Penggunaan Encripsi MD5 Untuk Pencegahan *SQL Injection* *Pada Aplikasi Berbasis Web*

[1]Benediktus Zebua, [2]Patah Herwanto, [3]Rosida
[1,2,3] Program Studi Teknik Informatika, STMIK IM
Bandung Indonesia
email : [1]benediktuszebua@gmail.com, [2]pherwanto@stmik-im.ac.id, [3]Rosida21@yahoo.com

ABSTRAK

Pengembangan website yang terus berkembang dan masif maka sistem keamanan adalah hal yang paling penting untuk melindungi informasi yang bersifat sensitif. SQL Injection merupakan salah satu serangan yang sangat populer dan sering digunakan untuk memanfaatkan celah keamanan pada sistem untuk menyerang database pada sebuah website. Oleh karena itu, pencegahan perlu dilakukan untuk menghindari serangan SQL Injection. Penelitian ini menggunakan metode pengumpulan data dan pengembangan sistem. Pengumpulan data menggunakan studi pustaka dan studi lapangan. Pengembangan sistem dilakukan dengan metode prototype, sebelum mendapatkan kesepakatan pada bentuk sistem yang akan dibangun maka perubahan dan presentasi pada prototype yang dirancang dapat dilakukan berkali-kali. Pencegahan adanya serangan SQL Injection pada suatu website, maka peneliti tertarik membuat sistem pencegahan dengan menggunakan Algoritma MD5 untuk mengenkripsi parameter query string. Enkripsi dilakukan untuk menjaga keaslian data yang dikirimkan ke sistem agar tidak terjadi penyisipan perintah-perintah yang membahayakan sistem. Dalam pembuktian implementasi MD5 untuk pencegahan SQL injection ini digunakan aplikasi website sederhana yang bisa menggambarkan proses bagaimana pencegahan dapat dilakukan.

Kata kunci: SQL Injection, Kerentanan, Algoritma MD5, Query String.

ABSTRACT

Website development is constantly evolving and massive, the system security is the most important thing to protect sensitive information. SQL Injection is one of the attacks that are very popular and are often used to exploit security loopholes in the system to attack the database on a website. Therefore, prevention needs to be done to avoid SQL Injection attacks. This study uses the method of data collection and the development of the system. Data collection using literature study and field study. The development of the system is carried out with the prototype method, before getting an agreement on the form of the system to be built then it changes and the presentation of the designed prototype can be done many times. Prevention of presence of SQL Injection attacks on a website, then the researcher is interested in creating a system of prevention by using the MD5 Algorithm to encrypt the parameters of the query string. The encryption is done to preserve the authenticity of the data that is sent to the system so that does not happen insertion orders-orders that harm the system. In proving the implementation of MD5 for SQL injection prevention, a simple website application is used that can describe the process by which prevention can be done.

Keywords: *SQL Injection, Vulnerability, Algorithm MD5, Query String.*

Pendahuluan

Di era sekarang ini, zaman menuntut untuk menyesuaikan diri pada perkembangan teknologi informasi yang berkembang sangat pesat. Mulai dari pelayanan bisnis yang dapat memberikan kemudahan kepada konsumen dan begitu juga pada pemerintahan, pendidikan dan bidang kesehatan yang mengandalkan teknologi untuk memberikan pelayanan yang mudah dan dapat dijangkau oleh siapa dan dimana saja. Tumpukan kertas hilang dan informasi disimpan secara digital sehingga mempermudah mengakses dan mengontrol data. Dalam hal ini basis data sebagai media penyimpanan informasi yang tentunya memiliki peranan yang sangat penting. Maka sistem keamanan adalah hal yang paling utama untuk menjaga dan melindungi data yang tersimpan dalam basis data.

