**SecuNik LogX - Complete Project Structure**

**📁 FINAL PROJECT STRUCTURE**

**SecuNik-LogX/**

**│**

**├── backend/**

**│ ├── main.py # FastAPI app entry point**

**│ ├── config.py # Configuration management**

**│ ├── requirements.txt # Python dependencies**

**│ ├── .env.example # Environment template**

**│ │**

**│ ├── api/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── upload.py # File upload endpoints**

**│ │ ├── analysis.py # Analysis endpoints**

**│ │ ├── history.py # History endpoints**

**│ │ ├── rules.py # Rule management endpoints**

**│ │ ├── virustotal.py # VirusTotal integration**

**│ │ └── websocket.py # Real-time updates**

**│ │**

**│ ├── core/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── parser\_factory.py # Parser selection logic**

**│ │ ├── analysis\_engine.py # Main analysis orchestrator**

**│ │ ├── storage\_manager.py # File storage operations**

**│ │ └── file\_identifier.py # File type detection**

**│ │**

**│ ├── parsers/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── base\_parser.py # Abstract parser interface**

**│ │ │**

**│ │ ├── logs/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── syslog\_parser.py # Linux syslog**

**│ │ │ ├── windows\_event\_parser.py # Windows event logs**

**│ │ │ ├── apache\_parser.py # Apache access/error logs**

**│ │ │ ├── nginx\_parser.py # Nginx logs**

**│ │ │ ├── iis\_parser.py # IIS logs**

**│ │ │ ├── auth\_log\_parser.py # Auth logs**

**│ │ │ ├── firewall\_parser.py # Firewall logs**

**│ │ │ ├── application\_parser.py # Generic app logs**

**│ │ │ └── json\_log\_parser.py # JSON formatted logs**

**│ │ │**

**│ │ ├── network/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── pcap\_parser.py # PCAP files**

**│ │ │ ├── pcapng\_parser.py # PCAPNG files**

**│ │ │ ├── netflow\_parser.py # NetFlow data**

**│ │ │ ├── zeek\_parser.py # Zeek/Bro logs**

**│ │ │ ├── suricata\_parser.py # Suricata logs**

**│ │ │ ├── snort\_parser.py # Snort logs**

**│ │ │ ├── tcpdump\_parser.py # TCPDump output**

**│ │ │ └── dns\_log\_parser.py # DNS query logs**

**│ │ │**

**│ │ ├── system/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── registry\_parser.py # Windows Registry**

**│ │ │ ├── memory\_dump\_parser.py # Memory dumps**

**│ │ │ ├── process\_list\_parser.py # Process dumps**

**│ │ │ ├── prefetch\_parser.py # Windows Prefetch**

**│ │ │ ├── mft\_parser.py # NTFS MFT**

**│ │ │ ├── usnjrnl\_parser.py # USN Journal**

**│ │ │ ├── evt\_evtx\_parser.py # EVT/EVTX files**

**│ │ │ ├── lnk\_parser.py # Windows shortcuts**

**│ │ │ └── scheduled\_task\_parser.py # Task scheduler**

**│ │ │**

**│ │ ├── mobile/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── android\_logcat\_parser.py # Android logs**

**│ │ │ ├── ios\_syslog\_parser.py # iOS logs**

**│ │ │ ├── whatsapp\_parser.py # WhatsApp exports**

**│ │ │ ├── sms\_parser.py # SMS backups**

**│ │ │ └── call\_log\_parser.py # Call logs**

**│ │ │**

**│ │ ├── cloud/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── aws\_cloudtrail\_parser.py # AWS logs**

**│ │ │ ├── azure\_activity\_parser.py # Azure logs**

**│ │ │ ├── gcp\_stackdriver\_parser.py # GCP logs**

**│ │ │ ├── office365\_parser.py # O365 audit logs**

**│ │ │ └── gsuite\_parser.py # Google Workspace**

**│ │ │**

**│ │ ├── email/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── eml\_parser.py # EML files**

**│ │ │ ├── msg\_parser.py # MSG files**

**│ │ │ ├── mbox\_parser.py # MBOX archives**

**│ │ │ ├── pst\_parser.py # PST files**

**│ │ │ └── smtp\_parser.py # SMTP logs**

**│ │ │**

**│ │ ├── documents/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── pdf\_parser.py # PDF analysis**

**│ │ │ ├── office\_parser.py # DOCX, XLSX, PPTX**

**│ │ │ ├── rtf\_parser.py # RTF files**

**│ │ │ ├── odt\_parser.py # OpenDocument**

**│ │ │ └── html\_parser.py # HTML files**

**│ │ │**

**│ │ ├── archives/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── zip\_parser.py # ZIP files**

**│ │ │ ├── rar\_parser.py # RAR files**

**│ │ │ ├── sevenz\_parser.py # 7Z files**

**│ │ │ ├── tar\_parser.py # TAR/TAR.GZ**

**│ │ │ └── iso\_parser.py # ISO images**

**│ │ │**

**│ │ ├── database/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── sqlite\_parser.py # SQLite files**

**│ │ │ ├── mysql\_dump\_parser.py # MySQL dumps**

**│ │ │ ├── postgres\_dump\_parser.py # PostgreSQL dumps**

**│ │ │ └── mongodb\_export\_parser.py # MongoDB exports**

**│ │ │**

**│ │ ├── forensics/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── disk\_image\_parser.py # DD, E01 images**

**│ │ │ ├── vmdk\_parser.py # VMware disks**

**│ │ │ ├── vhd\_parser.py # Hyper-V disks**

**│ │ │ ├── forensic\_image\_parser.py # FTK, EnCase**

**│ │ │ └── timeline\_parser.py # Plaso timelines**

**│ │ │**

**│ │ ├── code/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── source\_code\_parser.py # Source code analysis**

**│ │ │ ├── binary\_parser.py # EXE, DLL, ELF**

**│ │ │ ├── script\_parser.py # PS1, SH, BAT**

**│ │ │ ├── config\_parser.py # Config files**

**│ │ │ └── secrets\_scanner.py # API keys, passwords**

**│ │ │**

**│ │ ├── structured/**

**│ │ │ ├── \_\_init\_\_.py**

**│ │ │ ├── csv\_parser.py # CSV files**

**│ │ │ ├── json\_parser.py # JSON files**

**│ │ │ ├── xml\_parser.py # XML files**

**│ │ │ ├── yaml\_parser.py # YAML files**

**│ │ │ └── ini\_parser.py # INI files**

**│ │ │**

**│ │ └── generic/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── text\_parser.py # Plain text**

**│ │ ├── hex\_parser.py # Hex dumps**

**│ │ └── binary\_parser.py # Unknown binary**

**│ │**

**│ ├── analyzers/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── yara\_analyzer.py # YARA rule engine**

**│ │ ├── sigma\_analyzer.py # Sigma rule engine**

**│ │ ├── mitre\_analyzer.py # MITRE ATT&CK mapping**

**│ │ ├── stix\_analyzer.py # STIX/TAXII integration**

**│ │ ├── ai\_analyzer.py # OpenAI integration**

**│ │ ├── pattern\_analyzer.py # Pattern detection**

**│ │ ├── anomaly\_detector.py # Statistical anomalies**

**│ │ ├── ioc\_extractor.py # IOC extraction**

**│ │ └── correlation\_engine.py # Event correlation**

**│ │**

**│ ├── rules/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── rule\_manager.py # Rule CRUD operations**

**│ │ ├── rule\_validator.py # Rule syntax validation**

**│ │ ├── rule\_importer.py # Import rule packs**

**│ │ └── custom\_rule\_engine.py # Custom rule processor**

**│ │**

**│ ├── integrations/**

**│ │ ├── \_\_init\_\_.py**

**│ │ ├── virustotal\_client.py # VT API client**

**│ │ ├── openai\_client.py # OpenAI client**

**│ │ └── threat\_intel\_client.py # Other threat intel**

**│ │**

**│ └── utils/**

**│ ├── \_\_init\_\_.py**

**│ ├── file\_utils.py # File operations**

**│ ├── hash\_utils.py # File hashing**

**│ ├── time\_utils.py # Timestamp parsing**

**│ ├── encoding\_utils.py # Encoding detection**

**│ └── validation\_utils.py # Input validation**

**│**

**├── frontend/**

**│ ├── package.json # Dependencies**

**│ ├── vite.config.js # Vite configuration**

**│ ├── tailwind.config.js # Tailwind configuration**

**│ ├── index.html # Entry HTML**

**│ ├── .env.example # Environment template**

**│ │**

**│ ├── public/**

**│ │ ├── favicon.ico**

**│ │ └── logo.svg**

**│ │**

**│ ├── src/**

**│ │ ├── main.jsx # React entry point**

**│ │ ├── App.jsx # Main app component**

**│ │ ├── index.css # Global styles**

**│ │ │**

**│ │ ├── pages/**

**│ │ │ ├── Upload.jsx # File upload page**

**│ │ │ ├── Analysis.jsx # Analysis dashboard**

**│ │ │ ├── History.jsx # Analysis history**

**│ │ │ ├── Rules.jsx # Rule management**

**│ │ │ └── Settings.jsx # App settings**

**│ │ │**

**│ │ ├── components/**

**│ │ │ ├── layout/**

**│ │ │ │ ├── Sidebar.jsx # Navigation sidebar**

**│ │ │ │ ├── Header.jsx # Top header**

**│ │ │ │ └── Layout.jsx # Main layout wrapper**

**│ │ │ │**

**│ │ │ ├── upload/**

**│ │ │ │ ├── FileDropzone.jsx # Drag-drop area**

**│ │ │ │ ├── FileList.jsx # Uploaded files list**

**│ │ │ │ └── UploadProgress.jsx # Progress indicator**

**│ │ │ │**

**│ │ │ ├── analysis/**

**│ │ │ │ ├── EventsTimeline.jsx # Events tab**

**│ │ │ │ ├── IOCsTable.jsx # IOCs tab with VT**

**│ │ │ │ ├── PatternsView.jsx # Patterns tab**

**│ │ │ │ ├── AnomaliesView.jsx # Anomalies tab**

**│ │ │ │ └── AnalysisOverview.jsx # Summary view**

**│ │ │ │**

**│ │ │ ├── rules/**

**│ │ │ │ ├── RuleEditor.jsx # Rule creation/editing**

**│ │ │ │ ├── RuleList.jsx # Rules table**

**│ │ │ │ ├── RuleImporter.jsx # Import rules**

**│ │ │ │ └── RuleTester.jsx # Test rules**

**│ │ │ │**

**│ │ │ └── common/**

**│ │ │ ├── Button.jsx # Reusable button**

**│ │ │ ├── Card.jsx # Card component**

**│ │ │ ├── Modal.jsx # Modal dialog**

**│ │ │ ├── Table.jsx # Data table**

**│ │ │ ├── Tabs.jsx # Tab component**

**│ │ │ └── LoadingSpinner.jsx # Loading indicator**

**│ │ │**

**│ │ ├── services/**

**│ │ │ ├── api.js # Axios config**

**│ │ │ ├── uploadService.js # Upload API**

**│ │ │ ├── analysisService.js # Analysis API**

**│ │ │ ├── historyService.js # History API**

**│ │ │ ├── rulesService.js # Rules API**

**│ │ │ └── websocketService.js # WebSocket connection**

**│ │ │**

**│ │ ├── hooks/**

**│ │ │ ├── useWebSocket.js # WebSocket hook**

**│ │ │ ├── useAnalysis.js # Analysis state**

**│ │ │ └── useTheme.js # Dark theme**

**│ │ │**

**│ │ └── utils/**

**│ │ ├── constants.js # App constants**

**│ │ ├── formatters.js # Data formatters**

**│ │ └── helpers.js # Helper