Pemanfaatan Data Aktivitas Jaringan MikroTik pada Jaringan Intra Pemerintah Kabupaten Magetan untuk Pembangkitan Kunci Kriptografi dalam Pengamanan File

Dhea Budi Bagas Ramadhan[1], I Made Suartana[2]

[1,2] (Teknik Informatika/Teknik Informatika, Universitas Negeri Surabaya)
[1]dhea.23402@mhs.unesa.ac.id
[2]imadesuartana@unesa.ac.id

***Abstrak*— Keamanan data merupakan aspek krusial dalam sistem informasi, khususnya dalam lingkungan jaringan intra pemerintah yang rentan terhadap ancaman siber. Penelitian ini bertujuan untuk mengembangkan metode pembangkitan kunci kriptografi berbasis data aktivitas jaringan mikrotik, sebagai upaya meningkatkan keamanan file dalam sistem pemerintahan Kabupaten Magetan. Metode penelitian yang digunakan adalah Research and Development (R&D). Penelitian dilaksanakan di Dinas Komunikasi dan Informatika Kabupaten Magetan dengan data aktivitas jaringan diambil dari log router mikrotik. Teknik pengumpulan data dilakukan melalui observasi dan dokumentasi. Instrumen penelitian berupa perangkat lunak pengolah log jaringan serta pengujian statistik untuk mengukur kualitas kunci yang dihasilkan. Teknik analisis data meliputi uji frekuensi, uji poker, dan approximate entropy. Hasil penelitian menunjukkan bahwa kunci kriptografi yang dibangkitkan dari data aktivitas jaringan menunjukkan karakteristik acak yang baik dan memenuhi standar kelayakan dalam uji statistik. Kesimpulan dari penilitian inin adalah metode pembangkitan kunci berbasis aktivitas jaringan dapat diimplementasikan sebagai alternatif sistem keamanan file. Saran yang diajukan yaitu pengembangan lebih lanjut dalam integrasi sistem otomatisasi dan pengujian pada skala jaringan yang lebih besar untuk penguatan keamanan data di sektor pemerintahan.**

***Kata Kunci*— Mikrotik, aktivitas jaringan, kriptografi, kunci, keamanan file, uji statistik.**

I. PENDAHULUAN

Dalam era digital, kebutuhan akan keamanan data semakin meningkat, terutama di lingkungan pemerintahan yang menangani informasi sensitif dan rahasia. Pemerintah Kabupaten Magetan, melalui Dinas Komunikasi dan Informatika memiliki jaringan intra yang menghubungkan berbagai perangkat dan sistem informasi yang ada di seluruh Organisasi Perangkat Daerah (OPD) di Kabupaten Magetan. Jaringan ini sering digunakan untuk mengirimkan file yang mengandung data penting, seperti dokumen kebijakan, data kepegawaian, dan informasi strategis lainnya.

Namun, meningkatnya ancaman keamanan siber, seperti peretasan dan pencurian data, menuntut adanya mekanisme pengamanan data yang lebih baik. Bahkan, menurut data dari Badan Siber dan Sandi Negara (BSSN) sepanjang 2023, dugaan insiden siber paling banyak terdeteksi di sektor administrasi pemerintahan dengan jumlah 186 laporan, kemudian disusul sektor keuangan dan transportasi masing-masing 38 dan 24 laporan. BSSN juga mencatat, sepanjang 2023 ada sekitar 1,67 juta data yang tesebar dan terekspos tanpa izin di darknet dengan rincian paling banyak dari sektor administrasi pemerintahan (665 ribu data), sektor keuangan (165 ribu data), dan sektor teknologi informasi dan komunikasi (161 ribu data)[1].

Kerentanan sebuah data dapat dengan mudah tersebar dan terekspos disebabkan beberapa faktor salah satunya adalah faktor pengiriman data yang tidak aman. Untuk mengamankan sebuah file atau data diperlukan sebuah proses yang enkripsi dan dekripsi. Enkripsi merupakan teknik mengubah pesan asli atau plaintext menjadi bentuk terenkripsi yang disebut ciphertext. Ciphertext ini kemudian dikirimkan oleh pengirim melalui media komunikasi yang tidak aman. Ketika pesan tersebut sampai ke penerima, proses dekripsi dilakukan untuk mengubah ciphertext kembali menjadi plaintext, sehingga isi pesan dapat dimengerti dan dibaca oleh penerima[2].

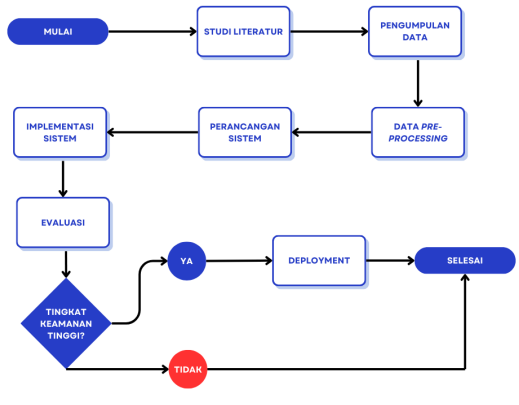
Dalam proses enkripsi dan dekripsi dibutuhkan sebuah kunci kriptografi. Untuk mendapatkan kunci kriptografi dapat dilakukan dengan beberapa cara seperti yang dilakukan pada penelitian-penelitian sebelumnya yang memiliki ruang lingkup yang relevan. Penelitian terkait pembangkitan kunci kriptografi salah satunya dilakukan oleh Nabilun Najib G.S dan Amang Sudarsono (2019) dengan memanfaatkan data Channel State Information (CSI) sebagai dasar pembagkitan kunci kriptografi untuk pengiriman gambar secara aman. Data CSI yang digunakan pada penelitian ini memuat keadaan kanal komunikasi wireless seperti bandwidth, channel, rate, RSS, dan sebagainya. Untuk mendapatkan informasi tersebut digunakan Network Interface Card (NIC) Atheros AR9485. Hasil penelitian ini menunjukkan kombinasi parameter-parameter CSI memiliki tingkat keacakan yang cukup tinggi sehingga menyulitkan pihak ketiga untuk mengetahui kunci kriptografi yang dibangkitkan [3].

