

# Windows Kayıt Defteri (Registry) Analiz Yazılımı

Elektronik Delil Toplanması ve Analizi Dersi

Ömer Eldehil - 220509004

21.10.2025

# 1. Giriş

## 1.1. Projenin Amacı:

- 2.1. Projenin Amacı: Bu projenin temel amacı, Windows Kayıt Defteri (Registry) hive dosyalarından (SYSTEM, SOFTWARE, NTUSER.DAT) ve Windows Güvenlik Olay Günlüklerinden (Security.evtx) belirli sistem ve kullanıcı aktivite bilgilerini otomatik olarak ayrıştıran (parse eden), analiz eden ve kullanıcı dostu bir arayüzde sunan bir yazılım geliştirmektir.
- Yazılım, özellikle Kayıt Defteri ve ilgili log dosyaları üzerinden kurulan programlar, çalıştırılan uygulamalar (UserAssist), oturum açma/kapatma olayları, takılan USB aygıtları ve geçmiş ağ bağlantıları gibi bilgileri çıkarmayı hedeflemektedir.

## 1.2. Kapsam:

Yazılımın analiz ettiği ana veri kaynakları

- **Windows Kayıt Defteri (Registry) Hive Dosyaları:** SYSTEM Hive'ından USB bilgisi, SOFTWARE'den kurulu programlar ve ağlar, NTUSER.DAT'tan UserAssist
- **Windows Güvenlik Olay Günlükleri (Security.evtx):** , Security.evtx'ten oturum olayları

# 1. Giriş

## 1.3. Kullanılan Teknolojiler:

Programlama Dili: Python

Ana Kütüphaneler:

- **PyQt5:** Görsel Kullanıcı Arayüzü (GUI) oluşturmak için.
- **pandas:** Verileri işlemek, filtrelemek ve tablo formatında saklamak için.
- **python-evtx:** Windows Olay Günlüğü (.evtx) dosyalarını okumak ve ayrıştırmak için.
- **python-registry:** Windows Kayıt Defteri (Registry Hive) dosyalarını okumak ve ayrıştırmak için.
- **pathlib:** Dosya yollarını yönetmek için.
- **struct, codecs, datetime:** İkili (binary) verileri işlemek, şifre çözmek ve zaman damgalarını yönetmek için.

## 2. Yazılım Mimarisi ve Metodoloji

Bu bölümde, geliştirilen Windows Kayıt Defteri (Registry) Analiz Yazılımı'nın yapısı, kullanılan yöntemler ve veri işleme akışı ayrıntılı olarak açıklanmaktadır.

### 2.1. Genel Yaklaşım

Yazılım, modüler bir yapıda tasarlanmış olup iki ana Python dosyasından oluşmaktadır:

**registry\_parser.py (Veri Ayırıştırma Modülü):** Bu modül, analiz edilecek ham verilerin (Kayıt Defteri hive dosyaları ve Olay Günlükleri) okunması, ilgili kayıtların filtrelenmesi, verinin ayrıştırılması (parsing) ve yapılandırılmış bir formatta (pandas DataFrame) sunulması görevlerini üstlenen fonksiyonları içerir. Analiz mantığının çekirdeğini oluşturur.

**main\_gui.py (Görsel Arayüz Modülü):** Bu modül, PyQt5 kütüphanesi kullanılarak geliştirilmiş görsel kullanıcı arayüzünü (GUI) içerir. Kullanıcının analiz edilecek veri dosyalarını içeren bir klasör ("Vaka Klasörü") seçmesini sağlar. Seçilen klasördeki dosyaların yollarını registry\_parser.py modülündeki ilgili fonksiyonlara gönderir. Analiz sonuçlarını alır, bellekte saklar (self.data\_frames sözlüğü) ve kullanıcının sol paneldeki listeden seçtiği kategoriye göre ilgili veriyi sağ paneldeki tabloda (QTableWidget) görüntüler. Ayrıca işlem durumunu alt durum çubuğunda (QStatusBar) gösterir.

## 2.2. Metodoloji - Kopya Üzerinde Çalışma:

Windows işletim sistemi, çalışması için kritik olan Kayıt Defteri hive dosyalarını (SYSTEM, SOFTWARE, NTUSER.DAT) ve aktif Olay Günlüklerini (Security.evtx) kullanım sırasında kilitler. Bu dosyalara doğrudan erişim genellikle "Erişim Engellendi" hatasıyla sonuçlanır ve canlı sistem üzerinde analiz yapmak, delillerin değişmesine neden olabilir. Bu nedenle, analiz edilecek sistemden bu dosyaların kopyalarının alınması gerekmektedir. Kayıt Defteri dosyaları için Windows'un *reg save HKLM\SYSTEM <hedef\_dosya>* gibi komutları kullanılarak güvenli kopyalar oluşturulmuş ve analizler bu kopyalar üzerinde gerçekleştirilmiştir. Security.evtx dosyası ise Dosya Gezgini üzerinden kopyalanmıştır. Yazılım, kullanıcının bu kopyaları içeren bir klasörü seçmesiyle çalışır.

