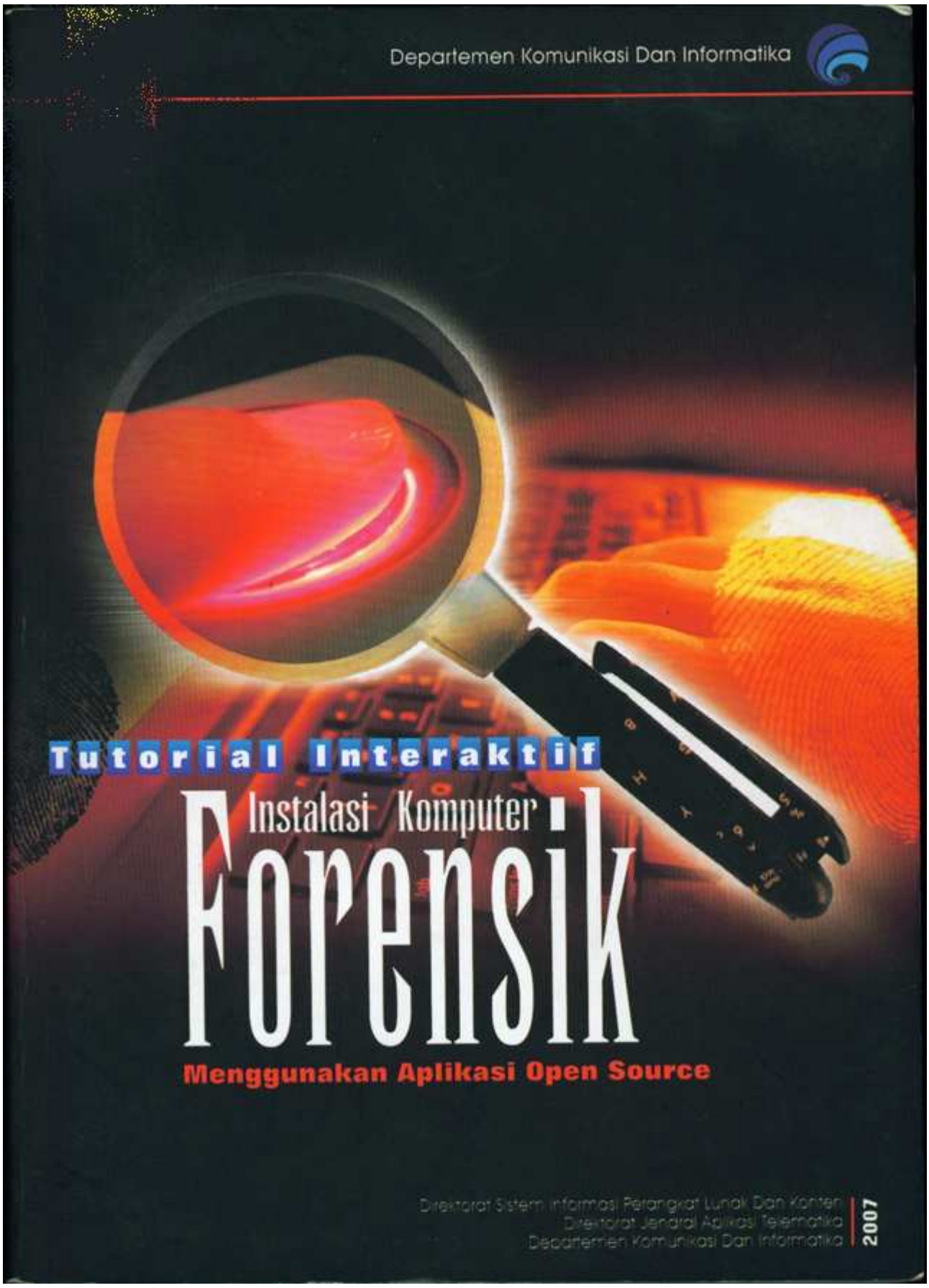

Departemen Komunikasi Dan Informatika
Tutorial Interaktif
Instalasi Komputer
Forensik
Menggunakan Aplikasi Open Source
Direktorat Sistem Informasi Perangkat Lunak Dan Konten
Direktorat Jenderal Aplikasi Telematika
Departemen Komunikasi Dan Informatika
2007

Tutorial Interaktif

Instalasi Komputer Forensik

Menggunakan Aplikasi Open Source

Achmad Syafa'at
achmad_mail@yahoo.com

Direktorat Sistem Informasi, Perangkat Lunak Dan Konten
Direktorat Jenderal Aplikasi Telematika
Departemen Komunikasi Dan Informatika
2007

1. Komputer Forensik

1.1. LatarBelakang dan Sejarah Komputer Forensik

Saat ini teknologi komputer dapat digunakan sebagai alat bagi para pelaku kejahatan komputer : seperti pencurian, penggelapan uang dan lain sebagainya. Barang bukti yang berasal dari komputer telah muncul dalam persidangan hampir 30 tahun. Awalnya, hakim menerima bukti tersebut tanpa membedakannya dengan bentuk bukti lainnya. Namun seiring dengan kemajuan teknologi komputer, perlakuan tersebut menjadi membingungkan.

Bukti yang berasal dari komputer sulit dibedakan antara yang asli ataupun salinannya, karena berdasarkan sifat alaminya, data yang ada dalam komputer sangat mudah dimodifikasi. Proses pembuktian bukti tindak kejahatan tentunya memiliki kriteria-kriteria, demikian juga dengan proses pembuktian pada bukti yang didapat dari komputer.

Di awal tahun 1970-an Kongres Amerika Serikat mulai merealisasikan kelemahan hukum yang ada dan mencari solusi terbaru yang lebih cepat dalam penyelesaian kejahatan komputer. US Federals Rules of Evidence 1976 menyatakan permasalahan tersebut. Hukum lainnya yang menyatakan permasalahan tersebut adalah:

- Economic Espionage Act 1996, berhubungan dengan pencurian rahasia dagang
- The Electronic Comunications Privacy Act 1986, berkaitan dengan penyadapan peralatan elektronik.
- The Computer Security Act 1987 (Public Law 100-235), berkaitan dengan keamanan sistem komputer pemerintah

1.2. Definisi Komputer Forensik

Berikut ini adalah beberapa buah definisi komputer forensik:

• Definisi sederhana : penggunaan sekumpulan prosedur untuk melakukan pengujian secara menyeluruh suatu sistem komputer dengan menggunakan software dan tool untuk mengambil dan memelihara barang bukti tindakan kriminal.
• Menurut Judd Robin, seorang ahli komputer forensik : "Penerapan secara sederhana dari penyelidikan komputer dan teknik analisisnya untuk menentukan bukti-bukti hukum yang mungkin".

- *New Technologies memperluas definisi Judd Robin dengan: "Komputer forensik berkaitan dengan pemeliharaan, identifikasi, ekstraksi dan dokumentasi bukti-bukti komputer yang tersimpan dalam wujud informasi magnetik".*
- *Menurut Dan Farmer & Wietse Venema : "Memperoleh dan menganalisa data dengan cara yang bebas dari distorsi atau bias sebisa mungkin, untuk merekonstruksi data atau apa yang telah terjadi pada waktu sebelumnya di suatu sistem".*

1.3. Kebutuhan akan Forensik

Dalam satu dekade terakhir, jumlah kejahatan yang melibatkan komputer telah meningkat pesat, mengakibatkan bertambahnya perusahaan dan produk yang berusaha membantu penegak hukum dalam menggunakan bukti berbasis komputer untuk menentukan siapa, apa, di mana, kapan, dan bagaimana dalam sebuah kejahatan. Akibatnya, komputer forensik telah berkembang untuk memastikan presentasi yang tepat bagi data kejahatan komputer di pengadilan. Teknik dan tool forensik seringkali dibayangkan dalam kaitannya dengan penyelidikan kriminal dan penanganan insiden keamanan komputer, digunakan untuk menanggapi sebuah kejadian dengan menyelidiki sistem tersangka, mengumpulkan dan memelihara bukti, merekonstruksi kejadian, dan memprakirakan status sebuah kejadian. Namun demikian, tool dan teknik forensik juga dapat digunakan untuk tugas-tugas lainnya, seperti :

- *Operational Troubleshooting. Banyak tool dan teknik forensik dapat digunakan untuk melakukan troubleshooting atas masalah-masalah operasional, seperti menemukan lokasi fisik dan virtual sebuah host dengan konfigurasi jaringan yang tidak tepat, mengatasi masalah fungsional dalam sebuah aplikasi.*
- *Log Monitoring. Beragam tool dan teknik dapat membantu dalam melakukan monitoring og, seperti menganalisis entri log dan mengkorelasi entri log dari beragam sistem. Hal ini dapat membantu dalam penanganan insiden, mengidentifikasi pelanggaran kebijakan, audit, ddan usaha lainnya.*
- *Data Recovery. Terdapat lusinan tool yang dapat mengembalikan data yang hilang dari sistem, termasuk data yang telah dihapus atau dimodifikasi baik yang disengaja maupun tidak.*
- *Data Acquisition. Beberapa organinasi menggunakan tool forensik untuk mengambil data dari host yang telah dipensiunkan. Sebagai contoh, ketika seorang user meninggalkan organisasi, data dari komputer user tersebut dapat diambil dan disimpan bilamana dibutuhkan di masa mendatang. Media komputer tersebut lalu dapat disanitasi untuk menghapus semua data usertersebut.*

- *Due Diligence/Regulatory Compliance. Regulasi yang ada dan yang akan muncul mengharuskan organisasi melindungi informasi sensitif dan memelihara beberapa catatan tertentu demi kepentingan audit. Juga, ketika informasi yang dilindungi terekspos ke pihak lain, organisasi mungkin diharuskan untuk memberitahu pihak atau individu yang terkena dampaknya. Forensik dapat membantu organisasi melakukan* due diligence *dan mematuhi persyaratan tersebut.*

1.4. Barang Bukti Digital

Bukti digital adalah informasi yang didapat dalam bentuk / format digital (scientific Working Group on Digital Evidence, 1999). Beberapa contoh bukti digital antara lain:

E-mail, alamat e-mail
Filewordprocessor/spreadsheet
Source code perangkat lunak
File berbentuk image(.jpeg, .tip, dan sebagainya)
Web Browser bookmarks, cookies
Kalender, to-do list

Bukti digital tidak dapat langsung dijadikan barang bukti pada proses peradilan, karena menurut sifat alamiahnya bukti digital sangat tidak konsisten. Untuk menjamin bahwa bukti digital dapat dijadikan barang bukti dalam proses peradilan maka diperlukan sebuah standar data digital yang dapat dijadikan barang bukti dan metode standar dalam pemrosesan barang bukti sehingga bukti digital dapat dijamin keasliannya dan dapat dipertanggung jawabkan.

Berikut ini adalah aturan standar agar bukti dapat diterima dalam proses peradilan:

- *Dapat diterima, artinya data harus mampu diterima dan digunakan demi hukum, mulai dari kepentingan penyelidikan sampai dengan kepentingan pengadilan.*
- *Asli, artinya bukti tersebut harus berhubungan dengan kejadian / kasus yang terjadi dan bukan rekayasa.*
- *Lengkap, artinya bukti bisa dikatakan bagus dan lengkap jika di dalamnya terdapat banyak petunjuk yang dapat membantu investigasi.*
- *Dapat dipercaya, artinya bukti dapat mengatakan hal yang terjadi di belakangnya. Jika bukti tersebut dapat dipercaya, maka proses investigasi akan lebih mudah.*

Syarat dapat dipercaya ini merupakan suatu keharusan dalam penanganan perkara.

Untuk itu perlu adanya metode standar dalam pegambilan data atau bukti digital dan pemrosesan barang bukti data digital, untuk menjamin keempat syarat di atas terpenuhi. Sehingga data yang diperoleh dapat dijadikan barang bukti yang legal di pengadilan dan diakui oleh hukum.

1.5. Metodologi Standar

Pada dasarnya tidak ada suatu metodologi yang sama dalam pengambilan bukti pada data digital, karena setiap kasus adalah unik sehingga memerlukan penanganan yang berbeda. Walaupun demikian dalam memasuki wilayah hukum formal, tentu saja dibutuhkan suatu aturan formal yang dapat melegalkan suatu investigasi.

Untuk itu menurut U.S. Department of Justice ada tiga hal yang ditetapkan dalam memperoleh bukti digital:

- *Tindakan yang diambil untuk mengamankan dan mengumpulkan barang bukti digital tidak boleh mempengaruhi integritas data tersebut.*
- *Seseorang yang melakukan pengujian terhadap data digital harus sudah terlatih.*
- *Aktivitas yang berhubungan dengan pengambilan, pengujian, penyimpanan atau pentransferan barang bukti digital harus didokumentasikan dan dapat dilakukan pengujian ulang.*

Selain itu terdapat pula beberapa panduan keprofesian yang diterima secara luas:

- *Pengujian forensik harus dilakukan secara menyeluruh. Pekerjaan menganalisa media dan melaporkan temuan tanpa adanya prasangka atau asumsi awal.*
- *Media yang digunakan pada pengujian forensik harus disterilisasi sebelum digunakan.*
- *Image bit dari media asli harus dibuat dan dipergunakan untuk analisa.*
- *Integritas dari media asli harus dipelihara selama keseluruhan penyelidikan.*

Dalam kaitan ini terdapat akronim PPAD pada Komputer forensik:

1. *Memelihara (Preserve) data untuk menjamin data tidak berubah.*
2. *Melindungi (Protect) data untuk menjamin tidak ada yang mengakses barang bukti.*

3. *Melakukan analisis (Analysis) data menggunakan teknik forensik.*
4. *Mendokumentasikan (Document) semuanya, termasuk langkah-langkah yang dilakukan.*

Dari berapa uraian di atas sudah sangat jelas bahwa tujuan pendefinisian metodologi standar adalah untuk melindungi bukti digital. Mengenai penentuan kebijakan dan prosedur teknis dalam pelaksanaan dapat disusun kemudian oleh instansi yang terkait, tentunya dengan mengacu pada metode-metode standar yang telah ada dan disesuaikan dengan hukum yang berlaku di negara yang bersangkutan. Dari beberapa metodologi di atas dapat digarisbawahi bahwa penggunaan bukti asli dalam investigasi sangat dilarang dan bukti ini harus dijaga agar jangan sampai ada perubahan di dalamnya karena akan sangat mempengaruhi kesimpulan yang diambil,

1.6. Pemrosesan Barang Bukti

Barang bukti sangat penting keberadaanya karena sangat menentukan keputusan di pengadilan, untuk itu pemrosesan barang bukti dalam analisa forensik sangat diperhatikan. Berikut ini adalah panduan umum dalam pemrosesan barang bukti menurut Lori Wilier dalam bukunya "Computer Forensic":

- shutdown *komputer, namun perlu dipertimbangkan hilangnya informasi proses yang sedang berjalan*
- *dokumentasikan konfigurasi hardware sistem, perhatikan bagaimana komputer disetup karena mungkin akan diperlukan restore kondisi semula pada tempat yang aman*
  pindahkan sistem komputer ke lokasi yang aman buat backup secara bit-by-bitdan *hardisk dan floppy barang bukti asli uji keotentikan data pada semua perangkat penyimpanan dokumentasikan tanggal dan waktu yang berhubungan dengan file komputer buat daftar keyword pencarian evaluasi swap file, evaluasi file slack evaluasi unallocated space* (erased file) *pencarian keyword pada file, file slack, dan unallocated space dokumentasikan nama file, serta atribut tanggal dan waktu identifikasikan anomali file, program untuk mengetahui kegunaannya dokumentasikan temuan dan software yang dipergunakan buat salinan software yang dipergunakan*

Untuk memastikan bahwa media bukti digital tidak dimodifikasi, sebelum ia digunakan untuk duplikasi, ia harus diset ke "Read Only", locked" atau "Write Protect", untuk mencegah terjadinya modifikasi yang tidak disengaja. Secara baku, SLAX4 menset seluruh device sebagai read only, *sehingga mereka tidak dapat dimodifikasi dengan mudah. Namun demikian, kami tetap menyarankan untuk melindungi media digital tersebut menggunakan* hardware write protector.

2. Data Digital

2.1. File

File adalah kumpulan informasi yang secara logika dikelompokkan ke dalam kesatuan tunggal dan diacu dengan menggunakan suatu nama unik. Suatu file dapat berupa banyak tipe data, termasuk dokumen, gambar, video atau aplikasi. Pengujian media komputeryang sukses bergantung pada kemampuan mengumpulkan, memeriksa, dan meneliti file yang berada pada media itu.

Sebelum mencoba memperoleh atau menguji file, analis perlu memahami pengetahuan dasar file dan filesistem. Bagian 2.1.2 menjelaskan bagaimana filesistem digunakan untuk mengorganisir file, dan menyediakan suatu ikhtisar beberapa filesistem umum Bagian 2.1.3 mendiskusikan bagaimana data file yang dihapus masih terdapat di dalam filesistem. Analis perlu juga menyadari variasi media yang dapat berisi file; Bagian 2.1.1 memberikan beberapa contoh media utama digunakan di dalam komputer pribadi.

2.1.1. Media Penyimpanan File

Penggunaan komputer yang tersebar luas dan alat digital lain telah mengakibatkan peningkatan banyaknya jenis media berbeda yang digunakan untuk menyimpan file. Sebagai tambahan terhadap jenis media yang biasa digunakan seperti disket dan hard drives, file juga disimpan pada alat seperti PDA dan telepon selular, serta jenis media yang lebih baru, seperti flash card yang dipopulerkan dengan adanya kamera digital label berikut mendaftarkan jenis media yang digunakan pada komputer dan alat digital.

Daftar ini tidak meliputi setiap jenis media yang tersedia; melainkan untuk menunjukkan variasi jenis media yang perlu diketahui seorang analis.

