SmartTV - Eine Sicherheits- und Datenschutzanalyse von internetfähigen TVs

SmartTV - A security and privacy analysis of Internet based TVs

Bachelor-Thesis von Florian Oswald aus Weinheim

November 2012

TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Informatik
Fachgebiet Sicherheit in der Informationstechnik

CASED

EC SPRIDE

SmartTV - Eine Sicherheits- und Datenschutzanalyse von internetfähigen TVs
SmartTV - A security and privacy analysis of Internet based TVs

Vorgelegte Bachelor-Thesis von Florian Oswald aus Weinheim

Prüfer: Prof. Dr. Michael Waidner
Betreuer: Marco Ghiglieri

Tag der Einreichung:

Erklärung zur Bachelor-Thesis

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 1. Dezember 2012

_____

(Florian Oswald)

Zusammenfassung

Die vorliegende Arbeit beschreibt eine Sicherheits- und Datenschutzanalyse aktueller Smart TVs, welche Fernseher um zusätzliche Computerfunktionen, vor allem eine Anbindung an das Internet, erweitert. Hierbei wurde ein Testgerät der Firma Samsung ausgewählt und einzelne Subsysteme des Smart TVs untersucht. Die insgesamt vier durchgeführten Analysen der einzelnen Systeme fanden auf Softwareebene statt. Im ersten Test, der drei aufeinander aufbauenden Testreihe, wird eine Datenschutzrelevante Übertragung von sensiblen Daten aufgezeigt, welche bei jedem einzelnen Start des Smart TVs geschieht. Im weiteren wird gezeigt, dass die aktuelle Implementierung der Smart Feature Oberfläche anfällig für sogenannte Phishing Attacken ist. Ein dritter Test untersucht den potenziellen Videotextnachfolger HbbTV auf Schwachstellen und zeigt ein mögliches Angriffsszenario auf, mit welchem das Sendeverhalten von dritten aufgezeichnet werden kann. Im vierten und abschließenden Test der Arbeit, wird der „Pairing“ Vorgang des Smart TVs mit weiteren Geräten untersucht und die Schwachstelle inklusive Angriffsszenario dokumentiert.

Inhaltsverzeichnis

1 Einleitung

Das Internet ist aus unserem Alltag nicht mehr wegzudenken. Egal ob auf dem Laptop, Smartphone oder auf dem Tablet, das Internet ist ein ständiger Begleiter ins unserem täglichen Leben geworden. Es gilt mittlerweile als Selbstverständlichkeit überall online gehen zu können. Dem gegenüber steht eine Studie der Arbeitsgemeinschaft Fernsehforschung (AGF) [6] welche besagt, dass das Fernsehgerät durchschnittlich 216 Minuten am Tag genutzt wird. Eine Studie der Firma „Bitkom“ ergab im Jahre 2011, dass 50% der Fernsehnutzer gleichzeitig das Internet benutzen. Deshalb war es nur eine Frage der Zeit, bis die beiden Technologien in einem Gerät vereint wurden. Inzwischen ist mindestens jeder dritte hergestellte Fernseher mit Internet ausgestattet, so „Bitkom“ weiter [**?**]. Den Käufern sogenannter Smart TVs ist es mittlerweile möglich im Internet zu surfen, ihre Mails abzurufen oder Videoinhalte aus dem Internet auf ihren Smart TVs wiederzugeben. Dabei rückt das einfache Fernsehen immer mehr in den Hintergrund und die Benutzung des Internets, mit den damit verbundenen Möglichkeiten, nimmt immer mehr zu. Die Auswahl der verfügbaren Funktionen scheint dabei mittlerweile unerschöpflich.

Große Hersteller, wie Samsung, Panasonic oder Toshiba, läuteten mit der Einführung von Internetfähigen Fernsehern, den Smart TV Geräten, ein neues Zeitalter der Fernsehtechnologie ein. Alle gemeinsam haben diese Geräte der verschiedenen Hersteller, dass Sie zusätzlich zum gewöhnlichen Fernsehprogramm eine Reihe unterschiedlichster Funktionalitäten anbieten die vor allem eine bestehende Internetverbindung voraussetzen. Dass dieser neuer Markt von den Verbraucher angenommen wird zeigt eine Studie des Marktanalysezentrums Goldmedia, welche besagt, dass die Zahl der Smart TVs bis 2016 auf 20,1 Millionen Geräte in Deutschland anwachsen wird [45]. Zusätzlich hat eine Studie des ZVEI (Zentralverband der Elektrotechnik- und Elektroindustrie EV) ergeben, dass mittlerweile jedes fünfte Fernsehgerät ein Internetfähiges Smart TV ist. Außerdem ergab diese Studie dass von all diesen Smart TVs mehr als drei Viertel (76%) dieser Geräte schon heute mit dem Internet verbunden sind und diese Verbindung auch regelmäßig von den Nutzern verwendet wird [51]. All diese Statistiken untermauern, dass Smart TVs die Zukunft des Fernsehmarktes sind und den einfachen Fernseher auf lange Sicht verdrängen werden, zumal nur noch sehr wenige Hersteller, ein Fernsehgerät herstellen, welches nicht über diese smarten Funktionen verfügt.

Durch die Verschmelzung von Fernseher und Internet, ergeben sich ähnliche Sicherheits- und Datenschutzrelevanten Szenarien auf einem Smart TV Gerät wie sie der Nutzer bisher nur von seinem Computer oder Smartphone gewohnt ist. Für etwaige Angriffe, auf die Daten des Nutzers, bietet diese neue Kombination aus Internet und Fernseher ein ganz neues Angriffsumfeld für potenzielle Angriffe. Schwachstellen in Soft- bzw. Hardware, können ausgenutzt werden und können unter Umständen mit erheblichen Datenverlust oder Schäden verbunden sein. Dabei setzen aktuelle Systeme auf ähnliche Betriebssysteme, wie sie auf Desktop Rechnern bzw. Smartphones verfügbar sind. Daher können auch bereits bekannte Lücken, auf dem Smart TV relevant werden.

In einem Artikel des Online Magazins heise.de [19] wurde, am 26.04.2012, von einer ersten Sicherheitslücke in einem Subsystems eines Samsung Smart TVs [20](später auch bei Geräten der Firma Sony in der Modellreihe „Bravia“ nachgewiesen [14]) berichtet. Dabei wird die Möglichkeit eines Angreifers beschrieben, das Smart TV über die Kommunikation mit einem zweiten Gerät, zum Beispiel einem Smartphone, faktisch lahmzulegen. Hierfür wurde eine Schwachstelle im Verbindungsaufbau, zwischen Smart TV und einem zweiten Gerät ausgenutzt. Die Kommunikation von einem Smart TV Gerät mit weiteren Geräten wird in Kapitel 4.4 dieser Arbeit beschrieben.

Inwieweit aktuelle Smart TV Geräte gegen diese neuen Angriffe gerüstet sind, beschreibt die vorliegende Arbeit. Dabei werden exemplarisch einzelne Subsysteme eines Testgerätes der Firma Samsung untersucht, die daraus entstandenen Risiken werden analysiert und eine Lösungsempfehlung wird gegeben.

In einem einführenden Abschnitt, wird eine kurze Definition aktueller Smart TVs gegeben auf die ein Vergleich zu anderen Smart TV Geräten anderer Hersteller folgt. Im darauf folgenden Abschnitt, wird die Vorgehensweise der einzelnen Test beschrieben und das getestete Smart TV Gerät, der Firma Samsung, wird vorgestellt. Danach werden insgesamt vier Tests durchgeführt. Diese werden eingeleitet durch eine kurze Beschreibung des vorliegenden Systems, mit anschließender Analyse. Darauf folgt eine Bewertung der Risiken der vermeintlichen Probleme und ein Angriffsszenario wird vorgestellt um die Risiken zu verdeutlichen. Ein abschließender Teil behandelt Lösungsansätze für die vorliegenden Lücken. Es wird abgewogen ob die Lücken hiermit beseitigt werden können und das Angriffsszenario abgewehrt werden kann. Zum Abschluss der Tests, wird kurz auf andere Probleme am vorgestellten Testgerät eingegangen. Die letzten Kapitel der Arbeit gliedern sich in einen Ausblick, welcher einerseits Forschungsmöglichkeiten aufdeckt und anderseits beschreibt inwieweit eine Kommunikation der Probleme, mit den betroffenen Stellen, erfolgt ist. In einem abschließenden Fazit werden die Probleme nochmals resümiert und unter dem Aspekt der Daten- und Sicherheitsanalyse reflektiert.

2 Grundlagen: Smart TV

Der Begriff Smart TV beschreibt ein Fernsehgerät, welches zusätzlich um Computer Funktionen erweitert ist. Eine eindeutige Definition gibt es für die junge Technologie nicht und aktuelle Hersteller dieser Geräte bezeichnen die Smart TVs auch mit unterschiedlichsten Namen. Panasonic zum Beispiel, vermarktet seine Smart TV Angebot unter dem Namen Smart Viera [32], wohingegen die Firma Samsung ihre Geräte, unter dem in der Arbeit verwendeten Namen, Smart TV [38] verkauft. Im Folgenden soll eine Definition für ein „Smart TV“ gegeben werden:

„Sogenannte Smart TV Geräte oder auch Connected- bzw. Hybrid TV-Geräte sind Fernsehgeräte, welche sich durch die Integration von Computer Funktionen, insbesondere dem Internet, auszeichnen“

Testgerät der Arbeit

Zu Beginn der Integration von Computer Funktionen mit einem Fernseher, bestand die Möglichkeit Inhalte von Wechselmedien auf dem Fernsehgerät wiederzugeben. Die Funktionalität wurde stets erweitert, bis es auch erstmals möglich war, in einer Browser-ähnlichen Umgebung Internetseiten abzurufen bzw. vorgefertigte Inhalte der Hersteller zu nutzen. Dies war auch die erste Funktion, welche einen Internetzugang voraussetzte. Dabei war der Funktionsumfang überschaubar und kann in keinster Weise mit dem aktueller Smart TVs konkurrieren.

Das Smart TV, welches in dieser Arbeit als Testgerät diente, ist ein Smart TV der koreanischen Firma Samsung. Dabei trägt das Gerät die Modellbezeichnung UE40ES6300 [40]. Das Testgerät bietet zahlreiche Smart TV Funktionen, welche weit über das einfache surfen im Browser hinausreicht. Samsung orientiert sich bei der Gestaltung der Smart TV Bedieneroberfläche an der von bekannten Smartphone Oberflächen. Ein zentrales Programm verwaltet alle Smart Features in einer Oberfläche. Samsung bezeichnet dieses Programm als Smart Hub (näher untersucht und erläutert in Kapitel 4.2). Der Nutzer selbst entscheidet ob er das aktuelle Fernsehprogramm schaut, Smart Features nutzt oder beides parallel tut. Dabei ist die Auswahl mitgelieferten Smart Features groß. Angefangen bei einem Browser, der im Punkt Funktionalität mit aktuellen Desktop Versionen von Browsern mithalten kann. Weiter können E-Mail Clients genutzt werden, um E-Mails zu verwalten oder Applikationen installiert werden mit denen soziale Netzwerke auf dem Smart TV genutzt werden können. Der Funktionsumfang kann zusätzlich über den mitgelieferten App-Store [35] erweitert werden.

Vergleicht man dieses Testgerät mit aktuellen Geräten am Ende dieser Arbeit, kann man sagen, dass dieses Gerät in Sachen Funktionsumfang fast schon als „veraltet“ bezeichnet werden kann. Dabei können neue Applikationen auf dem Testgerät auch genutzt werden, dennoch sind neuere Smart TVs mit neuer Hardware ausgestattet. Als Beispiel sind hier Kameras zu nennen, die es dem Nutzer ermöglicht Videotelefonie zu nutzen oder das Smart TV über Gesten zu steuern. Auch dies bestätigt den rasanten Aufstieg des noch relativen jungen Marktes der Smart TVs. Die Entwicklung der aktuellen Geräte geht noch weiter, sodass neuere Smart TVs auch in Zukunft mit neuen Innovationen aufwarten werden.

Die Auswahl unseres Testgerätes erfolgte anhand folgender Kriterien:

- Hersteller mit möglichst großen Nutzerkreis
- Einstiegspunkt für die Arbeit, z.B. gefundene Sicherheitsrelevante Schwachstelle

Diese beiden Kriterien werden von der Samsung Smart TV Serie erfüllt. Da die Firma Samsung die entsprechende Marktrelevanz besitzt und zusätzlich, wie schon in der Einführung angemerkt, eine Sicherheitsrelevante Lücke festgestellt werden konnte, war dies ein entsprechendes Gerät für diese Arbeit. Dabei wurde die Auswahl des Gerätes anhand der vorhandenen Smart Features ausgewählt, um ein möglichst breites Testspektrum abdecken zu können. Dabei stammt das Modell, UE40ES6300, aus der 6 Series von Samsung. Hier unterscheidet Samsung insgesamt in neun verschiedenen Serien. Je größer die Nummer der Serie, desto umfangreicher ist der Funktionsumfang und die Bildschirmgröße des Gerätes. Diesen Zusammenhang bestätigt auch die Firma „Bitkom“ [8] in ihrer Statistik über Smart TVs. Relevant für die hier vorliegende Arbeit, war ausschließlich der Funktionsumfang des Smart TVs.

Das Testgerät verfügt standardmäßig über einen LAN und WLAN Adapter um sich mit einem bestehenden Netzwerk zu verbinden. Der integrierte WLAN Adapter ist bei neueren Geräten mittlerweile eine Standardkomponente, wobei es für diverse ältere Geräte diese Adapter zum Nachrüsten gibt, in der Form eine USB WLAN Sticks. Die Konfiguration des Netzwerkadapters ist sowohl per LAN bzw. WLAN möglich. Standardmäßig versucht der Fernseher immer erst die Verbindung über ein entsprechendes WLAN Netz aufzubauen. Dabei unterstützt das Gerät alle aktuellen WLAN Verschlüsselungstypen. In allen Tests konnte die Netzwerkkonfiguration problemlos durchgeführt werden. Neben den Anschlüssen für die Verbindung mit einem Netzwerk, verfügt das Testgerät noch über USB Anschlüsse, welche sich zum direkten Zugriff auf

Massenspeichergeräten eignen. Hiermit können Filme, Bilder oder Musik direkt abgespielt werden. Hierfür ist in den Smart Hub eine entsprechende Anwendung standardmäßig integriert, die das Streamen von Inhalte von Wechselmedien unterstützt. Als Betriebssystem wurde eine Linux Version identifiziert. Dabei handelt es sich ein eigens entwickeltes Betriebssystem, welches auf keinen bekannten Distributionen aufbaut.

Als zweites Referenzgerät wurde ein weiteres Gerät der Firma Samsung verwendet. Es trägt die Modellbezeichnung UE40D6200 [39] und stammt ebenfalls aus der 6 Series. Es wurde verwendet um Ergebnisse bzw. Probleme zu verifizieren und um die Testergebnisse einzustufen. Das Referenzgerät stammt nicht aus der aktuellen Generation (Testgerät 2012 - Referenzgerät 2011) wie das primär verwendete Testgerät. Dabei unterscheidet sich das Modell UE40D6200 dadurch, dass der Smart Hub über keinen integrierten Browser verfügt.

3 Beschreibung des Testsystems

Zu Beginn der Arbeit war nicht genau definiert, welche Systeme des Fernsehers untersucht werden sollten, da wenige wissenschaftliche Arbeiten für den Aufbau und die Struktur aktueller Smart TVs existieren. Daraus resultierte, dass anfängliche Tests sehr breit und allgemein ausgelegt wurden. Ziel war es, Sicherheits- bzw. Datenschutzrelevante Probleme zu finden und diese in einem Angriffsszenario exemplarisch durchzuführen. Die im folgenden Abschnitt beschriebenen Tests decken deshalb nicht alle gefunden Probleme, Auffälligkeiten oder Besonderheiten ab. In Kapitel 4.5 folgt eine Übersicht der zusätzlich gefunden Ergebnisse. Dabei stellt diese Arbeit keine vollständige Analyse des Testgerätes da, sondern eher eine Grundlage für weitere Forschungsarbeiten.

In den nun folgenden Abschnitten wird immer das gleiche Analyseschema verfolgt: Analyse – Risiko - Lösung (Vgl. Grafik 1). Zu Beginn der Tests wird der Testaufbau beschrieben und eine Einführung in das Themengebiet bzw. in die im Test verwendeten Subsysteme gegeben. Darauf wird detailliert der eigentliche Testvorgang beschrieben, mit den entsprechenden Details und Parametern. Im zweiten Kapitel, werden die Testergebnisse analysiert und die draus resultierenden Risiken werden erläutert. Die möglichen Gefahren für den Nutzer werden aufgezeigt und falls möglich ein Angriffsszenario erstellt und durchgeführt. Im letzten Kapitel werden Lösungsmöglichkeiten für alle gefundenen Probleme gegeben. Am Ende dieses Kapitels wird überprüft, ob das gegebene Angriffsszenario damit verhindert werden kann und ob die Lücke damit geschlossen werden könnte oder zumindest die Risiken vermindert werde können.

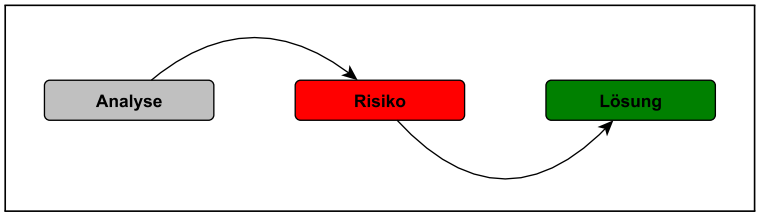

Abbildung 1: Vorgehensweise der Arbeit

Die Ausrichtung der Tests dieser Arbeit liegt auf der Analyse von Sicherheits- bzw. Datenschutzrelevanten Problemen welcher durch das Smart TV mir der darauf eingesetzten Software entstehen. Eine Analyse der darunterliegenden Hardware bzw. des Betriebssystem und die damit verbundenen Lücken, wurden in dieser Arbeit nicht untersucht und werden deshalb auch nicht weiter behandelt.

In den folgenden Abschnitten werden Bilder von Testaufbauten und Angriffsszenarien gegeben. Die in den unterstützenden Grafiken verwendeten Symbole werden hier erklärt. Außerdem wird die verwendete Hardware kurz erläutert. In den späteren Abschnitten symbolisieren die Grafiken die hier beschriebene Hardware. Außerdem werden die Symbole und Figuren aus den Angriffsszenarien erklärt. Weiter wird eine Einführung in die verwendeten Verbindungen gegeben.

In den verschiedenen Tests wurde das Smart TV Gerät mit dem Internet wahlweise über WLAN bzw. mit LAN verbundenen. Dabei wurden in Tests mit denen das Smart TV mit dem Test Laptop über WLAN bzw. Server per LAN verbunden war, die IP Adressen statisch zugewiesen. Im verwendeten Test über einen Router (Kapitel 4.3 und Kapitel 4.4), wurde die IP Adresse dynamisch via DHCP vom Router zugewiesen. Als DNS Eintrag wurde, in den Fällen ohne Router, der Google DNS Server mit der IP Adresse 8.8.8.8 verwendet [17]. Eine Lösung mit einem Proxy Server wurde ebenfalls betrachtet, dennoch fehlt dem Smart TV die Möglichkeit sich mit einem Proxy Server zu verbinden. Auch eine entsprechende Dokumentation seitens Samsung fehlt.

Für die Testfälle in denen das Smart TV mittels LAN Anschluss mit dem Gateway verbunden war, nutzten wir einen Linux Server mit dem Betriebssystem Ubuntu in der Server Edition 12.04 LTE. Der Rechner ist mit zwei separaten Netzwerkkarten ausgerüstet. Hierfür wurde die ausgehend Verbindung mit Internet als Gemeinsame Verbindung konfiguriert, sodass alle eingehenden Pakete vom Smart TV auf die zweite Netzwerkkarte weitergeleitet wurden. Als Analyse-Software auf dem Server, kam das Netzwerkanalysetool „Wireshark“ [49] in der Version 1.8.0 (Juni 2012) zum Einsatz. In Tests in denen das Smart TV per WLAN mit dem Netzwerk verbundenen war, nutzen wir einen Laptop (Dell Vostro 3350) mit Windows 7. Hier wurde das WLAN Signal auf den entsprechenden LAN Adapter weitergeleitet. Auch hier kam das Netzwerkanalysetool „Wireshark“ (Version 1.8.0) zum Einsatz. Im vierten Test wurde das Testgerät, Samsung Smart TV UE40ES6300, mit einem Smartphone gesteuert. Hierfür wurde ein Google Nexus S, ebenfalls von Samsung, verwendet. Die darauf verwendete Android Betriebssystemversion trägt die Nummer 4.1.2 alias Jelly Beans. Der eingesetzte TP Link WLAN Router verfügte über insgesamt 4 LAN Schnittstellen und einer Schnittstelle für ein WAN Signal. Er unterstützt alle bekannten Verschlüsselungstypen.

