SKRIPSI

IMPLEMENTASI TEKNOLOGI BLOKCHAIN UNTUK PENGIRIMAN JAWABAN CBT MENGGUNAKAN RSA-SHA256

Disusun dan diajukan oleh

KENNEDY

H071171516

PROGRAM STUDI SISTEM INFORMASI
DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR

2021

IMPLEMENTASI TEKNOLOGI BLOKCHAIN UNTUK PENGIRIMAN JAWABAN CBT MENGGUNAKAN RSA-SHA256

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi Departemen Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin

KENNEDY

H071171516

PROGRAM STUDI SISTEM INFORMASI DEPARTEMEN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HASANUDDIN

MAKASSAR

2021

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama	: Kennedy
NIM	: H071171516
Program Studi	: Sistem Informasi
Jenjang	: S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

Implementasi Teknologi Blockchain untuk Pengiriman Jawaban CBT menggunakan RSA-SHA256

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan belum pernah dipublikasikan dalam bentuk apapun.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut..

Makassar, 1 Desember 2021
Yang menyatakan,

Kennedy
NIM. H071171516

IMPLEMENTASI TEKNOLOGI BLOCKCHAIN UNTUK PENGIRIMAN JAWABAN CBT MENGGUNAKAN RSA-SHA256

Disusun dan diajukan oleh

KENNEDY

H071171516

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin pada tanggal 1 Desember 2021 dan dinyatakan telah memenuhi syarat kelulusan.

Menyetujui,

Pembimbing Utama	**Pembimbing Pertama**
Dr. Armin Lawi, S.Si, M.Eng	**Andi Muhammad Anwar, S.Si., M.Si.**
NIP. 197204231995121001	NIP. 199012282018031001

Ketua Program Studi,

Dr. Muhammad Hasbi, M.Sc.
NIP. 196307201989031003

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh:

Nama	:	Kennedy
NIM	:	H071171516
Program Studi	:	Sistem Informasi
Judul Skripsi	:	Implementasi Teknologi Blockchain untuk Pengiriman Jawaban CBT Menggunakan RSA-SHA256

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

DEWAN PENGUJI

		Tanda tangan
Ketua	: Dr. Armin Lawi, S.Si, M.Eng	(.....)
Sekretaris	: Andi Muhammad Anwar, S.Si., M.Si	(.....)
Anggota	: Dr. Muhammad Hasbi, M.Sc	(.....)
Anggota	: Rozalina Amran, S.T., M.Eng.	(.....)

Ditetapkan di : Makassar
Tanggal : 1 Desember 2021

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN UNIVERSITAS HASANUDDIN DEPARTEMEN MATEMATIKA FAKULTAS MIPA

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas berkat dan rahmat-Nya penulis dapat menyelesaikan pendidikan jenjang Strata 1 pada Program Studi SistemInformasi, Universitas Hasanuddin, dengan judul skripsi "Implementasi Teknologi Blokchain untuk Pengiriman Jawaban CBT Menggunakan RSA-SHA256" . Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Komputer. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini. Oleh karena itu, pada kesempatan ini dengan segala kerendahan hati penulis menyampaikan terimakasih yang setulus-tulusnya kepada :

1. Rektor Universitas Hasanuddin, Ibu **Prof. Dr. Dwia Aries Tina Pulubuhu, M.A.** beserta jajarannya; Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, Bapak **Dr. Eng. Amiruddin, S.Si.** beserta jajarannya; Ketua Departemen Matematika, Bapak **Prof. Dr. Nurdin, S.Si., M.Si**. beserta jajarannya; Ketua Program Studi Sistem Informasi, Bapak **Dr. Muhammad Hasbi, M.Sc**. beserta jajarannya; serta Bapak/Ibu dosen Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin atas ilmu-ilmu dan bantuannya dalam berbagai bentuk.
2. Bapak **Dr. Armin Lawi, S.Si, M.Eng,** sebagai pembimbing utama dan Bapak **Andi Muhammad Anwar, S.Si., M.Si,** sebagai pembimbing pertama, yang sementara menyusun disertas, atas ketersediaan waktunya yang telah diluangkan untuk membimbing penulis selama proses penyusunan tugas akhir; Bapak **Dr. Muhammad Hasbi, M.Sc** dan Ibu **Rozalina Amran, S.T., M.Eng.** , atas waktu dan kesediaannya sebagai penguji untuk tugas akhir penulis, serta ilmu yang dibagikan selama mengerjakan selama mengerjakan tugas akhir.
3. Orang Tua dan Keluarga yang telah memberikan dukungan dan moral, khususya untuk ayah, Bapak **Rudyanto Sulaiman, S.H.** dan Ibu **Ineke**

Kusumawati Tan, yang selalu memberi masukan dan saran dalam mengerjakan tugas akhir hingga penulis dapat menyelesaikan dengan baik.

4. Sahabat **Muhammad Muflihun Naim, Siti Rabiatul Adawiyah, Muhammad Fitrah, Khawaritzmi Abdallah Ahmad,** yang berperan banyak dalam pengerjaan dan penyelesaian tugas akhir dari awal hingga akhir, juga sebagai teman seperjuangan dari awal perkuliahan hingga saat ini.
5. **Dio Athagashi Rudy** sebagai teman curhat dari awal pengerjaan tugas akhir hingga awal hingga akhir. Teman-teman **Sistem Informasi** angkatan **2017** atas kebersamaan dan segala suka dan duka selama menjadi mahasiswa Universitas Hasanuddin.
6. Seluruh pihak yang tidak dapat disebutkan satu per satu atas segala bentuk kontribusi, partisipasi, serta motivasi yang diberikan selama ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Makassar, 1 Desember 2021

Penulis

PERYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Hasanuddin, saya yang bertanda tangan di bawah ini:

Nama	: Kennedy
NIM	: H071171516
Program Studi	: Sistem Informasi
Departemen	: Matematika
Fakultas	: Matematika dan Ilmu Pengetahuan Alam
Jenis Karya	: Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul:

Implementasi Teknologi Blockchain untuk Pengiriman Jawaban CBT menggunakan RSA-SHA256

berserta perangkat yang ada (jika diperlukan). Terkait dengan hal di atas, maka pihak universitas berhak menyimpan,mengalih-media/ format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat,dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar pada tanggal 1 Desember 2021

Yang menyatakan

(Kennedy)