Seiring dengan perkembangan pelayanan yang disediakan, jumlah serangan berbasis web juga semakin tinggi. Serangan yang ditargetkan pada sisi kerentanan dan kelemahan pada aplikasi web, salah satu serangan terhadap web yang masih populer hingga saat ini adalah SQL Injection. Sehingga pada penelitian ini, peneliti fokus pada keamanan database Structured Query Language atau disingkat SQL. SQL digunakan dalam jenis database untuk melakukan proses mengambil dan manajemen informasi dalam database. SQL dapat dieksploitasi oleh SQL Injection. Jenis serangan ini melakukan penyisipan perintah query SQL ke dalam parameter query. Kode perintah yang disisipkan ke dalam parameter query yang dikirim dari klien dan dijalankan dalam basis data dan dapat menyebabkan akses tidak sah ke data atau modifikasi di dalam basis data. Kode yang disisipkan dalam parameter query SQL akan dianggap sebagai kode SQL biasa dan dijalankan dalam basis data.

Algoritma MD5 merupakan fungsi hash satu arah yang dirancang sebagai pengembangan dari MD4 oleh Ron Rivest. Algoritma MD5 melakukan pemrosesan masukan ke dalam blok-blok bit sebanyak 512 bit dan kemudian dibagi ke dalam 32 bit sub-blok sebanyak 16 buah. Setelah melakukan pemrosesan maka terbentuk hasil keluaran berupa 4 (empat) buah blok yang tiap-tiap blok berjumlah 32 bit sehingga menjadi 128 bit dan disebut menjadi nilai hash (Kusuma, 2019). Fungsi pada hash kriptografik yang digunakan untuk pemeriksaan integritas pada data dalam berbagai situasi. Hash sendiri adalah fungsi yang digunakan untuk melakukan proses perubahan pada suatu data menjadi data yang lain dengan panjang tertentu, sehingga data tersebut tidak dapat dipulihkan kembali atau mengembalikan ke nilai semula. Dengan cara ini biasa diterapkan dalam enkripsi data-data penting, misalnya penyimpanan kata sandi supaya tidak mudah mengetahuinya walaupun dapat melihat kode hash dari kata sandi tersebut.

Pada penelitian ini penulis menggunakan Algoritma MD5 untuk Enkripsi Query String. Metode ini dapat digunakan sebagai sistem yang dapat memblokir akses bagi penyerang yang menginputkan permintaan berbahaya. Penelitian ini dibuat untuk tujuan mengatasi dan melakukan pencegahan serangan SQL Injection yang dilakukan oleh penyerang ke dalam sistem website, sehingga data yang tersimpan dalam basis data website menjadi lebih aman dan terhindar dari pencurian data yang dilakukan oleh penyerang.

Materi dan Metode

1. Kriptografi

Kriptografi (Cryptography) yang berasal dari bahasa Yunani yang terdiri dari suku kata kripto yang berarti menyembunyikan dan graphia yang berarti tulisan. Jadi, kriptografi yaitu ilmu yang mempelajari teknik matematika yang mempunyai hubungan pada aspek keamanan informasi, seperti kerahasian, keabsahan, integritas, serta autentikasi pada data. Namun, dengan kriptografi tidak semua aspek dalam keamanan sistem informasi dapat diselesaikan (Amin, 2016).

Berdasarkan definisi diatas dapat disimpulkan bahwa Kriptografi adalah ilmu yang memanfaatkan teknik enkripsi data dalam penyampaian pesan secara tersembunyi untuk menyamarkan dan mengamankan suatu informasi.