## 2.3. Veri Kaynakları ve Ayırıştırma Yöntemleri:

Yazılım, aşağıdaki veri kaynaklarından belirtilen bilgileri çıkarmak üzere tasarlanmıştır:

### A. Oturum Logları Analizi (Security.evtx):

- **Kaynak:** Windows Güvenlik Olay Günlüğü dosyası (Security.evtx).
- **Hedef Olaylar:** Başarılı oturum açma (4624), Başarısız oturum denemesi (4625), Oturum kapatma (4634, 4647).
- **Yöntem:**  
python-evtx kütüphanesi kullanılarak .evtx dosyası açılır ve log.records() ile her bir olay kaydına erişilir. Her kaydın XML verisi (record.xml()) alınır ve Python'un standart xml.etree.ElementTree kütüphanesi ile ayrıştırılır (ET.fromstring). XML içindeki ./e:System/e:EventID yolu (belirtilen namespace ns ile) kullanılarak olayın kimlik numarası (EventID) bulunur. EventID, logon\_filter listesindeki değerlerden biriye, olay işlenir. Olayın zaman damgası (record.timestamp()) alınır . ./e:EventData yoluyla olay verileri bulunur ve içindeki <e:Data Name="...">...</e:Data> etiketleri taranarak TargetUserName, SubjectUserName, LogonType, IPAddress gibi alanlar bir sözlüğe (data\_fields) çıkarılır. EventID ve LogonType değerleri, kod içinde tanımlanmış event\_id\_descriptions ve logon\_type\_descriptions sözlükleri kullanılarak okunabilir metinlere çevrilir. Sonuçlar, "Timestamp", "Olay", "Kullanıcı Adı", "Oturum Türü", "Kaynak IP" sütunlarını içeren bir pandas DataFrame olarak döndürülür (parse\_security\_log fonksiyonu).

```
11
12 # --- FONKSİYON 1: OTURUM LOGLARI (Değişiklik yok) ---
13 > def parse_security_log(evt_x_file_path): ...
81
82 # --- FONKSİYON 2: USB ANALİZİ (Değişiklik yok) ---
83 > def parse_usb_devices(system_hive_path): ...
161
162 # --- FONKSİYON 3: KURULU PROGRAMLAR (Değişiklik yok) ---
163 > def parse_installed_programs(software_hive_path): ...
216
217 # --- FONKSİYON 4: ÇALIŞTIRILAN PROGRAMLAR (Değişiklik yok) ---
218 > def parse_user_assist(ntuser_dat_path): ...
284
285 # --- FONKSİYON 5: AĞ (WIFI) BİLGİLERİ (Değişiklik yok) ---
286 > def parse_network_list(software_hive_path): ...
349
```

## Windows Forensic Artifact Analyzer

Dosya

Oturum Logları		Timestamp	Olay	Kullanıcı Adı	Oturum Türü	Kaynak IP
USB Depolama Aygıtları	1	2025-10-08 13:15:43	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Tüm USB Aygıtları	2	2025-10-08 13:15:43	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Kurulu Programlar	3	2025-10-08 13:15:49	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Çalıştırılan Programlar (UserAssist)	4	2025-10-08 13:17:00	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Ağ Geçmişi	5	2025-10-08 13:17:53	(4647) Oturum Kullanıcı Tarafından Kapatıldı	HP	N/A	N/A
	6	2025-10-08 13:46:36	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	7	2025-10-08 14:02:25	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	8	2025-10-08 14:03:14	(4624) Başarılı Oturum Açma	SYSTEM	(0) Sistem	-
	9	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	UMFD-0	(2) İnteraktif	-
	10	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	11	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	Local Service	(5) Hizmet	-
	12	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	NETWORK SERVICE	(5) Hizmet	-
	13	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-

## B. USB Cihaz Analizi (SYSTEM Hive):

- **Kaynak:** Windows Kayıt Defteri SYSTEM hive dosyası.
- **Hedef Bilgiler:**
  - USB Depolama Aygıtları: Cihaz Adı (Marka/Model), Seri Numarası, İlk Takılma Zamanı.
  - Tüm USB Aygıtları: VID/PID (Üretici/Ürün Kimliği), Instance ID (Örnek Kimliği)/Seri No, Aygıt Açıklaması, Kolay Ad, Konum Bilgisi, Son Güncelleme Zamanı.
- **Yöntem:**

python-registry kütüphanesi kullanılarak SYSTEM hive dosyası açılır (Registry.Registry).

