INOTEK, Vol.8 Agustus 2024
ISSN: 2580-3336 (Print) / 2549-7952 (Online)
Url: https://proceeding.unpkediri.ac.id/index.php/inotek/

File Signature Analyzer 2.0 : Identifikasi Ekstensi File Berdasarkan Metadata

Received:
10 Juni 2024
Accepted:
10 Juli 2024
Published:
1 Agustus 2024

[1]Fakhrul Rifqi Darmawan, [2*]Nur Widiyasono, [3]Husni Mubarok
[1,2,3]Informatika, Universitas Siliwangi Tasikmalaya
E-mail: [1]187006097@unsilac.id, [2]nur.widiyasono@unsil.ac.id, [3]husni@unsil.ac.id*

Abstrak— Kejahatan perangkat digital seperti kebocoran informasi, penggelapan uang di bank dan penipuan kartu kredit yang memanfaatkan metode modifikasi *file* untuk melakukan aksinya bisa berdampak buruk hingga merugikan beberapa pihak. Signature *file* atau angka ajaib adalah salah satu teknik ilmu forensik yang membantu proses identifikasi tipe *file*. Proses identifikasi *file* terbagi menjadi dua yaitu identifikasi *file* berdasarkan ekstensi dan identifikasi *file* berdasarkan signature *file* atau angka ajaib. Penelitian ini menyajikan *File Signature Analyzer* 2.0 sebagai pendekatan baru berbasis website dengan akurasi tinggi untuk identifikasi ekstensi *file* otomatis dengan menggunakan metode perulangan. Pengujian aplikasi dilakukan dengan menggunakan dua metode modifikasi yaitu mengubah ekstensi secara acak dan menghilangkan ekstensi dan hasilnya *file* orisinal terdeteksi benar (98%) dan Terdeteksi salah (2%), *file* modifikasi terdeteksi bisa dikembalikan (95%) dan terdeteksi, tidak bisa dikembalikan (5%).

Kata Kunci —– Angka Ajaib, Ekstensi; *File*; Identifikasi; Signature *File*.

Abstract— *Digital device crimes such as information leaks, embezzlement of money from banks and credit card fraud that utilize file modification methods to carry out their actions can have negative impacts and even harm several parties. File signatures or magic numbers are a forensic science technique that helps identify file types. The file identification process is divided into two, namely file identification based on the extension and file identification based on the file signature or magic number. This research presents File Signature Analyzer 2.0 as a new website-based approach with high accuracy for automatic file extension identification using an iterative method. Application testing was carried out using two modification methods, namely changing the extension randomly and removing the extension and the results were that the original file was detected correctly (98%) and detected incorrectly (2%), the detected modified file could be returned (95%) and detected, could not be returned (5%).*

Keywords— *Extension; File; File Signature; Identification; Magic Number.*

Penulis Korespondensi:

Nama Penulis [Nur Widiyasono],
Departemen Penulis [Informatika],
Institusi Penulis [Universitas Siliwangi Tasikmalaya],
Email [nur.widiyasono@unsil.ac.id]
ID Orcid :
Handphone: 089676416325