2. Enkripsi

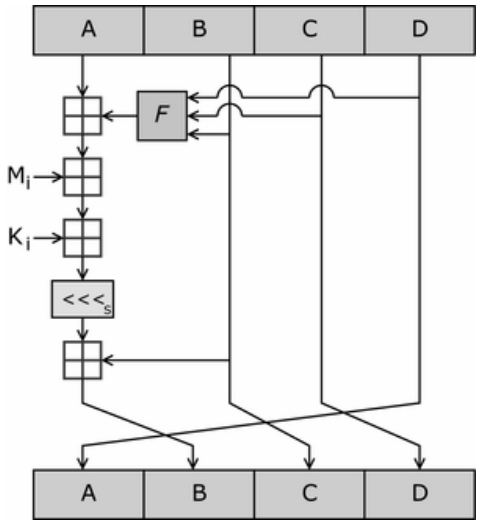
Enkripsi adalah proses penyamaran sebuah data pesan menjadi pesan unik, dengan cara mengubah plaintext menjadi ciphertext. Menurut Rusdianto dan Qashlim (2016) Enkripsi adalah suatu fungsi sebagai kunci pengontrol yang menerima banyak perhatian dalam organisasi saat ini. Kebutuhan pada organisasi untuk mengenkripsi informasi sensitif harus terpenuhi. Enkripsi dapat membantu mencegah data disalahgunakan oleh pihak yang tidak bertanggung jawab misalnya terjadi pencurian data, serta mencegah penipuan dalam sebuah organisasi. Dalam beberapa kasus, enkripsi juga digunakan diwajibkan untuk memenuhi persyaratan peraturan agar data konsumen aman dan terlindungi.

3. Algoritma MD5 (Message-Digest Algorithm 5)

Algoritma MD5 merupakan fungsi hash satu arah yang dirancang sebagai pengembangan dari MD4. Algoritma MD5 melakukan pemrosesan inputan pada blok - blok bit sebanyak 512 bit dan kemudian akan dibagi ke dalam 32 bit sub-blok yang banyaknya 16 buah. Setelah melakukan pemrosesan maka terbentuk hasil keluaran berupa 4 buah blok yang masing-masing berjumlah 32 bit sehingga menjadi 128 bit dan disebut menjadi nilai hash (Kusuma, 2019). MD5 sering digunakan untuk proses validasi dan memeriksa integritas data.

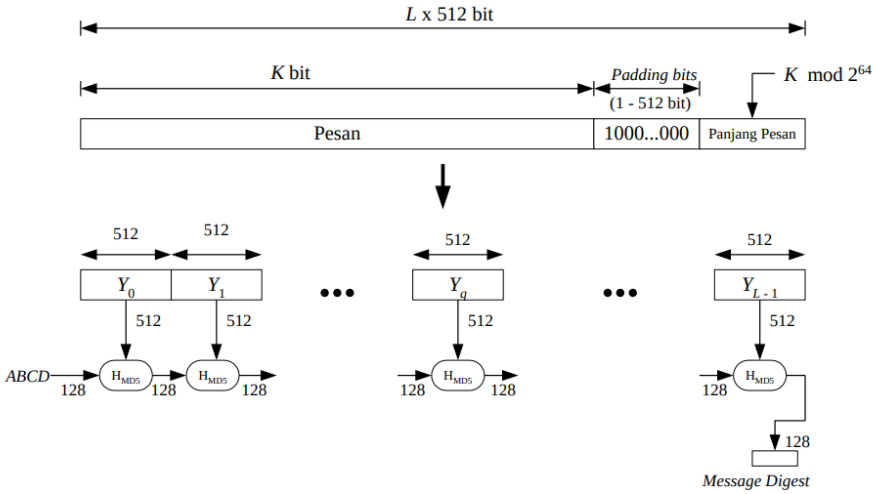
MD5 adalah salah satu algoritma message-digest yang dirancang oleh Prof. Ronald Rivest dari Massachusetts Institute of Technology (MIT). Kerja analitis menerangkan bahwa pendahulu MD5 yaitu MD4 mulai tidak aman, maka MD5 kemudian mulai dirancang pada tahun 1991 dan dipublish pada marete tahun 1992 sebagai pengganti dari MD4. Hash MD5 memiliki panjang 128-bit (16-byte), dikenal sebagai inti pesan, message-digest secara khas akan ditampilkan ke bilangan heksadesimal 32 digit. MD5 telah dimanfaatkan di berbagai macam sistem keamanan dalam aplikasi dan umumnya MD5 digunakan untuk proses pengujian integritas data. (Sibyan, 2017).

