Nederlands Forensisch Instituut
Ministerie van Justitie en Veiligheid

Terug naar de bestanden

Technische toelichting over identificeren, verbergen en verwijderen van bestanden

22 juni 2019

Opdrachtgever	Openbaar Ministerie Landelijk Parket Landelijk Expertisecentrum Kinderporno en Kindersextoerisme

technische toelichting

Nederlands Forensisch Instituut

Postadres
Postbus 24044
2490 AA Den Haag

Bezoekadres
Laan van Ypenburg 6
2497 GB Den Haag
T (070) 888 66 66
F (070) 888 65 55

www.forensischinstituut.nl

Inhoudsopgave

Inhoudelijke toelichtingen:

Drs. R.J.P. van Bree 070-8886452
Ir. R. Schramp 070-8886447

De meest actuele versie van dit document kunt u op www.forensischinstituut.nl vinden.

1 Inleiding

In de rechtspraak zijn in het afgelopen anderhalf jaar een aantal beslagbeslissingen over bestanden en gegevensdragers geweest. In deze uitspraken wordt opgenomen dat bestanden of gegevens dienen te worden teruggeven aan beslagenen of derden of dat strafbare afbeeldingen van de gegevensdragers moeten worden verwijderd waarna de gegevensdragers teruggegeven dienen te worden.

Het grootste risico hierbij is dat er strafbaar materiaal teruggeven wordt aan de verdachte. Aan de andere kant bestaat het recht van de verdachte op het terugkrijgen van de 'legale content', zoals privédocumenten of vakantiefoto's. Het is dus belangrijk dat er zowel bij het OM als de Rechtspraak kennis en inzicht bestaat om beslissingen hieromtrent weloverwogen te kunnen nemen.
Om over de digitaal technische achtergronden bij dit onderwerp over de juiste kennis en inzichten te beschikken is het NFI verzocht een technische toelichting over dit onderwerp op te stellen. De doelstelling van dit document is om zo duidelijk en begrijpelijk mogelijk soms complexe onderwerpen betreffende de werking van computers en digitale gegevens en hun eigenschappen uit te leggen. Het is zeker niet de bedoeling om de onderwerpen diepgaand, uitvoerig en wetenschappelijk uit te leggen, maar om ze in het licht van de discussie rondom teruggave van digitaal bewijs zo goed mogelijk toe te lichten.
In deze toelichting ligt de nadruk veelal op de gegevens en gegevensdragers van computers. De gebruikte informatiebronnen voor deze toelichting bestaan voornamelijk uit bij het NFI bestaande kennis en informatie evenals informatie uit de literatuur. Er hebben diverse deskundigen van de afdeling Forensische Digitale Technologie bijgedragen aan de totstandkoming van deze toelichting.

2 Onderwerpen

Op verzoek van het Landelijk Expertisecentrum Kinderporno en Kindersekstoerisme wordt in deze toelichting een aantal onderwerpen behandeld. In hoofdstuk 3 wordt de werking van digitale gegevensdragers (zoals usb sticks en harddisks) beschreven. Op deze gegevensdragers komen verschillende soorten 'bestandssystemen' voor. Ieder bestandssysteem heeft eigen manier waarop de bestanden en de bijbehorende 'metadata' worden opgeslagen op de gegevensdrager.
Als bestanden of gegevens dienen te worden teruggegeven aan beslagenen of derden, dan dienen eerst de juisten bestanden geselecteerd te worden.
In hoofdstuk 4 worden methoden voor de selectie of identificatie van bestanden besproken en de uitdagingen die hier bij bestaan. Hierbij wordt vooral ingegaan op de hoeveelheid van bestanden en op de talloze manieren waarop bestanden te verbergen zijn.
Het verwijderen van bestanden is het onderwerp van hoofdstuk 5. Hoe goed en hoe definitief kun je bestanden verwijderen op een gegevensdrager? Ten slotte wordt er in de bijlagen nog een extra toelichting gegeven op de onderwerpen bestandskenmerken, cryptografie en wachtwoorden.

In deze toelichting gaat het over bestanden met een 'illegale content'. Wat er precies in dit document wordt bedoeld met illegale content is beschreven in het onderstaande kader.

Illegale content

- gegevens of bestanden waarvan het bezit of voorhanden hebben of enige andere handeling daarmee strafbaar gesteld is. Voorbeelden: kinderpornografische afbeeldingen (240b Sr), technische hulpmiddelen en wachtwoorden als bedoeld in artikel 139d Sr, gegevens die van diefstal afkomstig zijn (strafbaar sinds CCIII in werking is getreden (art. 138c Sr)).
- gegevens of bestanden die vatbaar zijn voor vernietiging/onttrekking aan het verkeer, d.w.z. dat ze niet terug dienen te keren in de maatschappij. Het gaat om gegevens met betrekking tot welke of met behulp waarvan een strafbaar feit is begaan, voor zover de vernietiging noodzakelijk is ter voorkoming van nieuwe strafbare feiten. Bijvoorbeeld RAT's, DDoS-pakketten, malware, ransomware, contactgegevens van medeverdachten/leveranciers/afnemers, inloggegevens van internetomgevingen waar over de strafbare feiten gecommuniceerd wordt, (seksuele) chats met minderjarigen of 'gelijkgestemden', niet als strafbaar beoordeelde afbeeldingen van minderjarigen (bijvoorbeeld slachtoffers in badkleding) die tussen de kinderpornografische afbeeldingen staan en sporen van strafbaar materiaal, zoals bestandsnamen van kinderpornografische afbeeldingen.
- gegevens of bestanden die toegang geven tot cryptocurrency (vermogensbestanddelen)[1].

Met de term **legale** content worden gegevens of bestanden bedoeld waarvan is vastgesteld dat deze geen illegale content zijn.

[1] Die hoeven an sich natuurlijk niet illegaal te zijn. Omwille van de leesbaarheid plaatsen we die in dit stuk echter wel onder de noemer van illegale content.

3 De werking van een digitale gegevensdrager

3.1 Inleiding

Om te kunnen begrijpen hoe een bestand met illegale dan wel legale content kan worden geïdentificeerd of kan worden verwijderd is het van belang om enige kennis te hebben over de werking van een digitale gegevensdrager en de daarop opgeslagen informatie. Hiermee wordt bedoeld op welke wijze bestanden (gegevens) op een gegevensdrager worden opgeslagen en dus niet hoe een gegevensdrager zelf fysiek werkt.

De gegevensdragers beschreven in deze toelichting zijn gebaseerd op de niet vluchtige geheugens. In niet vluchtige geheugens kunnen de gegevens bewaard worden, ook als het geheugen geen spanning meer krijgt (computer wordt uitgezet). Dit in tegenstelling tot vluchtig geheugen (werkgeheugen) van apparaten.

In dit hoofdstuk zullen de volgende onderwerpen aan bod komen:

- Digitale gegevensdragers: de soorten, typen, houdbaarheid en kopieën.
- Bestandssystemen: de werking, typen en eigenschappen
- Bestanden: de soorten, versleuteling en metadata.

3.2 Soorten digitale gegevensdragers

3.2.1 Vormen en methoden van gegevensdragers

Een gegevensdrager is een fysieke plek om informatie op te slaan en bij digitale gegevensdragers bestaat de informatie uit nullen en enen. Er zijn verschillende technieken/manieren om nullen en enen op te slaan. In Tabel 1 zijn diverse vormen van gegevensdragers weergegeven.

Tabel 1: Diverse vormen van gegevensdragers

Vorm	**Techniek opslag**	**Pluspunten**	**Minpunten**	**gemiddelde houdbaarheid**
Harddisk (intern/extern)	Magnetisch	Goedkoop Capaciteit	Bewegende onderdelen	10 jaar*
SSD disks	Elektrisch	Snel Geen bewegende delen Shock proof	Minder lang houdbaar Relatief duur	3** maanden-10 jaar??
USB/Memory cards	Elektrisch	Compact Goedkoop Lang houdbaar	Kleine capaciteit Snel kwijt raken	10.000-1.000.000 keer herschrijven
CD/DVD/Blue ray	Optisch	Compact Lang houdbaar***	Kleine capaciteit Bewegende onderdelen Kort houdbaar***	2-10 jaar
Tapes	Magnetisch	Lang houdbaar	Alleen voor back-up	30 jaar
Floppy's	Magnetisch	Geen	Worden niet meer gebruikt	5-10 jaar
Cloud	Diversen	Grote capaciteit Verspreid over meerdere locaties Overal ter wereld te bereiken	Relatief langzaam Niet in eigen beheer.	onbekend

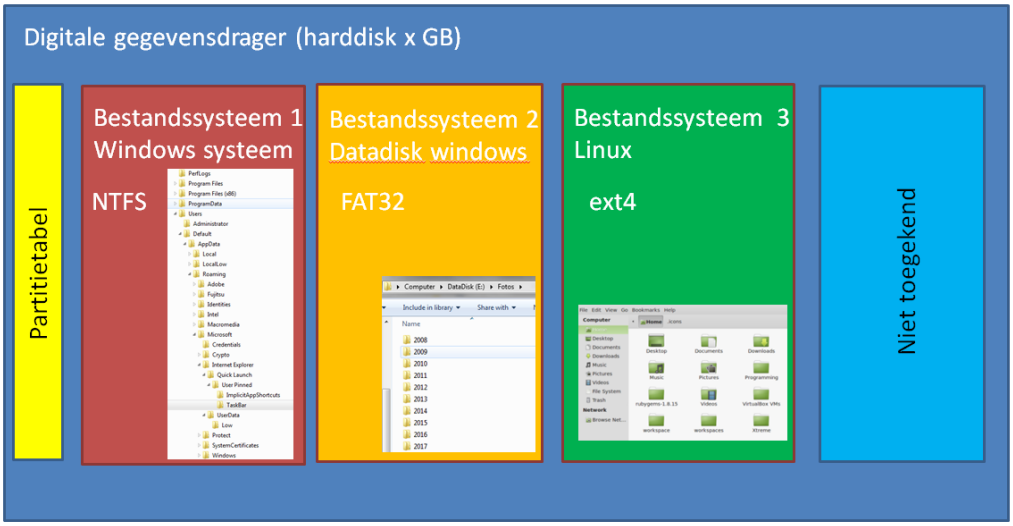
Toelichting:
De vorm, opslagtechniek, plus- en minpunten en gemiddelde houdbaarheid van de meest voorkomende vormen van gegevensdragers.
*afhankelijk van gebruik en omgevingsfactoren
** hangt van temperatuur af
*** afhankelijk van de kwaliteit van de gegevensdrager en de gebruikte techniek

De opslagtechniek (kolom 2) wordt verder niet toegelicht, maar bepaalt voor een groot deel wel hoe gevoelig de gegevensdrager is voor degradatie (ofwel verlies) van de gegevens.
Er bestaan drie natuurkundige technieken: de magnetische die gebruikt wordt voor harddisks, floppy's en tapes. De elektrische die gebruikt wordt voor de modernere SSD harddisks, USB sticks en andere met geheugenchips gemaakte permanente (niet vluchtige[2]) opslag. De laatste techniek is de mechanische die geldt bij cd's, dvd's en blue rays.
In de laatste drie kolommen zijn de plus- en minpunten van de soort van gegevensdrager genoemd samen met een schatting van de levensduur (houdbaarheid) van de gegevensdrager. Hiermee wordt bedoeld hoe lang de gegevens op de gegevensdrager er nog vanaf gehaald kunnen worden zonder gegevensverlies. Bij vele soorten hangt de levensduur erg af van de kwaliteit van de drager en de lokale gebruiksomstandigheden.

[2] Niet vluchtig geheugen bestaat uit geheugenmodules waar de informatie in opgeslagen blijft ook nadat de stroom er vanaf is gehaald (apparaat staat dan uit). Er bestaat ook vluchtig geheugen ook wel RAM geheugen genoemd waarbij de informatie verdwijnt als de stroom wordt uitgeschakeld.

3.3 Bestandssystemen

In figuur 1 is een schematisch voorbeeld gegeven van een digitale gegevensdrager met daarop diverse bestandssystemen. Alle onderdelen in figuur 1 zullen worden besproken.



Toelichting:
Schematische weergave van een harddisk met 3 verschillende delen (partities). Met 3 verschillende soorten van bestandssystemen: NTFS, FAT32 en ext4.

Figuur 1: Voorbeeld van een gegevensdrager (harddisk).

In de *partitietabel* (gele blok figuur 1) van een gegevensdrager worden de locaties van de verschillende partities opgeslagen en bijgehouden. Het is ook mogelijk om via deze partitietabel op te starten vanaf een andere partitie. Bijvoorbeeld Linux (groene blok) opstarten in plaats van Windows (rode of oranje blok). Dit wordt een multi (of dual) boot computer genoemd.

Een *bestandssysteem* is in de eerste plaats een administratie van bestanden op een gegevensdrager. Er bestaan vele soorten van bestandssystemen. Ze komen op alle soorten van gegevensdragers voor zoals op harde schijven van computers, USB-sticks en cd-roms. Zonder een bestandssysteem zijn bestanden niet zichtbaar en niet benaderbaar voor de gebruiker.

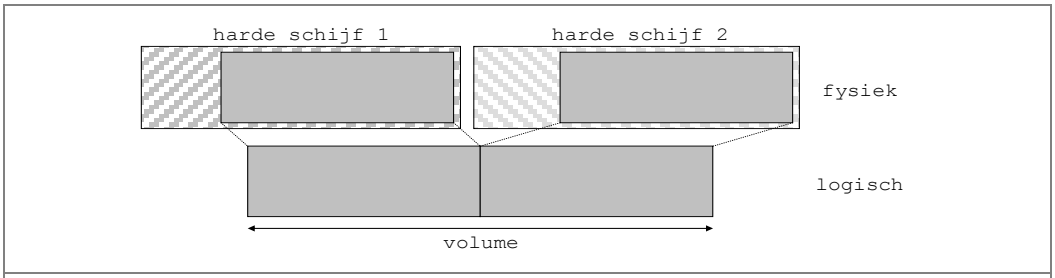
Een bestandssysteem maakt het mogelijk om meerdere bestanden op te slaan binnen één opslagruimte. De basisfuncties in bijna alle bestandsystemen maken het mogelijk om gegevens in bestanden op te slaan, te lezen en te wijzigen. De meeste bestandsystemen bieden ook de functionaliteit om bestanden hiërarchisch en nevenschikkend te groeperen in een mappenstructuur.

In de tweede plaats houdt een bestandssysteem naast de inhoud van bestanden gegevens over bestanden bij: *metadata*. Deze meta-informatie omvat gegevens die voor forensisch onderzoek van belang kunnen zijn, zoals de naam, de grootte van het bestand en wie er gebruikersrechten heeft over het bestand. Daarnaast zijn de datum en tijd waarop een bestand is aangemaakt of voor het laatst is gewijzigd er in te vinden.

