Implementasi *Message Authentication Code* dan *Digital Signature* untuk *API Content Integrity*

Jason Kanggara - 13520080
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13520080@std.stei.itb.ac.id

***Abstract*—Di era digital ini, pengembangan web sudah menjadi hal yang umum dan menjadi sarana informasi yang paling banyak digunakan oleh masyarakat. Dengan banyaknya jumlah pengguna web, sebagai pengembang perlu memperhatikan aspek keamanan untuk memastikan data pengguna tidak terancam dari serangan yang tidak diinginkan. Aplikasi web banyak menggunakan API, atau *Application Programming Interface*, untuk memproses seluruh kegiatan pendukung aplikasi, seperti autentikasi pengguna, menyimpan data, melihat data, dan sebagainya. Untuk mengamankan API tersebut, perlu dibuat suatu pengamanan API agar API tersebut dapat terhindar dari pengubahan oleh seorang penyerang. Dalam makalah ini, akan menerapkan ilmu kriptografi, dalam kasus ini adalah memanfaatkan MAC, atau *Message Authentication Code*, dan *Digital Signature* untuk mengamankan API terhadap pengubahan secara ilegal.**

Keywords—API, Digital Signature, Message Authentication Code, Web, Kriptografi

I. Pendahuluan

Di era modern ini, penggunaan teknologi sudah sangat berkembang hingga teknologi menjadi bagian dari kehidupan kita setiap harinya. Perkembangan ini tentu tidak lepas dari kebutuhan akan pengamanan pada website yang lebih baik untuk mencegah berbagai jenis upaya penyerangan yang terjadi. Penyerangan ini dapat terjadi dengan melakukan manipulasi atau pengubahan terhadap API. Jika terjadi pengubahan yang membahayakan pada suatu API, hal tersebut dapat membahayakan pengguna saat melakukan segala jenis transaksi data pada API tersebut.

Untuk mengatasi permasalahan ini, diperlukan suatu metode yang dapat mencegah adanya pengubahan API secara ilegal. Ada banyak metode yang dapat digunakan untuk mengamankan API dari modifikasi yang tidak diinginkan. Salah satu cara yang dapat digunakan untuk mencegah modifikasi ilegal adalah dengan memanfaatkan teknik kriptografi. Penggunaan teknik kriptografi dapat memastikan kebenaran atau integritas dari konten yang terdapat pada API. Jika ada modifikasi yang tidak terverifikasi pada API, akan terlihat adanya ketidakcocokan pada hasil enkripsi antara API sesungguhnya dengan API hasil modifikasi ilegal.

Terdapat beberapa teknik kriptografi yang dapat digunakan untuk menyelesaikan permasalahan ini. Pada makalah ini, akan digunakan teknik *Message Authentication Code* dan *Digital Signature* untuk memastikan keaslian dari isi serta fungsi dalam suatu API. *Message Authentication Code*, atau MAC, akan digunakan untuk memastikan integritas konten pada API tetap terjaga. Hal tersebut dapat mengamankan API dari modifikasi secara ilegal. *Digital Signature* akan digunakan untuk memastikan pengubahan pada API hanya dilakukan oleh mereka yang memiliki akses untuk mengubah API tersebut.

Pemanfaatan kedua teknik tersebut diharapkan dapat mencegah modifikasi yang tidak diinginkan pada API guna mengamankan segala transaksi data yang dilakukan oleh pengguna. Dengan adanya MAC serta *Digital Signature*, API tidak dapat dimodifikasi oleh seseorang yang tidak memiliki akses melakukan pengubahan. Aplikasi web nantinya akan melakukan dekripsi pada MAC untuk memvalidasi kecocokan terhadap isi konten API yang sesungguhnya. Hal ini memastikan aplikasi web di sisi client tidak menjalankan segala transaksi pada API yang dimodifikasi secara ilegal.

Pada makalah ini, akan dilakukan enkripsi pada suatu file API dengan MAC dengan tanda tangan digital untuk memastikan perubahan pada API dilakukan oleh mereka yang memiliki wewenang untuk mengubahnya. Nanti, saat melakukan permintaan kepada API dari sisi klien, akan dilakukan validasi pada MAC untuk memastikan tidak adanya modifikasi yang tidak sah pada API agar tidak membahayakan pengguna.

II. Dasar Teori

A. *Application Programming Interface*

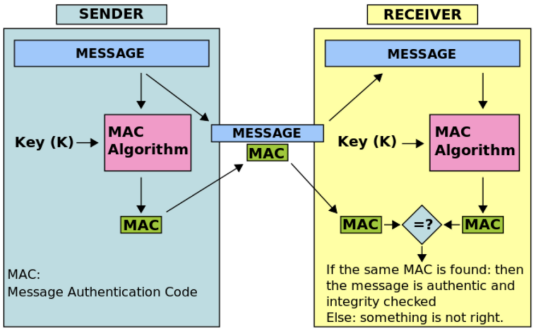
API, atau *Application Programming Interface*, adalah seperangkat definisi dan protokol yang digunakan untuk membangun dan mengintegrasikan perangkat lunak aplikasi. API memungkinkan komunikasi antara dua komponen perangkat lunak yang berbeda. Ini bisa diibaratkan sebagai perantara yang memungkinkan dua aplikasi untuk berkomunikasi satu sama lain.

Arsitektur API berkaitan dengan komunikasi dua arah antara klien dan server. Aplikasi yang mengirimkan permintaan disebut klien dan aplikasi yang mengirimkan respon disebut server. Hal ini memperlihatkan terjadinya transaksi data, yaitu klien, misalnya pengguna website, akan mengirimkan data ke server yang nantinya akan diproses lalu dikembalikan kepada klien.

Terdapat beberapa jenis API yang lazim digunakan saat ini, di antaranya adalah API SOAP, API RPC, API Websocket, dan API REST. Keempat jenis API tersebut memiliki cara kerjanya masing - masing sesuai kebutuhan dari aplikasi yang akan dikembangkan. Akan tetapi, keempatnya memiliki kesamaan yang erat kaitannya dengan keterhubungan antara klien dan server. Aplikasi akan mengirimkan permintaan disebut klien dan aplikasi yang mengirimkan respons disebut sebagai server.

B. Message Authentication Code

MAC, atau *Message authentication Code*, adalah kode kecil berukuran tetap (*fixed*) yang dihasilkan dari pesan dan kunci untuk mengotentikasi pengirim dan memeriksa integritas pesan. MAC diletakan pada pesan sebagai *signature*, digunakan untuk memeriksa integritas pesan dan otentikasi pengirim. Untuk memastikan keaslian dari suatu pesan, MAC yang dikirim harus dipastikan sama dengan MAC yang dihitung oleh penerima.



Gambar II.1 Cara kerja MAC

Seperti dengan fungsi hash, MAC dihasilkan dengan algoritma yang melakukan kompresi pesan yang bersifat *irreversible*. Algoritma dari MAC adalah fungsi *many to one*, artinya adanya kemungkinan yang tinggi pesan memiliki MAC yang sama. Akan tetapi, menemukan dua atau lebih pesan yang memiliki MAC yang sama sangatlah sulit. Hal ini memastikan bahwa MAC masih merupakan algoritma yang dapat dijamin keamanannya.

Terdapat beberapa cara untuk menghasilkan MAC. Algoritma MAC yang dapat digunakan adalah algoritma berbasis *block cipher* dan algoritma berbasis fungsi *hash* satu arah. Pada algoritma MAC berbasis *block cipher*, MAC dibangkitkan dengan mode CBC dan CFB. Nilai *hash*-nya (yang menjadi MAC) adalah hasil enkripsi pada blok terakhir. Algoritma MAC berbasis fungsi *hash* satu arah dapat menggunakan fungsi *hash* seperti MD5 dan SHA. Pesan disambung dengan kunci, lalu dihitung nilai hash dari hasil penggabungan tersebut dengan fungsi *hash*.

C. Digital Signature

Digital signature, atau tanda tangan digital, menerapkan konsep yang sama dengan tanda tangan tradisional. Tanda tangan itu sendiri digunakan untuk otentikasi dokumen cetak. Tanda tangan memiliki karakteristik sebagai berikut.

- Tanda tangan adalah bukti yang otentik
- Tanda tangan tidak dapat dilupakan
- Tanda tangan tidak dapat dipindah untuk digunakan ulang
- Dokumen yang telah ditandatangani tidak dapat diubah
- Tanda tangan tidak dapat disangkal

Fungsi tanda tangan tersebut kemudian diterapkan untuk otentikasi pada data digital seperti dokumen elektronik. Namun, dalam konteks kriptografi, tanda tangan digital tidak sama dengan tanda tangan konvensional yang di-digitasi dengan cara dipindai atau difoto, tetapi nilai kriptografis yang bergantung pada isi pesan dan kunci.

Saat tanda tangan seseorang pada dokumen cetak selalu sama, maka tanda tangan digital dalam ilmu kriptografi selalu berbeda - beda antara satu pesan dengan pesan lainnya, dan/atau antara satu kunci dengan kunci yang lain.

Terdapat beberapa persyaratan pada tanda tangan digital, persyaratan - persyaratan tersebut antara lain adalah sebagai berikut.