ABSTRAK

Teknologi pada masa ini sangat cepat berkembang seiring dengan berjalannya waktu. Terdapat banyak manfaat dari teknologi saat ini yang membantu pekerjaan manusia dalam berbagai aspek, baik aspek ekonomi, aspek sosial, khususnya aspek pendidikan, contohnya ujian berbasis komputer yang telah diterapkan di Indonesia pada tahun 2015. Dengan adanya ujian berbasis *online*, memungkinkan pihak bertanggung jawab untuk mencuri/ mengganti jawaban peserta ujian. Penelitian ini mengimplementasikan teknologi *blokchain* untuk pengiriman jawaban CBT sehingga pada saat pengiriman jawaban di server, riwayat pengiriman dapat disimpan ke dalam sebuah blok, sehingga jika terjadi penggantian data/ penghapusan data oleh pihak yang tidak berwajib, maka dapat dideteksi dan diselesaikan dengan cepat. Penelitian ini menggunakan Algoritma RSA dalam pengiriman jawaban CBT, dan Algoritma SHA256 untuk pengamanan blok dalam *blokchain*, dan *merkle tree* untuk *hashing* jawaban peserta ujian. Performa teknologi *blokchain* berpengaruh terhadap *difficulty,* sehingga semakin besar nilai *difficulty* maka semakin besar pula memori dan durasi yang dibutuhkan untuk proses *hashing*. Semakin besar pula data dalam blok dan banyak blok dalam *blokchain*, maka memori dan durasi yang dibutuhkan semakin lama.

Kata kunci : Ujian Berbasis Komputer, *Blokchain, merkle tree, difficulty*, SHA-256, RSA

ABSTRACT

Technology at this time is very fast developing along with the passage of time. There are many benefits from today's technology that helps human work in various aspects, both economic aspects, social aspects, especially educational aspects, for example computer-based exams that have been implemented in Indonesia in 2015. With online-based exams, it allows responsible parties to stealing/replacing the examinee's answers. This research implements blokchain technology for sending CBT answers so that when sending answers on the server, the sending history can be stored in a blok, so that in the event of data replacement/deletion of data by unauthorized parties, it can be detected and resolved quickly. This study uses the RSA Algorithm for sending CBT answers, and the SHA256 Algorithm for securing bloks in the blokchain, and the merkle tree for hashing the examinees' answers. The performance of blokchain technology affects difficulty, so the greater the difficulty value, the greater the memory and duration required for the hashing process. The larger the data in bloks and the more bloks in the blokchain, the longer the memory and duration required.

Keywords : *computer-based exams, Blokchain, Merkle Tree, difficulty*, SHA-256, RSA

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1. Latar Belakang

Teknologi pada masa ini sangat cepat berkembang seiring dengan berjalannya waktu. Terdapat banyak manfaat dari teknologi saat ini yang banyak membantu pekerjaan manusia dalam berbagai aspek, baik aspek ekonomi, aspek sosial, khususnya aspek pendidikan. Dengan memanfaatkan teknologi, pendidikan sangat mudah untuk didapatkan, mulai dari pembelajaran berbasis internet, materi pembelajaran, video pembelajaran, dan ujian yang dilaksanakan secara online berbasis komputer (CBT).

Computer Based Test (CBT) sudah mulai diterapkan di Indonesia sejak Ujian Nasional Sekolah Menengah atau Tes Masuk Perguruan Tinggi sejak 2015, tetapi dampak dari pandemik virus Covid-19, mengakibatkan CBT banyak diterapkan bahkan dalam proses belajar mengajar secara online. Secara umum, jenis soal yang terdapat pada CBT adalah pilihan ganda. Ujian Berbasis Komputer yang terhubung dengan jaringan internet memiliki kelemahan yaitu bahaya manipulasi data yang terjadi dalam proses pengiriman jawaban dari peserta ke admin dan server pusat. Penyadap dapat mengganti jawaban peserta ataupun membagikan jawaban peserta tersebut ke peserta lain yang masih bekerja.

Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikan ke dalam bentuk yang tidak dapat dipahami lagi maknanya (Meyer, 1982). Kriptografi bertujuan untuk memberikan keamanan berupa kerahasiaan, integritas data, otentikasi dan anti penyangkalan (Schneier, 1996). Dengan kriptografi, jawaban dari peserta akan diacak bentuknya sehingga ketika penyadap ingin mencuri jawaban peserta, penyadap tidak memiliki kunci untuk dekripsi jawaban menjadi bentuk aslinya.

Algoritma RSA merupakan salah satu algoritma kriptografi kunci publik yang digunakan untuk mengenkripsi sebuah pesan. Algoritma ini dibuat oleh 3 orang peneliti dari MIT (Massachussets Institute of Technology) pada tahun 1976.

Algoritma RSA dapat digunakan untuk enkripsi dan dekripsi berbagai jenis data, misalnya teks, gambar, suara dan video. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima (Rivest, dkk., 1977).

Jika penyadap ingin mengubah atau menambahkan data pada pesan yang telah dienkripsi, salah satu fungsi dari kriptografi adalah integritas data yang berfungsi untuk menjamin data keaslian suatu pesan selama pengiriman informasi tersebut melalui media tertentu (Schneier, 1996). Integritas data yang dimaksud adalah menggunakan fungsi hash satu arah. Hasil dari fungsi hash memiliki ukuran yang tetap, tidak berpengaruh terhadap panjang pesan aslinya.

Fungsi hash memiliki kelemahan pada bagaimana mendeteksi perubahan pesan sekaligus perubahan pada nilai hash. Apabila penyadap memiliki akses terhadap file atau dokumen nilai hash dari pesan, maka penyadap dapat mengubah nilai pesan kemudian menghitung ulang nilai hash pesan yang sudah diubah tersebut dan mengubahnya pada file nilai hash semula. Oleh karena itu, digunakan teknologi *blokchain* untuk mengatasi perubahan pesan /hash.

Blokchain, teknologi basis data transaksional, adalah cara terdesentralisasi untuk mengelola validasi dan transaksi yang tahan gangguan dengan konsistensi di sejumlah besar peserta, juga dikenal sebagai node (O. Ali, A. Jaradat, A. Kulakli and A. Abuhalimeh, 2021). Dengan menggunakan blokchain kita dapat menjaga informasi yang diarsipkan agar tidak rusak atau diambil orang lain.

