**CyberSecurity dan Forensik Digital** e-ISSN: 2615-8442
Vol. 1, No. 1, Mei 2018, hlm. 1-6

# PERANCANGAN APLIKASI PENGUJIAN CELAH KEAMANAN PADA APLIKASI BERBASIS WEB

**Fietyata Yudha[1], Andi Muhammad Panji Muryadi T[2]**

[1,2]Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia
Email: [1]yudha@uii.ac.id, [2]13523260@students.uii.ac.id

**Abstrak**

Kemajuan dunia digital membuat kehidupan manusia menjadi semakin mudah. Berbagai aplikasi digital bermunculan untuk memudahkan kehidupan umat manusia. Salah satu teknologi yang cukup berkembang pesat adalah aplikasi berbasis web. Aplikasi web dapat berjalan pada berbagai sistem operasi dengan bantuan perambah web. Aplikasi berbasis web juga termasuk aplikasi yang ringan untuk digunakan, karena itu aplikasi web menjadi pilihan bagi sebagian orang. Perkembangan aplikasi web menjadi tantangan tersendiri bagi para pengembangan aplikasi web dalam mengembangkan aspek keamanan. Aplikasi web merupakan platform aplikasi yang cukup rentan terhadap serangan dari peretas. SQL Injection, Phising, dan Cross-Site Scripting (XSS) merupakan beberapa jenis serangan yang dapat menyerang aplikasi berbasis web. Dalam meningkatkan sistem keamanan dari sebuah aplikasi berbasis web maka perlu dilakukan sebuah tahapan pengujian keamanan dari aplikasi berbasis web tersebut. Pengujian keamanan dilakukan dengan melakukan uji teknin-teknik serangan yang mungkin terhadap aplikasi target. Terdapat banyak sumber daya yang ditawarkan untuk melakukan pengujian aplikasi berbasis web. Namun, penggunaan sumber daya tersebut masih yang bersifat manual dan belum terintegrasi. Penelitian ini akan mengusulkan sebuah rancangan aplikasi terintegrasi yang dapat digunakan untuk melakukan pengujian terhadap aplikasi berbasis web.

**Kata kunci**: *web based application, keamanan, peretas, pengujian*

# *DESIGN OF WEB APPS SECURITY ASSESSMENT APPLICATIONS*

***Abstract***

*The evolution of digital things makes live easier for human being. Various digital applications emerge to facilitate the life of human. One of technology that's growing rapidly is the web based application. Web based application can be run on various operating system with the assistance of web browser. Web based application is lightweight apps, so the web apps is the option for some people. The evolution of web apps challenged for the developers to develop the security aspect. Web apps are fragile apps and being a target of hackers. SQL injection, phising, and Cross-Site Scripting are some attacks that targeted to web apps. To increase the security system of the web apps so it is important to doing an assessment stage on web apps development. The security assessment performed by using various possible technique for the target. There are many resources for performed web apps security assessment. However, the using of the resources are manual and not integrated. This research propose the design of integrated application for performed web app security assessment*

**Keywords**: *web based application, security, hackers, assessment*

## 1. PENDAHULUAN

Era digital membawa kehidupan manusia menjadi lebih mudah. Berbagai aplikasi digital digunakan untuk mempermudah pekerjaan manusia. Salah satu hal yang berkembang cukup pesat adalah aplikasi berbasis web. Aplikasi web dipilih karena aplikasi tersebut dapat berjalan di berbagai platform. Aplikasi berbasis web juga termasuk aplikasi yang ringan untuk digunakan, karena itu aplikasi web menjadi pilihan bagi sebagian orang. Menjamurnya aplikasi web in imenjadi tantangan tersendiri bagi para pengembangan aplikasi web dalam mengembangkan aspek keamanan pada aplikasi tersebut. Aplikasi berbasis web sendiri masih menjadi sasaran para peretas. Dari berbagai jenis yang dapat dilakukan pada aplikasi berbasis web, terdapat beberapa serangan yang paling sering dipergunakan untuk melakukan serangan pada aplikasi tersebut. Jenis serangan tersebut diantaranya SQL Injection, Phising, dan Cross-Site Scripting (XSS). Berdasarkan data yang di unggal oleh Passeri (2018) sql injection memiliki prosentase serangan sebesar 2,3 persen dari seluruh serangan pada sistem komputer.

