



**PLINTA - PLattform för säker INTegration av Användarfunktioner i bil
Publik slutrapport**

**Vinnovaprojekt inom FFI - Fordonstrategisk Forskning och Innovation
Vinnova reference, Diariernr: 2012-01703**

**PLINTA-projektet genomfördes mellan 2013-01-01 och 2015-01-30
Projektmedlemmar: Semcon, Pelagicore, HiQ och Volvo Personvagnar**

Sekretess

Innehållet i denna rapport är öppet för allmänheten. Dokumentet är avsett som en introduktion till PLINTA-projektet.

Datum 2015-01-30

Innehållsförteckning

1	Sammanfattning.....	3
2	Bakgrund	4
3	Syfte.....	4
4	Genomförande	5
5	Resultat och leverabler	6
5.1	IHU-plattform	6
5.2	HMI.....	7
5.3	Secure Gateway	8
5.3.1	Nätverkslager.....	8
5.3.2	Meddelandelager	9
5.3.3	Servicelager	9
5.4	Applikationsinstallation.....	9
5.5	Användaranpassning.....	10
5.6	Smart Device Link.....	10
5.7	Fjärrkontroll.....	10
5.8	Ansiktsgenkänning	11
5.9	Eftermarknadsinstallation	12
5.10	Bidrag till FFI-mål	12
6	Spridning och publicering	13
6.1	Publik presentation	13
6.2	Kunskaps- och resultatspridning.....	13
7	Slutsatser och fortsatt forskning	14
7.1	Slutsatser.....	14
8	Deltagande parter och kontaktpersoner	15

Kort om FFI

FFI är ett samarbete mellan staten och fordonsindustrin om att gemensamt finansiera forsknings-, innovations- och utvecklingsaktiviteter med fokus på områdena Klimat & Miljö samt Säkerhet. Satsningen innebär verksamhet för ca 1 miljard kr per år varav de offentliga medlen utgör hälften.

För närvarande finns fem delprogram Energi & miljö, Fordons- och trafiksäkerhet, Fordonsutveckling, Hållbar produktionsteknik och Transporteffektivitet. Läs mer på www.vinnova.se/ffi

1 Sammanfattning

PLINTA - PLattform för säker INTegration av Användarfunktioner i bil - är ett Vinnovafinansierat innovationsprojekt. Projektet ingår i Vinnovaprogrammet "Fordonstrategisk Forskning och Innovation" (FFI) och portfolion för Elektronik, mjukvara och kommunikation. Partners i projektet är Semcon, Pelagicore, HiQ och Volvo Car Corporation.

Syftet med projektet är att hitta lösningar för att minimera säkerhetsriskerna med användning av mobila enheter i samband med körning och samtidigt ge svenska företag ett försprång inom mjukvarutjänster för fordonsbranschen. De föreslagna lösningarna introducerar en infotainmentmiljö som är tillräckligt mångsidig och adaptiv för att kontinuerligt kunna följa den mobila utvecklingen och samtidigt vidhålla positionen som förarens föredragna gränssnitt. Projektet har resulterat i:

- Ett flexibelt och öppet Infotainment Head Unit (IHU)-system med en stor touchscreen har integrerats i en Volvo XC60. Systemet är baserat på en plattform som möjliggör utveckling av applikationer som enkelt kan laddas ner och installeras. Applikationerna kan dessutom utnyttja och kontrollera fordonsspecifik funktionalitet.
- Ökad systemsäkerhet har åstadkommits genom att använda ett så kallat "Secure Gateway"-koncept för att koppla samman fordonets kärnfunktionalitet och konsumentorienterade infotainmentfunktioner. Konceptet kan också användas till att applicera ny extern hårdvara, vilken kan användas av applikationer på ett enkelt och flexibelt sätt.
- Ett anpassningsbart människa-maskin-gränssnitt (Human-Machine Interface, HMI) har utvecklats. Detta HMI kan hantera olika skärmupplösningar och olika fordonskonfigurationer utan mjukvaruändringar. Detta är ett initialt steg mot att minska beroendet av IHU-utveckling och fordonsprogram. Gränssnittet är designat för att minimera förardistraktion, och bland annat ansiktsigenkänning har testats i detta syfte i prototypform.
- En applikation med funktionalitet som fjärrkontroll för IHU:er samt fjärridentifiering av förare har utvecklats för iPhone.
- Referensplattformen är molnbaserad, vilket möjliggör användaranpassning och låter systemet förutse förarens intentioner baserat på en kombination av data från olika datakällor. Exempelvis kan positionsdata från kalendern användas automatiskt som input till fordonets navigationssystem.
- En kombination av en flexibel utvecklingsmiljö för IHU:er och en säker anslutning till fordonets kärnfunktionalitet möjliggör enkel utveckling av ny funktionalitet, såsom applikationer för ekokörning.

En vanlig missuppfattning är att många av ovanstående funktioner redan har kommersialiserats inom fordonsbranschen. Detta är emellertid långt ifrån sanningen. Att lägga till funktionalitet eller nya användarupplevelser i dagens infotainmentsystem är mycket tidskrävande och kostsamt. De huvudsakliga barriärerna är:

- Nuvarande överenskommelser och kommersiella kontrakt inom branschen är inte anpassade efter öppna plattformar.
- Ekosystemet ligger fortfarande efter funktionaliteten. Det finns för många alternativa plattformar och ingen identifierad drivande kraft med en tydlig affärsmodell. En konsolidering krävs för att kunna bygga en stabil och långvarig plattform helt oberoende av fordonslivscyklar.

