

BAB V

KESIMPULAN DAN SARAN

V.1. Kesimpulan

Berdasarkan pembahasan dalam bab-bab sebelumnya, dapat ditarik kesimpulan dari penelitian ini, yaitu bahwa sistem penyelamatan dokumen yang "disandera" *malware* dapat menyelamatkan dokumen yang "disandera" *malware* secara otomatis, dan sekaligus mudah diakses serta digunakan para pengguna komputer.

V.2. Saran

Beberapa saran yang dapat diberikan dari proses studi pustaka hingga penulisan laporan penelitian ini adalah:

1. Sebaiknya sistem dikembangkan agar tidak memerlukan "birokrasi" yang panjang dan sumber daya yang banyak untuk dapat menyelamatkan sebuah dokumen yang "disandera" *malware*.
2. Sebaiknya subsistem Antarmuka Web dikembangkan agar memiliki fitur keanggotaan, agar para pengguna sistem dapat mengelola *file-file*-nya masing-masing (memeriksa status, menghapus, dsb.) dengan mudah dari sebuah antarmuka.
3. Sebaiknya subsistem Antarmuka Web dikembangkan agar mendukung pengajuan beberapa *file* sekaligus untuk diselamatkan, untuk memudahkan para pengguna sistem yang memiliki banyak *file* yang ingin diselamatkan dokumennya.

4. Sebaiknya subsistem Penjadwal dikembangkan agar menggunakan *virtual server* (atau pustaka pengendali *virtual server*) lain yang lebih handal dan stabil.
5. Sebaiknya subsistem Penyelamat dikembangkan agar tidak memerlukan instalasi program penampil (*viewer*) sebuah jenis dokumen dalam *virtual machine* untuk dapat menyelamatkan dokumen-dokumen berjenis tersebut.



DAFTAR PUSTAKA

F-Secure Corporation, 2007, *F-Secure Reports Amount of Malware Grew by 100% during 2007*, siaran pers, Helsinki, Finlandia, URL: http://www.f-secure.com/en_EMEA/about-us/pressroom/news/2007/fs_news_20071204_1_eng.html, diakses 1 November 2009.

Microsoft Press, 2002, *Microsoft Computer Dictionary*, ed. 5, Microsoft Press, Redmond, Washington.

Richter, J., 1999, *Programming Applications for Microsoft Windows*, ed. 4, Microsoft Press, Redmond, Washington.

Skoudis, E. dan Zeltser, L., 2003, *Malware: Fighting Malicious Code*, Prentice Hall PTR, Upper Saddle River, New Jersey.

VMware Incorporated, 2008, *Virtualization Basics*, informasi teknologi virtualisasi, Palo Alto, California, URL: <http://www.vmware.com/virtualization/>, diakses 1 November 2008.

SKPL

Spesifikasi Kebutuhan Perangkat Lunak

PANDORA

(Sistem Penyelamatan Dokumen yang
"Disandera" Malware)


untuk:

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Ronald Budi Gunawan (TF / 04366)

Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta	Nomor Dokumen		Halaman
		SKPL-PANDORA		1/62
		Revisi	-	1 Februari 2010

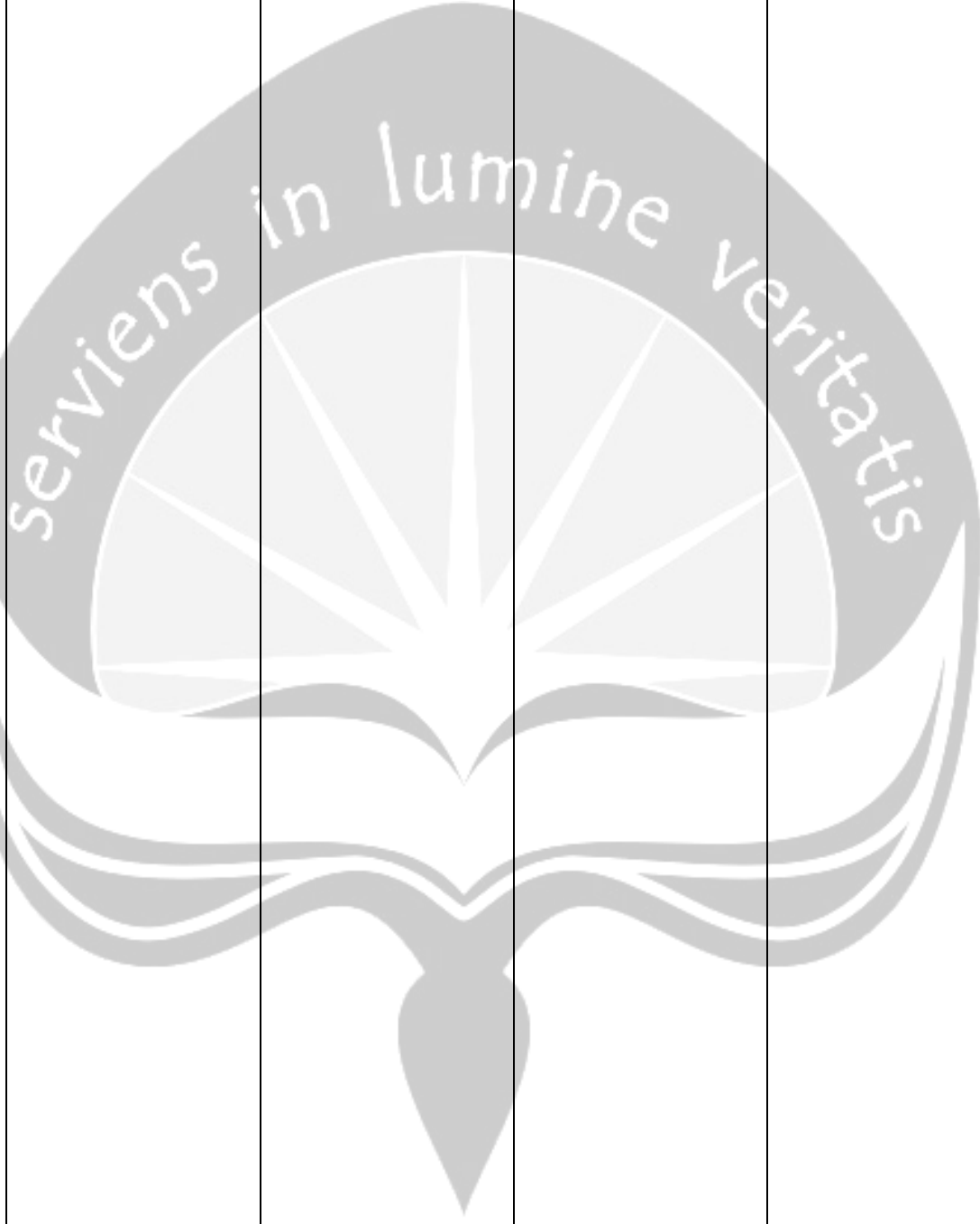
Daftar Perubahan

Revisi	Deskripsi
A	
B	
C	
D	
E	

INDEX TGL	A	B	C	D	E
Ditulis oleh					
Diperiksa oleh					
Disetujui oleh					

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



DAFTAR ISI

1. Pendahuluan.....	12
1.1. Tujuan.....	12
1.2. Lingkup Masalah.....	12
1.3. Definisi, Akronim, dan Singkatan.....	13
1.4. Referensi.....	13
1.5. Deskripsi Umum (Overview).....	14
2. Deskripsi Kebutuhan.....	15
2.1. Perspektif Produk.....	15
2.1.1. Antarmuka Sistem.....	15
2.1.2. Antarmuka Pemakai.....	15
2.1.3. Antarmuka Perangkat Keras.....	15
2.1.4. Antarmuka Perangkat Lunak.....	15
2.1.5. Antarmuka Komunikasi.....	17
2.1.6. Batasan Memori.....	17
2.1.7. Operasi.....	18
2.2. Fungsi Produk.....	18
2.3. Karakteristik Pengguna.....	20
2.4. Batasan-Batasan.....	20
2.5. Asumsi dan Ketergantungan.....	21
3. Kebutuhan Khusus.....	22
3.1. Kebutuhan Antarmuka Eksternal.....	22
3.1.1. Antarmuka Pemakai.....	22
3.1.2. Antarmuka Perangkat Keras.....	22
3.1.3. Antarmuka Perangkat Lunak.....	23
3.1.4. Antarmuka Komunikasi.....	23
3.2. Kebutuhan Fungsionalitas.....	24
3.2.1. Aliran Informasi.....	24
3.2.1.1. DFD Level 0 (Diagram Konteks) PANDORA..	24
3.2.1.1.1. Entitas Data.....	24
3.2.1.1.2. Proses.....	24
3.2.1.1.3. Topologi.....	25
3.2.1.2. DFD Level 1 Proses PANDORA.....	25
3.2.1.2.1. Entitas Data.....	25
3.2.1.2.2. Proses.....	25
3.2.1.2.3. Topologi.....	27
3.2.1.3. DFD Level 2 Proses Antarmuka Web.....	28
3.2.1.3.1. Entitas Data.....	28
3.2.1.3.2. Proses.....	28
3.2.1.3.3. Topologi.....	29
3.2.1.4. DFD Level 2 Proses Penjadwal.....	29
3.2.1.4.1. Entitas Data.....	29
3.2.1.4.2. Proses.....	29
3.2.1.4.3. Topologi.....	30
3.2.1.5. DFD Level 2 Proses Penyelamat.....	30

3.2.1.5.1. Entitas Data	30
3.2.1.5.2. Proses	30
3.2.1.5.3. Topologi	31
3.2.1.6. DFD <i>Level 3</i> Proses Menampilkan Rincian Pekerjaan Penyelamatan	31
3.2.1.6.1. Entitas Data	31
3.2.1.6.2. Proses	32
3.2.1.6.3. Topologi	33
3.2.1.7. DFD <i>Level 3</i> Proses Mengelola Pekerjaan Penyelamatan	33
3.2.1.7.1. Entitas Data	33
3.2.1.7.2. Proses	33
3.2.1.7.3. Topologi	34
3.2.2. Deskripsi Proses	34
3.2.2.1. Proses <i>Log In</i>	34
3.2.2.1.1. Entitas Data Masukan	34
3.2.2.1.2. Algoritma atau Formula dari Proses .	35
3.2.2.1.3. Entitas Data Terlibat	35
3.2.2.2. Proses Mengajukan Pekerjaan Penyelamatan	35
3.2.2.2.1. Entitas Data Masukan	35
3.2.2.2.2. Algoritma atau Formula dari Proses .	35
3.2.2.2.3. Entitas Data Terlibat	36
3.2.2.3. Menampilkan Informasi dan Status Penyelamatan	36
3.2.2.3.1. Entitas Data Masukan	36
3.2.2.3.2. Algoritma atau Formula dari Proses .	37
3.2.2.3.3. Entitas Data Terlibat	37
3.2.2.4. Proses <i>Download</i> Dokumen yang Diselamatkan	37
3.2.2.4.1. Entitas Data Masukan	37
3.2.2.4.2. Algoritma atau Formula dari Proses .	37
3.2.2.4.3. Entitas Data Terlibat	38
3.2.2.5. Proses <i>Download Executable Malware</i>	38
3.2.2.5.1. Entitas Data Masukan	38
3.2.2.5.2. Algoritma atau Formula dari Proses .	38
3.2.2.5.3. Entitas Data Terlibat	39
3.2.2.6. Proses Menampilkan Catatan Penyelamatan	39
3.2.2.6.1. Entitas Data Masukan	39
3.2.2.6.2. Algoritma atau Formula dari Proses .	39
3.2.2.6.3. Entitas Data Terlibat	40
3.2.2.7. Proses Menampilkan Daftar Pekerjaan Penyelamatan	40
3.2.2.7.1. Entitas Data Masukan	40
3.2.2.7.2. Algoritma atau Formula dari Proses .	40
3.2.2.7.3. Entitas Data Terlibat	40
3.2.2.8. Proses Menghapus Pekerjaan Penyelamatan	41
3.2.2.8.1. Entitas Data Masukan	41
3.2.2.8.2. Algoritma atau Formula dari Proses .	41
3.2.2.8.3. Entitas Data Terlibat	41

3.2.2.9.	Proses Mengulang Pekerjaan Penyelamatan	41
3.2.2.9.1.	Entitas Data Masukan	41
3.2.2.9.2.	Algoritma atau Formula dari Proses	42
3.2.2.9.3.	Entitas Data Terlibat	42
3.2.2.10.	Proses <i>Log Out</i>	42
3.2.2.10.1.	Entitas Data Masukan	42
3.2.2.10.2.	Algoritma atau Formula dari Proses	42
3.2.2.10.3.	Entitas Data Terlibat	43
3.2.2.11.	Proses Merawat Sistem	43
3.2.2.11.1.	Entitas Data Masukan	43
3.2.2.11.2.	Algoritma atau Formula dari Proses	43
3.2.2.11.3.	Entitas Data Terlibat	44
3.2.2.12.	Proses Menjadwalkan Pekerjaan Penyelamatan	44
3.2.2.12.1.	Entitas Data Masukan	44
3.2.2.12.2.	Algoritma atau Formula dari Proses	45
3.2.2.12.3.	Entitas Data Terlibat	46
3.2.2.13.	Proses Mengendalikan Penyelamatan	47
3.2.2.13.1.	Entitas Data Masukan	47
3.2.2.13.2.	Algoritma atau Formula dari Proses	47
3.2.2.13.3.	Entitas Data Terlibat	47
3.2.2.14.	Proses Meng-hook Windows API	47
3.2.2.14.1.	Entitas Data Masukan	47
3.2.2.14.2.	Algoritma atau Formula dari Proses	47
3.2.2.14.3.	Entitas Data Terlibat	48
3.2.2.15.	Proses Memantau Panggilan Windows API	48
3.2.2.15.1.	Entitas Data Masukan	48
3.2.2.15.2.	Algoritma atau Formula dari Proses	48
3.2.2.15.3.	Entitas Data Terlibat	48
3.2.3.	Spesifikasi Data	49
3.2.3.1.	Konstruksi Data Pekerjaan Penyelamatan	49
3.2.3.1.1.	Tipe <i>Record</i>	49
3.2.3.1.2.	<i>Field-Field</i>	49
3.2.4.	Kamus Data	50
3.2.4.1.	Elemen Data ID Pekerjaan Penyelamatan	50
3.2.4.1.1.	Nama	50
3.2.4.1.2.	Representasi	50
3.2.4.1.3.	Unit atau Format	50
3.2.4.1.4.	Presisi atau Keakuratan	50
3.2.4.1.5.	<i>Range</i>	50
3.2.4.2.	Elemen Data Nama <i>File Executable Malware</i>	50
3.2.4.2.1.	Nama	50
3.2.4.2.2.	Representasi	51
3.2.4.2.3.	Unit atau Format	51
3.2.4.2.4.	Presisi atau Keakuratan	51
3.2.4.2.5.	<i>Range</i>	51
3.2.4.3.	Elemen Data Ukuran <i>File Executable Malware</i>	51
3.2.4.3.1.	Nama	51

3.2.4.3.2. Representasi	51
3.2.4.3.3. Unit atau Format	51
3.2.4.3.4. Presisi atau Keakuratan	51
3.2.4.3.5. <i>Range</i>	51
3.2.4.4. Elemen Data <i>Hash MD5 Executable Malware</i>	52
3.2.4.4.1. Nama	52
3.2.4.4.2. Representasi	52
3.2.4.4.3. Unit atau Format	52
3.2.4.4.4. Presisi atau Keakuratan	52
3.2.4.4.5. <i>Range</i>	52
3.2.4.5. Elemen Data <i>Hash SHA-1 Executable Malware</i>	52
3.2.4.5.1. Nama	52
3.2.4.5.2. Representasi	52
3.2.4.5.3. Unit atau Format	52
3.2.4.5.4. Presisi atau Keakuratan	52
3.2.4.5.5. <i>Range</i>	53
3.2.4.6. Elemen Data Nama <i>File</i> Dokumen yang Diselamatkan	53
3.2.4.6.1. Nama	53
3.2.4.6.2. Representasi	53
3.2.4.6.3. Unit atau Format	53
3.2.4.6.4. Presisi atau Keakuratan	53
3.2.4.6.5. <i>Range</i>	53
3.2.4.7. Elemen Data Ukuran <i>File</i> Dokumen yang Diselamatkan	53
3.2.4.7.1. Nama	53
3.2.4.7.2. Representasi	53
3.2.4.7.3. Unit atau Format	53
3.2.4.7.4. Presisi atau Keakuratan	54
3.2.4.7.5. <i>Range</i>	54
3.2.4.8. Elemen Data <i>Hash MD5</i> Dokumen yang Diselamatkan	54
3.2.4.8.1. Nama	54
3.2.4.8.2. Representasi	54
3.2.4.8.3. Unit atau Format	54
3.2.4.8.4. Presisi atau Keakuratan	54
3.2.4.8.5. <i>Range</i>	54
3.2.4.9. Elemen Data <i>Hash SHA-1</i> Dokumen yang Diselamatkan	54
3.2.4.9.1. Nama	54
3.2.4.9.2. Representasi	54
3.2.4.9.3. Unit atau Format	55
3.2.4.9.4. Presisi atau Keakuratan	55
3.2.4.9.5. <i>Range</i>	55
3.2.4.10. Elemen Data Status	55
3.2.4.10.1. Nama	55
3.2.4.10.2. Representasi	55
3.2.4.10.3. Unit atau Format	55

