	+	+	++	+++

Legende:

- Nicht vorhanden: -
- Gut: +
- Mittelmäßig: ++
- Sehr gut: +++

4. Konzept zur Verbesserung der Benutzerführung

In diesem Kapitel werden verschiedene Ansätze zur Verbesserungen der Benutzerführung sowie Visualisierung von Sicherheitsaspekten mit dem Umgang von gesicherten Verbindungen im Browser gezeigt. Anhand der Analyse im vorangegangenen Kapitel sind Schwachstellen in der Darstellung und Benutzerführung bei HTTPS Verbindungen aufgefallen. Dies betrifft teilweise den Mangel an geeigneten Informationen zur Sicherheit der aktuellen Verbindung und zum eingesetzten Zertifikat sowie deren Visualisierungshilfe.

Wie in den Beispielen in Kapitel 3 gezeigt wurde, erhält der Benutzer in vielen Fällen nicht die nötigen Angaben, um über die Qualität einer gesicherten Verbindung entscheiden zu können. Die Hauptproblematik ergibt sich im besonderen dann, wenn ein Wurzelzertifikat nicht vorhanden ist und der Browser den Zugang zu der HTTPS gesicherten Webseite unterbricht. In diesem Fall muss der Benutzer selbständig entscheiden, ob er dem Zertifikat und damit der HTTPS Verbindung zum Server und der Webseite vertraut. Der Browser bietet dem Benutzer in einigen Fällen keine Hilfestellung um die Entscheidung zu vereinfachen. Ein großes Problem ist, dass die sicherheitsrelevanten Informationen nicht übersichtlich strukturiert oder schwer zu finden sind. Weiterhin sind die Informationen nicht so aufgearbeitet, dass sie ein nicht kryptographisch versierter Benutzer verstehen kann. Beispielsweise fehlen geeignete Grafiken um auch schwierige Sachverhalte mit einfachen Abbildungen darzustellen.

Eine Studie von Venafi [20] zum Thema Sicherheitswarnungen bei HTTPS Verbindungen ergab, dass 91% der Befragten schon einmal eine Sicherheitsmeldung im Browser gesehen haben. Die Ergebnisse zeigten dabei, dass 41% der Befragten diese Meldung ignorierten und den Zugang zur Internetseite fortsetzten, während 43% die Verbindung abgebrochen haben (Abb. 50).

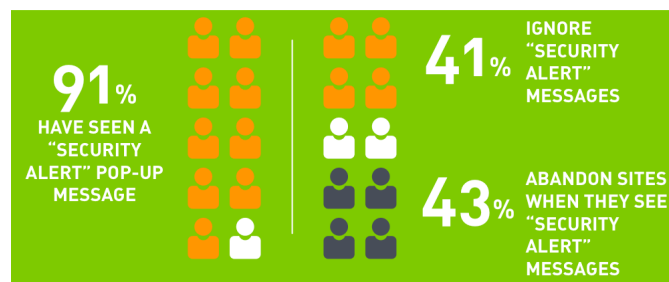


Abbildung 50: Studie zur Internetsicherheit [20]

Die Analyse ergab, dass nur 40% der Befragten eine Kompromitierung der Webseite vermuten, während knapp 30% von einem Fehler auf der Webseite oder im Browser ausgehen (Abb. 51).

Die restlichen 30% sind sich unsicher und können mit der Warnmeldung nichts anfangen, was sehr deutlich zeigt, dass die Informationen der Browser zu diesem Thema unzureichend sind.

Why do you think these messages appear?

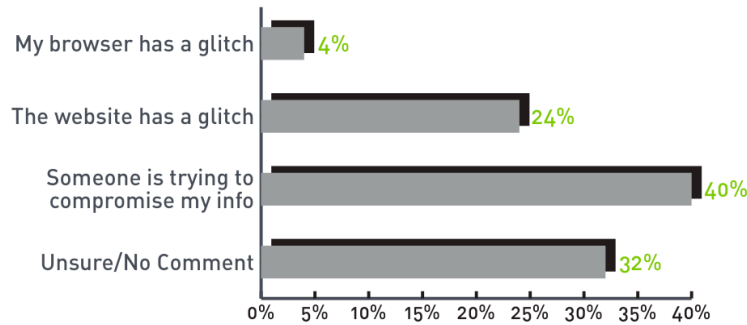


Abbildung 51: Umfrage zu den Ursachen einer Sicherheitswarnung [20]

Im folgenden werden Konzepte anhand eines Firefox Mock-Ups aufgezeigt, die es dem Benutzer erlauben die Qualität der gesicherten Verbindung einzustufen.

Ziel ist es, das allgemeine Sicherheitsbewusstsein der Nutzer zu sensibilisieren, indem die sicherheitskritischen Aspekte für den Benutzer analysiert sowie in geeigneter Form dargestellt werden.

Das Sicherheitsbewusstsein des Benutzers kann durch mehrere Schritte verbessert werden. Dazu zählt zum einen, die visuelle Form einer Anzeige über den Status der gesicherten Verbindung und zum anderen verständliche Informationen über den Sicherheitsfaktor dieser Verbindung, der die aktuell gültigen Sicherheitsnormen berücksichtigt. Wichtig ist, dass dazu nicht nur sichere Verbindungen mit gültigen Zertifikaten zählen, sondern speziell Zertifikate, welche durch ein fehlendes Wurzelzertifikat vom Browser nicht automatisch als gültig erkannt werden können. Es gilt dabei die Ursachen dieser Problematik herauszuarbeiten und sie entsprechend dem Benutzer aufzuzeigen, damit er schnell und sicher eine Entscheidung darüber treffen kann, ob er der aufgerufenen HTTPS-Seite trauen kann.

4.1. Adresszeile einfärben

Bedeutsame Sicherheitsaspekte, wie die Erkennung einer sicheren sowie vertrauenswürdigen Internetseite, sollten dem Nutzer direkt und deutlich sichtbar gemacht werden. Eine Möglichkeit ist das Einfärben der Adresszeile, was mittlerweile zu einem bekannten und "bewehrten" Konzept geworden ist.

Dieser Ansatz wird bisher im IE ab Version 7 in Verbindung mit EV-SSL und ungültigen Zertifikaten umgesetzt. Weiterhin wurde in der Version 2 des Firefox die Adresszeile bei allen HTTPS Verbindung gelb eingefärbt, jedoch in der aktuellen FF 3 wieder verworfen. Dieser Ansatz lässt sich allerdings in einer erweiterten Funktionalität im FF3 nachinstallieren.

In der verbesserten Variante sollen gesicherte Verbindungen mit gültigen Zertifikaten offensichtlicher hervorgehoben werden und gegenüber HTTPS Verbindungen ohne gültiges Zertifikat oder mit EV-SSL Zertifikat bewusst unterschiedlich dargestellt werden.

Es ergeben sich mehrere Szenarien, die jeweils eine andere Färbung der Adresszeile bewirken sollen. Die Szenarien teilen sich wie folgt auf:

- Gesicherte Verbindung mit gültigem Zertifikat und vorinstalliertem Wurzelzertifikat
- Gesicherte Verbindung mit gültigem Zertifikat ohne vorinstalliertem Wurzelzertifikat
- Gesicherte Verbindung mit gültigem EV-SSL Zertifikat und Wurzelzertifikat
- Gesicherte Verbindung mit ungültigem Zertifikat

4.1.1. Gesicherte Verbindung mit gültigem Zertifikat und vorinstalliertem Wurzelzertifikat

Wird eine HTTPS Verbindung zu einem Server aufgebaut, dessen Wurzelzertifikat von einer vorinstallierten Root-CA signiert wurde, erkennt der Browser dies automatisch. Ist das Zertifikat gültig, vertraut der Browser dem Zertifikat.

