Pattern-of-Life Forensics in Hansken

Maatwerk+ leveren met digitale sporen

AFSTUDEERVERSLAG
Versie 2.2

Timo Meconi
Stagiair Digitale en Biometrische Sporen
Nederlands Forensisch Instituut

Forensisch Onderzoek
Hogeschool van Amsterdam

STATUS: NIET VERTROUWELIJK

28 juli 2021, Amsterdam

Nederlands Forensisch Instituut
Ministerie van Justitie en Veiligheid

Dr. ir. H. Henseler
Senior Advisor Digital Forensics

Hogeschool van Amsterdam

Dr. J.G. Koster
Docent Forensisch Onderzoek

Voorwoord

Beste lezer,

Voor u ligt de scriptie 'Pattern-of-Life Forensics in Hansken, Maatwerk+ in digitale sporen'. Het onderzoek richt zich op het gebruik van Hansken in combinatie met PoLF voor de Forensische Opsporing. Deze scriptie is geschreven in het kader van mijn afstuderen voor de opleiding Forensisch Onderzoek aan de Hogeschool van Amsterdam. Van februari 2021 tot en met juni 2021 ben ik bezig geweest met dit onderzoek.

Deze scriptie is geschreven in opdracht van het Nederlands Forensisch Instituut. Samen met mijn stagebegeleider, Hans Henseler, heb ik de onderzoeksvraag bedacht. Het onderzoek dat ik heb uitgevoerd, was veelzijdig en complex. Met veel plezier heb ik gewerkt aan de verschillende onderdelen, wat een echte zoektocht van bit naar bewijs was. Graag wil ik Anneke Koster bedanken voor haar doorzettingsvermogen om digitaal onderzoek op de kaart te zetten. En natuurlijk Hans Henseler voor de begeleiding en vrijheid om dit onderzoek uit te voeren. Ondanks dat de begeleiding vooral online verliep, heb dit zeker niet als negatief ervaren. Ik ben hem zeer dankbaar voor zijn inzet, flexibiliteit en snelle reactie op mijn vragen.

Vanuit het lectoraat Forensisch Onderzoek wil ik Christianne en Renushka bedanken voor hun enthousiasme en prettige samenwerking. Ook bedank ik Mattijs, Jan Peter, Abdul, Tom en Ginger voor hun tijd om met me te sparren. Thanks to Matthew Sorell for his insights on PoLF and access to his personal dataset. Natuurlijk Elise en Sven voor peer-review gedurende mijn afstudeerstage. Daarnaast wil ik mijn collega's bij het NFI, DBS bedanken voor leerzame tijd in jullie organisatie.

Tot slot wil ik mijn familie en vrienden bedanken voor hun steun in de afgelopen periode. Jullie geduld en feedback hebben me geholpen om mijn afstudeerperiode in coronatijd door te komen.

Timo Meconi te Amsterdam, 7 juni 2021

Figuur 1: Ets van Rembrandt van Rijn uit 1638, waarin hij Adam en Eva uitbeeldt net voor de zondeval. Rechtsboven is de duivel uitgebeeld als draak, voordat hij zou veranderen in een slang. Rechtsonder naast de boom is Hansken te zien. De vrouwtjesolifant naar wie het DFaaS-systeem van het NFI is vernoemd.

Samenvatting

Met het analyseplatform Hansken van het Nederlands Forensisch Instituut kunnen opsporingsdiensten steeds meer sporen uit digitale gegevensdragers halen. Om hiervan gebruik te maken zullen rechercheurs getraind moeten worden. Zo leren ze onder andere hoe ze met Hansken.py kunnen werken. Daarvoor zal eerst gekeken moeten worden op welke manier rechercheurs deze functionaliteit het makkelijkst kunnen gebruiken. Met name forensische rechercheurs zouden hier voordeel uit kunnen halen met het gebruik van Pattern-of-Life Forensics. Hierbij gaat het om handelingen die worden geregistreerd door een smartphone.

Hiervoor is de volgende onderzoeksvraag opgesteld: In hoeverre kan Pattern-of-Life Forensics in combinatie met Hansken bijdragen aan scenariogericht onderzoek? Forensisch onderzoekers kunnen resultaten uit Pattern-of-Life Forensics gebruiken om specifieke onderdelen van hun scenario te toetsen. Zo kan de smartphone van de verdachte alleen uit gestaan hebben ten tijde van het delict. Of het tijdstip van overlijden van het slachtoffer kan achterhaald worden aan de hand van een smartwatch.

Om de hoofdvraag te beantwoorden, is een virtuele machine gebouwd met Hansken All-In-One en Jupyterlab. Digitale sporen werden met APOLLO geanalyseerd. Deze zijn teruggeplaatst als kindsporen van de database. Uit de resultaten bleek dat verschillende databasen van iPhone gebruikt kunnen worden om (opzet)handelingen of een mogelijke gebruiker aan te tonen. De betrouwbaarheid van digitale sporen blijft echter lastig aan te tonen door *dual-tool verification*. Een alternatief is door meerdere digitale sporen te toetsen aan de hand van verklaringen van verdachte of getuigen. Rechercheurs moeten kritisch zijn op de resultaten van digitale sporen en niet geneigd zijn om snel conclusies te trekken. Bewustwording wat wel en niet mogelijk is, zal daarom meer gestimuleerd moeten worden. Onder andere de kennis dat Pattern-of-Life sporen maar tijdelijk beschikbaar zijn.

Inhoudsopgave

1 Inleiding

1.1 Aanleiding van het onderzoek

Op 15 maart 2021 zetten verschillende strafrechtadvocaten in het NRC de aanval in op de Encrochat-berichten van de politie. Ze twijfelden niet alleen over de rechtmatigheid van de wijze waarmee de politie de berichten bemachtigde. Ook de verwerking met de zoekmachine Hansken roept vragen op bij meester Inez Weski. Zij staat zowel Naoufel F. als Ridouan T. bij in hun strafproces rondom de PGP-berichten. Volgens haar is niet te controleren of Hansken betrouwbare resultaten levert.[1] Echter, Hansken is speciaal gebouwd om grote hoeveelheden data op een forensische wijze te analyseren.[2] In een wereld waar alles steeds digitaler wordt, is een dergelijk analysesysteem hard nodig.

De Nationale Politie (NP) is zich ervan bewust dat digitale sporen steeds meer een rol gaan spelen in opsporingsonderzoeken. In een interview in het politieblad Blauw vertelt Ruud Staijen (programmadirecteur Forensische Opsporing) over de toekomst voor de forensisch onderzoeker:

> "De sporenzoeker van de toekomst kan niet alleen een bepaald facet uitvoeren, maar is iemand die voortdurend met een brede blik kijkt waar hij sporen vandaan kan halen: hetzij biologisch, hetzij digitaal. ... Voeg je de fysieke en digitale sporen op een plaats delict zo snel mogelijk bij de bevindingen van de tactische recherche, dan pak je waarschijnlijk meer daders. ... Uit recent onderzoek bleek dat de FO bij zwaardere onderzoeken in 40 procent van de gevallen digitale sporendragers links liet liggen. Daar is nog veel verbetering mogelijk."[3]

In 2014 is het Nederlands Forensisch Instituut (NFI) begonnen te werken aan een analyseplatform om digitale sporen door rechercheurs te laten analyseren. Het oude systeem Xiraf liep tegen limitaties aan qua snelheid en gebruikersgemak. Uitgangspunt van Hansken was om rechercheurs in staat te stellen sneller digitaal bewijsmateriaal te laten analyseren.[4] Het bouwen van Hansken is niet over een nacht ijs gegaan. Van Beek et al. beschrijven hoe Hansken tot stand is gekomen. Denk daarbij niet alleen aan de opslag- en analysecapaciteit, maar ook aan de privacy- en beveiligingsrisico's.[5] In de jaren daarna ontwikkelde het NFI Hansken verder. Van Beek et al. maakten na vijf jaar de balans op over geleerde lessen tijdens de ontwikkeling hiervan. Een aanbeveling was om binnen de Digitaal Forensisch Onderzoek (DFO)-gemeenschap samen te werken. Het tijdelijke Opschaling en Ketenimplemtatie Hansken

(OK Hansken)-programma van het NFI heeft als doel om trainingen te geven en een community te bouwen rondom Hansken. Een andere aanbeveling was dat een Tool Development Kit (TDK) gebruikt zou kunnen worden om computer scripts automatisch taken te laten uitvoeren. Met andere woorden, een gedeelte van de analyse wordt door Hansken zelf gedaan.[2]

1.2 Probleemstelling

Vanuit OK Hansken worden speciale community dagen georganiseerd, waar ketenpartners met elkaar in contact komen. Tijdens de eerste twee events stonden Hansken.py en de Hansken Extraction Plugin (EP) centraal. Deze twee services stellen de gebruiker in staat om zelf computerprogramma's te schrijven voor Hansken. Het NFI is van plan om hiervoor meerdere trainingen te ontwikkelen. Echter, de huidige opleidingen zijn in de ogen van rechercheurs niet effectief. Daarom zal gekeken worden of rechercheurs op een andere manier opgeleid kunnen worden. 'Learning-on-the-job' is een van de manieren, waarbij de praktijk een belangrijke rol speelt tijdens de opleiding.[6] OK Hansken is bezig om een Hansken Academy op te zetten, waar rechercheurs leren omgaan met het programma.

Een opkomend onderzoeksgebied binnen DFO heet Pattern-of-Life Forensics (PoLF), die de koppeling met de praktijk kan vormen. Deze tak richt zich niet alleen op foto's en video's in de smartphone. PoLF onderzoekt databasen die informatie geven over het gebruik van de telefoon, zoals aan-/uitzetten, batterijniveaus en datagebruik. Zo zou een reconstructie gemaakt kunnen worden vanaf het moment dat een foto wordt genomen tot het uploaden naar een Dropbox-account.[7] Voor de forensisch onderzoekers zouden dit soort digitale sporen specifieke delen van een scenario kunnen verifiëren of falsificeren. Data uit Hansken kan hiervoor gebruikt worden. Op dit moment is deze koppeling van PoLF in combinatie met Hansken nog niet beschikbaar, maar dit zou in combinatie met Hansken.py wel mogelijk zijn.

1.3 Doelstelling

Voor het onderzoek zal eerst een testomgeving gemaakt worden om rechercheurs te trainen en digitale onderzoeken te laten uitvoeren. In plaats van het verzorgen van een training wordt er een testomgeving gecreëerd, waar rechercheurs kunnen experimenteren met digitale sporen. Deze omgeving maakt onder andere gebruik van Hansken in combinatie met de Jupyterlab. Rechercheurs zijn dan niet alleen in staat om analyse uit te voeren, maar ze kunnen ook zelf scripts schrijven voor hun eigen (lopende) onderzoeken. Indien mogelijk zouden de scripts onderling gedeeld kunnen worden met andere onderzoeksteams.

Het creëren van een digitaal platform is slechts een technische oplossing. Om van enig nut te zijn, zal de omgeving ook moeten aansluiten op de praktijk.[8] Digitale sporen geven

in tegenstelling tot de fysieke sporen vaak een volgorde van gebeurtenissen en precieze tijden aan.[9] Met name smartphones zijn ondertussen geïntegreerd in het dagelijks leven.[10] De smartphone is inmiddels een uiterst persoonlijk apparaat geworden. Niet alleen om via biometrische beveiliging de eigenaar te determineren, maar de data schetst gedragspatronen van de eigenaar.[11] PoLF-sporen zouden hierdoor van een toegevoegde waarde kunnen zijn voor de opsporingspraktijk.

Het doel is om te onderzoeken of PoLF gebruikt kan worden met Hansken als de basis. De testomgeving wordt hierbij gebruikt als programmeerplatform. In principe is dit een stapje verder dan alleen het maken van training voor rechercheurs. De toegevoegde waarde en validatie van PoLF zal eerst bepaald moeten worden, voordat deze kennis gedeeld kan worden met de ketenpartners. Vandaar dat ook wordt onderzocht welke soort sporen interessant kunnen zijn.

1.4 Hoofd- en deelvragen

Om het doel te behalen, is een hoofdvraag opgesteld voor het onderzoek. Het gebruik van Hansken, PoLF en de opsporingspraktijk staan hierbij centraal. Hieruit volgt de hoofdvraag:

In hoeverre kan Pattern-of-Life Forensics in combinatie met Hansken bijdragen aan scenariogericht onderzoek?

Om de hoofdvraag te beantwoorden zijn de deelvragen verdeeld in literatuuronderzoek en experimenteel onderzoek. Het literatuuronderzoek zal de volgende onderwerpen behandelen:

1. Wat is Hansken?
2. Wat is Pattern-of-Life Forensics (PoLF)?
3. Welke juridische bezwaren zijn er voor het gebruik van PoLF?
4. Hoe gaan de opsporingsdiensten om met digitaal bewijsmateriaal?

De andere deelvragen worden aan de hand van het experimentele onderzoek beantwoord:

5. Hoe zouden digitale sporen ingezet kunnen worden?
6. Welke potentie hebben digitale sporen in scenariogericht onderzoek?
7. Op welke manier kunnen digitaal en forensisch onderzoek elkaar versterken?

1.5 Afbakening van het onderzoek

Een groot gedeelte van het onderzoek is gekaderd door het gebruik van PoLF-sporen. Hierdoor zal niet inhoudelijk naar sporen worden gekeken, zoals analyse van berichten en e-mails. In plaats daarvan zal de focus liggen op de metadata. Op gebied van PoLF-analyse bij iPhones is het werk van Sarah Edwards leidend met haar tool: Apple Pattern of Life Lazy Output'er (APOLLO).[12] In deze tool worden verschillende databasen onderzocht. In haar blog mac4n6 geeft ze verdere uitleg hoe precies deze databasen in elkaar zitten.[13] Voor Android is minder bekend wat qua PoLF mogelijk is. Daarom is ook gekozen om alleen iPhones te gebruiken in het onderzoek.

Qua digitale middelen zal een Hansken All-In-One (Hansken AIO) installatie gebruikt worden. Deze versie bevat minder functionaliteiten dan productie, maar is wel geschikt om testen uit te voeren. Er wordt documentatie geschreven om dit zelf te installeren in een 'virtual machine'. De programmeertaal Python in combinatie met Jupyterlab is gekozen als softwareomgeving. Codes zullen niet in detail besproken, maar de lezer kan in de comments de werking van het programma lezen.

1.6 Leeswijzer

In hoofdstuk 2 wordt de theoretische achtergrond van Hansken en PoLF besproken. Kort zal ingegaan worden op de juridische aspecten en opsporingspraktijk. Hoofdstuk 3 geeft een overzicht van gebruikte onderzoeksmethoden en datasets. In hoofdstuk 4 worden de resultaten doorgenomen van Hansken AIO, Jupyterlab en het gebruik van PoLF in de opsporing. De conclusie met betrekking tot de hoofdvraag staat in hoofdstuk 5. In hoofdstuk 6 wordt afgesloten met aanbevelingen naar aanleiding van het onderzoek. Verwijzingen naar literatuur zijn aangegeven met nummers. De juridische referenties met Romeinse cijfers.

1.7 Vetrouwelijke stukken

Dit document is een openbare versie van het afstudeerverslag. Tijdens het onderzoek is contact gelegd met meerdere ketenpartners binnen de strafrechtketen (inter)nationaal. Hierin zijn zowel lopende zaken besproken. In verband met de vertrouwelijkheid van deze informatie is gekozen, om dit in deze versie niet te vermelden. Eventuele verzoeken om een volledige kopie te ontvangen, kan worden opgevraagd.

2 Theoretisch kader

De deelvragen 1 tot en met 4 zullen aan de hand van literatuuronderzoek worden beantwoord. Vanuit de hoofdvraag zullen eerst Hansken en PoLF worden besproken en gedefinieerd in 2.1 en 2.2. Daarna zal ingegaan worden op de juridische aspecten voor het gebruik van PoLF in 2.3. Als laatste wordt besproken hoe opsporingsdiensten omgaan met digitaal bewijsmateriaal in 2.4.

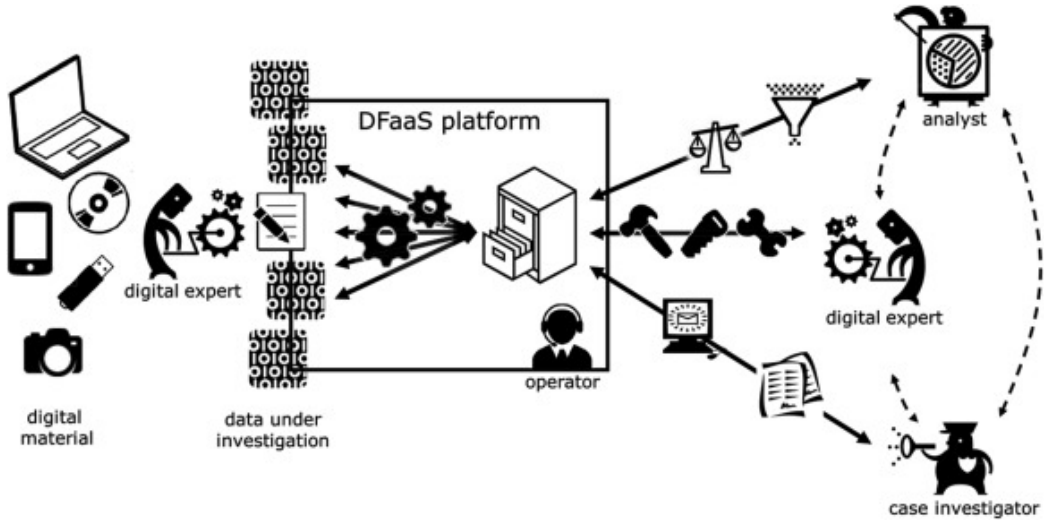
2.1 Hansken

Het NFI heeft niet stilgezeten, als het gaat om digitaal sporenonderzoek. Al in 2006 werd een poging gedaan om een platform te ontwikkelen dat grote hoeveelheden data en verschillende soorten computerbestanden kon analyseren. XML Information Retrieval Approach to digital Forensics (Xiraf) was het eerste prototype om dit te verwezenlijken. Hoewel Xiraf in staat was om een tijdlijn te maken, foto's te onderzoeken en kinderporno te detecteren, merkten Alink et al. op dat het niet de meest efficiënte tool is. Xiraf maakt niet gebruikt van cachegeheugen. Een ander punt is dat de opslag verloopt via Binary Large OBject (BLOB) en Extensible Markup Language (XML). Dit maakt dat Xiraf veel computerkracht nodig heeft. De gebruiker merkt dat het programma hierdoor langzaam is.[14]

In de jaren erna wordt Xiraf verder ontwikkeld met meer functionaliteiten. In het artikel van Bhoedjang et al. worden ook lessen getrokken om grote hoeveelheden data te verwerken. Door middel van het reduceren van *I/O-requests*, het gebruik van *parallel execution* en de selectie van specifieke algoritme kan de data sneller worden verwerkt. Het doel van dit programma is om niet alleen experts, maar ook tactische rechercheurs een gedeelte van het DFO uit te laten voeren. Een groot voordeel van Xiraf is verder de splitsing van een frontend en backend. Hierdoor is het mogelijk om de verwerking van de data gescheiden te houden van de analyse. De analyse kan daarna worden gedaan via een website, waar gebruikers op kunnen inloggen. Volgens Bhoedjang et al. kunnen programma's zoals Xiraf op een effectieve manier bijdragen aan strafrechtelijke onderzoeken.[15]

De ontwikkeling Xiraf loopt tegen zijn grens aan. Het NFI wil niet alleen een nieuw platform ontwikkelen, maar ook het proces rondom de verwerking van digitale sporen verbeteren. Digital Forensics as a Service (DFaaS) is de methodiek van het NFI, waarbij vooral de extractie fase wordt uitgevoerd door de computer zonder tussenkomst van een gebruiker. De resultaten worden hierna centraal opgeslagen op een server. De gebruiker kan via een webbrowser de

resultaten beoordelen. De methode heeft nadelen, zoals het centraal opslaan van alle data en het toegankelijk maken voor de gebruikers. Dat kan op zichzelf leiden tot beveiligings- en privacyrisico's. Hiermee dient rekening mee te worden gehouden tijdens de ontwikkeling. Vandaar dat het NFI is gaan werken aan een opvolger van Xiraf.[4]



Figuur 2.1: Digital Forensics as a Service (DFaaS)-model opgesteld door Van Baar et al.[4]

In 2012 is het NFI begonnen om de architectuur van de opvolger te bepalen: Hansken. In plaats van het meteen ontwikkelen van Hansken koos het NFI ervoor om eerst alle behoeftes en risico's van het systeem in kaart te brengen. Het doel van Hansken was om grote databestanden te kunnen verwerken, sneller resultaten te leveren aan het onderzoeksteam en zo veel mogelijk digitale artefacten te ondersteunen. In het artikel van Van Beek et al. worden acht ontwerpprincipes gegeven waar rekening mee dient te worden gehouden, waarvan beveiliging, privacy en transparantie de belangrijkste drie zijn. Hierna zijn twaalf onderwerpen gedefinieerd, die ervoor zorgen dat de ontwerpprincipes worden gewaarborgd (encryptie van data, logsystemen en datastromen). Van Beek laat in het artikel verder zien welke technische implementaties voor Hansken worden gebruikt.[5]

Vijf jaar later maken Van Beek et al. de balans op over het gebruik van Hansken. Daarbij geven ze ook een overzicht van de geleerde lessen. Hansken is gebruikt in meer dan 1000 zaken in Nederland, waaronder de Ennetcom-zaken. Ter uitbreiding van het artikel uit 2015 bespreken ze hoe Hansken in de praktijk wordt ontwikkeld. Zo wordt gewerkt met Scrum-teams, is een groot gedeelte van de verwerking van de testen geautomatiseerd en zijn beveiligingen ingebouwd in geval dat systeemfouten optreden. Uit de ontwikkeling van Hanksen zijn ook lessen getrokken wat betreft het gebruik van het DFaaS-systeem: agile-werken binnen een publieke sector is moeilijk, één systeem kan niet voldoen aan de behoefte van elke zaak en het systeem kan ook niet een DFO-expert vervangen. Als laatste bespreken ze de visie voor het gebruik van Hansken in de toekomst, waarvan zijn laatste punt pleit voor meer samenwerking.

De DFO-gemeenschap zal met elkaar moeten samenwerken, om bij te kunnen blijven dragen aan opsporingsonderzoeken.[2]

In de rechtbank is Hansken gebruikt voor het Ennetcom-onderzoek.[I] Meester Weski voerde aan dat de verdachte geen eerlijk proces kon hebben als gebruik gemaakt zou worden van een hulpmiddel, zoals Hansken.[II] Zo zouden er te weinig mogelijkheden zijn om de betrouwbaarheid van de data te testen. Ook is er geen sprake zijn van 'equality-of-arms'. De verdediging kan namelijk niet gebruik maken van Hansken in hun voorbereiding. Het is mogelijk om bij het NFI de data in te zien.[16] De rechtbank oordeelde in beide gevallen dat hiervan geen sprake was. Een rechter-commissaris heeft de betrouwbaarheid onder begeleiding van het NFI kunnen testen. Daarnaast had de verdediging de mogelijkheid om een expert in te schakelen. Hij oordeelde dat Hansken geen mankementen bevatten.[16]

Hansken heeft zich in de afgelopen jaren kunnen bewijzen als hulpmiddel voor de opsporingsdiensten als het gaat om beeldmateriaal en berichten. In een snel veranderende wereld is het moeilijk voor opsporingsdiensten om bij te blijven. Daarom moeten de krachten gebundeld worden, waarbij Hansken een rol kan invullen om kennis te delen en het digitale werk te automatiseren.[17] Kortom, Hansken zal zich moet blijven ontwikkelen. Om dit te bereiken, creëerde het NFI het programma Opschaling en Ketenimplemtatie Hansken (OK Hansken). Het doel van OK Hansken is om een community te bouwen, om te zorgen dat Hansken volwassener wordt. Zo wordt de samenwerking aangegaan met onderwijsinstellingen, zoals Norwegian University of Science and Technology (NTNU) en HSL. Een ander deelproject bestaat uit het inrichten van een inzagefunctie voor de advocatuur, om de verdediging ook toegang te geven tot Hansken.[2]

Resumé

Uit literatuuronderzoek blijkt dat Hansken de opvolger is van Xiraf om digitale sporen te kunnen analyseren. Een van de doelen van Hansken is om informatie zo snel mogelijk bij de opsporingsteams te brengen. Hiervoor is de DFaaS-methodiek ontwikkeld, waarbij alle onderdelen van Hansken de data op een forensische wijze indexeren en analyseren. Inmiddels is Hansken gebruikt in diverse strafzaken om het berichtenverkeer van PGP-telefoons inzichtelijk te maken. Om echter bij te blijven met de huidige trends zullen kennisinstellingen en opsporingsdiensten met elkaar moeten samenwerken. OK Hansken is een programma da als doel heeft om de Hansken-software volwassener te maken en de gebruikers beter te ondersteunen in het gebruik van Hansken. Onder andere door het opzetten van de Hansken Academy voor trainingen en de Hansken Community waarin gebruikers elkaar kunnen ontmoeten en kennis kunnen uitwisselen.

