

I - Pre-requisites for setting up a programming environment

1) A laptop with Mac OS High Sierra or higher:

- Install Appium (<http://appium.io/>)
- Install Anaconda Python 2.7 (<https://www.anaconda.com/distribution/>)
- Install Android Studio (<https://developer.android.com/studio>)
- Install Appium Python Client (<https://github.com/appium/python-client>)
- Install Selenium (<https://selenium-python.readthedocs.io/installation.html>)
- Install Chrome Driver (<https://sites.google.com/a/chromium.org/chromedriver/downloads>)
- Install Iterm2 terminal (<https://iterm2.com/>)
- Install Homebrew (<https://brew.sh/>)
- Install Microsoft Office (<https://www.office.com>)
- Install Text Editor (e.g., vim, emacs, vscode, sublime, notepad, etc.)
- Install Chrome web browser (<https://www.google.com/chrome/>)
- Install Latex (<http://www.tug.org/mactex/>)
- Install FileZilla (<https://filezilla-project.org/>)
- Install Eclipse IDE (<https://projects.eclipse.org/releases/kepler>)
- Install ipdb (<https://pypi.org/project/ipdb/>)
- Install sshpass (<https://gist.github.com/arunoda/7790979>)
- Install pyautogui (<https://pyautogui.readthedocs.io/en/latest/>)
- Install Cisco Anyconnect client
- Install Screaming Frog (<https://www.screamingfrog.co.uk/seo-spider/>)
- Install lxml (<https://lxml.de/parsing.html>)

2) A server with Ubuntu 16.04:

- Install Anaconda Python 2.7 (<https://www.anaconda.com/distribution/>)
- Install Apktool (<https://ibotpeaches.github.io/Apktool/install/>)
- Install tesseract (<https://github.com/tesseract-ocr/tesseract/wiki>)
- Install graphviz (<https://www.howtoinstall.co/en/ubuntu/xenial/graphviz>)
- Install openssh server (<https://help.ubuntu.com/lts/serverguide/openssh-server.html>)
- Install html2text (<http://manpages.ubuntu.com/manpages/trusty/man1/html2text.1.html>)
- Install Text Editor (e.g., vim, emacs, vscode, sublime, notepad, etc.)
- Install ipdb (<https://pypi.org/project/ipdb/>)
- Install Apache webserver (<https://httpd.apache.org/>)
- Install httrack (<https://www.httrack.com/>)

3) A mobile phone with Android OS 6.0:

- Make sure to turn on Developer Mode
- Have a Google Play Store account with gmail and password

Notes

- Make sure Bash version is at least 4.0
- Make sure the following GNU commands are available from the Bash command prompt:
find, grep, awk, sed, dos2unix, dot, curl, wget, adb, aapt, scp, apktool, split, sshpass, tesseract, html2text, uiautomatorviewer...
- Any text editor is sufficient to develop in Bash and Python
- Eclipse IDE (Kepler) is preferred if developing in Java with Soot framework
- gcc and GNU build system are preferred if developing in C
- Software version control (e.g., git, svn) is highly recommended for serious programmers
- Openssh is recommended if working in a team of at least 2 people
- Database management system can be used but not required
- Graphical softwares (e.g., Adobe products) can be used but not required
- Latex is recommended for serious programmers
- FileZilla can be used but not required
- Server secondary-storage should be large if running with thousands of apks
- Apache webserver can be replaced with a more light-weight webserver
- alias server='python -m SimpleHTTPServer' is a very useful snippet for launching a builtin python webserver
- Cisco Anyconnect client can be used for remote access from home to school if necessary
- Screaming Frog and httrack can be used but not required
- lxml can be used but not required

II – Project structure

The project consists of the following packages:

LeftoverAccountAnalyzer/lai

contains all the tools for automated testing android apps via appium

```

LeftoverAccountAnalyzer/lai_web
    contains all the tools for automated testing websites via selenium
AccountDeletionAnalyzer/analyzer
    contains all the tools related to account deletion functionality for android apps
AccountDeletionAnalyzer/analyzer_web
    contains all the tools related to account deletion functionality for the web
AccountDeletionAnalyzer/analyzer_combiner
    contains all the tools which combine results from the two packages analyzer and analyzer_web
RetentionPeriodAnalyzer/pystatparser or AccountDeletionAnalyzer/pystatparser
    contains a parser for natural language processing
AccountDeletionAnalyzer/nlp
    contains all the tools which use pystatparser for analyzing the semantics in natural language processing
AccountDeletionAnalyzer/duc
    contains all the tools to be used with Soot framework

```

III – Notations

./	run an executable bash script
< >	some user input
[X Y]	choose either X or Y
Z	software tool Z is developed or customized in-house
—	optional

IV - Software tools and usages

apkdl.sh

synopsis	./apkdl.sh
description	scrape all the android package names from a website
options	none the input website is already hardcoded as apk-dl.com the script will return a list of android package names
dependencies	curl awk
packages	lai_web
examples	<pre>base(master) 5 ~ Documents>Projects>lai_web> ./apkdl.sh amr.shohaeib.hikamWakhawater af.hindi.stories.booktwo info.plateaukao.callipplus.free com.miningsite.kitabbulughulmaram com.dekd.apps ja.bito.aobajisho net.andromo.dev280448.app265483 com.sirma.mobile.bible.android</pre>

apk2web.sh

synopsis	./apk2web.sh <apkfile>
description	find the equivalent website address for an apk file by looking into the manifest
options	<apkfile> an apk file given an apk file, the script will return the equivalent website address
dependencies	aapt dos2unix grep

packages	lai_web
examples	<pre>base(master) 5 ~ Documents > Projects > lai_web ./apk2web.sh ~/Desktop/files/com.remitano.remitano.apk remitano.com</pre>

apk2url.sh

synopsis	<code>./apk2url.sh <packagelist> <u>auto</u></code>
description	find the equivalent website addresses for a list of android package names via google
options	<p><packagelist> a text file containing android package names</p> <p>the <u>auto</u> flag is optional: if present, the script will automatically pick the first website address from google candidate search results if not present, the script will let testers pick a website manually from the google candidate search results</p>
dependencies	url.py
packages	lai_web
examples	<pre>base(master) 6 ~ Documents > Projects > lai_web ./apk2url.sh test_packages.txt auto ----- Searching url for apk: adidas.apk (search terms : adidas) [0] https://www.adidas.com/us [1] https://www.adidas-group.com/ [2] https://www.freepnglogos.com/pics/adidas-logo-png [3] https://en.wikipedia.org/wiki/Adidas [4] https://www.zappos.com/c/adidas [5] https://www.instagram.com/adidas/?hl=en [6] https://www.finishline.com/adidas-originals [7] https://twitter.com/adidas?lang=en [8] https://www.amazon.com/stores/adidas/adidas/page/5E398A61-45C7-46F9-A6C6-5B4797CC5063 [9] https://www.facebook.com/adidas ----- Searching url for apk: adobescan.apk (search terms : adobescan) [0] https://acrobat.adobe.com/us/en/mobile/scanner-app.html [1] https://apps.apple.com/app/adobe-scan-pdf-scanner-ocr/id1199564834 [2] https://www.ghacks.net/2017/06/02/why-i-wont-be-using-adobe-scan/ [3] https://www.cnet.com/how-to/turn-your-phone-into-a-document-scanner-with-adobe-scan/</pre> <pre>master 6 ~ Documents > Projects > lai_web cat test_packages.txt adidas.apk adobescan.apk air.thix.sciencesense.beaker.apk aliexpress.apk</pre>