Indonesia sendiri merupakan negara dengan jumlah pengguna internet terbanyak ke-5 di dunia di tahun 2017 berdasarkan data dari Miniwatts Marketing Group (2018). Perkembangan pengguna internet juga diimbangi dengan perkembangan aplikasi berbasis web di Indonesia. Banyak pengembang aplikasi berbasis web yang tumbuh di Indonesia. Banyaknya jumlah pengguna juga diprediksi menjadi negara yang rentan akan serangan atau disebut dengan Threat Exposure Rate(TER). Berdasarkan data dari shopos Indonesia menempati urutan pertama negara yang paling rentan terhadap serangan . Seperti terlihat pada gambar...

**10 Riskiest Countries**

| | TER | | TER |
|---|---|---|---|
| 1. Indonesia | 23.54% | 6. India | 15.88% |
| 2. China | 21.26% | 7. Mexico | 15.66% |
| 3. Thailand | 20.78% | 9. UAE | 13.67% |
| 4. Philippines | 19.81% | 9. Taiwan | 12.66% |
| 5. Malaysia | 17.44% | 10. Hong Kong | 11.47% |

Gambar 1. 10 Negara Paling Beresiko Terhadap Serangan(Sophos, 2013)

Dalam menjaga dan meningkatkan keamanan dari sebuah aplikasi berbasis web, maka perlu dilakukan pengujian aplikasi tersebut secara berkala . Pada umumnya, pengujian yang dilakukan dengan memanfaatkan sumber daya yang ada pada sistem operasi yang dapat dipergunakan untuk melakukan pengujian keamanan sistem, kali linux merupakan sistem operasi yang sering dipergunakan. Namun, penggunaan sumber daya tersebut masih yang bersifat manual dan belum terintegrasi. Diperlukan beberapa sumber daya untuk melakukan pengujian satu celah keamanan sistem, sehingga akan mengurangi efisiensi waktu dalam proses pengujian celah keamanan tersebut.

Adanya permasalahan tersebut, maka diperlukan aplikasi terintegrasi yang dapat dipergunakan untuk melakukan pengujian terhadap aplikasi berbasis web. Aplikasi tersebut berisi jenis serangan yang umumnya dilakukan oleh hacker ataupun cracker untuk mencari celah keamanan dari sebuah website, seperti SQL Injection, Phising, dan Cross-Site Scripting (XSS). Paper ini memberikan gambaran awal proses pengembangan aplikasi terintegrasi untuk melakukan npengujian kemanan aplikasi berbasis web.

## 2. KAJIAN PUSTAKA

Kie, Guo and Ernst, (2009) melakukan penelitian dengan membuat aplikasi bernama Ardilla. Aplikasi ini dikembangkan untuk melakukan pengujian terhadap sebuah halaman web.Adapun celah keamanan yang menjadi bahan uji serangan pada aplikasi ini adalah serangan SQL Injection dan Cross-Site Scripting (XSS). Pengujian yang dilakukan oleh keizun menunjukkan bahwa Ardilla berhasil menemukan 68 celah keamanan pada 5 aplikasi web. Namun, pada penelitian ini terdapat beberapa kekurangan salah satunya adalah program ini hanya dapa menguji 2 celah keamanan pada aplikasi web.

Elu (2013) membuat sebuah aplikasi pendeteksi celah SQL Injection untuk keamanan website. Aplikasi tersebut menerima masukan berupa alamat aplikasi web yang akan diuji dan memilih jenis pengujian yang akan digunakan. Aplikasi tersebut menyediakan 8 jenis pengujian, diantaranya uji komentar baris, uji komentar sebaris, uji perintah bertumpuk, uji kalimat jika, uji bilangan bulat, uji untaian, uji penggabungan hasil query, dan uji kesalahan. Aplikasi ini menggunakan teknik web crawling untuk mendapatkan data dari aplikasi web yang diuji.

Selain itu (Ahuja, Bal and Varnica, 2015) membuat penelitian mengenai teknik crawling pada aplikasi web untuk melakukan ekstraksi data dari halaman web. Program web crawler akan mengunduh halaman web yang akan di lakukan ekstraksi terhadap halaman tersebut dan akan melakukan proses parsing untuk mengambil data yang dibutuhkan sesuai algoritma parsing yang digunakan sesuai dengan tujuan dari penggunaan web crawler itu sendiri.