Een bestandssysteem staat op de gegevensdrager in een stuk opslagruimte dat daarvoor speciaal is gereserveerd. Deze opslagruimte is een vast stuk op de gegevensdrager. Het kan de volledige opslagruimte van een gegevensdrager betreffen, maar opslagruimten van gegevensdragers kunnen ook worden opgedeeld of gecombineerd. Dit wordt omschreven in de paragraaf over volumes en partities.

3.3.1 Volumes en partities

Een (logisch) volume is een opslagruimte die als één geheel benaderbaar is. Een volume kan enkele harde schijf zijn, maar het is ook mogelijk (delen van) de opslagruimte van één of meerdere fysieke gegevensdragers te combineren tot één geheel, zie figuur 2.



Toelichting:
(Delen van) één of meer fysieke harde schijven kunnen een logisch volume vormen

Figuur 2: (Delen van) één of meer harde schijven vormen volumes

De gangbare oplossing om meerdere harde schijven te combineren tot één aaneengesloten opslagruimte is het zogenaamde *redundante schijfsysteem* (RAID). Vaak worden RAID systemen met gespecialiseerde hardware samengesteld, maar RAID systemen kunnen ook met een softwarematige oplossing gerealiseerd worden. De meeste gangbare moderne besturingssystemen, zoals Microsoft Windows, Apple OS-X en Linux, bieden ook een softwarematig alternatief: de *volumemanager*. Een volumemanager of logische volumemanager beheert de opslagruimten voor een besturingssysteem en biedt veelal de mogelijkheid (delen van) deze opslagruimten te combineren.

Een *volume* kan in meerdere logische delen, partities, verdeeld worden. Een partitietabel geeft een overzicht van de partities in een volume. Soms bevat een volume meerdere partitietabellen, in dat geval is minstens één van deze tabellen altijd vindbaar omdat het zich op een vaste plaats bevindt. Vanuit deze tabel is het mogelijk de andere tabellen te vinden.

Een *partitie* bevat meestal een bestandssysteem of een deel daarvan. Sommige bestandssystemen kunnen over meerdere partities verdeeld worden, maar een partitie bevat niet meer dan één bestandssysteem. Het gebruik van partities maakt het onder andere mogelijk om verschillende soorten bestands- en besturingssystemen naast elkaar binnen één volume te laten bestaan (zie figuur 1).

3.3.2 Indelen van een bestandssysteem

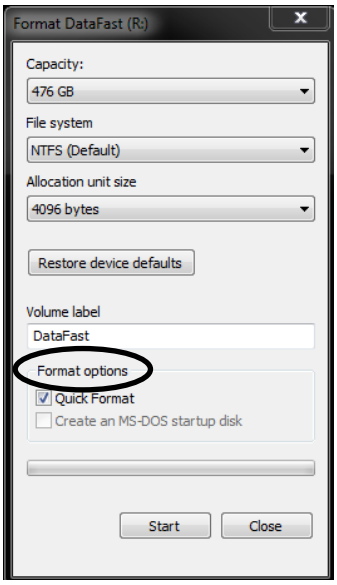
Voordat een bestandssysteem kan worden gebruikt moet de partitie worden ingedeeld of geformatteerd. Gegevens op een gegevensdrager zijn benaderbaar in eenheden met een vaste afmeting: *sectoren*.

Bij een partitie dat een bestandssysteem bevat, bevat de eerste sector van die partitie meestal de informatie die nodig is om het bestandssysteem te interpreteren. Deze sector heet afhankelijk van het soort bestandssysteem volume boot sector,

volume boot record of superblock. De informatie in deze sector is essentieel voor de interpretatie van het bestandssysteem. Daarom bevatten de meeste bestandssystemen één of meerdere kopieën van de bootsector van een volume als reserve, voor het geval de originele sector beschadigd raakt of per ongeluk wordt overschreven.

De overige sectoren van een partitie worden zo ingedeeld en ingericht dat ze ofwel bestanden kunnen opslaan ofwel metadata. Ieder bestandssysteem hanteert hiervoor een eigen indelingsprocedure. Bestandssystemen houden bij welke sectoren in gebruik zijn en welke beschikbaar zijn. Om het ruimtebeslag van deze administratie te beperken houden bestandssystemen dit meestal niet per sector maar per cluster bij. Een cluster bestaat uit een vast aantal sectoren achter elkaar.

Veel besturingssystemen bieden de keuze tot volledig formatteren of snel formatteren. De laatste optie wordt vaak gekozen omdat dit bij grote harddisks veel tijd bespaart. Beide methoden delen de gedeelten van de partitie voor de administratie van het bestandssysteem in of opnieuw in. Bij volledig formatteren worden bestanden meestal volledig gewist. Bij snel formatteren blijven de partitiedelen voor de opslag van bestanden meestal ongemoeid. In dat geval zijn mogelijk nog gegevens van voor het formatteren terug te vinden.



Toelichting:
De optie tot snel formatteren van een Windows harddisk met het NTFS bestandssysteem.

Figuur 3: Snel formatteren van een Windows harddisk

3.3.3 *Opslag van bestanden*

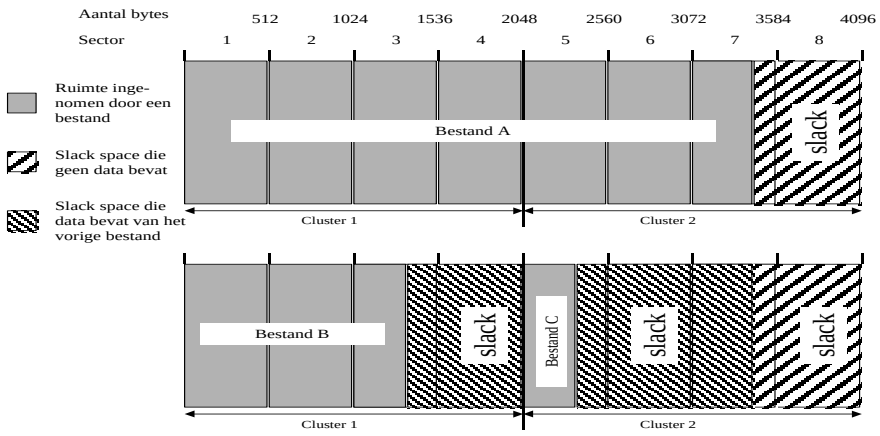
Ieder bestand beslaat één of meerdere clusters in een bestandssysteem. Om ruimte te besparen bevatten sommige bestandssystemen speciale functionaliteit om kleine bestanden zo op te slaan dat niet altijd een heel cluster gereserveerd hoeft te worden. Dat gebeurt bijvoorbeeld door ze op te slaan in gereserveerde ruimten voor kleine bestanden of door kleine bestanden samen te voegen in één cluster.

Wanneer een deel van een bestand gelezen of geschreven moet worden, is het nodig de clusters met dit deel van het bestand te vinden. Het is dus nodig bij te houden welk deel van een bestand in welke clusters is opgeslagen.

Bestandssystemen gebruiken verschillende technieken om een dergelijke administratie bij te houden als onderdeel van de meta-informatie.
Ongebruikte clusters voor bestandsopslag heten niet-gealloceerde clusters. Ongebruikte clusters samen heten niet-gealloceerde ruimten. Het bestandssysteem wijst een nieuw bestand één of meer niet-gealloceerde clusters toe, die daardoor gealloceerd worden. Als een bestand echter wordt verwijderd, komen de clusters waarin het staat vrij voor hergebruik.

> Het verwijderen leidt tot wijzigingen in de interne administratie van het bestandssysteem. Het betekent niet noodzakelijkerwijs dat de inhoud van het bestand daadwerkelijk wordt verwijderd. Als de inhoud van de betreffende clusters niet expliciet gewist wordt, blijft de inhoud intact totdat de betreffende clusters worden (her-)gebruikt door andere bestanden.

De omvang van een bestand is zelden precies gelijk aan de grootte van één of meer clusters. Daarom zal het laatste door het bestand gebruikte cluster bijna altijd deels ongebruikt blijven. De ongebruikte ruimte in een cluster heet *slack space*. *Slack space* kan gegevens bevatten uit bestanden die zijn verwijderd en later deels zijn overschreven. In deze gegevens kan zich interessante informatie bevinden. Met forensische software kan dit toegankelijk gemaakt worden. In *slack space* ontbreekt wel altijd het begin van de overschreven bestanden. Dit bemoeilijkt de interpretatie omdat juist het begin van een bestand vaak de benodigde informatie bevat die nodig is voor een goede of volledige interpretatie.



Toelichting:
Schematische weergave van het ontstaan van *slack space* en de veranderingen hierin bij het verwijderen van bestanden (zie uitleg tekst).

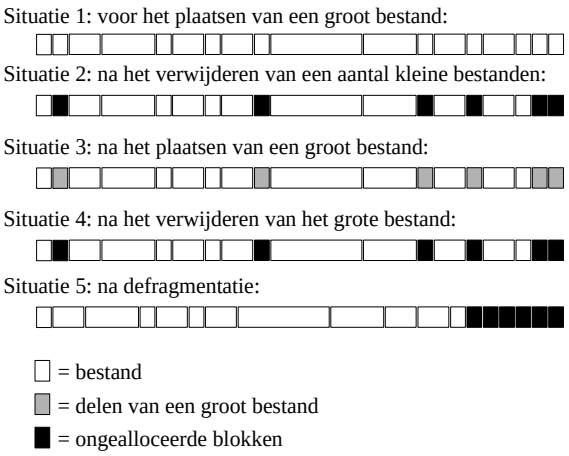
Figuur 4: *Slack space* binnen een cluster

Figuur 4 bevat twee situaties. In het bovenste deel van deze figuur staat bestand A op een schone harde schijf. De sectorgrootte is 512 bytes en de clustergrootte is vier sectoren (2.048 bytes). Bestand A is 3.400 bytes groot en beslaat twee clusters. De ongebruikte ruimte in het tweede cluster is *slack space*. Op een schone harde schijf bevat deze *slack space* geen gegevens. De tweede situatie in de figuur beschrijft dezelfde clusters, alleen is bestand A nu verwijderd. De vrijgekomen ruimte is gebruikt voor het opslaan van twee kleinere bestanden, B en C. B van 1200 bytes staat in het eerste cluster. Omdat dit bestand het cluster niet vol maakt, blijven er 848 bytes over als *slack*. Deze *slack* ruimte bevat dus nog gegevens van

bestand A. Hetzelfde geldt voor bestand C van 300 bytes in het tweede cluster; ook dit bevat nog gegevens uit bestand A.

3.3.4 Fragmentatie

De meeste bestandssystemen proberen bestanden in opeenvolgende clusters op te slaan, maar dit is niet altijd mogelijk. Als er geen aaneengesloten ruimte groot genoeg is om een bestand op te slaan, kan het in delen verspreid over het bestandssysteem worden opgeslagen. Een verspreid opgeslagen bestand kan ook ontstaan door een bestaand bestand te vergroten. Op een gegeven moment is de oorspronkelijke aaneengesloten ruimte waarin het bestand zich bevindt niet groot genoeg meer en moet elders ruimte worden gevonden.
Als een magnetische harde schijf veel van dergelijke verspreide bestanden bevat is er sprake van fragmentatie en een gefragmenteerde harde schijf.
Defragmentatieprogramma's dienen om bestandsfragmenten logisch te herschikken. Ze verplaatsen de clusters op de harde schijf zodat ze zoveel mogelijk in de juiste volgorde achter elkaar komen (zie Figuur 5). Door defragmenteren gaat veel 'verwijderde' informatie definitief verloren. Clusters worden gekopieerd naar plekken waar ze mogelijk interessante informatie overschrijven.

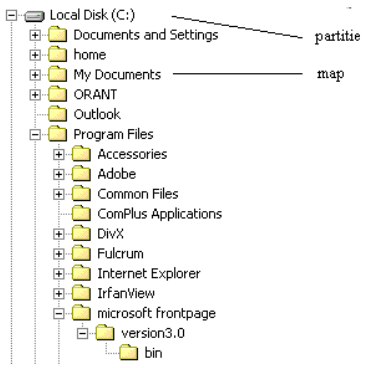


Toelichting:
Vijf situaties van fragmentatie naar defragmentatie:
1. De originele situatie;
2. Een aantal kleine bestanden is verwijderd om plaats te maken voor één groot bestand;
3. Eén groot bestand heeft de plaats ingenomen van de verwijderde kleine bestanden;
4. De niet-gealloceerde ruimte van het bestandssysteem is gefragmenteerd (na verwijderen van het grote bestand);
5. Hetzelfde bestandssysteem na defragmentatie.

Figuur 5: Fragmentatie en defragmentatie van een harde schijf

3.3.5 Mappen

Een map, folder of directory is een lijst met verwijzingen naar bestanden en andere mappen. Mappen hebben tot doel om bestanden logisch te groeperen. Doordat mappen ook referenties naar andere mappen kunnen bevatten, ontstaat een hiërarchische structuur, zoals in Figuur 6.



Toelichting:
Voor beeld van een mappenstructuur met mappen en submappen op de C partitie met een Windows installatie.

Figuur 6: Een hiërarchische mappenstructuur

3.3.6 *Verwijderen en terughalen van bestanden*

Een bestand is voor een eindgebruiker alleen toegankelijk als er vanuit de *hoofdmap* een pad door de mappenhiërarchie naartoe bestaat. Wanneer in geen enkele *map* een verwijzing naar een bestand voorkomt is het bestand voor een gebruiker niet meer toegankelijk. Daarom is het verwijderen van de verwijzing naar een bestand voldoende om een bestand te 'verwijderen'. Het bestand wordt hierdoor uit de inhoudsopgave verwijderd en daarmee onzichtbaar voor de eindgebruiker. Daarna geeft het bestandssysteem de clusters waarin het bestand is opgeslagen vrij om te overschrijven.

Daarnaast is een grondiger wijze mogelijk om bestanden te verwijderen: daadwerkelijk overschrijven of wissen, ook wel *wiping* genoemd. Gegevens zijn hierna niet meer terug te vinden. Sommige besturingssystemen hebben ingebouwde functionaliteit om bestanden te *wipen* en er bestaan ook aparte *wipe*-programma's.

Forensische en vrij verkrijgbare software is vaak in staat verwijderde, maar nog niet overschreven bestanden terug te vinden en weer zichtbaar te maken. Bij gedeeltelijke overschrijving kunnen soms delen van verwijderde bestanden worden teruggevonden. Meta-informatie wordt gescheiden van het bestand zelf opgeslagen. Daardoor kan het zijn dat slechts één van beide wordt teruggevonden tenzij de gegevens zijn *gewiped* dan is er niets meer terug te vinden.