auto.sh

synopsis	<code>./auto.sh [<apklist-directory> <server-username> <server-password> <wait-time-in-secs> collect <apkdir> apk2web]</code>
description	an automatic script to collect apks and websites without any manual intervention
options	<p>collect</p> <p>automatically go to google play store, download apks, and upload them to a dedicated server</p> <p><apklist-directory></p> <p>a directory which contains files whose contents are android package names</p> <p><server-username></p> <p>ssh username</p> <p><server-password></p> <p>ssh password</p> <p><wait-time-in-secs></p> <p>time to wait between each batch of apk files finish uploading to the server</p> <p>apk2web</p> <p>automatically find equivalent websites from a directory containing all apk files pulled from the mobile phone</p> <p><apkdir></p> <p>a directory which contains all apk files pulled from the android mobile phone</p>
dependencies	lac_batch.sh

	upload.sh apk2web.sh
packages	lai_web
notes	appium is required to run the script
examples	<pre>base > master ▶ 6 ? ~ Documents > Projects > lai_web ▶ ./auto.sh 200minus guest tseug 300 collect MAKE SURE APPIUM IS RUNNING OR CTRL-C TO ABORT..... ===== =====Using apk data file 200minus/xab=====</pre> <pre>base > master ▶ 6 ? ~ Documents > Projects > lai_web ▶ ./auto.sh ~/Desktop/files apk2web com.azure.authenticator.apk, com.betterhelp.apk.www.betterhelp.com com.opera.browser.beta.apk, com.remitano.remitano.apk.remitano.com net.wargaming.wot.blitz.apk,</pre>

batch.sh

synopsis	./batch.sh <apk-directory> [install uninstall remove signup login delete manual]
description	a tool for testers to work with a directory containing apk files (working in batch mode)
options	<p><apk-directory> a directory which contains apk files</p> <p>choose one of the following options:</p> <ul style="list-style-type: none"> install install all the apk files in the directory on the android mobile phone uninstall uninstall all the apk files in the directory from the android mobile phone remove remove the <apk-directory> and all the output directories _out and error directories _err, ready for next batch signup automatically signup a user account for each of the apk file in <apk-directory> login automatically login the user account for each of the apk file in <apk-directory> delete automatically delete the user account for each of the apk file in <apk-directory> manual manual intervention from testers, break into ipdb console and finish any of the options above testers can also use function <i>dump_text</i> builtin the tool to do screen scraping on the android phone
dependencies	adb aapt grep manual.py delete.py login.py signup.py
packages	lai lai_web
notes	appium is required to run the script depending on the chosen option, the tool will save log files into _out and _err directories, from which statistical analysis can be performed

```

examples base ➤ master ➤ ~ ➤ Documents > Projects > lai ➤ ./batch.sh dir_033 signup
>>>>>>>>>>>>>> Running APK : dir_033/zillow.apk <<<<<<<<<<<<
[SCREEN] #0 | BEGIN |
[Activity] com.zillow.android.re.ui.onboarding.view.LoginOnboardingActivity
Welcome!
Enter your email to get started
Email address
Continue
Or
Skip
googleplus
com.zillow.android.zillowmap:id/action_bar_root
android:id/content
com.zillow.android.zillowmap:id/fragment_container
com.zillow.android.zillowmap:id/login_layout_scroll
com.zillow.android.zillowmap:id/space
com.zillow.android.zillowmap:id/icon
com.zillow.android.zillowmap:id/login_title
com.zillow.android.zillowmap:id/login_explanation
com.zillow.android.zillowmap:id/zillow_login_layout
com.zillow.android.zillowmap:id/email_text_input_layout
com.zillow.android.zillowmap:id/label
com.zillow.android.zillowmap:id/input
com.zillow.android.zillowmap:id/login_button
com.zillow.android.zillowmap:id/third_party_signin_prefix
com.zillow.android.zillowmap:id/third_party_signin_layout
com.zillow.android.zillowmap:id/google_sign_in_button
com.zillow.android.zillowmap:id/facebook_login_button
com.zillow.android.zillowmap:id/register_terms_button
com.zillow.android.zillowmap:id/bottom_left_button
com.zillow.android.zillowmap:id/toolbar_as_actionbar
-----
[SCREEN] #1 | IN REGISTER |

base ➤ master ➤ ~ ➤ Documents > Projects > lai ➤ ls dir_033_err/ dir_033_out/
dir_033_err/:
Zalo.log  yelp.log  youper.log  yourphonecompanion.log

dir_033_out/:
Zalo.log  yelp.log  youper.log  yourphonecompanion.log

base ➤ master ➤ ~ ➤ Documents > Projects > lai ➤ ./batch.sh dir_033 manual
>>>>>>>>>>>>>> Running APK : dir_033/yourphonecompanion.apk <<<<<<<<<<<
--Return--
-----
          s/Projects/lai/manual.py(22)<module>()

---> 22 import ipdb; ipdb.set_trace(context=1)

ipdb> dump_text()
[Activity] Activity WelcomeActivity
Update your phone to link
Your phone needs to be on Android version 7.0 or higher to link with your PC.
Learn more
Continue to Home Screen
Update your phone to link, Heading, Your phone needs to be on Android version 7.0 or higher to link with your PC.
Learn more, Link
Continue to Home Screen, Button
android:id/content
com.microsoft.apppanager:id/activity_welcome_root
com.microsoft.apppanager:id/error_view_root
com.microsoft.apppanager:id/error_layout
com.microsoft.apppanager:id/error_img
com.microsoft.apppanager:id/error_title
com.microsoft.apppanager:id/error_detail
com.microsoft.apppanager:id/learn_more
com.microsoft.apppanager:id/error_go_back_btn
-----
ipdb> 

master ➤ ~ ➤ Documents > Projects > lai ➤ ./batch.sh dir_033/ install
Installing dir_033/Zalo.apk
dir_033//Zalo.apk: 1 file pushed. 5.4 MB/s (41884948 bytes in 7.407s)
      pkg: /data/local/tmp/Zalo.apk
Success
Installing dir_033/yelp.apk
dir_033//yelp.apk: 1 file pushed. 6.1 MB/s (48800284 bytes in 7.591s)
      pkg: /data/local/tmp/yelp.apk

master ➤ ~ ➤ Documents > Projects > lai ➤ ls dir_033/
Zalo.apk  yelp.apk  youper.apk  yourphonecompanion.apk  zillow.apk  zomato.apk

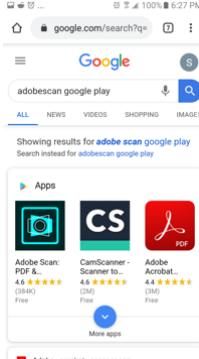
```

