BAB 2
TINJAUAN PUSTAKA

2.1 Profil Badan Pertanahan Nasional

Badan Pertanahan Nasional (BPN) adalah lembaga pemerintah yang mempunyai tugas di bidang pertanahan secara nasional, regional dan sektoral sesuai dengan ketentuan peraturan perundang-undangan. BPN menyediakan beberapa pelayanan untuk masyarakat, diantaranya pelayanan pendaftaran tanah, pelayanan pengukuran bidang tanah, pelayanan pemeliharaan data pendaftaran tanah, pelayanan pengaturan dan penataan pertanahan, pelayanan pencatatan dan informasi pertanahan dan pengelolaan pengaduan.

Badan Pertanahan Nasional mempunyai fungsi sebagai berikut:

1. Penyusunan dan penetapan kebijakan di bidang pertanahan.
2. Perumusan dan pelaksanaan kebijakan di bidang survei, pengukuran, dan pemetaan.
3. Perumusan dan pelaksanaan kebijakan di bidang penetapan hak tanah, pendaftaran tanah, dan pemberdayaan masyarakat.
4. Perumusan dan pelaksanaan kebijakan di bidang pengaturan, penataan dan pengendalian kebijakan pertanahan.
5. Perumusan dan pelaksanaan kebijakan di bidang pengadaan tanah.
6. Perumusan dan pelaksanaan kebijakan di bidang pengendalian dan penanganan sengketa dan perkara pertanahan.
7. Pengawasan atas pelaksanaan tugas di lingkungan BPN.
8. Pelaksanaan koordinasi tugas, pembinaan, dan pemberian dukungan administrasi kepada seluruh unit organisasi di lingkungan BPN.
9. Pelaksanaan pengelolaan data informasi lahan pertanian pangan berkelanjutan dan informasi di bidang pertanahan.
10. Pelaksanaan penelitian dan pengembangan di bidang pertanahan.
11. Pelaksanaan pengembangan sumber daya manusia di bidang pertanahan.

2.1.1 Sejarah Badan Pertanahan Nasional

Tahun 1988 merupakan tonggak bersejarah karena saat itu terbit keputusan Presiden Nomor 26 Tahun 1988 tentang Badan Pertanahan Nasional. Sejalan dengan meningkatnya pembangunan nasional yang menjadi tema sentral proyek ekonomi–politik orde baru, kebutuhan akan tanah juga makin meningkat. Persoalan yang dihadapi Direktorat Jenderal Agraria bertambah berat dan rumit. Untuk mengatasi hal tersebut, status Direktorat Jenderal Agraria ditingkatkan menjadi Lembaga Pemerintah Non Departemen dengan nama Badan Pertanahan Nasional. Dengan lahirnya keputusan Presiden Nomor 26 Tahun 1988 tersebut, Badan Pertanahan Nasional bertanggung jawab langsung kepada Presiden.

Berdasarkan Keputusan Presiden Nomor 96 Tahun 1993, tugas Kepala Badan Pertanahan Nasional kini dirangkap oleh Menteri Negara Agraria. Kedua lembaga tersebut dipimpin oleh satu orang sebagai Menteri Negara Agraria atau Kepala Badan Pertanahan Nasional. Dalam pelaksanaan tugasnya, Kantor Menteri Negara Agraria berkonsentrasi merumuskan kebijakan yang bersifat koordinasi.

Pada saat ini Badan Pertanahan Nasional Republik Indonesia berubah menjadi Kementerian Agraria dan Tata Ruang atau Badan Pertanahan Nasional berdasarkan peraturan Presiden Nomor 17 Tahun 2015 tentang Kementerian Agraria yang berfungsi Tata Ruang dan Peraturan Presiden Nomor 20 Tahun 2015 tentang Badan Pertanahan Nasional yang ditetapkan pada 21 Januari 2015.

2.1.2 Logo Badan Pertanahan Nasional

Logo Badan Pertanahan Nasional (BPN) dapat dilihat pada Gambar 2.1.



Gambar 2.1 Logo BPN

Logo Badan Pertanahan Nasional adalah bentuk suatu kesatuan gambar dan tulisan terdiri dari:

1. Gambar 4 (empat) butir padi melambangkan Kemakmuran dan kesejahteraan. Memaknai atau melambangkan 4 (empat) tujuan Penataan Pertanahan yang akan dan telah dilakukan BPN RI yaitu kemakmuran, keadilan, kesejahteraan sosial dan keberlanjutan.
2. Gambar lingkaran bumi melambangkan sumber penghidupan manusia. Melambangkan wadah atau area untuk berkarya bagi BPN RI yang berhubungan langsung dengan unsur-unsur yang ada didalam bumi yang meliputi tanah, air dan udara.
3. Gambar sumbu melambangkan poros keseimbangan. 3 (tiga) Garis Lintang dan 3 (tiga) Garis Bujur Memaknai atau melambangkan pasal 33 ayat 3 UUD 45 yang mandasari lahirnya Undang-undang Pokok Agraria (UUPA) nomor 5 tahun 1960.
4. Gambar 11 (sebelas) bidang grafis bumi memaknai atau melambangkan 11 (sebelas) agenda pertanahan yang akan dan telah dilakukan BPN RI. Bidang pada sisi sebelah kiri melambangkan bidang bumi yang berada diluar jangkauan wilayah kerja BPN RI.
5. Warna coklat melambangkan bumi, alam raya dan cerminan dapat dipercaya dan teguh.
6. Warna kuning emas melambangkan kehangatan, pencerahan, intelektual dan kemakmuran.
7. Warna abu-abu melambangkan kebijaksanaan, kedewasaan serta keseimbangan.