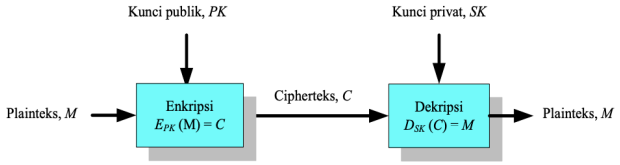
- Tanda tangan harus berupa rangkaian bit yang bergantung pada pesan yang ditandatangani
- Tanda tangan harus menggunakan informasi unik, berupa kunci, dari pengirim untuk mencegah pemalsuan dan penyangkalan
- Membangkitkan tanda tangan digital harus relatif mudah untuk dilakukan
- Mengenali dan memverifikasi tanda tangan digital harus relatif mudah untuk dilakukan
- Secara komputasi, hampir tidak mungkin untuk memalsukan tanda tangan digital, baik dengan merekonstruksi pesan baru, atau merekonstruksi tanda tangan curang untuk pesan yang sudah diberikan
- Menyimpan salinan tanda tangan digital ke dalam tampungan harus mudah dilakukan secara praktek

Terdapat dua proses dalam melakukan tanda tangan digital, pertama adalah menandatangani pesan yang disebut proses *signing*, dan memverifikasi pesan yang disebut proses *verification*. Kedua proses tersebut perlu diimplementasi dengan baik agar tanda tangan digital dapat berfungsi dengan baik sesuai dengan tujuan utamanya.

Pada proses *signing*, atau menandatangani pesan, terdapat dua cara yang dilakukan. Cara pertama adalah mengenkripsi pesan yang dikhususkan untuk pesan yang bersifat rahasia. Cara kedua adalah dengan menggunakan kombinasi fungsi *hash* dan kriptografi kunci - publik yang digunakan untuk pesan yang tidak perlu dijamin kerahasiaannya.

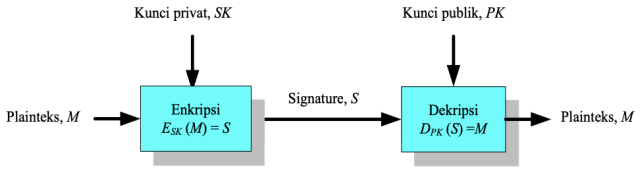
Terdapat beberapa algoritma untuk penandatanganan dengan cara mengenkripsi pesan. Cara pertama adalah enkripsi menggunakan algoritma kriptografi kunci - simetri. Pesan yang dienkripsi dengan algoritma simetri sudah memberikan solusi untuk otentikasi pengirim karena kunci simetri hanya diketahui oleh pengirim dan penerima. Namun cara ini tidak menyediakan cara untuk melakukan nir - penyangkalan. Oleh karena itu, penandatangan pesan dengan kriptografi kunci - simetri tidak dapat dilakukan jika hanya terdapat dua aktor saja. Untuk menangani masalah penyangkalan tersebut, diperlukan pihak ketiga yang dipercaya oleh pengirim/penerima. Pihak ketiga ini disebut penengah yang memiliki otoritas *arbitrase* yang dipercaya oleh pengirim dan penerima.

Cara kedua adalah enkripsi pesan menggunakan algoritma kunci - publik. Mengenkripsi pesan dengan menggunakan kunci publik penerima pesan dan mendekripsinya dengan kunci privat penerima pesan merupakan proses yang umum terjadi dalam kriptografi kunci - publik.



Gambar II.2 Kriptografi kunci - publik

Akan tetapi, cara ini tidak dapat mengotentikasi pengirim pesan karena kunci publik diketahui oleh siapapun. Oleh karena itu, agar kriptografi kunci - publik dapat berfungsi sebagai tanda tangan digital, maka prosesnya dibalik menjadi pesan dienkripsi dengan kunci privat pengirim dan didekripsi dengan kunci publik pengirim.



Gambar II.3 Pembalikkan proses kriptografi kunci - publik

Dengan diterapkannya pembalikkan proses ini, kerahasiaan pesan dan otentikasi si pengirim pesan dapat dicapai. Penerima pesan dapat mengotentikasi pengirim pesan karena kunci publik dan kunci privat adalah berpasangan. Penerima pesan juga dapat melakukan nir - penyangkalan jika pengirim menyangkal mengirim pesan. Ide ini ditemukan oleh Diffie dan Hellman untuk merealisasikan tanda tangan digital menggunakan algoritma kunci - publik.

Cara kedua untuk menandatangani pesan yang sudah disebutkan sebelumnya adalah dengan menggunakan kriptografi kunci - publik dan fungsi *hash*. Cara ini cocok digunakan untuk kasus disaat otentikasi diperlukan tetapi kerahasiaan pesan tidak diperlukan. Kasus lebih jelasnya adalah pesan tidak perlu dienkripsi karena tidak rahasia, sebab yang dibutuhkan hanya keotentikan pesan dan pengirimnya saja. Salah satu algoritma *signature* yang paling banyak digunakan untuk mengenkripsi pesan adalah algoritma RSA dan *ElGamal Signature*. Pada RSA, algoritma enkripsi dan dekripsi identik sehingga proses *signature* dan verifikasi juga identik. Pada *ElGamal Signature*, algoritma enkripsi dengan algoritma *signature* tidak sama.

III. IMPLEMENTASI

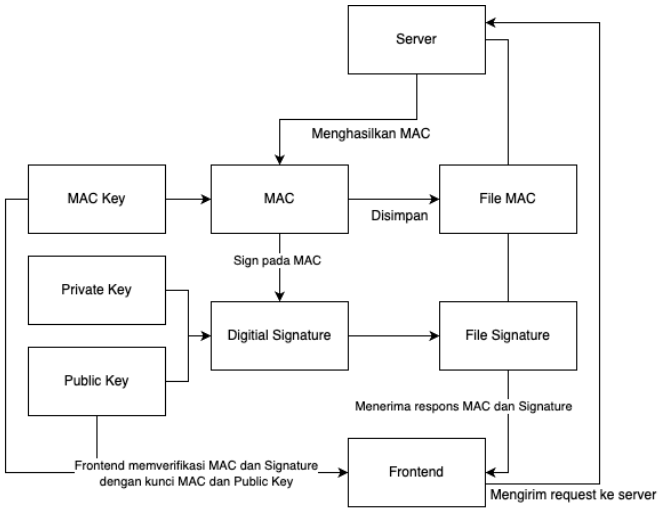
Implementasi pada topik ini akan dilakukan dengan pertama mengembangkan API dalam bentuk REST terlebih dahulu serta dengan mengembangkan aplikasi web sebagai klien yang akan mengirimkan permintaan kepada server, dalam kasus ini adalah API nya, lalu akan dikirimkan suatu respons sederhana kembali kepada klien. Implementasi pada sisi server dan klien akan dibuat sesederhana mungkin untuk difokuskan pada sisi verifikasi/validasi pada MAC dan implementasi *digital signature* setiap melakukan perubahan pada API.

Implementasi di sisi *frontend web* yang berperan sebagai klien akan menggunakan bahasa pemrograman *Typescript* serta menggunakan pustaka *React* untuk mempermudah pengembangannya. API akan dikembangkan dengan menggunakan bahasa pemrograman *Python* dengan *framework Flask* untuk menerima permintaan/*request* dan mengirimkan respon balik kepada klien. MAC akan disimpan pada suatu file teks yang nantinya MAC tersebut akan divalidasi di sisi *frontend* untuk memverifikasi apakah integritas API masih terjaga.

Implementasi pada makalah ini akan mensimulasikan skenario kasus nyata namun dalam lingkungan pengembangan secara lokal. Perubahan yang terotentikasi pada API akan dihasilkan sebuah MAC baru untuk menjaga integritas dari cara kerja API. Jika disimulasikan pada kasus nyata, MAC akan dihasilkan pada proses CI/CD, yaitu pada saat API akan siap diluncurkan untuk digunakan oleh klien.

A. Rancangan Solusi

Berikut ini merupakan rancangan solusi dalam bentuk diagram untuk memperlihatkan alur kerja dari sistem yang akan dikembangkan nantinya. Sistem akan mencakup dua jenis sub sistem, yaitu *backend* sebagai server dari sistem, dan *frontend* sebagai tampilan antarmuka sebagai klien dari sistem yang akan mengirimkan permintaan kepada server dan menerima respons balik dari server.



Gambar III.1 Flow dari Rancangan Sistem

Pada server, akan dihasilkan pasangan kunci publik dan kunci privat menggunakan RSA yang akan digunakan untuk melakukan tanda tangan digital pada MAC. MAC akan dihasilkan menggunakan kunci rahasia untuk mengenkripsi file API pada server. Lalu, MAC yang diperoleh pada langkah sebelumnya akan dilakukan penandatanganan menggunakan kunci privat yang sudah dihasilkan sebelumnya.

Kedua MAC dan tanda tangan digital yang sudah dihasilkan akan disimpan pada suatu file di server yang nantinya kedua file tersebut akan dikirimkan kepada klien untuk melakukan verifikasi API pada sisi MAC dan tanda tangan digital. Untuk memverifikasi MAC pada sisi klien, klien akan menerima konten dari API yang nantinya akan dihasilkan sebuah MAC dengan kunci yang sama digunakan untuk menghasilkan MAC pada sisi server. MAC yang dihasilkan pada klien akan dikomparasi dengan MAC yang diperoleh dari server. Jika keduanya sama, maka verifikasi integritas pada konten API berhasil.