Zusätzlich zu der verwendeten Hardware werden an dieser Stelle noch Erklärungen zu den Pfeilen für Verbindungen in den Testaufbauten bzw. Protokollen oder Angriffsszenarien gegeben (Vgl. Abbildung 3). Insgesamt unterscheiden wir bei Verbindungen zum einen zwischen einer Verbindung über WLAN bzw. über LAN. Weiter wird unterschieden zwischen verschlüsselt bzw. unverschlüsselt Kommunikationen. Dabei bedeutet eine durchgezogene Linie eine Verbindung über

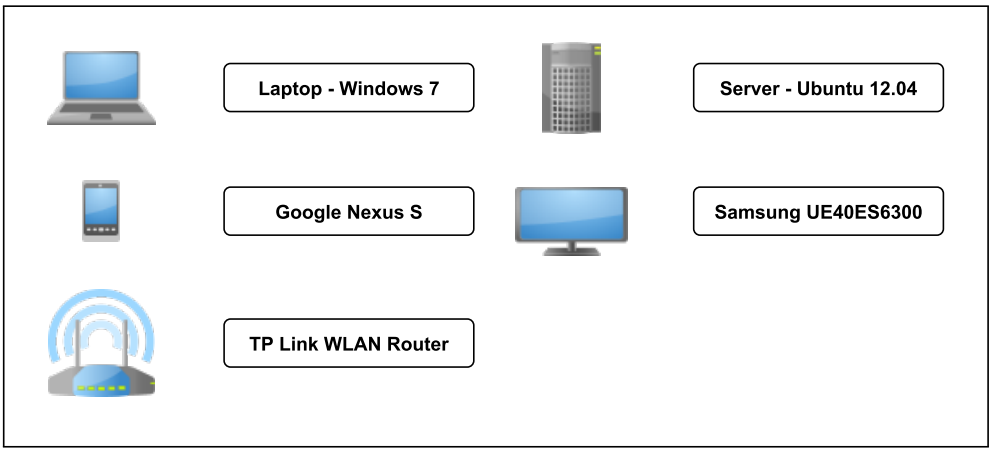

Abbildung 2: Verwendete Symbole der Hardware

LAN, eine gestrichelte Linie eine Übertragung über das WLAN Signal. Die Färbung der Linie gibt ihren Verschlüsselungsstatus an. Eine graue Linie bedeutet unverschlüsselte-, eine grüne Einfärbung dagegen eine verschlüsselte Verbindung. Die Pfeilspitze gibt jeweils die Kommunikationsrichtung an. Wird eine nicht gerichtete Kante verwendet (d.h. ohne Spitze) gilt die Kommunikation in beide Richtungen. In Grafik 3 sind Beispiele für Verbindungen gegeben. Dabei beschreibt Beispiel eins, eine verschlüsselte WLAN Übertragung von A nach B. Das zweite Beispiel, illustriert einen Datenaustausch zwischen A und B. Dieser hingegen findet unverschlüsselt über LAN statt.

Abbildung 3: Beispiele für Kommunikationen

Als letzter Punkt werden in den Angriffsszenarien der Begriff Angreifer bzw. Opfer verwendet. Diese beiden werden in Grafik 4 erklärt.

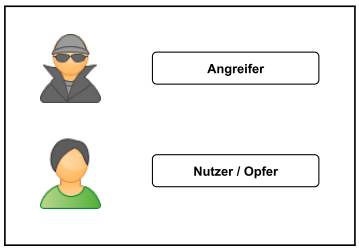

Abbildung 4: Angreifer und Nutzer/Opfer

Alle verwendeten Grafiken wurden mit der freien Software „yED Graph Editor“ erstellt, welche auch die entsprechenden Grafiken der einzelnen Bilder enthält [50].

An dieser Stelle sollen die verwendeten Schutzziele erklärt werden. Dabei werden diese eingesetzt, um die entstanden Risiken besser einordnen zu können bzw. zu klassifizieren. Dabei werden in folgenden Kapiteln die Schutzziele **Vertraulichkeit**, **Integrität**, **Verfügbarkeit** und **Authentizität** verwendet.

Vertraulichkeit beschreibt das Schutzziel, dass Daten zu jedem Zeitpunkt, d.h. während der Übertragung oder gespeichert, nur von autorisierten Personen gelesen und modifiziert werden können. Dass eine Modifizierung von Daten nicht unbemerkt geschehen kann, beschreibt das Schutzziel der **Integrität**. **Verfügbarkeit** stellt sicher, dass Daten erreichbar sind und Systemausfälle verhindert werden können. Das letzte Schutzziel **Authentizität** dient als Nachweis für die Echtheit der Daten. Ist das Schutzziel der **Authentizität** sichergestellt, kann der Nutzer sich sicher sein, dass die Daten von der entsprechenden Person stammen, von der er sie erwartet. [11]

4 Durchgeführte Analysen

Insgesamt wurden im Laufe der Arbeit vier Testszenarien durchgeführt, entsprechend analysiert und dokumentiert. Die ersten drei Tests bauen dabei logisch aufeinander auf, da sie jeweils eine Erweiterung des Testaufbaus aus dem vorangegangen Test sind.

- **Starttest:** Analysiert den Traffic des Smart TVs, welcher entsteht, wenn man das Gerät einschaltet und 120 Sekunden, ohne Interaktion, wartet.
- **Test der Smart Features** Testet neben dem Startverhalten des Smart TVs, den zusätzlichen Traffic, welcher durch die Nutzung der Smart Features entsteht.
- **Trafficveränderung durch ein Fernsehsignal** Erweitert die beiden ersten Tests um ein Fernsehsignal und untersucht Änderungen am Traffic.
- **„Pairing“ eines Smartphones mit dem Smart TV** Analysiert das Verbinden eines Smartphones mit dem Smart TV Gerät, um dieses mit dem Smartphone zu bedienen.

4.1 Starttest

Der Startest wurde am Anfang der Arbeit mit dem Smart TV Testgerät durchgeführt. Der Test sollte als Grundlage für neue Testfälle dienen und einen möglichen Rahmen für die Arbeit vorgeben. Dabei hatte er das Ziel, den Internettraffic näher zu untersuchen, welcher bei jedem einzelnen Start des Smart TVs entsteht, ohne das Nutzer eine Interaktion durchführt. Ziel war es, Datenpakete zu identifizieren und deren Herkunft zu bestimmten. Dabei sollten die einzelnen Verbindungen analysiert werden und falls möglich Standorten zugeordnet werden.

4.1.1 Analyse

Initial wurde das Smart TV mit dem Internet verbunden. Die Konfiguration der Interneteinstellungen wird persistent auf dem Smart TV gespeichert, sodass bei einem Neustart des Smart TVs, keine Neukonfiguration dieser Einstellungen notwendig ist. Hierfür wurde die LAN Schnittstelle des Fernsehers verwendet und mit dem entsprechenden Linux Server verbunden. Der genaue Testaufbau ist Grafik 5 zu entnehmen.

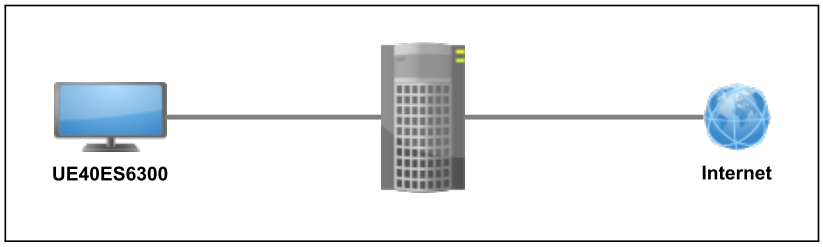

Abbildung 5: Testaufbau Starttest

Der Starttest wurde wie folgt durchgeführt: Das Smart TV wird mit konfiguriertem Internet eingeschaltet. Da kein Fernsehsignal verfügbar ist, erscheint ein schwarzes Bild auf dem Bildschirm. Der Test wird nach einem festgelegten Zeitfenster beendet. Dabei wird der gesamte Traffic am Server aufgezeichnet.

Insgesamt wurde ein Zeitfenster von 120 Sekunden gewählt in denen der Traffic gemessen wurde. Diese Zeit wurde daraus bestimmt, dass sich nach diesem Zeitfenster keine Änderungen mehr ergeben und alle Initialen Verbindungen terminieren. Dies konnte aus der Häufigkeit der Tests bestätigt werden. Zusätzlich zur dieser Zeit, mussten noch die Testbedienungen einheitlich gestaltet werden, d.h. Firmware Updates wurden vorher durchgeführt, da ein solches Update den Traffic einmalig verändert.

Nach dem Einschalten des Smart TVs dauert es einige Sekunden bis die ersten Verbindungen aufgebaut werden. Danach beginnt die Kommunikation zu verschiedensten Servern via HTTP und HTTPS. Im Folgenden, wird kurz auf die einzelnen Verbindungen eingegangen. Dabei wird jeweils auf die übertragenden Daten und den jeweiligen Standort des Servers eingegangen. Natürlich kann nur bei den ausgewählten HTTP Verbindungen eine Erklärung der Daten erfolgen.

Gefundene HTTP Verbindungen

Alle im Folgenden genannten IP Adressen sind beispielhaft für einen Testlauf. Bei anderen Tests konnten Änderungen in den Adressen festgestellt werden, wobei diese nur auf den letzten Eintrag der IP Adresse beschränkt war. In manchen Fällen wechselte die IP vollständig. Dennoch blieb der Provider der Adresse derselbe. Auch der Standort änderte sich dadurch nicht. Die Bestimmung des entsprechenden Providers und Hosters der einzelnen IP Adressen, sowie die Bestimmung des Standorts, wurde mit der Hilfe der Webseite utrace.de [31] durchgeführt.

In einer ersten DNS Anfrage werden verschiedene Namen von Servern aufgelöst. Darunter befinden sich Samsung spezifische Domänen. Diese Namen lauten **„samsung.com“**, **„samsungcloudsolutions.net“** und **„samsungrm.net“**.

Die darauffolgende erste HTTP Anfrage geht an den Server mit der IP **23.67.135.41** mit dem Standort **Boston** in den **USA**. Sie wird betrieben vom Anbieter für Online Technologien und Streaming Diensten Akamai Technologies [1]. Hierbei wird eine XML Datei angefordert, welche Informationen über Bilddateien und dazugehörige Links überträgt. Wie sich herausstellte, werden diese XML Dateien genutzt, um den die Software Smart Hub zu gestalten. Diese XML Dateien werden im Laufe des Starts immer wiederkehrend bezogen. Dabei handelt es sich unter Umständen um die gleichen Dateien. Generell werden von Servern von Akamai Technologies solche Daten angefordert. HTTPS Verbindungen zu Servern von Akamai Technologies konnten wir nicht feststellen. Hierbei handelt es sich um die Domain **„samsungcloudsolutions.net“**. In weiteren Tests konnten auch andere Standorte der Server festgestellt werden. Dieser wechselt je nach Test von den USA über England bis nach Deutschland.

In einer weiteren Verbindung wird zur IP Adresse **54.243.21.75** eine Kommunikation aufgebaut, welcher dem Cloud Service Provider „Amazon Cloud EC2“ [2] zugewiesen werden kann. Hier wird die Domäne **„samsungrm.net“** gehostet. Die Anfrage hierbei fordert eine XML Datei an, welche eine XML Liste mit den aktuellen Versionen vorinstallierter Applikationen enthält. Dabei ist die XML so strukturiert, dass sie die aktuelle Version der Applikation und einen entsprechenden Link zu den Dateien enthält. Der Download selbst, zeigt auf ein Ziel welches auf dem bekannten Samsung Server liegt (**„samsungcloudsolutions.net“**).Ein Test zeigte, das die Dateien auch über einen Browser auf einen PC heruntergeladen werden können. Die bezogenen zip-Dateien enthalten eine ausführbare .so Datei, welche dennoch auf normalen Desktop Rechnern nicht lauffähig ist. Zusätzlich enthält das Archiv, Textdateien über Version und Hinweise für die Installationsroutine. Die XML Datei ist einfach strukturiert und enthält weder Angaben im Kopf der XML Dateien, z.B. die benutzte Kodierung oder die Sprache, noch eine Signatur zur Überprüfung der Integrität der Daten.

Datenschutzrelevante Header Felder

Im Header dieser HTTP Anfragen, werden zusätzlich zu den Standardparametern noch die Felder „DUID“ und „MACAddr“ übertragen. Hinter dem Feld „MACAddr“ verbirgt sich die entsprechende MAC Adresse des LAN Anschlusses des Smart TVs. Bei der MAC Adresse handelt es sich um eine eindeutige Identifikationsnummer von Netzwerkgeräten. Das zweite Feld „DUID“ ist eine Device Unique ID. Laut dem Samsung Developer Guide [42] ist diese „DUID“ für jedes Smart TV Gerät eindeutig. Zusätzlich dazu, kann mit der Hilfe einer eingebauten Funktion die „DUID“ anhand der MAC Adresse des Smart TVs berechnet werden. Die Gegenrichtung, die Berechnung der MAC Adresse anhand der „DUID“, ist nicht möglich. Öffnet man die gegebene Seite (**„www.samsungrm.net/openapi/device/auth/query“**) mit einem Desktop Browser, wird eine leere XML Datei zurückgegeben, da der Zugriff verweigert wird. Ein Ausschnitt der entsprechenden Header sind in Bild 6 verdeutlicht. Der Screenshot stammt aus dem Netzwerkanalysetool „Wireshark“.

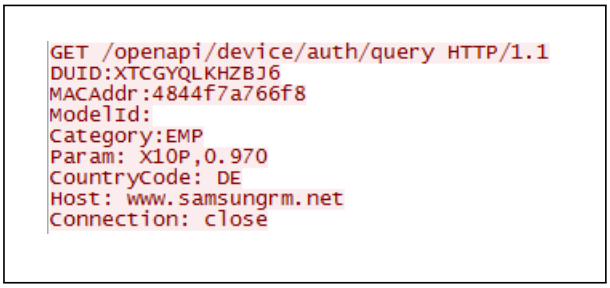

Abbildung 6: „Wireshark“ Ausschnitt mit Header Feldern der „DUID“ und MAC Adresse

Zusätzlicher Test der Header Felder

Auf der Grundlage dieses Ergebnisses wurden noch zwei weitere Tests durchgeführt. Hierbei sollte zum einen festgestellt werden, wie das Header Feld MAC Adresse sich verändert, wenn man anstelle des LAN Adapters die WLAN Schnittstelle des Fernsehers verwendet. Im zweiten Test sollte festgestellt werden, wie sich die Anfrage verhält, bei der Änderungen der einzelnen Header Felder in der Anfrage. D.h. es sollte festgestellt werden, welche Header Felder nötig sind um eine valide Anfrage abzusenden.

Der Testaufbau, dargestellt in Abbildung 7, für den ersten Test wurde dahingehend verändert, dass das Smart TV mit dem Gateway über WLAN kommuniziert. Dafür wurde der Test Laptop verwendet um die Messung des Traffics durchzuführen. Die Kommunikation zwischen Smart TV und Test Laptop wurde unverschlüsselt aufgebaut.

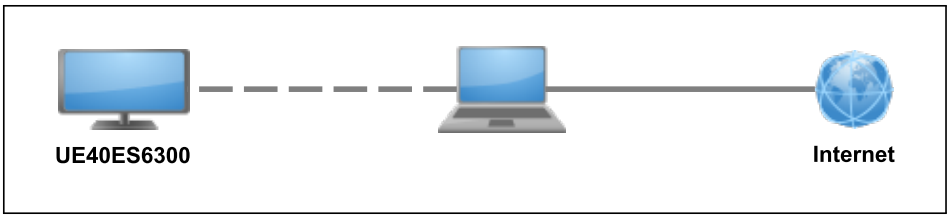

Abbildung 7: Veränderter Aufbau Testaufbau des Starttests

Es konnte festgestellt werden, obwohl das Smart TV mit der WLAN Schnittstelle konfiguriert ist, wird ebenfalls die MAC Adresse der LAN Schnittstelle übertragen. Auch kann ausgeschlossen werden, dass beide Schnittstellen über dieselbe MAC Adresse verfügen.

Minimierung der Header Felder

Der zweite Test wurde mit dem in Figure 2 vorgestellten Testaufbau durchgeführt. Hierfür wurde der HTTP Request des Smart TVs nachgebaut und abgesendet. Dies diente der Überprüfung, ob tatsächlich nur die Header Felder entscheidend für eine valide Anfrage sind. Nachdem dies erfolgreich verifiziert werden konnte, wurden MAC Adresse und „DUID“ erst auf einen ungültigen Wert geändert und zuletzt einfach weggelassen.

Als relevantes Feld wurde das Header Feld Category identifiziert. Die beiden sensiblen Header Felder „MACAddr“ und „DUID“ spielen für den eigentlichen Request keine Rolle und können auch weggelassen werden. Eine Veränderung der MAC Adresse im Header Bereich bzw. das Einfügen einer vermeintlich falschen „DUID“, hat keine Auswirkung auf die Antwort der Anfrage. Daraus lässt sich schließen, dass die Header Felder in der Anfrage minimiert werden können.

HTTPS Verbindungen

Neben den oben beschriebenen HTTP Anfragen, werden insgesamt zwei Kommunikationspartner per HTTPS angesprochen. Eine der IP Adresse ist dabei persistent, d.h. an ihr wurde keine Änderung in allen Tests festgestellt. Diese Adresse lautet **„210.118.88.200“**. Dabei handelt es sich um einen Server mit dem Standort **Süd Korea** und wird dem Unternehmen Samsung selbst zugeordnet. Auch der zweite Server kann identifiziert werden. Die IP Adresse dieses Servers lautet **„122.248.249.97“**. Er gehört dem Cloud Service Provider Amazon Cloud EC2 [2]. Als Standort dieses Servers konnte **Kuala Lumpur** ausgemacht werden. Dabei werden zu diesem Server zusätzlich auch HTTP Verbindungen aufgebaut, welche wie oben beschrieben, vor allem XML Dateien beziehen.

Welche Daten in den HTTPS Anfragen übertragen werden, konnte nicht festgestellt werden. In Kapitel Sonstiges am Ende dieser Arbeit wird das Thema der HTTPS Verbindungen nochmals aufgegriffen.

4.1.2 Risiko

Im nun folgenden Abschnitt, soll auf die Risiken eingegangen werden, welche durch die Übertragung per HTTP entstehen und zusätzlich die Datenschutzproblematik zur Übermittlung der „DUID“ und MAC Adresse erläutert werden. Im Test stellte sich heraus das zahlreiche HTTP Verbindung beim Start des Fernsehers zu verschiedensten Servern aufgebaut werden. Diese sind dafür bestimmt, Update Informationen über die aktuelle Firmware des Fernsehers und die drauf installierten Applikationen zu beziehen bzw. XML Dateien anzufordern um den Smart Hub des Fernsehers aktuell zu halten.

Durch die fehlende Übertragung der Daten über HTTP und fehlenden Signaturen der XML Dateien, kann das Smart TV Geräte zu keiner Zeit die Integrität bzw. Authentizität der übermittelten Daten prüfen. Daher ist es nicht möglich, veränderte oder modifizierte Inhalte zu entdecken bzw. zu verifizieren, ob die erhaltenen Daten tatsächlich vom Server stammen. Das Schutzziel der Integrität ist essentiell für die Sicherheit der eingesetzten Software auf dem Fernseher. Durch eine Modifikation der Daten ist es möglich, die Inhalte auf dem Fernseher zu ändern. Ohne Integritätsprüfung kann ein Angreifer die Inhalte verändern ohne dass der Nutzer davon Kenntnis erlangt. Wie ein solches Vorgehen umgesetzt werden kann und welche Möglichkeiten für einen Angreifer dadurch entstehen, wird in Kapitel 4.2.2 in einem Angriffsszenario näher beschrieben.

Übermittlung „DUID" und MAC Adresse

Eine Anfrage, die bei jedem Start des Smart TVs ausgeführt wird, enthält die „DUID" und die MAC Adresse des physikalischen LAN Adapters des Smart TVs. Bei der „DUID" handelt es sich laut Samsung um eine eindeutige Kennung welche eindeutig einem Smart TV zugeordnet ist. D.h. zwischen „DUID" und Smart TV besteht eine 1:1 Beziehung [42]. Dies wird dadurch bestätigt, dass die „DUID" anhand der MAC Adresse bestimmt werden kann. Diese sind eindeutig für ein Netzwerkgerät weltweit vergeben.