In diesem Fall sollte der Benutzer direkt erkennen können, ob er sich auf einer gesicherten Seite befindet. Dies geschieht in diesem Szenario, durch das Einfärben der Adresszeile in hell grün (Abb. 52).

Ähnlich zur Einfärbung der Adresszeile in dunkel grün bei einem EV-SSL Zertifikat (siehe Abschnitt 4.1.3), soll der Benutzer durch die ebenso grüne Farbe sehen, dass in beiden Fällen ein gültiges Wurzelzertifikat vorhanden ist und eine gesicherte Verbindung aufgebaut wurde.

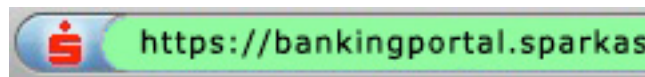


Abbildung 52: Adresszeile wird hell grün eingefärbt

4.1.2. Gesicherte Verbindung mit gültigem Zertifikat ohne vorinstalliertem Wurzelzertifikat

Wenn ein Benutzer eine HTTPS geschützte Seite aufruft, deren Zertifikat nicht von einem vorinstalliertem Root-CA Zertifikat signiert wurde, wird die Verbindung normalerweise im ersten Schritt unterbrochen. Diese Verbindungen sind verschlüsselt und somit gesichert, dennoch werden sie nicht vom Browser automatisch als vertrauenswürdig eingestuft. Um die Seite aufrufen zu können, muss der Benutzer das Zertifikat manuell installieren und somit selbst dem Zertifikat vertrauen.

In diesem Moment muss der Benutzer in der Lage sein zu erkennen, ob er dieser Seite vertrauen kann oder nicht. Eine Unterstützung zur Erkennung der Sicherheitsmerkmale dieser Webseite, werden in einem späteren Abschnitt beschrieben.

Wird das Zertifikat vom Benutzer manuell nachinstalliert, gilt es ab diesem Zeitpunkt für den Browser als vertrauenswürdig und die HTTPS Verbindung wird bei jedem Besuch ohne weitere Warnmeldungen aufgerufen.

Der Nutzer sollte jedoch in der Lage sein zu erkennen, ob er sich auf einer HTTPS geschützten Webseite befindet, die automatisch vom Browser als vertrauenswürdig erkannt wurde, oder auf einer Internetseite, die der Nutzer selbst als vertrauenswürdig eingestuft hat.

Gesicherten Verbindungen mit gültigem, aber vom Benutzer selbstständig nachinstalliertem Zertifikat, müssen sich somit von Zertifikaten mit vorinstalliertem Wurzelzertifikat unterscheiden können.

Dadurch erhält der Benutzer die optische Bestätigung, ob er sich auf einer gesicherten Seite befindet, deren Zertifikat er selbst vertraut hat, wodurch er dieser Internetseite kritischer gegenüberstehen sollte. Das Farbkodieren für die Adressleisten-Hintergrundfarbe sollte in diesem Fall gelb sein (Abb. 53).



Abbildung 53: Adresszeile wird gelb eingefärbt

Diese Farbkodierung in gelb sollte auch für alle vorinstallierten Zertifikate gelten, deren Wurzelzertifikat unbekannt ist. Dazu fallen beispielsweise Testzertifikate, welche bei einer Browserinstallation mitgeliefert werden können.

4.1.3. Gesicherte Verbindung mit gültigem EV-SSL Zertifikat und Wurzelzertifikat

HTTPS Verbindungen mit EV-SSL Zertifikat werden im FF3 durch eine erweiterte, in grün eingefärbte, Favicon-Anzeige dargestellt. Die Einfärbung der gesamten Adresszeile sollte in diesem Fall vollständigheitshalber zusätzlich integriert werden (Abb. 54). Dadurch wird die Einheitlichkeit sowie die Unterscheidung gegenüber HTTP Verbindungen, die einen weißen Hintergrund haben, gewährleistet.



Abbildung 54: Adresszeile wird grün eingefärbt

4.1.4. Gesicherte Verbindung mit ungültigem Zertifikat

Wird ein Zertifikat nicht akzeptiert, sollte die Adresszeile rot eingefärbt werden, um zu signalisieren, dass ein Problem aufgetreten ist (Abb. 55).

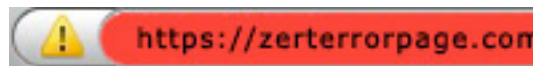


Abbildung 55: Adresszeile wird rot eingefärbt

4.1.5. Realisierung

Um die Hintergrundfarbe der Adresszeile im FF zu ändern, bedarf es der Anpassung der „userChrome.css“ Datei, welche sich im Profildrorder der Firefox Installation befindet.

Mit dem Befehl `#urlbar[level="___"]` erkennt FF um welche Art Zertifikat es sich handelt und kann derzeit zwischen drei Fällen unterscheiden:

Level="high" => EV-SSL Zertifikat
Level="low" => Gültiges Zertifikat
Level="broken" => Ungültiges Zertifikat

Eine zusätzliche CSS Farbformatierung sorgt anschließend für das Einfärben der Adresszeile. Der dazugehörige Code ist im Anhang zu finden.

Für die Umsetzung zur Unterscheidung sowie Farbkodierung von manuell hinzugefügten Zertifikaten, bedarf es einer erweiterten Implementierung eines Scripts, welches die hinzugefügten Zertifikate in einer Datenbank speichert. Dadurch kann der Browser zwischen den vorinstallierten und manuell hinzugefügten Zertifikaten unterscheiden und entsprechend die Adresszeile farblich markieren.

4.2. Passwortfelder einfärben

Die Aufmerksamkeit des Benutzers sollte weiterhin dann geweckt werden, wenn eine Internetseite die Eingabe von Logindaten verlangt.

In dem Fall, wenn Logindaten über eine ungesicherte HTTP Verbindung gesendet werden, besteht das Sicherheitsrisiko für den Benutzer, dass Benutzerdaten sowie Passwörter unverschlüsselt übertragen und von Angreifern leicht abgehört werden können.

Aus diesem Grund ist das Hervorheben von Passwortfeldern eine weitere Möglichkeit den Benutzer darauf aufmerksam zu machen, dass er darauf achten sollte, Logindaten nur über HTTPS Verbindungen zu übertragen.

Abbildung 56 zeigt eine Verwirklichung im FF3.

