Jurnal ELTIKOM :
Jurnal Teknik Elektro, Teknologi Informasi dan Komputer

Vol. 6, No. 2, Desember 2022, hal. 186-194
ISSN 2598-3245 (Print), ISSN 2598-3288 (Online)
DOI: http://doi.org/10.31961/eltikom.v6i2.574
Tersedia online di http://eltikom.poliban.ac.id

PENINGKATAN KEAMANAN DATA END-TO-END SMART DOOR MENGGUNAKAN ADVANCED ENCRYPTION STANDARD

Whisnumurti Adhiwibowo, Alauddin Maulana Hirzan*, Muhammad Sani Prayogi
Fakultas Teknologi Informasi Dan Komunikasi, Universitas Semarang, Semarang, Indonesia
e-mail: whisnu@usm.ac.id, maulanahirzan@usm.ac.id, yogie@usm.ac.id

Diterima: 2 Februari 2022 – Direvisi: 3 Juni 2022 – Disetujui: 29 September 2022

ABSTRACT

Smart Home is one form of implementation of Internet of Things technology in the form of smart homes that can carry out management, monitoring, even reporting. In addition, smart homes can be equipped with security equipment such as Smart Door that can open or lock the door automatically when recognizing the homeowner's face. However, the current Smart Door model has a disadvantage where the stored data on the server and the device are not secured end-to-end. The homeowners' image data on the device is not encrypted with a specific algorithm and validation. Thus, the outside parties can use this high-risk problem to enter the house unnoticed. They disguised themselves as the homeowner by entering false data on the device. Based on this problem, this study has a purpose to increase the model's end-to-end security by implementing the Advanced Encryption Standard algorithm. In addition to increase the security level, the Truncated Decimal-converted SHA-1 checksum validation is added to prevent modifications in each image data. From the results of the model comparison experiment, there was an increase in device resource needs as much as 0.81% increase in process time; 18% CPU usage; 5.3% data usage; and 5.04% for the use of the entire process of memory. But the increase in performance needs is not comparable to the security features presented by the Advanced Encryption Standard algorithm in securing data and servers. So that with improvisation this security is expected to improve the data security of homeowners from outside parties.

***Keywords**: Advanced Encryption Standard, Face Recognition, Internet of Things, OpenCV, Smart Door.*

ABSTRAK

Smart Home merupakan salah satu bentuk implementasi teknologi Internet of Things dalam bentuk rumah cerdas yang dapat melalukan manajemen, pemantauan, bahkan pelaporan. Selain itu rumah cerdas dapat dilengkapi dengan peralatan keamanan seperti smart door yang dapat membuka maupun mengunci pintu secara otomatis ketika mengenali wajah pemilik rumah. Namun model smart door ini memiliki kelemahan yang di mana data yang tersimpan di dalam server maupun perangkat tidak diamankan secara end-to-end. Perangkat yang menyimpan data-data gambar pemilik rumah tidak dienkripsi dengan algoritma tertentu maupun validasi keaslian data gambar. Sehingga masalah ini dapat dimanfaatkan pihak luar dengan melakukan masquerading atau menyamar dengan cara memasukkan data palsu di dalam perangkat. Berdasarkan masalah yang sudah dideskripsikan, penelitian ini memiliki tujuan untuk meningkatkan keamanan data end-to-end model dengan algoritma Advanced Encryption Standard. Selain itu, penelitian ini juga melengkapi tingkat keamanan dengan validasi integritas data terenkripsi menggunakan teknik Truncated Decimal-converted SHA-1 Checksum untuk membuat nilai hash unik yang dapat mencegah modifikasi di masing-masing data gambar. Dari hasil eksperimen perbandingan model yang dilakukan, terjadi kenaikan kebutuhan sumber daya perangkat sebanyak 0,81% peningkatan waktu proses, 18% penggunaan CPU, 5,3% penggunaan data, dan 5,04% untuk penggunaan memori seluruh proses. Namun peningkatan kebutuhan kinerja ini tidak sebanding dengan fitur keamanan yang dihadirkan oleh algoritma Advanced Encryption Standard dalam mengamankan data perangkat dan server. Sehingga dengan improvisasi keamanan ini diharapkan dapat meningkatkan keamanan data pemilik rumah dari pihak luar.

***Kata Kunci**: Internet of Things, OpenCV, Pengenalan Wajah, Smart Door, Standar Enkripsi Tingkat Lanjut.*

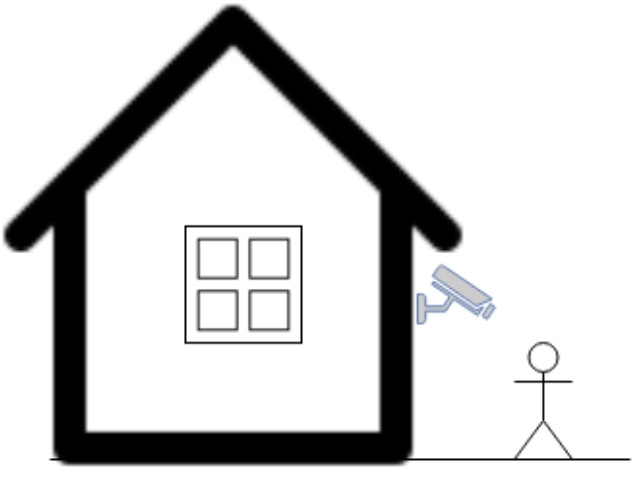
I. PENDAHULUAN

PERKEMBANGAN teknologi dengan hadirnya *Internet of Things* memungkinkan perangkat-perangkat maupun objek benda untuk dapat terkoneksi dengan internet. Benda-benda tersebut dapat mengirimkan informasi ke pemilik melalui internet sehingga para pemilik dapat melakukan monitoring perangkat tersebut di manapun dan kapanpun. Salah satu bentuk implementasi dari *Internet of Things* adalah pintu cerdas yang telah dilengkapi dengan kamera, rumah cerdas ini dapat mengenali pola wajah pemilik rumah yang bisa disebut sebagai *Smart Door.*