3.2.4.10.4.	Presisi atau Keakuratan	55
3.2.4.10.5.	<i>Range</i>	55
3.2.4.11.	Elemen Data Alamat IP Pengaju	55
3.2.4.11.1.	Nama	55
3.2.4.11.2.	Representasi	56
3.2.4.11.3.	Unit atau Format	56
3.2.4.11.4.	Presisi atau Keakuratan	56
3.2.4.11.5.	<i>Range</i>	56
3.2.4.12.	Elemen Data Waktu Pengajuan	56
3.2.4.12.1.	Nama	56
3.2.4.12.2.	Representasi	56
3.2.4.12.3.	Unit atau Format	56
3.2.4.12.4.	Presisi atau Keakuratan	56
3.2.4.12.5.	<i>Range</i>	56
3.2.4.13.	Elemen Data Waktu Pemrosesan	57
3.2.4.13.1.	Nama	57
3.2.4.13.2.	Representasi	57
3.2.4.13.3.	Unit atau Format	57
3.2.4.13.4.	Presisi atau Keakuratan	57
3.2.4.13.5.	<i>Range</i>	57
3.2.4.14.	Elemen Data Waktu Selesai	57
3.2.4.14.1.	Nama	57
3.2.4.14.2.	Representasi	57
3.2.4.14.3.	Unit atau Format	57
3.2.4.14.4.	Presisi atau Keakuratan	57
3.2.4.14.5.	<i>Range</i>	58
3.2.4.15.	Elemen Data <i>Executable Malware</i>	58
3.2.4.15.1.	Nama	58
3.2.4.15.2.	Representasi	58
3.2.4.15.3.	Unit atau Format	58
3.2.4.16.	Elemen Data Dokumen yang Diselamatkan .	58
3.2.4.16.1.	Nama	58
3.2.4.16.2.	Representasi	58
3.2.4.16.3.	Unit atau Format	58
3.2.4.17.	Elemen Data Catatan Penyelamatan	58
3.2.4.17.1.	Nama	58
3.2.4.17.2.	Representasi	59
3.2.4.17.3.	Unit atau Format	59
3.2.4.18.	Elemen Data Keterhapusan	59
3.2.4.18.1.	Nama	59
3.2.4.18.2.	Representasi	59
3.2.4.18.3.	Unit atau Format	59
3.2.4.18.4.	Presisi atau Keakuratan	59
3.2.4.18.5.	<i>Range</i>	59
3.2.5.	<i>Entity-Relationship Diagram</i>	60
3.3.	Kebutuhan Performansi	60
3.3.1.	Kebutuhan Numerik Statik	60
3.3.2.	Kebutuhan Numerik Dinamis	60
3.4.	Batasan Perancangan	61

3.5. Atribut Sistem Perangkat Lunak..... 61



DAFTAR TABEL

Tabel 1.1. Definisi, Akronim, dan Singkatan..... 13



DAFTAR GAMBAR

Gambar 3.1. DFD <i>Level</i> 0 PANDORA	25
Gambar 3.2. DFD <i>Level</i> 1 PANDORA	27
Gambar 3.3. DFD <i>Level</i> 2 Antarmuka Web	29
Gambar 3.4. DFD <i>Level</i> 2 Penjadwal	30
Gambar 3.5. DFD <i>Level</i> 2 Penyelamat	31
Gambar 3.6. DFD <i>Level</i> 3 Menampilkan Rincian Pekerjaan Penyelamatan	33
Gambar 3.7. DFD <i>Level</i> 3 Mengelola Pekerjaan Penyelamatan	34
Gambar 3.8. <i>Entity-Relationship Diagram</i>	60



SISTEM PENYELAMATAN DOKUMEN YANG "DISANDERA" MALWARE

(PANDORA)

1. Pendahuluan

1.1. Tujuan

Tujuan dari dokumen spesifikasi kebutuhan perangkat lunak (SKPL-PANDORA) dalam pengembangan perangkat lunak PANDORA (Sistem Penyelamatan Dokumen yang "Disandera" Malware) adalah untuk mendefinisikan kebutuhan perangkat lunak yang meliputi: antarmuka eksternal (antarmuka antara sistem dengan sistem lain, perangkat lunak, perangkat keras, dan pengguna), performansi (kemampuan perangkat lunak dari segi kecepatan, tempat penyimpanan yang dibutuhkan, serta keakuratan), dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta untuk mendefinisikan fungsi perangkat lunak. SKPL-PANDORA ini juga mendefinisikan batasan perancangan perangkat lunak.

Dokumen SKPL ini menjadi dasar kesepakatan antara pihak pelanggan dengan pihak pengembang mengenai perangkat lunak yang akan dikembangkan.

1.2. Lingkup Masalah

Perangkat lunak PANDORA dikembangkan untuk menangani penyelamatan dokumen yang "disandera" malware, dan terdiri atas tiga subsistem, yaitu:

1. Antarmuka Web, merupakan subsistem yang menyajikan antarmuka web agar para pengguna sistem dapat berinteraksi dengan sistem,

2. Penjadwal, merupakan subsistem yang menangani perawatan sistem dan penjadwalan pengerjaan setiap pekerjaan penyelamatan dalam Penyelamat, dan
3. Penyelamat, merupakan subsistem yang mengerjakan setiap pekerjaan penyelamatan yang diberikan Penjadwal.

1.3. Definisi, Akronim, dan Singkatan

Daftar definisi, akronim, dan singkatan:

Tabel 1.1. Definisi, Akronim, dan Singkatan

Keyword / Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL_XX	Kode yang merepresentasikan kebutuhan pada perangkat lunak PANDORA.
DFD	<i>Data Flow Diagram</i> , merupakan teknis grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan saat data bergerak dari <i>input</i> menjadi <i>output</i> .
ERD	<i>Entity-Relationship Diagram</i> , merupakan teknis grafis / diagram yang menggambarkan objek dan hubungan antar objek.
PANDORA	Perangkat lunak yang menangani penyelamatan dokumen yang "disandera" <i>malware</i> .

1.4. Referensi

Dokumen yang digunakan sebagai acuan dalam rencana pengembangan perangkat lunak ini adalah:

1. Software Engineering Standards Committee, IEEE Computer Society. 1998. *IEEE Std 830-1998: IEEE*

Recommended Practice for Software Requirements Specification. Piscataway, New Jersey: IEEE Press.

2. Pressman, Roger S. 2002. *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku I)*. Ed. II. Terj. LN Harnaningrum. Yogyakarta: ANDI.
3. Pressman, Roger S. 2003. *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku II)*. Ed. II. Terj. LN Harnaningrum. Yogyakarta: ANDI.

1.5. Deskripsi Umum (Overview)

Secara umum, dokumen SKPL ini terbagi atas 3 bagian utama. Bagian pertama berisi penjelasan mengenai dokumen SKPL yang mencakup: tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak, definisi, referensi, dan deskripsi umum tentang dokumen SKPL.

Bagian kedua berisi penjelasan umum tentang perangkat lunak PANDORA yang akan dikembangkan, mencakup: perspektif produk yang akan dikembangkan, fungsi perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak, dan asumsi yang dipakai dalam pengembangan perangkat lunak PANDORA.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak PANDORA yang akan dikembangkan.

2. Deskripsi Kebutuhan

2.1. Perspektif Produk

Perangkat lunak PANDORA merupakan perangkat lunak yang menangani penyelamatan dokumen yang "disandera" malware.

2.1.1. Antarmuka Sistem

Perangkat lunak PANDORA berdiri sendiri dan tidak berinteraksi dengan sistem lainnya.

2.1.2. Antarmuka Pemakai

Karakteristik antarmuka pemakai adalah:

interaksi anonim dan administrator dengan sistem dilakukan dengan *keyboard* dan *mouse*, melalui halaman web (dalam *web browser*) yang ditampilkan pada layar monitor beresolusi 1024×768.

2.1.3. Antarmuka Perangkat Keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak PANDORA adalah:

1. *keyboard*,
2. *mouse*, dan
3. *modem* atau *LAN card*.

2.1.4. Antarmuka Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam pengoperasian perangkat lunak PANDORA adalah:

1. nama : Debian GNU/Linux,
nomor versi : 4.0 "Etch",
sumber : Software in the Public Interest,

sebagai sistem operasi tempat perangkat lunak PANDORA berjalan,

2. nama : Apache HTTP Server,
nomor versi : 2.2,
sumber : Apache Software Foundation,
sebagai *web server* tempat proses Antarmuka Web PANDORA berjalan,
3. nama : PHP,
nomor versi : 5.2,
sumber : PHP Group,
sebagai *interpreter* yang menjalankan proses Antarmuka Web PANDORA,
4. nama : MySQL Community Server,
nomor versi : 5.0,
sumber : Oracle Corporation,
sebagai *database server* yang menyimpan data-data perangkat lunak PANDORA,
5. nama : NFS Server,
nomor versi : 3,
sumber : komunitas,
sebagai *file server* yang menyimpan data-data perangkat lunak PANDORA,
6. nama : Perl,
nomor versi : 5.8,
sumber : Larry Wall,
sebagai *interpreter* yang menjalankan proses Penjadwal PANDORA,

7. nama : VMware Server,
nomor versi : 1.0,
sumber : VMware,
sebagai *virtual server* tempat proses Penyelamat
PANDORA berjalan,

8. nama : Windows,
nomor versi : XP Professional Service Pack 2,
sumber : Microsoft,
sebagai sistem operasi tempat proses Penyelamat
PANDORA berjalan, dan

9. nama : C++ Run-time Library,
nomor versi : 6.0,
sumber : Microsoft,
sebagai pustaka untuk menjalankan proses Penyelamat
PANDORA.

2.1.5. Antarmuka Komunikasi

Antarmuka komunikasi yang digunakan dalam pengoperasian perangkat lunak PANDORA adalah protokol TCP/IP.

2.1.6. Batasan Memori

Batasan memori primer yang dibutuhkan dalam pengoperasian PANDORA yaitu RAM minimal 1 GB ditambah 512 MB per *virtual machine*. Sedangkan memori sekunder perangkat lunak PANDORA minimal 5 GB ditambah 2,5 GB per *virtual machine*.

2.1.7. Operasi

Hanya ada satu modus operasi yang didukung perangkat lunak PANDORA, yaitu modus operasi baku.

2.2. Fungsi Produk

Fungsi perangkat lunak PANDORA adalah:

1. Antarmuka Web (SKPL_01), merupakan fungsi yang menyajikan antarmuka web agar para pengguna sistem dapat berinteraksi dengan sistem.

Fungsi-fungsi yang ada dalam SKPL_01 ini adalah:

- a. *Log In* (SKPL_01_01), merupakan fungsi yang menangani otentikasi administrator ke dalam sistem melalui *password*.
- b. Mengajukan Pekerjaan Penyelamatan (SKPL_01_02), merupakan fungsi yang menangani pengajuan dan penyimpanan pekerjaan penyelamatan.
- c. Menampilkan Rincian Pekerjaan Penyelamatan (SKPL_01_03), merupakan fungsi yang menangani penampilan berbagai informasi tentang sebuah pekerjaan penyelamatan.

Fungsi-fungsi yang ada dalam SKPL_01_03 ini adalah:

- 1) Menampilkan Informasi dan Status Penyelamatan (SKPL_01_03_01), merupakan fungsi yang menangani penampilan informasi dan status penyelamatan sebuah pekerjaan penyelamatan.
- 2) *Download* Dokumen yang Diselamatkan (SKPL_01_03_02), merupakan fungsi yang menangani *download* dokumen yang diselamatkan dari sebuah pekerjaan penyelamatan.
- 3) *Download Executable Malware* (SKPL_01_03_03), merupakan fungsi yang menangani *download executable malware* sebuah pekerjaan penyelamatan.

4) Menampilkan Catatan Penyelamatan (SKPL_01_03_04), merupakan fungsi yang menangani penampilan catatan penyelamatan sebuah pekerjaan penyelamatan.

d. Mengelola Pekerjaan Penyelamatan (SKPL_01_04), merupakan fungsi yang menangani pengelolaan pekerjaan-pekerjaan penyelamatan.

Fungsi-fungsi yang ada dalam SKPL_01_04 ini adalah:

1) Menampilkan Daftar Pekerjaan Penyelamatan (SKPL_01_04_01), merupakan fungsi yang menangani penampilan daftar pekerjaan penyelamatan.

2) Menghapus Pekerjaan Penyelamatan (SKPL_01_04_02), merupakan fungsi yang menangani penandaan pekerjaan penyelamatan untuk dihapus.

3) Mengulang Pekerjaan Penyelamatan (SKPL_01_04_03), merupakan fungsi yang menangani pengulangan pemrosesan pekerjaan penyelamatan.

e. *Log Out* (SKPL_01_05), merupakan fungsi yang menangani deotentikasi administrator dari dalam sistem.

2. Penjadwal (SKPL_02), merupakan fungsi yang menangani perawatan sistem dan penjadwalan pengerjaan setiap pekerjaan penyelamatan dalam Penyelamat.

Fungsi-fungsi yang ada dalam SKPL_02 ini adalah:

a. Merawat Sistem (SKPL_02_01), merupakan fungsi yang menangani pengenalan dan perbaikan atas masalah-masalah yang sering terjadi dalam sistem.

b. Menjadwalkan Pekerjaan Penyelamatan (SKPL_02_02), merupakan fungsi yang menangani penjadwalan pengerjaan setiap pekerjaan penyelamatan dalam Penyelamat.

3. Penyelamat (SKPL_03), merupakan fungsi yang mengerjakan setiap pekerjaan penyelamatan yang diberikan Penjadwal.

Fungsi-fungsi yang ada dalam SKPL_03 ini adalah:

- a. Mengendalikan Penyelamatan (SKPL_03_01), merupakan fungsi yang mengendalikan keseluruhan proses penyelamatan.
- b. Meng-hook Windows API (SKPL_03_02), merupakan fungsi yang meng-hook fungsi-fungsi Windows API yang berkaitan dengan penyelamatan dokumen.
- c. Memantau Panggilan Windows API (SKPL_03_03), merupakan fungsi yang memantau panggilan-panggilan Windows API dan mencoba menyelamatkan dokumen.

2.3. Karakteristik Pengguna

Karakteristik pengguna perangkat lunak PANDORA adalah:

1. anonim
 - a. memahami *malware* dalam sistem operasi Windows,
 - b. mampu mengoperasikan komputer secara umum, dan
 - c. mampu mengoperasikan Internet dan *web browser*.
2. administrator
 - a. memahami *malware* dalam sistem operasi Windows,
 - b. memahami cara kerja sistem,
 - c. memahami lingkungan tempat sistem bekerja,
 - d. mampu mengoperasikan komputer secara umum,
 - e. mampu mengoperasikan Internet dan *web browser*, dan
 - f. mampu merawat sistem dan lingkungannya.

2.4. Batasan-Batasan

Batasan dalam pengembangan perangkat lunak PANDORA adalah:

1. kebijaksanaan umum

berpedoman pada tujuan pengembangan perangkat lunak PANDORA, dan

2. keterbatasan perangkat keras

dapat diketahui setelah sistem berjalan (sesuai kebutuhan).

2.5. Asumsi dan Ketergantungan

Asumsi yang digunakan dalam pengembangan perangkat lunak PANDORA adalah:

tersedianya semua perangkat keras dan perangkat lunak yang diperlukan untuk mengoperasikan perangkat lunak PANDORA.

3. Kebutuhan Khusus

3.1. Kebutuhan Antarmuka Eksternal

3.1.1. Antarmuka Pemakai

Kebutuhan antarmuka pemakai yang dibutuhkan dalam pengembangan perangkat lunak PANDORA adalah:

1. hanya menampilkan informasi yang relevan dengan konteks yang sedang dilakukan,
2. menggunakan istilah-istilah, warna-warna, dan simbol-simbol baku yang mudah dimengerti,
3. menampilkan pesan kesalahan jika terjadi kesalahan (*include* dalam setiap modul fungsional),
4. menggunakan *upper-* dan *lowercase*, indentasi, serta pengelompokan teks untuk mempermudah pengertian,
5. meminimalkan jumlah masukan yang harus dilakukan pengguna sistem, dan
6. mengelola konsistensi antara tampilan informasi dan masukan data.