Da diese Informationen bei jedem Start des Smart TVs übermittelt werden, ergibt sich daraus ein Datenschutz relevantes Risiko. Durch die Eindeutigkeit der Beziehung zwischen Smart TV und „DUID" kann der entsprechende Server, welche die Anfrage bearbeitet, genau nachvollziehen wann ein entsprechendes Smart TV Gerät eingeschaltet wird. Wie sich in Kapitel 4.2.1 herausstellte, werden diese Pakete auch beim Start des Smart Hub übertragen. Dadurch ist es der Gegenstelle möglich das Nutzungsverhalten vom Einschaltzeitpunkt des Smart TVs und der Nutzung des Smart Hubs zu erstellen. Durch die Kombination der benutzten IP Adresse und der „DUID" lässt sich der aktuelle Standort des Fernsehgerätes feststellen.

Zusätzlich dazu, dass die Daten, wie oben beschrieben, über das unsichere HTTP Protokoll übertragen werden, ist es jedem Knotenpunkt zwischen dem Smart TV und dem Server möglich die Daten mitzulesen. Dadurch können die Nutzungsstatistiken nicht nur von der Gegenstelle selbst, sondern auch von allen Punkten dazwischen, erstellt werden.

4.2 Lösung

Der letzte Abschnitt des ersten Tests, gibt Lösungsmöglichkeiten für die gefunden Probleme im Starttest an. Dabei soll auf das Problem der Übertragung der XML Dateien via HTTP eingegangen werden. Des Weiteren soll ein Lösungsvorschlag für die Übertragung der MAC Adresse und „DUID" gegeben werden.

Durch die fehlende Möglichkeit die Integrität der Dateien zu prüfen, welche einerseits auf die Übertragung der Daten via HTTP zurückzuführen ist, andererseits auf die fehlende Signatur innerhalb der XML Dateien, ergibt sich ein relevantes Angriffsszenario welches in Kapitel 4.2.2 vorgestellt wird. Damit die Problematik behoben werden kann, müssen die Dateien Signaturen erhalten. Hierfür geeignet sind sogenannte Digitale Signaturen [13].

Digitale Signaturen ermöglichen es, Dateien elektronisch zu signieren. Dadurch kann die Herkunft der Datei eindeutig verifiziert werden, da nur der Aussteller der Datei diese signieren kann. Das Verfahren beruht dabei auf einem Asymmetrischen Schlüsselpaar. Dieses Paar besteht aus einem privaten Schlüssel und einem öffentlichen, welcher aus dem geheimen Teil berechnet werden kann. Die Gegenrichtung, die Berechnung von privatem Teil anhand des öffentlichen Schlüssels, ist bei sicheren Verfahren nicht möglich bzw. sehr schwer [25]. Der geheime private Schlüssel wird zum Signieren der Daten genutzt, wohingegen der öffentliche Schlüssel zum Überprüfen der Signatur benutzt wird und frei verfügbar sein kann.

Für den Fall auf dem Testgerät würde es bedeuten, dass Samsung auf allen Geräten den öffentlichen Schlüssel verteilt. Werden XML Dateien über das Netzwerk übertragen, werden diese mit dem auf dem Samsung Server gespeicherten privaten Schlüssel signiert. Der Fernseher kann anhand des öffentlichen Schlüssels die Integrität und Authentizität der Dateien überprüfen. Ist die Signatur nicht echt, können so Integritätsfehler entdeckt werden. Eine Modifizierung der Daten kann damit unterbunden und ein Angriffsszenario, wie in Kapitel 4.2.2 beschrieben, abgewehrt werden. Ein Lösungsvorschlag ist in Grafik 8 gegeben.

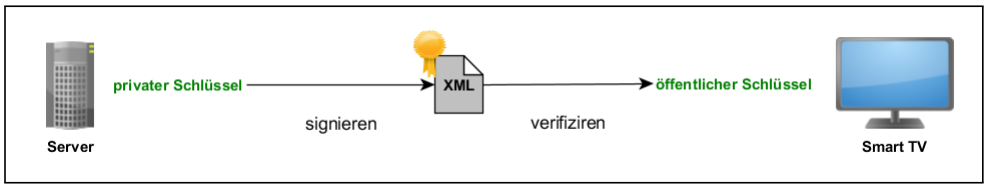

Abbildung 8: Signaturverfahren zwischen Smart TV und Server

Minimierung der Header Felder

Aus der Übertragung der MAC Adresse und „DUID" bei jedem Start des Smart TVs und öffnen des Smart Hub ergab sich aus Kapitel 4.2.1 eine Datenschutzproblematik, da sensible Daten über einen unverschlüsselten Kanal übertragen werden. Anhand dieser Daten kann ein direktes Tracking des Nutzungsverhaltens erstellt werden.

Solche Daten sollten zu keinem Zeitpunkt übertragen werden. Weder unverschlüsselt noch verschlüsselt. Zudem stellte sich in einem Test heraus, dass die Daten vom Server nicht benötigt werden um die Anfrage bearbeiten zu können. Deshalb ist die einfachste und sinnvollste Lösung, die Abfrage der Softwareupdates ohne sensible Daten durchzuführen.

Dennoch stellt sich die Frage, wieso Pakete beim Einschalten des Smart TVs übertragen werden. Die Übertragung der Datenpakete im Hintergrund, verletzt das mentale Model des Nutzers. Dies beschreibt das Bewusstsein des Nutzers dafür,

dass wenn Inhalte eines Providers genutzt werden, kann dieser auch feststellen, dass ich diesen Dienst derzeit nutze. Am Beispiel des Webbrowsers bedeutet das, besucht ein Nutzer eine Webseite, weiß er, dass Daten von ihm an den Betreiber der Seite weitergeleitet werden. Der Nutzer stimmt dieser Nutzung der Daten implizit zu. Möchte er dies nicht, kann er selbst bestimmten, ob er diese Seite nicht mehr besucht.

Im Falle des Smart TVs werden die Daten für die Nutzung der Samsung Dienste von den Servern bezogen, ohne dass ein Dienst wirklich genutzt wird. Das bedeutet, dass schon im Vorfeld Daten übertragen werden, ohne dass der Nutzer darüber informiert wird. Dies ist eine klare Verletzung des mentalen Modells und der Privatsphäre.

Damit dies nicht verletzt wird, sollten die Daten erst angefordert werden, wenn sie wirklich gebraucht werden. Dies passiert erst, wenn der Nutzer den Smart Hub aktiviert. Im normalen Betrieb des Smart TVs, z.B. das schauen von Fernsehsendungen, werden keinerlei Daten von Samsung benötigt. Deshalb sollte eine Implementierung in Erwägung gezogen werden, bei der die Daten erst beim Start des Smart Hubs übertragen werden.

Ein weiteres Beispiel für die Verletzung des mentalen Modells wird in Kapitel 4.3.1 vorgestellt.

4.3 Smart Features

In einer Erweiterung zum ersten Test, wurde der Traffic aufgezeichnet, welcher durch die Nutzung der sogenannten Smart Features entsteht. Anschließend wurde die Aufzeichnung analysiert und ausgewertet.

4.3.1 Analyse

Eine Sammlung von Smart TV Applikationen bzw. „Smart Apps“ (im Folgenden Applikationen) die man auf Smart TV benutzen kann, findet man bei Samsung Geräten innerhalb des sogenannten Smart Hubs [37]. Diese Anwendung ist der zentrale Punkt für alle Smart Features des Smart TVs. Das Starten des Smart Hubs erfolgt über einen Tastendruck auf der Fernbedienung, über das Smart Hub Symbol (Vgl. Abbildung 9). Ist das Gerät mit dem Internet verbunden erscheint nach kurzer Ladezeit der Smart Hub, als eigenständige Applikation. Ist eine Aktualisierung des Smart Hubs verfügbar, wird dieser hier eingeblendet. Besteht keine Verbindung zum Internet, wird eine Fehlermeldung ausgegeben und der Nutzer dazu aufgefordert seine Einstellungen zu überprüfen bzw. die Verbindung neu zu konfigurieren.

Smart Hub

Alle momentan installierten Applikationen sind auf dieser Oberfläche verknüpft. Diese erinnert stark an die von aktuellen Smartphone- oder Tabletgeräten. Installierte Anwendungen werden in einer Art Kachel-Ansicht auf dem Bildschirm abgelegt und können dort vom Nutzer frei angeordnet werden. Dabei erfolgt die Steuerung über die mitgelieferte Fernbedienung. Des Weiteren besteht die Möglichkeit eine Maus bzw. Tastatur an das Smart TV Gerät anzuschließen. Auch Bluetooth Peripherie wird von dem eingesetzten Testgerät unterstützt. Neben zahlreichen eigenen Applikationen von Samsung sind auch standardmäßig Programme von Drittanbietern installiert. Unter diesen finden sich bekannte Applikationen wie die der Software Skype oder das Client Programm für das soziale Netzwerk Facebook. Aber auch zahlreiche eher unbekannte Applikationen sind vertreten. Alle Smart Features werden hier zentral verwaltet und können ggf. auch deinstalliert werden. Neue Applikationen können über den von Samsung bereitgestellten App-Store erweitert werden. Durch regelmäßige Updates der Smart TV Firmware werden neue Applikationen, auch von Drittanbietern, automatisch installiert bzw. deinstalliert. Außerdem wird dem Nutzer die Möglichkeit gegeben, eigene Nutzer Accounts für die Oberfläche anzulegen. Diese verfügen über separate Oberflächen, individuell von jedem Nutzer angepasst.

Zusätzlich zu den Paketquellen von Samsung, gibt es die Möglichkeit über einen Developer Account beliebige Applikationen zu installieren. Dieser ist dafür gedacht, eigene Entwicklungen zu Testen. Dennoch ermöglicht es beliebige Software zu installieren. Durch die Angabe eines Link werden die entsprechenden Dateien heruntergeladen und auf dem Smart TV Gerät installiert. Der Developer Account kann mit dem Benutzernamen „develop“ und einem beliebigen sechsstelligen nummerischen Passwort aktiviert werden. Das Vorgehen wird offiziell von Samsung erklärt [41].

Abbildung 9: Aktuelles Smart Hub Logo

Neben den Icons der Applikationen auf der Oberfläche des Smart Hub gibt es auch variable Bereiche, in denen Werbung oder neue Inhalte präsentiert werden. Diese Live-Inhalte werden mit Informationen aus den in Kapitel 4.1.1 beschriebenen XML Daten gefüllt. Die XML Dateien enthalten Links in denen die einzelnen Kacheln beschrieben sind, jeweils mit Text und wahlweise einem Medium dass eingefügt werden soll. Als Inhalt werden meist externe Grafiken oder Flash Inhalte geladen. Zusätzlich hierzu kann eine Aktion definiert werden, welche beim einen Klick auf das Symbol ausgeführt wird. Samsung nutzt diese Fläche um Neues zu präsentieren oder Hinweis auf bestimmte Inhalte in anderen Applikationen zu geben.

Viele Applikationen können nur im Online Modus, d.h. wenn das Smart TV erfolgreich mit dem Internet verbunden ist, ausgeführt werden. Dennoch kann der Smart Hub auch im Offline Modus betrieben werden. Lediglich eine kryptische

Fehlermeldung mit der Möglichkeit der Konfiguration macht den Nutzer darauf aufmerksam. Der Funktionsumfang im Offline Modus ist dennoch äußert gering.

Test des integrierten Browsers

Als Testapplikation wurde speziell der integrierte Browser untersucht. Bei dem eingesetzten Browser (nur im Testgerät vorhanden nicht im Referenzgerät) handelt es sich scheinbar um eine Eigenentwicklung seitens Samsung [41]. Der Browser in unserem Test trägt die Versionsnummer *VER.1207101*. Die festgestellten Ergebnisse konnten noch bei der aktuellen Version *VER.1210271* festgestellt werden. Als User-Agent verwendet der Browser die Kennung Mozilla 5.0, wobei keinerlei Ähnlichkeiten mit dem Browser Firefox von Mozilla festgestellt werden konnte. Dieser unterstützt alle gängigen Webinhalte, wie z.B. Adobe Flash. Beim Aufruf diverser Webseiten konnten keine Fehler bzgl. der Darstellung oder Anordnung der Seiten festgestellt werden. Cookies werden standardmäßig angenommen, auch von Drittanbieter. Eine Funktion um Cookies zu verwalten bzw. die Regulierung der Cookies konnte nicht gefunden werden. Dennoch wird diese in einer ersten Datenschutzerklärung erklärt.

Beim Aufrufen von HTTPS Seiten wurde festgestellt, dass alle Seiten - mit gültigem und auch ungültigem Zertifikat - ohne Prüfung als vertrauenswürdig angezeigt werden. Der Browser kennzeichnet alle HTTPS Seiten am rechten Rand der Adressleiste mit einem goldenen Schloss. Aktuelle Browser klassifizieren hierbei HTTPS Seiten je nach ihrer Zertifizierungsautorität, deren Gültigkeit und der verwendeten TLS/SSL Version [13].

Die Sicherheitsprüfung der Zertifikate von HTTPS Seiten unterstützt der integrierte Browser im Smart TV nicht. Hier wird lediglich zwischen HTTP bzw. HTTPS Seiten unterschieden.

4.3.2 Risiko

In diesem Kapitel werden die aus dem zweiten Test gefundenen Risiken untersucht. Dabei wird das Kapitel in insgesamt drei Abschnitte unterteilt. Im ersten werden die Risiken analysiert, welche durch die fehlende Integritätsprüfung der Übertragung der XML Dateien zur Veränderung der Smart Hub Oberfläche entstehen. Danach folgt eine Erklärung der Risiken durch die nicht vorhandene Prüfung der Zertifikate von HTTPS Seiten. Am Ende des Kapitels werden die beiden beschriebenen Probleme kombiniert und ein Angriffsszenario wird präsentiert.

Durch die Übertragung von Daten über das Netz per HTTP, werden die Pakete unverschlüsselt übertragen. Sowohl die Anfragen an den Server, als auch die Antwort des Servers, werden dabei im Klartext versendet. Hat ein Angreifer Zugriff auf das Netz, kann er alle übertragenen Daten mitlesen. Dabei kann der Angreifer übertragene Daten nicht nur mitlesen, sondern auch verändern, löschen oder neue Daten hinzufügen. In den folgenden Abschnitten nehmen wir an, dass der Angreifer vollen Zugriff auf das Netz hat.

Ein Angreifer hat nun die Möglichkeit Inhalte welche für den Smart Hub bestimmt sind, zu modifizieren. Speziell wollen wir hierbei die XML Dateien hervorheben, welche zur Gestaltung des Smart Hubs verwendet werden. Wie im vorherigen Abschnitt beschrieben, werden hier Informationen und Quellen angegeben um bestimmte Kacheln auf der Oberfläche zu gestalten. Die Dateien werden jeweils beim Start des Fernsehers und beim Öffnen des Smart Hubs angefordert. Die übertragenen Ressourcen tragen jeweils den gleichen Namen und können somit in unterschiedlichen Testläufen des Smart TVs identifiziert werden. Außerdem besitzen die XML Dateien keinerlei Signatur oder Prüfsumme. Dadurch können Änderungen an der XML Dateien vom Smart TVs nicht erkannt werden. Dieses Problem wurde inklusive Lösung in Kapitel 4.2.2 und 4.2.3 behandelt. Der Inhalt der XML Dateien kann vom Angreifer so modifiziert werden, dass neue gefälschte Inhalte auf der Smart Hub Oberfläche angezeigt werden.

Fehlende Überprüfung von HTTPS Zertifikaten

Bei der zweiten Sicherheitslücke handelt es sich um die fehlende Prüfung von Zertifikaten auf HTTPS Seiten. HTTP via TLS/SSL, kurz HTTPS, hat die Schutzziele Vertraulichkeit, Integrität und Authentizität. Aktuelle Browser basieren auf der TLS Version 1.1 bzw. 1.2. Zusätzlich zu den Verschlüsselungsverfahren die das TLS/SSL Protokoll unterstützt, werden Zertifikate für die Authentifizierung verwendet.

Ohne diese besteht die Möglichkeit von „Man-In-The-Middle Attacken“, da der Schlüsselaustausch abgehört und damit die TLS/SSL Verbindung entschlüsselt werden kann. Hierbei ist der Angreifer zwischen den beiden Kommunikationspartnern und hört die Verbindung mit dem Start des Verbindungsaufbaus ab. Hier kann er die nötigen Informationen für den verwendeten Schlüssel erhalten und damit die Kommunikationsverschlüsselung brechen.[46]

Generell soll ein solches Zertifikat die Identität des Kommunikationspartners bestätigen. Durch den Besitz des Zertifikates kann die eigene Identität bestätigt werden.

Bei den ausgestellten Zertifikaten unterscheidet man zwischen selbst erstellten oder signierten Zertifikaten einer vertrauenswürdigen „Trusted Third Party“ (TTP). Da die Möglichkeit besteht, HTTPS Zertifikate selbst auszustellen, können HTTPS Seiten nicht immer vertraut werden. Deshalb werden EV-SSL (Extended Validation) [9] Zertifikate von dritten Parteien ausgestellt. Diese haben den Hintergrund, die Identität und den Betreiber der HTTPS Seite sicherzustellen.

Aktuelle Browser führen in der Regel die Prüfung der HTTPS Zertifikate automatisch beim Aufruf einer HTTPS Seite durch. Dabei wird sowohl die verwendete TLS/SSL Version und das mit übermittelte Zertifikat auf seine Gültigkeit und Ausstellung geprüft. Sind alle Informationen korrekt so wird die HTTPs Seite farblich für den Nutzer hervorgehoben. Eine besondere Hervorhebung erhalten Webseiten mit dem erweiterten EV-SSL Zertifikat. Dabei werden Webseiten mit ungültigem Zertifikat (abgelaufen oder selbst ausgestellt) oder veralteten TLS/SSL Version nicht aufgerufen. Hierbei wird der Nutzer auf eine Warnseite geleitet und vor der weiteren Nutzung der Webseite muss der Nutzer explizit fortsetzen (Beispiel Grafik 10).

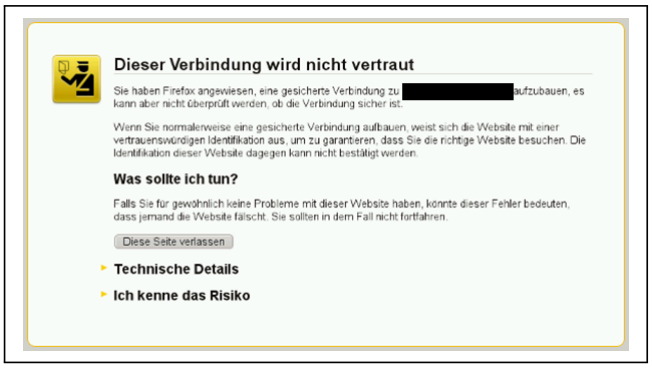

Abbildung 10: Aktuelle Warnmeldung im Browser (Firefox)

All die oben genannten Prüfungen bzw. Warnungen besitzt der integrierte Browser in unserem Testgerät nicht. Die bedeutet, dass der Nutzer nicht vor etwaig gefälschten oder nicht vertrauenswürdigen HTTPS Webseiten gewarnt wird. Dies birgt das Risiko von sogenannten „Phishing Attacken".

Phishing Attacken

Allgemein versteht man unter einer „Phishing Attacke", den Diebstahl der Identität des Nutzers. Dies versucht ein Angreifer durch gefälschte Inhalte im Internet, Webseiten oder E-Mails, zu erreichen, indem er das Opfer so täuscht, dass dieses sensible Daten an den Angreifer übermittelt. Typisch für solche Angriffe ist, dass der Angreifer mit der Ähnlichkeit der Inhalte das Vertrauen des Opfers gewinnen möchte. Hierfür wird die z.B. die Seite so detailliert wie möglich nachgebaut, Firmeneigene Logos werden eingefügt und das Design wird übernommen.

Auch durch die Bekämpfung von Phishing in aktuellen Browsern und E-Mail Client Programmen, z.B. Microsoft Outlook [27], sind die Schäden die durch Phishing entstehen immer noch enorm. Das Phishing ein aktuelles Problem ist, zeigt auch ein Quartalsbericht, vom zweiten Quartal 2012, der „Anti Phishing Working Group" [4]. Dabei erreichten die Phishing Seiten im April 2012 ein Spitzenwert. Insgesamt 175.229 Seiten wurden von der „Anti Phishing Working Group" als Phishing Seiten identifiziert. Weiter wird beschrieben, das der Finanzsektor der am stärksten betroffene Bereich ist [5].

