Vicnum Manual

Mordecai Kraushar
December 29, 2009

Table of Contents

Introduction

This manual describes Vicnum - a flexible and vulnerable web application which demonstrates common web security problems such as cross site scripting, sql injections, and session management issues. The program is especially useful to IT auditors honing web security skills and setting up 'capture the flag' type games.

Being a small web application with no complex framework involved, Vicnum can easily be invoked and tailored to meet a specific need. For example if a test vulnerable application is needed in evaluating a web security scanner or a web application firewall, you might want to control a target web application to see what the scanner can find and what the firewall can protect.

Ultimately the major goal of this project is to strengthen security of web applications by educating different groups (students, management, users, developers, auditors) as to what might go wrong in a web app. And of course it's OK to have a little fun.

NOTE THAT THIS IS AN INTENTIONLLY VULNERABLE APPLICATION AND CERTAIN COMMON SENSE MEAURES SHOULD BE TAKEN. FOR EXAMPLE DON'T USE THIS PROGRAM WITH A PRODUCTION DATABASE. THE AUTHOR WILL NOT BE HELD RESPONSIBLE.

Comments welcome either via:
the OWASP project page http://www.owasp.org/index.php/Category:OWASP_Vicnum_Project)
the SourceForge project page which has the code http://sourceforge.net/projects/vicnum/
or my linkedin page http://www.linkedin.com/in/mkraushar

Mordecai Kraushar

Vicnum the Game

“The computer will think of a three digit number with unique digits. After you attempt to guess the number, the computer will tell you how many of your digits match and how many are in the right position. Keeping on submitting three digit numbers until you have guessed the computer's number.”

The Vicnum game is similar to many other games commonly played to kill time. These games might have different names such as Guessnum and could certainly have different rules. For example in Vicnum the player might guess a number with digits repeated, which could never be the computer’s guess which must involve unique digits. A more rigorous implementation might require guesses to have unique digits and even be possible solutions by analyzing the previous guesses and responses. The code to do this is left as a challenge to the reader.

Another related game based on letters in a five letter word is Jotto, for which multiple names and rules also exist. There has been some development on a vulnerable web application related to this game and it may resume shortly.

Vicnum the Program

The software program was originally written as a line mode PERL program. The roots of this are clearly visible by looking in the cgi-bin folder at the three perl programs

vicnum1.pl
vicnum2.pl
vicnum3.pl

Much later in having the program work with a database two additional PHP programs were added to the root of the Vicnum web directory.

vicnum4.php
vicnum5.php

These five files form the heart of the program with the usual index page, icons, images, doc and help folders present as well.

Over the years numerous scripts have been written mainly to create capture the flag type scenarios or to modify the database. A fair number of these scripts remain in very raw format in the admin folder which is protected with basic authentication (default password of 55Broadway) Some of these scripts have also been copied to the root folder and are used for game playing, for example the top.php script will show those who have guessed the computer's number in one guess and a capture the flag scenario could be established to see who is first in the top display.

Besides installing the programs in the right folders, it may necessary to set up the MySQL database.

create table results (idnum int(4) NOT NULL auto_increment PRIMARY KEY, name char(100), guess int(3) ZEROFILL, count int(2), tod TIMESTAMP);
grant ALL on *.* to root@localhost IDENTIFIED BY 'vicnum' ;

Of course if you download the VM you don't worry about this.

As one of Vicnums vulnerabilities, the help folder is indexable and one can see all the help files not just the help1.html file about how to play the game which is advertised from every page. In particular the help2.html file has the following tips as to how to set Vicnum up.

- **Make sure appropriate Apache modules are loaded**
- **Modify http config file to find perl code in the cgi-bin directory**
- **Consider putting an .htaccess file to protect the admin folder**
- **Review Magic quotes setting in php.ini (forces escapes on certain characters)**

In addition the help3.html file is also present by default and includes multiple tips for customizing the game. A key tip of course is to delete the help3,html file shown below as well as this manual.