3.1.2. Antarmuka Perangkat Keras

Kebutuhan antarmuka perangkat keras yang dikembangkan PANDORA berkenaan dengan antarmuka perangkat keras pada bagian Perspektif Produk adalah:

1. *mouse*, digunakan untuk mengenali masukan data yang dilakukan pengguna sistem berkaitan dengan *event click, drag and drop*, dan *on focus*,
2. *keyboard*, digunakan untuk melakukan masukan data berupa karakter atau teks atau menu *pull down* yang harus dimasukkan pengguna sistem, dan
3. *modem* atau *LAN card*, digunakan untuk melakukan transfer data yang dikirimkan pengguna sistem dan

sekaligus untuk menerima informasi yang diberikan sistem berkaitan dengan interaksi sistem dengan pengguna sistem.

3.1.3. Antarmuka Perangkat Lunak

Antarmuka perangkat lunak *server-side* mengacu pada bagian Perspektif Produk, sedangkan untuk perangkat lunak *client-side* digunakan perangkat lunak pendukung berikut:

1. nama : Windows,
nomor versi : XP Professional Service Pack 2,
sumber : Microsoft,
sebagai sistem operasi tempat *web browser* berjalan,
dan
2. nama : Internet Explorer,
nomor versi : 6,
sumber : Microsoft,
sebagai *web browser* untuk mengakses antarmuka web PANDORA.

3.1.4. Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak PANDORA ini menggunakan protokol TCP/IP (mengacu pada bagian Perspektif Produk), karena perangkat lunak ini digunakan untuk mendukung model sistem *client-server* dengan media komunikasi *intranet* atau Internet.

3.2. Kebutuhan Fungsionalitas

3.2.1. Aliran Informasi

3.2.1.1. DFD Level 0 (Diagram Konteks) PANDORA

3.2.1.1.1. Entitas Data

Entitas eksternal data yang terlibat dalam pengembangan perangkat lunak PANDORA dapat dinyatakan dalam tabel berikut:

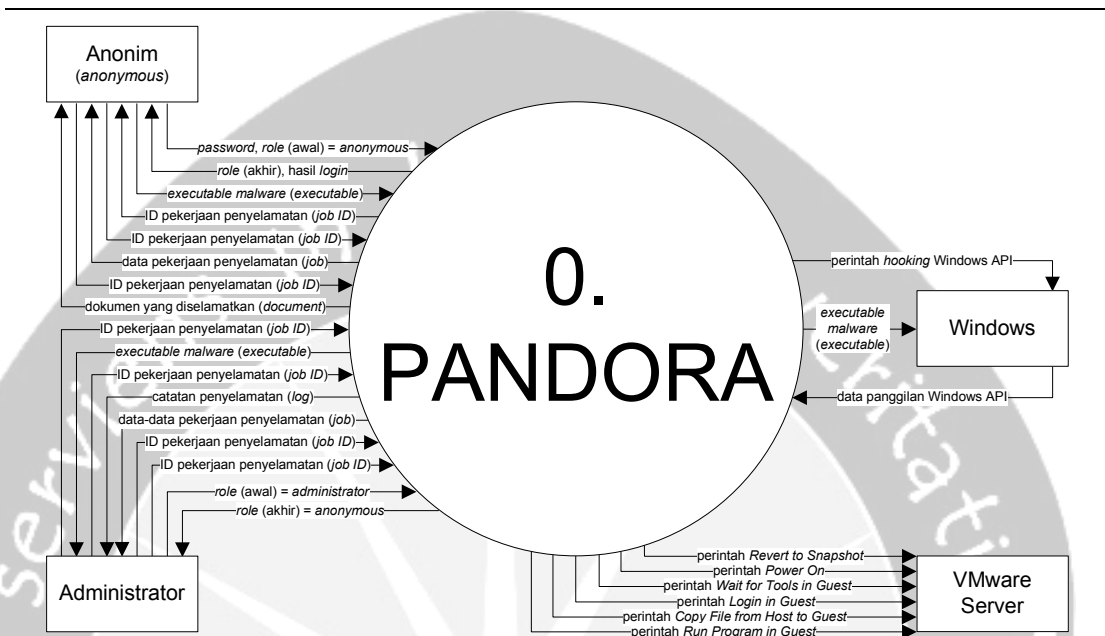
Nama	Kode
anonim	ANONIM
administrator	ADMINISTRATOR
VMware Server	VMWARE SERVER
Windows	WINDOWS

3.2.1.1.2. Proses

Proses dalam perangkat lunak PANDORA adalah menerima masukan data dari pengguna sistem kemudian memprosesnya menjadi informasi yang dikehendaki.

3.2.1.1.3. Topologi

Topologi proses dari perangkat lunak PANDORA dapat digambarkan sebagai berikut:



Gambar 3.1. DFD Level 0 PANDORA

3.2.1.2. DFD Level 1 Proses PANDORA

3.2.1.2.1. Entitas Data

Mengacu pada entitas data DFD Level 0 (Diagram Konteks) PANDORA.

3.2.1.2.2. Proses

Proses dalam DFD Level 1 proses PANDORA adalah:

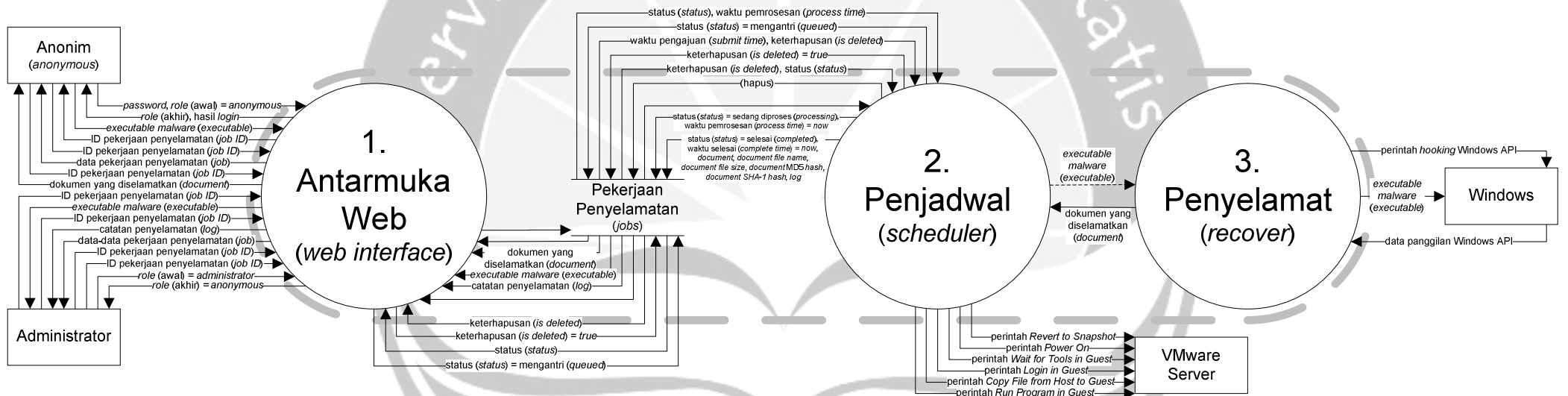
1. Antarmuka Web (*web interface*), merupakan proses yang menyajikan antarmuka web agar para pengguna sistem dapat berinteraksi dengan sistem,
2. Penjadwal (*scheduler*), merupakan proses yang menangani perawatan sistem dan penjadwalan pengerjaan setiap pekerjaan penyelamatan dalam Penyelamat, dan

3. Penyelamat (*recover*), merupakan proses yang mengerjakan setiap pekerjaan penyelamatan yang diberikan Penjadwal.



3.2.1.2.3. Topologi

Topologi DFD *Level 1* proses PANDORA dapat digambarkan sebagai berikut:



Gambar 3.2. DFD *Level 1* PANDORA

3.2.1.3. DFD Level 2 Proses Antarmuka Web

3.2.1.3.1. Entitas Data

Entitas data yang terlibat dalam proses Antarmuka Web PANDORA adalah:

1. anonim (*anonymous*), dan
2. administrator.

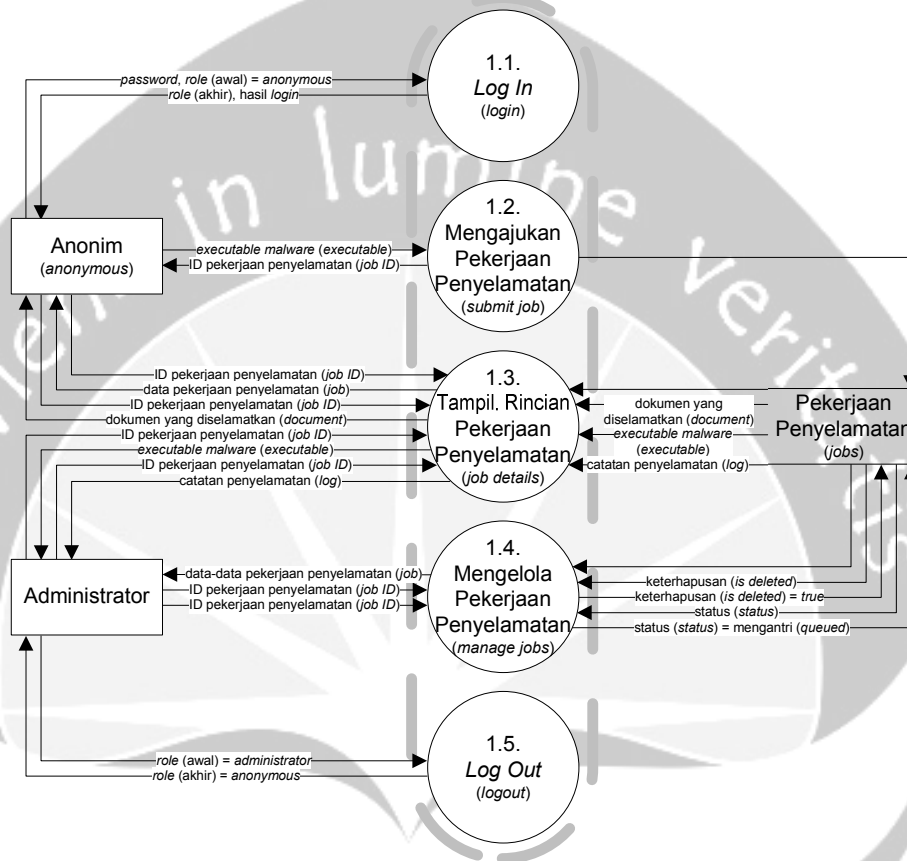
3.2.1.3.2. Proses

Secara garis besar, proses yang dikelola pada proses Antarmuka Web PANDORA adalah:

1. *Log In (login)*, merupakan proses yang menangani otentikasi administrator ke dalam sistem melalui *password*,
2. Mengajukan Pekerjaan Penyelamatan (*submit job*), merupakan proses yang menangani pengajuan dan penyimpanan pekerjaan penyelamatan,
3. Menampilkan Rincian Pekerjaan Penyelamatan (*job details*), merupakan proses yang menangani penampilan berbagai informasi tentang sebuah pekerjaan penyelamatan,
4. Mengelola Pekerjaan Penyelamatan (*manage jobs*), merupakan proses yang menangani pengelolaan pekerjaan-pekerjaan penyelamatan, dan
5. *Log Out (logout)*, merupakan proses yang menangani deotentikasi administrator dari dalam sistem.

3.2.1.3.3. Topologi

Topologi DFD *Level 2* proses Antarmuka Web dapat digambarkan sebagai berikut:



Gambar 3.3. DFD *Level 2* Antarmuka Web

3.2.1.4. DFD *Level 2* Proses Penjadwal

3.2.1.4.1. Entitas Data

Entitas data yang terlibat dalam proses Penjadwal PANDORA adalah:

1. VMware Server.

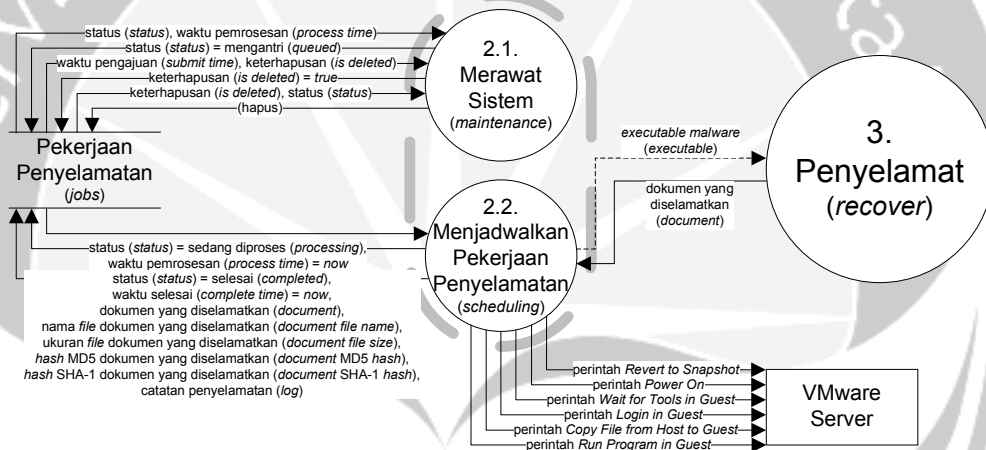
3.2.1.4.2. Proses

Secara garis besar, proses yang dikelola pada proses Penjadwal PANDORA adalah:

1. Merawat Sistem (*maintenance*), merupakan proses yang menangani pengenalan dan perbaikan atas masalah-masalah yang sering terjadi dalam sistem, dan
2. Menjadwalkan Pekerjaan Penyelamatan (*scheduling*), merupakan proses yang menangani penjadwalan pengerjaan setiap pekerjaan penyelamatan dalam Penyelamat.

3.2.1.4.3. Topologi

Topologi DFD Level 2 proses Penjadwal dapat digambarkan sebagai berikut:



Gambar 3.4. DFD Level 2 Penjadwal

3.2.1.5. DFD Level 2 Proses Penyelamat

3.2.1.5.1. Entitas Data

Entitas data yang terlibat dalam proses Penyelamat PANDORA adalah:

1. Windows.

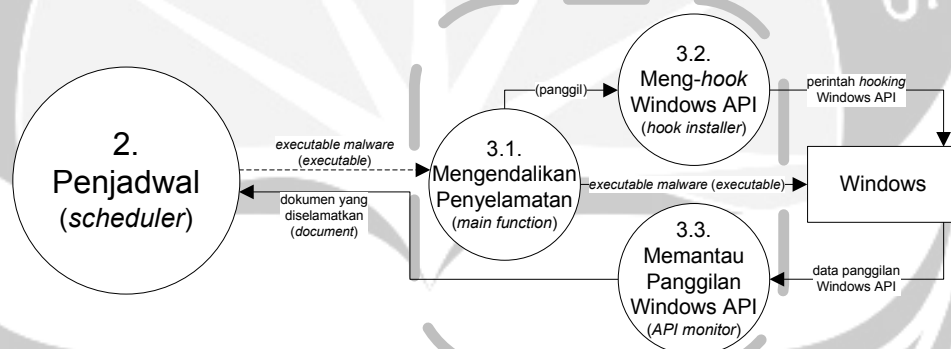
3.2.1.5.2. Proses

Secara garis besar, proses yang dikelola pada proses Penyelamat PANDORA adalah:

1. Mengendalikan Penyelamatan (*main function*), merupakan proses yang mengendalikan keseluruhan proses penyelamatan,
2. Meng-hook Windows API (*hook installer*), merupakan proses yang meng-hook fungsi-fungsi Windows API yang berkaitan dengan penyelamatan dokumen, dan
3. Memantau Panggilan Windows API (*API monitor*), merupakan proses yang memantau panggilan-panggilan Windows API dan mencoba menyelamatkan dokumen.

3.2.1.5.3. Topologi

Topologi DFD *Level 2* proses Penyelamat dapat digambarkan sebagai berikut:



Gambar 3.5. DFD *Level 2* Penyelamat

3.2.1.6. DFD *Level 3* Proses Menampilkan Rincian Pekerjaan Penyelamatan

3.2.1.6.1. Entitas Data

Entitas data yang terlibat dalam proses Menampilkan Rincian Pekerjaan Penyelamatan PANDORA adalah:

1. anonim (*anonymous*), dan
2. administrator.