Walaupun kedua MAC sama, tetapi diperlukan verifikasi berdasarkan tanda tangan digital sebagai lapisan kedua pengamanan integritas dari suatu API. Tanda tangan digital yang sudah dihasilkan di server akan dikirimkan ke klien untuk nantinya diverifikasi. Klien juga akan menerima kunci publik yang sudah dihasilkan untuk nantinya digunakan dalam proses verifikasi tanda tangan digital.

Hasil perbandingan dari MAC yang dihasilkan di klien dan di server akan diperiksa bersamaan kebenarannya dengan verifikasi dari tanda tangan digital yang diverifikasi di sisi klien. Jika salah satu aspek dari kedua kriptografi tersebut bernilai salah/tidak benar, maka hal tersebut menunjukkan adanya ketidakcocokan pada API yang ada dengan API sesungguhnya. Jika hal tersebut terjadi, maka dapat diindikasikan terjadinya modifikasi secara tidak terotentikasi pada API, sehingga API tidak dapat digunakan pada sisi klien.

B. Batasan dan Asumsi

Berdasarkan rancangan yang telah diajukan, hasil implementasi pada sisi yang sudah dikembangkan memiliki batasan serta asumsi sebagai berikut

- Bahasa pemrograman yang digunakan pada sisi server adalah Python dengan *framework* Flask untuk menyediakan API kepada klien
- Bahasa pemrograman yang digunakan pada sisi klien adalah Typescript dengan pustaka React
- Kunci publik dan kunci privat dihasilkan terlebih dahulu sebelum menerapkan tanda tangan digital pada MAC dan dikirimkan ke klien
- Penandatanganan MAC dilakukan pada file server menggunakan kunci yang didefinisikan oleh pemilik/pembuat API
- Kunci publik, kunci privat, MAC, dan tanda tangan digital disimpan secara lokal pada direktori server.
- Klien akan menerima kunci publik, MAC, dan tanda tangan digital untuk selanjutnya dilakukan verifikasi
- Kunci untuk mengenkripsi API menjadi MAC sudah didefinisikan secara langsung pada server dan klien dalam bentuk string sederhana

Untuk kebutuhan prototype, skala sistem yang dikembangkan tidak besar, dengan ukuran yang cukup kecil, khususnya pada sisi server untuk memperlihatkan bagaimana proses melakukan enkripsi pada API serta melakukan verifikasi MAC dan tanda tangan digital dari API tersebut pada sisi klien/*frontend*.

IV. Hasil dan Pembahasan

Pada bagian ini, dilakukan pengujian terhadap hasil implementasi beserta analisis terhadap hasil pengujian yang didapatkan.

A. Inisialisasi Pengujian

Pada tahap pengujian, kasus tidak bersifat adanya kasus uji berupa masukkan yang berbeda, akan tetapi pengujian dilakukan dengan melakukan modifikasi pada beberapa aspek yang terdapat pada sisi server dan klien. Berikut adalah aspek - aspek yang digunakan dalam proses pengujian sistem

TABLE I. Aspek Pengujian Sistem

Kode Uji	Aspek Pengujian
kasus-uji-1	Kasus berhasil dengan kunci privat, kunci publik, kunci MAC, dan tidak adanya modifikasi pada API
kasus-uji-2	Seluruh kunci sesuai dengan adanya modifikasi pada API

kasus-uji-3	Tidak ada modifikasi pada API, tetapi kunci MAC tidak sesuai antara server dengan klien
kasus-uji-4	Tidak ada modifikasi pada API, tetapi dengan kunci privat dan kunci publik yang tidak sesuai dengan kunci privat yang digunakan untuk melakukan tanda tangan digital
kasus-uji-5	Modifikasi pada API, dengan kunci MAC yang berbeda antara server dan klien
kasus-uji-6	Modifikasi pada API, dengan kunci privat dan kunci publik yang tidak sesuai dengan kunci privat dan kunci publik yang digunakan untuk tanda tangan digital

B. Pengujian dan Analisis

Berdasarkan kasus uji yang sudah didefinisikan pada sub bab A, yaitu Inisialisasi Pengujian, akan dilakukan analisis pada hasil keluaran yang diberikan oleh klien terkait proses verifikasi dari integritasi API.

1. kasus-uji-1

   Kasus uji 1 akan dilakukan pengujian dengan semua kondisi sesuai dan mengeluarkan keluaran bahwa integritas API masih terjaga dan verifikasi berhasil. Berikut adalah data serta hasil dari kasus uji pertama.

TABLE II. INISIASI KUNCI PADA KASUS UJI 1

Kunci MAC	'jason_kanggara'
Kunci publik	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w 0BAQEFAAOCAQ8AM IIBCgKCAQEAowNWs sWyeo3f5JN1+y3S InWwJClKYvrlVzbFgw S1CdyblPp3lFClxYGU9 x4pkxRzeR4VhLjG7rH DcFPHSn5U 4Grlpo7V0Sn10iucAdlD 43B7zB79Sy2Aq8NzLs 6aaD4qhK6DSU42hfCG pDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8RdOe6Op6a 32Qos5o7A72ABx0Io18 neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp430Tj4QP +IFvbBT8GlP8g6+nMK dbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QIA4MF914 5bt7LDb8SUSl1+AX5U fBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----
Kunci Privat	-----BEGIN RSA PRIVATE KEY----- MIIEowIBAAKCAQEA owNWssWyeo3f5JN1+y 3SInWwJClKYvrlVzbFg wS1CdyblPp3 lFClxYGU9x4pkxRzeR 4VhLjG7rHDcFPHSn5 U4Grlpo7V0Sn10iucAdl D43B7zB79 Sy2Aq8NzLs6aaD4qhK 6DSU42hfCGpDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8R dOe6Op6a32Qos5o7A7 2ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp 430Tj4QP+IFvbBT8GlP 8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QI A4MF9145bt7LDb8SUS l1+AX5UfBOfkNL5gna EQIDAQABAoIBAAO1 4EcOPkaRMPKm t4yAoBlyTN1sP/sS6urN jWwIKpjkvBKngNahA XHCPZeoyKICA4/smY LsZx8PPMhm jreHGJ3jFUccM96G8j7 Ob08qZYX+t9sxTrf9N wZtlpu5m15nQ09tkgGU WehLGlaH xJdV1hpgiN9f99UAWS Jj9j1PXBWrMYeHdweP ikA1tRqXAsUcjwhEA0 V0S9RwD7ck aStd2jURY/wOEHuEzq +bJ3MOsxcenI1l9/G210

	5miadEhDH3qtZtduvTW/KyIL9L wY7+Q0Bzn/Z3dmQZLIhmPavIZ4Y6ARcflBATBL5tpLMu8aF202XzKvneiXxbHlvI ptyibWECgYEAyJ3WEsoKQ33ZodR/HKX5Zh/ilO2/1m1nOVHjdbCpiuRyYz77I61S P8yj8BDz5fawCIXNJ0h/RkKESVShbF6Jg2XbC/SG21jtQRCkcZI0GR1cmcLzqw/g eYJ75t2fL3b4vYO3zvKnOzIIqLwU6PLqwtyEAPwD6aV3yXswvhVV3AcCgYEA0APznnUEsUQNjovKURbN+a34+QiwW0WXqery/TgDmjYldFGLatrWJzf0dvWAsJIDalxbfPZcQ+9fC8DWQ9bhmOIXpOt8CGan2ItVmGFJKHLe7ZKoWG2np3SdEkEbyM68OqXVSspEIRtTYYMG7dd6moL8zCsowBklfyjcGWbEwycCgYEAqhNlUTiHyv/+aPvFZ3wuGy4c4TVN1XOadzW9KCFFGsLhVtImBvEBMaVDtloo62RfiFomuo7zl5T8oU5mK4Fa A1NuiIwdaAfT271ILCBOmpwRzSlBelxTaa/9apOoSI0MA16PjZqZu82HGcWZ1INcEybvogYFKydkjiMqB3v+9jECgYAxFT2e3i3FRjGMuR98Es13ZL2ixZIXd0xaI75CUb1H3g7oa3ViOVDe4C0nBLbg8xMrPa+4xMjwnsLmmxHz3/hLgs3hdE9IDDD+v6Hq2snQaYqAtGJUu0scEEAf1o0fzOhNTmQNybxZs4DxOWGhLX8Yhl/wLR7odQaMArAsrjsvcQKBgBvBwjV8HnXiSJwKgrqXh0EeZe0KqS4+yinzrOGnfLd2OZnDHv+1SwNnX2dxgIwiizrpgheXTPRNJmLVNjOej5PMy6uV	

	KMVWzXY55FuhGT6GNRAfdwxc2W70avWKn9QNESf/Q3eFsR4SC4jlqcX9J6d7qFf41cAQj5gTsVYa4OAO -----END RSA PRIVATE KEY-----

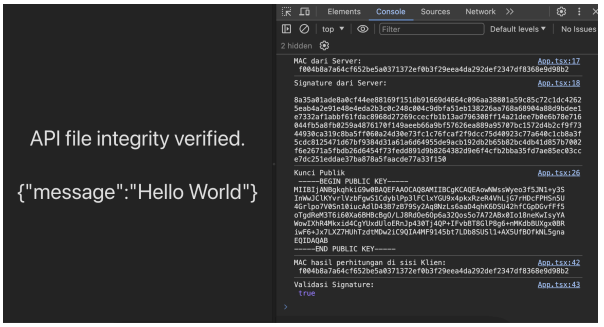
Inisiasi API	`@app.route('/api/hello')`
	`def hello():`
	`  return jsonify({`
	`  'message': 'Hello World'`
	`  })`

Berikut adalah isi dari API yang digunakan

```
@app.route('/api/hello')
def hello():
  return jsonify({
    'message': 'Hello World'
  })
```

Gambar IV.1 API yang digunakan untuk menghasilkan MAC

Setelah dilakukan verifikasi di sisi klien, berikut adalah hasil yang diperoleh.