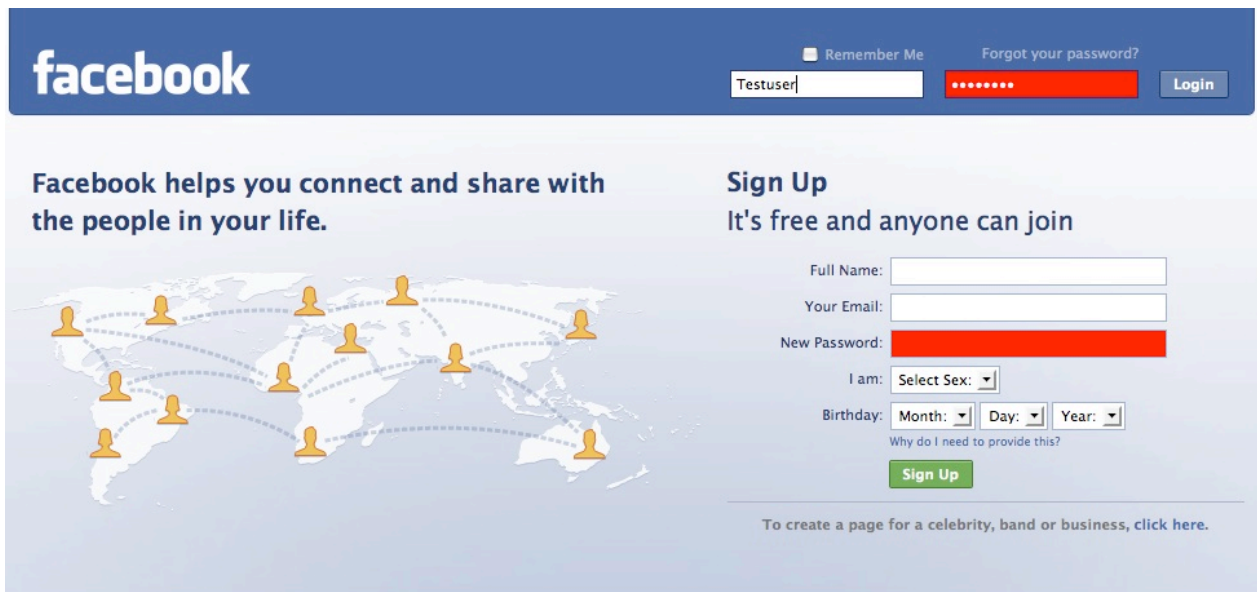


Abbildung 56: Passwortfelder auf Webseiten einfärben

Für die Implementierung dieses Konzeptes im FF muss die Datei „userContent.css“ aus dem Profilordner der FF Installation erweitert werden. Wie im Abschnitt 7.1.5 gezeigt wurde, erfolgt als erstes die Analyse ob es sich um eine gesicherte Verbindung handelt. Ist dies nicht der Fall, und der Benutzer befindet sich auf einer HTTP Verbindung, wird folgender Code in der „userContent.css“ Datei ausgeführt:

```
input[type="password"] {  
    background-color: red !important;  
}
```

Dieser Befehl bewirkt, dass alle Inputfelder, welche in HTML mit dem Typ „Password“ deklariert sind, einen roten Hintergrund erhalten.

4.3. Public-Key Verifizierung durch unabhängige Notar-Server

Oft ist es schwierig die Echtheit des Besitzers eines Zertifikats auf den ersten Blick zu erkennen. Um sich von der Richtigkeit des Zertifikats zu vergewissern, bedarf es beispielsweise der manuellen Verifizierung des eingetragenen Fingerprints im Zertifikat.

Dies kann dadurch geschehen, dass der Benutzer den Fingerprint im Zertifikat mit dem veröffentlichten Fingerprint auf der Webseite des Betreibers vergleicht. Jedoch kann diese Methode trügerisch sein, falls die Seite selbst kompromittiert wurde.

Die zweite Möglichkeit den Fingerprint zu verifizieren ist es, die Aussteller CA des Zertifikats telefonisch zu kontaktieren und den Fingerprint abzugleichen.

Problematisch ist jedoch, dass kaum jemand diese Verfahren nutzt, um die Echtheit zu überprüfen, da oft das Sicherheitsrisiko vernachlässigt wird und diese Methoden viel Zeit in Anspruch nehmen.

Ein anderes und automatisiertes Vorgehen wäre es, den im Zertifikat eingetragenen Public-Key auf seine Konsistenz hin zu prüfen. Da Phishing-Angriffe auf Webseiten in der Regel nur über einen kurzen Zeitraum möglich sind, ohne entdeckt zu werden, ist es essentiell nicht nur die Gültigkeit eines Zertifikats zu prüfen, sondern des darin enthaltenen Public-Keys, welcher zum Betreiber der Webseite gehört. Eine Kontrolle der Konsistenz des Public-Keys könnte somit die Echtheit eines Zertifikats plausibilisieren.

Dieses Konzept wird mit dem Firefox Add-on „Perspectives“ [17] umgesetzt. Perspectives nutzt für eine solche zur Kontrolle der Public-Keys mehreren öffentliche Server, welche in regelmäßigen Abständen die Public-Keys von diversen Zertifikaten abfragen und für jeden dieser so genannten Notar-Server den Public-Key in ihrer eigenen Datenbank speichern. Die Notar-Server bestätigen mit diesem Verfahren, dass sie von einem Webseitenbesitzer B, zum Zeitpunkt Z, einen Public-Key K gesehen haben. Um Angriffen auf ein bestimmtes Netz entgegenzuwirken, sind die Server in einer dezentralen Architektur über mehrere Netze verteilt.

Das Add-on ist nach der Installation immer aktiv und befindet sich in der Statuszeile des Browsers. Ruft der Benutzer eine HTTPS-Seite auf, überprüft Perspectives auf Clientseite ob der Public-Key des Zertifikats mit den Public-Key Informationen übereinstimmt, die von den Notar-Servern aufgezeichnet wurden. Abbildung 57 zeigt den Ablauf.

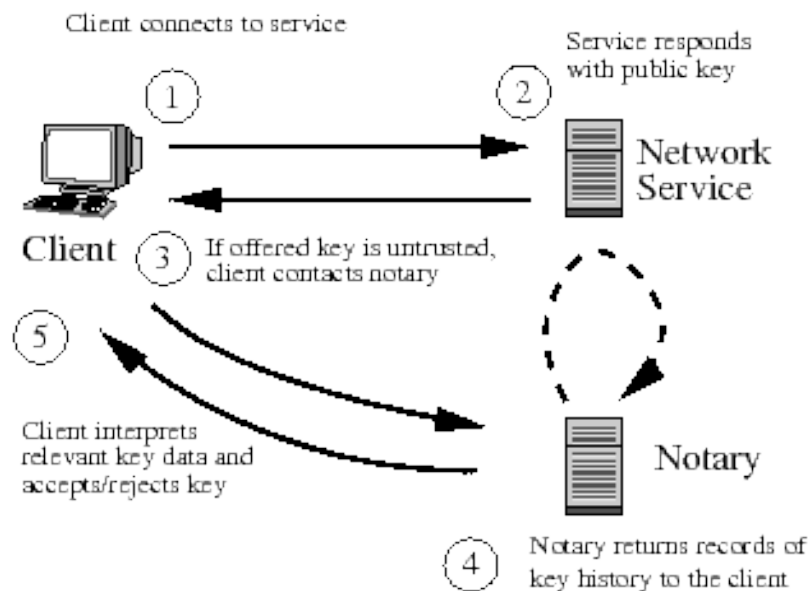


Abbildung 57: Funktionsweise von Perspectives

Wird erkannt, dass das Zertifikat über eine vordefinierte Zeitspanne konsistent war, wird in der Statuszeile des FF3 ein kleines grünes Häkchensymbol angezeigt. Ein Mouse-Over-Text zeigt dem Benutzer wie lange das Zertifikat konsistent war (Abb. 58)

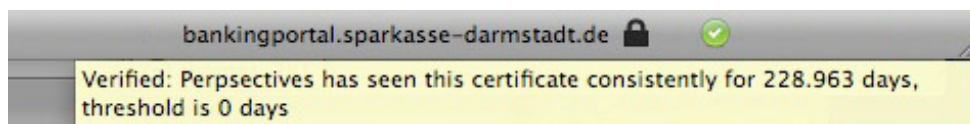


Abbildung 58: Statuszeilenanzeige über die Dauer der Konsistenz des Public-Keys

Mit einem Klick auf das Häkchensymbol kann der Nutzer in einer Grafik die Konsistenzaufzeichnungen der Notar-Server kontrollieren (Abbildung 59). Links zu sehen sind die IP-Adressen der Notar-Server,

rechts das mit einem blauen Balken visualisierte Zeitintervall, in welchem der Public-Key in seiner Konsistenz aufgezeichnet wurde.