För att överbrygga dessa barriärer är förhoppningen att PLINTA kan inspirera den svenska fordonsbranschen att hitta nya möjligheter för att realisera innovativa idéer. En viktig del av PLINTA-projektet har varit att demonstrera och framhäva möjligheterna med en öppen och flexibel plattform.



2 Bakgrund

Utvecklingsmöjligheterna inom fordonsbranschen verkar nuförtiden vara gränslösa. Bilar är uppkopplade mot molntjänster, har multipla displayer, en myriad av sensorer och är till och med autonoma. För att kunna utnyttja alla dessa möjligheter till fullo och samtidigt tillhandahålla en suverän användarupplevelse, behöver tillverkarna förkorta utvecklingscyklerna, minska svårighetsgraden för att implementera ny funktionalitet samt ta till sig feedback från användare.

Användarnas förbindelse till den uppkopplade upplevelsen är mobila enheter som ständigt uppdateras. Det kommer alltid att vara frestande att använda dessa enheter i bilen, trots att de är distraherande för förarna. Apple Car Play och Android Projected Mode är initiativ som strävar efter att göra mobila enheter till navet i infotainmentsystemet, samtidigt som bilens hårdvara används för att göra användningen säkrare och mer intuitiv.

Det tar tre år att utveckla en bil som kan leva upp till 15-20 år och hittills har branschen huvudsakligen fokuserat på att leverera uppdateringar fokuserade på att lösa buggar. Konkurrensen mellan biltillverkare har skiftat fokus till att kretsa kring erbjudanden i den digitala domänen, och den dominerande 17-tums-tryckskrämen i Teslabilar har bringat uppmärksamhet till det digitala paradigmskiftet.

För att positionera bilen som en säker möjliggörare för digitala innovationer behövs nya mjukvaruarkitekturer och utvecklingsmetoder.

För att erbjuda mer digital innovation måste bilindustrin minska kostnaden för infotainmentutveckling. En logisk väg är att dela utvecklingen av standardmjukvara tvärsöver branschen och koncentrera resurserna på differentierande element som uppfattas som gediget innovativa. Allt detta kräver användning av open source-komponenter, branschstandardisering och stark utveckling.

3 Syfte

PLINTA har som mål att skapa en handfast referenslösning med tillhörande demonstrator av en modern infotainmentplattform, baserad på öppen källkod och en säker arkitektur. Projektet drivs genom agil systemutveckling.

Demonstratorerna fungerar som ”proof of concept”, såväl som ett fundament för att marknadsföra fördelarna och därigenom etablera drivkraft och entusiasm för att föra lösningarna närmare kommersialisering.

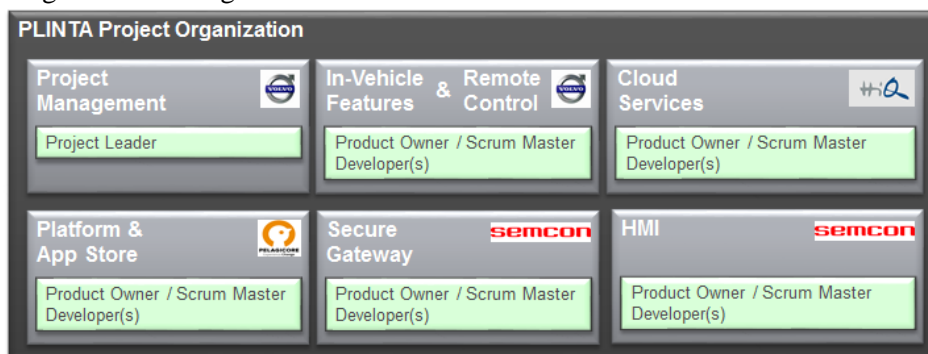
Plattformen ämnar minska tröskeln för utveckling av lösningar som förhöjer användarnas upplevelser och på så sätt stärka Sveriges position i den internationella fordonsbranschen. Plattformen bidrar till utvecklingen av säkrare bilar.

Viktiga utvecklingsmål:

- Linuxbaserad infotainmentplattform baserad på öppen källkod
- Referens-HMI som möjliggör enkel och säker användning av ny funktionalitet
- Utforska möjligheter för smartphoneintegration
- Koncept för säker och managerad tillgång till fordonsnätverk
- Utforska möjligheter för innovation genom molnuppkoppling
- Utforska metoder av projektutförande mellan deltagande partners

4 Genomförande

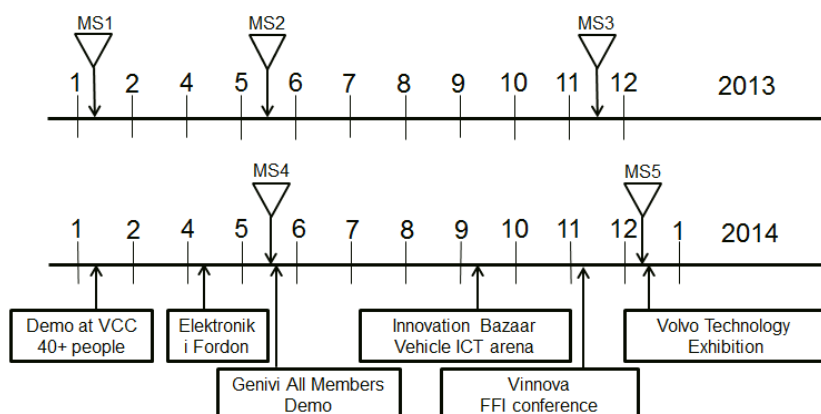
Projektet har genomförts i form av ett samarbete mellan Volvo Car Corporation, HiQ, Semcon och Pelagicore. I enlighet med kontraktsförbindelser för PLINTA-projektet bär respektive part ansvar för implementering och integration av komponenter i referenssystemet. Tillämpade högnivåroller och ansvarsfördelning illustreras i figur 1.