  1. Depolama Aygıtları: ControlSet001\Enum\USBSTOR anahtarı açılır (reg.open). Bu anahtarın altındaki her bir alt anahtar (cihaz tipi, örn: Disk&Ven\_...) taranır. Onun altındaki alt anahtarlar (seri numarası/instance id) taranır. Seri numarası anahtarının adı (serial\_key.name()) Seri Numarası olarak, anahtarın zaman damgası (serial\_key.timestamp()) ise İlk Takılma Zamanı olarak alınır.
  2. Tüm Aygıtlar:ControlSet001\Enum\USB anahtarı açılır. Bu anahtarın altındaki her bir VID/PID anahtarı (vid\_pid\_key) taranır. Onun altındaki her bir instance anahtarı (instance\_key) taranır. Instance anahtarının adı (instance\_key.name()) Instance ID/Seri No olarak, zaman damgası (instance\_key.timestamp()) ise Son Güncelleme Zamanı olarak alınır. Ayrıca instance\_key içerisinde DeviceDesc, FriendlyName ve LocationInformation değerleri (instance\_key.value(...)) okunmaya çalışılır (bulunamazsa "N/A" atanır).

İki analiz sonucunda elde edilen veriler, iki ayrı pandas DataFrame (df\_storage, df\_all\_usb) olarak parse\_usb\_devices fonksiyonu tarafından döndürülür. Zaman damgaları pd.Timestamp objelerine çevrilir ve tablolar bu zamana göre sıralanır.



Windows Forensic Artifact Analyzer			
Dosya			
Oturum Logları USB Depolama Aygıtları Tüm USB Aygıtları Kurulu Programlar Çalıştırılan Programlar (UserAssist) Ağ Geçmişi	Cihaz Adı		Seri Numarası
	İlk Takılma Zamanı		
	1	Disk&Ven_VendorCo&Prod_ProductCode&Rev_2.00	4713691254353567775&0
	2	Disk&Ven_ATA&Prod_TOSHIBA_MQ01ACF0&Rev_1D	0123456789ABCDE&0