2.2 Pattern-of-Life Forensics (PoLF)

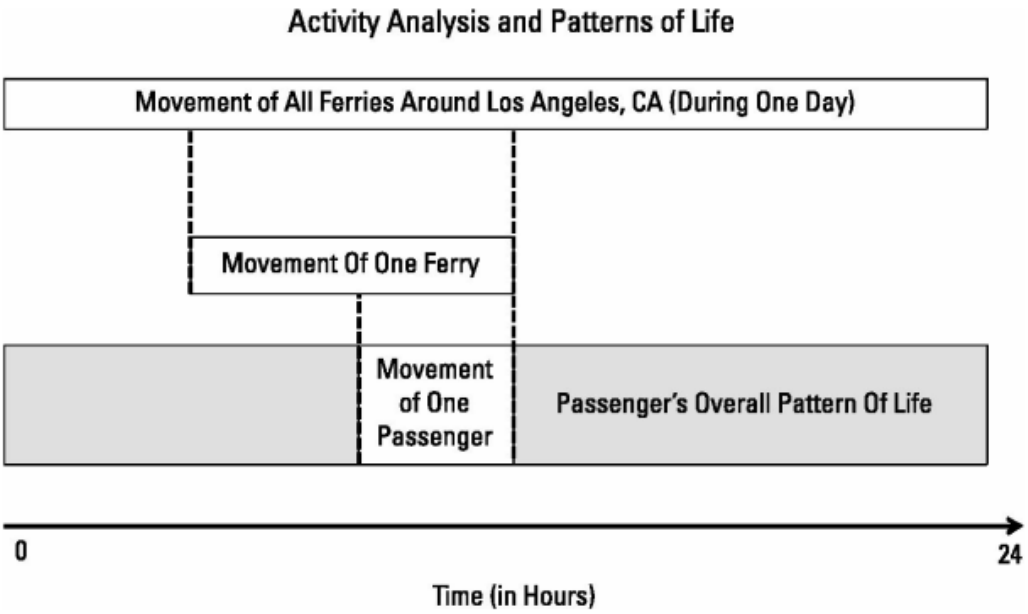
Het NFI maakt gebruik van hiërarchie van hypothesen op bron-, activiteit- en delictniveau, waarbij aan elk niveau andere eisen worden gesteld.[18] In de rechtszaal is de discussie bij bijvoorbeeld vingersporenonderzoek niet (alleen) meer of vingerspoor A van de verdachte komt (bronniveau), maar op welke manier dit gezet is (activiteitniveau). De plaatsing, de locatie van het vingerspoor en welke vinger kunnen gezamenlijk inzicht geven welke activiteit er mogelijk vooraf is gegaan.[19,20] Maar digitale sporen bieden, in tegenstelling tot de fysieke sporen, vaak precieze tijden met daaraan gekoppeld activiteiten. Hiermee zou niet alleen de 'wie'-vraag, maar ook de 'wat, waar, wanneer'-vragen beantwoord kunnen worden.[9]

Onderzoek naar dit soort sporen valt onder Pattern-of-Life (PoL). In de fysieke wereld kan het gaan om observaties van individuen of locaties. De methode wordt veelvuldig gebruikt in oorlogsgebieden, om informatie te verzamelen over vijandige netwerken.[21] Echter, DFO heeft geen officiële definitie voor PoL. Daarom wordt de volgende definitie van Biltgen dn Ryan gebruikt:

Definitie 2.1. Pattern-of-Life (PoL) is the specific set of behaviors and movements associated with a particular entity over a given period of time.[22]

Dit komt overeen met de dagelijkse gebruiken/gewoontes van een individu. Het kan van alles zijn, zoals de krant lezen, de hond uitlaten of om zes uur eten. In de definitie wordt echter gesproken van een 'particular entity'. Biltgen beschrijft dat er meerdere niveaus zijn van PoLF. Neem het vliegverkeer op luchthaven Schiphol. Op het hoogste niveau kan gekeken worden naar de bewegingen van alle vliegtuigen rondom de luchthaven. Daarnaast kan gefocust worden op de beweging van één vliegtuig in het bijzonder. Op het laagste niveau kan ingezoomd worden op de bewegingen van een passagier.[22]

Voor opsporingsdoeleinden kunnen dezelfde soort principes gebruikt worden in de vorm van Pattern-of-Life Forensics (PoLF). Met name het gebruik van een smartphone is zeer interessant. Een smartphone registreert namelijk veel meer dan de mensen denken. Sinds 2018 bevat zowel iOS en Android respectievelijk de functies Screen Time en Digital Wellbeing, wat onder meer bijhoudt hoe de smartphone gebruikt wordt.[9] Sarah Edwards heeft veel onderzoek gedaan naar de KnowledgeC-database, waar dit soort artefacten zijn opgeslagen bij iPhones. Zij houdt een blog bij genaamd mac4n6. In haar blog schrijft ze over waar deze artefacten gevonden kunnen worden.[13] Daarvoor heeft ze Apple Pattern of Life Lazy Output'er (APOLLO) ontwikkeld voor iPhone en Android Review Timeline Events Modular Integrated Solution (ARTEMIS) voor Android.[12] Als voorbeeld geeft ze welke informatie gelogd wordt als ze hardloopt. Niet alleen is terug te vinden welke route ze heeft gelopen, maar ook wanneer ze haar muziek heeft



Figuur 2.2: Voorbeeld van meerdere niveaus van PoLF met verschillende entiteiten uit het boek (p. 100) van Biltgen. Dit betreft alle bewegingen rondom de veerboten in Los Angeles.[22]

opgestart.[23]

Bij de Digital Forensic Research Workshop (DFRWS) in 2020 verzorgde Brignoni een presentatie over het gebruik van digitale sporen in opsporingsonderzoeken.[24] Hij ontwikkelde de tool iLEAPP voor iOS en ALEAPP voor Android. Deze tool is vergelijkbaar met die van APOLLO met als extra voordeel dat deze een Graphical User Interface (GUI) heeft en als triagemiddel gebruikt kan worden. In plaats van een export van database kunnen de resultaten vanuit xLEAPP in worden gezien via een webbrowser. Voor mensen die geen affiniteit hebben met computers, verzorgd Brignoni een speciale Python-webserie. Dit stelt rechercheurs in staat om hun eigen scripts te schrijven voor xLEAPP in Python.[25]

De resultaten uit APOLLO en xLEAPP komen voornamelijk uit .sqlite-databases. Deze databasen zijn compact, snel (voor kleine datasets) en met SQL-queries uit te lezen. Ze worden vaak gebruikt in smartphones om data lokaal op te slaan.[26] Voor Apple geldt dat de meeste logbestanden worden weggeschreven in .sqlite-databases (of .plist-en).[27] Het voordeel van .sqlite is dat de data met behulp van SQL-statements opgevraagd kan worden. Van Zandwijk en Boztas hebben onderzoek gedaan naar *healthdb_secure.sqlite*. Deze database registreert onder andere het aantal stappen, de afgelegde afstand en het aantal verdiepingen. De data staat wel verspreid over meerdere tabellen (*tables*): *samples* en *quantity_samples*. Door de tabellen te combineren kunnen de stappen op een bepaalde tijd worden herleid (zie Code C.1). De tijd is gedefinieerd als *Mac absolute time/Apple Cocoa Core Data timestamp* (aantal seconden sinds 2001).[28] De mogelijke afgelegde looproute kan aan de hand van deze database worden gereconstrueerd. Uit het onderzoek blijkt verder dat stappen accurater zijn

dan de afgelegde afstand. Hiermee dient dus ook rekening gehouden te worden voor de interpretatie van deze data.[29]

De *healthdb_secure.sqlite* zou gebruikt kunnen worden om de gezamenlijke looproutes van het slachtoffer en de verdachte na te gaan. Als beide personen een Apple Watch of iPhone hebben, dan is het mogelijk om de twee datasets met elkaar te vergelijken. Dit kan een rol spelen als het scenario ervan uit gaat dat beiden een stukje samen hebben gelopen of als sprake is van stalking. Jennings vergeleek de stappentellers van Apple Watch en iPhone van twee personen. Een verglijkbaar onderzoek hebben Bosma et al. uitgevoerd met zendmastgegevens.[30] Hierbij keek hij alleen naar de data van drie dagen, waarbij de twee personen bij elkaar waren. Met behulp van de Kullback-Leibler Divergence (zie eq. 3.3) achterhaalde hij of het aannemelijk is of de twee iPhones dicht bij elkaar waren. Uit het onderzoek blijkt dat een duidelijk verschil is te zien tussen de dagen dat de twee personen bij elkaar waren in vergelijking tot de andere dagen. Jennings merkt op dat het mogelijk is dat de twee personen op hetzelfde moment dezelfde afstand hebben gelopen op andere locaties. Zendmastgegevens zouden hierover uitsluitsel kunnen geven.[31]

Resumé

Uit literatuuronderzoek blijkt dat PoLF wordt gebruikt door militaire organisaties om informatie te vergaren over vijandelijke doelen. Biltgen en Ryan beschrijven dat een specifieke set van gedragingen of bewegingen over een bepaalde tijd iets kunnen zeggen over een entiteit.[22] Sinds 2018 is het mogelijk om zulke gedragingen of bewegingen te monitoren met behulp van Screen Time en Digital Wellbeing op een smartphone. De functies houden onder andere bij wanneer de gebruiker de teleoon aan/uit zet, wat het batterijgebruik is en welke route is gelopen. Sarah Edwards heeft een open-source-programma APOLLO geschreven, die dit soort sporen uit iPhones haalt. Specifiek kan dan gedacht worden stappenteller op een iPhone. Van Zandwijk en Boztas laten zien dat stappenteller gebruikt kan worden om uitspraak te doen over looproutes. Jennings voegt toe dat stappenteller-datasets onderling met elkaar vergeleken kan worden, om een indicatie te krijgen of twee iPhones mogelijk bij elkaar in de buurt waren.

2.3 Juridische bezwaren

Biltgen en Ryan schrijven dat hoe meer informatie verzameld wordt, hoe beter men een beeld van de entiteit kan vormen.[22] Dit levert problemen op als het gaat om de privacy van de burgers. Binnen deze sectie zal vooral de Nederlandse situatie worden besproken. In de grondwet is namelijk vastgelegd dat de overheid niet zomaar de persoonlijke levenssfeer van haar burgers mag aantasten.[III] Hieronder vallen ook geheime observaties, het noteren van persoonlijke gegevens en fotograferen.[32] Dit is verder uitgebreid in het art. 8 EVRM: recht op eerbiediging

van privé-, familie- en gezinsleven.[IV]

Tot 2015 kende de rechtbank geen jurisprudentie voor het onderzoeken van smartphones en computers. Bij het Gerechtshof Arnhem-Leeuwarden kwam een zaak voor over openlijke geweldpleging, waarbij WhatsApp-berichten werden gebruikt in de bewijsvoering. Het hof oordeelde dat dergelijk bewijs een ernstige inbreuk was op de persoonlijke levenssfeer van de verdachte, zoals besproken in art. 8 EVRM.[V] De Hoge Raad deed uitspraak in het smartphone-arrest.[VI]. Lassche heeft hieruit de volgende conclusies getrokken:[33]

- Onderzoek aan gegevensdragers mag slechts worden uitgevoerd bij voldoende grondslag voor een niet meer dan geringe inbreuk op de persoonlijke levenssfeer.
- Voor een meer dan geringe inbreuk moet derhalve een andere grondslag worden gezocht.
- Dit betreft het onderzoek van gegevensdragers in zijn algemeenheid, smartphones daaronder inbegrepen.

Simpel gezegd: rechercheurs mogen gericht in een specifieke chat zoeken. Voor andere soorten onderzoeken zal toestemming gevraagd moeten worden aan de officier van justitie (OvJ) of rechter-commissaris (RC).[33] Anders is alle (verschoningsgerechtigde) data beschikbaar voor de rechercheurs. Vertrouwelijke informatie tussen bijvoorbeeld een verdachte en advocaat zou niet toegankelijk moeten zijn voor het OM. Als niet duidelijk is welke informatie wordt onderzocht, kan dit grote consequenties hebben in de vervolging van een verdachte. In Groot-Brittannië schreef de toezichthouder dat in meer dan de helft van de zaken tekortkomingen waren geconstateerd bij de informatieverstrekking van het OM en de politie.[34] In artikel van Seyyar en Gerardts wordt uitgelegd hoe binnen Hansken omgegaan kan worden als er een vermoeden is dat het gaat om vertrouwelijke informatie.[35]

Dat werpt de vraag op hoeveel data gebruikt mag worden voor het onderzoek. Het is niet wenselijk dat er te weinig informatie wordt verzameld om antwoord te geven op een onderzoeksvraag. Door juist meer informatie te gebruiken, kan een beter beeld worden geschetst voor een eventuele reconstructie. Dit kan zowel belastend als ontlastend zijn voor de verdachte.[35] Maar weegt dit aan de andere kant op tegen het feit dat een grote inbreuk op de privacy van het individu wordt gepleegd? Als blijkt dat verdachte onschuldig is, dan kan deze inbreuk niet zomaar teruggedraaid worden. Een juiste afweging moet dus gemaakt worden in welke mate de privacy geschonden mag worden. In Nederland kent de proportionaliteits- en subsidiariteitstoets, die waarborgen dat er zo min mogelijk inbreuk is op art. 8 EVRM.[33] Het is, echter, aan de staande en zittende magistratuur om dit te toetsen.

Een voorbeeld van het gebruik van PoLF is de Bûterwei-zaak, 2017.[VII] Hierin werd verdachte ten laste gelegd dat ze haar man heeft vermoord nabij een festivalterrein. De oriëntatie van de

telefoon van het slachtoffer is gebruikt om te onderbouwen dat hij tussen 00:30:00 en 00:46:34 om het leven is gebracht. Aan de hand van de telefoongegevens van verdachte is ook een reconstructie gemaakt van haar activiteiten rond die nacht. Tot op de seconde nauwkeurig konden haar activiteiten achterhaald worden. Echter, tussen de hiervoor genoemde tijden werd geen data geregistreerd. Verdachte gaf op dat haar telefoon plotseling was uitgevallen. Iets wat de deskundigen niet waarschijnlijk achtten gezien de logbestanden van december 2017. Deze loggegevens zijn niet rond de tijd van het delict veiliggesteld. Verdachte was eerst aangemerkt als getuige (echtgenoot van het slachtoffer). De politie heeft daarom later pas een volledige forensische kopie van de telefoon van de verdachte kunnen maken, om de onder andere de batterijgegevens de onderzoeken. Het gerechtshof ging niet mee in de redenering van de verdachte en veroordeelde haar tot een gevangenisstraf van 20 jaar.

Uiteindelijk zal altijd een discussie blijven hoeveel digitale data gebruikt mag worden in een strafzaak. Op dit moment is er te weinig jurisprudentie om een degelijke uitspraak te doen wat er wel en niet onder valt. In de hiervoor genoemde zaken is vooral bewijs gebruikt uit berichten, foto's en locaties. In de Bûterwei-zaak is gebruik van digitale sporen verschoven van bronniveau naar activiteitniveau, maar er is nog steeds geen juridisch kader waarin getoetst kan worden wat wel en niet mag wat betreft DFO. Voor nu is het dus de grenzen van de wet opzoeken, wat voorgelegd kan worden aan de rechtbank.

Resumé

Uit literatuuronderzoek en jurisprudentie blijkt dat het gebruik van PoLF-analyses een grote inbreuk kunnen zijn op de persoonlijke levenssfeer van de burger. Bij onderzoek aan een smartphone gelden strenge regels welk soort informatie doorzocht mag worden. Zo mogen rechercheurs alleen zoeken naar een specifieke chat. Om verdiepend onderzoek uit te voeren, zal toestemming gevraagd moeten worden aan de OvJ of zelfs de RC. Dat levert beperkingen op als het gaat om het maken van een waarheidsgetrouwe reconstructie, als een gedeelte van de data onderzocht mag worden. Het gebruik van meer digitale informatie kan zowel belastend als ontlastend zijn voor een verdachte. Een goede afweging zal dus gemaakt moeten worden door de staande en zittende magistratuur om dit per geval te toetsen. De Bûterwei-zaak is een voorbeeld waarbij PoLF-sporen wel hebben geleidt tot een reconstructie. Volgens de rechtbank zette de verdachte haar telefoon zelf uit ten tijde van het delict. Met de Bûterwei-zaak is een eerste stap gezet hoe de rechtbank digitale sporen gebruikt in onderbouwing van het vonnis.

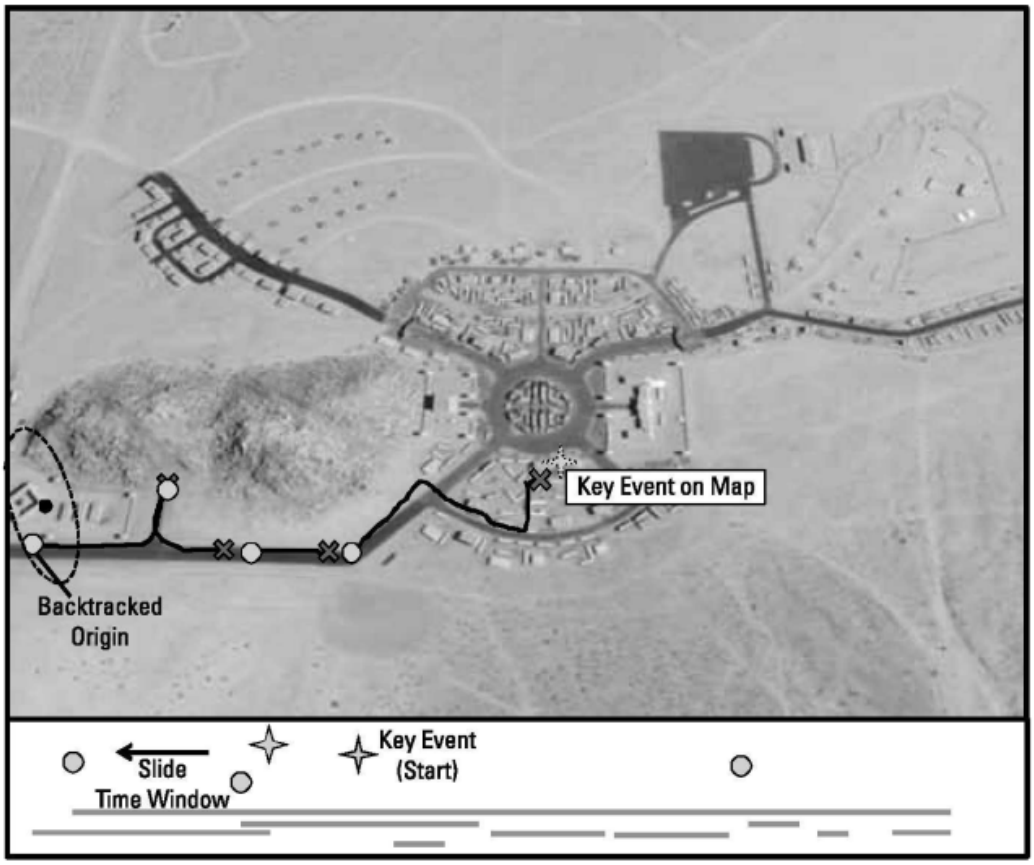
2.4 Gebruik van digitale sporen

Hansken staat bekend om haar snelle verwerkingskracht en big data analyses.[2,4,5] Dat betekent niet dat met één druk op de knop het antwoord uit Hansken komt rollen als een rechercheur het programma opstart. Uit twee rapporten van Cybersafety Research Group blijkt dat de kennis van de NP tekortschiet betreffende het gebruik van DFO-analysesoftware. Rechercheurs zien vaak wel de potentie ervan in, maar weten niet hoe ze dit kunnen gebruiken. Vaak moet een rechercheur een bepaalde expertise hebben om dit te kunnen inzetten. Rechercheurs kunnen hiervoor speciale cursussen volgen, maar de kennis blijft niet hangen.[6,8]

In de vorige paragraaf werd gefocust op: "Hoe kunnen digitale sporen helpen binnen onderzoeken?" Dit gaat vooral om de potentie van het digitale bewijs zelf. Digitale sporen worden dan leidend in het onderzoek. Dit kan leiden tot tunnelvisie.[36] Daar komt bij dat rechercheurs digitale sporen betrouwbaarder en objectiever achtten dan analoge sporen. Het probleem is dat digitale sporen wel degelijk te manipuleren zijn. Dit is overigens wel te achterhalen met verdiepend onderzoek.[6] Casey geeft aan dat de politie soms te weinig kennis heeft om digitaal onderzoek te verrichten, maar ook een kritische blik mist voor limitaties van de methode en het oplossen van gebruikersproblemen. Dit zou ertoe kunnen leiden dat een verkeerde interpretatie van het digitale sporenbeeld volgt.[37]

Casey et al. geven aan dat door Bayesiaanse statistiek te gebruiken verkeerde interpretaties voorkomen kunnen worden. In het artikel worden geolocaties gebruikt als voorbeeld. Het is namelijk afhankelijk van de technologie in combinatie met het aangedragen scenario van de aanklager en verdediging. Daarbij wordt niet alleen gefocust op de locatiedata, maar ook eventuele andere digitale sporen die het scenario kunnen verifiëren of falsificeren.[38] Waar Casey zich vooral richt op de digitaal deskundige, benadrukt Epskamp dat het gebruik van hypothesen breder ingezet kan worden met scenariodenken. Binnen het scenariodenken zou centraal moeten staan: "Wat is er gebeurd?" Hieruit worden (alternatieve) scenario's opgesteld voor een reconstructie. Hieruit kunnen indicatoren geformuleerd worden, die gebruikt kunnen worden om de scenario's te toetsen.[39]

Tot op zekere hoogte kunnen de problemen voor digitaal bewijs nog verder worden opgelost, zoals het doorzoeken van grote hoeveelheden data.[40] Scenarioreconstructies helpen namelijk niet alleen om scenario indicatoren te identificeren.[39] Scenarioreconstructies kunnen de hoeveelheid digitale informatie reduceren, die rechercheurs moeten onderzoeken. De aanvankelijk grote dataset wordt verkleind tot behapbare stukken via filters ('keywords-logic' en 'timestamp intervals'). In het geval van PoLF wordt ingezoomd rondom een specifieke tijd of een locatie, waarbij meerdere informatiebronnen aan elkaar gekoppeld kunnen worden. Biltgen en Ryan noemen een dergelijke reconstructie *forensic backtracking* (zie fig. 2.3).[22]

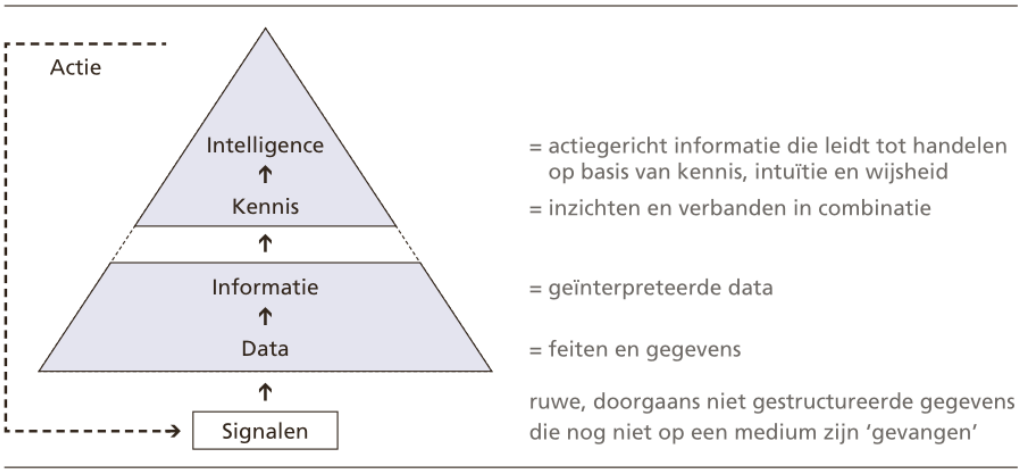


Figuur 2.3: Voorbeeld van het gebruik van 'forensic backtracking' uit het boek (p. 303) van Biltgen. Hierbij worden verschillende informatiebronnen met elkaar gecombineerd en teruggeredeneerd vanaf het 'key event'.[22]

Het gebruik van *forensic backtracking* zou in combinatie met de verschillende niveaus van PoLF kunnen leiden tot forensische intelligentie. Hierbij wordt gekeken naar correlaties tussen zaken, zoals alle gebeurtenissen die voor het delict hebben afgespeeld. Denk aan het vinden van bepaalde sporen, gedrag, context of locaties.[41,42] De NP noemt dit intelligencegestuurd politiewerk (IGP) (zie fig. 2.4). Uiteindelijk is het de koppeling van de tactiek, forensisch en digitaal met elkaar die kan zorgen voor intelligence. Immers de PoLF-artefacten kunnen zowel tactisch als forensisch extra ondersteunen, om een betere scenarioreconstructie te kunnen opmaken. Het identificeren van bepaalde gedragingen zorgt voor een beter inzicht hoe opgetreden kan worden op een PD.[43] Digitaal bewijsmateriaal (en PoLF in het bijzonder) kan zeker een waardevolle bijdrage leveren aan de opsporing.

Resumé

Uit literatuuronderzoek blijkt dat kennis op het gebied van DFO bij de NP tekort schiet, wat betreft het gebruik van DFO-analysesoftware . Rechercheurs achtten digitale sporen betrouwbaarder en objectiever dan analoge sporen. Dit zou kunnen leiden tot een verkeerde interpretatie van de data. Het gebruik van Bayesiaanse statistiek en scenariogericht onderzoek zouden kunnen



Figuur 2.4: Het cyclische proces waarbij vanuit signalen wordt opgetreden om informatie te verzamelen. Door alle informatie met elkaar te verbinden zou intelligence verzameld kunnen worden. Deze kennis kan daarna weer gebruikt worden op een andere PD, wat weer hetzelfde proces triggert.[43]

helpen om niet blindelings te vertrouwen op digitale sporen. De volgende stap is het combineren van verschillende informatiebronnen om een reconstructie te maken (*forensic backtracking*). Inzichten door het vormen van deze reconstructies kunnen leiden tot 'forensic intelligence', waar digitale sporen zowel tactisch als forensisch een bijdrage kunnen leveren.

3 Methodologie

Om antwoord te kunnen geven op de experimentele deelvragen zal eerst een testomgeving worden voorbereid. In deze omgeving is het mogelijk om computercodes te schrijven voor Hansken.py. Het opzetten hiervan wordt besproken in 3.1. Om analyses te kunnen uitvoeren is gebruik gemaakt van meerdere softwarepakketten (*libraries*). Welke *libraries* dit zijn en de functie van elke *library* worden besproken in 3.2. De communicatie met de Hansken Py API en de voorbereidingshandelingen van een Hansken project worden besproken in 3.3. Daarna wordt kort ingegaan op welke datasets gebruikt zijn voor het onderzoek in 3.4. Ter afsluiting wordt het gebruik van de Jupyterlab-omgeving en de statische berekeningen doorgenomen in 3.5.

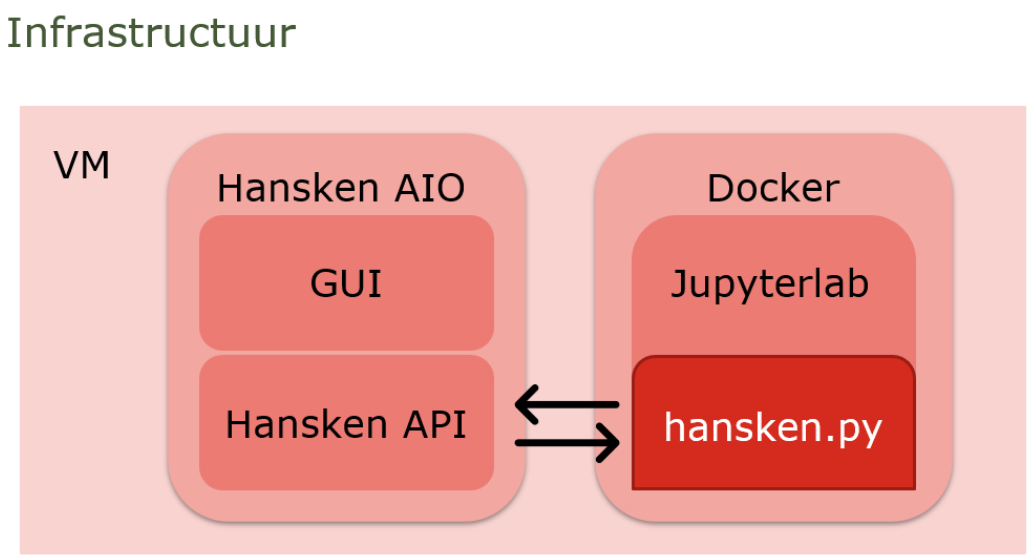
3.1 Testomgeving opzetten

Voor het onderzoek is een standalone server gebruikt met VirtualBox. Deze server is een virtuele computer, die dient als testomgeving. Deze omgeving is makkelijk in te stellen en op te zetten. De server heeft een Ubuntu Server (*20.04 LTS*) OS. Ubuntu is de basis waar de andere programma's op draaien. Hansken heeft alleen een Java-interpreter nodig om te kunnen werken. Voor deze installatie is gekozen voor de *openjdk-11-jre-headless*. Om digitaal bewijsmateriaal te uploaden naar Hansken werd de imaging-tool van het NFI gebruikt (*image-tool-40.19.0-community.jar*). Dit is een tool die de gebruiker in staat stelt, om een forensische kopie (*image*) te formatteren naar het NFI-formaat. De laatste stap bestaat uit het configureren van de services, zoals het instellen van de netwerk-routing. Dit valt buiten dit onderzoek, maar de handleiding Hansken AIO met deze stappen kan opgevraagd worden.