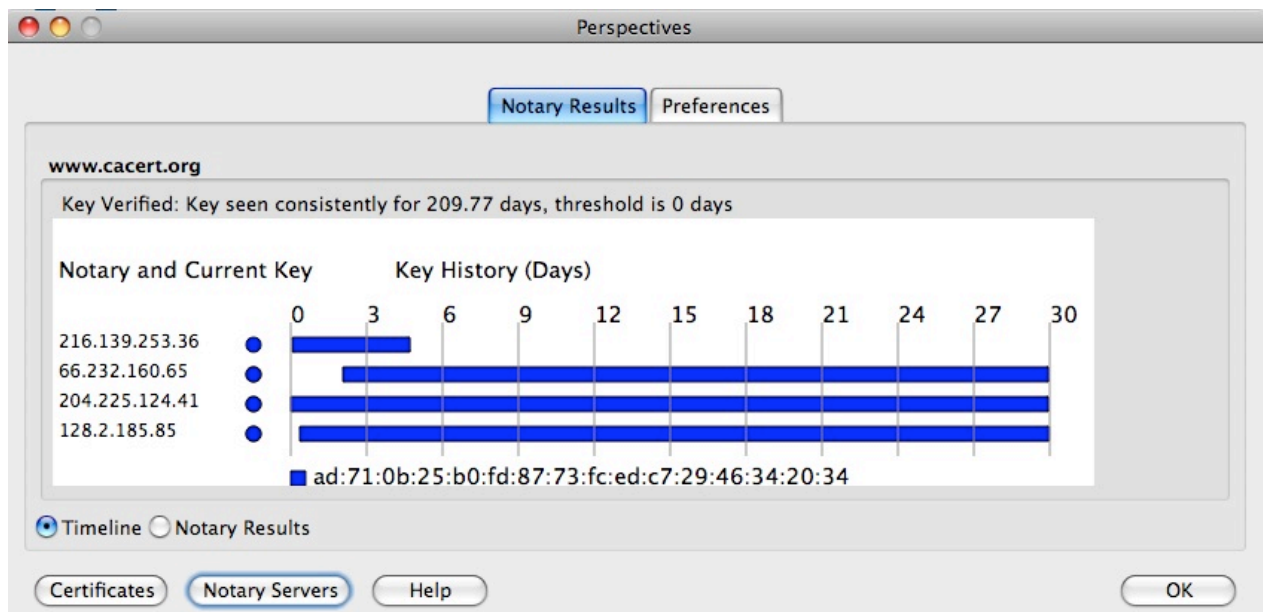


Abbildung 59: Kontrollgrafik der Konsistenz des Public-Key, aufgezeichnet von den Notar-Servern

4.4. Sicherheits-Add-on: Sec-Rank (FF Mock-up)

Zusätzlich zu dem vorgestellten Konzept der Visualisierung von Zertifikatseigenschaften in der Adresszeile, muss der Benutzer über die Qualität der Sicherheit der aktuellen HTTPS Verbindung informiert werden.

Diese Informationen sollen durch das „Sec-Rank“ Add-on für den FF3 realisiert und nachfolgend durch ein Mock-Up konzeptuell beschrieben werden.

Das Add-on soll mittels einer Grafik in der Browserleiste eine Gesamtbewertung des Sicherheitszustands der HTTPS verschlüsselten Verbindung sowie des eingesetzten Zertifikats anzeigen (Abb. 60).



Abbildung 60: Add-on Sec-Rank in der Browserleiste des FF

Anhand von vordefinierten Sicherheitscharakteristika, welche nachfolgend erläutert werden, zeigen die Grafiken eine Skala an, wie hoch die kryptographischen Sicherheitsaspekte der aktuellen HTTPS Verbindung klassifiziert wurden.



Abbildung 61: „Sec Rank“ - Bewertungsskala

Die farblichen Abstufungen der Grafik symbolisieren dem Benutzer wie hoch der Sicherheitsgrad der aktuellen Verbindung ist. Umso weiter die Grafik ausgefüllt ist, desto sicherer wurde die Verbindung eingestuft (Abb. 61). Wird keine gesicherte Verbindung unterstützt, oder ist das Zertifikat ungültig, ist der Rand der Grafik rot eingefärbt (Abb. 61 links).

Mit dieser Grafik kann der Benutzer den Sicherheitszustand verfolgen und entscheiden, wie kritisch er dieser Verbindung gegenüber stehen sollte und wie vertrauenswürdig das Zertifikat ist.

Durch eine Klassifizierung des kryptographischen Verschlüsselungsverfahrens der SSL-Verbindung (siehe Abschnitt 7.4.2), sowie einem Ranking über die Güte eines Zertifikats (siehe Abschnitt 7.4.1), wird die Gesamtqualität der Sicherheit dieser Verbindung in einer vierstufigen Skala dargestellt.

Zusätzlich zur Gesamtauswertung, können die einzelnen Klassifizierungsdetails der analysierten Sicherheitsmerkmale durch einen Klick auf die Grafik angezeigt werden (Abb. 62).



Abbildung 62: „Sec-Rank“ – Erweiterte Anzeige der Klassifizierung der Sicherheitsmerkmale

Der Benutzer erhält durch diese visuelle Darstellung ein schnellen Überblick über die Sicherheitsmerkmale der aktuellen Verbindung. Dadurch soll die Entscheidung vereinfacht werden, wann einem Zertifikate vertraut werden sollte und in welchem Fall nicht.

Das Konzept des „Sec-Rank“ Add-ons basiert auf einer vierstufigen Analyse der folgenden Sicherheitsmerkmale:

1) Art des Zertifikats

- Gesicherte Verbindung mit gültigem Zertifikat und vorinstalliertem Wurzel-Zertifikat
- Gesicherte Verbindung mit gültigem Zertifikat ohne vorinstalliertem Wurzelzertifikat
- Gesicherte Verbindung mit gültigem EV-SSL Zertifikat
- Gesicherte Verbindung mit ungültigem Zertifikat
- Public-Key Konsistenz durch Notar-Server bestätigt

Klassifizierung der Sicherheit

- 2) Verwendete Hash-Algorithmen zur Generierung des Fingerprints
- 3) Signaturverfahren zur Unterzeichnung des Zertifikats
- 4) Verschlüsselungsverfahren der SSL Verbindung

Nachfolgend werden die Klassifizierungen der einzelnen Sicherheitsmerkmale beschrieben.