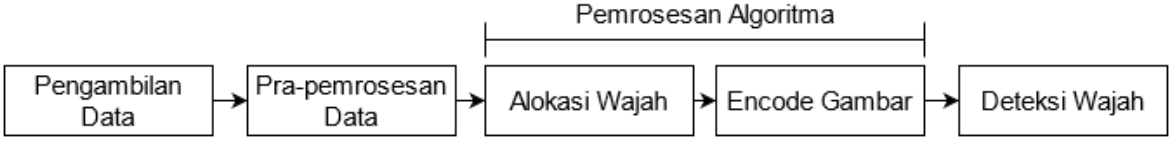
Smart Door dilengkapi dengan perangkat pemrosesan data pola, sensor kamera untuk mendeteksi wajah, dan aktuator untuk membuka kunci pintu. Gambar 4 ini merupakan ilustrasi daripada *smart house* yang memiliki teknologi *smart door* berkamera. Kamera yang terdapat di perangkat *smart door* ini berfungsi sebagai sensor visual yang membaca wajah manusia sebagai masukkan. Ketika pemilik rumah atau tamu datang dan berdiri tepat di depan pintu, perangkat *smart door* akan melakukan pembacaan wajah. Dari titik ini *smart door* dapat menentukan apakah pemilik rumah sudah pulang, atau tamu pemilik rumah yang datang ke rumah tersebut. Jika pemiliki rumah yang tedeteksi, maka kunci pintu akan terbuka. Dan sebaliknya, *smart door* akan mengabaikan jika bukan pemilik rumah yang terdeteksi.

Teknik deteksi perangkat ini menggunakan algoritma deteksi wajah dengan data pemilik rumah yang sudah dipelajari sebelumnya. Gambar 5 adalah alur proses bagaimana proses pembelajaran dan deteksi wajah berjalan. Proses ini dimulai dengan mengambil data-data gambar wajah yang disimpan dalam basis data internal rumah. Setelah data selesai diambil, proses berlanjut dengan melakukan pra-pemrosesan data untuk memastikan data bersih dan siap digunakan. Data yang sudah siap diproses kemudian dilanjutkan dengan deteksi atau alokasi wajah yang sudah dilabeli sebelumnya. Setelah proses alokasi wajah, dilanjutkan dengan *encoding* dengan cara memasangkan label alokasi dengan gambar utamanya. Hasil dari *encode* ini kemudian digunakan untuk mendeteksi wajah yang ada di kamera [1-4].

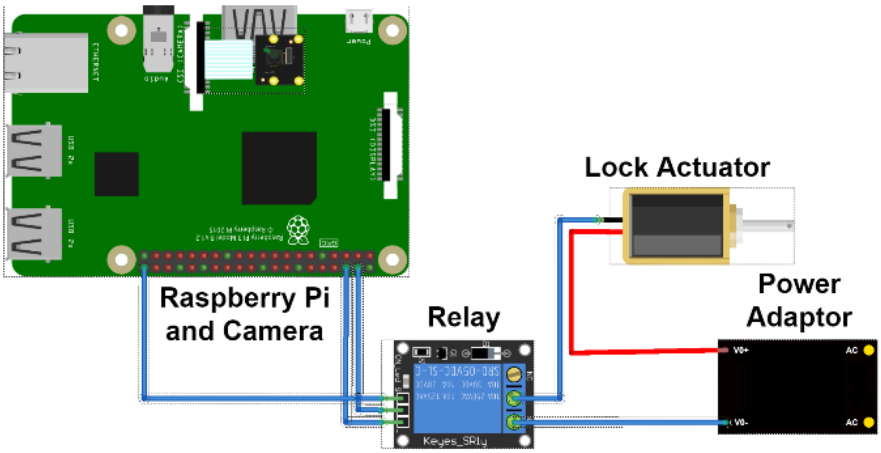
Jika pola wajah yang terdeteksi memiliki kemiripan dengan pola yang terdata, maka pintu akan terbuka otomatis. Dan begitu juga sebaliknya. Teknologi pengenalan wajah sebelumnya telah diimplementasikan untuk beberapa hal, salah satunya untuk kehadiran mahasiswa. Model deteksi satu ini menggunakan teknologi *OpenCV* dan Raspberry Pi untuk mendeteksi kehadiran siswa di sekolah [5][6]. Salah satu bentuk implementasi dari teknologi yang juga menjadi fokus daripada penelitian ini adalah *smart door* bentuk pengawasan keamanan dari *smart home* [7-11].