I. PENDAHULUAN

Forensik digital merupakan salah satu teknologi yang mendukung proses pencarian bukti-bukti hukum. Forensik digital digunakan untuk membantu menyelidiki kejahatan dunia maya atau mengidentifikasi bukti langsung kejahatan yang dibantu komputer [1]. Identifikasi jenis *file* adalah tugas yang sangat kompleks bagi pemeriksa forensik digital [2]. Analisis terhadap ekstensi *file* diperlukan untuk membantu upaya mendeteksi manipulasi suatu *file* dengan menggunakan metode modifikasi yang bertujuan untuk menyembunyikan isi aslinya [3]. Jika ekstensi suatu *file* diubah dari ekstensi aslinya, aplikasi yang digunakan untuk menjalankan *file* tersebut tidak dapat mengidentifikasinya. Hal ini merupakan masalah keamanan yang harus ditingkatkan dalam proses pertumbuhan teknologi saat ini karena penjahat sering memanipulasi keaslian suatu *file* [4]. Signature dapat merujuk pada *header file* karena setiap ekstensi memiliki *signature* sendiri sebagai mekanisme autentikasi [5]. *Signature file* adalah nomor unik di *header file* yang digunakan untuk mengidentifikasi atau memverifikasi integritas konten *file*. Ekstensi *file* adalah akhiran berupa nama atau identitas *file* komputer yang menunjukkan jenis *file*. Suatu *file* diidentifikasi berdasarkan ekstensinya, Namun ada hal lain yang lebih mendalam pada proses identifikasi suatu file adalah mengetahui *header* suatu *file* [6]. Penelitian terkait *signature file* [2], [3], [7], [8], [9], [10], [11]. Penelitian [12] Terkait modifikasi *file* yang pernah dilakukan sebelumnya: mengubah ekstensi file untuk mengelabui sistem operasi. Penelitian yang dilakukan oleh Michael Yip ini mencoba mengubah ekstensi *file* gambar dari .jpg menjadi .doc yang memberikan bukti betapa mudahnya mengelabui sistem operasi agar menampilkan jenis *file* yang salah hanya dengan menggunakan metode mengubah ekstensi file dan caranya. untuk memeriksa *signature file* menggunakan alat *HexEdit* untuk mengetahui keaslian suatu ekstensi *file*. Pemeriksaan dengan metode ini memerlukan waktu yang lama.

ISSN: 2580-3336 (Print) / 2549-7952 (Online)
Url: https://proceeding.unpkediri.ac.id/index.php/inotek/

II. METODE

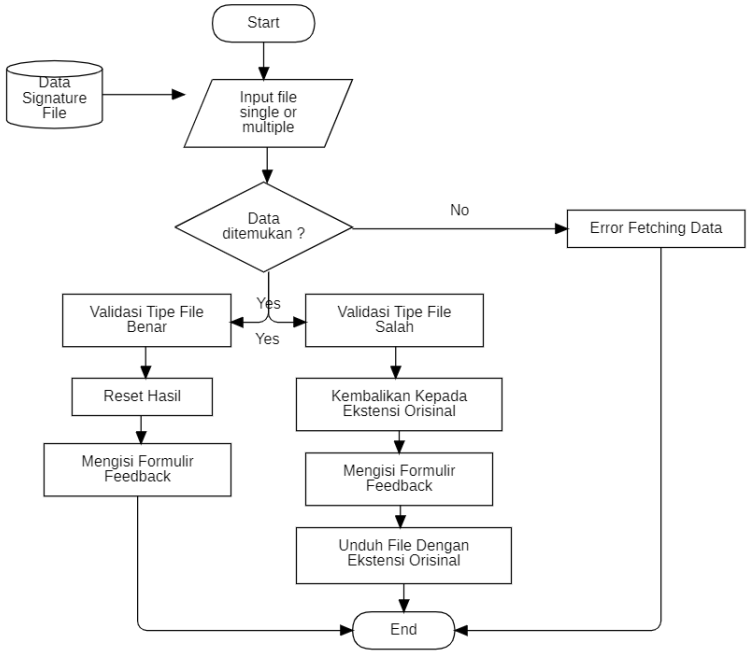
```
function mulai() {

    let daftarFile = ambilDaftarFile();
    let hasilAnalisis = [];

    for (let file of daftarFile) {
        let ekstensiFile = bacaEkstensiFile(file);
        let sigantureFile = bacaSignatureFile(file);

        if (cocokkanSignatureFile(sigantureFile)) {
            hasilAnalisis.push({ file: file, status: 'orisinal' });
        } else {
            hasilAnalisis.push({ file: file, status: 'modifikasi' });
        }
    }

    simpanHasilAnalisis(hasilAnalisis);

    tampilkanLaporan(hasilAnalisis);

    selesai();
}

function ambilDaftarFile() {
    return ['file1.txt', 'file2.pdf', 'file3.jpg'];
}

function bacaEkstensiFile(file) {
    return file.split('.').pop();
}

function bacaSignatureFile(file) {
    return 'signature_sample';
}

function cocokkanSignature(signature) {
    let basisDataSignature = ['signature_sample'];
    return basisDataSignature.includes(signature);
}

function simpanHasilAnalisis(hasilAnalisis) {
    let hasil = '';
    hasilAnalisis.forEach(hasilItem => {
        hasil += `File: ${hasilItem.file}, Status: ${hasilItem.status}\n`;
    });
    console.log(hasil);
}

function tampilkanLaporan(hasilAnalisis) {
    hasilAnalisis.forEach(hasilItem => {
        console.log(`File: ${hasilItem.file}, Status: ${hasilItem.status}`);
    });
}

function selesai() {
    console.log("Analisis selesai.");
}
```