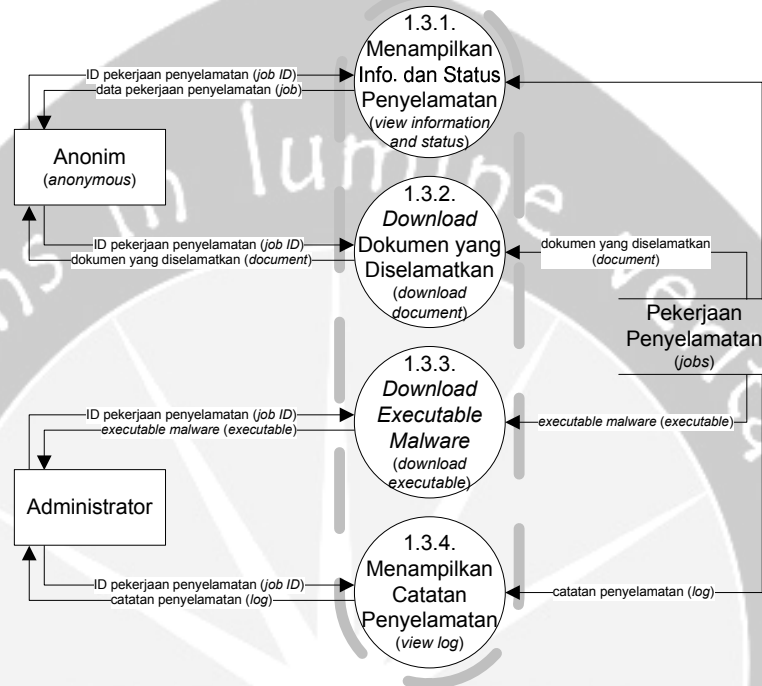
3.2.1.6.2. Proses

Secara garis besar, proses yang dikelola pada proses Menampilkan Rincian Pekerjaan Penyelamatan PANDORA adalah:

1. Menampilkan Informasi dan Status Penyelamatan (*view information and status*), merupakan proses yang menangani penampilan informasi dan status penyelamatan sebuah pekerjaan penyelamatan,
2. *Download* Dokumen yang Diselamatkan (*download document*), merupakan proses yang menangani *download* dokumen yang diselamatkan dari sebuah pekerjaan penyelamatan,
3. *Download Executable Malware* (*download executable*), merupakan proses yang menangani *download executable malware* sebuah pekerjaan penyelamatan, dan
4. Menampilkan Catatan Penyelamatan (*view log*), merupakan proses yang menangani penampilan catatan penyelamatan sebuah pekerjaan penyelamatan.

3.2.1.6.3. Topologi

Topologi DFD *Level 3* proses Menampilkan Rincian Pekerjaan Penyelamatan dapat digambarkan sebagai berikut:



Gambar 3.6. DFD *Level 3* Menampilkan Rincian Pekerjaan Penyelamatan

3.2.1.7. DFD *Level 3* Proses Mengelola Pekerjaan Penyelamatan

3.2.1.7.1. Entitas Data

Entitas data yang terlibat dalam proses Mengelola Pekerjaan Penyelamatan PANDORA adalah:

1. administrator.

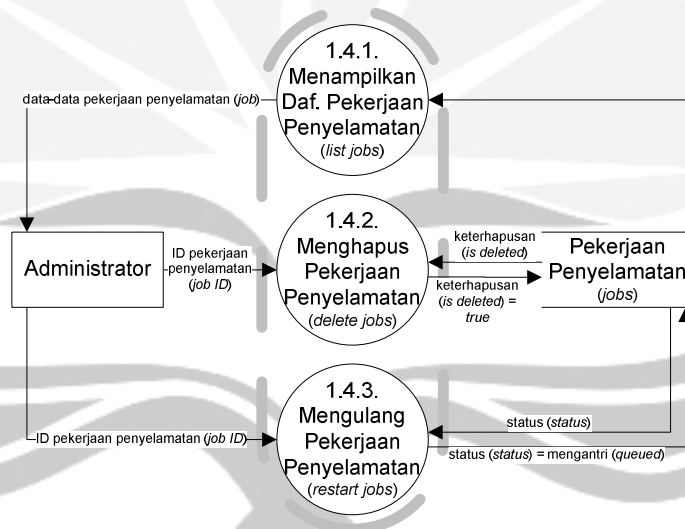
3.2.1.7.2. Proses

Secara garis besar, proses yang dikelola pada proses Mengelola Pekerjaan Penyelamatan PANDORA adalah:

1. Menampilkan Daftar Pekerjaan Penyelamatan (*list jobs*), merupakan proses yang menangani penampilan daftar pekerjaan penyelamatan,
2. Menghapus Pekerjaan Penyelamatan (*delete jobs*), merupakan proses yang menangani penandaan pekerjaan penyelamatan untuk dihapus, dan
3. Mengulang Pekerjaan Penyelamatan (*restart jobs*), merupakan proses yang menangani pengulangan pemrosesan pekerjaan penyelamatan.

3.2.1.7.3. Topologi

Topologi DFD *Level 3* proses Mengelola Pekerjaan Penyelamatan dapat digambarkan sebagai berikut:



Gambar 3.7. DFD *Level 3* Mengelola Pekerjaan Penyelamatan

3.2.2. Deskripsi Proses

3.2.2.1. Proses Log In

3.2.2.1.1. Entitas Data Masukan

Entitas data masukan proses *Log In* adalah:

1. *password*, dan
2. *role*.

3.2.2.1.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses *Log In* adalah:

1. menerima masukan berupa *password* dan *role*,
2. membandingkan *password* dengan *password* administrator: jika sama, mengatur *role* menjadi "administrator" dan hasil *login* menjadi "Login Berhasil"; sebaliknya, mengatur hasil *login* menjadi "Login Gagal", dan
3. mengembalikan keluaran berupa *role* dan hasil *login*.

3.2.2.1.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses *Log In* adalah:

1. *password*, dan
2. *role*.

Entitas data keluaran dari proses ini adalah:

1. *role*, dan
2. hasil *login*.

3.2.2.2. Proses Mengajukan Pekerjaan Penyelamatan

3.2.2.2.1. Entitas Data Masukan

Entitas data masukan proses Mengajukan Pekerjaan Penyelamatan adalah:

1. *executable malware (executable)*.

3.2.2.2.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Mengajukan Pekerjaan Penyelamatan adalah:

1. menerima masukan berupa *executable malware* yang diajukan (*executable*),

2. membangkitkan ID pekerjaan penyelamatan (*job ID*) secara acak,
3. mencatat nama *file* (*executable file name*) dan ukuran *file* (*executable file size*) *executable malware* (*executable*),
4. menghitung *hash* MD5 (*executable MD5 hash*) dan *hash* SHA-1 (*executable SHA-1 hash*) *executable malware* (*executable*),
5. mencatat alamat IP pengakses sebagai alamat IP pengaju (*submitter IP*),
6. mencatat waktu sekarang sebagai waktu pengajuan (*submit time*),
7. menyimpan semua data di atas sebagai data pekerjaan penyelamatan (*job*) dalam *data store* Pekerjaan Penyelamatan (*jobs*), dan
8. mengembalikan keluaran berupa ID pekerjaan penyelamatan yang dibangkitkan (*job ID*).

3.2.2.2.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Mengajukan Pekerjaan Penyelamatan adalah:

1. *executable malware* (*executable*),
2. *data store* Pekerjaan Penyelamatan (*jobs*).

Entitas data keluaran dari proses ini adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.3. Menampilkan Informasi dan Status Penyelamatan

3.2.2.3.1. Entitas Data Masukan

Entitas data masukan proses Menampilkan Informasi dan Status Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.3.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Menampilkan Informasi dan Status Penyelamatan adalah:

1. menerima masukan berupa ID pekerjaan penyelamatan yang akan ditampilkan informasi dan status penyelamatannya (*job ID*),
2. membaca data pekerjaan penyelamatan (*job*) dengan ID (*job ID*) tersebut dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. mengembalikan keluaran berupa data pekerjaan penyelamatan yang dibaca (*job*).

3.2.2.3.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Menampilkan Informasi dan Status Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*),
2. data pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Entitas data keluaran dari proses ini adalah:

1. data pekerjaan penyelamatan (*job*).

3.2.2.4. Proses *Download* Dokumen yang Diselamatkan

3.2.2.4.1. Entitas Data Masukan

Entitas data masukan proses *Download* Dokumen yang Diselamatkan adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.4.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses *Download* Dokumen yang Diselamatkan adalah:

1. menerima masukan berupa ID pekerjaan penyelamatan yang akan di-*download* dokumennya (*job ID*),
2. membaca dokumen yang diselamatkan (*document*) milik pekerjaan penyelamatan (*job*) dengan ID (*job ID*) tersebut dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. mengembalikan keluaran berupa dokumen yang dibaca (*document*).

3.2.2.4.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses *Download* Dokumen yang Diselamatkan adalah:

1. ID pekerjaan penyelamatan (*job ID*),
2. dokumen yang diselamatkan (*document*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Entitas data keluaran dari proses ini adalah:

1. dokumen yang diselamatkan (*document*).

3.2.2.5. Proses *Download Executable Malware*

3.2.2.5.1. Entitas Data Masukan

Entitas data masukan proses *Download Executable Malware* adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.5.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses *Download Executable Malware* adalah:

1. menerima masukan berupa ID pekerjaan penyelamatan yang akan di-*download executable malware*-nya(*job ID*),
2. membaca *executable malware* (*executable*) milik pekerjaan penyelamatan (*job*) dengan ID (*job ID*)

- tersebut dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. mengembalikan keluaran berupa *executable malware* yang dibaca (*executable*).

3.2.2.5.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses *Download Executable Malware* adalah:

1. ID pekerjaan penyelamatan (*job ID*),
2. *executable malware* (*executable*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Entitas data keluaran dari proses ini adalah:

1. *executable malware* (*executable*).

3.2.2.6. Proses Menampilkan Catatan Penyelamatan

3.2.2.6.1. Entitas Data Masukan

Entitas data masukan proses Menampilkan Catatan Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.6.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Menampilkan Catatan Penyelamatan adalah:

1. menerima masukan berupa ID pekerjaan penyelamatan yang akan ditampilkan catatan penyelamatannya (*job ID*),
2. membaca catatan penyelamatan (*log*) milik pekerjaan penyelamatan (*job*) dengan ID (*job ID*) tersebut dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. mengembalikan keluaran berupa catatan penyelamatan yang dibaca (*log*).

3.2.2.6.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Menampilkan Catatan Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*),
2. catatan penyelamatan (*log*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Entitas data keluaran dari proses ini adalah:

1. catatan penyelamatan (*log*).

3.2.2.7. Proses Menampilkan Daftar Pekerjaan Penyelamatan

3.2.2.7.1. Entitas Data Masukan

Proses Menampilkan Daftar Pekerjaan Penyelamatan tidak memiliki entitas data masukan.

3.2.2.7.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Menampilkan Daftar Pekerjaan Penyelamatan adalah:

1. membaca data-data pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
2. mengembalikan keluaran berupa data-data pekerjaan penyelamatan yang dibaca (*job*).

3.2.2.7.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Menampilkan Daftar Pekerjaan Penyelamatan adalah:

1. data pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
2. *data store* Pekerjaan Penyelamatan (*jobs*).

Entitas data keluaran dari proses ini adalah:

1. data-data pekerjaan penyelamatan (*job*).

3.2.2.8. Proses Menghapus Pekerjaan Penyelamatan

3.2.2.8.1. Entitas Data Masukan

Entitas data masukan proses Menghapus Pekerjaan Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.8.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Menghapus Pekerjaan Penyelamatan adalah:

1. menerima masukan berupa ID pekerjaan penyelamatan yang akan dihapus (*job ID*), dan
2. memperbarui keterhapusan (*is deleted*) milik pekerjaan penyelamatan (*job*) dengan ID (*job ID*) tersebut dalam *data store* Pekerjaan Penyelamatan (*jobs*) menjadi *true*.

3.2.2.8.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Menghapus Pekerjaan Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*),
2. keterhapusan (*is deleted*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Proses ini tidak memiliki entitas data keluaran.

3.2.2.9. Proses Mengulang Pekerjaan Penyelamatan

3.2.2.9.1. Entitas Data Masukan

Entitas data masukan proses Mengulang Pekerjaan Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*).

3.2.2.9.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Mengulang Pekerjaan Penyelamatan adalah:

1. menerima masukan berupa ID pekerjaan penyelamatan yang akan diulang pemrosesannya (*job ID*), dan
2. memperbarui status (*status*) milik pekerjaan penyelamatan (*job*) dengan ID (*job ID*) tersebut dalam *data store* Pekerjaan Penyelamatan (*jobs*) menjadi "mengantri" (*queued*).

3.2.2.9.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Mengulang Pekerjaan Penyelamatan adalah:

1. ID pekerjaan penyelamatan (*job ID*),
2. status (*status*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Proses ini tidak memiliki entitas data keluaran.

3.2.2.10. Proses Log Out

3.2.2.10.1. Entitas Data Masukan

Entitas data masukan proses *Log Out* adalah:

1. *role*.

3.2.2.10.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses *Log Out* adalah:

1. menerima masukan berupa *role*,
2. mengatur *role* menjadi "*anonymous*", dan
3. mengembalikan keluaran berupa *role* yang telah diatur.

3.2.2.10.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses *Log Out* adalah:

1. *role*.

Entitas data keluaran dari proses ini adalah:

1. *role*.

3.2.2.11. Proses Merawat Sistem

3.2.2.11.1. Entitas Data Masukan

Proses Merawat Sistem tidak memiliki entitas data masukan.

3.2.2.11.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Merawat Sistem adalah:

1. mengulang pemrosesan pekerjaan-pekerjaan penyelamatan yang macet:
 - a. membaca status (*status*) dan waktu pemrosesan (*process time*) milik setiap pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
 - b. jika suatu pekerjaan penyelamatan sedang diproses (*status = processing*) tetapi waktu pemrosesannya (*process time*) telah jauh terlampaui, memperbarui status (*status*) pekerjaan penyelamatan (*job*) tersebut menjadi "mengantri" (*queued*),
2. menandai pekerjaan-pekerjaan penyelamatan yang telah kadaluwarsa untuk dihapus:
 - a. membaca waktu pengajuan (*submit time*) dan keterhapusan (*is deleted*) milik setiap pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
 - b. jika suatu pekerjaan penyelamatan waktu pengajuannya telah melebihi satu tahun (*now -*

submit time > 1 year), memperbarui keterhapusan (*is deleted*) pekerjaan penyelamatan (*job*) tersebut menjadi *true*,

3. menghapus pekerjaan-pekerjaan penyelamatan yang telah ditandai untuk dihapus:
 - a. membaca keterhapusan (*is deleted*) dan status (*status*) milik setiap pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
 - b. jika suatu pekerjaan penyelamatan telah ditandai untuk dihapus (*is deleted = true*) dan tidak sedang diproses (*status ≠ processing*), menghapus pekerjaan penyelamatan (*job*) tersebut.

3.2.2.11.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Merawat Sistem adalah:

1. status (*status*) dari *data store* Pekerjaan Penyelamatan (*jobs*),
2. waktu pemrosesan (*process time*) dari *data store* Pekerjaan Penyelamatan (*jobs*),
3. waktu pengajuan (*submit time*) dari *data store* Pekerjaan Penyelamatan (*jobs*),
4. keterhapusan (*is deleted*) dari *data store* Pekerjaan Penyelamatan (*jobs*), dan
5. *data store* Pekerjaan Penyelamatan (*jobs*).

Proses ini tidak memiliki entitas data keluaran.

3.2.2.12. Proses Menjadwalkan Pekerjaan Penyelamatan

3.2.2.12.1. Entitas Data Masukan

Proses Menjadwalkan Pekerjaan Penyelamatan tidak memiliki entitas data masukan.

3.2.2.12.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Menjadwalkan Pekerjaan Penyelamatan adalah:

1. membaca sebuah data pekerjaan penyelamatan (*job*) yang mengantri (*status = queued*) dari *data store* Pekerjaan Penyelamatan (*jobs*),
2. memperbarui status (*status*) pekerjaan penyelamatan (*job*) tersebut menjadi "sedang diproses" (*processing*) dan waktu pemrosesannya (*process time*) menjadi sekarang,
3. memberikan perintah *Revert to Snapshot* kepada VMware Server untuk mengembalikan keadaan *virtual machine* ke *snapshot*,
4. memberikan perintah *Power On* kepada VMware Server untuk menghidupkan *virtual machine*,
5. memberikan perintah *Wait for Tools in Guest* kepada VMware Server untuk menunggu hingga sistem operasi *guest* siap digunakan,
6. memberikan perintah *Login in Guest* kepada VMware Server untuk melakukan *login* ke sistem operasi *guest*,
7. memberikan perintah *Copy File from Host to Guest* kepada VMware Server untuk menyalin Penyelamat (*recover*) dan *executable malware* (*executable*) pekerjaan penyelamatan (*job*) tersebut ke dalam sistem operasi *guest*,
8. memberikan perintah *Run Program in Guest* kepada VMware Server untuk menjalankan Penyelamat (*recover*) dalam sistem operasi *guest*,

CATATAN: kedua langkah di atas praktis memberikan *executable malware* (*executable*) kepada Penyelamat (*recover*) untuk dikerjakan.