2.1.3 Visi dan Misi

Visi dan misi Badan Pertanahan Nasional (BPN) adalah sebagai berikut:

2.1.3.1 Visi

Menjadi lembaga yang mampu mewujudkan tanah dan pertanahan untuk sebesar-besar kemakmuran rakyat, serta keadilan dan keberlanjutan sistem kemasyarakatan, kebangsaan dan kenegaraan Republik Indonesia.

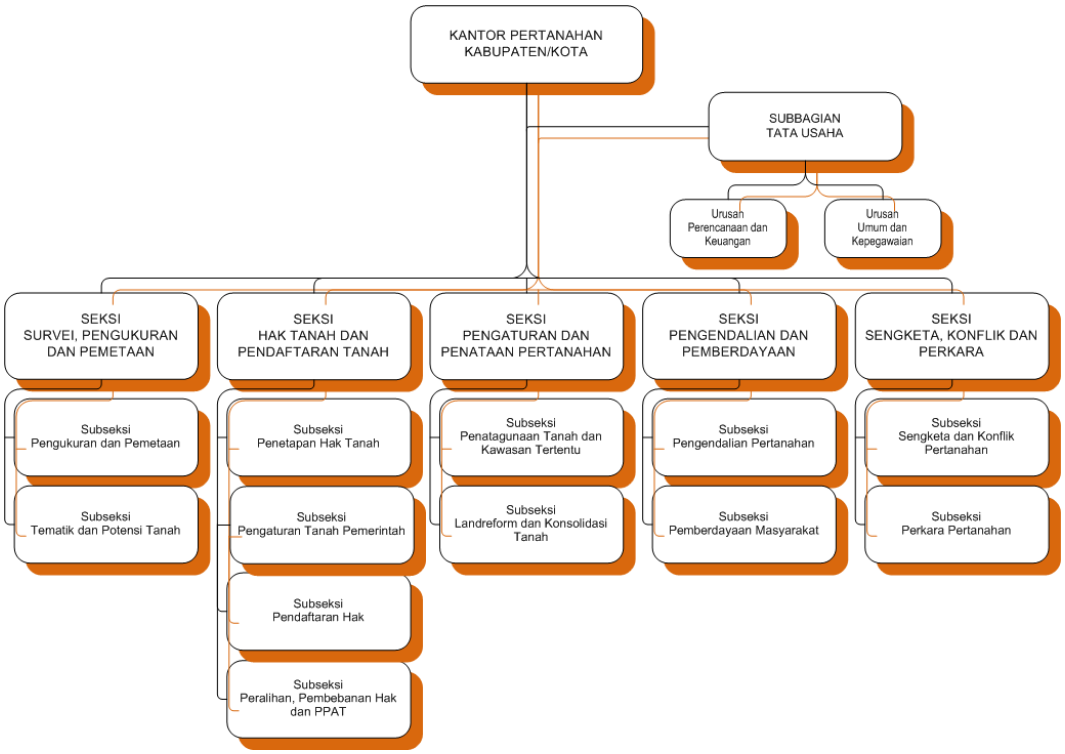
2.1.3.2 Misi

Mengembangkan dan menyelenggarakan politik dan kebijakan pertanahan untuk:

1. Peningkatan kesejahteraan rakyat, penciptaan sumber-sumber baru kemakmuran rakyat, pengurangan kemiskinan dan kesenjangan pendapatan, serta pemantapan ketahanan pangan.
2. Peningkatan tatanan kehidupan bersama yang lebih berkeadilan dan bermartabat dalam kaitannya dengan penguasaan, pemilikan, penggunaan dan pemanfaatan tanah.
3. Perwujudan tatanan kehidupan bersama yang harmonis dengan mengatasi berbagai sengketa, konflik dan perkara pertanahan di seluruh tanah air dan penataan perangkat hukum dan sistem pengelolaan pertanahan sehingga tidak melahirkan sengketa, konflik dan perkara di kemudian hari.
4. Keberlanjutan sistem kemasyarakatan, kebangsaan dan kenegaraan Indonesia dengan memberikan akses seluas-luasnya pada generasi yang akan datang terhadap tanah sebagai sumber kesejahteraan masyarakat. Menguatkan lembaga pertanahan sesuai dengan jiwa, semangat, prinsip dan aturan yang tertuang dalam UUPA dan aspirasi rakyat secara luas.

2.1.4 Struktur Organisasi

Struktur organisasi Badan Pertanahan Nasional Kota Cimahi dapat dilihat seperti pada Gambar 2.2.



Gambar 2.2 Struktur Organisasi BPN

2.2 Landasan Teori

Landasan teori bertujuan untuk memberikan gambaran sumber dan kajian dari teori–teori yang berkaitan dengan pembangunan sistem. Berikut landasan teori yang digunakan dalam melakukan penelitian.

2.2.1 Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan. Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan mengambil mengubah mengolah dan mengkomunikasikan informasi

yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya [5].

