BAB 2
TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Kriptografi

Kriptografi berasal dari bahasa Yunani, crypto dan graphia. Crypto berarti secret (rahasia) dan graphia berarti writing (tulisan). Menurut terminologinya, Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan atau data dikirim dari suatu tempat ke tempat yang lain. Menurut penelitian tentang Mengenal Proses Perhitungan Enkripsi Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES)[7].

Menurut (Dafid, D, 2006), Kata kriptografi *(Cryptography)* berasal dari bahasa Yunani yaitu dari kata *Cryptos* yang artinya tersembunyi dan *Graphein* yang artinya menulis. Kriptografi dapat diartikan sebagai suatu ilmu ataupun seni yang mempelajari bagaimana sebuah data dikonversi ke bentuk tertentu yang sulit untuk dimengerti (Schneier B, 1996). Ada 4 tujuan mendasar dari ilmu kriptografi ini yang juga merupakan aspek keamanan informasi, yaitu [8]:

1. Kerahasiaan, adalah layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki otoritas atau kunci rahasia untuk membuka/mengupas informasi yang telah disandi.
2. Integritas data, adalah berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain kedalam data yang sebenarnya.
3. Autentikasi, adalah berhubungan dengan identifikasi/pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keaslian, isi datanya, waktu pengiriman, dan lain-lain.

4. Non-repudiasi., atau nirpenyangkalan adalah usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman/terciptanya suatu informasi oleh yang mengirimkan/membuat.

Suatu data yang tidak disandikan disebut *plaintext* atau *cleartext*. Sedangkan data yang telah tersandikan disebut *ciphertext*. Proses yang dilakukan untuk mengubah *plaintext* menjadi *ciphertext* disebut enkripsi *(encryption)* atau *encipherment*. Sedangkan proses untuk mengubah *ciphertext* kembali ke *plaintext* disebut dekripsi *(decryption)* atau *decipherment*. Dalam kriptografi diperlukan parameter yang digunakan untuk proses konversi data yaitu suatu set kunci. Enkripsi dan dekripsi data dikontrol oleh sebuah kunci atau beberapa kunci.

Berdasarkan kunci yang dipakai, algoritma kriptografi dapat dibedakan atas dua jenis yaitu algoritma simetrik *(symmetric)* dan asimetrik *(asymmetric)*.

Kriptografi Simetrik

Sebuah algoritma juga dikenal sebagai algoritma simetris atau algoritma enkripsi tradisional. Gunakan kunci yang sama untuk enkripsi dan dekripsi[9]. Algoritma Enkripsi simetris dapat dibagi menjadi dua kategori: enkripsi aliran dan enkripsi aliran. Blok algoritma (enkripsi blok). Dalam algoritma aliran, proses pengkodean adalah sebagai berikut: Data 1-bit atau 1-byte. Proses pengkodean menggunakan algoritma blok. Hal ini dimaksudkan untuk satu set data bit atau byte (blok demi blok). Contoh algoritma kunci simetris DES (Standar Enkripsi Data), *Blowfish*, *Twofish*, MARS, IDEA, 3DES (berlaku tiga kali), AES (Advanced Encryption Standard), yang nama aslinya adalah Rijndael.

2.1.2 *Advanced Encryption Standard (AES)*

AES merupakan sistem penyandian blok yang bersifat non-Feistel karena AES menggunakan komponen yang selalu memiliki invers dengan panjang blok 128 bit. Kunci AES menggunakan proses yang berulang yang disebut dengan ronde. Proses di dalam AES merupakan transformasi terhadap state. Sebuah teks asli dalam blok (128 bit) terlebih dahulu diorganisir sebagai state. Enkripsi AES

adalah transformasi terhadap state secara berulang dalam beberapa ronde. State yang menjadi keluara ronde k menjadi masukan untuk ronde ke-k +1. Pada Proses enkripsi awalnya teks asli dibentuk sebagai sebuah state. Kemudian sebelum ronde 1 dimulai blok teks asli dicampur dengan kunci ronde ke-0 (transformasi ini disebut AddRoundKey). Setelah itu, ronde ke-1 sampai dengan ronde ke-(Nr-1) dengan Nr adalah jumlah ronde. AES menggunakan 4 jenis transformasi yaitu:

1. SubBytes, sebagai transformasi subtutusi.
2. ShiftRows, sebagai transformasi permutasi.
3. MixColumns, sebagai transformasi pengacakan.
4. AddRoundKey, sebagai transformasi penambahan kunci.

ada ronde terakhir, yaitu ronde ke-Nr dilakukan transformsi serupa dengan ronde lain namun tanpa transfomasi serupa dengan ronde lain namun tanpa transformasi MixColumns. Adapun proses enkripsi yang dilakukan menggunakan algoritma AES yaitu:

1. *AddRoundKey*

Proses ini dilakukan di awal ronde dengan melakukan operasi XOR tiap *byte* pada matriks *state*(*plaintext*)dengan tiap *byte* pada *cipherkey*, tahap ini disebut juga *initial round*.

2. *SubBytes*

SubBytes merupakan transformasi *byte* dimana setiap elemen pada *state* akan dipetakan dengan menggunakan sebuah tabel substitusi (S-Box). Untuk setiap *byte* pada *state* dinyatakan dengan *S*'[*r*, *c*]. *S*'[*r*, *c*] adalah elemen di dalam tabel subtitusi yang merupakan perpotongan baris (x)dengan kolom (y).

3. *Shiftrows*

Shiftrows pada dasarnya adalah proses pergeseran *bit* dimana *bit* paling kiri akan dipindahkan menjadi *bit* paling kanan (rotasi *bit*). Namun jumlah pergeseran yang dilakukan berbeda, tergantung untuk setiap barisnya. Baris pertama tidak terjadi pergeseran. Setiap *byte* dari baris kedua pada matriks *state* digeser satu *byte* ke kiri. Selanjutnya baris ketiga digeser ke kiri

sebanyak dua *byte* dan pada baris keempat digeser ke kiri sebanyak tiga byte. Proses ini bertujuan untuk menghasilkan *diffusion* yakni dengan menyebarkan pengaruh transformasi nonlinear pada baris-baris matriks *state* untuk putaran selanjutnya.

4. *MixColumns*

Pada proses *MixColumns*, tiap kolom dari matriks *state* dilakukan operasi perkalian. Hal ini bertujuan untuk menyebarkan pengaruh setiap bit *plaintext* dan *cipherkey* terhadap *ciphertext* yang dihasilkan, pada arah kolom matriks *state.* Setiap kolom matriks *state* diperlakukan sebagai polinomial empat suku dalam *Galois field,* kemudian dikalikan dengan modulo $(X^8+X^4+X^3+X+1)$. Operasi *MixColumns* juga dapat dipandang sebagai perkalian matriks, dengan mengalikan empat bilangan di dalam *Galois field MixColumns* juga disebut sebagai proses mengalikan setiap kolom dengan matriks berikut:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

5. *AddRoundKey*

Dalam tahap *AddRoundKey* ini, *cipherkey* yang telah ada di ekspansikan terlebih dahulu maka akan di dapat *roundkey* yang akan digunakan untuk proses selanjutnya. Kemudian setiap *byte* dari matriks *state* keluaran proses *MixColumns* dilakukan operasi XOR dengan setiap *byte* dari *roundkey*. Proses *round* atau proses *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey* dilakukan hingga putaran ke-n dengan cara yang sama. Sedangkan utuk putaran terakhir atau disebut juga *final round* proses *SubBytes, ShiftRows,* dan *AddRoundKey* tetap dilakukan tetapi proses *MixColumns* tidak dilakukan. Proses dekripsi yang dilakukan menggunakan algoritma *Advanced Encryption Standard* (AES) yaitu:

1. *AddRoundKey*

Inverse atau kebalikan dari tahap *AddRoundKey* adalah operasi XOR antara byte-byte matriks *state* yangdisusun dari *ciphertext* dengan byte-byte *roundkey* yang dibangkitkan sebelumnya. *Roundkey* yang digunakandi setiap

iterasinya berkebalikan dengan *roundkey* yang ada pada proses enkripsi. *Inverse* dari transformasiini digunakan untuk proses dekripsi.