Penelitian “BLOSOM : Blokchain technology for Security of Medical Records” oleh R.Johari, V. Kumar, dan K.Gupta, 2021, membahas bagaimana penerapan *blokchain* untuk keamanan rekaman medis, sehingga rekaman medis tersebut tidak dapat dihapus ataupun diubah.

Penelitian “Pengamanan jawaban ujian *Computer Based Test* (CBT) dengan menggunakan algoritma RSA dan fungsi HMAC berbasis algoritma SHA-1” oleh Deo Valiandro (2020), membahas bagaimana mengamankan jawaban CBT yang akan dikirim, sehingga jawaban tersebut tetap aman dan tidak diubah pada saat pengiriman berlangsung.

Penelitian ini mengimplementasikan teknologi blokchain pada sistem ujian berbasis komputer dengan menggunakan fungsi Hash SHA-256 dan Enkripsi menggunakan algoritma RSA untuk tetap menjaga jawaban yang telah dimasukkan sah dan tidak rusak oleh campur tangan pihak ketiga.

1.2. Rumusan Masalah

Rumusan masalah berdasarkan latar belakang yang telah dijelaskan diatas adalah :

1) Bagaimana cara implementasi *Merkle Tree* dengan Algoritma SHA-256 pada pesan?
2) Bagaimana cara implementasi Enkripsi dan Dekripsi dengan Algoritma RSA?
3) Bagaimana cara implementasi Blokchain dalam pengiriman pesan?
4) Bagaimana simulasi aplikasi pada pengiriman pesan?
5) Bagaimana performa keseluruhan metode terhadap banyak blok dan pesan?

1.3. Batasan Masalah

Adapun batasan masalah yang diteliti sehingga jangkauan penelitian tidak akan melewati permasalahan sebenarnya, yaitu :

1) Para admin diberi tanggung jawab untuk mengelola blokchain.
2) Sistem ujian yang disimulasikan adalah pilihan ganda (5 pilihan) sebanyak 100 soal.
3) Sistem ujian yang disimulasikan adalah sistem semi-online. Sistem semi-online dipilih karena beragam lokasi peserta ujian, maka internet yang digunakan tidak selamanya bagus.

1.4. Tujuan Penelitian

Berdasarkan rumusan masalah dan latar belakang yang telah dijelaskan, penelitian ini bertujuan untuk :

1) Mampu mengimplementasikan *Merkle Tree* dengan Algoritma SHA-256 pada pesan.
2) Mampu mengimplementasikan Enkripsi dan Dekripsi dengan Algoritma RSA.
3) Mampu mengimplementasikan Blokchain dalam pengiriman pesan.

4) Mampu melakukan simulasi aplikasi pada pengiriman pesan.
5) Mengetahui performa keseluruhan metode terhadap banyak blok dan pesan.

1.5. Manfaat Penelitian

Penelitian ini diharapkan dapat dimanfaatkan dalam pengamanan hasil ujian berbasis komputer, sehingga hasil jawaban tidak mengalami perubahan dan keaslian jawaban tetap terjaga. Penelitian ini juga diharapkan dapat dikembangkan dan diimplementasikan ke berbagai aspek lainnya.

BAB II
TINJAUAN PUSTAKA

2.1. Ujian

Ujian adalah cara untuk mengukur kemampuan seseorang. Pelaksanaan ujian bertujuan untuk mengukur pengetahuan seseorang atau dalam hal ini peserta didik, tujuan ini misalnya pada pelaksanaan ujian nasional di jenjang pendidikan SD, SMP/MTs, SMPLB, SMA/MA/SMAK/SMTK, SMALB, maupun di SMK/MAK dan juga sebagai salah satu tolak ukur pencapaian pembelajaran dalam rangka penjaminan dan peningkatan mutu pendidikan (Badan Standar Nasional Pendidikan (BSNP) Dan Badan Penelitian Dan Pengembangan, Kementerian Pendidikan Dan Kebudayaan (Balitbang Kemdikbud), 2018). Selain itu, pelaksanaan ujian juga digunakan untuk menyeleksi para calon untuk memasuki suatu institut tertentu, hal ini terlihat pada ujian masuk perguruan tinggi yang dilaksanakan baik oleh perguruan tinggi negeri maupun oleh perguruan tinggi swasta (Menteri Pendidikan dan Kebudayaan Republik Indonesia, 2018).

2.2. Ujian Berbasis Komputer (CBT)

Ujian Nasional Berbasis Komputer (UNBK) disebut juga *Computer Based Test* (CBT) adalah sistem pelaksanaan ujian nasional dengan menggunakan komputer sebagai media ujiannya. Dalam pelaksanaannya, UNBK berbeda dengan sistem ujian nasional berbasis kertas atau *Paper Based Test* (PBT) yang selama ini sudah berjalan. (Kementrian Pendidikan dan Kebudayaan, 2020). Namun, ujian CBT sekarang ini telah diterapkan juga dalam tes masuk perguruan tinggi (SBMPTN), dan juga ujian dalam perguruan tinggi ataupun ujian harian sekolah, khususnya dalam masa pandemik Covid-19 yang mengharuskan segala aktivitas dilakukan di rumah masing-masing.

Ujian CBT bekerja dengan cara menggunakan klien dan server. Ujian CBT bisa menggunakan 3 jenis sistem yaitu (Kementrian Pendidikan dan Kebudayaan, 2020):

1. Sistem *online* atau terhubung langsung dengan server pusat secara *realtime*, soal langsung dari server pusat dan hasil pengerjaan soal tersebut juga langsung masuk ke server pusat,
2. Sistem *semi-online* yaitu soal dikirim dari server pusat secara online melalui jaringan (sinkronisasi) ke server lokal (sekolah/pusat ujian), kemudian ujian siswa dilayani oleh server lokal (sekolah/pusat ujian) secara *offline*. Selanjutnya hasil ujian dikirim kembali dari server lokal (sekolah/pusat ujian) ke server pusat secara online (melalui proses upload).
3. Sistem *offline*, yaitu ujian dilayani oleh server lokal, soal dan hasil pengerjaan soal dikirim dengan menggunakan media penyimpanan.

Penyelenggaraan UNBK saat ini menggunakan sistem semi-online yaitu soal dikirim dari server pusat secara online melalui jaringan (sinkronisasi) ke server lokal (sekolah), kemudian ujian siswa dilayani oleh server lokal (sekolah) secara offline. Selanjutnya hasil ujian dikirim kembali dari server lokal (sekolah) ke server pusat secara online (*upload*).