2.2.2 Internet

Internet (*International Network*) dapat diartikan sebagai jaringan komputer internasional, ribuan sistem komputer saling berhubungan satu dengan lainnya. Kehadiran internet telah membiaskan batas-batas negara sehingga berbagai informasi penting dapat dengan cepat dan sangat mudah didistribusikan ke seluruh penjuru dunia, maka abad ini kemudian disebut sebagai abad informasi. Dengan adanya komputer yang saling terhubung tersebut, memungkinkan terjadinya pertukaran *file* data dan informasi yang terdapat pada masing-masing komputer. Hingga kini telah tercatat lebih dari 80.000 jaringan yang saling terhubung di seluruh dunia. Hal ini menjadikan internet suatu komunitas baru dari masyarakat yang sangat besar yang dikenal dengan istilah *cyberspace* [5].

2.2.3 Keamanan dan Kerahasiaan Data

Keamanan atau *security* adalah mekanisme dan teknik untuk melindungi sesuatu yang dapat berupa data atau informasi di dalam sistem. Pada dasarnya *security* adalah sistem yang digunakan untuk melindungi sistem dalam suatu jaringan keamanan agar tetap terjaga. Masalah keamanan dan kerahasiaan data merupakan salah satu aspek penting dari suatu sistem informasi. Dalam hal ini, sangat terkait dengan betapa pentingnya informasi tersebut diterima oleh orang yang berkepentingan. Informasi akan menjadi tidak *valid* lagi jika informasi tersebut di ketahui atau dibajak oleh orang yang tidak berhak [6].

Informasi atau data saat ini sudah menjadi sesuatu yang sangat penting bagi sebuah organisasi, perguruan tinggi, lembaga pemerintahan maupun individual, termasuk kemampuan dalam mengakses dan menyediakan informasi secara cepat serta akurat. Karena pentingnya sebuah informasi, seringkali informasi yang diinginkan hanya dapat diakses oleh orang tertentu misalnya pihak penerima yang diinginkan, dan jika informasi ini sampai diterima oleh pihak yang tidak diinginkan akan berdampak kerugian pada pihak pengirim.

2.2.4 Aspek Keamanan Informasi

Keamanan informasi adalah bagaimana kita dapat mencegah penipuan, atau paling tidak mendeteksi adanya penipuan di sebuah sistem yang berbasis informasi, untuk melindungi informasi dari pengaksesan, penggunaan, penyebaran, perusakan, perubahan, dan penghancuran tanpa otorisasi yang sah. Untuk itu diperlukanlah sebuah pendekatan dalam melakukan pengamanan pada informasi, seperti melakukan enkripsi, *steganografi*, *cipher, digital signature* dan *hashing* terhadap informasi tersebut.

Aspek keamanan komputer adalah bentuk pertimbangan yang menyatakan sebuah komputer bisa dinyatakan aman. Aspek-aspek keamanan di dalam kriptografi adalah sebagai berikut: [7]

1. *Confidentiality* (kerahasiaan) yaitu merupakan usaha untuk menjaga informasi dari orang yang tidak berhak mengakses.
2. *Authentication* (otentikasi) yaitu agar penerima informasi dapat memastikan keaslian pesan tersebut datang dari orang yang dimintai informasi
3. *Non-repudiation* (nir penyangkalan) yaitu merupakan hal yang yang bersangkutan dengan si pengirim, si pengirim tidak dapat mengelak bahwa dia lah yang mengirim informasi tersebut.
4. *Integrity* (data Integritas) yaitu keaslian pesan yang dikirim melalui sebuah jaringan dan dapat di pastikan bahwa informasi yang dikirim tidak di modifikasi oleh orang yang tidak berhak dalam perjalanan informasi tersebut.

2.2.5 Basis Data

Basis data atau *database* adalah kumpulan data yang salaing berelasi yang disimpan dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil (*query*) basis data disebut *Database Management System* (DBMS), dan sistem basis data yang dipelajari disebut sistem informasi.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dan pengetahuan. Sebuah basis data memiliki penjelasan yang tersimpan

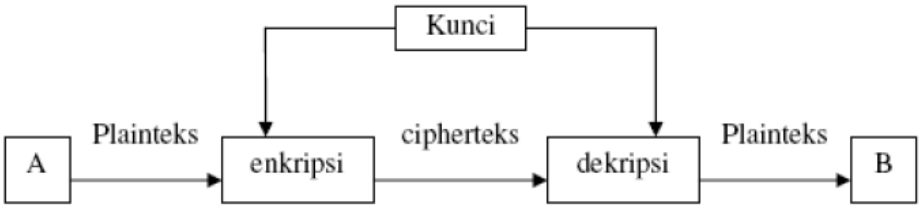
di dalamnya yang di sebut skema. Skema menggambarkan objek yang dimiliki oleh suatu basis data dan relasi (hubungan). Jika berbicara basis data, maka seluruh data disimpan dalam basis data pada masing-masing tabel atau *file* sesuai dengan fungsinya. Karena itu dapat melakukan penelusuran data yang diinginkan dengan mudah sehingga akan cepat mendapatkan informasi yang diperlukan [11].