Selain itu penelitian yang melakukan pengaman file melalui enkripsi dan dekripsi adalah penelitian yang dilakukan oleh Arief Dharmawan dan Haris Munandar (2023). Objek yang digunakan pada penelitian ini adalah data atau file dari

PT Pelangi Sentral Kreasi. Penelitian ini menunjukkan bahwa sistem dapat melakukan proses enkripsi dan dekripsi file dengan efisiensi tinggi serta kemudahan bagi pengguna. Penggunaan algoritma SHA-256 dan AES-256, disertai dengan sedikit penyesuaian pada hasil enkripsi, memungkinkan file berukuran 266 kilobyte dienkripsi hanya dalam waktu 29,7 milidetik dan didekripsi dalam waktu 31,4 milidetik [4].

Berdasarkan latar belakang dan penelitian sebelumnya, pada penelitian ini akan dilakukan proses pembangkitan kunci kriptografi dengan memanfaatkan data aktivitas jaringan mikrotik pada jaringan intra pemerintah Kabupaten Magetan dengan menerapkan algoritma SHA-256 dan AES-256 sebagai proses enkripsi dan dekripsi untuk pengamanan sebuah file. Peneltian ini akan berfokus pada proses pembangkitan kunci kriptografi menggunakan data aktivitas jaringan mikrotik. Untuk proses enkripsi dan dekripsi, AES-256 dipilih karena lebih aman dari algoritma lain seperti DES dan 3DES karena memiliki panjang kunci 256 bit yang tahan terhadap serangan brute force. SHA-256 dipilih karena menghasilkan nilai hash tetap yang tidak dapat dibalik, berbeda dengan MD5 atau SHA-1 yang lebih rentan terhadap *collison attack*. Sehingga kombinasi SHA-256 dan AES-256 memastikan integritas serta kerahasiaan data dengan SHA-256 untuk pembuatan kunci acak dari data, sedangkan AES untuk mengamankan file secara langsung. Hasil penelitian ini diharapkan mampu meningkatkan keamanan file yang dikirim serta menjadi model awal untuk implementasi sistem keamanan data yang lebih luas di lingkungan pemerintahan.

II. METODE PENELITIAN



Gbr. 1 Alur Penelitian

Penelitian ini menggunakan metode *Research and Development* (R&D). Pada Gbr. 1 Alur Penelitian yang dimulai dari studi literatur, pengumpulan data, data pre-processing, perancangan sistem, implementasi sistem, evaluasi, dan deployment.

A. Studi Literatur

Tahap studi literatur merupakan tahapan awal dimana peneliti melakukan pencarian dan pembelajaran terhadap sumber literatur yang berhubungan dengan penelitian yang dilakukan sebagai pendukung dan acuan penelitian. Sumber literatur yang dicari dan digunakan dalam penelitian ini berhubungan dengan proses pembangkitan kunci, penggunaan algoritma SHA-256 dan proses enkripsi dan dekripsi menggunakan AES-256. Literatur yang digunakan bersumber dari karya tulis ilmiah dan jurnal nasional maupun internasional.

B. Pengumpulan Data

Pada penelitian ini mengembangkan teknik pembangkitan kunci kriptografi menggunakan data aktivitas jaringan pada *router* mikrotik. Pada sebuah *router* mikrotik dalam sebuah jaringan terdapat beberapa data yang bersifat unik sehingga dapat digunakan sebagai dasar dalam pembangkitan kunci kriptografi. Contoh data unik tersebut berupa data *MAC-Address* dari perangkat yang terhubung dalam jaringan *router* tersebut, data *signal strength* dari perangkat yang terhubung melalui *Wi-Fi router* mikrotik tersebut, dan data lalu lintas jaringan. Pada tahap ini akan dilakukan ekstraksi data aktivitas jaringan dari mikrotik seperti data *MAC-Address* pada tabel ARP, data tabel *Tx/Rx Signal*, *Tx Rate*, dan *Rx Rate* pada tabel registration, serta data *Destination-Address* pada tabel *Firewall-connection*.

TABEL I
DATA DAN WAKTU YANG DIAMBIL

Router *MikroTik*	**Data yang diambil**	**Letak Data**	**Waktu Pengambilan (28 April 2025)**
Kec. Karangrejo	Dst-Address	Firewall-Connections	08.30, 10.30, 13.30
	Mac-Address	IP-ARP	
	Signal, Tx Rate, Rx Rate	Wireless-Registration	
Kec. Barat	Dst-Address	Firewall-Connections	08.30, 11.00, 14.00
	Mac-Address	IP-ARP	
	Signal, Tx Rate, Rx Rate	Wireless-Registration	

Pada Tabel 1 adalah data dan waktu tertentu yang ditentukan pada penelitian ini. Penelitian ini mengumpulkan data aktivitas jaringan dari dua *router MikroTik* yang berada di jaringan intra Pemerintah Kabupaten Magetan, yaitu *router* di Kecamatan Barat dan Kecamatan Karangrejo. Pengambilan data dilakukan secara otomatis dengan menanamkan script pada menu scheduler *MikroTik,* menggunakan informasi IP address, username, dan password masing-masing *router*.