Tipe Media	Kapasitas	Keterangan
Floppy disk	1.44MB	Disk berukuran 35 inch: popularitasnya mulai menurun
CD-ROM	650 - 800MB	Meliputi CD -R dan CD -RW; media yang biasanya banyak digunakan
DVD-ROM	1.67- 15.9GB	Meliputi DVD -R dan DVD -RW drive , baik single dan dual layer disks ,
Hard drive	20 300 GB	
Tipe Media	Kapasitas	Keterangan
Zip disk	100-7^0MB	
Jaz disk	1 2GB	Serupa dengan Zip disks; tidak lagi diproduksi
Backup tape	80MB - 320GB	Banyak menyerupai kaset tape au dio; peka terhadap kerusakan karenakondisi lingkungan
Magneto Optical	600MB-9.1GB	lebih sedikit peka untuk kondisi lingkungan dibanding backup tape
ATA flash card	8MB - 2GB	PCMCIA flash memory card; berukuran 85.6 x 54 x 5 mm
Flash/Jump drive	16MB -4GB	Merupakan media penyimpanan yang paling banyak digunakan. Kapasitas ukurannya semakin bertambah.
CompactFlash card	16MB-6GB	Kartu tipe 1 berukuran 43 x 36 x 3.3 mm; kartu tipe 2 berukuran 43 x 36 x 5 mm
MultiMediaCard	16MB-512MB	Berukuran 24 x 32 x 1 .4 mm
Secure Digital (SD) card	32 MB - 1 GB	Memenuhi kebutuhan dengan Secure DigitalMusic Initiative (SDMI); menyediakan data built -in yang dienkripsi; dari luamya serupa MMC
Memory Stick	16 MB - 2 GB	Mencakup Memory Stick (50 x 21.5 x 2.8 mm), Memory Stick Duo (3 1 x 20 x 1 6 mm), MemoryStick PRO, Memory Stick PRO Duo; beberapa memenuhi kebutuhan SDMI dan menyediakan enkripsi built-in
SmartMedia Card	8MB- 128MB	Berukuran 37 x 45 x 0.76 mm
xD-Picture Card	16MB 512MB	Sekarang ini hanya digunakan di dalam kamera digital Fujifilm dan Olympus ; berukuran 20 x 25 x 1 .7 mm

2.1.2.FileSistem

Sebelum media dapat digunakan untuk menyimpan data, biasanya media tersebut harus dipartisi dan diformat ke dalam logical volume terlebih dulu. Mempartisi adalah suatu aktivitas untuk membagi media secara logikal ke dalam bagian-bagian yang berfungsi sebagai unit fisik terpisah. Logical volume *adalah sebuah partisi atau kumpulan partisi yang berfungsi sebagai satu kesatuan yang telah diformat dengan suatu filesistem. Beberapa jenis media, seperti disket, dapat berisi paling banyak satu partisi (dan sebagai konsekuensi, satu logical volume). Format logical volume ditentukan oleh filesistem yang dipilih.*
Suatu filesistem menentukan cara file dinamai, disimpan, diorganisir dan diakses pada logical volumes. Terdapat beragam filesistem, masing-masing menyediakan fitur dan struktur data yang unik. Namun demikian, semua filesistem memiliki beberapa ciri umum. Pertama, mereka menggunakan konsep direktori dan file untuk mengorganisir dan menyimpan data. Direktori adalah struktur organisasional yang digunakan untuk mengelompokkan file. Selain file, direktori dapat berisi direktori lain yang disebut subdirektori. Kedua, filesistem menggunakan beberapa struktur data untuk menunjuk lokasi file pada media. Mereka juga menyimpan masing-masing file data yang ditulis ke media dalam satu atau lebih unit alokasi file. Hal ini dikenal sebagai cluster *oleh beberapa filesistem (misalnya File Allocation Table [FAT], NT File System [NTFS]) dan* blok *oleh filesistem lainnya (misalnya filesistem Unix dan Linux). Sebuah unit alokasi file adalah sebuah kelompok sektor, yang merupakan unit terkecil yang dapat diakses pada suatu media.*

Berikut ini adalah beberapa filesystem yang umum digunakan:

- *FAT12. FAT12 digunakan hanya pada disket dan volume FAT yang lebih kecil daripada 16 MB. FAT12 menggunakan sebuah entri FAT 12-bit untuk menunjuk entri dalam filesistem.*
- *FAT16. MS-DOS, Windows 95/98/Nt/2000/Xp, Server Windows 2003, dan beberapa sistem operasi UNIX mendukung FAT16 secara asli. FAT16 biasanya juga digunakan untuk alat multimedia seperti audio player dan kamera digital. FAT16 menggunakan sebuah entri FAT 16-bit untuk menunjuk entri dalam filesistem. Volume FAT16 terbatas hingga maksimum 2 GB di dalam MS-DOS dan Windows 95/98. Windows NT dan sistem operasi yang lebih baru meningkatkan ukuran volume maksimum FAT16 menjadi 4 GB.*
- *FAT32. Windows 95 OEM Service Release 2 (OSR2), Windows 98/2000/XP, dan Windows Server 2003 mendukung FAT32 secara asli, seperti halnya beberapa alat multimedia. FAT32 menggunakan sebuah entri FAT 32-bit untuk menunjuk entri dalam filesistem. Ukuran maksimum volume FAT32 adalah 2 terabytes (TB).*
- *NTFS. Windows NT/2000/XP dan Server Windows 2003 mendukung NTFS secara asli. NTFS adalah suatu filesistem dapat dipulihkan, yang berarti bahwa ia dapat secara otomatis mengembalikan konsistensi filesistem manakala terjadi kesalahan. Sebagai tambahan, NTFS mendukung data kompresi dan enkripsi, dan memungkinkan ijin tingkat user dan grup didefinisikan untuk file dan direktori. Ukuran maksimum volume NTFS adalah 2TB.*

- *Second Extended Filesystem (ext2fs). ext2fs didukung secara langsung oleh Linux. ext2fs mendukung jenis file dan pemeriksaan filesistem standar Unix untuk memastikan konsistensi filesistem. Ukuran volume ext2fs maksimum adalah 4 TB.*

- Third Extended Filesystem (extSfs) extSfs didukung secara langsung o!eh Linux extSfs didasarkan pada ext2fs filesystem dan menyediakan kemampuan menjurnai yang memungkmkan pemeriksaan konsistensi filesistem dilakukan dengan cepat pada sejumlah data yang besar. Ukuran volume extSfs maksimum adalah 4 TB.
- ReiserFS.21 ReiserFS didukung oleh Linux dan merupakan filesistem baku bagi beberapa versi Linux, la memberikan kemampuan menjurnai dan jauh lebih cepat dibandingkan filesistem ext2fs dan extSfs. Ukuran volume maksimum adalah 16 TB.
- Hierarchical File System (HFS). HFS didukung secara langsung oleh Mac OS. HFS sebagian besar digunakan di dalam versi lama Mac OS tetapi masih didukung dalam versi lebih baru. Ukuran volume maksimum HFS di Mac OS 6 dan 7 adalah 2 GB. Ukuran volume maksimum HFS dalam Mac OS 7.5 adalah 4 GB. Mac O 7.5.2 dan sistem operasi Mac yang terbaru meningkatkan ukuran volume maksimum HFSmenjadi2TB.
- HFS Plus. HFS Plus didukung secara langsung oleh Mac OS 8.1 dan versi selanjutnya dan ia merupakan sebuah filesistem berjurnal di Mac OS X. HFS Plus adalah penerus HFS dan memberikan banyak peningkatan seperti mendukung nama file yang panjang dan nama file Unicode untuk penamaan file internasional. Ukuran volume maksimum HFS Plus adalah 2 TB.
- Unix File System (UFS). UFS didukung secara asli oleh beberapa jenis sistem operasi Unix, termasuk Solaris, FreeBSD, OpenBSD, dan Mac OS X Namun demikian, kebanyakan sistem operasi sudah menambahkan fitur khusus, sehingga detil UFS berbeda antar implementasi.
- Compact Disk File System (CDFS). Seperti indikasi namanya, CDFS filesistem digunakan untuk CD.
- International Organization for Standardization (ISO) 9660 dan Joliet ISO 9660 filesistem biasanya digunakan pada CD-ROM.

Filesistem CD-ROM yang popular lainnya adalah Joliet, suatu varian ISO9660. ISO 9660 mendukung panjang nama file sampai 32 karakter, sedangkan Joliet mendukung sampai 64 karakter. Joliet juga mendukung karakter Unicode di dalam pemberian nama file.

- Universal Disk Format (UDF). UDF adalah filesistem yang digunakan untuk DVD dan juga digunakan untuk beberapa CD.

2.1,3. Data lain pada Media

Seperti yang diuraikan di dalam bagian 2.1.2, filesistem dirancang untuk menyimpan file *pada* suatu media. Namun demikian, filesistem dapat pula berisikan data dan dihapusnya file atau versi lebih awal dari file yang ada. Data ini dapat menyediakan informasi penting

Beberapa hal berikut menguraikan bagaimana data masih dapat berada pada berbagai media:

- File yang dihapus. Manakala suatu file dihapus, umumnya ia tidak dihapus dan media; melainkan informasi struktur data direktori yang menunjuk ke lokasi file ditandai sebagai terhapus. Hal ini berarti file masih disimpan pada media tetapi hal itu tidak lagi dihitung oleh sistem operasi tersebut. Sistem operasi menganggap ini sebagai ruang kosong dan dapat mengisi sebagian atau seluruh file terhapus kapanpun dibutuhkan.
- Slack Space. Seperti dijelaskan sebelumnya, filesistem menggunakan unit-unit alokasi file untuk menyimpan file. Sekalipun suatu file memerlukan ruang lebih sedikit daripada ukuran unit alokasi file, seluruh unit alokasi file masih disediakan untuk file itu. Sebagai contoh, jika ukuran unit alokasi file adalah 32 KB dan suatu file hanya berukuran 7 KB, keseluruhan 32 KB dialokasikan untuk file tetapi hanya 7 KB yang digunakan, akibatnya 25 KB ruang yang tak terpakai. Ruang yang tak terpakai ini dikenal sebagai file slack space, ia dapat berisikan data sisa seperti bagian file yang dihapus.
- Free Space. Free Space adalah area pada media yang tidak dialokasikan untuk partisi apapun. la terdiri dari cluster atau blok yang tidak teralokasi. Hal ini sering meliputi ruang pada media tempat file (dan bahkan keseluruhan volume) mungkin berada pada satu waktu namun kemudian telah dihapus Ruang kosong masih dapat berisi potongan data.

Cara lain data mungkin disembunyikan adalah melalui Alternate Data Streams (ADS) di dalam volume NTFS. NTFS telah lama mendukung berbagai arus data untuk direktori dan file. Masing-masing file pada suatu volume NTFS terdiri dari suatu stream tak bernama yang digunakan untuk menyimpan data primer file, dan satu atau lebih stream bernama (misalnya file.txt:Stream1, file.txt:Stream2) yang dapat digunakan untuk menyimpan informasi tambahan seperti properti file dan gambar thumbnail data Sebagai contoh, jika seorang user mengklik kanan pada suatu file di dalam Windows Explorer, melihat properti file dan kemudian memodifikasi informasi yang diperlihatkan di dalam summary tab, OS menyimpan ringkasan informasi file dalam arus bernama *(named stream)*.

Semua arus data dalam suatu file berbagi atribut file (seperti timestamp, atribut keamanan). Walaupun stream bernama mempengaruhi kuota penyimpanan file, mereka sebagian besar disembunyikan dari user sebab utilitas file Windows standar seperti Explorer hanya melaporkan ukuran stream tak bernama *(unnamed stream)* file. Akibatnya, user tidak dapat menentukan apakah suatu file berisi ADS dengan menggunakan utilitas file Windows standard. Hal ini memungkinkan data tersembunyi berada dalam filesistem NTFS. Memindahkan file dengan ADS ke filesistem non-NTFS secara efektif melepaskanADS dari file, maka ADS dapat hilangjika analis tidak paham akan kehadiran mereka.

2.2. *Mengumpulkan File*

Selama mengumpulkan data, analis perlu membuat satu atau lebih salinan file atau filesistem yang relevan. Analis kemudian bisa bekerja dengan salinan file tanpa mempengaruhi file yang asli. Bagian 2.2.1 menguraikan tools dan teknik yang utama untuk penyalinan file dan data file sisa dari suatu media. Bagian 2.2.2 mendiskusikan pentingnya pemeliharaan integritas file dan memberikan panduan tentang hardware dan software yang dapat membantu memelihara dan memverifikasi integritas file. Seringkali penting untuk mengumpulkan tidak hanya file, tetapi timestamp signifikan untuk file, seperti saatterakhir file diakses atau dimodifikasi.

Bagian 2.2.3 menguraikan timestamp dan menjelaskan bagaimana mereka dapat dipelihara. Isu teknis lain yang berhubungan dengan pengumpulan data, seperti menemukan file yang tersembunyi dan penyalinan file dari implementasi Redundant Arrays of Inexpensive Disks (RAID) ditunjukkan dalam Bagian 2.2.4.

2.2.1. Penyalinan file dari media

File dapat disalin dari media menggunakan dua teknik berbeda sebagai berikut:

- Logical Backup. Suatu logical backup menyalin file dan direktori logical volumes. Hal tersebut tidak menangkap data lain yang mungkin ada pada media, seperti file yang dihapus atau data sisa yang disimpan di dalam slack space.
- Physical Backup. Juga yang dikenal sebagai *disk imaging,* bit stream imaging menghasilkan penyalinan bit-per-bit media asli, termasuk free space dan slack space. Bit stream image memerlukan ruang penyimpanan yang lebih besar dan membutuhkan waktu lebih panjang untuk pelaksanaannya daripada logical backup.

Jika bukti dibutuhkan untuk tindakan hukum atau disiplmer, analis harus memperoleh image bit stream dari media asli, memberi

label media asli, dan menyimpannya secara aman sebagai bukti. Analisis berikutnya harus dilakukan pada media salinan untuk memastikan bahwa media asli tidak dimodifikasi dan bahwa salinan media asli dapat selalu dibuat bila dibutuhkan. Seluruh langkah yang diambil untuk membuat salinan image harus didokumentasikan. Hal ini akan memungkinkan analis lain menghasilkan salinan tepat media asli menggunakan prosedur yang sama. Sebagai tambahan, dokumentasi yang tepat dapat digunakan untuk mendemonstrasikan bahwa bukti tidak ditangani dengan sembarangan selama proses pengumpulan. Selain langkah-langkah yang diambil untuk merekam image, analis juga harus mendokumentasikan informasi tambahan seperti model dan nomor serial hard drive, kapasitas media penyimpanan, dan informasi mengenai software atau hardware imaging yang digunakan (misalnya, nama, nomor versi, informasi lisensi). Kesemua tindakan ini mendukungterpeliharanya *the chain of custody.*

Ketika image bit stream dieksekusi, kita dapat melakukan penyalinan disk-to-disk atau disk-to-file. Penyalinan disk-to-disk, menyalin seluruh isi media secara langsung ke media lain Penyalinan disk-to-file menyalin isi media ke sebuah file data logikal tunggal. Penyalinan disk-to-disk bermanfaat karena media yang disalin dapat dihubungkan langsung ke sebuah komputer dan isinya dapat segera dilihat. Namun dalam penyalinan disk-to-disk media kedua harus sama dengan media asli. Penyalinan disk-to-file memungkinkan image file data dipindah dan dibackup dengan mudah. Namun untuk melihat isi logikal file image, analis harus merestore image ke media atau membuka atau membacanya dari aplikasi yang mampu menampilkan isi logikal image bit stream

Beragam tool hardware dan software dapat melakukan imaging bit stream dan backup logikal. Tool hardware umumnya portabel, terhubung secara langsung ke drive atau komputer yang ingin diimage, dan memiliki fungsi hash built-in. Solusi software umumnya terdiri dari disket startup, CD atau program terinstalasi yang berjalan pada stasiun kerja yang terhubung dengan media yang ingin diimage.

Sebagai tambahan atas fungsi utama mereka, beberapa tool imaging disk juga melakukan pencatatan record forensik, seperti pelacakan audit otomatis dan *chain of custody.* Dengan semakin banyaknya tool imaging disk, NIST Computer Forensics Tool Testing (CFTT) telah mengembangkan prosedur testing yang rinci untuk memvalidasi hasil tool tersebut. Saat ini, baru sedikit tool imaging disk yang mengikuti testing tersebut.

Umumnya, tool yang melakukan imaging bit stream tidak digunakan untuk memperoleh salinan bit-by-bit seluruh device fisik dari sistem live, karena file dan memori pada sistem tersebut berubah secara konstan dan karenanya tidak dapat divalidasi. Namun demikian, penyalinan bit-by-bit logikal pada sistem live dapat diselesaikan dan divalidasi. Ketika dilakukan backup logikal, tetap disarankan untuk tidak menyalin file dari sistem live; perubahan mungkin dilakukan ke file selama proses backup, dan file yang dibuka oleh sebuah proses mungkin tidak dapat disalinkan. Selain itu, analis harus memutuskan apakah menyalin file dari sistem live *feasible* berdasarkan pada file apa yang ingin diperoleh, seberapa akurat dan lengkap penyalinan dibutuhkan, dan seberapa penting sistem live tersebut. Sebagai contoh, tidaklah perlu mematikan sebuah server kritis yang digunakan oleh ratusan orang hanya untuk mengambil file dari direktori home seorang user.

2.2.2. Integritas File Data

Selama backup dan imaging, integritas media asli harus dipelihara. Untuk memastikan bahwa proses backup atau imaging tidak merubah data pada media yang asli, analis dapat menggunakan write-blocker ketika memback-up atau mengimage media. Write-blocker adalah tool hardware atau software yang mencegah komputer menulis ke media penyimpanan komputer yang terhubung dengannya. Write-blocker hardware secara fisik terhubung ke komputer dan media penyimpanan yang sedang diproses untuk mencegah penulisan apapun ke media tersebut. Write-blocker software diinstalasi pada sistem analis dan saat ini hanya tersedia sistem MS-DOS dan Windows. (Beberapa sistem operasi, misalnya Mac OS dan Linux, mungkin tidak memerlukan write-blocker software karena mereka dapat diset untuk boot ke device sekunder yang belum dikaitkan). Bagaimanapun, memasang suatu alat perangkat keras writeblocking akan memastikan terpeliharanya integritas. Software write-blocker berbasis MS-DOS bekerja dengan menjebak Interrupt 13 dan extended Interrupt 13 disk write. Software write-blocker berbasis Windows menggunakan filter untuk mengurutkan Interrupt yang dikirim ke alat untuk mencegah penulisan ke media penyimpanan.