Damit eine solche Phishing Attacke erfolgreich durchgeführt werden kann, muss das Opfer eine vom Angreifer präparierte Seite aufrufen und darauf seine sensiblen bzw. geheimen Daten freigeben. Dies geschieht durch gefälschte E-Mails mit einem vorbereiteten Link auf eine nicht vertrauenswürdige Seite oder durch Links auf Webseiten. Seltener gibt der Nutzer die URL selbst in den Browser ein.

kombiniertes Angriffsszenario

Im letzten Teil dieses Kapitels werden beide Lücken kombiniert und ein Angriffsszenario aufgezeigt. Ziel ist es, zu zeigen, dass es ohne komplexe Maßnahmen möglich ist mit Hilfe der beiden gefundenen Lücken, den Nutzer auf eine vorbereitete Seite zu leiten auf der er sensible Daten herausgeben soll. Unterstützend wird das Protokoll in den Grafiken 11-14 verdeutlicht.

Im ersten Schritt (Abbildung 12) wird die fehlende Prüfung auf Integrität der XML Dateien ausgenutzt um eine Kachel auf der Smart Hub Oberfläche so zu modifizieren, dass sie den Nutzer auf eine gefälschte Webseite leitet. Nachdem das vermeintliche Opfer den Smart Hub startet, beginnt der Fernseher damit Verbindungen aufzubauen um den Smart Hub zu aktualisieren. Hierbei bezieht er auch XML Dateien um die Inhalte auf der Oberfläche zu gestalten. Bei einer Anfrage des Smart TVs (Abbildung 13), wird die XML Datei vom Angreifer durch seine modifizierte XML Datei ausgetauscht. Dabei wird das Format der Datei nicht verändert, lediglich der Eintrag für eine bestimmte Kachel. Das Resultat hierbei ist, dass eine Kachel auf dem Smart Hub des Nutzers modifiziert wurde, ohne dass dies vom Fernseher bzw. vom Nutzer entdeckt wurde. Der Inhalt auf der Kachel kann vom Angreifer frei gestaltet werden. Ziel ist es, dass der Nutzer mit großer

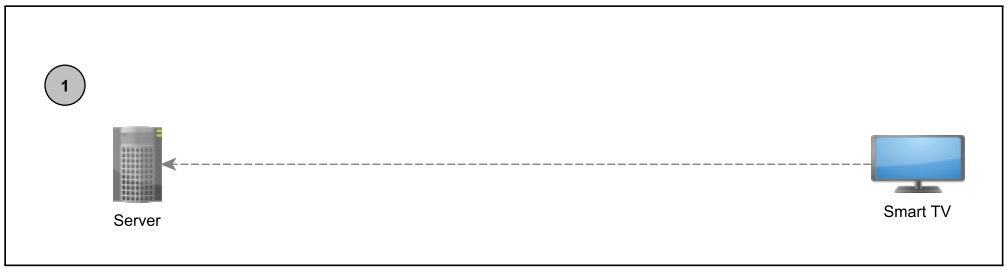

Abbildung 11: Protokollschritt 1

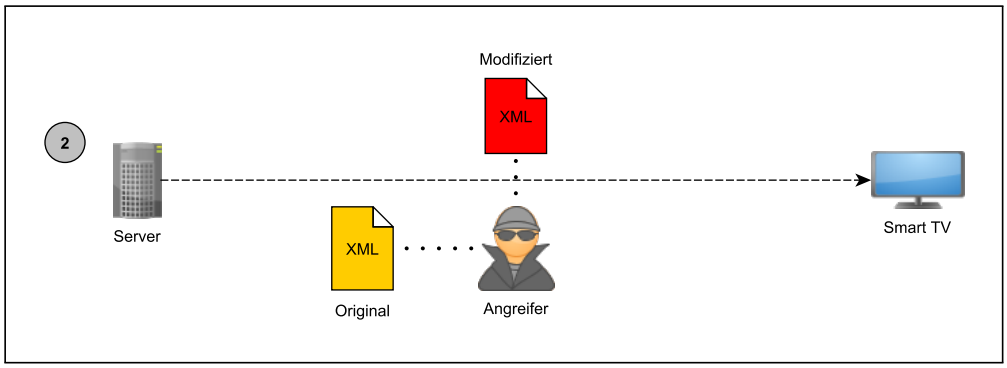

Abbildung 12: Protokollschritt 2

Wahrscheinlichkeit auf diese Kachel navigiert und den dahinter versteckten Link im Web Browser prüft. Ein Beispiel wäre hierfür z.B. einen Hinweis auf ein Gewinnspiel oder ein Logo der Webseite einer Bank. Navigiert nun das Opfer auf diese HTTPS Seite öffnet sich der gefälschte Inhalt des Angreifers (Abbildung 14).

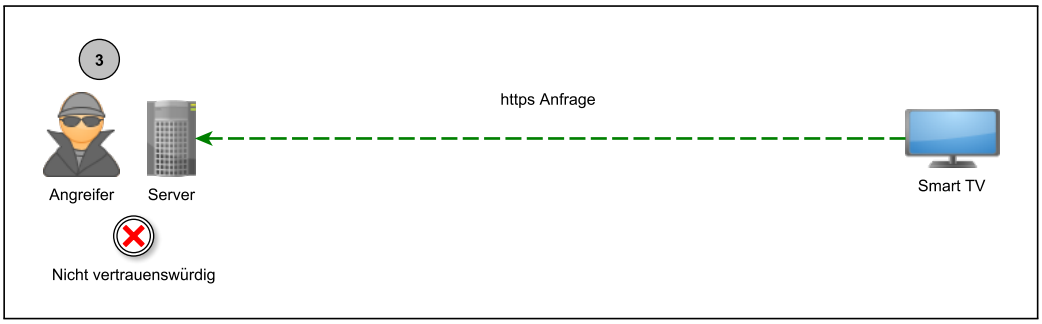

Abbildung 13: Protokollschritt 3

Die Webseite des Angreifers ist nicht vertrauenswürdig, da das Zertifikat selbst ausgestellt wurde. Aktuelle Webbrowser würden nachdrücklich vor dem Besuch der nachfolgenden Webseite warnen und den Nutzer durch explizites anklicken fortfahren lassen. Die HTTPS Anfrage wird mit der Webseite ohne gültiges Zertifikat bestätigt (Abbildung 15).

Auf dem Fernseher wird die HTTPS als Vertrauenswürdig eingestuft und erhält ein kleines goldenes Schloss an der Adressleiste. Für den Nutzer sieht es aus, als ob es sich bei Seite um eine vertrauenswürdige Webseite handelt. Der Angreifer, welcher die Seite betreibt, fordert das Opfer nun auf, sensible oder geheime Daten einzugeben. Hierbei können verschiedenste Szenarien eintreten. Handelt es sich um das Beispiel des Online Bankings, könnte die Webseite vortäuschen, dass zur Bestätigung des Banking Accounts auf dem Smart TV, TAN Nummern an die Webseite geschickt werden müssen. Hierbei wird der Nutzer aufgefordert eine Anzahl von geheimen TAN Nummer einzugeben und diese abzuschicken. Nehmen wir das Beispiel des Gewinnspiels, könnte der Nutzer aufgefordert werden seine Kreditkarteninformationen einzugeben, um den vermeintlichen Gewinn zu erhalten. Nach der Übermittlung der Daten, ist der Angreifer nun im Besitz von den sensiblen Daten des Nutzers.

Dieses Beispiel sollte zeigen, wie die beiden Sicherheitslücken kombiniert werden können, um eine erfolgreiche Phishing Attacke auf dem Smart TV auszuführen.

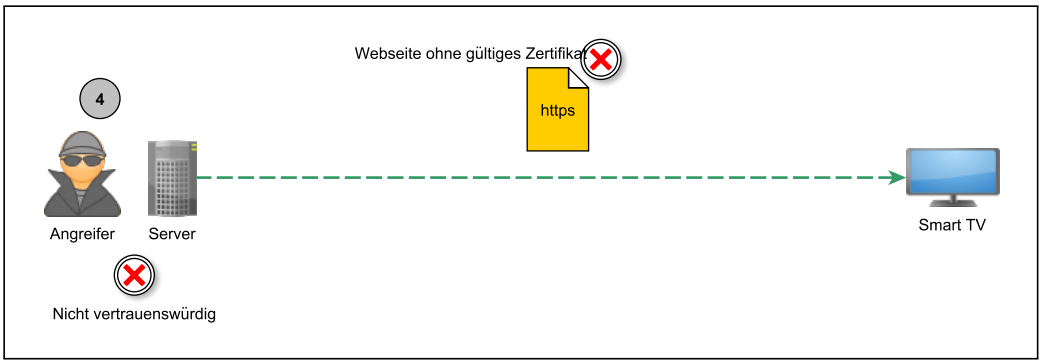

Abbildung 14: Protokollschritt 4

4.3.3 Lösung

In diesem Abschnitt sollen die Lösungsmöglichkeiten für die im Smart Hub Test festgestellten Lücken vorgestellt werden. Es soll außerdem untersucht werden, ob die Risiken damit beseitigt werden können.

Die Übertragung der XML Dateien über das unsichere HTTP Protokoll, diente im Angriffsszenario zuvor als Einstiegspunkt für die durchgeführte Phishing Attacke. Damit solche Angriffe vermieden werden können muss das Smart TV sicher sein, dass er sich bei den Dateien um die angeforderten Ressourcen handelt und nicht um ausgetauschte Daten. Damit dies erfüllt werden kann muss das Übertragungsprotokoll die Schutzziele Integrität und Authentizität erfüllen. Wie dieses Problem gelöst werden kann, wurde in Kapitel 4.2.3 durch die Einführung von digitalen Signaturen gezeigt.

Überprüfung der HTTPS Zertifikate

Das zweite Problem welches identifiziert wurde, ist die fehlende Zertifikatsprüfung im integrierten Browser. Diesem fehlt eine Prüfung der Zertifikate von HTTPS Seiten, um diese nach ihrer Vertrauenswürdigkeit einzustufen. Diese Klassifizierung anhand der SSL Zertifikate fehlt dem Browser komplett. Aktuelle Browser klassifizieren die HTTPS Seiten und warnen (Vgl. Abbildung 10) den Nutzer hier vor.

Eine Möglichkeit wäre, diese Klassifizierung von aktuellen Browser zu übernommen. Dennoch besteht zusätzlich die Gefahr, dass die vermeintliche Eigenentwicklung des Browsers, neben der gefundenen Schwachstellen noch zusätzliche, in diesem Test nicht behandelt, besitzt. Mittlerweile existieren erprobte Browser für Smart TV Geräte. Ein Beispiel hierfür ist der vom Desktop bekannte Browser Opera, welcher für aktuelle Smart TVs verfügbar ist [12]. Hier kann davon ausgegangen werden, dass er aktuellen Sicherheitsstandards entspricht.

4.4 HbbTV

Kapitel 4.3 beschreibt eine gefundene Schwachstelle im Datenschutz des HbbTV Standards. Dieser gilt als Nachfolger des aktuell eingesetzten Videotextes und wird von den meisten Fernsehsendern mittlerweile implementiert. Dabei handelt es sich um keine Schwachstelle am System des Testgerätes der Firma Samsung.

4.4.1 Analyse

Im dritten Test wird die Testreihe der ersten beiden Tests erweitert. Daher wurde der Testaufbau leicht verändert, indem ein Fernsehsignal hinzugefügt wurde. Der veränderte Aufbau kann Figure 15 entnommen werden. Als Signalgeber für das Fernsehsignal wurde eine DBV-T Antenne genutzt. Ziel des Tests war es, Unterschiede zwischen dem Traffic ohne und mit Fernsehsignal zu entdecken. Hierbei wurden die Ergebnisse aus den ersten beiden Tests, Kapitel 4.1 und 4.2 als Vergleichswerte genutzt. Auch dieser Test wurde aus Gründen der besseren Testmöglichkeit über die LAN Schnittstelle durchgeführt.

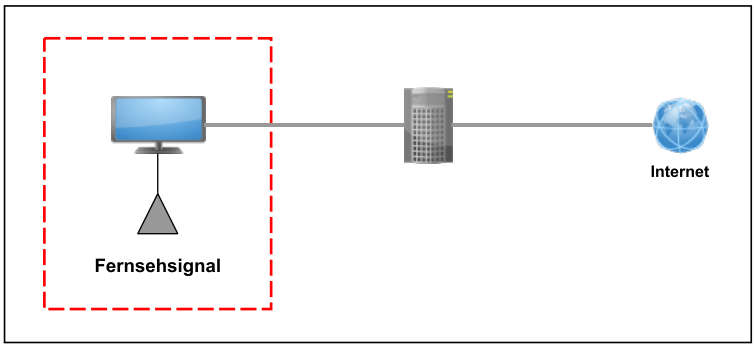

Abbildung 15: Veränderter Testaufbau mit Fernsehsignal

Bevor das Smart TV Fernsehsender empfängt, müssen diese hinzugefügt werden. Hierfür gibt es zwei Varianten. Samsung unterscheidet hier zwischen der manuellen Einrichtung der Sender oder der automatischen Suche auf allen Frequenzen des Empfängers. Für diesen Test wurde der automatische Suchlauf gewählt. Der in Kapitel 4.1.1 beschriebene Starttest wurde wiederholt. Das Smart TV Gerät wurde eingeschaltet. Der Test terminiert nach 120 Sekunden, diesmal auf einem Fernsehsender. Welche Sender dies ist, spielt für den Test keine Rolle. Die ersten Pakete beim Start entsprechen genau den Paketen, welche im ersten Test auch beobachtet werden konnten. Zusätzlich zu diesem Traffic konnte auch neue Verbindungen und Pakete entdeckt werden, welche zu neuen, bisher unbekannten, Servern aufgebaut wurden. Anhand der IP Adresse und den Änderungen auf dem Bildschirm wurden diese Pakete dem Datendienst HbbTV zugeordnet.

Datenanalyse des HbbTV Datendienstes

Hybrid Broadcast Broadband TV oder kurz HbbTV (im Folgenden HbbTV) ist ein kostenloser Datendienststandard, welcher von den meisten Fernsehsender implementiert wird [22]. Den HbbTV Standard gibt es aktuell in der Version 1.2.1 (November 2012) [24]. ARD und ZDF waren die ersten Sender, welche 2009 auf der Internationalen Funkausstellung in Berlin erste Implementierungen der neuen Technik vorstellten [7].

HbbTV bietet die Möglichkeit zusätzliche Informationen auf dem Smart TV Gerät anzeigen zu lassen. Ähnlich zum eingesetzten Videotext, kann der Datendienst Informationen zum aktuellen Fernsehprogramm liefern. Dennoch bietet HbbTV viel mehr als der klassische Videotext. Durch die Möglichkeit von hochauflösenden Oberflächen, bei aktiver Internetanbindung, können die Sender ihre Inhalte frei gestalten. Dabei handelt es sich bei den HbbTV Oberflächen um Weboberflächen. Den Sendern bleibt überlassen was sie implementieren, der Standard legt lediglich einen Rahmen fest. Der Datendienst kann so gestaltet werden, dass er in das Fernsehprogramm integriert wird (Vgl. Bild 16) oder als vollständige Oberfläche im Vollbildmodus. Die Inhalte können dynamisch angepasst werden. Viele Sender bieten mittlerweile ihre „Mediathek“ über HbbTV an.

Auf den meisten Smart TV Geräten ist der Datendienst verfügbar, wobei das Angebot mancher Sender zusätzlich den Besitzt von Smart Card Modulen voraussetzt bzw. spezielle Receiver benötigen. Eine vollständige Liste der HbbTV Datendienste wird auf der offiziellen HbbTV Seite angeboten [23], inklusive der Verfügbarkeit.

Standardmäßig ist der HbbTV Datendienst auf dem Testgerät deaktiviert. In den Menüeinstellungen kann dieser manuell aktiviert werden. Gelangt man danach auf einen Fernsehsender, welcher HbbTV unterstützt, erscheint am rechten unteren Bildrand, nach einer kurzen Wartezeit, eine rote Schaltfläche, welche über die Fernbedienung aktiviert werden kann. Nach dem drücken des sogenannten „Red Buttons“ (Vgl. Bild 17), gelangt man in die vom Sender gestalteten

Abbildung 16: HbbTV Beispiel ARD

HbbTV Oberfläche. Wird der „Red Button“ nicht gedrückt, verschwindet die Schaltfläche nach kurzer Zeit, eine Aktivierung ist dennoch über die Fernbedienung jederzeit möglich.

In den Tests konnte festgestellt werden, dass einzelne Sender stellenweise die gleiche Oberfläche benutzen. Dies gilt für die einzelnen Sendergruppen von gemeinsamen Sendeanstalten. Als Beispiel können hier die Sender ARD mit allen dritten Programmen genannt werden. Alle verwenden Sie eine in das Fernsehprogramm integrierte Oberfläche mit dem aktuellen Programmplan (für jeden Sender individuell). Startet man darauf die Vollbildanwendung, gelangt man in die gleiche Mediathek, die des Hauptsenders ARD.

Abbildung 17: “Red Button” exemplarisch

Neben des neuen Traffics, welcher dem Datendienst HbbTV zugeordnet werden konnte, entstanden zusätzlich keine Änderungen im Vergleich zu den durchgeführten Test in Kapitel 4.1 und 4.2. Daher wurde der Fokus dieses Tests auf den Datendienst HbbTV gelegt. Angemerkt sei an dieser Stelle, dass HbbTV keine Entwicklung von Samsung ist, sondern lediglich ein Standard, welcher von aktuellen Smart TV Geräten von Samsung unterstützt wird. Andere Fernsehhersteller unterstützen diesen Standard mittlerweile auch. Auch die Implementierung der HbbTV Oberfläche oder die Übertragung der Pakete ist unabhängig von dem verwendeten Testgerät. Daher grenzt sich der Test ab, da es sich in den folgenden Ergebnissen um keine Samsung bezogene Probleme handelt. Für die weiteren Tests, musste der Testaufbau nicht verändert werden. Der Aufbau aus Figure 5 wurde weiter verwendet.

Paketverhalten HbbTV

Eine erste Analyse der Verbindungen ergab, dass alle HbbTV Dienste über das HTTP Protokoll aufgebaut werden. Damit die benötigten Informationen über den Inhalt und die Gestaltung der HbbTV Oberflächen bezogen werden kann, werden Verbindungen zu den jeweiligen Servern der Sender aufgebaut. Dies konnte anhand des verwendeten Tools utrace.de [31] verifiziert werden. Es konnte eine klare Struktur im Verhalten beim Beziehen der Pakete beobachtet werden, welches in Figure 18 und 19 dargestellt ist.

Schaltet der Nutzer auf einen Sender, werden im Hintergrund Initial Pakete I_{Sender} für den HbbTV Inhalt vom entsprechenden Sender heruntergeladen. Ist diese Übertragung abgeschlossen, erscheint in der unteren rechten Ecke der bekannte „Red-Button“. Dieses Verhalten ist in Punkt eins der Abbildung 12 beschrieben. Besucht man nach dem Umschalten einen Sender zu einem späteren Zeitpunkt, werden die Initialen Informationen nochmals bezogen. Dabei bleiben die Pakete für jeden Sender, bei jedem Download, identisch. Die Größe der Pakete variiert bei Sendern unterschiedlicher Sendergruppen. Dieses Verhalten wurde auf allen Fernsehsendern beobachtet und auch durch das Referenzgerät bestätigt. Dies bedeutet bei eingeschalteter Datendienstfunktion, dass auch wenn der Nutzer keine HbbTV Datendienste aktiv nutzt, immer Inhalte von den Servern der Sender, im Hintergrund, bezogen werden.

Aktiviert der Nutzer nun den HbbTV Inhalt öffnet sich die Seite des Senders und neue Informationen werden über das Netzwerk nachgeladen. Der Traffic hierbei variiert von Sender zu Sender. Dieses Verhalten ist in Punkt zwei in Abbildung

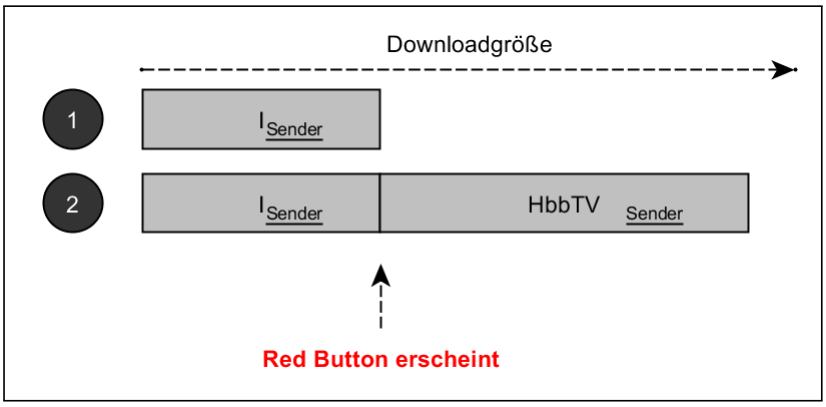

Abbildung 18: Pakete bei Nutzung von HbbTV

12 verdeutlicht. Gemeinsam haben diese beiden Fälle, dass immer zuerst die Pakete I_{Sender} des Senders heruntergeladen werden müssen.