### 2.1. SQL Injection

SQL (Structured Query Language) Injection merupakan suatu teknik yang memungkinkan penyerang mendapatkan akses yang tidak sah kedalam database(Kie, Guo and Ernst, 2009). Teknik SQL Injection ini digunakan dengan cara memasukkan perintah-perintah SQL melalui alamat

URL (Uniform Resource Locator) atau melalui formulir masukan yang nantinya akan dieksekusi oleh server ketika meminta data ke dalam database. SQL Injection menggunakan teknik tertentu untuk melakukan manipulasi data. Teknik yang digunakan adalah dengan menambahkan karakter seperti Double Minus (--) dan Union kedalam URL antaupun formulir masukan. Terdapat beberapa beberapa hal yang dapat dilakukan untuk mengetahui apakah sebuah website rentan terhadap serangan SQL Injection :

1. Memeriksa kode apakah website telah menggunakan compiler dengan aman atau tidak.
2. Verifikasi bahwa semua penggunaan compiler secara tegas memisahkan data yang tidak dapat dipercaya dari perintah atau query.

Phising merupakan salah satu jenis serangan yang dilakukan dengan mengirimkan sebuah tautan yang akan mengarahkan korban ke sebuah halaman website. Halaman tersebut pada umumnya memiliki tampilan seperti tampilan seperti halaman aslinya dan dapat dipercaya . Tujuan dari teknik phising ini sendiri adalah untuk mendapatkan informasi sensitif seperti e-mail, username, password, dan data penting lainnya dari korban(Rachmawati, 2014).

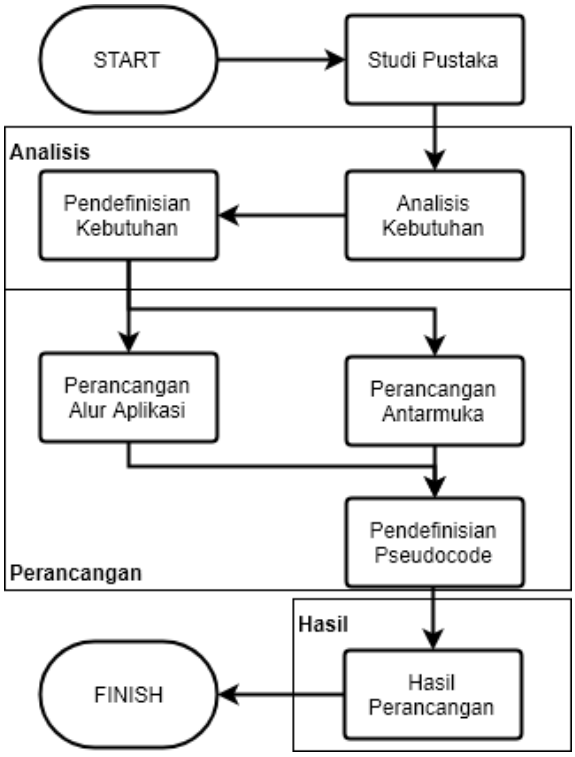
Cross-Site Scripting (XSS) merupakan sebuah jenis serangan yang dilakukan dengan memanfaatkan kelemahan server dalam melakukan validasi masukan yang diberikan oleh pengguna. XSS memungkinkan penyerang mengeksekusi kode-kode didalam perambah web yang dimiliki korban, sehingga dapat mengubah tampilan halaman web atau mengarahkan mengarahkan penggunak ke halaman-halaman yang mengandung aplikasi berbahaya(OWASP, 2017). Selain itu, jenis serangan ini juga memungkinkan penyerang untuk mencuri cookies pengguna lain.

## 3. METODOLOGI

Dalam menyelsaikan permasalahan tersebut dibangunlah sebuah metodologi. Metodologi dibangun agar proses yang dilakukan untuk menyelsaikan permasalahan tersebut diatas dapat berjalan secara sistematis dan terstruktur. Gambar 2 menunjukkan metodologi yang dibangun untuk menyelsaikan permasalah perancangan aplikasi pengujian celah keamanan pada aplikasi berbasis web.

Penelitian ini dilaksanakan dalam 4 tahapan Utama. Tahapan tersebut antara lain :