Secara umum, ketika menggunakan write-blocker hardware, device atau media yang digunakan untuk membaca media harus terhubung langsung ke write-blocker, dan write-blocker harus dihubungkan ke komputer atau device yang digunakan untuk melakukan backup. Ketika menggunakan write-blocker software, softwarenya harus sudah ada dalam komputer sebelum alat atau media yang digunakan untuk membaca media dihubungkan ke komputer itu. Write-blocker dapat juga mengijinkan write-blocking untuk diset on atau off bagi alat tertentu. Adalah penting ketika write-blocking digunakan, ia harus diset on untuk semua alat yang dihubungkan. Write-blocker juga harus diuji secara rutin untuk memastikan bahwa mereka mendukung alat lebih baru. Setelah suatu backup atau imaging dilakukan, penting untuk memverifikasi bahwa data yang disalinkan adalah salinan yang tepat dari data asli. Menghitung message digest data yang disalin dapat digunakan untuk memverifikasi dan memastikan integritas data.

Sebuah message digest adalah sebuah hash yang secara unik mengidentifikasi data dan memiliki sifat yaitu merubah satu bit data akan menghasilkan message digest yang berbeda sama sekali. Terdapat banyak algoritma untuk menghitung message digest data, namun dua buah yang umum digunakan adalah MD5 dan SHA1. Kedua algoritma ini mengambil data masukan dengan ukuran sembarang dan menghasilkan 128-bit message digest untuk MD5, sedangkan SHA1 menghasilkan 160-bit message digest.

Ketika dilakukan image bit stream, message digest media asli perlu dihitung dan disimpan sebelum image dilakukan. Setelah proses imaging, message digest media salinan perlu dihitung juga dan dibandingkan dengan message digest asli untuk memverifikasi bahwa integritas data telah dipelihara. Message digest media asli kemudian perlu dihitung kembali untuk memastikan bahwa proses imaging tidak merubah media asli, dan seluruh hasil harus didokumentasikan. Proses harus dilakukan untuk backup logikal, kecuali bahwa message digests harus dihitung dan dibandingkan untuk setiap file data. Untuk stream images dan logical backups, message digests dibuat untuk memastikan bahwa integritas data harus disimpan pada media read-only atau write-once atau dicetak lalu diamankan pada lokasi yang tepat.

2.2.3. Modifikasi File, Akses, dan Waktu penciptaan

Seringkali penting untuk mengetahui kapan suatu file digunakan atau dimanipulasi, dan kebanyakan sistem operasi mencatat timestamps tertentu yang terkait dengan file. Timestamps yang biasanya digunakan adalah waktu modifikasi, akses, dan penciptaan *modification, access,* and *creation',* MAC), sebagai berikut:

- Waktu Modifikasi Ini adalah waktu terakhir file diubah dengan berbagai cara, meliputi ketika suatu file ditulis dan ketika file tersebut diubah oleh program lain.
- Waktu Akses. Ini adalah waktu terakhir dilakukannya akses apapun pada file (misalnya, dilihat, dibuka, dicetak).
- Waktu penciptaan. Ini biasanya merupakan waktu dan tanggal file diciptakan. Bagaimanapun, ketika file disalinkan ke sistem, waktu penciptaan akan menjadi waktu file dicopy ke sistem yang baru.Waktu modifikasi akan tetap utuh.

Filesistem yang berbeda mungkin saja menyimpan jenis waktu yang berbeda. Sebagai contoh, Sistem Windows menyimpan waktu modifikasi terakhir, waktu akses terakhir, dan waktu penciptaan file. Sistem UNIX menyimpan waktu modifikasi terakhir, perubahan inode terakhir, dan waktu akses terakhir, namun beberapa sistem UNIX (termasuk versi BSD dan SunOS) tidak memperbaharui waktu akses terakhir file eksekutabel ketika mereka dijalankan. Beberapa sistem UNIX merekam waktu terkini ketika metadata file diubah. Metadata adalah data tentang data; untuk filesistem, metadata adalah data yang menyediakan informasi mengenai isi file.

Jika suatu analis ingin menetapkan garis waktu yang akurat atas suatu peristiwa, maka waktu file harus dipelihara. Analis harus sadar bahwa tidak semua metode untuk memperoleh file dapat memelihara waktu file. Image bit stream dapat mempertahankan waktu file karena dilakukan penyalinan bit-for-bit; melakukan logical backup menggunakan beberapa tool dapat menyebabkan waktu penciptaan file terubah ketika file data disalinkan. Oleh *karena* itu, bila waktu file penting, harus digunakan imaging bit stream untuk mengumpulkan data.

Analis juga harus menyadari bahwa waktu file tidak selalu akurat. Beberapa alasan ketidakakuratan itu adalah sebagai berikut:

- Jam komputer tidak mempunyai waktu yang benar itu. Sebagai contoh, jam mungkin tidak selalu disinkronisasi secara teratur dengan sumber waktu resmi.
- Waktu tidak mungkin direkam dengan tingkat detil yang diharapkan, seperti menghilangkan detikatau beberapa menit.
- Penyerang mungkin telah mengubah waktu file yang direkam.

2.2.4. IsuTeknis

Beberapa isu teknis mungkin muncul dalam memperoleh file data. Seperti yang diuraikan dalam Bagian 2.2.1, isu utama adalah memperoleh sisa-sisa file dan file yang dihapus yang masih ada dalam free space dan slack space pada media. Individu dapat menggunakan berbagai teknik untuk merintangi pengumpulan data seperti itu. Sebagai contoh, terdapat banyak tool yang tersedia untuk melaksanakan wiping—overwriting media (atau bagian media, seperti file tertentu) dengan nilai-nilai tetap atau acak (misalnya semua 0). Tool seperti itu berbeda dalam reliabilitas dan keandalan, tetapi umumnya efektif dalam mencegah pengumpulan file secara mudah, terutama bila dilakukan beberapa penghapusan. Individu dapat juga menggunakan alat-alat fisik untuk mencegah di dapatnya data, seperti demagnetizing harddrive (juga yang dikenal sebagai *degaussing)* atau secara fisik merusakkan atau menghancurkan media. Kedua teknik berbasis fisik dan software dapat membuat sangat sulit, atau bahkan mustahil, untuk memulihkan semua data yang menggunakan perangkat lunak. Usaha pemulihan dalam kasus ini mengharuskan penggunaan tenaga ahli khusus forensik dengan fasilitas, perangkat keras, dan teknik yang canggih, tetapi usaha dan biaya dalam pelaksanaan hal tersebut menjadi penghalang penggunaan cara ini secara umum. Dalam beberapa hal, data tidak dapat

dipulihkan.

Isu umum lainnya adalah memperoleh data yang tersembunyi. Banyak sistem operasi mengijinkan user menandai file, direktori, atau partisi tertentu sebagai tersembunyi, yang berarti secara baku mereka tidak ditampilkan di dalam listing direktori. Beberapa sistem operasi dan aplikasi menyembunyikan konfigurasi file untuk mengurangi kemungkinan user secara tidak sengaja memodifikasi atau menghapusnya. Juga, pada beberapa sistem operasi, direktori yang telah dihapus mungkin ditandai sebagai tersembunyi. Data yang tersembunyi dapat berisi banyak informasi; sebagai contoh, suatu partisi yang tersembunyi bisa berisi suatu sistem operasi terpisah dan banyak file data. User dapat menciptakan partisi yang tersembunyi dengan mengubah label partisi untuk mengganggu manajemen disk dan mencegah aplikasi melihat adanya area data. Data tersembunyi dapat juga ditemukan di dalam ADS pada volume NTFS, di akhirfile slack space dan free space pada medium, dan dalam Host Protected Area (HPA) pada beberapa hard drive, yang merupakan area drive yang ditujukan hanya untuk vendor. Banyak tool pengumpul data yang dapat mengenali beberapa atau semua metode penyembunyian data ini dan dapat memulihkan data yang terkait. Isu lain yang mungkin muncul adalah pengumpulan data dari array RAID yang menggunakan stripping (e.g., RAID-0, RAID-5). Di dalam konfigurasi ini, stripped volume terdiri atas partisi berukuran sama yang berada pada disk drive terpisah. Ketika data ditulis ke volume, ia didistribusikan secara merata ke seluruh partisi untuk meningkatkan performa disk. Hal ini akan menjadi masalah karena semua partisi dari stripped volume harus ada untuk menguji isinya, namun dalam hal ini partisi berada pada physical disk drives terpisah. Oleh karena itu, untuk menguji suatu stripped volume, masing-masing disk drive di dalam array RAID perlu diimage dan konfigurasi RAID harus dibuat ulang pada sistem pengujian. Sistem pengujian harus diboot menggunakan disk boot forensik yang dapat mengenali dan menggunakan array RAID dan dapat mencegah penulisan ke array. Beberapa tool dapat mengambil *stripped volume* dan mampu memelihara area data tak terpakai dari sebuah volume, seperti free space dan *slack space.*

2.3. Pengujian File

Setelah dilakukan backup logikal atau imaging bit stream, backup atau image harus direstorasi ke media lain sebelum data dapat diperiksa. Hal ini tergantung pada tool forensik yang akan digunakan untuk melakukan analisis. Beberapa tool dapat menganalisis data secara langsung dari file image, sedangkan tool lain mengharuskan backup atau image direstorasi dulu ke media. Tanpa mengindahkan apakah digunakan file image atau image yang telah direstore dalam pengujian, data harus diakses secara read-only untuk memastikan bahwa data yang diperiksa tidak dimodifikasi dan akan memberikan hasil yang konsisten pada proses berikutnya. Selama proses ini dapat digunakan write-blocker. Setelah mengembalikan backup (bila dibutuhkan), analis mulai menguji data yang terkumpul dan melakukan penilaian file-file dan data terkait dengan menemukan semua file, termasuk file terhapus, file sisa di slack dan free space, dan file tersembunyi. Kemudian, analis mungkin perlu mengekstraksi data dari beberapa atau semua file, yang mungkin dipersulit oleh enkripsi atau dilindungi password.

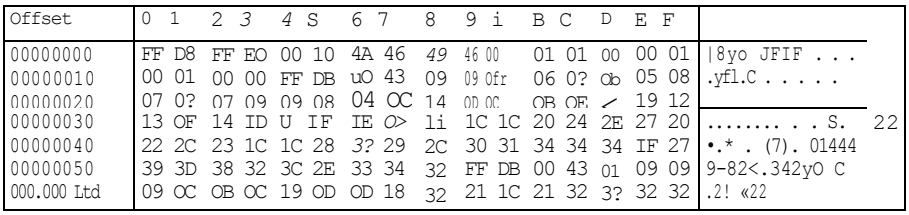
2.3.1. Mencari File

Langkah pertama dalam pengujian adalah mencari file. Sebuah image disk dapat menangkap banyak slack space dan free space, yang dapat berisikan ribuan file dan potongan file. Mengekstraksi data secara manual dari ruang tidak terpakai dapat merupakan proses yang butuh waktu dan sulit, karena ia membutuhkan pengetahuan format filesistem yang digunakan. Untungnya, beberapa tool tersedia untuk mengotomasi proses mengekstraksi data dari ruang tidak terpakai dan menyimpannya ke file data, dan juga mengembalikan file terhapus dan file yang berada dalam recycling bin. Analis dapat juga menampilkan isi slack space dengan editor heks atau tool recovery khusus slack.

2.3.2. Mengekstraksi Data

Proses pengujian selanjutnya mencakup ekstraksi data dari beberapa atau seluruh file. Untuk memahami isi file, seorang analis perlu mengetahui tipe data isi file. Tujuan ekstensi file adalah menjelaskan isi file; sebagai contoh ekstensi meng-indikasikan sebuah file grafis, dan ekstensi mp3 mengindi-kasikan file musik. Namun, user dapat memberikan sembarang ekstensi file ke sembarang tipe file, seperti menamakan file teks mysong.mp3 atau menghapus ekstensi file. Selain itu, beberapa ekstensi file mungkin tersembunyi atau tidak didukung pada sistem operasi lain. Karenanya, analis tidak boleh mengasumsikan bahwa ekstensi file adalah akurat. Analis dapat mengidentifikasikan tipe data yang disimpan dalam file dengan melihat header file tersebut. Sebuah header file berisikan informasi untuk mengidentifikasi file dan mungkin metadata yang memberikan informasi mengenai isi file. Seperti tampak pada gambar 2.1, header file berisikan signature file yang mengindikasikan tipe data isi file. Contoh pada gambar 2.1 memiliki header file FF D8, yang mengindikasikan bahwa ini adalah sebuah file JPEG. Teknik efektif lainnya untuk mengidentifikasi tipe data dalam sebuah file adalah histogram sederhana yang menunjukkan distribusi nilai ASCII sebagai persentase total karakter dalam file. Sebagai contoh. meningkatnya garis "spasi", "a", dan "e" umumnya

mengindikasikan sebuah file teks, sementara konsitensi di seluruh histogram mengindikasikan file terkompresi. Pola lain mengindikasikan file terenkripsi atau dimodifikasi melalui steganography.



Gambar 2.1. Informasi Header File

2.3.3. Toolkit

Analis harus memiliki akses ke beragam tool yang memungkinkan mereka melakukan pengujian dan analisis data, dan juga aktivitas pengumpulan. Banyak produk forensik memungkinkan analis melakukan sejumlah proses untuk menganalisis file dan aplikasi, serta mengumpulkan file, membaca image disk, dan mengekstraksi data dari file. Kebanyakan produk analis juga menawarkan kemampuan membuat laporan dan mencatat seluruh kesalahan yang terjadi selama analisis.

Proses-proses berikut adalah proses yang harus dapat dilakukan oleh analis dengan beragam tool:

- Menggunakan Penglihat *(viewer)* File. Menggunakan penglihat file alih-alih aplikasi sumber asli untuk menampilkan isi tipe tertentu file merupakan sebuah teknik penting untuk mempreview data, dan lebih efisien karena analis tidak membutuhkan aplikasi natif untuk melihat setiapjenis file
- Mendekompresi file. File terkompresi mungkin berisikan file dengan informasi berguna. Karenanya penting bagi analis untuk mencari dan membuka file terkompresi tersebut. Proses dekompresi harus dilakukan di awal proses forensik untuk memastikan bahwa isi file terkompresi disertakan dalam pencarian dan tindakan lain. Namun demikian, analis perlu memperhatikan bahwa file terkompresi dapat berisikan *malicious content,* seperti bom kompresi, yaitu file yang dikompresi berulang kali. Untuk meminimalkan dampaknya, mesin penguji harus menggunakan antivirus yang up-to-date dan merupakan sistem standalone. Selain itu image mesin pengujian harus dibuat, sehingga bila dibutuhkan, sistem dapat direstorasi.
- Menampilkan struktur direktori secara grafis. Praktek ini memudahkan dan mempercepat analis untuk mengumpulkan informasi umum tentang isi media, seperti tipe software yang terinstalasi dan kemampuan teknikal user yang membuat data.
- Identifikasi File Dikenal. Keuntungan menemukan file penting adalah jelas, namun juga bermanfaat untuk menghilangkan file-file tidak penting untuk dipertimbangkan, seperti file aplikasi dan sistem operasi yang dikenal bagus. Analis perlu menggunakan set hash yang telah tervalidasi, seperti yang dibuat oleh NIST National Software Reference Library (NSRL), sebagai dasar mengindentifikasi file jahat yang dikenal. Set hash biasanya menggunakan algoritma SHA-1 dan MD5 untuk memberikan nilai message digest untuk setiap file yang dikenal.
- Melakukan Pencarian String dan Pencocokan Pola. Pencarian string membantu dalam mencari kata kunci atau string dalam sekumpulan data. Contoh pencarian yang umum mencakup pencarian banyak kata dalam satu file dan pencarian versi salah eja kata tertentu. Beberapa hal yang perlu dipertimbangkan dalam melakukan pencarian string adalah sebagai berikut:

    - Beberapa format file proprietary tidak dapat dicari stringnya tanpa menggunakan tool tambahan. Selain itu, file terkompresi, terenkripsi atau dilindungi password membutuhkan proses tambahan sebelum dilakukan pencarian string.
    - Penggunaan set data multi-karacter yang menyertakan karakter asing atau Unicode dapat menyebabkan masalah dengan pencarian string.
    - Adanya keterbatasan tool atau algoritma pencarian. Sebagai contoh, sebuah kecocokan mungkin tidak ditemukan untuk pencarian string jika sebagian string ada dalam satu kluster dan sisa string berada dalam kluster lain yang tidak bersebelahan.

- Mengakses File Metadata. File metadata memberikan rincian tentang file tertentu. Sebagai contoh, mengumpulkan metadata tentang file grafis mungkin memberikan informasi mengenai waktu penciptaan, informasi hak cipta, dan deskripsi file. Metadata untuk grafis yang dihasilkan oleh kamera digital mungkin menyertakan pembuat dan model kamera digital yang digunakan untuk mengambil gambar, serta seting F-stop, flash, dan aperture.

2.3.4.Analisis

Setelah pengujian selesai, langkah berikutnya adalah melakukan analisis atas data yang diekstraksi. Seperti yang disebutkan

dalam Bagian 2.3.3, terdapat banyak tool yang dapat membantu dalam menganalisis berbagai tipe data. Ketika menggunakan tool ini atau melakukan review manual atas data, analis harus sadar tentang nilai waktu sistem dan file. Mengetahui kapan sebuah insiden terjadi, sebuah file tercipta atau dimodifikasi, atau sebuah email dikirim dapat bersifat kritikal bagi analisis forensik.

Sangatlah menguntungkan bagi analis bila sebuah orggnisasi memelihara pewaktuan sistemnya yang akurat. Network Time Protocol (NTP) mensinkronkan waktu pada komputerdengan sebuah jam atomik yang dijalankan oleh NIST atau organisasi lain.

