Aplikasi Absensi Kehadiran Berbasis Web dengan Memanfaatkan Sandi Rahasia

Aditio Pangestu
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jalan Ganesha 10 Bandung 40132, Indonesia
aditiopangestu@gmail.com

***Abstract*—Dalam dunia pendidikan, kehadiran di kelas menjadi tolak ukur yang penting dalam evaluasi proses belajar mengajar. Sehinggap pelacakan kehadiran pelajar perlu untuk dilakukan. Pelacakan ini dilakukan dnegan pencatatan kehadiran pelajar di suatu kelas yang dilakukan setiap pertemuan. Terdapat beberapa metode pencatatan kehadiran pelajar. Secara umum, metode pencatatan kehadiran pelajar dapat dibagi dua yaitu tradisional dan modern. Pada metode tradisional, perhitungan secara manual maupun pemindahan data secara digital menjadi pekerjaan yang sulit dan membosankan bila jumlah kelas dan jumlah anggota kelas cukup besar. Sedangkan pada metode modern, membutuhkan perangkat keras tambahan yang spesifik dan memiliki nilai beli yang cukup besar. Untuk menangani permasalahan tersebut dibagun aplikasi absensi berbasis website yang memanfaatkan sandi rahasia. Untuk meningkatkan keaman pada aplikasi ini akan digunakn fungsi sipHash dan localStorage.**

Keywords—component; sipHash; localStorage; website; absensi

I. PENDAHULUAN

Dalam dunia pendidikan, kehadiran di kelas menjadi tolak ukur yang penting dalam evaluasi proses belajar mengajar. Selain itu jumlah kehadiran di kelas juga mampu mempengaruhi prestasi pelajar. Secara umum, kehadiran yang baik mampu memberikan prestasi yang baik. Atas dasar tersebut, banyak instansi pendidikan yang melacak jumlah kehadiran pelajar di kelas. Bahkan, terdapat beberapa instansi pendidikan yang memberikan sanksi yang berat terhadap pelajar yang memiliki jumlah kehadiran yang di bawah batasan minimum yang ditetapkan oleh instansi pendidikan tersebut. Contohnya, pada Institut Teknologi Bandung (ITB) dan Universitas Padjadjaran (Unpad) diterapkan peraturan tidak dapat mengikuti Ujian Akhir Semester (UAS) apabila jumlah kehadiran mahasiswa di kelas dibawah dari 80% dari total kelas yang diadakan.

Pencatatan kehadiran pelajar di suatu kelas dilakukan setiap pertemuan. Metode pencatatan kehadiran sangat mempengaruhi tingkat keakuran atau kebenaran pelacakan kehadiran pelajar. Terdapat beberapa metode pencatatan kehadiran pelajar. Secara umum, metode pencatatan kehadiran pelajar dapat dibagi dua yaitu tradisional dan modern. Metode pencatatan tradisional masih melakukan perhitungan manual terhadap jumlah kehadiran total atau membutuhkan tindakan pemindahan data ke bentuk digital agar dapat dilakukan secara otomatis. Contoh dari metode pencatatan tradisional adalah pencatatan dengan memanfaatkan kertas absensi yang disebarkan pada tiap pertemuannya. Sedangkan metode pencatatan modern tidak memerlukan tindakan pemindahan data ke bentuk digital sehingga perhitungan total langsung dilakukan secara otomatis. Contoh dari metode pencatatan modern adalah pencatatan dengan memanfaatkan mesin absensi sidik jari, NFC atau RFID.

Metode pencatatan yang tersedia, masih memiliki beberapa kekurangan. Pada metode tradisional, perhitungan secara manual maupun pemindahan data secara digital menjadi pekerjaan yang sulit dan membosankan bila jumlah kelas dan jumlah anggota kelas cukup besar. Sedangkan pada metode modern, membutuhkan perangkat keras tambahan yang spesifik dan memiliki nilai beli yang cukup besar, contoh dibutuhkan pembelian mesin absensi sidik jari, pembaca NFC, atau pembaca RFID.

Dari kekurangan metode pencatatan yang tersedia, diajukan sebuah metode modern berupa pencatatan dengan memanfaatkan sandi rahasia yang menjadi inputan pada aplikasi absensi berbasis web. Metode ini hanya membutuhkan sebuah aplikasi website dan kertas yang berisi sandi rahasia untuk melakukan absensi.