2.3. Kriptografi

Kriptografi adalah ilmu menjaga rahasia. Asumsikan Alice ingin mengirimkan pesan kepada Bob melalui jaringan terbuka, seperti telepon umum atau jaringan komputer. Pesan Alice kemungkinan besar dapat diubah atau diambil oleh Charlie tanpa diketahui oleh Alice dan Bob. Salah satu tujuan dari Kriptografi adalah untuk mencegah hal tersebut terjadi.

Kriptografi memiliki tujuan memberikan keamanan yang terbagi menjadi beberapa aspek sebagai berikut (Schneier, 1996):

1. Kerahasiaan (*confidentiality*), yaitu menjaga keamanan informasi sehingga tidak dapat diketahui oleh siapa pun kecuali pihak yang memiliki otoritas untuk mengetahui pesan tersebut. Di dalam kriptografi, aspek ini direalisasikan dengan enkripsi dan dekripsi informasi yang bersifat rahasia.
2. Integritas data (*data integrity*), yaitu penjaminan informasi masih asli atau tidak diubah oleh pihak lain selama pengiriman informasi tersebut. Di dalam

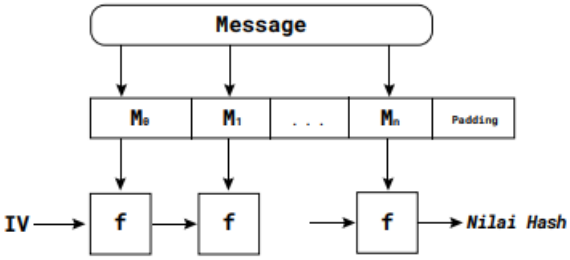
kriptografi, aspek ini direalisasikan dengan menggunakan fungsi hash dan tanda tangan digital (*digital signature*).

3. Otentikasi (*authentication*), yaitu identifikasi kebenaran pihak-pihak yang saling berkomunikasi mempertukarkan informasi bahwa pesan yang diterima atau dikirim benar dari pengirim atau penerima yang sesungguhnya. Di dalam kriptografi, aspek ini direalisasikan dengan menggunakan tanda tangan digital (*digital signature*).
4. Anti-penyangkalan (*non-repudiation*), yaitu pencegahan pengirim informasi menyangkal bahwa pengirim yang telah mengirim informasi. Di dalam kriptografi, aspek ini direalisasikan dengan menggunakan tanda tangan digital (*digital signature*).

2.4. SHA

SHA merupakan keluarga fungsi hash satu-arah dengan enam varian dengan perbedaan pada parameter yang digunakan.. Variasi pertama yang dipublikasikan adalah SHA-0 pada 1993 dan sudah ditemukan kolisinya. SHA-1 dipublikasikan pada tahun 1995. Empat variasi lain kemudian dipublikasikan yang dikenal dengan SHA-2 yaitu SHA-224, SHA-256, SHA-384 dan SHA-512. NIST kemudian mengumumkan fungsi hash Keccak sebagai SHA-3 pada tahun 2012 (Paar & Pelzl, 2010).

Algoritma SHA bekerja dengan membagi naskah atau pesan menjadi beberapa blok, setiap blok biasanya 512 atau 1024 bit. Naskah atau pesan diberi padding agar panjang pesan merupakan kelipatan dari besarnya blok dan *padding* diberi akhiran berupa panjang dari pesan. Algoritma biasanya terdiri dari dua tahap, yaitu: 1) *preprocessing* data, yaitu terdiri dari padding dan *parameter setup*, 2) *hashing*, yaitu tahap membuat *message digest* dengan mengompresi data. Kompresi dilakukan dengan berurutan tiap blok dan hasil blok sebelumnya dijadikan *feedback* untuk blok berikutnya. Cara kerja ini bekerja secara iteratif. Konstruksi seperti ini dinamakan konstruksi Merkle– Damgård (Merkle, 1979).



Gambar 2.1. Konstruksi Merkle-Damgard

2.5. SHA-256

SHA-256 merupakan salah satu variasi dari SHA-2 yang merupakan penerus dari SHA-1. SHA-256 menerima maksimum 2^{64} bit dan menghasilkan *message digest* sebesar 256 bit. SHA-256 memproses pesan yang diinput blok tiap blok, dimana pada setiap aplikasi fungsi f_k(m) adalah fungsi dari 64 pengulangan pada tiap langkah. (Nigen, 2016).

Cara kerja fungsi menggunakan fungsi f dan g yang berbeda dengan yang digunakan dalam MD-4 dan SHA-1. Fungsi f dan g yang digunakan SHA-2 seperti berikut :

$$f'(u,v,w) = (u \wedge v) \oplus ((\neg u) \wedge w)$$

$$g'(u,v,w) = (u \wedge v) \oplus (u \wedge w) \oplus (v \wedge w)$$

SHA-2 juga menggunakan fungsi berikut dalam 32-bit

$$\sum\nolimits_0 (x) = (x \ggg 2) \oplus (x \ggg 13) \oplus (x \ggg 22)$$

$$\sum\nolimits_1 (x) = (x \ggg 6) \oplus (x \ggg 11) \oplus (x \ggg 25)$$

$$\sigma_0(x) = (x \ggg 7) \oplus (x \ggg 18) \oplus (x \gg 3)$$

$$\sigma_1(x) = (x \ggg 17) \oplus (x \ggg 19) \oplus (x \gg 10)$$

Dengan $\ggg$ melambangkan rotasi kanan, dan $\gg$ melambangkan shift kanan. Terdapat 64 konstanta yang mewakli 32 bit pertama dari bagian pecahan akar pangkat tiga dari 64 bilangan prima pertama. Untuk SHA-256, keadaan internal

dalam algoritma merupakan delapan pasang nilai 32-bit, yaitu H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8.

Tahap *preprocessing* pada SHA-256 sama dengan algoritma SHA-2 yang lainnya, hanya saja yang membedakannya ialah pada tahap kompresi fungsi. Fungsi kompresi SHA-256 seperti berikut. (Nigen, 2016)

Algoritma SHA-256 | Fungsi Kompresi

$A, B, C, D, E, F, G, H \leftarrow H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8$.