Jika menggunakan banyak tool untuk menyelesaikan pengujian dan analisis, analis harus mengetahui bagaimana setiap tool mengekstraksi, memodifikasi, dan menampilkan waktu modifikasi, akses, dan penciptaan file. Write-blocker dapat digunakan untuk mencegah tool ini merubah waktu MAC; namun demikian, meskipun write-blocker dapat mencegah waktu dimodifikasi pada media, namun ia tidak dapat mencegah sistem operasi mencaching perubahan dalam memori (misalnya, menyimpan perubahan di random access memory [RAM]). Sistem operasi kemudian mungkin melaporkan waktu MAC yang dicache alih-alih waktu aktual.

Analis dapat menggunakan tool khusus yang dapat membuat garis waktu forensik berdasarkan data kejadian. Tool tersebut biasanya memberikan analis interface grafis untuk melihat dan menganalisis urutan kejadian. Fitur umum tool ini adalah mengijinkan analis mengelompokkan kejadian terkait ke meta-event. Hal ini membantu analis memperoleh "gambaran besar" kejadian.

Dalam banyak kasus, analisis forensik melibatkan tidak hanay data dari file, namun juga data dari sumber lainnya, seperti status sistem operasi, lalu lintas jaringan, atau aplikasi. Bagian berikut memberikan contoh bagaimana data dari file dan data dari sumber lainnya dikorelasi melalui analisis.

2.4. Penggunaan Data Sistem Operasi

Sebuah sistem operasi (OS) adalah program yang dijalankan pada komputer dan menyediakan platform perangkat lunak agar program lain dapat dijalankan. Sebagai tambahan, suatu OS bertanggung jawab untuk mengolah perintah masukan dari pemakai, mengirimkan keluaran ke layar, berinteraksi dengan alat penyimpanan untuk menyimpan dan mengambil data, serta mengendalikan perangkat lain seperti printer dan modem. Beberapa OS umum untuk server atau workstation meliputi berbagai versi Windows, Linux, Unix, dan Mac OS. Beberapa device jaringan, seperti router, mempunyai OS sendiri (misalnya Cisco Internetwork Operating System [IOS]). Bagian ini mendiskusikan komponen OS yang mungkin relevan bagi forensik dan menyediakan panduan untuk mengumpulkan, menguji, dan menganalisis data dari OS server dan workstation umum.

2.4.1. Dasar OS

Data OS ada dalam dua jenis, volatil dan non volatil. *Data non-volatil* mengacu pada data yang tetap ada bahkan setelah komputer dimatikan, seperti filesistem yang disimpan pada hard drive. *Data volatil* mengacu pada data sistem live yang hilang setelah komputer dimatikan, seperti koneksi jaringan saat ini ke dan dari sistem itu. Dari perspektif forensik, banyak jenis data volatil dan non volatil yang mungkin menarik.

Data Non Volatil

Sumber utama data non volatil dalam suatu OS adalah filesistem. Filesistem juga biasanya merupakan sumber data yang paling kaya dan yang paling besar di dalam OS, berisikan informasi yang dipulihkan selama peristiwa forensik. Filesistem menyediakan penyimpanan untuk OS pada satu atau lebih media. Suatu filesistem umumnya berisi banyak jenis file yang berbeda, yang mungkin bernilai bagi analis dalam situasi yang berbeda.

Daftar berikut menguraikan beberapa jenis data yang biasanya ditemukan di dalam filesistem OS:

- File Konfigurasi. OS dapat menggunakan file konfigurasi untuk menyimpan seting OS dan aplikasi. Sebagai contoh, file konfigurasi dapat mendaftarkan layanan yang dimulai secara otomatis setelah sistem diboot, dan menetapkan lokasi log files dan file temporer. User juga dapat memiliki file konfigurasi OS dan aplikasi sendiri yang berisi pilihan dan informasi spesifik untuk user, seperti pengaturan yang terkait dengan perangkat keras (misalnya, resolusi layar, setingan printer) dan asosiasi file. File konfigurasi yang menarik adalah sebagai berikut:

1. User dan Kelompok. OS menyimpan catatan tentang akun user dan kelompok. Informasi akun ini dapat meliputi keanggotaan kelompok, uraian dan nama akun, ijin akun, status akun (misalnya, aktif, ditiadakan), dan path ke direktori home akun.
2. File Sandi. OS mungkin menyimpan hash password dalam file data. Berbagai tool password cracking dapat digunakan untuk mengkonversi hash password ke clear text ekivalennya untuk OS tertentu.
3. Pekerjaan Yang dijadwalkan. OS memelihara daftar tugas yang dijadwalkan supaya dapat dilakukan secara otomatis pada waktu tertentu (misalnya, melaksanakan pemeriksaan virus tiap minggu). Informasi yang dapat diperoleh meliputi nama tugas, program yang digunakan untuk melaksanakan tugas, argumen dan switch perintah baris, waktu dan hari saat tugas dilakukan.

Log. File log OS berisi informasi tentang berbagai peristiwa sistem operasi, dan dapat juga berisi informasi peristiwa spesifik. Tergantung pada OS, log mungkin disimpan dalam file teks, file biner dengan format *proprietary,* atau database. Beberapa OS menulis entri log ke dua atau lebih file terpisah. Jenis informasi yang umum ditemukan dalam log OS adalah:

Peristiwa Sistem. Peristiwa sistem adalah tindakan operasional yang dilakukan oleh komponen OS, seperti mematikan sistem atau memulai suatu layanan. Umumnya, peristiwa gagal dan peristiwa penting yang sukses akan dicatat, tetapi banyak sistem operasi mengijinkan admin sistem menetapkan peristiwa mana yang akan dicatat Setiap peristiwa biasanya diberi penanda waktu; informasi pendukung lain dapat meliputi kode peristiwa, kode status, dan nama user.

Arsip Audit. Arsip audit berisi informasi peristiwa keamanan seperti sukses dan tidaknya usaha otentikasi dan perubahan kebijakan keamanan. OS umumnya mengijinkan admin sistem menetapkan jenis peristiwa mana yang harus diaudit.

3. Peristiwa Aplikasi. Peristiwa aplikasi adalah tindakan operasional penting yang dilakukan oleh aplikasi, seperti awal dan akhir aplikasi, kegagalan aplikasi, dan perubahan besar konfigurasi aplikasi.
4. Sejarah Perintah. Beberapa sistem operasi mempunyai log file yang terpisah (umumnya untuk setiap pemakai) yang berisi sejarah perintah OS yang dilakukan oleh pemakai tersebut. File Yang Baru Diakses. Suatu sistem operasi mungkin mencatat file terakhir yang diakses atau pemakaian lain, menciptakan daftar file yang terkini diakses.

File Aplikasi. Aplikasi dapat terdiri atas banyak jenis file, termasuk executable, skrip, dokumentasi, file konfigurasi, file log, file sejarah, grafik, suara, dan ikon. File Data. File data menyimpan informasi aplikasi. Beberapa contoh file data umum adalah file teks, pengolah kata dokumen, spreadsheet, database, file audio, dan file grafik.

- Swap Files. Kebanyakan OS menggunakan swap file bersama dengan random access memory (RAM) untuk menyediakan penyimpanan sementara bagi data yang sering digunakan aplikasi. Pada dasarnya swap file meningkatkan jumlah memori yang tersedia bagi suatu program dengan memungkinkan halaman (atau segmen) data untuk ditukarkeluar dan masuk RAM.
- Dump File. Beberapa OS memiliki kemampuan menyimpan isi memori secara otomatis selama kondisi kesalahan untuk membantu dalam troubleshooting berikutnya. File yang berisikan isi memori yang disimpan dikenal sebagai dump file.
- File Hibernasi. File hibernasi dibuat untuk menyimpan status sistem saat ini (khususnya adalah laptop) dengan merekam memori dan file terbuka sebelum mematikan sistem itu. Ketika sistem dinyalakan setelahnya, status sistem dikembalikan.
- File Sementara. Selama instalasi sistem operasi, aplikasi; update atau upgrade OS atau aplikasi, file sementara sering dibuat. Meskipun file ini umumnya dihapus pada akhir proses instalasi, hal ini tidak selalu terjadi. File sementara juga diciptakan ketika banyak aplikasi yang dijalankan, file ini akan dihapus ketika aplikasi berakhir, tetapi hal ini tidak selalu terjadi.

Walaupun filesistem adalah sumber utama data non volatil, sumber menarik lainnya adalah Sistem Input Output Dasar (Basic Input/Output System, atau sering dikenal dengan BIOS). BIOS berisi jenis informasi yang terkait dengan perangkat keras, seperti alat yang dipasang (misalnya CD-ROM drives, hard drives), jenis koneksi dan interrupt request line (IRQ) assignments (misalnya serial, USB, kartu jaringan), komponen-komponen motherboard (misalnya, tipe dan kecepatan prosesor, cache size, informasi memori), seting keamanan sistem, dan hot key. BIOS juga ber-komunikasi dengan driver RAID dan menampilkan informasi yang disajikan driver itu. Sebagai contoh, BIOS memandang perangkat keras RAID sebagai single drive dan perangkat lunak RAID sebagai multiple drive. BIOS biasanya mengijinkan user menetapkan password yang membatasi akses ke pengaturan BIOS dan dapat mencegah sistem booting tanpa password. BIOS juga menyimpan waktu dan tanggal sistem.

Data Volatile

OS berjalan dalam RAM suatu sistem. Ketika OS sedang berfungsi, isi RAM berubah secara konstan. Di setiap waktu, RAM dapat berisi banyak jenis informasi dan data yang mungkin menarik. Sebagai contoh, RAM sering berisi data yang sering diakses serta terakhir diakses, seperti file data, password hashes, dan perintah terbaru. Seperti filesistem, RAM dapat berisi data sisa dalam slack dan free space, sebagai berikut:

- Slack Space. Slack space memori lebih kurang deterministik daripada file slack space. Sebagai contoh, OS

biasanya mengatur memori dalam unit yang dikenal sebagai halaman atau blok, dan mengalokasikannya bagi aplikasi yang meminta. Kadang-kadang meskipun aplikasi tidak meminta keseluruhan unit, tetapi diberikan juga. Jadi data sisa dapat saja berada dalam unit memori yang dialokasikan ke aplikasi, meskipun ia tidak dapat diakses oleh aplikasi. Untuk efisiensi dan kinerja, beberapa sistem operasi memvariasikan ukuran unit yang mereka alokasikan, yang cenderung menghasilkan space memori slack yang lebih kecil.

- Free Space. Halaman memori dialokasikan dan didealokasikan seperti himpunan file. Ketika mereka tidak dialokasikan, halaman memori sering dikumpulkan dalam kelompok umum halaman yang tersedia, prosesnya dikenal sebagai *garbage collection.* Tidaklah aneh bagi data sisa berada di halaman memori yang dapat digunakan kembali, yang serupa dengan kluster file yang belum dialokasikan.

Beberapa jenis data volatil penting lainnya yang mungkin ada dalam suatu OS adalah :

1. Konfigurasi Jaringan. Walaupun banyak elemen jaringan, seperti driver Kartu Penghubung Jaringan (Network Interface Card atau NIC) dan seting konfigurasi, umumnya disimpan dalam filesistem, secara alami jaringan bersifat dinamis. Sebagai contoh, banyak host yang diberikan alamat IP secara dinamis oleh host lainnya, yang berarti bahwa alamat IP mereka tidak menjadi bagian konfigurasi yang disimpan. Banyak host juga mempunyai beragam interface jaringan, seperti wired, wireless, virtual private network (VPN), dan modem; konfigurasi jaringan yang sekarang menandai interface yang digunakan. User mungkin dapat mengubah konfigurasi interface jaringan, seperti mengubah alamat IP secara manual. Jika memungkinkan, analis perlu menggunakan konfigurasi jaringan yang sekarang, bukan konfigurasi yang tersimpan.
2. Hubungan Jaringan. OS memfasilitasi koneksi antara sistem dan sistem lainnya. Kebanyakan OS dapat menyediakan daftar koneksi jaringan yang keluar dan masuk saat ini, dan beberapa OS dapat menampilkan daftar koneksi terkini.
3. Proses Yang Berjalan. Proses adalah program yang sedang dijalankan suatu komputer. Proses meliputi layanan yang ditawarkan oleh OS dan aplikasi yang dijalankan oleh administrator dan user. Kebanyakan OS menawarkan cara untuk melihat daftar proses yang saat ini sedang berjalan. Daftar ini dapat dipelajari untuk menentukan layanan yang aktif pada sistem. Mengidentifikasi proses yang berjalan bermanfaat untuk mengidentifikasi program yang harus berjalan namun telah ditiadakan atau dihapus.
4. File Terbuka. OS memelihara daftar file terbuka, yang biasanya meliputi user atau proses yang membuka file.
5. Sesi Login. OS umumnya memelihara informasi tentang user yang login saat ini (dan waktu awal serta durasi setiap sesi), login yang gagal dan sukses sebelumnya, penggunaan istimewa, serta impersonasi. Namun demikian, informasi sesi login hanya tersedia bila komputer telah dikonfigurasi untuk mengaudit usaha login. Catatan logon dapat membantu menentukan kebiasaan penggunaan user dan mengkonfirmasi apakah akun user aktif ketika terjadi sebuah peristiwa.
6. Waktu Sistem Operasi. OS memelihara waktu sekarang dan menyimpan waktu daylight saving serta informasi zona waktu. Informasi ini dapat bermanfaat ketika membuat garis waktu peristiwa atau mengkorelasikan peristiwa di antara sistem yang berbeda. Analis harus tahu bahwa waktu yang diberikan oleh sistem operasi mungkin berbeda dengan BIOS karena seting khusus OS, seperti wilayah waktu.

2.4.2. Memperoleh Data OS

Sebagaimana yang diuraikan di dalam Bagian 2.4.1, data OS ada di dalam bagian volatil dan non volatil Data OS non volatil seperti data filesistem dapat diperoleh dengan menggunakan pendekatan yang dibahas di dalam Bagian 2.3 untuk melakukan logical and physical backups. Data OS yang volatil harus dikumpulkan sebelum komputer dimatikan. Bagian 2.4.2.1 dan 2.4.2.2 secara berturut-turut menyediakan rekomendasi untuk memperoleh data OS yang volatil dan non volatil. Bagian 2.4.2.3 mendiskusikan isu teknis yang dapat menghalangi pengumpulan data.

2.4.2.1 Memperoleh Data OS yang Volatil

Data volatil OS yang melibatkan suatu peristiwa hanya dapat diperoleh dari live sistem yang belum direboot atau dimatikan sejak peristiwa terjadi. Setiap tindakan yang dilakukan pada sistem, apakah inisiatif seseorang atau oleh OS sendiri, hampir bisa dipastikan akan mengubah data OS yang bersifat votetil. Oleh karena itu, analis perlu memutuskan secepat mungkin apakah data OS volatil perlu disifnpan. Idealnya, kriteria dalam pembuatan keputusan ini harus telah didokumentasikan sebelumnya sehingga analis dapat membuat keputusan yang terbaik dengan seketika.

Pada sisi lain, mengumpulkan data OS yang bersifat volatil dari suatu komputer yang sedang dijalankan memiliki risiko. Sebagai contoh, selalu ada kemungkinan file pada komputer berubah dan data volatil lain mungkin diubah. Sebagai tambahan, pihak jahat mungkin telah menginstalasi rootkit yang dirancang untuk memberikan informasi palsu, menghapus file, atau melaksanakan tindakan jahat lainnya Dalam memutuskan untuk mengumpulkan data volatil, risiko yang terkait dengan proses

tersebut harus pula dipertimbangkan dengan potensi mengembalikan informasi penting. Jika usaha yang diperlukan untuk mengumpulkan data volatil tidak sesuai, analis dapat memutuskan untuk melakukan shutdown komputer.

Berikut ini adalah beberapa jenis data OS yang bersifat volatil dan menjelaskan bagaimana tools dapat digunakan dalam pengumpulan masing-masing tipe data:

1. Isi memori. Terdapat kebutuhan untuk dapat menyalin isi RAM ke file data dan membantu menganalisa data berikutnya. Pada kebanyakan sistem, hal ini akan menyebabkan perubahan pada RAM ketika menjalankan fungsi yang mencoba membuat salinan RAM. Oleh karena itu, tujuannya adalah melakukan dengan langkah-langkah kecil yang mungkin dapat memperkecil gangguan pada RAM
2. Konfigurasi jaringan. Umumnya sistem operasi memiliki fasilitas untuk menampilkan konfigurasi jaringan yang ada pada saat itu, seperti **ifconfig** pada sistem Unix dan **ipconfig** pada sistem Windows. Informasi tersebut dapat disediakan melalui fasilitas konfigurasi jaringan yang meliputi hostname, network interface fisik dan logika dan informasi konfigurasi untuk masing - masing interface (misalnya, alamat IP, alamat MAC, dan status yang sekarang).
3. Koneksi jaringan. Sistem operasi umumnya menyediakan metode untuk menampilkan daftar koneksi jaringan yang ada. Sistem berbasis Windows dan Unix umumnya memiliki program **netstat** untuk menampilkan daftar koneksi jaringan yang berisi alamat IP dan port sumber dan tujuan, dan juga daftar port yang terbuka di masing-masing interface.

4. Menjalankan proses. Semua sistem berbasis Unix memiliki perintah **ps** untuk menampilkan proses yang sedang berjalan. Walaupun Windows menyediakan utilitas daftar proses berbasia GUI, Task Manager, umumnya lebih disukai memiliki daftar berbasis teks.

5. File terbuka. Semua sistem berbasis Unix menyediakan perintah **lsof** untuk menampilkan daftarfile yang terbuka.

6 Sesi login. Beberapa sistem operasi memiliki perintah built-in untuk menampilkan user yang sedang logon saat ini, seperti perintah w di sistem UNIX. 7. Waktu sistem operasi. Terdapat beberapa utilitas untuk mengambil waktu sistem saat ini, informasi zona waktu, serta seting waktu daylight saving. Pada sistem UNIX, perintah **date** dapat digunakan untuk mengambil informasi ini. Pada sistem Windows, perintah **date, time, nlsinfo** dapat digunakan secara bersama untuk memperoleh informasi ini.

