Pi Exam

## Summary

This lab is designed to be attacked by hand, rather than require a set of tools. Five common attacks using just a browser.

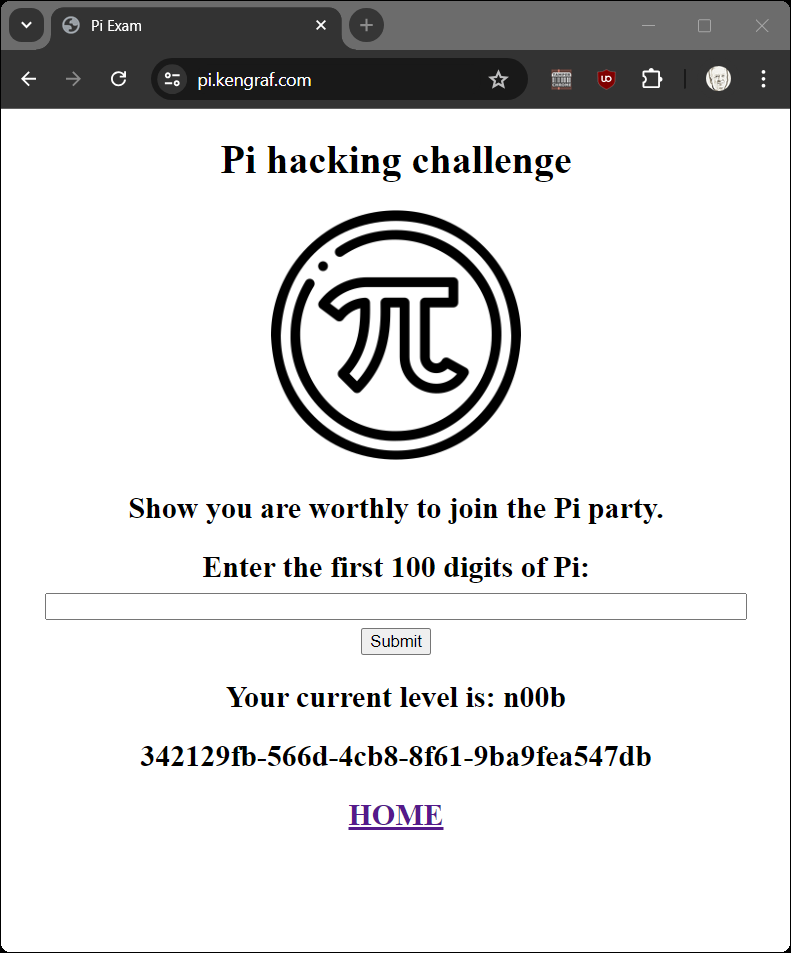
**Capture screenshots of your successes for submission.**

## Tools/Resources

A browser with inspect capability.