Zie voor meer informatie over het verwijderen van bestanden hoofdstuk 5.

3.3.7 *Typen bestandssystemen*

Bestandssystemen zijn er in vele soorten. Ze zijn vaak ontworpen door de ontwikkelaars van besturingssystemen zoals Microsoft Windows, Apple OS-X en Linux. Veel producenten hebben meerdere bestandssystemen ontwikkeld en laten gebruikers kiezen in welk bestandssysteem ze hun bestanden willen opslaan. Tabel 2 geeft een overzicht van populaire bestandssystemen. Bestandssystemen horen in het algemeen bij een bepaald *besturingssysteem* of een generatie daarvan. Toch zijn veel bestandssystemen ook door besturingssystemen van andere producenten benaderbaar. Zo kan vrijwel elk besturingssysteem bestanden in een FAT-bestandssysteem benaderen.
De (historische) lijst met bestaande bestandssystemen is uiteraard veel langer. Zie voor een actuele impressie op
`https://en.wikipedia.org/wiki/List_of_file_systems`.

Bestandssysteem	Ontwikkeld voor	Sinds
FAT	MS-DOS	1981
NTFS	Windows	1993
UFS	Unix	1969
Ext2, Ext3, Ext4	Linux	1993, 2001, 2008
HFS, HFS+	Mac OS	1985, 1998
APFS	MacOS en iOS	2017
ISO 9660	Optische media	1988

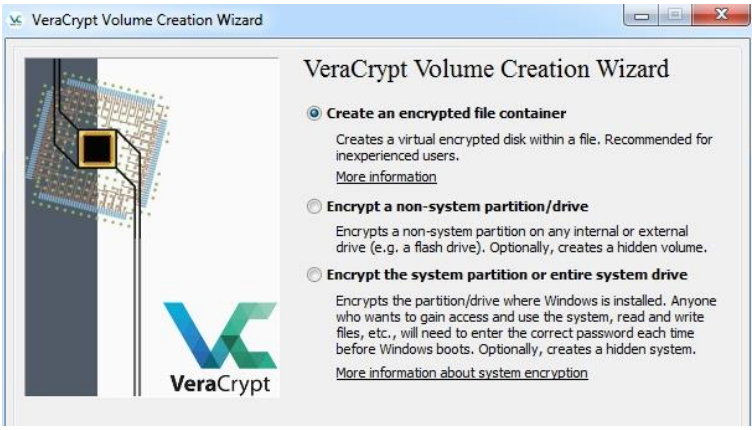
Tabel 2: Oorsprong van bestandssystemen

3.3.8 Bijzondere bestandssystemen

De meeste bestandssystemen schrijven gegevens direct in leesbare vorm naar het opslagmedium, er zijn echter uitzonderingen: comprimerende bestandssystemen en versleutelende bestandssystemen.

Comprimerende bestandssystemen maken gebruik van compressietechnieken om bestandsdata in zo min mogelijk ruimte op te slaan. Gegevens worden eerst gecomprimeerd en dan pas opgeslagen, dit bespaart ruimte ten koste van verwerkingstijd. Een bijeffect van compressie is dat het problematisch kan zijn om teruggevonden gedeelten van bestanden te interpreteren. Dit probleem wordt veroorzaakt doordat het veelal noodzakelijk is om complete clusters of zelfs bestanden te hebben om gegevens te kunnen decomprimeren.

Versleutelende bestandssystemen versleutelen gegevens om deze alleen voor bevoegden toegankelijk te maken. Bij versleuteling of encryptie zijn de gegevens niet zonder de juiste sleutel toegankelijk. In de meeste gevallen voert een gebruiker de sleutel in als wachtwoord, maar het is ook mogelijk dat een sleutel digitaal aangeboden wordt (bijvoorbeeld met een USB stick of smart card). Sommige bestandssystemen hebben functionaliteit ingebouwd om (individuele) bestanden versleuteld op te slaan. Een alternatief voor ingebouwde versleuteling is de zogeheten *versleutelde container*. Dat is een stuk opslagruimte, zoals een harde-schijfpartitie, dat in zijn geheel versleuteld is. Omdat containers in hun geheel versleuteld worden, is binnen een container meestal een niet-versleutelend bestandssysteem in gebruik. Sommige besturingssystemen hebben ingebouwde functionaliteit om versleutelde containers aan te maken en te gebruiken. Andere kennen speciale programma's hiervoor.



3.4 Kopieën van gegevensdragers

In forensische onderzoeken worden in beginsel nooit de fysieke gegevensdragers zelf onderzocht maar de digitale kopieën ervan. De reden hiervoor is dat op deze

wijze gegarandeerd kan worden dat de gegevensdragers niet of zo weinig mogelijk worden veranderd door het onderzoek zelf.
Er zijn afhankelijk van het soort en type gegevensdrager diverse mogelijkheden tot het maken van digitale kopieën van deze gegevensdragers.

3.4.1 Eén-op-één kopie (fysieke kopie)

Digitale kopieën voor forensisch gebruik bevatten bij voorkeur buiten alle inhoudelijke informatie ook alle metadata en niet toegewezen data. Het beste is dan om een exacte kopie van de gegevensdrager te maken: een zogenaamde één-op-één kopie, ook wel een fysieke kopie (Engels: image) genoemd. Hierbij worden alle nullen en enen van de gegevensdrager gekopieerd. Fysieke kopieën bevatten dezelfde informatie als het origineel en dat is controleerbaar met behulp van bestandskenmerken (*hashes,* zie paragraaf 3.5.5).

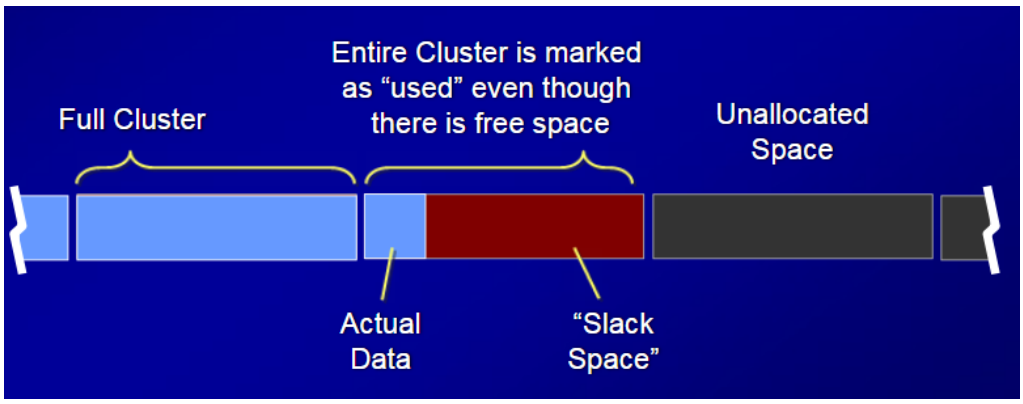
3.4.2 Logische kopie

In sommige gevallen kan er geen fysieke kopie gemaakt worden van een gegevensdrager; bijvoorbeeld wanneer het apparaat is beveiligd of vercijferd. Maar zelfs als er toegang tot het apparaat is kan er doordat de gebruiker te weinig rechten heeft (geen *root access*) geen toegang worden verkregen tot bepaalde bestanden. Dit laatste komt vaak voor bij mobiele telefoons.
Bij mobiele telefoons moet er dan genoegen worden genomen met een zogenaamde logische kopie. De logische kopie bestaat er in verschillende variaties. Soms is het bijvoorbeeld alleen maar mogelijk om de gebruikersgegevens van een mobiele telefoon te kopiëren, maar in andere gevallen kunnen ook alle bestanden van het bestandssysteem worden gekopieerd. Dit wordt dan een bestandssysteem kopie genoemd. Deze kopie kan veel extra (log) gegevens bevatten. De locatie waar systeembestanden worden opgeslagen kan ook worden gebruikt om bepaalde bestanden te verbergen. De mogelijkheden van het maken van een logische kopie hangt af van zowel soort en type telefoon als van de gebruikte software waarmee de kopie wordt gemaakt.

3.4.3 Carven in 'unallocated space' en in 'slack space'

Het zoeken naar gewiste data (Eng: *carven*) in de niet toegekende[3] (Engels: *unallocated*) ruimte of *slack space* op een opslagmedium is alleen mogelijk als er een één-op-één kopie wordt gebruikt.
Bij het wegschrijven van gegevens op een harddisk worden deze gegevens weggeschreven in clusters. Als een bestand net iets groter is dan een cluster dan wordt een volgend cluster gebruikt. De resterende ruimte in een cluster wordt *slack space* genoemd (rode stuk in Figuur 7). De ruimte met compleet lege of niet toegewezen clusters wordt de *unallocated space* genoemd. Deze structuren zijn alleen beschikbaar bij een één-op-één kopie omdat dan alle nullen en enen van het hele opslagmedium worden gekopieerd tegenover alleen de bestanden (licht blauwe blok in Figuur 7) bij een logische kopie.

[3] Niet toegekende ruimte, dit is ruimte op een gegevensdrager die niet is toegewezen aan aanwezige bestanden van een bestandssysteem. Dit is de zogenaamde 'lege' ruimte op een gegevensdrager. In de meeste gevallen is deze ruimte echter niet echt leeg.



Toelichting:
Weergave van verschil tussen *slack space* en *unallocated space*. Files worden per cluster opgeslagen. Resterende ruimte in een cluster wordt *slack space* genoemd. Compleet lege clusters zijn *unallocated space*.

Figuur 7: verschil *slack* en *unallocated space*.

Het is soms wel mogelijk om te *carven* binnen bijvoorbeeld databasebestanden waar soms nog gewiste records binnen het bestand zelf aanwezig kunnen zijn. Hier is dan geen fysieke kopie voor nodig.

3.5 Bestanden

3.5.1 Soorten van bestanden

Wat is een bestand? Volgens een definitie is een computer bestand een bij elkaar horende verzameling van digitale gegevens, die door een elektronisch apparaat onder één naam kan worden aangesproken. Voorbeelden van een bestand zijn een Word-bestand, een PDF-bestand of een afbeelding. Een bestand heeft een bestandsnaam wat bestaat uit een hoofdbestandsnaam en een extensie. Bijvoorbeeld *rapport.docx, bijlage.pdf of IMG0345.JPG*. De extensies van de bestandsnamen geven aan welk type /soort bestand het is. De digitale gegevens in een bestand bestaan uit eenheden die bits genoemd worden. Een bit is een nul of een één. Een groep van 8 bits wordt een byte genoemd. Onderstaande tabel geeft de naamgevingen van oplopende groottes.

1 byte	=	8 bits
1 kilobyte	=	1.000 bytes
1 megabyte	=	1.000.000 bytes (1.000 kB)
1 gigabyte	=	1.000.000.000 bytes (1.000 MB)
1 terabyte	=	1.000.000.000.000 bytes (1.000 GB)
1 petabyte	=	1.000.000.000.000.000 bytes (1.000 TB)

Typische bestandsgroten zijn enkele kB voor een e-mail, een Word document enkele 100-en kB en een afbeelding enkele MB. Om te illustreren hoeveel informatie een te analyseren zaak kan bevatten het volgende voorbeeld. Als een zaak ongeveer 4 TB (terabyte) aan informatie bevat en je zou deze informatie uitprinten dan kun je ongeveer 8 kB op 1 A4'tje kwijt. In een pak papier zitten 500 A4'tjes, in een doos zitten vijf pakken papier, er passen twintig dozen op een pallet en er kunnen 33 pallets in een vrachtwagen. Dit zou resulteren in een file van vijf kilometer aan vrachtwagens als je die 4 Terabyte zou uitprinten op A4'tjes. Standaard hebben computers op dit moment ergens tussen de 1 en 4 Terabyte aan opslagcapaciteit. Mobiele devices hebben per toestel ergens tussen de 16 en 512 gigabyte aan

opslagruimte. Bij een inbeslagname van een aantal laptops, PC's en mobiele telefoons wordt de totale hoeveelheid aan data snel meerdere terabytes.

3.5.2 Afbeeldingen en filmpjes

Een veel voorkomende vorm van bestanden zijn afbeeldingen en filmpjes. Het grootste deel wat betreft omvang en aantal van de inhoud op mobiele telefoons en computers zijn afbeeldingen en filmpjes. Afbeeldingen zijn er in vele formaten. Deze formaten verschillen in de manier waarop de afbeeldingen of filmpjes worden opgeslagen. Ook zijn er per formaat diverse manier om ruimte te besparen door middel van compressietechnieken. Alle formaten hebben standaard hun eigen extensies. De meest voorkomende extensies van afbeeldingen zijn bijvoorbeeld *JPG, BMP, TIFF, PNG* en *GIF*. Voor videobestanden zijn dit bijvoorbeeld *MP4, AVI, WMV*, *MOV* en *DIVX.* De bestandsgroottes kunnen variëren van enkele KB (klein GIF-plaatje) tot meerdere GB (hoge resolutie films in DIVX). Een uitgebreider overzicht van deze formaten is gegeven in bijlage 3.

3.5.3 Software

Remote Acces Tools (RAT), ransomware, malware, DDoS-pakketten zijn voorbeelden van software die kan worden aangetroffen op een gegevensdrager. Soms staan ze op zeer goed bereikbare locaties op de gegevensdrager maar soms staan ze ook opgeslagen in zip-bestanden of zogenaamde cryptocontainers (zie paragraaf 4.3.3). De diversiteit in deze soort software is zeer groot en daarom ook niet altijd even gemakkelijk om als zodanig te herkennen. Er bestaan diverse databases (Virustotal, NIST ect.) waarin schadelijke software wordt geregistreerd. Hiermee is het mogelijk om sommige schadelijke software te herkennen maar zeker niet alle schadelijke software wordt hiermee ondervangen. In hoofdstuk 4 zal verder ingegaan worden op methodieken waarmee bestanden (schadelijke, illegaal en legaal) geïdentificeerd kunnen worden.

3.5.4 Gegevensbestanden

Als laatste categorie bestanden met informatie die in rechtstreeks verband staat met strafbare feiten of die door een strafbaar feit zijn verkregen. Een aantal veel voorkomende voorbeelden van deze gegevens in strafzaken is:

- de contactgegevens van 'online' medeverdachten of leveranciers of afnemers van strafbare content of producten,
- de inloggegevens van internetomgevingen waar over de strafbare feiten gecommuniceerd wordt,
- (seksuele) chats met minderjarigen of 'gelijkgestemden',
- sporen van strafbaar materiaal, zoals bestandsnamen van kinderpornografische afbeeldingen,
- gegevens of bestanden die toegang geven tot cryptocurrency,
- buitgemaakte creditcard of inloggegevens van personen.

