

**Computer Security**

**Tutorial 6: Authentication**

**Part 1: Google Reconnaissance**

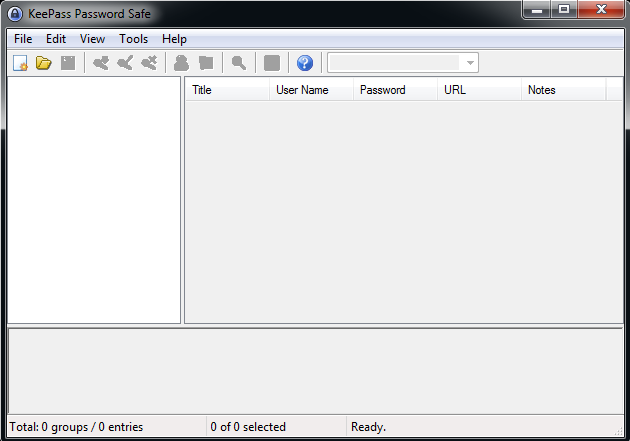
Many people use Google purely for the purpose of locating items of interest usually of an entertaining nature. However, the capabilities of Google can far extend into areas such identifying vulnerabilities or locating web cameras which have not been properly secured and hence have configured in a manner which has no authentication implemented. In this instance, we will use Google to locate potential usernames and password lists.

1. Open your web browser presumably Internet Explorer and go to the default Google search engine www.google.com.au
2. Click *Advanced Search* so that the Advanced Search page is displayed on your browser.
3. First we are going to see if there are any publicly available Microsoft Excel spreadsheets containing a list of usernames and passwords. Please note, we are not going to and you are not perrmited to use any usernames and passwords we find for malicious purposes.
4. We are going to use the search terms "login:" and "password=" to hopefully acquire some results.
5. Under the *Find web pages that have...*  there is a text box namely *all these words* which you need to enter "login:\*" "password=\*". For those students who are unsure the asterix represents a wild car search.
6. Next under the *File type* field select *Microsoft Excel* as we are looking for Excel based spreadsheets. Click the *Advanced Search*.
7. Have a browse around the the Google search results. Some files may be password protected, others will contain actual authentication credentials.
8. What are the security issues associated with performing Google reconnaissance? How can this be mitigated?

**Part 2: Password Storage Program**

In today’s online sphere people are often required to remember numerous passwords for all the different types of online and offline accounts. In this activity you will be exposed to a an open source software which will allow you to securely store your passwords. Please note downloading programs from the Internet may have a negative affect on your computer and so you untake the task at your own risk.

1. Go to [www.keepass.info](http://www.keepass.info) and follow the prompts to locate and download the latest **portable** version of keepass.
2. Because the version that you installed is a portable version, it does not install under Windows. As a result double click the KeePass.exe file.
3. As per the diagram below you will notice that the program allows you to create a database of passwords for your own safe keeping.



1. Click *File* and *New* to start a password database. Enter a strong master password which will be utilised to protect the entire database of passwords that you will storing in this exercise.
2. Click *Edit* and *Add Entry*. You will enter information about an online account that has a password.
3. Under *Group*  select an appropriate group for this account.
4. Enter a title for this account under *Title*. Under *Username* enter the username that you use to login to this account. Erase the entries under *Password:* and *Repeat:* and enter the password that you use for this account and confirm it.
5. Enter the URL for this account URL:
6. Click OK.
7. Click *File* and *Save* Enter your last name as the filename and click Save.
8. Exit KeePass.
9. If necessary navigate to the location of KeePass and double click the file KeePass.exe to launch the application.
10. Enter your master password to open it.
11. Play around with testing out KeePass and attempt to answer the following questions.
    1. Outline three benefits to using KeePass or a similar program?
    2. Outline three issues to using KeePass or a similar program?
    3. How could the program be improved to increase the level security?

**Part 3: Questions**

1. Mary developed a password authentication system which only permits eight character passwords. Assuming the full ASCII character set can be used, what is the total number of passwords that could be constructed from such a character set based on the password length?

Assuming Mary has not implemented any password lockout features and an attacker could test 1 password every nanosecond, how long (on average) would it take an attacker to guess such a password?

1. Bob tried to break into an Automated Teller Machine (ATM) using a screwdriver and a paper clip. Fortunately, the only damage he caused was breaking five different keys on the numeric keypad and jamming the card reader, at which point he heard Amy approaching, so he hid. Amy walked up, put in her ATM card, successfully entered her secret 4-digit PIN, and withdrew some money. Because the card reader was jammed she was not able to get card back and decided to go home and call the bank to report the incident.

Bob went back to the ATM, and was able to push Amy's card back in which activated the ATM and requested a pin number. He started entering numbers to try and discover Amy's PIN number and steal money from her account. What is the worst-case number of PINs that Bob has to enter before correctly discovering Amy's PIN?

1. A thief walks up to an electronic lock with a 10-digit keypad and he notices that all but four of the keys are covered in dust. On this type of lock each key is only used once in the four digit combination. He assumes that the keys which are not covered in dust must be the ones that are continually utilized. What is the worst case number of combinations he must now test to try to open this lock using a brute-force attack? *Think permutations!*