**CS378 Ethical Hacking**

**Final Project**

##### Alex Irion, Tony Sampson, Krzysztof Drewniak

##### What is the tool?

##### Our tool, elit3\_h4ck.py, runs an automated phishing attack.

##### First, to set up the attack, we cloned the UT login page (<https://login.utexas.edu/)> and hosted it locally on the cs378-ethical-hacking network. Ideally, we would host our cloned site where it is publically viewable, but for this proof of concept the site can only be accessed from the cs378 network. The purpose of the cloned site is to trick students into believing they are visiting the real utexas login page, however we altered the javascript code on our site so that when the victim hits the submit button their username:password credentials are sent to our script!

##### After the cloned/altered website is up and running we are ready to start our attack! The attack is designed to be directed towards current classmates, other UT students/faculty, or specific email addresses. The body of the email can be configured to look socially engineered or directed at the victim(s) specifically. We chose to send an advising bar notification email since it is likely to be noticed and clicked on by our peer students. Our program finds email addresses of people we wish to the attack, sends the phishing email out to each person with a malicious link, and starts a server listening for victims that click the submit button on our cloned website!

##### Use these credentials however you wish!

##### /Users/alexirion/Desktop/Screen Shot 2017-05-04 at 12.58.22 PM.png

##### /Users/alexirion/Desktop/Screen Shot 2017-05-04 at 1.01.52 PM.png

##### How do I use the tool?

##### Our tool can be run 3 different ways:

##### python elit3\_h4ck.py –efile emails.txt

##### The file ‘emails.txt’ provided should be a list of email addresses separated by a new line. Running the script with the –efile option doesn’t do any email collection since the file has all of the direct addresses we want to send our attack to preloaded.

##### python elit3\_h4ck.py –nfile names.txt

##### The file ‘names.txt’ provided should be a list of names of UT students, professors, and/or faculty separated by a newline. Running the script with the –nfile option will query the UT directory service (https://directory.utexas.edu) for each name in the file. Our script sends a GET request to the directory service and scrapes for the submitted persons email address on the page. After the script queries the directory service for each name in the provided file, the program has a list of email addresses to direct the phishing attack towards.

##### The names in the file must be found in the UT directory service.

##### python elit3\_h4ck.py –cid #

##### The # provided should be a valid canvas UT course id that the person running the attack has access to (has membership of). The person running the script will first need to obtain a special oauth token from canvas for 3rd party application access. Next the person running the script must provide a valid canvas course ID.

##### The script gets all students/professor/TA’s email addresses from the canvas API in the course with the course ID provided.

##### This attack is directed at people in one specific class.

##### Tech specs

##### Cloned login.utexas.edu from the firefox web browser

##### Altered the website javascript code for login button on submit function

##### Hosted website on VMWare apache web server inside of cs378-ethical-hacking network

##### Python script to run exploit (python version 2.7.10)

##### Collects email addresses from UT directory web scraping, canvas API

##### Sends email through Gmail SMTP server

##### Starts infinite looped server listening for post requests

##### Problems we encountered

##### Rate limited requests to UT directory, I even got completely blocked after too many requests

##### Canvas will only give email addresses of people in courses that you are currently or formerly enrolled in

##### login.utexas.edu didn’t have any xss exploits that we could find, so we resorted to cloning the website and making it look realistic as possible

##### Attempted to alter email header fields to spoof the from mail address

##### What could have been improved?

##### The body of our email could be very directed/social engineered more so since we know that we are sending the email to other students in our class. This could include advising bars, google docs, assignment submission, canvas message notifications etc.

##### We could have used a different SMTP server that would have allowed us to spoof the from email address from address

##### We could have publicly hosted our fake website so students don’t have to be on the cs387-ethical hacking network to access it (we were lazy and financially limited)

##### We could have made our cloned website look more legit by getting a certificate, real URL, etc.