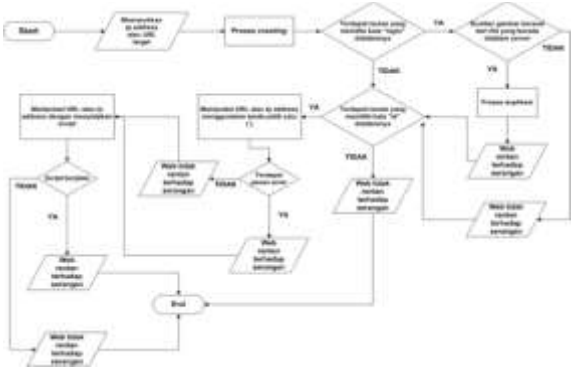
1. Studi Pustaka
   Pada tahap ini referensi mengenai aplikasi yang akan dirancang dikumpulkan. hasil dari referensi yang dikumpulkan dipergunakan untuk rekomendasi perancangan aplikasi yang dilakukan.
2. Analisis
   Rekomendasi yang didapatkan dari proses studi dipustakan dipergunakan pada tahapan analisis. Tahapan ini dibagi menjadi 2 tahapan yaitu analisis kebutuhan dan pendefinisian kebutuhan. Analisis kebutuhan merupakan proses penguraian kebutuhan dalam proses perancangan aplikasi tersebut sesuai dengan rekomendasi yang didapatkan pada tahapan pertama. Sedangkan pendefinisian kebutuhan merupakan tahapan pengembangan dari analisis kebutuhan, yaitu penentuan spesifikasi aplikasi yang akan dirancang dari segi bahasa pemgrograman, modul yang akan digunakan, dan lain-lain
3. Perancangan
   Pada tahapan ini dilakukan pembuatan rancangan dilakukan. Rancangan yang dibuat dibagi menjadi 3 bagian yaitu: rancangan alur aplikasi, pendefinisian pseudocode, dan rancangan antarmuka aplikasi. Pada tahapan rancangan alur aplikasi akan menghasilkan diagram alir aplikasi (flowchart) yang sesuai dengan aplikasi yang akan dirancang, selain itu juga dilakukan perancangan antarmuka untuk aplikasi tersebut.
4. Hasil
   Merupakan hasil akhir dari proses penelitian ini. Berdasarkan hasil perancangan yang sudah dilakukan pada tahapan sebelumnya, dilakukan pendefinisian pseudo code. Pseudocode ini yang nantinya akan dipergunakan untuk melakukan pembuatan aplikasi selanjutnya.




Gambar 2 Metodologi Penelitian

## 4. HASIL DAN PEMBAHASAN

Berdasarkan metodologi yang sudah dibuat pada bagian dan juga hasil dari tahap studi pustaka maka dapat dilakukan analisis. Hasil analisis yang dilakukan tools yang dirancang harus dapat dijalankan pada semua platform sistem operasi, maka dari itu bahasa pemrograman yang dipilih untuk membangun aplikasi ini nantinya adalah bahasa pemrograman Python dengan menggunakan interpreter Python3. Adapun untuk melakukan proses crawling pada aplikasi web yang akan di uji akan menggunakan modul BeatutifulSoup. Antarmukan dari aplikasi ini akan dibangun dengan menggunakan platform Qt5. Adapun diagram alir dari aplikasi yang akan dibuat nantinya dapat dilihat pada Gambar 3



Gambar 3 Diagram Alir Aplikasi

Berdasarkan diagram alir diatas terdapat 3 hal yang akan dilakukan oleh aplikasi yang akan dibangun. Hal tersebut dilakukan dengan urutan sebagai berikut:

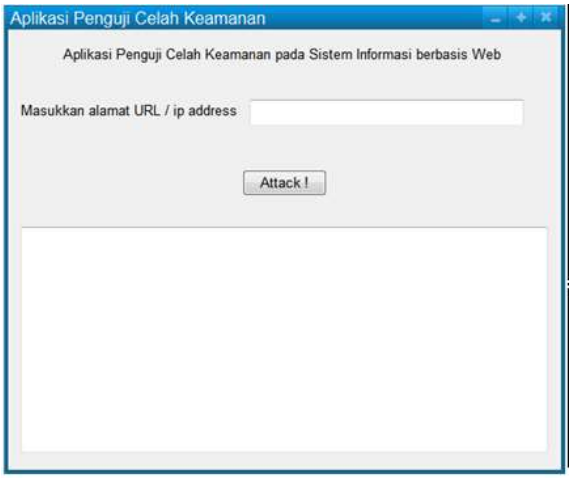
1. Pengujian Phising
2. Pengujian SQL Injection
3. Pengujian Cross-site Scripting

Pada saat awal penggunaan aplikasi hal pertama yang dilakukan oleh aplikasi adalah meminta pengguna untuk memasukkan alamat target, baik berupa alamat URL maupun ip address.