Figur 1: Projektroller och ansvarsfördelning fördelat per partner.

Projektet har utförts i enlighet med figur 2.

PLINTA MILESTONES AND PRESENTATION EVENTS



Figur 2: Tidsplan.

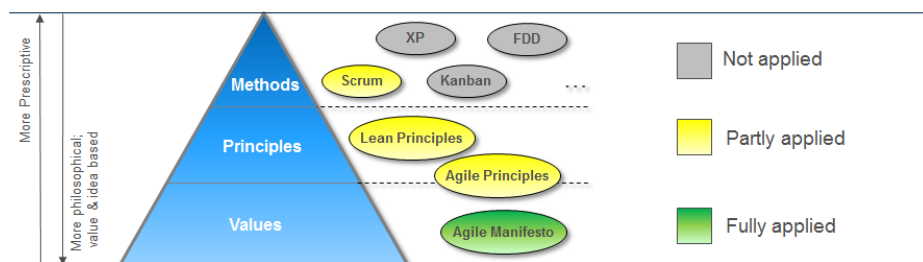
Projektet har genomförts med ett agilt tankesätt, baserat på manifestet för agil systemutveckling:

”Individer och interaktioner framför processer och verktyg
Fungerande programvara framför omfattande dokumentation
Kundsamarbete framför kontraktförhandling
Anpassning till förändring framför att följa en plan”

<http://agilemanifesto.org/iso/sv>

Övriga agila utvecklingsprinciper som har använts inom projektet illustreras i figur 3. Det har emellertid varit svårt att fullt ut arbeta med Agila metoder och principer eftersom parterna i PLINTA-projektet redan från start har haft egna dedicerade områden.

En projektwiki tillhandahållen av Semcon har använts för projektdokumentation och administration av backlog.



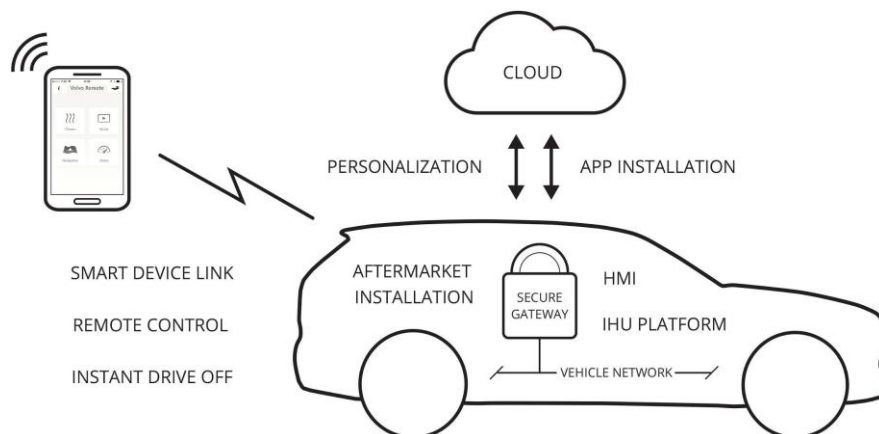
Process Highlights

- ✔ Four week sprints
- ✔ Sprint demo
- ✔ Common repository
- ✔ Sprint planning
- ✔ Retrospective

Figur 3: Agila principer som har tillämpats i projektet

5 Resultat och leverabler

Figur 4 nedan visar PLINTA-systemet och dess relation till omnämnda användningsscenarier.



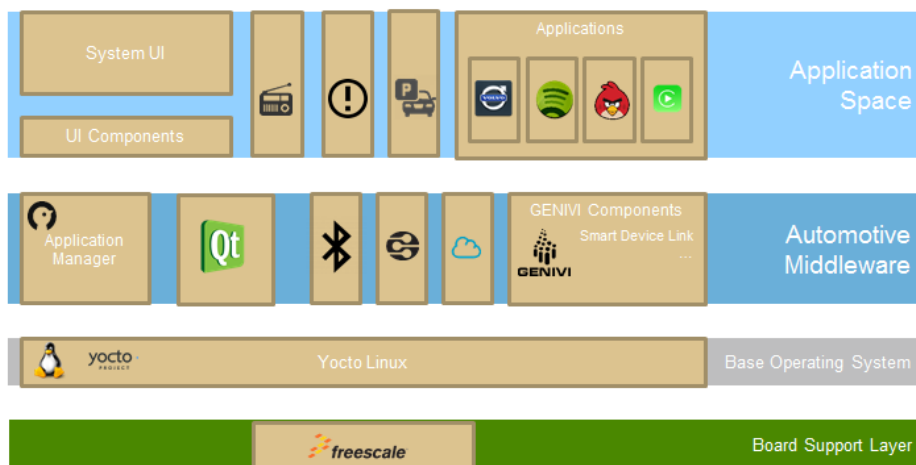
Figur 4: PLINTA-systemöversikt och användningsscenarier

