Proposal of a computer security scheme for a Comprehensive Planning System

Propuesta de un esquema de seguridad informática para un Sistema Integral de Planeación

ROJAS-ALONZO, Jhon Henry†*, ARRIOLA-ESCALANTE, Claudia Ivette, MENA-CANTORAN, Rocio Lilia and CEJAS-LEYVA, Nohemí

Tecnológico Nacional de México Campus Cancún / Tecnológico Nacional de México Campus Agua Prieta

ID 1st Author: *Rojas-Alonzo, Jhon Henry* / **ORC ID:** 0000-0002-2873-0414, **Researcher ID Thomson:** ABB-9539-2021, **CVU CONACYT ID:** 546576

ID 1st Co-author: *Arriola-Escalante, Claudia Ivette* / **ORC ID:** 0000-0003-3183-6997, **Researcher ID Thomson:** ABB-9531-2021, **CVU CONACYT ID:** 944922

ID 2nd Co-author: *Mena-Cantoran, Rocío Lilia* / **ORC ID:** 0000-0003-3628-321X, **Researcher ID Thomson:** ABC-6081-2021, **CVU CONACYT ID:** 555196

ID 3rd Co-author: *Cejas-Leyva, Nohemí* / **ORC ID:** 0000-0002-1282-626X, **Researcher ID Thomson:** ABA-2730-2021, **CVU CONACYT ID:** 700396

DOI: 10.35429/EJDRC.2021.13.7.10.16 Received July 28, 2021; Accepted December 30, 2021

Abstrac

At present, Comprehensive Planning Systems are a fundamental part of any organization, since they function as a tool in the scheduling of activities as well as their Budgeting, Control, and Monitoring. The improvement and continuous updating of said systems are necessary to provide reliability in any context, that is why this research proposes to design and implement a computer security scheme focused on the entire administrative system, benefiting users who interact operating in the various related processes. In addition to this, the methodology used will be divided into two phases constituted by activities and tasks that will allow to satisfactorily find critical, severe, and moderate vulnerabilities in the server. Consequently, by obtaining a comprehensive diagnosis of the server, a security scheme is implemented to solve the problems detected and improve the organization.

Comprehensive Planning Systems, IT Security, Vulnerabilities

Resumen

En la actualidad, los Sistemas Integrales de Planeación son parte fundamental para cualquier organización, ya que funcionan como una herramienta en la programación de actividades así como su Presupuestación, Control y Seguimiento. El mejoramiento y la actualización continua de dichos sistemas son necesarios para brindar confiablidad ante cualquier contexto, es por eso que la presente investigación propone diseñar e implementar un esquema de seguridad informática enfocada a todo el sistema administrativo, beneficiando a los usuarios que interactúan operando en los diversos procesos relacionados. Aunado a esto, la metodología empleada se dividirá en dos fases constituidas por actividades y tareas que permitirá encontrar de manera satisfactoria vulnerabilidades críticas, severas y moderadas en el servidor. Consecuentemente, con la obtención del diagnóstico integral del servidor, se procede a la implementación de un esquema de seguridad para solucionar las problemáticas detectadas y mejorar la organización.

Sistemas Integrales de Planeación, Seguridad informática, Vulnerabilidades

Citation: ROJAS-ALONZO, Jhon Henry, ARRIOLA-ESCALANTE, Claudia Ivette, MENA-CANTORAN, Rocio Lilia and CEJAS-LEYVA, Nohemí. Proposal of a computer security scheme for a Comprehensive Planning System. ECORFAN Journal-Democratic Republic of Congo. 2021, 7-13: 10-16

* Correspondence to Author (e-mail: jhon.ra@cancun.tecnm.mx)
† Researcher contributing first author.

Introduction

Currently, the Comprehensive Planning System (SIPlan), has solved the problems related to the annual programming of activities, as well as the Budgeting, Control, and Monitoring of these.

This system is hosted on a server of the National Technological Institute of Mexico (TecNM) with the name of Administration System (SISAD) under a free license from SIPlan, developed at the Technological Institute of Cancun (IT Cancun), the Institute that has coined the project.

The aforementioned system has benefited 132 Federal Institutions of the TecNM, of which 2,865 areas carry their budgetary control in this system, as well as the 4,528 users who interact daily for the operation of their different activities in which the following are carried out processes:

- Annual Work Program
- Annual Operating Program
- Sub-budgets
- Budget Transfers
- Budget adequacy
- Programmatic Budget Evaluation
- Requisitions
- Travel expenses
- Purchase Orders
- Simultaneous Warehouse Entry and Exit
- Service Request
- Payment request
- Payment Registration
- Educational Structure

Currently, in the Technological Institutes, Planning, Programming and Budgeting is systematized in the SIPlan, which has solved the problem of keeping strict control of the budget exercise in general of the Institution.

