**CyberSecurity dan Forensik Digital** e-ISSN: 2615-8442
Vol. 3, No. 1, Mei 2020, hlm. 39-44

# BANGKOLO: APLIKASI *VULNERABILITY IDENTIFICATION* BERBASIS *HYBRID APPS*

**Dedy Hariyadi[1], Fazlurrahman[2], Hendro Wijayanto[3]**

[1]Universitas Jenderal Achmad Yani Yogyakarta
[2]Komunitas NgeSec Yogyakarta
[3]STMIK Sinar Nusantara
Email: [1]dedy@unjaya.ac.id, [2]fazlurbima@gmail.com, [2]hendrowijayanto.lecturer@sinus.ac.id

**Abstrak**

Keamanan merupakan hal penting dalam sistem maupun jaringan dalam melindungi data informasi. Tingginya tingkat laporang celah keamanan dari Edgescan menunjukkan masih minimnya pengembang sistem dan jaringan dalam hal menutamakan keamanan. *Information System Security Assesment Framework* (ISSAF) merupakan metodologi *penetration testing* yang dikembangkan oleh *Open Information Systems Security Group*. Dalam *framework* tersebut terdiri dari tiga fase, yaitu *Planing and Preparation*, *Assessment* dan *Reporting*, *Clean-up and Destroy Astefacts*. Dalam melakukan *Vulnerabilities Identification* diperlukan *tools* untuk mengetahui potensi celah keamanan dalam bentuk laporan. Ini sangat diperlukan untuk mempermudah analisis, penggunaan dan meminimalisir biaya *pentesting*. Selama ini tools pentesting kebanyakan masih menggunakan model *Command Line Interface* (CLI) sehingga sulit digunakan oleh orang awam. Sehingga diperlukan *tools* berbasis *Graphic User Interface (GUI)*. Dengan pendekatan *Hybrid Apps* dapat dikembangkan aplikasi *pentesting* berbasis *Graphic User Interface* yang memanfaatkan kelebihan teknologi *native* dan web. Bangkolo merupakan aplikasi untuk *pentesting* yang dikembangkan dari framework ISSAF dan pendekatan Hybrid Apps.

**Kata kunci**: *Hybrid Apps, Information Security, Penetration Testing, Security Assessment, Vulnerability Identification*

# *BANGKOLO: VULNERABILITY IDENTIFICATION APPLICATION BASED ON HYBRID APPS*

***Abstract***

*Security is very important in the system and network for data protection. The high level of vulnerability reports from Edgescan shows the still lack of system and network developers in terms of prioritizing security. Information System Security Assessment Framework (ISSAF) is a penetration testing methodology developed by the Open Information Systems Security Group. Where in the framework consists of three phases, namely Planing and Preparation, Assessment and Reporting, Clean-up and Destroy Astefacts. In conducting Vulnerabilities Identification, tools are needed to find out potential security holes in the form of reports. This is very necessary to facilitate analysis, use and minimize the cost of pentesting. So far, most pentesting tools still use the Comman Line Interface (CLI) model, making it difficult for ordinary people to use. So we need tools based on Graphic User Interface (GUI). With the Hybrid Apps approach it can be developed pentesting applications based on Graphic User Interface that utilize the advantages of native technology. Bangkolo is a native pentesting application developed from the ISSAF framework and Hybrid Apps approach.*

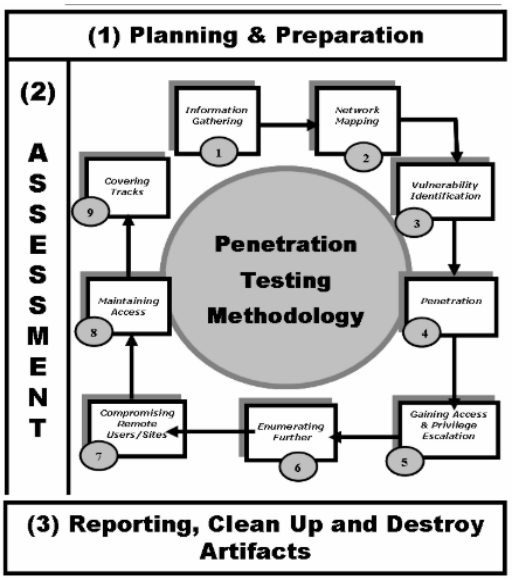
**Keywords**: *Hybrid Apps, Information Security, Penetration Testing, Security Assessment, Vulnerability Identification*