Hal yang pertama kali dilakukan sebelum proses MD5 yaitu pengubahan data sehingga jadi data bit. Jika tahap pertama selesai, maka MD5 melanjutkan memproses data sehingga menjadi blok-blok data yang terdiri dari 512 bit blok, dan ini dibagi ke dalam 16 sub-blok yang terdiri dari 32 bit. hash value 128 bit alam didapatkan setelah proses dari blok-blok tersebut.



Gambar 2.2 Satu Operasi MD5

Gambar 2.2 adalah merupakan satu operasi MD5 yang terdiri atas 64 operasi, digabungkan pada 4 putaran dari 16 proses operasi. F digunakan pada tiap-tiap putaran yang fungsinya non-linear. Mi menandakan blok 32 bit yang dihasilkan dari inputan pesan, dan Ki menunjukkan hasil konstanta 32 bit yang berbeda pada tiap-tiap operasi. Pergeseran pada kiri bit oleh s, dan s bervariasi pada tiap-tiap operasi dengan notasi <<<s yaitu operasi circular left.



Gambar 2.3 Pembuatan message digest algoritma MD5
(Munir, 2006)

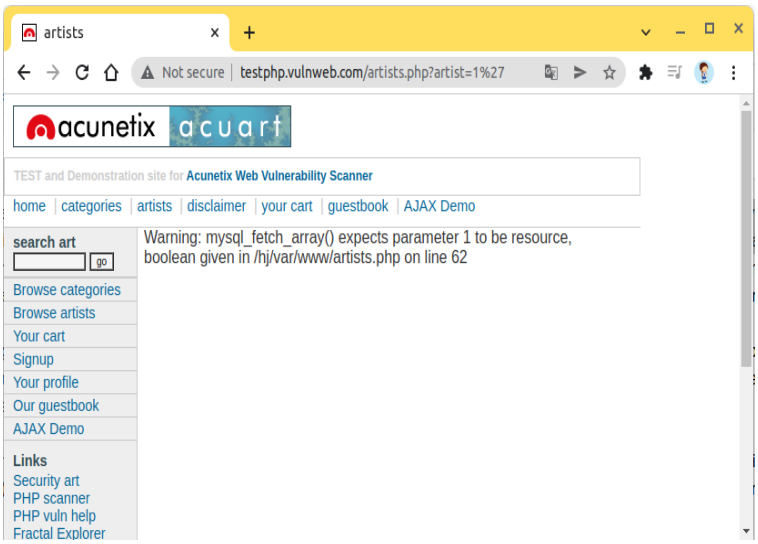
Pada Gambar 2.3 menerangkan secara garis besar tahapan pembuatan hash value dimulai dari tahapan penambahan bit-bit pengganjal (padding bits), penambahan nilai panjang pada pesan awal, setelah itu dilakukan proses pembentukan penyangga (buffer) MD, pada langkah terakhir dilakukan pengolahan pesan pada blok berukuran 512 bit.

4. SQL Injection

SQL Injection atau disingkat dengan SQLi yaitu salah satu jenis serangan injeksi yang memanfaatkan celah kerentanan pada website yang memungkinkan untuk mengeksekusi perintah-perintah SQL yang berbahaya. Perintah berbahaya memungkinkan untuk mengontrol server database sistem aplikasi web. Sehingga pihak yang tidak bertanggung jawab dapat menemukan akses dan otorisasi ke halaman web atau aplikasi web dan mengambil konten dari seluruh database SQL.