single.sh

synopsis	<code>./single.sh <apkfile> [download signup install uninstall login delete launch search manual]</code>
description	a tool for testers to work with one single apk file
options	<p><apkfile> an apk file</p> <p>choose one of the following options:</p> <ul style="list-style-type: none"> download download the <apkfile> from the server signup automatically signup a user account for the <apkfile> install install the <apkfile> on the android mobile phone uninstall uninstall the <apkfile> from the android mobile phone login automatically login the user account for the <apkfile> delete automatically delete the user account for the <apkfile> launch launch the <apkfile> and fire random events with monkey tool search automatically search apk file for a given android package name by automating google web browser manual manually break into the ipdb console
dependencies	adb monkey scp <code>signup.py</code> <code>manual.py</code> <code>delete.py</code> <code>login.py</code>
packages	lai lai_web
notes	appium is required to run the script this script is similar to batch.sh except it works on only 1 apk file at a time, can be used in tandem with batch.sh it also provides some extra commands not in batch.sh
examples	<pre>base(master) ~> Documents > Projects > lai > ./single.sh youper.apk search ===== Searching Google Play Store for APK file with package name: youper.apk =====</pre> <pre>base(master) ~> Documents > Projects > lai > ./single.sh dir_033/youper.apk signup >>>>>>>>>>>> Running APK : dir_033/youper.apk <<<<<<<<<<<<<<<<<<<</pre>