5.1 IHU-plattform

IHU-plattformen i PLINTA-demonstratorn byggdes ovanpå en Yoctobaserad plattform och genom ett IVI (In-Vehicle Infotainment)-metalager uppfyller plattformen kraven för GENIVI-kompatibilitet. Ovanpå det gemensamma servicelagret användes Pelagicore Application Manager (AM) som värd för användargränssnittet och för att hantera applikationsinstallationer samt övervaka applikationslivscykeln. Ett högnivådiagram av arkitekturen som ligger till grund för PLINTA-demonstratorn visas i figur 5. Nedan följer en rudimentär begreppsdefinition:

- **GENIVI Alliance** – Ett ideellt konsortium vars mål är att etablera en global och konkurrenskraftig Linuxbaserad plattform för IVI-sammanhang, inklusive operativsystem och middleware.
- **Qt** – Ett plattformsoverskridande ramverk för användargränssnitt baserat på C++ och QML, ett CSS- och JavaScriptliknande språk. Qt, Qt Quick och tillhörande verktyg är utvecklade i form av open source.

- **Yocto** – Yoctoprojektet är ett open sourceprojekt som drivs av Linux Foundation. Målet med projektet är att producera verktyg och processer som möjliggör Linuxdistributioner för inbyggda system, oberoende av underliggande arkitektur och inbyggd systemmjukvara såväl som underliggande Board Support Layer.

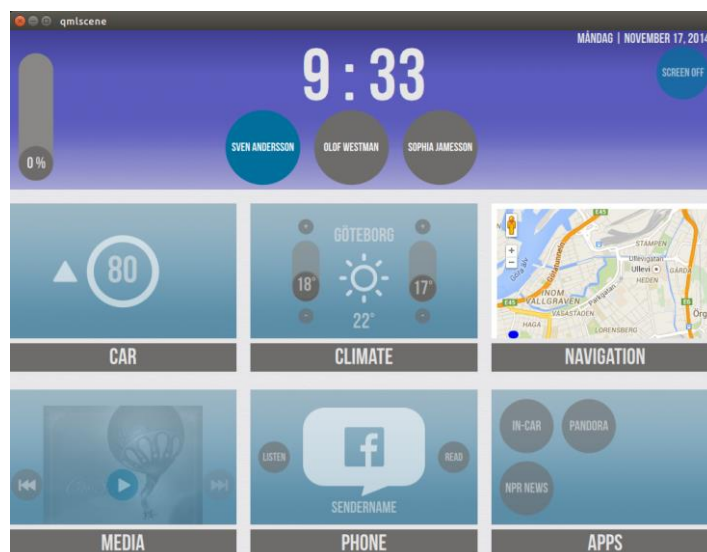


Figur 5: Översikt av PLINTA:s referenssystem.

5.2 HMI

Ett märkesoberoende användargränssnitt designades med avsikt att demonstrera PLINTA:s funktionalitet. Gränssnittet är designat för att enkelt tillhandahålla relevant information och möjliggöra snabb tillgång till viktiga och frekvent använda funktioner, med mål att låta användaren spendera mer tid på primära och sekundära uppgifter utan att bli distraherad av mindre viktiga dito.

Den välbalanserade informationsarkitekturen presenteras i ett överskådligt raster, se figur 6, som motverkar långa blickar och omständlig interaktion. Detta håller nere förarens distraktion i samband med användning. Gränssnittet tillhandahåller därför en säkrare och effektivare miljö än dagens lösningar för mobila enheter, genom att standardisera såväl visuell presentation som interaktionsparadigm för alla sammankopplade enheter och applikationer.



Figur 6: PLINTA:s användargränssnitt, med dess rasterlayout

De flesta regioner introducerar riktlinjer och regler för att minska förardistraktion, genom att kontrollera och begränsa förarens interaktion med IVI-system. HMI:t måste motsvara såväl OEM-krav som internationella och nationella riktlinjer såsom ESoP, NHTSA, AAM och JAMA. NHTSA:s riktlinjer betonar två användningssituationer (körning och icke-körning), vilka medför ökad komplexitet för applikationsutvecklare. De senaste standarderna inom Geniviprojektet specificerar emellertid hela fem olika distraktionsnivåer.

En separat rapport¹ presenterar ett HMI-säkerhetsramverk med avsikt att hantera komplexitet och underlätta utveckling utan att tynga processen med kontextuella restriktioner från fordonsbranschen, och istället möjliggöra innovativa tjänster som lever upp till förväntningar, riktlinjer och standarder. En annan fördel är att applikationer tillåts fungera likvärdigt även utan koordinering mellan utvecklare, vilket resulterar i en mer unison användarupplevelse.

Ramverket underlättar genom att informera om såväl distraktionsnivåer som ändringsförslag med avseende på aspekter som animeringar, bilder, ljud, input, knapp- och teckenstorlekar samt textlängd. Som ett konkret exempel på utfallet av detta, skulle en applikation automatiskt kunna visa större text och knappar såväl som att exkludera alla rörliga texteffekter i takt med att trafikkomplexiteten ökar, exempelvis i händelse av accelerationer, nattkörning och omgivande fordon.