Verder bevat de testomgeving een Docker-service. Met Docker kunnen softwareontwikkelaars applicaties ontwikkelen, die OS onafhankelijk zijn. In theorie zou elke applicatie kunnen draaien als een Docker-service aanwezig is. Ontwikkelaars hoeven zich hierdoor geen zorgen te maken dat hun applicatie wel op de ene computer werkt, maar niet op de ander. Verder kan gekozen worden om één of meerdere services te bouwen. Het begint met een Dockerfile als basis voor elke service. De Dockerfile voor de testomgeving is te vinden in code C.2. Hierin staat onder andere welke software gebruikt wordt binnen Ubuntu. Ook zijn er specifieke stappen voor de Jupyterlab-installatie, die tevens zijn vermeld in de handleiding Hansken AIO.

Om te kunnen programmeren, is gekozen voor Jupyterlab. Dit is een interactieve programmeeromgeving in de vorm van een webapplicatie. De service is zo opgezet dat mensen Jupyterlab kunnen bereiken met hun webbrowser. Het voordeel is dat Jupyterlab gebruikt maakt

van een interactieve Python-omgeving. Stukjes computercode kunnen meteen worden uitgevoerd. Als fouten in het programma optreden, dan wordt dit direct gemeld. Het installeren van de *libraries* kan worden geautomatiseerd in Docker. In de Dockerfile worden deze geïnstalleerd via Pip (*pip install -r requirements*). In de 3.2 wordt verder uitgewerkt welke *libraries* gebruikt worden binnen het onderzoek.



Figuur 3.1: De verschillende onderdelen van de virtuele machine zijn weergegeven. Deze bestaat voornamelijk uit twee delen: Hansken AIO en Jupyterlab. De rechercheurs zullen vooral in de GUI onderzoek doen. Digitale rechercheurs kunnen met de Hansken Py communiceren met de Hansken API in Jupyterlab.

3.2 Analyse en visualisatie tools

Op GitHub zijn twee PoLF-softwarepakketten beschikbaar: APOLLO[12] (Sarah Edwards, BlackBag Technologies) en iLEAPP[25] (Alexis Brignoni, FBI) voor onderzoeken aan iOS-apparaten. Beide pakketten zijn geschreven in Python, waarbij iLEAPP nog een visuele weergave geeft van de data via een webbrowser. Daarnaast is het ook mogelijk om zelf scripts te schrijven voor databasen die niet worden ondersteund. In theorie zouden databasebestanden kunnen worden geanalyseerd met deze software. Daar onder vallen ook geëncrypte databasen, mits de sleutel beschikbaar is. Vaak is de sleutel ergens anders in de database weggeschreven. Voor dit onderzoek wordt alleen APOLLO gebruikt, omdat dit pakket meer PoLF-artefacten ondersteunt dan iLEAPP.

Binnen Jupyterlab is het mogelijk om verschillende *libraries* toe te voegen. In code C.3 staat een overzicht van de geïnstalleerde *libraries*. Voor data-analyse is Pandas gebruikt voor het inladen en verwerken. Binnen de Pandas kunnen verschillende soorten transformaties en analyses worden uitgevoerd. Deze acties zijn vergelijkbaar met Excel-sheets. Alleen bij Pandas moeten deze operaties worden geprogrammeerd. Voor de visualisatie van de data is gekozen voor

Matplotlib en Seaborn. Met Matplotlib kunnen eenvoudige grafieken, zoals lijn-, staafgrafieken en histogrammen. Hiervoor moet de data vanuit Pandas worden ingelezen in Matplotlib. Ter uitbreiding is Seaborn ook toegevoegd als library. Seaborn is gebouwd op Matplotlib en heeft meer functies, die van pas komen bij maken van grafieken. Het inlezen en operaties uitvoeren op de data gaat makkelijker in Seaborn dan in Matplotlib voor het maken van heatmaps.

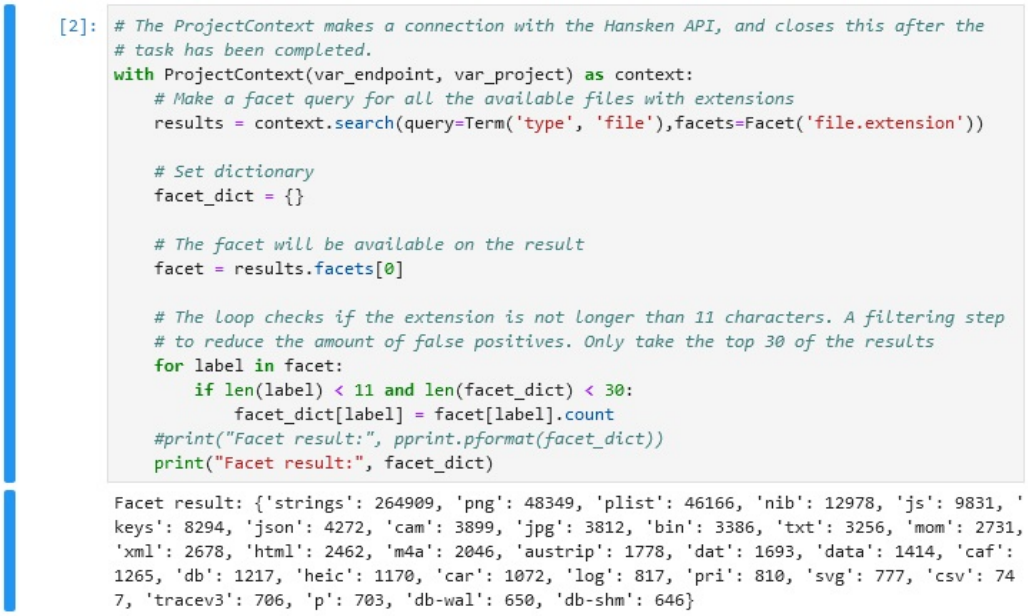
3.3 Hansken Python API

Om te kunnen communiceren met Hansken is door het NFI een Hansken Python API ontwikkeld. Dit is een interface, waar verzoeken naar toe gestuurd kunnen worden. Zo haalt de tactische en technische website (frontend) de data op vanuit Hansken. Hanksen.py heeft verschillende functies die gebruikt kunnen worden om data te versturen en te ontvangen. Rechercheurs kunnen gebruik maken van dezelfde functies. Dit vergt enige programmeerkennis in Python. Welke functies gebruikt kunnen worden, staat gedocumenteerd in Hansken Python API. Voor dit onderzoek zijn eigen scripts geschreven om beter te kunnen aansluiten op Jupyterlab. Om te kunnen communiceren met de Hansken Python API voert de gebruiker eerst het IP-adres en project-id in. Deze gegevens zullen verschillen per Hansken installatie. Advies is om naar de technische omgeving te gaan. Klik daar op het gewenste project. Rechts naast de zoekfunctie staat `'Sla zoekopdracht op'` en klik hierop. Een pop-up scherm komt naar voren met de `endpoint` en `project`. Deze gegevens worden ingevuld om te kunnen communiceren met Hansken (zie fig. 3.2 en code C.6). De volgende stap is het aanroepen van een functie. Om efficiënt gebruik te maken van Hansken wordt de facet-functie aangeroepen. Deze functie groepeert de resultaten aan de hand van de zoekopdracht. Aan Hansken wordt gevraagd welke bestandstype (.docx, .plist, enz.) aanwezig zijn binnen het project. Als resultaat geeft Hansken een *dict* terug, die het aantal per bestandstype weergeeft (zie fig. 3.3 en code C.4).

Als laatste is de data gevisualiseerd met behulp van Matplotlib. Uit de vorige paragraaf blijkt dat Hansken een *dict* teruggeeft. Deze *dict* kan in een plot functie geplaatst worden als argument. Zo worden de bestandstypen op de x-as en aantal bestanden op de y-as geplaatst. Indien nodig kan de gebruiker nog extra informatie invoeren, zoals een titel en eenheden (zie code C.5).

3.4 Datasets

Vier datasets zijn beschikbaar gesteld voor het onderzoek. De aanname binnen dit onderzoek is dat de datasets in overeenstemming met de digitale forensische richtlijnen zijn veiliggesteld en de juiste extractiemethoden zijn toegepast. De datasets worden met behulp van de NFI-imaging-tool ingelezen in Hansken. Daarnaast zijn de datasets geanalyseerd met Magnet Axiom



```
[2]: # The ProjectContext makes a connection with the Hansken API, and closes this after the
     # task has been completed.
     with ProjectContext(var_endpoint, var_project) as context:
         # Make a facet query for all the available files with extensions
         results = context.search(query=Term('type', 'file'),facets=Facet('file.extension'))

         # Set dictionary
         facet_dict = {}

         # The facet will be available on the result
         facet = results.facets[0]

         # The loop checks if the extension is not longer than 11 characters. A filtering step
         # to reduce the amount of false positives. Only take the top 30 of the results
         for label in facet:
             if len(label) < 11 and len(facet_dict) < 30:
                 facet_dict[label] = facet[label].count
         #print("Facet result:", pprint.pformat(facet_dict))
         print("Facet result:", facet_dict)

Facet result: {'strings': 264909, 'png': 48349, 'plist': 46166, 'nib': 12978, 'js': 9831, '
keys': 8294, 'json': 4272, 'cam': 3899, 'jpg': 3812, 'bin': 3386, 'txt': 3256, 'mom': 2731,
'xml': 2678, 'html': 2462, 'm4a': 2046, 'austrip': 1778, 'dat': 1693, 'data': 1414, 'caf':
1265, 'db': 1217, 'heic': 1170, 'car': 1072, 'log': 817, 'pri': 810, 'svg': 777, 'csv': 74
7, 'tracev3': 706, 'p': 703, 'db-wal': 650, 'db-shm': 646}
```

Figuur 3.2: De interactieve Python omgeving gebruikt blokken, om de codes uit te voeren. In dit voorbeeld wordt een verzoek gestuurd met Hansken.py met een facet-query. De resultaten worden daarna opgeslagen in een `dict` (Python type). De top 30 hiervan zijn opgeslagen en weergegeven.

v4.11.0.24063 om eventueel resultaten met elkaar te vergelijken.

1. **Schiphol Airport Case (SAC)** Deze casus is gemaakt voor Magnet Forensics en wordt gebruikt voor onderwijsdoeleinden op de HvA. De producten waren onderdeel van een derdejaarsstage van Timo Meconi bij Magnet Forensics in samenwerking met HSL.[44] Het betreft een drugszaak met meerdere telefoons, computer en USB-stick.

2. **De Perfecte Plofkraak (DPP)** Deze dataset is gemaakt door studenten van de HvA voor een projectopdracht Van bit naar bewijs in module I jaargang 2021-2022. Hierbij bereiden studenten een fictieve plofkraak voor. Ze maken daarbij gebruik van smartphones voor communicatie en voorbereidingshandelingen.

3. **Joshua Hickman Dataset (JHD)** De dataset is beschikbaar gesteld door Joshua Hickman voor onderzoeksdoeleinden en staat op Digital Copora.[45]

4. **Matthew Sorell Dataset (MSD)** De dataset is beschikbaar gesteld door Matthew Sorell, docent van University of Adelaide. Tijdens de DFRWS EU-meeting van 29 maart tot 1 april 2021[46] is contact gelegd met Sorell, om zijn onderzoek rondom PoLF te bespreken.

3.5 Data analyse

De verwerking van de data verloopt via drie stappen. De eerste stap is het inlezen in Hansken AIO met behulp van de imaging-tool. De images worden geconverteerd naar het NFI-formaat.

In de technische interface wordt een project aangemaakt met de knop `Nieuw Project`. In het scherm is een projectnaam en korte omschrijving toegevoegd. Vervolgens is gedrukt op de knop `Maak project`. Aan het project is een nieuwe gegevensdrager toegevoegd via de knop `Nieuw image`. Voor elke image is een korte beschrijving opgegeven, om een images te kunnen uploaden en van elkaar te onderscheiden. Na het uploaden is gekozen om de data te analyseren met het extractieprofiel `Herstel`. Dit profiel bevat verschillende modules, die niet te veel rekenkracht vergen. De modules `crypto/ntlm-hash` en `metadata/hashmatch` zijn uitgeschakeld, omdat Hansken AIO geen hash-service bevat. Na het voltooien van deze extractiestap is Hansken AIO klaar voor gebruik.

Jupyterlab

De data is via de Hansken Python API door Jupyterlab opgevraagd. De scripts die zijn gebruikt, zijn een combinatie van Python en SQL. Python haalt de data op, zorgt voor de communicatie met Hansken (zie code C.6) en verzorgt de analyse in Pandas. De (aangepaste) SQL-queries zijn overgenomen uit APOLLO. Deze repository bevat het hoogste aantal *queries* op gebied van PoLF.[12] Ter illustratie zijn de codes uit sectie 3.3 gebruikt, om de Jupyterlab omgeving te laten zien (zie fig. 3.3 en 3.4).

Statistiek

Voor de statistische analyse worden de volgende formules gebruikt: correlatiecoëfficiënt van Pearson (eq. 3.1) en Kullback-Leibler Divergence (eq. 3.3). De Python-library Pandas heeft de functie `pandas.DataFrame.corr` voor de correlatiecoëfficiënt. De Python-library SciPy heeft de functie `scipy.stats.entropy` voor de Kullback-Leibler Divergence.

De eerste formule wordt gebruikt om een correlatie tussen twee grootheden te onderzoeken. De correlatiecoëfficiënt geeft een indicatie of sprake is van een lineair verband tussen de twee variabelen. Het interval ligt tussen $[-1, 1]$, waarbij de hoogste waarde een positieve en lage waarde een negatieve correlatie weergeeft. Een waarde rond de 0 betekent dat grootheden geen (lineair) verband hebben met elkaar.[47]

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.1}$$

Jennings maakte gebruik van Kullback-Leibler Divergence, om twee distributies met elkaar te vergelijken.[31] Hierbij wordt gekeken in welke mate de een distributie past in de andere distributie. Eventueel kunnen de variabelen eerst worden genormaliseerd (eq. 3.2). Deze stap zorgt ervoor dat de waarden vallen in een waarschijnlijkheidsverdeling $[0, 1] \subset \mathcal{X}$. De distributie wordt hierdoor een verdelingsfunctie.

Facets

With this code an investigator can make a bar histogram of the available extensions in the Hansken project.

[1]:
```
import matplotlib.pyplot as plt

from hansken.remote import ProjectContext, Facet
from hansken.query import TermFacet, RangeFacet, Term

url_hansken = "http://192.168.56.42"                    # Change to your own Hansken service
var_project =  "d3dd6aaa-5eb4-4eb9-8f5b-24e513aa55be"   # Change to your own project number

var_endpoint = url_hansken + ":9091/gatekeeper/"
var_keystore = url_hansken + ":9090/keystore/"
```

[2]:
```
# The ProjectContext makes a connection with the Hansken API, and closes this after the
# task has been completed.
with ProjectContext(var_endpoint, var_project) as context:
    results = context.search(query=Term('type', 'file'),facets=Facet('file.extension'))

  # Set dictionary
    facet_dict = {}

    # The facet will be available on the result
    facet = results.facets[0]

    # The loop checks if the extension is not longer than 11 characters. A filtering step
    # to reduce the amount of false positives
    for label in facet:
        if len(label) < 11:
            facet_dict[label] = facet[label].count
```

[3]:
```
plt.rcParams['figure.figsize'] = [20, 10]
plt.xticks(rotation=90)
plt.yscale("log")
plt.bar(facet_dict.keys(), facet_dict.values())
```

[3]:
```
<BarContainer object of 93 artists>
```

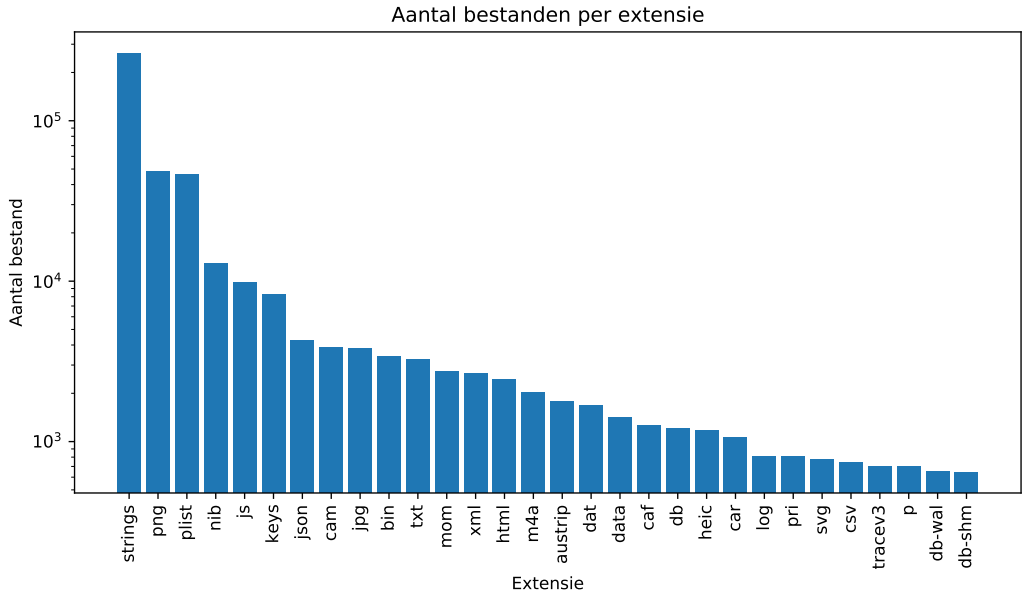
Figuur 3.3: De verschillende codes uit sectie 3.3 zijn geplaatst in de aparte blokken (*cells*). Door deze cellen te runnen wordt de code uitgevoerd in Python. Alleen de top 30 en extensies korter dan elf karakters zijn meegenomen in deze analyse.

$$\hat{\mathbf{v}} = \frac{\mathbf{v}}{\|\mathbf{v}\|} \tag{3.2}$$

De genormaliseerde waarden worden geplaatst in eq. 3.3. De uitkomst ligt tussen $[0, \inf)$. Hoe lager deze waarde bij 0 ligt, hoe beter de twee distributies bij elkaar liggen. Bij Kullback-Leibler Divergence worden twee distributies met elkaar vergeleken. Dit verschilt met de correlatiecoëfficiënt (eq. 3.1). Deze geeft een mogelijke samenhang tussen twee variabelen aan.

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log \left(\frac{P(x)}{Q(x)} \right) \tag{3.3}$$

Merk op dat de functie niet commutatief is. De deling in de logaritme zorgt voor een verschil in uitkomst, als de twee distributies worden omgewisseld.



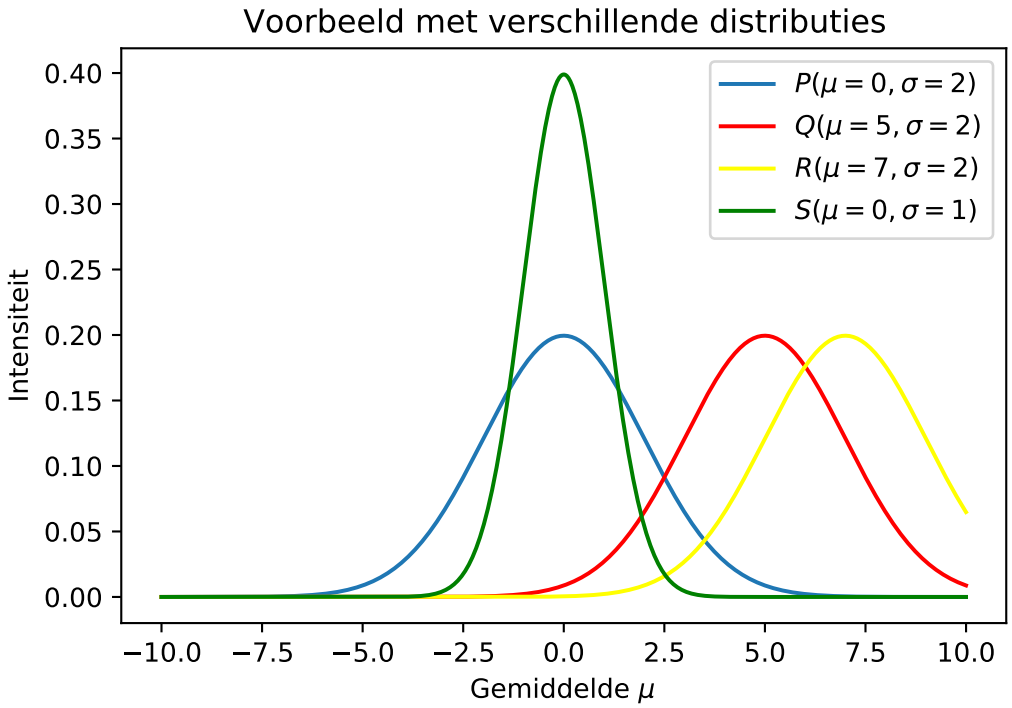
Figuur 3.4: Aan de hand van de verkregen resultaten uit de facets is een histogram gemaakt. De waarden komen uit de `dict` van 3.2.

$$D_{\mathrm{KL}}(P \parallel Q) \neq D_{\mathrm{KL}}(Q \parallel P) \tag{3.4}$$

Ter illustratie zijn vier verschillende distributies weergegeven in figuur 3.5. Daarna zijn in tabel 3.1 de Kullback-Leibler Divergence berekend. Hoe verder de distributie van de referentie distributie ligt, hoe hoger de *distance* is. Zo lijken distributies P en S meer op elkaar dan P en Q.

Tabel 3.1: De Kullback-Leibler Divergence van de verschillende distributies. Zo als in eq. 3.4 blijkt, zijn de waarden niet hetzelfde als de noemer en deler worden omgewisseld.

D_{KL}	P	Q	R	S
P	0.0	3.118767	6.055822	0.806831
Q	3.087111	0.0	0.454704	13.07042
R	5.708214	0.424098	0.0	23.119825
S	0.318147	3.436914	6.373969	0.0



Figuur 3.5: Verschillend normale verdelingen zijn weergegeven met een eigen gemiddelde (μ) en standaarddeviatie (σ).

4 Resultaten & discussie

Voor het experimentele onderzoek zijn drie deelvragen opgesteld. De eerste deelvraag 4.1 gaat in op potentiële digitale sporen die relevant zijn voor het onderzoek. Centraal hierbij zijn opzet indicatoren, eigenaarschap van digitale gegevensdrager en de betrouwbaarheid van data. De tweede deelvraag 4.2 bespreekt het gebruik van digitale sporen in de rechtszaal en de juridische basis om PoLF te gebruiken. De laatste deelvraag 4.3 neemt de huidige praktijk door en de mogelijke rol van Hansken in opsporingsonderzoeken.

4.1 Inzet digitale sporen

Uit een gesprek met Tom van de PA blijkt dat om de vraag te beantwoorden “Hoe zouden digitale sporen ingezet kunnen worden?” gekeken wordt naar drie onderdelen. Het begint met op zoek gaan naar opzet indicatoren. Het is lastig om in juridische zin opzet aan te tonen, dus daarom zijn voorbeelden gegeven welke databasen handelingen van de gebruiker registreren. *knowledgeC* houdt onder andere bij welke applicaties in beeld zijn en voor hoelang. *ineractionC* geeft meer inzicht in de contactmomenten van de gebruiker dan alleen de telefoongesprekken. *healthdb_secure* registreert het aantal stappen en hartslag. Een ander onderdeel is het individualiseren van de gebruiker. Een iPhone registreert met behulp van de *ADDataStore* per dag hoeveel keer op welke manier wordt ingelogd. In het bijzonder worden ook TouchID-registraties bijgehouden. TouchID stelt de gebruiker in staat om een iPhone met behulp van een vingerafdruk te openen. Hierdoor is het mogelijk om personen te koppelen aan een iPhone. Als laatste stap is de betrouwbaarheid toetsen van de digitale sporen. Dit kan op meerdere manieren: *dual-tool verification*, ruwe data analyse of scenario elementen toetsen aan de hand van de verklaringen. In volgende paragrafen komen deze onderdelen aan bod.