## 1. PENDAHULUAN

Keamanan merupakan prioritas utama sebuah sistem informasi. Beberapa vendor penyedia jasa sistem informasi terkadang sedikit mengesampingkan tingkat keamanannya karena beberapa hal, mulai dari tingginya biaya pembuatan sampai pendeknya waktu pembuatan sebuah sistem. Sehingga tidak dapat dipungkiri kalau beberapa sistem terdapat celah yang berpontensi terjadi serangan. Berdasarkandata statistik *Vulnerability Statistics Report* Tahun 2019 oleh Edgescan, menunjukkan bahwa 19% celah keamanan terjadi di *Application Layer* dan 81% di *Network Layer*. Dari total persentase celah keamanan yang terjadi di *Application layer*, 19% berstatus *High Risk* dan 81% *Critical Risk.*

*Information System Security Assessment Framework* (ISSAF) yang diterbitkan oleh *Open Information Systems Security Group* merupakan sebuah metodologi penetration testing yang bertujuan mengevaluasi sistem komputer dan jaringan pada sebuah organisasi atau institusi. Metodologi ini terbagi menjadi tiga fase. Fase pertama yaitu *Planning and Preparation* merupakan langkah awal *penetration testing* untuk merencanakan dan mempersiapkan tahapan dan hal apa saja yang nantinya akan dilakukan. Fase kedua adalah *Assessment*, pad fase ini dilakukan pengumpulan informasi berkaitan dengan celah keamanan, *penetration* sampai dengan *covering tracks*. Tahapan ketiga yaitu *Reporting, Clean-up, and Destroy Artefacs*, pada tahapan ini sudah menghasilkan laporan tentang celah kemanan pada sistem / jaringan. Pada fase Assessment terdiri 9 tahapan, yaitu, seperti tampak pada Gambar 1 (Open Information Systems Security Group, 2006):

1. Information Gathering.
2. Network Mapping.
3. Vulnerability Identification.
4. Penetration.
5. Gaining Access and Privilege Escalation.
6. Enumerating Further.
7. Compromise Remote Users/Sites.
8. Maintaining Access.
9. Covering Tracks.




Gambar 1. Metodologi ISSAF

Pada tulisan ini fokus pada tahapan *Vulnerability Identification* pada fase *Assessment*. Tahapan *Vulnerability Identification* diperlukan sebuah tools untuk membantu *Penetration Testers/Analyst* mengetahui potensi-potensi celah keamanan dalam bentuk sebuah laporan.

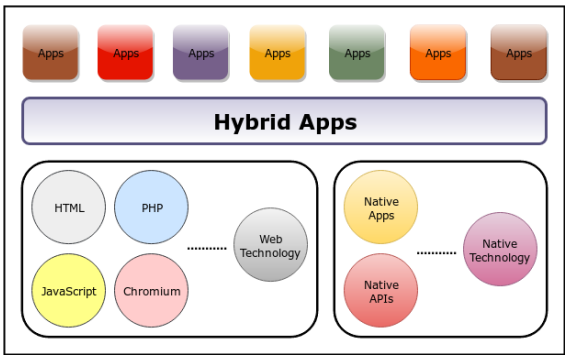
*Network Mapper (Nmap)* merupakan *tools* yang biasanya digunakan pada tahapan *Network Mapping* dengan tujuan diantaranya *network discovery*, pemindaian protokol, deteksi sistem operasi, dan host discovery (Zeeshan et al., 2017). Walaupun Nmap juga menyediakan antarmuka berbasis *Graphical User Interface (GUI)* yang *cross-platform* tetapi fitur-fiturnya tidak seperti pada antarmuka *Command Line Interface (CLI)*. Nmap tidak hanya digunakan pada tahapan *Network Mapping* (Mandal and Jadhav, 2016), bisa juga digunakan pada tahapan *Vulnerability Identification* (Markowsky and Markowsky, 2015). Namun, fitur tersebut masih tersedia pada antarmuka CLI.