The Comprehensive Planning System until 2019 was hosted on a server at IT Cancun. Since 2015, this server was tuned and made ready for the operation of the SIPlan, sufficient vulnerability tests were carried out to avoid intrusion both to the server and to the Database. This server has the Red Hat Linux Operating System installed which is installed with a graphical environment called Anaconda.

Today, the Management System is hosted on a Microsoft Azure platform server, with the Free BSD Operating System which is a free and open-source derivative of BSD (Berkeley Software Distribution) with a focus on speed, stability, security, and consistency, among other characteristics.

However, the aforementioned is a new server that has not been tuned, so it is necessary to carry out the necessary vulnerability and attack tests to be able to do the corresponding tuning and generate confidence in the stability and security of the information. Due to the above, it is important to develop this proposal for a computer security scheme for the TecNM Administration System.

Thematic scheme

The emergence of the idea

The present investigation has been delimited with the purpose of guiding the study with a systematic order. Next, figure 1 shows a diagram where this delimitation is observed:

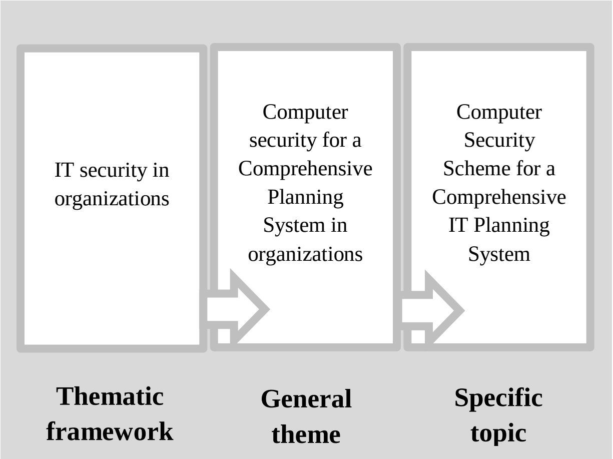

IT security in organizations
Computer security for a Comprehensive Planning System in organizations
Computer Security Scheme for a Comprehensive IT Planning System
Thematic framework
General theme
Specific topic

Figure 1 General scheme for the delimitation of the subject
Source: Author's perception

In phase one, IT security in organizations forms the thematic framework, thus providing a starting point for this study. The general research topic has been called Computer Security for a Comprehensive Planning System in organizations, consolidating the research.

Phase three is called Information Security Scheme for a Comprehensive Planning System in IT, delimiting the investigation towards a specific sector.

Identification of experts

This project is developed with personnel from the Information and Communication Technologies Directorate and personnel from the Cancun Technological Institute, both belonging to the National Technological Institute of Mexico. The staff is interviewed by the researcher, sectioning the process by rounds, addressing important points on the subject of study. The researcher subjectively determines the knowledge of the personnel regarding the Comprehensive Planning System. The selected personnel is called Experts.

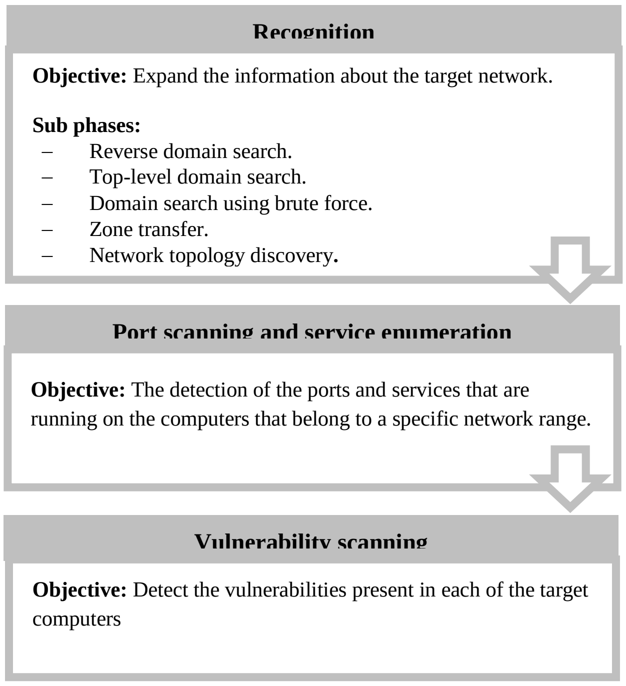
Methodology

The methodology used in the development of this study is "Methodology for the Detection of Vulnerabilities in Data Networks". (Franco, D., Perea J. and Puello, P., 2012).