Windows Forensic Artifact Analyzer						
Dosya						
Oturum Logları USB Depolama Aygıtları Tüm USB Aygıtları Kurulu Programlar Çalıştırılan Programlar (UserAssist) Ağ Geçmişi		VID_PID	Instance ID / Seri No	Açıklama	Kolay Ad	Konum
	1	VID_8087&PID_0AF1&MI_00	7&3733247f8&0&0000	@oem132.inf,%devicename4g%;HP GNSS Sensor	@oem132.inf,...	0000.0000.0000.001.000.000.000.000.000
	2	VID_8087&PID_0AF1	05022016	@usb.inf,%usb\composite.devicedesc%;USB Composite Device	N/A	Port_#0001.Hub_#0002
	3	VID_30FA&PID_0400	5&1d3f3aca&0&4	@input.inf,%hid.devicedesc%;USB Input Device	N/A	Port_#0004.Hub_#0001
	4	VID_30FA&PID_0400	5&1d3f3aca&0&1	@input.inf,%hid.devicedesc%;USB Input Device	N/A	Port_#0001.Hub_#0001
	5	VID_8087&PID_0AC9&MI_00	7&4fa6012&0&0000	@netwmbclass.inf,%wmbclass.devicedesc%;Generic Mobile Broadband ...	Intel(R) XMM(TM) 7360 LTE-A	0000.0000.0000.003.000.000.000.000.000
	6	VID_8087&PID_0AC9	05022016	@usb.inf,%usb\composite.devicedesc%;USB Composite Device	N/A	Port_#0003.Hub_#0002
	7	VID_8087&PID_0ACA&MI_04	7&1e2cd3d3&0&0004	@oem133.inf,%devicename%;ModemControl Device	N/A	0000.0000.0000.002.000.000.000.000.000
	8	VID_0408&PID_5374&MI_00	6&2a13ead0&0&0000	@oem257.inf,%rtsuvc.devicedesc%;HP Universal Camera Driver	HP HD Camera	0000.0014.0000.002.000.000.000.000.000
	9	VID_8087&PID_0026	5&1d3f3aca&0&10	@oem252.inf,%ibt_usb%;Intel(R) Wireless Bluetooth(R)	N/A	Port_#0010.Hub_#0001
	10	VID_0408&PID_5374	200901010001	@usb.inf,%usb\composite.devicedesc%;USB Composite Device	N/A	Port_#0002.Hub_#0001
	11	VID_8087&PID_0ACA&MI_02	7&1e2cd3d3&0&0002	@usbser.inf,%usbserial.devicedesc%;USB Seri Cihaz	USB Seri Cihaz (COM3)	0000.0000.0000.002.000.000.000.000.000
	12	VID_8087&PID_0ACA&MI_00	7&1e2cd3d3&0&0000	@usbser.inf,%usbserial.devicedesc%;USB Seri Cihaz	USB Seri Cihaz (COM5)	0000.0000.0000.002.000.000.000.000.000
	13	VID_8087&PID_0ACA	05022016	@usb.inf,%usb\composite.devicedesc%;USB Composite Device	N/A	Port_#0002.Hub_#0002
	14	ROOT_HUB30	7&9e81f02&0&0	@usbhub3.inf,%usbhub3.roothubdevicedesc%;USB Root Hub (USB 3.0)	N/A	N/A
	15	ROOT_HUB30	5&2078b4e3&0&0	@usbhub3.inf,%usbhub3.roothubdevicedesc%;USB Root Hub (USB 3.0)	N/A	N/A
	16	ROOT_HUB30	4&17877d47&0&0	@usbhub3.inf,%usbhub3.roothubdevicedesc%;USB Root Hub (USB 3.0)	N/A	N/A
	17	VID_346D&PID_5678	4713691254353567775	@usbstor.inf,%genericbulkonly.devicedesc%;USB Mass Storage Device	N/A	Port_#0004.Hub_#0001
	18	ROOT_HUB30	7&3a9bf513&0&0	@usbhub3.inf,%usbhub3.roothubdevicedesc%;USB Root Hub (USB 3.0)	N/A	N/A
	19	VID_0204&PID_6025	MSFT30WD-...	@uaspstor.inf,%generic.devicedesc%;USB Attached SCSI (UAS) Mass Stora...	N/A	Port_#0014.Hub_#0001
	20	VID_2537&PID_1068	0123456789ABCDE	@usbstor.inf,%genericbulkonly.devicedesc%;USB Mass Storage Device	N/A	Port_#0013.Hub_#0001
	21	VID_06CB&PID_00DF	7ff461c4fd26	@oem26.inf,%wbdiusbdevicename_fs7605%;Synaptics FS7605 Touch ...	N/A	Port_#0007.Hub_#0001
	22	VID_06CB&PID_00DF	13d9fd0c8cb5	@oem26.inf,%wbdiusbdevicename_fs7605%;Synaptics FS7605 Touch ...	N/A	Port_#0007.Hub_#0001
	23	VID_8888&PID_7777	MSFT3022222222222222...	@uaspstor.inf,%generic.devicedesc%;USB Attached SCSI (UAS) Mass Stora...	N/A	Port_#0014.Hub_#0001

## C. Kurulu Programlar Analizi (SOFTWARE Hive):

- **Kaynak:** Windows Kayıt Defteri SOFTWARE hive dosyası.
- **Hedef Bilgiler:** Program Adı, Yayıncı, Sürüm, Kurulum Tarihi.
- **Yöntem:**

python-registry ile SOFTWARE hive dosyası açılır. Hem 64-bit (Microsoft\Windows\CurrentVersion\Uninstall) hem de 32-bit (Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall) programların kayıtlarının bulunduğu yollar taranır. Her bir Uninstall anahtarı altındaki alt anahtarlar (program GUID'leri veya isimleri) gezilir. Her program anahtarından (prog\_key) DisplayName, Publisher, DisplayVersion ve InstallDate değerleri okunur (prog\_key.value(...)). DisplayName değeri yoksa veya boşsa, bu kayıt atlanır. InstallDate değeri (genellikle YYYYMMDD formatında bir string) okunur ve Python'un datetime.strptime fonksiyonu ile tarih objesine, ardından strftime('%Y-%m-%d') ile YYYY-MM-DD formatında bir string'e çevrilir. Formatlama başarısız olursa orijinal değer kullanılır. Sonuçlar, ilgili sütunları içeren bir pandas DataFrame olarak parse\_installed\_programs fonksiyonu tarafından döndürülür ve Program Adı'na göre sıralanır.