/* EXPANSION */

For i = 16 to 63 do

$w_i \leftarrow \sigma_1(w_{i-2}) + \sigma_0(w_{i-15}) + w_{i-7} + w_{i-16}$

/* ROUND /*

For i = 0 to 63 do

$t_1 \leftarrow H + \sum_1(E) + f'(E, F, G) + K_i + X_i$

$t_2 \leftarrow \sum_0(A) + g'(A, B, C)$

$(A, B, C, D, E, F, G, H) \leftarrow (t_1 + t_2, A, B, C, D + t_1, E, F, G)$

/* ***MESSAGE DIGEST /**

$(H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8) \leftarrow s_r =$

$(H_1 + A, H_2 + B, H_3 + C, H_4 + D, H_5 + E, H_6 + F, H_7 + G, H_8 + H)$

Delapan pasang nilai 32-bit sebagai berikut :

$H_1 \leftarrow 0x6A09E667$ $H_2 \leftarrow 0xBB67AE85$

$H_3 \leftarrow 0x3C6EF372$ $H_4 \leftarrow 0xA54FF53A$

$H_5 \leftarrow 0x510E527F$ $H_6 \leftarrow 0x9B05688C$

$H_7 \leftarrow 0x1F83D9AB$ $H_8 \leftarrow 0x5BE0CD19$

Langkah-langkah pembentukan *message digest* dari SHA-256 terdiri dari tahapan *preprocessing* dan Kompresi Fungsi, contoh pesan “hello world” :

1) Pada Tahap *preprocessing* , konversi pesan “hello world” menjadi biner 01101000 01100101 01101100 01101100 01101111 00100000 01110111 01101111 01110010 01101100 01100100
2) Setelah itu tambahkan 1 di akhir pesan yang telah diubah menjadi biner. 01101000 01100101 01101100 01101100 01101111 00100000 01110111 01101111 01110010 01101100 01100100 1
3) Langkah berikutnya tambahkan nilai k yang bernilai 0 hingga panjang pesan l menjadi $l + 1 + k = 448 \, mod \, 512$, maka kita perlu menambahkan 0 sebanyak $448 - 88 - 1 = 359$.
   01101000 01100101 01101100 01101100 01101111 00100000 01110111 01101111 01110010 01101100 01100100 10000000 00000000
4) Lalu 64-bits data terakhir diisi dengan panjang pesan yang diinput, yaitu 88 diubah menjadi biner menjadi 01011000, dan akan menjadi seperti : 01101000 01100101 01101100 01101100 01101111 00100000 01110111 01101111 01110010 01101100 01100100 10000000 01011000

5) Setelah itu pesan yang telah diperoleh dikelompokkan menjadi 16 bagian $\frac{512}{16} = 32$ bit per kelompok yang kita sebut dengan *words*. Karena nilai w[3] sampai w[14] sama, dapat kita singkat.

$$w[0] = 01101000011001010110110001101100$$

$$w[1] = 01101111001000000111011101101111$$

$$w[2] = 01110010011011000110010010000000$$

$$w[3] = 00000000000000000000000000000000$$

….

$$w[14] = 00000000000000000000000000000000$$

$$w[15] = 00000000000000000000000001011000$$

6) Setelah itu kita memakai fungsi kompresi untuk mendapatkan nilai w[16] sampai w[63]. Berikut kita mencari nilai dari w_{16}.

$$w_{16} = \sigma_1(w_{16-2}) + \sigma_0(w_{16-15}) + w_{16-7} + w_{16-16}$$

$$w_{16} = \sigma_1(w_{14}) + \sigma_0(w_1) + w_9 + w_0$$

$$\sigma_0(w_1) = (w_1 \ggg 7) \oplus (w_1 \ggg 18) \oplus (w_1 \gg 3)$$

$$\sigma_0(w_1) = 11011110110111100100000011101110$$
$$\oplus\ 00011101110110111101101111001000$$
$$\oplus\ 00001101111001000000111011101101$$

$$\sigma_0(w_1) = 11001110111000011001010111001011$$

$$\sigma_1(w_{14}) = (w_{14} \ggg 17) \oplus (w_{14} \ggg 19) \oplus (w_{14} \gg 10)$$

$$\sigma_1(w_{14}) = 00000000000000000000000000000000$$
$$\oplus\ 00000000000000000000000000000000$$
$$\oplus\ 00000000000000000000000000000000$$

$$\sigma_1(w_{14}) = 00000000000000000000000000000000$$

$$w_{16} = 00000000000000000000000000000000$$
$$+\ 11001110111000011001010111001011$$
$$+\ 00000000000000000000000000000000$$
$$+\ 01101000011001010110110001101100$$
$$\boldsymbol{w_{16} = 00110111010001110000001000110111}$$

7) Dengan cara yang sama, dilakukan perulangan untuk mencari nilai dari w[17] sampaiw[63], dengan hasil seperti berikut ini :

$$w[17] = 10000110110100001100000000110001$$
$$w[18] = 11010011101111010001000100001011$$
$$w[19] = 01111000001111110100011110000010$$
$$w[20] = 00101010100100000111110011101101$$
$$w[21] = 01001011001011110111110011001001$$
$$w[22] = 00110001111000011001010001011101$$
$$w[23] = 10001001001101100100100101100100$$
$$w[24] = 01111111011110100000011011011010$$
$$w[25] = 11000001011110011010100100111010$$
$$w[26] = 10111011111010001111011001010101$$
$$w[27] = 00001100000110101110001111100110$$
$$w[28] = 10110000111111100000110101111101$$
$$w[29] = 01011111011011100101010110010011$$
$$w[30] = 00000000100010011001101101010010$$
$$w[31] = 00000111111100011100101010010100$$
$$w[32] = 00111011010111111110010111010110$$
$$w[33] = 01101000011001010110001011100110$$
$$w[34] = 11001000010011100000101010011110$$
$$w[35] = 00000110101011111001101100100101$$
$$w[36] = 10010010111011110110010011010111$$
$$w[37] = 01100011111110010101111001011010$$
$$w[38] = 11100011000101100110011111010111$$
$$w[39] = 10000100001110111101111000010110$$
$$w[40] = 11101110111011001010100001011011$$

$$w[41] = 10100000010011111111001000100001$$
$$w[42] = 11111001000110001010110110111000$$
$$w[43] = 00010100101010001001001000011001$$
$$w[44] = 00010000100001000101001100011101$$
$$w[45] = 01100000100100111110000011001101$$
$$w[46] = 10000011000000110101111111101001$$
$$w[47] = 11010101101011100111100100111000$$
$$w[48] = 00111001001111110000010110101101$$
$$w[49] = 11111011010010110001101111101111$$
$$w[50] = 11101011011101011111111100101001$$
$$w[51] = 01101010001101101001010100110100$$
$$w[52] = 00100010111111001001110011011000$$
$$w[53] = 10101001011101000000110100101011$$
$$w[54] = 01100000110011110011100010000101$$
$$w[55] = 11000100101011001001100000111010$$
$$w[56] = 00010001010000101111110110101101$$
$$w[57] = 10110000101100000001110111011001$$
$$w[58] = 10011000111100001100001101101111$$
$$w[59] = 01110010000101111011100000011110$$
$$w[60] = 10100010110101000110011110011010$$
$$w[61] = 00000001000011111001100101111011$$
$$w[62] = 11111100000101110100111100001010$$
$$w[63] = 11000010110000101110101100010110$$