Usulan yang disampaikan pada tulisan adalah pengembangan aplikasi *Vulnerability Identification* dengan antarmuka GUI. Pengembangan Aplikasi dengan antarmuka GUI menggunakan pendekatan pengembangan *Hybrid Apps* yang kompatibel dengan sistem operasi GNU/Linux, MS Windows, ataupun Mac OS X. Tujuan pengembangan aplikasi *Vulnerability Identification* mempermudah kinerja *Penetration Testers/Analyst* menggunakan aplikasi yang *low-cost* seperti Nmap dalam menganalisis potensi-potensi celah keamanan.

## 2. TINJAUAN PUSTAKA

### 2.1. *Hybrid Apps*

*Hybrid Apps* merupakan sebuah pendekatan pengembangan aplikasi dengan antarmuka GUI yang memanfaatkan masing-masing kelebihan dari teknologi web dan teknologi native (Hariyadi and Irawan, 2014). Dengan pendekatan *Hybrid Apps* aplikasi yang dibangun berupa aplikasi yang dapat berjalan pada *desktop environment*. Teknologi *hybrid* mengakomodir beberapa bahasa pemrograman untuk dapat dijadikan satu sehingga mampu mengolah data dengan baik dan berjalan secara responsif. Selain itu ditambahkan pendekatan teknologi *native* yang nantinya dapat melakukan proses-luaran sesuai yang diinginkan. Adapun sistem arsitektur *Hybrid Apps* dapat dilihat pada Gambar 2.



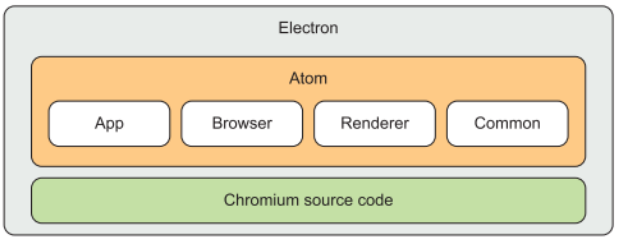

Gambar 2. Arsitektur Hybrid Apps

### 2.2. Electron *Framework*

Pengembang dari Github, Cheng Zhao mengembangkan sebuah *framework* untuk membuat aplikasi desktop yang *cross-platform* menggunakan HTML, CSS, dan JavaScript pada proyek pengembangan aplikasi editor Atom (Cai et al., 2019). Framework yang dikembangkan oleh Cheng Zhao tersebut dikenal sebagai Electron *Framework*,

sebuah *framework* yang memanfaatkan teknologi Chromium, Node.js dan *Native Apps* untuk membuat aplikasi desktop yang *cross-platform*. Komponen kode-kode pengembangan berbasis Electron *Framework* terdiri dari, dapat dilihat pada Gambar 3 (Jensen, 2017):

1. App, direktori yang berisi berkas dengan kode C untuk menangani kode yang perlu dimuat pada awal Elektron dijalankan.
2. Browser, direktori yang berisi berkas untuk menangani interaksi dengan bagian antarmuka dari aplikasi
3. Renderer, direktori yang berisi berkas untuk menangani kode yang berjalan dalam proses renderer Electron.
4. Common, direktori yang berisi berkas dengan kode utilitas yang digunakan oleh proses utama dan renderer untuk menjalankan aplikasi.



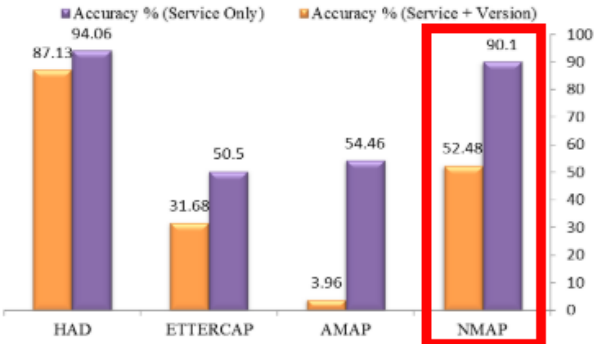

Gambar 3. Arsitektur Electron *Framework*

### 2.3. *Vulnerability Identification*

*Vulnerability identification* merupakan tahapan asesmen pada *Information System Security Assesment Framework* (ISSAF) yang melakukan analisis potensi celah keamanan sebuah sistem / jaringan. *Tools* yang membantu *Penetration Testers/Analyst* untuk mengetahui potensi-potensi celah keamanan sangat diperlukan pada tahapan *Vulnerability Identification* (Zeeshan et al., 2017). Melalui *tools Vulnerability Identification* dapat diketahui beberapa potensi-potensi celah keamanan dari penyedia databases celah keamanan, seperti *Common Vulnerabilities and Exposures* (CVE), *SecurityFocus*, *Computer Emergency Response Team* (CERT), OpenVAS, OSVDB, *Vulnerability Database*, dan IBM X-Force. Hasil laporan dari *tools Vulnerability Identification* dapat menentukan langkah-langkah yang tepat untuk melakukan tahapan selanjutnya yaitu *Penetration*.