5.3 Secure Gateway

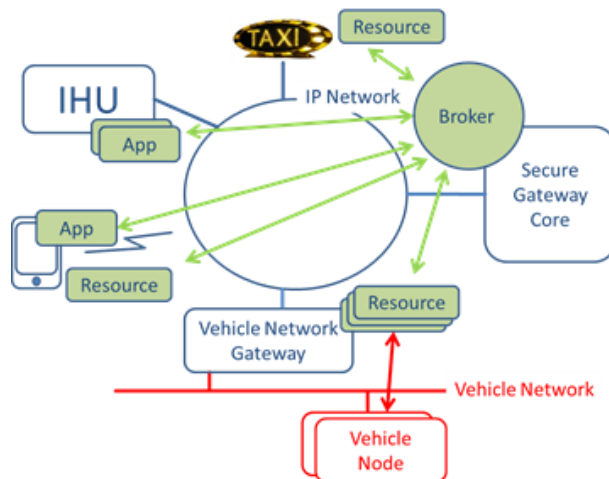
Ett viktigt användningsfall i moderna fordon, som också har utforskats inom PLINTA-projektet, är möjligheten att utnyttja fordonets sensorer och aktuatorer genom applikationer som körs i infotainmentmiljön. Detta innefattar tillgång från applikationer som körs i bilens egna system, antingen förinstallerade eller nedladdningsbara, såväl som applikationer som körs i mobila enheter anslutna till fordonet, exempelvis genom en WiFi-hotspot. Sådana anslutningar kräver en hög säkerhetsnivå för att motverka att illvilliga eller felaktigt fungerande applikationer åsamkar skada genom att påverka fordonet på oönskade sätt. Ett osäkrat system kan i värsta fall leda till att fordonet betar sig farligt.

Secure Gateway (SG) är ett koncept som adresserar behovet av applikationers tillgång till fordonsnätverket samtidigt som det tillhandahåller skydd för illvilliga eller felaktigt fungerande applikationer. Grunden i konceptet är ett nätverk bestående av tre lager – ett nätverkslager, ett meddelandelager och ett servicelager. Genom inkapsling ökar säkerheten. En schematisk bild visas nedan.

5.3.1 Nätverkslager

Grunden i Secure Gateway är ett TCP/IP-nätverk baserat på WiFi, Ethernet eller en kombination av båda. Standardprotokoll för kryptering som TLS/SSL och användandet av certifikat, enkel konfiguration enligt RFC 3927 samt Service Discovery genom mDNS (RFC 6762) har tillämpats. Dessa protokoll utgör fundamentet i Secure Gateway-konceptet och utgör det lägsta lagret i Secure Gateway – nätverkslagret. Lagret är schematiskt presenterat i figur 7 (blå färg). Kommunikation med fordonsnätverk, som CAN eller Flexray, är en viktig del av konceptet och hanteras genom nätverksgateways (inte att förväxla med Secure Gatewaykonceptet självt). IP-nätverket och fordonsnätverket möts i konceptets nätverksgateway och tillåts kommunicera med varandra med hjälp av meddelande- och servicelager i Secure Gateway. Observera att det inte görs någon direkt översättning mellan IP-paket och fordonsnätverkets dataformat, eftersom kommunikationen behöver säkras.

¹ Finns i den tekniska rapporten .Baserat på ett koncept utvecklat i ett annat Vinnvasponsrat projekt - SICS



Figur 7: SG med nätverkslager (blått) och meddelandelager (grönt)

5.3.2 Meddelandelager

Kommunikationen i Secure Gateway tillämpar ett ämnesbaserat prenumerationsformat där meddelanden skickas från en avsändare till en eller fler prenumeranter. Istället för att avsändaren vet vilka prenumeranter som meddelanden behöver nå, sköts denna kommunikation via en så kallad broker. Prenumeranterna registrerar sig själva i brokern och anger vilka ämnen som är relevanta. Alla meddelanden från avsändarna märks med ett ämne som förmedlas till brokern. Baserat på den information kan brokern bestämma vilka prenumeranter som behöver meddelandet, och således även vidarebefordra det. Secure Gateway använder MQTT som protokoll, och det är detta som utgör meddelandelagret som är lagret ovanför nätverkslagret. Alla meddelanden är krypterade och en gatekeeper i brokern använder accesslistor (ACL) för att fastställa accessrättigheter.

5.3.3 Servicelager

Det översta lagret i Secure Gateway är servicelagret. I detta lager återfinns två viktiga aktörer – applikationer och resurser. En resurs kan kort sammanfattas som en mjukvarurepresentation av en enhet, exempelvis en mediaspelare eller en temperaturgivare. Resursen tillhandahåller API:er som använder meddelandelagret (MQTT). Applikationer, exempelvis installerade i IHU, använder dessa API:er för att läsa data eller kontrollera ställdon. För att få åtkomst till enheter på fordonsnätverket används resurser i omnämnt nätverk. Certifikat används av gatekeepern i meddelandelagret för att autentisera både resurser och applikationer.