9. menerima dokumen yang diselamatkan (*document*) dari Penyelamat (*recover*),

10. memberikan perintah *Revert to Snapshot* kepada VMware Server untuk mengembalikan keadaan *virtual machine* ke *snapshot*,
11. mencatat nama *file* (*document file name*) dan ukuran *file* (*document file size*) dokumen yang diselamatkan (*document*),
12. menghitung *hash* MD5 (*document MD5 hash*) dan *hash* SHA-1 (*document SHA-1 hash*) dokumen yang diselamatkan (*document*),
13. menulis catatan penyelamatan (*log*), yang berisi rincian kejadian-kejadian yang terjadi selama pemrosesan pekerjaan penyelamatan, dan
14. memperbarui status (*status*) pekerjaan penyelamatan (*job*) tersebut menjadi "selesai" (*completed*), waktu selesainya (*complete time*) menjadi sekarang, dan dokumen yang diselamatkan (*document*), berikut nama *file* (*document file name*), ukuran *file* (*document file size*), *hash* MD5 (*document MD5 hash*), dan *hash* SHA-1-nya (*document SHA-1 hash*), serta catatan penyelamatannya (*log*) sesuai hasil-hasil perhitungan di atas.

3.2.2.12.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Menjadwalkan Pekerjaan Penyelamatan adalah:

1. data pekerjaan penyelamatan (*job*) dari *data store* Pekerjaan Penyelamatan (*jobs*),
2. dokumen yang diselamatkan (*document*) dari proses Penyelamat (*recover*), dan
3. *data store* Pekerjaan Penyelamatan (*jobs*).

Proses ini tidak memiliki entitas data keluaran.

3.2.2.13. Proses Mengendalikan Penyelamatan

3.2.2.13.1. Entitas Data Masukan

Entitas data masukan proses Mengendalikan Penyelamatan adalah:

1. *executable malware (executable)*.

3.2.2.13.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Mengendalikan Penyelamatan adalah:

1. menerima masukan berupa *executable malware (executable)*,
2. memanggil proses Meng-hook Windows API (*hook installer*) untuk meng-hook fungsi-fungsi Windows API yang berkaitan dengan penyelamatan dokumen, dan
3. memberikan *executable malware (executable)* kepada Windows untuk dijalankan.

3.2.2.13.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Mengendalikan Penyelamatan adalah:

1. *executable malware (executable)*.

Proses ini tidak memiliki entitas data keluaran.

3.2.2.14. Proses Meng-hook Windows API

3.2.2.14.1. Entitas Data Masukan

Proses Meng-hook Windows API tidak memiliki entitas data masukan.

3.2.2.14.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Meng-hook Windows API adalah memerintahkan Windows untuk meng-hook fungsi-

fungsi Windows API yang berkaitan dengan penyelamatan dokumen, sehingga semua panggilan terhadap fungsi-fungsi tersebut akan melewati proses Memantau Panggilan Windows API (*API monitor*) terlebih dahulu sebelum dijalankan.

3.2.2.14.3. Entitas Data Terlibat

Tidak ada entitas data yang terlibat proses Meng-hook Windows API. Selain itu, proses ini juga tidak memiliki entitas data keluaran.

3.2.2.15. Proses Memantau Panggilan Windows API

3.2.2.15.1. Entitas Data Masukan

Entitas data masukan proses Memantau Panggilan Windows API adalah:

1. data panggilan Windows API.

3.2.2.15.2. Algoritma atau Formula dari Proses

Algoritma atau formula dari proses Memantau Panggilan Windows API adalah:

1. menerima masukan berupa data panggilan Windows API, dan
2. jika ada dokumen yang dapat diselamatkan dari data panggilan Windows API tersebut, menyelamatkan dokumen tersebut dan memberikan dokumen yang diselamatkan (*document*) kepada Penjadwal (*scheduler*).

3.2.2.15.3. Entitas Data Terlibat

Entitas data yang terlibat pada proses Memantau Panggilan Windows API adalah:

1. data panggilan Windows API.

Entitas data keluaran dari proses ini adalah:

1. dokumen yang diselamatkan (*document*).

3.2.3. Spesifikasi Data

3.2.3.1. Konstruksi Data Pekerjaan Penyelamatan

3.2.3.1.1. Tipe Record

Tipe *record* dalam konstruksi data Pekerjaan Penyelamatan terdiri atas:

1. data pekerjaan penyelamatan (*job*).

3.2.3.1.2. Field-Field

Field-field dalam masing-masing tipe *record* dalam konstruksi data Pekerjaan Penyelamatan adalah:

1. data pekerjaan penyelamatan:

- a. ID pekerjaan penyelamatan (*job ID*),
- b. nama *file executable malware* (*executable file name*),
- c. ukuran *file executable malware* (*executable file size*),
- d. *hash MD5 executable malware* (*executable MD5 hash*),
- e. *hash SHA-1 executable malware* (*executable SHA-1 hash*),
- f. nama *file* dokumen yang diselamatkan (*document file name*),
- g. ukuran *file* dokumen yang diselamatkan (*document file size*),
- h. *hash MD5* dokumen yang diselamatkan (*document MD5 hash*),
- i. *hash SHA-1* dokumen yang diselamatkan (*document SHA-1 hash*),
- j. status (*status*),
- k. alamat IP pengaju (*submitter IP*),

- l. waktu pengajuan (*submit time*),
- m. waktu pemrosesan (*process time*),
- n. waktu selesai (*complete time*),
- o. *executable malware* (*executable*),
- p. dokumen yang diselamatkan (*document*),
- q. catatan penyelamatan (*log*), dan
- r. keterhapusan (*is deleted*).

3.2.4. Kamus Data

3.2.4.1. Elemen Data ID Pekerjaan Penyelamatan

3.2.4.1.1. Nama

Nama elemen data: *job ID* (KD_SKPL_01).

3.2.4.1.2. Representasi

Representasi dalam tipe karakter dengan panjang 32 karakter.

3.2.4.1.3. Unit atau Format

Format sembarang.

3.2.4.1.4. Presisi atau Keakuratan

Presisi atau keakuratan: tidak *case sensitive*.

3.2.4.1.5. Range

Range: angka heksadesimal ('0' - '9' dan 'a' - 'f').

3.2.4.2. Elemen Data Nama *File Executable Malware*

3.2.4.2.1. Nama

Nama elemen data: *executable file name* (KD_SKPL_02).

3.2.4.2.2. Representasi

Representasi dalam tipe karakter dengan panjang maksimum 255 karakter.

3.2.4.2.3. Unit atau Format

Format: "namafile.ekstensi".

3.2.4.2.4. Presisi atau Keakuratan

Presisi atau keakuratan: *case sensitive*.

3.2.4.2.5. Range

Range: semua karakter.

3.2.4.3. Elemen Data Ukuran File Executable Malware

3.2.4.3.1. Nama

Nama elemen data: *executable file size* (KD_SKPL_03).

3.2.4.3.2. Representasi

Representasi dalam tipe bilangan bulat.

3.2.4.3.3. Unit atau Format

Unit: byte, format sembarang.

3.2.4.3.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga satu byte.

3.2.4.3.5. Range

Range: 1 hingga $2^{31} - 1$ byte (2×10^9 byte).

3.2.4.4. Elemen Data Hash MD5 Executable Malware

3.2.4.4.1. Nama

Nama elemen data: *executable MD5 hash* (KD_SKPL_04).

3.2.4.4.2. Representasi

Representasi dalam tipe karakter dengan panjang 32 karakter.

3.2.4.4.3. Unit atau Format

Format sembarang.

3.2.4.4.4. Presisi atau Keakuratan

Presisi atau keakuratan: tidak *case sensitive*.

3.2.4.4.5. Range

Range: angka heksadesimal ('0' - '9' dan 'a' - 'f').

3.2.4.5. Elemen Data Hash SHA-1 Executable Malware

3.2.4.5.1. Nama

Nama elemen data: *executable SHA-1 hash* (KD_SKPL_05).

3.2.4.5.2. Representasi

Representasi dalam tipe karakter dengan panjang 40 karakter.

3.2.4.5.3. Unit atau Format

Format sembarang.

3.2.4.5.4. Presisi atau Keakuratan

Presisi atau keakuratan: tidak *case sensitive*.

3.2.4.5.5. Range

Range: angka heksadesimal ('0' - '9' dan 'a' - 'f').

3.2.4.6. Elemen Data Nama *File* Dokumen yang Diselamatkan

3.2.4.6.1. Nama

Nama elemen data: *document file name* (KD_SKPL_06).

3.2.4.6.2. Representasi

Representasi dalam tipe karakter dengan panjang maksimum 255 karakter.

3.2.4.6.3. Unit atau Format

Format: "namafile.ekstensi".

3.2.4.6.4. Presisi atau Keakuratan

Presisi atau keakuratan: *case sensitive*.

3.2.4.6.5. Range

Range: semua karakter.

3.2.4.7. Elemen Data Ukuran *File* Dokumen yang Diselamatkan

3.2.4.7.1. Nama

Nama elemen data: *document file size* (KD_SKPL_07).

3.2.4.7.2. Representasi

Representasi dalam tipe bilangan bulat.

3.2.4.7.3. Unit atau Format

Unit: byte, format sembarang.

3.2.4.7.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga satu byte.

3.2.4.7.5. Range

Range: 1 hingga $2^{31} - 1$ byte (2×10^9 byte).

3.2.4.8. Elemen Data Hash MD5 Dokumen yang Diselamatkan

3.2.4.8.1. Nama

Nama elemen data: *document MD5 hash* (KD_SKPL_08).

3.2.4.8.2. Representasi

Representasi dalam tipe karakter dengan panjang 32 karakter.

3.2.4.8.3. Unit atau Format

Format sembarang.

3.2.4.8.4. Presisi atau Keakuratan

Presisi atau keakuratan: tidak *case sensitive*.

3.2.4.8.5. Range

Range: angka heksadesimal ('0' - '9' dan 'a' - 'f').

3.2.4.9. Elemen Data Hash SHA-1 Dokumen yang Diselamatkan

3.2.4.9.1. Nama

Nama elemen data: *document SHA-1 hash* (KD_SKPL_09).

3.2.4.9.2. Representasi

Representasi dalam tipe karakter dengan panjang 40 karakter.

3.2.4.9.3. Unit atau Format

Format sembarang.

3.2.4.9.4. Presisi atau Keakuratan

Presisi atau keakuratan: tidak *case sensitive*.

3.2.4.9.5. Range

Range: angka heksadesimal ('0' - '9' dan 'a' - 'f').

3.2.4.10. Elemen Data Status

3.2.4.10.1. Nama

Nama elemen data: *status* (KD_SKPL_10).

3.2.4.10.2. Representasi

Representasi dalam tipe enumerasi dengan 4 elemen.

3.2.4.10.3. Unit atau Format

Format sembarang.

3.2.4.10.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga setiap elemen.

3.2.4.10.5. Range

Range: "mengantri" (*queued*), "sedang diproses" (*processing*), "selesai" (*completed*), dan "ada kesalahan" (*error*).

3.2.4.11. Elemen Data Alamat IP Pengaju

3.2.4.11.1. Nama

Nama elemen data: *submitter IP* (KD_SKPL_11).

3.2.4.11.2. Representasi

Representasi dalam tipe karakter dengan panjang maksimum 15 karakter.

3.2.4.11.3. Unit atau Format

Format: "999.999.999.999".

3.2.4.11.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga satu *host*.

3.2.4.11.5. Range

Range: angka desimal ('0' - '9') dan titik ('.').

3.2.4.12. Elemen Data Waktu Pengajuan

3.2.4.12.1. Nama

Nama elemen data: *submit time* (KD_SKPL_12).

3.2.4.12.2. Representasi

Representasi dalam tipe *timestamp*.

3.2.4.12.3. Unit atau Format

Format sembarang.

3.2.4.12.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga satu detik.

3.2.4.12.5. Range

Range: 1 Januari 1970 00:00:00 UTC hingga 19 Januari 2038 10:14:07 UTC.

3.2.4.13. Elemen Data Waktu Pemrosesan

3.2.4.13.1. Nama

Nama elemen data: *process time* (KD_SKPL_13).

3.2.4.13.2. Representasi

Representasi dalam tipe *timestamp*.

3.2.4.13.3. Unit atau Format

Format sembarang.

3.2.4.13.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga satu detik.

3.2.4.13.5. Range

Range: 1 Januari 1970 00:00:00 UTC hingga 19 Januari 2038 10:14:07 UTC.

3.2.4.14. Elemen Data Waktu Selesai

3.2.4.14.1. Nama

Nama elemen data: *complete time* (KD_SKPL_14).

3.2.4.14.2. Representasi

Representasi dalam tipe *timestamp*.

3.2.4.14.3. Unit atau Format

Format sembarang.

3.2.4.14.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga satu detik.

3.2.4.14.5. Range

Range: 1 Januari 1970 00:00:00 UTC hingga 19 Januari 2038 10:14:07 UTC.

3.2.4.15. Elemen Data Executable Malware

3.2.4.15.1. Nama

Nama elemen data: *executable* (KD_SKPL_15).

3.2.4.15.2. Representasi

Representasi dalam tipe *file*.

3.2.4.15.3. Unit atau Format

Format: *file* yang dapat dijalankan (*executable*).

3.2.4.16. Elemen Data Dokumen yang Diselamatkan

3.2.4.16.1. Nama

Nama elemen data: *document* (KD_SKPL_16).

3.2.4.16.2. Representasi

Representasi dalam tipe *file*.

3.2.4.16.3. Unit atau Format

Format: *file* dokumen.

3.2.4.17. Elemen Data Catatan Penyelamatan

3.2.4.17.1. Nama

Nama elemen data: *log* (KD_SKPL_17).

3.2.4.17.2. Representasi

Representasi dalam tipe *file*.

3.2.4.17.3. Unit atau Format

Format: *file* teks.

3.2.4.18. Elemen Data Keterhapusan

3.2.4.18.1. Nama

Nama elemen data: *is deleted* (KD_SKPL_18).

3.2.4.18.2. Representasi

Representasi dalam tipe *boolean*.

3.2.4.18.3. Unit atau Format

Format sembarang.

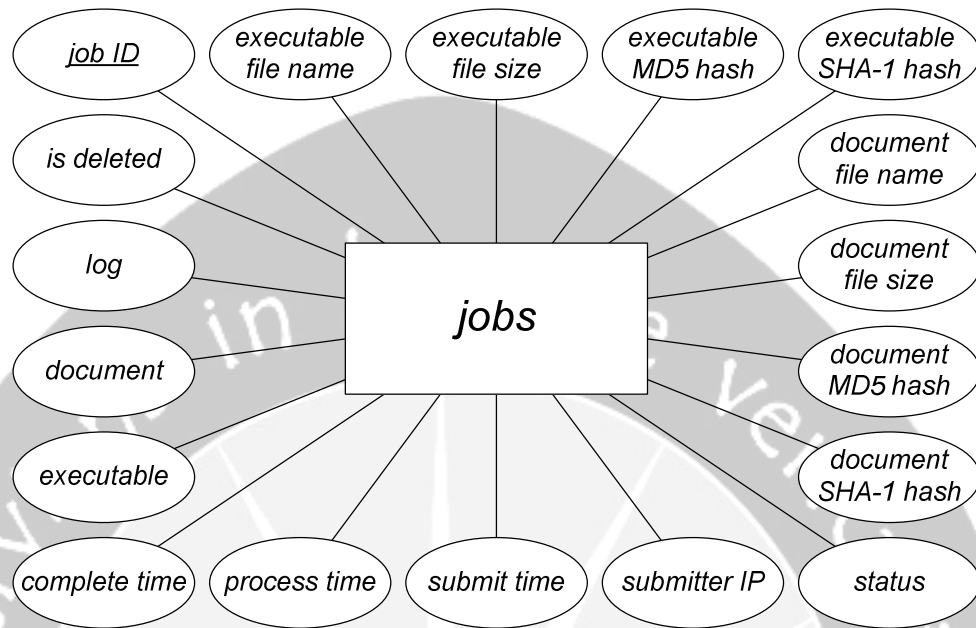
3.2.4.18.4. Presisi atau Keakuratan

Presisi atau keakuratan: akurat hingga setiap nilai kebenaran.

3.2.4.18.5. Range

Range: *true* dan *false*.

3.2.5. Entity-Relationship Diagram



Gambar 3.8. Entity-Relationship Diagram

3.3. Kebutuhan Performansi

3.3.1. Kebutuhan Numerik Statik

Kebutuhan numerik statis PANDORA adalah:

1. tipe *record* yang ditangani adalah data pekerjaan penyelamatan, dan
2. jumlah maksimum *record* data pekerjaan penyelamatan yang dapat disimpan adalah 2^{128} *record* (3×10^{38} *record*).

