Monitoring Government Web Portals in Cameroon

Paul Dayang, Theodore Efoumena and Franklin Tchakounte
The University of Ngaoundéré
Faculty of Science, Department of Mathematics and Computer Science
P.O. Box 454 Ngaoundéré, Cameroon
pdayang@univ-ndere.cm, efoumenatheodore@yahoo.fr and tchafros@gmail.com

ABSTRACT

This research aims to analyse the most frequent vulnerabilities of Cameroonian governmental web portals and to suggest a suitable monitoring infrastructure. In fact, the technological evolution and the need to intensify the local communication enabled companies and governmental institutions to set up web portals for e-governance, e-education, e-advertising, e-payments, e-registration, e-commerce etc. These e-services manipulate quite sensitive data of individuals and governmental institutions. Attackers are therefore attracted to launch malicious actions to take control over these web systems. After analysing several Cameroonian web portals, one of our main findings is that, unfortunately, administrators of such systems lack expertise and even are not aware of the use of diverse systems for monitoring and securing their infrastructures, exposing therefore the life of users. As a consequence, several attacks perpetrated against web portals are increasingly being registered since 2012. We have analysed 55 web portals. Three groups of vulnerabilities have been identified and categorised. As a solution approach, we propose a distributed monitoring architecture for national web portals. The architecture aims to gather automatically information and analyse them for assistance in security decision-making. This work constitutes a considerable step towards preventive measures for strengthening cybersecurity in Cameroon.

KEYWORDS
Web portal, intrusion testing, vulnerabilities, web monitoring, web site, Cameroon.

1 INTRODUCTION

The development of computer applications is an ongoing activity. With the technological evolution, we are witnessing the emergence of new programming languages and the software development companies are changing their way of proceeding, looking for new ways of programming that facilitate more and more work. In the range of the products being developed, we have software systems and application software which encompass web portals which are more or less easy to design and implement.

Unlike desktop software that performs specific tasks of a company’s business, web portals are nowadays developed for processing serious transactions and business.

With the expansion and appearance of web portals since the 2000s, the security problem starts arising more and more with acuity in Cameroon [1]. We have in mind the hacking of the web portal of the Cameroonian Customs in November 2008 [2], the portal of the former Cameroonian Ministry of Domains and Land Affairs on 18 May 2008 [2], that of the University of Yaoundé I in 2009 [2], [3], that of the daily newspaper La Nouvelle Expression on 14 September 2008, the former web portal of the National Assembly of Cameroon on 22 September 2014 [4] and the web portal of the Presidency of the Republic of Cameroon on 11 March 2015 [5]. These attacks enable accesses to confidential information and proliferation of phenomena such as espionage and fraud.

The main objective of a hacker is to hinder the proper functioning of an application, i.e. the confidentiality of the stored information, the availability of the application and the integrity of the data exchanged without having any intention of stealing information. To achieve this goal, there are several methods of attack and tools for accomplishing these malicious activities. Among these methods, intrusions, code injections and redirection of users to destinations other than those desired are among the most common and used attacks. While attacking an application, the
ultimate goal of the malicious user is to get access illegally to a system without being identified. In regard to the security incidents encountered, it is clear that web portals, mostly governmental, face the illegal intrusion problem. We may ask the following question: Is it possible to propose a web server monitoring architecture allowing us to monitor upstream web portals in Cameroon in order to identify and block subsequently the illegal access and devastating intrusions? The study of various web portals of the governmental institutions and the associated structures in this work is on an experimental basis and therefore the objective is not to cause any damage or harm to the institutions. We proposed to conduct tests related to some governmental web portals because we would also like to evaluate their vulnerability level with regard of hacker penetration. The goal is to sensitise decision-makers about security practices associated to their respective web portals.

The first section of this work presents the state of the art on the intrusion test and supervision tools, furthermore we will focus on the Zero Entry Hacking (ZEH) method and present the results of the conducted experiment. Before problems affect critical business processes, it is important to protect IT infrastructure. Finally, we make a proposal of an architecture for monitoring of web portals in Cameroon based on a selected monitoring tool.