Windows Forensic Artifact Analyzer				
Dosya				
Oturum Logları				
USB Depolama Aygıtları				
Tüm USB Aygıtları				
Kurulu Programlar				
Çalıştırılan Programlar (UserAssist)				
Ağ Geçmişi				
Program Adı	Yayıncı	Sürüm	Kurulum Tarihi	
1 DB Browser for SQLite	DB Browser for SQLite Team	3.13.1	2025-10-08	
2 Exterro FTK Imager	Exterro	4.7.3.81	2025-10-08	
3 GNS3	GNS3 Technology Inc.	2.2.54	N/A	
4 Git	The Git Development Community	2.51.0.2	2025-10-12	
5 Google Chrome	Google LLC	141.0.7390.108	2025-10-18	
6 HP Connection Optimizer	HP Inc	2.0.20.0	2025-10-04	
7 HP PC Hardware Diagnostics UEFI	Şirketinizin Adı	10.2.0.0	2025-10-04	
8 HP Thunderbolt Dock G2 Firmware Installer	HP Inc.	1.0.71.1	2025-10-04	
9 Herramientas de corrección de Microsoft Office 2016: ...	Microsoft Corporation	16.0.4266.1001	2025-10-08	
10 Microsoft .NET Host - 9.0.10 (x64)	Microsoft Corporation	72.40.40927	2025-10-18	
11 Microsoft .NET Host FX Resolver - 9.0.10 (x64)	Microsoft Corporation	72.40.40927	2025-10-18	
12 Microsoft .NET Runtime - 9.0.10 (x64)	Microsoft Corporation	72.40.40927	2025-10-18	
13 Microsoft Access MUI (English) 2016	Microsoft Corporation	16.0.4266.1001	2025-10-08	
14 Microsoft Access Setup Metadata MUI (English) 2016	Microsoft Corporation	16.0.4266.1001	2025-10-08	
15 Microsoft DCE MUI (English) 2016	Microsoft Corporation	16.0.4266.1001	2025-10-08	

## D. Çalıştırılan Programlar Analizi (NTUSER.DAT Hive):

- **Kaynak:** Kullanıcıya özel NTUSER.DAT hive dosyası.
- **Hedef Bilgiler:** Program Adı, Yayıncı, Sürüm, Kurulum Tarihi.
- **Yöntem:**

python-registry ile NTUSER.DAT dosyası açılır.

Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist anahtarı açılır. Bu anahtar altındaki her bir {GUID} anahtarı gezilir ve onun altındaki Count alt anahtarı açılır (guid\_key.subkey("Count")). Count anahtarının içindeki her bir değer (value) işlenir. Değerin adı (value.name()) alınır, (Default) değilse codecs.decode(..., 'rot\_13') ile ROT13 şifresi çözülür. Değerin ikili (binary) verisi (value.value()) alınır. Verinin uzunluğu (data\_len) kontrol edilir:

- Eğer 72 byte veya daha uzunsa (modern format): Çalıştırma Sayısı (Run Count) struct.unpack('<I', binary\_data[4:8]) ile 4. ofsetten, Odaklanma Sayısı (Focus Count) struct.unpack('<I', binary\_data[8:12]) ile 8. ofsetten, 64-bit Windows FILETIME zaman damgası struct.unpack('<Q', binary\_data[60:68]) ile 60. ofsetten okunur.

- Eğer 16 byte veya daha uzunsa (eski format): Çalıştırma Sayısı 4. ofsetten okunur ve Windows XP/7'deki mantığa göre (sayaç 5'ten başladığı için) run\_count = run\_count\_raw - 5 hesaplaması yapılır. Zaman damgası son 8 bayttan okunur (binary\_data[-8:]).

- Diğer uzunluktaki veriler atlanır.

Okunan FILETIME zaman damgası (filetime\_raw), filetime\_to\_datetime yardımcı fonksiyonu ile UTC datetime objesine çevrilir (Epoch farkı ve nanosaniye hesaplaması yapılarak). Sonuçlar, ilgili sütunları içeren bir pandas DataFrame olarak parse\_user\_assist fonksiyonu tarafından döndürülür ve Son Çalıştırma Zamanı'na göre sıralanır.

Oturum Logları  
USB Depolama Aygıtları  
Tüm USB Aygıtları  
Kurulu Programlar  
Çalıştırılan Programlar (UserAssist)  
Ağ Geçmişi