3.3.2. Kebutuhan Numerik Dinamis

Kebutuhan numerik dinamis PANDORA adalah:

1. jumlah proses Antarmuka Web, Penjadwal, dan Penyelamat yang berjalan pada suatu saat dapat lebih dari satu,
2. jumlah pengunjung yang ditangani sebuah proses Antarmuka Web dapat lebih dari satu,

3. jumlah proses Penyelamat yang ditangani sebuah proses Penjadwal dapat lebih dari satu, dan
4. jumlah pekerjaan penyelamatan dapat melebihi jumlah proses Penyelamat.

3.4. Batasan Perancangan

Batasan dalam perancangan PANDORA ini adalah:

1. *malware* yang akan diproses sistem adalah *malware-malware* yang dapat berjalan di sistem operasi Windows XP,
2. *malware* yang akan diproses sistem adalah *malware-malware* yang hanya terdiri atas sebuah *file* EXE dan hanya "menyandera" sebuah *file* dokumen, dan
3. *malware* yang akan diproses sistem adalah *malware-malware* yang "kooperatif" dengan sistem.

3.5. Atribut Sistem Perangkat Lunak

Atribut perangkat lunak PANDORA adalah:

1. keandalan (*reliability*)
keandalan perangkat lunak dilakukan melalui perawatan otomatis yang dapat mengenali dan memperbaiki masalah-masalah yang sering terjadi secara otomatis,
2. keamanan (*security*)
keamanan perangkat lunak dilakukan dengan pembatasan hak akses berdasarkan peran (anonim atau administrator) dan otentikasi administrator melalui *password*, dan
3. kepemindahan (*portability*)
kepemindahan perangkat lunak *server-side* dilakukan dengan menggunakan perangkat-perangkat lunak (selain yang berjalan dalam *virtual machine*) yang kepemindahannya telah terbukti (telah tersedia untuk *platform* Windows dan Linux), sedangkan kepemindahan

perangkat lunak *client-side* dilakukan dengan merancang antarmuka web yang keluarannya memenuhi standar XHTML 1.0 dan CSS 2.0 dan dengan mengujinya dalam berbagai *web browser* populer.



DPPL

Deskripsi Perancangan Perangkat Lunak

PANDORA

(Sistem Penyelamatan Dokumen yang
"Disandera" Malware)


untuk:

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Ronald Budi Gunawan (TF / 04366)

Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta	Nomor Dokumen		Halaman
		DPPL-PANDORA		1/31
		Revisi	-	1 Februari 2010

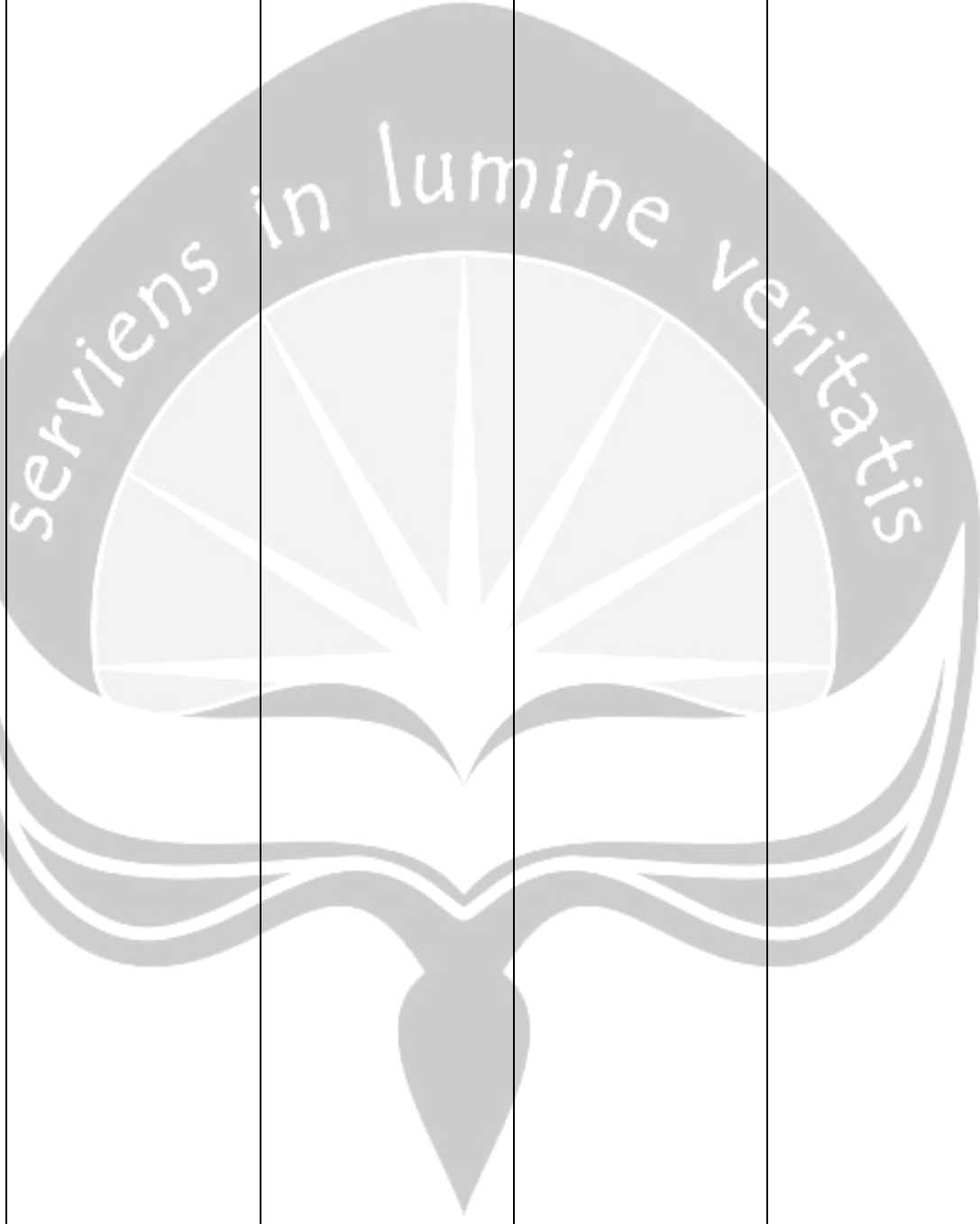
Daftar Perubahan

Revisi	Deskripsi
A	
B	
C	
D	
E	

INDEX TGL	A	B	C	D	E
Ditulis oleh					
Diperiksa oleh					
Disetujui oleh					

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



DAFTAR ISI

1. Pendahuluan	8
1.1. Tujuan	8
1.2. Lingkup Masalah	8
1.3. Definisi, Akronim, dan Singkatan	8
1.4. Referensi	9
1.5. Deskripsi Umum Dokumen	10
2. Deskripsi Dekomposisi	11
2.1. Dekomposisi Data	11
2.1.1. Deskripsi Entitas Data Pekerjaan Penyelamatan	11
2.1.2. <i>Physical Data Model</i>	12
2.2. Dekomposisi Modul	12
3. Deskripsi Perancangan Antarmuka dan Fungsional	13
3.1. Deskripsi Antarmuka untuk Anonim	13
3.1.1. Deskripsi Antarmuka Pengajuan <i>File</i>	13
3.1.1.1. Deskripsi Tombol <i>Browse</i>	13
3.1.1.2. Deskripsi Tombol <i>Submit</i>	13
3.1.2. Deskripsi Antarmuka Pengajuan <i>File</i> Berhasil	14
3.1.2.1. Deskripsi <i>Link</i> Status Penyelamatan <i>File</i>	15
3.1.3. Deskripsi Antarmuka Informasi dan Status Penyelamatan <i>File</i>	15
3.1.3.1. Deskripsi <i>Link Download</i> Dokumen yang Diselamatkan	16
3.1.3.2. Deskripsi <i>Link Refresh</i>	17
3.1.4. Deskripsi Antarmuka <i>Login</i>	17
3.1.4.1. Deskripsi Tombol <i>Log In</i>	18
3.1.5. Deskripsi Antarmuka <i>Login</i> Berhasil	18
3.2. Deskripsi Antarmuka untuk Administrator	19
3.2.1. Deskripsi Antarmuka Pengelolaan <i>File</i>	19
3.2.1.1. Deskripsi Tombol <i>Select All</i>	20
3.2.1.2. Deskripsi Tombol <i>Select None</i>	21
3.2.1.3. Deskripsi Tombol <i>Delete</i>	21
3.2.1.4. Deskripsi Tombol <i>Restart Processing</i>	21
3.2.1.5. Deskripsi <i>Link</i> Nama <i>File</i>	22
3.2.1.6. Deskripsi <i>Link</i> Tampilkan Catatan Penyelamatan	22
3.2.1.7. Deskripsi <i>Link Download File</i>	23
3.2.1.8. Deskripsi <i>Link Download</i> Dokumen yang Diselamatkan	23
3.2.1.9. Deskripsi <i>Link</i> Urutkan Menaik	23
3.2.1.10. Deskripsi <i>Link</i> Urutkan Menurun	24
3.2.1.11. Deskripsi <i>Link</i> Halaman Pertama	24
3.2.1.12. Deskripsi <i>Link</i> Halaman Sebelumnya	24
3.2.1.13. Deskripsi <i>Link</i> Nomor Halaman	24

3.2.1.14. Deskripsi <i>Link</i> Halaman Berikutnya	25
3.2.1.15. Deskripsi <i>Link</i> Halaman Terakhir	25
3.2.1.16. Deskripsi <i>Link Refresh</i>	25
3.2.2. Deskripsi Antarmuka Informasi dan Status Penyelamatan <i>File</i>	25
3.2.2.1. Deskripsi <i>Link</i> Tampilkan Catatan Penyelamatan	26
3.2.2.2. Deskripsi <i>Link Download File</i>	27
3.2.2.3. Deskripsi <i>Link Download</i> Dokumen yang Diselamatkan	27
3.2.2.4. Deskripsi <i>Link Delete</i>	28
3.2.2.5. Deskripsi <i>Link Restart Processing</i>	29
3.2.2.6. Deskripsi <i>Link Refresh</i>	30
3.2.3. Deskripsi Antarmuka Catatan Penyelamatan ...	30
3.2.4. Deskripsi Antarmuka <i>Logout</i>	30



DAFTAR TABEL

Tabel 1.1. Definisi, Akronim, dan Singkatan.....	9
Tabel 2.1. Deskripsi Entitas Data Pekerjaan Penyelamatan	11



DAFTAR GAMBAR

Gambar 2.1. <i>Physical Data Model</i>	12
Gambar 2.2. <i>Arsitektur Modul</i>	12
Gambar 3.1. <i>Antarmuka Pengajuan File</i>	13
Gambar 3.2. <i>Antarmuka Pengajuan File Berhasil</i>	14
Gambar 3.3. <i>Antarmuka Informasi dan Status Penyelamatan File</i>	16
Gambar 3.4. <i>Antarmuka Login</i>	17
Gambar 3.5. <i>Antarmuka Login Berhasil</i>	18
Gambar 3.6. <i>Antarmuka Pengelolaan File</i>	20
Gambar 3.7. <i>Antarmuka Informasi dan Status Penyelamatan File</i>	26
Gambar 3.8. <i>Antarmuka Catatan Penyelamatan</i>	30
Gambar 3.9. <i>Antarmuka Logout</i>	31

1. Pendahuluan

1.1. Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

1.2. Lingkup Masalah

Perangkat lunak PANDORA dikembangkan untuk menangani penyelamatan dokumen yang "disandera" malware, dan terdiri atas tiga subsistem, yaitu:

1. Antarmuka Web, merupakan subsistem yang menyajikan antarmuka web agar para pengguna sistem dapat berinteraksi dengan sistem,
2. Penjadwal, merupakan subsistem yang menangani perawatan sistem dan penjadwalan pengerjaan setiap pekerjaan penyelamatan dalam Penyelamat, dan
3. Penyelamat, merupakan subsistem yang mengerjakan setiap pekerjaan penyelamatan yang diberikan Penjadwal.

1.3. Definisi, Akronim, dan Singkatan

Daftar definisi, akronim, dan singkatan:

Tabel 1.1. Definisi, Akronim, dan Singkatan

Keyword / Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak, disebut juga <i>Software Design Description</i> (SDD), merupakan deskripsi dari perancangan produk atau perangkat lunak yang akan dikembangkan.
PANDORA	Perangkat lunak yang menangani penyelamatan dokumen yang "disandera" malware.

1.4. Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Budi Gunawan, Ronald. 2009. *Spesifikasi Kebutuhan Perangkat Lunak Sistem Penyelamatan Dokumen yang "Disandera" Malware (SKPL-PANDORA)*. Yogyakarta: Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Atma Jaya Yogyakarta.
2. Software Engineering Standards Committee, IEEE Computer Society. 1998. *IEEE Std 1016-1998: IEEE Recommended Practice for Software Design Descriptions*. Piscataway, New Jersey: IEEE Press.
3. Pressman, Roger S. 2002. *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku I)*. Ed. II. Terj. LN Harnaningrum. Yogyakarta: ANDI.
4. Pressman, Roger S. 2003. *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku II)*. Ed. II. Terj. LN Harnaningrum. Yogyakarta: ANDI.

1.5. Deskripsi Umum Dokumen

Secara umum, dokumen DPPL ini terbagi atas 3 bagian utama. Bagian pertama berisi penjelasan mengenai dokumen DPPL yang mencakup: tujuan pembuatan DPPL, ruang lingkup masalah dalam pengembangan perangkat lunak PANDORA, definisi, referensi, dan deskripsi umum tentang dokumen DPPL.

Bagian kedua berisi deskripsi dekomposisi perangkat lunak PANDORA yang akan dikembangkan, mencakup dekomposisi data dan dekomposisi modul dari perangkat lunak PANDORA.

Bagian ketiga berisi deskripsi perancangan antarmuka dan fungsional dari masing-masing *form* yang akan digunakan dalam pengembangan perangkat lunak PANDORA tersebut.

2. Deskripsi Dekomposisi

2.1. Dekomposisi Data

2.1.1. Deskripsi Entitas Data Pekerjaan Penyelamatan

Tabel 2.1. Deskripsi Entitas Data Pekerjaan Penyelamatan

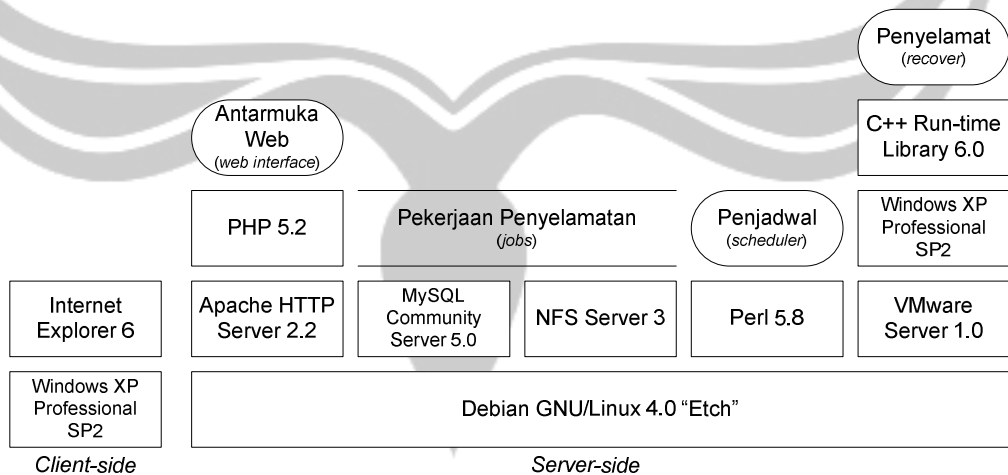
Nama	Tipe	Panjang	Keterangan
job_id	CHAR	32	ID pekerjaan penyelamatan
executable_file_name	VARCHAR	255	Nama file executable malware
executable_file_size	INT	-	Ukuran file executable malware
executable_md5_hash	CHAR	32	Hash MD5 executable malware
executable_shal_hash	CHAR	40	Hash SHA-1 executable malware
document_file_name	VARCHAR	255	Nama file dokumen yang diselamatkan
document_file_size	INT	-	Ukuran file dokumen yang diselamatkan
document_md5_hash	CHAR	32	Hash MD5 dokumen yang diselamatkan
document_shal_hash	CHAR	40	Hash SHA-1 dokumen yang diselamatkan
status	TINYINT	-	Status
has_log	BOOL	-	Keberadaan catatan penyelamatan
submitter_ip	VARCHAR	15	Alamat IP pengaju
submit_time	TIMESTAMP	-	Waktu pengajuan
process_time	TIMESTAMP	-	Waktu pemrosesan
complete_time	TIMESTAMP	-	Waktu selesai
executable	FILE *)	-	Executable malware
document	FILE *)	-	Dokumen yang diselamatkan
log	FILE *)	-	Catatan penyelamatan
is_deleted	BOOL	-	Keterhapusan
<i>Primary key: field (job_id)</i>			
*) disimpan sebagai file dalam network share.			