Tipe data volatil yang harus dikumpulkan tergantung pada kebutuhan. Sebagai contoh, jika dicurigai terjadi penyusupan jaringan, mungkin perlu mengumpulkan informasi konfigurasi jaringan, koneksi jaringan, sesi login, proses yang sedang berjalan untuk menentukan seseorang telah memperoleh akses ke sistem. Jika investigasi mengkuatirkan pencurian identitas, maka isi RAM, daftar proses yang berjalan, daftar file terbuka, informasi konfigurasi jaringan, dan koneksi jaringan mungkin dapat menampilkan nomor kartu kredit dan keamanan sosial, program yang digunakan untuk memperoleh atau mengenkripsi data, hash password, dan metode yang mungkin telah digunakan untuk memperoleh informasi melalui jaringan.

2.4.2.2. Memperoleh Data OS yang Non-Volatil

Setelah mendapatkan data OS volatil, analis perlu juga mendapatkan data OS yang non volatil. Untuk melakukannya, pertama analis perlu memutuskan apakah sistem harus dimatikan, Mematikan sistem tidak hanya mempengaruhi kemampuan melakukan physical backups dan banyak logical backups, tetapi dapat juga mengubah data OS yang dipelihara. Kebanyakan sistem dapat dimatikan melalui dua metode, seperti berikut:

- Melakukan pematian sistem operasi secara baik. Hampir setiap OS menawarkan pilihan cara mematikan OS. Hal ini menyebabkan OS melakukan aktifitas cleanup, seperti menutup semua file yang terbuka, menghapus file sementara dan kemungkinan membersihkan file swap, sebelum mematikan sistem. Mematikan sistem secara baik dapat juga memicu penghapusan materi berbahaya; contohnya, memori resident rootkits dapat hilang dan trojan horses mungkin menghilangkan bukti aktivitas berbahaya mereka OS secara umum dimatikan dari account administrator ataupun user yang sedang menggunakan sistem (jika user yang sedang menggunakan sistem memiliki cukup ijin).
- Meniadakan sumber daya listrik sistem. Pemutusan hubungan tenaga listrik dari belakang komputer (dan memindahkan baterai pada laptop atau device portable lain) dapat memelihara file swap, file data sementara, dan informasi lainnya yang mungkin diubah atau dihapus selama sistem dimatikan secara baik. Sayangnya, kehilangan daya secara tiba-tiba dapat menyebabkan beberapa OS rusak datanya. Pada PDA dan telepon selular, meniadakan tenaga baterai dapat menyebabkan kehilangan data.

Analis harus memahami karakteristik tiap sistem operasi dan memilih suatu metode shutdown berdasarkan perilaku OS dan jenis data yang perlu dipelihara. Sebagai contoh, sistem Windows 95/98 dan DOS pada umumnya tidak merusakkan data ketika

tenaga listriknya ditiadakan tiba - tiba, jadi meniadakan tenaga listrik akan melindungi data. Sistem operasi lainnya mungkin dapat merusak data, seperti file yang terbuka atau file yang sedang diakses ketika itu, jika hilang tenaga listrik, maka mematikan komputer secara baik merupakan tindakan yang dianjurkan kecuali kalau file swap dan file sementara merupakan bagian yang menarik atau jika sistem mungkin berisi rootkit, trojan horse atau program berbahaya lainnya, yang dipicu oleh mematikan sistem secara benar. Setelah melakukan *"shutdown",* analis kemudian perlu memperoleh data filesistem dari media penyimpanan sistem menggunakan metode yang dibahas dalambagian2.3.

Daftar berikut merupakan jenis lain data sistem operasi yang bersifat non volatil dan penjelasan bagaimana tool dapat berguna dalam mendapatkan masing - masing jenis dari filesistem:

- User dan Grup user. Sistem operasi memelihara daftar user dan grup yang memiliki akses ke suatu sistem.
  Pada sistem Unix, user dan user disimpan dalam /etc/passwd dan /etc/groups. Sebagai tambahan, perintah groups dan users dapat digunakan untuk mengidentifikasi user yang telah dimasukkan ke dalam sistem dan grup mereka menjadi anggota. Pada sistem Windows, perintah net user dan net group dapat digunakan untuk menampilkan user dan grup.

- Password. Kebanyakan sistem operasi menyimpan hash password di disk. Pada sistem Windows, utilitas pihak ketiga dapat digunakan untuk menampilkan hash password dari database Security Account Manager (SAM). Pada sistem Unix, potongan kata sandi biasanya disimpan dalam file/etc/passwd atau /etc/shadow.
- Share Jaringan. Suatu sistem dapat membuat sumber daya lokal dipakai bersama dalam jaringan. Pada sistem Windows, utilitas SrvCheck dapat digunakan untuk menampilkan share jaringan.
- Log. Log yang tidak disimpan dalam bentuk file teks mengharuskan penggunaan utilitas ekstraksi log. Sebagai contoh, utilitas khusus dapat mengambil informasi mengenai usaha logon yang berhasil dan gagal terkini pada sistem Windows. Kebanyakan entri log pada sistem Unix disimpan dalam file teks oleh syslog atau dalam direktori /var/log/.

2.4.3. Isu Teknis Dalam Memperoleh Data

Terdapat beberapa isu teknis yang berpotensi menghalangi perolehan data OS yaitu:

- Akses OS. Mendapatkan data volatil mungkin sulit karena analis tidak dapat segera memperoleh akses ke sistem operasi. Sebagai contoh, seorang user mungkin menjalankan screensaver yang terlindung password atau sistemnya dikunci. Dalam kasus ini analis perlu mengatasi proteksi tersebut atau mencari cara lain memperoleh akses ke data volatil.
- Modifikasi log. User mungkin mencoba mengurangi manfaat log dengan cara menon-aktifkan fitur log, modifikasi seting log, sehingga mengurangi tempat penyimpanan log atau menulis banyak kejadian ke log. Salah satu cara mengurangi dampak perubahan log adalah mengkonfigurasi sistem untuk mengarsip entri log ke server terpusat.
- Hard Drives dengan Flash Memory. Adakalanya, seorang analis mendapatkan hard drive yang juga berisi flash memory. Flash memory ini dapat berisi password yang dibutuhkan untuk mengakses drive, bahkan ketika drive telah dipindahkan dari komputer tersebut. Biasanya, analis perlu menemukan, menduga, atau memecahkan password untuk mendapatkan akses ke drive tersebut.
- Key remapping. Pada beberapa komputer, kunci perorangan atau kombinasi penekanan tombol dapat dipetakan kembali untuk melakukan fungsi berbeda dari fungsi awal. Sebagai contoh, user dapat memetakan kunci Ctrl, Alt dan Del untuk menghapus hard drive alih-alih tindakan yang dfharapkan.Jalan terbaik untuk menghindarinya.

7. Dokumentasi Tool

Bab ini akan berisikan dokumentasi untuk beberapa buah tool yang disertakan dalam SLAX4. Untuk manual lengkapnya dapat dilihat dalam SLAX4 dengan memberikan perintah "man <nama_tool>".

7.1.antiword

Antiword merupakan sebuah aplikasi yang digunakan untuk menampilkan teks dan gambar dokumen Microsoft Word. Antiword hanya mendukung dokumen yang dibuat oleh MS Word versi 2 dan versi 6 atau yang lebih baru. Perintah penggunaan antiword adalah:

antiword [*options] wordfiles*

Opsi yang tersedia adalah sebagai berikut:

-a *papersize*
Hasil dalam format Adobe PDF. Dapat dicetak dalam kertas berukuran : 10x14, a3, a4, a5, b4, b5, executive, folio, legal, letter, note, kuarto, statement atau tabloid.

-f
Hasil dalam bentuk teks berformat. Artinya teks bold akan dicetak *tebal*, cetak miring dengan /miring/dan bergarisbawah dengan _garisbawah_

-l *image level*
Menentukan bagaimana gambar akan ditampilkan.

0:

Menggunakan ekstensi non-standar dari Ghostscript. Hasil ini mungkin tidak dapat dicetak pada sembarang printer PostScript, namun ia berguna bila tidak dibutuhkan hard copy, la juga bermanfaat ketika Ghostscript digunakan sebagai filter untuk mencetak file PostScript ke printer non-PostScript.

1: tampilkan tiada gambar.

2: Kompatibel dengan PostScript level 2. (default)

Kompatibel dengan PostScript level 3. (EXPERIMENTAL, image Portable Network Graphics (PNG) tidak dapat dicetak dengan tepat)

-w *width*
Dalam mode teks ini merupakan lebar baris untuk karakter. Nilai nol menempatkan seluruh paragraf dalam satu baris, bermanfaat ketika teks akan

digunakan sebagai input bagi pengola kata lainnya. Nilai ini diabaikan dalam mode PostScript

-x *documenttype definition*
Hasil dalam format XML. Saat ini hanya mendukung defmisi tipe dokumen db
(untuk DocBook).

7.2. Autopsy

The Autopsy Forensic Browser merupakan antarmuka grafis untuk tool analisis investigasi diginal perintah baris The Sleuth Kit. Bersama, mereka dapat menganalisis disk dan filesistem Windows dan UNIX (NTFS, FAT, UFS1/2, Ext2/3).

The Sleuth Kit dan Autopsy bersifat Open Source dan berjalan pada platform UNIX. Oleh karena Autopsy berbasis HTML, anda dapat koneksi ke server Autopsy dari sembarang platform dengan menggunakan browser HTML.

ModeAnalisis

- Analisis offline (dead analysis) terjadi ketika digunakan sistem analisis khusus untuk memeriksa data dari sistem tersangka. Autopsy dan The Sleuth Kit dijalankan dalam lingkunganterpercaya, biasanya dalam sebuah laboratorium.
- Analisis hidup (live analysis) terjadi ketika sistem tersangka dianalis ketika sedang berjalan. Dalam hal ini Autopsy dan The Sleuth Kit dijalankan dari sebuah CD (SLAX4) dalam lingkungan yang tidak terpercaya Hal ini sering dilakukan selama proses *incident response* ketika insiden sedang dikonfirmasi. Setelah ia dikonfirmasi, sistem dapat diambil dan dilakukan analisis offline.

Teknik Pencarian Bukti

- File Listing: Menganalisis file dan direktori, termasuk nama file yang terhapus dan file dengan nama berbasis Unicode.
- File Content: Isi file dapat dilihat dalam format raw, heksadesimal, atau string ASCII. Ketika menginterpretasi data,

Autopsy membersihkannya untuk mencegah kerusakan ke sistem analisis lokal. Autopsy tidak menggunakan bahasa skrip

client-side.

- Hash Databases: Mencari file tidak dikenal dalam database hash untuk mengetahui apakah ia benar atau tidak. Autopsy menggunakan NIST National Software Reference Library (NSRL) dan database yang dibuat user untuk file benar atau tidak yang dikenal..
- File Type Sorting: Mengurutkan file berdasarkan tanda internal mereka untuk mengidentifikasi file dengan tipe yang dikenal. Autopsy dapat juga mengekstraksi hanya image grafis (termasuk thumbnail). Ekstensi file akan dibandingkan pula dengan tipe file untuk mengidentifikasi file yang ekstensinya diubah untuk menyembunyikannya • Timeline of File Activity: Dalam beberapa kasus. memiliki garis waktu aktivitas file dapat membantu mengidentifikasi area filesistem yang mungkin berisi bukti, Autopsy dapat membuat garis waktu yang berisi entri untuk waktu Modified, Access, dan Change (MAC) bagi file yang teralokasi dan tidak.
- Keyword Search: Pencarian kata kunci pada image filesistem dapat dilakukan dengan menggunakan string ASCII dan ekspresi reguler grep. Pencarian dapat dilakukan untuk seluruh image filesistem atau hanya untuk ruang yang belum dialokasi. Sebuah file indeks dapat dibuat untuk mempercepat pencarian. String yang sering dicari dapat dikonfigurasi ke dalam Autopsy untuk pencarian otomatis.
- Meta Data Analysis: Struktur Meta Data berisikan detil file dan direktori. Autopsy memungkinkan anda melihat detil sembarang struktur meta data dalam filesistem. Hal ini berguna untuk mengembalikan materi yang terhapus. Autopsy akan mencari direktori untuk mengidentifikasi path lengkap file yang telah mengalokasikan struktur.
- Data Unit Analysis: Data Unit adalah tempat isi file disimpan. Autopsy memungkinkan anda melihat isi sembarang data unit dalam berbagai format termasuk ASCII, hexdump, dan string. Tipe file juga diberikan dan Autopsy akan mencari struktur meta data untuk mengidentifikasi yang telah mengalokasikan data unit.
- Image Details: Detil filesistem dapat dilihat, termasuk layout disk dan waktu aktivitas. Mode ini memberikan informasi yang berguna selama proses pengembalian data.

Manajemen Kasus

- Case Management: Penyelidikan diorganisir berdasar kasus, yang dapat berisikan satu atau lebih host. Setiap host dikonfigurasi untuk memiliki seting zona waktu dan penyesuaian waktu sendiri, sehingga waktu yang ditampilkan sama seperti yang dilihat oleh pengguna asli. Setiap host dapat berisikan satu atau lebih image filesistem untuk dianalisis.
- Event Sequencer: Kej'adian berbasis waktu dapat ditambahkan dari aktivitas file atau log IDS dan firewall. Autopsy mengurutkan kejadian sehingga urutan insiden kejadian dapat ditentukan dengan lebih mudah.

Perhatikan : Catatan dapat disimpan per-host atau per-penyelidik. Hal ini memungkinkan anda membuat catatan singkat tentang file dan struktur. Lokasi asli dapat dengan mudah dipanggil dengan penekanan tombol ketika catatan dibaca kembali. Semua catatan disimpan dalam file ASCII.

- Image Integrity: Sangatlah penting untuk memastikan bahwa file tidak termodifikasi saat analisi. Autopsy, secara default, akan membuat nilai MD5 untuk seluruh file yang diimpor atau dibuat. Integritas sembarang file yang digunakan Autopsy dapat divalidasi setiap saat.
- Reports: Autopsy dapat membuat laporan ASCII untuk file dan struktur filesistem lainnya. Hal ini memungkinkan anda membuat informasi yang konsisten dan cepat selama proses penyelidikan.
- Logging: Log audit dibuat untuk setiap tingkatan kasus, host, dan penyelidik sehingga setiap tindakan dapat ditelusuri dengan mudah. Perintah lengkap Sleuth Kit yang dieksekusi juga dicatat.
- Open Design: Kode Autopsy bersifat terbuka dan seluruh file yang digunakan adalah dalam format raw. Seluruh file konfigurasi dalam teks ASCII dan kasus diorganisir dalam direktori. Hal ini memudahkan proses ekspor dan pengarsipan data. Hal ini juga tidak membatasi anda menggunakan tool lain yang mungkin lebih tepat digunakan untuk menyelesaikan masalah

Client Server Model: Autopsy adalah berbasis HTML dan karenanya anda tidak perlu berada dalam sistem yang sama dengan image filesistem. Hal ini memungkinkan banyak penyelidik menggunakan server yang sama dan terhubung dari sistem pribadi mereka.

7.3. *binhash*

binhash merupakan sebuah program sederhana untuk melakukan hashing terhadap berbagai bagian file ELF dan PE untuk perbandingan. Saat ini ia melakukan hash terhadap segmen header dari bagian header segmen obyek ELF dan bagian segmen header obyekPE.

Hash yang tersedia adalah hash yang disertakan bersama librari OpenSSL (md2/md4/md5/sha1). Ia bermanfaat ketika membandingkan dua varian malware. Bagian mana yang cocokdan bagian manayang tidak cocok.

```
$ binhash

BinHash v.03
http://www.structsoftware.net
Chris Rohlf - 2007

        -f      [ Filename ] (PE and ELF Only)
        -h      [ Hash (MD2 |MD4 IMD51SHA1) ]
```

Mari kita gunakan binhash untuk melakukan hashing terhadap dirinya sendiri

```
$ binhash -f binhash -h SHA1

File: [binhash] 085c429dc040b4e653724c72f19634cc28da80e7

ELF Header:    ed2e2c60a297912dee9cccb0abb!3b3476ae50ce

Phdr [0] (Address: 08048034) (Offset: 00000034) (Size: 00000120) PHDR Phdr:
7bea746b2219ceffbf0f6321134b62c27c8f7b77 Phdr Data: bldfac59d482c73a847bfee05010c3af903befcl

Phdr [1] (Address: 08048154) (Offset: 00000154) (Size: 00000013) INTERP Phdr:
a42957aae066f08b92bcffa14d5b0bd52e6e20b0 Phdr Data: Iabl77049f14482a43fc24babe5826e81efb51e8
 Phdr [3] (Address: 0804aefc) (Offset: OOOOlefc) (Size: 0000017c) LOAD Phdr:
 ac76eb202cc2d3e02e2c249eda05203b0798049a Phdr Data: 7745035e8ce7257ff63f96d8cc9f7a84f74c9b67

 Phdr [4] (Address: 0804aflO) (Offset: OOOOlflO) (Size: OOOOOOeO) DYNAMIC Phdr:
 ed0b76ccf69818fc38df3455c2b8d9f5612aed9f Phdr Data: 3b3de45b27dc498e5517148aac212f80b0ae3e56

 Phdr [5] (Address: 08048168) (Offset: 00000168) (Size: 00000020) NOTE Phdr:
 9c8d7aaebafdl3385db72be01e7edebbc907a206 Phdr Data: 80614b8f262ad92ealdfd061a64328bf45f79d8b

 Phdr [6] (Address: 08048188) (Offset: 00000188) (Size: 00000018) NOTE Phdr:
 a500faefc95c6720e5e892466180418924763dc5 Phdr Data: 97cl5cd690abc9c2b0cl6fb5bada3a6f9dc9419c

 Phdr [8] (Address: 0804aefc) (Offset: OOOOlefc) (Size: 00000104) GNU RELRO Phdr:
 0bf524elel4974850c319936d5f9ead5a88cl0d9 Phdr Data: 1915664907888691e9b8095322982ca21385ca5e

 Shdr [1] (Addr: 08048154) (Offset: 00000154) (Size: 00000013) Shdr:
 7b42024d09e000b6eb9759d0a20cc787e2c8d654 Shdr Data: Iabl77049f14482a43fc24babe5826e81efb51e8



 Shdr [37] (Addr: 00000000) (Offset: 00005184) (Size: 00000650) Shdr:
 26d88f37305562ab32d2a652b6ea8dldffc756eb Shdr Data: e48f5677a7fcl9dca87bec9fcc!4clfeac4959b6

 Shdr [38] (Addr: 00000000) (Offset: 000057d4) (Size: 0000036b) Shdr:
 2fId82862a99e5e5e34a373ff!86ele204edla30 Shdr Data: 140e85dc5b63aaf22eelbc0ec0e2daccd59aae63

 Written by Chris Rohlf 2007 http://www.structsoftware.net
```

7.4. ClamAVAnti Virus Scanner

Clam AntiVirus merupakan sebuah toolkit antivirus untuk UNIX, ia mulanya dirancang untuk memeriksa email pada gateway email. Namun saat ini ia telah menyertakan kemampuan untuk memeriksa virus pada workstation. Dalam SLAX4, hanya disertakan tool clamscan dan freshclam saja.