functions**

**│**

**├── storage/ # File-based storage**

**│ ├── uploads/ # Original uploaded files**

**│ ├── parsed/ # Parsed data (JSON)**

**│ ├── analysis/ # Analysis results**

**│ ├── temp/ # Temporary files**

**│ └── history.json # Analysis history index**

**│**

**├── rules/ # Detection rules**

**│ ├── yara/**

**│ │ ├── malware/ # Malware rules**

**│ │ ├── exploits/ # Exploit rules**

**│ │ ├── suspicious/ # Suspicious patterns**

**│ │ └── index.yar # Main YARA file**

**│ │**

**│ ├── sigma/**

**│ │ ├── windows/ # Windows rules**

**│ │ ├── linux/ # Linux rules**

**│ │ ├── network/ # Network rules**

**│ │ └── cloud/ # Cloud rules**

**│ │**

**│ ├── mitre/**

**│ │ ├── techniques.json # ATT&CK techniques**

**│ │ └── mappings.json # Rule mappings**

**│ │**

**│ ├── custom/**

**│ │ └── user\_rules.json # User-created rules**

**│ │**

**│ └── index.json # Rule registry**

**│**

**├── docker/**

**│ ├── Dockerfile.backend # Backend container**

**│ ├── Dockerfile.frontend # Frontend container**

**│ └── nginx.conf # Nginx config**

**│**

**├── scripts/**

**│ ├── setup.sh # Initial setup**

**│ ├── download\_rules.py # Download rule packs**

**│ └── test\_parsers.py # Parser testing**

**│**

**├── tests/**

**│ ├── test\_files/ # Sample files for testing**

**│ ├── test\_parsers.py # Parser tests**

**│ └── test\_analyzers.py # Analyzer tests**

**│**

**├── docs/**

**│ ├── README.md # Main documentation**

**│ ├── PARSERS.md # Parser documentation**

**│ ├── API.md # API documentation**

**│ └── DEPLOYMENT.md # Deployment guide**

**│**

**├── .gitignore**

**├── .dockerignore**

**├── docker-compose.yml # Full stack deployment**

**├── LICENSE**

**└── README.md**

**📋 Implementation Handoff Plan**

**BATCH 1: Core Backend Infrastructure (10 files)**

1. **backend/main.py**
2. **backend/config.py**
3. **backend/requirements.txt**
4. **backend/core/storage\_manager.py**
5. **backend/core/file\_identifier.py**
6. **backend/api/upload.py**
7. **backend/api/analysis.py**
8. **backend/utils/file\_utils.py**
9. **backend/utils/hash\_utils.py**
10. **docker-compose.yml**

**BATCH 2: Parser Foundation (10 files)**

1. **backend/parsers/base\_parser.py**
2. **backend/core/parser\_factory.py**
3. **backend/parsers/logs/syslog\_parser.py**
4. **backend/parsers/logs/windows\_event\_parser.py**
5. **backend/parsers/network/pcap\_parser.py**
6. **backend/parsers/structured/json\_parser.py**
7. **backend/parsers/structured/csv\_parser.py**
8. **backend/parsers/archives/zip\_parser.py**
9. **backend/parsers/generic/text\_parser.py**
10. **backend/parsers/\_\_init\_\_.py**

**BATCH 3: Advanced Parsers (10 files)**

1. **backend/parsers/logs/apache\_parser.py**
2. **backend/parsers/network/netflow\_parser.py**
3. **backend/parsers/system/registry\_parser.py**
4. **backend/parsers/email/eml\_parser.py**
5. **backend/parsers/documents/pdf\_parser.py**
6. **backend/parsers/forensics/disk\_image\_parser.py**
7. **backend/parsers/cloud/aws\_cloudtrail\_parser.py**
8. **backend/parsers/mobile/android\_logcat\_parser.py**
9. **backend/parsers/code/source\_code\_parser.py**
10. **backend/parsers/database/sqlite\_parser.py**