Penyerang juga dapat menggunakan SQL Injection untuk menambah, mengubah, dan menghapus catatan dalam database. Menurut (Nugraha dkk., 2013) SQL Injection adalah serangan yang merubah query normal pada aplikasi menjadi query berbahaya yang memungkinkan pengaksesannya dan pemrosesan basis data secara tidak normal. Adapun pendapat lain menurut Zam, E. (2012) yang dikutip oleh Gunadhi & Nugraha (2016) SQL Injection terdiri dari dua kata, yaitu SQL merupakan bahasa yang digunakan untuk memanajemen suatu database, sedangkan kata injection jika diterjemahkan yaitu menyuntik. SQL Injection adalah sebuah metode yang digunakan untuk menginjeksi perintah SQL berbahaya sebagai nilai masukan melalui sebuah web guna mendapatkan akses database.

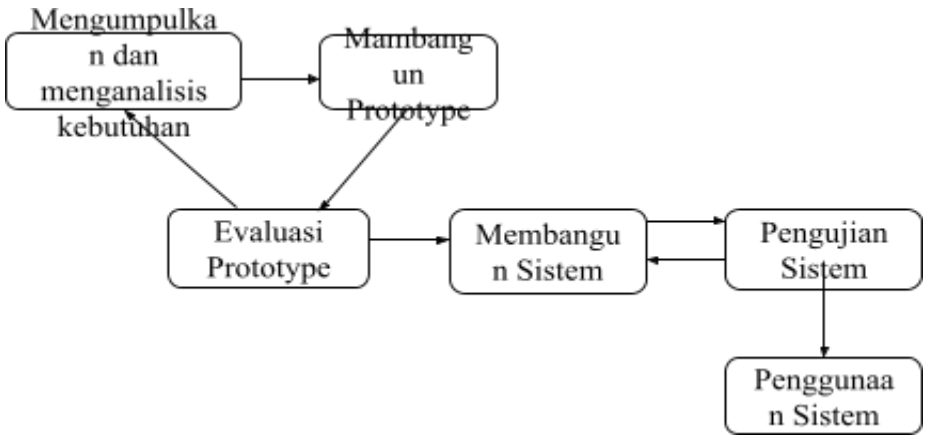
Memanipulasi query normal pada url target http://testphp.vulnweb.com/artists.php?artist=1 dengan mengubah atau menambahkan nilai parameter artist dengan (') pada akhiran url sehingga menjadi http://testphp.vulnweb.com/artists.php?artist=1', bila website yang ditargetkan menampilkan pesan kesalahan dari database seperti pada gambar 2.1 maka dapat disimpulkan website tersebut sangat rentan terhadap serangan SQL Injection.



Gambar 2.4 Acunetix Acuart (http://testphp.vulnweb.com)

5. Metode Pengembangan Sistem

Metode yang digunakan pada pengembangan sistem ini adalah metode prototype.



Gambar 2.5 Metode Prototype (O'Brien, 2005)

Prototype merupakan salah satu metode pengembangan pada perangkat lunak untuk membuat perancangan yang relatif cepat dan bertahap sehingga dapat dapat segera dievaluasi oleh calon pengguna/klien.

1. Mengumpulkan dan Menganalisis Kebutuhan
   Pada awal pekerjaan yang sifatnya teknis, diawali dengan mengumpulkan dan menganalisis kebutuhan untuk kelancaran pengambangan pada setiap tahap yang dilakukan.
2. Membangun Prototype
   Membangun prototype dengan membuat rancangan sesuai pembicaraan terhadap calon pengguna/klien.
3. Evaluasi Prototype
   Evaluasi ini dilakukan untuk menerima masukan dan saran untuk memperbaiki prototype yang sudah dibangun sebelum melanjutkan pada tahap berikutnya.
4. Membangun Sistem.
   Di tahap ini adalah melakukan implementasi hasil prototype yang selesai dirancang ke dalam bahasa pemrograman.
5. Pengujian Program
   Tahap ini dilakukan setelah pembangunan sistem telah selesai untuk mengetahui hasil dan fungsi-fungsi dari sistem yang bangun.
6. Menggunakan Sistem
   Setelah sistem berhasil dibangun dan telah diuji coba, dapat diimplementasikan ke sistem yang lainnya.