2.2.6 Kriptografi

Kriptografi adalah ilmu yang mempelajari bagaimana supaya pesan atau dokumen kita aman, tidak bisa dibaca oleh pihak yang tidak berhak. Terdapat dua jenis algoritma kriptografi berdasar jenis kuncinya yaitu: [8]

1. Algoritma Simetri

Algoritma simetri disebut juga sebagai algoritma konvensional adalah algoritma yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Algoritma simetrik sering juga disebut sebagai algoritma kunci rahasia, algoritma kunci tunggal, atau algoritma satu kunci. Pada Gambar 2.3 dinyatakan bahwa kriptografi simetris pada saat di enkripsi dan di dekripsi menggunakan kunci yang sama.

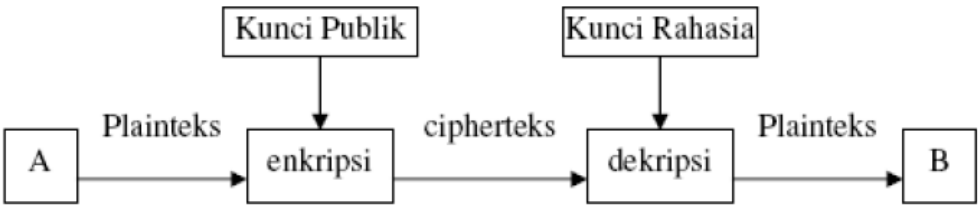


Gambar 2.3 Proses Enkripsi dan Dekripsi Kriptografi Simetri

2. Algoritma Asimetri

Algoritma asimetri didesain sedemikian sehingga kunci yang digunakan untuk enkripsi berbeda dari kunci yang digunakan untuk dekripsi, pada Gambar 2.4 terlihat kunci untuk enkripsi tidak sama dengan kunci untuk dekripsi. Kunci untuk enkripsi tidak rahasia, sehingga dinamakan juga kunci publik (*public key*), sedangkan kunci untuk dekripsi rahasia, sehingga dinamakan kunci privat (*privat key*). Pengirim pesan mengenkripsi pesan dengan menggunakan kunci publik si

penerima pesan, hanya penerima pesan yang dapat mendekripsi pesan menjadi plainteks semula dengan menggunakan kunci privatnya. Contoh algoritma kriptografi asimetris diantaranya adalah RSA (Rivest-Shamir-Adleman), Rabin, dan ElGamal.



Gambar 2.4 Proses Enkripsi dan Dekripsi Kriptografi Asimetri

2.2.7 Digital Signature

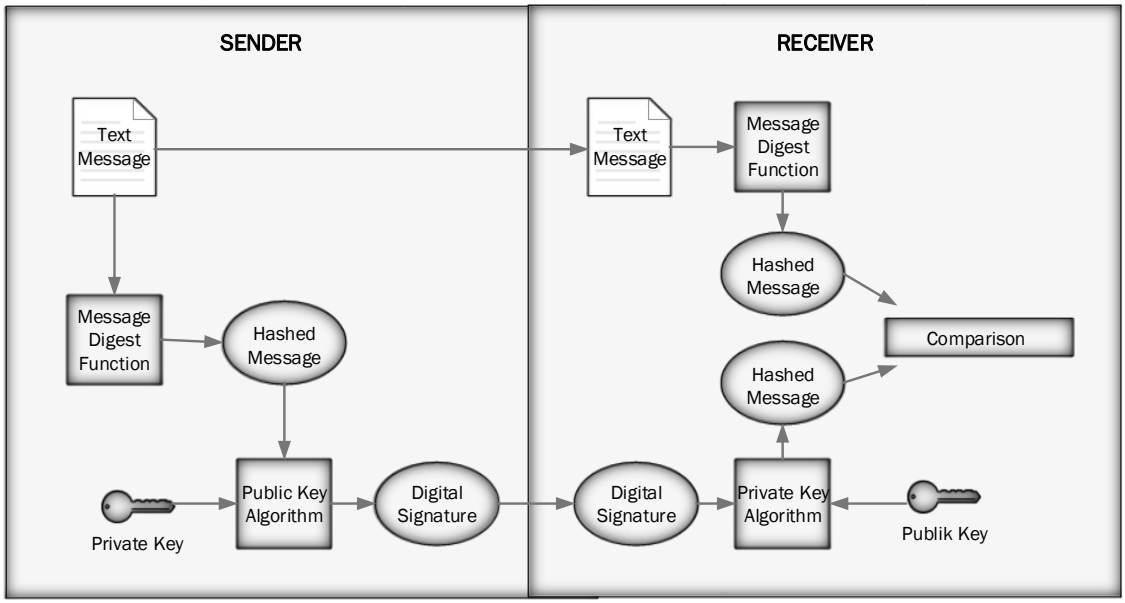
Salah satu konsep pada kriptografi modern adalah *digital signature*. Cara kerja dan kegunaan *digital signature* mirip dengan tanda tangan dalam versi nyata, yaitu untuk memberikan kepastian keaslian dan persetujuan dokumen oleh penanda tangan. Dalam *digital signature*, "tanda tangan" adalah dalam bentuk digital yang digunakan untuk mensahkan sebuah dokumen digital.

Prinsip yang digunakan dalam tanda tangan digital ini adalah dokumen yang dikirimkan harus ditandatangani oleh pengirim dan tanda tangan bisa diperiksa oleh penerima untuk memastikan keaslian dokumen yang dikirimkan. Fungsinya adalah untuk melakukan validasi terhadap data yang dikirim. Tanda tangan digital menggunakan algoritma yang disebut dengan istilah *hashing algorithm*. Fungsi tersebut akan menghasilkan sebuah kombinasi karakter yang yang unik yang disebut *message digest*. dengan cara ini pengirim bertanggungjawab terhadap isi dokumen dan dapat di cek keaslian dokumen oleh penerima.

