

Final Proposal

Author: Wolfgang C. Strack

CIS29 Spring 2015

Purpose

Use threading techniques

Background

You have been tasked by the government to break into a corrupt private banking firm's database. This bank caters to criminals and is infamous for secretly providing monetary oversight of illegal activities. The reason why you are breaking into their user database is to track and arrest their clients, nearly all of which have committed serious crimes. You are working with teams of other government hackers to complete this mission. While the other team is in charge of breaking in, your team is in charge of processing the user data as quickly as possible so that you will not be detected.

Description

You will be in charge of handling a multi-threaded XML data processor, because the bank database stores parts of the encrypted user data in XML. You will also be in charge of taking this encrypted data and decrypting it to uncover the identities of this illegal bank's clients.

Fortunately, the bank's database designers were not very bright so data has leaked involving the security design. From these leaks, it turns out that the usernames have been sorted in alphabetical order and are stored into three files; one file is for usernames A-H, the second file is for usernames I-P, and the third file is for usernames Q-Z.

Since you want to get out of the database as quickly as possible, you will use threading to parse the three different files at the same time and store the data into three XML node lists (use the XML node list you built in the previous lab). Since you cannot copy or download this data (and risk getting detected), you will also design a class that decrypts the user data immediately after parsing and storing it into the three lists.

Instructions

Your employer has given you three sample files to work with that they obtained through the security leaks. Design a class (or classes) that use(s) your XML parser from the previous lab and use threading to asynchronously parse the XML data from the three files. Here is what one user node looks like in the XML file:

```
<user>
    <name>sampleUsername</name>
    <first>ax71ef$%^adf9</first>
    <last>#@afd#a^d&jkl</last>
</user>
```

After parsing this data and storing it into a list, decrypt the data in the nodes named <first> and <last>. This data represents the “first” and “last” names of the bank’s clients. Again, because of the security leaks, your employer has obtained the encryption/decryption algorithm which you will use to decrypt this data. Use multi-threading again to process the encrypted data in separate lists to maximize speed. Finally, print to screen the decrypted names next to their username in a format similar to this:

```
sampleUsername: lastName, firstName
```

Steps

1. Design a class (or classes) that use(s) your XML parser from the previous lab and use(s) multi-threading to parse multiple files at the same time. Parse the XML files provided in “Lab materials”.
2. After parsing the data, decrypt the data stored in the nodes named <first> and <last>.
3. Print the decrypted data to the screen next to the username they belong to in a format similar to this: sampleUsername: lastName, firstName .

Note to Instructor:

This document is simply a proposal of the final lab I am designing. Therefore, it should be considered a rough draft as minor changes may be made.