2.1.2. Physical Data Model

jobs	
PK	<u>job_id</u>
	executable_file_name executable_file_size executable_md5_hash executable_sha1_hash document_file_name document_file_size document_md5_hash document_sha1_hash status has_log submitter_ip submit_time process_time complete_time executable document log is_deleted

Gambar 2.1. Physical Data Model

2.2. Dekomposisi Modul



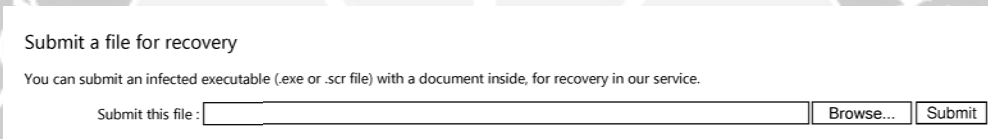
Gambar 2.2. Arsitektur Modul

3. Deskripsi Perancangan Antarmuka dan Fungsional

3.1. Deskripsi Antarmuka untuk Anonim

3.1.1. Deskripsi Antarmuka Pengajuan *File*

Antarmuka Pengajuan *File* (*Submit a file for recovery*) merupakan antarmuka yang digunakan untuk mengajukan sebuah *file* untuk diselamatkan dokumennya. Pengguna memilih *file* yang akan diajukan dengan mengklik tombol *Browse...*, lalu mengklik tombol *Submit* untuk mengajukan *file* tersebut.



Gambar 3.1. Antarmuka Pengajuan *File*

3.1.1.1. Deskripsi Tombol *Browse...*

Tombol *Browse...* merupakan tombol yang digunakan untuk memilih *file* yang akan diajukan. Proses pemilihan *file* ditangani sepenuhnya oleh *web browser*.

3.1.1.2. Deskripsi Tombol *Submit*

Tombol *Submit* merupakan tombol yang digunakan untuk menyimpan *file* yang diajukan dalam sistem, menampilkan *link* ke antarmuka Informasi dan Status Penyelamatan *File* untuk *file* tersebut dalam antarmuka Pengajuan *File* Berhasil, lalu menampilkan antarmuka Pengajuan *File* Berhasil tersebut.

Secara prosedural:

```
<?php
...
if( isset( $_FILES[ 'executable' ] ) && isset( $_POST[ 'submit_job' ] ) ) {
```

```

$job_id = uid();
$executable_file_name = $_FILES[ 'executable' ][ 'name' ];
$executable_file_size = $_FILES[ 'executable' ][ 'size' ];
$executable_md5_hash = md5_file( $_FILES[ 'executable' ][ 'tmp_name' ] );
$executable_shal_hash = sha1_file( $_FILES[ 'executable' ][ 'tmp_name' ] );
$submitter_ip = $_SERVER[ 'REMOTE_ADDR' ];

mkdir( $NETWORK_SHARE_PATH . '/jobs/' . $job_id . '/' );

move_uploaded_file(
    $_FILES[ 'executable' ][ 'tmp_name' ],
    $NETWORK_SHARE_PATH . '/jobs/' . $job_id . '/executable'
);

$databse = new PDO( ... );

$insert_job_statement = $databse->prepare(
    'INSERT INTO jobs (
        ' job_id ' .
        ' , executable_file_name ' .
        ' , executable_file_size ' .
        ' , executable_md5_hash ' .
        ' , executable_shal_hash ' .
        ' , submitter_ip ' .
    ' ) VALUES ( ? , ? , ? , ? , ? , ? ) '
);

$insert_job_statement->bindParam( 1, $job_id, PDO::PARAM_STR );
$insert_job_statement->bindParam( 2, $executable_file_name, PDO::PARAM_STR );
$insert_job_statement->bindParam( 3, $executable_file_size, PDO::PARAM_INT );
$insert_job_statement->bindParam( 4, $executable_md5_hash, PDO::PARAM_STR );
$insert_job_statement->bindParam( 5, $executable_shal_hash, PDO::PARAM_STR );
$insert_job_statement->bindParam( 6, $submitter_ip, PDO::PARAM_STR );
$insert_job_statement->execute();

print( '<a href="' . get_page_uri( 'job_details', array( 'job_id' => $job_id
) ) . '">...</a>' );

}
...
?>

```

3.1.2. Deskripsi Antarmuka Pengajuan File Berhasil

Antarmuka Pengajuan *File* Berhasil (*File submitted*) merupakan antarmuka yang digunakan untuk memberitahukan keberhasilan pengajuan sebuah *file*. Pengguna dapat mengikuti *link* yang ada untuk menampilkan status penyelamatan *file*.

File submitted

Your file has been submitted for recovery. To check your file's status, please visit the following page (make sure to save the link for future reference):

http://server/pandora/index.php?page=job_details&job_id=0123456789abcdef0123456789abcdef

Gambar 3.2. Antarmuka Pengajuan *File* Berhasil

3.1.2.1. Deskripsi *Link* Status Penyelamatan *File*

Link Status Penyelamatan *File* merupakan *link* yang digunakan untuk menampilkan status penyelamatan *file* yang diajukan, dengan cara menampilkan antarmuka Informasi dan Status Penyelamatan *File* untuk *file* tersebut.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxxx">...</a>
...
</html>
```

3.1.3. Deskripsi Antarmuka Informasi dan Status Penyelamatan *File*

Antarmuka Informasi dan Status Penyelamatan *File* (*File information and recovery status*) merupakan antarmuka yang digunakan untuk menampilkan informasi dan status penyelamatan sebuah *file*. Sebelum antarmuka ditampilkan, informasi dan status penyelamatan *file* tersebut akan ditampilkan dalam antarmuka. Setelah antarmuka ditampilkan, pengguna dapat mengikuti *link Download Dokumen yang Diselamatkan* (dalam bagian *Recovered document*) untuk men-*download* dokumen yang diselamatkan dari *file* tersebut. Selain itu, pengguna juga dapat mengikuti *link Refresh* untuk memperbarui informasi yang ditampilkan.

Secara prosedural:

```
<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
    $job_id = $_GET[ 'job_id' ];
    ...
    $database = new PDOk

    $get_job_statement = $database->prepare(
        'SELECT ' .
            ' job_id ' .
            ', executable_file_name ' .
            ', executable_file_size ' .
            ', executable_md5_hash ' .
            ', executable_shal_hash ' .
            ', document_file_name ' .
            ', document_file_size ' .
            ', document_md5_hash ' .
            ', document_shal_hash ' .
```



```

        ', status ' .
        ', has_log ' .
        ', submitter_ip ' .
        ', UNIX_TIMESTAMP( submit_time ) AS submit_time ' .
        ', UNIX_TIMESTAMP( process_time ) AS process_time ' .
        ', UNIX_TIMESTAMP( complete_time ) AS complete_time ' .
    'FROM jobs ' .
    'WHERE ' .
        'job_id = ? ' .
        'AND is_deleted = 0 '
    );
    $get_job_statement->bindParam( 1, $job_id, PDO::PARAM_STR );
    $get_job_statement->execute();
    $job = $get_job_statement->fetch( PDO::FETCH_ASSOC );
    ...
}
...
?>

```

File information and recovery status for malware1.exe [refresh](#)

You can view a file's information and recovery status, as well as perform various tasks related to the file.


Submitted file	
File name :	malware1.exe
File size :	120.56 KB (123,456 bytes)
MDS hash :	0123456789abcdef0123456789abcdef
SHA-1 hash :	123456789abcdef0123456789abcdef012345678

Recovered document download	
File name :	document1.doc
File size :	22.91 KB (23,456 bytes)
MDS hash :	23456789abcdef0123456789abcdef01
SHA-1 hash :	3456789abcdef0123456789abcdef0123456789a

Recovery status	
Status :	Completed
Submitted at :	Fri 01 Jan 2010 12:00:00 AM UTC
Processed at :	Fri 01 Jan 2010 12:00:00 AM UTC
Completed at :	Fri 01 Jan 2010 12:05:00 AM UTC

Gambar 3.3. Antarmuka Informasi dan Status Penyelamatan File

3.1.3.1. Deskripsi [Link Download Dokumen](#) yang Diselamatkan

[Link Download Dokumen](#) yang Diselamatkan ( [download](#)) merupakan *link* yang digunakan untuk *download* dokumen yang diselamatkan dari *file*, dengan cara membaca dokumen yang diselamatkan dari *file* tersebut lalu menyajikannya dalam bentuk *download*.

Secara prosedural:

```

<html>
...

```

```

<a href="index.php?page=job_details&job_id=xxxxxxxx&download_document=1">
...</a>
...
</html>
<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
...
$job_id           = $job[ 'job_id' ];
$document_file_name = $job[ 'document_file_name' ];
$document_file_size = $job[ 'document_file_size' ];
$complete_time     = $job[ 'complete_time' ];
...
if( isset( $_GET[ 'download_document' ] ) ) {

    ob_end_clean();
    header( 'Content-Disposition: attachment; filename="' .
        $document_file_name . '"' );
    header( 'Content-Length: ' . $document_file_size );
    header( 'Content-Type: application/octet-stream' );
    header( 'Last-Modified: ' . date( 'r', $complete_time ) );
    readfile( $NETWORK_SHARE_PATH . '/jobs/' . $job_id . '/document' );

}
...
}
...
?>

```

3.1.3.2. Deskripsi *Link Refresh*

Link Refresh merupakan *link* yang digunakan untuk memperbarui tampilan informasi dan status penyelamatan *file*, dengan cara menampilkan kembali antarmuka.

Secara prosedural:

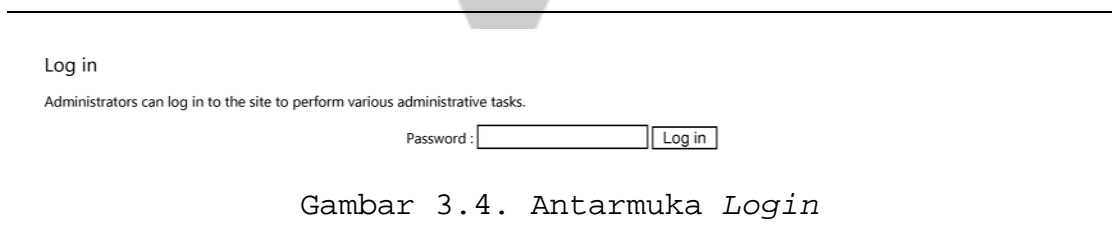
```

<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxxx">...</a>
...
</html>

```

3.1.4. Deskripsi Antarmuka *Login*

Antarmuka *Login* (*Log in*) merupakan antarmuka yang digunakan untuk *log in*. Pengguna memasukkan *password* dalam *text box*, lalu mengklik tombol *Log In* untuk *log in*.



Gambar 3.4. Antarmuka *Login*

3.1.4.1. Deskripsi Tombol *Log In*

Tombol *Log In* merupakan tombol yang digunakan untuk memproses *login*, dengan cara membandingkan *password* yang dimasukkan dalam *text box* dengan *password* administrator: jika sama, antarmuka *Login Berhasil* akan ditampilkan; sebaliknya, jika berbeda, pesan kesalahan *login* akan ditampilkan.

Secara prosedural:

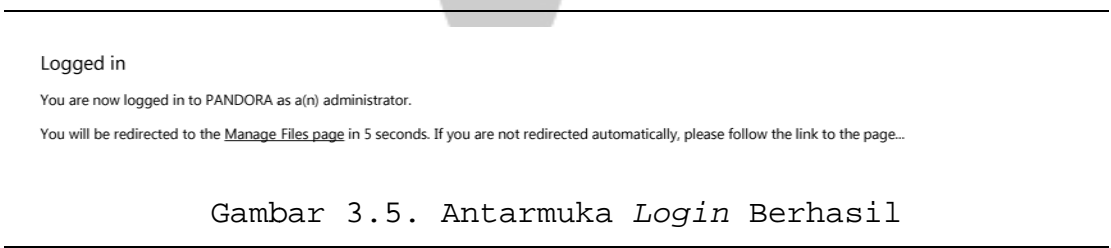
```
<?php
...
if( isset( $_POST[ 'password' ] ) && isset( $_POST[ 'login' ] ) ) {
    $password = $_POST[ 'password' ];
    $result = '';
    if( $password == $ADMINISTRATORS_PASSWORD ) {
        set_role( 'administrator' );
        $result = 'Login successful';
    } else {
        $result = 'Login failed';
    }
}
...
?>
```

3.1.5. Deskripsi Antarmuka *Login Berhasil*

Antarmuka *Login Berhasil (Logged in)* merupakan antarmuka yang digunakan untuk memberitahukan keberhasilan *login*. Beberapa saat setelah antarmuka ditampilkan, antarmuka *Pengelolaan File* akan ditampilkan.

Secara prosedural:

```
<html>
...
<meta content="5; url=index.php" http-equiv="refresh" />
...
</html>
```



Gambar 3.5. Antarmuka *Login Berhasil*

3.2. Deskripsi Antarmuka untuk Administrator

3.2.1. Deskripsi Antarmuka Pengelolaan *File*

Antarmuka Pengelolaan *File* (*Manage submitted files*) merupakan antarmuka yang digunakan untuk mengelola *file-file* yang diajukan. Sebelum antarmuka ditampilkan, daftar *file* yang diajukan akan ditampilkan dalam antarmuka. Setelah antarmuka ditampilkan, pengguna dapat mengelola *file-file* yang diajukan dengan memilih *file-file* yang akan dikenai tindakan dengan mencentang *check box file-file* tersebut, mengklik tombol *Select All* (untuk memilih semua *file* yang ditampilkan), atau mengklik tombol *Select None* (untuk tidak memilih semua *file* yang ditampilkan), lalu mengklik tombol *Delete...* untuk menghapus *file-file* yang dipilih atau tombol *Restart Processing...* untuk mengulang pemrosesan *file-file* yang dipilih. Selain itu, pengguna juga dapat mengikuti *link* Nama *File* (dalam kolom *Submitted file*) untuk menampilkan informasi dan status penyelamatan *file* tersebut, *link* Tampilkan Catatan Penyelamatan (dalam kolom *Recovery status*) untuk menampilkan catatan penyelamatan *file* tersebut, *link* *Download File* (dalam kolom *Submitted file*) untuk *download file* tersebut, *link* *Download Dokumen* yang Diselamatkan (dalam kolom *Recovered document*) untuk *download* dokumen yang diselamatkan dari *file* tersebut, *link-link* pengurutan (dalam judul-judul kolom) untuk mengurutkan daftar *file*, *link-link* halaman (di bawah daftar *file*) untuk berpindah halaman daftar *file*, dan *link* *Refresh* untuk memperbarui daftar *file*.

Secara prosedural:

```
<?php
...
$sort_column = $_GET[ 'sort_column' ];
$sort_order = $_GET[ 'sort_order' ];
$page_number = $_GET[ 'page_number' ];
```

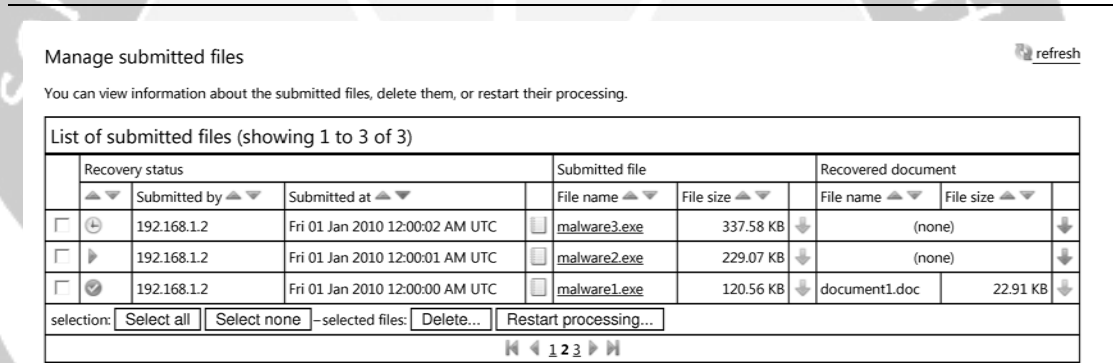
```

$page_row_count = 10;
$page_row_start = $page_number * $page_row_count;

$database = new PDO( ... );