5.4 Applikationsinstallation

Alternativ för användarhantering och sätt att tillhandahålla ett återanvändbart ramverk för användarprofiler, inklusive privat data såsom kontodetaljer och inställningar, har utforskats inom projektet. Sandboxing-konceptet visade sig möjliggöra tydlig användarseparering utan att påverka applikationerna. Detta gör det möjligt att kombinera vanliga fordonsapplikationer, såsom radio och navigering, vilka tillhandahåller sömlös interoperabilitet mellan användare, samtidigt som det tillåter tredjepartsapplikationer på plattformen utan att informationen riskerar att läcka mellan användarprofilerna. En sandlåda implementeras genom att mjukvaran körs i en begränsad operativsystemsmiljö.

5.5 Användaranpassning

Under våren 2014 inledde PLINTA-projektet ett samarbete med Ericsson för att utveckla funktionalitet baserat på Ericsson Multiservice Delivery Platform (MSDP), vilket stödde lagring och nedladdning av personliga inställningar (anpassningar) för fordonet. Molnintegrering möjliggör helt ny funktionalitet.

Följande tjänster integrerades:

- Strömningstjänster, dvs. musikapplikationer såsom Rdio, där personliga spellistor och autentisering för tjänsterna laddas ner
- Anpassade fordonsklimatinställningar
- Google Calendar – möteshantering
- Google Maps – där en Google Calendar-händelse kan bestämma nästa destination, vilken sedan understöds av Maps i form av avstånd, tid och kartor

5.6 Smart Device Link

Smart Device Link är en open sourcekonkurrent till liknande teknik såsom Mirrorlink och Apple Car Play. Det möjliggör kontroll av smartphoneappar från ett fordonsgränssnitt genom röst-, pek- och knappstyrning. Arbetet involverade även uppgiften att förstå källkoden och designa ett gränssnittslager, för att möjliggöra ett HMI baserat på Qt istället för HTML5.

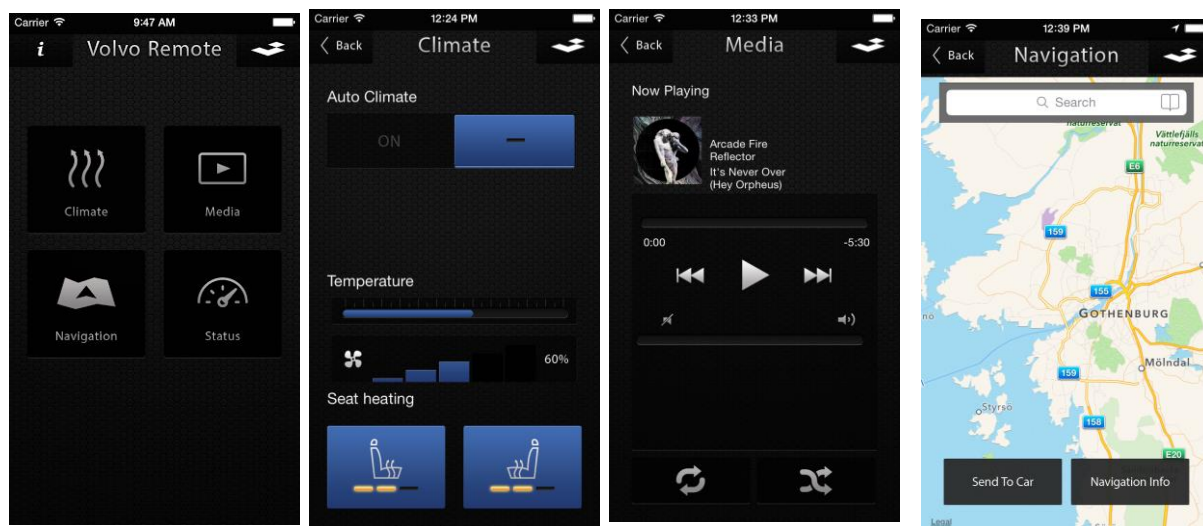
En implementering av PLINTA-plattformen demonstrerades på GENIVI All Member Meeting i Göteborg. Strategin var att släppa den utvecklade koden för Smart Device Link i Genivi-projektet. Under tiden hade emellertid Genivi-projektet släppt nyare källkod, vilket gjorde VCC-varianten inaktuell.

Open Sourceprojektet med ansvar för Smart Device Link har fortsatt att släppa nya versioner av källkoden som förbättrar funktionaliteten. Den har däremot begränsad support from fordonstillverkare, vilket delvis beror på dålig kvalitet av källkod och dålig dokumentation, men även på grund av att Apple och Google har släppt konkurrerande projekt som tillhandahåller motsvarande funktionalitet.

5.7 Fjärrkontroll

För att testa plattformen och utforska aspekter relaterade till smartphoneintegration, identifierades användningsfall under VCC-användartester i Kina. Bland annat tillåts baksätespassagerare att modifiera klimatinställningar, media och navigering genom en smartphone eller tablet-PC.

En fjärrkontrollapplikation skapades för iOS. När passagerare antrar bilen ansluter den mobila enheten till bilens WiFi-hotspot varpå fjärrkontrollen automatiskt hittar brokern och försöker ansluta (mer om anslutningshantering under sektion ”Secure Gateway”). Efter anslutning kontrollerar applikationen vilka resurser som är tillgängliga för styrning.



Figur 8: Grafiskt användargränssnitt för fjärrkontrollsapplikation.