- **Modify or delete this file**
- **Modify or delete the backdoors**
- **Don't allow indexing of the help file, don't call the help file help1.html**
- **Names with < > might be interesting but can trash the db, consider javascript to disable on the client, server side code to sanitize or code to check before writing to the database**
- **Change the names of cookies and hidden fields, extra cookies and/or hidden fields are ok too**
- **Remove encoding of a certain hidden field, showing it in plain text will make it trivial. There are other ways to obfuscate, for example add or rotate integers**

- **Likewise cookies content can be obfuscated**
- **Remove the hidden field that announces admin, maybe admin folder shouldn't be called admin**
- **Lock down the admin folder with basic authentication, optionally make some of the admin scripts available**
- **Let them find an htaccess user file**
- **Code to require a different guessed number can be commented out which enables the easy back button refresh**
- **Where and when to do the db write? if done with the perl code it is written at once, otherwise pass it to the php program via cookies , However Perl code modules may be needed to do db queries (this is not yet done)**
- **With magic quotes off sql injection should get the entire db**

Additional customization is facilitated by reading the code and examining the comments. For example base64 encoding of the answer in a hidden field can be disabled thus making the answer easily viewable in a view source.

Hacking Vicnum

By default there are three capture the flag targets available in the Vicnum application. As seen as on the home page:

Click here to see those who may have played a perfect game.
Or here to see those who have clearly hacked the game.
Or here to see those who have hacked the game and the database

To play a perfect game means to guess the computer's number in one guess which is unlikely given that there are 810 possibilities. More likely they have hacked the application. Possible ways to do this include:

1. A central issue here is the question of how the web server remembers its number through repeated guesses since the http protocol is stateless. Typically state is maintained either via session cookies, URL strings or via hidden form fields. Since none of the first two are present in initial playing of the game it would seem worthwhile to examine the hidden fields. One of the hidden fields called VIEWSTATE contains the answer in base64 encoded format which can be decoded with proxy apps as well as many online web sites. True techies will take great delight in noting that the 3 digit guess number encodes very nicely into a four digit base 64 number with none of the telltale = signs appended.

2. A related question is how does the app track of the number of guesses. This too is maintained within the hidden fields and can duped by backing the browser up to the first guess once you have figured out the answer. Note that once you have hit the CONTINUE button and have been entered into the database it is too late to reenter your name with the same guess.

3. A third way to ace the program is to try the backdoor. Read the code or guess !

To 'clearly hack the game' it is probably best to look at the cookies being passed to the vicnum4.php program. These cookies named for large European cities  have the game fields contained within cookie values and can be modified via proxy tools or browser plugins such as Tamper.

To go one step further and hack the game and the database try SQL injecting the search form on the home page of the application. Picturing the backend query as taking input from the app and then quoting it for the SQL query, one merely needs to fill the query out to be a valid query that would obtain all the record such as ' or "" = ' .

For this particular CTF to work a single userid of 55Broadway with the maximum negative score must exist. A dbhack.php script exists in the admin folder to delete 55Broadway accounts with positive scores which are required to win the game.

Other Vicnum Vulnerabilities

Beside the vulnerabilities woven into the 'capture the flag' portions of the application and the indexable directories mentioned above there are a few other vulnerabilities to be probed.

There is an admin option seen in the source of the home page and it is obvious where this folder is to be found. An exercise can be set up to access this folder with tools and execute those administrator scripts. Some of those scripts therein such as the backup.php script can easily be used to DoS the database by creating multiple backups of the results database.

And of course what would a vulnerable web application be without a cross site scripting (XSS) vulnerability? Try entering trivial tags in the name field on the home page such as <i>, escalate then to <script> alerts, and eventually iframe script tags. They should all succeed as no server side validation is being performed.

Note that these XSS vulnerabilities are reflective only and that some validation is enabled by default prior to the MySQL write in vicnum4.php to prevent a stored XSS. This can be disabled to muck up the database for those that come late to the party.

If you find or develop others let me know!

Acknowledgements

Thanks to multiple people at CipherTechs and OWASP for their reviews and support.
And a special thanks to my wife and children who contributed heavily to the initial development of the program.