	Program Adı (Deşifre Edilmiş)	Çalıştırma Sayısı	Odaklanma Sayısı	Son Çalıştırma (UTC)
1	C:\Users\HP\Downloads\program setups\cmdex\Cmdex.exe	5	7	2025-10-19 17:58:48
2	C:\Users\HP\Desktop\Cmder - Kısayol.Ink	5	0	2025-10-19 17:58:48
3	{1AC14E77-02E7-4E5D-B744-2EB1AE5198B7}\notepad.exe	1	0	2025-10-19 16:14:54
4	C:\Windows\System32\config\SYSTEM - Shortcut.Ink	1	0	2025-10-19 16:14:52
5	Microsoft.Windows.Photos_8wekyb3d8bbwe!App	6	0	2025-10-19 16:12:49
6	{1AC14E77-02E7-4E5D-B744-2EB1AE5198B7}\eventvwr.exe	1	0	2025-10-19 10:35:19
7	C:\Users\HP\Desktop\Visual Studio Code.Ink	1	0	2025-10-19 09:52:49
8	Microsoft.VisualStudioCode	1	89	2025-10-19 09:52:49
9	{1AC14E77-02E7-4E5D-B744-2EB1AE5198B7}\DriverStore\FileRepository\realtekservice.inf_amd64_da83c4764fd71b24\RtkAudUService64.exe	1	0	2025-10-19 09:45:51
10	Chrome	7	0	2025-10-19 07:31:23
11	C:\Users\HP\Downloads\program setups\RegistryExplorer\RegistryExplorer.exe	0	18	2025-10-18 08:00:09
12	{6D809377-6AF0-444B-8957-A3773F02200E}\WinRAR\WinRAR.exe	0	0	2025-10-18 07:53:00
13	{F38BF404-1D43-42F2-9305-67DE0B28FC23}\regedit.exe	0	0	2025-10-18 07:11:44
14	{0139D44E-6AFE-49F2-8690-3DAFCAE6FFB8}\Administrative Tools\Registry Editor.Ink	0	0	2025-10-18 06:23:33
15	{6D809377-6AF0-444B-8957-A3773F02200E}\DB Browser for SQLite\DB Browser for SQLite.exe	0	0	2025-10-18 05:43:18
16	C:\Users\Public\Desktop\DB Browser (SQLite).Ink	0	0	2025-10-18 05:43:18
17	Microsoft.Windows.Explorer	0	39	2025-10-18 05:42:01
18	{9E3995AB-1F9C-4F13-B827-48B24B6C7174}\TaskBar\File Explorer.Ink	0	0	2025-10-18 05:42:01
19	C:\Users\HP\Desktop\Tor Browser.Ink	0	0	2025-10-17 17:28:37
20	C:\Users\HP\Desktop\Tor Browser\Browser\firefox.exe	0	0	2025-10-17 17:28:37
21	C:\Users\Public\Desktop\Oracle VirtualBox.Ink	0	0	2025-10-17 17:24:14
22	{6D809377-6AF0-444B-8957-A3773F02200E}\Oracle\VirtualBox\VirtualBox.exe	0	0	2025-10-17 17:24:14
23	{6D809377-6AF0-444B-8957-A3773F02200E}\Wireshark\Wireshark.exe	0	0	2025-10-17 17:19:19
24	C:\Users\HP\Desktop\Wireshark.Ink	0	0	2025-10-17 17:19:19

## E. Ağ Geçmişi Analizi (SOFTWARE Hive):

- **Kaynak:** Windows Kayıt Defteri SOFTWARE hive dosyası.
- **Hedef Bilgiler:** Geçmişte bağlanılan ağların adı (SSID), İlk Bağlantı Zamanı.
- **Yöntem:**  
python-registry ile SOFTWARE hive dosyası açılır. Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles anahtarı açılır. Bu anahtar altındaki her bir {GUID} alt anahtarı (ağ profili) gezilir. Her profilden ProfileName (Ağ Adı) değeri okunur. DateCreated değeri okunur. Bu değer REG\_BINARY tipinde ve 16 byte uzunluğunda olduğu tespit edilmiştir (SYSTEMTIME formatı). Bu 16 byte'lık veri, struct.unpack('<HHHHHHHH', raw\_value) ile 8 adet 2 byte'lık unsigned short değerine (yıl, ay, gün, saat, dakika, saniye, milisaniye) ayrıştırılır. Bu değerler kullanılarak systemtime\_to\_datetime yardımcı fonksiyonu ile UTC datetime objesi oluşturulur. Sonuçlar, "Ağ Adı (SSID)", "İlk Bağlantı (UTC)", "Profil Yolu (GUID)" sütunlarını içeren bir pandas DataFrame olarak parse\_network\_list fonksiyonu tarafından döndürülür ve İlk Bağlantı Zamanı'na göre sıralanır.