Gambar IV.2 Hasil verifikasi di sisi klien

Berikut adalah komponen - komponen yang dilakukan dalam proses verifikasi API di sisi klien

TABLE III. DATA YANG DIGUNAKAN DALAM PROSES VERIFIKASI

Kunci MAC	'jason_kanggara'
MAC dari server	f004b8a7a64cf652be5a0371372ef0b3f29eea4da292def2347df8368e9d98b2
MAC perhitungan dari klien	f004b8a7a64cf652be5a0371372ef0b3f29eea4da292def2347df8368e9d98b

	2
Signature dari server	8a35a01ade8a0cf44ee88 169f151db91669d4664c 096aa38801a59c85c72c 1dc42625eab4a2e91e48e 4eda2b3c0c248c004c9d bfa51eb138226aa768a68 904a88d9bdee1e7332af1 abbf61fdac8968d27269c cecfb1b13ad796308ff14 a21dee7b0e6b78e71604 4fb5a8fb0259a4876170f 149aeeb66a9bf57626ea8 89a95707bc1572d4b2cf 9f7344930ca319c8ba5ff 060a24d30e73fc1c76fca f2f9dcc75d40923c77a64 0c1cb8a3f5cdc8125471d 67bf9384d31a61a6d649 55de9acb192db2b65b82 bc4db41d857b7002f6e2 671a5fbdb26d6454f73fe dd891d9b8264382d9e6f 4cfb2bba35fd7ae85ec03 cce7dc251eddae37ba878 a5faacde77a33f150
Kunci publik yang digunakan untuk validasi tanda tangan	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w 0BAQEFAAOCAQ8AM IIBCgKCAQEAowNWs sWyeo3f5JN1+y3S InWwJClKYvrlVzbFgw S1CdyblPp3lFClxYGU9 x4pkxRzeR4VhLjG7rH DcFPHSn5U 4Grlpo7V0Sn10iucAdlD 43B7zB79Sy2Aq8NzLs 6aaD4qhK6DSU42hfCG pDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8RdOe6Op6a 32Qos5o7A72ABx0Io18 neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp430Tj4QP +IFvbBT8GlP8g6+nMK dbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QIA4MF914 5bt7LDb8SUSl1+AX5U fBOfkNL5gna EQIDAQAB -----END PUBLIC

	KEY-----
Validitas Signature	True

2. kasus-uji-2

Kasus uji 2 akan dilakukan pengujian dengan semua kunci sesuai dengan adanya perubahan pada isi API.

TABLE IV. INISIASI KUNCI PADA KASUS UJI 2

Kunci MAC	'jason_kanggara'
Kunci publik	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w 0BAQEFAAOCAQ8AM IIBCgKCAQEAowNWs sWyeo3f5JN1+y3S InWwJClKYvrlVzbFgw S1CdyblPp3lFClxYGU9 x4pkxRzeR4VhLjG7rH DcFPHSn5U 4Grlpo7V0Sn10iucAdlD 43B7zB79Sy2Aq8NzLs 6aaD4qhK6DSU42hfCG pDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8RdOe6Op6a 32Qos5o7A72ABx0Io18 neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp430Tj4QP +IFvbBT8GlP8g6+nMK dbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QIA4MF914 5bt7LDb8SUSl1+AX5U fBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----
Kunci Privat	-----BEGIN RSA PRIVATE KEY----- MIIEowIBAAKCAQEA owNWssWyeo3f5JN1+y 3SInWwJClKYvrlVzbFg wS1CdyblPp3 lFClxYGU9x4pkxRzeR 4VhLjG7rHDcFPHSn5 U4Grlpo7V0Sn10iucAdl D43B7zB79 Sy2Aq8NzLs6aaD4qhK 6DSU42hfCGpDGvfFf5

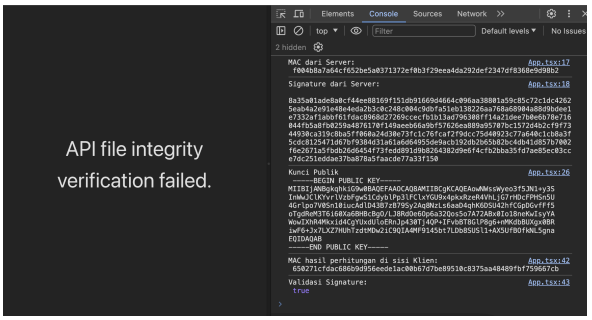
	oTgdReM3T6i60Xa6BH BcBgO/LJ8R dOe6Op6a32Qos5o7A7 2ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp 430Tj4QP+IFvbBT8GlP 8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QI A4MF9145bt7LDb8SUS l1+AX5UfBOfkNL5gna EQIDAQABAoIBAAO1 4EcOPkaRMPKm t4yAoBlyTN1sP/sS6urN jWwIKpjkvBKngNahA XHCPZeoyKICA4/smY LsZx8PPMhm jreHGJ3jFUccM96G8j7 Ob08qZYX+t9sxTrf9N wZtlpu5m15nQ09tkgGU WehLGlaH xJdV1hpgiN9f99UAWS Jj9j1PXBWrMYeHdweP ikA1tRqXAsUcjwhEA0 V0S9RwD7ck aStd2jURY/wOEHuEzq +bJ3MOsxcenI1l9/G210 5miadEhDH3qtZtduvT W/KyIL9L wY7+Q0Bzn/Z3dmQZL IhmPavIZ4Y6ARcflBAT BL5tpLMu8aF202XzKv neiXxbHlvI ptyibWECgYEAyJ3WEs oKQ33ZodR/HKX5Zh/il O2/1m1nOVHjdbCpiuR yYz77I61S P8yj8BDz5fawCIXNJ0h /RkKESVShbF6Jg2XbC /SG21jtQRCkcZI0GR1c mcLzqw/g eYJ75t2fL3b4vYO3zvK nOzIIqLwU6PLqwtyEA PwD6aV3yXswvhVV3 AcCgYEA0APz nnUEsUQNjovKURbN+ a34+QiwW0WXqery/Tg DmjYldFGLatrWJzf0dv WAsJIDalxb fPZcQ+9fC8DWQ9bhm OIXpOt8CGan2ItVmGF JKHLe7ZKoWG2np3Sd EkEbyM68OqXV SspEIRtTYYMG7dd6m oL8zCsowBklfyjcGWbE

	wycCgYEAqhNlUTiHy v/+aPvFZ3wu Gy4c4TVN1XOadzW9 KCFFGsLhVtImBvEB MaVDtloo62RfiFomuo7 zl5T8oU5mK4Fa A1NuiIwdaAfT271ILC BOmpwRzSlBelxTaa/9a pOoSI0MA16PjZqZu82 HGcWZ1INc EybvogYFKydkjiMqB3 v+9jECgYAxFT2e3i3FR jGMuR98Es13ZL2ixZI Xd0xaI75C Ub1H3g7oa3ViOVDe4C 0nBLbg8xMrPa+4xMjw nsLmmxHz3/hLgs3hdE9 IDDD+v6Hq 2snQaYqAtGJUu0scEE Af1o0fzOhNTmQNybx Zs4DxOWGhLX8Yhl/w LR7odQaMArAs rjsvcQKBgBvBwjV8Hn XiSJwKgrqXh0EeZe0K qS4+yinzrOGnfLd2OZn DHv+1SwNn X2dxgIwiizrpgheXTPR NJmLVNjOej5PMy6uV KMVWzXY55FuhGT6 GNRAfdwxc2W70 avWKn9QNESf/Q3eFsR 4SC4jlqcX9J6d7qFf41c AQj5gTsVYa4OAO -----END RSA PRIVATE KEY-----
Inisiasi API	

Berikut adalah isi dari API yang digunakan

Gambar IV.3 API yang digunakan untuk menghasilkan MAC

Setelah dilakukan verifikasi di sisi klien, berikut adalah hasil yang diperoleh.