8) Setelah itu kita masuk ke fungsi kompresi, dimana delapan nilai hash kita ubah kedalam biner terlebih dahulu.

A = 0x6a09e667 = 01101010000010011110011001100111

B = 0xbb67ae85 = 10111011011001111010111010000101

C = 0x3c6ef372 = 00111100011011011111001101110010

D = 0xa54ff53a = 10100101010011111111010100111010

E = 0x510e527f = 01010001000011100101001001111111

F = 0x9b05688c = 10011011000001010110100010001100

G = 0x1f83d9ab = 00011111100000111101100110101011

H = 0x5be0cd19 = 01011011111000001100110100011001

9) Nilai biner tersebut akan kita gunakan dalam formula kompresi seperti dibawah ini dengan perulangan sebanyak 64 kali. *K* merupakan nilai hex yang didapatkan dari sebagian nilai akar pangkat tiga dari 64 bilangan prima pertama (2, 3, 5, 7, … , 311).

$$t_1 \leftarrow H + \sum_1(E) + f'(E,F,G) + K_i + X_i$$

$$t_2 \leftarrow \sum_0(A) + g'(A,B,C)$$

$$(A,B,C,D,E,F,G,H) \leftarrow (t_1 + t_2, A, B, C, D + t_1, E, F, G)$$

Kita ambil contoh i = 0, maka :

$$t_1 = H + \sum_1(E) + f'(E,F,G) + K_0 + X_0$$

$$\sum_1(E) = (E \ggg 6) \oplus (E \ggg 11) \oplus (E \ggg 25)$$

$$\sum_1(E) = 11111101010001000011100101001001$$
$$\oplus\ 01001111111010100010000111001010$$
$$\oplus\ 10000111001010010011111110101000$$

$$\sum_1(E) = \ 00110101100001110010011100101011$$

$$f'(E,F,G) = (E \wedge F) \oplus \big((\neg E) \wedge G\big)$$

$$E \wedge F = \ 01010001000011100101001001111111$$
$$\wedge\ 10011011000001010110100010001100$$

$$E \wedge F = \ 00010001000001000100000000001100$$

$$(\neg E) \wedge G = 10101110111100011010110110000000$$
$$\wedge\ 00011111100000111101100110101011$$

$$(\neg E) \wedge G = 00001110100000011000100110000000$$

$$f'(E,F,G) = 00010001000001000100000000001100 \oplus$$
$$00001110100000011000100110000000$$

$$f'(E,F,G) = 00011111100001011100100110001100$$

$$t_1 = 01011011111000001100110100011001$$
$$+\ 00110101100001110010011100101011$$
$$+ 00011111100001011100100110001100$$
$$+ 01000010100010100010111110011000$$
$$+\ 01101000011001010110110001101100$$
$$t_1 = 01011011110111010101100111010100$$

$$t_2 = \sum_0 (A) + g'(A,B,C)$$

$$g'(A,B,C) = (A\ \wedge B) \oplus (A \wedge C) \oplus (B \wedge C)$$

$$A\ \wedge B = 01101010000010011110011001100111$$
$$\wedge\ 10111011011001111010111010000101$$

$$A\ \wedge B = 00101010000000011010011000000101$$

$$A \wedge C = 01101010000010011110011001100111\ \wedge$$
$$00111100011011011111001101110010$$

$$A \wedge C = 00101000000010001110001001100010$$

$$B \wedge C = 10111011011001111010111010000101$$
$$\wedge\ 00111100011011011111001101110010$$

$$B \wedge C = 00111000011001101010001000000000$$

$$g'(A,B,C) = 00101010000000011010011000000101$$
$$\oplus 00101000000010001110001001100010$$
$$\oplus 00111000011001101010001000000000$$

$$g'(A,B,C) = 00111010011011111110011001100111$$

$$\sum_0(A) = (A \ggg 2) \oplus (A \ggg 13) \oplus (A \ggg 22)$$

$$\sum_0(A) = 11011010100000100111100110011001$$
$$\oplus\ 00110011001110110101000001001111$$
$$\oplus\ 00100111100110011001110110101000$$

$$\sum_0(A) = 11001110001000001011010001111110$$

$$t_2 = \ 11001110001000001011010001111110$$
$$+\ 00111010011011111110011001100111$$

$$t_2 = \ 00001000100100001001101011100101$$

$$(A,B,C,D,E,F,G,H) \leftarrow (t_1 + t_2, A, B, C, D + t_1, E, F, G)$$

Lalu langkah berikutnya kita mengganti nilai A – H, menjadi seperti formula diatas.

$$H = G = 00011111100000111101100110101011$$
$$G = F = 10011011000001010110100010001100$$
$$F = E = 01010001000011100101001001111111$$
$$E = D + t_1 = 10100101010011111111010100111010$$
$$+\ 01011011110111010101100111010100$$
$$E = 00000001001011010100111100001110$$
$$D = C = 00111100011011011011001101110010$$
$$C = B = 10111011011001111010111010000101$$

$$A = t_1 + t_2 = 01011011110111010101100111010100$$
$$+\ 00001000100100001001101011100101$$

$$A = \ 01100100011011011111010010111001$$

10) Setelah itu, lakukan perulangan sampai iterasi i = 63, nilai H_1 – H_8 dijumlahkan dengan A – H :