Verbindungen zu Dritten

Zusätzlich zu den IP Adressen der Sender wurden bei einzelnen Sendern Verbindungen zu Servern dritter nachgewiesen. Dabei konnte die Verwendung von Webanalyse Tools festgestellt werden. Hierbei handelt es sich bei den gefundenen Tools um das Google eigene Tool „Google Web Analytics“[16], „Chartbeat“ [10] und „Webtrek“ [48]. Auch eine Verwendung von Cookies konnte bei einigen Sendern nachgewiesen werden. Diese wurden mit einer sehr langen Laufzeit gesetzt, jedoch nicht mehr zurück übertragen.

Weiter konnte bei aktivem Datendienst eine regelmäßige Verbindung zum entsprechenden HbbTV Server des Senders festgestellt werden. Dieses Verhalten konnte beobachtet werden, bis der Nutzer die HbbTV Oberfläche über seine Fernbedienung aktivierte. Die Anfragen erfolgten dabei periodisch, wobei die Größe der Intervalle zwischen einer Sekunde und einer Minute betragen. Entsprechenden wurde das Datenverhalten durch Abbildung 13 erweitert.

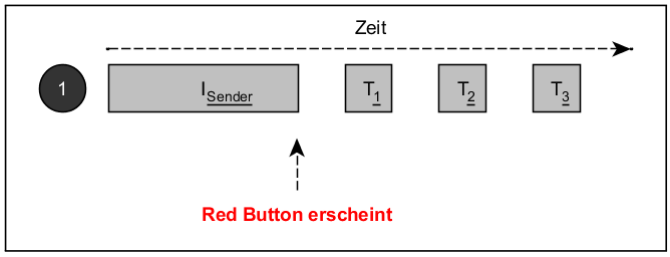

Abbildung 19: Datenpakete der Intervallanfragen

Erstellen einer Sendersignatur

Ein weiterer Test, sollte die Eindeutigkeit der HbbTV Datenpakete bestätigen. Ziel war es, die Pakete I_{Sender}, welcher bei jedem Einschalten eines Senders übertragen wird, allein anhand der Größe zu identifizieren.

In ersten Messungen wurden Signaturen, anhand der Größe, für die einzelnen Sender erstellt. Dabei wurde eine Liste der Reihenfolge der entsprechenden Paketgrößen pro Sender erstellt. Diese dient als Signatur der einzelnen Sender. Der Inhalt der Pakete spielte für diesen Test keine Rolle, auch wenn die Möglichkeit durch die Übertragung über HTTP möglich wäre.

Die erstelle Liste zeigte für die Sender jeweils eine eindeutige Signatur in der Größe und Abfolge der einzelnen Pakete. Wie in der Einführung des HbbTV Standards erwähnt, besitzen Sender gleicher Sendergruppen auch dieselbe HbbTV Oberfläche. Ein Unterschied der Signaturen innerhalb einer Sendergruppe konnte nicht festgestellt werden. Dennoch lässt sich anhand der Testergebnisse darauf schließen, dass es allein durch die Größe der Signatur möglich ist, den Sender zu bestimmen. Die Signaturen wurden immer für den gleichen Zeitabschnitt erstellt. Die Protokollierung der Pakete erfolgte vom Einschalten des Senders, bis zum Erscheinen des „Red Buttons“ am unteren rechten Bildschirmrand. Diese Zeitspanne gilt als Übertragung der I_{Sender} Pakete, welcher als Signatur des Senders bezeichnet wird.

Im aktuellen Testaufbau wird die Messung und Überwachung des Traffics durch einen Netzwerkteilnehmer durchgeführt. Das bedeutet, dass die Messung mit vollem Zugriff im Netzwerk erfolgt. Ein zweiter Test sollte bestätigen, dass

es auch möglich ist, die Erkennung der Signaturen durch einen nicht im Netzwerk authentifizierten Teilnehmer erfolgen kann. Der veränderte Testaufbau ist in Abbildung 20 gegeben. Dabei wurde eine WLAN zwischen dem Smart TV und dem gegeben Router erstellt. Die Verbindung wurde mit einer WPA 2 Verschlüsselung abgesichert [21].

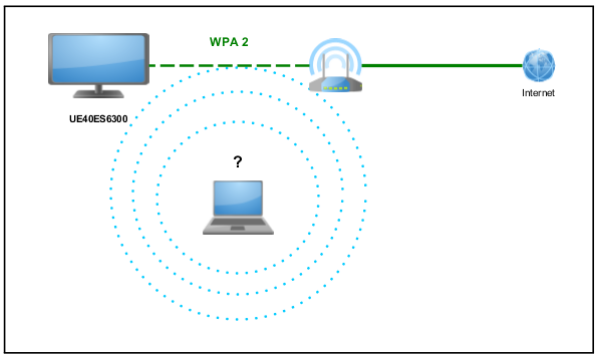

Abbildung 20: Vergleichsmessung der Signatur

Der zweite Test sollte das gleiche Ergebnis verifiziert werden, wie der Test über die LAN Schnittstelle. Auch hier ist es möglich das aktuelle Programm anhand der Signatur zu erkennen. Der Messpunkt verfügt dieses Mal zwar nicht über den Klartext der Verbindung, dennoch ändert eine Verschlüsselung keine Paketgrößen. Deshalb ist die Messung ebenso erfolgreich wie bei der Protokollierung über die LAN Schnittstelle.

4.4.2 Risiko

Aus dem Test des Smart TVs mit Fernsehsignal ergaben sich Sicherheits- und Datenschutzrelevante Risiken in Bezug auf die Nutzung des HbbTV Datendienstes. Diese werden in diesem Abschnitt näher betrachtet. Dabei beziehen sich die Risiken und die daraus entstehenden Probleme nicht auf Samsung spezifische Sicherheitsprobleme, sondern sind auf den Datendienst HbbTV zurückzuführen. Der erste Teil dieses Kapitels beschäftigt sich mit dem offensichtlichen Tracking der Fernsehsender anhand Intervall basierenden Anfragen an den jeweiligen Server und die Nutzung von Drittanbietern. Im zweiten Abschnitt des Kapitels, werden die Risiken analysiert welche durch die fehlende Vertraulichkeit entsteht, durch die nicht vorhandene Verschlüsselung des HTTP Protokolls. Am Schluss wird noch auf die eindeutige Paketgröße der Initialen Pakete der HbbTV Sender eingegangen und ein mögliches Angriffsszenario wird vorgestellt.

Der Test aus Kapitel 4.3.1 ergab, dass Sender ein Intervall basierendes Tracking nutzen. Nachdem ein Fernsehsender eingeschaltet wurde, werden zuerst die Initialen HbbTV Daten I_{Sender} bezogen. Danach wird, in einem Intervall von einer Sekunde bis mehreren Minuten [26], eine Anfrage an den jeweiligen Server des Senders gesendet. Die Dauer des Intervalls unterscheidet sich innerhalb der Sender. Die komplette Kommunikation dieser Daten findet im Hintergrund, während dem laufenden Programm statt, ohne dass der Nutzer dies bemerkt.

Mithilfe dieses offensichtlichen Trackings können die Sender mehrere Informationen über den Nutzer ermitteln. Durch die Initialen Pakete welche vom Server bezogen werden, kann der Sender erkennen wann ein Nutzer den jeweiligen Fernsehsender einschaltet. Dies ist möglich, da die notwendigen HbbTV Daten I_{Sender} schon vor der Nutzung des Angebotes angefordert werden. Zusätzlich hierzu ist es durch die Intervall basierenden Anfragen außerdem möglich zu bestimmen, wie lange ein Nutzer auf einem bestimmten Sender bleibt. Aus diesen beiden Informationen kann jeder HbbTV nutzende Fernsehsender äußerst detaillierte Verhaltensmuster des Nutzers erstellen. Zum Beispiel, wie lange ein Nutzer auf einem bestimmten Programm verweilt oder welche Inhalte er bevorzugt. Dennoch kann der Sender, nur für seinen Sender Nutzungsstatistiken erstellen. Eine Nutzungsstatistik kann nur innerhalb der Sendergruppe erfolgen, nicht übergreifend.

Zusätzlich hierzu nutzen Sender zur Analyse dieser Daten das Angebot eines Drittanbieters. Hierbei wurden die beiden „Web Analyse Tools Google Analytics“, „Chartbeat“ und „Webtrek“ gefunden. Diese Tools sind darauf spezialisiert um Statistiken und Auswertungen über die Nutzung von bestimmten Inhalten zu liefern. Hierbei werden verschiedenste Daten erhoben und kombiniert. In allen HbbTV Portalen konnte weder eine Datenschutzerklärung noch eine Hinweis auf die Benutzung von Drittanbieterseiten gefunden werden.

Ein Sicherheitsrelevantes Risiko ergibt sich, wie schon im vorangegangen Test der Smart Features des Smart TVs, durch die Nutzung von HTTP zur Übertragung der HbbTV Inhalte. Hierdurch kann die Integrität der Ressourcen nicht verifiziert werden. Ein Angriffsszenario, welches dieses fehlende Schutzziel ausnutzt, wurde im vorangegangenen Phishing Szenario gegeben und kann für diese Inhalte angepasst werden.

Angriffsszenario anhand der Sendersignaturen

Im letzten Abschnitt dieses Kapitels soll anhand eines Angriffsszenarios gezeigt werden, dass es auch für dritte möglich ist, das aktuelle Fernsehprogramm und somit Nutzungsstatistiken über den Nutzer zu ermitteln. In Test drei wurde festgestellt, dass die Initialen Pakete I_{Sender} eine eindeutige Signatur anhand der einzelnen Paketgrößen besitzen. Schaltet ein Nutzer ein HbbTV nutzendes Fernsehprogramm ein, werden die benötigten Daten bis zum Erscheinen des „Red Button“ heruntergeladen. Wie schon im Analyse Teil dieses Tests festgestellt wurde, ist die Erkennung der Signatur nicht nur im Klartext möglich, d.h. bei Aufzeichnung des Traffics im Netzwerk, sondern auch bei der Nutzung eines verschlüsselten WLANs. Wir nehmen an, dass der Angreifer über eine Liste der entsprechenden Sender Signaturen I_{Sender} verfügt. Diese können zuvor bestimmt werden, über einen Test, beschrieben im ersten Kapitel dieses Tests.

Der Angreifer muss nun das WLAN des Opfers mitschneiden indem er z.B. mit einem Laptop das entsprechende Signal abhört. Es wird angenommen, dass die WLAN Übertragung verschlüsselt ist. Hierfür nehmen wir an, dass eine WPA 2 [21] Verschlüsselung mit ausreichend langem und sicherem Passwort eingesetzt wird. Die verschlüsselte Kommunikation hat die Eigenschaft, dass der Inhalt verschlüsselt wird und Inhalte nicht im Klartext mitgelesen werden können. Sender und Empfänger können trotzdem erkannt werden, anhand der Kommunikationsendpunkte.

Die Verschlüsselung spielt hierbei keine Rolle, da der Angriffsvektor auf der Größe der einzelnen Datenpakete beruht, welche von der Verschlüsselung unberührt bleibt. Lediglich ein Overhead, durch die Verschlüsselung, wird an jedes Paket angehängt. Diese Verschiebung der Größte der Signaturen kann leicht erkannt werden.

Hat der Angreifer die Daten erkannt (welche vom Smart TV versendet werden) kann er die relevanten Initialen Pakete anhand der Signatur erkennen bzw. abgesendete Tracking Informationen mitschneiden. Dadurch ist es dem Angreifer möglich die gleichen Nutzungsstatistiken wie die Fernsehsender zu erzeugen. Dabei ist es dem Angreifer möglich Sender übergreifende Statistiken zu erstellen, da er Zugang zu allen Sendern hat. Der Angriffsverlauf ist in Grafik 21 dargestellt.

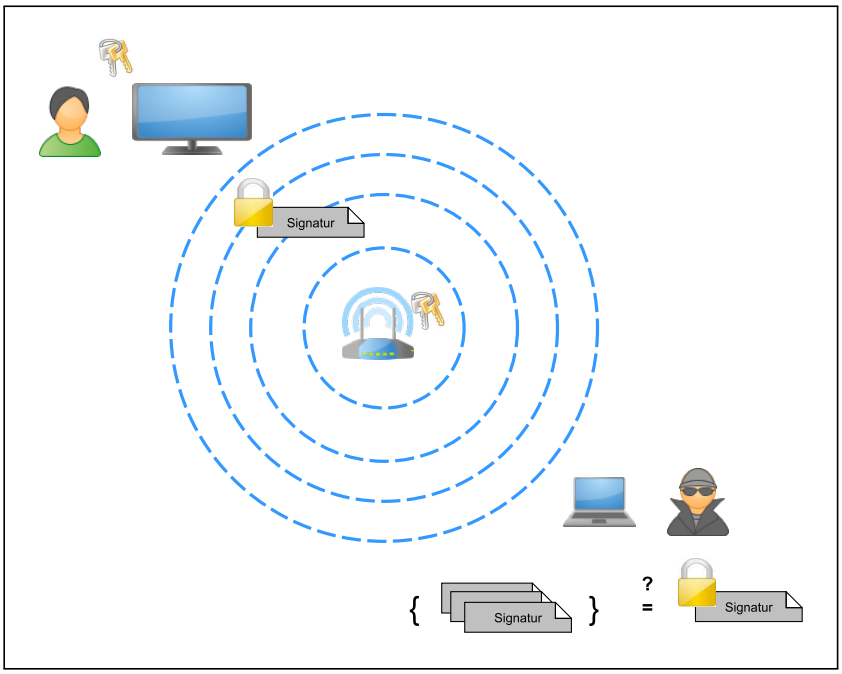

Abbildung 21: Angriffsszenario HbbTV

Diese Datenschutz- und Sicherheitsrelevante Lücke ist als kritisch einzuschätzen, da es einem Angreifer ohne komplexere Maßnahmen möglich ist, ein sichere Verschlüsselung System zu umgehen und der Nutzer keine Möglichkeit hat sich davor zu schützen. Diese eingesetzte Technik wird als Seitenkanalangriff bezeichnet.

4.4.3 Lösung

In diesem Abschnitt sollen Lösungsvorschläge zur Datenschutzproblematik, welche durch den Datendienst HbbTV auf Smart TV Testgerät genutzt wird, aufgezeigt werden.

Die HbbTV Daten werden per HTTP auf den Fernseher übertragen. Da diese Implementierung des Protokolls keine Integritätsprüfung besitzt, können übertragene Ressourcen vor dem Empfang modifiziert werden und an den Smart TV weitergeleitet werden. Besteht zu einem späteren Zeitpunkt der Implementierung von Bezahlinhalten in der HbbTV Oberfläche, sollte zusätzlich das Schutzziel der Vertraulichkeit erreicht werden. Hierfür könnte ein Wechsel auf HTTPS in Erwägung gezogen werden. Damit wird gleichzeitig auch das Schutzziel der Integrität und Authentizität erfüllt.

Im Angriffsszenario (Vgl. Abbildung 21) wurde verdeutlicht, dass es für einen Angreifer möglich ist, detaillierte Nutzungsstatistiken zu erstellen. Dies ist möglich, da der Angreifer anhand der eindeutigen Signatur der Paketgrößen der Initialen HbbTV Pakete I_{Sender} und der darauffolgenden Tracking Pakete erkenne kann was ein Nutzer gerade tut. Wie hier schon gezeigt wurde, kann dieses Problem nicht durch kryptographische Mittel gelöst werden, den selbst durch eine, als sicher anzusehende WPA 2 Verschlüsselung, kann dieser Seitenkanalangriff nicht verhindert werden. Dabei beruht die Stärke des Angriffs auf der Größe der einzelnen Pakete, diese wird nicht durch die eingesetzte Verschlüsselung verändert. Generell sind keine sicheren WLAN Verschlüsselungen bekannt, welche die Paketgröße so ändern, dass zwischen den Paketen nicht mehr unterschieden werden kann.

Als Lösung für dieses explizite Problem kann ein Mittel aus der Kryptographie verwendet werden, welches einzelne Pakete „auffüllt“, sodass Sie einer bestimmten Größe entsprechen. Dieses Verfahren wird als „Padding“ [25] bezeichnet. Ist ein Paket „zu klein“, wird es vom Sender mit einer bestimmten Zeichenfolge aufgefüllt, bis es der gewünschten Größe entspricht. Auf der Seite des Senders wiederum, müssen diese Zeichenfolgen erkannt werden und entfernt werden, damit die Semantik der Pakete erhalten bleibt. Diese Größe könnte anhand des Standards definiert werden, und somit die Pakete nur noch in einer bestimmten Größe übertragen werden. Damit wäre eine Entscheidung des Senders anhand der Eindeutigkeit der Paketgrößen nicht mehr gegeben und für dritte nicht mehr möglich Nutzungsstatistiken zu erstellen. Weiterhin könnte eine Lösung im Standard angestrebt werden, in dem die Paketgrößen für jeden Sender zufällig aufgefüllt wird. Dennoch sei an diese Stelle gesagt, dass nur wenn alle Sender diese Änderungen vornehmen, der Angriff abgewehrt werden kann. Falls nur weniger und gar nur ein Sender diese Änderung vornimmt, kann anhand der zufälligen Größe genau dieser Sender bestimmt werden. Deshalb ist bei beiden Lösungen, der Änderung der Paketgröße, eine Zusammenarbeit aller HbbTV Sender notwendig.

Weiterführende Probleme

Die beschrieben Lösung würde zwar das explizite Problem der Paketgröße lösen, das Tracking durch die Sender würde trotzdem weiterhin möglich sein. Sowohl der Angreifer im Angriffsszenario als auch die Fernsehsender bilden ihre Nutzungsstatistiken anhand der Initialen Pakete des HbbTV Dienstes, welche ohne Kenntnis des Nutzers bezogen werden bevor eine Nutzung von HbbTV stattfindet. Aus technischer Sicht ist keinerlei Notwendigkeit die HbbTV Daten vor der Nutzung des Dienstes herunterzuladen. Noch weniger ist es nötig die Daten bei jedem einschalten eines Sender erneut zu beziehen. Eine aktuelle Liste der HbbTV fähigen Sender könnte über z.B. über die Aktualisierung der Firmware auf den Fernseher gelangen. Hierbei wäre die Datei zentral auf einem Server gespeichert. Ein Tracking wäre nicht möglich. Auch eine Art einmaliger „Suchlauf“ bei der Aktivierung der Datendienste auf dem Fernseher ist denkbar. Hierbei wird der Nutzer darüber informiert, dass nun nach HbbTV Sendern gesucht wird.

Durch die genannten Lösungsvorschläge wäre die Nutzung der HbbTV Inhalte nicht eingeschränkt. Zusätzlich hierzu müssten bei allen HbbTV Sendern die Datenschutzerklärungen hinzugefügt werden, da diese bei keinem Sender vorhanden waren. Nutzt ein Sender, wie teilweise auch festgestellt wurde, einen Drittanbieter für die Analyse der Daten, muss ausdrücklich auf die Weitergabe der Daten hingewiesen werden.

In Kapitel 4.1.3 wurde Bezug auf das mentale Modell des Nutzers gegeben. Auch dieses wird durch die aktuelle Implementierung des HbbTV Datendienstes verletzt, da Informationen über einen Dienst heruntergeladen werden, ohne das der Nutzer den Inhalt aktiv nutzt.

Die gesamten Ergebnisse, mit zusätzlichen Informationen, aus Kapitel 4.3 werden in einem eigenen Paper, „HbbTV -I Know What You Are Watching“, veröffentlicht und auf dem 13. Deutschen IT Sicherheitskongress präsentiert [26].

4.5 „Pairing“

Der vierte Test löst sich von den vorangegangenen Testreihe und untersucht die Kombination des Smart TVs mit einem zweiten Gerät. Dabei wird die Möglichkeit der Steuerung des Smart TVs über ein Smartphone untersucht. Dabei wird eine kritische Schwachstelle im „Pairing“ des Smart TVs aufgedeckt und weiter analysiert.

4.5.1 Analyse

Der vierte Test der Testreihe, an einem Samsung Smart TV Testgerät, untersucht die Möglichkeit das Smart TV Gerät über eine auf einem Smartphone installierte Applikation zu steuern. Hierfür wurde der „Use Case“ Fernbedienung gewählt. Da auch hier nicht die Applikation untersucht werden sollte, wurde das durchgeführte „Pairing“ Protokoll auf eventuelle Schwachstellen untersucht.