Pengiriman sandi rahasia akan memanfaatkan SipHash untuk menjaga otentikasi dan integritas dari sandi rahasia yang dikirim. Lalu, untuk mencegah pemalsuan absensi, sandi rahasia yang diberikan spesik untuk seseorang dalam satu pertemuan kelas yang dibankitkan menggunakan SipHash dan memanfaatkan *local storage* yang dimiliki *browser* untuk mendeteksi keberadaan kunci rahasia.

Pada kesempatan ini, juga dibahas kekurangan dari aplikasi Absensi Kehadiran Berbasis Web dari segi keamanan. Beberapa kekurangan yang akan dibahas akan diberikan solusi yang berupa teknis maupun non teknis.

II. DASAR TEORI

A. *Metode Penyimpanan State pada Aplikasi Website*

Aplikasi website memanfaatkan HTTP untuk komunikasi antara *client* dan *server*. Tetapi, HTTP merupakan *stateless protocol*, dibutuhkan tag tertentu untuk *client* dan *server* saling mengenal status terbaru. Tag tertentu tersebut disimpan di suatu

tempat penyimpanan. Terdapat dua jenis tempat penyimpanan yaitu *local storage* dan *cookies*.

Local storage dan *cookies* memiliki perbedaan yang signifikan terhadap entitas yang membutuhkan. Apabila data yang disimpan dibutuhkan oleh *server* maka penyimpanan pada *cookies* lebih tepat sebab *cookies* memanfaatkan header dari HTTP untuk mempertahankan data. Lalu, apabila data yang disimpan dibutuhkan oleh *client* maka penyimpanan pada *local storage* lebih tepat sebab *local storage* memanfaatkan tempat penyimpanan yang dimiliki oleh browser.

B. Hash-based Message Authentication Code (HMAC)

HMAC merupakan salah satu metode implementasi MAC yang mengkombinasikan fungsi hash satu arah dan kunci rahasia simetris. Fungsi hash satu arah merupakan suatu fungsi yang mengubah suatu pesan dengan panjang tertentu menjadi suatu pesan yang memiliki panjang tetap (*message-digest*) yang tidak dapat dikembalikan menjadi pesan awal. Sedangkan kunci rahasia simetris merupakan kunci bernilai sama yang dimiliki oleh dua belah pihak yang ingin melakukan komunikasi.

Fungsi hash satu arah memiliki sifat berupa dua pesan yang berbeda tidak mungkin memiliki *message-digest* yang sama. Lalu, kunci rahasia simetris memiliki sifat berupa spesifik dimiliki pihak tertentu. Dari dua sifat tersebut, dapat dihasilkan cara untuk menjaga otentikasi dan integeritas dari sebuah pesan yang dikirim (tujuan MAC).

Selanjutnya akan dibahas tahap-tahap yang terjadi pada HMAC. Berikut tiga tahap utama yang terjadi pada HMAC :

- Pengirim dan penerima telah memiliki kunci rahasia simetris.
- Nilai MAC dibangkitkan menggunakan fungsi hash satu arah dengan inputan berupa pengolahan pesan dan kunci rahasia.
- Pesan diverifikasi kerahasiannya menggunakan nilai MAC yang diterima dan kunci rahasia simetris.

Pada tahap kedua dan ketiga terdapat pembangkitan nilai MAC. Pembangkitan nilai MAC dapat dilakukan menggunakan persamaan berikut :

$$HMAC(K,m) = H((K' \oplus opad) \vee (K' \oplus ipad) \vee m)$$

dengan keterangan sebagai berikut :

- H, fungsi hash satu arah yang digunakan,
- K, kunci rahasia simetris,
- m, pesan,
- K', hasil pembangkitan kunci rahasia lain dari kunci rahasia, K, menggunakan fungsi tertentu,
- opad, *outer padding*, bilangan konstanta heksadesimal 0x5c5c...5c5c,
- ipad, *inner padding*, bilangan konstanta heksadesimal 0x3636...3636.

C. ShipHash

ShipHash merupakan fungsi hash satu arah yang menjadi sebuah solusi dari permasalahan waktu eksekusi yang berlebihan ketika membangkitkan nilai MAC dari pesan yang singkat, pada metode HMAC menggunakan fungsi SHA. ShipHash memiliki tingkat keamanan yang tinggi dan ketika dievaluasi termasuk ke dalam *cryptographically strong PRF* (*pseudorandom function*).