Gambar IV.4 Hasil verifikasi di sisi klien

Berikut adalah komponen - komponen yang dilakukan dalam proses verifikasi API di sisi klien

TABLE V. DATA YANG DIGUNAKAN DALAM PROSES VERIFIKASI

Kunci MAC	'jason_kanggara'
MAC dari server	f004b8a7a64cf652be5a0371372ef0b3f29eea4da292def2347df8368e9d98b2
MAC perhitungan dari klien	650271cfdac686b9d956eede1ac00b67d7be89510c8375aa48489fbf759667cb
Signature dari server	8a35a01ade8a0cf44ee88169f151db91669d4664c096aa38801a59c85c72c1dc42625eab4a2e91e48e4eda2b3c0c248c004c9dbfa51eb138226aa768a68904a88d9bdee1e7332af1abbf61fdac8968d27269ccecfb1b13ad796308ff14a21dee7b0e6b78e716044fb5a8fb0259a4876170f149aeeb66a9bf57626ea889a95707bc1572d4b2cf9f7344930ca319c8ba5ff060a24d30e73fc1c76fcaf2f9dcc75d40923c77a640c1cb8a3f5cdc8125471d67bf9384d31a61a6d64955de9acb192db2b65b82bc4db41d857b7002f6e2671a5fbdb26d6454f73fedd891d9b8264382d9e6f4cfb2bba35fd7ae85ec03cce7dc251eddae37ba878a5faacde77a33f150
Kunci publik yang digunakan untuk validasi	-----BEGIN PUBLIC KEY-----
tanda tangan	MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5U4Grlpo7V0Sn10iucAdlD43B7zB79Sy2Aq8NzLs6aaD4qhK6DSU42hfCGpDGvfFf5oTgdReM3T6i60Xa6BHBcBgO/LJ8RdOe6Op6a32Qos5o7A72ABx0Io18neKwIsyYAWowIXhR4Mkxid4CgYUxdUloERnJp430Tj4QP+IFvbBT8GlP8g6+nMKdbBUXgx0BRiwF6+Jx7LXZ7HUhTzdtMDw2iC9QIA4MF9145bt7LDb8SUSl1+AX5UfBOfkNL5gnaEQIDAQAB-----END PUBLIC KEY-----
Validitas Signature	True

3. kasus-uji-3

Kasus uji 3 akan dilakukan pengujian kunci MAC yang berbeda antara klien dan server.

TABLE VI. INISIASI KUNCI PADA KASUS UJI 1

Kunci MAC	'jason_kanggara'
Kunci publik	-----BEGIN PUBLIC KEY-----MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5U4Grlpo7V0Sn10iucAdlD43B7zB79Sy2Aq8NzLs6aaD4qhK6DSU42hfCGpDGvfFf5oTgdReM3T6i60Xa6BHBcBgO/LJ8RdOe6Op6a32Qos5o7A72ABx0Io18

	neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp430Tj4QP +IFvbBT8GlP8g6+nMK dbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QIA4MF914 5bt7LDb8SUSl1+AX5U fBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----
Kunci Privat	-----BEGIN RSA PRIVATE KEY----- MIIEowIBAAKCAQEA owNWssWyeo3f5JN1+y 3SInWwJClKYvrlVzbFg wS1CdyblPp3 lFClxYGU9x4pkxRzeR 4VhLjG7rHDcFPHSn5 U4Grlpo7V0Sn10iucAdl D43B7zB79 Sy2Aq8NzLs6aaD4qhK 6DSU42hfCGpDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8R dOe6Op6a32Qos5o7A7 2ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp 430Tj4QP+IFvbBT8GlP 8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QI A4MF9145bt7LDb8SUS l1+AX5UfBOfkNL5gna EQIDAQABAoIBAAO1 4EcOPkaRMPKm t4yAoBlyTN1sP/sS6urN jWwIKpjkvBKngNahA XHCPZeoyKICA4/smY LsZx8PPMhm jreHGJ3jFUccM96G8j7 Ob08qZYX+t9sxTrf9N wZtlpu5m15nQ09tkgGU WehLGlaH xJdV1hpgiN9f99UAWS Jj9j1PXBWrMYeHdweP ikA1tRqXAsUcjwhEA0 V0S9RwD7ck aStd2jURY/wOEHuEzq +bJ3MOsxcenI1l9/G210 5miadEhDH3qtZtduvT W/KyIL9L wY7+Q0Bzn/Z3dmQZL

	IhmPavIZ4Y6ARcflBAT BL5tpLMu8aF202XzKv neiXxbHlvI ptyibWECgYEAyJ3WEs oKQ33ZodR/HKX5Zh/il O2/1m1nOVHjdbCpiuR yYz77I61S P8yj8BDz5fawCIXNJ0h /RkKESVShbF6Jg2XbC /SG21jtQRCkcZI0GR1c mcLzqw/g eYJ75t2fL3b4vYO3zvK nOzIIqLwU6PLqwtyEA PwD6aV3yXswvhVV3 AcCgYEA0APz nnUEsUQNjovKURbN+ a34+QiwW0WXqery/Tg DmjYldFGLatrWJzf0dv WAsJIDalxb fPZcQ+9fC8DWQ9bhm OIXpOt8CGan2ItVmGF JKHLe7ZKoWG2np3Sd EkEbyM68OqXV SspEIRtTYYMG7dd6m oL8zCsowBklfyjcGWbE wycCgYEAqhNlUTiHy v/+aPvFZ3wu Gy4c4TVN1XOadzW9 KCFFGsLhVtImBvEB MaVDtloo62RfiFomuo7 zl5T8oU5mK4Fa A1NuiIwdaAfT271ILC BOmpwRzSlBelxTaa/9a pOoSI0MA16PjZqZu82 HGcWZ1INc EybvogYFKydkjiMqB3 v+9jECgYAxFT2e3i3FR jGMuR98Es13ZL2ixZI Xd0xaI75C Ub1H3g7oa3ViOVDe4C 0nBLbg8xMrPa+4xMjw nsLmmxHz3/hLgs3hdE9 IDDD+v6Hq 2snQaYqAtGJUu0scEE Af1o0fzOhNTmQNybx Zs4DxOWGhLX8Yhl/w LR7odQaMArAs rjsvcQKBgBvBwjV8Hn XiSJwKgrqXh0EeZe0K qS4+yinzrOGnfLd2OZn DHv+1SwNn X2dxgIwiizrpgheXTPR NJmLVNjOej5PMy6uV KMVWzXY55FuhGT6 GNRAfdwxc2W70 avWKn9QNESf/Q3eFsR

	4SC4jlqcX9J6d7qFf41c AQj5gTsVYa4OAO -----END RSA PRIVATE KEY-----

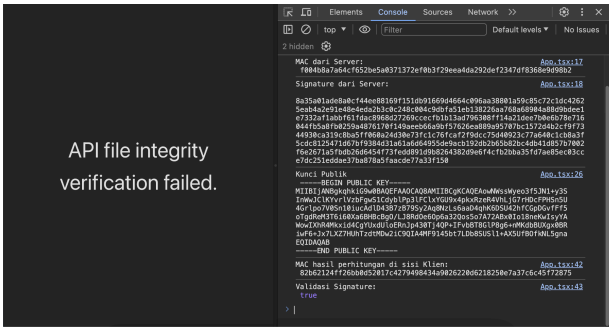
Inisiasi API	`@app.route('/api/hello')`
	`def hello():`
	`  return jsonify({`
	`  'message': 'Hello World'`
	`  })`

Berikut adalah isi dari API yang digunakan

```
@app.route('/api/hello')
def hello():
  return jsonify({
    'message': 'Hello World'
  })
```

Gambar IV.5 API yang digunakan untuk menghasilkan MAC

Setelah dilakukan verifikasi di sisi klien, berikut adalah hasil yang diperoleh.



Gambar IV.6 Hasil verifikasi di sisi klien

Berikut adalah komponen - komponen yang dilakukan dalam proses verifikasi API di sisi klien

TABLE VII. DATA YANG DIGUNAKAN DALAM PROSES VERIFIKASI

Kunci MAC	'kriptografi'
MAC dari server	f004b8a7a64cf652be5a0 371372ef0b3f29eea4da2 92def2347df8368e9d98b 2
MAC perhitungan dari klien	82b62124ff26bb0d5201 7c4279498434a9026220 d6218250e7a37c6c45f72 875
Signature dari server	8a35a01ade8a0cf44ee88 169f151db91669d4664c 096aa38801a59c85c72c 1dc42625eab4a2e91e48e 4eda2b3c0c248c004c9d bfa51eb138226aa768a68 904a88d9bdee1e7332af1 abbf61fdac8968d27269c cecfb1b13ad796308ff14 a21dee7b0e6b78e71604 4fb5a8fb0259a4876170f 149aeeb66a9bf57626ea8 89a95707bc1572d4b2cf 9f7344930ca319c8ba5ff 060a24d30e73fc1c76fca f2f9dcc75d40923c77a64 0c1cb8a3f5cdc8125471d 67bf9384d31a61a6d649 55de9acb192db2b65b82 bc4db41d857b7002f6e2 671a5fbdb26d6454f73fe dd891d9b8264382d9e6f 4cfb2bba35fd7ae85ec03 cce7dc251eddae37ba878 a5faacde77a33f150
Kunci publik yang digunakan untuk validasi tanda tangan	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w 0BAQEFAAOCAQ8AM IIBCgKCAQEAowNWs sWyeo3f5JN1+y3S InWwJClKYvrlVzbFgw S1CdyblPp3lFClxYGU9 x4pkxRzeR4VhLjG7rH DcFPHSn5U 4Grlpo7V0Sn10iucAdlD 43B7zB79Sy2Aq8NzLs 6aaD4qhK6DSU42hfCG pDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8RdOe6Op6a 32Qos5o7A72ABx0Io18 neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp430Tj4QP +IFvbBT8GlP8g6+nMK dbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QIA4MF914 5bt7LDb8SUSl1+AX5U fBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----

Validitas Signature	True

4. kasus-uji-4

Kasus uji 4 akan dilakukan pengujian dengan adanya kunci privat dan kunci publik yang berbeda saat penandatanganan MAC dan pada saat verifikasi tanda tangan di sisi klien.