2. *Inverse SubBytes*

Inverse SubBytes juga merupakan transformasi *bytes* yang berkebalikan dengan transformasi *SubBytes*. Pada *InvSubBytes*, tiap elemen pada state dipetakan dengan menggunakan tabel *Inverse S-Box.*

3. *InverseShiftRow*

Untuk proses dekripsinya dilakukan proses *Inverse* dari transformasi *ShiftRows. InverseShiftRow s*adalah transformasi byte yang berkebalikan dengan transformasi *ShiftRows*. Pada transformasi *InverseShiftRows*, dilakukan pergeseran *bit* ke kanan sedangkan pada *ShiftRows* dilakukan pergeseran *bit* ke kiri.

4. Inverse MixColumns

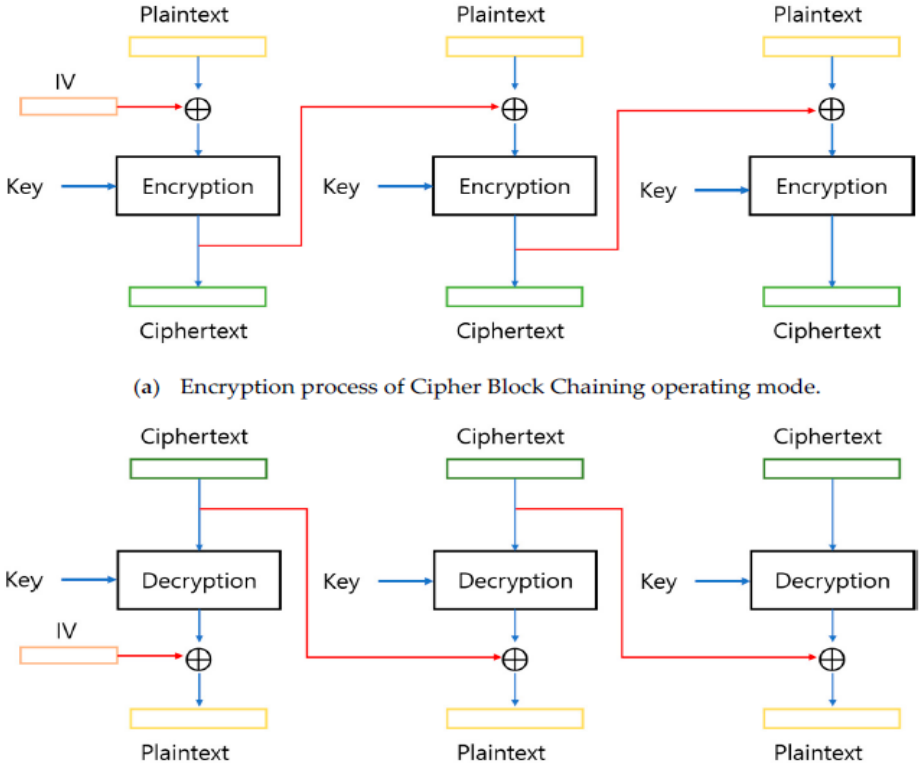
Untuk melakukan dekripsi pesan, dilakukan inverse dari transformasi MixColumns yakni mengalikan setiap kolom hasil dari Inverse AddRoundKey dengan matriks berikut:

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix}$$

2.1.3 *Advanced Encryption Standard – Cipher Block Chaining(AES-CBC)*

Teknologi enkripsi blok memiliki lima mode operasi: Mode *Electronic CodeBook (ECB)*, mode *Cipher Block Chaining (CBC)*, mode *Cipher FeedBack (CFB)*, mode *Output Feedback (OFB)*, dan mode *Counter (CTR)*. Mode *CBC* adalah yang paling aman metode enkripsi di antara mode operasi enkripsi blok dan merupakan yang paling sering digunakan. Penggunaan *Cipher Block Chaining (CBC)* pada sistem yang dikembangkan dapat menghasilkan sistem yang lebih baik. Dengan mode CBC, setiap blok *ciphertext* bergantung tidak hanya pada blok *plaintext*-nya tetapi juga pada

seluruh blok *plaintext* sebelumnya. Dekripsi dilakukan dengan memasukkan blok *ciphertext* yang *current* ke fungsi dekripsi, kemudian meng-XOR-kan hasilnya dengan blok *ciphertext* sebelumnya. Blok *ciphertext* sebelumnya berfungsi sebagai umpan-maju *(feedforward)* pada akhir proses dekripsi. blok *plaintext* pertama menggunakan *Initialization Vector* (IV) sebagai vektor awal. IV tidak perlu rahasia. Algoritma AES dengan CBC membutuhkan input yang memiliki ukuran tepat multiplikasi dari ukuran blok. Penggunaan *padding* dengan standarisasi tertentu tidak diperlukan. Blok dapat dipenuhi agar sesuai dengan spesifikasinya cukup dengan karakter kosong (spasi), selain itu mikrokontroler sendiri tidak menerapkan standarisasi tertentu untuk *padding*[10].



Gambar 1. *AES-CBC*

2.1.4 Firebase

Firebase merupakan model layanan yang bekerja di belakang layar dan menghubungkan aplikasi *mobile* ke *cloud storage*. *Firebase Realtime Database*

adalah database yang di-*host* di *cloud*. Data disimpan sebagai *JSON* dan disinkronkan secara *realtime* ke setiap klien yang terhubung. Ketika anda membuat aplikasi lintas-*platform* dengan *SDK Android*, *iOS*, dan *JavaScript*, semua *klien* akan berbagi sebuah *instance Realtime Database* dan menerima *update* data terbaru secara otomatis.

2.1.4.1 Firebase Realtime Databse

Firebase Realtime Database adalah database yang menyimpan data di cloud. Data yang disimpan di cloud dihasilkan sebagai JSON dan disinkronkan langsung dengan setiap produk aplikasi yang terhubung. Saat pengembang membangun aplikasi lintas platform menggunakan iOS, Android, dan JavaScript SDK, semua klien berbagi satu instance Realtime Database dan secara otomatis menerima pembaruan ke data terbaru[11][12]. Realtime Database adalah database NoSQL dengan optimasi dan fitur yang berbeda dari database relasional. Realtime Database API hanya dikembangkan untuk memungkinkan manipulasi data yang cepat. Hal ini memungkinkan pengembang untuk menciptakan pengalaman waktu nyata yang dapat melayani jutaan pengguna tanpa mengorbankan daya tanggap[13].

2.2 Sistem Monitoring Kualitas Air

Tujuan dari pengecekan kualitas air adalah untuk mengkontrol penyakit dan bakteri pada air kolam budidaya sehingga dapat dilakukan tindakan dengan segera jika kualitas air dalam keadaan buruk. Berdasarkan permasalahan tersebut diperlukan terobosan teknologi untuk mempermudah petani ikan dalam mengkontrol kualitas air kolam budidaya. Sistem monitoring kualitas air berbasis Internet of Things (IoT) merupakan solusi yang tepat untuk permasalahan kontrol kalitas air pada kolam budidaya. Prinsip kerja sistem tersebut adalah dengan cara mentransfer data dari beberapa sensor kualitas air (Ph, Total Dissolve Solid, Suhu) melalui embedded sistem pada sistem cloud computing yang kemudian data tersebut

di transfer ke web server dan smartphone android sehingga petani ikan dapat memonitoring kualitas air kolam menggunakan smartphone secara realtime dan terintegrasi dengan sistem notifikasi[14].