4.1.1 Opzet indicatoren

Tijdens een opsporingsonderzoek kan de opzetvraag een rol spelen. Daarvoor voert de politie in samenwerking met het OM een onderzoek uit. In het onderzoek zal niet ingegaan worden op de juridische zin van opzet-/schuldvraag.[48] Deze classificatie behoort tot het domein van de juristen. Echter, in zekere mate is de vraag wel essentieel voor de strafvervolging. Digitale sporen kunnen in tegenstelling tot de fysieke sporen meer context geven binnen een zaak. Dit kan niet alleen informatie opleveren ten tijde van het delict, maar ook ruime tijd daarvoor en

erna.[9] De berg aan digitale sporen eist wel dat er gerichte onderzoeksvragen opgesteld moeten worden. Aan de andere kant moet een digitaal rechercheur weten wat digitaal mogelijk is. Voor iPhones zijn onder andere *knowledgeC*, *interactionC*, *healthdb_secure* en cache-databasen interessant.

knowledgeC

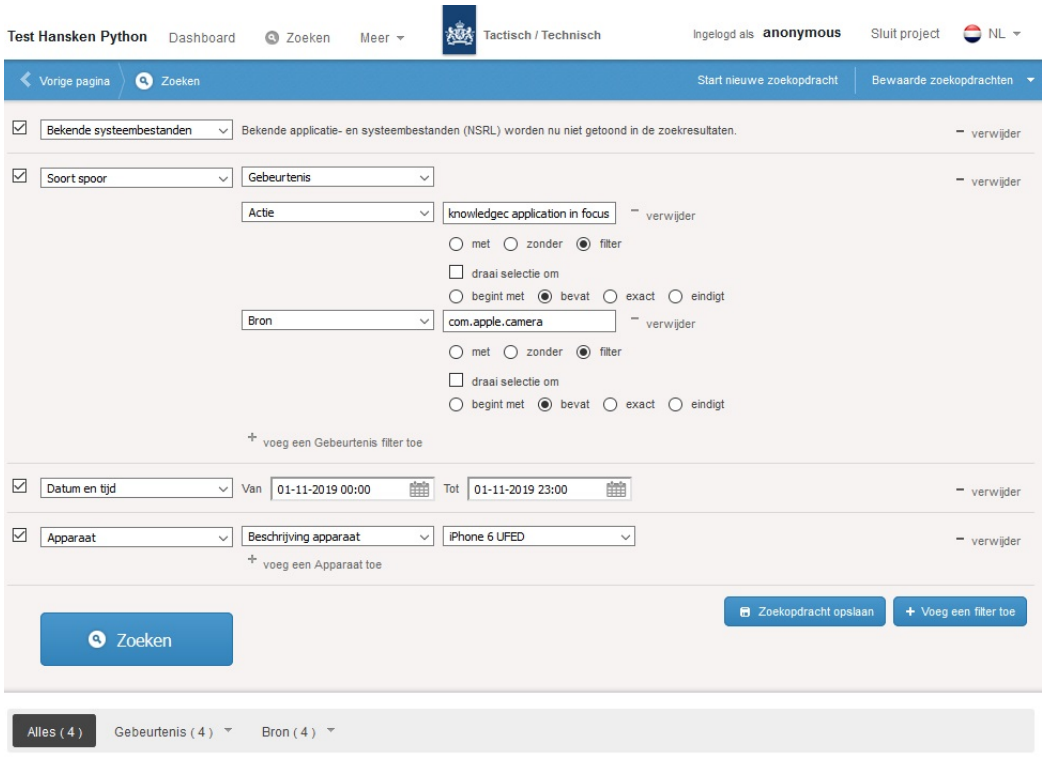
De grootste troef om een reconstructie te maken met digitale sporen, is de knowledgeC-database. Deze gegevens blijven voor ongeveer vier weken beschikbaar.[7] *knowledgeC* is beschikbaar op zowel macOS als iOS. Het bevat de browsergeschiedenis van Safari, applicatiegebruik en applicatie activiteiten. iOS op een iPhone geeft extra informatie over de vergrendeling van de smartphone, batterijgebruik en audio status.[49] Door deze informatie te combineren met elkaar ontstaat een beeld over de handelingen van de gebruiker. In het artikel van Henseler en De Poot staat een voorbeeld hoe met behulp van *knowledgeC* achterhaald kan worden of de gebruiker een foto heeft gemaakt.[9] Hierop voortbordurend is het mogelijk interessant om een locatie te bepalen. Dit is vergelijkbaar met de *forensic backtracking*, omdat meerdere digitale sporen aan elkaar worden gekoppeld.[22]

Hansken ondersteunt *knowledgeC* niet, maar dit kan wel met behulp van Hansken.py. De eerder gebouwde testomgeving biedt de mogelijk om een specifiek bestand uit Hanksen op te vragen. In dit geval gaat het om de *knowledgeC.db*-database op iPhone 6 van SAC. Een SQL-commando is uitgevoerd, die de applicaties op de voorgrond indexeert. APOLLO noemt het SQL-commando *knowledge_app_inFocus*.[12] De uitgevoerde code lijkt op het voorbeeld van de facets. De inhoudelijke behandeling van elke regel code en het proces valt buiten de strekking van dit onderzoek. De resultaten van code D.2 zijn toegevoegd als *gebeurtenis* onder het *knowlegdeC*-spoor in Hansken. Verdere uitleg is te vinden in bijlage D. Hierin staat onder andere het sporenmodel van Hansken en het terugplaatsen van resultaten als kindsporen.

De eerste vraag is: zijn door de gebruiker mogelijk meerdere foto's gemaakt op die dag (1 november 2019) met de iPhone 6?[9] Hiervoor is gebruik gemaakt van de tactische GUI van Hansken (zie fig. 4.1). Met behulp van de *Zoeken*-functie zijn twee filters geselecteerd: *Datum en tijd* en *Apparaat*. Als extra filter is *Soort spoor*→*Gebeurtenis* toegevoegd. Om de camera applicatie te specificeren zijn extra filters gebruikt: *Actie*→*knowledgec application in focus* en *Bron* →*com.apple.camera*.

De ingestelde filters geven vier resultaten terug vanuit Hansken, waarbij de gebruiker de camera applicatie heeft opgestart (zie tabel 4.1). De volgende stap is het koppelen van foto's rond het tijdstip (*timestamp*) dat de camera is ingeschakeld. Drie foto's komen hiermee overeen, die mogelijk zijn gemaakt met de iPhone 6.

Door de metadata te vergelijken is de hypothese dat de foto's zijn gemaakt met deze iPhone 6



Figuur 4.1: De filters die zijn gebruikt, in de tactische GUI-interface van Hansken. Door middel van filters wordt de zoekvraag specifieker. In het voorbeeld hierboven is de camera-applicatie opgegeven als zoekopdracht. Onderin is te zien dat de zoekopdracht vier resultaten oplevert.

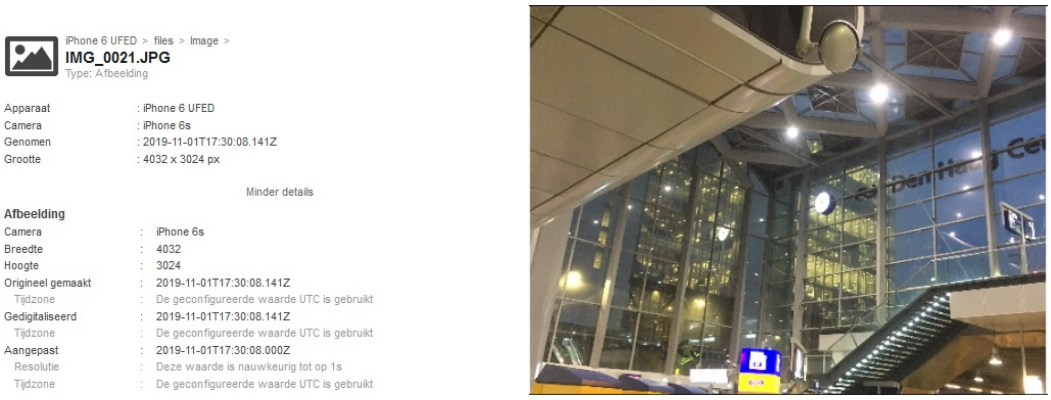
waarschijnlijker dan dat de foto's gemaakt zijn door een andere digitale gegevensdrager dan iPhone 6. Een van deze foto's is zeer interessant: IMG_0021.JPG (zie fig. 4.2). Op deze foto is de binnenkant van een treinstation gefotografeerd. Door foto naar rechts te spiegelen is in de rechterbovenhoek te lezen: "Den Haag Ce".

In dit geval is gekeken naar het camera applicatie van de iPhone. Dit voorbeeld is slechts ter illustratie, hoe *knowledgeC* gebruikt kan worden in Hansken. Belangrijk punt om op te merken is dat het opstarten van de camera applicatie niet automatisch betekent dat de gebruiker een foto heeft gemaakt. *knowlegdeC* geeft aan welke applicatie is opgestart en hoe lang deze in beeld is op de voorgrond. Extra digitale sporen rond dit tijdstip zullen gekoppeld moeten worden om een dergelijke uitspraak te kunnen doen. Zo kan de positie van de foto en telefoon (horizontaal of verticaal) een goede aanwijzing zijn.[9] De gebruiker zou ook meerdere foto's gemaakt kunnen hebben in de tijd dat de camera aanstond. Dit voorbeeld bevat geen aanleiding dat dit het geval is geweest. Wel dat de camera applicatie is opgestart, maar dat er geen foto is gemaakt rond tijdstip 16:23:16. Verder zijn tussen IMG_0014.JPG en IMG_0020.JPG waarschijnlijk vijf screenshots gemaakt in plaats van een foto met een camera. De camera applicatie is vier seconden na het nemen van IMG_0014.JPG afgesloten. *knowlegdeC* is hierdoor een goede indicator om mee te starten.

Een ander voordeel van het gebruik van *knowledgeC* is dat de rechercheur de metadata

Tabel 4.1: Overzicht van de tijden wanneer de camera applicatie is opgestart. Het verschil in een uur komt door de UTC weergave in APOLLO. Axiom heeft een optie om automatisch de tijden aan de lokale tijdzones aan te passen. Laatste kolom vermeldt de foto die gemaakt is rond het tijdstip.

Tijd knowledgeC	Tijd foto	Naam foto
2019-11-01 08:41:53	2019-11-01 09:42:05	IMG_0014.JPG
2019-11-01 16:13:37	2019-11-01 17:13:44	IMG_0020.JPG
2019-11-01 16:23:16	-	-
2019-11-01 16:29:50	2019-11-01 17:30:08	IMG_0021.JPG



(a) (b)

Figuur 4.2: De Exif-data (a) van foto IMG_0021.JPG geeft aan dat de foto genomen is met een iPhone 6. De originele foto is gespiegeld (b), zodat in de rechterbovenhoek de tekst van het station is te lezen.

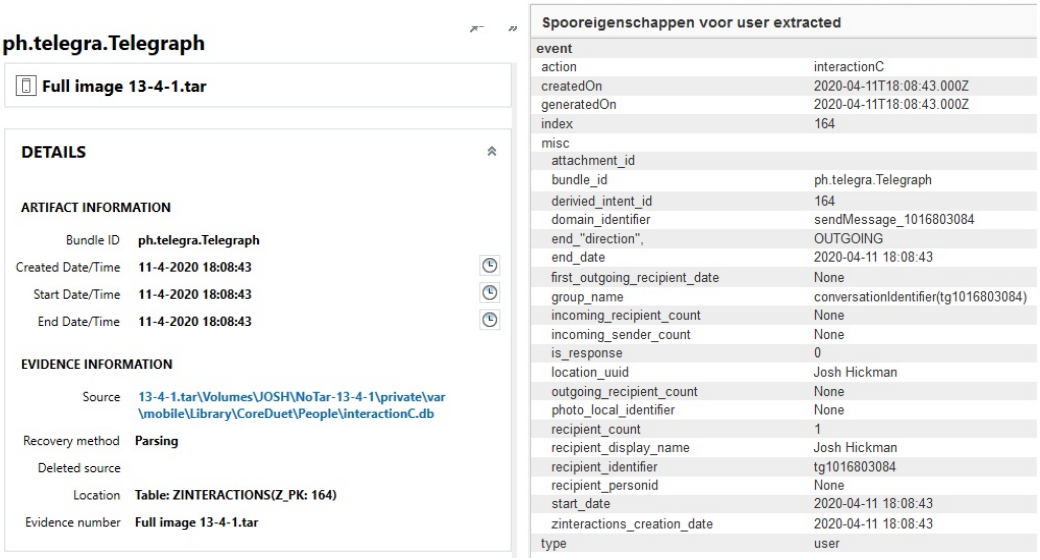
onderzoekt. Dit is een verschuiving om inhoudelijk de data te analyseren. In het geval van de casus hierboven kan een rechercheur ervoor kiezen om de foto's één voor één te bekijken. Het doornemen van een persoon zijn foto's kan gezien worden als (ernstige) inbreuk op de privacy en tijdrovend. In het geval van een levensdelict zou het van cruciaal belang kunnen zijn, maar bij lichtere vergrijpen zou mogelijk niet proportioneel zijn. De Hoge Raad oordeelt dat een opsporingsambtenaar bevoegd is om een fotogalerij op een smartphone[VIII] en "alle" foto's op een forensische kopie te onderzoeken. In het laatste arrest wijst de Hoge Raad wel erop dat het hof een motivering moet geven op welke manier "de politie selectief is geweest in het onderzoek aan de telefoon" voor foto's, maar dat dit niet meteen leidt tot bewijsuitsluiting.[IX] Oerlemans is het niet eens met deze redenering. Volgens hem ervaren jongere generaties het doorzoeken van een telefoon als ingrijpende bevoegdheid.[50] Analyse met behulp van *knowlegdeC* kan een mogelijkheid zijn, om alsnog onderzoek te verrichten zonder inhoudelijk te kijken naar de data.

interactionC

Naast knowledgeC bestaat interactionC voor registraties van contacten. Deze database is uitgebreider dan *callHistory.storedata*-database (inkomende en uitgaande telefoongesprekken).

interactionC houdt bij wanneer een applicatie die te maken heeft met communicatie, wordt opgestart. Denk hierbij aan e-mail, sms en chatdiensten. Het gebruik van deze database kan een vertekend beeld geven door de weergave in UFED PA en Axiom. UFED PA geeft de aangeroepen applicaties weer met de afzender bij sms. Axiom geeft ook de applicatie weer wanneer deze is gebruikt. Als het kan ook de (af)zender bij sms en e-mail. Het is dus niet mogelijk om een duidelijk verificatie stap uit te voeren met deze twee analyseprogramma's voor e-mails. Volgens Hermesdorf kan dit voorkomen bij UFED PA en Axiom en is manuele verificatie nodig om de resultaten te controleren.[51]

Edwards schreef in haar blogpost 'Socially Distant…' dat *interactionC* onder belicht is ten opzichte van *knowledgeC*. In haar dataset zijn 28.000 registraties beschikbaar die tot zes maanden teruggaan in de tijd.[52] Met APOLLO kan achterhaald worden voor wie de (af)zender is geweest bij chatdiensten vanaf iOS 12 (zie code C.7). Alleen de uitgaande berichten worden door *interactionC* geregistreerd. Echter, het weten met wie een gebruiker contact heeft gehad kan van cruciaal belang zijn in een onderzoek. Ter illustratie zijn in figuur 4.3 de verschillen getoond tussen Axiom en APOLLO in Hansken. *interactionC* kan een duidelijker beeld creëren



(a) (b)

Figuur 4.3: Data presentatie van Magnet Axiom (a) en APOLLO in Hansken (b). Axiom laat alleen de tijden en applicatie zien. APOLLO legt de link met mogelijk contactpersoon. De code om APOLLO-resultaten in Hansken te zetten, staat beschreven in bijlage D.

van de communicatie van de gebruiker. In een tijd waarin steeds meer gebruikers appjes sturen naar elkaar, is alleen het doornemen van de telefoongesprekken niet meer voldoende. Het is wel mogelijk om via UFED en Axiom elke chat rond een bepaald tijdstip door te nemen, maar dit kost veel tijd. Veel gebruiksvriendelijker is om de data uit de *interactionC* te halen. Hierbij geldt net zoals bij *knowlegdeC* dat het privacy-technisch beter te verantwoorden is om *interactionC* te gebruiken in het onderzoek. Inhoudelijk wordt niet in de berichten gekeken, maar wel in de

meta-data.

healthdb_secure en cache_encryptedC

De healthdb_secure.db is door Van Zandwijk en Boztas onderzocht op betrouwbaarheid van stappen en afstanden. Hieruit kwam naar voren dat de stappen nauwkeuriger waren dan de afstanden.[29] Binnen iOS zijn er twee databasen die de stappen bijhouden: healthdb_secure.sqlite ('alles') en cache_encryptedC.db (laatste zes dagen).[23] Om te controleren of deze twee databasen overeenkomen is een vergelijking uitgevoerd met behulp van de correlatiecoëfficiënt en Kullback-Leibler Divergence. De databasen uit de SAC-dataset geëxporteerd vanuit Hansken. Van deze dataset is bekend dat de twee telefoons dicht bij elkaar zijn vervoerd in een rugzak. De laatste vijf dagen zijn gebruikt met tijdsintervallen van 30 minuten.

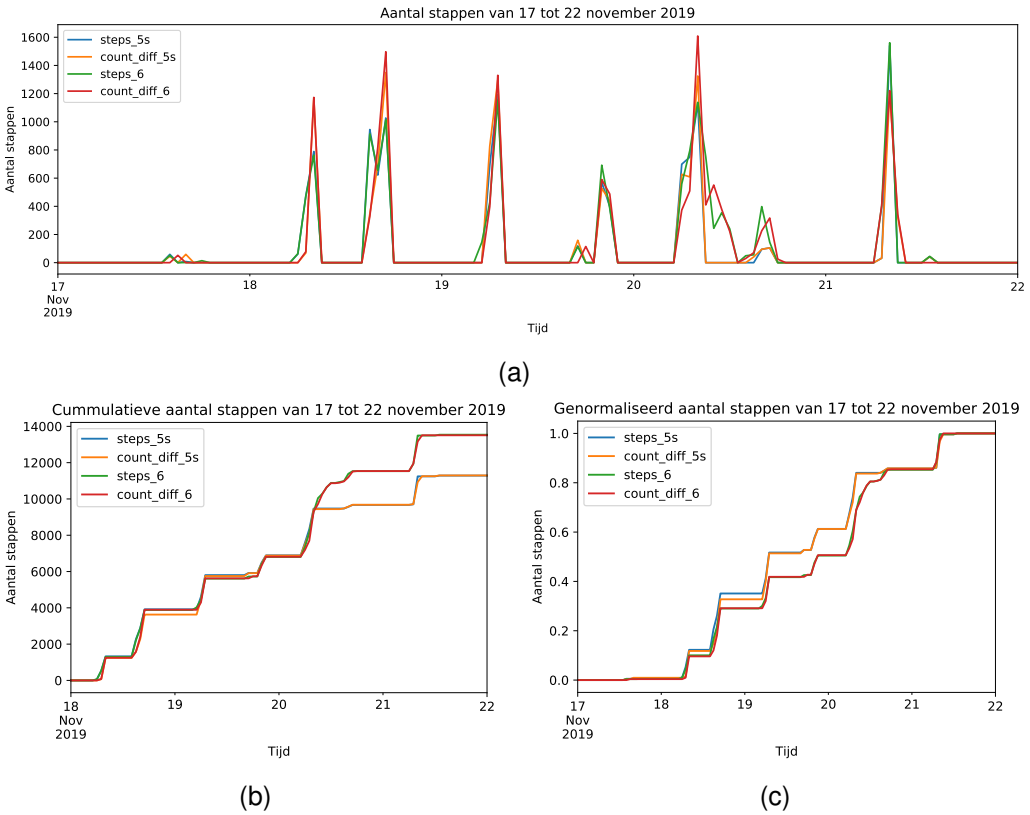
Uit tabel 4.2 blijkt dat er een hoge correlatie is tussen de tussen de *healthdb_secure* (zie code C.1) en *cache_encryptedC* (zie code C.8). Zoals verwacht liggen de waarden van de twee databasen per iPhone dichter bij elkaar, omdat ze ongeveer dezelfde afstand hebben afgelegd.

Tabel 4.2: Correlatiecoëfficiënt tussen de *healthdb_secure* en *cache_encryptedC*. Alle waarde zijn hoog postief, wat kan duiden op een correlatie tussen de twee variabelen.

R^2	**Stappen 5S**	**Cache 5S**	**Stappen 6**	**Cache 6**
Stappen 5S	1.000000	0.943441	0.940921	0.878683
Cache 5S	0.943441	1.000000	0.882596	0.941205
Stappen 6	0.940921	0.882596	1.000000	0.911545
Cache 6	0.878683	0.941205	0.911545	1.000000

Om te bepalen in welke mate de twee grafieken overeenkomen met elkaar is de Kullback-Leibler Divergence gebruikt (zie fig. 4.4). In de grafiek is te zien dat de metingen per iPhone dicht bij elkaar liggen. In tabel 4.3 zijn de waarden berekend voor de cummalatieve datapunten. Hierbij is te zien dat de meeste waarden dicht bij 0 liggen. Zoals is uitgelegd in het bijschrift zijn de inf-waarden vals positief. Hieruit kan geconcludeerd worden dat de waarden van metingen van de *healthdb_secure* en *cache_encryptedC* nagenoeg gelijk zijn. Deze resultaten komen overeen met de aanname van Van Zandwijk en Boztas dat deze twee databasen hetzelfde aantal stappen registreren.[29]

Hoewel de grafieken op elkaar lijken, hoeft het niet te betekenen dat de telefoons bij elkaar in de buurt zijn geweest.[31] In dit specifieke geval bleek uit de contextinformatie dat de twee telefoons met elkaar vervoerd waren. Daarom wordt kritischer ingezoomd op de verwerking van de data. Uit de resultaten van Van Zandwijk en Boztas blijkt dat gemeten afstand van de iPhones in de rugzak verschillen ondanks dat deze afstand vooraf vaststond.[29] De gekozen



Figuur 4.4: Bovenste grafiek geeft het aantal stappen weer van uit de *healthdb_secure* (steps) en *cache_encryptedC* (count_diff) van de twee iPhones uit de SAC-dataset. In de onderste twee grafieken zijn de stappen bij elkaar opgeteld (b) en genormaliseerd (c).

methode om eerst de dataset te normaliseren zorgt ervoor (eq. 3.2) dat eventuele verschillen in metingen zoals stapgrootte in mindere mate een rol spelen. Over dezelfde afstand kunnen twee personen immers een verschillend aantal stappen hebben gezet. Het gebruik van deze methode kan alleen als vooraf bekend is tussen welke twee tijdstippen wordt gemeten. Hiervoor kunnen onder andere zendmastgegevens een rol in spelen. Een beperkende factor is verdelingsfunctie van $[0, 1] \subset \mathcal{X}$. Zoals blijkt uit dit interval zal altijd functie altijd beginnen bij 0 en/of eindigen bij 1. Dit kan een vertekend beeld geven door de normalisatie. Aan te raden is om als niet bekend is of twee telefoons in elkaars buurt zijn geweest, eerst de Kullback-Leibler Divergence toe te passen zonder normalisatie.[31]

Cache-bestanden

Net zoals computers hebben iPhones ook cache-bestanden. Cache staat voor informatie die tijdelijk wordt bewaard op een digitale gegevensdrager. In het geval van de iPhone zijn dat bestanden zoals *cache_encryptedA/B/C* en *CurrentPowerlog.PLSQL*. Afhankelijk van het type en iOS-versie kan verschillen welke van de *cache_encryptedA/B/C*-databasen aanwezig zijn. De *CurrentPowerlog.PLSQL* bevat veel meer informatie dan de *knowledgeC*-database. Voor dit onderzoek is niet uitvoerig gekeken welke informatie beschikbaar is, omdat bij de gebruikte

Tabel 4.3: De verschillende waarden van de Kullback-Leibler Divergence liggen vlak bij 0. Dat betekent dat de distributies erg op elkaar lijken. In de laatste kolom staan drie keer inf. Deze waarden zijn vals positieven, omdat de metingen uit *Cache 6* 0 bevatten. Hierdoor wordt de deling een limiet, die de functie berekent als inf.

D_{KL}	**Stappen 5S**	**Cache 5S**	**Stappen 6**	**Cache 6**
Stappen 5S	0.0	0.001102	0.004357	inf
Cache 5S	0.000943	0.0	0.004334	inf
Stappen 6	0.004285	0.004321	0.0	inf
Cache 6	0.005532	0.004376	0.000749	0.0

datasets niet alles is gedocumenteerd tot op de minuut nauwkeurig. Gezien de soort informatie die uit deze database gehaald kan worden, is het de moeite waard om dit verder te onderzoeken. Omdat het tijdelijke bestanden zijn slaat de iPhone de laatste zeven dagen op. Willen de opsporingsdiensten deze informatie gebruiken, dan zal een onderzoeksteam hiermee rekening moeten houden voor een eventuele aanhouding of doorzoeking.

4.1.2 Eigenaar koppelen aan digitale sporendrager

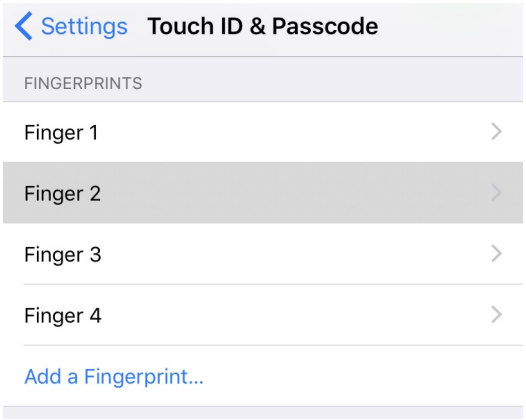
Digitale sporen koppelen aan een persoon is lastiger dan bij fysieke sporen. Uit het gesprek met Van Dam blijkt dat verdachten vaak een alternatief scenario opgeven. Zo zou iemand anders dan de verdachte handelingen hebben uitgevoerd op hun digitale gegevensdrager (zie bijlage B.1). Gina Doekhie gaf in een presentatie tijdens E-discovery aan dat een verdachte zei dat iemand anders illegale bestanden op zijn computer had gedownload. Hierbij zou de verdachte gehackt zijn.[53] Voor een digitale gegevensdrager blijkt het lastig om een specifieke gebruiker aan te wijzen. Verschillende personen kunnen handelingen uitvoeren op één apparaat. Denk daarbij aan een computer in een bibliotheek of flexwerkplekken, die van dit principe gebruik maken. De smartphone is inmiddels een uiterst persoonlijk apparaat.[10] Mensen zullen niet snel geneigd zijn om onbeperkt toegang te verlenen aan iedere willekeurige persoon. Hooguit een paar mensen mogen dit, en nog een beperkter aantal mensen zullen waarschijnlijk het wachtwoord kennen van de gebruiker. Zo wordt het aantal personen dat toegang heeft tot de smartphone gelimiteerd. Toch kan met behulp van biometrische informatie de mogelijke authenticatie methode achterhaald worden.

Biometrische ontgrendeling

Om te achterhalen op welke manier is ingelogd zal eerst bepaald moeten worden welke authenticatie methode mogelijk zijn gebruikt. De meest eenvoudige optie om de telefoon te ontgrendelen is geen authenticatiemethode te gebruiken. Geen beveiliging heeft als gevolg dat alle gegevens op de telefoon niet worden versleuteld via de Keybag-functionaliteit. Keybag zorgt ervoor dat bestanden versleuteld kunnen worden op een iPhone.[54] Wang et al. maken

onderscheid in de verschillende mogelijkheden om een smartphone te vergrendelen. Zo kan de gebruiker een wachtwoord te kiezen (*text-based secret*) of patroon te tekenen (*graphical secret*). Een andere manier is door middel van een vingerafdruk, palmscan, ogen of gezicht (*physiological biometrics*).[55] De laatste optie is forensisch gezien het meest individualiserend, omdat dit fysiologische kenmerken bevat van een 'unieke' gebruiker.