Gambar 4. Ilustrasi Deteksi Wajah *Smart Door*



Gambar 5. Metode Deteksi Wajah Model Standar



Gambar 3. Rangkaian elektronik dari model smartdoor yang diusulkan

Dibalik kenyamanan pengamanan yang dihadirkan oleh *keyless smart door* ini, terdapat kelemahan yang cukup berbahaya yang tidak disadari. Kelemahan dari model saat ini terletak di keamanan datanya adalah tidak adanya enkripsi *end-to-end* untuk melindungi data yang berada di dalam server hingga perangkat. Data gambar pemilik rumah yang tersimpan di server maupun perangkat tidak terenkripsi dengan algoritma keamanan maupun dilengkapi dengan validasi keaslian dapat berakibat buruk bagi pemilik rumah. Keamanan data yang rendah ini dapat dieksploitasi oleh pihak luar untuk masuk ke dalam rumah secara senyap dengan cara *masquerading* atau penyamaran. Cara ini dilakukan dengan menambah atau mengubah data internal rumah sehingga pihak luar bisa masuk rumah dengan mudah [12][13]. Karena tidak adanya enkripsi maupun validasi data tersebut, *smart door* langsung mengakses data palsu tersebut dan mengenali pihak luar tersebut seolah-olah sebagai pemilik rumah. Hal ini dapat berakibat serius yang dapat menyebabkan kerugian material yang banyak. Pihak luar yang memiliki akses bebas ke seluruh ruangan rumah, dapat mengambil barang-barang yang ada di dalam rumah tanpa kecurigaan.

Berdasarkan masalah keamanan mengenai teknologi *smart door* saat ini, penelitian ini memiliki tujuan untuk meningkatkan keamanan dalam segi data *end-to-end*. Untuk dapat meningkatkan keamanan daripada model *smartdoor* saat ini, penelitian ini membuat sebuah model *smart door* dengan algoritma *Advanced Encryption Standard* untuk mengamankan data di dalam server dan perangkat pendeteksi. Algoritma ini menyediakan pengamanan data dengan kekuatan enkripsi yang kuat dan memerlukan waktu yang lama untuk memecahnya tanpa kunci yang sesuai [14]. Selain itu algoritma ini juga sangat ringan dijalankan di perangkat IoT tanpa menghambat proses lainnya [15][16].

Kontribusi maupun keterbaruan utama dalam penelitian ini adalah peningkatan keamanan model *smart door* saat ini yang tidak memiliki teknologi pengamanan kriptografi maupun validasi data gambar yang rentan eksploitasi pihak luar. Model yang diteliti ini diharapkan dapat meningkatkan keamanan data gambar dalam bentuk enkripsi dan nilai *checksum hash* yang unik untuk memastikan integritas file tetap terjaga. Sehingga pemilik rumah tetap merasa aman meskipun rumah dalam keadaan kosong dalam waktu yang panjang.

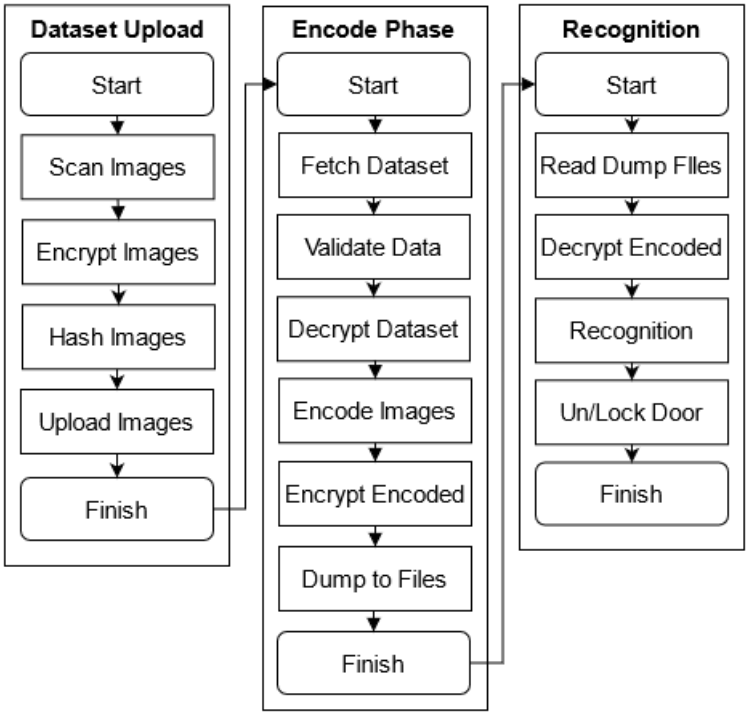
II. METODE PENELITIAN

A. Desain Perangkat, Topologi dan Algoritma Model

Untuk membangun sebuah model yang memiliki tingkat keamanan yang baik dari model sebelumnya, penelitian ini menggunakan perangkat pemrosesan *Internet of Things* dengan *Raspberry Pi 3B plus*. Perangkat ini memiliki tingkat pemrosesan yang lebih baik dibandingkan perangkat IoT lainnya sehingga sering digunakan untuk *smart door*. Selain itu, perangkat ini dapat menyediakan kinerja komputasi yang mencukupi untuk algoritma *Advanced Encryption Standard* [17].

Rangkaian elektronik yang ditunjukkan oleh Gambar 3 sebagai panduan peletakkan komponen yang sesuai agar berfungsi dengan baik. Rangkaian ini terdiri dari beberapa bagian utama seperti Raspberry Pi sebagai perangkat pemrosesan data terletak di atas kiri, *Pi Camera* yang sudah terpasang di dalam perangkat Raspberry Pi sebagai sensor pendeteksi wajah, *relay* yang terletak dibagian tengah bawah

Gambar 4. Topologi jaringan nirkabel untuk komunikasi model



Gambar 5. Alur Proses Pengenalan Wajah

untuk mengatur aliran listrik dari *Power Adaptor* ke *Lock Actuator.* Agar bisa mudah mengontrol *relay, relay* harus dihubungkan dengan Raspberry Pi melalui pin *General Purpose Input Output* yang telah ditentukan agar bisa dikendalikan via Program. *Lock Actuator* berupa kunci *solenoid* yang terletak di bagian kanan tengah, terhubung dengan *relay* dan *power adaptor.* Kunci ini jika diberi tegangan listrik, maka kunci akan terbuka. Komponen terakhir adalah *power adaptor* yang menyediakan tegangan listrik 12V ke kunci *solenoid* melalui pengendalian *relay.*