Fjärrkontrollen är gjord för iOS och består av fyra separata vyer som återges i figur 8:

- **Vy 1:** adaptiv hemskärm för fjärrkontroll
- **Vy 2:** klimatvy som ger användaren möjlighet att kontrollera klimatinställningar
- **Vy 3:** mediavy som agerar fjärrkontroll för infotainmentsystemet. Inom projektet implementerades endast grundläggande funktionalitet, men denna vy har stor utvecklingspotential
- **Vy 4:** navigeringsvy som tillåter inmatning av adress till det inbyggda navigeringssystemet, vilket i bilens HMI kan upplevas som besvärligt och leda till förardistraktion. Genom att använda Google Maps eller iOS till att söka efter destinationer blir processen enklare och säkrare och tillåter dessutom passagerarstöd

Så kallad “Instant Drive Off” är ett av användningsfallen som identifierades i Kina. Konceptet går ut på att alla personliga inställningar och destinationer automatiskt är tillgängliga omedelbart när föraren antrar bilen.

Idag identifieras användaren via bilnyckeln. Räckvidden på den trådlösa sändaren i bilnyckeln är begränsad för att spara energi. Bluetooth Low Energy (BTLE) är en nyare understandard inom Bluetooth 4.0, som kan användas i enheter för att möjliggöra nyckellös tillgång. Målet är att identifiera användaren, starta infotainmentsystemet och förhandsladda användarens inställningar.

I PLINTA används en iPhone som Bluetooth LE-enhet med avsikt att ersätta traditionella nycklar. Telefonen tillkännager konstant sin närvaro och i samband med anslutning, inleder servern autentiseringsprocessen. När enheten är identifierad kommer infotainmentsystemet att informeras om användarens närvaro och således ladda ner inställningar från molnet. Systemet ska vara helt förberett redan när användaren öppnar dörren, och möjliggöra omedelbar användning, så kallad Instant Drive Off.

Googlekonton skapades för tre användare med olika kalenderevent på olika platser. I samband med identifiering hämtas förarens personliga inloggningsuppgifter från molnet och används för att logga in med Googlekontot. Googles API används för att hämta kalenderdata, inklusive positioneringsinformation. Läget, om tillgängligt, förmedlas till navigeringssystemet och en eller flera rutter kan planeras.

5.8 Ansiktsgenkänning

Ett ansiktsgenkänningssystem adderades till PLINTA-plattformen. Avsikten var att låta infotainmentsystemet känna av förarens uppmärksamhet och använda denna information för att utföra lämpliga åtgärder, såsom att

stänga ner distraherande applikationer. Ansiktsigenkänningen implementerades med hjälp av en webbkamera och Open CV-biblioteket. Open CV (Open Source Computer Vision Library) är ett bibliotek av programmeringsmetoder och algoritmer fokuserade på realtidsbehandling, objektidentifiering, ansiktsidentifieringsalgoritmer och bildmanipulering.

Förmågan att kunna urskilja ett ansikte är ett viktigt första steg i automatisk ansiktsigenkänning. Detektering av ansikte och ögon uppnåddes genom användning av en detektor med inlärningsförmåga baserad på OpenCV biblioteket. Biblioteket baseras på en metod kallad "Haar feature-based cascade classifiers", som bygger på inlärning baserad på ett antal positiva och negativa bilder. Resultatet från inlärningen används sedan för att detektera objekt inom andra bilder.

Medan systemet används letar det efter ansikten och ögon i den avsökte bilden. Om ögonen är stängda eller om ansiktet är utanför räckvidd mer än 2,5 sekunder, skickar systemet ut en varning. Systemet kan dessutom lära sig att känna igen ansikten genom att analysera insamlad bilddata med EigenFaces-algoritmen. Tack vare detta kan säkerhetsvarningar visas, samtidigt som förare kan identifieras och få en anpassad användarupplevelse.

5.9 Eftermarknadsinstallation

Genom att använda Secure Gateway-konceptet är det möjligt att utveckla nya eftermarknadslösningar. I projektet demonstreras detta med en taxiskylt. Skylten monteras på fordonet och ansluter automatiskt till Secure Gateway. En taxameterapplikation laddas sedan ner och installeras i IHU. Med hjälp av meddelandeprotokollet (MQTT) kan applikationen hitta och använda det API som tillhandahålls av skylten. Detta enkla exempel demonstrerar ett effektivt sätt att expandera ett system med ny funktionalitet.

5.10 Bidrag till FFI-mål

FFI-programmets mål att öka kunskap och reducera utvecklingstid samt "time to market" adresseras enligt följande:

- Elektronisk arkitektur för inbyggda och uppkopplade system
 - PLINTA:s referenssystem utgör en öppen och flexibel plattform som kan användas till grund för vidare utveckling av infotainmentsystem. Plattformen och de agila arbetssätten inom projektet har influerat VCC:s strategi för infotainmentutveckling
 - De modulära och öppna miljöerna använda i såväl Secure Gateway som IHU-plattformen öppnar upp för extern utveckling, som i sin tur reducerar time to market
 - Anslutning till molnet och cloud computing
 - Ökad flexibilitet i arkitekturen för att addera nya funktioner
- Tekniker för "green, safe and connected functions"
 - Proof of concept att ansiktsigenkänning kan användas för att mäta och hantera förardistraktion
 - Ett koncept för hur applikationsutveckling med förardistraktion i åtanke kan förenklas
- Digitala användargränssnitt
 - Ett referensanvändargränssnitt som motverkar förardistraktion och underlättar standardisering av gränssnitt mellan anslutna enheter
 - Ett ramverk för att tillåta tredjepartsutveckling av moderna och enhetliga HMI-lösningar i enlighet med OEM- och säkerhetsstandarder
- Nationell tillväxt
 - Genom att använda öppna och flexibla plattformar, ökar PLINTA-projektet såväl fart som volym av innovationer. Implementeringskostnaderna minskar samtidigt som utvecklarna kan arbeta autonomt