ShipHash memiliki dua parameter bilangan bulat yaitu c dan d, umunnya ditulis menjadi SipHash-c-d. Nilai c merupakan jumlah eksekusi fungsi SipRound pada proses kompresi, sedangkan d merupakan jumlah eksekusi fungsi SipRound pada proses finalisasi. Kunci yang digunakan SipHash untuk menghasilkan nilai MAC harus berukuran 128-bit. Sedangkan untuk ukuran pesan tidak terdapat batasan. Misalkan terdapat kunci k dan sebuah pesan m, maka ShipHash-2-4 akan mengembalikan nilai MAC berukuran 64 bit dengan perhitungan sebagai berikut

- **Inisialisasi**, proses ini menghasilkan 4 buah 64-bit buffer $v0$, $v1$, $v2$, dan $v3$ yang dihasilkan dari perhitungan dibawah ini :

  $$v0 = k0 \oplus 736f6d6570736575$$
  $$v1 = k1 \oplus 646f72616e646f6d$$
  $$v2 = k0 \oplus 6c7967656e657261$$
  $$v3 = k1 \oplus 7465646279746573$$

  dengan $k0$ dan $k1$ merupakan 64-bit pertama k dan 64-bit kedua k dengan format *little-endian*.

- **Kompresi**, proses ini dimulai dengan penguraian m yang berukuran b-byte menjadi $w = \lceil (b+1)/8 \rceil$ buah buffer yang berukuran 64-bit dalam format *little-endian*, $m0$, $m1$, ... dan $m(w-1)$ dengan mi merupakan 64-bit ke-i pada m, kecuali untuk $i = w - 1$ merupakan penambahan pading 0 di depan m yang tersisa sehingga memiliki ukuran 7-byte lalu di depan 7-byte tersebut ditambahkan nilai b mod 256.

  Selanjutnya, untuk setiap $mi \in \{m0, m1, \dots, m(w-1)\}$ secara berurutan dilakukan perhitungan di bawah ini :

  $$v3 \oplus = mi,$$

  lalu 2 kali eksekusi fungsi SipRound dan diakiri dengan

  $$v0 \oplus = mi$$

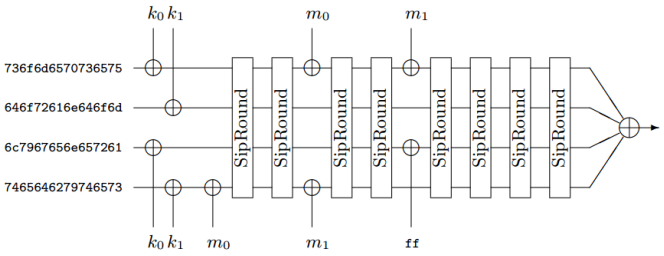
- **Finaliasi**, proses untuk menghasilkan nilai MAC dengan melakukan perhitungan

  $$v3 \oplus = 255,$$

  lalu 4 kali eksekusi fungsi SipRound dan diakiri dengan

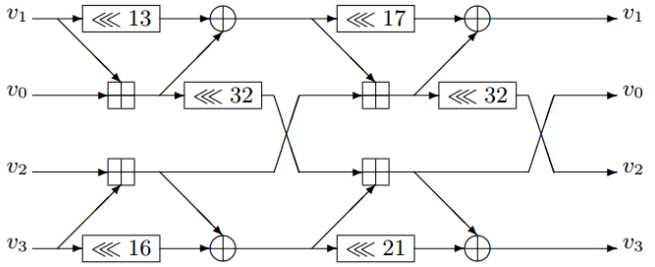
  $v0 \oplus v1 \oplus v2 \oplus v3$ yang menjadi nilai MAC

Pada gambar 2, dapat dilihat proses sipHash yang terjadi saat menghasilkan nilai MAC untuk pesan yang berukuran 15-byte.



Gambar 1 Contoh Proses SipHash pada Pesan yang Berukuran 15-byte

Selanjutnya, untuk mengetahui proses dari fungsi SipRound dapat dilihat pada gambar 1.