Weinig literatuur is beschikbaar om te achterhalen op welke manier de gebruiker is ingelogd. Edwards schrijft in haar blog 'Pincodes, Passcodes, & TouchID on iOS', hoe dit mogelijk wel zou kunnen. Daarvoor gebruikt ze de *ADDataStore.sqlite*-database. Volgens Edwards somt deze database bepaalde acties op, zoals inlogpogingen. Het aantal acties wordt bijgehouden per dag, waarbij de tijd gelogd wordt volgens *unixepoch*, UTC. Ze geeft in een disclaimer aan dat deze database verder onderzocht moet worden door middel van eigen tests. Als extra tip geeft ze mee, dat achterhaald kan worden welke vingers op de iPhone geregistreerd zijn.[56] In figuur 4.5 is weergegeven hoe dit via instellingen gecontroleerd kan worden. Opsporingsambtenaren



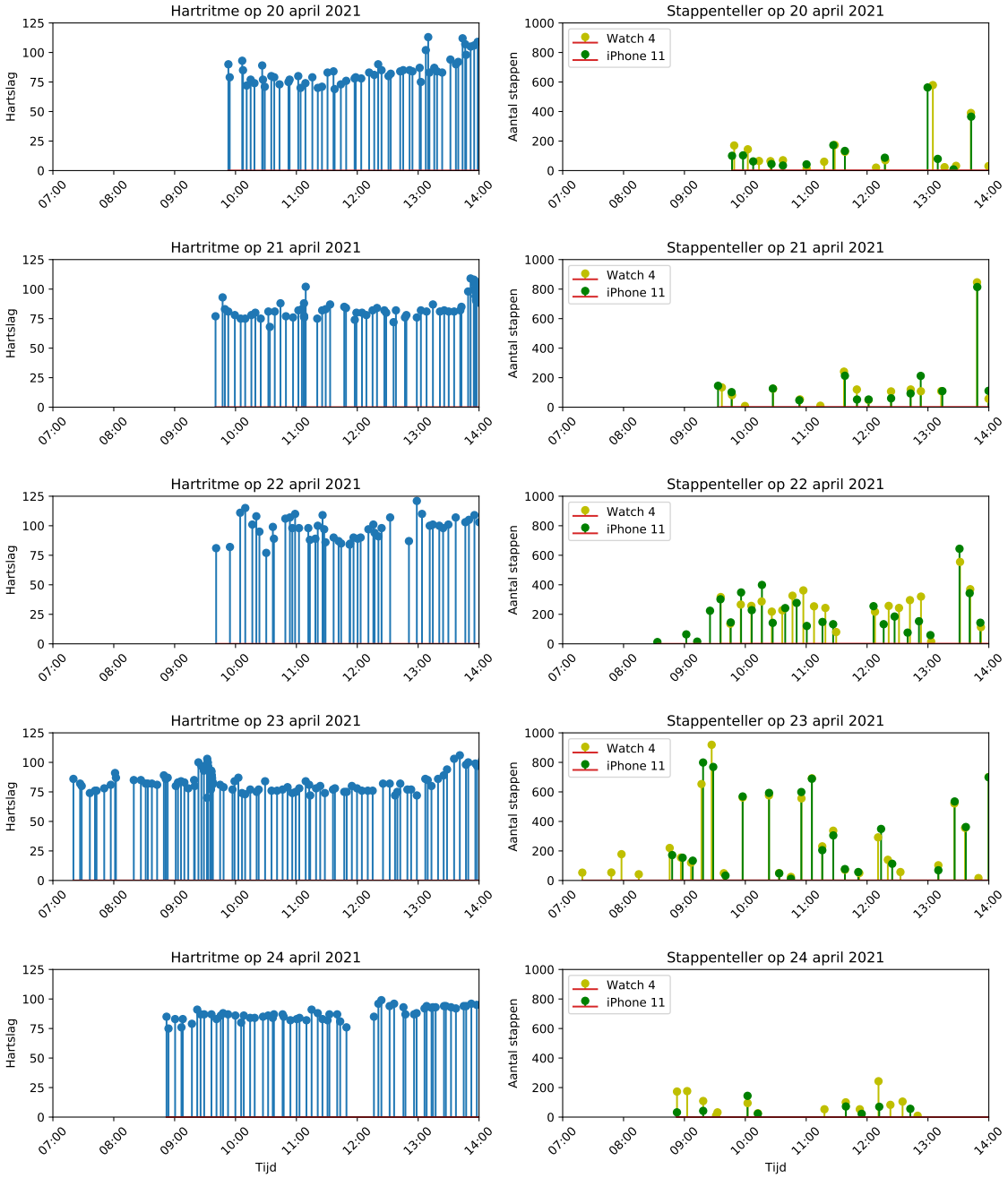
Figuur 4.5: Screenshot van een iPhone waar de TouchID onder Instellingen is weergegeven. Als de gebruiker wil weten welke vinger geregistreerd is per afdruk, dan drukt hij op de scanner. De vingerafdruk op de iPhone kleurt dan licht grijs.[56]

mogen de telefoon met behulp van een vingerafdruk ontgrendelen volgens de Hoge Raad.[X] Het scherm om te kunnen bepalen welke vingerafdruk het geweest kan helaas niet volgens de methode van figuur 4.5. Hiervoor moet namelijk de pincode van de iPhone worden ingetoetst. Wat wel kan is het registreren welke vinger gebruikt is. In de *ADDataStore* wordt het aantal pogingen per vingerafdruk geregistreerd. Deze database houdt het voor de hele dag bij, dus handelingen per tijdstip kunnen niet exact achterhaald worden. Daarnaast is het mogelijk dat drie vingerafdrukken worden geregistreerd in een afdruk op de iPhone.[57] Dat betekent dat een of meerdere personen toegang kunnen hebben tot de iPhone. Mocht het nodig zijn in het onderzoek, dan is het verstandig om alle vingers van de verdachte over de scanner heen te halen. Achteraf kan achterhaald worden of al deze vingers in de geregistreerde vingerafdrukken op de iPhone aanwezig zijn.

Biometrische trends

Naast onderzoek naar de individualiserende kenmerken biedt de iPhone in combinatie met de Apple Watch ook mogelijkheden. Een smartwatch houdt niet alleen de tijd bij. Inspanningen en work-outs worden geregistreerd door de Apple Watch. Deze gegevens worden opgeslagen in iCloud of de iPhone in de *healthdb_secure*-database.[31] Deze data wordt niet veelvuldig gebruikt in opsporingsonderzoeken, blijkt uit een artikel op Medium. Zo is aan hand van de *steps climbed* een reconstructie door de Duitse politie gemaakt, waarbij de verdachte het lichaam van het slachtoffer naar de rivier heeft de gedragen en daar heeft gedumpt. De gegevens van de verdachte zijn iPhone komen na een reconstructie van de Duitse politie hiermee overeen.[58]

Bij ander voorbeeld in Australië verklaarde de schoondochter dat een groep mannen haar schoonmoeder (slachtoffer) hadden achtervolgd naar huis. Eenmaal thuis hebben deze mannen twintig minuten gediscussieerd, voordat ze het slachtoffer om het leven brachten. De schoondochter heeft dit niet gemerkt, omdat de deur naar de keuken dichtzat. Hierna werd de schoondochter ontdekt. De mannen bonden haar vast en zijn daarna gevlucht. Om 10:10 PM lukte het de schoondochter om een buurman op de hoogte te stellen. Nader onderzoek naar de Apple Watch van het slachtoffer wees uit dat het slachtoffer rond 6:38 PM is aangevallen en vermoedelijk rond 6:45 PM is overleden. Het verhaal van de schoondochter wordt hiermee in twijfel getrokken. Volgens de aanklager is het gat van drie uur genoeg tijd om op te ruimen en bebloede kleding weg te gooien. De aanklager gaat ervan uit dat de schoondochter haar schoonmoeder om het leven heeft gebracht.[58]



Figuur 4.6: Aan de hand van de analyse van de hartslagmeter en stappenteller kunnen bepaalde patronen worden herleid. In de stappenteller is onderscheid gemaakt tussen de Apple Watch en de iPhone. Deze data komt uit MSD-dataset met een tijdsverschil van +09:30 UTC voor 20, 21 en 22 april en +08:00 UTC voor 23 en 24 april 2021. De tijdverschillen zijn verwerkt naar lokale tijd van de gebruiker (zie code C.9)

Volgens Matthew Sorell is de essentie van PoLF niet het herkennen van patronen in lange datareeksen. Het gaat volgens hem om een bepaalde handeling die een gebruiker uitvoert op een specifieke tijd of plaats. Door deze handeling te vergelijken met andere dagen kan een correlatie gemaakt worden, wat wel en niet behoort tot het dagelijks patroon. Uit figuur 4.6 blijkt dat de hartslagmetingen altijd voor 10:00 beginnen. Ook neemt de intensiteit van het

aantal meetpunten van de hartslagmeter toe bij het zetten van meer stappen. Deze informatie is interessant om te gebruiken voor een onderzoek, als bijvoorbeeld een ander patroon te zien is op de dag van een delict. Echter, een expert moet ervoor waken dat hij geen sterke conclusies hieruit trekt.[59]

4.1.3 Betrouwbaarheid

De betrouwbaarheid controleren van digitale sporen blijft een lastig proces.[60] Zo is het wel eens voorgekomen dat de verkeerde conclusies zijn getrokken aan de hand een verkeerde interpretatie van tijdzones.[61] Waar FO in de loop der jaren een stevige fundering heeft opgebouwd, staat DFO nog aan het begin.[62] Dit biedt kansen om te leren van FO, maar creëert ook risico's in de uitvoering. Rechercheurs krijgen geen effectieve trainingen, kunnen vaak niet kritisch kijken naar digitale resultaten of merken eventuele gaten in de analyse niet op.[37] Volgens Page et al. heeft DFO zelfs het minst robuuste kwaliteitswaarborgingssysteem van alle forensische disciplines.[63] Dat komt de betrouwbaarheid van DFO-onderzoeken niet ten goede. Technologieën veranderen in een snel tempo en dataopslag neemt alleen maar toe. Vooral dat laatste heeft als gevolg dat grote hoeveelheden data verwerkt moet worden en veel relevante artefacten niet onderzocht kunnen worden.[37]

Verificatie

Een belangrijke stap om te controleren of de resultaten betrouwbaar zijn, is door middel van *dual-tool verification*. Door twee verschillende analysemethoden kan gecontroleerd worden of de resultaten uit de ene tool overeenkomen met een andere tool.[64] Dit is vergelijkbaar met de identificatie van drugs bij FO.[65] Voor digitale sporen kan het lastig zijn om *dual-tool verification* uit te voeren.[60] Uit de vorige deelvraag 4.1 blijkt namelijk dat niet alle data wordt opgepakt door de verschillende analyseprogramma's.[51]

Bij *interactionC* kwam naar voren dat niet "alle" beschikbare informatie door UFED en Axiom worden weergeven. Terwijl met APOLLO en Hansken wel achterhaald kan worden met wie de gebruiker contact heeft gehad. Idem voor de *healthdb_secure*, waar UFED het aantal registraties van stappen weergeeft, terwijl APOLLO en Axiom wel de stappen eruit halen. Dit maakt de verificatie stap niet onmogelijk, maar dit kost extra tijd en middelen om de resultaten te controleren.[60] Elke rechercheur zou moeten weten wat een tool wel en niet kan.[66] Toch blijven de resultaten uit het analyseprogramma leidend als de waarheid in de ogen van rechercheurs.[6,67]

Het schrijven van eigen scripts helpt in het begrijpen hoe de data geanalyseerd wordt. Het traint de rechercheur om (on)bewust te onderbouwen wat achter de schermen gebeurt tijdens de analyse.[60] In de ideale wereld zou elke rechercheur een programmeur zijn met zijn eigen set

aan scripts en tools. De opsporingsdiensten kunnen echter niet verwachten dat tactische en forensische rechercheurs experts worden op gebied van programmeren. Tegelijkertijd is het wel wenselijk om deze groep bepaalde analyses uit te laten voeren. Daarom is een tweedeling qua rapporteren nodig, vergelijkbaar met de Hansken tactische en technische interface. Tactische en forensische rechercheurs mogen alleen beschrijven wat ze zien: “op de smartphone is een foto gevonden”. Hierbij mogen ze geen conclusies trekken zoals: “de foto is gemaakt met deze smartphone”. Zulke interpretaties en conclusies moeten ze overlaten aan hun digitale collega’s. Aan de digitaal rechercheur is het uiteindelijk om deze verificatie stap uit te voeren en de betrouwbaarheid te beoordelen.

Ruwe data

Volgens Adrie Stander (docent Forensische ICT, Hogeschool Leiden) zal een digitaal rechercheur weer terug moeten naar de bits en bytes.[68] IoT-apparaten zijn hiervan een voorbeeld, omdat deze vaak uitgelezen moeten worden op byte-niveau.[64] Voor smartphones geldt hetzelfde met de nadruk op onderzoek naar het bronbestand. APOLLO en Hansken lezen de data in en voeren een analyse uit. De resultaten hiervan worden gepresenteerd aan de gebruiker. Hierdoor is de gebruiker afhankelijk van de programmeur die de software verzorgt. Met Hansken.py is het mogelijk om zelf tools te ontwikkelen. Dat heeft zowel een voordeel dat meer bestanden onderzocht kunnen worden, maar als nadeel dat de scripts ook gevalideerd moeten worden.

Scenario element toetsen

De derde methode om de betrouwbaarheid te bepalen, is aan de hand van de data de verklaring van de verdachte te toetsen. Zoals al eerder is gezegd, kan een verdachte een alternatieve verklaring geven waarom specifieke digitale sporen op zijn telefoon zijn gevonden. Mogelijk hoort dit in het dagelijks patroon van de verdachte. Hierbij moet niet alleen gekeken worden naar de artefacten op de dag van het delict. Het zijn de artefacten de dagen voor het delict die interessant zijn. Met behulp van figuur 4.6 kan een verklaring van de gebruiker interessant zijn. Voor het onderzoek zijn vragen gesteld aan Matthew Sorell, om contextinformatie te krijgen over de genereerde data uit zijn dataset.

- De dagen 20, 21 en 22 april lijken in hartslag en stappenteller van de Apple Watch rond dezelfde tijd te meten. Op 22 april zijn stappen geregistreerd bij de iPhone voor het hartslagmetingen. *Sorell geeft aan dat zijn Watch pas om doet, als hij aan zijn bureau gaat zitten. In de nacht laadt hij zijn horloge op. Soms neemt hij zijn iPhone ook mee, terwijl hij zijn horloge niet om heeft.*
- Op 23 april start de meting van de hartslag eerder dan de andere dagen. *Sorell staat meestal rond 06:30 op in de ochtend. Hij drinkt dan koffie en leest e-mails. Als hij zijn*

kinderen naar school brengt, dan draagt hij soms ook zijn horloge. Anders doet hij dat pas later op de dag.

- Op 24 april zijn weinig metingen te zien op de stappenteller.

Uit de gevonden resultaten blijkt niet dat Sorell vanaf 06:30 opstaat. Uit zijn verklaring komen geen tegenstrijdigheden voor. Dagen 20, 21 en 22 later de hartslag meten dan 23 april komen overeen met de verklaring dat Sorell soms zijn kinderen naar school brengt. Nu lijkt 24 april eruit te springen als dag waarbij weinig stappen gemeten wordt. Dat kan kloppen met de informatie dat dit een zaterdag is geweest. Door telkens stukjes informatie te koppelen aan de data ontstaat een beeld wat mogelijk gebeurd is. Dit vergelijkbaar met *forensic backtracking* en het CAI-model.[18] In een strafrechtelijk onderzoek zou het voor kunnen komen dat rond de tijd van het delict niks wordt gemeten. Als verdachte kan beamen dat de dagen ervoor hij specifieke handelingen uit heeft gevoerd, dan kan dit getoetst worden aan de hand van de data. Als het verhaal klopt met de gevonden gegevens, dan neemt de betrouwbaarheid toe. Voor de niet gemeten data rond het delict zal de verdachte een alternatief scenario met de reden waarom de iPhone niks heeft gemeten.

Resumé

Uit de resultaten blijkt dat in de praktijk vaak drie dezelfde vragen worden gezet met betrekking tot het inzetten van digitale sporen. Is er sprake van opzet, wie is de eigenaar van de digitale gegevensdrager en hoe betrouwbaar zijn de resultaten? Voor de opzet-vraag zijn meerdere voorbeelden gegeven hoe digitale sporen uit *knowledgeC*, *interactionC*, *healthdb_secure* en cache-databasen gebruikt kunnen worden. Hierbij valt op dat de analyse meer betrekking heeft op de meta-data, dan de inhoudelijke data. Verder kan de combinatie van verschillende digitale gegevensdragers zorgen voor een betere reconstructie, zoals blijkt uit de hartslagmeter en de stappenteller. Om de eigenaar te kunnen bepalen, kan TouchID of biometrische gegevens gebruikt worden. TouchID registreert het aantal inlogpogingen met vingerafdruk en biometrische gegevens kunnen in combinatie met elkaar een beeld creëren over de gebruiken van ene persoon. Als laatste is de betrouwbaarheid van digitale sporen doorgenomen. De verschillende analyseprogramma's geven op een eigen manier de resultaten weer. Het uitvoeren van *dual-tool verification* maakt dit lastig. Ruwe data analyse aan de andere kant vergt expertise en kost tijd. Een makkelijker manier is de verklaringen van verdachte en getuigen te toetsen aan de hand van digitale sporen. Dit is vergelijkbaar met *forensic backtracking* en het CAI-model.

4.2 Potentie digitale sporen in scenariogericht onderzoek

Als in 40% van de gevallen digitale sporen blijven liggen[3], dan betekent dat ze niet in de rechtszaal komen. Om een beeld te krijgen wat de meerwaarde van digitale sporen in de

rechtszaal is, wordt het hoger beroep van de Bûterwei doorgenomen. De punten waar het gerechtshof op let in de beoordeling van de digitaal bewijsmateriaal, staan hierbij centraal. Daarnaast wordt gekeken onder welke bevoegdheid onderzoeken van PoLF valt. Hoewel PoLF-analyse voordelen heeft tijdens onderzoeken, geeft het geen garantie dat het altijd ingezet mag worden. Met het oog op de modernisering Wetboek van Strafvordering wordt getoetst hoe de wetgever kijkt naar dit soort analyses.

Hoger beroep Bûterwei[XI]

In het literatuuronderzoek is kort ingegaan op de Bûterwei-zaak aan de hand van vonnis van de rechtbankVII, krantenartikel van Modderkolk[69] en het artikel van Henseler en De Poot[9]. Naar aanleiding van de uitspraak van het Gerechtshof Arnhem-Leeuwarden op 1 december 2020 zal een korte review worden uitgevoerd, die betrekking hebben op PoLF-artifacten.

Voor het Bûterwei-onderzoek waren de bewegingen van de telefoon van het slachtoffer interessant. Uit Google-Timeline gegevens worden locatiegegevens en activiteiten teruggevonden. Google geeft hierbij een waarde van 0 tot 100, wat aangeeft hoe zeker het bedrijf is dat de activiteit heeft plaatsgevonden.[69] In de uitspraak van het hof worden deze locatie en activiteiten benoemd (zie tabel 4.4).

Tabel 4.4: Tijdlijn van de digitale sporen die zijn gevonden op de Google-Timeline van het slachtoffer. Het gaat om gegevens van 9 juli 2017.[XI]

Tijd	Confidence level	Soort activiteit (nauwkeurigheid)
00:01:57	-	Gebruiker bevindt zich op festival terrein (11m)
00:27:38	95	Gebruiker is aan het lopen
00:40:09	100	Hoek van toestel verandert aanzienlijk
00:43:18	100	Toestel beweegt niet
00:44:56	100	Hoek van toestel verandert aanzienlijk
00:46:34	100	Toestel beweegt niet
00:49:54	100	Toestel beweegt niet
00:56:03	100	Toestel beweegt niet
01:05:48	100	Toestel beweegt niet
12:44:07	-	Toestel verandert niet van locatie (12m)

Om de exacte locatie in combinatie met de betrouwbaarheid vast te stellen gebruikt Google een Gaussische (normale) distributie. Hierbij geldt dat de betrouwbaarheid van de punten afhankelijk is van de grootte van straal ($r \sim \sigma$). Uit onderzoek van de politie blijkt dat vindplaats overeenkomt de Google-data met inachtneming van de betrouwbaarheid. Telecomdeskundige Pluijmers van het Nationaal Forensisch Onderzoeksbureau (NFO) voerde contra-expertise uit in eerste aanleg. Hij kwam tot dezelfde conclusie als het politieonderzoek met een eigen computerscript. In hoger beroep achtte de verdediging de Google-data niet accuraat en onbetrouwbaar. Deze conclusie kwam naar voren uit een deskundigerapportage van het Nederlands Forensisch

Incident Response (NFIR). In de uitspraak komt niet goed naar voren waar het NFIR deze bevindingen op baseert. Het NFIR heeft namelijk geen ruwe data onderzoek uitgevoerd, terwijl de politie en Pluijmers dat wel hebben gedaan. Omdat de gegevens van deze twee partijen overeenkwamen, twijfelt het hof niet over de betrouwbaarheid van de Google-data. Het hof heeft de Google-data gebruikt als indicatie van de locatie van de telefoon, omdat dit niet met absolute zekerheid vastgesteld kan worden.

Een vergelijkbaar locatieonderzoek is uitgevoerd met de telefoon van de verdachte. Het hof heeft dezelfde soort conclusies overgenomen als bij de telefoon van het slachtoffer. De verdachte reed rond in een auto in de nacht van 8 op 9 juli 2017. De locatiegegevens van de telefoon zijn vergeleken met de camerabeelden langs de route. Deze gegevens blijken nagenoeg overeen met elkaar te komen. In een verklaring gaf de verdachte aan dat ze een specifiek adres had bezocht. Het gerechtshof was van oordeel dat de verklaring van verdachte over de middag overeenkwamen met de digitale sporen. Daarom twijfelde ze niet over de betrouwbaarheid hiervan. Over de nacht verklaart ze dat ze na een gesprek met verdachte om 00:22 in een Mercedes is gestapt en haar telefoon in het middenconsole van de auto heeft gelegd. Ze heeft haar auto zonder telefoon verlaten en is gaan lopen. Rond 00:45 kwam ze terug bij de auto. Ze wilde het slachtoffer bellen, maar haar telefoon was zwart. Daarop is ze naar huis gereden. De politie heeft haar verhaal getoetst aan de hand van gegevens uit haar telefoon.

Het batterijgebruik van de iPhone rond het tijdstip van het delict kon niet worden achterhaald. Een *full image* met deze data werd pas later veiliggesteld, omdat verdachte eerst was aangemerkt als getuige. Vandaar dat zowel de politie als Pluijmers de loggegevens van de iPhone hebben onderzocht. Pluijmers verklaarde in eerste aanleg dat alleen de laatste vijf dagen beschikbaar waren qua batterijgegevens. Volgens hem was de batterij niet in slechte staat. Verder zijn verschillende logregels aangemaakt rond 00:46 uur. Dat paste niet bij het uitschakelen van een lege accu.

Op drie punten is het arrest belangrijk voor PoLF. Het eerste punt is de bepaling van de betrouwbaarheid van de locatiegegevens. Zowel de politie als Pluijmers kwamen op dezelfde conclusie aan de hand van de ruwe data. Pluijmers gebruikte een eigen script en kwam op exact op dezelfde posities uit als de politie. Met andere woorden, analyses met een eigen computerscript kan gebruikt worden als controle van data. Het tweede punt gaat over de toetsing van de verklaring van de verdachte aan de hand van de digitale data. De verklaring en digitale sporen betreffende de middag achtte het gerechtshof betrouwbaar. Het derde punt is het batterijgebruik. Pluijmers gaf aan dat de gegevens tot vijf dagen teruggaan. Het klopt inderdaad dat dit voor het cache-bestand *CurrentPowerlog* geldt.

Echter, *knowlegdeC* houdt de batterijgebruik ook bij voor vier weken. Het delict speelde zich

af in juli, maar de gegevens waren beschikbaar van de maand december. *knowlegdeC* had meer datapunten kunnen opleveren, om zekerder te weten of de telefoon niet onder specifieke omstandigheden uitvalt. Uiteindelijk kon met behulp van de loggegevens en batterijgegevens de verklaring van de verdachte getoetst worden. De *throttling mode* zorgt ervoor dat de iPhone geen piekprestaties levert. Deze mode schakelt in als het batterijgebruik onder 40 procent ligt. Vanaf de avond voor het delict is deze service op de iPhone van de telefoon niet meer ingeschakeld, wat suggereert dat de telefoon voor minimaal 40 procent was op geladen.[69] Het gebruik van de ruwe data analyse en batterijgegevens biedt kansen om PoLF in juridische zin te gebruiken in de rechtszaal.

Gemoderniseerd Wetboek van Strafvordering

Deze wet ligt ten tijde van het schrijven in concept bij het Nederlands parlement. Het is mogelijk dat delen hiervan wijzigen als het wetboek definitief wordt ingevoerd.

In 2026 staat de invoering van het nieuwe Wetboek van Strafvordering (Sv) gepland. Deze vernieuwing was volgens de Commissie implementatie nieuw Wetboek van Strafvordering (Commissie-Letschert) hard nodig. Doelstelling is om het Sv toegankelijker en begrijpelijker te maken voor de burger en strafrechtketen. Een van de onderdelen hierin is ook het bieden van een juridische basis voor 'digitale gegevensdragers'. Deze term is vrij abstract, om te zorgen dat het nieuwe Sv toekomstbestendig is.[XII] In het bijzonder worden meerdere artikelen gewijd aan onderzoek aan digitale gegevensdragers. In de memorie van toelichting wordt vaak verwezen naar de Commissie modernisering opsporingsonderzoek in het digitale tijdperk (Commissie-Koops). Deze commissie had de opdracht om het conceptvoorstel Boek 2, Sv te beoordelen en hierover te adviseren.[XIII] Met name de onderwerpen biometrische vergrendeling en criterium van stelselmatigheid zijn relevant voor de toepasbaarheid van PoLF.

De Commissie-Koops is zich ervan bewust dat biometrische vergrendelingen meer een rol gaan spelen in opsporingsonderzoeken. Dit zou geen belemmering moeten zijn voor het opsporingsonderzoek. Het ontgrendelen van een digitaal gegevensdrager met behulp van biometrie zou niet in strijd zijn met het nemo-teneturbeginsel.[X XIV 50] Onder lichte dwang zou een digitale gegevensdrager ontgrendeld mogen worden, zoals is bevestigd door de Hoge Raad. In de Memorie van Toelichting van concept Sv staat dat dit zowel voor de verdachte als niet-professioneel verschoningsgerechtigden geldt. Deze ontgrendeling kan op bevel van de OvJ plaatsvinden.[XV] Voor de opsporing biedt dit kansen om met behulp van de biometrische vergrendeling gebruikers te identificeren rond een bepaald tijdstip. In 4.1.2 is besproken op welke manier dit zou kunnen. Hierbij ligt dus een juridische basis om gebruikers te identificeren aan de hand van biometrische gegevens. Echter, de interpretatie om dit te kunnen concluderen vereist meer onderzoek.

Het criterium van stelselmatigheid bepaalt de juridische basis, om onderzoek uit te voeren op een digitale gegevensdrager. Zowel de Commissie-Koops als Memorie van Toelichting van concept Sv gaan uitvoerig in op wat onder deze stelselmatigheid wordt verstaan. Hierbij wordt een driedeling gemaakt in normeringssystematiek: 'geringe', 'meer dan geringe' en 'zeer ingrijpende inbreuk'. Respectievelijk zou dat onder de bevoegdheid van de opsporingsambtenaar, OvJ en RC vallen.[XVI] Omdat PoLF-databasen alleen toegankelijk zijn na het maken van een forensisch kopie, zal altijd toestemming gevraagd moeten worden aan de OvJ.[XVII] Zoals blijkt uit 4.1.1 beschikt een iPhone uit genoeg databasen die kunnen zorgen voor een inbreuk op de persoonlijke levenssfeer. Valt PoLF-analyse dan onder 'meer dan geringe' of 'zeer ingrijpende inbreuk'?