Gambar 2.1 Metode Yang Digunakan

Merujuk gambar 2.1 ditunjukan *pseudocode* yang digunakan aplikasi. Aplikasi menggunakan metode pengulangan sebagai inti dari proses yang berlangsung ketika aplikasi melakukan *check document*. Cara kerja metode pengulangan ini sistem memproses *file input* dengan cara memeriksa keseluruhan *magic number* dan berhenti ketika mendapatkan *magic number* yang sesuai antara *file input* dengan *magic number* yang tersimpan di dalam *database* aplikasi.

III. HASIL DAN PEMBAHASAN

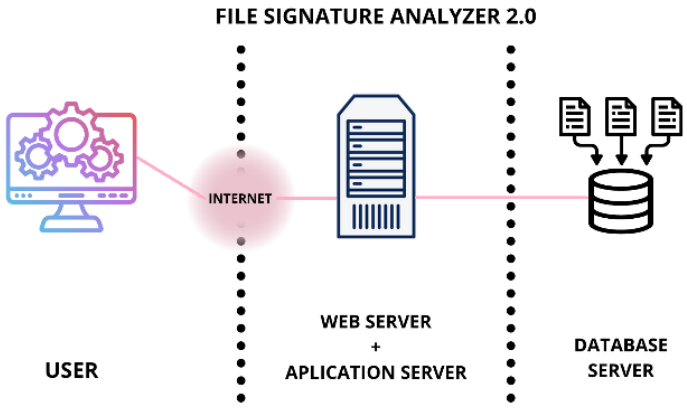
a. *Flowchart* Aplikasi



Gambar 3.1 Flowchart Aplikasi

Merujuk gambar 2.1 ditunjukan *flowchart* aplikasi yang bisa diakses oleh aktor pengguna yang sudah memiliki akun. Aplikasi *File Signature Analyzer* 2.0 hanya memiliki aktor utama yaitu user.

b. Arsitektur Aplikasi



Gambar 3.2 Arsitektur Aplikasi

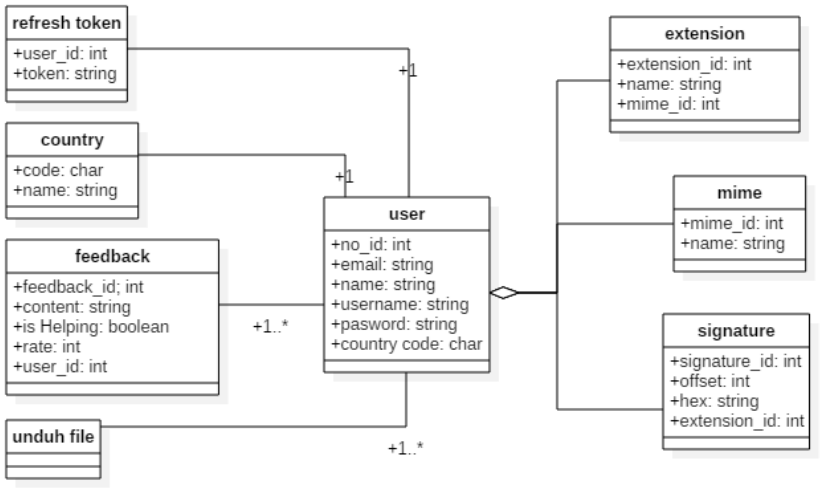
Merujuk gambar 3.2 dijelaskan sistem yang dibangun menggunakan arsitektur web tiga tingkat. Arsitektur web 3 tingkat adalah arsitektur yang terdiri dari tingkat presentasi, tingkat aplikasi, dan tingkat data. Tingkat data menyimpan informasi, tingkat aplikasi menangani logika

dan tingkat presentasi adalah antarmuka pengguna grafis yang berkomunikasi dengan dua tingkat lainnya.

c. Dukungan Aplikasi

Gambar 3.3 Dukungan Aplikasi

Merujuk gambar 3.3 ditunjukan dukungan untuk aplikasi yang meliputi data ekstensi file, mimetype, signature dan negara untuk melacak pengguna.



