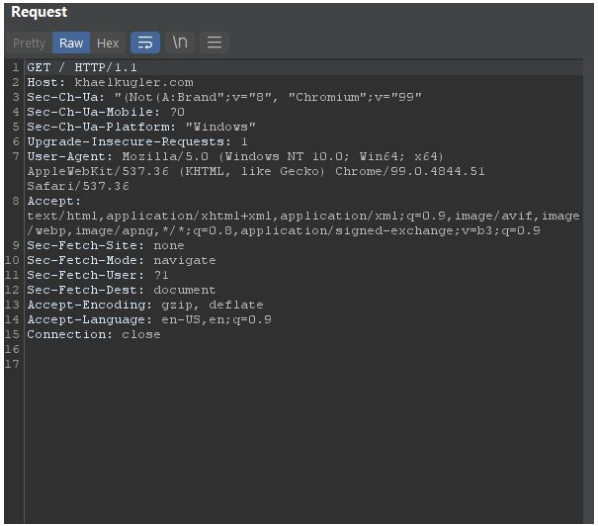
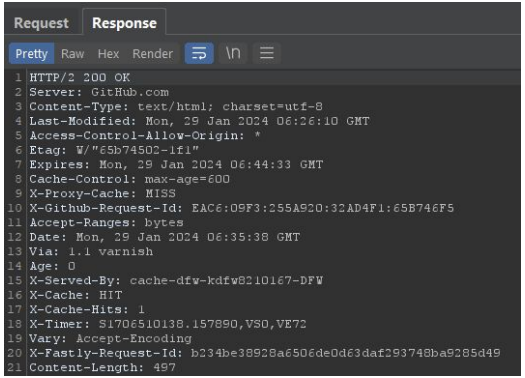
Web Hacking

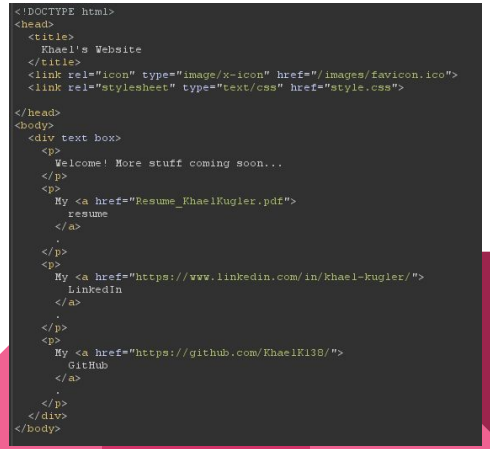
Khael Kugler

What is this “web”? Requests and Responses:

- HTTP requests and responses
- Request:
  - Asking a server for some data
- Response:
  - The server sends that data