Which consists of three phases:

- Recognition
- Port scanning and service enumeration
- Vulnerability scanning

Each of the phases is supported by software tools. The results of each phase provide data necessary for the execution of the following stages. (Franco, D., Perea J. and Puello, P., 2012).



Recognition

Objective: Expand the information about the target network.

Sub phases:

- Reverse domain search.
- Top-level domain search.
- Domain search using brute force.
- Zone transfer.
- Network topology discovery**.**

Port scanning and service enumeration

Objective: The detection of the ports and services that are running on the computers that belong to a specific network range.

Vulnerability scanning

Objective: Detect the vulnerabilities present in each of the target computers

Figure 2 Scheme of the methodology for the detection of vulnerabilities in data networks
Source: Methodology for Detecting Vulnerabilities in Data Networks (2012)

Additionally, for this study, two separate phases were implemented for the operation of the Application, which are:

- Preparation
- Implementation

Activities

The activities are developed according to the phases of the Methodology used:

1. Preparation
1.1. Installation
2. Recognition
2.1. Diagnosis of security schemes
3. Port scanning and service enumeration
3.1. Scan with port tools
4. Vulnerability scanning
4.1. Vulnerability detection
5. Implementation
5.1. Determination of threats

Chores

The Tasks are developed according to the phases of the Methodology used and the proposed activities:

1. Preparation
1.1. Installation
1.1.1. Server Installation
1.1.2. Application Installation
2. Recognition
2.1. Diagnosis of security schemes
2.1.1. User Identification
2.1.2. Information Identification
2.1.3. Infrastructure Identification
3. Port scanning and service enumeration
3.1. Scan with port tools
3.1.1. Analysis application with the Nmap tool
3.1.2. Scan report analysis
4. Vulnerability scanning
4.1. Vulnerability detection
4.1.1. Analysis application with the ZAP tool
4.1.2. Analysis application with the Nessus tool
5. Implementation
5.1. Determination of threats
5.1.1. Server Tuning
5.1.2. Application Tuning

Diagnosis of security schemes

SIPlan security is carried out from the following three parts:

- Users: Here there is an Entry Verification system (Login) where the Session Name and Password are verified, once this filter has been passed through there are 23 Types of Users to which the Menu Options are assigned , as well as their respective privileges.

- Information: In this aspect, the information is housed in a Relational Database, the verification of the information that enters has two filters, one from the code itself where the validations are made so that what we want to happen and from the code itself Database Model as it has restrictions to save integrated information.

- Infrastructure: The system is hosted on the Azure platform of TecNM in the cloud, as for the server there is a Dell PowerEdge R320 Server - Xeon E5-2407V2 - 2.4GHz - 8GB - 2 x 1TB - Raid 1 - Free Dos.

The System is developed in the PHP Programming Language that runs on the Server side, interacting with JavaScript that runs on the client-side, the views are developed in HTML and the Database Manager is MySQL.

As preventive mechanisms, the backup of the database is done through a process programmed in the TecNM Server which is launched at two in the morning to a remote server which is located in the ITCancún, this backup is done in the SQL.

It is worth mentioning that the system is in the general implementation stage at the National Level, so it is necessary to be updating the system modules.

Vulnerability detection

At the launch of the project at the national level, a vulnerability analysis was carried out and Development Vulnerabilities were detected, especially in possible SQL injections due to the use of the GET method in the passing of parameters between PHP files, this was detected through scans that were they did with the ZAP Scanning tool and Nessus.

The problem was that when making a hyperlink with an HTML <a> tag embedded in a PHP file, the URL of the address where the tag refers as well as the variables are displayed in the status bar of the browser which is passing to the other PHP file where it is directed, this generated a high vulnerability since if that URL was copied and the values were changed, it was prone to logical SQL injections for the system and for the base engine data, then the solution was immediately made as shown in Figure 3.

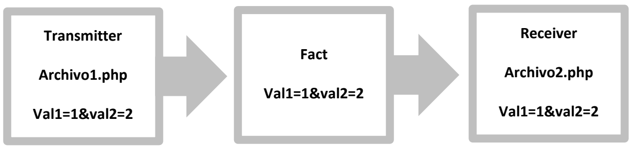


Figure 3 Parameter passing by PHP's GET method
Source: Author's perception

As mentioned in the Diagnosis, the System is hosted on Microsoft's Azure platform, and it is known that in this type of platform the tools are provided to implement the most convenient server, and therefore the security of this server is totally low. responsibility of the platform administrator.