2 INTRUSION TESTING AND MONITORING TOOLS

The intrusion test or pentest is an authorised attempt to simulate the activities of a hacker who wishes to get access to unauthorised resources or hinder the proper functioning of an information system, for example by making it unavailable [6]. The general idea in this operation is to find the security issues using the same tools and techniques as those used by the hackers. The discovered vulnerabilities will be corrected before a real hacker exploits them. The intrusion test can be used to help secure various types of targets, such as an information system in its entirety, a specific subset of an information system (one or more subnets, a group of machines, etc.) or a particular application. This operation may be legal, assuming that the alleged hacker has a test authorisation from the responsible of the infrastructure to be tested. Thus, as far as this operation relies on attack actions of making systems unstable, it can also provide solutions for repairing and preventing attacks. As objectives of an intrusion test in an information system we can cite [7], [8], [9].

- List a set of information found, which may be sensitive or critical;
- Identify vulnerabilities or weaknesses in the security system that can be exploited;
- Demonstrate that an attacking potential is able to find vulnerabilities and exploit them to enter the system.
- Test the effectiveness of the intrusion detecting systems and the responsiveness of the security team and sometimes also the users;
- Report and make a final presentation of its progress and discoveries to the client;
- Provide guidance and advices on methods for resolving and correcting the discovered vulnerabilities.

To achieve these objectives, the intrusion test team according to the conditions defined in the contract of employment must make a choice and respect the procedure of the types of test to be carried out. The types of intrusion test are defined according to the nature and the domain of the elements to be tested [10]. Generally, there are six types of tests:

1. Network intrusion test;
2. application intrusion test;
3. websites intrusion test;
4. physical intrusion test;
5. Cloud intrusion test;
6. social engineering intrusion test.

2.1 Approaches for Testing of Web Portals

Since our target is constituted of web portals, we will focus on approaches testing such type of applications [10].
2.1.1 Black Hat or Black Box Mode

In the Black Box context, the tester is really in the position of an external attacker and begins his intrusion test with as little information as possible about his target. This test aims to find and demonstrate the presence of an action plan that can be exploited by an external person to take control of the system or to get hold of certain information. In this operation, the hacker has no authorisation from the target company.

2.1.2 White Hat or White Box Mode

Here, the ethical tester working in close collaboration with the Director of the Information System (DSI), the Responsible of the Security of Information Systems (RSSI), and the technical team of the system of information. The aim is then to obtain 100% control of the information on the system and of the accompany as the DSI or RSSI in the detection of vulnerabilities. One of the advantages of White Box mode is that you can then detect security flaws more widely than those the Black Box mode would not have detected.

2.1.3. Mode Gray Box or Gray Box

In this mode, the tester starts its intrusion test with a limited number of information, being for example in the position of a user of the IS. It will start with a computer in the system, a valid user account and an access to certain applications. Here, one can simulate the harm that a user can cause to system of its own company from a precise position in the information system.

2.1.4. Red Team Approach

This approach aims to test more thoroughly the detection and response capabilities of the defence team (Blue Team) and the system protection put in place. The goal of the attacking team is to trace a path that someone can take from outside the company to the completion of one or more critical actions within the targeted system. This approach aims to describe the technically advanced attacks and generally carried out by organisations or very large companies. They are carried out over a long period of time, unlike the others which are carried out over a short period. To perform any of the above tests, there are defined methodologies.

2.2. Testing Methodologies

A methodology aims to organise in different stages the techniques and the technologies allowing to solve in a rational way a problem. Therefore, we will review some important methodologies.

2.2.1 Penetration Testing Execution Standard (PTES)

The Penetration Testing Execution Standard (PTES) is a standard created to provide a common framework for companies and security teams helping them to perform intrusion tests. The PTES is divided into seven phases namely [7], [8], [11].

- Pre-engagement Interactions: This step consists of establishing a work contract;

- Intelligence Gathering: During this phase information is collected;

- Threat Modelling: Here we make a list of threats for the company;

- Vulnerability Analysis: This step involves searching for exploitable vulnerabilities;

- Exploitation: The main objective is to try to test the vulnerabilities found in the previous phase and to progress in the intrusion of the system.

- Post Exploitation: This step includes several interesting phases, acting as a real attacker, the tester will have to erase his tracks so that these actions remain as discrete as possible;

- Reporting: Here we explain the progress of the intrusion test and its fruit to the company that ordered it.