7.5. Clamscan

Clamscan digunakan untuk memeriksa virus pada file dan direktori.
Sedangkan opsi-opsinya antara lain adalah:

-D FILE/DIR, ~database=FILE/DIR
Memuat database virus dari FILE atau memuat semua file database virus dari DIR.

-|FILE,--log=FILE
Menyimpan hasil pemeriksaan ke FILE.

-r, —recursive
Memeriksa direktori secara rekursif. **~exclude=PATT, --exclude-dir=PATT**
Tidak memeriksa nama file/direktori yang berisikan PATT. **~include=PATT, --include-dir=PATT**
Hanya memeriksa nama file/direktori yang berisikan PATT.

-i,-- infected
Hanya mencetak file yang terinfeksi.

--remove
Hapus file yang terinfeksi. Hati-hati.

-move=DIRECTORY
Pindahkan file yang terinfeksi ke DIRECTORY. Direktori tersebut harus dapat ditulisi oleh user'clamav'atau user unprivileged yang menjalankan clamscan.

~copy=DIRECTORY
Salinkan file yang terinfeksi ke DIRECTORY. Direktori tersebut harus dapat ditulisi oleh user'clamav' atau user unprivileged yang menjalankan clamscan.

-no-algorithmic
Dalam beberapa kasus (misalnya malware yang kompleks, eksploitasi dalam file grafis, dan lainnya), ClamAV menggunakan algoritma khusus untuk menyediakan deteksi yang akurat. Opsi ini meniadakan deteksi algoritma tersebut.

--no-pe

PE singkatan dari Portable Executable - ia merupakan sebuah format file yang digunakan dalam sistem operasi Windows 32-bit. Bakunya, ClamAV akan melakukan analisis rinci atas file eksekutabel dan berusaha membuka pengompres eksekutabel populer seperti UPX, Petite, dan FSG. Opsi ini meniadakan dukungan PE dan harus digunakan dengan hati-hati!

-no-elf

Executable and Linking Format merupakan format standar file eksekutabel UN*X. Opsi ini meniadakan dukungan ELF.

--no-ole2

Meniadakan dukungan untuk dokumen dan file Microsoft Office .msi.

-no-pdf ,

Tidak memeriksa dalam file PDF.

--no-html

Meniadakan dukungan untuk pemeriksaan dan normafisasi HTML.

--no-archive

Meniadakan dukungan arsip yang ada dalam libclamav.

-detect-broken

Tandai eksekutabel yang rusak sebagai virus.

--max-files=#n

Ekstraksi #n file pertama dari setiap arsip. Opsi ini melindungi sistem anda dari serangan DoS attacks (default: 500)

-max-space=#n

Ekstraksi #n kilobyte pertama dari setiap arsip. Anda dapat memberikan nilainya dalam format megabyte xM atau xm, dengan x adalah sebuah angka. Opsi ini melindungi sistem anda terhadap serangan (default: 10MB)

--max-recursion=#n

Menset batasan tingkatan rekursi arsip. Opsi ini melindungi sistem anda terhadap serangan DoS (default: 8).

-max-dir-recursion=#n

Kedalaman direktori maksimum yang diperiksa (default: 15).

Di bawah ini adalah beberapa contoh penggunaan clamscan:

- **Memeriksa sebuah file:**

  clamscan : i1e

- **Memeriksa direktori kerja saat ini:**

  clamscan

- **Memeriksa seluruh file (dan subdirektori) dalam /home/tedi:**

  clamscan - : /home/tedi

- **Memuatkan database dari file dan membatasi penggunaan disk hingga 50 MB:**

  clamscan - 1 /tmp/newclamdb —rax-space=50m -r /tmp

7.6. *Freshclam*

freshclam merupakan sebuah tool untuk memperbarui database virus bagi ClamAV.

Perintahnya adalah:

freshclam [option-;!

Freshclam membaca konfigurasi dari file freshdam.conf. Seting yang ada di dalam file tersebut dapat dioverwrite dengan opsi perintah bans di antaranya adalah:

-IFILE,-log=FILE

Menulis laporan download ke FILE.

.-datadir=DIRECTORY

Instalasi database baru dalam DIRECTORY. Direktori harus dapat ditulisi oleh user 'clamav' atau user unprivileged yang menjalankan freshclam.

-no-dns

Opsi ini memaksa metode verifikasi lawas non-DNS (tanpa delay TTL).

-c#n,-checks=#n

Memeriksa #n kali per hari untuk database baru. #n harus antara 1 dan 50.

Berikut ini adalah contoh penggunaan freshclam:

- Download database ke direktori baku: freshclam
- Download database ke direktori saat ini: freshclam —datadir=.

7.7. *sigtool*

sigtcol merupakan tool untuk manajemen signature dan database ClamAV. sigtool dapat digunakan untuk rnenghasilkan checksum MD5, konversi data ke dalam format heksadesimal, menampilkan daftar signature virus dan build/unpack/test/verify database CVD dan skrip update.

Penggunaannya adalah sebagai berikut:

sigtool [options] Opsi yang didukung sigtool antara lain adalah:

--hex-dump
Baca data dari stdin dan tulis string heksa ke stdout.

-md5 [FILES]
Hasilkan checksum MD5 dari stdin atau MD5 sigs untuk FILES.

~vba=FILE
Ekstraksi makro VBA/Word6 dari dokumen MS Office yang diberikan.

~vba-hex=FILE
Ekstraksi makro Word6 dari dokumen MS Office yang diberikan dan tampilkan nilai heksnya.

-i,-info
Cetak informasi CVD dan verifikasi MD5 dan tanda tangan digital

-I, -list-sigs
Tampilkan nama signature.

Berikut ini adalah contoh penggunaan sigtool:
Membuat string heksadesimal dari testfile dan menyimpannya ke dalam file testfile.hex:

```
cat testfile I sigtool—hex-dump >testfile.hex
```

7.8. ChaosReader

ChaosReader merupakan sebuah tool freeware untuk melacak sesi TCP/UDP/... dan mengambil data aplikasi dari log tcpdump. la akan mengambil sesi telnet, file FTP, transfer HTTP (HTML, GIF, JPEG,...), email SMTP, dan sebagainya, dari data yang ditangkap oleh log lalu lintas jaringan. Sebuah file index html akan tercipta yang berisikan link ke seluruh detil sesi, termasuk program replay realtime untuk sesi telnet, rlogin, IRC, X11 atau VNC; dan membuat laporan seperti laporan image dan laporan isi HTTP GET/POST.

Chaosreader dapat juga berjalan dalam mode *standalone,* yaitu ia memanggil tcpdump (jika tersedia) untuk membuat file log dan memproses mereka.

Contoh Penggunaan:

tcpdump -s9000 atau :	
ethereal (save atau :	-w out; chaosreader out; netscape index.html as "out!"); chaosreader out;
chaosreader -s	
	netscape index.html 5; netscape index.html

7.9. chkrootkit

chkrootkit merupakan sebuah tool untuk memeriksa tanda-tanda adanya rootkit secara lokal. la akan memeriksa utilitas utama apakah terinfeksi, dan saat ini memeriksa sekitar 60 rootkit dan variasinya.

Penggunaan sederhana chkroot adalah sebagai berikut:

```
# ./chkrootkit
```

Perintah di atas akan melakukan seluruh tes. Anda dapat juga memberikan tes tertentu yangdiinginkan:

```
# ./chkrootkit [options] [testname ...]
```

Opsi:
Dengan nama tes adalah sebagai berikut:

aliens	chkutmp	egrep	init	P°P[2]	tcpd
asp bindshel	amd	env	killall	pop3	tcpdump
	basenam	find	ldsopreloa ps	top	
1	e	fingerd	d login	pstree	telnetd
1km	biff	gpm	ls	rpcinfo	timed
rexedcs	chfn	grep	lsof	rlogind	traceroute
sniffer	chsh	hdparm	mail	rshd	vdir

w55808 cron su mingetty slogin
wted date ifconfig netstat sendmail
scalper du inetd named shd
slapper dirname inetdconf passwd syslogd
z2 echo identd pidof tar

Sebagai contoh, perintah berikut akan memeriksa biner ls dan ps yang bertrojan dan juga memeriksa apakah interface jaringan berada dalam mode promiscuous:

./chkrootkit ps Is sniffer

chkrootkit menggunakan perintah-perintah berikut untuk melakukan pemeriksaan : : awk, cut, egrep, find, head, id, Is, netstat, ps, strings, sed, uname. Untuk memberikan path alternatif sehingga ia tidak menggunakan biner sistem bila dicurigai, gunakan opsi

'-P'-

Untuk menggunakan biner dalam /cdrom/bin, perintahnya adalah:

4 ./chkrootkit -p /cdrom/bin Untuk menambahkan lebih banyak path, gunakan ':'

t ./chkrootkit -p /cdrom/bin:/floppy/mybin

Pesan-pesan berikut akan ditampiikan oleh chkrootkit (kecuali diberikan opsi -x dan -q) selamapengujiannya:

- "INFECTED": tes mengidentifikasikan perintah yang mungkin telah dimodifikasi oleh rootkit;
- "not infected": tes tidak menemukan tanda rootkit yang dikenal.
- "not tested": tes tidak dilakukan, hal ini dapat terjadi dalam situasi berikut:
  a) tesnya khusus untuk sistem operasi;
  b)testergantung pada program eksternal yang tidak tersedia;
  c) diberikan opsi perintah baris tertentu. (misalnya -r).
- "notfound": perintah yang akan dites tidak tersedia;
- "Vulnerable but disabled": perintah terinfeksi namun tidak digunakan. (tidak berjalan atau dikomentari dalam inetd.conf)
- Berikut ini adalah hasil menjalankan chkrootkit:

```
*chkrootkit
ROOTDIR is V
                    not found basename' ...  Checking 'amd
not infecte-1       not   infected   not     Checking
chfn'               infected not infected    Checking
chsh'1              . . not infected not     Checking
cron'1              infected                 Checking
date'                                        Checking
du'..                                        Checking

Checking 'dirname' . . . not infected Checking 'echo1 . . . not
infected Checking 'egrep' . . . not infected

Checking 'asp1 . . . not infected
Checking "bindshell' . . . not infected
Checking '1km'. . . chkproc: nothing detected
Checking 'rexedcs' . . . not found
Checking 'sniffer' . . . ethO: not promise and no PF PACKET sockets
Checking 'w55808' . . . not infected
Checking 'wted' . . . chkwtmp: nothing deleted
Checking 'scalper' . . . not infected
Checking 'slapper' . . . not infected
Checking ~z2'... chklastlog: nothing deleted
Checking 'chkutmp'. . . chkutmp: nothing deleted
```

7.10.dcfldd

Tool ini mulanya dikembangkan di Department of Defense Computer Forensics Lab (DCFL). Meskipun saat ini Nick Harbour tidak lagi berafiliasi dengan DCFL, ia tetap memelihara tool ini.

Dcfldd merupakan versi perbaikan GNU dd dengan fitur yang bermanfaat untuk forensik dan keamanan. dcfldd memiliki fitur-fitur tambahan berikut bila dibandingkan dd:

- Hashing on-the-fly - dcfldd dapat menghasilkan hash data input saat mentransfernya, untuk memastikan integritasdata. « Menginformasikan status - dcfldd dapat memberitahukan user mengenai perkembangan data yang telah ditransfer dan berapa lama lagi proses berlangsung.
- Menghapus disk secara fleksibel - dcfldd dapat digunakan untuk menghapus disk secara cepat dengan pola tertentu bila diinginkan.
- Verifikasi Image/hapus - dcfldd dapat memverifikasi bahwa drive sasaran cocok escara bit demi bit dari file input atau pola yang ditentukan.
- Multi output - dcfldd dapat mengeluarkan ke berbagai file atau disk pada saat bersamaan.

- Membagi output - dcfldd dapat membagi output ke beragam file dengan lebih mudah dibandingkan dengan perintah split.
- Mem-piped output dan log - dcfldd dapat mengirim seluruh log data dan output ke perintah serta file dengan alami.

7.11. ddrescue

GNU ddrescue merupakan sebuah tool penyelamat data, la menyalinkan data dari satu file atau device blok (hard disc, cdrom, dsb.) ke yang lain, berusaha keras menyelamatkan data dalam hal kegagalan pembacaan.
Ddrescue tidak memotong file output bila tidak diminta. Sehingga setiap kali anda menjalankannya kefile output yang sama, ia berusaha mengisi kekosongan.

Operasi dasar ddrescue otomatis penuh. Yaitu, anda tidak perlu menunggu kesalahan, menghentikan program, membaca log, menjalankannya dalam mode reverse, dsb, Jika anda menggunakan fitur logfile ddrescue, data diselamatkan dengan sangat efisien (hanya blok yang dibutuhkan yang dibaca),

Anda dapat pula menginterupsi penyelamatan di setiap waktu dan melanjutkannya di waktukemudian.

Penggabungan backup secara otomatis : bila anda memiliki dua atau lebih salinan file, cdrom, dsb, yang rusak, dan menjalankan ddrescue ke mereka semua, satu per satu, dengan file output yang sama, anda mungkin akan memperoleh file lengkap dan bebas kesalahan. Hal ini disebabkan karena probabilitas memiliki kerusakan di area yang sama pada file input berbeda adalah sangat rendah. Dengan menggunakan logfile, hanya blok yang dibutuhkan yang dibaca dari salinan kedua dan seterusnya.

logfile secara periodik disimpan ke disk. Sehingga bila terjadi crash anda dapat meneruskan penyelamatan dengan sedikit penyalinan ulang.

logfile yang sama juga dapat digunakan untuk beragam perintah yang menyalin area berbeda dari file, dan untuk usaha penyelamatan ganda atas subset yang berbeda.

Ddrescue menyelaraskan buffer i/O-nya ke ukuran sektor sehingga ia dapat digunakan untuk membaca dari device raw. Demi alasan efisiensi, ia juga menyelaraskannya dengan ukuran page memori jika ukuran page merupakan kelipatan ukuran sektor.

7.12. dd_rescue

Seperti dd, dd_rescue menyalinkan data dari satu file atau device blok ke yang lainnya Namun dd_rescue berbeda dengan dd dalam hal:
- dd_rescue tidak menyediakan fitur konversi karakter
- Sintaks perintahnya berbeda. Jalankan dd_rescue -h.
- dd_rescue tidak batal bila ada kesalahan pada file input, kecuali anda menentukan jumlah kesalahan maksimum, lalu dd_rescue akan membatalkan proses ketika angkainiterlampaui.
- dd_rescue tidak memotong file output, kecuali diminta.
- Anda dapat memberitahu dd_rescue untuk memulai dari akhir file dan bergerak mundur.
- la menggunakan ukuran 2 blok, ukuran blok besar (soft) dan ukuran blok kecil (hard). Bila ada kesalahan, ukuran kembali ke yang kecil dan akan dipromosikan kembali bila tidak ada kesalahan.

Ketiga fitur terakhir membuatnya cocok untuk menyelamatkan data dari media yang rusak, misalnya harddisk dengan beberapa sektor rusak. Mengapa ?

- Bayangkan, salah satu partisi anda *crash,* dan karena ada beberapa kesalahan, anda tidak ingin menulis ke harddisk ini lagi. Anda hanya ingin mengambil seluruh data yang ada dan mempensiunkannya. Namun anda tidak dapat mengakses file-file tersebut, karena filesistemnya rusak.
- Sekarang, anda ingin menyalinkan seluruh partisi ke sebuah file Anda membakarnya ke CDROM agar tidak hilang lagi. Anda dapat mensetup device loop, dan memperbaikinya (fsck) dan berharap dapat memountnya.
- Menyalinkan partisi ini dengan tool normal Unix seperti cat atau dd akan gagal, karena tool tersebut akan berhenti ketika menemui kesalahan. Sebaliknya dd_rescue akan berusaha membaca dan bila gagal, ia akan melanjutkan ke sektor berikutnya. File hasilnya tentu saja akan memiliki celah. Anda dapat menulis sebuah file log, untuk melihat di mana lokasi kesalahan-kesalahan tersebut.
- Rate data akan jatuh ke tingkat yang sangat rendah, ketika menemui kesalahan. Jika anda menginterupsi proses penyalinan, anda tidak kehilangan apapun. Anda dapat meneruskan di saat lain. File output akan diisi dengan data lanjutan dan tidak terpotong seperti tool Unix lainnya.
- Jika anda memiliki satu celah sektor rusak dalam partisi, merupakan ide yang baik untuk mendekati celah ini dari dua sisi. Penyalinan dari arah belakang dapat berguna.
- Ukuran dua blok merupakan optimisasi kinerja. Ukuran blok besar menghasilkan kinerja yang tinggi, namun bila ada kesalahan, anda ingin mencoba menyelamatkan setiap sektor. Jadi hardbs terbaik diset ke ukuran sektor hardware (seringkali 512 bytes) dan softbs ke nilai besar misalnya 64k default.

7.13. foremost

Foremost merupakan sebuah tool yang dapat digunakan untuk me-recover file berdasarkan header, footer, atau struktur data file tersebut. la mulanya dikembangkan oleh Jesse Kornblum dan Kris Kendall dari the United States Air Force Office of Special Investigations and The Center for Information Systems Security Studies and Research. Saat ini foremost dipelihara oleh Nick Mikus seorang Peneliti di the Naval Postgraduate School Center for Information Systems Security Studies and Research.