Selain penataan perangkat, konfigurasi komunikasi antara perangkat dan server juga diatur untuk mensimulasikan *smart door.* Gambar 4 mengilustrasikan bagaimana jaringan nirkabel dikonfigurasikan agar perangkat *smart door* dapat berkomunikasi dengan server tanpa harus menggunakan kabel fisik. Selain itu, topologi juga menunjukkan peran-peran yang dipunyai masing-masing perangkat, *Smart Door* bertindak sebagai klien yang meminta data dari server, dan sebaliknya server bertugas menyediakan data wajah untuk klien. Pemilihan jaringan nirkabel ini dikarenakan bebas dari pemasangan kabel yang rumit, rendahnya biaya yang digunakan, dan efisiensi ketika menambahkan perangkat baru [18].

Tahap berikutnya dari pengembangan model pengenalan wajah adalah pengaturan algoritma kripografi dan pengecekan integritas data seperti yang ditunjukkan di Gambar 5. Alur proses ini memiliki tiga fase utama yang dimulai dari *Dataset Upload* yang berfungsi sebagai pengunggahan dataset, dilanjutkan dengan *Encode Phase* yang berfungsi sebagai pengubah data gambar menjadi pola, dan *Recognition* yang berfungsi sebagai pendeteksi wajah menggunakan pola yang sudah disiapkan. Proses pendeteksian wajah dimulai dengan pengunggahan data dilakukan dengan membaca semua file gambar yang akan diunggah. Kemudian proses ini berlanjut dengan mengenkripsi setiap gambar yang ada, mengambil *hash* unik dari file terenkripsi lalu mengunggahnya ke server basis data. Proses enkripsi dan dekripsi sendiri menggunakan algoritma *Advanced Encryption Standard* dengan ukuran kunci 128-

bit, dan mode *Cipher FeedBack* (CFB). Sedangkan untuk validasi daripada data-data yang tersimpan di dalam server menggunakan *truncated decimal-converted SHA-1 checksum.* Teknik validasi memadukan tiga macam hal yaitu: *hashing* dengan algoritma *SHA-1,* konversi hexadesimal ke desimal, dan *truncation* atau pemendekan. Sehingga nilai daripada validasi data-data yang ada di dalam server menjadi unik dan sulit untuk ditiru oleh pihak luar.

Dalam pembuatan *hash* unik, penelitian ini mengambil pendekatan yang berbeda dengan *checksum hash* biasa. Untuk meningkatkan keamanan serta keunikan nilai *hash,* penelitian ini menambahkan proses baru berupa konversi nominal dan pemendekkan (*truncation).* Sehingga dapat mengurangi kemungkinan terjadinya nilai *hash* yang bertabrakan. Alur proses pembuatan *hash* unik dimulai dengan pengubahan data gambar berbentuk *binary* ke bentuk *String.* Selain itu, untuk memperjelas jenis *string* yang digunakan dalam *checksum hash,* data tersebut dikonfigurasikan dengan *encoding UTF-8.* Data yang telah dibentuk dalam format *string* ini kemudian diambil nilai *hash* nya dengan menggunakan algoritma *SHA-1* dalam nominal hexadesimal. Untuk membuat nilai *hash* ini menjadi lebih unik, proses dilanjutkan dengan pengubahan nominal hexadesimal ke dalam bentuk desimal. Persamaan 1 digunakan dalam mengubah nila hexadesimal menjadi desimal.

$$Nilai\ Desimal = \sum h_{n-1} x 16^{r-1} = h_n x 16^n + \cdots + h_1 x 16^1 + h_0 x 16^0 \quad (1)$$

Untuk mendapatkan nilai desimal dari hasil *checksum* dengan menggunakan *SHA-1*, Persamaan 1 ini membutuhkan nilai hexa (*h*) dengan letak digit (*n*) yang dikalikan dengan eksponen (*r*) dari kuadrat nilai 16. Kalkulasi ini dilakukan terus menerus hingga mencapai nilai digit *n* adalah 0 dan kemudian dijumlahkan secara total keseluruhannya. Hasil akhir dari konversi ini kemudian dipendekkan menjadi 10 karakter agar lebih mudah diidentifikasi dan mengurangi kebutuhan memori penyimpanan basis data. Proses penambahan proses kriptografi dan validasi ini dilakukan agar alur proses pengenalan wajah tidak mengalami perubahan yang banyak.

Proses berikutnya dilanjutkan dengan *Encoding Phase* yang dilakukan oleh perangkat *smartdoor* sendiri. Untuk memulai proses ini, perangkat perlu mengambil data yang ada di server disertai dengan pengecekan integritas data. Jika integritas data gagal, maka proses akan berhenti. Jika tidak, proses dilanjutkan ke proses *encoding.* Dalam proses *encoding* sendiri, perangkat akan menggunakan CPU untuk mengolah data gambar menjadi pola. Pola yang sudah selesai kemudian dipasangkan dengan gambar asli (tagging) kemudian di enkripsi kembali dan disimpan dalam bentuk file. Untuk proses pengenalan wajah *(Recognition),* perangkat akan membaca file, dan mendekripsi file tersebut. Pola-pola yang terdaftar di dalam file dijadikan acuan untuk mengenali pola wajah ketika proses pengenalan berlangsung. Ketika pola wajah pemilik rumah dikenali, maka secara otomatis pintu akan dibuka dan dikunci kembali.