Keunikannya adalah jika di tengah perjalanan data mengalami modifikasi, penghapusan maupun di sadap diam-diam oleh *hacker* walaupun hanya 1 karakter saja, maka *message digest* yang berada pada si penerima akan berbeda dengan yang dikirimkan pada awalnya. Keunikan lainnya adalah *message digest* tersebut tidak

bisa dikembalikan lagi ke dalam bentuk awal seperti sebelum disentuh dengan fungsi algoritma, sehingga disebutlah sebagai *one-way hash* [2].

Fungsi utama dari tanda tangan digital pada pada aspek keamanan kriptografi adalah *non-repudiation* atau anti penyangkalan dimana apabila dokumen valid maka pengirim tidak bisa menyangkal bahwa keberadaan dokumen benar dikirim oleh pengirim yang bersangkutan.



Gambar 2.5 Skema Digital Signature

Cara kerja *digital signature* seperti yang terlihat pada Gambar 2.5 adalah sebagai berikut:

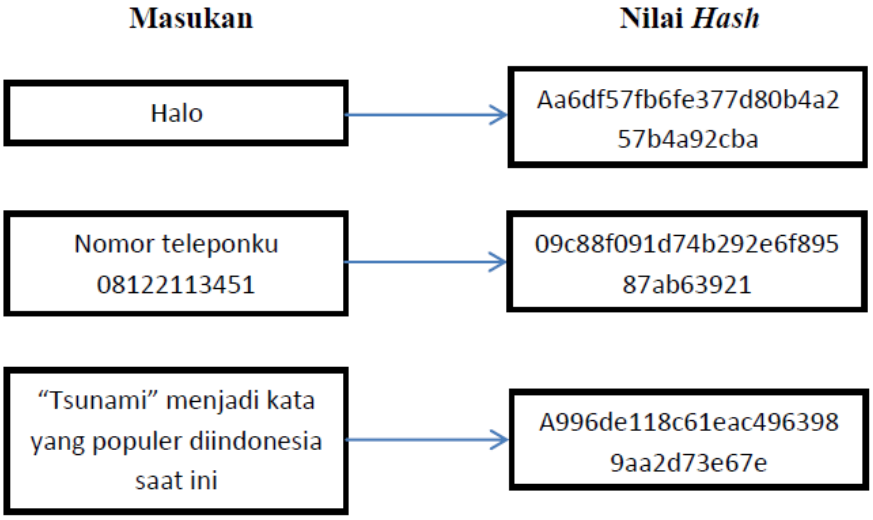
1. Sender melakukan proses *hashing algorithm* untuk menghasilkan *message digest* dari sebuah pesan yang terdapat dalam sebuah dokumen yang akan dikirim.
2. Setelah dilakukan *hashing*, Sender melakukan *sign* terhadap *message digest* dengan menggunakan kunci publik yang digunakan untuk membentuk *digital signature*.
3. Kemudian Sender mengirimkan *digital signature* bersama dokumen tersebut kepada Receiver.
4. Receiver merima pesan yang dikirimkan oleh Sender.

5. Setelah itu Receiver mengverifikasi pesan yang dikirimkan oleh Sender. Pada proses verifikasi tersebut pesan di *hashing* terlebih dahulu sehingga menghasilkan *message digest* dan *digital signature* akan di *unsign* menggunakan kunci private. Jika *message digest*-nya sama, maka pesan ini adalah asli dan pesan berasal dari pengirim yang sebenarnya. Bila pesan telah diubah oleh pihak luar, maka *message digest* juga ikut berubah.

Proses diatas mampu membuktikan bahwasanya pesan adalah asli (*message authentication*) dan orang yang mengirim adalah orang yang sebenarnya (*user authentication*). Ini berarti *digital signature* memenuhi salah satu syarat keamanan jaringan yaitu *non-repudiation* atau nir penyangkalan.

2.2.8 Fungsi Hash

Fungsi *hash* adalah suatu fungsi yang menerima masukan berupa string yang panjangnya sembarang dan mengonversi masukan tersebut menjadi string yang mempunyai panjang tetap (*fixed*) dan umumnya menjadi lebih kecil dari panjang semula. Keluaran dari fungsi *hash* disebut juga nilai *hash* atau pesan ringkas (*message digest*). Fungsi *hash* merupakan fungsi satu arah (*one way function*) yang dapat menghasilkan ciri (*signature*) dari data. Perubahan satu bit saja akan mengubah keluaran *hash* secara drastis. Fungsi *hash* biasanya digunakan untuk menjamin integritas dan *digital signature*.



Gambar 2.6 Contoh Penggunaan Fungsi Hash

Pada Gambar 2.6 [7], memperlihatkan fungsi *hash* yang mengubah suatu string dengan panjang berapapun menjadi sebuah *message digest* yang memiliki panjang tetap. Fungsi *hash* pada dasarnya bekerja satu arah, berarti pesan asli atau pesan semula akan diubah menjadi sebuah *message digest*, namun *message digest* yang dihasilkan tidak dapat dikembalikan menjadi pesan asli atau pesan semula kembali. Pengirim dan penerima memiliki cara sehingga keutuhan data dapat diselidiki, parameter dari fungsi *hash* yang biasa digunakan dapat dilihat pada Tabel 2.1 serta daftar algoritma fungsi *hash* pada Tabel 2.2.