$get_jobs_statement = $database->prepare(
    'SELECT ' .
        ' job_id ' .
        ', executable_file_name ' .
        ', executable_file_size ' .
        ', document_file_name ' .
        ', document_file_size ' .
        ', status ' .
        ', has_log ' .
        ', submitter_ip ' .
        ', UNIX_TIMESTAMP( submit_time ) AS submit_time ' .
    'FROM jobs ' .
    'WHERE is_deleted = 0 ' .
    'ORDER BY ' . $sort_column . ' ' . $sort_order . ' ' .
    'LIMIT ?, ? '
);
$get_jobs_statement->bindParam( 1, $page_row_start, PDO::PARAM_INT );
$get_jobs_statement->bindParam( 2, $page_row_count, PDO::PARAM_INT );
$get_jobs_statement->execute();
...
?>

```



Gambar 3.6. Antarmuka Pengelolaan File

3.2.1.1. Deskripsi Tombol *Select All*

Tombol *Select All* merupakan tombol yang digunakan untuk memilih semua file yang ditampilkan, dengan cara mencentang semua *check box* dalam antarmuka.

Secara prosedural:

```

<html>
...
<script type="text/javascript">
function select_all_jobs( root ) {
    if( /^select_job_.*_control$/ .test( root.id ) ) {
        root.checked = true;
    } else {
        var child = root.firstChild;
        while( child != null ) {
            select_all_jobs( child );
            child = child.nextSibling;
        }
    }
}

```

```

    }
  }
</script>
...
<input onclick="select_all_jobs( document )" type="button" value="Select all" />
...
</script>

```

3.2.1.2. Deskripsi Tombol *Select None*

Tombol *Select None* merupakan tombol yang digunakan untuk tidak memilih semua *file* yang ditampilkan, dengan cara menghapus centangan semua *check box* dalam antarmuka.

Secara prosedural:

```

<html>
...
<input type="reset" value="Select none" />
...
</script>

```

3.2.1.3. Deskripsi Tombol *Delete...*

Tombol *Delete...* merupakan tombol yang digunakan untuk menghapus *file-file* yang dipilih dari dalam sistem.

Secara prosedural:

```

<?php
...
if( isset( $_POST[ 'delete_jobs' ] ) ) {
    $database = new PDO( ... );

    $delete_job_statement = $database->prepare(
        'UPDATE jobs ' .
        'SET is_deleted = 1 ' .
        'WHERE ' .
        'job_id = ? ' .
        'AND is_deleted = 0 '
    );
    $delete_job_statement->bindParam( 1, $job_id, PDO::PARAM_STR );

    foreach( $_POST as $p => $q ) {
        if( preg_match( '/^select_job_(.*)$/i', $p, $subpatterns ) ) {
            $job_id = $subpatterns[ 1 ];
            $delete_job_statement->execute();
        }
    }
}
...
?>

```

3.2.1.4. Deskripsi Tombol *Restart Processing...*

Tombol *Restart Processing...* merupakan tombol yang digunakan untuk mengulang pemrosesan *file-file* yang dipilih, dengan cara mengatur status *file-file* tersebut

menjadi "mengantri" (*queued*), praktis mengulang pemrosesannya apabila telah selesai diproses.

Secara prosedural:

```
<?php
...
if( isset( $_POST[ 'restart_jobs' ] ) ) {

    $database = new PDO( ... );

    $restart_job_statement = $database->prepare(
        'UPDATE jobs ' .
        'SET status = ' . $JOB_STATUS_QUEUED . ' ' .
        'WHERE ' .
        'job_id = ? ' .
        'AND is_deleted = 0 '
    );
    $restart_job_statement->bindParam( 1, $job_id, PDO::PARAM_STR );

    foreach( $_POST as $p => $q ) {
        if( preg_match( '/^select_job_(.*)$/i', $p, $subpatterns ) ) {
            $job_id = $subpatterns[ 1 ];
            $restart_job_statement->execute();
        }
    }
}
...
?>
```


3.2.1.5. Deskripsi Link Nama File

Link Nama File (malware1.exe) merupakan *link* yang digunakan untuk menampilkan informasi dan status penyelamatan *file*, dengan cara menampilkan antarmuka Informasi dan Status Penyelamatan *File* untuk *file* tersebut.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxxx">...</a>
...
</html>
```

3.2.1.6. Deskripsi Link Tampilkan Catatan Penyelamatan

Link Tampilkan Catatan Penyelamatan () merupakan *link* yang digunakan untuk menampilkan catatan penyelamatan *file*, dengan cara menampilkan antarmuka Catatan Penyelamatan untuk *file* tersebut.

Secara prosedural:

```
<html>
...
```

```
<a href="index.php?page=job_details&job_id=xxxxxxx&view_log=1">...</a>
...
</html>
```

3.2.1.7. Deskripsi *Link Download File*

Link Download File (↓ dalam kolom *Submitted file*) merupakan *link* yang digunakan untuk men-*download file*, dengan cara membaca *file* tersebut lalu menyajikannya dalam bentuk *download*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxx&download_executable=1">
...</a>
...
</html>
```

3.2.1.8. Deskripsi *Link Download Dokumen yang Diselamatkan*

Link Download Dokumen yang Diselamatkan (↓ dalam kolom *Recovered document*) merupakan *link* yang digunakan untuk men-*download* dokumen yang diselamatkan dari *file*, dengan cara membaca dokumen yang diselamatkan dari *file* tersebut lalu menyajikannya dalam bentuk *download*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxx&download_document=1">...</a>
...
</html>
```

3.2.1.9. Deskripsi *Link Urutkan Menaik*

Link Urutkan Menaik (▲) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar terurut menaik menurut kolom tertentu.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&sort_column=xxxxxxx&sort_order=asc">
...</a>
...
</html>
```


3.2.1.10. Deskripsi *Link* Urutkan Menurun

Link Urutkan Menurun (▼) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar terurut menurun menurut kolom tertentu.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&sort_column=xxxxxxx&sort_order=desc">
...</a>
...
</html>
```

3.2.1.11. Deskripsi *Link* Halaman Pertama

Link Halaman Pertama (◀) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar menampilkan halaman pertama daftar *file*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&page_number=0">...</a>
...
</html>
```

3.2.1.12. Deskripsi *Link* Halaman Sebelumnya

Link Halaman Sebelumnya (◀) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar menampilkan halaman sebelumnya daftar *file*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&page_number=xx">...</a>
...
</html>
```

3.2.1.13. Deskripsi *Link* Nomor Halaman

Link Nomor Halaman (1 dan 3) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar menampilkan halaman tertentu daftar *file*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&page_number=xx">...</a>
...
...
</html>
```

```
</html>
```

3.2.1.14. Deskripsi *Link* Halaman Berikutnya

Link Halaman Berikutnya (►) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar menampilkan halaman berikutnya daftar *file*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&page_number=xx">...</a>
...
</html>
```

3.2.1.15. Deskripsi *Link* Halaman Terakhir

Link Halaman Terakhir (►) merupakan *link* yang digunakan untuk mengubah tampilan daftar *file* agar menampilkan halaman terakhir daftar *file*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...&page_number=xx">...</a>
...
</html>
```

3.2.1.16. Deskripsi *Link Refresh*

Link Refresh merupakan *link* yang digunakan untuk memperbarui tampilan daftar *file*, dengan cara menampilkan kembali antarmuka.

Secara prosedural:

```
<html>
...
<a href="index.php?page=manage_jobs&...">...</a>
...
</html>
```

3.2.2. Deskripsi Antarmuka Informasi dan Status Penyelamatan *File*

Antarmuka Informasi dan Status Penyelamatan *File* (*File information and recovery status*) merupakan antarmuka yang digunakan untuk menampilkan informasi dan status penyelamatan sebuah *file*. Sebelum antarmuka

ditampilkan, informasi dan status penyelamatan *file* tersebut akan ditampilkan dalam antarmuka. Setelah antarmuka ditampilkan, pengguna dapat mengikuti *link* Tampilkan Catatan Penyelamatan (dalam bagian *Recovery status*) untuk menampilkan catatan penyelamatan *file* tersebut, *link* Download File (dalam bagian *Submitted file*) untuk men-download *file* tersebut, dan *link* Download Dokumen yang Diselamatkan (dalam bagian *Recovered document*) untuk men-download dokumen yang diselamatkan dari *file* tersebut. Selain itu, pengguna juga dapat mengikuti *link* Delete untuk menghapus *file* tersebut, *link* Restart Processing untuk mengulang pemrosesan *file* tersebut, dan *link* Refresh untuk memperbarui informasi yang ditampilkan.

File information and recovery status for malware1.exe

[delete](#) |
 [restart processing](#) |
 [refresh](#)

You can view a file's information and recovery status, as well as perform various tasks related to the file.

Submitted file download	
File name :	malware1.exe
File size :	120.56 KB (123,456 bytes)
MDS hash :	0123456789abcdef0123456789abcdef
SHA-1 hash :	123456789abcdef0123456789abcdef012345678

Recovered document download	
File name :	document1.doc
File size :	22.91 KB (23,456 bytes)
MDS hash :	23456789abcdef0123456789abcdef01
SHA-1 hash :	3456789abcdef0123456789abcdef0123456789a

Recovery status view log	
Status :	Completed
Submitted by :	192.168.1.2
Submitted at :	Fri 01 Jan 2010 12:00:00 AM UTC
Last processed at :	Fri 01 Jan 2010 12:00:00 AM UTC
Last completed at :	Fri 01 Jan 2010 12:05:00 AM UTC

Gambar 3.7. Antarmuka Informasi dan Status Penyelamatan *File*

3.2.2.1. Deskripsi *Link* Tampilkan Catatan Penyelamatan


Link Tampilkan Catatan Penyelamatan ([view log](#)) merupakan *link* yang digunakan untuk menampilkan catatan

penyelamatan *file*, dengan cara menampilkan antarmuka Catatan Penyelamatan untuk *file* tersebut.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxxx&view_log=1">...</a>
...
</html>
```

3.2.2.2. Deskripsi *Link Download File*

Link Download File ( download dalam bagian *Submitted file*) merupakan *link* yang digunakan untuk men-*download file*, dengan cara membaca *file* tersebut lalu menyajikannya dalam bentuk *download*.


Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxxx&download_executable=1">
...</a>
...
</html>
<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
...
$job_id           = $job[ 'job_id' ];
$executable_file_name = $job[ 'executable_file_name' ];
$executable_file_size = $job[ 'executable_file_size' ];
$submit_time      = $job[ 'submit_time' ];
...
if( isset( $_GET[ 'download_executable' ] ) ) ) {

    ob_end_clean();
    header( 'Content-Disposition: attachment; filename="' .
        $executable_file_name . '"' );
    header( 'Content-Length: ' . $executable_file_size );
    header( 'Content-Type: application/x-msdownload' );
    header( 'Last-Modified: ' . date( 'r', $submit_time ) );
    readfile( $NETWORK_SHARE_PATH . '/jobs/' . $job_id . '/executable' );

}
...
}
...
?>
```

3.2.2.3. Deskripsi *Link Download Dokumen yang Diselamatkan*

Link Download Dokumen yang Diselamatkan ( download dalam bagian *Recovered document*) merupakan *link* yang digunakan untuk men-*download* dokumen yang diselamatkan dari *file*, dengan cara membaca dokumen yang diselamatkan

dari *file* tersebut lalu menyajikannya dalam bentuk *download*.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxxx&download_document=1">
...</a>
...
</html>

<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
...
$job_id      = $job[ 'job_id' ];
$document_file_name = $job[ 'document_file_name' ];
$document_file_size = $job[ 'document_file_size' ];
$complete_time  = $job[ 'complete_time' ];
...
if( isset( $_GET[ 'download_document' ] ) ) {
...
ob_end_clean();
header( 'Content-Disposition: attachment; filename="' .
$document_file_name . '"' );
header( 'Content-Length: ' . $document_file_size );
header( 'Content-Type: application/octet-stream' );
header( 'Last-Modified: ' . date( 'r', $complete_time ) );
readfile( $NETWORK_SHARE_PATH . '/jobs/' . $job_id . '/document' );
...
}
...
}
...
?>
```

3.2.2.4. Deskripsi *Link Delete*

Link Delete merupakan *link* yang digunakan untuk menghapus *file*, dengan cara mengatur keterhapusan *file* tersebut menjadi *true*. Setelah *file* dihapus, antarmuka Pengelolaan *File* akan ditampilkan.

Secara prosedural:

```
<html>
...
<form action="" id="delete_job_form" method="post">
  <input name="delete_job" type="hidden" value="1" />
</form>
...
<a href="#" onclick="document.getElementById( 'delete_job_form' ).submit();
return( false );">...</a>
...
</html>

<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
...
$job_id = $_GET[ 'job_id' ];
...
if( isset( $_POST[ 'delete_job' ] ) ) {
...
$database = new PDO( ... );
```

```

        $delete_job_statement = $database->prepare(
            'UPDATE jobs ' .
            'SET is_deleted = 1 ' .
            'WHERE ' .
            'job_id = ? ' .
            'AND is_deleted = 0 '
        );
        $delete_job_statement->bindParam( 1, $job_id, PDO::PARAM_STR );
        $delete_job_statement->execute();

        ob_end_clean();
        header( 'HTTP/1.0 303 See Other' );
        header( 'Location: ' . get_absolute_page_uri( 'manage_jobs' ) );
    }
    ...
}
...
?>

```

3.2.2.5. Deskripsi Link Restart Processing

Link Restart Processing merupakan link yang digunakan untuk mengulang pemrosesan *file*, dengan cara mengatur status *file* tersebut menjadi "mengantri" (*queued*), praktis mengulang pemrosesannya apabila telah selesai diproses.

Secara prosedural:

```

<html>
...
<form action="" id="restart_job_form" method="post">
  <input name="restart_job" type="hidden" value="1" />
</form>
...
<a href="#" onclick="document.getElementById( 'restart_job_form' ).submit();
return( false );">...</a>
...
</html>
<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
    $job_id = $_GET[ 'job_id' ];
    ...
    if( isset( $_POST[ 'restart_job' ] ) ) {
        $database = new PDO( ... );

        $restart_job_statement = $database->prepare(
            'UPDATE jobs ' .
            'SET status = ' . $JOB_STATUS_QUEUED . ' ' .
            'WHERE ' .
            'job_id = ? ' .
            'AND is_deleted = 0 '
        );
        $restart_job_statement->bindParam( 1, $job_id, PDO::PARAM_STR );
        $restart_job_statement->execute();

        }
    ...
}
...
?>

```

3.2.2.6. Deskripsi *Link Refresh*

Link Refresh merupakan *link* yang digunakan untuk memperbarui tampilan informasi dan status penyelamatan *file*, dengan cara menampilkan kembali antarmuka.

Secara prosedural:

```
<html>
...
<a href="index.php?page=job_details&job_id=xxxxxxx">...</a>
...
</html>
```

3.2.3. Deskripsi Antarmuka Catatan Penyelamatan

Antarmuka Catatan Penyelamatan (*Recovery log*) merupakan antarmuka yang digunakan untuk menampilkan catatan penyelamatan sebuah *file*. Sebelum antarmuka ditampilkan, catatan penyelamatan sebuah *file* akan dibaca lalu ditampilkan dalam antarmuka.

Secara prosedural:

```
<?php
...
if( isset( $_GET[ 'job_id' ] ) ) {
...
$job_id = $job[ 'job_id' ];
...
if( isset( $_GET[ 'view_log' ] ) ) {
    readfile( $NETWORK_SHARE_PATH . '/jobs/' . $job_id . '/log' );
}
...
}
...
?>
```

Recovery log for malware1.exe

A file's recovery log shows what happened during the file's recovery.

```
[2010-01-01 00:00:00Z] pandoral scheduling: Processing job #0123456789abcdef0123456789abcdef...
[2010-01-01 00:05:00Z] pandoral scheduling: Completed processing job #0123456789abcdef0123456789abcdef.
```

Gambar 3.8. Antarmuka Catatan Penyelamatan

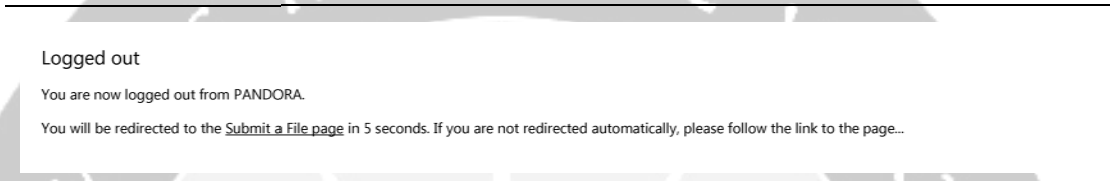
3.2.4. Deskripsi Antarmuka *Logout*

Antarmuka *Logout* (*Logged out*) merupakan antarmuka yang digunakan untuk *log out* dari dalam sistem. Beberapa

saat setelah antarmuka ditampilkan, antarmuka Pengajuan *File* akan ditampilkan.

Secara prosedural:

```
<?php
...
    reset_role();
...
?>
<html>
...
    <meta content="5; url=index.php" http-equiv="refresh" />
...
</html>
```



Gambar 3.9. Antarmuka *Logout*