C. Data Pre-Processing

Tahap *data pre-processing* adalah tahapan yang memiliki *output* untuk membuat dataset final dari data mentah, yang kemudian akan dilakukan pemodelan dengan algoritma.

- ***Cleaning data*** : Membersihkan data dari informasi yang tidak relevan agar siap diproses.

```python
def extract_and_convert_mac(input_file):
    mac_addresses = []
    try:
        with open(input_file, "r") as file:
            for line in file:
                match = re.search(r"([0-9A-Fa-f]{2}[:-]){5}[0-9A-Fa-f]{2}", line)
                if match:
                    mac_addresses.append(match.group().replace(":", "").replace("-", ""))
    except FileNotFoundError:
        print(f"File '{input_file}' tidak ditemukan.")

    mac_decimal = []
    for mac in mac_addresses:
        decimal_values = [int(mac[i:i+2], 16) for i in range(0, len(mac), 2)]
        mac_decimal.extend(decimal_values)

    return mac_decimal
```

Gbr. 2 Source Code Pengolahan Data MAC-Address

Pada Gbr. 2 adalah *source code* untuk proses pengolahan data *Mac-Address*. *Mac-Address* adalah identitas unik yang ada pada perangkat keras jaringan yang terdiri dari beberapa bilangan heksadesimal.

```python
def clean_data(input_file):
    signal_pattern = r"Signal: (-?\d+)dBm"
    tx_rate_pattern = r"Tx-rate: ([\d.]+)Mbps"
    rx_rate_pattern = r"Rx-rate: ([\d.]+)Mbps"
    output_values = []

    try:
        with open(input_file, "r") as file:
            for line in file:
                if line.strip():
                    signal = re.search(signal_pattern, line)
                    tx_rate = re.search(tx_rate_pattern, line)
                    rx_rate = re.search(rx_rate_pattern, line)

                    if signal:
                        output_values.append(int(signal.group(1).replace("-", "")))
                    if tx_rate:
                        output_values.append(float(tx_rate.group(1)))
                    if rx_rate:
                        output_values.append(float(rx_rate.group(1)))
    except FileNotFoundError:
        print(f"File '{input_file}' tidak ditemukan.")
    return output_values
```

Gbr. 3 Source Code Pengolahan Signal, Tx/Rx Rate

Pada Gbr. 3 adalah *source code* untuk proses pengolahan data signal, tx/rx rate. Data-data tersebut diolah dengan menghilangkan data selain data angka signal, tx/rx rate.

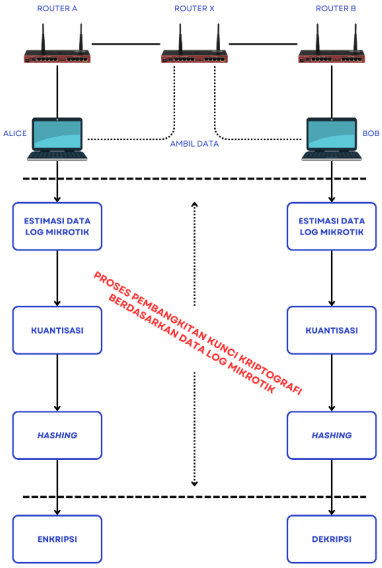
```python
def process_file(input_file):
    processed_lines = []

    try:
        with open(input_file, 'r') as file:
            unique_lines = list(file.readlines())

        for line in unique_lines:
            line = line.strip()
            if line:
                if ':' in line:
                    line = line.split(':')[0]
                segments = line.replace(':', '.').split('.')
                processed_lines.extend(map(lambda x: int(float(x)), segments))
    except FileNotFoundError:
        print(f"File '{input_file}' tidak ditemukan.")
    return processed_lines
```

Gbr. 4 Source Code Pengolahan Data Destination-Addess

Pada Gbr. 4 adalah *source code* untuk proses mengolah data Destination-address. Data tersebut diolah dengan menghapus data yang tidak diperlukan yaitu data port dan hanya mengambil IP Address saja.

- **Penyesuaian data** : Mengubah data bersih ke dalam format template input yang sesuai dengan metode yang digunakan.

D. Perancangan Sistem

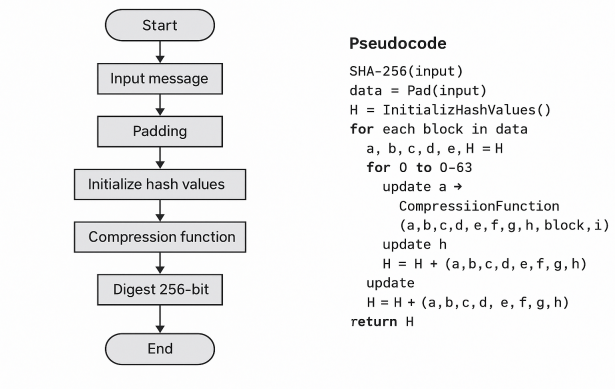


Gbr. 5 Rancangan Sistem

Pada Gbr. 5 Rancangan Sistem yaitu proses pembangkitan kunci kriptografi berdasarkan data log mikrotik. Dalam proses pembangkitan kunci ini terdapat 3 subproses yaitu estimasi data log mikrotik, kuantisasi data, dan terakhir adalah proses *hashing*. Sehingga output dari proses ini adalah sebuah kunci kriptografi yang dapat digunakan untuk proses enkripsi dan dekripsi file.