**BATCH 4: Analysis Engines (10 files)**

1. **backend/core/analysis\_engine.py**
2. **backend/analyzers/yara\_analyzer.py**
3. **backend/analyzers/sigma\_analyzer.py**
4. **backend/analyzers/mitre\_analyzer.py**
5. **backend/analyzers/ai\_analyzer.py**
6. **backend/analyzers/ioc\_extractor.py**
7. **backend/analyzers/pattern\_analyzer.py**
8. **backend/integrations/virustotal\_client.py**
9. **backend/integrations/openai\_client.py**
10. **backend/api/websocket.py**

**BATCH 5: Rule Management & APIs (10 files)**

1. **backend/rules/rule\_manager.py**
2. **backend/rules/rule\_validator.py**
3. **backend/api/rules.py**
4. **backend/api/history.py**
5. **backend/api/virustotal.py**
6. **backend/analyzers/anomaly\_detector.py**
7. **backend/analyzers/correlation\_engine.py**
8. **backend/utils/time\_utils.py**
9. **backend/utils/encoding\_utils.py**
10. **backend/.env.example**

**BATCH 6: Frontend Foundation (10 files)**

1. **frontend/package.json**
2. **frontend/vite.config.js**
3. **frontend/tailwind.config.js**
4. **frontend/src/main.jsx**
5. **frontend/src/App.jsx**
6. **frontend/src/index.css**
7. **frontend/src/components/layout/Layout.jsx**
8. **frontend/src/components/layout/Sidebar.jsx**
9. **frontend/src/services/api.js**
10. **frontend/index.html**

**BATCH 7: Frontend Pages (10 files)**

1. **frontend/src/pages/Upload.jsx**
2. **frontend/src/pages/Analysis.jsx**
3. **frontend/src/pages/History.jsx**
4. **frontend/src/pages/Rules.jsx**
5. **frontend/src/pages/Settings.jsx**
6. **frontend/src/components/upload/FileDropzone.jsx**
7. **frontend/src/components/analysis/EventsTimeline.jsx**
8. **frontend/src/components/analysis/IOCsTable.jsx**
9. **frontend/src/components/common/Button.jsx**
10. **frontend/src/components/common/Card.jsx**

**BATCH 8: Frontend Components & Services (10 files)**

1. **frontend/src/components/analysis/PatternsView.jsx**
2. **frontend/src/components/analysis/AnomaliesView.jsx**
3. **frontend/src/components/rules/RuleEditor.jsx**
4. **frontend/src/components/rules/RuleList.jsx**
5. **frontend/src/services/uploadService.js**
6. **frontend/src/services/analysisService.js**
7. **frontend/src/services/websocketService.js**
8. **frontend/src/hooks/useWebSocket.js**
9. **frontend/src/utils/constants.js**
10. **frontend/src/utils/formatters.js**

**BATCH 9: Additional Parsers (10 files)**

1. **backend/parsers/logs/nginx\_parser.py**
2. **backend/parsers/logs/firewall\_parser.py**
3. **backend/parsers/network/zeek\_parser.py**
4. **backend/parsers/system/memory\_dump\_parser.py**
5. **backend/parsers/system/prefetch\_parser.py**
6. **backend/parsers/email/msg\_parser.py**
7. **backend/parsers/documents/office\_parser.py**
8. **backend/parsers/archives/rar\_parser.py**
9. **backend/parsers/structured/xml\_parser.py**
10. **backend/parsers/code/secrets\_scanner.py**

**BATCH 10: Final Components & Setup (Remaining files)**

1. **Initial YARA rules**
2. **Initial Sigma rules**
3. **Docker configurations**
4. **Setup scripts**
5. **Documentation files**
6. **Test files**
7. **Remaining utility files**

**🎯 Next Steps**

1. **Confirm Structure: Review and confirm this structure meets all requirements**
2. **Begin Implementation: Start with BATCH 1 (Core Backend Infrastructure)**
3. **Maintain Context: Each batch handoff will include:** 
   * **Complete file contents**
   * **Integration points with other files**
   * **Current progress status**
   * **Next batch preview**

**Ready to begin implementation? Just say "Start BATCH 1" and I'll provide the first 10 files with complete, production-ready code.**