Unter dem Begriff „Pairing“ versteht man einen Protokollablauf, welcher dazu dient, dass sich zwei Geräte gegenseitig authentifizieren können und danach miteinander sicher kommunizieren können.

Wie in Kapitel drei schon beschrieben wurde, verwendeten wir eine Android basiertes Smartphone für den Test. Die Möglichkeit das Smart TV zu steuern ist auch über Smartphone mit anderen Betriebssystemen (z.B. iOS) möglich. Im Gegensatz zu den vorangegangen Tests musste der Testaufbau modifiziert werden (Vgl. Bild 22). Das Smart TV ist an dem verwendeten Test Laptop über ein unverschlüsseltes WLAN Netz angeschlossen. Das Smartphone kommuniziert ebenfalls über diese unverschlüsselte WLAN Verbindung mit dem Test Laptop, an welchem die Messung durchgeführt wurde. Eine aktivierte Internetverbindung wurde für diesen Test nicht verwendet, da weder die Applikation auf dem Smartphone noch das Smart TV für diesen Test über eine Internetverbindung verfügen müssen.

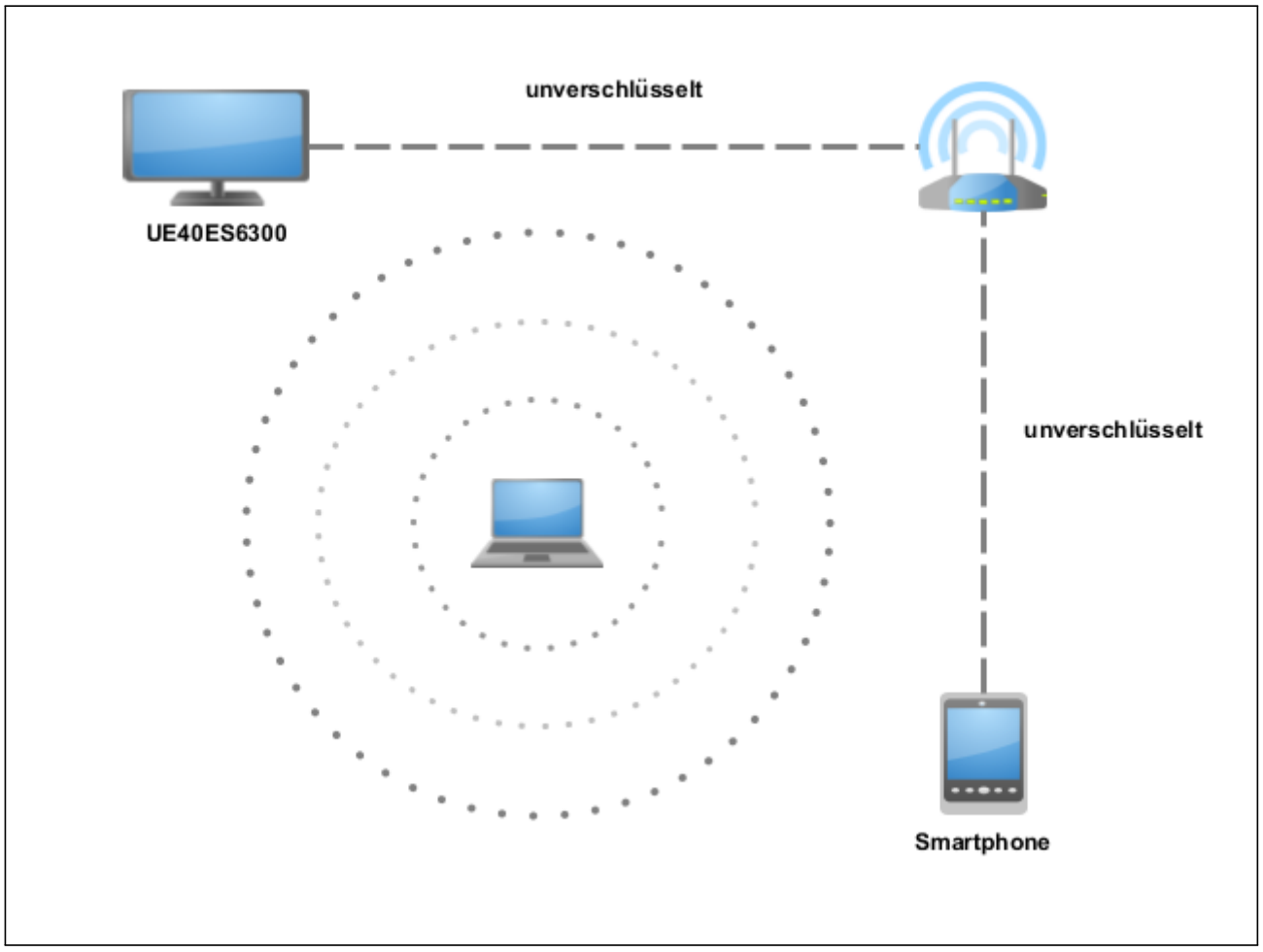

Abbildung 22: Testaufbau für den „Pairing“ Versuch

Ausgewählte Applikation

Als Applikation wurde die kostenlos im „Google Play Store“ erhältliche Smartphone Fernbedienung „SamyGo Remote“ [43] ausgwählt. Die Applikation bietet die Möglichkeit den Fernseher mit einer emulierten Fernbedienung auf dem Smartphone zu steuern. Sie bietet unterschiedliche Layouts für verschiedenen Modelle von Samsung Smart TVs. Dabei wird die Fernbedienung mit dem Original Layout auf dem Smartphone nachgebaut. Neben der kostenlos verfügbaren „SamyGo Remote“ Applikation gibt es auch eine Originale Version von Samsung selbst. „Samsung Remote“ ist ebenfalls kostenlos erhältlich für verschiedene Mobile Betriebssystem [36]. Ausschlaggebend für die Auswahl war, dass der Quelltext der „SamyGo Remote“ Applikation frei auf Github verfügbar [44] ist.

Hierdurch bestand im Test die Möglichkeit des „White-Box-Testing“ [3]. Das Vorgehen beim „Pairing“ konnte unmittelbar am Quelltext der Applikation nachvollzogen werden. Im Falle eines Einsatzes der Samsung proprietären Software, bestünde nur die Möglichkeit des „Black-Box-Testing“. Dennoch konnte festgestellt werden, dass das Verhalten beider Applikationen die gleiche Reaktion am Smart TV auslöste und somit auch identische Testergebnisse lieferte.

Bevor der „Pairing“ Vorgang eingeleitet werden kann, muss die Applikation konfiguriert werden. Hierfür wird die entsprechende IP Adresse des Smart TVs benötigt. Diese kann wahlweise manuell eingegeben werden oder über einen automatischen Suchlauf gefunden werden. Standardmäßig erfolgt die Kommunikation der externen Geräte über den Port 1234. Wird versucht die Kommunikation aufzubauen, erscheint auf dem Bildschirm eine Nachricht mit der Aufforderung ein neues Gerät zu den vertrauenswürdigen Geräten hinzuzufügen. Dabei wird in einem Bestätigungstext der entsprechende Name des Gerätes eingeblendet. Dieser kann über die Applikation auf dem Smartphone konfiguriert werden. Die Bestätigung erfolgt über einen Tastendruck mit einem bereits registrierten Gerät oder über die mitgelieferte Fernbedienung. Mit einem Tastendruck wird das „Pairing“ beendet und das neue Gerät ist erfolgreich hinzugefügt worden. Wahlweise kann der Nutzer diesen Vorgang abbrechen. Der Protokollablauf ist in Grafik 23 dargestellt.

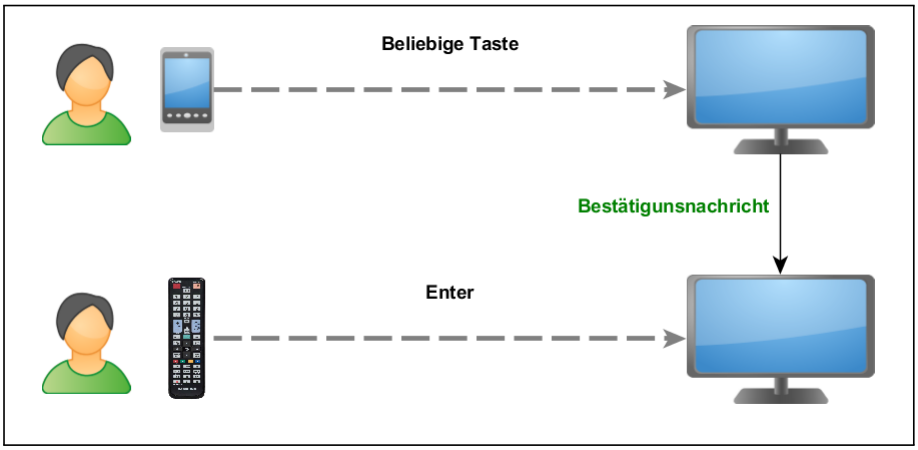

Abbildung 23: „Pairing“ Vorgehen Smart TV

All Share Technologie

Die Technologie All Share ist eine eigene Entwicklung Samsungs. Samsung beschreibt auf seiner Webseite [34] diese Technologie kurz als „Einfachste, kabellose Multimedia-Vernetzung auf allen Geräten“. Gilt ein Gerät als registriert, erscheint es in der sogenannten All Share Liste. Angeschlossene Geräte haben authentifizierten Zugriff auf den laufenden „SOAP Server“.

Das „Simple Object Access Protocol“ (kurz SOAP) [47] ermöglicht es über Anfragen, in Form einer XML Datei, Informationen über das Netzwerk abzufragen bzw. zu setzen. Dabei bietet der Server definierte Schnittstellen für Funktionen. Der auf dem Smart TV Gerät laufende Server bietet eine Reihe von Funktionen, welche ohne Authentifizierung aufgerufen werden können. Zu diesen zählen z.B. die Veränderungen der Lautstärke und die damit verbundene Funktion der Abfrage der aktuellen Lautstärke. Der größte Anteil der Funktionen können nur von authentifizierten Geräten durchgeführt werden, d.h. nur die welche eine „Pairing“ mit dem Smart TV abgeschlossen haben und damit in der All Share Liste stehen.

Eine Analyse der Netzwerkpakete ergab, dass die Pakete sowohl beim „Pairing“ als auch beim übertragen der späteren Signale, alle Pakete im Klartext gesendet werden. In einem nächsten Schritt, wurde die Struktur der Pakete durch Klartextübertragung erkannt. Die Übertragung von Befehlen an den Smart TV erfolgen immer in zwei Schritten. In einer ersten Übertragung wird ein entsprechendes Paket mit der Kennung, ID_{Device}, des angeschlossenes Gerätes übertragen. Dieser enthält die IP Adresse und den Namen des Gerätes. Zusätzlich besitzt das Paket eine klare Paketstruktur mit

Anfangs und Endzeichen, diese können vernachlässigt werden, da sie für die Analyse keine Rolle spielen. Nach der Übertragung von ID_{Device} wird ein zweites Paket mit dem Befehl, $Com_{Behfehl}$, gesendet. Die entsprechenden Befehle können aus dem Quelltext der Applikation entnommen werden. Alle Pakete sind mit der Base64 Kodierung kodiert [28].

Simulation des „Pairings“

Für weitere Tests, wurde die Funktionalität der Fernbedienung auf dem Test Laptop nachgebaut. Hierfür sollten die gewonnen Paketstrukturen verwendet werden, um eigene Befehle an den Smart TV zu senden. Dafür wurde unser Testgerät und der Laptop im gleichen WLAN Netz verbunden. Dieser übernahm für die weiteren Tests die Funktionalität des Smartphones. Der Testaufbau (Vgl. Grafik 24) wurde um den WLAN Router erweitert.

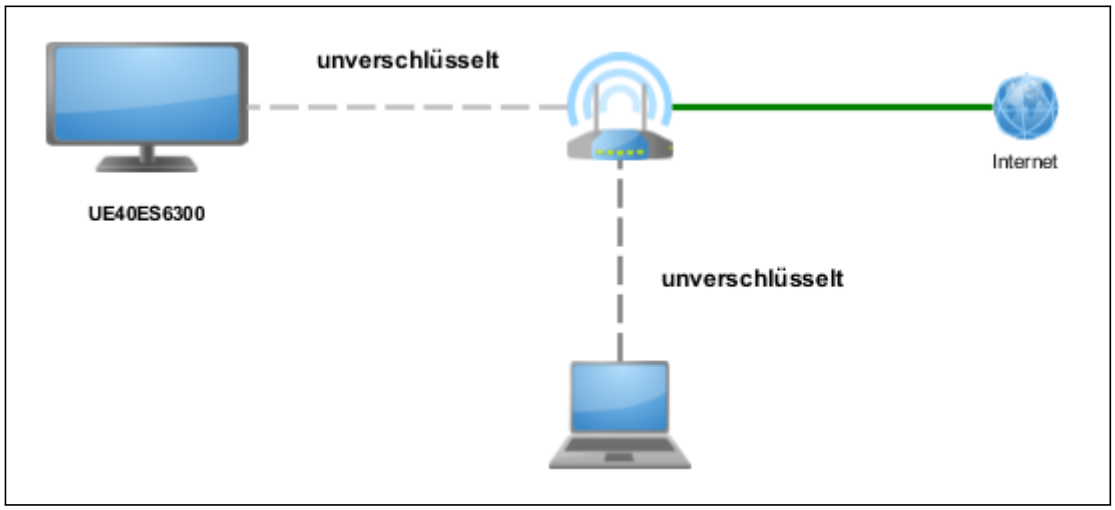

Abbildung 24: Veränderter Testaufbau „Pairing“

In ersten Versuchen, wurde versucht ein erfolgreiches „Pairing“ zwischen Smart TV und Laptop abzuschließen. Hierfür wurde das Paket ID_{Laptop} entsprechend angepasst. Und das in Figure 16 beschriebene „Pairing“ durchgeführt. Das Paket ID_{Laptop} wurde mit der IP Adresse des Laptops angepasst. Der Name konnte frei gewählt werden. Das „Pairing“ wurde erfolgreich beendet und eine Steuerung konnte durch das Senden einiger Befehle verifiziert werden. Hierfür wurde zuerst immer das Paket ID_{Laptop} gesendet und darauf verschiedene Kommandos, z.B. $Com_{VolumeUp}$ oder $Com_{ChannelUp}$.

In einem weiteren Test wurde versucht die Kennung ID_{Laptop} so zu ändern, dass sie nicht auf das entsprechende Gerät passt. Hierbei wurde untersucht, ob es trotzdem möglich ist, mit einer falschen Kennung das Smart TV zu steuern. Als Identifizierung konnte festgestellt werden, dass nur die IP Adresse relevant für die Kennung sein kann. Da das Paket ID_{Device} nur über zwei Merkmale verfügt, IP Adresse und Name, und der Name wie festgestellt werden konnte frei gewählt werden kann, bleibt nur noch die IP Adresse als Kennung übrig.

Die IP Adresse wurde im Paket ID_{Laptop} entsprechend verfälscht, dennoch war sie syntaktisch korrekt. Der „Pairing“ Vorgang aus Figure 16 wurde wiederholt. Auch bei diesem Test konnte das Gerät registriert werden. Im Folgenden wurden weitere Test durch die Veränderung des Paktes ID_{Laptop} durchgeführt. Die Tests ergaben, dass es auch möglich ist ein namenloses Gerät hinzuzufügen. Behält man die Paketstruktur, Anfangs- und Endzeichen, und entfernt alle restlichen Informationen, wird das „Pairing“ dennoch durchgeführt. Der entsprechende All Share Eintrag trägt die IP Adresse 0.0.0.0 und das Namensfeld bleibt leer. Auch ist es möglich für ein physikalisches Gerät mehrere Kennungen auf dem Fernseher zu platzieren. Außerdem konnten gleiche Kennung doppelt abgelegt werden. Alle registrieren Geräte wurden mit der entsprechenden Kennung auch zum Steuern genutzt werden.

Das Verhalten beim Entfernen der verbundenen Geräte ist nicht fehlerfrei. Registrierte All Share Geräte können über die Liste per Fernbedienung entfernt werden. Dennoch bleibt die Funktionalität des registrieren Gerätes welches bis zum Neustart des Smart TVs erhalten.

Während den durchgeführten Tests wurde festgestellt, dass das Smart TV Befehle von registrierten Geräten zu jeder Zeit annimmt. Zu welchen Problemen dies führen kann, wird in Kapitel 4.5 am Ende dieser Arbeit beschrieben.

4.5.2 Risiko

Im letzten vorangehenden Analyse Abschnitt des vierten Tests, wurde das „Pairing“ untersucht, welches eingesetzt wird, um in diesem Testfall, ein Smartphone mit unserem Testgerät zu verbinden und dies als Fernbedienung zu verwenden.

Dafür wurde eine Android Applikation verwendet und später wurde das Verhalten einem Laptop simuliert.

Dabei wurde festgestellt, dass die Kommunikation über ein Kennungspaket ID_{Device}, IP Adresse und Name, eingeleitet wird. Darauf wird ein Befehlspaket $Com_{Command}$ an das Smart TV gesendet. Diese können abgefangen werden, da es sich um eine unverschlüsselte Verbindung handelt. Des Weiteren gibt es keine Prüfung anhand des ID_{Device} Paketes auf die Identität des steuernden Gerätes. Das bedeutet, die Kennungspakete können selbst gestaltet werden (Tupel aus IP Adresse und Name) oder von anderen Geräten einfach kopiert werden. Dass es möglich ist mit diesen Kennungen das Smart TV zu steuern, wurde im Analyse Abschnitt gezeigt. Um die Gefahren zu verdeutlichen wird ein Testszenario vorgestellt welches es ermöglicht das Smart TV zu steuern, ohne dass der Nutzer des Smart TV dies bestätigt.

Das Testszenario benutzt den gleichen Testaufbau welcher auch in Grafik 24 verwendet wurde. Das Szenario zeigt, dass es möglich ist eine vorhandene Kennung eines Gerätes zu nutzen, welches schon in der „All Share“ Liste registriert ist um ein eigenes Gerät zu registrieren und anonym, Daten über das Smart TV zu beziehen bzw. das Gerät unbrauchbar zu machen. Wir nehmen an, dass der Angreifer vollen Zugriff zum Netzwerk hat. Das bedeutet, dass das gesamte Szenario mit einem unverschlüsselten WLAN durchgeführt wird. Zusätzlich dazu muss ein Gerät bereits auf dem Smart TV registriert worden sein und muss zum Zeitpunkt des Angriffs, zumindest anfänglich, auch benutzt werden.

Angriffsszenario „Pairing“

Das Angriffsszenario kann grob aufgeteilt werden in drei Schritte: Stehlen der Kennung (Vgl. Grafik 25) – Registrierung eines eigenen Gerätes (Vgl. Grafik 26) – Anonymisierung.

Im ersten Schritt wird die fehlende Verschlüsselung der Übertragung dazu genutzt um das Kennungspaket $ID_{Reg.\ Device}$ des registrierten Gerätes Reg. Device zu stehlen. Hierfür wird die WLAN Verbindung vom Angreifer abgehört und auf ein $ID_{Reg.\ Device}$ Paket gewartet. Da dieses Paket bei jedem einzelnen Befehl übertragen wird, muss die Smartphone Fernbedienung nur benutzt werden, bis ein solches Paket übertragen wird. Wurde das Paket erfolgreich abgefangen kann

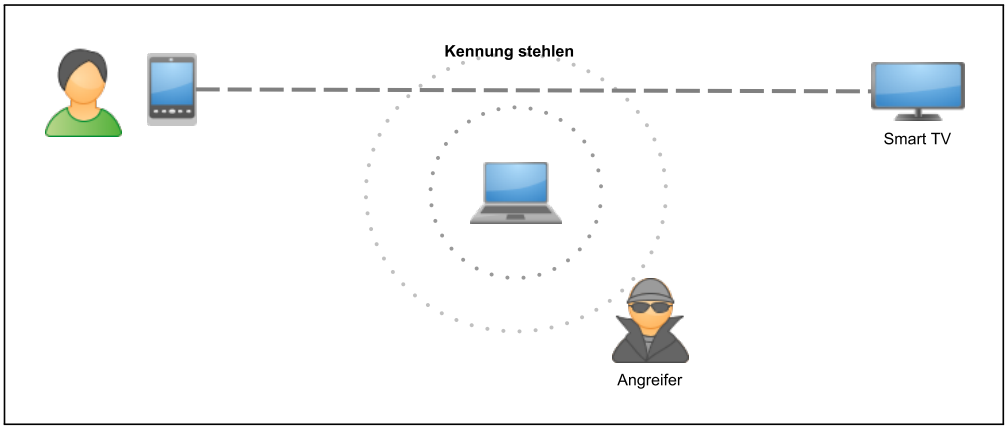

Abbildung 25: Stehlen der Kennung

$ID_{Reg.\ Device}$ dazu genutzt werden um das eigene Gerät des Angreifers zu registrieren. Dafür wird die Kennung $ID_{Reg.\ Device}$ benutzt um das eigene Gerät zu bestätigen. Insgesamt besteht die Registrierung aus zwei Schritten. Es wird eine eigene Kennung $ID_{Own\ Device}$ erstellt und an den Smart TV gesendet. Daraufhin erscheint eine Authentifizierungsnachricht am unteren linken Bildrand. Im zweiten Schritt wird das gestohlene Paket $ID_{Reg.\ Device}$ genutzt um diese Nachricht zu bestätigen. Hierzu wird das bekannte Kennungspaket und darauf der Befehl für die Taste „Enter“ Com_{Enter} gesendet. Damit wird das neue Gerät hinzugefügt, ohne das der Nutzer vor dem Fernseher dies wirklich bestätigt. Lediglich die Kennung des Gerätes, welches der Nutzer zuvor genutzt hat, wird verwendet. Wie schon in Kapitel 4.5 festgestellt wurde, kann die Abfolge der Befehle – $ID_{Reg.\ Device}$ $ID_{Own\ Device}$ Com_{Enter} – ohne Verzögerung nacheinander gesendet werden. Damit ist es dem Nutzer nicht möglich vor dem Smart TV den Angriff zu bemerken. Im Szenario konnte lediglich ein leichtes flackern des Bildschirms beobachtet werden. Wurden diese beiden Schritte ausgeführt, ist das Gerät des Angreifers registriert. Damit hat er völligen Zugriff auf die „All Share“ Dienste und damit auch auf den SOAP Server.