B. Evaluasi Kinerja Model

Model ini kemudian diuji untuk mengevaluasi efek kinerja perangkat terhadap improvisasi keamanan yang telah dilakukan. Evaluasi yang dilakukan ini bertujuan untuk melihat perbedaan kinerja model yang telah ditingkatkan keamanannya dengan model standar pasaran saat ini. Dalam evaluasi yang dilakukan, ada beberapa indikator yang dijadikan pertimbangan evaluasi kinerja perangkat. Indikator evaluasi performa perangkat dalam ini adalah sebagai berikut: *Processing Time* berupa waktu yang dibutuhkan untuk menyelesaikan satu tugas (dalam satuan *second*), *CPU Usage* berupa penggunaan CPU ketika proses atau tugas berjalan (dalam satuan persen), *Data Size* berupa penggunaan memori yang dikhususkan untuk data saja (dalam satuan *MegaBytes*), dan *Unique Set Size* berupa penggunaan memori keseluruhan termasuk data dan kode itu sendiri (dalam satuan *MegaBytes*). Objek evaluasi dari penilitian ini adalah fase-fase proses deteksi wajah dari awal hingga akhir. Fase-fase tersebut berupa tiga fase utama seperti *Data Fetching; Encode Image;* dan *Face Recognition, dan* 3 fase keamanan tambahan berupa *Decrypt Image; Encrypt File;* dan *Decrypt File.* Untuk melihat persentase perbandingan kinerja model berdasarkan indikator-indikator yang sudah ditentukan, Persamaan 2 digunakan untuk mendapatkan nilai selisih persentase kinerja.

$$\Delta Kinerja(\%) = \frac{(I_{improvisasi} - I_{standar})}{I_{standar}} x100\% \quad (2)$$

TABEL 1
HASIL *BENCHMARK* KEDUA MODEL PENGENALAN WAJAH

Phase(s)	Indicator	Model Standar	Model Improvisasi
Data Fetching	Processing Time	12,36	11,14
	CPU Usage	0,45%	0,43%
	Data Usage	150,07	150,78
	Unique Set Size	179,27	179,52
Decrypt Image	Processing Time	0	5,20
	CPU Usage	0	24,68%
	Data Usage	0	216,18
	Unique Set Size	0	239,12
Encode Image	Processing Time	380,36	386,00
	CPU Usage	24,98%	24,97%
	Data Usage	219,28	220,70
	Unique Set Size	242,38	243,24
Encrypt Pickle	Processing Time	0	0,16
	CPU Usage	0	24,68%
	Data Usage	0	225,43
	Unique Set Size	0	247,57
Decrypt Pickle	Processing Time	0	0,16
	CPU Usage	0	25,14%
	Data Usage	0	213,60
	Unique Set Size	0	236,08
Face Recognition	Processing Time	34,47	28,00
	CPU Usage	28,03%	26,28%
	Data Usage	233,03	241,96
	Unique Set Size	245,92	256,95
Total	Processing Time (s)	427,19	430,67
	CPU Usage (%)	17,82%	21,03%
	Data Size	200,79	211,44
	Unique Set Size	222,52	233,74

Persamaan 2 ini terdiri dari selisih sumber daya komputer ($\Delta Kinerja$) yang didapatkan dari selisih hasil indikator dari model improvisasi ($I_{improvisasi}$) dengan hasil indikator model standar ($I_{standar}$). Hasil dari selisih ini kemudian dibagi dengan indicator model standar ($I_{standar}$) sebagai dasar penilaian. Untuk mendapatkan persentase perbandingan kinerja model, hasl ini dikalikan dengan 100% sebagai kalkulasi akhir evaluasi. Hasil dari perhitungan ini akan mendapatkan nilai perbedaan kinerja (*performance gap*) dengan model standar sebagai acuan dasar.

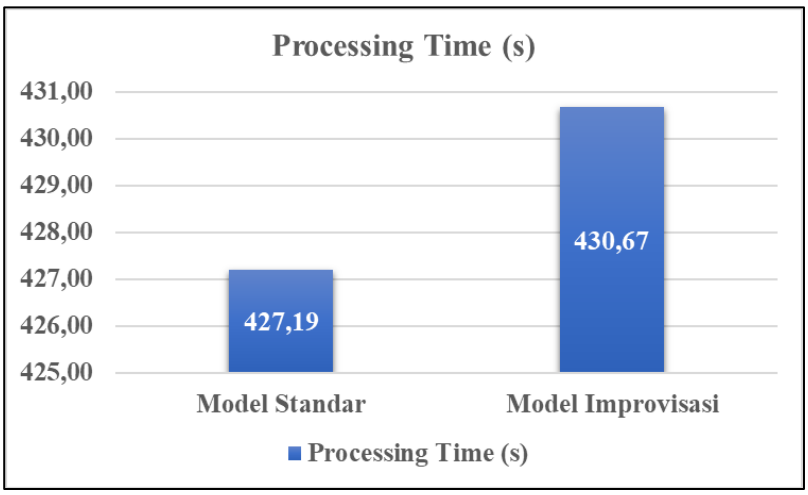
III. HASIL DAN PEMBAHASAN

Dari eksperimen yang dilakukan dengan menggunakan model standar dan model improvisasi, menghasilkan data indikator kinerja berupa: penggunaan CPU, penggunaan memori (*Data Size* dan *Unique Set Size*), dan waktu pemrosesan di masing-masing fase.