Tabel 2.1 Parameter fungsi hash

Parameter	Keterangan
M	Pesan
h	Fungsi *hash*
y	Merupakan *h(M)* atau *message digest*

Tabel 2.2 Daftar algoritma fungsi hash

Algoritma	Output size	Internal state size	Block size	Lenght size	Word size	Collision
HAVAL	256/224/192/160/128	256	1024	64	32	Yes
MD2	128	384	128	No	8	Almost
MD4	128	128	512	64	32	Yes
MD5	128	128	512	64	32	Yes
PANAMA	256	8736	256	No	32	With flaws
RIPEMD	128	128	512	64	32	Yes
RIPEMD-128/256	128/256	128/256	512	64	32	No
RIPEMD-160/320	160/320	160/320	512	64	32	No
SHA-0	160	160	512	64	32	Yes
SHA-1	160	160	512	64	32	With flaws
SHA-256/224	256/224	256	512	64	32	No
SHA-512/384	512/384	512	1024	128	64	No
Tiger(2)-192/160/128	192/160/128	192	512	64	64	No
VEST-4/8	160/256	176/304	8	80	1	No
VEST-16/32	320/512	424/680	8	88	1	No
WHIRLPOOL	512	512	512	256	8	No

Fungsi *hash h* memiliki sifat-sifat sebagai berikut:

1. *Preimage resistant*

   Jika diberikan suatu nilai *hash y*, akan sulit mencari *M* sedemikian sehingga *h(M)=y*

2. *Second preimage resistant*

   Jika diberikan sebuah masukan *M*, akan sulit mencari *M'* sedemikian sehingga *h(M)=h(M')*

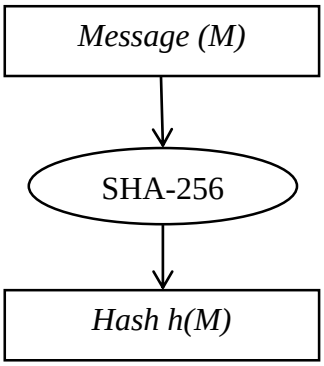
3. *Collision resistant*

   Akan sulit untuk mencari *M* dan *M'* sedemikian sehingga *h(M)=h(M')*

2.2.9 Secure Hash Algorithm-256 *(SHA-256)*

Secure Hash Algorithm-256 adalah salah satu jenis *hash* yang masih umum digunakan. Fungsi ini adalah varian dari SHA-1, SHA-256 dibuat karena telah ditemukan bentrokan dari SHA-1, SHA-1 sendiri adalah pengganti dari SHA-0.

Sampai saat ini belum ada yang dapat memecahkan algoritma untuk SHA-256. SHA-256 umumnya digunakan sebagai fungsi antara untuk fungsi lain, termasuk fungsi *hash* MAC, HMAC, dan beberapa fungsi penghasil *digital signature* [3]. Fungsi utama SHA-256 dapat dilihat pada Gambar 2.7.



Gambar 2.7 Fungsi utama SHA-256

Fungsi utama SHA-256 menerima masukan berupa data atau pesan *M* dengan panjang sembarang, lalu akan menghasilkan nilai *hash h(M)* dengan panjang 256-bit. SHA-256 mempergunakan 6 fungsi logika, yang setiap fungsi tersebut beroperasi pada 32-bit *words* yang direperesentasikan sebagai *x*, *y*, and *z*. Hasil dari

setiap fungsi merupakan sebuah 32-bit *word* baru. Berikut fungsi logika dalam SHA-256:

$Ch(x,y,z) = (x \wedge y) \oplus (\neg x \wedge z)$

$Maj(x,y,z) = (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z)$

$\sum_0^{\{256\}}(x) = \text{ROTR}^2(x) \oplus \text{ROTR}^{13}(x) \oplus \text{ROTR}^{22}(x)$

$\sum_1^{\{256\}}(x) = \text{ROTR}^6(x) \oplus \text{ROTR}^{11}(x) \oplus \text{ROTR}^{25}(x)$

$\sigma_0^{\{256\}}(x) = \text{ROTR}^7(x) \oplus \text{ROTR}^{18}(x) \oplus \text{SHR}^3(x)$

$\sigma_1^{\{256\}}(x) = \text{ROTR}^{17}(x) \oplus \text{ROTR}^{19}(x) \oplus \text{SHR}^{10}(x)$

Keterangan:

ROTR (*Right-rotate*)

SHR (*Right-shift*)

2.2.10 Algoritma RSA

RSA ditemukan oleh tiga orang yang kemudian disingkat menjadi RSA. Ketiga penemu itu adalah Ron Rivest, Adi Shamir, dan Leonard Adleman. RSA termasuk algoritma asimetri, yang berarti memiliki dua kunci, yaitu kunci publik dan kunci privat [7].

RSA menjadi sistem kriptografi kunci-publik yang terpopuler karena merupakan sistem pertama yang sekaligus dapat digunakan untuk *key distribution*, *confidentiality* dan *digital signature*.