Gambar 3.4 Class Diagram Aplikasi

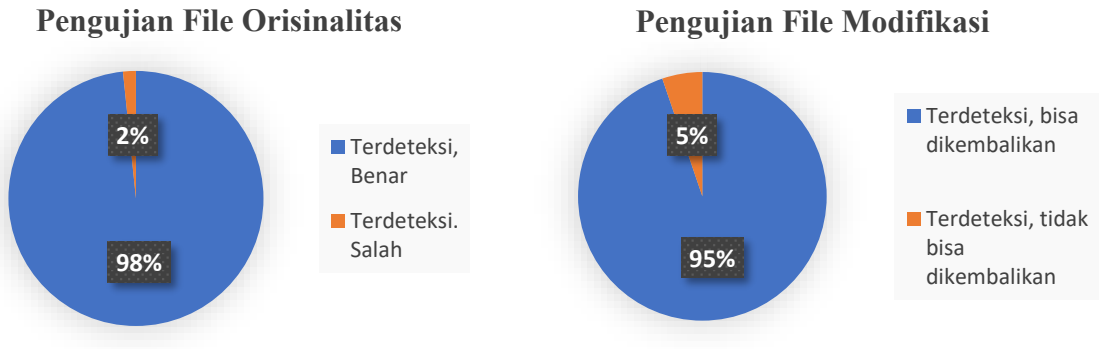
Merujuk gambar 3.4 ditunjukan Hubungan antar class – class yang dituangkan dalam class diagram yang bertujuan untuk membangun dan mendukung jalannya aplikasi.

d. Pengujian Aplikasi

Hasil pengujian merupakan bagian dari implementasi terhadap aplikasi yang dibuat. Untuk mengetahui sejauh mana aplikasi yang dibuat mencapai tujuan penelitian dilakukan pengujian

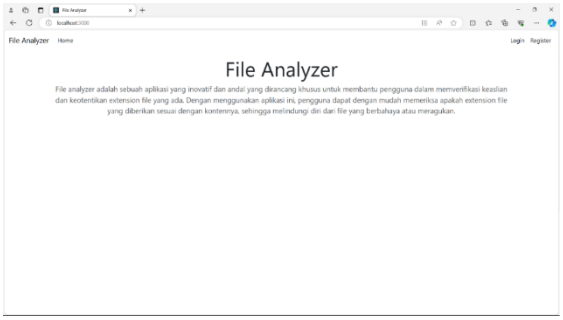
terhadap aplikasi yang telah dibuat. Banyak *file* yang akan dilakukan pengujian sebanyak 500 *file* yang bersumber dari *filesamples.com* dengan rincian:

1. 246 *file* orisinal,
2. 250 *file* modifikasi,
   - 20 ekstensi dihilangkan
   - 228 ekstensi dirubah secara acak
   - 2 ekstensi ganda
3. 1 *file* diatas 20mb.
4. 3 *file* pemrograman



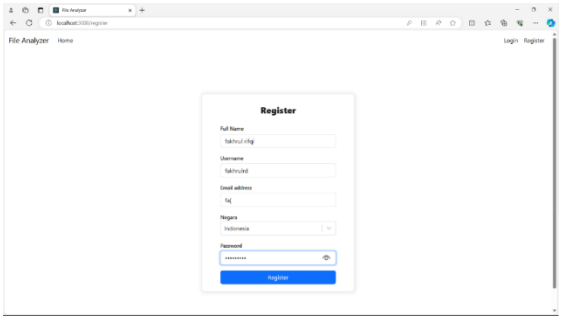
Gambar 3.5 Hasil Pengujian Aplikasi

Merujuk gambar 3.5 ditunjukan pengujian aplikasi yang memberikan hasil cukup akurat; File orisinal terdeteksi benar (98%) dan Terdeteksi salah (2%), File modifikasi terdeteksi bisa dikembalikan (95%) dan terdeteksi, tidak bisa dikembalikan (5%).