Proses yang dilakukan oleh foremost dikenal sebagai *data carving.* Foremost dapat digunakan pada file image, seperti yang

dihasilkan oleh dd, Safeback, Encase, dsb., atau secara langsung di drive. Header dan footer dapat diberikan dalam sebuah file konfigurasi atau anda dapat memberikan opsi perintah baris untuk menentukan tipe file *built-in.* Tipe built-in ini mencari struktur data format file yang diberikan sehingga memberikan recovery yang cepat dan handal. Jika tidak tersedia tipe built-in untuk format yang anda inginkan, anda dapat mendefisikan formatnya dalam file konfigurasi foremost.conf.

Beberapa format yang didukung secara built-in adalah gif, jpg, png, bmp, avi, mov, doc, html, pdf, wav, zip, rar, wmv, ppt, xls, sxw, sxc, dan sxi.
Berikut ini adalah beberapa buah contoh penggunaan foremost:

- Mencari format jpeg dan mengabaikan 100 blok pertama:
  ```
  for --most-s 109 -tjpg - image, cr:
  ```

- Mencari seluruh tipe yang telah didefmisikan secara built-in:
  ```
  for-'most -"-ail -i image . dd
  ```

- Mencari dokumen office dan file jpeg dalam mode rinci:
  ```
  foremost -"." -t oie,jpeg -i image, id
  ```

7.14. gqview

Gqview merupakan sebuah program untuk melihat gambar berbasis GTK la mendukung beragam format gambar, zooming, panning, thumbnails, dan pengurutan gambar.

7.15.galleta

Galleta merupakan sebuah tool yang ditulis oleh Keith J Jones untuk melakukan analisis forensikterhadap cookie Internet Explorer.

Banyak file penting dalam sistem operasi Microsoft Windows memiliki struktur yang tidak didokumentasikan. Salah satu prinsip utama forensik komputer adalah seluruh metodologi analisi harus didokumentasikan dengan baik dan dapat digunakan berulang-ulang, dan mereka harus memiliki margin kesalahan yang dapat diterima. Saat ini terdapat kekurangan metode dan tool open source yang dapat diandalkan oleh analis forensik untuk menguji data yang ditemukan dalam file proprietary Microsoft.

Banyak penyelidikan kejahatan komputer membutuhkan rekonstruktsi file cookie Internet Explorer. Karena teknik analisis ini dilakukan secara teratur, Keith menyelidiki struktur data yang ditemukan dalam file cookie. Galleta, yang berasal dari bahasa Spanyol dan berarti "cookie", dikembangkan untuk menguji isi file cookie. Galleta akan memeriksa informasi dalam file cookie dan mengeluarkan hasil dalam field delimited sehingga dapat diimpor ke program spreadsheet favorit Anda

Berikut ini adalah contoh penggunaan galleta:

```
gali-'ta antihackertoolki~.txt > cookies.txt
```

Bukalah file cookies.txt sebagai file TAB delimited dalam MS Excel untuk melakukan pengurutan dan pemfilteran.

7.16. lshw

lshw (Hardware Lister) merupakan sebuah tool kecil yang memberikan informasi detil mengenai konfigurasi hardware dalam mesin. la dapat melaporkan konfigurasi memori dengan tepat, versi firmware, konfigurasi mainboard, versi dan kecepatan CPU, konfigurasi cache, kecepatan bus, dsb. pada sistem t>MI-capable x86 atau sistem EFI

(IA-64) dan pada beberapa mesin PowerPC. Informasi dapat dihasilkan dalam bentuk teks biasa, XML, atau HTML.

Saat ini ia mendukung DMI (hanya x86 dan EFI), device OpenFirmware (hanya PowerPC), PCI/AGP, ISA PnP (x86), CPUID (x86), IDE/ATA/ATAPI, PCMCIA (hanya diuji pada x86), USB dan SCSI. Penggunaan:

```
Ishw [format] 'options. . . ]
```

dengan format adalah:

-x	menjalankanGUI(jikatersedia)
-html	mengaktifkan mode HTML
-xml	mengaktifkan mode XML
-short	mencetak path hardware
-bus info	mencetak informasi bus

dan opsi dapat berupa:

-enable TEST	mengadakantes
-disable TEST	meniadakantes
-class CLASS	membatasi output ke kelas tertentu
-C CLASS	alias bagi -class CLASS

PERHATIKAN: untuk menggunakan beberapa fitur (seperti DMI pada platform x86), anda perlu menjalankan lshw sebagai root atau ia hanya akan menampilkan laporan informasi parsial.

7.17. mac-robber

mac-robber merupakan sebuah tool Forensics & Incident Response yang digunakan untuk mengumpulkan waktu Modified, Access, dan Change (MAC) dari file yang dialokasikan. la secara rekursif membaca waktu MAC file dan direktori dan mencetaknya dalam format "mesin waktu" ke output standar. Format ini sama dengan yang dibaca oleh tool mactime dari TSK dan TCT.

Program ini memiliki beberapa keunggulan dibandingkan menggunakan grave-robber:

- la tidak membutuhkan Perl, secara default program dikompilasi dengan link statik.
- la menggunakan kode C yang sangat mendasar sehingga dapat dikompilasi dalam platform apapun.
- C lebih cepat dibandingkan Perl untuk operasi semacam ini.

Perhatikan bahwa tool ini tidak akan menampilkan file terhapus, file tidak teralokasi, atau file yang telah disembunyikan oleh rootkit. Untuk melihat informasi mengenai tipe-tipe file tersebut, gunakan tool khusus dari The Sleuth Kit.

mac-robber menerima daftar direktori untuk dianalisis sebagai argumen
Sebagai contoh. untuk menganah.sis direktori 'mnt' dan 'mnt2' dan menginm hasilnya ke

sebuah file perintahnya adalah

Jika anda ingin menganaiis sistem dari direktori root dan menginm datanya ke sebuah server yang menjalankan netcat, gunakan :

Server tersebut akan menjalankan sesuatu seperti:

```
* nc -1 -p ? 000 -• body.mac
```

Untuk menganalisis data, dibutuhkan tool mactime dari The Sleuth Kit Gunakan flag -b untuk mengimpor file badan:

7.18. Magicrescue

Magic Rescue membuka device untuk dibaca, memeriksanya untuk tipe file yang dikena! dan memanggil program eksterna! untuk mengekstraksi mereka. la melihat pada "magic bytes" dalam isi file, sehingga ia dapat digunakan sebagai utilitas undelete dan untuk mengembalikan drive atau partisi yang rusak. la bekerja pada sembarang filesistem, namun pada filesistem yang sangat terfragmentasi ia hanya dapat mengembalikan bagian pertama untuk setiap file. Namun demikian bagian ini seringkali

sebesarSOMB.

Untuk memanggil magicrescue, anda harus menspesifikasikan paling tidak satu

device dan opsi -d dan -r.

Perintah umumnya adalah sebagai berikut:

```
magicrescue [ options ] devices
```

Opsi yang didukung oleh magicrescue antara lain:

-b blocksize

Default: 1. Opsi ini akan mengarahkan magicrescue untuk hanya memperhatikan file yang dimulai dari kelipatan argumen *blocksize* Opsi ini hanya berlaku bagi resep yang menyertainya. sehingga dengan menspesifikasikannya berulang kali dapat digunakan untuk memperoleh perilaku resep yang berbeda. Dengan menggunakan opsi ini anda biasanya akan memperoleh kinerja yang *lebih baik, namun akan menemukan lebih sedikit file. Secara khusus, file dengan* leading garbage *(misalnya file mp3) dan file yang berada di dalam file lain biasanya akan dilewati. Dan juga beberapa filesistem tidak menyelaraskan file-file kecil ke ukuran blok, sehingga mereka juga tidak akan ditemukan.*

Jika anda tidak tahu ukuran blok filesistem anda, gunakan saja nilai 512, yang merupakan ukuran sektor hardware.

-D directory

Wajib. Direktori untuk menampung file yang ditemukan. Pastikan anda memiliki banyak ruang kosong dalam direktori ini, terutama ketika mengekstraksi tipe file yang sangat umum seperti jpeg atau gzip. Dan juga pastikan bahwa filesistem mampu menangani ribuan file dalam satu direktori, misalnya jangan gunakan FAT bila anda mengekstraksi banyak file.

Anda seharusnya tidak meletakkan direktori hasil pada device blok yang sama dengan file yang ingin anda selamatkan. Hal ini mungkin menambah file yang sama ke device blok sebelum pembacaan, yang menyebabkan magicrescue menemukan file yang sama kemudian. Daam kasus teoritis yang paling buruk, hal ini dapat menyebabkan terjadinya loop ketika file yang sama diekstraksi berulang kali hingga ruang disk habis. Anda juga mungkin akan menimpa file terhapus yang ingin anda cari.

-r recipe

Wajib. Nama resep, file atau direktori. Spesifikasikan sebagai nama file biasa (misalnya jpeg-jfif) atau sebuah path

(misalnya recipes/jpeg-jfif). *Secara default, ia akan mencari di* ./recipes *dan* PREFIX/share/magicrescue/recipes, *dengan PREF/Xadalah path instalasi magicrescue.*

Berikut ini adalah contoh penggunaan magicrescue:

Misalkan anda telah menghancurkan filesistem /dev/hdb1 dan anda ingin mengambil kembali seluruh file jpeg yang telah hilang. Panduan ini mengasumsikan anda telah menginstalasi MagicRescue di /usr/bin. *Pastikan DMA dan optimisasi lainnya telah diaktifkan pada disk anda, atau proses ini akan memakan waktu lama. Di Linux, gunakan hdparm untuk menset opsi ini:*

```
hdparm -d 1 -c 1 -u 1 /dev/hdb
```

Pilih direktori output yang memiliki ruang spasi besar:

```
mkdir -/output
```

Lihat di direktori /usr/share/magicrescue/recipes *untuk resep yang anda inginkan. MagicRescue menyertakan beberapa resep untuk tipe file umum, dan anda dapat juga membuat resep sendiri. Kebanyakan resep membutuhkan software pihak ketiga untuk bekerja, dan anda mungkin ingin memodifikasi beberapa parameter (seperti min_output_file) untuk memenuhi kebutuhan anda. Lalu panggil magicrescue:*

```
magicrescue -r jpeg-jfif -r jpeg-exif -d -/output /dev/hdbl
```

Ia akan memeriksa seluruh hard disk anda, jadi bersabarlah. Anda dapat menghentikannya dan melanjutkan kemudian bila anda ingin. Untuk melakukan hal itu, interupsi dengan menekan tombol CTRL+C dan perhatikan informasi kemajuan mengenai alamat selanjutnya. Lalu restart-lah kemudian dengan opsi -O. Bila telah selesai anda mungkin akan menemukan ribuan file jpg dalam -/output, *termasuk hal-hal yang tidak anda ketahui berada dalam cache browser anda. Penyortiran seluruh file tersebut merupakan tugas yang berat, sehingga anda mungkin akan menggunakan software atau skrip untuk melakukannya. Pertama, cobalah menghilangkan duplikasi dengan tool dupemap:*

```
dupemap delete,report -/output
```

Jika anda melakukan operasi undelet anda mungkin ingin menghapus seluruh file yang diselamatkan yang juga ada dalam filesistem live. Jika hal tersebut belum cukup, anda dapat menggunakan magicsort untuk memperoleh gambaran yang lebih baik:

```
magicsort ~/output
```

7.19.md5deep

mdSdeep adalah sebuah tool lintas platform yang menghitung signature MD5 file. mdSdeep serupa dengan mdSsum namun ia dapat memproses direktori secara rekursif, menghasilkan prakiraan waktu selesai, membandingkan file ke known hash set, *dan dapat diset hany untuk memproses tipe file tertentu saja.*

7.20. Memdump

Memdump adalah program yang dibuat untuk men-dump memori sistem ke stream output standar, melewati beberapa lubang dalam peta memori. Secara default, program akan mendump isi memori fisik (/dev/mem). Outputnya dalam bentuk dump raw, bila diperlukan gunakan opsi -m untuk menyimpan informasi layout memori.

Output sebaiknya dikirimkan ke host lain melalui jaringan, untuk menghindari perubahan memori dalam cache filesistem. Berikut ini adalah contoh mendump memori fisik ke host lain dengan menggunakan nc dan SSL:

```
memdump | nc host port
memdump | openssl s_client -connect host:port
```

Perintah memdump adalah sebagai berikut:
memdump t~kv] [-b buffer_size] [-d dump_size] [-mmap_file] [-p page_size]

Argumen ukuran di bawah ini mengerti akhiran k (kilo), m (mega) dan g (giga). Opsi:

-k
berusaha mendump memori kernel (/dev/kmem) alih-alih memori fisik.

-b *buffer_size* (default: 0)
Jumlah byte untuk operasi baca memori. Secara default, program menggunakan nilai *page_size.*

-d *dump-size* (default: 0)
Jumlah byte memori yang akan didump. Secara default, program berjalan hingga device memori melaporkan sebuah end-of-file (Linux), atau hingga ia mendump sebanyak mungkin memori dari /dev/mem yang dilaporkan kernel

(FreeBSD, Solaris), atau hingga pointer kembali ke semula.

-p *page_size* (default: 0)
Gunakan *page_s/ze* sebagai ukuran halaman memori. Secara default, program menggunakan ukuran page sistem.

7.21. Ophcrack

ophcrack merupakan sebuah password cracker Windows berbasiskan trade-off waktu-memori yang lebih cepat dengan menggunakan tabel rainbow.

Interface memungkinkan 3 cara menghasilkan hash password:

- encrypted SAM: hasillkan hash dari file SAM dan SYSTEM yang diperoleh dari mesin Windows dengan membooting menggunakan disk lain. Perhatikan dalam kasus ini anda tidak perlu mengetahui password Windows administrator untuk memperoleh hash tersebut.
- local SAM (hanya untuk ophcrack versi Windows): hasilkan hashes dari mesin Windows yang menjalankan program. Anda butuh password administrator ke mesin lokal.
- remote SAM (hanya untuk ophcrack versi Windows): hasilkan hash mesin Windows remote, anda harus mengetahui user dan password administrator dan nama share.

Anda dapat juga membongkar hash yang telah disimpan dalam sesi sebelumnya atau diperoleh dari tool lainnya.

Untuk membongkar hash, cukup tekan tombol Launch. Proses dapat diinterupsi dan hasilnya disimpan dalam file, yang dapat dimuatkan kembali.
Ophcrack membongkar hash NTLM dengan menggunakan set tabel bernama NTHASH. la membongkar 99% :

- password dengan panjang kurang dari 6 karakter yang terdiri dari karakterberikut:
  0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXY Z!"#$%&' ()*+,-./:;<=>?@[\]^_'{|}~
  (including the space character)
- password alfanumerik dengan panjang 7 (lower- dan uppercase)
- password alfanumerik dengan panjang 7 (hanya lowercase)

Tabel-tabel dapat didownload dari:

`http: //lasecwww.epf 1 . ch/~oechsiin/pro]ects/ophcr-ick`

Perhatikan bahwa Jumlah operasi yang dibutuhkan untuk membongkar password berkurang dengan meningkatnya ukuran tabel. Tabel besar akan lebih cepat sekitar4 kali.

Tabel yang digunakan oleh ophcrack tidak kompatibel dengan yang dihasilkan oleh tool lain bernama rainbowcrack. Tabel ophcrack lebih kompak dan memungkinkan pembongkaran password secara lebih cepat.

7.22.pasco

Banyak file penting dalam sistem operasi Microsoft Windows memiliki struktur yang tidak didokumentasikan. Salah satu prinsip utama forensik komputer adalah seluruh metodologi analisi harus didokumentasikan dengan baik dan dapat digunakan berulang-ulang, dan mereka harus memiliki margin kesalahan yang dapat diterima. Saat ini terdapat kekurangan metode dan tool open source yang dapat diandalkan oleh analis forensik untuk menguji data yang ditemukan dalam file proprietary Microsoft.

Banyak penyelidikan kejahatan komputer membutuhkan rekonstruksi aktivitas Internet tersangka. Karena teknik analisis ini dilakukan secara teratur, Keith menyelidiki struktur data yang ditemukan dalam file aktivitas Internet Explorer (file index.dat). Pasco, yang berasal dari bahasa Latin dan berarti "browse", dikembangkan untuk menguji isi file cache Internet Explorer. Pasco akan memeriksa informasi dalam file index.dat dan mengeluarkan hasil dalam field delimited sehingga dapat diimpor ke program spreadsheet favorit Anda.

Berikut ini adalah contoh penggunaan Pasco:

```
pasco index.dat > index.txt
```

Bukalah file index.txt sebagai file TAB delimited dalam MS Excel untuk melakukan pengurutan dan pemfilteran.

7.23. *PhotoRec*
PhotoRec merupakan software recovery data file yang dirancang untuk memulihkan file hilang termasuk video, dokumen dan arsip dari Hard Disk dan CDROM dan gambar yang hilang dari memori kamera digital.

PhotoRec mengabaikan filesistem dan mencari data yang ada di dalamnya, jadi ia akan tetap bekerja meskipun filesistem media anda telah rusak atau telah diformat. PhotoRec aman digunakan, ia tidak akan menulis ke drive atau memori yang berisikan data yang ingindirecover.

Photorecdapat mengembalikan file dari filesistem :
*FAT,
*NTFS,
*EXT2/EXT3filesystem
*HFS+

PhotoRec dapat digunakan untuk media HardDisk, CDROM, Compact Flash, Memory Stick, SecureDigital, SmartMedia, Microdrive, MMC, USB Memory Drives.

PhotoRec telah berhasil dicoba untuk kamera digital berikut:

* Canon EOS300D, 10D
* HP PhotoSmart620,850, 935
* Nikon CoolPix 775,950,5700
* Olympus C350N, C860L, Mju 400 Digital, Stylus 300
*SonyDSC-P9
* Praktica DCZ-3.4
* Casio ExilimEX-Z 750

PhotoRec mencari header file yang dikenal dan karena biasanya tidak ada fragmentasi data, ia dapat mengembalikan seluruh file. Photorec mengenali beragam format file termasuk ZIP, Office, PDF, HTML, JPEG dan beragam format file grafts lainnya. Format file yang dikenali PhotoRec lebih dari 80 buah.