Voor al deze gegevens geldt dat deze niet op een standaard manier worden opgeslagen. De gegevens kunnen bijvoorbeeld in tekstbestanden (.txt), Excelbestanden (.xls(x)) of databasebestanden worden opgeslagen. Ook hiervoor geldt dat de grote diversiteit aan soorten bestanden het tot een uitdaging maakt om ze allemaal te kunnen identificeren.

3.5.5 Bestandskenmerken (hash waarden) [4]

Om te controleren of de inhoud van een kopie van digitaal materiaal nog identiek is aan het origineel wordt gebruik gemaakt van de zogenoemde bestandskenmerken. Deze bestandskenmerken kunnen hiernaast ook gebruikt worden voor classificatie en identificatie van bestanden.

[4] Tekst uit de NFI vakbijlage Forensisch gebruik van bestandskenmerken en bijbehorende hashalgoritmen.

De bestandskenmerken worden uitgerekend met programma's die gebruik maken van specifieke hiervoor geschikte rekenkundige methoden, *hashalgoritmen* genaamd.

De (Engelse) termen *hash*, *hash value* en *hashwaarde* zijn veelgebruikte synoniemen voor de term bestandskenmerk. Hiernaast wordt een bestandskenmerk ook wel een 'digitale vingerafdruk'[5] van een bestand genoemd. Een voorbeeld van een hash waarde (SHA-256) is hieronder gegeven:

```
b628 0802 dfa9 b18a 8f0e 3069 3a65 9636
f417 7412 3d56 b61b 0a67 af76 a939 6ee5
```

Een bestandskenmerk is een compacte representatie van de *binaire* inhoud van digitaal materiaal, maar verschaft verder geen informatie over de door een persoon *interpreteerbare*[6] inhoud van het materiaal. Waar bijvoorbeeld voor een persoon de inhoud van twee afbeeldingen gelijk zijn (dezelfde interpreteerbare inhoud bevatten) kunnen de bestanden binair gezien verschillend zijn (bijvoorbeeld wanneer het opslagformaat van de afbeeldingen verschilt). Waar in deze paragraaf over de inhoud van een bestand wordt gesproken word de binaire inhoud bedoeld, tenzij anders vermeld.

Een digitaal bestand bestaat binair gezien uit een reeks nullen en enen (*bits*). Het aantal en de volgorde van deze nullen en enen bepalen de inhoud van het bestand en zijn daarmee kenmerkend voor dat bestand. Door gebruik te maken van een hashalgoritme worden de specifieke nullen en enen van een bepaald bestand omgezet in een veel eenvoudigere en compactere notatie. Een hashalgoritme is een eindige (en meestal complexe) reeks (mathematische) instructies waarmee op basis van de inhoud van het oorspronkelijke bestand een reeks nullen en enen van vaste lengte wordt gegenereerd. Deze gegeneerde reeks is kenmerkend voor de inhoud van het oorspronkelijke bestand en wordt dan ook aangeduid als het 'bestandskenmerk' van het oorspronkelijke bestand.
Hoewel hier vooral over (digitale) bestanden wordt gesproken geldt het (forensisch) gebruik van bestandskenmerken voor al het digitale materiaal, dus bijvoorbeeld ook voor de volledige inhoud (één-op-één-kopie, veiliggestelde gegevens, Engels: *image*) van een digitale gegevensdrager.

Zoals eerder aangegeven moet men er zich van bewust zijn dat twee bestanden er voor een gebruiker hetzelfde uit kunnen zien, maar wanneer de bestanden op bit-niveau verschillend zijn deze met extreem grote waarschijnlijkheid andere bestandskenmerken zullen hebben.
Voor een uitgebreidere uitleg over bestandskenmerken zie bijlage 1.

3.5.6 Metadata van bestanden

Alle bestanden bevatten naast hun inhoud ook zogenaamde metadata. Deze metadata bevat informatie over het bestand. De metadata van bijvoorbeeld een bestand in Windows kan worden bekeken door rechts te klikken en dan naar *properties* te gaan. Hier kan allerlei informatie worden gegeven over het bestand.

[5] Deze vergelijking geldt alleen voor het overeenkomstige algemene gevoel van onderscheidbaarheid van een vingerafdruk dan wel bestandskenmerk. Strikt genomen loopt deze vergelijking mank: bijna gelijke bestandskenmerken horen (extreem waarschijnlijk) bij bestanden met compleet verschillende inhoud en zijn dan ook zeer onderscheidend te noemen. Bijna gelijke (*matchende*) vingerafdrukken zijn juist niet onderscheidend.

[6] Voorbeelden van door een persoon interpreteerbare bestanden zijn digitale tekstdocumenten en afbeeldingen (visueel) en digitale audiobestanden (auditief).

De meest interessante gegevens zijn:

- soort bestand,
- datum-tijdstempels van creatie, gewijzigd en bekeken.
- de opslaglocatie,
- de eigenaar,
- bestandsgrootte,
- herkomst van de data,
- eigenschappen van het bestand (*read-only, hidden*)

Afbeeldingen en videobestanden hebben vaak uitgebreidere metadata. Deze wordt Exif (Exchangeable image file format) data genoemd. Typische velden die hier in voorkomen zijn:

- Datum en tijd van de opname en van de laatste bestandswijziging.
- Merk en model van de camera.
- Naam van de eigenaar van de camera.
- Camera-instellingen zoals belichtingstijd, diafragmagetal (F-getal), diafragma, brandpuntsafstand.
- Locatie-gegevens zoals de breedtegraad en lengtegraad.

Er zijn echter nog veel meer informatievelden die kunnen voorkomen in de Exif of metadata van bestanden. Er bestaan allerlei tools waarmee deze informatie kan worden verkregen uit de bestanden.

De belangrijkste metadata binnen een forensisch onderzoek zijn de verschillende tijden, locaties, herkomst en eigenaar van de bestanden. Met behulp van deze gegevens kan worden achterhaald waar bestanden vandaan komen. Zijn ze bijvoorbeeld gedownload of gekopieerd van een externe bron of ontvangen via e-mail.

Analyse van tijdstempels van bestanden is niet vanzelfsprekend. Tijdstempels zijn belangrijk in forensisch onderzoek naar de herkomst en geschiedenis van bestanden. In het doel van deze toelichting voert het wat ver om hier heel diep technisch op in te gaan. De interpretatie van de diverse aanwezige tijdstempels is afhankelijk van het bestandssysteem (FAT, NTFS ect) en van het besturingssysteem (Windows, Linux enz.). Zie voor meer informatie in Carrier[7] en Knutson[8]. Tijdstempels kunnen gemanipuleerd worden (*Eng: timestomping*) waardoor bijvoorbeeld het tijdstip waarop een bestand is gedownload of voor het laatst is gewijzigd kan worden aangepast. De analyse van tijdstempels en eventuele manipulatie is regelmatig een element in forensisch bestandsonderzoek. Er bestaan diverse manieren om tijdstempels van bestanden aan te passen maar deze aanpassing is vaak ook weer te detecteren. Binnen deze toelichting zal hier inhoudelijk niet verder op ingegaan worden. Voor meer informatie hierover zie Knutson[8].

3.5.7 Staat van bestanden

Bestanden hebben nog een belangrijke eigenschap die van invloed is of een bestand geïdentificeerd of gevonden kan worden. Deze eigenschap wordt hier omschreven als de ‘staat’ van het bestand.

De meest voorkomende staat is dat het bestand normaal toegankelijk is voor alle gebruikers. Het kan echter ook voorkomen dat bestanden verborgen zijn (*Eng:*

[7] Carrier, B. File System Forensic Analysis, ISBN: 978-0-321-26817-4, 2005.
[8] Knutson, T., Filesystem Timestamps: What Makes Them Tick?, https://www.sans.org/reading-room/whitepapers/forensics/filesystem-timestamps-tick-36842 (laatst bezocht 19-7-2018).

hidden). Deze bestanden zijn over het algemeen niet zichtbaar voor gebruikers, maar met de juiste instellingen van het computersysteem zijn deze bestanden wel zichtbaar voor de gebruiker.

Bestanden kunnen gecomprimeerd zijn. Dit komt meestal voor om de bestandsgrootte te verkleinen voor transport of voor back-up toepassingen. De meest voorkomende compressie methoden van bestanden zijn ZIP of TAR. Afhankelijk van de compressie methoden of software wordt de metadata van deze bestanden wel of niet overgenomen van de originele bestanden. Gecomprimeerde bestanden zijn met behulp van de juiste software meestal wel toegankelijk, tenzij de gecomprimeerde bestanden zijn beveiligd met een wachtwoord.

Bestanden of een set van bestanden kunnen ook versleuteld zijn. Dit wordt encryptie van de bestanden genoemd. Er bestaan diverse vormen en manieren van encryptie waarbij sommige methoden gemakkelijk te kraken zijn en sommige (bijna) niet. Berucht zijn de zogenoemde cryptocontainers waar vaak een hele verzameling aan (strafbaar) materiaal is ondergebracht, of nog weer meer cryptocontainers.
De laatste besproken staat is de verwijderde staat. Bij het gewoon verwijderen van een bestand in bijvoorbeeld Windows wordt niet de inhoud van het bestand weggegooid maar alleen de verwijzing naar het bestand in de index. Hier zal nog uitgebreider op worden ingegaan in hoofdstuk 5.

4 Identificatie van bestanden

4.1 Inleiding

Als bestanden of gegevens dienen te worden teruggegeven aan beslagenen of derden, dan dienen eerst de juisten bestanden geselecteerd te worden.
In dit hoofdstuk worden methoden voor de selectie of identificatie van bestanden besproken en de uitdagingen die bij de selectie van (il)legale bestanden bestaan.

Er bestaan diverse methoden waarop bestanden al dan niet automatisch kunnen worden geselecteerd. In de eerste paragraaf zullen diverse bestaande selectiemethoden van bestanden worden besproken, gevolgd in de tweede paragraaf door de uitdagingen die er zijn bij de identificatie van alle soorten van bestanden (zowel legaal als illegaal). Een rekenvoorbeeld volgt in paragraaf drie. De laatste paragraaf gaat over de mogelijkheden van het uitsluiten van de aanwezigheid van illegale content (bij teruggave van bestanden).

4.2 Beschrijving van bestaande methodes en benodigdheden

4.2.1 Handmatige selectie van niet illegale content (handmatige whitelisting)

Methode
Een van de meest logische methoden voor het selecteren van bestanden is de handmatige *whitelisting* methode: er wordt een lijst gemaakt met niet illegale bestanden die teruggegeven kunnen worden. Hierbij is het van belang dat er kennis bestaat over welke bestanden geselecteerd moeten worden en waar deze zich (ongeveer) bevinden. Hierbij kan gedacht worden aan foto- en filmmateriaal, aan (privé)documenten, administraties of andere voor personen belangrijke digitale gegevens. Deze bestanden kunnen met behulp van een bestandsbrowser (bijv. Windows Explorer) worden gezocht, geselecteerd en naar een externe gegevensdrager worden gekopieerd. In het geval van teruggeven van bestanden in een strafzaak zullen de geïdentificeerde bestanden voorafgaand aan teruggave moeten worden beoordeeld op (verborgen) illegaliteit zodat er geen bestanden met strafbare inhoud worden terug gegeven. Hiervoor zullen de bestanden leesbaar en niet versleuteld moeten zijn. Voor de beoordeling zal een persoon moeten worden ingezet die beschikt over de nodige kennis om het onderhavige materiaal te kunnen beoordelen.

Voordelen
- Handmatige selectie is geschikt voor teruggave privébestanden, want de meeste privébestanden hebben vaak persoonlijke bestandsnamen die met automatische methoden niet gevonden worden.

Nadelen
- Handwerk is een zeer tijdrovend en kostbaar proces.
- Wordt er wegens efficiëntie steekproefsgewijs op illegale content wordt gecontroleerd dan is de kans op teruggave van illegale content een stuk groter.
- Er is een expert nodig die de vormen van illegaliteit kan herkennen en beoordelen en die verborgen illegaliteit kan herkennen.

Deze methode is het meest bruikbaar als het om een beperkt aantal en eenvoudig te identificeren bestanden gaat.

4.2.2 Handmatige en automatische selectie van illegale content (blacklisting)

Methode
Blacklisting is het tegenovergestelde van whitelisting: er wordt een lijst opgesteld van (in dit geval) bestanden die niet mogen worden teruggeven wegens illegale content. De vraag is hoe bestanden op deze lijst terecht kunnen komen. Voor afbeeldingen met kinderpornografische (KP) content bestaat een zogenaamde hash-lijst met *hashes* van geïdentificeerde KP bestanden. Deze *hashes* kunnen vergeleken worden met de *hashes* van de bestanden op een gegevensdrager. Komen ze overeen dan kan een bestand aan de blacklist worden toegevoegd. Een zelfde soort hash-lijst bestaat van virus, malware, ransomware bestanden. Ook zouden er handmatig bestanden aan de blacklist kunnen worden toegevoegd, bijvoorbeeld bestanden die tijdens een strafzaak zijn geïdentificeerd als illegaal. Vervolgens kunnen alle bestanden op een gegevensdrager worden vergeleken met de blacklist en als ze overeenkomen kunnen ze worden gewist (zie hoofdstuk 5). De resterende, niet gewiste bestanden, worden dan teruggeven aan de eigenaar. Bij deze methode kan de originele gegevensdrager na het wissen van de geïdentificeerde illegale bestanden worden teruggegeven aan de eigenaar.

Voordelen
- Proces kan grotendeels geautomatiseerd worden en kan daardoor tijd en kosten effectief worden ingezet.
- Eigenaar krijgt zo in totaal meer bestanden terug, in de structuur waarin ze opgeslagen stonden. Ook de gegevensdager zelf kan teruggegeven worden.

Nadelen
- Doordat de lijsten met *hashes* van bestanden met illegale content nooit volledig kunnen zijn omdat er steeds nieuw materiaal komt, is het geen garantie dat alle bestanden met illegale content worden geïdentificeerd.
- Illegale bestanden kunnen verborgen zijn (zie paragraaf 4.3). Als ze niet gevonden worden, worden ze met deze methode teruggegeven.
- Er bestaat een risico dat de geïdentificeerde illegale bestanden niet op de juiste wijze worden gewist en terug gehaald kunnen worden (zie hoofdstuk 5).
- De politie heeft in een strafrechtelijk onderzoek niet de taak alle illegale bestanden te vinden. Het onderzoek kan gestopt worden als voor de tenlastelegging voldoende onderbouwing is. De kans is dus zeer reëel dat er niet geïdentificeerde illegale bestanden zijn, met name bij KP-zaken.