De Commissie-Koops is daar niet echt duidelijk over: "de normering in het gemoderniseerde wetboek zal moeten werken met (tamelijk) abstracte criteria, die van geval tot geval geïnterpreteerd moeten en kunnen worden."[XVIII] Volgens de commissie is sprake van een ingrijpend beeld van iemands privéleven, als een wezenlijk of aanzienlijk beeld tot stand komt.[XIX] Voor PoLF is niet een direct een link te leggen met dit criterium, maar de commissie geeft wel een andere theorie om dit te toetsen: "De mozaïektheorie komt er kort gezegd op neer dat, voor de beoordeling van de mate van een privacyinbreuk, niet moet worden gekeken naar losse steentjes, maar naar het beeld dat ontstaat als je de nodige steentjes bij elkaar legt."[XX] Aan de hand van deze theorie is te concluderen dat PoLF onder een 'zeer ingrijpende inbreuk' valt. In de Memorie van Toelichting komt de mozaïektheorie niet terug. De wetgever laat aan praktijk over wanneer wel of niet sprake is van 'zeer ingrijpende inbreuk'. Het is zelfs niet de bedoeling om veel machtigingen aan de RC te vragen.[XXI] De machtiging van een OvJ acht de wetgever dus in veel gevallen voldoende, om onderzoek uit te voeren aan een digitale gegevensdrager. Oerlemans is het hier niet mee eens. Hij stelt dat vooral de jongere generatie onderzoek aan een smartphone gezien kan worden als zeer privacy-intrusief. Daarom is volgens hem een machtiging van een RC vereist.[50]

Resumé

Uit het arrest van de Bûterwei-zaak blijkt dat digitale sporen een belangrijke rol hebben gespeeld in de bewijsvoering van het hof. Een zeer relevante opmerking van het hof was de bepaling van de betrouwbaarheid van de locaties. De Google-data geeft een indicatie van de locatie van de telefoon en geen absolute zekerheid. Zowel de politie als telecomdeskundige Pluijmers kwamen op exact dezelfde locaties uit aan de hand van de ruwe data. Het toetsen van de verklaring van de verdachte aan de hand van digitale data speelde ook een rol in de reconstructie van de middag en avond van het delict. Verder blijkt uit het arrest dat verdachte de telefoon heeft uitgezet. De verdachte verklaart dat de telefoon batterijproblemen had. Uit analyse van de politie en Pluijmers wordt deze verklaring niet waarschijnlijk geacht. Aan de hand van loggegevens en

batterijgebruik bleek dat de iPhone niet door een te lage accu was uitgeschakeld. In juridische zin zou PoLF gebruikt kunnen worden in de rechtszaal.

Om de deelvraag kort te beantwoorden, digitale sporen kunnen potentieel gebruikt worden om bepaalde scenario-elementen te toetsen. Potentie betekent per definitie niet dat PoLF-artefacten standaard onderzocht mogen worden volgens de wet. In de huidige wetgeving mogen opsporingsambtenaren gelimiteerd onderzoek uitvoeren aan een digitale gegevensdrager, maar dat wil de wetgever veranderen. Met het oog op de modernisering van het Wetboek van Strafvordering is getoetst onder welke bevoegdheid PoLF-onderzoek valt. Uit de memorie van toelichting is te lezen dat het onder de bevoegdheid van de OvJ valt. Echter, Oerlemans is van mening dat het doorzoeken van een telefoon een ernstige inbreuk is op de privésfeer van de burger. Daarom zou het moeten vallen onder de RC. De wetgever laat aan de praktijk over hoe precies deze scheiding wordt bepaald.

4.3 Versterking DFO en FO

Casey et al. benoemen in meerdere artikelen dat DFO en FO meer met elkaar zouden moeten samenwerken.[17,70,71] DFO zou hierbij een rigide raamwerk van FO kunnen adopteren. Hij draagt meerdere oplossingen aan om te zorgen dat deze samenwerking kan plaatsvinden. Een van deze aanbevelingen is de politie te laten investeren om niet-wetenschappers digitale sporen uit Hansken te laten begrijpen.[37] De vraag is of de politiepraktijk hier klaar voor is en welke rol Hansken hier in kan spelen.

4.4 Kanttekeningen bij experimenteren in het lab

In de vorige sectie is uitvoerig besproken welke data meer informatie kan geven over mogelijke handelingen van de gebruiker. Deze analyses zijn uitgevoerd onder lab-omstandigheden, omdat vooraf bekend was wat de resultaten zouden moeten zijn. Vandaar dat in praktijk resultaten kritischer geïnterpreteerd moeten worden. Zo is een vergelijking gemaakt met een Axiom, UFED Reader en APOLLO.

Het eerste wat opviel is de verwerking van de *healthdb_secure*. UFED laat alleen het aantal stappensessies per uur zien. Zowel Magnet Axiom als APOLLO geeft de stappen en afstand als aparte artefact weer. Met de kennis dat stappen accurater zijn dan afstanden kan dit tot een verkeerde interpretatie leiden van de rechercheur.[29] Daar schuilt een gevaar. Politiemensen denken dat digitale sporen 'niet liegen'. Hierdoor zijn ze betrouwbaarder en objectiever dan analoge sporen.[6] Casey waarschuwt dat deze misinterpretatie onbewust kan optreden. De rechercheur schrijft op wat hij waarneemt.[72] Het uitvoeren van *dual-tool verification* is dus niet mogelijk met UFED in combinatie met Axiom, als rechercheurs het aantal stappen willen

vergelijken met de *artefact view* in UFED PA en Axiom. Deze bewustwording moet duidelijk overgebracht worden aan rechercheurs die met digitaal bewijs omgaan. Hoewel een analyseprogramma data op een specifieke manier weergeeft, betekent per definitie niet dat dit ook waarheid is. De rechercheur moet kritisch zijn of de getoonde data consistent is met andere digitale sporen.

Een ander punt is de benodigde expertise voor het gebruik van de tools. De meeste rechercheurs gebruiken tools zoals UFED, Axiom of Hansken om hun analyse uit te voeren. Een groot gedeelte van de analyse wordt door deze programma's gedaan. De resultaten uit de vorige secties zouden met behulp van APOLLO gereproduceerd kunnen worden door een rechercheur met basiskennis Python. Het script wordt aangeroepen met de iPhone als input. De resultaten kunnen daarna doorgenomen worden door de rechercheur. Rechercheurs kunnen (onbewust) conclusies trekken, die soms interpretatie stap vragen van een expert. Dat kan het geval zijn met UTC-tijden, waarbij het tijdsverschil meegenomen moet worden in de analyse.[61] Een rechercheur zal voorzichtig moeten zijn in het trekken van conclusies aan de hand van digitale sporen met betrekking tot tijden.

Echter, de analyse van de data vindt plaats met behulp van SQL. Simpel gezegd, SQL-commando's worden uitgevoerd op sqlite-databasen van de iPhone. Sommige commando's zijn vrij eenvoudig te begrijpen, terwijl andere heel complex zijn. In geval van de *interactionC.db* is dit een commando van 45 regels. Daarbij komen de verschillende soorten 'timestamps' of 'time offsets', die gebruikt worden in database. Deze tijden moeten geïnterpreteerd worden door een expert om ze op een correcte wijze in een tijdlijn te plaatsen. Als laatste wordt een gedeelte van de data weergegeven als twee tabellen met elkaar gecombineerd worden. Een voorbeeld hiervan is het gebruik van een 'left join'-argument in het SQL-commando. Van de twee tabellen wordt een gedeelte van weergegeven, waardoor ook data gemist wordt. Een expert zal al de hiervoor benoemde punten moeten meenemen om tot een conclusie te komen.

Belangrijkste advies is het veiligstellen van een iPhone binnen zes dagen. Cachebestanden werken volgens het 'first-in-first-out'-principe. Het woord *cache* geeft in computertermen aan dat deze bestanden tijdelijk zijn. Voor de opsporing is het wenselijk om zoveel mogelijk informatie te verzamelen. Bij het maken van keuzes dient dus rekening gehouden te worden dat cachebestanden op iPhones na zes dagen worden overschreven. Indien dit niet mogelijk is, dan kan terug worden gevallen op *knowledgeC* met vier weken aan beschikbare data. Het verschil zit vooral in dat *cache* veel gedetailleerder bijhoudt welke handelingen zijn uitgevoerd.

Hansken in de praktijk

Het gebruik van Hansken in combinatie met PoLF biedt veel mogelijkheden voor de opsporing.

In 4.1.1 zijn een paar voorbeelden gegeven hoe PoLF in de praktijk gebruikt kan worden. Deze lijst is in werkelijkheid veel langer, dan beschreven kan worden in dit onderzoek. Of informatie van pas komt tijdens een opsporingsonderzoek zal nog moeten blijken. Als rechercheurs niet weten dat dit soort informatie beschikbaar is op iPhones, dan is het niet raar dat FO digitale sporen laat liggen. Want hoewel Hansken meer voor tactische en digitale rechercheurs is gebouwd, kan de forensische rechercheur wel degelijk profijt hebben van PoLF-artefacten. Een tijdstip van overlijden bepalen van een slachtoffer.[58] Of onderzoeken of verdachte en slachtoffer gezamenlijk wandelden, kunnen rechercheurs helpen in het opstellen van scenario's.[31] Het zou zelfs mogelijk zijn om een speciale externe module te bouwen in Hansken voor FO, zodat ze dit soort vragen kunnen beantwoorden zonder extra informatie van andere modules.

Om PoLF effectief te kunnen inzetten in Hansken zal eerst gekeken moeten worden hoe betrouwbaar en valide de APOLLO-scripts zijn. Het zijn vrij eenvoudige scripts, dus voor onderzoekers is het makkelijk te controleren of het werkt. Deze stap zal niet betrouwbaarheid van de resultaten doen toenemen. De extra controle zorgt ervoor dat PoLF-scripts beter worden gevalideerd en een kwaliteitssysteem wordt ontwikkeld voor dit soort sporen. Als eenmaal alle scripts op betrouwbaarheid zijn getest, dan kunnen ze geïmplementeerd worden in Hansken. Aan te raden is om dit middels de Hansken Extraction Plugin (EP) te doen, omdat dit vele malen sneller is dan de Hansken.py. Hansken EP is vergelijkbaar met de Hansken.py qua functionaliteit. De kracht van Hansken EP is de mogelijkheid om de analyse tijdens de extractie uit te voeren. Hansken.py kan alleen gebruikt worden als de *image* is verwerkt door Hansken.[73]

Uiteindelijk zullen de rechercheurs PoLF-artefacten ook zelf moeten gebruiken in hun onderzoeken. Een Hansken EP maken voor PoLF is niet struikelblok in het proces. Zoals blijkt in 4.1.3 staat centraal wie de interpretatie op PoLF-artefacten gaat uitvoeren. Mag een digitaal rechercheur uitspraken doen over hartslagmetingen of valt dit onder bio-informaticus van het NFI? En in hoeverre is sprake van expertise in PoLF? Verder blijkt uit 4.1.2 dat sinds de eerste successen rond 2018 dat PoLF niet vaak ingezet wordt in onderzoeken. Deels is dit te wijten aan onwetendheid wat digitaal allemaal mogelijk is, maar omdat tussen DFO en FO weinig samenwerking is. Meer onderzoek, het delen van kennis en bewustwording hoe PoLF ingezet kan worden, zal daarom het succes bepalen van PoLF in Hansken met betrekking tot scenariogericht opsporen.

Resumé

Als laatste deelvraag wordt beantwoord: "Op welke manier kunnen digitaal en forensisch onderzoek elkaar versterken?". Hiervoor is in de praktijk getest hoe deze analyses uitgevoerd kunnen worden. Drie belangrijke resultaten kwamen naar voren. Een rechercheur moet kritisch zijn op de gepresenteerde data. Niet elk analyseprogramma laat de resultaten op dezelfde

manier zien. Het tweede punt is de interpretatie van de data. Een rechercheur zal zich bewust moeten zijn welke conclusie hij verbindt aan de gevonden resultaten. Het laatste punt is het veiligstellen van iPhones bij verdachten. Om een goede reconstructie te kunnen uitvoeren, is het gewenst om zoveel mogelijk data te verzamelen. Aangezien *cache*-bestanden zes dagen bewaard blijven en *knowlegdeC* vier weken beschikbaar blijft, kan dit invloed hebben op de tijd wanneer een aanhouding of huiszoeking zal moeten plaatsvinden.

Hansken kan voor vele doeleinden ingezet worden. Om dit te kunnen laten slagen, zal wel meer onderzoek nodig zijn om de resultaten te controleren en valideren. Onderzoek naar PoLF zou de eerste stap kunnen zijn, omdat dit vrij eenvoudige scripts zijn. Vooral FO kan dit soort sporen gebruiken in hun scenario's. De volgende stap is het omzetten van Hansken.py naar Hansken EP, om gebruik te maken snelheid van het systeem. Indien aan deze voorwaarden voldaan is, kunnen rechercheurs dit niet meteen in de praktijk gebruiken. De rechercheur moet immers genoeg expertise hebben om uitspraken te kunnen doen over PoLF-analyses. En hoewel successen bekend zijn, zal meer geïnvesteerd moeten worden in onderzoek, het delen van kennis en bewustwording.

5 Conclusie

Digitale sporen gaan steeds meer een rol spelen in opsporingsonderzoeken. Met behulp van Hansken.py kunnen rechercheurs met programmeerervaring zelf aan de slag met de analyse van digitale data uit Hansken. Ze kunnen zelf scripts schrijven om nieuwe digitale sporen uit bestanden te halen. Forensische Opsporing (FO) zou hier gebruik van kunnen maken door te focussen op Pattern-of-Life Forensics (PoLF). Hiervoor is een testomgeving met Hansken All-In-One (Hansken AIO) en Jupyterlab gecreëerd, om te bepalen wat de toegevoegde waarde is voor de opsporingspraktijk. Het doel is om te onderzoeken of de combinatie PoLF met Hansken mogelijk is. De hoofdvraag die hieruit volgt, is: “In hoeverre kan Pattern-of-Life Forensics in combinatie met Hansken bijdragen aan scenariogericht onderzoek?”

Hansken wordt steeds meer gebruikt in opsporingsonderzoeken in Nederland als digitaal analyseplatform. Hiermee kunnen opsporingsdiensten op een forensische wijze digitale data indexeren en analyseren. Hansken moet zich in de komende jaren verder blijven ontwikkelen. Een van de ontwikkelingen die mogelijk interessant is voor FO is PoLF. Dit onderzoeksgebied richt zich op specifieke handelingen of activiteiten van gebruiker: telefoon aan-/uitzetten, batterijverbruik en/of stappenteller. Een open-source-programma dat speciaal voor de iPhone ontwikkeld is, is Apple Pattern of Life Lazy Output’er (APOLLO) van Sarah Edwards.

PoLF-sporen kunnen bijdragen aan de reconstructies, zoals blijkt in de Bûterwei-zaak. Onder andere kunnen *knowledgeC*, *interactionC*, *healthdb_secure* en cache-bestanden gebruikt worden, om handelingen van de gebruiker aan te tonen. In combinatie van deze digitale sporen zou opzet kunnen aantonen. Een stapje verder is het identificeren van de gebruiker via biometrische gegevens. Zo kan TouchID uitsluitsel geven wie de iPhone ontgrendeld heeft. Een stappenteller in combinatie met de hartslagmeter geeft inzicht in de levensstijl van de gebruiker. Aan de hand hiervan kan een beter beeld verkregen worden van de eigenaar van de iPhone. Deze gegevens moeten wel betrouwbaar en controleerbaar zijn. Hoewel *dual-tool verification* wordt aangeraden, blijkt in praktijk dat niet alle data op de juiste manier wordt weergegeven door analysesoftware. Manueel doorzoeken kost veel tijd. Vandaar dat het makkelijker is om de verklaring van gebruiker te combineren met de gevonden digitale sporen. Deze methode is vergelijkbaar met de CAI-model.

PoLF-onderzoek in combinatie met Hansken is gezien de huidige wetgeving lastig uit te voeren. Wettelijk gezien mogen opsporingsambtenaren weinig handelingen uitvoeren aan een digitale gegevensdrager. Om een goede analyse uit te voeren, is het wel nodig om zoveel mogelijk data

te verzamelen. Het kan zowel belastend als ontlastend zijn voor de verdachte. Met het oog op de modernisering van Strafvordering is wel meer ruimte voor dit soort onderzoeken. Hoewel PoLF-onderzoek als zeer ingrijpend kan worden ervaren, is de wetgever (in concept) van mening dat een machtiging van een officier van justitie voldoende is. De praktijk zal moeten uitwijzen of deze bevoegdheid onder de officier van justitie en zelfs de rechter-commissaris valt.

Als een rechercheur PoLF met Hansken mag gebruiken, dan zal hij wel met een paar punten rekening moeten houden. Zo moet een rechercheur een kritische blik hebben op de gepresenteerde data en voorzichtig zijn in het trekken van conclusies. Hierbij kan een digitaal rechercheur als expert optreden. Verder zal een onderzoeksteam op de hoogte moeten zijn dat digitale informatie overschreven wordt bijvoorbeeld bij *knowledgeC* en cache-bestanden. Voor een eventuele aanhouding of huisdoorzoeking kan dit van belang zijn.

DFO zal zich in de komende jaren verder moeten ontwikkelen als forensische discipline. Met Hansken en Hansken.py staan rechercheurs aan de vooravond, om veel gerichter digitaal onderzoek uit te voeren. Een testomgeving met Hansken AIO en Jupyterlab biedt onderzoekers een platform om met digitale sporen te experimenteren. FO kan hierop meeliften en zelf een bijdrage vervullen. PoLF is een onderzoeksgebied, wat snijdt tussen DFO en FO. In specifieke gevallen kunnen ze elkaar zelfs versterken, om de betrouwbaarheid te waarborgen en specifieke scenario elementen te toetsen.

6 Aanbevelingen

Uit dit onderzoek blijkt dat in grote lijnen de aanbevelingen hetzelfde zijn als bij Van Zandwijk en Boztas. Onder andere voorzichtig zijn met het trekken van sterke conclusies uit de interpretatie van de data. Of meer onderzoek doen naar combineren van de artefacten uit de verschillende databasen komen overeen met resultaten uit dit onderzoek.[59] Om het wat breder te trekken, kan juist de combinatie van Hansken en PoLF een effectieve oplossing zijn.

Onderzoek naar de betrouwbaarheid en validiteit van digitale sporen zal als eerst uitgevoerd moeten worden voor FO met PoLF aan de slag kan. Dit document heeft een eerste opzet gegeven om te kunnen experimenteren met digitale sporen met behulp van Hansken AIO en Jupyterlab. Zowel digitale rechercheurs als studenten kunnen hiermee zelf bepaalde digitale sporen toetsen. In 4.1.1 is slechts een klein aantal PoLF-artefacten besproken, die beschikbaar zijn via APOLLO. Hier liggen kansen om de andere scripts ook toe te passen. Met name meer onderzoek naar de ontgrendelingsmechanisme van de telefoon kan uitgebreider worden onderzocht. Een ander onderdeel is meer onderzoek naar de cache-bestanden. Een voorbeeld hiervan is *CurrentPowerlog*, die veel gedetailleerder data bijhoudt dan *knowlegdeC*.

Inmiddels kunnen deze experimenten vrij eenvoudig worden uitgevoerd. Het NFI heeft een analyseprogramma dat bijhoudt welke bestanden gewijzigd zijn en een snapshot maakt van deze bestanden.[59] Voor experimenteel onderzoek is dit een oplossing, om een bepaald scenario element te toetsen. Afhankelijk van het PoLF-artefact zal wel bepaald moeten worden wat voor soort statische analyse uitgevoerd moet worden. Dit is vergelijkbaar met de keuze om een correlatiecoëfficiënt of relatieve entropie te gebruiken bij de stappenteller.

Als deze kennis beschikbaar is en gevalideerd, begint de volgende stap om het te delen met de ketenpartners van het NFI. De Hansken Academy heeft hierbij een belangrijke rol om digitale rechercheurs op de hoogte te stellen van PoLF. Bijvoorbeeld hoe zij met behulp van Hansken gebruik kunnen maken van PoLF, om digitale sporen te duiden (opzet, eigenaarschap en betrouwbaarheid). Hoewel de kennis technisch aanwezig is, zal ook extra geïnvesteerd moeten worden in de bewustwording wat digitaal mogelijk is. Het is niet vanzelfsprekend dat de digitale, forensische, tactische rechercheurs en analisten meteen met elkaar Hansken gebruiken. Iedereen binnen de keten zal basiskennis moeten hebben om digitale kansen te zien. Deze bewustwording is vooral voor de tactische en forensische rechercheur. Zij hebben de contextinformatie die nodig is om bepaalde scenario's te verifiëren of falsificeren.

Op korte termijn bestaat het gevaar dat een niet-digitale rechercheur zelf PoLF-analyses uitvoert.

Uit de resultaten blijkt dat interpretatie van PoLF enige vorm van expertise vereist. In plaats van de taak van PoLF neer te leggen bij de tactische of forensische rechercheur is advies om gezamenlijk onderzoeksvragen te bepalen met DFO. Denk daarbij aan de drie eenheid van spelers binnen DFaaS-model.[2] Ieder met zijn eigen expertise kan bijdrage leveren. Dat kan in de vorm van een brainstormsessie, die moet leiden tot een advies richting de leiding van het onderzoek. Eventuele behaalde successen of mislukkingen kunnen weer gedeeld worden binnen de Hansken community. Precies waarvoor het platform voor bedoeld is: samenwerken en delen van kennis. Alleen dan kan *forensic intelligence* verder ontwikkeld worden.

De laatste aanbeveling heeft betrekking tot de uitvoering van DFO. Een smartphone is een uiterst persoonlijk gegevensdrager. Daar komt bij dat een smartphone steeds meer data bevat. Het is bijna onmogelijk om een test-telefoon voor te bereiden, die overeenkomt met de dagelijkse handelingen van een persoon. Voor het trainen en onderzoeken van PoLF zou daarom zoveel mogelijk gewerkt moeten worden met datasets uit de praktijk. Voor eventuele trainingen en vervolgonderzoek is de voorkeur om te kiezen voor echte data uit de praktijk.

Bibliografie

[1] Driessen C, Meeus J. Advocaten strijden tegen informatie uit 'cryptophones' [Internet];. Beschikbaar via : https://www.nrc.nl/nieuws/2021/03/15/advocaten-strijden-tegen-informatie-uit-cryptophones-a4035704 [Geraapleegd 7 juni 2021].

[2] Van Beek H, Van den Bos J, Boztas A, Van Eijk E, Schramp R, Ugen M. Digital forensics as a service: Stepping up the game. Forensic Science International: Digital Investigation. 2020;35:301021.

[3] Van der Veen E. 'De forensische opsporing heeft nog veel meer potentie'. Blauw. 2021 3:53–54.

[4] Van Baar R, Van Beek H, Van Eijk E. Digital Forensics as a Service: A game changer. Digital Investigation. 2014;11:S54–S62.

[5] Van Beek H, Van Eijk E, Van Baar R, Ugen M, Bodde J, Siemelink A. Digital forensics as a service: Game on. Digital Investigation. 2015;15:20–38.

[6] Zuurveen R, Stol W. Benutten van Digitale Sporen. Politiekunde; 2020.

[7] Miller C. Forensic Pattern Of Life Analysis [Internet];. Beschikbaar via : https://www.forensicfocus.com/articles/forensic-pattern-of-life-analysis/ [Geraapleegd 7 juni 2021].

[8] Zuurveen R, Van Valkengoed T, Veenstra S, Stol W. Kennis voor politiewerk in een digitale samenleving. Cybersafety Research Group; 2020.

[9] Henseler H, De Poot C. De betekenis van digitale sporen voor bewijs op activiteitniveau. Expertise en Recht. 2020;2:50–59.

[10] Bommisetty S, Tamma R, Mahalik H. Practical mobile forensics. Packt Publishing Ltd; 2014.

[11] Humphries G, Nordvik R, Manifavas H, Cobley P, Sorell M. Law Enforcement educational challenges for mobile forensics. Digital Investigation. 2021.

[12] mac4n6. APOLLO [Internet];. Beschikbaar via : https://github.com/mac4n6/APOLLO [Geraapleegd 7 juni 2021].

[13] mac4n6. mac4n6 blog [Internet];. Beschikbaar via : http://www.mac4n6.com/ [Geraapleegd 7 juni 2021].

[14] Alink W, Bhoedjang R, Boncz P, De Vries A. XIRAF–XML-based indexing and querying for digital forensics. digital investigation. 2006;3:50–58.

[15] Bhoedjang R, Van Ballegooij A, Van Beek H, Van Schie J, Dillema F, et al. Engineering an online computer forensic service. Digital Investigation. 2012;9(2):96–108.

[16] Henseler H. Het inzagerecht en de groeiende omvang van digitaal bewijs. Expertise en Recht. 2020;6:215–217.

[17] Casey E. Strengthening trust: Integration of digital investigation and forensic science. Elsevier; 2020.

[18] Cook R, Evett I, Jackson G, Jones P, Lambert J. A model for case assessment and interpretation. Science and Justice. 1998;38(3):151–156.

[19] De Ronde A, Van Aken M, De Puit M, De Poot C. A study into fingermarks at activity level on pillowcases. Forensic science international. 2019;295:113–120.

[20] De Ronde A, Kokshoorn B, De Poot C, De Puit M. The evaluation of fingermarks given activity level propositions. Forensic science international. 2019;302:109904.

[21] Flynn M, Juergens R, Cantrell T. Employing ISR SOF best practices. National Defense Univ Washington DC Inst for National Strategic Studies; 2008.

[22] Biltgen P, Ryan S. Activity-Based Intelligence: Principles and Applications. Artech House; 2016.

[23] mac4n6. Presentations [Internet];. Beschikbaar via : https://github.com/mac4n6/Presentations [Geraapleegd 7 juni 2021].

[24] DFRWS presentations and Brignoni, A. iLEAPP & ALEAPP: Parse and validate mobile forensic artifacts with Python [Internet];. Beschikbaar via : http://dfrws.org/presentation/ileapp-aleapp-parse-and-validate-mobile-forensic-artifacts-with-python/ [Geraapleegd 7 juni 2021].

[25] Brignoni A. iLEAPP [Internet];. Beschikbaar via : https://github.com/abrignoni/iLEAPP [Geraapleegd 7 juni 2021].

[26] SQLite Consortium. What Is SQLite [Internet];. Beschikbaar via : https://sqlite.org/index.html [Geraapleegd 7 juni 2021].

[27] Shimmi S, Dorai G, Karabiyik U, Aggarwal S. Analysis of iOS SQLite schema evolution for updating forensic data extraction tools. In: 2020 8th International Symposium on Digital Forensics and Security (ISDFS). IEEE; 2020. p. 1–7.