Gambar 2 Jaringan Arx pada SipRound

D. Pertukaran Kunci Diffie-Helman

Pertukaran kunci diffie-helman merupakann metode berbagi kunci semetris antara dua orang yang aman untuk dilakukan pada kanal publik. Pertukaran kunci ini memanfaatkan kesukaran dalam pencarian solusi dari sebuah persamaan logaritma diskrit, sebuah permasalahan pencarian nilai x jika diketahui $a^x = b \ mod \ p$ dengan a, b suatu bilangan buat dan p suatu bilangan bulat prima.

Selanjutnya akan dibahas protokol kriptografi pada pertukaran kunci *diffie-helman*. Berikut protokol kritografi pada pertukaran kunci *diffie-helman*:

- Sebelum melakukan pertukaran kunci, Alice dan Bob telah meyepakati dua buah bilangan prima yang besar, n dan g, dengan g < n. Bilangan n dan g tidak perlu dirahasiakan, dapat disepakati menggunakan kanal publik.
- Alice membangkitkan bilangan bulat acak yang besar, x, dan mengirimkan hasil perhitungan dari $X = g^x \ mod \ n$ ke Bob.
- Bob membangkitkan bilangan bulat acak yang besar, y, dan mengirimkan hasil perhitungan dari $Y = g^y \ mod \ n$ ke Alice.
- Alice menghitung kunci rahasia simetris $K = Y^x \ mod \ n$.
- Bob menghitung kunci rahasia simetris $K = X^y \ mod \ n$.

III. DESAIN APLIKASI ABSENSI KEHADIRAN BERBASIS WEBSITE

Pada aplikasi ini terdapat tiga alur penting yang perlu diperhatikan yaitu pertukaran kunci rahasia simetris, pembangkitan sandi rahasia dan pengiriman sandi rahasia. Berikut desain dari ketiga alur tersebut.

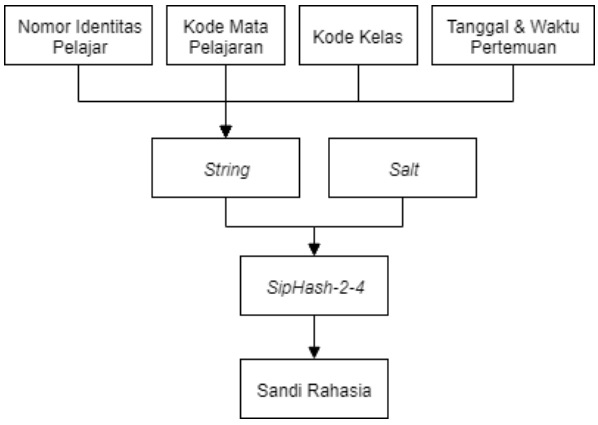
A. Pertukaran Kunci Rahasia

Alur pertukaran kunci rahasia akan menggunakan metode pertukaran kunci *Diffie-Helman*. Pada pertukaran kunci *Diffie-Helman* dibutuhkan kesepakatan dalam menentukan dua buah bilangan prima besar. Pada aplikasi ini, dua buah bilangan prima besar tersebut ditentukan secara sepihak oleh *client*. Lalu *client* akan mengirim dua buah bilangan prima besar tersebut ke *server* bersamaan dengan kunci publik *client*. Selanjutnya server akan menghasilkan kunci publik berdasarkan dua buah bilangan prima besar yang dikirim oleh *client*. Lalu server akan mengirim kunci publik ke *client* bersamaan dengan dua buah bilangan prima besar yang dikrim oleh *client*.

Pengiriman kembali bertujuan untuk mengkonfirmasi kepada *client* bahwa dua buah bilangan yang diterima oleh *server* merupakan dua buah bilangan yang benar. Apabila tidak sama, maka *client* akan mengirim ulang kembali dua buah bilangan prima besar hingga *client* menerima dua buah bilangan prima besar yang benar. Jika sama, *client* akan mengkonfirmasi ke *server* dan menyimpan kunci rahasia simetris yang didapatkan. Ketika *server* menerima konfirmasi bahwa bilangan prima yang digunakan benar, maka *server* akan menyimpan kunci rahasia yang didapatkan tersebut.