4.4.1. Art des Zertifikats

Die oberste Grafik in der erweiterten Anzeige des Add-ons (Abb. 62) zeigt die Güte des Zertifikats an, welche in vier Abstufungen den Grad der Sicherheit bestimmt. Vergleichbar mit der Klassifizierung in Abschnitt 7.1, erfolgt auch in diesem Fall die Einstufung der Qualität des Zertifikats. Zusätzlich wird die Einstufung des Zertifikats jeweils um eine Stufe aufgewertet, falls Perspectives den Public-Key als konsistent markiert hat.

Abbildung 63 zeigt die farblichen Abstufungen der Sicherheitsskala.

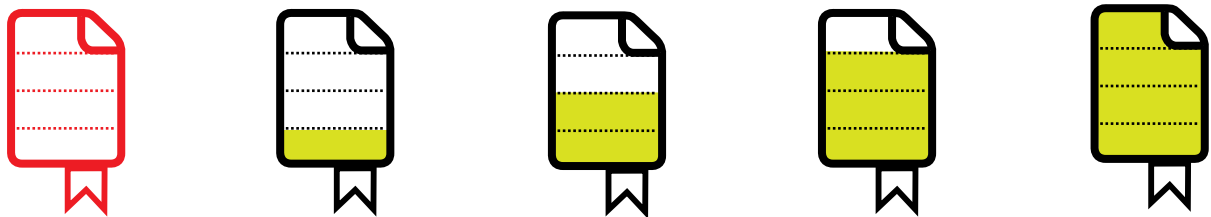


Abbildung 63: Skala der Zertifikatseigenschaften

Die Höhe der Sicherheitsstufe wird durch die Güte des Zertifikats bestimmt und wie folgt klassifiziert:

Stufe	Sicherheitsgrad	Sicherheitsgrad mit konsistentem Public-Key
4.		<ul style="list-style-type: none"> • Das Zertifikat ist gültig • Ein Wurzelzertifikat ist vorinstalliert • EV-SSL Zertifikat • Der Public-Key war konsistent
3.	<ul style="list-style-type: none"> • Das Zertifikat ist gültig • Ein Wurzelzertifikat ist vorinstalliert • EV-SSL Zertifikat 	<ul style="list-style-type: none"> • Das Zertifikat ist gültig • Ein Wurzelzertifikat ist vorinstalliert • Der Public-Key war konsistent
2.	<ul style="list-style-type: none"> • Das Zertifikat ist gültig • Ein Wurzelzertifikat ist vorinstalliert 	<ul style="list-style-type: none"> • Das Zertifikat ist gültig • Kein Wurzelzertifikat vorinstalliert • Der Public-Key war konsistent
1.	<ul style="list-style-type: none"> • Das Zertifikat ist gültig • Kein Wurzelzertifikat vorinstalliert 	
0.	<ul style="list-style-type: none"> • Keine Zertifikat vorhanden 	

Wurde der Public-Key als konsistent verifiziert, färbt sich zusätzlich der Wimpel am unteren Ende der Grafik gelb. Dies zeigt an, dass die Sicherheitsstufe um eins aufgewertet wurde.

Ist das Zertifikat fehlerhaft und ungültig, werden alle Stufen in rot ausgefüllt, um den Benutzer vor einem Problem zu warnen.

Umso höher der Skala ausgefüllt ist, desto vertrauensvoller ist das Zertifikat und damit der Server mit dem kommuniziert wird.

4.4.2. Klassifizierung der Sicherheit

Die Klassifizierung der verwendeten kryptographischen Verfahren und Algorithmen einer HTTPS Verbindung ist einer der entscheidendsten Aspekte der Qualität einer sicheren Verbindung. Zur Bewertung der eingesetzten Verfahren wurden die Empfehlungen von BSI [2], NIST [9] und ECRYPT [9] verwendet, welche in einer Übersicht auf Blukrypt [9] abrufbar sind.

Die Analyse sowie Auswertung der Sicherheitsaspekte einer HTTPS Verbindung geschieht in drei Schritten.

Im ersten Schritt wird der Hash-Algorithmus untersucht, welcher von der CA eingesetzt wurde, um den Fingerprint eines Zertifikats zu berechnen. Dies ist wichtig, um auf die Güte des Zertifikats zu schließen. Wurde beispielsweise ein Hash-Algorithmus verwendet, der schwach kollisionsresistent ist, wie der MD5 Algorithmus, so besteht eine größere Gefahr der Fälschung dieses Zertifikats.

Die Klassifizierung des Hash-Algorithmus wird anhand der zweiten Grafik in der erweiterten Anzeige visualisiert (Abb. 63). Abbildung 64 zeigt die einzelnen Sicherheitsstufen. Es gilt ebenso, umso mehr Stufen grün eingefärbt sind, desto sicherer ist der Algorithmus. Ist die Grafik rot, ist ein Fehler aufgetreten oder kein Zertifikat vorhanden.

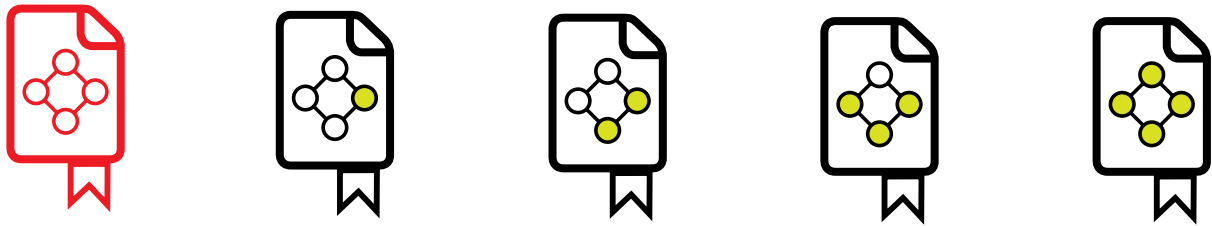


Abbildung 64: Klassifizierung des Hash-Algorithmus

Die folgende Tabelle gibt ein Ranking wieder, auf dessen Basis eine Einstufung der Sicherheit und damit der Qualität errechnet wird. Umso höher ein Algorithmus in der Liste steht, desto sicherer ist damit das Zertifikat.

Stufe	Hash-Algorithmus	Anzahl Bits
4.	SHA-256	256
3.	SHA-224	224
2.	SHA-1	160
1.	MD5	128
0.	-	-

Im zweiten Schritt wird der Algorithmus klassifiziert, der zur Erzeugung der Signatur benutzt wurde:

Stufe	Algorithmus	Blockgröße in Bits	Keylänge in Bits
4.	RSA/DSA	2048	224
3.	RSA/DSA	1728	224
2.	RSA/DSA	1248	160
1.	RSA/DSA	1024	160
0.	-	-	-

Die farblichen Abstufungen werden in Abbildung 65 symbolisiert.



Abbildung 65: Klassifizierung des Signaturverfahrens

Im letzten Schritt wird das eingesetzte Verschlüsselungsverfahren des SSL-Tunnels analysiert. Besitzt die HTTPS Verbindung zum Server einen kryptographisch sichereren Verschlüsselungsalgorithmus, sinkt das Risiko eines Angriffs.

Abbildung 66 zeigt die hierfür verwendeten farblichen Abstufungen.