6 Spridning och publicering

Resultaten har vid flertalet tillfällen demonstrerats inför utvalda deltagare, för att utlösa diskussioner om såväl lösningar som resultat. Deltagarna representerar de domäner som behöver ta till sig resultaten och rekommendationerna, för att etablera den drivkraft som krävs för att föra lösningarna närmare implementering. Det rör sig framförallt om systemarkitekter, HMI-designers, User Experience-konceptutvecklare, applikationsutvecklare samt säkerhetsspecialister inom Volvo Personvagnar.

PLINTA-projektet har proaktivt närmat sig kringliggande projekt och intressenter för att identifiera de användningsscenarier som potentiellt kan locka en stor publik och samtidigt demonstrera möjligheter med plattformen, såsom ”Instant Drive Off” och den tidigare omnämnda fjärrkontrollen.

Övriga exempel på PLINTA-exponering är genom Pelagicores arbete i GENIVI, exempelvis i EG HAF, men också genom den öppna källkoden av de utvecklade komponenterna, såsom IVI-logger. PLINTA-demonstratorn visades dessutom under GENIVI AMM, som arrangerades i Göteborg under maj 2014.

6.1 Publik presentation

Projektet har presenterats i följande forum:

- Vehicle ICT Arena, Lindholmen, Göteborg, den 4 september 2014
- Genivi all member meeting, Göteborg, 20-23 maj 2014
- Elektronik i fordon, Göteborg, 8-9 april 2014
- Vinnovas FFI-konferens, Lindholmen, Göteborg, den 6 november 2014
- Multipla demonstrationer på Volvo Cars för särskilt utvalda personer från olika avdelningar och domäner

Arbetet har även presenterats på ett privat evenemang:

- Volvo Cars Technology Exhibition, Göteborg, 2-4 december, 2014

6.2 Kunskaps- och resultatspridning

HMI-säkerhetsramverket som beskrivs i mjukvaruarkitekturrapporten, baseras på ett koncept utvecklat i ett annat Vinnovasponsrat projekt – SICS. Det kan vara värt att följa upp vidare utveckling av det säkerhets-API som är utvecklat i SICS-projektet, och samtidigt överväga att inkorporera denna funktionalitet i en PLINTA-kontext.

Resultaten relaterade till Secure Gateway kommer att återanvändas och vidareutvecklas inom det Vinnovasponsrade projektet ”Second Road fas 2”.

7 Slutsatser och fortsatt forskning

7.1 Slutsatser

Slutsatserna från PLINTA-projektet är att det bör vara möjligt att använda en öppen och flexibel miljö för utveckling av ny funktionalitet i en infotainmentkontext. En vanlig missuppfattning är att många av ovanstående funktioner redan har kommersialiserats inom fordonsbranschen. Detta är emellertid långt ifrån sanningen. Att addera funktionalitet eller nya användarupplevelser i dagens infotainmentsystem är mycket tidskrävande och kostsamt. De huvudsakliga barriärerna är:

- Nuvarande överenskommelser och kommersiella kontrakt inom branschen är inte anpassade efter öppna plattformar
- Ekosystemet ligger fortfarande efter funktionaliteten. Det finns för många alternativa plattformar och ingen identifierad drivande kraft med en tydlig affärsmodell. En konsolidering krävs för att kunna bygga en stabil och långvarig plattform helt oberoende av fordonslivscyklar.

För att överbrygga dessa barriärer är förhoppningen att PLINTA kan inspirera den svenska fordonsbranschen att hitta nya möjligheter för att realisera innovativa idéer. En viktig del av PLINTA-projektet har varit att demonstrera och framhäva möjligheterna med en öppen och flexibel plattform.

Nedan följer några områden för fortsatt forskning och arbete baserat på utmaningar som hittats under projektets gång.

- Stödja utveckling av nya affärsmodeller som tar tillvara på slutsatser från detta projekt.
- Mer arbete krävs för att passa in secure gateway i fordonsystemarkitekturen för att uppnå kraven för industrialisering.
- Det krävs mycket arbete inom Public Key Infrastructure (PKI)-domänen. Delar av säkerheten inom Secure Gateway-konceptet förlitar sig på certifikat och befintliga metoder för att distribuera och skydda certifikaten är otillräckliga
- Systemen för IHU-integration av ansiktsgenkänning behöver utforskas vidare. Aktiva säkerhetssystem kan med fördel dra nytta av ansiktsgenkänning för att optimera användningen
- En fullt implementerad skyddad miljö för applikationer i enlighet med resultaten i detta projekt, tillsammans med automatisk applikationsvalidering, tredjeparts applikationsutruddning etc.

8 Deltagande parter och kontaktpersoner



Anders Bengtsson
HiQ Göteborg



Jacob Juul
Volvo Cars Corporation



Johan Kristensson
Semcon Sweden AB



Alwin Bakkenes
Pelagicore AB