Gambar 3.6 *Homepage* Aplikasi

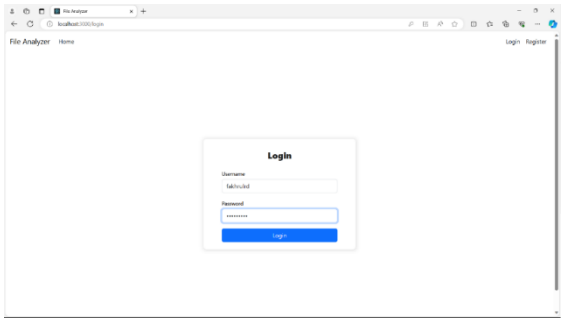
Merujuk gambar 3.6 ditunjukan halaman awal aplikasi. Terdapat informasi singkat aplikasi dan fitur register untuk membuat akun.



Gambar 3.7 Registrasi

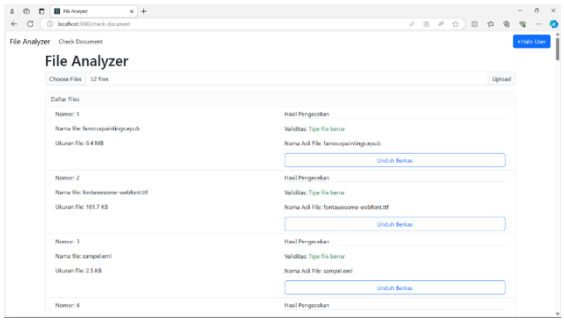
Gambar 3.7 ditunjukan halaman registrasi atau pendaftaran aplikasi. Tampil formulir untuk diisi oleh *user*.

ISSN: 2580-3336 (Print) / 2549-7952 (Online)
Url: https://proceeding.unpkediri.ac.id/index.php/inotek/



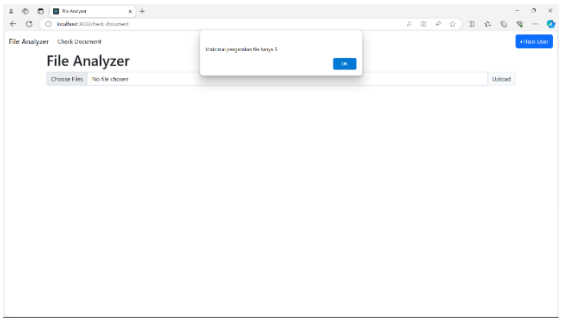
Gambar 3.8 *Login*

Gambar 3.8 ditunjukan halaman *login* untuk *user* diminta meng-*input*-kan *username* dan *password.*



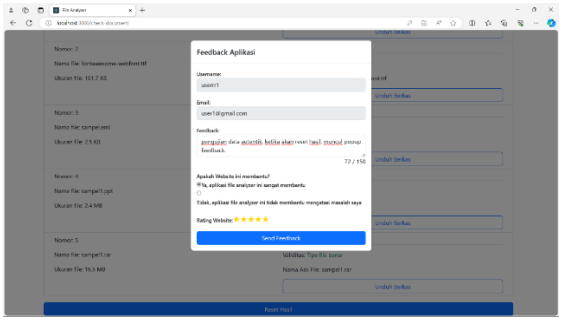
Gambar 3.9 *Check Documents*

Gambar 3.9 ditunjukan halaman *check documents*. Dapat dilihat user meng-input-kan file lebih dari satu secara bersamaan dan aplikasi dapat memproses.



Gambar 3.10 Maksimal Pengujian

Gambar 3.10 ditunjukan peringatan maksimal pengujian *file* yang dapat di unggah oleh *user* dan dapat di proses oleh aplikasi.



Gambar 3.11 *Pop-up Feedback*

Gambar 3.11 ditunjukan formulir *feedback yang* akan tampil ketika *user* melakukan riset hasil.