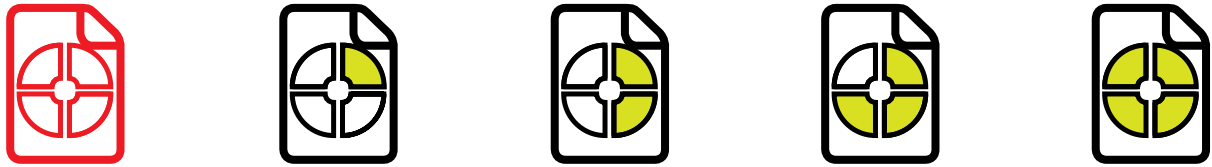


Abbildung 66: Klassifizierung der Verschlüsselung

Die Klassifizierung über den Sicherheitsstand der Algorithmen folgt in nachstehender Tabelle:

Stufe	Algorithmus	Anzahl Bits
4.	AES	256
3.	AES/3DES	192
2.	AES/2DES	128
1.	RC4	128
0.	-	-

Die Sicherheitseinstufung des Add-ons, ergibt sich damit aus der Summe der vier Rankings. Dazu zählen die Klassifizierungen der Verschlüsselung, der Signaturen, der Hash-Algorithmen sowie die Klassifizierung des Zertifikats.

Des Weiteren sollte eine regelmäßige Aktualisierung der Rankings durchgeführt werden, um das Add-on an die jeweilige Entwicklung der Sicherheitstechnik anzupassen.

4.5. Error-Code-Page im FF bei nicht vorhandenem Wurzelzertifikat

Die Warnmeldung des FF, wie sie genauso vom IE bekannt ist (Abb. 19 und 29), wird durch das installierte Sicherheitsplugin umgangen, sobald ein gültiges Zertifikat erkannt wurde. Mit einer Meldung unter der Adresszeile wird der Benutzer darauf hingewiesen, dass diese HTTPS Verbindung ein gültiges Zertifikat besitzt, andernfalls wird die Warnmeldung des FF angezeigt.

Abbildung 67 zeigt eine mögliche Realisierung, wie sie bereits vom Add-on Perspectives umgesetzt wird.

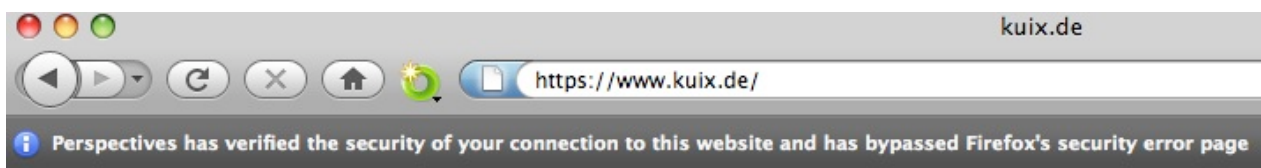


Abbildung 67: Error Code Page Umgehung

4.6. Erweiterte Hilfe im Zertifikats-Manager

Der Umgang mit sicheren Verbindungen und Zertifikaten im Browser kann außerdem dadurch verbessert werden, indem die einzelnen Einträge eines Zertifikats verständlicher erläutert werden.

Bislang werden in den Browsern nur die Zertifikate ausgelesen und ausgewertet. Der Benutzer wird jedoch nicht ausreichend mit Informationen unterstützt, was sich hinter den Einträgen wie Fingerprint, Public-Key, Zertifikatsunterzeichnungs-Algorithmus oder ähnlichen Stichworten verbirgt.

Eine Erweiterung zur Auswertung der Feld-Werte wäre ein zusätzlicher Hilfe-Button zu jedem Eintrag, mit einer Erläuterung dessen Bedeutung (Abb. 68).

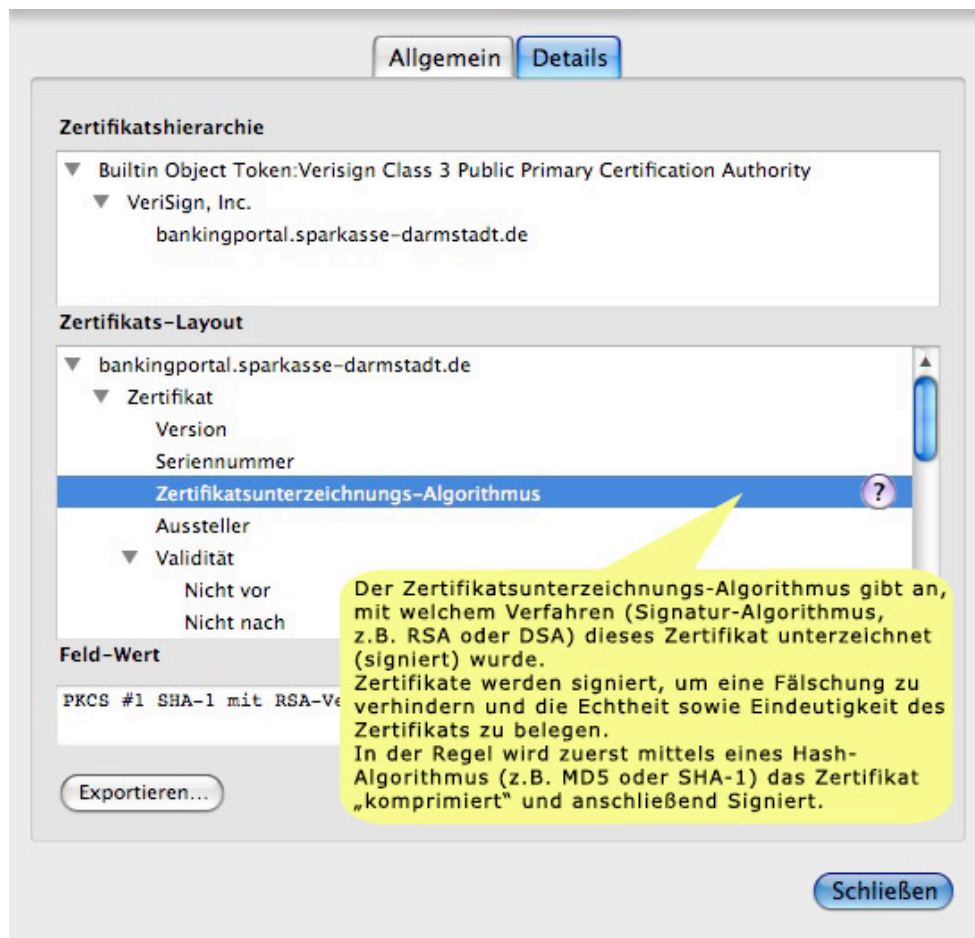


Abbildung 68: Erweiterte Hilfe im Zertifikats Manager des FF

5. Erweiterungsmöglichkeiten

Zusätzlich zum beschriebenen Konzept über die Klassifizierung der kryptographischen Merkmale einer gesicherten Verbindung, ergeben sich weitere Erweiterungsmöglichkeiten des „Sec-Rank“ Add-ons.

Ein mögliche Erweiterung wäre die Anzeige des Standortes des Servers, auf den aktuell zugegriffen wird. Befindet sich beispielsweise der Server einer deutschen Web-Seite nicht innerhalb von Deutschland, steigt das Risiko eines Phishing-Angriffs. Durch eine entsprechende Hinweismeldung kann der Benutzer darüber informiert werden, das ein Sicherheitsrisiko bei diesem Server aufgefallen ist.

Weitere Klassifizierungen könnten durch die Benutzer selbst oder eine zentrale Instanz vorgenommen werden und werden in den folgenden Abschnitten beschrieben.