2.2.2 Open Source Security Testing Methodology Manual (OSSTMM)

Its interest is to describe a set of operational tests for the security evaluation of the system, network, human, Wi-Fi, etc. at the physical level. This method is global and therefore not limited to an aspect. It is organised along the following steps [6]:

---

International Journal of Cyber-Security and Digital Forensics (IJCSDF) 6(3): 109-120
The Society of Digital Information and Wireless Communications (SDIWC), 2017 ISSN: 2305-0012

111
1. Intrusion test;
2. Safety assessment;
3. Vulnerability assessment;
4. Red Team;
5. Blue Team.

2.2.3 The Open Web Application Security Project (OWASP) Testing Guide.

It's about a very comprehensive document for intrusion tester and also for the developers of applications and websites. Beyond an intrusion test methodology, it is more a list of different security points to be checked by developers of web applications and testers as well. The particularity of this method is that it can be applied throughout the life cycle of an application. The Testing Guider version 4.0 presents in detail the stages of the life cycle of a web application, the testing points, and if necessary the tests and tools to be used.

2.2.4 Zero Entry Hacking (ZEH) Method

Zero Entry Hacking [9] is a methodology used by Hackers to get into networks or web portals. Generally, it is carried out during the implementation phase of an information system. It put special emphasis on the phase of reconnaissance by proposing an approach and the stage of social engineering. There is no work commitment between the parties. It gives for each phase, the necessary tools and advices. This method which is used in our work, is based on the following key points:

**Reconnaissance:** It is subdivided into the stages of active reconnaissance, passive reconnaissance and social engineering.

**Scanning:** It is divided into four stages: ping and ping scanning, port scanning with Nmap, using the Nmap script engine and finally scanning vulnerabilities.

**Exploitation:** It can be summed up as network operation, web exploitation and social engineering.

**Operation:** During this phase, tester rely on the social engineering approach to get access to the network, the Web and to maintain the established access.

In order to have an overall view of the functioning information system, enterprise usually install some supervisory tools for analysing some important data. Different data or files can be analysed.

- Analysis of log files;
- Retrieval and analysis of results from local or remote commands and scripts;
- Use the Simple Network Management Protocol (SNMP).

ZEH has two main objectives: failure prevention and recovery after an incident [12], [13]. The existing solutions in the supervision market are divided into two main categories: proprietary solutions and free solutions. The proprietary solutions are intended for large companies, small and medium-sized enterprises and solutions based on Software as a Service (SaaS, known as part of Cloud Computing) are part of this group. The second category of solutions is from the world of free and open source software solutions which encompass tools such as Nagios, Zabbix, Zenoss, Shinken, Cacti (graphs) and Eyes of Network. The advantages of free and open source are numerous, for instance, the availability of source code and the freedom to study and modify the code accordingly and to disseminate it [14]. After comparing these solutions based on the following criteria: user interface, functionalities and architectures, Nagios appeared to be the most widely used and appropriate for our analysis.

3 METHODOLOGY

The methodology applied to our work consists of experimenting the ZEH method and describing the results. The active reconnaissance aims to hack journals, forums, search engines for getting information on the targets. In our case study, we worked with a list of 55 targets or web portals.

The reconnaissance phase is the first step and the basic element of the ZEH method. Its main objective is to focus on a list of web portals and the corresponding IP addresses of the targets.
The results presented in this phase are based on two main points which are the passive reconnaissance and the active reconnaissance.

The active reconnaissance consisted of analysing the copies (HTTrack) of the web portals of the targets, for that purpose we used the Neteract tool during four days to exploring the targets and we include also Google operators.

Figure 1 shows a part of the results provided by the Neteract tool while applying it on the web portal of the Ministry of Public Service and Administrative Reform of Cameroon.

In these results we obtained, two important details are displayed and interesting for attackers: the email address of the administrator and the server version hosting the web portal. The first one can be used to conduct attack by social engineering [15], [16]. The problem linked the server version is that an experienced hacker may use the server vulnerability of its given version to attack the system.

On the login web page of the web portal of the digital training environment of the University of Yaounde 1 (visited on 17 July 2016), the error message shown indicates that the login field is the one being incorrectly filled in. In this situation, the error message must be a generic one and must not give any information to the wrong field.

Furthermore, six web portals allow users to browse a portion of their directory tree at the server level. The danger with this practice is that one can forget files containing sensitive data in these directories. We have the directory of the web portal of the Ministry of Transport (visited on 17 July 2016), which gives the possibility to users to access.sql files which may deliver very crucial on databases.