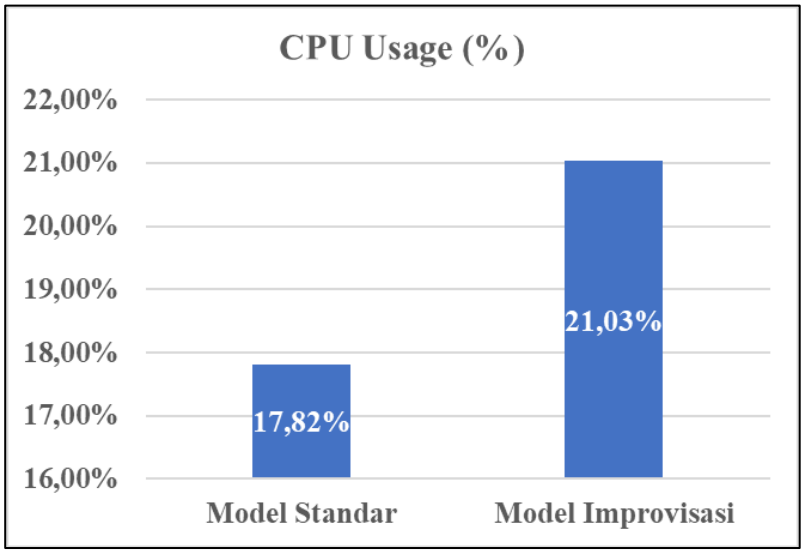
Berdasarkan hasil *benchmark* yang ditunjukkan dengan Tabel 1, model standar membutuhkan waktu selama 427,19 detik, penggunaan CPU sebanyak 17,82%, penggunaan memori data sebanyak 200,79 MB, dan penggunaan memori oleh proses sebanyak 222,52 MB. Evaluasi untuk model standar tidak menghitung fase-fase tambahan enkripsi karena bernilai 0. Sedangkan untuk model improvisasi membutuhkan waktu selama 430,67 detik, penggunaan CPU sebanyak 21,03%, penggunaan memori untuk data sebanyak 211,44 MB, dan penggunaan memori untuk seluruh proses sebanyak 233,74 MB. Hasil yang didapatkan oleh model improvisasi ini lebih tinggi dibandingkan oleh model standar.

Dari hasil evaluasi *benchmark* yang dilakukan, model standar mendapatkan beberapa nilai nol di fase-fase tambahan ini. Hal ini dikarenakan oleh model yang tidak menggunakan pengamanan data *end-to-end* sehingga perangkat hanya menjalani tiga fase utama pengenalan wajah. Berbeda dengan model standar, model improvisasi ini selain melalui tiga fase pengenalan wajah juga melalui tiga fase tambahan peningkatan keamanan. Sehingga jika dilihat dari hasil di Tabel 1, model improvisasi memiliki hasil *benchmark* yang lebih tinggi dibandingkan dengan model standar.

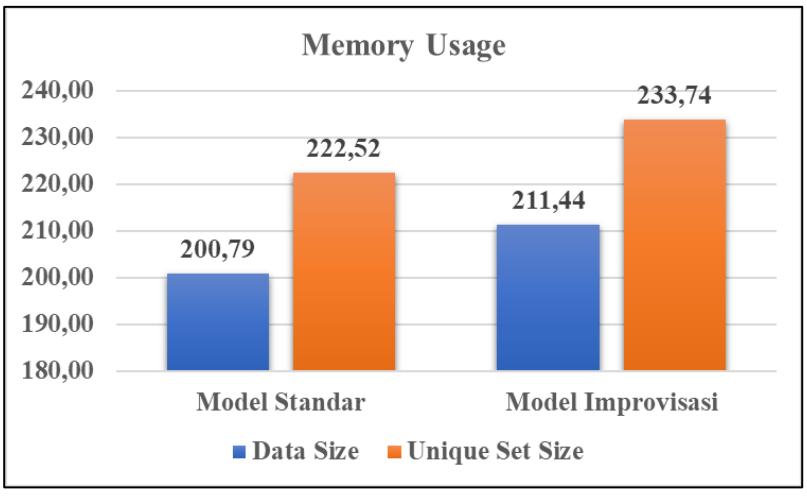
Proses evaluasi berikutnya adalah melakukan perhitungan persentase *performance gap* dari kedua model. Evaluasi ini menghitung kenaikan kinerja perangkat yang dilakukan oleh model improvisasi dengan menggunakan model standar sebagai acuan penilaian *performance gap* tersebut. Untuk dapat melihat secara detail perbedaan penggunaan sumber daya komputer maupun selisih dari kinerja masing-



Gambar 6. Perbandingan Waktu Pemrosesan Model



Gambar 7. Perbandingan Penggunaan CPU



Gambar 8. Perbandingan Penggunaan Memori Model

masing model, Gambar 6-8 mengilustrasikan kinerja masing-masing model ditiap-tiap indikator kinerja yang sudah ditentukan.

Dari Gambar 6 menampilkan kebutuhan total waktu yang diperlukan untuk model melakukan pengambilan data hingga mengenali wajah. Berdasarkan gambar tersebut terjadi perbedaan waktu

proses mencapai 3,48 detik dengan *performance gap* mencapai 0,81% lebih lama dari model standar. Dari sisi penggunaan CPU yang ditunjukkan di Gambar 7, model yang telah diimprovisasi membutuhkan CPU dengan rata-rata 3,21%, dan *performance gap* mencapai 18% lebih tinggi dibandingkan model standar. Dalam penggunaan memori penyimpanan data yang ditunjukkan dengan Gambar 8, model improvisasi ini membutuhkan 10,65MB, dan *performance gap* 2,58% lebih banyak dibandingkan model standar. Evaluasi terakhir di gambar ini adalah penggunaan memori untuk keseluruhan proses, model improvisasi juga mengalami peningkatan mencapai 11,22MB, dan *performance gap* mencapai 2,46% lebih tinggi dibandingkan model standar.