### 2.4. *Network Mapper (NMAP)*

Aplikasi *Network Mapper* atau dikenal sebagai Nmap merupakan aplikasi pemindai keamanan yang memiliki *database* yang berisi sekitar 2200 *well-known services*. Sebagai contoh Nmap mendeteksi port 22/TCP, 110/TCP, dan 25/TCP maka hasil dari pemindaian dapat disimpulkan mesin menjalankan layanan SSH Server, POP3 Server dan SMTP Server. Nmap dapat memindai tipe layanan saja dengan akurasi 90,1% sedangkan pemindaian tipe layanan dan versinya memiliki akurasi 52,48%, seperti tampak pada Gambar 4 (Ghanem and Belaton, 2013).




Gambar 4. Komparasi Nmap dengan *Tools* Lainnya

Hasil atau luaran dari proses pemindaian dari Nmap baik berbasis GUI atau CLI hasilnya sama. Gambar 5 merupakan contoh hasil pemindaian menggunakan Nmap pada localhost. Nmap memiliki fitur yang bagus untuk mempermudah kinerja *Penetration Testers/Analyst* dalam menganalisis celah keamanan yang berpotensi. Fitur ini disebut *Nmap Scripting Engine* (NSE) yang memperbolehkan pembuatan skrip yang disesuaikan dengan kebutuhan. NSE dirancang supaya fleksibel dalam menangani hal-hal seperti *Network Discovery*, deteksi yang lebih canggih, deteksi celah keamanan, deteksi *Backdoor*, dan ekploitasi celah keamanan (Gordon Lyon, 2011). Pada penelitian ini menggunakan NSE untuk mendeteksi celah keamanan.

```
Starting Nmap 7.60 ( https://nmap.org ) at 2019-10-07
18:58 WIB
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000097s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE
80/tcp    open  http
631/tcp   open  ipp
1234/tcp  open  hotline
3306/tcp  open  mysql
5432/tcp  open  postgresql
50002/tcp open  iiimsf

Nmap done: 1 IP address (1 host up) scanned in 0.06
seconds
```

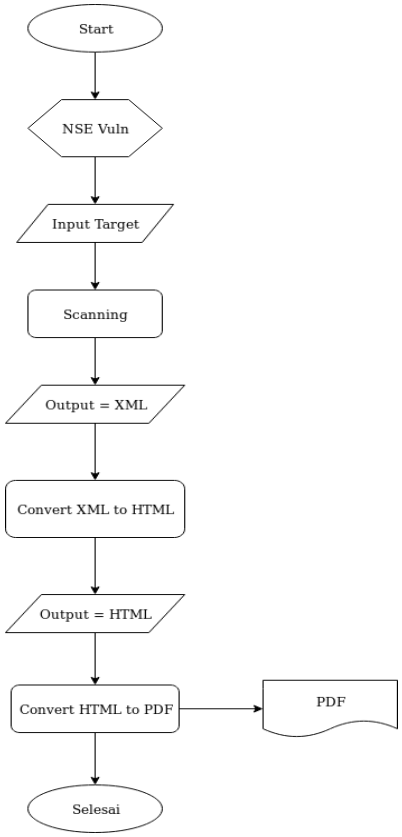
Gambar 5. Hasil Pemindaian NMap

## 3. RANCANGAN APLIKASI

Pada prinsipnya *Network Mapping* (Nmap) telah menyediakan NSE dengan skrip yang dapat disesuaikan dengan kebutuhan *Penetration Testers/Analyst*. Namun, penelitian ini menggunakan NSE *Vulnerabilities* yang telah disediakan oleh Nmap. Saat ini Nmap belum menyediakan laporan berupa berkas dengan format PDF. Laporan dalam format PDF ini memudahkan pengguna dalam melakukan asesmen pada sistem dan jaringan. Maka pada penelitian ini dikembangkan aplikasi *Vulnerability Identification* menggunakan Nmap

dengan fitur tambahan berupa laporan dalam format berkas PDF yang mudah dibaca dan disajikan dalam dokumen laporan *Security Assessment*.