After collecting the data during the reconnaissance phase, we performed the scan phase. It aims to link with each target the services associated with each of its ports and to perform the vulnerability scan defined in the Open Source Vulnerability Data Base (OSVDB) [17], [18]. This phase is divided into three major stages:

- **Scanning targets** with Ping scanning: the step is to list the active targets. A target that does not respond to this request is classified as inaccessible;
- **Port scanning** for assigning the associated services to each port;
- **Vulnerability scan** for looking for universally recognised vulnerabilities defined in the OSVDB.

We used two tools Nmap and Nikto for port scanning. All the vulnerabilities identified during this phase were classified into three categories:

- **Configuration error**: it groups all the vulnerabilities resulting from the configurations errors of the components of the web server;
- **Programming error**: this category includes vulnerabilities related to poor
programming practices;

- **Use of failed Components**: this category includes all vulnerabilities resulting from the use of components which are obsolete and presenting certain vulnerabilities.

### Table 1. Categories of found vulnerabilities

<table>
<thead>
<tr>
<th>Category 1: Configuration error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identified vulnerabilities:</td>
</tr>
<tr>
<td>- OSVDB-3092: phpMyAdmin is not protected or limited to authorized hosts.</td>
</tr>
<tr>
<td>- OSVDB-561: Apache information are available.</td>
</tr>
<tr>
<td>- OSVDB-877: HTTP TRACE method is active, therefore vulnerable to XST.</td>
</tr>
<tr>
<td>- OSVDB-3268: Directory indexing found.</td>
</tr>
<tr>
<td>- OSVDB-637: Enumeration of users is possible.</td>
</tr>
<tr>
<td>- OSVDB-3233: Apache default file found</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category 2: Programming error</th>
</tr>
</thead>
<tbody>
<tr>
<td>- OSVDB-27071: PHP Image View 1.0 is vulnerable to XSS.</td>
</tr>
<tr>
<td>- OSVDB-12184: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category 3: Using Failed Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>- OSVDB-682: Webalizer may be installed. Versions lower than 2.01-09 vulnerable to XSS.</td>
</tr>
</tbody>
</table>

Throughout our work, we noted various shortcomings which may be due to negligence.

- The lack of regular updating of certain web portals: for example, we have the Prime Minister's services portal which lists www.assembleenationale.cm as the portal of the national assembly instead of www.assnat.cm;
- The web portal of the Telecommunications Regulatory Agency is out of service: www.art.cm;
- Some web portals are offline on weekends;
- The portal of the Ministry of Communication was inaccessible during days: www.mincom.gov.cm;
- The portal of the Ministry of Communication was inaccessible during days: www.mincom.gov.cm;
- Some web portals are hardly accessible the day after some appointments;
- The collection of data was generally effective on weekends because the portals were less solicited;
- The ability to obtain relevant information from the server of each portal using some tools.

After the scan phase, it was a question of carrying out the phase of web exploitation and maintenance of access. In respect to the regulations and laws on cyber criminality in Cameroon and without any official authorisation we could not push deeper this phase because it can harm the functioning of the system put under exploitation.

### Table 2. Summary of Intrusion Test Results Using the ZEH Method

<table>
<thead>
<tr>
<th>Operation</th>
<th>Targets</th>
<th>Successful</th>
<th>Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Test</td>
<td>55</td>
<td>53</td>
<td>From 55 web portals two (2) are unavailable:</td>
</tr>
<tr>
<td>Testing with Googles Operators</td>
<td>53</td>
<td>6</td>
<td>From 53 web portals six (6) have vulnerabilities</td>
</tr>
<tr>
<td>Accessing server-specific data</td>
<td>53</td>
<td>51</td>
<td>From 53 web portals, only two protect their server against scanning with NETCRAFT</td>
</tr>
<tr>
<td>Port Scan</td>
<td>53</td>
<td>51</td>
<td>From 53 web portals, only two (2) protect their server against the scan of the ports</td>
</tr>
<tr>
<td>Scan</td>
<td>53</td>
<td>12</td>
<td>Of the 53 web portals, twelve (12) are vulnerable.</td>
</tr>
</tbody>
</table>
Web exploitation includes all attacks on web portals that are intended to interfere with the proper functioning, integrity, availability and confidentiality of a web portal and the data it contains.