Berdasarkan data-data dari hasil eksperimen maupun grafik perbandingan kinerja model, bisa diketahui bahwa model yang telah mengalami improvisasi ini membutuhkan sumber daya yang lebih. Dari segi kebutuhan waktu pemrosesan, penggunaan CPU, penggunaan memori Data, dan memori proses, semua aspek ini mengalami peningkatan. Kenaikan-kenaikan ini disebabkan adanya tiga fase tambahan yang digunakan untuk meningkatkan keamanan model pendeteksi wajah. Tiga fase tambahan ini menggunakan algoritma *Advanced Encryption Standard* untuk melindungi dataset dan pola gambar dari perubahan yang tidak diinginkan, serta verifikasi integritas data untuk memastika data tidak mengalami perubahan dari pihak luar. Jika dilihat dari segi waktu pemrosesan, kenaikan yang dibutuhkan untuk memroses enkripsi tidak signifikan dan tidak mempengaruhi respon yang dibutuhkan untuk membuka pintu. Dari sisi penggunaan CPU, terjadi kenaikan yang cukup banyak mencapai 18% dibandingkan model standar. Hal ini diakibatkan oleh algoritma enkripsi yang membutuhkan kemampuan CPU untuk melakukan kalkulasi enkripsi maupun dekripsi di satu inti prosesor. Meskipun terjadi kenaikan, namun hal ini juga tidak mempengaruh kinerja sistem. Perangkat yang digunakan dalam perangkat ini memiliki empat inti pemrosesan, sehingga masih terdapat ruang lain untuk sistem operasi bekerja. Dan sisi terakhir dari evaluasi kinerja ini adalah penggunaan memori, model improvisasi ini mengalami kenaikan hingga 5% saja. Kenaikan penggunaan memori ini disebabkan oleh produk tambahan yang dihasilkan oleh algoritma enkripsi seperti *Initialization Vector* yang digunakan kembali untuk mendekripsikan *cipher* data gambar. Kenaikan 5% dari penggunaan memori model standar ini juga tidak memiliki efek yang buruk terhadap perangkat, karena perangkat memiliki memori yang cukup banyak dan teknologi *memory management* oleh kernel Linux yang dapat menghindari terjadinya *system crash* dengan cara *swapping*.

Peningkatan-peningkatan yang terjadi ini pada dasarnya hanya mempertimbangkan salah satu faktor saja. Terdapat beberapa faktor lainnya yang dapat meningkatkan evaluasi kinerja di tiap-tiap fase seperti:jumlah data gambar, ukuran data gambar, banyaknya proses yang berjalan secara *background* di perangkat, spesifikasi perangkat, bahkan sistem operasi yang digunakan oleh perangkat dapat mempengaruhi kinerja perangkat secara internal. Semakin besar maupun semakin banyak data gambar yang digunakan sebagai pendeteksi pola wajah, maka semakin lama, dan banyak pula sumber daya komputer yang digunakan untuk mengolah data-data tersebut. Faktor eksternal yang dapat mempengaruhi waktu lamanya pemrosesan seperti kurangnya penerangan yang menyebabkan kamera tidak dapat mendeteksi wajah secara baik.

Berdasarkan fakta-fakta yang didapatkan melalui eksperimen evaluasi ini, didapatkan selisih perbedaan yang menjadi bukti efek implementasi algoritma *Advanced Encryption Standard* untuk meningkatkan keamanan model pengenalan wajah *smart door*. Namun jika ditinjau dari segi *feasibility* dan s*ecurity,* kenaikan penggunaan sumber daya ini tidak sebanding dengan keamanan yang ditawarkan oleh algoritma kriptografi ini. Selain itu, kenaikan-kenaikan kebutuhan sumber daya ini tidak memiliki efek jangka pendek maupun panjang yang buruk ke perangkat. Sehingga dapat disimpulkan bahwa model pengenalan wajah *smart door* dapat ditingkatkan keamanannya dengan efek kinerja perangkat yang tidak signifikan baik dalam jangka pendek maupun panjang.

IV. KESIMPULAN

Model *smart door* dengan saat ini memiliki kelemahan yang cukup berbahaya bagi para pemilik *smart home.* Karena teknologi *smart door* ini mengandalkan data yang disimpan di dalam perangkat dan tidak terenkripsi secara *end-to-end,* dapat meningkatkan resiko keamanan rumah. Kelemahan ini memungkinkan pihak luar untuk melakukan *masquerading* dengan cara memasukkan gambar-gambar mereka ke dalam sistem untuk masuk ke dalam rumah. Oleh karena itu untuk memitigasi hal itu terjadi, penelitian ini meningkatkan keamanan model *smart door* dengan menggunakan algoritma *Advanced Encryption Standard* dan verifikasi integritas data menggunakan teknik *truncated decimal-converted*

SHA-1 checksum. Dari hasil evaluasi perbandingan kinerja yang dilakukan dengan model improvisasi dan standar, penelitian ini mendapatkan data kenaikan penggunaan sumber daya dengan nilai 0,81% lebih banyak waktu untuk memroses, 18% lebih banyak CPU, 5,3% lebih banyak memori untuk data, dan 5,04% lebih banyak memori untuk keseluruhan proses model. Meski terjadi kenaikan kebutuhan penggunaan sumber daya perangkat, namun peningkatan ini tidak memiliki dampak yang signifikan terhadap perangkat. Sehingga hal ini tidak setimpal dengan keamanan yang dihadirkan oleh enkripsi algoritma kriptografi dan teknik *hashing* yang unik melindungi data pemilik rumah. Kedepannya diharapkan model pengenalan wajah di *smart door* dapat ditingkatkan kembali dengan teknologi seperti *block chain* untuk menghindari modifikasi salah satu data wajah pemilik rumah.