5.1. Website Rating

Als Erweiterung zum Sicherheits-Plugin ist ein interaktives Bewertungssystem durch Internetnutzer denkbar. Dadurch können zusätzlich zu den Sicherheitsinformationen die Besucher eigene Erfahrungen mit der Webseite mitteilen und eine Aussage über den Sicherheitsstand treffen. Durch diese Erfahrungen können weitere Gäste dieser Webseite ein umfangreiches Bild darüber erhalten, ob die Seite vertrauenswürdig ist.

Ein solches Konzept verfolgt das Firefox Add-on WOT (Web of Trust) [21].

Die Zielsetzung von WOT ist das Schützen des Benutzer vor Online-Betrug, Identitätsbetrug, Spyware, Spam, Viren und unseriösen Shopping-Websites.

Durch ein Bewertungssystem können Besucher einer Internetseite diese in mehreren Kriterien bewerten. Die Summe über alle Bewertungen und Kriterien ergibt die Reputation dieser Webseite, welche durch ein farbliches Symbol angezeigt wird.

Das Add-on installiert ein kreisförmiges Symbol links von der Adresszeile, welches anhand der Reputation einer Webseite seine Farbe von grün, die Seite ist vertrauenswürdig, bis hin zu rot, die Seite ist nicht vertrauenswürdig, ändert. Durch einen Klick auf das Icon können die Bewertungen der Nutzer eingesehen und eigene Bewertungen eingetragen werden (Abb. 69)

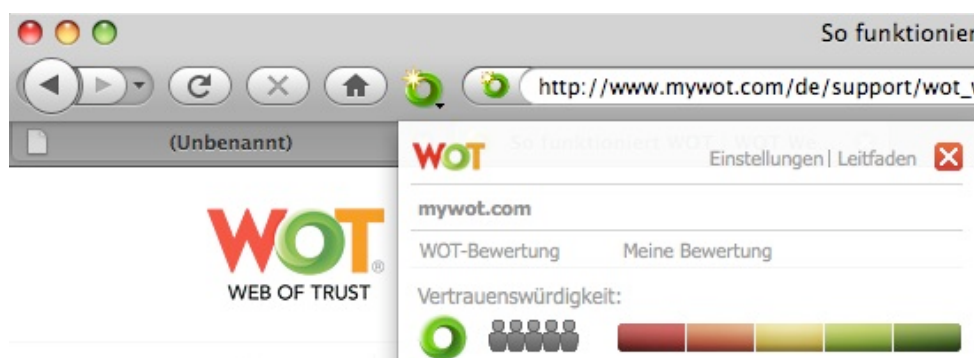


Abbildung 69: WOT Bewertungssystem

WOT bewertet eine Website in vier Kategorien, wobei die Einstufungen rein subjektiv vom Benutzer abhängig sind und deshalb immer mit Vorsicht betrachtet werden sollten. Allerdings bei einer genügend großen Anzahl von Bewertungen durchaus sehr aussagekräftig sind.



Abbildung 70: WOT Bewertungskriterien

- Vertrauenswürdigkeit:** Wird die Webseite als vertrauenswürdig betrachtet? Erscheint die Nutzung als sicher?
 Eine schlechte Bewertung kann auf Identitätsbetrugsrisiko, Internet-Betrug, Kreditkartenbetrug, Phishing, Gewinnspielbetrug, Viren, Adware oder Spyware hinweisen.
 Websites mit der Bewertung "ungenügend" enthalten möglicherweise lästige Werbung, exzessive Popups oder Inhalte, die den Browser abstürzen lassen.
- Händlerzuverlässigkeit:** Ist das Kaufen oder Verkaufen – bzw. sind Geschäftstransaktionen im Allgemeinen – auf dieser Website sicher?
 Eine "schlechte" Bewertung weist auf möglichen Betrug oder schlechte Erfahrungen beim Einkauf hin.
- Datenschutz:** Kann dem Besitzer der Website vertraut werden? Besteht ein Sicherheitsrisiko die E-Mail-Adresse anzugeben oder Dateien herunterzuladen?
 Eine "schlechte" Bewertung deutet auf Spam, Adware oder Spyware hin.
- Jugendschutz:** Enthält die Website Inhalte, die für Kinder oder Jugendliche nicht geeignet sind (pornografische Inhalte, Darstellung von Hass oder Gewalt), oder Material, das zu gefährlichen oder illegalen Handlungen ermutigt?

Die Zuverlässigkeit dieser Bewertung wird anhand der in grau schattierten Symbole dargestellt und gibt an wie viele Benutzer bereits abgestimmt haben.

5.2. Zentrale Zertifikatsdatenbank

Zertifikate die nicht als Wurzelzertifikat im Browser vorinstalliert sind, ergeben die größte Schwierigkeit für Benutzer zu erkennen, ob sie sich auf einer vertrauenswürdigen gesicherten Webseite befinden. Durch die derzeitige Fehlermeldung des Browsers und das Stoppen des Verbindungsaufbaus zum jeweiligen Server, wird ein Handeln des Benutzers verlangt, dessen Entscheidungsfindung die Kompetenz des Nutzers in der Regel übersteigt.

Ein zusätzlicher Lösungsansatz für Zertifikate, die nicht standardmäßig im Browser vorinstalliert sind, ist eine zentrale Verifizierungsinstanz.

Vergleichbar zu vorinstallierten Wurzelzertifikaten, ist eine zentrale Zertifikatsdatenbank in der Lage, Zertifikate als vertrauensvoll zu klassifizieren sowie den Benutzer mit einer Benachrichtigung zum eingesetzten Zertifikat, den Zugang zur Webseite freizugeben.

Die Klassifizierung der Zertifikate kann durch die in Kapitel 4 und 5 vorgestellten Methoden stattfinden. Die Richtlinien zur Verifizierung durch die zentrale Datenbank könnten dabei wie folgt aussehen:

- Die Notar-Server müssen den Public-Key als konsistent für mindestens 100 Tage klassifizieren
- Der Wert des „Sec-Rank“ Add-ons muss mindestens 50% betragen
- Mindestens 100 Benutzer müssen diese Seite als vertrauenswürdig bewertet haben

Durch die Kombination dieser verschiedenen Kriterien wird es möglich ein Zertifikat, welches nicht durch eine Root-CA signiert wurde, als vertrauensvoll einzustufen. Die Markierung der Adresszeile in gelb sollte zusätzlich dem Nutzer symbolisieren, dass es sich um eine vertrauensvolle und gesicherte Verbindung handelt, das Benutzen der Seite muss dennoch mit Bedacht erfolgen.

6. Fazit

Die Analyse der verschiedenen Browser hat gezeigt, dass der Benutzer fast gar nicht über die eingesetzten Sicherheitsmerkmale einer HTTPS-Seite aufgeklärt wird.

Zum einen wird der Nutzer nicht über Güte der verwendeten kryptographischen Verfahren aufgeklärt, welche zur Verschlüsselung der Verbindung sowie zur Qualität des Zertifikats dienen. Es fehlt gänzlich eine Klassifizierung der eingesetzten kryptographischen Sicherheitsmerkmale, um die Qualität einer HTTPS Verbindung aufzuzeigen. Diese Informationen werden besonders dann benötigt, wenn Wurzelzertifikate einer HTTPS-Seite nicht vorinstalliert sind und der Benutzer diese selbstständig nachinstallieren muss. In diesem Moment muss er sich eigenständig davon überzeugen, ob er dieser Verbindung vertrauen kann. Eine Hilfe für diese Entscheidung wird vom Browser nicht gegeben – hier kommen die in dieser Arbeit vorgestellten Konzepte zum Einsatz. Durch die Klassifizierung der Sicherheitsstufe der HTTPS-Seite, erhält der Nutzer eine Auswertung über die Qualität der verwendeten kryptographischen Verfahren und kann dadurch besser entscheiden ob die Seite vertrauenswürdig und damit sicher ist oder nicht.