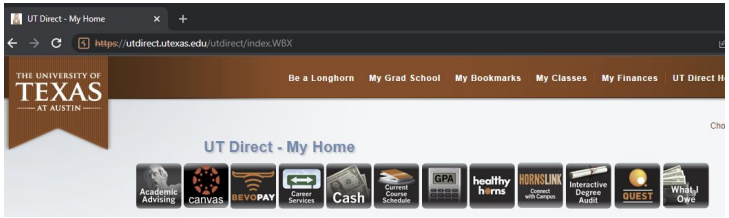


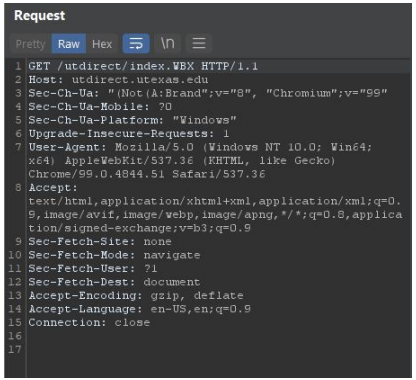


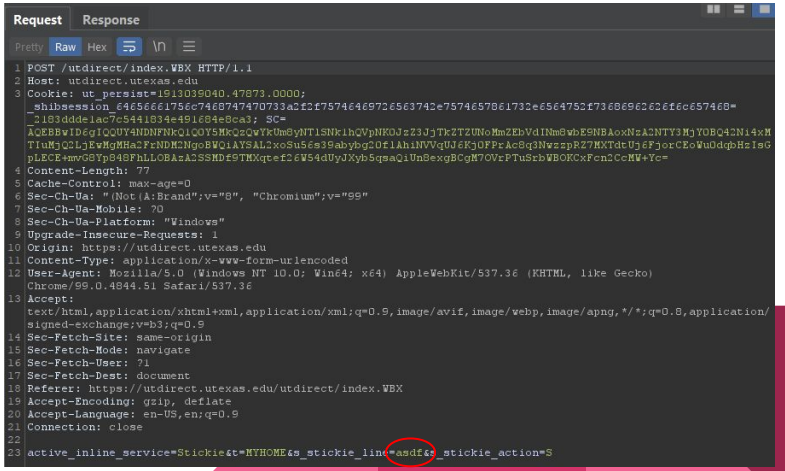


Two Important HTTP Requests

- GET Request:
  - Retrieving data from a server
- POST Request:
  - Sending data to a server





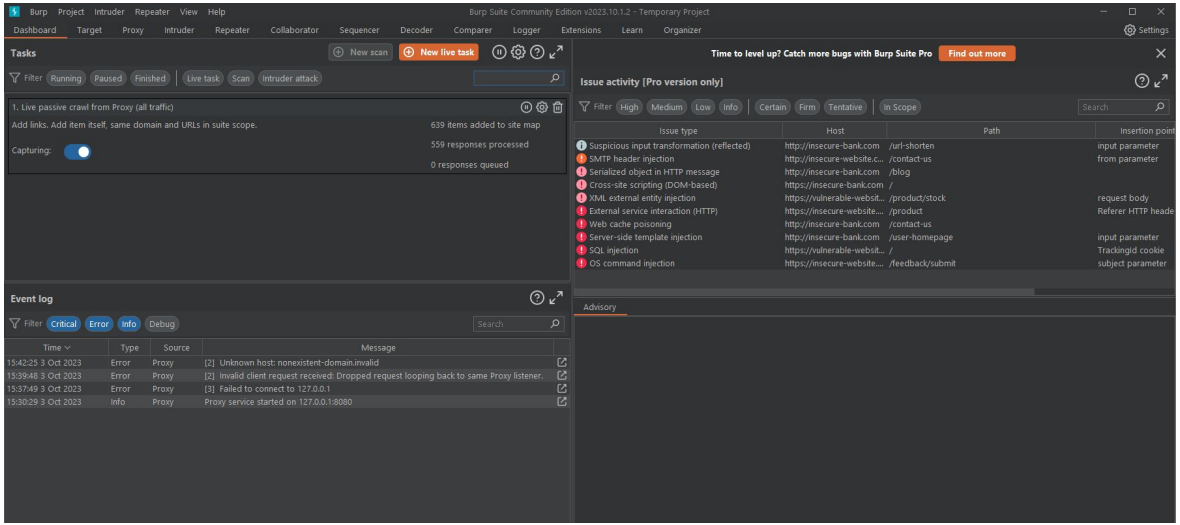


A quick cURL demonstration

How do I modify these requests?

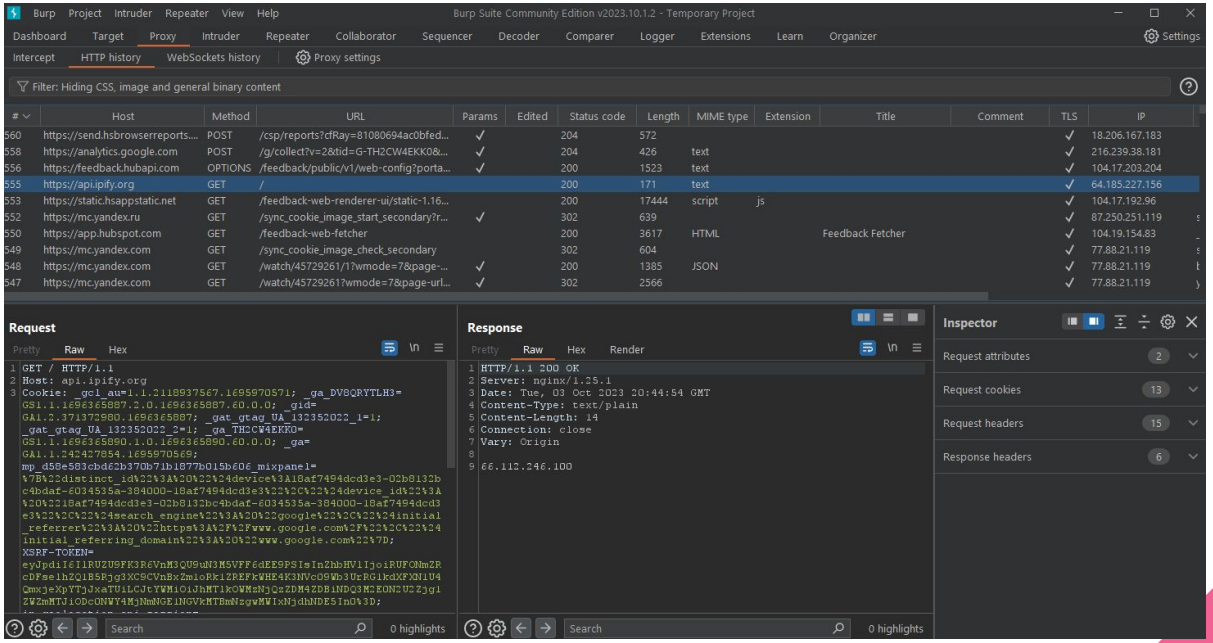
- Burp Suite - acts as a proxy between you and the web page