IV. KESIMPULAN

Penelitian ini telah mempresentasikan *File Signature Analyzer* 2.0, pendekatan untuk identifikasi jenis ekstensi *file* secara otomatis dengan waktu yang relatif cepat dibandingkan dengan cara konvensional untuk melakukan validasi terhadap ekstensi *file*. Pendekatan yang dipresentasikan dalam penelitian ini menggunakan algoritma perulangan yang berbasis kepada *file signature* atau *magic number*. *File Signature Analyzer* 2.0 berbasis web dapat digunakan oleh pengguna yang tidak ahli untuk melakukan investigasi forensik. Pengujian memberikan hasil yang akurat terdeteksi benar (98%) dan Terdeteksi salah (2%), *file* modifikasi terdeteksi bisa dikembalikan (95%) dan terdeteksi, tidak bisa dikembalikan (5%).

DAFTAR PUSTAKA

[1] M. Tembely and S. M. Musa, "Digital Forensics," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 7, no. 4, pp. 274–276, 2017, doi: 10.23956/ijarcsse/v7i4/01404.

[2] K. Karampidis and G. Papadourakis, "File Type Identification - Computational Intelligence for Digital Forensics," *J. Digit. Forensics, Secur. Law*, vol. 12, no. 2, 2017, doi: 10.15394/jdfsl.2017.1472.

[3] R. Hegarty and J. Haggerty, "SlackStick: Signature-Based File Identification for Live Digital Forensics Examinations," *Proc. - 2015 Eur. Intell. Secur. Informatics Conf. EISIC 2015*, pp. 24–29, 2016, doi: 10.1109/EISIC.2015.28.

[4] A. Ardiansyah, N. Hardi, and W. Gata, "Identifikasi dan Recovery File JPEG dengan Metode Signature-Based Carving dalam Model Automata," *Komputika J. Sist. Komput.*, vol. 9, no. 1, pp. 75–83, 2020, doi: 10.34010/komputika.v9i1.2733.

[5] B. M. B. Eloff J., "Software Failure Investigation A Near-Miss Analysis Approach."

[6] B. M. B. Eloff, *Software Failure Investigation A Near-Miss Analysis Approach*. 2017. [Online]. Available: https://books.google.co.id/books?hl=en&lr=&id=7cA0DwAAQBAJ&oi=fnd&pg=PP6&dq=Eloff,+Jan,+%26+Bella,+2017&ots=BKnnR8MAZt&sig=eSyJ--tYFs0bLPQZN-KR-NCJfOc&redir_esc=y#v=onepage&q=Eloff%2C Jan%2C %26 Bella%2C 2017&f=false

[7] M. C. Amirani, M. Toorani, and A. A. B. Shirazi, "A new approach to content-based file type detection," *Proc. - IEEE Symp. Comput. Commun.*, no. July 2008, pp. 1103–1108, 2008, doi: 10.1109/ISCC.2008.4625611.

[8] M. Subli, "Metadata Forensik Untuk Mendukung Proses Investigasi Digital," *Data Manag. dan Teknol. Inf.*, vol. 18, no. 1, pp. 44–50, 2017.

[9] B. Mainoo, "Digital Multimedia Tampering Detection for Forensics Analysis," *Adv. Multidiscip. Sci. Res. J. Publ.*, vol. 1, no. 1, pp. 81–90, 2022, doi: 10.22624/aims/crp-bk3-p14.

[10] O. A. Rosso, R. Ospina, and A. C. Frery, "Classification and verification of handwritten signatures with time causal information theory quantifiers," *PLoS One*, vol. 11, no. 12, pp. 1–20, 2016, doi: 10.1371/journal.pone.0166868.

[11] R. Rizal, R. Ruuhwan, and S. Chandra, "Signature File Analysis Using The National Institute Standard Technology Method Base on Digital Forensic Concepts," *J. Inform. Univ. Pamulang*, vol. 5, no. 3, p. 364, 2020, doi: 10.32493/informatika.v5i3.6073.

[12] M. Yip, "Signature analysis and Computer Forensics," *Sch. Comput. Sci. Univ. Birmingham*, pp. 1–11, 2008, [Online]. Available: http://www.michaelyip.me.uk/projects/SaCF.pdf