- **Estimasi Data Aktivitas**: Menggabungkan lima jenis data dari MikroTik (MAC address, Tx/Rx signal, Tx rate, Rx rate, dan destination address) untuk membentuk dasar pembangkitan kunci yang acak dan unik.
- **Kuantisasi**: Mengubah hasil kombinasi data menjadi bit (0 dan 1) menggunakan skema Aono, berdasarkan nilai median sebagai threshold. Nilai di atas threshold menjadi 1, di bawah menjadi 0, dan yang sama dengan threshold dibuang. Kuantisasi Aono menggunakan nilai median sebagai threshold karena median tahan terhadap nilai ekstrem dan *noise* serta meningkatkan entropy *bitstream.* Median dipilih untuk menentukan batas ambang pada proses kuantisasi bit dari nilai RSSI. [5]
- **Hashing**: Hasil kuantisasi diacak, kemudian dibagi menjadi blok 256 bit dan diubah menjadi bentuk hash menggunakan algoritma SHA-256, menghasilkan sejumlah kandidat kunci untuk proses enkripsi dan dekripsi. SHA-256 merupakan salah satu algoritma hash

yang dirancang oleh *National Institute of Standards and Technology* (NIST) pada tahun 2002.

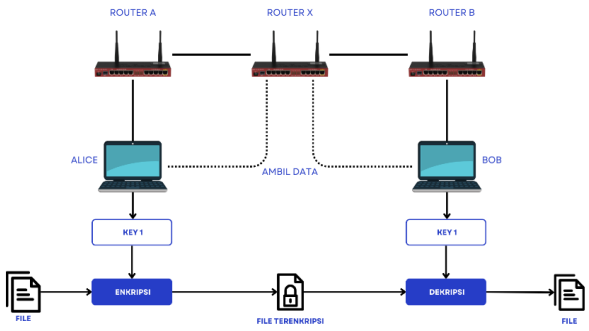


Gbr. 6 Flowchart dan Pseudocode SHA-256

Pada Gbr. 6 Flowchart dan Pseudocode SHA-256, algoritma ini menghasilkan *message digest* sepanjang 256 bit. SHA-256 dianggap aman karena dirancang sedemikian rupa sehingga sangat kecil kemungkinannya untuk menemukan dua pesan berbeda yang menghasilkan *message digest* yang identik. [6].

E. Implementasi Sistem

Sistem ini memanfaatkan algoritma kriptografi AES-256 untuk melindungi kerahasiaan dan integritas file melalui proses enkripsi. Enkripsi tersebut mengubah data asli (*plaintext*) menjadi bentuk terenkripsi (*ciphertext*) sehingga tidak dapat diakses atau diubah oleh pihak yang tidak memiliki otorisasi.



Gbr. 8 Implementasi Sistem

Pada Gbr. 7 Implementasi Sistem, dilakukan satu kali proses enkripsi dan dekripsi. Enkripsi dilakukan oleh User 1 (Alice) dan dekripsi oleh User 2, keduanya menggunakan kunci yang dihasilkan dari data *router MikroTik* yang sama.

```
Cipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
   byte  state[4,Nb]

   state = in

   AddRoundKey(state, w[0, Nb-1])                  // See Sec. 5.1.4

   for round = 1 step 1 to Nr-1
      SubBytes(state)                              // See Sec. 5.1.1
      ShiftRows(state)                             // See Sec. 5.1.2
      MixColumns(state)                            // See Sec. 5.1.3
      AddRoundKey(state, w[round*Nb, (round+1)*Nb-1])
   end for

   SubBytes(state)
   ShiftRows(state)
   AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1])

   out = state
end
```

Gbr. 7 Pseudocode AES-256[7]

Pada Gbr. 8 Pseudocode AES-256, Algoritma AES (Advanced Encryption Standard) dapat melakukan enkripsi dan dekripsi dengan panjang kunci 128, 192, atau 256 bit. Proses enkripsi dilakukan dalam beberapa ronde transformasi, dimulai dengan AddRoundKey, lalu diikuti oleh ronde utama yang terdiri dari empat transformasi:

1. SubBytes (substitusi)
2. ShiftRows (permutasi)
3. MixColumns (pengacakan)
4. AddRoundKey (penambahan kunci)

Pada ronde terakhir, MixColumns tidak digunakan. Dekripsi AES dilakukan dengan transformasi invers dari tiap langkah enkripsi, yaitu InvSubBytes, InvShiftRows, InvMixColumns, sedangkan AddRoundKey bersifat self-invers asalkan menggunakan kunci yang sama.

F. Evaluasi

Tahap evaluasi dalam penelitian ini bertujuan untuk menilai dua aspek utama: tingkat keacakan kunci kriptografi yang dihasilkan serta efisiensi proses enkripsi dan dekripsi file. Penilaian terhadap keacakan kunci dilakukan melalui tiga jenis uji statistik berdasarkan standar dari National Institute of Standards and Technology (NIST)[8], yakni:

- Uji Frekuensi Bit (Monobit Test) – Mengukur keseimbangan jumlah bit ‘0’ dan ‘1’ dalam kunci.
- Uji Poker – Mengamati pola 4-bit untuk menilai distribusi blok bit dalam kunci.
- Uji Approximate Entropy – Menilai tingkat ketidakpastian atau kompleksitas pola dalam bitstream kunci.

Setelah ketiga uji tersebut dilakukan, kunci terbaik dipilih berdasarkan hasil yang paling memenuhi kriteria keacakan, dengan prioritas utama pada uji poker, disusul oleh approximate entropy, dan frekuensi bit.

Selain mengevaluasi keacakan kunci, penelitian ini juga menguji kinerja proses enkripsi dan dekripsi. Evaluasi dilakukan dengan mengukur waktu pemrosesan enkripsi dan dekripsi pada berbagai ukuran file. Hal ini bertujuan untuk memastikan bahwa sistem tidak hanya menghasilkan kunci yang aman secara kriptografi, tetapi juga dapat bekerja secara efisien dalam konteks penggunaan nyata.