Deze methode is het meest bruikbaar als het gaat om een groot aantal terug te geven bestanden en kan worden toegepast als de a priori kans op aanwezigheid van illegale bestanden klein is (bijvoorbeeld bij een harddisk van de computer van de bejaarde buurvrouw).

4.2.3 Automatische selectie van legale content (automatische whitelisting)

Methode
De automatische *whitelisting* methode is precies het omgekeerde van de automatische blacklist methode. Hierbij worden alle bekende en legale bestanden automatisch vergeleken met de inhoud van een gegevensdrager. Er bestaan hiervoor lijsten van alle standaard bestanden van bijvoorbeeld Windows en andere software pakketten. Deze stap kan in grote mate geautomatiseerd worden maar is niet geschikt voor het automatisch identificeren van gebruikersdata zoals foto’s, filmpjes en privédocumenten. De informatie van deze gegevens moet dan handmatig worden toegevoegd aan de *whitelist* en dat kost weer veel tijd (zie

paragraaf 4.2.1). Groot nadeel van de methode is dat de eigenaar van de gegevensdrager zijn privébestanden (zoals foto's en dergelijke) niet terugkrijgt en het dus voor de omschreven doeleinden ongeschikt is.

Voordelen

- Er worden geen bestanden met illegale content teruggeven.
- Het proces verloopt automatisch.

Nadelen

- De eigenaar van de gegevensdrager heeft niets aan deze methode omdat de privébestanden niet (automatisch) worden teruggegeven.

4.2.4 Datum-tijdstempelselectie

Voor zover kan worden vastgesteld wanneer de illegale bestanden voor het eerst op een gegevensdrager zijn terecht gekomen, zouden alle bestanden met een vroeger datum-tijdstempel kunnen worden teruggegeven. Deze selectie procedure zou kunnen worden geautomatiseerd. De risico's hier zijn tweeledig. Ten eerste moeten dan alle illegale bestanden zijn geïdentificeerd want als er nog bestanden bestaan met vroegere datum-tijdstempels worden deze volgens deze methode teruggegeven. Ten tweede moet er onderzoek gedaan worden naar tijdmanipulatie van tijdstempels van bestanden (*Eng: timestomping*). Er bestaan vrij verkrijgbare tools waarmee de tijdstempels van bestanden kunnen aangepast (zie paragraaf 3.5.6).

Voordelen

- Proces kan geautomatiseerd worden en kost dus relatief weinig tijd.
- Eigenaar krijgt deel van privébestanden terug

Nadelen

- Kans bestaat dat illegale content wordt teruggegeven
- Eigenaar krijgt niet alle privébestanden terug.

4.3 Uitdagingen bij identificatie van bestanden

Bij het identificatieproces van bestanden met zowel illegale als legale content bestaat er een aantal uitdagingen. Deze uitdagingen worden vooral veroorzaakt door de diversiteit aan bestanden, de specifieke vakkennis die nodig is voor de herkenning, de soms gigantische hoeveelheid aan bestanden en 'last but not least' de vele manieren om bestanden met illegale content te verbergen.

4.3.1 Soorten bestanden

Zoals al aangegeven in paragraaf 3.4 bestaan er vele soorten bestanden. Ze zijn te identificeren aan de hand van de extensies van de bestanden of aan structuur van de inhoud van de bestanden. Ook is de inhoud van de bestanden soms te identificeren aan de hand van hashwaarden. Om van alle soorten van bestanden (foto's, video's, software, systeembestanden, tekstbestanden, databases enz.) te weten waar ze voor zijn of hoe ze werken is een uitgebreide technische kennis van IT en computers noodzakelijk. Illegale content betreft niet alleen kinderporno, maar steeds vaker ook cryptocurrency, malware of ransomware, inloggegevens enz. Actuele vakkennis, juridische kennis en/of kennis van de zaak is noodzakelijk om bestanden met illegale content te herkennen. Bij het identificeren van bestanden is de aanname dat bestanden niet op enige manier gemanipuleerd of verborgen zijn.

Er bestaan vele (soms vernuftige) manieren om bestanden te verbergen, meer hierover in paragraaf 4.3.3.

4.3.2 Hoeveelheid van bestanden

Hoeveel is veel? Die vraag is niet zo gemakkelijk te beantwoorden. Zoals al eerder als voorbeeld gegeven ontstaat een file van vrachtwagens van 5 km lang als alle nullen en enen van 4TB aan data worden uitgeprint. Maar wat zijn nu veel bestanden? Een typisch voorbeeld hiervan is een thuiscomputer met een Windows installatie en ongeveer 1 TB aan harddisk opslag. Als deze 1TB aan opslag gevuld is met afbeeldingen en/of videobestanden dan loopt het aantal bestanden al snel op tot enkele honderdduizenden (zie Figuur 8).

Toelichting:
Voorbeeld van het aantal bestanden en folders 900GB groot.

Figuur 8: aantal bestanden op een harddisk

In bovenstaande figuur een voorbeeld van hoeveel bestanden (van groot tot klein) er al in 900 GB passen. Verder bevat een typische Windows installatie al meer dan 100.000 bestanden.
Stel dat er in een zaak 500.000 afbeeldingen moeten worden beoordeeld op illegale content (KP). Stel dat een ervaren deskundige 3 seconden per afbeelding nodig heeft voor de beoordeling. Dan duurt het ruim 17 volledige dagen van 24 uur om alle afbeeldingen te bekijken. Uitgedrukt in werkdagen (effectief 6 uur per dag kijken) kost dit meer dan 69 werkdagen. Met andere woorden: alle bestanden handmatig bekijken is praktisch niet uitvoerbaar. Hiervoor moeten er zeker (gedeeltelijke) automatisering en slimme selectiemethoden worden toegepast.

4.3.3 Verbergen van bestanden in computers en telefoons

Er bestaan veel manieren om bestanden op een computer te verbergen. Dit kan variëren van gemakkelijk door bijvoorbeeld het zetten van de ‘hidden’ vlag van een bestand tot moeilijk door het aanmaken van versleutelde bestandscontainers die zelf ook nog op ‘geheime’ of op moeilijk toegankelijke locaties zijn opgeslagen. Het belangrijkste doel van het verstoppen van bestanden is het afschermen van de bestanden voor andere gebruikers. De redenen hiervoor kan privacy zijn maar ook dat de bestanden illegale content bevatten.

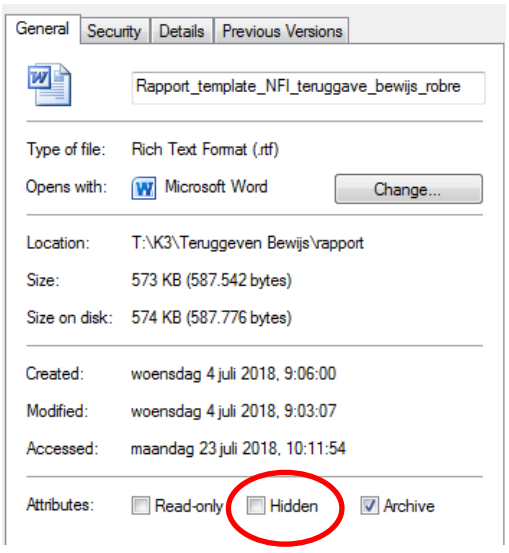
In de literatuur en op internet zijn vele boeken te vinden over het verbergen van data. Een boek dat specifiek gaat over het verbergen van data binnen Windows is ‘Data Hiding Techniques in Windows OS’ van N.A. Hassan (ISBN13: 9780128044964, september 2016). In dit boek worden vele technieken voor het

verbergen van data besproken van hele simpele tot zeer geavanceerde. Hieronder een aantal veel voorkomende, beginnend met een aantal simpele manieren en vervolgens een aantal moeilijkere en complexere manieren. Voor deze laatste manieren is een vrij uitgebreide kennis van Windows systemen noodzakelijk, maar op internet zijn voldoende uitgebreide handleidingen te vinden over hoe je zoiets als niet-Windowsspecialist moet doen.

Bij alle gegeven voorbeelden van methoden om bestanden te verbergen moet worden opgemerkt dat er forensische technieken bestaan om de verborgen bestanden toch te detecteren. Bij de simpelere vormen van verbergen zoals het *hidden* maken van bestanden of partities, het gebruik van *alternate datastreams* of zelf aangemaakt *Volume Shadow Copies* bestaan er diverse tools die de verborgen bestanden inzichtelijk maken. Bestanden die verborgen zijn met behulp van vercijfertechnieken (encryptie) in bijvoorbeeld containers of hele versleutelde harddisks zijn niet moeilijk te detecteren maar wel vaak moeilijk te benaderen omdat het kraken van een wachtwoord (vaak) een lastige en langdurige of soms zelfs een onmogelijke opgave is. Ook is dan niet bekend hoeveel gegevens er zich precies in een container bevinden. Een container met een vaste grootte hoeft niet helemaal gevuld te zijn.

Voor het verbergen van bestanden op mobiele telefoons werken vaak dezelfde technieken als bij het Windows operating-systeem. Van het *hidden* maken van een bestand/map tot het gebruik van cryptocontainers. Een mobiele telefoon is niets anders dan een ‘mini’-computer met zijn eigen bestands- en besturingssysteem. De in deze paragraaf beschreven verbergingstechnieken zijn vaak voor zowel computers als telefoons toepasbaar. Over het algemeen zijn mobiele telefoons beter beveiligd dan de meeste (thuis)computers met een (externe) harddisk. Ook vanuit de fabrikanten van telefoons wordt heel veel gedaan om de privacy gebruikers van de telefoon bij bijvoorbeeld verlies te waarborgen. Dit geeft een extra uitdaging om bij de gegevens van een mobiele telefoon te komen.

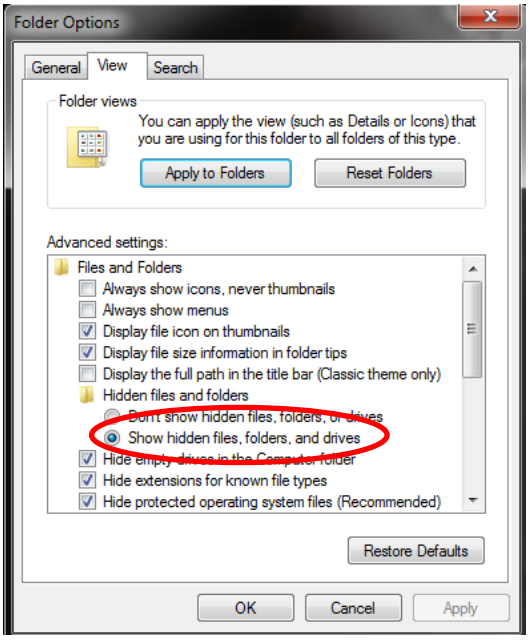
Verbergen bestanden en directories (hidden)
Binnen Windows kun je bestanden en directories ‘hidden’ maken. Hierbij stel je in het bestandssysteem in dat een bepaald bestand niet zichtbaar is in de Windows Explorer.



Toelichting:
De *hidden attribute* die per bestand of directory kan worden gezet.

Figuur 9: eigenschappen van een bestand

Deze verbergmethode kan echter gemakkelijk omzeild worden door de *Folder opties* instellingen van Windows. Hiermee kun je instellen dat de Windows Explorer de *Hidden files en folders* gewoon laat zien.

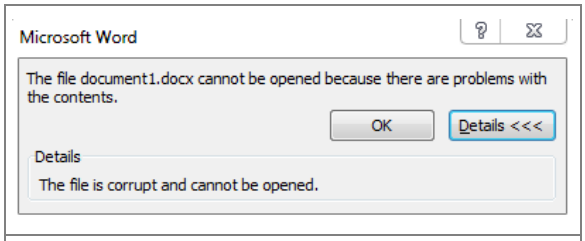


Toelichting:
De *hidden files and folders* optie. Hiermee kan de Windows Explorer de hidden bestanden wel of niet laten zien.

Figuur 10: Folder opties Windows Explorer

Verbergen partities
Behalve bestanden en directories is het ook mogelijk om hele partities van een harddisk te verbergen. Dit is te doen door bijvoorbeeld de driveletter (D:, E:, enz.) te verwijderen van een partitie. De partitie wordt dan door het bestandssysteem niet meer herkend en niet meer weergegeven. Er bestaan ook simpele tools waarmee partities kunnen worden verborgen voor een gebruiker. Bij het maken van forensische kopieën van een harddisk worden deze *hidden* partities gewoon mee gekopieerd en door de analysesoftware herkend als partities. Er zijn ook nog twee andere soorten partities (DCO en HPA partities) die niet zomaar worden herkend maar deze zijn ook niet heel gemakkelijk te gebruiken door een gebruiker en vergen veel expertise van de gebruiker.

Extensie aanpassingen
Het aanpassen van een extensie van een bestand is een gemakkelijke verbergingsmethode. Het is alleen voor de gebruiker van de bestanden relatief onhandig. Windows koppelt meestal programma's aan de extensies van bestanden. Zo worden bijvoorbeeld .doc of .docx bestanden aan Microsoft Word gekoppeld en beeldbestanden (.jpg, .bmp enz.) aan een imageviewer. Als een bestand genaamd *foto1.jpg* wordt hernoemt naar *document1.docx*, dan zal het bestand door Windows automatisch geopend worden door Microsoft Word met de onderstaande foutmelding als gevolg.



Toelichting:
Fout melding van Microsoft Word bij het openen van het bestand document.docx wat in werkelijkheid een beeldbestand is.

Figuur 11: foutmelding Windows bij extensieaanpassing

Deze methode is bruikbaar om bestanden te maskeren maar niet erg praktisch voor het (regelmatig) gebruik van deze bestanden. Als het van naam gewijzigde bestand met het juiste programma wordt geopend (bijvoorbeeld Windows Photo Viewer) wordt de inhoud van de foto wel zichtbaar.

Splitsen of combineren van bestanden
Voor het splitsen van bestanden geldt hetzelfde. De bruikbaarheid van de bestanden gaat er niet op vooruit. Het is mogelijk om met gebruik van deze techniek de bestanden die verborgen moeten worden op te splitsen in bijvoorbeeld 5 losse delen. Deze delen kunnen dan voor een deel op een harddisk en voor een deel op een losse USB schijf worden opgeslagen. Als je maar beschikking hebt over alleen de harddisk dan is de informatie niet meer terug te vormen naar het originele bestand.

Bestanden kunnen ook samen gekopieerd worden tot één nieuw bestand. Zo kan bijvoorbeeld door een simpele kopieer actie een 'verboden afbeelding' (Koala.jpg) worden toegevoegd aan een standaard afbeelding (Desert.jpg).