Zum anderen ist eine Schwachstelle in der Erläuterung der eingesetzten Sicherheitsverfahren aufgefallen. Der Benutzer wird nicht darüber aufgeklärt, was die einzelnen Funktionen und Einträge, des im Zertifikats-Manager aufgeführten Merkmale, bedeuten. Hier besteht ein großes Verbesserungspotential, welches mithilfe des in Kapitel 5 beschriebenen Konzeptes, zur erweiterten Hilfe im Zertifikats-Manager, behoben werden soll.

Weitere Erweiterungen, wie beispielsweise eine subjektive Bewertung der Besucher der Internetseiten ist denkbar und wurde in Kapitel 5 beschrieben. Dadurch ist es möglich, zusätzlich zur reinen Auswertung der Sicherheitsfaktoren, eine direkte Beurteilung der Nutzer für eine Seite einfließen zu lassen. Diese menschliche Bewertung kann zur Entscheidungsfindung beitragen und einen weiteren Aspekt der Sicherheit dieser Seite hinzufügen.

Anhang

Realisierung zur Einfärbung der Adresszeile im Firefox:

```
#urlbar[level="high"] > .autocomplete-textbox-container,  
#urlbar[level="high"] > .autocomplete-history-dropmarker  
{  
  background-color: green !important;  
  color: #000000 !important;  
}  
  
#urlbar[level="low"] > .autocomplete-textbox-container,  
#urlbar[level="low"] > .autocomplete-history-dropmarker  
{  
  background-color: blue !important;  
  color: #000000 !important;  
}  
  
#urlbar[level="broken"] > .autocomplete-textbox-container,  
#urlbar[level="broken"] > .autocomplete-history-dropmarker  
{  
  background-color: red !important;  
  color: #000000 !important;  
  font-weight: bold !important;  
}
```

Literaturverzeichnis

- [1] **A-CERT Zertifizierungsdienst:** Welche Schritte sind notwendig um ein Zertifikat zu validieren / zu prüfen?, http://www.a-cert.at/php/cms_monitor.php?q=PUB-TEXT-A-CERT&s=20446cuj [Online, zugegriffen am 10.04.2009]
- [2] **Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen:** Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung, <http://www.bundesnetzagentur.de/media/archive/15549.pdf> [Online, 12.04.2009]
- [3] **Boosmann, Dana:** Kryptographie – Authentifikation und digitale Signatur, <http://www.wi.hs-wismar.de/~cleve/vorl/projects/krypto/ss04/Boosmann-Kryptographie.pdf> [Online, zugegriffen am 12.04.2009]
- [4] **Cases Luxembourg:** HTTPS HyperText Transfer Protocol Secure, <http://www.cases.public.lu/de/publications/dossiers/https/index.html> [Online, zugegriffen am 09.04.2009]
- [5] **CRYPTAS it-Security GmbH:** Verschiedene digitale Signaturen, http://www.cryptoshop.com/de/images/digsigappendix_560.jpg [Online, zugegriffen am 12.04.2009]
- [6] **Eckert, Claudia:** IT-Sicherheit, Oldenbourg Verlag München Wien 2008
- [7] **European Network of Excellence in Cryptology:** Yearly Report on Algorithms and Keysizes (2007-2008), <http://www.ecrypt.eu.org/ecrypt1/documents/D.SPA.28-1.1.pdf> [Online, zugegriffen am 05.05.2009]

-
- [8] **Froehling, Willem:** Konzept und exemplarische Implementation eines gesicherten Kanals zur Übertragung biometrischer Daten, <http://www.koram.de/doks/sa/node31.html> [Online, zugegriffen am 10.04.2009]
- [9] **Giry, Damien:** Cryptographic Key Length Recommendation, <http://www.keylength.com> [Online, zugegriffen am 12.04.2009]
- [10] **Hesse, Friedrich:** Client-Server, <http://www.e-teaching.org/technik/vernetzung/architektur/client-server> [Online, zugegriffen am 08.04.2009]
- [11] **Institut für Internet-Sicherheit - Fachhochschule Gelsenkirchen:** Verschlüsselung mit TLS, <http://www.internet-sicherheit.de/de/forschung/aktuelle-projekte/internet-frhwarnsysteme/ergebnisse/verschluesselung-mit-tlsssl> [Online, zugegriffen am 11.04.2009]
- [12] **Maj, Artur:** Apache 2 with SSL/TLS - Step-by-Step, <http://www.securityfocus.com/infocus/1818> [Online, zugegriffen am 09.04.2009]
- [13] **Maximov, Alexander und Khovratovich, Dmitry:** New State Recovery Attack on RC4, <http://eprint.iacr.org/2008/017.pdf> [Online, zugegriffen am 13.04.2009]
- [14] **Mozilla Foundation:** NSS and SSL Error Codes, <http://www.mozilla.org/projects/security/pki/nss/ref/ssl/sslerr.html> [Online, zugegriffen am 12.04.2009]
- [15] **National Institute of Standards and Technology:** Recommendation for Key Management - Publication 800-57 Part 1, http://csrc.nist.gov/groups/ST/toolkit/key_management.html [Online, zugegriffen am 05.05.2009]
- [16] **Net Applications:** Browser Market Share, <http://marketshare.hitslink.com> [Online, zugegriffen am 15.04.2009]
- [17] **Perspectives:** Firefox Add-on, <http://www.cs.cmu.edu/~perspectives/> [Online, zugegriffen am 14.04.2009]
- [18] **SoftEd Systems:** Wie funktioniert HTTPS?, <http://www.softed.de/fachthema/https.aspx> [Online, zugegriffen am 09.04.2009]
- [19] **TU-Darmstadt:** Zertifikate, <http://www1.hrz.tu-darmstadt.de/www/hilfe/zertifikate.tud> [Online, zugegriffen am 09.04.2009]
- [20] **Venafi:** Encryption Study 2007, http://www.venafi.com/Collateral_Library/VenafiEncryptionStudy2007.pdf [Online, zugegriffen am 04.05.2009]
- [21] **Web of Trust:** Firefox Add-on, <http://www.mywot.com> [Online, zugegriffen am 15.04.2009]
- [22] **Wikimedia:** EV-SSL, <http://de.wikipedia.org/wiki/EV-SSL> [Online, zugegriffen am 11.04.2009]
- [23] **Wikimedia:** MD5, <http://de.wikipedia.org/wiki/MD5> [Online, zugegriffen am 11.04.2009]
- [24] **Wikimedia:** Registrierungsstelle, <http://de.wikipedia.org/wiki/Registrierungsstelle> [Online, zugegriffen am 10.04.2009]
- [25] **Wikimedia:** SHA-3, <http://en.wikipedia.org/wiki/SHA-3> [Online, zugegriffen am 03.05.2009]
- [26] **Wikimedia:** Transport Layer Security, http://de.wikipedia.org/wiki/Transport_Layer_Security [Online, zugegriffen am 09.04.2009]
- [27] **Winkler, Jan:** HTTP-Transaktionen, http://www.html-world.de/program/http_2.php [Online, zugegriffen am 08.04.2009]