B. Pembangkitan Sandi Rahasia

Pembangkitan sandi rahasia yang akan digunakan sebagai masukan aplikasi absensi kehadiran berbasis website memanfaat sipHash-2-4, lihat gambar 3. Penggunaan sipHash pada pembangkitan sandi rahasia ini dikarenakan sipHash memiliki kelebihan performa dalam menghasilkan nilai acak dari sebuah pesan yang singkat. Pesan yang akan dijadikan parameter masukan sipHash merupakan sekumpulan *string* yang tersusun dari nomor identiatas pelajar, kode mata pelajaran, kode kelas dan tanggal dan waktu pertemuan. Pemilihan pesan tersebut diharapkan setiap orang dari sebuah pertemuan di sebuah kelas pada mata pelajaran tertentu akan memiliki sandi rahasia yang berbeda-beda. Selanjutnya untuk parameter kunci pada sipHash akan digunakan *salt* yang berukuran 128-bit dan telah tersimpan untuk masing-masing pelajar.



Gambar 3Proses Pembangkitan Sandi Rahasia

Hasil dari sipHash berupa buffer yang berukuran 64-bit atau 16 karakter dalam bentuk heksadesimal. 16 karakter memiliki ukuran yang cukup panjang untuk dimasukan secara manual pada aplikasi absensi kehadiran berbasis website. Oleh karena itu dari 16 karakter tersebut akan diambil 5 karakter pertama yang akan dijadikan sandi rahasia.

C. *Alur Pengiriman Sandi*

Setiap pelajar akan mengambil sebuah kertas berisikan sandi rahasia yang bersesuaian. Mahasiswa akan memasukkan sandi rahasia tersebut ke dalam sebuah form yang tersedia pada aplikasi absensi kehadiran berbasis website. Setelah pelajar memasukkan sandi tersebut, *client* akan membangkitkan nilai MAC dengan memanfaatkan fungsi sipHash. Pesan yang dijadikan parameter merupakan sandi rahasia yang dimasukkan oleh pelajar, sedang kunci yang dijadikan parameter merupakan kunci rahasia simetris yang telah dimiliki. Kunci rahasia simestris diproses terlebih dahulu agar memiliki ukuran sebesar 128-bit.

Setelah memiliki sandi rahasia dan nilai MAC, *client* akan mengirimkan informasi tersebut ke s*erver*. Lalu, *server* akan memvalidasi otentikasi dan integritas pesan menggunakan nilai MAC. Setelah valid, *server* baru memvalidasi sandi rahasia yang diterima. Apabila terjadi kegagalan/keberhasilan pada validasi maka *server* akan mengkonfirmasi hal tersebut kepada *client*.

IV. IMPLEMENTASI

Pada kesempatan kali ini, aplikasi yang dihasilkan berupa *high fidelity prototype* yang telah memiliki tiga alur penting pada aplikasi absensi kehadiran berbasis website. *Framework* aplikasi website yang digunakan adalah Node.js. Untuk pengembangan *client-side* digunakan framework Reactjs. Berikut deskripsi detail dari pengembangan tiga alur tersebut.

A. *Pertukaran Kunci Rahasia*

Pertukaran kunci rahasia terjadi saat melakukan *login* ke aplikasi. Komunikasi antara *client* dengan *server* menggunakan Ajax. Setelah berhasil login data berupa informasi user dan kunci rahasia simetris disimpan pada *local storage* pada *client*. Sedangkan pada *server*, sandi rahasia akan diperbaharui dalam file json.

Saat penyimpanan data terdapat pengecekan khusus pada *client*. Apabila terdapat data lama pada *local storage* maka akan dilakukan pengecekan *id* pengguna. Apabila *id* pengguna pada *local storage* sama dengan *id* pengguna yang akan disimpan, maka akan dilakukan pembaharuan data. Apabila sebaliknya, maka data tidak akan di simpan dan pengguna akan diberikan pesan peringatan bahwa satu browser atau perangkat hanya boleh digunakan oleh satu pengguna.

B. *Pembangkitan Sandi Rahasia*

Pada pembangkitan rahasia tidak terdapat perlakuan khusus dalam proses implementasi. Dengan sistem pendidikan yang terjadwal dan peserta yang tetap. Pembangkitan sandi rahasia dapat dilakukan sekali. Sandi rahasia dibangkitkan menggunakan *library* yang tersedia pada laman https://github.com/jedisct1/siphash-js.

Tiap pertemuannya sandi rahasia akan dicetak dengan tambahan informasi berupa nomor identitas mahasiswa. Lalu setiap pasangan sandi rahasia dan nomor identitas mahasiswa dijadikan sebuah kupon yang sedemikan rupa membuat informasi sandi rahasia sulit dikenal dan nomor indentitas mudah dikenal.