Desert.jpg


Koala.jpg

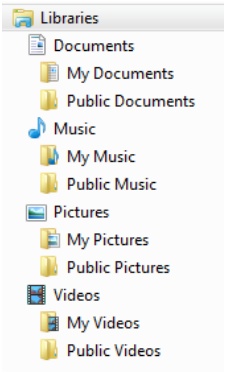
Bovenstaande twee afbeeldingen kunnen met een *copy* commando zoals hieronder weergegeven worden gecombineerd tot een nieuwe afbeelding.

```
copy /B Desert.jpg+Koala.jpg Desert_2.jpg
```


Desert_2.jpg

De nieuwe afbeelding laat alleen het plaatje van Desert.jpg zien. De bestandsgrootte van het nieuwe bestand is nu de grootte van de twee bronbestanden bij elkaar. Als Desert_2.jpg in software voor beeldverwerking wordt geladen zijn beide afbeeldingen wel weer zichtbaar.

Locatie van de bestanden
Een gebruiker van een Windows computer heeft een eigen directory structuur waar bestanden kunnen worden opgeslagen. In de Windows Explorer zien deze er meestal als volgt uit.

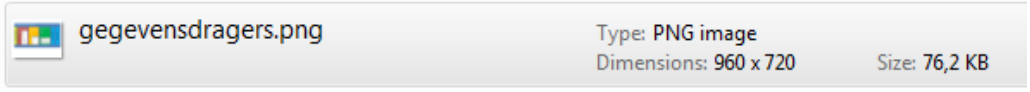


Op deze locaties kunnen door gebruikers bestanden zoals foto's, documenten, video's en muziek worden opgeslagen. Omdat standaard deze directories voor een gebruiker worden aangemaakt op de harddisk waar het operating systeem op staat (meestal C:\) kan de opslagcapaciteit beperkt zijn. Dit gebeurt meestal als er bij installatie van een computer is gekozen om het operating systeem te plaatsen op een aparte partitie (zie hoofdstuk 3). Deze partitie is dan vaak een kleiner gedeelte van de totale harddisk. Het is ook mogelijk om de hiernaast weergegeven directories op een andere partitie te plaatsen. Meestal worden er echter door een gebruiker op een aparte locatie van de harddisk directories aangemaakt waar bestanden

worden opgeslagen. Deze locaties kunnen best goed verstopt zitten achter een hele lange rij aan directories. Voor het opsporen van bestanden worden echter vaak automatische processen gebruikt die geen problemen hebben met lange directory-structuren.

Windows herstelpunten (Volume Shadow Copies)
Windows heeft een service, *Volume Shadow Copies* genaamd, die er voor zorgt dat er zogenaamde *restore points* (herstelpunten) worden aangemaakt van het huidige besturingssysteem van de computer. Deze service werkt alleen als het bestandssysteem NTFS is. De herstelpunten worden aangemaakt bij de installatie of de-installatie van nieuwe software of wanneer er een systeemupdate wordt gedaan. Mocht het systeem hierna niet meer stabiel zijn en bijvoorbeeld crashen dan kan er via een herstelpunt terug gegaan worden naar een vorige (en werkende) situatie. Deze service kan ook gebruikt worden om zelf een zogenaamd *Volume Shadow Copy* aan te maken waarin illegale content verstopt kan worden. Deze content kan met de juiste analyse ook gevonden worden. Er kunnen best grote bestanden worden opgeslagen in de Windows herstelpunten maar omdat er maar een beperkte maximale ruimte hiervoor is gereserveerd zullen oudere herstelpunten na verloop van tijd worden overschreven. De houdbaarheid van de herstelpunten voor een 'normale' PC gebruiker is minstens een maand of 5.

Alternate datastreams
Het fenomeen *alternate datastreams* komt alleen voor bij het NTFS bestandssysteem. Er kan aan één bestandsnaam meerdere datastromen (bestanden) gekoppeld worden. Dit kunnen alle soorten van bestanden zijn en het aantal bestanden dat aan één bestandsnaam gekoppeld kan worden loopt in de duizenden. Bij gebruik van de Windows Explorer zullen alleen de originele bestandsnamen worden getoond en niet de alternatieve bestanden die er eventueel ook aanhangen. Als voorbeeld hier gegeven zijn er aan een beeldbestand (gegevensdragers.png) twee andere bestanden gekoppeld, een ander beeldbestand (foto1.jpg) en een document (document1.docx). In de Windows Explorer is dit niet te zien, ook de bestandsgrootte is die van het originele beeldbestand.



Bekijken we deze afbeelding in de zogenaamde *command prompt* en bekijken we de directory met het commando *dir /r* dan zien we dat er meer verstopt zit achter dit bestand.

Achter het bestand gegevensdragers.png zit nog een bestand document1.docx van 12.796 bytes en een foto1.jpg bestand met een grote van 845.941 bytes. Windows zelf heeft geen mogelijkheden om *alternate datastreams* te detecteren. Er bestaan wel diverse gratis tools die deze mogelijkheid wel hebben.

Tot nu toe zijn allerlei relatief simpele manieren besproken waarbij gebruikers van computers gegevens kunnen verstoppen in Windowssystemen. Voor al deze methoden geldt bijna altijd dat ze met de juiste tools ook weer gevonden kunnen worden. De vraag is wel of er altijd naar alle mogelijke manieren gezocht wordt bij

een onderzoek aan bijvoorbeeld een harddisk door de politie. Het is en blijft een kat en muis spelletje tussen beide partijen met het vinden van nieuwe manieren van verstoppen en ontdekken van bestanden, waarbij altijd een afweging gemaakt moet worden tussen de benaderbaarheid dan wel bruikbaarheid van desbetreffende bestanden en het verstoppen ervan.

De beste methoden waarbij de inhoud van de bestanden niet meer zichtbaar zijn is encryptie (versleuteling) van deze bestanden. Een versleuteld bestand kan gemakkelijk in 'plain sight' op de desktop van een computer staan maar als deze met een goede betrouwbare versleuteling met een lang (en/of complex) wachtwoord beveiligd is kan de inhoud ervan niet worden geopend.
Niet alleen bestanden, maar ook directories, partities en hele harde schijven kunnen versleuteld worden, al dan niet in een cryptocontainer. Meer informatie over cryptografie en wachtwoorden is te vinden in bijlage 2.

Steganografie
Steganografie is een vorm van cryptografie waarbij informatie wordt verborgen in andere (onschuldige) bestanden zoals afbeeldingen, muziek, video en tekstbestanden. Een vrij uitgebreide behandeling van al deze mogelijkheden wordt beschreven in hoofdstuk 3 en 6 van het boek '*Data Hiding Techniques in Windows OS*'.

Samengevat
De enorme hoeveelheid aan mogelijkheden om gegevens te verstoppen geeft wel een indicatie dat het vinden van alle mogelijk verstopte gegevens bijna een onmogelijke taak voor de opsporings– en onderzoeksteams is.

4.4 Uitsluiting van aanwezigheid van illegale content

Er zijn in deze technische toelichting een aantal methoden en technieken beschreven voor het verbergen van bestanden met illegale content. Zelfs als bestanden niet verborgen zijn kunnen ze nog door de enorme hoeveelheid en diversiteit aan bestanden niet eenvoudig gevonden worden.

In een forensisch onderzoek in een strafzaak is het vaak niet haalbaar om alle illegale bestanden te vinden. Aangezien de zakendruk voor alle ketenpartijen hoog is, stopt het onderzoek meestal op het moment dat de tenlastelegging voldoende onderbouwd is en begint men aan de volgende zaak.

De beperkte onderzoekstijd van de opsporings- en onderzoeksdiensten in zaken, de vele manieren van verstoppen, het eeuwige kat en muis spel tussen verstoppen en vinden, de eenvoudige wijze om aan informatie via het internet over het verbergen van bestanden te komen, enzovoort, maken het onmogelijk om uit te sluiten dat er zich op een gegevensdrager geen niet-geïdentificeerde bestanden met illegale content meer bevinden.
Deze zekerheid is alleen te geven als de originele gegevensdrager en alle gegevensdragers waar kopieën van de originele gegevensdrager op hebben gestaan fysiek worden vernietigd, als er tenminste geen kopieën in de *cloud* van deze gegevens bestaan.

5 Verwijderen van geïdentificeerde bestanden

5.1 Inleiding

Bestanden komen voor op gegevensdragers met verschillende bestandssystemen, waarbij geldt dat voor ieder bestandssysteem verschillende methoden bestaan waarop deze bestanden verwijderd kunnen worden. In dit hoofdstuk worden deze manieren van de meest voorkomende bestandssystemen (en dus soorten van gegevensdragers) beschreven. Zoals in paragraaf 3.3.6 al is aangegeven, bestaat er een onderscheid tussen het normaal en definitief verwijderen (wipen) van bestanden. Met definitief wordt hier bedoeld dat de bestanden (of gegevens) met behulp van forensische software niet meer te achterhalen zijn.
Bij het normaal of definitief verwijderen horen verschillende eisen aan de verwijdermethoden. Sommige verwijdermethoden kunnen zeer eenvoudig en effectief zijn maar ook nevenschade hebben met als extreem voorbeeld het fysiek versnipperen van de harde schijf. In onderstaande paragrafen zal ingegaan per soort gegevensdrager worden op verschillende verwijdermethoden en hun beperkingen.

5.2 Bestandssystemen computers (NTFS, FAT32)

5.2.1 Normaal verwijderen van bestanden

Als een computergebruiker een bestand verwijdert dan is de reden hiervoor over het algemeen dat hij het bestand niet meer nodig heeft en de vrije ruimte op zijn computer wil vergroten of zijn gegevens overzichtelijker wil indelen. Een besturingssysteem biedt herstelopties via de prullenbak of geïntegreerde back-up oplossingen zoals bestandssysteem-snapshots (*Volume Shadow Copy*, zie paragraaf 4.3.3). Maar zelfs als het verwijderde bestand geen onderdeel is van de prullenbak of back-up, blijft de inhoud van het bestand opgeslagen in de niet toegewezen ruimte en is soms terug te vinden door te *carven* (zie paragraaf 3.4.3). Ook bestaat de mogelijkheid dat het bestand eerder op een andere locatie (soms onder een andere naam) van het bestandssysteem heeft gestaan. Ook deze al verwijderde bestanden kunnen nog weer worden teruggehaald door middel van *carving*-technieken.
Als een bestand normaal wordt verwijderd dan wordt alleen de verwijzing naar het bestand verwijderd dan wel aangepast, de inhoud van het bestand blijft gewoon bestaan. Vervolgens wordt de ruimte die de bestanden innemen wel weer vrijgegeven voor nieuwe bestanden. Na verloop van tijd kunnen de originele gegevens van de bestanden geheel of gedeeltelijk worden overschreven.
Zogenaamd verwijderde bestanden kunnen gedeeltelijk nog terug te vinden zijn in *slack space* dan wel *unallocated space* (zie paragraaf 3.3.2 en 3.3.3).
Verder kunnen bestanden zijn opgeslagen in een archiefbestand (bijvoorbeeld zip, zie 3.5.7.). Een besturingssysteem heeft over het algemeen geen toegang tot bestanden die opgeslagen zijn in archiefbestanden. Bijvoorbeeld het selectief verwijderen van één bestand uit een zip-archiefbestand is feitelijk het kopiëren van het zip-archiefbestand waarbij één gecomprimeerd bestand wordt overgeslagen. Op de harde schijf staat vervolgens één verwijderd zipbestand waarin het te verwijderen bestand nog aanwezig is, en een nieuw zip bestand zonder het te verwijderen bestand.
Bij het normaal verwijderen van bestanden, al dan niet in archiefbestanden, is het dus verre van zeker dat het bestand dan ook echt weg is.

5.2.2 Definitief verwijderen van bestanden

Een andere methode voor het verwijderen van bestanden heet wipen. Bij deze methode worden de gegevens van een bestand overschreven door andere data. Als

een bestand is gefragmenteerd (zie paragraaf 3.3.4) moet de wiping software alle losse fragmenten (sectoren) op de gegevensdrager overschrijven. Of dit overschrijven op de juiste fragmenten op de correcte wijze wordt uitgevoerd hangt af van de implementatie van de *wiping*methode en het besturingssysteem. Daarom is het verstandig om wipingsoftware te gebruiken die specifiek voor een bepaald besturingssysteem is gemaakt. In het algemeen wordt tegenwoordig aangenomen dat eenmalig overschrijven met nullen of enen of random data voldoende is om een bestand definitief te verwijderen. In de jaren negentig bestonden er nog technieken die het mogelijk maakten of informatie terug te halen als een bestand al meer dan twintig keer was overschreven, maar met de huidige stand van de techniek in harde schijven is dit niet meer mogelijk.

5.3 Mobiele apparaten

5.3.1 Verwijderen bestanden op mobiele telefoons

Mobiele telefoons zijn ook gegevensdragers met bepaalde besturings- en bestandssystemen en diverse soorten opslagcapaciteiten (interne vaste opslag en sd-kaarten). De meeste mobiele apparaten bevatten een versie van het Android besturingssysteem[9], gevolgd door versies van het iOS besturingssysteem[10]. Een paar specifieke telefoonmerken gebruiken nog andere varianten van besturingssystemen.
Het verwijderen van bestanden op telefoons werkt anders dan het verwijderen van bestanden op computers. Het voert in deze toelichting te ver om in te gaan op alle mogelijkheden dan wel onmogelijkheden van het verwijderen van bestanden. Dit hangt heel erg af van onder andere het type telefoon, soort besturingssysteem, soort bestandssysteem, aanwezigheid van versleutelingsmechanismen, soort van opslagcapaciteit, toegangsrechten tot de telefoon of opslagcapaciteit. De vraag of een bestand definitief verwijderd is of kan worden van een mobiele telefoon moet dus per zaak apart beoordeeld worden.

5.3.2 Back-up mogelijkheden van mobiele apparaten en computers

Bijna alle mobiele apparaten en computers hebben de mogelijkheid tot het maken van een back-up van de gebruikersbestanden en/of applicatiebestanden naar een externe locatie. Dit kan een andere computer zijn (iTunes back-up van een iPhone) of in de meeste gevallen een back-up in de *cloud*. Voorbeelden hiervan zijn de iCloud van Apple, Google Drive van Google en Onedrive van Microsoft. Er zijn ook vele cloud oplossingen die door commerciële partijen (dus tegen betaling) worden aangeboden. Via deze back-up oplossingen is het voor gebruikers vaak erg gemakkelijk om foto's, video's, tekstbestanden en chats terug te zetten op een mobiele telefoon of een computer. Bestanden die definitief verwijderd zijn (gewiped) op de computer of telefoon kunnen dan toch weer teruggehaald worden. In sommige gevallen (bijvoorbeeld bij de iCloud) bestaat er een synchronisatie van de afbeeldingen op de telefoon en in de cloud. Na bepaalde tijd worden dan verwijderde foto's ook in de cloud back-up automatisch gewist.
Dus als eventuele illegale content niet wordt verwijderd in externe back-up locaties is het verwijderen van die bestanden op de telefoon of computer weinig doeltreffend.