H_1 = 6A09E667 = 01101010000010011110011001100111

H_2 = BB67AE85 = 10111011011001111010111010000101

H_3 = 3C6EF372 = 00111100011011101111001101110010

H_4 = A54FF53A = 10100101010011111111010100111010

H_5 = 510E527F = 01010001000011100101001001111111

H_6 = 9B05688C = 10011011000001010110100010001100

H_7 = 1F83D9AB = 00011111100000111101100110101011

H_8 = 5BE0CD19 = 01011011111000001100110100011001

A = 4F434152 = 01001111010000110100000101010010

B = D7E58F83 = 11010111111001011000111110000011

C = 68BF5F65 = 01101000101111110101111101100101

D = 352DB6C0 = 00110101001011011011011011000000

E = 73769D64 = 01110011011101101001110101100100

F = DF4E1862 = 11011111010011100001100001100010

G = 71051E01 = 01110001000001010001111000000001

H = 870F00D0 = 10000111000011110000000011010000

$H_1 = H_1 + A$ = 10111001010011010010011110111001

$H_2 = H_2 + B$ = 10010011010011010011111000001000

$H_3 = H_3 + C$ = 10100101001011100101001011010111

$H_4 = H_4 + D$ = 11011010011111011010101111111010

$H_5 = H_5 + E$ = 11000100100001001110111111100011

$H_6 = H_6 + F$ = 01111010010100111000000011101110

$H_7 = H_7 + G$ = 10010000100010001111011110101100

$H_8 = H_8 + H$ = 11100010111011111100110111101001

11) Setelah dijumlahkan nilai H_1 – H_8 diubah kedalam bentuk hex lalu digabungkan menjadi satu nilai *message digest*.

H_1 = 10111001010011010010011110111001 = B94D27B9

H_2 = 10010011010011010011111000001000 = 934D3E08

H_3 = 10100101001011100101001011010111 = A52E52D7

H_4 = 11011010011111011010101111111010 = DA7DABFA

H_5 = 11000100100001001110111111100011 = C484EFE3

H_6 = 01111010010100111000000011101110 = 7A5380EE

H_7 = 10010000100010001111011110101100 = 9088 F7AC

H_8 = 11100010111011111100110111101001 = E2EF CDE9

MD = B94D27B9934D3E08A52E52D7DA7DABFAC484EFE37A53 80EE9088F7ACE2EFCDE9

2.6. Algoritma RSA

Algoritma RSA adalah salah satu algoritma kunci publik yang dibuat oleh 3 orang peneliti dari MIT (Massachussets Institute of Technology) pada tahun 1976, yaitu Ron Rivest, Adi Shamir dan Leonard Adleman. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima. Pemfaktoran ini dilakukan untuk mencari kunci privat. (Deo Valiandro, 2020)

Skema enkripsi RSA terdiri dari tiga bagian proses, yaitu proses pembangkitan sepasang kunci, proses enkripsi dan proses dekripsi. Rumus enkripsi dan dekripsi plaintext M dan ciphertext C sebagai berikut :

$$C = M^e \ mod \ n$$

$$M = C^d \ mod \ n$$

Maka Bob (Pengirim) dan Alice (Penerima) keduanya harus mengetahui nilai dari n . Hanya Bob mengetahui nilai e, dan hanya Alice mengetahui nilai dari d. Jadi, algoritma enkripsi kunci publik dengan kunci publik $PU = \{e, n\}$ dan kunci privat $PK = \{d, n\}$. Maka pertama-tama Alice harus memilih 2 bilangan prima p dan q, dengan syarat $p \neq q$, untuk menghindari nilai p diperoleh dengan mencari nilai $\sqrt[2]{n^2}$. Kemudian Alice menghitung nilai $n = p.q$, yang nantinya akan digunakan untuk enkripsi pesan dan dekripsi pesan.

Setelah itu Alice menghitung nilai *totient* $\phi(n) = (p-1)(q-1)$. Setelah mendapatkan nilai *totient* , hubungan antara nilai e dan d dapat diekspresikan sebagai berikut :

$$ed \; mod \; \phi(n) = 1$$

Sehingga dapat dikatakan ekuivalen dengan :

$$e.d \; \equiv 1 \; mod \; \phi(n)$$

$$d \; \equiv e^{-1} \; mod \; \phi(n)$$

Artinya, nilai *e* dan *d* merupakan invers perkalian modulus $\phi(n)$, dimana menurut aturan matematika modular, pernyataan diatas hanya benar hanya jika *e* relatif prima terhadap $\phi(n)$ atau $\gcd(\phi(n), e) = 1$.

1. **Pembentukan Kunci oleh Alice.**
   a) Alice memilih nilai p dan q, keduanya merupakan bilangan prima, dan nilai p ≠ q.
   b) Alice menghitung nilai $n = p.q$.
   c) Alice menghitung nilai $\phi(n) = (p-1)(q-1)$
   d) Alice memilih nilai *e*, dengan syarat
   $$\gcd(\phi(n), e) = 1 \; ; \; 1 < e < \; \phi(n)$$
   e) Alice menghitung nilai $d \; \equiv e^{-1} \; mod \; \phi(n)$
   f) Maka Public Key (*PU*) = {*e*, *n*}
   g) Private Key (PK) = {*d*, *n*}.
2. **Enkripsi pesan oleh Bob lalu dikirim ke Alice.**

   Misalnya Bob mengirim pesan m ke Alice, nilai m haruslah terletak di dalam selang [0, n -1], jika panjang pesan m melebihi nilai n – 1, maka pesan dibagi menjadi blok-blok dengan ukuran lebih kecil. Kemudian dengan menggunakan kunci publik Alice, yaitu e dan n, Bob menghitung :

   $$C = M^e \; mod \; n$$

3. **Dekripsi pesan oleh Alice.**

   Alice mendekripsi pesan menggunakan kunci privat *d* dan *e* dengan menghitung :

   $$M = C^d \; mod \; n$$

Contoh : Misalkan kata “hello” ingin di enkripsi, pertama-tama kita harus menentukan nilai p dan q yang merupakan bilangan prima, dengan syarat p tidak boleh sama dengan q.

Misalnya nilai **p = 37, q = 41, dan e = 53**.