2.2.10.1 Pembangkit Kunci RSA

Hal pertama yang dilakukan algoritma pembangkit kunci adalah membangkitkan 2 bilangan prima besar. Pembangkitan bilangan prima besar menggunakan algoritma pengujian bilangan prima misalnya algoritma Miller-Rabin. Untuk membangkitkan kunci RSA terdapat beberapa parameter yaitu *p*, *q*, *n*, *ø (n)*, *e*, *d*, *Kpublik*, dan *Kprivat*. Keterangan dari masing-masing parameter dapat dilihat pada Tabel 2.3.

Tabel 2.3 Parameter pembangkit kunci RSA

Parameter	Keterangan
p	Bilangan prima
q	Bilangan prima
n	Merupakan hasil dari $p \times q$
ø (n)	Merupakan hasil dari $(p-1) \times (q-1)$
e	Dengan ketentuan, gcd (ø (n), e) = 1 gcd = *greatest common divisor*
d	e^{-1} *(mod ø (n))* Menggunakan algoritma *extended euclid*
Kpublik	*(e,n)*
Kprivat	*d*

Contoh pembangkitan kunci publik dan kunci privat sistem kriptografi RSA:

1. Pilih bilangan prima *p* dan *q*, misalnya $p = 17$ dan $q = 31$
2. Cari $n = p \times q = 17 \times 31 = 527$
3. Hitung $ø(n) = (p-1) \times (q-1) = (17-1) \times (31-1) = 16 \times 30 = 480$
4. Misalnya $e = 77$, nilai 77 dipilih karena memenuhi syarat gdc(480,77) = 1
5. Didapat $d = e^{-1} = 77^{-1} \bmod 480$, dengan menggunakan algoritma *extended euclid* $d = 293$
6. Maka *Kpublik* = *(e,n)* = (77,527) dan
7. *Kprivat* = *d* = 293

2.2.10.2 Algoritma Enkripsi RSA

Setelah kunci publik *Kpublik* dibangkitkan oleh pendekripsi, maka sembarang orang dapat menggunakan kunci publik tersebut. Algoritma enkripsi RSA menggunakan fungsi eksponensial dalam modular *n*, seperti yang dijelaskan pada Tabel 2.4.

Tabel 2.4 Algoritma enkripsi RSA

Algortima Enkripsi RSA	
Input	*Kpublik = (e,n)*
Output	$C = P^e \bmod n$ Menggunakan algoritma *square and multipy*

Contoh enkripsi RSA, menyambung dari contoh pembangkitan kunci publik dan kunci privat sistem kriptografi RSA sebelumnya:

1. Misal diberikan teks asli $P = 51$

2. Karena kunci publik *Kpublik* = (77,527) berarti *e* = 77 dan *n* = 527
3. Maka teks sandi C = P^e mod n = 51^{77} mod 527 = 493

2.2.10.3 Algoritma Dekripsi RSA

Seperti halnya enkripsi, algoritma dekripsi RSA merupakan fungsi eksponensial modular *n* dengan menggunakan kunci privat, seperti yang dijelaskan pada Tabel 2.5.

Tabel 2.5 Algoritma dekripsi RSA

Algortima Dekripsi RSA	
Input	*Kprivat = d* *Kpublik = (e,n)*
Output	$P = C^d \bmod n$

Contoh dekripsi RSA, menyambung dari contoh pembangkitan kunci publik dan kunci privat sistem kriptografi RSA dan contoh enkripsi sebelumnya:

1. Telah diketahui kunci privat *Kprivat* = 293 berarti *d* = 293
2. Juga telah diketahui *n* = 527 dan teks sandi *C* = 493
3. Maka $P = C^d \bmod n = 493^{293} \bmod 527 = 51$
4. Jadi teks asli *P* = 51 bernilai sama dengan teks asli pada contoh enkripsi.

2.2.11 Metode Analisis dan Perancangan Terstruktur

Metode analisis dan perancangan terstruktur merupakan aktivitas mentransformasikan suatu hasil analisis ke dalam suatu perencanaan untuk dapat diimplementasikan. Pendekatan terstruktur dilengkapi dengan alat-alat (*tools*) yang dibutuhkan dalam pengembangan sistem seperti *Flowmap, Data Flow Diagram* (DFD), Kamus Data, dan *Entity Relationship Diagram* (ERD).

2.2.11.1 Flowmap

Bagan alir dokumen (dokumen *flowchart*) atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan didalam bagan alir sistem [9].

2.2.11.2 Data Flow Diagram

Data Flow Diagram (DFD) merupakan suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunanya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. DFD juga merupakan alat untuk membuat diagram sederhana. *Data Flow Diagram* terdiri dari notasi penyimpanan data (*data store*), proses (*process*), aliran data (*data flow*), dan sumber masukan (*external entity*) [11].

2.2.11.3 Kamus Data

Kamus data (*data dictionary*) merupakan daftar elemen data yang terorganisir dengan definisi yang tetap dan sesuai dengan system, sehingga *user* dan analis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan *data storage*. Kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari sistem informasi. Selain digunakan untuk dokumentasi dan mengurangi redudansi, kamus data juga dapat digunakan untuk:

1. Memvalidasi diagram arus data dalam hal kelengkapan dan keakuratan.
2. Menyediakan suatu titik awal untuk mengembangkan layar dan laporan-laporan.
3. Menentukan muatan data yang disimpan dalam *file-file.*
4. Mengembangkan logika untuk proses-proses diagram arus data.