[9] Zie voor een actueel overzicht: https://nl.wikipedia.org/wiki/Android_(besturingssysteem)
[10] Zie voor een actueel overzicht: https://nl.wikipedia.org/wiki/IOS_(Apple)

5.4 Niet vluchtige geheugens

Tegenwoordig zijn de meest gebruikte vorm van opslag geheugenchips. Voorbeelden hiervan zijn: USB sticks, SD-kaarten (ook in telefoons), vaste opslag in mobiele apparaten en natuurlijk in steeds grotere getale de SSD harddisks in computers. Wat precies de verschillen in allerlei soorten geheugenchips zijn wordt hier niet besproken, maar een belangrijk element dat voor kan komen bij het verwijderen van bestanden bij vluchtige gegevens heet TRIM en is van belang bij het (definitief) verwijderen van bestanden.
Samengevat zorgt TRIM (of *wear levelling*) ervoor dat bestanden die verwijderd worden op een SSD/geheugenchip meestal ook echt definitief verwijderd zijn. Zoals altijd werkt het niet in alle gevallen zo en bestaan er situaties waarbij bestanden nog niet definitief weg zijn, maar bestanden op een SSD worden wel vaker definitief verwijderd dan bij magnetische harddisks!

6 Bijlage 1: Bestandskenmerken

Onder meer om de leesbaarheid te vergroten worden de bestandskenmerken niet weergegeven in enen en nullen, maar wordt daarbij de zogenaamde zestientallige (hexadecimale) notatie gebruikt. In die notatie worden vier opeenvolgende nullen en enen genoteerd als één van de volgende zestien tekens: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f. Er wordt hierbij geen onderscheid gemaakt tussen hoofdletters en kleine letters. Een voorbeeld van een 256 bits (32 byte) bestandskenmerk is de volgende:

```
2584 878d 8694 6001 3fab 045c 4de2 03d1
ca8a bb44 3b6b 2169 c9a1 4021 8d2b 21bb
```

Bij het gebruik van (goede) hashalgoritmen leidt een wijziging van slechts één bit in een bestand met extreem grote waarschijnlijkheid tot een compleet ander bestandskenmerk dan dat van het bestand voor de wijziging. Dit wordt ook wel het 'watervaleffect' (Engels: *avalanche effect*) genoemd. Zie ter illustratie in onderstaande tabel de bestandskenmerken (in dit voorbeeld MD5) berekend over bijna gelijke bit-reeksen zijn totaal verschillend.

Bitreeks	**MD5-bestandskenmerk**
1000000000000000	fc60 9b43 cb85 95a9 a832 cbc2 591e d83a
1000000000000001	9d26 f82a f654 8210 454c 017a 3179 c0ec
1000000010000000	688e c893 e933 aff7 2480 5d61 4e43 f68e

Tabel: MD5-bestandskenmerken van bijna gelijke bit-reeksen

Deze paragraaf beperkt zich tot een specifieke categorie hashalgoritmen en bijbehorende bestandskenmerken, de zogeheten *cryptografische* of *niet-omkeerbare* hashalgoritmen. In het Engels heten deze *one-way hash algorithms*. Waar hier de term hashalgoritmen wordt vermeldt, gaat het steeds om dergelijke niet-omkeerbare hashalgoritmen. Ze hebben, naast het genoemde watervaleffect, minimaal de volgende eigenschappen:

1. Éénrichting (Engels: *one-way*, *preimage resistant*): het is praktisch gezien onmogelijk[11] om bij een gegeven bestandskenmerk een bestand te vinden/creëren dat bij dat bestandskenmerk hoort.
2. Doel-botsing-bestendig (Engels: *target collision resistant*, *second preimage resistant*): het is praktisch gezien onmogelijk een bestand met andere inhoud te vinden/creëren waarbij het bestandskenmerk identiek is aan het bestandskenmerk van een vooraf gegeven bestand.
3. Botsing-bestendig (Engels: *random collision resistant*): het is praktisch gezien onmogelijk om twee bestanden te vinden met andere inhoud maar met hetzelfde bestandskenmerk.

De beschreven eigenschappen van een niet-omkeerbaar hashalgoritme zijn aannames die werkbaar zijn tot het tegendeel is aangetoond. Er is momenteel zelfs geen wiskundig bewijs bekend voor het bestaan van niet-omkeerbare hashalgoritmen. Een niet-omkeerbaar hashalgoritme verliest zijn veronderstelde niet-omkeerbare status zodra deze *gekraakt* is. Het kraken van een hashalgoritme is meestal een zoektocht naar bestanden die hetzelfde bestandskenmerk hebben. Vervolgens wordt een methode gezocht om efficiënt dergelijke bestanden te vinden.

[11] 'Praktisch onmogelijk' kan hier worden gelezen als: 'Zelfs wanneer alle computerkracht van de wereld tegelijk gebruikt zou kunnen worden, is het nog steeds onmogelijk'.

Zodra een dergelijke methode gevonden is voor een hashalgoritme, dan zijn de beschreven eigenschappen (deels) niet meer van toepassing op dit algoritme. Als een gangbaar hashalgoritme gekraakt is of dreigt te worden, zal dit doorgaans plaatsmaken voor een nieuw hashalgoritme dat niet op dezelfde manier valt te kraken. Het overstappen op een nieuw hashalgoritme is daarom na verloop van tijd mogelijk noodzakelijk.

Voor identificatie en classificatie van bestanden is het gebruik van bestandskenmerken over het algemeen veel efficiënter omdat bestandskenmerken van een bestand relatief klein zijn. Tegenwoordig zijn bestanden zelf vaak (zeer) groot (enkele gigabytes). De momenteel vaak gebruikte maximale lengte van een bestandskenmerk is slechts 32 bytes (64 tekens). Het is dan ook veel eenvoudiger en sneller om bestandskenmerken van bestanden te vergelijken, dan de inhoud van de bestanden zelf. Daarnaast is het veel eenvoudiger en sneller om te communiceren over bestandskenmerken van bestanden, dan over (de inhoud van) de bestanden zelf.

Bestandskenmerken helpen verder bij het identificeren van bestanden. Bij deze forensische toepassing wordt gebruik gemaakt van een database met bestandskenmerken van bekende, geclassificeerde bestanden. Een voorbeeld is het identificeren van bestanden met kinderpornografische afbeeldingen. Om een bestand te identificeren wordt eerst het bestandskenmerk van dat bestand berekend. Vervolgens gaat de onderzoeker na of dit bestandskenmerk voorkomt in de database[12]. Is dit het geval, dan kijkt de onderzoeker hoe het bestandskenmerk is geclassificeerd. De database kan zowel bestandskenmerken bevatten van bestanden die eerder aangemerkt zijn als kinderpornografisch als van bestanden die bekende niet-relevante gegevens bevatten. Deze methode maakt het daarom mogelijk om zowel relevante bestanden te ontdekken als om niet-relevante bestanden vroegtijdig voor verder onderzoek uit te sluiten. Omdat het extreem onwaarschijnlijk is dat door deze methode verschillende bestanden hetzelfde bestandskenmerk zullen hebben, is de kans op een foutieve classificatie verwaarloosbaar klein (zo goed als 0).

De wereldwijd meest gebruikte hashalgoritmen om de integriteit van digitale gegevens te controleren en voor bestandsidentificatie zijn MD5 met een lengte van 128 bits en SHA-1 met een lengte van 160 bits. Zowel binnen als buiten de forensische wereld vinden ze veel toepassing. De veiligheidsgaranties die ze bieden zijn, ondanks voortdurende 'aanvallen' op deze algoritmes, nog steeds extreem hoog voor integriteitscontrole en bestandsidentificatie. Een preventieve overstap naar één van de nog veiliger geachte SHA-2 hashalgoritmen met een lengte van 256 bits (ook wel SHA-256 genoemd) is aan te bevelen voor forensische toepassingen.

De kans dat een willekeurig (ander) bestand hetzelfde bestandskenmerk heeft als een bepaald *gegeven* bestand wordt hier uitgelegd aan de hand van het volgende voorbeeld. Neem een database met bestandskenmerken van bestanden waarvan de inhoud geclassificeerd is als kinderpornografisch. Stel nu dat in een onderzoek bestand met de naam *B* wordt aangetroffen, waarbij de bestandskenmerken van *B* voorkomen in deze database. Wat is nu de kans dat **B** toevallig toch een andere inhoud heeft dan het kinderpornografische bestand? Dit wordt hier uitgerekend per gebruikt bestandskenmerk.

[12] KP database: De database met hashes van kinderpornografische afbeeldingen is zeker niet volledig. Er staan zelfs maar een beperkt aantal afbeeldingen in. Het veranderen van slecht 1 bit van de afbeelding levert al een andere hash waarde op terwijl dit geen zichtbaar effect op de afbeelding zelf heeft.

- **MD5**
  Een MD5-bestandskenmerk bestaat uit 128 bits. Ervan uitgaande dat elk MD5- bestandskenmerk met dezelfde kans voor kan komen zijn er in totaal 2^{128} ($\approx 3{,}40\ x\ 10^{38}$) aan mogelijke MD5- bestandskenmerken. Dit betekent dat de kans dat een willekeurig ander bestand met andere inhoud toch hetzelfde MD5- bestandskenmerk heeft gelijk is aan $\frac{1}{2^{128}} \approx 2{,}9\ x\ 10^{-39}$.
- **SHA-1**
  Een SHA-1- bestandskenmerk bestaat uit 160 bits. Een vergelijkbare berekening als bij de MD5- bestandskenmerk geeft dat de kans dat een willekeurig ander bestand met andere inhoud toch dezelfde SHA-1- bestandskenmerk heeft gelijk is aan $\frac{1}{2^{160}} \approx 6{,}8\ x\ 10^{-49}$.
- **SHA-256**
  Een SHA-256 bestandskenmerk uit 256 bits. Een vergelijkbare berekening als bij de MD5- en SHA-1- bestandskenmerken geeft dat de kans dat een willekeurig ander bestand met andere inhoud toch dezelfde SHA-256- bestandskenmerk heeft gelijk is aan $\frac{1}{2^{256}} \approx 8{,}6\ x\ 10^{-78}$.

Belangrijk is nog om op te merken dan de hash waarden worden berekend over de inhoud van een bestand en niet over de metadata. Twee dezelfde bestanden met twee verschillende bestandsnamen leveren toch dezelfde hash waarden op.
(zie voorbeeld hieronder, verschillende bestandsnamen zelfde hash)

7 Bijlage 2: Cryptografie en wachtwoorden

7.1 Cryptografie

7.1.1 Cryptografische algoritmen en sleutels

Het proces van vercijferen en ontcijferen van digitale gegevens gebeurt met cryptografische algoritmen. Een dergelijk algoritme voert bepaalde transformaties op klare tekst uit, met als resultaat cijfertekst. Cryptografische algoritmen maken gebruik van zogeheten cryptografische sleutels. Een ieder die kennis heeft van het algoritme en in het bezit is van de ontcijferingssleutel kan de daarmee bijbehorende vercijferde gegevens ontcijferen. Over het algemeen is het cryptografische algoritme (bijvoorbeeld de gebruikte programmatuur) niet geheim, dit in tegenstelling tot de cryptografische sleutel. Hoe moeilijker het is om een cryptografische sleutel te kunnen achterhalen, hoe moeilijker het is om de bijbehorende vercijferde gegevens te kunnen ontcijferen.

Bij het maken van cryptografische ontcijferingssleutels zijn de volgende aspecten van belang:

- de sleutel moet lang genoeg zijn dat het niet eenvoudig is om alle sleutels uit te proberen;
- er mag geen informatie in de vercijferde gegevens aanwezig zijn waardoor de sleutel eenvoudig te bepalen is;
- het gebruikte algoritme om de sleutel te maken moet ervoor zorgen dat in principe elke willekeurige sleutel van bepaalde lengte te maken is.

7.1.2 Sleutellengtes

In de moderne cryptografie bestaan cryptografische sleutels uit een lange reeks bits. Veelgebruikte lengten zijn 128 tot 4.096 bits. Vaak dwingt het te gebruiken algoritme af welke bepaalde (vaste) lengte in bits de sleutel moet hebben om gebruikt te kunnen worden. Hoe langer de sleutellengte in bits, hoe meer mogelijke sleutels te gebruiken zijn. Bij een sleutelengte van bijvoorbeeld 128 bits zijn er in totaal 2^{128} ($\approx 3{,}4 * 10^{38}$) verschillende sleutels mogelijk. Bij een lengte van bijvoorbeeld 4.096 bits is dit $2^{4.096}$ ($\approx 1{,}0 * 10^{1.233}$).

7.1.3 Wachtwoorden

Wachtwoorden zijn (over het algemeen door een gebruiker) gekozen reeks karakters waarmee bepaalde informatie afgeschermd wordt. Het NFI onderscheidt wachtwoorden in twee hoofdsoorten: wachtwoorden die een cryptografische sleutel afschermen en wachtwoorden die nodig zijn om toegang tot een (onvercijferd) systeem te verkrijgen.

Wachtwoorden voor sleutels

Omdat het voor een gebruiker ondoenlijk is om een sleutel bestaande uit een lange reeks willekeurige bits te onthouden worden cryptografische sleutels vaak afgeschermd met (afgeleid van) een wachtwoord. Met kennis van het (juiste) wachtwoord kan de ontcijferingssleutel worden achterhaald; hiermee kan de daarmee vercijferde informatie ontcijferd worden. De hoofreden om sleutels afgeleid van of beschermd door wachtwoorden te gebruiken in plaats van de wachtwoorden zelf, heeft er mee te maken dat de cryptografische algoritmen vaak een sleutel van vaste bit-lengte nodig hebben. Veelgebruikte algoritmen om van wachtwoorden cryptografische sleutels te maken zijn gebaseerd op zogeheten cryptografische of niet-omkeerbare hashalgoritmen.

Toegangswachtwoorden
Naast vercijferde kunnen wachtwoorden ook onvercijferde gegevens afschermen. In dit geval geeft het (juiste) wachtwoord direct toegang tot de (onvercijferde) achterliggende gegevens. Een dergelijk wachtwoord wordt toegangswachtwoord genoemd. Een voorbeeld van een toegangswachtwoord is een gebruikerswachtwoord voor Microsoft Windows computers.