TABLE VIII. INISIASI KUNCI PADA KASUS UJI 1

Kunci MAC	'jason_kanggara'
Kunci publik	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w 0BAQEFAAOCAQ8AM IIBCgKCAQEAowNWs sWyeo3f5JN1+y3S InWwJClKYvrlVzbFgw S1CdyblPp3lFClxYGU9 x4pkxRzeR4VhLjG7rH DcFPHSn5U 4Grlpo7V0Sn10iucAdlD 43B7zB79Sy2Aq8NzLs 6aaD4qhK6DSU42hfCG pDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8RdOe6Op6a 32Qos5o7A72ABx0Io18 neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp430Tj4QP +IFvbBT8GlP8g6+nMK dbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QIA4MF914 5bt7LDb8SUSl1+AX5U fBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----
Kunci Privat	-----BEGIN RSA PRIVATE KEY----- MIIEowIBAAKCAQEA owNWssWyeo3f5JN1+y 3SInWwJClKYvrlVzbFg wS1CdyblPp3 lFClxYGU9x4pkxRzeR 4VhLjG7rHDcFPHSn5 U4Grlpo7V0Sn10iucAdl D43B7zB79 Sy2Aq8NzLs6aaD4qhK 6DSU42hfCGpDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8R dOe6Op6a32Qos5o7A7 2ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp 430Tj4QP+IFvbBT8GlP 8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QI A4MF9145bt7LDb8SUS l1+AX5UfBOfkNL5gna EQIDAQABAoIBAAO1 4EcOPkaRMPKm t4yAoBlyTN1sP/sS6urN jWwIKpjkvBKngNahA XHCPZeoyKICA4/smY LsZx8PPMhm jreHGJ3jFUccM96G8j7 Ob08qZYX+t9sxTrf9N wZtlpu5m15nQ09tkgGU WehLGlaH xJdV1hpgiN9f99UAWS Jj9j1PXBWrMYeHdweP ikA1tRqXAsUcjwhEA0 V0S9RwD7ck aStd2jURY/wOEHuEzq +bJ3MOsxcenI1l9/G210 5miadEhDH3qtZtduvT W/KyIL9L wY7+Q0Bzn/Z3dmQZL IhmPavIZ4Y6ARcflBAT BL5tpLMu8aF202XzKv neiXxbHlvI ptyibWECgYEAyJ3WEs oKQ33ZodR/HKX5Zh/il O2/1m1nOVHjdbCpiuR yYz77I61S P8yj8BDz5fawCIXNJ0h /RkKESVShbF6Jg2XbC /SG21jtQRCkcZI0GR1c mcLzqw/g eYJ75t2fL3b4vYO3zvK nOzIIqLwU6PLqwtyEA PwD6aV3yXswvhVV3 AcCgYEA0APz nnUEsUQNjovKURbN+ a34+QiwW0WXqery/Tg DmjYldFGLatrWJzf0dv WAsJIDalxb fPZcQ+9fC8DWQ9bhm OIXpOt8CGan2ItVmGF JKHLe7ZKoWG2np3Sd EkEbyM68OqXV SspEIRtTYYMG7dd6m oL8zCsowBklfyjcGWbE wycCgYEAqhNlUTiHy

	v/+aPvFZ3wu Gy4c4TVN1XOadzW9 KCFFGsLhVtImBvEB MaVDtloo62RfiFomuo7 zl5T8oU5mK4Fa A1NuiIwdaAfT271ILC BOmpwRzSlBelxTaa/9a pOoSI0MA16PjZqZu82 HGcWZ1INc EybvogYFKydkjiMqB3 v+9jECgYAxFT2e3i3FR jGMuR98Es13ZL2ixZI Xd0xaI75C Ub1H3g7oa3ViOVDe4C 0nBLbg8xMrPa+4xMjw nsLmmxHz3/hLgs3hdE9 IDDD+v6Hq 2snQaYqAtGJUu0scEE Af1o0fzOhNTmQNybx Zs4DxOWGhLX8Yhl/w LR7odQaMArAs rjsvcQKBgBvBwjV8Hn XiSJwKgrqXh0EeZe0K qS4+yinzrOGnfLd2OZn DHv+1SwNn X2dxgIwiizrpgheXTPR NJmLVNjOej5PMy6uV KMVWzXY55FuhGT6 GNRAfdwxc2W70 avWKn9QNESf/Q3eFsR 4SC4jlqcX9J6d7qFf41c AQj5gTsVYa4OAO -----END RSA PRIVATE KEY-----

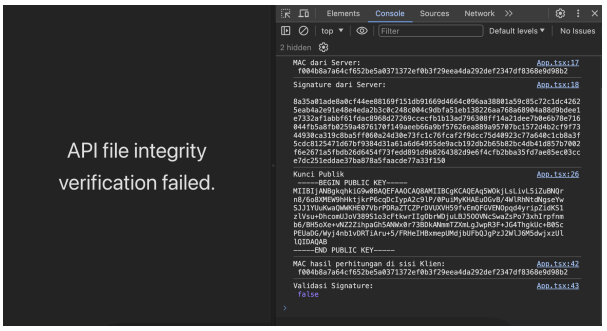
Inisiasi API	`@app.route('/api/hello')`
	`def hello():`
	`  return jsonify({`
	`  'message': 'Hello World'`
	`  })`

Berikut adalah isi dari API yang digunakan

```
@app.route('/api/hello')
def hello():
  return jsonify({
    'message': 'Hello World'
  })
```

Gambar IV.7 API yang digunakan untuk menghasilkan MAC

Setelah dilakukan verifikasi di sisi klien, berikut adalah hasil yang diperoleh.



Gambar IV.8 Hasil verifikasi di sisi klien

Berikut adalah komponen - komponen yang dilakukan dalam proses verifikasi API di sisi klien

TABLE IX. DATA YANG DIGUNAKAN DALAM PROSES VERIFIKASI

Kunci MAC	'jason_kanggara'
MAC dari server	f004b8a7a64cf652be5a0 371372ef0b3f29eea4da2 92def2347df8368e9d98b 2
MAC perhitungan dari klien	f004b8a7a64cf652be5a0 371372ef0b3f29eea4da2 92def2347df8368e9d98b 2
Signature dari server	8a35a01ade8a0cf44ee88 169f151db91669d4664c 096aa38801a59c85c72c 1dc42625eab4a2e91e48e 4eda2b3c0c248c004c9d bfa51eb138226aa768a68 904a88d9bdee1e7332af1 abbf61fdac8968d27269c cecfb1b13ad796308ff14 a21dee7b0e6b78e71604 4fb5a8fb0259a4876170f 149aeeb66a9bf57626ea8 89a95707bc1572d4b2cf 9f7344930ca319c8ba5ff 060a24d30e73fc1c76fca f2f9dcc75d40923c77a64 0c1cb8a3f5cdc8125471d 67bf9384d31a61a6d649 55de9acb192db2b65b82 bc4db41d857b7002f6e2 671a5fbdb26d6454f73fe dd891d9b8264382d9e6f 4cfb2bba35fd7ae85ec03 cce7dc251eddae37ba878 a5faacde77a33f150
Kunci publik yang digunakan untuk validasi	-----BEGIN PUBLIC KEY-----

tanda tangan	MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAq5WOkjLsLivL5iZuBNQrn8/6o8XMEW9hHktjkrP6cqDcIypA2c9lP/0PuiMyKHAEuOGvB/4WlRhNtdNgseYwSJJ1YUuKwaQWWKHE07VbrPDRaZTCZPrDVUXVH59fvEmQFGVENOpqd4yripZidKS1zlVsu+DhcomUJoV389S1o3cFtkwrIIgObrWDjuLBJ5OOVNcSwaZsPo73xhIrpfnmb6/BH5oXe+vNZ2ZihpaGh5ANWx0r73BDkANmmTZXmLgJwpR3F+JG4ThgkUc+B0ScPEUaDG/Wyj4nb1vDRTiAru+5/FRHeIHBxmepUMdjbUFbQJgPzJ2WlJ6M5dwjxzUllQIDAQAB -----END PUBLIC KEY-----
Validitas Signature	False

5. kasus-uji-5

Kasus uji 5 akan dilakukan pengujian dengan melakukan modifikasi pada API dan menggunakan kunci MAC saat verifikasi di sisi klien.