1. Pembentukan Kunci
   a. $p = 37, q = 41, e = 53$
   b. $n = p.q = 37 * 41 = 1517$
   c. $\phi(n) = (p-1)(q-1) = 36 * 40 = 1440$

$$\gcd(\phi(n), e) = 1 \; ; \; 1 < e < \phi(n)$$

   d. $de \equiv 1 \; mod \; \phi(n)$ atau $d \equiv \frac{1+i*1440}{53}$, dimana i merupakan suatu konstanta yang akan bertambah 1 hingga mendapatkan nilai *d* untuk memenuhi persamaan $de \equiv 1 \; mod \; \phi(n)$. Lalu didapatkan dengan nilai i = 47, nilai *d* = 1277
   e. Didapatkan Kunci Publik = {*53, 1517*}
   f. Kunci Privat = {*1277, 1517*}
2. Enkripsi
   a. Misalkan kita akan mengenkripsi huruf “h”, ubah terlebih dahulu menjadi kode ASCII yaitu 104.
   b. $C = M^e mod \; n = 104^{53} mod \; 1517 = 354$
   c. Cara a dan b dilakukan sampai kata “hello world” terenkripsi menjadi 3541491093709371258023712071258109809370010(peran “0” digunakan untuk memisahkan kata per kata sehingga pada dekripsi menjadi lebih mudah dipisah)
3. Dekripsi
   a. $M = 354^{1277} \; mod \; 1517 = 104$
   b. Lalu ubah kembali dari ASCII ke string, jadi 104 = h
   c. Lakukan cara a dan b, hingga mendapatkan kembali kata “hello world”.

2.7. Blokchain

(Shuyun Shi, 2020) Blokchain dibuat populer oleh suksesnya *Bitcoin* yang dibuat oleh Nakamoto pada tahun 2008 dan dapat dipercaya untuk memfasilitasi transaksi yang aman di seluruh jaringan asing tanpa tergantung dengan pihak ketiga yang terpusat. Blokchain adalah urutan kronologis blok yang termasuk daftar transaksi yang lengkap dan valid. Blok-blok pada blokchain saling terikat dengan blok sebelumnya dengan sebuah nilai *hash*, juga membentuk sebuah rantai. Blok yang paling pertama disebut juga dengan *genesis blok*.

Sebuah blok terdiri atas kepala blok dan badan blok. Dimana kepala blok terdiri atas :

a. Versi Blok : Index Blok
b. Previous Blok Hash : nilai hash dari blok sebelumnya.
c. Timestamp: Waktu pembentukan blok
d. Nonce : sebuah 4-bit acak yang *miners* sesuaikan untuk memecahkan teka-teki dari *proof-of-work* .

Badan blok terdiri dari :

a. Body root hash : nilai hash dari *merkle tree* atas transaksi / data dalam badan blok.
b. Target Hash: nilai ambang hash atas blok baru. *Target hash* digunakan untuk menentukan *difficulty* dari *proof-of-work* puzzle.

Proof-of-work (PoW) merupakan salah satu algoritma konsensus yang digunakan di Bitcoin. Penambang yang berhasil menghitung nilai *hash* dari blok akan mendapatkan sebuah imbalan, penambang harus melakukan tugas yang besar untuk menambang agar dapat dipercaya. Tugas tersebut memakan banyak biaya komputasi. Tugas *miners* adalah melakukan perhitungan nilai *hash* untuk mendapatkan nilai *nonce* yang cocok supaya nilai *hash* didapatkan sesuai dengan *target hash* yang telah ditentukan sebelumnya.

Merkle tree digunakan untuk menyimpan semua transaksi yang valid, di mana setiap *leaf node* adalah transaksi dan setiap *non-leaf node* adalah nilai hash dari dua

simpul anak yang digabungkan. Struktur pohon seperti itu efisien untuk verifikasi keberadaan dan integritas transaksi, karena setiap node dapat mengkonfirmasi validasi transaksi apa pun dengan nilai hash dari cabang yang sesuai daripada seluruh pohon *Merkle*. Sementara itu, setiap modifikasi pada transaksi akan menghasilkan nilai hash baru di lapisan atas dan ini akan menghasilkan hash root yang dipalsukan. Selain itu, jumlah maksimum transaksi yang dapat ditampung oleh satu blok tergantung pada ukuran setiap transaksi dan ukuran blok.

Blokchain memiliki karakteristik sebagai berikut : (Pankaj Dutta, 2020)

- *Decentralized* : Data dalam sistem dapat diakses, disetor, dimonitor, dan diupdate pada banyak sistem.
- *Transaparent* : Data direkam dan disetor ke dalam jaringan, dengan persetujuan dengan jaringan dan terbuka dan terlacak selamanya.
- *Immutable*: *Blokchain* memiliki *timestamp* dan memastikan dengan pasti pengendaliannya.
- *Irreversible* : Setiap transaksi yang dibuat, rekaman yang pasti dan terverfikasi tersimpan dalam *blokchain*.
- *Autonomy* : Setiap *node* dalam blokchain dapat mengakses, mengirim, menyetor, dan update data dengan sendiri tanpa pihak ketiga.
- *Open Source*: *Blokchain* terbuka untuk semua orang di dalam jaringan dengan sebuah hirarki.
- *Anonymity*: Saat transfer data terjadi antar node, identitas individu tetap anonim.
- *Ownership and uniqueness*: Setiap dokumen memiliki tanda kepemilikan dengan kode hash yang unik.
- *Provenance* : Setiap produk memiliki dokumen catatan digital di *blokchain* yang membuktikan keaslian dan asalnya.

Alur Kerja dari Blokchain seperti berikut ini : (R. Johari, 2021)

1. *Inisialisasi Blokchain.*

   Blokchain();

   Create_blok(prefix = 1, previousHash = '0') // Genesis Blok

2. Buat Blok baru ke dalam Blokchain :

   Create_blok(prefix, previousHash)

3. Print nilai atau isi setiap blok dalam *Blokchain* :

   Print $Blok_{index}$

   Print $Blok_{timestamp}$

   Print $Blok_{prefix}$

   Print $Blok_{records}$ *// Data*

   Print $Blok_{timestamp}$

4. Verifikasi dan Validasi *Blokchain* :

   Untuk setiap blok dalam *blokchain* Cek apakah Nilai

   $$previousHash_{currentBlok} = Hash_{previousBlok}$$

BAB III
METODE PENELITIAN

3.1. Waktu dan Tempat

Penelitian ini dilaksanakan dari bulan Juli 2021 dan dilaksankan di Laboratorium Rekayasa Perangkat Lunak Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

3.2. Tahapan Penelitian

Untuk melaksanakan penelitian ini, terdapat beberapa tahap yang dilakukan seperti :