Nmap secara umum menghasilkan laporan berupa berkas berformat XML. Pada penelitian ini terdapat dua konversi yaitu konversi XML ke HTML dan konversi HTML ke PDF. Adapun *flowchart* dari rancangan pengembangan aplikasi dapat dilihat pada Gambar 6.




Gambar 6. *Flowchart* Pengembangan Bangkolo

## 4. HASIL DAN PEMBAHASAN

Persiapan sebelum melakukan pengembangan aplikasi *Vulnerability Identification* memasang beberapa aplikasi pendukung diantaranya:

1. Electron *Framework*, dasar pengembangan aplikasi berbasis Hybrid Apps.
2. PHP, skrip interpreter untuk menjalankan aplikasi berbasis web.
3. Nmap, pemindai keamanan sistem informasi.
4. xsltproc, aplikasi yang berfungsi mengkonversi berkas XML ke HTML.
5. wkhtmltopdf, aplikasi yang berfungsi mengkonversi berkas HTML ke PDF.

Pada sisi teknologi web, Electron *Framework* menggunakan PHPServer sebagai web server. Jika pada sistem operasi MS Windows PHP harus dikonfigurasi *path environment*-nya, sedangkan pada GNU/Linux dan Mac OS X telah menyesuaikan saat proses installasi. Berikut cuplikan kode runphp.js yang berfungsi menjalankan PHPServer.

```
1  const PHPServer = require('php-
   server-manager');
2  const server = new PHPServer({
3      port: 3000,
4      directives: {
5          display_errors: 1,
6          expose_php: 1
7      }
8  });
```

Berikut baris kode utama aplikasi *Vulnerability Identification* yang tersimpan pada berkas index.php. Baris pertama dan baris kesebelas menunjukan kode tersebut adalah kode dengan kaidah PHP. Baris keempat memproses masukan dari bari kedua yang berisi IP atau host dari target yang akan dipindai dengan output yang disimpan di direktori output. Baris kelima mengkonversi output dari baris keempat berupa XML ke HTML. Baris keenam mengkonversi berkas HTML ke PDF.

```
1  <?php
2   if (isset($_POST["ip"]))
3   {
4   $com = "/usr/bin/nmap --script=vuln
   -oX output/output.xml ".
   $_POST["ip"];
5   $tml = "/usr/bin/xsltproc
   output/output.xml -o
   output/output.html";
6    $tpdf = "/usr/bin/wkhtmltopdf
   output/output.html
   output/output.pdf";
7   $result = shell_exec($com);
8   $result = shell_exec($tml);
9   $result = shell_exec($tpdf);
10   }
11 ?>
```

Aplikasi *Vulnerability Identification* yang diberi nama Bangkolo ini, memiliki fitur melakukan pemindaian menggunakan skrip NSE Vuln dan mengkonversi luaran menjadi berkas PDF. NSE Vuln merupakan skrip bawaan dari Nmap. Bangkolo masih memiliki kekurangan diantaranya belum memanfaatkan atau pemilihan fitur dengan NSE *Vulnerability* lainnya yang berdasarkan penyedia basisdata celah keamananan yang bersifat publik secara dinamis. Gambar 7 merupakan tampilan antarmuka dari aplikasi *Vulnerability Identification*, Bangkolo.

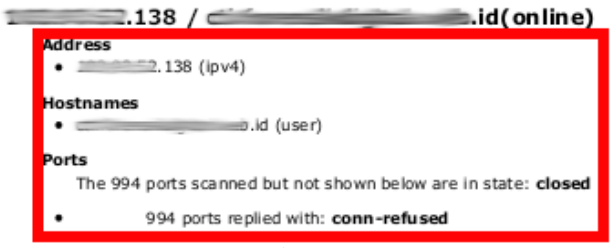

Gambar 7. Antarmuka Bangkolo

Setelah proses pemindaian selesai maka proses selanjutnya adalah proses pembuatan laporan *vulnerability identification*. Pada laporan ini terbagi menjadi 2 bagian utama, *scan summary* dan hasil pemindaian. *Scan summary* berisi informasi, seperti tampak pada Gambar 8 :