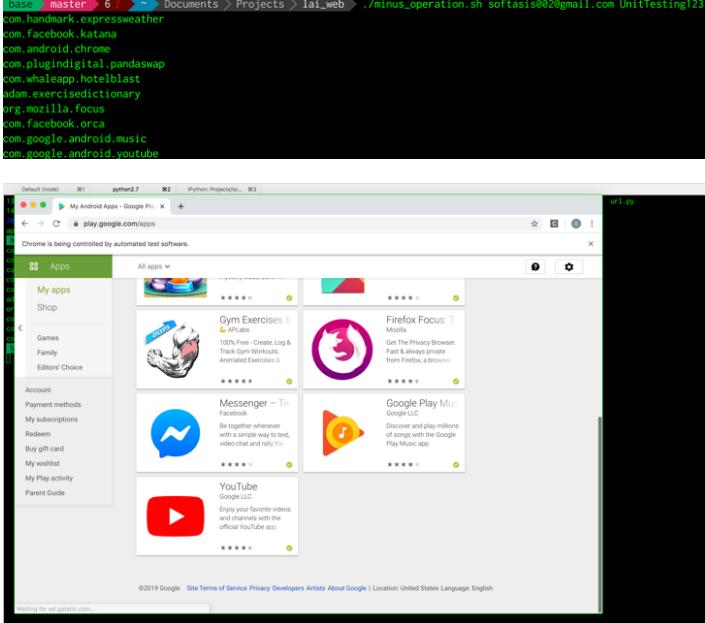
pm.sh

synopsis	<code>./pm.sh [pull uninstall list count <package-file> search <package-name> pull]</code>
description	a package manager tool to help testers manage android packages efficiently
options	<ul style="list-style-type: none"> pull pull all non-system packages from the android phone to your laptop uninstall uninstall all non-system packages currently residing in the android phone list

	<p>list all non-system packages in the android phone</p> <p>count count the number of packages after pulling</p> <p>search automatically search for android packages via google browser in the phone <package-file> a text file which contain android packages</p> <p>pull pull a specific package from the android phone to your laptop <package-name> an android package</p>
dependencies	dos2unix sed cut adb
packages	lai
examples	<pre>base(master) ~ Documents > Projects > lai > ./pm.sh packages.txt search ===== Searching Google Play Store for APK file with package name: adam.exercisedictionary ===== Press [enter - continue] ctrl-c to exit] ::: ===== Searching Google Play Store for APK file with package name: adidas ===== Press [enter - continue] ctrl-c to exit] ::: ===== Searching Google Play Store for APK file with package name: adobescan ===== Press [enter - continue] ctrl-c to exit] :::</pre>  <pre>base(master) ~ Documents > Projects > lai > ./pm.sh list com.microsoft.appmanager io.appium.settings com.betterhelp com.zing.zalo com.roku.remote com.sega.streetsofrafge2</pre>

minus_operation.sh

synopsis	<code>./minus_operation.sh <gmail> <gpassword></code>
description	auto login google play store website (<i>My apps</i> page) and calculate the difference between all the android packages listed there versus all the android packages listed in the mobile phone
options	<gmail> google email address to login play store <gpassword> google password to login play store
dependencies	my_apps.py awk dos2unix
packages	lai_web

notes	a compatible <i>chromedriver</i> is required for using with Chrome web browser
examples	<pre>base master > 6 / ~ Documents > Projects > lai_web > ./minus_operation.sh softasis00@gmail.com UnitTesting123</pre> 

lac batch.sh

synopsis	<pre>./lac_batch.sh [<gmail> <gpassword> minus </minus/laXX> install <package-name> <app-email> <app-password> clean pull uninstall </minus/laXX> remove]</pre>
description	when users uninstall apps on their android phones, they may have forgotten to delete their user accounts for those apps, the script can help to automatically cleanup those accounts, it also other functions to manage apk files
options	<p>minus see minus_operation.sh <gmail> google email address to login play store <gpassword> google password to login play store</p> <p>install automatically searching the play store for packages provided in </minus/laXX> and install them on android phone </minus/laXX> a text file which contains a list of android package names</p> <p>clean given <app-email> and <app-password>, this will automatically delete the user account on a given <package-name> <package-name> the package-name of an app currently residing on android phone <app-email> the email that user has provided when signing up with an app on android phone <app-password> the password that user has provided when signing up with an app on android phone alternate- we can even delete using ./single.sh <apk> delete script, refer to lai directory</p> <p>pull see pm.sh</p> <p>uninstall see pm.sh</p> <p>remove remove all log directories _out and _err, ready for the next batch</p>