UCAPAN TERIMA KASIH

Ucapan terima kasih kepada Lembaga Penelitian dan Pengabdian Kepada Masyarakat (LPPM) Universitas Semarang atas dana penelitian serta kesempatannya.

DAFTAR PUSTAKA

[1] A. Munir, S. Kashif Ehsan, S. M. Mohsin Raza, and M. Mudassir, 'Face and speech recognition based smart home', in 2019 International Conference on Engineering and Emerging Technologies, ICEET 2019, Lahore, Pakistan, 2019, pp. 1–5. doi: 10.1109/CEET1.2019.8711849.

[2] T. S. Gunawan, M. H. H. Gani, F. D. A. Rahman, and M. Kartiwi, 'Development of face recognition on raspberry pi for security enhancement of smart home system', Indones. J. Electr. Eng. Inform., vol. 5, no. 4, pp. 317–325, 2017, doi: 10.11591/ijeei.v5i4.361.

[3] N. Mustakim, N. Hossain, M. M. Rahman, N. Islam, Z. H. Sayem, and Md. A. Z. Mamun, 'Face Recognition System Based on Raspberry Pi Platform', in 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1–4. doi: 10.1109/ICASERT.2019.8934485.

[4] M. Bansal, 'Face Recognition Implementation on Raspberrypi Using OpenCV and Python', Int. J. Comput. Eng. Technol. IJCET, vol. 10, no. 3, pp. 141–144, May 2019, doi: 10.34218/ijcet.10.3.2019.016.

[5] P. Pasumarti and P. Purna Sekhar, 'Classroom Attendance Using Face Detection and Raspberry-Pi', Int. Res. J. Eng. Technol., vol. 5, no. 3, pp. 167–171, Mar. 2018.

[6] A. S. Hasban et al., 'Face recognition for Student Attendance using Raspberry Pi', in 2019 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE), Malacca, Malaysia, 2019, pp. 1–5. doi: 10.1109/APACE47377.2019.9020758.

[7] M. Nakrani, A. D. Deshmukh, A. D. Deshmukh, M. G. Nakrani, D. L. Bhuyar, and U. B. Shinde, 'Face Recognition Using OpenCv Based On IoT for Smart Door', in International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur - India, 2019, pp. 1066–1073. doi: http://dx.doi.org/10.2139/ssrn.3356332.

[8] A. Nag, N. J. N, and M. Kalmath, 'IoT Based Door Access Control Using Face Recognition', in 2018 3rd International Conference for Convergence in Technology (I2CT), Pune, India, 2018, pp. 1–3. doi: 10.1109/I2CT.2018.8529749.

[9] K. Sethi, S. Kaul, I. Patel, and R. Sujatha, 'FaceLock Homes: A Contactless Smart Home Security System to Prevent COVID Transmission', in 2021 Sixth International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 2021, pp. 75–79. doi: 10.1109/WiSPNET51692.2021.9419453.

[10] G. Lulla, A. Kumar, G. Pole, and G. Deshmukh, 'IoT Based Smart Security and Surveillance System', in 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2021, pp. 385–390. doi: 10.1109/ESCI50559.2021.9396843.

[11] G. S. Nagpal, G. Singh, J. Singh, and N. Yadav, 'Facial Detection and Recognition using OpenCV on Raspberry Pi Zero', in 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 2018, pp. 945–950. doi: 10.1109/ICACCCN.2018.8748389.

[12] S. Kumar, S. Singh, and J. Kumar, 'A comparative study on face spoofing attacks', in 2017 International Conference on Computing, Communication and Automation (ICCCA 2017), May 2017, pp. 1104–1108. doi: 10.1109/CCAA.2017.8229961.

[13] S. Mare, L. Girvin, F. Roesner, and T. Kohno, 'Consumer Smart Homes: Where We Are and Where We Need to Go', in Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications, Santa Cruz CA USA, Feb. 2019, pp. 117–122. doi: 10.1145/3301293.3302371.

[14] Pasquale Arpaia, Francesco Bonavolontà, and Antonella Cioffi, 'Security vulnerability in Internet of Things sensor networks protected by Advanced Encryption Standard', in 2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT, Roma, Italy, 2020, pp. 452–457. doi: 10.1109/MetroInd4.0IoT48571.2020.9138236.

[15] C.-W. Hung and W.-T. Hsu, 'Power Consumption and Calculation Requirement Analysis of AES for WSN IoT', Sensors, vol. 18, no. 6, p. 1675, May 2018, doi: 10.3390/s18061675.

[16] P. S. Munoz, N. Tran, B. Craig, B. Dezfouli, and Y. Liu, 'Analyzing the Resource Utilization of AES Encryption on IoT Devices', in 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Honolulu, HI, USA, Nov. 2018, pp. 1200–1207. doi: 10.23919/APSIPA.2018.8659779.

[17] E. Fernando, D. Agustin, M. Irsan, D. F. Murad, H. Rohayani, and D. Sujana, 'Performance Comparison of Symmetries Encryption Algorithm AES and DES With Raspberry Pi', in 2019 International Conference on Sustainable Information Engineering and Technology (SIET), Lombok, Indonesia, Sep. 2019, pp. 353–357. doi: 10.1109/SIET48054.2019.8986122.

[18] M. Li, W. Gu, W. Chen, Y. He, Y. Wu, and Y. Zhang, 'Smart home : architecture, technologies and systems', in Procedia Computer Science, 2018, vol. 131, pp. 393–400. doi: 10.1016/j.procs.2018.04.219.