Homepage:

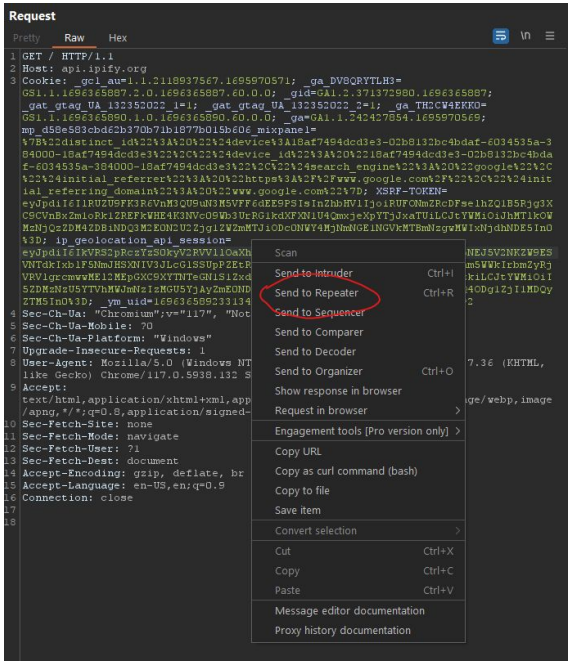


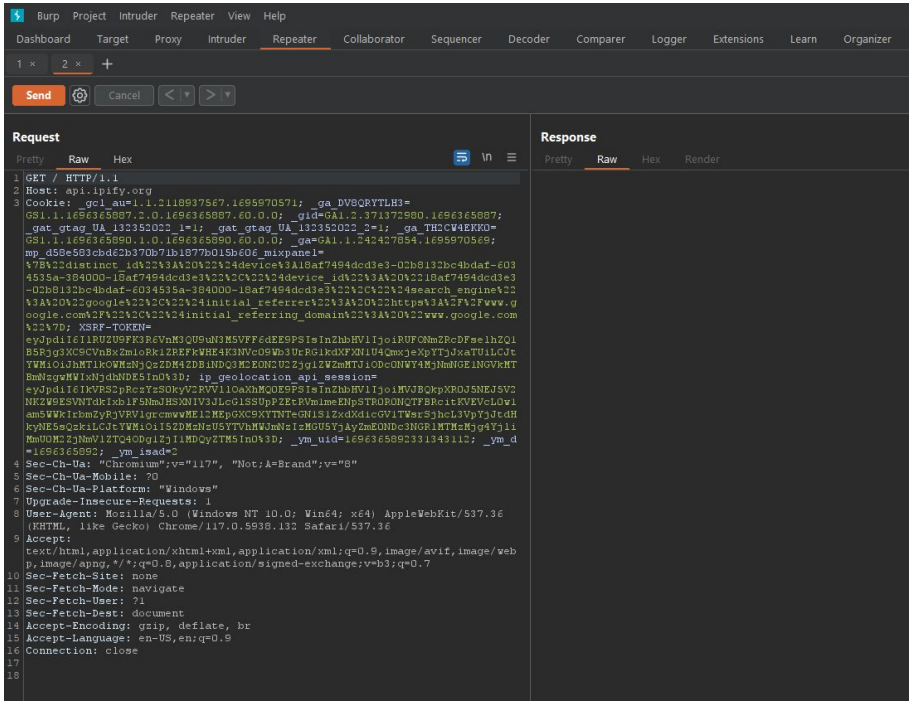
Burp Proxy

- Proxy Tab



Using Repeater





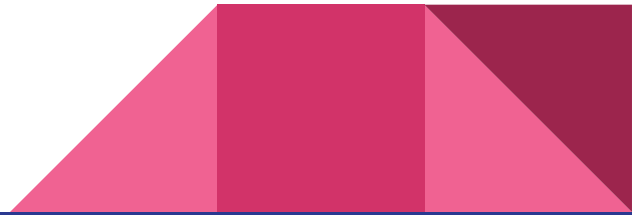
Download →

Scoping out a target

Ok, I have a target…

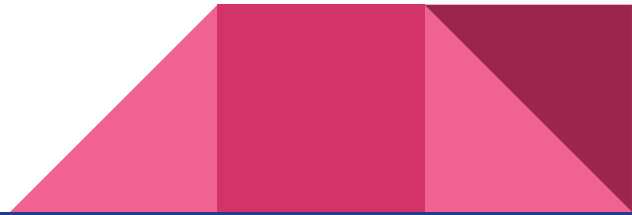
How do I figure out the attack surface (for example, of utexas.edu)?

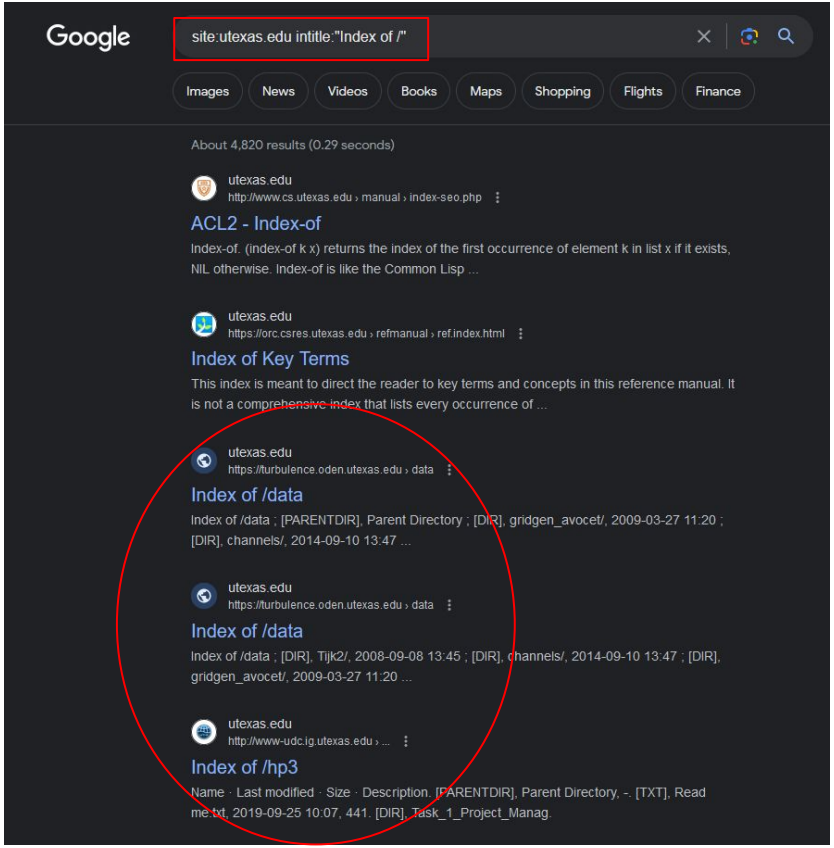
- Google Dorking
  - Searching for insecure sites index by Google
  - Very useful
- Exploring for sources of user input
  - Items like search boxes and HTML forms
- Automated scanners (dangerous; don’t use without explicit permission)
  - E.g. UT bug bounty doesn’t allow scanners



Google dorking utexas.edu

- UT’s web surface is enormous
- Many old, out-of-date webpages to mess with
  - Lots of already known vulnerabilities
  - Pages last updated before I was born
- Google dork techniques are made plentiful online
  - Find a top 30 list and copy/paste for interesting files