C. *Alur Pengiriman Sandi*

Pada alur pengiriman sandi, *client* dan *server* saling berkomunikasi menggunakan Ajax. Request yang dikirimkan terdapat informasi *id* pengguna, *id pertemuan,* nilai MAC dan sandi rahasia. Nilai MAC dibangkitkan menggunakan library yang tersedia pada laman https://github.com/jedisct1/siphash-js.

V. KEAMANAN APLIKASI

Aplikasi absensi berbasis website ini masih memiliki kekurangan pada penggunaan satu pengguna untuk sebuah perangkat. Hal ini dikarenakan kemampuan dari sebuah browser tidak mampu menangkap nomor identitas dari satu perangkat yang menggunakan browser tersebut.

Walaupun nomor identitas perangkat dapat dibuat berupa bayangan dengan memanfaatkan *localStorage*, masih memiliki kekurangan dikarenakan *localStorage* mendeteksi alamat IP dari pengguna. Ketika mendeteksi alamat IP yang berbeda, maka *localStorage* yang dihasilkan berbeda pula. Hal ini lah yang dapat dimanfaatkan untuk penggunaan beberapa pengguna dalam sebuah perangkat.

Untuk mencegah hal ini, dapat dilakukan dengan memperketat pembagian kupon sandi rahasia. Walaupun sebuah perangkat dapat digunakan untuk beberapa akun, apabila pengguna tidak dapat memiliki sandi rahasia maka pengguna tersebut tidak dapat melakukan absensi.

VI. KESIMPULAN DAN SARAN

Aplikasi absensi berbasis website memungkinkan untuk dibangun dengan tingkat keamanan yang sebanding dengan absensi tradisional dan memiliki kemampuan seperti absensi modern, serta lebih murah dibandingkan absensi modern. Kemungkinan ini dihasilkan dari kemampuan browser untuk menyimpan data secara lokal (*localStorage)* dan terdapat algoritma SipHash yang memiliki performa yang baik untuk pengolahan pesan yang berukuran kecil. SipHash memiliki tingkat keaman yang setara dengan MAC-SHA, tetapi lebih cepat dibandingkan dengan MAC-SHA dalam menghasilkan nilai MAC dari pesan yang berukuran kecil.

Terdapat tiga alur utama yang perlu diperhatikan dalam pembuatan aplikasi absensi berbasis website yaitu :

- Pertukaran kunci rahasia yang memanfaatkan metode pertukaran kunci rahasia Diffie-Helman dan localStorage untuk menyimpan kunci simetris
- Pembangkitan Sandi Rahasia yang memanfaatkan fungsi sipHash dengan parameter pesan berupa gabungan string dari nomor identiatas pelajar, kode mata pelajaran, kode kelas dan tanggal dan waktu pertemuan dan parameter kunci berupa salt berukuran 128-bit.

- Alur pengiriman sandi yang memanfaatkan fungsi sipHash dengan parameter pesan berupa sandi rahasia dan parameter kunci berupa kunci simetris yang disimpan

UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih kepada Allah SWT, karena atas rahmat dan karunia-Nya lah makalah ini dapat selesai pada waktunya. Penulis juga ingin mengucapkan terima kasih kepada kedua orang tua yang tidak pernah letih mendukung dan mendoakan anaknya, serta Bapak Dr. Ir. Rinaldi Munir selaku dosen mata kuliah Kriptografi. Tidak lupa penulis juga ingin mengucapkan terima kasih kepada pihak-pihak lain yang telah membantu dalam penyelesaian makalah ini.

REFERENSI

[1] Munir, R. “MAC”. 2018. Slide Kuliah IF4020 Kriptografi.

[2] Munir, R. “Diffie-Helman”. 2018. Slide Kuliah IF4020 Kriptografi.

[3] “localStorage vs. Cookies for Analytics Implementations: What You Should Know” https://www.webanalyticsworld.net/2017/09/localstorage-vs-cookies-for-analytics-implementations.html diakses tanggal 14 Mei 2018.

[4] Aumasson, Jean-Philippe, and Daniel J. Bernstein. "SipHash: a fast short-input PRF." International Conference on Cryptology in India. Springer, Berlin, Heidelberg, 2012.

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, 18 Mei 2018

Aditio Pangestu / 13514030