Hasil dan Pembahasan

1. Implementasi Kode

Untuk melakukan implementasi rancangan enkripsi pada query string dengan menggunakan algoritma MD5, perlu dilakukan dalam beberapa tahap. Yang penting untuk diperhatikan dalam implementasi ini adalah pada sisi kodenya. kode enkripsi query string diletakkan pada halaman yang memuat konten yang dapat diklik atau dilihat rinciannya, pada rancangan ini kode enkripsi akan diletakkan di file index.php dan file blog.php.

2. Pembuatan Private Key

Sebelum implementasi fungsi enkripsi m5 pada query string, yang perlu dilakukan lebih dulu yaitu pembuatan private key. Pembuatan private key cukup mudah dengan menggunakan text secara acak dan menambahkan karakter unik, dan ukurannya tidak dibatasi. Private key ini sangat penting dan rahasia, key ini dapat disimpan file constants.php mendefinisikan variabel dengan nama "crypKey" agar mudah dipanggil pada saat dibutuhkan.

Gambar 3.3 File constants.php

3. Implementasi fungsi enkripsi MD5 pada file index

```
<?php foreach($result_post as $data): ?>
  <article class="entry">
      <header class="entry__header">

          <h2 class="entry__title h1">
              <a href="<?php echo 'blog.php?id='.$data['id'].'&key='.md5
              ('id='.$data['id'].$cryptKey);?>" title="<?php echo $data
              ['title']?>"><?php echo $data['title']?></a>
          </h2>

          <div class="entry__meta">
          </div>
      </header>
      <div class="entry__content">
        <p><?php echo $data['summary']?></p>
      </div>
  </article>
<?php endforeach;?>
```

Gambar 3.1 Kode enkripsi query string pada url

Pada gambar diatas dapat dilihat kode untuk melakukan enkripsi pada query string di setiap url data yang ditampilkan di halaman index.php, hasil dari enkripsi tersebut disimpan pada query stringnya sendiri dan diberi nama sebagai key, yang nantinya key tersebut akan digunakan sebagai verifikasi untuk permintaan data sesuai konten yang diakses. Apabila tidak ada perubahan pada query stringnya maka konten yang dipilih dapat ditampilkan.
Agar kode enkripsi unik dan tidak mudah ditebak oleh siapapun, maka perlu penambahan private key sebagai kata kunci pembeda pada enkripsi query stringnya.

4. Implementasi fungsi verifikasi key enkripsi pada file blog

```php
<?php
    // Create database connection using config file
    include_once "app/init.php";

    $id = $_GET['id'];
    $key = $_GET['key'];

    if(!is_numeric($id)) {
        header("location: ./error.php", FORBIDDEN);
    }
    if(strlen(trim($key)) != 32) {
        header("location: ./error.php", FORBIDDEN);
    }
    if($key === md5('id='.$id.$cryptKey)) {
        // apabila $key dengan hasil enkripsi pada query string sama dengan key yang di terima
        // maka selanjutnya query ke dalam database dapat dilakukan
         $query = "
            SELECT
                post.id,
                post.title,
                post.summary,
                post.content,
                post.publishedAt,
                user.firstName AS authorName,
                category.title as categoryName,
                category.slug AS categorySlug
            FROM post
            LEFT JOIN user ON user.id=post.authorId
            LEFT JOIN post_category ON post_category.postId = :id
            LEFT JOIN category ON category.id=post_category.categoryId
            WHERE post.id = :id AND published = 1;
        ";

        $sth = $conn->prepare($query);
        $sth->bindValue(':id', $id, PDO::PARAM_INT);
        $sth->execute();
        $sth->setFetchMode(PDO::FETCH_ASSOC);
        // ...
    } else {
        http_response_code(403);
        header("location: ./error.php");
    }
```

Gambar 3.2 Kode verifikasi pada key query string

Pada gambar diatas dapat dilihat kode cara verifikasi pada key hasil enkripsi md5 pada halaman blog.php. Membuat satu kondisi untuk melakukan verifikasi key dengan hasil enkripsi ulang pada query string yang diterima dengan penambahan private key, apabila hasilnya sama maka kode di dalam dijalankan atau melakukan query ke database. dan apabila hasil verifikasinya tidak sama maka diarahkan ke halaman error.