In einem letzten Schritt soll der Zugriff des Angreifers noch anonymisiert werden. Hierfür wird das eben hinzugefügte Gerät aus der „All Share Liste“ entfernt. Aus dem Fehler beim Entfernen des Gerätes, wie in Kapitel 4.4.1 festgestellt wurde, kann das Gerät noch bis zum Neustart bedient werden. Selbst wenn der Nutzer nun nach fremden Geräten in der „All Share“ Liste sucht, wird das Gerät des Angreifers nicht gefunden. Wurde einmal eine Kennung gestohlen, kann das Gerät bei einem Neustart hinzugefügt werden.

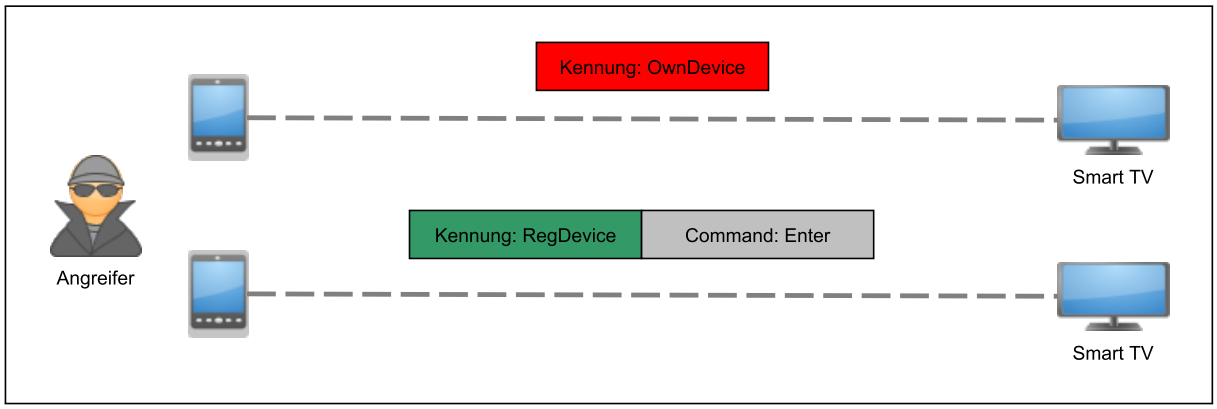

Abbildung 26: Registrierung des eigenen Gerätes

Nach diesen drei Schritten ist es für einen Angreifer möglich anonym auf Funktionen des Smart TVs zuzugreifen. Dabei beschränken sich die für den Angreifer interessanten Funktionen auf die Funktionalität des SOAP Servers. Im Folgenden werden nun einige Beispiele genannt, was mit einem anonymen registrierten Gerät durchgeführt werden können. Durch eine sequentielle Befehlsfolge kann der Fernseher, anhand der bekannten Tastenfolge, in Werkseinstellungen versetzt werden. Auch die Bestätigung der Abfrage erfolgt über das Gerät des Angreifers. Danach hat der Angreifer zwar keinen Zugriff mehr auf das Smart TV kann später sein Gerät über das oben beschriebene Protokoll erneut registrieren.

Möglichkeiten eines regestrierten Gerätes

Eine andere Möglichkeit ergibt sich aus dem Zugriff auf den SOAP Server. Hierbei können Datenschutzrelevante Informationen über das Nutzungsverhalten des Nutzers abgefragt werden. Z.B. aktuelle Programmierungen von Aufnahmen oder eine Liste der häufig eingeschalteten Sender. Zusätzlich hierzu, kann über den SOAP Server die Smart TV PIN (zum Schutz des Fernsehers) über das „Brute Force“ Verfahren ermittelt werden, ohne dass der Nutzer dies mitbekommt. Hierbei kann dem SOAP Server eine viertstellige Nummer gesendet werden. Nach der Validierung wird dem Nutzer entweder 0, falsche PIN, oder 1 für richtige PIN gesendet. Da es sich um eine 4 stellige PIN handelt, gibt es maximal 10000 Kombinationen. „Brute Force“ bedeutet, alle möglichen Kombinationen auszuprobieren. Die Laufzeit dieses Verfahrens ist in den meisten Fällen sehr schlecht. In einem zusätzlichen Schritt kann diese PIN geändert werden und der Fernseher damit für Nutzer unbenutzbar gesperrt werden. Neben ausprobieren bleibt dem Opfer nur die Möglichkeit den Fernseher in Werkseinstellungen zu versetzen.

4.5.3 Lösung

In Test vier gefundenen Lücken im „Pairing“ des Smart TVs können zu Sicherheits- und Datenschutzrelevanten Risiken führen. In diesem Abschnitt wird ein aktuelles „Pairing“ Protokoll vorgestellt, mit denen alle vorgestellten Lücken behoben werden können.

Als Lösungsvorschlag wird ein „Pairing“ Protokoll von Google benutzt [18]. Dieses ist frei verfügbar und zielt darauf ab, dass ein Client sich mit einem Server Gerät verbindet. Nach diesem Verbindungsaufbau findet eine verschlüsselte Verbindung statt. Das Protokoll wahrt die Schutzziele Authentizität und Vertraulichkeit. Damit können die in Kapitel 4.4.3 beschriebenen Risiken beseitigt werden.

Das in Grafik 27 verdeutlichte Protokoll kann grob in drei Unterschritte unterteilt werden. Im ersten Schritt dem Initialisierungsschritt meldet der Client dem Server, dass er sich gerne gegenüber dem Server authentifizieren möchte. Hier kann eine bestimmte Paketstruktur gefordert werden oder ein einfacher unverschlüsselter Befehl z.B. auf einem bestimmten Port. Im zweiten Schritt, dem Konfigurationsschritt, werden Optionen ausgetauscht, welche Authentifizierungsmöglichkeiten der jeweilige Kommunikationspartner unterstützt. Dabei sendet der Client ein Liste von unterstützen Verfahren an den Server. Der Server antwortet mit einer Liste, welche Funktionen er aus der Liste des Client unterstützt. Es geht dabei einerseits um die Eingesetzte Verschlüsselungsalgorithmen die in der späteren verschlüsselten Verbindung verwendet werden. Zusätzlich dazu wird die eingesetzte Authentifizierungsmethode ausgehandelt.

Im zweiten Teil dieser Konfiguration sendet der Client die ausgewählten Einstellungen zurück und bekommt diese vom Server bestätigt. Nachdem auch der Konfigurationsschritt beendet ist, beginnt ein „Challenge Response“ Verfahren anhand der Konfiguration aus Schritt zwei. Über einen „Out of Bounds Channel“ (OoB Channel) wird eine Nachricht an Client gesendet. Als OoB Channel wird ein Kanal bezeichnet welcher nicht über die standardmäßige Verbindung erreichbar ist.

Als Beispiel ist hier das SMS-TAN Verfahren bei diversen Online Banking Diensten genannt. Hier benutzt man das Mobilfunknetz und der Übertragungsweg von Bankserver zu Handy als OoB Channel. Hierbei wird die geheime TAN

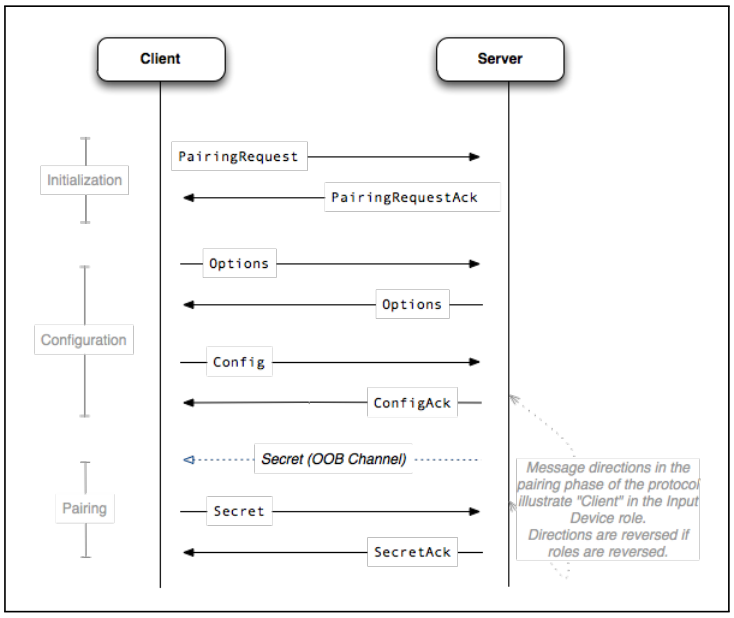

Abbildung 27: Protokollvorschlag zur Lösung (Google)

über das Mobilfunknetz übertragen, um zu verhindern dass die Nachricht abgefangen werden kann, sollte der aktuelle Übertragungsweg kompromentiert sein. Beantwortet der Client die Challenge, welche über den OoB Channel gesendet wurde zurück an den Server, kann nun eine verschlüsselte Verbindung aufgebaut werden in der beide Parteien sicher sein können, dass Sie mit dem richtigen Kommunikationspartner kommunizieren.

Sicheres „Pairing" am Smart TV

Im Folgenden soll ein beispielhafter Ablauf dargestellt werden. Im ersten Schritt sendet der Nutzer mit dem Smartphone eine beliebige Nachricht an das Smart TV und beginnt somit das Protokoll. Im zweiten Schritt werden die möglichen Optionen ausgetauscht werden (z.B. PIN, Passwort oder QR Code). Nachdem eine Authentifizierungsmethode des Client ausgewählt wurde, erscheint eine Challenge auf dem Fernseher. Die Übertragungsweg Fernsehbild – Auge wird hier als OoB Channel bezeichnet, da er unabhängig von dem Übertragungsweg über das WLAN ist. Nachdem die Challenge durch den Smart TV bestätigt wurde kann nun eine verschlüsselte Verbindung zwischen Smart TV und Smartphone stattfinden. Zusätzlich müssen Zufallszahlen in die übertragenen Pakete eingebaut werden. Diese Zufallszahlen werden als „Nonce" bezeichnet. Sie sollen sogenannte Replay Attacken [25] verhindern. Hierdurch kann die Identität der Protokollteilnehmer garantiert werden. Danach kann eine vertrauenswürdige Verbindung zwischen Smartphone und Smart TV aufgebaut werden.

In Protokollbild 28 wird das Verhalten veranschaulicht. Dabei beschreiben den ersten beiden Punkten die Übertragung und Bestätigung der Konfiguration. Hierbei wurde ein Challenge Response Verfahren anhand der Eingabe einer vierstelligen Nummer gewählt. Dieser wird auf dem Bildschirm angezeigt. Der Nutzer liest dieser ab und gibt sie in das „Pairing" Smartphone ein. Die Übertragung zwischen Fernseher und Nutzer, über das Sehen der Nummer, wird als OoB Channel bezeichnet. Nachdem alle Schritte erfolgreich durchgeführt worden sind, kann die Verbindung sicher verwendet werden.

Durch die Verschlüsselung werden gleichzeitig mehrere Dinge erreicht. Das Kennungspaket kann nicht mehr einfach mitgelesen werden und somit die Kennung gestohlen werden. Als zweites kann ein Gerät eindeutig identifiziert werden indem es diesen Schlüssel immer benutzt.Dadurch kann eine 1:1 Beziehung zwischen Gerät und Fernseher hergestellt werden. Zusätzlich wird der ausgetauschte Schlüssel für jede Kommunikation verwendet werden, da dies die Identität des Gerätes bestätigt. Da dieser Schlüssel geheim auf dem Gerät gespeichert wird, kann man davon ausgehen, dass die Kennung nicht gestohlen werden kann.

Das Problem beim Entfernen der Geräte, dass diese immer noch aktiv sind, auch nach dem Entfernen der Geräte bis zum Neustart des Smart TVs muss behoben werden. Hierbei handelt es sich um einen Implementierungsfehler.

Zusammenfassend lässt sich sagen, dass durch die Implementierung des vorgestellten Protokolls, alle gefundenen Lücken behoben werden können und das vorgestellte Angriffsszenario vermieden werden kann.

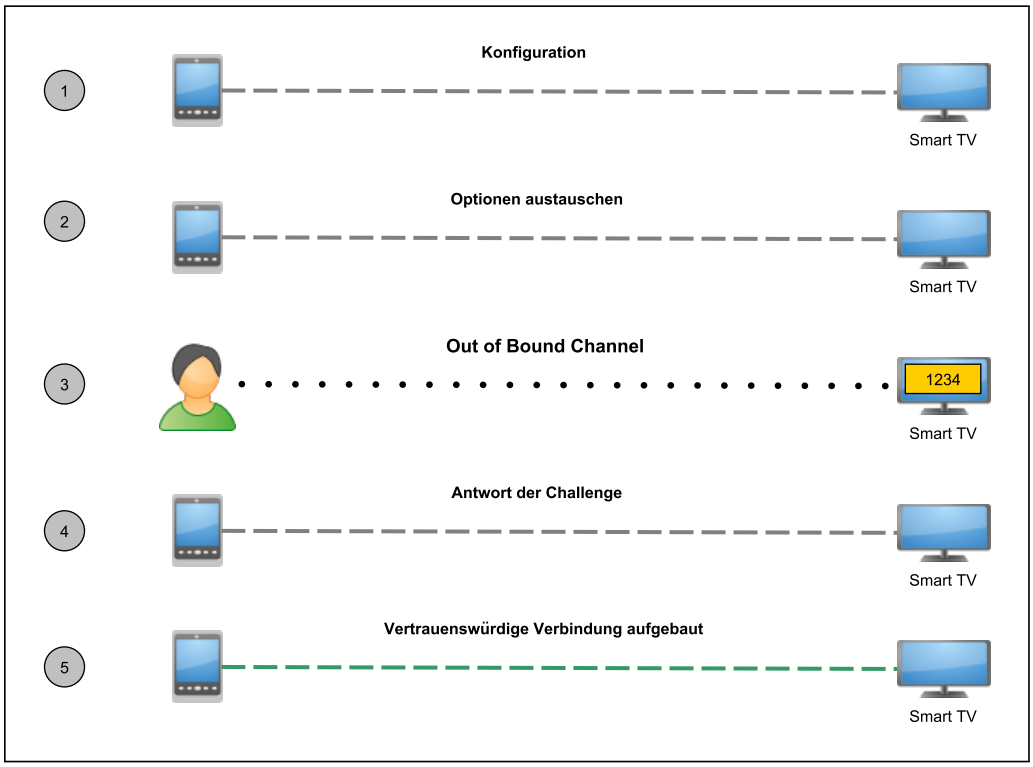

Abbildung 28: Beispiel für ein sicheres „Pairing“

4.6 Sonstige Ergebnisse

Zusätzlich zu den hier vorgestellten Testfällen sind während der kompletten Arbeit auch andere Schwachstellen gefunden worden. Diese sind zusammenhangslos mit den hier vorgestellten Szenarien, sodass sie keinem Testfall zugeordnet werden sollen bzw. zu klein um einen eigenen Testfall zu beschreiben. Dieser Abschnitt soll die nicht zugeordneten Auffälligkeiten am Smart TV beschreiben. Dabei umfasst ein Abschnitt jeweils die komplette Reihenfolge des Vorgehens: Analyse – Risiko – Lösung.

Schwachstelle Fernwartung

Das Smart TV verfügt über eine eingebaute Fernwartungslösung. Diese Funktion kann über das Hauptmenü im Smart Hub erreicht werden und über den Menüpunkt Hilfe aktiviert werden. Nachdem ein AGB Text angezeigt wird und der Nutzer diese bestätigt hat, wird eine Support Telefonnummer und eine sechsstellige Zahlenfolge auf dem Fernseher eingeblendet. Das „Pairing“ mit dem Support Mitarbeiter erfolgt nach folgendem Schema: Der Nutzer nimmt per Telefon Kontakt mit dem Supportmitarbeiter auf und teilt diesem seine sechsstellige Zahlenfolge mit. Danach kann dieser sich über eine Verbindungssoftware auf das Smart TV verbinden.

In einem Test wurde festgestellt, dass die sechsstellige Nummer in einer XML Datei über HTTP an den entsprechenden Fernwartungsserver geschickt wird. Dies birgt einige Risiken und verletzt das eigentliche Prinzip der Authentifizierung zur Fernwartung nach dem vorgestellten Schema. An dieser Stelle sei erwähnt, dass kein Test mit einem Fernwartungsmitarbeiter durchgeführt wurde, sodass nicht bestätigt werden konnte, ob der zugreifende Mitarbeiter von der gleichen Lokation wie der empfangenden Server der sechsstelligen Nummer stammt.

Der angezeigt Code auf dem Fernseher wird dem Supportmitarbeiter über einen sogenannten „Out-of-Bounds-Channel“ (OoB Channel) mitgeteilt. Dies hat den Sicherheitsvorteil, dass sich keine unbeteiligten dritten Zugang zum Smart TV verschaffen können. Dadurch, dass der Nutzer per Telefon mit dem Supportmitarbeiter verbunden ist, reguliert der Nutzer den Zugriff auf sein Smart TV. Nur wenn er die Nummer mitteilt, ist es für diesen Mitarbeiter möglich sich mit dem Smart TV zu verbinden.

In Falle des vorgestellten Testgerätes, wird die sechsstellige Nummer über HTTP an den Server geschickt. Das bedeutet, dass alle Knoten auf dem Weg vom Fernseher zum empfangenden Server diese Datei empfangen und lesen können, da es sich um das unverschlüsselte HTTP Protokoll zur Übertragung handelt. Da HTTP nicht das Schutzziel der Vertraulichkeit erfüllt, kann nicht sichergestellt werden, dass kein dritter, mit entsprechender Software, Zugriff auf das Smart TV erhält.

Das Schutzziel der Vertraulichkeit muss dennoch nicht erreicht werden, das ein solches „Pairing“ Protokoll auch ohne die Übertragung der Nummer funktioniert. Damit eine sichere Verbindung aufgebaut werden kann, darf die Nummer nur über den OoB Channel übertragen werden.

Möglichkeit einer DoS Attacke

Der zweite Abschnitt beschreibt eine Lücke welche in Test vier gefunden wurde. Dabei wurde festgestellt, dass das Smart TV unbegrenzt Befehle von einem registrierten Gerät annimmt, ohne die Verbindung nach einer gegebenen Grenze, zu blockieren.

In einem zusätzlichen Test wurde getestet, wie viele Befehle das Smart TV annehmen kann, ohne dass Befehle verloren gehen oder ein Fehlverhalten des Gerätes erzeugt wird. Hierfür wurde der Testaufbau aus Kapitel 4.4.1 verwendet. Mit dem Test Laptop wurde in einer Endlosschleife, jeweils ein Kennungspaket ID_{Device} plus ein Befehl $Com_{Command}$ gesendet.

Die Befehle wurden von unserem Testgerät ohne Pause gesendet. Es wurde festgestellt, dass das Smart TV diese Befehle alle annimmt und versucht diese auch zu bearbeiten. Dadurch ergibt sich, dass der Fernseher unbenutzbar wird. Eine Bedienung über die originale Fernbedienung ist nicht mehr möglich. Auch andere registrierte Geräte können nicht verwendet werden bis die Befehlsfolge beendet ist.