Secara keseluruhan, tahap evaluasi ini memberikan dasar yang kuat untuk menilai kualitas kriptografi dan performa praktis dari sistem yang dikembangkan.

Gbr. 9 Skema Deployment Sistem

Pada Gbr. 9 Skema Deployment Sistem adalah skema dari rancangan sistem yang akan dikembangkan dengan *Mikrotik* sumber data yang mencatat log aktivitas jaringan. Python server memiliki fungsi menjalankan *script* pengambilan log dari *Mikrotik,* pembangkitan kunci, dan proses enkripsi dan dekripsi. *Frontend Application* dari sistem ini menggunakan *django* yang akan berinteraksi langsung dengan python server dan menyimpan file kunci, enkripsi dan dekripsi ke *storage*.

III. HASIL DAN PEMBAHASAN

A. *Pemanfaatan Data Aktivitas Mikrotik*

TABEL III
HASIL HASHING DARI DATA ROUTER KECAMATAN BARAT

Waktu	SHA-256	Label
08.30	fb91ea2eaa760dafa5ac6fbcd4fbf8f1277d5d9c727c7de8b636ee845082cdae	Kunci 1
	32685652c52d612fa21370b09c62b9dcaf86533806e617a25177acacfd5b217d	Kunci 2
	5f5fe0fc6c6d711045165d966e530717ddad4c6fcbb355d31352926138eb89a1	Kunci 3
	801635f52871dccb68764c3c1e25265168cbaa999d99067e2791f1066a6b9813	Kunci 4
11.00	e7dd30d5267f51d37f694ec242f27ea35b0788c8504c0984264e77360965bc52	Kunci 1
	534c7329914bfe2a1845b3b9f37bdb85c847915e5fe01dbead4ed2e978ef5918	Kunci 2
14.00	efff1480120f7230facee0604caea5a494c22da680801caf6935e8bce0f370b5	Kunci 1
	0a18850448e61b8af174dad0b272c10cd7f8fde888f819c7f308873b7146dad4	Kunci 2
	0e2a92d128d6c520d256cddff784a9840dc63b93d4c11d87ab61fffb0b36dc8c	Kunci 3
	91c31a430cc2850be8dcdfae104e920b4ad3b860eed66049448ee6714b50810d	Kunci 4

Pada Tabel 2 adalah hasil hashing dari data kuantisasi router kecamatan Barat menggunakan algoritma SHA-256.. Hasil tersebut adalah kunci yang dapat digunakan untuk proses enkripsi dan dekripsi.

TABEL IIIII
HASIL HASHING DARI DATA ROUTER KECAMATAN KARANGREJO

Waktu	SHA-256	Label
08.30	44a33dfb9393c321e3c833e7841128e64085aed5b238b4b42729ec8f87cbd138	Kunci 1
	72f0ae9fb6e07989ae8e54e40e1f96763d7b026b0d4b4b3296f481193ffdfc02	Kunci 2
10.30	43632448f39a0527c434407674e167ab6b054d61c5120d51949406ba4204674a	Kunci 1
	e7ea8996e5b3b5640b5a2c4dd7de92da839438a4a0df5cff1d7f3dcba8e69a16	Kunci 2
13.30	6f8cc46b6e0c7e57cd76a143e6ad3a23ac358a3c1babadeef4be63c6a5f6cd4f	Kunci 1
	7ea95335710b48bcb17b08fc0097125bf654a976344eeebdac07f73eb5410d0f	Kunci 2
	343e1db949708b243dc25ac898c00e8cfa5bc88fb39152e4545d27c8e70b8f4f	Kunci 3

Pada Tabel 3 adalah hasil hashing dari data kuantisasi router kecamatan Karangrejo menggunakan algoritma SHA-256. Hasil tersebut adalah kunci yang dapat digunakan untuk proses enkripsi dan dekripsi.

B. *Keacakan Kunci*

Untuk mengetahui kualitas kunci kriptografi yang telah dihasilkan dari data aktivitas jaringan *router MikroTik* Kecamatan Barat dan Kecamatan Karangrejo dilakukan tahap

```python
def frequency_test(bit_sequence):
    n = len(bit_sequence)
    count = bit_sequence.count('1') - bit_sequence.count('0')
    s_obs = abs(count) / math.sqrt(n)
    return math.erfc(s_obs / math.sqrt(2))
```

evaluasi. Kunci kriptografi yang dihasilkan akan diuji menggunakan uji NIST.

Gbr. 10 Source Code Uji Frekuensi Bit (Monobit)

Pada Gbr. 10 adalah *source code* uji frekuensi bit (monobit). Uji ini dilakukan untuk mengukur apakah barisan bit memiliki proporsi bit 0 dan 1 yang seimbang. Apabila nilai

```python
def poker_test(bit_sequence, m=2):
    if len(bit_sequence) % m != 0:
        bit_sequence = bit_sequence[:-(len(bit_sequence) % m)]
    n = len(bit_sequence)
    k = n // m
    blocks = [bit_sequence[i * m:(i + 1) * m] for i in range(k)]
    counts = Counter(blocks)
    x3 = (2 ** m / k) * sum(freq ** 2 for freq in counts.values()) - k
    p_value = math.exp(-x3 / 2)
    return p_value
```

dari p-value uji statistik frekuensi bit lebih dari sama dengan 0,01 menandakan keseragaman antara bit 0 dan 1.