2.2.11.4 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah diagram yang memperlihatkan entitas-entitas yang terlibat dalam suatu sistem serta relasi antar entitas. ERD merupakan tabel-tabel yang merepresentasikan entitas-entitas serta tabel-tabel yang merepresentasikan relasi antar entitas itu sendiri.

ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga simbol yang digunakan, yaitu: [11]

a. *Entity*

   Entity merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain . Simbol dari *entity* ini biasanya digambarkan dengan persegi panjang.

b. Atribut

   Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai suatu yang dapat mengidentifikasikan isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

c. Hubungan / Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

2.2.12 PHP

PHP (*Hypetext Preprocessor)* pertama kali diciptakan oleh Rasmus Lerdorf pada tahun 1994. PHP merupakan bahasa pemograman yang digunakan dalam dunia *website* dan bisa digunakan bersamaan dengan HTML (*HyperText Markup Language*). PHP dapat diartikan sebagai *Hypertext Preprocessor* yang merupakan bahasa yang dapat digunakan pada *server* yang hasilnya dapat ditampilkan pada *client*.

PHP memiliki kemampuan untuk mencatat seluruh informasi dari pengunjung situs *online*-nya, *interface* atau *parser* tersebut dimodifikasi dengan mendukung database MySQL atau *Mini Structur Query Language* dengan menggunakan *parser* SQL. Dalam hal ini *user* menggunakan kode-kode PHP untuk mengirimkan permintaan ke *server*.

Kegunaan dari PHP adalah untuk membuat tampilan *website* menjadi lebih dinamis, dengan php kita bisa menampilkan atau menjalankan beberapa *file* dalam 1 *file* dengan cara di *include* atau *require*, dan php itu sendiri sudah bisa beriteraksi dengan beberapa *database* walaupun dengan kelengkapan yang berbeda, seperti *FilePro (Personic, Inc)*, DBM, *Informix, Ingres, InterBase, Microsoft Access*, MSSQL, MySQL, *Oracle*, PostgrSQL, *Sybase*.

2.2.13 MySQL

MySQL (Mini Structur Query Language) adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisesnsi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya, SQL (*Structured Query Language*) adalah sebuah konsep pengoperasian basis data terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara automatis.

MySQL merupakan aplikasi *database* yang mudah digunakan karena telah banyak digunakan di berbagai aplikasi dan *website*. Pada *web hosting* juga sudah banyak yang meneyediakan *database* MySQL ini, dalam MySQL ada beberapa operasi dasar yang lebih dikenal dengan operasi CRUD yaitu:

1. *Create*

   Create adalah operasi penambahan data baru kedalam tabel. Terdapat 2 *query* untuk menambah data, yang pertama adalah penambahan yang tidak menspesifikasikan nama kolom yang akan ditambahkan dan hanya memberikan isi dari tabelnya. *Query*-nya adalah sebagai berikut:

   INSERT INTO table_name

   VALUES (value1,value2,value3,...);

   Untuk yang kedua adalah menginputkan data dengan mencantumkan nama kolom yang akan diisikan dengan isi di dalamnya. *Query*-nya adalah sebagai berikut:

   INSERT INTO *table_name* (*column1,column2,column3,...*)

   VALUES (*value1,value2,value3,...*);

2. *Read*

   Read adalah operasi untuk menampilkan semua atau sebagian data yang berada di dalam *database*. *Query*-nya adalah sebagai berikut:

   SELECT *column_name,column_name*

FROM *table_name*;

3. *Update*

   Update adalah operasi untuk mengubah data yang ada di dalam *database*.

   Query-nya adalah sebagai berikut:

   UPDATE *table_name*

   SET *column1=value1,column2=value2,...*

   WHERE *some_column=some_value*;

4. *Delete*

   Delete adalah operasi untuk menghapus data yang ada di dalam tabel.

   Query-nya adalah sebagai berikut:

   DELETE FROM *table_name*

   WHERE *some_column=some_value*;

2.2.14 Pengujian Sistem

Pada pengujian sistem terdapat rencana pengujian untuk menentukan jenis pengujian yang akan dilakukan.

2.2.14.1 Pengujian Black Box

Pengujian *black box* merupakan pendekatan komplementer dari teknik *white box*, karena pengujian *black box* diharapkan mampu mengungkap kelas kesalahan yang lebih luas dibandingkan teknik *white box*. Pengujian *black box* berfokus pada pengujian persyaratan fungsional perangkat lunak, untuk mendapatkan serangkaian kondisi input yang sesuai dengan persyaratan fungsional suatu program.

Pengujian *black box* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian *black box* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkitkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan [12].

2.2.14.2 Pengujian Beta

Pengujian Beta (*beta testing*) didefinisikan sebagai pengujian tahap akhir suatu produk sebelum diluncurkan secara komersial. Pengujian beta biasanya dilakukan terutama pada produk yang berkaitan dengan teknologi informasi. Analogi pengujian beta ini dapat diterapkan untuk produk atau jasa yang lain. Umumnya pengujian tersebut dilakukan dengan mengirimkan atau memberi kesempatan kepada pengguna untuk dapat mencoba produk atau jasa tersebut [12].