In this case, the Information and Communication Technologies (ICT) staff of TecNM, when scanning the server, detected that the Server Security was not well implemented since it was possible to access the servers and view the contents of the Directories and Files in such a way that by having the code it was susceptible to intrusion into the database by means of some SQL injection, so all this type of privilege was closed for users who visited or browsed the server.

Reviewing the requestlog file on the server where SISAD was hosted at that time, the following vulnerabilities were found:

- The server instance was not new, it had not been created for the SISAD implementation.

- It was detected that the server was very compromised, since March.

- There was a clear sign that they had already had an Apache service there and had already hit it previously.

- The detail was to know if they had left a shell on the server.

- This indicates that they did not enter through your app, since the server was already compromised long before.

- According to the log, it indicates that March 27 was the first time that SISAD was installed on that server, but it had already installed things before and they had already gotten into them.

- In line 9912 it indicates that it is the first time that they sent a request for administracion2 with http: //5.188.210.101/echo.php, from there it is detected that the server is open without protection.

- On the server is the phpmyadmin application, which is super vulnerable for SQL injections.
- From line 9750 they put the nmap to scan that server overnight and the next day with a program called ZmEu that is to attack phpmyadmin.

- Already here you can detect that they entered.

- From here almost daily they hit the server and from the same IP 129.211.50.227.

- Similarly, it was detected that the SQL queries had vulnerabilities.

Proposal

Having already had a diagnosis and having detected the vulnerabilities, a security scheme was implemented to solve the problems encountered in the detected vulnerabilities.

As a first point was the solution of the server issue for which the following actions were carried out:

- Users were advised that SISAD would be out of service for maintenance.

- A new instance was created on the Azure platform for the creation of a new server.

- Security rules were already applied to this new server so that it was not a completely open server but a completely closed one. Only the services that were necessary for the SISAD Application were opened and it was duly controlled.

- A fresh installation of the required software was made: Apache, MySQL, PHP.

- phpmyadmin is no longer installed, for security reasons.

- The System load was made as new from the original scripts, a copy of the previous server was not made.

- The database was loaded.

- SQL injection is a danger to data integrity and it is known that attacks must protect data. For this, the .htaccess file of our server was reconfigured to protect against this type of attack, and it was added to the .htaccess file with the following lines:

```
RewriteCond %{QUERY_STRING}
(;|<|>|’|”|\)|%0A|%0D|%22|%27|%3C|%3E|%00
).*(/\*|union|select|insert|cast|set|declare|drop|up
date|md5|benchmark) [NC,OR]

RewriteCond %{QUERY_STRING} \.\./\.\.
[OR]

RewriteCond %{QUERY_STRING}
(localhost|loopback|127\.0\.0\.1) [NC,OR]
RewriteCond %{QUERY_STRING} \.[a-z0-9]
[NC,OR]

RewriteCond %{QUERY_STRING}
(<|>|’|%0A|%0D|%27|%3C|%3E|%00) [NC]

RewriteRule .* - [F]
```

It is important to remember that you need to have the server configured so that .htaccess is active and allows URL filtering. This helps to improve the security of the System.

As a second point was the solution to the issue of passing parameters between PHP files and SQL queries, for which the following actions were carried out:

To solve the problem of the visibility of the <a> tag, all the calls to other files with buttons of the Input type were changed, with this you no longer saw the references to where the hyperlink was pointing, but thinking wrong and that If someone could have already obtained those references, the whole process for passing parameters was changed.

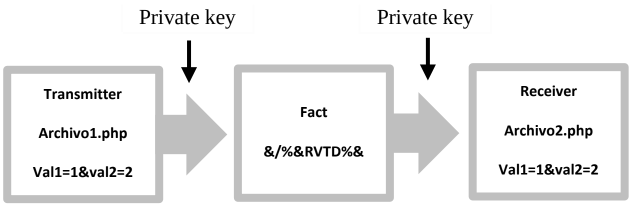


Figure 4 Solution to passing parameters in PHP with POST method
Source: Author's perception

The first thing that was done is that the data after "?" in file1.php, but a function called "encode" is called to which all the parameters to be protected are passed, upon receiving this function the chain assigns a control key before and after the chain, encoded with md5, the resulting string is encoded in PHP base64, the resulting string is passed to another function called "encrypt_one", which upon receipt of the string is applied another PHP encryption method "encrypt" and applies a word key encoded in md5 base 64, the resulting string is encoded in PHP base64 and this is the one that is passed through the URL with POST method.