[28] Misja com. Cocoa Core Data Timestamp Converter [Internet];. Beschikbaar via : https://www.epochconverter.com/coredata [Geraapleegd 7 juni 2021].

[29] Van Zandwijk J, Boztas A. The iPhone Health App from a forensic perspective: can steps and distances registered during walking and running be used as digital evidence? Digital Investigation. 2019;28:S126–S133.

[30] Bosma W, Dalm S, Van Eijk E, El Harchaoui R, Rijgersberg E, Tops H, et al. Establishing phone-pair co-usage by comparing mobility patterns. Science & Justice. 2020;60(2):180–190.

[31] Jennings L, Sorell M. identifying Patterns and activities froM iPhone and aPPle watch steP-count data for use in a digital investigation. In: 5th interdisciPlinary cyber research conference 2019; 2019. p. 78.

[32] Koekkoek A. De Grondwet: een systematisch en artikelsgewijs commentaar. WEJ Tjeenk Willink; 2000.

[33] Lassche H. Digitalisering en de opsporingspraktijk. Politieacademie; 2019.

[34] Reporters T. Police and prosecution lawyers fail to correctly disclose evidence in nearly half of cases, watchdog says [Internet];. Beschikbaar via : https://www.telegraph.co.uk/news/2018/07/12/policeand-prosecution-lawyers-fail-correctly-disclose-evidence/ [Geraapleegd 7 juni 2021].

[35] Seyyar M, Geradts Z. Privacy impact assessment in large-scale digital forensic investigations. Forensic Science International: Digital Investigation. 2020:200906.

[36] Epskamp-Dudink C. Niet te filmen! Over retrospectief scenariodenken in de opsporingspraktijk, Lectoraat Intelligence. 2016.

[37] Casey E. The chequered past and risky future of digital forensics. Australian Journal of Forensic Sciences. 2019;51(6):649–664.

[38] Casey E, Jaquet-Chiffelle D, Spichiger H, Ryser E, Souvignet T. Structuring the Evaluation of Location-Related Mobile Device Evidence. Forensic Science International: Digital Investigation. 2020;32:300928.

[39] Epskamp-Dudink C, Winter J. Benefits of scenario reconstruction in cold case investigations. Journal of Criminal Psychology. 2020.

[40] Sunde N, Dror I. Cognitive and human factors in digital forensics: Problems, challenges, and the way forward. Digital Investigation. 2019;29:101–108.

[41] Ribaux O, Margot P. Case based reasoning in criminal intelligence using forensic case data. Science & Justice. 2003;43(3):135–143.

[42] Ribaux O, Wright B. Expanding forensic science through forensic intelligence. Science & justice. 2014;54(6):494–501.

[43] Kop N, Klerks P. Intelligencegestuurd Politiewerk. Politieacademie; 2009.

[44] Meconi T. A future proof FOIT-function; 2020. Stageverslag bij Magnet Forensics.

[45] Corpora D. downloads.digitalcorpora.org S3 Browser [Internet];. Beschikbaar via : https://downloads.digitalcorpora.org/corpora/mobile/ios_13_4_1/ [Geraapleegd 7 juni 2021].

[46] DFRWS. DFRWS EU 2021; 2021. Conferentie rondom DFO.

[47] Lucy D. Introduction to statistics for forensic scientists. John Wiley & Sons; 2013.

[48] Enschedé C, Blom T. Beginselen van strafrecht: een syllabus. Kluwer; 1987.

[49] mac4n6. Knowledge is Power! Using the macOS/iOS knowledgeC.db Database to Determine Precise User and Application Usage [Internet];. Beschikbaar via : https://www.mac4n6.com/blog/2018/8/5/knowledge-is-power-using-the-knowledgecdb-database-on-macos-and-ios-to-determine-precise-user-and-application-usage [Geraapleegd 7 juni 2021].

[50] Oerlemans J. Beschouwing rapport Commissie-Koops: strafvordering in het digitale tijdperk. Tijdschrift Modernisering Strafvordering. 2018;2018(2):92–105.

[51] Hermesdorf L. A Methodology for Verification Testing of Data Evidence in Mobile Forensics; 2020.

[52] mac4n6. Socially Distant but Still Interacting! New and Improved Updates to macOS/iOS CoreDuet interactionC.db APOLLO Modules [Internet];. Beschikbaar via : https://www.mac4n6.com/blog/2020/6/21/socially-distant-but-still-interacting-new-and-improved-updates-to-macosios-coreduet-interactioncdb-apollo-modules [Geraapleegd 7 juni 2021].

[53] Doekhie G. Schuldig of onschuldig? Digitale sporen in bewijsvoering.; 2021. Conferentie presentatie, Hogeschool Leiden E-discovery Symposium https://www.hsleiden.nl/digital-forensics/agenda/symposium/e-discovery-2021.

[54] Apple Inc . Keybags for Data Protection [Internet];. Beschikbaar via : https://support.apple.com/nl-nl/guide/security/sec6483d5760/web [Geraapleegd 7 juni 2021].

[55] Wang C, Wang Y, Chen Y, Liu H, Liu J. User authentication on mobile devices: Approaches, threats and trends. Computer Networks. 2020;170:107118.

[56] mac4n6. Pincodes, Passcodes, & TouchID on iOS - An Introduction to the Aggregate Dictionary Database (ADDataStore.sqlite) [Internet];. Beschikbaar via : https://www.mac4n6.com/blog/2017/3/12/introduction-to-the-aggregate-dictionary-database-addatastoresqlite [Geraapleegd 7 juni 2021].

[57] Hamilton D. Touch ID Trick: Train Multiple Fingerprints at Once [Internet];. Beschikbaar via : https://www.macobserver.com/tips/how-to/touch-id-train-multiple-fingerprints/ [Geraapleegd 7 juni 2021].

[58] MacMullen T. How an Apple Watch Could Decide a Murder Case [Internet];. Beschikbaar via : https://medium.com/s/story/how-an-apple-watch-could-decide-a-murder-case-94314c8d95a2 [Geraapleegd 7 juni 2021].

[59] Van Zandwijk J, Boztas A. The phone reveals your motion: Digital traces of walking, driving and other movements on iPhones. Forensic Science International: Digital Investigation. 2021;37:301170.

[60] Horsman G. Tool testing and reliability issues in the field of digital forensics. Digital Investigation. 2019;28:163–175.

[61] Boyd C, Forster P. Time and date issues in forensic computing—a case study. Digital Investigation. 2004;1(1):18–23.

[62] Horsman G. Opinion: Does the field of digital forensics have a consistency problem? Forensic Science International: Digital Investigation. 2020:300970.

[63] Page H, Horsman G, Sarna A, Foster J. A review of quality procedures in the UK forensic sciences: What can the field of digital forensics learn? Science & Justice. 2019;59(1):83–92.

[64] Årnes A. Digital forensics. John Wiley & Sons; 2017.

[65] SWGDRUG. Scientific Working Group for the Analysis of Seized Drugs (SWGDRUG) recommendations. United States Department of Justice Drug Enforcement Administration . . . ; 2019.

[66] Williams J. ACPO good practice guide for digital evidence. Metropolitan Police Service, Association of chief police officers, GB. 2012.

[67] Casey E. Trust in digital evidence. Elsevier; 2019.

[68] Stander A. Digital Forensics education in a fast changing world; 2021. Conferentie presentatie, Hogeschool Leiden E-discovery Symposium https://www.hsleiden.nl/digital-forensics/agenda/symposium/e-discovery-2021.

[69] Modderkolk, H. Hoe Google-data in een moordzaak leidden naar de echtgenote [Internet];. Beschikbaar via : https://www.volkskrant.nl/nieuws-achtergrond/hoe-google-data-in-een-moordzaak-leidden-naar-de-echtgenote~b092755e/ [Geraapleegd 7 juni 2021].

[70] Casey E, Zehnder A. Inter-regional digital forensic knowledge management: needs, challenges, and solutions. Journal of Forensic Sciences. 2020.

[71] Casey E. Standardization of forming and expressing preliminary evaluative opinions on digital evidence. Forensic Science International: Digital Investigation. 2020;32:200888.

[72] Casey E. The epic story of scientific interpretation in digital investigations. Elsevier; 2020.

[73] Hansken Community Dagen 2. Hansken Seminar; 2021. Conferentie presentatie in maart 2021.

Juridische lijst

[I]Rechtbank Amsterdam 19 april 2018, ECLI:NL:RBAMS:2018:2504 en Rechtbank Gelderland 26 juni 2019, ECLI:NL:RBGEL:2019:2832

[II]Recht op een eerlijk proces is opgenomen in art. 6 EVRM.

[III]10 lid 1 Gw.

[IV]Art. 8 EVRM;onder lid 1 valt ook 'zijn woning en zijn correspondentie'. In lid 2 wordt ingegaan in welke situaties wel inbreuk gepleegd mag worden op de persoonlijke levenssfeer van de burger.

[V]Hof Arnhem-Leeuwarden 22 april 2015, ECLI:NL:GHARL:2015:2954

[VI]HR 4 april 2017, ELCI:NL:HR:2017:584

[VII] Rechtbank Noord-Nederland 11-07-2019, ELCI:NL:GHARL:2020:9865

[VIII]HR 10 juli 2018, ECLI:NL:HR:2018:1121

[IX]HR 9 juli 2019, ECLI:NL:HR:2019:1079

[X]HR 9 februari 2021, ECLI:NL:HR:2021:202

[XI]Hof Arnhem-Leeuwarden 1 december 2020, ELCI:NL:GHARL:2020:9865

[XII]Commissie implementatie nieuw Wetboek van Strafvordering (Commissie-Letschert), p. 2

[XIII]Commissie modernisering opsporingsonderzoek in het digitale tijdperk (Commissie-Koops), p. 7

[XIV]Commissie-Koops, p. 104-108

[XV]Ambtelijke versie juli 2020 Memorie van Toelichting Wetboek van Strafvordering (Memorie van Toelichting van concept Sv), p. 424-426

[XVI]Commissie-Koops, p. 165-168

[XVII]Memorie van Toelichting van concept Sv, p. 412-413

[XVIII]Commissie-Koops, p. 36

[XIX]Commissie-Koops, p. 39-41

[XX]Commissie-Koops, p. 40

[XXI]Memorie van Toelichting van concept Sv, p. 414

Appendices

A Afkortingen

B Gesprekken

B.1 Tom

Tom is docent digitaal onderzoek aan de PA. In een gesprek met hem stond centraal: Wat komt een gemiddelde rechercheur tegen op gebied van DFO? Aan de hand van een paar praktijkvoorbeelden kwamen een paar vraagstukken naar voren. Een veelvoorkomend vraagstuk is of sprake is van opzet. Hiermee wordt bedoeld dat een verdachte opzettelijk een bepaalde handeling heeft uitgevoerd. Een mogelijke strategie is om artefacten rondom het tijdstip van de handeling te onderzoeken. Dat vraagt wel van de rechercheur dat deze in staat is om de verschillende artefacten erom heen te herkennen en te destilleren. Wel moet de opmerking gemaakt worden dat digitale sporen gemanipuleerd kunnen worden. Dit is in mindere mate het geval bij biologische/fysische sporen.

Vaak wordt nog teruggegrepen op de 'whodunnit'-mentaliteit. Dit kan leiden tot tunnelvisie binnen DFO. Leidend zou moeten zijn om de 7 gouden W's te beantwoorden in een onderzoek. Als het goed is, zou bij de beantwoording van deze vragen genoeg informatie beschikbaar moeten zijn voor een reconstructie. Dit levert uiteindelijk een totaalplaatje op.

Sommige rechercheurs weten niet welke digitale sporen ze achterlaten in het dagelijks leven. In de eerste instantie denken ze dat ze geen digitale sporen achterlaten. Met wat meer doorvragen komen ze er uiteindelijk achter dat dit wel het geval is. Het kunnen herkennen van digitale kansen begint echter bij de bewustwording van de digitale wereld.

Uiteindelijk zijn er drie punten die meerdere malen terugkomen tijdens het interview: is er sprake van opzet, is verdachte alleen gebruiker en hoe betrouwbaar is de data? Vooral de eerste twee vragen zijn van belang voor de tactische rechercheur. Sprake van opzet is nodig voor een verdenking. Om juist de verdachte als gebruiker aan te wijzen moeten digitale sporen ondersteunend zijn voor het tactische werk.

B.2 Matthew Sorell

In het gesprek met Matthew Sorell is besproken hoe hij PoLF gebruikt in zijn werk. Hij is onder andere telecomdeskundige en doet veel onderzoek aan smartwatches. Tijdens het kwamen verschillende praktijkvoorbeelden voorbij, waarin PoLF-data wordt gebruikt. In het bijzonder was dit het onderzoek naar de moord in Adelaide. Die informatie die in dit gesprek voorbij

kwam is vertrouwelijk, dus er kan niet in details worden getreden waar precies op gelet is in de onderzoeken.

C Techniek

C.1 Instellingen

Tabel C.1: Instellingen van de gebruikte telefoons.

Dataset	Soort apparaat	Soort vergrendeling
SAC	iPhone 5S	Geen vergrendeling
SAC	iPhone 6	Geen vergrendeling
DPP	iPhone 5S A	Patroon tekenen
DPP	iPhone 5S B	Patroon tekenen
JHD	iPhone SE	4-cijferige pincode

C.2 Computer codes

Code C.1: Het aantal stappen per registratie in de *healthdb_secure*-database. Pad naar de database: ***/private/var/mobile/Library/Health/healthdb_secure.sqlite*.

```
1  SELECT
2      DATETIME(SAMPLES.START_DATE + 978307200, 'UNIXEPOCH') AS "START DATE",
3      DATETIME(SAMPLES.END_DATE + 978307200, 'UNIXEPOCH') AS "END DATE",
4      QUANTITY AS "STEPS",
5      (SAMPLES.END_DATE–SAMPLES.START_DATE) AS "TIME IN SECONDS",
6      SAMPLES.DATA_ID AS "SAMPLES TABLE ID",
7      DATA_PROVENANCES.ORIGIN_PRODUCT_TYPE AS "ORIGIN PRODUCT TYPE"
8  FROM
9      SAMPLES,
10     DATA_PROVENANCES,
11     OBJECTS
12     LEFT OUTER JOIN
13         QUANTITY_SAMPLES
14         ON SAMPLES.DATA_ID = QUANTITY_SAMPLES.DATA_ID
15 WHERE
16     SAMPLES.DATA_TYPE = 7
17     AND SAMPLES.DATA_ID = OBJECTS.DATA_ID
18     AND OBJECTS.PROVENANCE = DATA_PROVENANCES.ROWID
```

Code C.2: De instellingen van de Dockerfile om de Jupyterlab-omgeving op te zetten.

```
1  # Image
```

```
FROM ubuntu:latest
LABEL version="0.2"
LABEL description="Jupyterlab environment for Hansken.py and Hansken All-In-One"

# Install Python (tools) and editor
RUN apt-get update && apt-get -y update
RUN apt-get install -y --no-install-recommends \
                npm \
        vim \
        bash \
        python3.8 \
        python3-pip \
        python3-dev \
        build-essential \

# Install PyPi with npm
RUN pip3 -q install pip --upgrade
RUN npm install -g n
RUN n lts

# Create a working directory for code
RUN mkdir src
WORKDIR src

# Install the Python libraries
COPY ./requirements.txt requirements.txt
RUN pip3 install -r requirements.txt

# Add Tini. Tini operates as a process subreaper for jupyter. This prevents kernel crashes.
ENV TINI_VERSION v0.6.0
ADD https://github.com/krallin/tini/releases/download/${TINI_VERSION}/tini /usr/bin/tini
RUN chmod +x /usr/bin/tini
ENTRYPOINT ["/usr/bin/tini", "--"]

# OPTIONAL Create a special user nfi
#RUN useradd -ms /bin/bash nfi
#USER nfi
#WORKDIR /home/nfi

# Run the Docker image
CMD jupyter lab --port=8888 --no-browser --ip=0.0.0.0 --allow-root --shell=bash
```

Code C.3: Een overzicht van de verschillende *libraries*, die zjin gebruikt voor dit onderzoek.

```
pytz
pandas
hansken
seaborn
streamlit
matplotlib
jupyterlab
```

Code C.4: Een facet query uitvoeren om de verschillende soorten bestanden te groeperen in Hansken.

```
"""
    Create a facet search, which consists of a facet query for extensions.
"""

# The ProjectContext makes a connection with the Hansken API, and closes this after the
# task has been completed.
with ProjectContext(var_endpoint, var_project) as context:
    results = context.search(query=Term('type', 'file'),facets=Facet('file.extension'))

    # Set empty dictionary
    facet_dict = {}

    # The facet will be available on the result
    facets = results.facets[0]

    # The loop checks if the extension is not longer than 11 characters. A filtering step
    # to reduce the amount of false positives. Only top 30 will be displayed.
    for facet in facets:
        if len(facet) < 11 and len(facet_dict) < 30:
            facet_dict[facet] = facets[facet].count

    # Print the results
    print("Facet result:", facet_dict)
```

Code C.5: Visualisatie met behulp van Matplotlib. De gebruiker kan zelf aanpassingen doen aan de code om de assen en weergave aan te passen.

```
"""
    Add the plot information and configurations
"""

```

```
# Determine the figure size and rotation of x-ticks
plt.rcParams['figure.figsize'] = [10, 5]
plt.xticks(rotation=90)

# Add labels such as titles, x-axis and y-axis
plt.title("Aantal bestanden per extensie")
plt.xlabel("Extensie")
plt.ylabel("Aantal bestand")

# Use a bar-diagram with a logarithmic y-axis
plt.bar(facet_dict.keys(), facet_dict.values())
plt.yscale("log")

# Save the result as a .png-file
plt.savefig('code_facet_plt_hansken.png', bbox_inches = 'tight', pad_inches = 0.1)
```

Code C.6: Configuratie om te kunnen communiceren met de Hansken API.

```
"""
    Simple setup for the communication with the Hansken API.
"""

# Loading dependencies of Hansken
import matplotlib.pyplot as plt
from hansken.remote import ProjectContext, Facet
from hansken.query import TermFacet, RangeFacet, Term

# Create URL-destinations
url_hansken = "http://192.168.56.42" # Change to your own Hansken service
var_project = "d3dd6aaa-5eb4-4eb9-fa36-24e513aa55be" # Change to your own project number
var_endpoint = url_hansken + ":9091/gatekeeper/"
var_keystore = url_hansken + ":9090/keystore/"
```

Code C.7: De SQL-code voor het achterhalen van interacties van de gebruiker met contacten in de telefoon. Vanaf iOS-versie 13 kan dit commando gebruikt worden. Oorspronkelijk komt deze code uit APOLLO.[12] Pad naar de database: ***/private/var/mobile/Library/CoreDuet/People/interactionC.db*.

```
SELECT
    DATETIME(ZINTERACTIONS.ZSTARTDATE + 978307200, 'UNIXEPOCH') AS 'START DATE',
    DATETIME(ZINTERACTIONS.ZENDDATE + 978307200, 'UNIXEPOCH') AS 'END DATE',
    ZINTERACTIONS.ZBUNDLEID AS 'BUNDLE ID',
    ZINTERACTIONS.ZACCOUNT AS 'ACCOUNT',
```

```
6   ZINTERACTIONS.ZTARGETBUNDLEID AS 'TARGET BUNDLE ID',
7   CASE ZINTERACTIONS.ZDIRECTION
8       WHEN '0' THEN 'INCOMING'
9       WHEN '1' THEN 'OUTGOING'
10  END 'DIRECTION',
11  ZCONTACTS.ZDISPLAYNAME AS 'SENDER DISPLAY NAME',
12  ZCONTACTS.ZIDENTIFIER AS 'SENDER IDENTIFIER',
13  ZCONTACTS.ZPERSONID AS 'SENDER PERSONID',
14  RECEIPIENTCONACT.ZDISPLAYNAME AS 'RECIPIENT DISPLAY NAME',
15  RECEIPIENTCONACT.ZIDENTIFIER AS 'RECIPIENT IDENTIFIER',
16  RECEIPIENTCONACT.ZPERSONID AS 'RECIPIENT PERSONID',
17  ZINTERACTIONS.ZRECIPIENTCOUNT AS 'RECIPIENT COUNT',
18  ZINTERACTIONS.ZDOMAINIDENTIFIER AS 'DOMAIN IDENTIFIER',
19  ZINTERACTIONS.ZISRESPONSE AS 'IS RESPONSE',
20  ZATTACHMENT.ZCONTENTTEXT AS 'CONTEXT TEXT',
21  ZATTACHMENT.ZUTI AS 'UTI',
22  ZATTACHMENT.ZCONTENTURL AS 'CONTENT URL',
23  ZATTACHMENT.ZSIZEINBYTES AS 'SIZE IN BYTES',
24  ZATTACHMENT.ZPHOTOLOCALIDENTIFIER AS 'PHOTO LOCAL IDENTIFIER',
25  HEX(ZATTACHMENT.ZIDENTIFIER) AS 'ATTACHMENT ID',
26  ZATTACHMENT.ZCLOUDIDENTIFIER AS 'CLOUD IDENTIFIER',
27  ZCONTACTS.ZINCOMINGRECIPIENTCOUNT AS 'INCOMING RECIPIENT COUNT',
28  ZCONTACTS.ZINCOMINGSENDERCOUNT AS 'INCOMING SENDER COUNT',
29  ZCONTACTS.ZOUTGOINGRECIPIENTCOUNT AS 'OUTGOING RECIPIENT COUNT',
30  DATETIME(ZINTERACTIONS.ZCREATIONDATE + 978307200, 'UNIXEPOCH') AS '
        ZINTERACTIONS CREATION DATE',
31  DATETIME(ZCONTACTS.ZCREATIONDATE + 978307200, 'UNIXEPOCH') AS 'ZCONTACTS
        CREATION DATE',
32  DATETIME(ZCONTACTS.ZFIRSTINCOMINGRECIPIENTDATE + 978307200, 'UNIXEPOCH')
        AS 'FIRST INCOMING RECIPIENT DATE',
33  DATETIME(ZCONTACTS.ZFIRSTINCOMINGSENDERDATE + 978307200, 'UNIXEPOCH') AS '
        FIRST INCOMING SENDER DATE',
34  DATETIME(ZCONTACTS.ZFIRSTOUTGOINGRECIPIENTDATE + 978307200, 'UNIXEPOCH')
        AS 'FIRST OUTGOING RECIPIENT DATE',
35  DATETIME(ZCONTACTS.ZLASTINCOMINGSENDERDATE + 978307200, 'UNIXEPOCH') AS '
        LAST INCOMING SENDER DATE',
36  CASE ZCONTACTS.ZLASTINCOMINGRECIPIENTDATE
37      WHEN '0' THEN '0'
38      ELSE DATETIME(ZCONTACTS.ZLASTINCOMINGRECIPIENTDATE + 978307200, '
            UNIXEPOCH')
39  END 'LAST INCOMING RECIPIENT DATE',
40  DATETIME(ZCONTACTS.ZLASTOUTGOINGRECIPIENTDATE + 978307200, 'UNIXEPOCH')
```

```
        AS 'LAST OUTGOING RECIPIENT DATE',
41      ZCONTACTS.ZCUSTOMIDENTIFIER AS 'CUSTOM IDENTIFIER',
42      ZINTERACTIONS.ZCONTENTURL AS 'CONTENT URL',
43      ZINTERACTIONS.ZLOCATIONUUID AS 'LOCATION UUID',
44      ZINTERACTIONS.ZGROUPNAME AS 'GROUP NAME',
45      ZINTERACTIONS.ZDERIVEDINTENTIDENTIFIER AS 'DERIVIED INTENT ID',
46      ZINTERACTIONS.Z_PK AS 'ZINTERACTIONS TABLE ID'
47  FROM ZINTERACTIONS
48  LEFT JOIN ZCONTACTS ON ZINTERACTIONS.ZSENDER = ZCONTACTS.Z_PK
49  LEFT JOIN Z_1INTERACTIONS ON ZINTERACTIONS.Z_PK == Z_1INTERACTIONS.
        Z_3INTERACTIONS
50  LEFT JOIN ZATTACHMENT ON Z_1INTERACTIONS.Z_1ATTACHMENTS == ZATTACHMENT.Z_PK
51  LEFT JOIN Z_2INTERACTIONRECIPIENT ON ZINTERACTIONS.Z_PK==
        Z_2INTERACTIONRECIPIENT.Z_3INTERACTIONRECIPIENT
52  LEFT JOIN ZCONTACTS RECEIPIENTCONACT ON Z_2INTERACTIONRECIPIENT.
        Z_2RECIPIENTS== RECEIPIENTCONACT.Z_PK
```

Code C.8: Het aantal stappen per registratie in de *cache_encryptedC*-database. *Count* telt alle stappen bij elkaar op, dus het verschil wordt berekend aan de hand van de vorige registratie. Conversie van *Timestamp Apple Time* naar UTC-tijd. Pad naar de database: ***/private/var/root/Library/Caches/locationd/cache_encryptedC.db*.

```
1  SELECT
2       DATETIME(STARTTIME + 978307200, 'UNIXEPOCH') AS "START TIME",
3       TIMESTAMP AS "MOVEMENT TIME",
4       COUNT AS "COUNT",
5       count – lag(count, 1) OVER (ORDER BY "START TIME") AS "COUNT_DIFF",
6       DISTANCE AS "DISTANCE"
7  FROM STEPCOUNTHISTORY
```