## Walkthrough

1. Home page: <https://pi.kengraf.com>

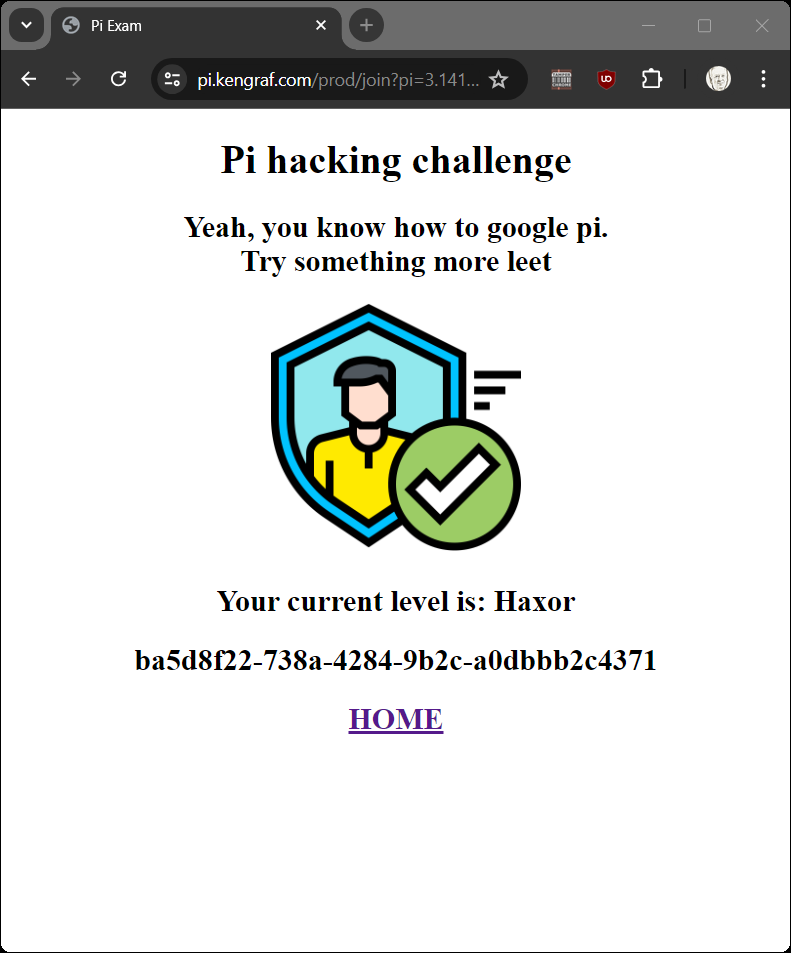


1. Review the source code. Dev can leave interesting comments

A screenshot of a computer

Description automatically generated

1. Get 100 digits from Google



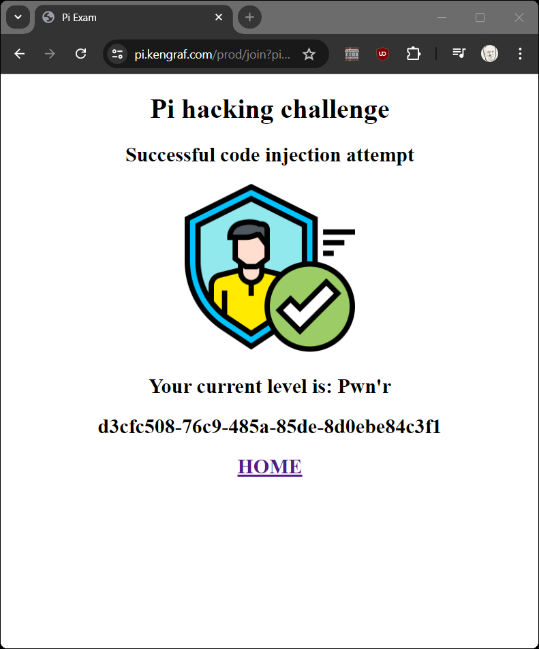
1. Observe in code or the submitted URL the digits parameter. Maybe we don’t need 100 digits

<https://pi.kengraf.com/prod/join?pi=3&digits=1>



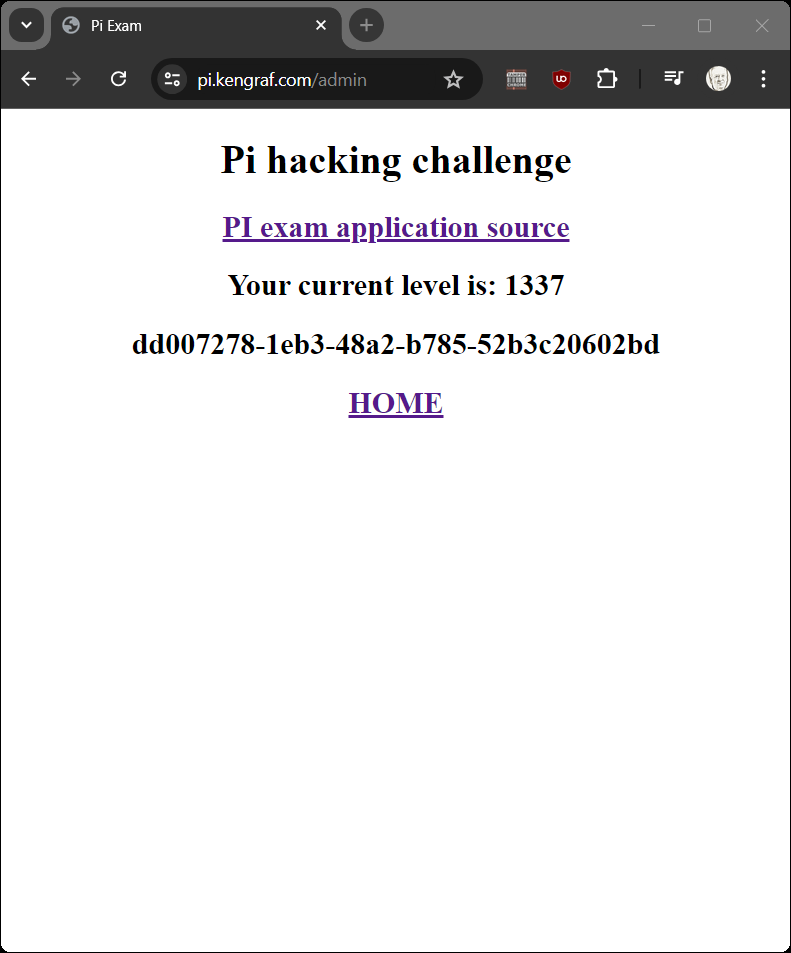
1. The site is looking for a number. Maybe we can add an expression to back the back end code eval to TRUE. This is the basis for SQLi and other forms of injection attacks.

<https://pi.kengraf.com/prod/join?pi=1+or+1%3D%3D1&digits=100>



1. Comments in source code refer to an “admin” page we should try that. Real world we would use a tool like dirbuster to automatically guess 100s of common file names

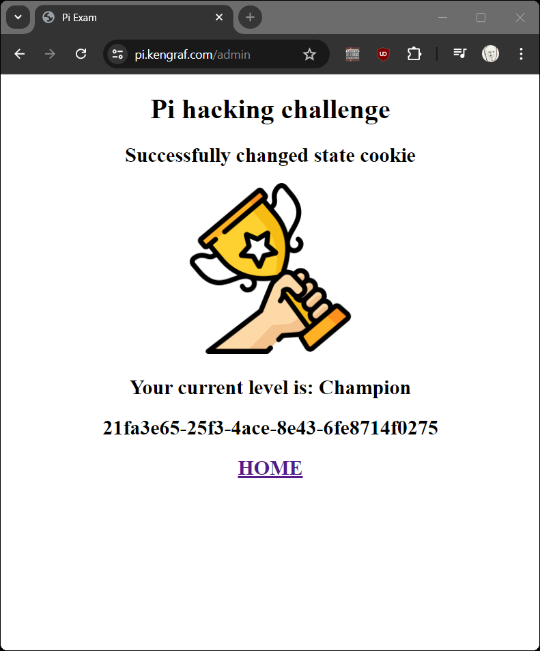
<https://pi.kengraf.com/admin>



1. Cookies can be manipulated by clients and should not trusted by the application. We have noticed a “level” element has been incrementing as we progressed. Maybe we can just jack it up to 99 and refresh?

A screenshot of a computer

Description automatically generated



## Submission

Upload to Canvas a docx or pdf containing screenshots showing your (5) successful web submits.

## Additional Thoughts

How does the PI-exam python program support HTTPS?

Is this application vulnerable to OS level attacks?

What is the cost for running this in the cloud?