The basic idea is to use the browser in the same way as browsing tool over the web portal, but all traffic is sent through a proxy. In this way, we can collect and analyse all the queries, as well as the answers provided by the web portal. The operations carried out during this phase are: SQL injections, Cross-Site Scripting (XSS) attacks and query interception. In order to have an overall knowledge of the operations carried out, Table 2 summarises the tests carried out and their importance.

4 PROPOSAL OF A MONITORING ARCHITECTURE
A web portal is a tool for popularising the activities of a given structure. It can be used as a tool for marketing, communication, work, consulting and even sometimes for doing business like for product sale. In Cameroon, the web portals of governmental institutions, universities and hospitals are mainly information portals. They present the activities of the structure, the tasks of the different persons in charge of these structures, the functioning of the structure, the texts and decrees issued or signed by the structure, the personnel and geographical location of the structure etc.

Apart from the informative service, there are certain portals, which provide real online services for recording, modifying and consulting important processes. We have, for example, the web portals of certain academic institutions that provide online pre-registration: The University of Yaoundé I and the University Douala. Furthermore, we have the online process of awarding public contracts, the online process of choosing and assigning regions to new school teachers at the Ministry of Secondary Education, the online process for creating enterprises or declaring taxes and many others.

In our analysis, we have decided to group the collected web portals into three categories: the academic institutions, hospitals, and ministerial and related structures.

We studied the data collected from the web portals at the level of each monitoring server in order to be able to trigger alerts, or to suspend the access to the servers to certain corrupt clients. This involves setting up a supervisory system coupled with a web server log file analyser, which, thanks to the various features it offers, can block client systems that can cause problems to the web servers. In addition, it should prevent failures by following the actions of monitoring system by controlling the status of the servers and associated services: CPU load, disk space, available memory, number of connected clients etc.

Since the web servers that host the portals are not all in the same network, we need at a first level to place an agent of monitoring tool on each server, then at a second level to group the web portals in category and to place at the level of each category another program of the same supervision tool, which analyses and presents the data collected by the tools at the first level (supervision server: client). Thus, one will obtain a structure in tree where the sheets are servers that host the web portals and the internal nodes of the central control are the supervision tools [20], [21].

Figure 3 shows an architectural proposal of a generalised solution. In its implementation, it will be extended only to all web portals hosted in Cameroon.

(1) Represent the server part of the architecture in which the server of the
monitoring tool will be installed.

(2) Represents the flow of information packages between the corresponding server and the clients. The packages contain the report of events that took place on the server and at the agent level.

(3) Represents the client part of the architecture on which the agent that will execute diverse commands of the supervision server will be installed.

In this architecture, we perform a collection of monitoring information either periodically or at the request of the monitoring server sent to each web server. Each agent placed at the server level of a portal has the IP address of its supervisor server with which it communicates. Each monitoring server maintains a list of these different agents. The data collection is coordinated by level two supervision.

The information from the agents is transferred to each level of the supervision server of its category. Thus, the supervisory administrator has the possibility to have in real time all the movements that occur in all the servers of a given category. It may therefore decide to block a server or service access to a given client or to suspend a client that saturates a server with queries. Thus, an illegal operation that escapes the vigilance of the web server administrator can be overtaken and corrected by the central supervisor. Implementing this architecture requires a convenient choice of supervisory tools to be used and configured.

The previous solution gives a general overview of the monitoring architecture without any specification of the tool and the monitoring architecture to be used and how it will be implemented. Nagios has been chosen in order to implement the proposed architecture. One of the main advantages provided by this tool is the huge number of existing plugins [22]. After the choice of Nagios as the main tool, the implementation of this solution must respect tool architecture and the following modalities:

- On each second level, supervisory cell grouping the servers of the same category, a Nagios XI server must be installed;
• The *NDOUtils* component must also be installed on each second level supervisory machine of the supervisory unit;

• Still on the same machines, the *Centreon* component must be installed at the second level.

• For the supervision of Windows servers, the client part of *NSClient* named *check_nt* must be installed on all Nagios servers;

• For the monitoring of Linux servers, one must check if the *NRPEclient* part named *check_nrpe* on all the Nagios servers is installed;

• Install the *NSClient* server part named *NSClient ++* on all Windows servers;

• Install the *NRPE* server part on all Linux servers.