TABLE X. INISIASI KUNCI PADA KASUS UJI 1

Kunci MAC	'jason_kanggara'
Kunci publik	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5U4Grlpo7V0Sn10iucAdlD43B7zB79Sy2Aq8NzLs6aaD4qhK6DSU42hfCGpDGvfFf5oTgdReM3T6i60Xa6BHBcBgO/LJ8RdOe6Op6a32Qos5o7A72ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgYUxdUloERnJp430Tj4QP+IFvbBT8GlP8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzdtMDw2iC9QIA4MF9145bt7LDb8SUSl1+AX5UfBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----
Kunci Privat	-----BEGIN RSA PRIVATE KEY----- MIIEowIBAAKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3 lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5U4Grlpo7V0Sn10iucAdlD43B7zB79 Sy2Aq8NzLs6aaD4qhK6DSU42hfCGpDGvfFf5oTgdReM3T6i60Xa6BHBcBgO/LJ8R dOe6Op6a32Qos5o7A72ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgYUxdUloERnJp 430Tj4QP+IFvbBT8GlP8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzdtMDw2iC9QI A4MF9145bt7LDb8SUSl1+AX5UfBOfkNL5gna EQIDAQABAoIBAAO14EcOPkaRMPKmt4yAoBlyTN1sP/sS6urNjWwIKpjkvBKngNahA XHCPZeoyKICA4/smYLsZx8PPMhmjreHGJ3jFUccM96G8j7Ob08qZYX+t9sxTrf9N wZtlpu5m15nQ09tkgGUWehLGlaHxJdV1hpgiN9f99UAWSJj9j1PXBWrMYeHdweP ikA1tRqXAsUcjwhEA0V0S9RwD7ck aStd2jURY/wOEHuEzq+bJ3MOsxcenI1l9/G2105miadEhDH3qtZtduvT 32Qos5o7A72ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgYUxdUloERnJp430Tj4QP+IFvbBT8GlP8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzdtMDw2iC9QIA4MF9145bt7LDb8SUSl1+AX5UfBOfkNL5gna EQIDAQAB -----END PUBLIC KEY-----

	W/KyIL9L wY7+Q0Bzn/Z3dmQZL IhmPavIZ4Y6ARcflBAT BL5tpLMu8aF202XzKv neiXxbHlvI ptyibWECgYEAyJ3WEs oKQ33ZodR/HKX5Zh/il O2/1m1nOVHjdbCpiuR yYz77I61S P8yj8BDz5fawCIXNJ0h /RkKESVShbF6Jg2XbC /SG21jtQRCkcZI0GR1c mcLzqw/g eYJ75t2fL3b4vYO3zvK nOzIIqLwU6PLqwtyEA PwD6aV3yXswvhVV3 AcCgYEA0APz nnUEsUQNjovKURbN+ a34+QiwW0WXqery/Tg DmjYldFGLatrWJzf0dv WAsJIDalxb fPZcQ+9fC8DWQ9bhm OIXpOt8CGan2ItVmGF JKHLe7ZKoWG2np3Sd EkEbyM68OqXV SspEIRtTYYMG7dd6m oL8zCsowBklfyjcGWbE wycCgYEAqhNlUTiHy v/+aPvFZ3wu Gy4c4TVN1XOadzW9 KCFFGsLhVtImBvEB MaVDtloo62RfiFomuo7 zl5T8oU5mK4Fa A1NuiIwdaAfT271ILC BOmpwRzSlBelxTaa/9a pOoSI0MA16PjZqZu82 HGcWZ1INc EybvogYFKydkjiMqB3 v+9jECgYAxFT2e3i3FR jGMuR98Es13ZL2ixZI Xd0xaI75C Ub1H3g7oa3ViOVDe4C 0nBLbg8xMrPa+4xMjw nsLmmxHz3/hLgs3hdE9 IDDD+v6Hq 2snQaYqAtGJUu0scEE Af1o0fzOhNTmQNybx Zs4DxOWGhLX8Yhl/w LR7odQaMArAs rjsvcQKBgBvBwjV8Hn XiSJwKgrqXh0EeZe0K qS4+yinzrOGnfLd2OZn DHv+1SwNn X2dxgIwiizrpgheXTPR NJmLVNjOej5PMy6uV KMVWzXY55FuhGT6

	GNRAfdwxc2W70 avWKn9QNESf/Q3eFsR 4SC4jlqcX9J6d7qFf41c AQj5gTsVYa4OAO -----END RSA PRIVATE KEY-----

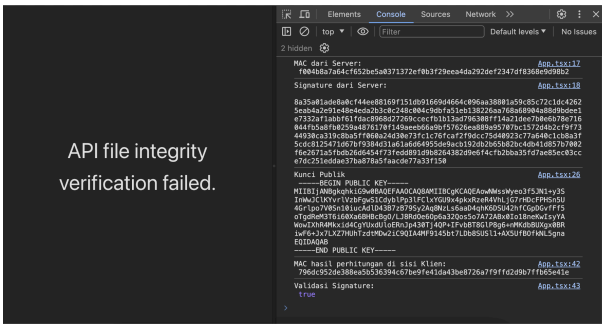
Inisiasi API	`@app.route('/api/hello')`
	`def hello():`
	`  return jsonify({`
	`  'message': 'Hello World'`
	`  })`

Berikut adalah isi dari API yang digunakan

```
@app.route('/api/hello')
def hello():
  return jsonify({
    'message': 'Tampered Data'
  })
```

Gambar IV.9 API yang digunakan untuk menghasilkan MAC

Setelah dilakukan verifikasi di sisi klien, berikut adalah hasil yang diperoleh.



Gambar IV.10 Hasil verifikasi di sisi klien

Berikut adalah komponen - komponen yang dilakukan dalam proses verifikasi API di sisi klien

TABLE XI. DATA YANG DIGUNAKAN DALAM PROSES VERIFIKASI

Kunci MAC	'kriptografi'
MAC dari server	f004b8a7a64cf652be5a0 371372ef0b3f29eea4da2 92def2347df8368e9d98b 2
MAC perhitungan dari klien	796dc952de388ea5b536 394c67be9fe41da43be87

	26a7f9ffd2d9b7ffb65e41e
Signature dari server	8a35a01ade8a0cf44ee88169f151db91669d4664c096aa38801a59c85c72c1dc42625eab4a2e91e48e4eda2b3c0c248c004c9dbfa51eb138226aa768a68904a88d9bdee1e7332af1abbf61fdac8968d27269ccecfb1b13ad796308ff14a21dee7b0e6b78e716044fb5a8fb0259a4876170f149aeeb66a9bf57626ea889a95707bc1572d4b2cf9f7344930ca319c8ba5ff060a24d30e73fc1c76fcaf2f9dcc75d40923c77a640c1cb8a3f5cdc8125471d67bf9384d31a61a6d64955de9acb192db2b65b82bc4db41d857b7002f6e2671a5fbdb26d6454f73fedd891d9b8264382d9e6f4cfb2bba35fd7ae85ec03cce7dc251eddae37ba878a5faacde77a33f150
Kunci publik yang digunakan untuk validasi tanda tangan	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5U4Grlpo7V0Sn10iucAdlD43B7zB79Sy2Aq8NzLs6aaD4qhK6DSU42hfCGpDGvfFf5oTgdReM3T6i60Xa6BHBcBgO/LJ8RdOe6Op6a32Qos5o7A72ABx0Io18neKwIsyYAWowIXhR4Mkxid4CgYUxdUloERnJp430Tj4QP+IFvbBT8GlP8g6+nMKdbBUXgx0BRiwF6+Jx7LXZ7HUhTzdtMDw2iC9QIA4MF9145bt7LDb8SUSl1+AX5UfBOfkNL5gnaEQIDAQAB -----END PUBLIC KEY-----
Validitas Signature	True

6. kasus-uji-6

Kasus uji 6 akan dilakukan pengujian dengan melakukan modifikasi pada API dan menggunakan kunci privat dan kunci publik yang berbeda pada saat penandatangan dan pada saat proses verifikasi di sisi klien

TABLE XII. INISIASI KUNCI PADA KASUS UJI 1

Kunci MAC	'jason_kanggara'
Kunci publik	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5U4Grlpo7V0Sn10iucAdlD43B7zB79Sy2Aq8NzLs6aaD4qhK6DSU42hfCGpDGvfFf5oTgdReM3T6i60Xa6BHBcBgO/LJ8RdOe6Op6a32Qos5o7A72ABx0Io18neKwIsyYAWowIXhR4Mkxid4CgYUxdUloERnJp430Tj4QP+IFvbBT8GlP8g6+nMKdbBUXgx0BRiwF6+Jx7LXZ7HUhTzdtMDw2iC9QIA4MF9145bt7LDb8SUSl1+AX5UfBOfkNL5gnaEQIDAQAB -----END PUBLIC KEY-----
Kunci Privat	-----BEGIN RSA PRIVATE KEY----- MIIEowIBAAKCAQEAowNWssWyeo3f5JN1+y3SInWwJClKYvrlVzbFgwS1CdyblPp3lFClxYGU9x4pkxRzeR4VhLjG7rHDcFPHSn5