7.24.rifiuti7

Banyak file penting dalam sistem operas! Microsoft Windows memiliki struktur yang tidak didokumentasikan. Salah satu prinsip utama forensik komputer adalah seluruh metodologi analisi harus didokumentasikan dengan baik dan dapat digunakan berulang-ulang, dan mereka harus memiliki margin kesalahan yang dapat diterima. Saat ini terdapat kekurangan metode dan tool open source yang dapat diandalkan oleh analis forensik untuk menguji data yang ditemukan dalam file proprietary Microsoft. Banyak penyelidikan kejahatan komputer membutuhkan rekonstruksi Recycle Bin tersangka. Karena teknik analisis ini dilakukan secara teratur, Keith menyelidiki struktur data yang ditemukan dalam file repository Recycle Bin (file INFO2). Rifiuti, yang berasal dari bahasa Italia dan berarti "trash", dikembangkan untuk menguji isi file INFO2 dalam Recycle Bin. Rifiuti akan memeriksa informasi dalam file INFO2 dan mengeluarkan hasil dalam field delimited sehingga dapat diimpor ke program spreadsheet favorit Anda.
Berikut ini adalah contoh penggunaan Rifiuti:

```
rifiuti INFO2 > INF02.txt
```

Bukalah file INFO2.txt sebagai file TAB delimited dalam MS Excel untuk melakukan pengurutan dan pemfilteran.

7.25. *rkhunter*

rkhunter merupakan sebuah tool untuk memeriksa rootkit, backdoor, dan eksploitasi lokal dengan menjalankan beberapa uji sebagai berikut:

- membandingkan hasil hash Md5
- memeriksa file baku yang digunakan oleh rootkit
- permisi file yang salah untuk file biner
- memeriksa string yang dicurigai dalam modul LKM dan KLD
- memeriksa file-file tersembunyi
- pemeriksaan opsional dalam file teks dan biner

rkhunter mampu mendeteksi berbagai ancaman sebagai berikut:

55808 Trojan - Variant
ADM WOrm
AjaKit
aPa Kit
Apache Worm
Ambient (ark) Rootkit
Balaur Rootkit
BeastKit
beX2
BOBKit
CiNIK Worm (Slapper.B
Kit
Danny-Boy's Abuse Kit
Devil RootKit
Dica
Dreams Rootkit
Duarawkz Rootkit
Flea Linux Rootkit
FreeBSD Rootkit
Fuck'it Rootkit
GasKit
Heroin LKM
HjC Rootkit
ignoKit
ImperalsS-FBRK
Irix Rootkit Kitko
Knark LiOn Worm
Lockit / LJK2 rootme
mod_rootme (Apache backdoor) TeLeKiT
MRK
variant)NiO Rootkit
NSDAP (RootKit for SunOS)URK (Universal RootKit)
Optic Kit (Tux) Oz Rootkit
Portacelo RSdstorm Toolkit
RH-Sharpe's rootkit
RSHA's rootkit Scalper
Worm Shutdown SHV4
Rootkit
SHV5 Rootkit
Sin Rootkit
Slapper
Sneakin Rootkit
Suckit
SunOS Rootkit
Superkit
TBD (Telnet BackDoor)
TOrn Rootkit
Trojanit
VcKit
Vole Rootkit
X-Org SunOS Rootkit
zaRwT.KiT Rootkit
Anti Anti-sniffer
LuCe LKM
THC Backdoor

Perangkat lunak yang dikembangkan oleh Michael Boelen ini telah diuji coba pada kebanyakan sistem operasi Linux dan BSD, selain itu juga bekerja di AIX4.1.5 /4.3.S, Solaris (SunOS). Pada versi terakhir rkhunter (1.2.7) ia dapat mendeteksi sekitar 60 rootkit, worm dan backdoor.

```
# rkhunter -c
Rootkit Hunter 1.2.7 is running
Determining OS... Unknown
Warning: This operating system is not fully supported!
```

```
Warning: Cannot find md5_not_known All MD5 checks will be skipped!

Check rootkits
* Default files and directories
   Rootkit '55808 Trojan - Variant A[1]                                    OK
   ADM Worm. . .                                                           ] t
   Rootkit 'AjaKit' . . .                                                  OK

Security advisories
* Check: Groups and Accounts
   Searching for /etc/passwd. . .                                           [ Found ]
   Checking users with UID '0' (root)...                                  [OK]
* Check: SSH
  Searching for sshd config... Found /etc/ssh/sshd config
Checking for allowed root login... Watch out Root login possible. Possible risk! info:
    Hint: See logfile for more information about this issue
   Checking for allowed protocols.. .                                       [OK (Only
SSH2 allowed) ]
* ———————— — Scan results ----------------------------
Md5
MD5 compared: 0
Incorrect MD5 checksums: 0
File scan
Scanned files: 342
Possible infected files: 0
Application scan
Vulnerable applications: 0
Scanning took 151 seconds
```

7.26. scalpel

calpel adalah sebuah tool forensik yang dirancang untuk mengidentifikasikan, mengisolasi dan merecover data dari media komputer selama proses investigasi forensik. Scalpel mencari hard drive, bit-stream image, unallocated space file, atau sembarang file komputer untuk karakteristik, isi atau atribut tertentu, dan menghasilkan laporan mengenai lokasi dan isi artifak yang ditemukan selama proses pencarian elektronik. Scalpel juga menghasilkan *(carves)* artifak yang ditemukan sebagai file individual.

7.27. Sha1 deep

shaldeep adalah sebuah tool lintas platform yang menghitung signature SHA1 file, shaldeep serupa dengan shalsum namun ia dapat memproses direktori secara rekursif, menghasilkan prakiraan waktu selesai, membandingkan file ke *known hash set,* dan dapat diset hany untuk memproses tipe file tertentu saja.

7.28.sha256eep

sha256deep adalah sebuah tool lintas platform yang menghitung signature SHA256-bit file sha256deep dapat memproses direktori secara rekursif, menghasilkan prakiraan waktu selesai, membandingkan file ke *known hash set,* dan dapat diset hany untuk memproses tipe file tertentu saja.

7.29. stegdetect

Stegdetect merupakan sebuah tool otomatis untuk mendeteksi isi steganografi dalam gambar. Ia mampu mendeteksi beberapa metode steganografi berbeda untuk menyembunyikaninformasitersembunyi dalam gambar JPEG. Saat ini, skema yang dapat dideteksi adalah:

```
jsteg,
jphide (unix dan windows),
invisible secrets,
out guess 01.3b,
F5 (analisis header),
appendX dan camouflage.
```

Stegdetect mendukung beragam fitur vektor berbeda dan secara otomatis menghitung karakteristik operasi penerima yang dapat digunakan untuk mengevaluasi kualitas fungsi deteksi yang dipelajari secara otomatis. Sebagai contoh:

```
stegdetect                          * . jpg
 cold  dvd. jpg                     : outguess(old) (***) jphide(*)
 dscf0001.jpg      : negative       steg(wonderland)
 dscf0002.jpg      : jsteg(***(     s, found 1 embeddings.
 dscf0003.jpg : jph del***)         : Cracks: 324123,   89:

   t stegbreak -t]  Perintah:
   Loaded I files.
   . dscf0002.jpg : stegdetect -t p auto.jpg
   Processed 1 file
   Time: 36
```

berusaha mendeteksi keberadaan informasi yang disembunyikan jphide dalam auto.jpg.

7.30. Tcpreplay

tcpreplay merupakan sebuah tool yang digunakan untuk mereplay lalu lintas jaringan dari file yang disimpan dengan tcpdump atau

tool lainnya yang menulis file pcap.

Operas! dasar tcpreplay adalah mengirim kembali seluruh paket dari file input dengan kecepatan saat mereka direkam, atau sesuai yang ditentukan, hingga kecepatan yang mampu ditangani oleh hardware.
Secara opsional, lalu lintas dapat dibagi antara dua interface, ditulis ke file, disaring dan diedit dengan beragam cara, memberikan cara untuk menguji firewall, NIDS dan device jannganlainnya.
Bentuk perintah tcpreplay adalah sebagai berikut:

tcpreplay [-flag [value] ! . . . [--opt-name [[=,] value]] . . . <Pc^.p_ flie (s) Berikut ini adalah sedikit opsi yang

bermanfaat:

-p number, --pps=number

Replay paket dengan kecepatan paket/detik.

-Mstring, --mbps=string

Replay paket dengan kecepatan Mbps yang diberikan.

7.37. *TestDisk*

TestDisk merupakan software *recovery* yang *powerful,* Ia mulanya dirancang untuk membantu mengembalikan partisi hilang dan/atau membuat disk *non-booting* menjadi *bootable* lagi ketika efek ini diakibatkan oleh software yang rusak, beberapa tipe virus tertentu atau kesalahan manusia. TestDisk dapat menemukan partisi hilang untuk seluruh filesistem ini:

*BeFS(BeOS)
* BSD disklabel (FreeBSD/OpenBSD/NetBSD)
* CramFS, Compressed File System
* DOS/Windows FAT12, FAT16 and FAT32
* HFS and HFS+, Hierarchical File System
* JFS, IBM's Journaled File System
* Linux Ext2 and Ext3
* Linux Raid
  + RAID 1: mirroring
  + RAID 4: striped array with parity device
  + RAID 5: striped array with distributed parity information
  + RAID 6: striped array with distributed dual redundancy information
* Linux Swap (versi 1 dan 2)
* LVM and LVM2, Linux Logical Volume Manager
* Mac partition map
* Novell Storage Services NSS
* NTFS (Windows NT/2K/XP/2003A/ista)
*ReiserFS3.5,3.6and4

7.32. *vinetto*
Vinetto merupakan sebuah tool forensik untuk memeriksa file Thumbs.db. Ia merupakan sebuah skrip Python perintah baris yang bekerja di platform Linux, Mac OS X dan Cygwin(win32).

Sistem operasi Windows (98, ME, 2000 dan XP) dapat menyimpan *thumbnails* dan metadata file gambar yang berada dalam direktori filesistem FAT32 atau NTFS. Thumbnails serta metadata yang terkait dengannya disimpan dalam file Thumbs.db. File Thumbs.db merupakan file berstruktur OLE yang tidak didokumentasikan.

Setelah file gambar dihapus dari filesistem, thumbnail dan metadata yang terkait dengannya masih tersimpan dalam file Thumbs.db. Sehingga data yang berada dalam file thumbs.db merupakan sumber informasi yang berharga bagi penyelidik forensik.

Vinetto dapat digunakan untuk mengekstraksi thumbnails metadatanya dari file Thumbs.db.

Ketika vinetto mencapai versi 0.98 beta, ia akan berfungsi dalam 3 mode::
- mode elementer: dalam mode ini vinetto hanya akan mengekstraksi thumbnails dan metadata dari file thumbs.dbterpilih.
- mode directory : dalam mode ini vinetto akan memeriksa konsistensi antara isi direktori dan file thumbs.db terkait, misalnya ia akan melaporkan thumbnails yang tidak memiliki file terkait dalam direktori.
- mode filesystem : dalam mode ini vinetto akan memproses seluruh partisi FAT atau NTFS. Vinetto akan membantu penyelidik forensik berbasis UNIX untuk:

- secara mudah melihat thumbnails gambar-gambar yang telah dihapus dalam sistem Windows,
- memperoleh informasi (tanggal, path, dsb.) tentang file yang telah dihapus tersebut.

Untuk dapat menjalankan vinetto dibutuhkan software-software berikut ini:

- Python-2.3 atau yang lebihbaru
- PIL (Python Imaging Library) 1.1.5 atau yang lebih baru. PIL digunakan untuk memperbaiki thumbnails tipe 1.

Saat ini vinetto masih memiliki keterbatasan yaitu ia tidak dapat merekonstruksi dengan benar beberapa thumbnails tipe 1 (format dalam keluarga mirip JPEG dengan header khusus, tabel huffman dan kuantisasi).

Perintah penggunaan vinetto adalah sebagai berikut:

```
vinetto [OPTIONS] [-s] [-U] [-o DIR] file
```

opsi: —version menampilkan nomor versi program dan keluar

-h, --help menampilkan pesan bantuan ini dan keluar
-o DIR menulis thumbnails ke DIR
-H menulislaporan HTML ke DIR
menggunakan encoding utf8

membuat symlink nama image ke nama bernomor dalam DIR/.thumbs

Berikut ini adalah beberapa contoh penggunaan perintah vinetto:

Menampilkan metadata yang ada dalam file thumbs.db: * Sun Solaris i386 disklabel
* Unix File System UFS and UFS2 (Sun/BSD/...)
* XFS, SGI's Journaled File System

TestDisk melakukan query ke BIOS atau sistem operas! untuk mencari Hard Disk dan karakteristiknya (ukuran LBA dan geometri CHS). TestDisk melakukan pemeriksaan singkat untuk struktur disk anda dan membandingkannya dengan label partisi. Jika tabel partisi memiliki kesalahan, TestDisk dapat memperbaikinya. Jika anda kehilangan partisi atau memiliki tabel partisi kosong, TestDisk dapat mencari partisi dan membuat tabel baru atau bahkan membuat MBR baru bila diperlukan.

```
$ vinetto Thumbs.db Root Entry modify time stamp : Mon Apr  3 13:35:58 2006

0001    Thu Dec  1 10:10:34 2005    netcat-1.png
0002    Thu Dec  1 10:20:46 2005    netcat-2.png
0003    Thu Dec  1 10:15:20 2005    netcat-3.png
```

Mengekstraksi thumbnails ke sebuah direktori:

```
$ vinetto -o /tmp/vinetto_output /path/to/Thumbs.db
```

Mengekstraksi thumbnails terkait ke sebuah direktori dan menghasilkan laporan HTML untuk melihatthumbnails melalui browser:

```
$ vinetto -Ho /tmp/vinetto_output /path/to/Thumbs.db
```

LAMPIRAN : Studi Kasus

Kasus Joe Jacobs

Joe Jacob, 28, telah ditangkap polisi kemarin dengan tuduhan menjual obat-obatan terlarang kepada para siswa menengah. Seorang polisi menyamar menjadi siswa menengah yang didekati oleh Joe Jacob di area parkir Smith Hill High School. Jacob bertanya kepada polisi yang sedang menyamar apakah ia bermaksud untuk membeli mariyuana. Sebelum polisi yang menyamar memberikan jawaban, Jacob mengeluarkan beberapa dari kantongnya dan menunjukkannya kepada petugas itu. Jacob berkata kepada petugas"Lihat ini, Kolombia tidakmemiliki yang lebih baikdari ini! Pemasok saya tidak hanya menjualnya langsung kepada saya tetapi mereka menanamnya sendiri."

Jacob sering terlihat berada di area parkir beberapa sekolah menegah sekitar jam 2:30 siang, waktu yang biasanya jam sekolah berakhir untuk hari itu. Pihak sekolah dari berbagai sekolah menengah sudah menghubungi polisi mengenai keberadaan Jacob di sekolah mereka dan mencatat peningkatan penggunaan obat-obatan terlarang di antara pelajar, semenjak kedatangannya.

Para polisi membutuhkan bantuan. Mereka ingin membuktikan apakah Joe Jacob juga menjual obat-obatan terlarang kepada pelajar di sekolah lain di samping Smith Hill. Masalahnya adalah tidak ada pelajar yang akan datang untuk membantu polisi. Berdasarkan komentar Joe mengenai Kolombia, polisi tertarik untuk mencari siapakah yang memasok mariyuana kepada Joe Jacob.

Jacob menolak mengatakan jika ia menjual obat-obatan tertarang selain di Smith Hill dan dia juga menolak untuk mengatakan siapakah nama pemasoknya. Jacob juga menolak membenarkan pernyataan yang ia katakan kepada polisi yang menyamar sebelum ia ditangkap. Setelah menerima surat perintah untuk menggeledah rumah yang dicurigai polisi hanya menemukan sejumlah kecil mariyuana. Polisi juga menyita sebuah floppy disk, tetapi tidak ada komputer atau media lain yang ada di rumah itu. Polisi telah membuat file image dari floppy tersebut dan memberikan salinannya kepada anda Mereka meminta bantuan anda untuk menguji floppy disk tersebut dan memberikan jawaban terhadap beberapa pertanyaan berikut:

1. Siapakah supplier mariyuana untuk Joe Jacob dan di manakah alamat supplier tersebut ?
2. Apakah data penting yang ada di dalam file coverpage.jpg dan mengapa data ini penting ?
3. Ke sekolah mana lagi Joe Jacob sering berkunjung selain Smith Hill (bila ada) ?
4. Untuk setiap file, proses apa saja yang dilakukan oleh tersangka untuk melindunginya dari pihak lain ?
5. Proses apakah yang anda (penyelidik) gunakan untuk menguji isi seluruh file dengan sukses ?

Keterangan yang didapat dari kepolisian : Disk Properties

Total Size 1.4MB (2,880 sectors)
Volume "Volume A"
Parameters File System: FAT12
Sectors Per Cluster: 1
Total Sectors: 2,880
Total Clusters: 2,847
Free Clusters: 2, 805
Volume Kame: NO NAME OEM
Version: MSDOS5.0
Heads: 2
Unused Sectors: 0
Sectors Per FAT: 9
Drive Type: Removable
Bytes Per Sector: 512
Total Capacity: 1,457,664 bytes (1.4MB)
Unallocated: 1,436,160 bytes U-4MB)
Allocated: 21504 bytes (21.0KB)
Volume Offset: 0
Volume Serial *: C4B1-CDCF
Sectors Per Track: 18
Number of FATs: 2
Boot Sectors: *I*

DAFTAR PUSTAKA

Barry J Grundy, The Law Enforcement and Forensic Examiner's Introduction to Linux, *versi 3.20, Oktober 2007.*
BJ Gleason dan Drew Fahey, Helix 1.7 for Beginners, *Maret 2006.*
NIST, Guide to Integrating Forensic Technique into Incident Response, *Special Publication 800-86, Agustus 2006.*