Windows Forensic Artifact Analyzer			
Dosya			
Oturum Logları USB Depolama Aygıtları Tüm USB Aygıtları Kurulu Programlar Çalıştırılan Programlar (UserAssist) Ağ Geçmişi	Ağ Adı (SSID)	İlk Bağlantı (UTC)	Profil Yolu (GUID)
1	TurkTelekom_ZY146A	2025-10-09 10:25:24	{A0FB5251-64E4-4FFF-8371-1E49165702D0}
2	TurkTelekom_ZTK7HU	2025-10-07 15:20:58	{E3177D2A-2BD2-47AB-9754-5D2FA5F50F28}
3	Ekider Müdüriyet	2025-10-07 15:13:51	{5C4D4BBC-14BA-46B3-97DA-CA6AE03A0450}
4	FU_KUTUPHANE	2025-10-07 12:02:31	{98774BA8-D4D5-439E-9959-EEA864102C58}
5	FU_WIFI	2025-10-07 09:07:43	{B740A12B-1B57-4E24-BA3E-611BC07BD218}
6	Ekider Müdüriyet_5G	2025-10-06 16:47:45	{F625D7CB-8E7C-41F5-9F9A-145CFAFD4EF2}
7	Redmi Note 8 Pro	2025-10-06 13:06:55	{AC455675-54F1-405E-BF03-614703FF9560}
8	Ertam#LAB	2025-10-06 12:55:12	{EAD34542-5B73-4CAF-B29B-0F1AD99836F1}
9	Turk 5GHz	2025-10-04 18:14:45	{C9195D99-E17E-4304-9B30-A8B434E25F57}
10	YepYeniBilgisayar_5G	2025-05-22 17:15:13	{8484B24D-6C37-4A2C-B765-08BC2588CE8D}
11	ATOM_BILISIM	2025-03-13 15:07:21	{35F78E77-FE55-4C7D-B909-0BEC2A084EFD}

## 2.4. Görsel Arayüz (GUI) Tasarımı (PyQt5):

Yazılımın kullanıcı arayüzü, PyQt5 kütüphanesi kullanılarak geliştirilmiştir. Bu kütüphane, platformdan bağımsız, modern ve esnek masaüstü uygulamaları oluşturmak için tercih edilmiştir. Arayüzün ana bileşenleri şunlardır:

- **Ana Pencere (QMainWindow):** Uygulamanın temel çerçevesini oluşturur, başlık çubuğu ve menü çubuğunu içerir.
- **Menü Çubuğu (QMenuBar):** Üst kısımda yer alır. "Dosya" menüsü altında "Vaka Klasörü Yükle..." ve "Çıkış" seçeneklerini sunar.
- **Sol Panel (QListWidget):** Analiz kategorilerini ("Oturma Logları", "USB Depolama Aygıtları" vb.) listeleyen bir pencere öğesidir. Kullanıcı bu listeden bir öğeye tıkladığında, currentItemChanged sinyali tetiklenir ve displayData fonksiyonu çağrılır.
- **Sağ Panel (QTableWidget):** Seçilen kategoriye ait analiz sonuçlarını tablo formatında gösteren ana pencere öğesidir. Satırların dönüşümlü renklendirilmesi (setAlternatingRowColors), hücre içeriğinin düzenlenmesinin engellenmesi (setEditTriggers) ve sütun başlıklarına tıklayarak sıralama yapılabilmesi (setSortingEnabled) gibi özellikler etkinleştirilmiştir.
- **Durum Çubuğu (QStatusBar):** Pencerenin en altında yer alır ve kullanıcıya o an yapılan işlem (dosya yükleme, analiz, veri gösterme) veya programın genel durumu hakkında bilgi verir (showMessage).

Oturum Logları		Timestamp	Olay	Kullanıcı Adı	Oturum Türü	Kaynak IP
USB Depolama Aygıtları	1	2025-10-08 13:15:43	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Tüm USB Aygıtları	2	2025-10-08 13:15:43	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Kurulu Programlar	3	2025-10-08 13:15:49	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Çalıştırılan Programlar (UserAssist)	4	2025-10-08 13:17:00	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
Ağ Geçmişi	5	2025-10-08 13:17:53	(4647) Oturum Kullanıcı Tarafından Kapatıldı	HP	N/A	N/A
	6	2025-10-08 13:46:36	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	7	2025-10-08 14:02:25	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	8	2025-10-08 14:03:14	(4624) Başarılı Oturum Açma	SYSTEM	(0) Sistem	-
	9	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	UMFD-0	(2) İnteraktif	-
	10	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	11	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	Local Service	(5) Hizmet	-
	12	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	NETWORK SERVICE	(5) Hizmet	-
	13	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	14	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	UMFD-1	(2) İnteraktif	-
	15	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	16	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	17	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	18	2025-10-08 14:03:15	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	19	2025-10-08 14:03:16	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	20	2025-10-08 14:03:16	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	21	2025-10-08 14:03:16	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	22	2025-10-08 14:03:16	(4624) Başarılı Oturum Açma	SYSTEM	(5) Hizmet	-
	23	2025-10-08 14:03:16	(4624) Başarılı Oturum Açma	DWM-1	(2) İnteraktif	-
	24	2025-10-08 14:03:16	(4624) Başarılı Oturum Açma	DWM-1	(2) İnteraktif	-