7.2 Achterhalen van wachtwoorden

Wanneer cryptoanalyse is afgerond en bekend is geworden hoe bepaalde programmatuur omgaat met een door een gebruiker ingevoerd wachtwoord, kan gestart worden met proberen te achterhalen van een wachtwoord voor het specifieke onderzoek. Het komt voor dat een programmatuur zelf al genoeg informatie over het wachtwoord prijsgeeft, dat een wachtwoord zo goed als direct te achterhalen is. In dit geval is verder onderzoek niet meer nodig.

Wanner niet direct een wachtwoord achterhaald kan worden, wordt naar aanleiding van de resultaten van de cryptoanalyse begonnen met het aanvallen van het wachtwoord. Onder aanvallen wordt het (automatisch) testen van (vele) wachtwoorden verstaan. Dit aanvallen gebeurt met steeds andere series van wachtwoorden totdat een juist wachtwoord gevonden (achterhaald) is. De aanvallen kunnen uitgevoerd worden met zelf ontwikkelde en (commercieel) beschikbare programmatuur en apparatuur. Voorbeelden van commerciële programmatuur die gebruikt kunnen worden om wachtwoorden te achterhalen zijn AccessData PRTK, Passware en Elcomsoft. Vanwege opsporingsbelang wordt verder niet ingegaan op welke specifieke aanvallen het NFI zelf ontwikkeld heeft.

Het aanvallen zelf bestaat hoofdzakelijk uit twee naast elkaar te gebruiken methodes: brute rekenkracht en woordenboek gebaseerd.

7.2.1 *Brute rekenkracht (Eng.: Brute force)*

Bij deze methode worden wachtwoorden bestaande uit een vaste set van karakters met vooraf gedefinieerde lengtes geprobeerd. Wanneer bijvoorbeeld wordt gekozen voor een aanval met wachtwoorden van lengte 6, waarbij het te testen woord alleen kleine letters uit het Nederlandse alfabet worden gebruikt (Eng.: *lowercase*) worden er in totaal 26^6 = 308.915.776 wachtwoorden getest. Wanneer naast kleine letters ook hoofdletters worden toegestaan (Eng.: uppercase), is het totaal aantal mogelijke wachtwoorden $(26+26)^6$ = 19.770.609.664. Ook aanvallen van wachtwoorden waarbij per positie in het te testen woord een reeks van bepaalde karakters is toegestaan valt onder een brute rekenkracht aanval. Stel dat een aanval test op wachtwoorden van lengte 7, beginnend met een hoofdletter, gevolgd door drie kleine letters en afgesloten met drie cijfers. In totaal worden dan $26*26^3*10^3$ = 456.976.000 mogelijkheden geprobeerd. Hieronder vallen bijvoorbeeld de volgende twee (wacht)woorden: `Pass123` en `Ydqt901`.

De complexiteit (doorlooptijd) van een brute rekenkracht aanval wordt groter bij grotere lengte en meerdere mogelijk karakters per positie. Bij bijvoorbeeld wachtwoorden van lengte 12, waarbij op alle posities hoofdletters, kleine letters, cijfers en de dertien leestekens `~!@#$%^&*()_+` mogen worden gebruikt, zijn $(26+26+10+13)^{12}$ = 31.676.352.024.078.369.140.625; ruim 31 triljard.

7.2.2 *Woordenboek (Eng. Dictionary)*

Algemeen bekend is dat gebruikers wachtwoorden gebruiken die gebaseerd zijn op ‘normale woorden’ of standaard reeksen, afgeleid uit de omgeving van de gebruiker. Onder normale woorden vallen reguliere woorden uit een specifieke taal

(bijvoorbeeld uit het Nederlands of Engels, maar ook straattaal (Eng.: *slang*)), eigennamen (bijvoorbeeld van personen of steden) en merken (bijvoorbeeld van auto's of telefoons). Dit heeft als achtergrond dat personen wachtwoorden beter kunnen onthouden wanneer (een deel van) het wachtwoord iets 'bekends' is. Bij standaard reeksen kan gedacht worden aan wachtwoorden als `12345678` en `qazwsx`. Dat soort reeksen worden *keywalks* genoemd, omdat de gebruikte karakters naast elkaar liggen op een toetsenbord.

Het NFI beschikt over diverse standaard woordenboeken die gebruikt kunnen worden als basis voor een aanval. Dit zijn geen statische woordenboeken; deze worden op geregelde tijden gecontroleerd en zo nodig uitgebreid. Ook zaak-specifieke woordenboeken worden gemaakt van woorden die uit onderzoek naar de gebruiker (verdachte) naar voren zijn gekomen. Denk hierbij aan biografische gegevens, taalgebruik, hobby's en interesses. Per te testen woord kunnen een of meerdere variaties worden toegepast, zoals het toevoegen van cijfers, het aanpassen van kleine letters naar hoofdletters of het combineren van woorden.

De complexiteit (doorlooptijd) van de aanval wordt groter naarmate het aantal variaties groter wordt. Stel dat een woordenboek bestaat uit 20.000 unieke woorden (in kleine letters), en als variaties worden de volgende gebruikt:

- eerste letter van elk woord wordt hoofdletter;
  - 20.000 extra wachtwoorden (maximaal);
- achter elk woord worden 2 cijfers geplakt:
  - 20.000.000 extra wachtwoorden voor woorden bestaande uit kleine letters;
  - 20.000.000 extra wachtwoorden voor woorden bestaande uit een hoofdletter gevolgd door kleine letters.

In totaal worden hier 40.040.000 wachtwoorden geprobeerd.

7.2.3 Verificatie achterhaald wachtwoord

Wanneer een wachtwoord achterhaald is, wordt geverifieerd of dit wachtwoord inderdaad het te onderzoeken vercijferde of afgeschermde materiaal kan ontcijferen en/of inzichtelijk kan maken. Hierbij wordt in principe gebruik gemaakt van de (originele) programmatuur waar het wachtwoord mee gezet is.

7.3 Achterhalen van sleutels

Zoals eerder aangegeven groeit de complexiteit, dus de doorlooptijd, wanneer gezocht moet worden naar steeds langere wachtwoorden met steeds meerdere variaties in de te gebruiken karakters. Voor vercijferde gegevens afgeschermd met een wachtwoord kan het een optie zijn om niet het wachtwoord maar de cryptografische sleutel zelf te achterhalen. Immers, met de juiste sleutel kunnen de vercijferde gegevens direct ontcijferd worden. Voorwaarde hiervoor is dat het algoritme dat gebruikt moet worden voor ontcijferen bekend is. Uit dit algoritme kan de lengte van een mogelijke sleutel bepaald worden. Hiernaast kan het zijn dat de programmatuur extra informatie over de sleutel bewaard waardoor het zoeken naar sleutels eenvoudiger wordt. Deze informatie kan bekend zijn geworden na cryptoanalyse van de gebruikte programmatuur (zie ook paragraaf 6.1).

7.3.1 Alle sleutels genereren en testen?

De vraag kan gesteld worden waarom niet alle mogelijke sleutels gegenereerd en getest worden, vergelijkbaar met de brute rekenkracht methode voor wachtwoorden. Dit heeft te maken met de complexiteit (lengte) van de sleutels. Uit

paragraaf 6.1.2 blijkt dat de tegenwoordig gebruikte sleutels uit zoveel bits bestaan, dat er heel (te) veel verschillende sleutels zouden moeten worden gegenereerd en getest. Zelfs met gebruik van alle computers aanwezig op de wereld zal het onmogelijk zijn om deze allemaal te kunnen testen. Een rekenvoorbeeld: stel we zoeken een sleutel met een lengte van 256 bits, er zijn dan ook 2^{256} mogelijke sleutels. Stel verder dat we de beschikking hebben over 10 miljard computers die allemaal 10 miljard sleutels per seconde kunnen testen. De tijd die het kost om alle sleutels te proberen is $(2^{256}) / (10^9 * 10^9) \approx 1{,}15 * 10^{59}$ seconden. Dit komt overeen met ongeveer $3{,}7^{49}$ eeuwen. Tabel 1 geeft een overzicht van een sleutel lengtes afgezet tegen de tijd die het kost om ze allemaal te genereren en testen, uitgaande van dezelfde hoeveelheid en snelheid van computers.

Bitlengte	**Complexiteit**
64	0,18 seconden
80	3,5 uur
96	25 jaar
128	1.078.289.752 eeuwen
256	$3{,}7^{49}$ eeuwen
4.096	$3{,}7^{1.203}$ eeuwen

7.3.2 Waar zoeken naar sleutels?

Als het redelijkerwijs niet mogelijk is om alle sleutels te genereren en te testen, hoe kunnen we dan toch sleutels achterhalen om gegevens mee te ontcijferen? Hier zijn een aantal onderzoeksmogelijkheden voor uit te voeren. Afhankelijk van de resultaten van de cryptoanalyse worden een of meerdere van de volgende mogelijkheden uitgevoerd:

- vooral voor performanceredenen en gebruikersgemak bewaren sommige programma's na invoering van een wachtwoord de hierdoor gegenereerde ontcijferingssleutel in het geheugen van de computer; wanneer het lukt om de sleutel uit het geheugen te halen, kan deze gebruikt worden voor ontcijferen;
- het kan zijn dat een sleutel door het programma wordt opgeslagen op bijvoorbeeld de harde schijf van een computer; in dit geval kan gericht gezocht worden in de gegevens aanwezig op de harde schijf om de sleutel te achterhalen;
- het kan zijn dat het algoritme dat gebruikt wordt om sleutels te genereren bewust of door implementatiefouten slechts een beperkt gedeelte van alle mogelijke sleutels kan genereren; Als het aantal mogelijke sleutels hierdoor klein genoeg wordt, is de juiste sleutel mogelijk wel door bijvoorbeeld brute rekenkracht te achterhalen.

7.3.3 Verificatie achterhaalde sleutel

Wanneer een sleutel achterhaald is, wordt geverifieerd of de hiermee ontcijferde gegevens zodanig te interpreteren (leesbaar) zijn, dat de sleutel inderdaad de juiste is. Neem als voorbeeld de ontcijfering met een sleutel van met Full Disk Encryption vercijferde gegevens. De ontcijferde gegevens zullen in dit geval over het algemeen een valide bestandssysteem bestaande uit systeem- en gebruikersbestanden en -mappen moeten bevatten.

8 Bijlage 3: formaten van afbeeldingen en videobestanden

Afbeeldingen	
JPG	Het populairste fotoformaat, omdat de meeste camera's en telefoons .jpg- of .jpeg-bestanden maken. Bestanden kunnen tijdens het opslaan flink worden verkleind, maar naarmate de compressie groter is, zal de beeldkwaliteit sterker teruglopen: de foto wordt onscherper en vlekkerig. In fotobewerkingsprogramma's is de mate van compressie in te stellen.
BMP	Een oud formaat van Microsoft waarin beeld zonder compressie (zonder kwaliteitsverlies) wordt opgeslagen. Een .BMP is dan ook snel erg groot en werkt alleen met Windows-computers.
GIF	Een GIF-plaatje kan maar 256 kleuren bevatten en is daarom alleen geschikt voor simpele tekeningen en kleine animaties. De bestanden zijn klein. .PNG is de opvolger van de .GIF
TIF	Een oudere bestandsindeling die in 2009 door Aldus werd ontwikkeld voor afdrukken in hoge kwaliteit. Ook populair voor commercieel afdrukken, scannen en computer- en faxtoepassingen. Er gaat geen kwaliteit verloren zoals bij .jpg en er zijn verschillende compressiemogelijkheden. Hoewel de indeling niet kan worden gebruikt voor webafbeeldingen, wordt TIF geprezen om zijn flexibiliteit.
EPS	Scherpe lijnen werken goed voor tekst, informatieve illustraties en lijntekeningen. Vectorafbeeldingen kunnen eenvoudig worden bewerkt en Pantone of andere steunkleuren voor commerciële printtoepassingen kunnen gemakkelijk worden toegepast. De uitzondering is Photoshop EPS, dat eigenlijk een TIF-bestand is met een EPS-extensie.
RAW	Ideaal voor fotografie, omdat deze de meeste gegevens van uw camera vastlegt. Zo krijgt u de mooist mogelijke afbeeldingen. Een minpunt is dat de mogelijkheden om foto-indelingen te bewerken beperkt zijn tot Photoshop en de software die met uw camera wordt meegeleverd.
PNG	Kleine bestandsgrootte en werkt goed op internet. Ondersteunt transparantie en gebruikt alleen geïndexeerde kleuren.
PDF	Hoewel PDF niet echt een afbeeldingsindeling is, is PDF de hedendaagse universele bestandsindeling voor topprofessionals. Een PDF-bestand wordt meestal gemaakt in een andere indeling en daarna geconverteerd. De meeste gebruikers kunnen zo geen aanpassingen maken, waardoor de indeling ideaal is voor e-books, formulieren en grotere afbeeldingen. De kwaliteit kan hoog of laag zijn, afhankelijk van hoe het PDF-bestand wordt gemaakt.
Video	
MPG	Een universeel videoformaat dat 25 jaar geleden is ontworpen voor televisie. Het bevat de compressietechnieken (codecs) MPEG-1 of -2; bij het opslaan gaat kwaliteit verloren.
AVI	Een veelgebruikt, maar verouderd videoformaat op Windows-PC's. .AVI kan veel verschillende codecs voor video en audio bevatten wat tot afspeelproblemen kan leiden.
MP4	Opvolger van MPEG-2 en ontworpen voor live streaming via het internet. .MP4-bestanden zijn sterker gecomprimeerd dan .MPG-bestanden.
WMV	Standaard van Microsoft gebaseerd op .MPEG-4. Het is vrij algemeen geaccepteerd en daardoor met veel software af te spelen. Werkt niet op de Mac van Apple.
MOV	Een extensie die hoort bij de QuickTime-standaard van Apple. Voor het afspelen is de QuickTime Player nodig.

3GP	Specifiek ontwikkeld voor mobiele telefoons, is 3GP een simpele versie van MP4. In vergelijking met het MP4 formaat dat op PCs word gebruikt, is 3GP veel kleiner, wat het makkelijker maakt om op mobiele toepassingen te bekijken. Er zijn twee soorten 3GP formaten. De eerste is het 3GP formaat met de bestandsextensie .3gp, en 3G2 met de extensie .3g2.
DIVX	DivX is een standaard om digitale videobestanden compact op te slaan door gebruik te maken van een compressiealgoritme dat geoptimaliseerd is voor videobeelden. DivX is een zeer populair videoformaat op het internet. Via torrentprogramma's worden duizenden films (illegaal) te downloaden aangeboden.
Toelichting: In deze tabel zijn de meest voorkomende soorten van beeld en video bestanden omschreven. Bronnen: https://www.compx.nl/2016/03/08/de-meestgebruikte-bestandsformaten-op-een-rijtje/ https://www.shutterstock.com/nl/support/article/Veelvoorkomende-bestandsindelingen-van-afbeeldingen https://wikipedia.nl/	