As a second step, upon receiving the string in the file2.php, it calls a function called "decode" which does the reverse process of "encoding", thereby ensuring that even if the data sent by the URL is intercepted, they will not be able to be decrypted and thus ensuring that no SQL injections are made through these processes. To solve the SQL queries, we proceed to add the function "mysql_real_escape_string" before making a query to the Database, since it escapes special characters in a string for use in an SQL statement.

The function operates as follows:

- Escapes special characters in the string given by unescaped_string, taking into account the character set in the use of the connection, so that it is safe to use in mysql_query (). If binary data is to be inserted, use this function.

- Calls the mysql_real_escape_string function from the MySQL library, which prepends backslashes to the following characters: \ x00, \ n, \ r, \, ', "and \ x1a.

- This function should always be used (with few exceptions) to make data safe before sending a query to MySQL.

- The character set must be set either at the server level or with the mysql_set_charset () API function to affect mysql_real_escape_string ().

- Returns the escaped string or FALSE on error.

- Running this function without a MySQL connection present will also throw PHP E_WARNING level errors. It should only be run with a valid MySQL connection present.

An example SQL injection attack:

```
<? php
// We haven't checked $ _POST ['password'], it could be anything the user wanted! For instance:
$ _POST ['username'] = 'aidan';
$ _POST ['password'] = "'OR' '='";// Consultar la base de datos para comprobar si existe algún usuario que coincida
$consulta = "SELECT * FROM users WHERE user='{$_POST['username']}' AND password='{$_POST['password']}'";
mysql_query($consulta);

// This means that the query sent to MySQL would be:
echo $ query;
?>
```

The query sent to MySQL:

```
SELECT * FROM users WHERE user = 'Aidan AND password = '' OR '' = ''
```

This would allow someone to access a session without a valid password.

How to solve it:

```
<? php

// We haven't checked $ _POST ['password'], it could be anything the user wanted! For instance:
$ _POST ['username'] = 'aidan';
$ _POST ['password'] = "'OR' '='";
```

```
// Query the database to check if there is a matching user
$ query = "SELECT * FROM users WHERE user = '{mysql_real_escape_string ($ _POST [' username '])}' AND password = '{mysql_real_escape_string ($ _ POST [' password '])}'";
mysql_query ($ query);

// This means that the query sent to MySQL would be:
echo $ query;
?>
The query sent to MySQL:
SELECT * FROM users WHERE user = 'aidan' AND password = '\' OR \ '\' = \ ''
```

This would prevent someone from being able to access a session without a valid password.

Conclusions

Given the study carried out, it is evident that the topic of Security in Systems is very delicate.

The level and security schemes to be implemented will depend on the type of information that is handled.

For the present work, both the vulnerability detection systems and the implemented schemes are free of charge, while for more sensitive information it is necessary to take into account the investment of software for both detection and implementation of security schemes.

Among the types of sensitive information, we can mention bank data were the situation, investments, and savings of users are found. Other types of national security, among which can be mentioned, are data on state secrets, information on weapons (from basic to nuclear), and data on secret missions to name a few.

Computer Security systems are of utmost importance in an organization, since the more sensitive the data, the greater the security schemes used.

References

Baizán, E. (2002). Como elaborar un proyecto. Asturias. España: Gráficas Eujoa.

Franco, D., Perea, J., & Puello, P. (2012). Metodología para la Detección de Vulnerabilidades en Redes de Datos. Obtenido de Scielo: https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0718-07642012000300014

Goñi, I. (2008). El qué y el cómo del diagnóstico del sistema de información gerencial. Obtenido de Scielo: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352008000500004&lng=es&tlng=es

Roa, J., & Bijani, G. (2013). Seguridad Informática. Madrid, España: McGrawHill.

Romero, M., Figueroa, G., Vera, D., Álava, J., Parrales, G., Álava, C., Castillo, M. (2018). Introducción a la Seguridad Informática y el Análisis de Vulnerabilidades. Manabí, Ecuador: 3Ciencias.

Ruiz, M. (2018). Propuesta de modelo para diagnosticar sistemas de información en las organizaciones. Obtenido de Revista CientíficoTécnica de la Empresa de Telecomunicaciones de Cuba, S.A.: http://www.revistatonoetecsa.cu/articulo/propuesta-de-modelo-para-diagnosticar-sistemas-de-informacion-en-las-organizaciones

Vidal, E. (2004). Diagnostico Organizacional-Evaluación sistémica del desempeño empresarial de la era digital. Bogotá, Colombia: Oceo Ediciones.