	U4Grlpo7V0Sn10iucAdl D43B7zB79 Sy2Aq8NzLs6aaD4qhK 6DSU42hfCGpDGvfFf5 oTgdReM3T6i60Xa6BH BcBgO/LJ8R dOe6Op6a32Qos5o7A7 2ABx0Io18neKwIsyYA WowIXhR4Mkxid4CgY UxdUloERnJp 430Tj4QP+IFvbBT8GlP 8g6+nMKdbBUXgx0BR iwF6+Jx7LXZ7HUhTzd tMDw2iC9QI A4MF9145bt7LDb8SUS l1+AX5UfBOfkNL5gna EQIDAQABAoIBAAO1 4EcOPkaRMPKm t4yAoBlyTN1sP/sS6urN jWwIKpjkvBKngNahA XHCPZeoyKICA4/smY LsZx8PPMhm jreHGJ3jFUccM96G8j7 Ob08qZYX+t9sxTrf9N wZtlpu5m15nQ09tkgGU WehLGlaH xJdV1hpgiN9f99UAWS Jj9j1PXBWrMYeHdweP ikA1tRqXAsUcjwhEA0 V0S9RwD7ck aStd2jURY/wOEHuEzq +bJ3MOsxcenI1l9/G210 5miadEhDH3qtZtduvT W/KyIL9L wY7+Q0Bzn/Z3dmQZL IhmPavIZ4Y6ARcflBAT BL5tpLMu8aF202XzKv neiXxbHlvI ptyibWECgYEAyJ3WEs oKQ33ZodR/HKX5Zh/il O2/1m1nOVHjdbCpiuR yYz77I61S P8yj8BDz5fawCIXNJ0h /RkKESVShbF6Jg2XbC /SG21jtQRCkcZI0GR1c mcLzqw/g eYJ75t2fL3b4vYO3zvK nOzIIqLwU6PLqwtyEA PwD6aV3yXswvhVV3 AcCgYEA0APz nnUEsUQNjovKURbN+ a34+QiwW0WXqery/Tg DmjYldFGLatrWJzf0dv WAsJIDalxb fPZcQ+9fC8DWQ9bhm OIXpOt8CGan2ItVmGF

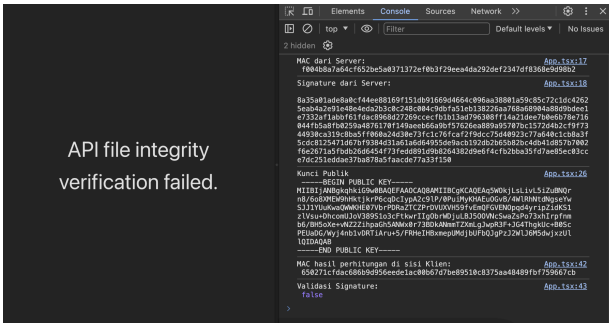
	JKHLe7ZKoWG2np3Sd EkEbyM68OqXV SspEIRtTYYMG7dd6m oL8zCsowBklfyjcGWbE wycCgYEAqhNlUTiHy v/+aPvFZ3wu Gy4c4TVN1XOadzW9 KCFFGsLhVtImBvEB MaVDtloo62RfiFomuo7 zl5T8oU5mK4Fa A1NuiIwdaAfT271ILC BOmpwRzSlBelxTaa/9a pOoSI0MA16PjZqZu82 HGcWZ1INc EybvogYFKydkjiMqB3 v+9jECgYAxFT2e3i3FR jGMuR98Es13ZL2ixZI Xd0xaI75C Ub1H3g7oa3ViOVDe4C 0nBLbg8xMrPa+4xMjw nsLmmxHz3/hLgs3hdE9 IDDD+v6Hq 2snQaYqAtGJUu0scEE Af1o0fzOhNTmQNybx Zs4DxOWGhLX8Yhl/w LR7odQaMArAs rjsvcQKBgBvBwjV8Hn XiSJwKgrqXh0EeZe0K qS4+yinzrOGnfLd2OZn DHv+1SwNn X2dxgIwiizrpgheXTPR NJmLVNjOej5PMy6uV KMVWzXY55FuhGT6 GNRAfdwxc2W70 avWKn9QNESf/Q3eFsR 4SC4jlqcX9J6d7qFf41c AQj5gTsVYa4OAO -----END RSA PRIVATE KEY-----
Inisiasi API	

Berikut adalah isi dari API yang digunakan

```python
@app.route('/api/hello')
def hello():
    return jsonify({
        'message': 'Tampered Data'
    })
```

Gambar IV.11 API yang digunakan untuk menghasilkan MAC

Setelah dilakukan verifikasi di sisi klien, berikut adalah hasil yang diperoleh.



Gambar IV.12 Hasil verifikasi di sisi klien

Berikut adalah komponen - komponen yang dilakukan dalam proses verifikasi API di sisi klien

TABLE XIII. DATA YANG DIGUNAKAN DALAM PROSES VERIFIKASI

Kunci MAC	'jason_kanggara'
MAC dari server	f004b8a7a64cf652be5a0371372ef0b3f29eea4da292def2347df8368e9d98b2
MAC perhitungan dari klien	650271cfdac686b9d956eede1ac00b67d7be89510c8375aa48489fbf759667cb
Signature dari server	8a35a01ade8a0cf44ee88169f151db91669d4664c096aa38801a59c85c72c1dc42625eab4a2e91e48e4eda2b3c0c248c004c9dbfa51eb138226aa768a68904a88d9bdee1e7332af1abbf61fdac8968d27269ccecfb1b13ad796308ff14a21dee7b0e6b78e716044fb5a8fb0259a4876170f149aeeb66a9bf57626ea889a95707bc1572d4b2cf9f7344930ca319c8ba5ff060a24d30e73fc1c76fcaf2f9dcc75d40923c77a640c1cb8a3f5cdc8125471d67bf9384d31a61a6d64955de9acb192db2b65b82bc4db41d857b7002f6e2671a5fbdb26d6454f73fedd891d9b8264382d9e6f4cfb2bba35fd7ae85ec03cce7dc251eddae37ba878a5faacde77a33f150
Kunci publik yang digunakan untuk validasi tanda tangan	-----BEGIN PUBLIC KEY----- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAq5WOkjLsLivL5iZuBNQrn8/6o8XMEW9hHktjkrP6cqDcIypA2c9lP/0PuiMyKHAEuOGvB/4WlRhNtdNgseYwSJJ1YUuKwaQWWKHE07VbrPDRaZTCZPrDVUXVH59fvEmQFGVENOpqd4yripZidKS1zlVsu+DhcomUJoV389S1o3cFtkwrIIgObrWDjuLBJ5OOVNcSwaZsPo73xhIrpfnmb6/BH5oXe+vNZ2ZihpaGh5ANWx0r73BDkANmmTZXmLgJwpR3F+JG4ThgkUc+B0ScPEUaDG/Wyj4nb1vDRTiAru+5/FRHeIHBxmepUMdjbUFbQJgPzJ2WlJ6M5dwjxUllQIDAQAB -----END PUBLIC KEY-----
Validitas Signature	False

C. *Hasil Analisis Keseluruhan*

Berdasarkan hasil pengujian yang sudah dilakukan pada sub bab B, terlihat bahwa jika adanya ketidakcocokan atau perubahan pada API dan kunci kunci yang digunakan, proses verifikasi API di sisi klien tidak akan berhasil yang mengakibatkan klien tidak dapat menggunakan API. Untuk memastikan verifikasi berhasil, seluruh kunci serta API yang digunakan untuk menghasilkan MAC harus sesuai.

Untuk melakukan perubahan API yang terotentikasi, setelah melakukan perubahan dilanjutkan dengan menghasilkan MAC baru dari API yang sudah diubah agar

klien dapat memverifikasi API tersebut dengan berhasil dan dapat menggunakannya.

V. Kesimpulan

Dari makalah ini, dapat disimpulkan bahwa pemanfaatan MAC dan tanda tangan digital dapat menjaga integritas dari suatu API untuk memastikan keamanan pada API dengan terbebas dari modifikasi modifikasi eksternal oleh pihak yang tidak berwenang. Perubahan sekecil apapun pada API akan mengubah MAC secara drastis sehingga verifikasi API antara klien dan server menjadi tidak berhasil. Tidak hanya itu, klien dan server juga harus memastikan adanya kesamaan antara pasangan kunci privat dan kunci publik agar API dapat digunakan. Dengan diimplementasikannya kedua aspek kriptografi tersebut, API memiliki perlindungan yang berlapis untuk menjadi integritas dari konten API yang dikembangkan.

Ucapan Terima Kasih

Penulis mengucapkan puji dan syukur kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga makalah dengan judul "Implementasi *Message Authentication Code* dan *Digital Signature* untuk *API Content Integrity"* dapat diselesaikan dengan baik. Penulis juga mengucapkan terima kasih kepada dosen pengajar IF4020 Kriptografi, Dr. Rinaldi Munir, S.T, M.T. yang telah memberikan bimbingan dan ilmu terkait materi Kriptografi ini, khususnya pada ilmu *Message Authentication Code* dan *Digital Signature*. Penulis juga mengucapkan terima kasih kepada para penulis sumber referensi karena sudah memberikan informasi tambahan yang membantu menyelesaikan makalah ini.

Referensi

[1] Amazon; Apa Itu API?; Amazon Web Service

[2] Munir, Rinadli, 2024; MAC (Message Authentication Code); Institut Teknologi Bandung

[3] Munir, Rinaldi, 2024; Tanda-Tangan Digital; Institut Teknologi Bandung

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 12 Juni 2024

Jason Kanggara, 13520080