Nach kurzer Zeit, reagiert der Fernseher nicht mehr und startet neu. So ist es möglich, den Fernseher unbrauchbar zu machen. Zusätzlich dazu, müssen die Pakete der TCP/IP Verbindung keine Kennungspakete sein, wie ein zweiter Test bestätigte. Hierbei wurde ID_{Device} Paket an das Smart TV gesendet, sondern ein beliebiges Paket an die IP Adresse des Smart TV auf Port 1234. Auch hier stellte sich das getestete Verhalten ein. Nach kurzer Zeit reagiert das Smart TV nicht mehr und startet neu. Das bedeutet, dass sich die Schwachstelle auch auf nicht registrierte Geräte ausweitet. Ist es einem Angreifer möglich auf das Netzwerk zuzugreifen, kann dieser den Gebrauch des Smart TVs, bei Nutzung einer Netzwerkverbindung, beträchtlich einschränken. Eine Verteilung eines solchen Tools könnte an beliebigen Orten im Netzwerk geschehen, z.B. als schädliche Software auf einem PC im Netzwerk.

Dieser Fehler kann behoben werden, indem die Befehle von Fernbedienungen nacheinander abgearbeitet werden und nur eine bestimmte Anzahl von Befehlen pro Zeiteinheit angenommen werden können. Wird diese Größe überschritten, werden neue Pakete nicht mehr angenommen oder ein Gerät temporär blockiert.

Analyse der HTTPS Verbindungen

Im ersten Test in Kapitel 4.1.1 wurden HTTPS Verbindungen entdeckt, welche nicht identifiziert werden konnten in Bezug auf den Inhalt und Nutzen dieser Verbindungen. Der Testaufbau aus Grafik 5 wurde benutzt um einen weiteren Test durchzuführen. Hierfür wurde der HTTPS Port 443 standardmäßig auf dem Gateway des Fernsehers blockiert um die damit die Verbindungen zu unterbinden bzw. sie auf das unsichere HTTP Protokoll umzuleiten.

Der Start des Fernsehers wurde dadurch nicht beeinträchtigt. Auf dem Gateway ergab sich das Bild, dass die HTTP Verbindungen weiterhin aufgebaut werden und die benötigten Daten bezogen werden. Die HTTPS Verbindungen werden aufgebaut, erhalten nach kurzer Zeit einen Timeout und werden abgebrochen. Weiter werden im Folgenden zu zusätzlichen HTTPS Server die gleichen Verbindungen aufgebaut. Dennoch werden alle Kommunikationen über HTTPS blockiert und werden abgebrochen. Ein zusätzlicher Test sollte den Test in Kapitel 4.2.1 erweitern, indem die Smart Features ohne HTTPS untersucht worden.

Der Testablauf dieses Zusatztests war wie folgt: Der Port 443 wurde im laufenden Betrieb blockiert. Danach wurde der eingebaute Internetverbindungstest durchgeführt. Ist dieser positiv, wird mit der Nutzung der Smart Features fortgefahren. Nachdem der Port 443 blockiert wurde ergab sich zunächst keine Änderungen am Smart Hub und Fehlermeldungen blieben aus. Auch der Internetverbindungstest bestätigte, dass eine Verbindung zum Internet möglich war. Dennoch war ein Starten des Smart Hubs nicht mehr möglich. Nach überdurchschnittlich langer Wartezeit, erschien eine kryptische Fehlermeldung auf dem Fernseher. Im Hintergrund konnte anhand des Traffics festgestellt werden, dass wie im ersten Test zahlreiche HTTPS Verbindungen zu unterschiedlichen Servern aufgebaut werden. Da diese Verbindungen nicht aufgebaut werden können, wird die Smart Hub Verbindung unterbrochen und schlägt fehl. Die HTTP Verbindungen laufen problemlos weiter, auch die XML Dateien oder Aktualisierungsinformationen zu den eingebauten Applikationen werden weiterhin übertragen.

Zusammengefasst kann gesagt werden, dass durch die Blockierung des HTTPS Ports 443 keine Nutzung von Smart Features innerhalb des Smart Hubs möglich sind. Der Datendienst HbbTV, welcher in Kapitel 4.3.1 beschrieben wird, ist davon nicht betroffen.

5 Ausblick

Im letzten Kapitel der Arbeit wird zunächst die Kommunikation der festgestellten Datenschutz- und Sicherheitslücken mit Samsung bzw. den TV Sendern beschrieben. In einem weiteren Abschnitt wird ein Ausblick gegeben zu „Related Work“, speziell auf eine wissenschaftliche Arbeit über eine Ausarbeitung von Kapitel 4.3 und Hinweise zu weiteren Forschungsthemen im Bereich Smart TV. Im letzten Abschnitt der Arbeit werden die Ergebnisse nochmals kurz resümiert und ein Fazit gezogen.

5.1 Kommunikation mit Samsung und TV Sendern

Die Ergebnisse aus Kapitel 4.1, 4.2 und 4.4 wurden der Firma Samsung kommuniziert. Hierbei handelt es sich um Design und Implementierungsfehler, welche durch Softwareupdates behoben werden können, da hierdurch sicherheits- und datenschutzrelevante Lücken entstanden sind. Dabei wurden folgende Punkte kommuniziert:

- Fehlende Sicherheit bei der Übertragung der Datenpakte über HTTP
- Unsichere Übertragung einer „Unique ID“ („DUID“ und MAC Adresse)
- Fehlende SSL Prüfung des Browser
- Unzureichende Prüfung des „Pairing“ Vorgangs

Bis zum Abschluss der Arbeit wurden die letzten beiden Punkte bestätigt und es wurde mitgeteilt, dass diese Lücken geschlossen werden. Die fehlende Sicherheit zur Übertragung via HTTP wurde bestätigt, diesen wurde keine Relevanz eingeräumt und wird nicht geändert. Zur Übertragung der „DUID“ bei jedem Start des Smart TVs, gab es keine Aussage seitens Samsung. Seit dem Test der einzelnen Szenarien und der Kommunikation mit Samsung gab es diverse Updates der gesamten Firmware bzw. der einzelnen Komponenten, vor allem des Browsers. Dennoch wurde keiner der bisher angesprochenen Fehler behoben.

Die Lücke und die daraus entstehenden Risiken aus Kapitel 4.3 wurden den betroffenen Sendern, bei denen der HbbTV Dienst auf dem vorgestellten Testgerät genutzt wird, kommuniziert.

- Direktes Tracking durch Sender: Initiale Pakete, periodische Anfragen und Cookies
- Indirektes Tracking durch Dritte: Abhören durch dritte (WPA2 Schwachstelle)
- Verletzung des mentalen Modells

Die Gefahren wurden insgesamt von einem Senderverbund bestätigt. Dieser kommunizierte, dass man bereit sei, an diesen Lücken zu Arbeiten und sie zu beheben. Dabei wurde nur das Indirekte Tracking behandelt. Bezug auf das mentale Model und das direkte Tracking wurde mit Implementierungsgründen gerechtfertigt. Die restlichen angeschriebenen Sender meldeten sich bis zu diesem Zeitpunkt nicht.

5.2 Weitere Forschungsthemen

In diesem Abschnitt soll der Blick auf andere Forschungen im Bereich Smart TV geworfen werden. Alle Tests dieser Arbeit zielten auf die Softwareumgebung und die Implementierung der einzelnen Subsysteme ab. Dabei wurden jeweils die Samsung eigenen Entwicklungen näher betrachtet. Es sollen nun drei Gebiete angesprochen werden, welche als Grundlage für neue Arbeiten dienen könnten.

Die Entwicklung der aktuellen Smart TV Serien ist rasant. Wie in der Einleitung der Arbeit schon erwähnt, kann das in dieser Arbeit verwendete Testgerät heutzutage, zumindest aus der Sicht der Nutzung der Smart Features, als veraltet bezeichnet werden. Immer neue Techniken kommen auf den Markt die immer mehr potenzielle Angriffsmöglichkeiten bieten.

Zusätzlich wurde in dieser Arbeit lediglich die Softwareebene betrachtet. Eine Untersuchung des darunterliegenden Betriebssystems oder der Hardwarekomponenten wurde nicht durchgeführt. Thema einer weiteren Untersuchung könnte die Hardwareebene sein und etwaige Schwachstellen mit denen der Softwareebene zu kombinieren.

Zuletzt muss erwähnt werden, dass alle Tests nur auf Smart TVs der Marke Samsung durchgeführt wurden. Um einen Eindruck des gesamten Marktes zu erhalten müssen natürlich auch Geräte anderer Hersteller in Betracht gezogen werden, welche mit unterschiedlichsten Systemen Arbeiten. Ob die gefunden Fehler nur Samsung spezifisch sind oder ob andere Hersteller ähnliche oder andere Schwachstellen haben, könnte durch weitere Test in einer zusätzlichen Arbeit festgestellt werden.

Am Ende der Arbeit wurde ein Artikel veröffentlicht, ähnlich zu dieser Arbeit. Der Bericht unter dem Namen „The TV is Watching You“ beschreibt Schwachstellen in aktuellen Smart TV Geräten der Firma Samsung [33]

5.3 Zusammenfassung und Fazit

Im abschließenden Fazit soll eine kurze Zusammenfassung der Ergebnisse der vorgestellten vier Tests erfolgen. Die gesammelten Ergebnisse werden dann in Bezug mit der eigentlichen Zielstellung der Arbeit verglichen. Das Ziel der Arbeit war anfänglich nicht klar definiert. Eine klare Abgrenzung der Ziele konnte nicht erfolgen, da bisher keine vergleichbaren Arbeiten auf diesem Gebiet existieren. Daher wurde auch das hier benutzte Testgerät der Firma Samsung ausgewählt, um einen Einstiegspunkt für die Arbeit zu haben, da dieses eine entsprechende Schwachstelle besitzt.

Die einzelnen Ergebnisse der hier vorgestellten Tests, vor allem Kapitel 4.1, 4.2 und 4.4 lassen einen unfertigen Eindruck des eingesetzten Systems auf dem Smart TV entstehen. Die Ergebnisse in 4.3 sind nicht auf das Smart TV selbst zurückzuführen, sondern auf den noch jungen HbbTV Standard. Dennoch kann man sagen, dass unser Testgerät eher mit funktionaler Quantität glänzt, anstelle von Qualität in Bezug auf die Implementierung der Smart Features. Dabei wurden auch die Aspekte Datenschutz und Sicherheit klar vernachlässigt. Die durchgeführten Tests erweiterten die Schwachstellen der Samsung Smart TV Serie.

Zusammengefasst kann man sagen, dass sich bei den gefundenen Problemen und relevante Schwachstellen handelt, welche von den entsprechenden Stellen behoben werden sollten. Bei den vorgestellten Schwachstellen und den dazugehörigen Angriffsszenarien handelt es sich um keine akademischen Szenarien sondern um tatsächliche relevante Schwachstellen.

Anzumerken ist, dass die Arbeit keine neuartigen Fehler aufdeckt, sondern es sich bei allen Ergebnissen um bekannte Schwachstellen handelt, welche zuvor bei anderen Systemen entdeckt wurden und teilweise auf diesen nicht mehr existieren. Als Beispiel sei hier die fehlende Überprüfung der HTTPS Zertifikate genannt. Aktuelle Browser auf anderen Systeme, Computer oder mobile Geräte, haben diese Prüfung in allen aktuellen Browsern korrekt implementiert.

Hieraus lässt sich schließen, das in den hier getesteten Smart TV Systemen, viele Fehler hätten vermieden werden können, wenn man auf bekannte Implementierungen gesetzt hätte (Bsp. Opera Browser [12]). Samsung selbst, stellt eine entsprechende Lösung schon in Aussicht. Durch die Portierung eines etablierten Betriebssystems, zum Beispiel Android, auf den Fernseher, könnte die Qualität der Implementierung deutlich angehoben werden und bekannte Fehler, unter Umständen, vermieden werden [15].

Dennoch kann diese Arbeit keine Aussage über den Status aktueller Smart TVs treffen, da sich die Arbeit nur mit Geräten der Firma Samsung beschäftigte. Um einen besseren Eindruck über den gesamten Markt zu erlangen, müssten mehrere Geräte anderer Firmen in Betracht gezogen werden und diese unter den gleichen Aspekten analysiert werden. Zusätzlich verfügt das vorgestellte Testgerät über weit mehr Funktionalität wie in der Arbeit untersucht. Auch die weitere Entwicklung des Smart TV Marktes muss betrachtet werden. Immer neue Funktionen, mit potenziellen Schwachstellen, werden auf das Smart TV portiert. Eine Webcam für Videotelefonie gehört mittlerweile zur Ausstattung aktueller Smart TVs. Auch die darunterliegende Hardware des Systems wurde nicht behandelt. All dies führt zu zusätzlichen Betrachtungsmöglichkeiten für weitere Arbeiten.

Literatur

[1] Akamai Network. Akamai Fast Forward. http://www.akamai.de/, 2012.

[2] Amazon Web Services. Amazon Elastic Compute Cloud. http://aws.amazon.com/de/ec2/, 2012.

[3] Andreas Spillner, Tilo Linz. Basiswissen Softwaretest: Aus- und Weiterbildung zum Certified Tester - Foundation Level nach ISTQB-Standard. dpunkt.verlag Seite 114;149, 2012.

[4] Anti-Phishing Working Group, Inc. Anti-Phishing Working Group, Inc. http://www.antiphishing.org/, 2012.

[5] Anti-Phishing Working Group, Inc. Anti-Phishing Working Group, Inc. Quartal Report 2Q 2012. http://docs.apwg.org/reports/apwg_trends_report_q2_2012.pdf, 2012.

[6] Arbeitsgemeinschaft Fernsehforschung (AGF). Fernsehkonsum: Tägliche Sehdauer der Deutschen in Minuten nach Altersgruppen (6. Dezember 2012). http://de.statista.com/statistik/daten/studie/2913/umfrage/fernsehkonsum-der-deutschen-in-minuten-nach-altersgruppen/, 2012.

[7] ARD. Die ARD auf der IFA 2012. http://www.ard.de/intern/-/id=2606668/property=download/nid=1886/pmuz25/Die+ARD+auf+der+IFA.pdf.pdf, 2012.

[8] Bitkom, Jürgen Boyny. Smart TV: Der Markt in Zahlens. http://www.bitkom.org/files/documents/smart_tv_der_markt_in_zahlen_l_juergen_boyny_l_gfk.pdf, 2011.

[9] CA Browser Forum. Guidelines for the issuance and management of extended validation certificates. CA Browser Forum, 2007.

[10] Chartbeat. Chartbeat. http://chartbeat.com/, 2012.

[11] Claudia Eckert. IT Sicherheit - Konzepte - Verfahren - Protokolle. Oldenbourg Wissenschafts Verlag, 2012.

[12] Developement Opera. Creating Web Content for TV. http://dev.opera.com/tv, 2012.

[13] Frank Bitzer, Klaus M. Brisen. Digitale Signaturen: Grundlagen, Funktionen und Einsatz. Springer Verlag, 1999.

[14] Gabriel Menezes Nunes. Sony Bravia Remote Denial of Service - CVE-2012-2210. http://archives.neohapsis.com/archives/bugtraq/2012-04/0043.html, 2012.

[15] golem.de: IT News für Profis. Google-Fernseher von Samsung kommt Ende 2012. http://www.golem.de/news/google-tv-google-fernseher-von-samsung-kommt-ende-2012-1208-94254.html, 2012.

[16] Google Inc. Google Analytics. http://www.google.com/analytics/, 2012.

[17] Google Inc. Google Public DNS. https://developers.google.com/speed/public-dns/, 2012.

[18] Google Inc. Google TV Pairing Protocol. https://developers.google.com/tv/remote/docs/pairing, 2012.

[19] Heise Online. http://www.heise.de. http://www.heise.de, 2012.

[20] Heise Security. Sendepause durch Firmware-Lücken. http://www.heise.de/security/meldung/Sendepause-durch-Firmware-Luecken-1557589.html, 2012.

[21] IEEE Standard. IEEE 802.11i. http://standards.ieee.org/getieee802/download/802.11i-2004.pdf, 2004.

[22] Institut für Rundfunktechnik GmbH. Hybrid Broadcast Broadband TV. http://www.hbbtv.org/, 2012.

[23] Institut für Rundfunktechnik GmbH. Hybrid Broadcast Broadband TV. http://www.hbbtv-infos.de/sender/hbbtv-senderliste.php, 2012.

[24] Institut für Rundfunktechnik GmbH. Hybrid Broadcast Broadband TV Standard. http://www.etsi.org/deliver/etsi_ts/102700_102799/102796/01.02.01_60/ts_102796v010201p.pdf, 2012.

[25] Johannes Buchmann. Einführung in Kryptographie. Springer Verlag, 2010.

[26] Marco Ghigleri, Florian Oswald, Dr. Erik Tews. HbbTV -I Know What You Are Watching. Wird veröffentlicht auf dem 13. IT Sicherheitskongress, Mai 2013 Bonn, 2012.

[27] Microsoft. Schutzmechanismen in Outlook (2007) gegen Viren, Spam und Phishing. http://office.microsoft.com/de-de/outlook-help/schutzmechanismen-in-outlook-gegen-viren-spam-und-phishing-HA001230571.aspx, 2009.

[28] Network Working Group. The Base16, Base32, and Base64 Data Encodings. http://tools.ietf.org/html/rfc4648, 2006.

[29] Network Working Group. RFC HTTP 1.1. http://tools.ietf.org/html/rfc2616, 2012.

[30] Network Working Group. RFC HTTPS over TLS. http://tools.ietf.org/html/rfc2818, 2012.

[31] Pagedesign GmbH. utrace. http://www.utrace.de/, 2012.

[32] Panasonic. Panasonic Viera Produktübersicht. http://www.panasonic.de/html/de_DE/Produkte/Flachbildfernseher/Willkommen/27825/index.html, 2012.

[33] ReVuln. The TV is Watching You. http://nakedsecurity.sophos.com/2012/12/12/samsung-tv-vulnerability/, 2012.

[34] Samsung. Samsung All-Share Technologie - Smart Verbinden. http://smart-tv.samsung.de/verbinden, 2012.

[35] Samsung. Samsung App Store. http://www.samsungapps.com/, 2012.

[36] Samsung. Samsung Remote Google Play Store. https://play.google.com/store/apps/details?id=com.samsung.remoteTV&hl=de, 2012.

[37] Samsung. Samsung Smart Hub. http://de.samsung.com/de/microsites/smarttv/anwendungen_feature.aspx, 2012.

[38] Samsung. Samsung Smart TV. http://www.samsung.de/smart-tv/, 2012.

[39] Samsung. Samsung Smart TV UE40D6200. http://www.samsung.com/de/consumer/tv-audio-video/television/led-tv/UE40D6200SXZG, 2012.

[40] Samsung. Samsung Smart TV UE40ES6300. http://www.samsung.com/de/consumer/tv-audio-video/telision/led-tv/UE40ES6300SXZG, 2012.

[41] Samsung. Testing Your Application on a TV. http://www.samsungdforum.com/upload_files/files/guide/data/html/html_3/getting_started/test_app_on_tv.html, 2012.

[42] Samsung Developer. Get DUID. http://www.samsungdforum.com/upload_files/files/guide/data/html/html_3/api_reference/javascript_apis/sef_plugin_api/GetDUID.html, 2012.

[43] SamyGo Remote. SamyGo Remote Google Play Store. https://play.google.com/store/apps/details?id=de.quist.app.samyGoRemote&hl=de, 2011.

[44] SamyGo Remote. SamyGo Remote on Github. https://github.com/tomquist/SamyGo-Android-Remote/, 2012.

[45] Statista. Anzahl der Smart TV-Haushalte in Deutschland im Jahr 2010 und Prognose bis 2016. http://de.statista.com/statistik/daten/studie/208236/umfrage/prognose-zur-entwicklung-der-smart-tv-haushalte-in-deutschland/, 2012.

[46] W. Cerroni, M. Ramilli. Man-in-the-Middle Attack to the HTTPS Protocol. Univ. of Bologna, Bologna, 2009.

[47] WC3. SOAP Version 1.2. http://www.w3.org/TR/soap/, 2012.

[48] Webtrek. Webtrek Live Raw Fast. http://www.webtrekk.com/de/home.html, 2012.

[49] Wireshark Foundation. Wireshark. http://www.wireshark.org/, 2012.

[50] yWorks. yEd Graph Editor. http://www.yworks.com/en/index.html, 2012.

[51] ZVEI. 2,2 Fernseher stehen in jedem deutschen Haushalt. http://www.zvei.org/Presse/Presseinformationen/Seiten/Jeder-fuenfte-ist-ein-Smart-TV.aspx, 2012.