Gbr. 11 Source Code Uji Poker

Pada Gbr. 11 adalah *source code* dari uji poker. Uji ini adalah uji frekuensi dengan blok tertentu. Sebuah kunci dikatakan acak jika mendapat nilai lebih dari sama dengan 0,01.

```python
def approximate_entropy_test(bit_sequence, m=2):
    def phi(m):
        padded = bit_sequence + bit_sequence[:m]
        counts = Counter(padded[i:i + m] for i in range(len(bit_sequence)))
        total = sum(counts.values())
        result = 0
        for count in counts.values():
            prob = count / total
            if prob > 0:
                result += prob * math.log(prob)
        return result

    phi_m = phi(m)
    phi_m1 = phi(m + 1)
    ap_en = phi_m - phi_m1
    chi_squared = 2 * len(bit_sequence) * (math.log(2) - ap_en)
    return math.erfc(math.sqrt(chi_squared / 2))
```

Gbr. 12 Source Code Uji Approximate Entropy

Pada Gbr. 12 adalah *source code* dari uji approximate entropy. Uji ini dilakukan untuk mengukur ketidakpastian atau keacakan pola berulang dalam string bit dengan menghitung entropi empiris. Apabila nilai p-value approximate entropy lebih dari sama dengan 0,01 maka menunjukkan hasil yang acak.

TABEL IVV
HASIL UJI NIST PADA KUNCI YANG DIHASILKAN (KECAMATAN BARAT)

Waktu	Kandidat Kunci	Uji Frekuensi	Uji Poker	Uji Approximate entropy
08.30	Kunci 1	0,0244	0,0342	0,0151
	Kunci 2	0,7077	0,6258	0,1624
	Kunci 3	0,6171	0,0302	0,2014
	Kunci 4	0,1232	0,3173	0,0360
11.00	Kunci 1	0,7077	0,3566	0,1229
	Kunci 2	0,2606	0,3796	0,1935
14.00	Kunci 1	0,3173	0,0413	0,0739
	Kunci 2	0,3816	0,3147	0,1788
	Kunci 3	0,9005	0,6661	0,6926
	Kunci 4	0,0455	0,1022	0,0213

Pada Tabel 4 Berdasarkan uji frekuensi terhadap kandidat kunci dari router Mikrotik Kecamatan Barat, seluruh kunci berhasil lolos uji keacakan berdasarkan standar NIST. Namun, satu kandidat kunci (Kunci 1) menunjukkan nilai p-value yang paling rendah, yaitu 0,0244 untuk uji frekuensi monobit, 0,0342 untuk uji poker, dan 0,0151 untuk uji approximate entropy, meskipun masih berada di atas batas minimum kelolosan (≥ 0,01).

TABEL V
HASIL UJI NIST PADA KUNCI YANG DIHASILKAN (KECAMATAN KARANGREJO)

Waktu	Kandidat Kunci	Uji Frekuensi	Uji Poker	Uji Approximate entropy
08.30	Kunci 1	0,5320	0,1582	0,0049
	Kunci 2	0,5320	0,5879	0,2266
10.30	Kunci 1	0,0087	0,0037	0,0026
	Kunci 2	0,1691	0,1486	0,0490
13.30	Kunci 1	0,0801	0,0990	0,0109
	Kunci 2	0,4301	0,7077	0,0321
	Kunci 3	0,3796	0,5320	0,0256

Pada Tabel 5 hasil uji frekuensi terhadap kandidat kunci dari router Mikrotik Kecamatan Karangrejo, terdapat satu kunci (Kunci 1 dari jam 10.30) yang dikategorikan tidak acak karena seluruh nilai uji NIST berada di bawah batas minimum (p-value < 0,01). Selain itu, satu kunci lain (Kunci 1 dari jam 08.30) juga gagal pada uji approximate entropy dengan nilai 0,0049, meskipun lolos pada dua uji lainnya.

TABEL VI
TINGKAT KELOLOSAN UJI NIST

	Uji Frekuensi	Uji Poker	Uji Approximate entropy
Tingkat Kelolosan	94.11 %	94.11 %	88.23 %

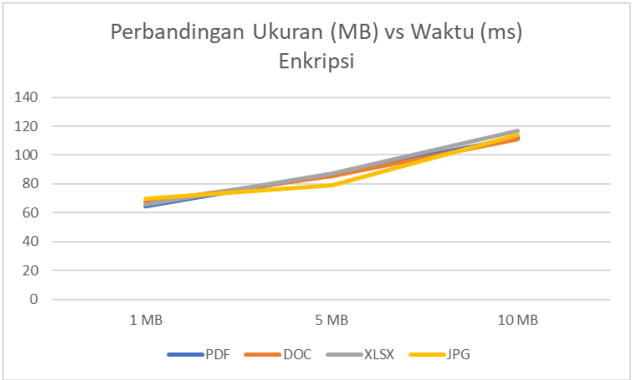
Dari Tabel 6 uji NIST yang dilakukan terhadap kandidat kunci yang telah dihasilkan memiliki tingkat kelolosan yang cukup tinggi. Sehingga hal ini dapat disimpulkan bahwa variasi data yang dimiliki oleh data aktivitas jaringan mikrotik cukup baik untuk digunakan sebagai dasar dari pembangkitan sebuah kunci kriptografi.

C. *Efektifitas Proses Enkripsi dan Dekripsi*

TABEL VII
HASIL RATA-RATA UJI ENKRIPSI FILE

Format File	1 MB	5 MB	10 MB
PDF	64.73 ms	86.73 ms	112.26 ms
DOC	67.4 ms	85.2 ms	111 ms
XLSX	65.63 ms	86.9 ms	116.66 ms
JPG	69.46 ms	78.81 ms	113.98 ms

Pada Tabel 7 adalah tabel hasil rata-rata uji enkripsi, dimana uji enkripsi pada penelitian ini berhasil dilakukan dengan rata-rata waktu yang digunakan adalah kurang dari 1 detik bahkan kurang dari 200 milidetik. Hal tersebut membuat enkripsi pada penelitian ini dilakukan dengan sangat efektif tanpa ada kendala dan proses yang lama.