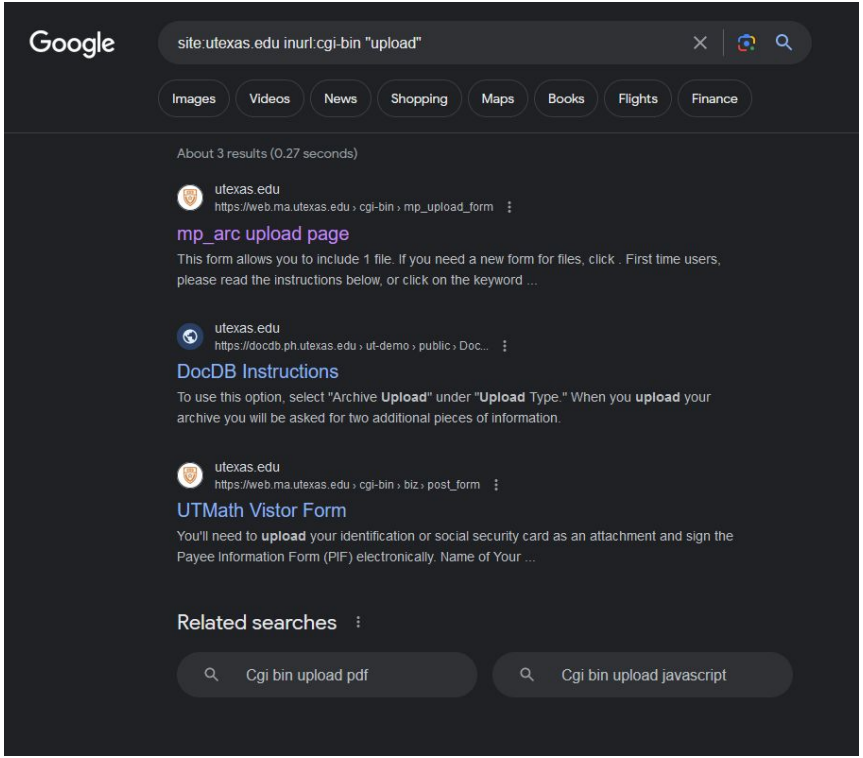


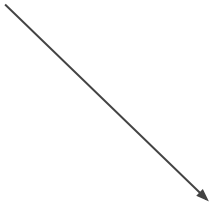


Index of /data

Name	Last modified	Size	Description
Parent Directory		-	
gridgen_avocet/	2009-03-27 11:20	-	
channels/	2014-09-10 13:47	-	
Tijk2/	2008-09-08 13:45	-	
Tijk/	2007-10-04 17:13	-	
MKM/	2010-12-17 11:11	-	
AJZM_old/	2007-02-01 13:58	-	
AJZM/	2014-05-30 14:01	-	

Let’s look at the CS





Submission Page for mp_arc Preprints

This form allows you to include 1 file. If you need a new form for files, click here.
First time users, please read the instructions below,
or click on the keyword for each entry field before filling out that part of the form.

REQUEST: archive paper

PASSWORD: nhNDD (can be ignored for now)

FORMAT:

KEYWORDS:

AUTHORS:

TITLE:

ABSTRACT:

COMMENTS: (optional)

PAPER:
Browse... No file selected.

After clicking the button Submit to submit the above to mp_arc, please wait for the reply:
You must respond to that reply before your paper can be archived.

If you need to reset the various form elements to their default states, you can press this button: Reset To Default Values

My Own Reconnaissance

- Compiled a list of 3000+ “utexas.edu” subdomains
- Attempted to connect to a webserver on each, so 99% should be live

Available here https://khaelkugler.com/misc/subdomains.txt:

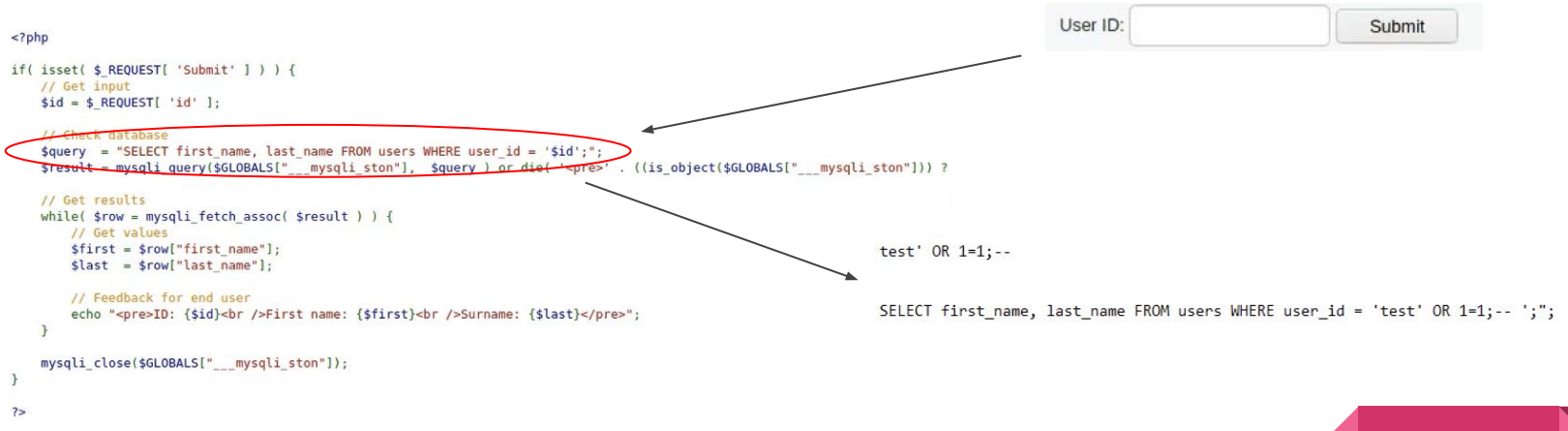
Common Vulnerabilities

Cross-Site Scripting (XSS)

- Injecting JavaScript into a page
  - Generally through a URL or through stored page data (like a comment section)
- Easily the most common vulnerability on UT - it’s basically free money

SQL Injection

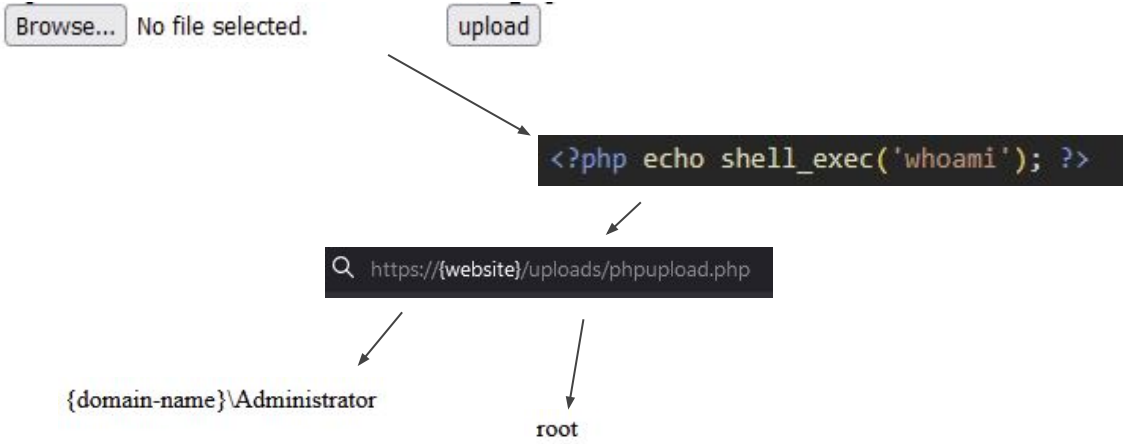
- Accessing a SQL database without permission
- Usually via apostrophe or quotation mark



- Cs.utexas.edu databases

Arbitrary File Upload

- Uploading a dangerous file to a server, which it then executes
- More common than you'd expect
  - This is because many filetypes, like PHP, must be executed to be displayed
  - Thus, accessing a PHP file on a server will execute the PHP code inside



Default Credentials

- This is WAY more common than it should be
- With many services, the default credentials can be looked up
  - Usually admin/admin, guest/guest, test/test

User: admin
Password: admin

Some Other Common Vulnerabilities

- Poor Authorization Methods
  - I've found a couple serious examples of this on UT
- Cryptographic Failures
- Insecure Configurations
- Outdated Components
  - Look for service version numbers, and search them up

Book.hacktricks.xyz has explanations and exploitations of all of these

UT Bug Bounty

What you'll look for

Find all kinds of bugs on UT's domains

- Cross-site scripting (XSS) **$$$**
- Remote Code Execution (RCE)
- SQL injection
- XML External Entity Injection (XXE)
- Authorization bypass/escalation
- Sensitive information leaks
- Cross-site request forgery (CSRF)

Be sure to log everything you find

What you won't do

- Run automated scanners
- Leak sensitive data
- Break stuff badly
- Mess with regular business operations
- Run ANY TESTING off of the UT VPN
- Share the vulnerabilities you find (like I DIDN'T just do)

Wanna get started?

Thanks! Questions???