Week 12 Tutorial (Data Privacy)

Part A: Legal Requirements for Dealing with Private Data

You can look at Australian's data privacy act (Privacy Act of 1988, Privacy Regulation 2013 & Privacy Amendment (Notifiable Data Breaches) Act 2017), which is under the Office of the Australian Information Commissioner (OAIC). Have a look at the "Guide to securing personal information" website from the Office of the Australian Information Commissioner (OAIC), which "provides guidance on the reasonable steps entities are required to take under the Privacy Act 1988":

https://www.oaic.gov.au/agencies-and-organisations/guides/guide-to-securing-personal-information

It includes all the amendments to the act. The article on the website is very long, so no need to read it all. Skim the sections on (they should be in order in the document):

- The information Lifecycle
- Encryption (about half way down the document)
- Access security (especially the part on "Passwords and passphrases")
- Data breaches
- Destruction or de-identification of personal information

What obligations do Australian companies have with respect to handling personal information?

- What information must be encrypted?
- How should passwords be chosen?
- What must happen in the case of a data breach? Must the company prepare for them?
- How can personal information be destroyed?

Part B: Understanding Data Leaks

B1: An Email Leak

In this section we investigate the ramifications of a leak of sensitive information. The information in this case is all emails for a particular corporation. (For further background with regards to this case, https://en.wikipedia.org/wiki/Enron_Corpus)

Compared with transaction databases that contain financial information (such as credit card numbers and bank account numbers), emails are often considered not quite as sensitive when it comes to privacy, but we will show that leaks of this type of information can have huge ramifications on individuals' privacy.

Open the BASH (UNIX) Shell terminal, then download (or copy to the Linux/MacOS machine) the file `ENRON_emails_sample.tgz` from Moodle. Extract the file by using the "tar" command as follows:

```
tar -xvzf ENRON_emails_sample.tgz
```

Find out what the `tar` command does, especially with the `-z` option (should already been introduced from previous tutorial).

A new directory called "`ENRON_emails_sample`" should be produced. You can check the size of the directory by typing:

```
du -sh ENRON_emails_sample
```

The directory contains a sample of emails written by senior executives of the now defunct ENRON energy corporation. As a result of a court case into illegal activities at the energy company, the data was made public (`https://www.cs.cmu.edu/~./enron/`). Imagine this email corpus wasn't public information, but had been stolen as part of a leak from the corporation. What could a hacker do with the information?

Perhaps the most obvious and least sensitive information in people's emails is the email addresses of the people they send emails to. Given a dump of emails, a hacker can easily identify lines containing email addresses using a simple grep for the '`@`' character:

```
cd ENRON_emails_sample
grep -r "@" * | less
```

Here the '`-r`' flag tells grep to descend recursively into subdirectories, and the '`*`' character is expanded by the shell to be all files/directories in the current directory.

Furthermore, a hacker can quickly extract just the email addresses using a slightly more complicated regular expression (RegEx) (you will encounter this in other Units later):

```
grep -iroh "[A-Z0-9]\+@[A-Z0-9]\+\.[A-Z0-9]\+" * | less
```

Here the '`i`' flag tells grep to ignore case, the '`o`' flag says to output only the matching part (word) from each line and the '`h`' flag suppresses the input filename in the output. Piping the results to "`sort`" and "`uniq`" (i.e. replacing the "`less`" with "`sort | uniq > emails.txt`") would result in a list of unique email addresses that would be very useful to a hacker for spamming purposes.

Similar to email addresses, a hacker could easily extract private cell phone numbers from the dataset:

```
grep -ir "[0-9]\{3\}" * | grep "cell" | grep "phone" | less
```

But email addresses and cell phone numbers are just the tip of the iceberg as far as privacy invasions go. Other regular expressions would easily turn up people's home addresses as well as very sensitive information such as credit card numbers, tax file numbers, usernames and passwords for other websites, etc.

As well as identity theft, hackers can easily find information useful for other crimes. For instance, they easily find out about people's plans, such as who might be going on vacation soon, e.g. by doing a simple grep for the term "on vacation":

```
grep -ir "on vacation" * | less
```

Obviously that information combined with people's home addresses (that can also often be found from their emails) can be quite dangerous. Hackers could also look for compromising information that might be useful for blackmail purposes (e.g. by searching for words like "affair" or "bribe").

Some questions that you can think about.

- What other types of private information might a hacker get their hands on from people's emails?

- Since email accounts contain so much sensitive information, providers of email services usually don't even allow their own employees access to the email accounts of their subscribers. Why would Google not trust their own employees to access the contents of emails on your Gmail account?

- How then is access control to email enforced? Hint: by simply encrypting all messages, only the person with access to the private key will be able to decrypt and view them.

- How can leaks of private email information be prevented?

    - Encryption of all sensitive data within the organisation. (Sensitive information such as emails should never be stored unencrypted.)

    - Use of firewalls and other network access controls.

- Continual logging of access metadata and the training of machine learning algorithms to detect unusual access behaviour.

- Use of strong and frequently updated passwords or other more reliable forms of user identification.

- How else might leaks be prevented?

B2: Famous Data Breaches

The following website provides a visualisation of recent data breaches. Play around with the visualisation to understand what types of data breaches have occurred and why.

http://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/

Click on some of the bubbles (maybe use Zoom, Nintendo or Facebook) to investigate further the leaks. For a couple of the breaches try to answer the following questions (we will leave it to you to start forum discussions if you like):

- What type of sensitive data was stolen?
- Who was affected by the data breach?
- How were the victims (those whose private data was taken) informed of the breach?
- How was the data stolen?
- Could the breach have been avoided? I.e. what controls could have been put in place to prevent it?

Part C: Website Privacy Policy / Terms of Use Agreements

C1: Google's Privacy Policy

Have a look through the Privacy Policy page for Google's website:

`https://policies.google.com/privacy?hl=my`

- What types of data does Google collect?
  - "We want you to understand the types of information we collect as you use our services"
- What does Google use your data for?
  - "We use data to build better services"
- What controls do you have over what data is collected?
  - "You have choices regarding the information we collect and how it's used"
- How does Google protect your data?
  - "We build security into our services to protect your information"

Have a look also at your account information and history at the following link to better understand what controls users have over what personal information Google stores about them:

`https://myaccount.google.com/privacy?hl=en#accounthistory`

C2: Developing a Privacy Policy page

Imagine you are setting up your own website for a new Data Science project. You can make a simple Privacy Policy page for your new company, by filling in the details here:

`https://visser.io/tools/living-in-australia/privacy-policy-generator/`

Depending on the type of user information your company collects, you would of course need to update the basic Privacy Policy statement produced above.