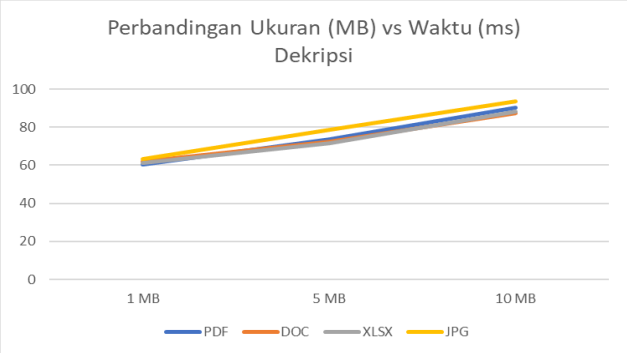
Gbr. 13 Grafik Perbandingan Ukuran File dengan Waktu Enkripsi

Pada Gbr. 13 adalah grafik perbandingan ukuran file dengan waktu enkripsi yang dibutuhkan. Dari grafik tersebut dapat disimpulkan bahwa semakin besar file target untuk enkripsi maka akan semakin lama proses yang dibutuhkan untuk enkripsi. Jadi ukuran file berbanding lurus dengan waktu yang dibutuhkan untuk proses enkripsi file tersebut.

TABEL VIII
HASIL RATA-RATA UJI DEKRIPSI FILE

Format File	1 MB	5 MB	10 MB
PDF	60.4 ms	73.53 ms	90.41 ms
DOC	62.11 ms	72.35 ms	87.15 ms
XLSX	61.11 ms	71.78 ms	88.15 ms
JPG	63.2 ms	78.8 ms	93.65 ms

Pada Tabel 8 proses dekripsi pada penelitian ini berhasil dilakukan dengan rata-rata waktu yang digunakan adalah kurang dari 1 detik bahkan rata-rata dapat dilakukan kurang dari 100 milidetik.

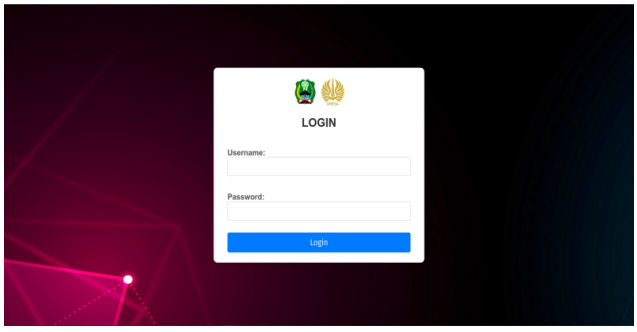


Gbr. 14 Grafik Perbandingan Ukuran File dengan Waktu Dekripsi

Pada Gbr. 14 adalah grafik perbandingan ukuran file dengan waktu dekripsi yang dibutuhkan. Dari grafik tersebut dapat disimpulkan bahwa semakin besar file target untuk dekripsi maka akan semakin lama proses yang dibutuhkan untuk dekripsi. Jadi ukuran file berbanding lurus dengan waktu yang dibutuhkan untuk proses dekripsi file tersebut.

D. *Deployment Sistem*

- Halaman Login



Gbr. 15 Halaman Login Sistem

Pada Gbr. 15 menampilkan halaman login untuk pengguna. Pemberian halaman login pada sistem ini bertujuan untuk menjaga dan meningkatkan keamanan dalam penggunaan sistem ini. Pada halaman pengguna akan memasukkan *username* dan *password* yang telah diberikan oleh administrator.

- Halaman Utama Pembangkitan Kunci

Gbr. 16 Halaman Utama Pembangkitan Kunci

Pada Gbr 16 menunjukkan halaman utama pembangkitan kunci yang menampilkan header sistem, ucapan selamat datang kepada pengguna, dan tombol logout. Halaman ini memiliki tiga tab utama: “Key”, “Enkripsi”, dan “Dekripsi”. Pada tab “Key”, pengguna dapat memilih router target sebagai dasar pembangkitan kunci dan menekan tombol “Generate” untuk memulai proses tersebut.

- Halaman Utama Enkripsi

Gbr. 17 Halaman Utama Enkripsi

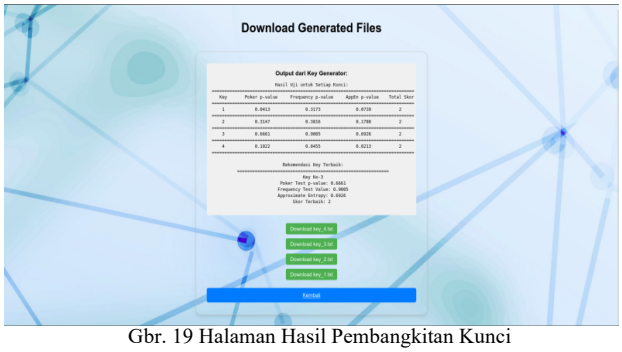
Pada Gbr. 17 menampilkan halaman utama dengan tab yang dipilih adalah “Enkripsi”. Pada halaman ini terdapat 2 field sebagai tempat untuk mengunggah file kunci dan file target enkripsi. Terdapat pula tombol “Encrypt” untuk menjalankan proses enkripsi.