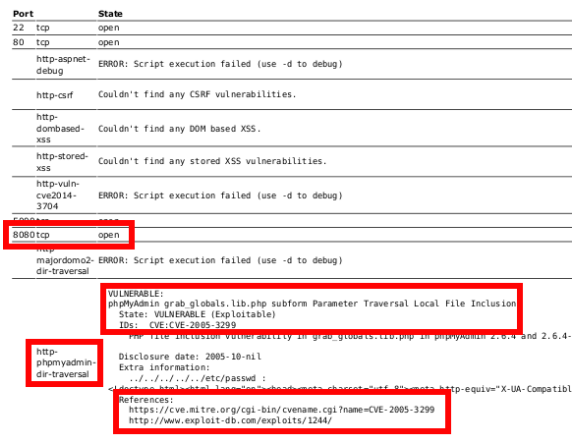
1. Waktu memulai pemindaian.
2. Perintah pemindaian.
3. *Reporting type*.
4. Informasi waktu yang dibutuhkan untuk menyelesaikan pemindaian.


Gambar 8. *Summary Scan*

Pada bagian hasil pemindaian terdapat 3 bagian, yaitu IP target, *hostname target*, dan port yang terbuka, dapat dilihat pada Gambar 9. Bagian port yang terbuka disajikan laporan terkait celah keamanan yang merujuk dari sebuah penyedia basis data celah keamanan (NSE) seperti tampak pada Gambar 10.




Gambar 9. Informasi Port Terbuka




Gambar 10. Informasi Celah Keamanan dari NSE Vuln

## 5. KESIMPULAN

Menggunakan Electron *Framework* dapat digunakan untuk mengembangan aplikasi *Vulnerability Identification* atau aplikasi pebgujian keamanan informasi lainya yang masih memiliki antarmuka CLI. Aplikasi Bangkolo dapat diunduh pada situs pengembangnya yaitu https://github.com/orangmiliter/bangkolo. Bangkolo masih terdapat kekurangan dan perlu pengembangan terkait pemanfaatan NSE Vulnerability yang bersifat lebih fleksibel. Kekurangan lainya yaitu luaran berupa berkas PDF belum bersifat dinamis dalam penyimpanan berkasnya. Misal seperti penamaannya menggunakan *timestamp* atau obyek. Harapannya kekurang-kekurang tersebut dapat diperbaiki pada pengembangan atau penelitian selanjutnya.

## DAFTAR PUSTAKA

CAI, R., RAO, Y., WANG, J., GUAN, H., SHI, X. AND WANG, Y., 2019. NetPadBrowser : An Offline Browser for Web-Based Dynamic Geometric Resources. In: 2019 14th International Conference on Computer Science & Education (ICCSE). IEEE.pp.434–438.

GHANEM, W.A.H.M. and BELATON, B., 2013. Improving Accuracy of Applications Fingerprinting on Local Networks using NMAP-AMAP-ETTERCAP as a Hybrid Framework. In: Proceedings - 2013 IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2013. IEEE.pp.403–407.

GORDON LYON, 2011. Nmap Network Scanning. [online] Available at: <https://nmap.org/book> [Accessed 6 Jul. 2018].

HARIYADI, D. and IRAWAN, E.T., 2014. Purwarupa Forensik BBM di Telepon Seluler Android Menggunakan IGN-SDK. Indonesia Security Conference 2014, .

JENSEN, P.B., 2017. Cross-Platform Desktop Applications Using Electron and NW.js. New York: Manning Publications Co.

MANDAL, N. and JADHAV, S., 2016. A Survey on Network Security Tools for Open Source. 2016 IEEE International Conference on Current Trends in Advanced Computing, ICCTAC 2016, pp.1–6.

MARKOWSKY, L. and MARKOWSKY, G., 2015. Scanning for Vulnerable Devices in the Internet of Things. In: Proceedings of the 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2015. pp.463–467.

OPEN INFORMATION SYSTEMS SECURITY GROUP, 2006. Information Systems Security Assessment Framework (ISSAF). Draft 0.2. ed.

ZEESHAN, M., NISA, S.U., MAJEED, T., NASIR, N. AND ANAYAT, S., 2017. Vulnerability Assessment and Penetration Testing: A

proactive approach towards Network and Information Security. International Journal of Digital Information and Wireless Communications, 7(2), pp.124–142.