5. Pengujian

Pengujian dilakukan untuk mengetahui Enkripsi Algoritma MD5 agar sesuai dengan perancangan yang dibuat dan mempunyai hasil yang diinginkan serta menemukan kesalahan-kesalahan yang mungkin terjadi pada bagian implementasi.

6. Pengujian Black Box

Sistem pencegahan SQL Injection dan user tidak terverifikasi yang telah dibangun, selanjutnya diuji dengan menggunakan metode Black Box Testing. Tahap pengujian

dilakukan dengan tujuan untuk menunjukan bahwa sistem yang telah dibangun sesuai dengan hasil analisis dan perancangan sehingga sistem tersebut sesuai dengan yang diharapkan.

7. Pengujian Dengan Menggunakan Unit Test Secara Otomatis

Pengujian dengan unit test secara otomatis dilakukan untuk mengetes query string yang terenkripsi dan tidak enkripsi secara mudah.

8. Pengujian dengan SQLMap

Penerapan enkripsi md5 pada query string pada url website terbukti mampu mencegah terjadi SQL Injection atau menjaga keaslian pada query string yang dikirim lewat url website. Penerapan sql injection pada website yang sudah diterapkan enkripsi pada query string tidak menemukan error yang akan tampil di website dan tahapan sql injection tidak dapat dilanjutkan.

Kesimpulan

SQL Injection sangat mudah terjadi bila web yang dibangun tidak memperhatikan keamanan pada data yang dalam parameter query string, dan minimnya validasi yang dilakukan pada saat ada permintaan pada website. Sistem Pencegahan SQL Injection dengan menggunakan Algoritma MD5 untuk mengenkripsi query string, terbukti berhasil bahwa dengan enkripsi query string dapat melakukan pencegahan dari perilaku user yang tidak terverifikasi dan pencegahan dari serangan yang menggunakan alat tertentu misalnya SQLmap.

Daftar Pustaka

Amin, M. M. (2016). Implementasi Kriptografi Klasik Pada Komunikasi Berbasis Teks. *Jurnal Pseudocode*, *3*(2), 129-136.

Gunadhi, E., & Nugraha, A. P. (2016). Penerapan Kriptografi Base64 Untuk Keamanan Url (Uniform Resource Locator) Website Dari Serangan Sql Injection. *Jurnal Algoritma*, *13*(2), 391-398.

Kusuma, E. D. (2019). Penerapan Algoritma MD5 untuk Menjaga Keamanan Terhadap File yang Di-download. *ALGOR*, *1*(1), 38-41.

Nugraha, S. G., Djanali, S., & Pratomo, B. A. (2013). Sistem Pendeteksi dan Pencegah Serangan SQL Injection dengan Penghapusan Nilai Atribut Query SQL dan Honeypot. *Jurnal Teknik Pomits*, *2*(1), 1-5.

Rusdianto, & Qashlim, A. (2016). Implementasi Algoritma MD5 Untuk Keamanan Dokumen. *Jurnal Ilmiah Ilmu Komputer Fakultas Ilmu Komputer Universitas Al Asyariah Mandar*, *2*(2), 10-15.

Sibyan, H. (2017). Implementasi Enkripsi Basis Data Dengan Algoritma Md5 (Message Digest Algorithm 5) Dan Vigenere Cipher. *Jurnal PPKM*, *1*, 114-121.