- Halaman Utama Dekripsi

Gbr. 18 Halaman Utama Dekripsi

Pada Gbr. 18 menampilkan halaman utama dengan tab yang dipilih adalah "Dekripsi". Pada halaman ini terdapat juga 2 field sebagai tempat untuk mengunggah file kunci dan file target dekripsi. Terdapat pula tombol "Decrypt" untuk menjalankan proses dekripsi

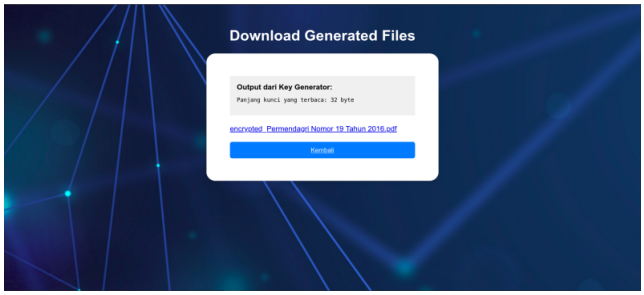
- Halaman Hasil Pembangkitan Kunci



Gbr. 19 Halaman Hasil Pembangkitan Kunci

Pada Gbr. 19 menampilkan hasil pembangkitan kunci, termasuk informasi hasil uji NIST (uji poker, frekuensi, dan approximate entropy), total skor uji, serta rekomendasi kunci terbaik. Halaman ini juga menyediakan file kunci yang dapat diunduh secara otomatis saat diklik, serta tombol "Kembali" untuk menuju halaman utama.

- Halaman Hasil Enkripsi



Gbr. 20 Halaman Hasil Proses Enkripsi

Pada Gbr 20 menampilkan halaman hasil dari proses enkripsi suatu file. Terdapat informasi tentang status proses enkripsi dan informasi mengenai panjang kunci yang terbaca. Dimana pada penelitian ini menggunakan algoritma AES-256 sehingga panjang kunci yang bisa digunakan adalah 32 byte. Pada halaman ini juga tersedia file hasil enkripsi, dimana ketika file hasil enkripsi tersebut diklik maka pengguna akan mendownload file tersebut secara otomatis. Tersedia juga tombol "Kembali" untuk kembali ke halaman utama.

- Halaman Hasil Dekripsi

Gbr. 21 Halaman Hasil Proses Dekripsi

Pada Gbr. 21 menampilkan halaman hasil dari proses dekripsi suatu file. Terdapat informasi tentang status proses enkripsi apakah berhasil atau tidak dan terdapat juga informasi mengenai panjang kunci yang terbaca. Pada penelitian ini menggunakan algoritma AES-256 untuk proses dekripsi sehingga panjang kunci harus 32 byte. Pada halaman ini juga tersedia file hasil dekripsi dengan didahului dengan kata "clear_" dan tersedia juga file hasil enkripsi yang sudah diunggah sebelumnya, dimana ketika file hasil dekripsi tersebut diklik maka pengguna akan mendownload file tersebut secara otomatis. Tersedia juga tombol "Kembali" untuk kembali ke halaman utama.

Gbr. 22 Halaman Hasil Proses Dekripsi Jika Salah Dalam Penggunaan Kunci

Pada Gbr. 22 adalah tampilan apabila proses dekripsi gagal karena kunci yang digunakan dalam proses dekripsi berbeda dengan kunci yang digunakan dalam proses enkripsi.

IV. KESIMPULAN

Penelitian membuktikan bahwa data aktivitas jaringan Mikrotik seperti MAC-Address, IP Destination, Signal Strength, dan Tx/Rx Rate dapat dikonversi menjadi bitstream acak yang cocok sebagai dasar pembangkitan kunci kriptografi. Dari 17 bitstream yang dihasilkan, 15 di antaranya lolos uji statistik NIST (frekuensi, poker, dan approximate entropy), memenuhi standar keacakan (p-value ≥ 0,01). Selain itu, proses enkripsi dan dekripsi menggunakan algoritma AES-256 dengan kunci tersebut terbukti efektif, dengan waktu proses di bawah 1 detik dan hasil dekripsi yang sesuai dengan file asli, sehingga mendukung keamanan file secara praktis dan andal.

REFERENSI

[1] Direktorat Operasi Keamanan Siber, Badan Siber dan Sandi Negara, *Lanskap Keamanan Siber Indonesia*, pp. 31–32, 2024.
[2] Warkim and I. Lewelusa, "Implementasi Kriptografi Menggunakan Algoritma Advanced Encryption Standard (AES) Dengan Metode CBC (Chiper Block Chaining) Dan Pengecekan Error Detection Cyclic Redundancy Check," *Jurnal Ilmu Komputer*, vol. 11, no. 2, 2015.
[3] N. Najib and A. Sudarsono, "Pembangkitan Key Menggunakan Channel State Information untuk Pengiriman Gambar Secara Aman," 2019.
[4] A. Dharmawan and H. Munandar, "Penerapan Algoritme Kriptografi SHA-256 dan AES-256 untuk Pengamanan File Pada PT Pelangi Sentral Kreasi," *3rd Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, vol. 2, no. 2, 2023.
[5] T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka, "Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels," *IEEE Trans. Antennas Propag.*, vol. 53, no. 11, pp. 3776–3784, 2005.
[6] F. Febriyadi, F. Kurnia, N. S. Harahap, F. Yanto, and Pizaini, "Implementasi AES ECB dan Hashing MD5/SHA-256 Pada Aplikasi Penyuratan Android," *Journal of Computer System and Informatics (JoSYC)*, vol. 5, no. 1, pp. 113–126, 2023.
[7] National Institute of Standards and Technology, *Advanced Encryption Standard (AES) (FIPS PUB 197-upd1)*, U.S. Dept. of Commerce, 2023.
[8] National Institute of Standards and Technology, *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*, NIST SP 800-22 Rev. 1a, 2010.