Code C.9: Python-code om de hartslagmeter en stappenteller te plotten met Pandas en Matplotlib. Pad naar de database: ***/private/var/mobile/Library/Health/healthdb_secure.sqlite*.

```
1  """
2      Graphing the healthdb_secure.sqlite–data of the MSD–dataset.
3  """
4
5  # Import libraries
6  import pytz
7  import sqlite3
8  import pandas as pd
9  import matplotlib.pyplot as plt
10 import matplotlib.dates as md
```

```
11
12 # Make a connection with the healthdb_secure.sqlite-databse. Offline version is
13 # used to reduce the connections with Hansken AIO
14 database = sqlite3.connect('./healthdb_secure.sqlite')
15
16 # Create a database cursor
17 cursor = database.cursor()
18
19 # Get the heart-rate from the Apple Watch using the adjusted APOLLO-code.
20 rows = cursor.execute("""
21         SELECT
22             DATETIME(SAMPLES.START_DATE + 978307200, 'UNIXEPOCH') AS "DATE",
23             ORIGINAL_QUANTITY AS "HEART RATE",
24             UNIT_STRINGS.UNIT_STRING AS "UNITS",
25             QUANTITY AS "QUANTITY",
26             SAMPLES.DATA_ID AS "SAMPLES TABLE ID"
27         FROM
28             SAMPLES
29             LEFT OUTER JOIN
30                 QUANTITY_SAMPLES
31                 ON SAMPLES.DATA_ID = QUANTITY_SAMPLES.DATA_ID
32             LEFT OUTER JOIN
33                 UNIT_STRINGS
34                 ON QUANTITY_SAMPLES.ORIGINAL_UNIT = UNIT_STRINGS.ROWID
35         WHERE
36             SAMPLES.DATA_TYPE = 5
37         """).fetchall()
38
39 # Put the results in a Pandas dataframe
40 df_heart_rate = pd.DataFrame.from_records(rows, columns=[
41                         "date", "heart_rate", "units", "quantity", "sampl_table_id"
42                     ])
43
44 # Get the steps from the Apple Watch and iPhone using the adjusted APOLLO-code.
45 rows = cursor.execute("""
46         SELECT
47             DATETIME(SAMPLES.START_DATE + 978307200, 'UNIXEPOCH') AS "START DATE",
48             DATETIME(SAMPLES.END_DATE + 978307200, 'UNIXEPOCH') AS "END DATE",
49             QUANTITY AS "STEPS",
50             (SAMPLES.END_DATE-SAMPLES.START_DATE) AS "TIME IN SECONDS",
51             SAMPLES.DATA_ID AS "SAMPLES TABLE ID"
52         FROM
```

```
            SAMPLES
            LEFT OUTER JOIN
                QUANTITY_SAMPLES
                ON SAMPLES.DATA_ID = QUANTITY_SAMPLES.DATA_ID
        WHERE
            SAMPLES.DATA_TYPE = 7
        """).fetchall()

# Put the results in a Pandas dataframe
df_steps = pd.DataFrame.from_records(rows, columns=[
                    "start_date", "end_date", "steps", "time_in_seconds", "sample_table_id"
                ])

# Close both the connection with the database and cursor
cursor.close()
database.close()

# Create a seperate dataframe with date as index for heart rate
df_plot = df_heart_rate
df_plot["date"] = pd.to_datetime(df_plot["date"], format="%Y-%m-%d %H:%M:%S")
df_plot.set_index('date', inplace=True)

# Create a seperate dataframe with date as index for steps
df_plot_steps = df_steps
df_plot_steps["start_date"] = pd.to_datetime(df_plot_steps["start_date"], format="%Y-%m-%d %H:%M:%
    S")
df_plot_steps.set_index('start_date', inplace=True)

# Set the dates as well as timezones
dates = [20, 21, 22, 23, 24]
adelaide = [20, 21, 22]
perth = [23, 24]

# Set timezones
adelaide_tzinfo = pytz.timezone("Australia/Adelaide")
perth_tzinfo = pytz.timezone("Australia/Perth")

# Add the x-axis interval
delta_minus = 3
delta_plus = 4

# Make a plot with multiple subplots
```

```
fig, ax1 = plt.subplots(len(dates),2,figsize=(12, 14))

# Iterate over the input dates
for i in range(len(dates)):
    # Set the day for strings
    day = dates[i]
    day_minus = day - 1
    day = str(day)
    begin_time = '2021-04-' + str(day_minus) + ' 19:00:00'
    middle_time = '2021-04-' + day + ' 10:00:00'
    end_time ='2021-04-' + day + ' 08:00:00'

    # Set timezone
    if int(day) in adelaide:
        timezone = adelaide_tzinfo
    if int(day) in perth:
        timezone = perth_tzinfo

    # Create two plots:
    for j in range(2):
        # Heart rate (left)
        if j == 0:

            # Get subset of the original dataset and convert to timezone
            result = df_plot.loc[begin_time:end_time]
            result.tz_convert(timezone)

            # Insert the data as stem-graph with hour-interval on x-axis and max y-axis
            ax1[i,j].stem(result.index, 'heart_rate', data=result)
            ax1[i,j].set_xlim(pd.Timestamp(middle_time, tz=timezone)-pd.Timedelta(delta_minus,'h'),
                    pd.Timestamp(middle_time, tz=timezone)+pd.Timedelta(delta_plus,'h'))
            ax1[i,j].xaxis.set_major_locator(md.HourLocator(interval = 1, tz=timezone))
            ax1[i,j].xaxis.set_major_formatter(md.DateFormatter('%H:%M', tz=timezone))
            ax1[i,j].set_ylim(0,125)

            # Set title and labels
            ax1[i,j].set_title("Hartritme op " + day + " april 2021")
            ax1[i,j].set_xlabel(" ")
            ax1[i,j].set_ylabel("Hartslag")

            # Add on the last plot a x-axis label
            if i == len(dates)-1:
```

```
136                 ax1[i,j].set_xlabel("Tijd")
137
138         # Steps (right)
139         elif j == 1:
140
141             # Get subset of the original dataset and convert to timezone
142             result = df_plot_steps.loc[begin_time:end_time]
143             result.tz_convert(timezone)
144
145             # Determine the if the data is from the Apple Watch or iPhone with
146             # hour-interval on x-axis, max y-axis and legend
147             res_watch = result[result['origin_product_type'] == 'Watch4,4']
148             res_iphone = result[result['origin_product_type'] == 'iPhone11,6']
149             ax1[i,j].stem(res_watch.index, 'steps','y', markerfmt='yo', data=res_watch, label='Watch 4')
150             ax1[i,j].stem(res_iphone.index, 'steps','g', markerfmt='go', data=res_iphone, label='iPhone
                    11')
151             ax1[i,j].set_xlim(pd.Timestamp(middle_time, tz=timezone)-pd.Timedelta(delta_minus,'h'),
152                     pd.Timestamp(middle_time, tz=timezone)+pd.Timedelta(delta_plus,'h'))
153             ax1[i,j].xaxis.set_major_locator(md.HourLocator(interval = 1, tz=timezone))
154             ax1[i,j].xaxis.set_major_formatter(md.DateFormatter('%H:%M', tz=timezone))
155             ax1[i,j].legend(loc='upper left')
156             ax1[i,j].set_ylim(0,1000)
157
158             # Set title and labels
159             ax1[i,j].set_title("Stappenteller op " + day + " april 2021")
160             ax1[i,j].set_xlabel(" ")
161             ax1[i,j].set_ylabel("Aantal stappen")
162
163             # Add on the last plot a x-axis label
164             if i == len(dates)-1:
165                 ax1[i,j].set_xlabel("Tijd")
166
167         # Rotate all x-ticks
168         for tick in ax1[i,j].get_xticklabels():
169             tick.set_rotation(45)
170
171 # Save figure
172 fig.tight_layout()
173 fig.savefig('health_trans.png', bbox_inches = 'tight', pad_inches = 0.1)
174 fig.savefig('health.png', bbox_inches = 'tight', pad_inches = 0.1, facecolor='white')
```

Code C.10: De berichten worden gekoppeld aan de metadata. In de metadata staat de tijd vermeld. *unixtime* is de ruwe tijd aangegeven in de database. Aan de hand hiervan kan de opsteller van het bericht mogelijk bepaald worden. *date* geeft de UTC-tijd aan. Pad naar de database: ***/private/var/mobile/Containers/Shared/AppGroup/*/fts/ChatSearchV*.sqlite*

```
SELECT
    metadata.date AS ”unixtime”,
        datetime(metadata.date+978307200, ’UNIXEPOCH’) AS ”date”,
        c1contents AS ”message”,
        c0chatSession AS ”session”
FROM docs_content
LEFT JOIN metadata
WHERE docs_content.docid = metadata.docID
AND c0chatSession LIKE ”%%”
```

Code C.11: SQL-commando om de berichten uit de WhatsApp-database te halen. Zowel de tijd in ruwe data vorm en conversie naar UTC wordt weergegeven. De richting naar wie de data is verzonden door de laatste variabelen. Pad naar de database: ***/private/var/mobile/Containers/Shared/AppGroup/*/ChatStorage.sqlite*.

```
SELECT
        ZMESSAGEDATE,
        datetime(ZMESSAGEDATE+978307200, ’UNIXEPOCH’) AS ”date”,
        ZTEXT,
        ZFROMJID,
        ZTOJID
FROM ZWAMESSAGE
```

D Hansken.py

D.1 Werkingsmechanisme

In het onderzoek is vaak Hansken.py voorbijgekomen als functionaliteit. Hiermee haalt Hansken data op. Om te begrijpen hoe precies Hansken de data verwerkt, wordt eerst uitgelegd wat een digitaal spoor is. De definitie van een digitaal spoor is binnen het DFaaS-model een digitaal artefact met (een link naar de) data en metadata. Tijdens de verwerking worden een *images* worden deze digitale sporen geïdentificeerd en de metadata geëxtraheerd. De metadata wordt gebruikt in het sporenmodel om de digitale sporen te categoriseren. De informatie uit de metadata moet wel binnen het sporenmodel passen. Door deze stap is het mogelijk om gemakkelijk een zoekopdracht uit te voeren binnen Hansken. Het is niet alleen mogelijk om data op te halen uit Hansken. Gebruikers kunnen ook eigen digitale sporen uploaden of bestaande sporen uitbreiden.[2]

Het sporenmodel bevat meerdere categorieën: *chat, contact, data, file, gps, phoneCall,* etc. Elke categorie heeft een eigen set aan spoortypen. Zo heeft een contact de spoortypen: *application, emailAddresses, firstName, id, lastName, name, phoneNumbers and timestamps*. Elk spoortype heeft daarnaast een eigen beschrijving, mogelijke collectie, type (*int, string, date, latLong*, etc.). In de documentatie binnen de expert interface staan deze categorieën uitgebreider beschreven.

De PoLF-sporen die uit APOLLO-scripts gevonden worden, kunnen terug in Hansken geplaatst worden met behulp van het sporenmodel. Deze sporen lijken het meest op loggegevens, die afkomstig zijn van een smartphone. Het spoortype dat het meest hierop lijkt is *event* (*"An event, mainly used for operating system events."*). Het voordeel van dit spoortype is de type *misc*. Hiermee kan de gebruiker extra informatie toevoegen die niet voorkomt in het standaard sporenmodel. Een voorbeeld is het aantal stappen of de hartslag uit de *healthdb_secure*-database, die niet standaard gedefinieerd zijn in Hansken.

De codes die gebruikt zijn voor *knowledgeC* en *interactionC*, worden doorgenomen, om te laten zien hoe de sporen uit APOLLO weer teruggeplaatst kunnen worden in Hanksen. De eerste stap is het converteren van de SQL-query naar strings (zie code D.1). Hiermee hoeft de gebruiker alleen de juiste database en query code in te voeren. Anders moet de gebruiker per query uitschrijven welke kolommen worden aangeroepen. Als extra toevoeging voert de code ook de extractie uit en heeft de resultaten hiervan terug.

De volgende stap is de connectie maken met de database met Hansken.py. Voor de configuratie van Hansken connectie zie code C.6. In plaats van een facet-zoekopdracht wordt nu aan Hansken de database opgevraagd. De uitvoering van de code is verglijkbaar met de facet voorbeeld. Met behulp van de code D.1 worden de resultaten opgehaald. Elk nieuw gevonden spoor kan nu worden teruggeplaatst in Hansken als kindspoor van de database. Alles zou in de *misc* geplaatst kunnen worden. Ter volledigheid wordt dit ook gedaan. Om het zoeken in Hansken wat makkelijker te maken, zijn voor *knowledgeC* (zie code D.2) en *interactionC* (zie code D.3) extra spoortype toegevoegd. Zo zijn de start en eind tijden geconverteerd naar UTC. Deze zijn daarna als *createdOn* en *generatedOn* toegevoegd. Het toevoegen van het spoortype kan met behulp van de `trace.child_builder()`-funcite in Hansken.py.

Een gebruiker heeft hiermee de mogelijkheid om elk script uit APOLLO of eigen codes uit te voeren. Het kunnen uitbreiden van digitale sporen binnen een dataset een krachtige functionaliteit, om verdiepende analyses te kunnen uitvoeren.

D.2 Kindspoor codes

Code D.1: Een *class* om SQL-commando's te converteren naaar strings. Deze strings worden daarna gebruikt als *column headers* voor de resultaatverwerking.

```
1  """
2      Class to convert, process and substitute data in a SQL-query for Hansken.py.
3  """
4
5  class ExtractPOL():
6      def __init__(self, database, query="", results=[]):
7
8          # Set initial values
9          self.database = database
10         self.query = query
11         self.results = results
12
13         # Substitute map for SQL-query
14         self.replace_map = {
15             " " : "_",
16             "(" : "",
17             ")" : ""
18         }
19
20     # Set query
21     def set_query(self, query):
```

```
22          self.query= query
23
24      # Set results
25      def set_result(self, results):
26          self.results = results
27
28      # Get the rows from database
29      def get_rows(self):
30          # Return all rows from the database
31          cursor = self.database.cursor()
32          rows = cursor.execute(self.query).fetchall()
33          return rows
34
35      # Set replace with map and set string to lowercase
36      def transform(self, temp):
37          for key, val in self.replace_map.items():
38              temp = temp.replace(key,val)
39          return temp.lower()
40
41      # Convert the column names to strings
42      def convert(self):
43          # Only take the part between SELECT and first FROM
44          self.result = []
45          query_before_from = self.query.splitlines()
46          index = [idx for idx, s in enumerate(query_before_from) if 'FROM' in s]
47
48          # Seperate every line
49          line_separated = query_before_from[:index[0]]
50          for l in line_separated[1:]:
51              if "WHEN" in l or "CASE" in l or "SELECT" in l:
52                  continue
53              elif "AS" in l:
54                  sub_result = l.split("\'")[1]
55              else:
56                  sub_result = l.strip()
57              if sub_result != '':
58                  self.result.append(self.transform(sub_result))
59
60      # Get the results back
61      def get_result(self):
62          return self.result
```

Code D.2: De code om Python de *knowledgeC App in focus*-commando uit APOLLO uit te laten voeren en de resultaten hiervan terug te plaatsen in Hansken als kindsporen. ***/private/var/mobile/Library/Health/healthdb_secure.sqlite*.

```
"""
    Get the knowlegdeC–data from the iPhones.
"""

# Set up libraries
from pytz import timezone
from datetime import datetime

# Convert time to UTC, which is used by Hansken
def convert_time(input_time):
    date_time_obj = datetime.strptime(str(input_time), "%Y–%m–%d %H:%M:%S")
    datetime_obj_utc = date_time_obj.replace(tzinfo=timezone('UTC'))
    return datetime_obj_utc

# Get the knowledgeC–database and put the results back as child–traces
def extract_knowlegdeC_app_in_focus(database_name, query_sql):

    # Create a connection with Hansken, which closes automatically
    with ProjectContext(var_endpoint, var_project, keystore_url=var_keystore) as context:
        with context:

            # Only take the interactionC–database from iPhone SE.
            query = '"file.name:"' + database_name
            results = context.search(query)

            # Get the trace
            for trace in results:

                # Export the trace to a temporary file
                export.to_file(trace, '/tmp/temporary–file.db')
                database = sqlite3.connect('/tmp/temporary–file.db')

                # Extract the data and get the column names
                extract = ExtractPOL(database, query_sql)
                extract.convert()
                columns_sql = extract.get_result()

                # Get every row in the results
```

```
39                  for i in extract.get_rows():
40                      builder = trace.child_builder()
41                      child_dict = {
42                          'name' : 'knowledgeC ' + i[0],
43                          'event.action' : 'knowledgeC application in focus',
44                          'event.id' : i[11],
45                          'event.index' : i[12],
46                          'event.type' : 'user',
47                          'event.source' : i[2],
48                          'event.generatedOn' : convert_time(i[0]),
49                          'event.createdOn' : convert_time(i[10])
50                      }
51                      elem = 0
52                      for j in i:
53                          child_dict["event.misc."+columns_sql[elem]] = str(j)
54                          elem += 1
55
56                      # Clear None values
57                      elem = 0
58                      for j in i:
59                          child_dict["event.misc."+columns_sql[elem]] = str(j)
60                          elem += 1
61                      filtered = {k: v for k, v in child_dict.items() if v != 'None'}
62                      child_dict.clear()
63                      child_dict.update(filtered)
64
65                      # Add the results back in Hansken as child traces
66                      builder.update(child_dict).build()
67
68  # Set database name and query command
69  database_name = '''knowledgeC.db'''
70  query_sql = """
71  SELECT
72                  DATETIME(ZOBJECT.ZSTARTDATE+978307200,'UNIXEPOCH') AS "START",
73                  DATETIME(ZOBJECT.ZENDDATE+978307200,'UNIXEPOCH') AS "END",
74                  ZOBJECT.ZVALUESTRING AS "BUNDLE ID",
75                  (ZOBJECT.ZENDDATE-ZOBJECT.ZSTARTDATE) AS "USAGE IN SECONDS",
76                  (ZOBJECT.ZENDDATE-ZOBJECT.ZSTARTDATE)/60.00 AS "USAGE IN MINUTES",
77                  ZSTRUCTUREDMETADATA .Z_DKAPPLICATIONMETADATAKEY__LAUNCHREASON
                        AS "LAUNCH REASON",
78                  ZSTRUCTUREDMETADATA .
                        Z_DKAPPLICATIONMETADATAKEY__EXTENSIONCONTAININGBUNDLEIDENTIFIER
```

```
                    AS "BUNDLE ID",
79                  ZSTRUCTUREDMETADATA .
                        Z_DKAPPLICATIONMETADATAKEY__EXTENSIONHOSTIDENTIFIER AS "
                        EXTENSION HOST ID",
80                  CASE ZOBJECT.ZSTARTDAYOFWEEK
81                      WHEN "1" THEN "Sunday"
82                      WHEN "2" THEN "Monday"
83                      WHEN "3" THEN "Tuesday"
84                      WHEN "4" THEN "Wednesday"
85                      WHEN "5" THEN "Thursday"
86                      WHEN "6" THEN "Friday"
87                      WHEN "7" THEN "Saturday"
88                  END "DAY OF WEEK",
89                  ZOBJECT.ZSECONDSFROMGMT/3600 AS "GMT OFFSET",
90                  DATETIME(ZOBJECT.ZCREATIONDATE+978307200,'UNIXEPOCH') AS "ENTRY
                        CREATION",
91                  ZOBJECT.ZUUID AS "UUID",
92                  ZOBJECT.Z_PK AS "ZOBJECT TABLE ID"
93              FROM ZOBJECT
94                      LEFT JOIN
95              ZSTRUCTUREDMETADATA
96              ON ZOBJECT.ZSTRUCTUREDMETADATA = ZSTRUCTUREDMETADATA.Z_PK
97          LEFT JOIN
98              ZSOURCE
99              ON ZOBJECT.ZSOURCE = ZSOURCE.Z_PK
100             WHERE ZSTREAMNAME IS "/app/inFocus" """
101
102
103 # Execute function
104 extract_knowlegdeC_app_in_focus(database_name, query_sql)
```

Code D.3: De code om Python de *interactionC Contact intents*-commando uit APOLLO uit te laten voeren en de resultaten hiervan terug te plaatsen in Hansken als kindsporen. ***/private/var/mobile/Library/CoreDuet/People/interactionC.db*.

```
1 """
2     Get the interactionC–data from the iPhone SE.
3 """
4
5 # Set up libraries
6 from pytz import timezone
7 from datetime import datetime
```

```
8
9  # Convert time to UTC, which is used by Hansken
10 def convert_time(input_time):
11     date_time_obj = datetime.strptime(str(input_time), "%Y-%m-%d %H:%M:%S")
12     datetime_obj_utc = date_time_obj.replace(tzinfo=timezone('UTC'))
13     return datetime_obj_utc
14
15 # Get the interactionC-database and put the results back as child-traces
16 def extract_interactionC(database_name, query_sql):
17
18     # Create a connection with Hansken, which closes automatically
19     with ProjectContext(var_endpoint, var_project, keystore_url=var_keystore) as context:
20         with context:
21
22             # Only take the interactionC-database from iPhone SE.
23             query = '"file.name:"' + database_name + '" image:'9ca6dd42-8266-4446-a4f6-4
                   e91ed3d3610' "'
24             results = context.search(query)
25
26             # Get the trace
27             for trace in results:
28
29                 # Export the trace to a temporary file
30                 export.to_file(trace, '/tmp/temporary-file.db')
31                 database = sqlite3.connect('/tmp/temporary-file.db')
32
33                 # Extract the data and get the column names
34                 extract = ExtractPOL(database, query_sql)
35                 extract.convert()
36                 columns_sql = extract.get_result()
37
38                 # Get every row in the results
39                 for i in extract.get_rows():
40
41                     # Set builder function for child traces
42                     builder = trace.child_builder()
43                     child_dict = {
44                         'name' : 'interactionC ' + i[0], # Add name of database with timestamp
45                         'event.action' : 'interactionC ', # Add action
46                         'event.index' : i[38], # Add internal database index
47                         'event.type' : 'user', # Add type user added
48                         'event.generatedOn' : convert_time(i[0]), # Convert start date
```

```
                    'event.createdOn' : convert_time(i[1]) # Convert end date
                }

                # Clear None values
                elem = 0
                for j in i:
                    child_dict["event.misc."+columns_sql[elem]] = str(j)
                    elem += 1
                filtered = {k: v for k, v in child_dict.items() if v != 'None'}
                child_dict.clear()
                child_dict.update(filtered)

                # Add the results back in Hansken as child traces
                builder.update(child_dict).build()

# Set database name and query command
database_name = '''interactionC.db'''
query_sql = """
    SELECT
        DATETIME(ZINTERACTIONS.ZSTARTDATE + 978307200, 'UNIXEPOCH') AS 'START DATE',
        DATETIME(ZINTERACTIONS.ZENDDATE + 978307200, 'UNIXEPOCH') AS 'END DATE',
        ZINTERACTIONS.ZBUNDLEID AS 'BUNDLE ID',
        ZINTERACTIONS.ZACCOUNT AS 'ACCOUNT',
        ZINTERACTIONS.ZTARGETBUNDLEID AS 'TARGET BUNDLE ID',
        CASE ZINTERACTIONS.ZDIRECTION
            WHEN '0' THEN 'INCOMING'
            WHEN '1' THEN 'OUTGOING'
        END 'DIRECTION',
        ZCONTACTS.ZDISPLAYNAME AS 'SENDER DISPLAY NAME',
        ZCONTACTS.ZIDENTIFIER AS 'SENDER IDENTIFIER',
        ZCONTACTS.ZPERSONID AS 'SENDER PERSONID',
        RECEIPIENTCONACT.ZDISPLAYNAME AS 'RECIPIENT DISPLAY NAME',
        RECEIPIENTCONACT.ZIDENTIFIER AS 'RECIPIENT IDENTIFIER',
        RECEIPIENTCONACT.ZPERSONID AS 'RECIPIENT PERSONID',
        ZINTERACTIONS.ZRECIPIENTCOUNT AS 'RECIPIENT COUNT',
        ZINTERACTIONS.ZDOMAINIDENTIFIER AS 'DOMAIN IDENTIFIER',
        ZINTERACTIONS.ZISRESPONSE AS 'IS RESPONSE',
        ZATTACHMENT.ZCONTENTTEXT AS 'CONTEXT TEXT',
        ZATTACHMENT.ZUTI AS 'UTI',
        ZATTACHMENT.ZCONTENTURL AS 'CONTENT URL',
        ZATTACHMENT.ZSIZEINBYTES AS 'SIZE IN BYTES',
        ZATTACHMENT.ZPHOTOLOCALIDENTIFIER AS 'PHOTO LOCAL IDENTIFIER',
```

```
91          HEX(ZATTACHMENT.ZIDENTIFIER) AS 'ATTACHMENT ID',
92          ZATTACHMENT.ZCLOUDIDENTIFIER AS 'CLOUD IDENTIFIER',
93          ZCONTACTS.ZINCOMINGRECIPIENTCOUNT AS 'INCOMING RECIPIENT COUNT',
94          ZCONTACTS.ZINCOMINGSENDERCOUNT AS 'INCOMING SENDER COUNT',
95          ZCONTACTS.ZOUTGOINGRECIPIENTCOUNT AS 'OUTGOING RECIPIENT COUNT',
96          DATETIME(ZINTERACTIONS.ZCREATIONDATE + 978307200, 'UNIXEPOCH') AS '
                ZINTERACTIONS CREATION DATE',
97          DATETIME(ZCONTACTS.ZCREATIONDATE + 978307200, 'UNIXEPOCH') AS 'ZCONTACTS
                CREATION DATE',
98          DATETIME(ZCONTACTS.ZFIRSTINCOMINGRECIPIENTDATE + 978307200, 'UNIXEPOCH')
                AS 'FIRST INCOMING RECIPIENT DATE',
99          DATETIME(ZCONTACTS.ZFIRSTINCOMINGSENDERDATE + 978307200, 'UNIXEPOCH') AS '
                FIRST INCOMING SENDER DATE',
100         DATETIME(ZCONTACTS.ZFIRSTOUTGOINGRECIPIENTDATE + 978307200, 'UNIXEPOCH')
                AS 'FIRST OUTGOING RECIPIENT DATE',
101         DATETIME(ZCONTACTS.ZLASTINCOMINGSENDERDATE + 978307200, 'UNIXEPOCH') AS '
                LAST INCOMING SENDER DATE',
102         CASE ZCONTACTS.ZLASTINCOMINGRECIPIENTDATE
103             WHEN '0' THEN '0'
104             ELSE DATETIME(ZCONTACTS.ZLASTINCOMINGRECIPIENTDATE + 978307200, '
                    UNIXEPOCH')
105         END 'LAST INCOMING RECIPIENT DATE',
106         DATETIME(ZCONTACTS.ZLASTOUTGOINGRECIPIENTDATE + 978307200, 'UNIXEPOCH')
                AS 'LAST OUTGOING RECIPIENT DATE',
107         ZCONTACTS.ZCUSTOMIDENTIFIER AS 'CUSTOM IDENTIFIER',
108         ZINTERACTIONS.ZCONTENTURL AS 'CONTENT URL',
109         ZINTERACTIONS.ZLOCATIONUUID AS 'LOCATION UUID',
110         ZINTERACTIONS.ZGROUPNAME AS 'GROUP NAME',
111         ZINTERACTIONS.ZDERIVEDINTENTIDENTIFIER AS 'DERIVIED INTENT ID',
112         ZINTERACTIONS.Z_PK AS 'ZINTERACTIONS TABLE ID'
113     FROM ZINTERACTIONS
114     LEFT JOIN ZCONTACTS ON ZINTERACTIONS.ZSENDER = ZCONTACTS.Z_PK
115     LEFT JOIN Z_1INTERACTIONS ON ZINTERACTIONS.Z_PK == Z_1INTERACTIONS.
            Z_3INTERACTIONS
116     LEFT JOIN ZATTACHMENT ON Z_1INTERACTIONS.Z_1ATTACHMENTS == ZATTACHMENT.Z_PK
117     LEFT JOIN Z_2INTERACTIONRECIPIENT ON ZINTERACTIONS.Z_PK==
            Z_2INTERACTIONRECIPIENT.Z_3INTERACTIONRECIPIENT
118     LEFT JOIN ZCONTACTS RECEIPIENTCONACT ON Z_2INTERACTIONRECIPIENT.
            Z_2RECIPIENTS== RECEIPIENTCONACT.Z_PK
119 """
120
121 # Execute function
```

122 extract_interactionC(database_name, query_sql)