registry\_parser.py • main\_gui.py

registry > registry\_parser.py > ...

```
1 import Evtx.Evtx as evt
2 import pandas as pd
3 import xml.etree.ElementTree as ET
4 from pathlib import Path
5 from Registry import Registry
6 import struct
7 import codecs
8 from datetime import datetime, timedelta, UTC
9
```

```
10
11 # --- FONKSİYON 1: OTURUM LOGLARI
12 > def parse_security_log(evt_file_path): ...
80
```

```
81 # --- FONKSİYON 2: USB ANALİZİ
82 > def parse_usb_devices(system_hive_path): ...
160
```

```
161 # --- FONKSİYON 3: KURULU PROGRAMLAR
162 > def parse_installed_programs(software_hive_path): ...
215
```

```
216 # --- FONKSİYON 4: ÇALIŞTIRILAN PROGRAMLAR
217 > def parse_user_assist(ntuser_dat_path): ...
283
```

```
284 # --- FONKSİYON 5: AĞ (WIFI) BİLGİLERİ
285 > def parse_network_list(software_hive_path): ...
348
```

```
349
350
351
352 # --- if __name__ == '__main__':
353 > if __name__ == '__main__': ...
```

registry\_parser.py main\_gui.py X

registry > main\_gui.py > ...

```
1 import sys
2 import os
3 from pathlib import Path
4 import pandas as pd
5 from PyQt5.QtWidgets import (QApplication, QMainWindow, QAction, QFileDialog,
6                               QListWidget, QTableWidgetItem,
7                               QHBoxLayout, QWidget, QVBoxLayout, QAbstractItemView,
8                               QStatusBar, QLabel, QMessageBox)
9 from PyQt5.QtCore import Qt
10
```

```
11 # Bizim analiz fonksiyonlarımızı içeren dosyayı import et
12 # (main_gui.py ve registry_parser.py aynı klasörde olmalı)
13 import registry_parser
14
```

```
15 class ForensicAnalyzerApp(QMainWindow):
```

```
16 > def __init__(self): ...
26
```

```
27 > def initUI(self): ...
77
```

```
78 > def loadCaseFolder(self): ...
151
```

```
152
153 # displayData fonksiyonu aynı kalabilir
154 > def displayData(self, current_item): ...
181
```

```
182
183 # --- Uygulamayı Başlat ---
184 if __name__ == '__main__':
```

```
185     app = QApplication(sys.argv)
186     mainWin = ForensicAnalyzerApp()
187     mainWin.show()
188     sys.exit(app.exec_())
```



## Kullanıcı Etkileşimi ve Veri Akışı:

1. Kullanıcı programı çalıştırır (python main\_gui.py).
2. "Dosya -> Vaka Klasörü Yükle..." menüsünü seçer.
3. QFileDialog.getExistingDirectory ile bir klasör seçer.
4. loadCaseFolder fonksiyonu tetiklenir.
5. Seçilen klasörde gerekli hive/log dosyalarının olup olmadığı kontrol edilir. Eksik varsa uyarı verilir.
6. registry\_parser modülündeki ilgili parse\_\* fonksiyonları sırayla çağrılır. Her fonksiyonun sonucu (bir pandas DataFrame) self.data\_frames sözlüğünde ilgili kategori adıyla saklanır. Bu işlemler sırasında durum çubuğu güncellenir.
7. Tüm analizler bittikten sonra, sol listedeki ilk kategori (setCurrentRow(0)) otomatik olarak seçilir.
8. displayData fonksiyonu tetiklenir. Seçilen kategoriye ait DataFrame self.data\_frames sözlüğünden alınır.
9. DataFrame boş değilse, QTableWidgetItem'ın satır/sütun sayısı ayarlanır, başlıklar (setHorizontalHeaderLabels) yazılır.
10. DataFrame'deki her bir hücre değeri okunur, pd.Timestamp ise YYYY-MM-DD HH:MM:SS formatına çevrilir, None veya NaT ise boş string'e çevrilir ve QTableWidgetItem olarak tabloya eklenir (setItem).
11. Son olarak, sütun genişlikleri içeriğe göre ayarlanır (resizeColumnsToContents).
12. Kullanıcı sol listeden farklı bir kategoriye tıkladığında 8-11. adımlar o kategori için tekrarlanır.