Tahap pertaman yang dilakukan oleh aplikasi adalah pengujian terhadap teknik phising. Aplikasi akan menggunakan teknik crawling untuk mencari informasi tautan yang berada di halaman awal target. Jika hasil crawling dari halaman utama mendapatkan informasi tautan dengan kata “login” maka, aplikasi akan membuka halaman tersebut. Selanjutnya, aplikasi akan melakukan pengecekan terhadap sumber dari gambar yang berada pada halaman tersebut, jika sumber gambar pada halaman tersebut berasal dari berkas yang berada di dalam server dan tidak dapat diakses publik, maka aplikasi akan membaca dan memberikan informasi bahwa aplikasi web target tidak rentan terhadap serangan dengan teknik Phising. Namun, jika sumber gambar pada halaman tersebut berasal dari alamat lain diluar server dan dapat diakses oleh publik, maka aplikasi akan membacadan memberikan informasi bahwa aplikasi web target rentan terhadap serangan Phising. Apabila aplikasi membaca bahwa target rentan terhadap serangan Phising, maka aplikasi akan melakukan duplikasi terhadap halaman yang rentan terhadap serangan phising tersebut secara otomatis ke penyimpanan lokal.

Pengujian terhadap serangan SQLinjection pada aplikasi web dilakukan dengan kembali dimulainya proses crawling. Perbedaanya terletak pada proses pencarian informasi tautan. Informasi tautan yang dicari adalah informasi tautan pada halaman awal target yang mengandung string “id=”. Jika tidak terdapat tautan yang sesuai, maka aplikasi akan membaca dan memberikan infomasi bahwa aplikasi web tersebut tidak rentan terhadap serangan SQL Injection maupun Cross-Site Scripting (XSS). Sedangkan, jika terdapat tautan yang mengandung string “id=”, maka aplikasi tersebut akan menyisipkan tanda petik satu (‘) diakhir tautan tersebut. Setelah dilakukan penyisipan tanda petik satu, aplikasi akan mengunjungi tautan tersebut dan mendeteksi error pada halaman yang dikunjungi. Jika terdapat pesan error SQL pada halaman tersebut, maka aplikasi akan membaca dan memeberikan informasi bahwa aplikasi web tersebut rentan terhadap serangan SQL Injection. Sebaliknya, bila tidak terdeteksi pesan error SQL, maka aplikasi akan membaca dan memberikan informasi bahwa aplikasi web tersebut tidak rentan terhadap serangan SQL Injection.

Pengujian terhadap teknik serangan Cross-Site Scripting dilakukan dengan cara menyisipkan script diakhir tautan yang memiliki kata “id=”. Terdapat 2 hal yang membedakan pengujian ini dengan pengujian SQLInjection. Pada serangan cross-site scripting yang disisipkan pada tautan adalah script dan yang dideteksi oleh aplikasi adalah hasil dari script tersebut berhasil dijalankan atau tidak. Script yang disisipkan adalah script javascript. Apabila script yang disisipkan berhasil berjalan, maka aplikasi akan membaca bahwa aplikasi web tersebut rentan terhadap serangan Cross-Site Scripting. Sedangkan, apabila script yang disisipkan tidak berjalan, maka aplikasi akan membaca bahwa aplikasi web tersebut tidak rentan terhadap serangan Cross-Site Scripting.




Gambar 4 Desain Antarmuka

pada tahap desain antarmuka, atarmuka dirancang dengan bantuan aplikasi Pencil. Aplikasi yang akan dibangun merupakan aplikasi berbasis desktop dengan memanfaatkan 1 jendela saja. Pada jendela tersebut, pengguna diminta untuk memasukkan alamat URL atau ip address target. Setelah memasukkan alamat target, selanjutnya pengguna dapat menekan tombol "Attack !" untuk menjalankan aplikasi tersebut. Hasil akhir dari proses pengujian akan ditampilkan pada textbox yang berada pada bagian bawah jendela aplikasi. Desain antarmuka halaman awal dapat dilihat pada Gambar 4

Selain perancangan antarmuka yang sudah dibuar diatas, dibuat juga pseudocode untuk memudahkan dalam proses pembangunan aplikasi nantinya. Pseudocode yang dibuat dibagi kedalam 3 bagian. Bagian 1 adalah bagian judul aplikasi dan keterangan aplikasi, bagian 2 adalah kamus yang bersisi variabel dan daftar modul yang digunakan, dan bagian terakhir adalah deskripsi algoritma. Pseudocode dari aplikasi yang akan dibangun dapat dilihat pada tabel dibawah ini