This configuration will allow us to scan the log files of each web server for traces of vulnerability scanners. This operation will be carried out continuously on each web server.

Figure 4 shows the operating architecture of a second level supervision unit of the control centre. Its operating mode is described as follows:

1. The *NRPE* and *NSClient ++* modules execute the commands of the monitoring server and return the results. These results are saved in files.
2. The server's *NDOMOD* program retrieves the information from the files and places them in the Socket of *NDutils*.
3. The *NDO2DB* module of *NDutils* extracts the data from the socket to an external MySQL database.
4. The data saved in the MySQL Database are extracted, processed by *Centstorage* which is the processing daemon of *Centreon* and stored them in other secondary databases such Database Design Report (DDR).

**Figure 4:** Operating Architecture of a Level II Supervisory Machine
5. The *Centstorage* daemon automatically updates the secondary databases (DDR) as soon as the backup files are modified.

6. These processed data will be used to display the results as graphs. This module deals also with the configurations on the tool Nagios.

7. *Centreon’s Centcore daemon* will allow this architecture to communicate together, taking care of the transmission of data between the central server and its dedicated servers. It is also responsible for deploying the Centreon configuration towards Nagios.

Among the supervision architectures proposed by Nagios Enterprises [23], [24], the federated architecture is the best suited to our case. It provides an overall viewing element of the supervisory system by adding a central server to the system that receives information from the dedicated servers. But configurations are always performed at each level of the dedicated server and on the central server.

Figure 5 presents the overall and complete architecture of the operation of the control centre. In this architecture, we have installed on each cell of the control centre the components necessary for its operation. And on each server, depending on the running operating system, we installed the corresponding plugins [25].

In view of the analysis of the observations made and the problems encountered during our work, we have put together a list of recommendations to be followed by an administrator of a supervision system on the one hand and by a web portal administrator on the other:

**Figure 5**: Complete Supervision Architecture
1. Secure access to the different directories of the web portal server;
2. Prohibit clients from accessing different information about the servers (version, administrator etc.) and in particular accessing configuration files;
3. Always know and get documentation of a component before installing and using it;
4. Avoid using components which are obsolete and vulnerable to attacks;
5. Never trust user data but check them before storing (web portal administrator);
6. Always update the used components for web portals;
7. Never allow a user attempting to connect to multiple ports of a monitoring system;
8. Restrict access by a client to a machine in the system that generates multitude requests towards a machine in the supervisory system;
9. When detected, block access to any attempt to scan ports and vulnerabilities;
10. Seek to use the best available supervisory tool that meets the requirements according to the means at our disposal;
11. The gap between two visualisations and full analyses of the supervisory system should not be large (preferably every day);
12. Avoid configuration errors on servers and the use of obsolete components;
13. Always monitor associated services such as social networks to see what activities are taking place there.

5 CONCLUSION
There are security issues everywhere and approaches also exist. However, the main challenge is to find and to set up suited system starting by an infrastructure for monitoring for traffic and access to resources in a network. In order to set up a monitoring system within an existing system, we need to understand how the systems work, to list eventual vulnerabilities by conduction diverse penetration tests.

In this work, we have analysed diverse Cameroonian web portals so that to understand the potential attacks targeted. The ZEH was used as the methodology for conduction our penetration test focusing on four important steps: Reconnaissance, Scanning, Exploitation and Operation.

To consider as findings, we listed diverse vulnerabilities of some selected web portals. We have grouped them into three categories: configuration errors, programming errors and use of failed components. After identifying the vulnerabilities, we then propose and justify a distributed architecture based mainly on Nagios system to perform at large scale. The aim of the distributed architecture is to monitor web portals and therefore to reduce attacks based on scanning vulnerabilities. Furthermore, we have put together a list of 13 recommendations to be followed by system administrators.

Because, security remains an ongoing issue, we think that we should keep testing and re-evaluating the proposed infrastructure as an ongoing process aiming to identify its weaknesses that will arise as the time passes.

REFERENCES
3. Ingenieris Mobile 1.0. Le site internet de l'université de Yaoundé I piraté – Cameroun.
4. Ncedem, A. Cameroon is a country vulnerable to cyber-criminality.  