Tabel 1 *Pseudocode* Aplikasi

```
Aplikasi: Aplikasi Pengujian
Celah Keamanan pada Aplikasi
Berbasis Web
{Aplikasi yang berfungsi untuk
menguji celah keamanan pada
aplikasi berbasi web dengan
menggunakan jenis serangan SQL
Injection, Phising, dan Cross-
Site Scripting (XSS)}
```

```
Kamus :
alamatURL : string
soup : string
kondisiPhising : string
kondisiSQLXSS : string
kondisiSQL : string
kondisiXSS : string
URL : string
kata : string
cekSumber : string
copasSource : string
BeautifulSoup(response,
'html5lib') : library
get(value):library
urllib.request.open(value) :
library
```

```
Deskripsi Algoritma :
input(alamatURL)
BeautifulSoup(response,
'html5lib') ← soup
soup.find(kata : 'login') ←
kondisiPhising
if (kondisiPhising) then
soup.find(kata : 'src') ←
cekSumber
if (cekSumber == "http") then
soup.find(kata : 'html') ←
copasSource
write (copasSource) in
'index.html'
output("Website rentan terhadap
serangan Phising")
else
 output("Website tidak rentan
terhadap serangan Phising")
else
 output("Website tidak rentan
terhadap serangan Phising")


soup.find(kata : 'id=') ←
kondisiSQLXSS
if (kondisiSQLXSS) then
get('href') ← URL
 urllib.request.open(URL + '%27')
 soup.find(kata : 'SQL syntax;')
← kondisiSQL
if(kondisiSQL)then
 output("Website rentan terhadap
serangan SQL Injection")
 else
 output("Website tidak rentan
terhadap serangan SQL Injection")
urllib.request.open(URL +
'<script>alert(222)</script>')
soup.find(kata : 'alert') ←
kondisiXSS
if(kondisiXSS) then
 output("Website rentan terhadap
serangan XSS")
 else
 output("Website tidak rentan
terhadap serangan XSS")
```

```
else
 output("Website tidak rentan")
```

## 5. KESIMPULAN DAN SARAN

Setelah melakukan analisis dan perancangan maka dapat ditarik adalah dengan proses analisis dan perancangan tersebut di atas pembuatan program nantinya akan lebih mudah, selain itu dengan pemanfaat bahasa Python sebagai media pengembangan aplikasi menyebabkan aplikasi dapat berjalan pada berbagai platform sistem operasi . Adapun saran yang dapat diberikan dari proses perancangan yang sudah dibuat adalah pada bagian deteksi serangan SQL *injection* dapat dideteksi hingga mengecek apakah serangan dapat mengakses basisdata dari aplikasi web target, karena terdapat beberapa aplikasi web yang memberikan informasi kesalahan SQL saat dilakukan pengujian secara manual, namun tidak bisa mendapatkan informasi mengenai basisdata.

## DAFTAR PUSTAKA

Ahuja, M. S., Bal, J. S. and Varnica (2015) 'Web Crawler : Extracting The Web Data', *International Engineering Research Journal (IERJ)*, 1(8), pp. 629–632.

Elu, A. M. (2013) 'RANCANG BANGUN APLIKASI PENDETEKSIAN VULNERABILITY STRUCTURED QUERY LANGUAGE (SQL) INJECTION UNTUK KEAMANAN WEBSITE', *Jurnal Teknologi Informasi*, 7(22), pp. 111–124.

Kie, A., Guo, P. J. and Ernst, M. D. (2009) 'Automatic Creation of SQL Injection and Cross-Site Scripting Attacks', in *Proceeding of International Conference Software Engineering*, pp. 199–209.

Miniwatts Marketing Group (2018) *Internet Top 20 Countries - Internet Users 2018*. Available at: https://www.internetworldstats.com/top20.htm (Accessed: 24 April 2018).

OWASP (2017) *The Ten Most Critical Web Application Security Risks*, *OWASP Top 10*. Available at: https://www.owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf%0Ahttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:OWASP+Top+10+-2010#1.

Passeri, P. (2018) *February 2018 Cyber Attacks Statistics – HACKMAGEDDON, hackmageddon Cyber Attacks Statistics*. Available at: https://www.hackmageddon.com/2018/04/06/february-2018-cyber-attacks-statistics/ (Accessed: 24 April 2018).

Rachmawati, D. (2014) 'Phising Sebagai Salah Satu Bentuk Ancaman Dalam Dunia Cyber', *Jurnal Ilmiah Saintikom*, 13(3), pp. 209–216.

Sophos (2013) *Security Threat Report 2013*, *Sophos Ltd*. doi: 10.1145/358438.349303.