	<pre></minus/laXX></pre> <p>a text file which contains a list of android package names</p>
dependencies	<pre>minus operation.sh gsplit store.py dos2unix adb grep lac adf.sh cleanup.py</pre>
packages	<pre>lai_web lai</pre>
notes	appium is required to run the script
examples	<pre>base[master]:8->Documents>Projects>lai_web>./lac_batch.sh minus/la03 softasis007@yahoo.com UnitTesting[23]: clean Cleaning com.jawbone.upopen ----- Not AD ----- Cleaning com.withings.wiscale2 ----- AD Already cleaned! ----- Cleaning fi.polar.polarflow ----- AD BEGIN Activity .activity.login.LoginActivity Decline Accept Accept com.withings.wiscale2:id/action_bar_root android:id/context com.withings.wiscale2:id/logo com.withings.wiscale2:id/viewpager com.withings.wiscale2:id/foreground_image com.withings.wiscale2:id/title com.withings.wiscale2:id/text com.withings.wiscale2:id/foreground_image com.withings.wiscale2:id/title com.withings.wiscale2:id/text com.withings.wiscale2:id/indicator com.withings.wiscale2:id/create_account com.withings.wiscale2:id/login ----- STATE LOGIN Activity .account.ui.LoginActivity Log In Email Password Forgot your password? Log In Navigate up</pre> 

upload.sh

synopsis	<pre>./upload.sh <ssh-username> <ssh-password></pre>
description	this little snippet will upload all apk files in <i>pull</i> directory and upload to a specific directory on the dedicated server, it can replace FileZilla
options	<pre><ssh-username> ssh username to login the dedicated server <ssh-password> ssh password to login the dedicated server</pre>
dependencies	sshpass

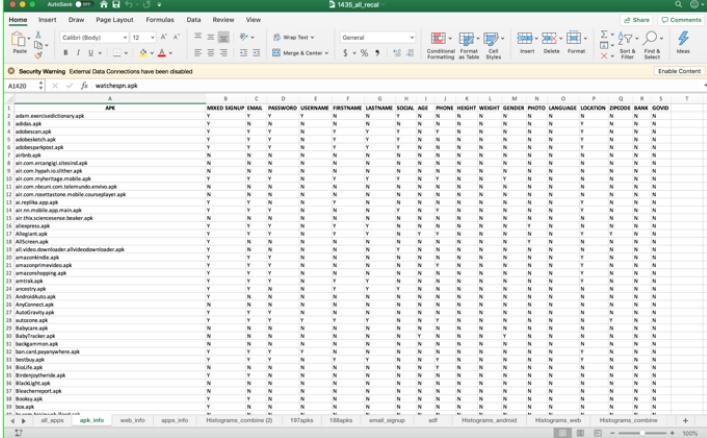
packages	lai_web
notes	a pull directory which contains all apk files must exist before running this script (e.g., after running <code>pm.sh pull</code>)
examples	<pre>base(master) ~ Documents > Projects > lai_web > ./lai.sh [...] base(master) ~ Documents > Projects > lai_web > ./lai.sh [...]</pre> <pre>base(master) ~ Documents > Projects > lai > ls -1 pull/ com.betterhelp.apk com.jawbone.upopen.apk com.microsoft.appmanager.apk com.misfitwearables.prometheus.apk com.zing.zalo.apk</pre>

lac_adf.sh

synopsis	<code>./lac_adf <package-name></code>
description	check on-the-fly whether an existing package on the android mobile device has user account deletion function or not (without using any natural language processing technique)
options	<p><package-name> an android package on the mobile device</p> <p>the script will return: <i>AD</i> means account deletion <i>Not AD</i> means not account deletion</p>
dependencies	adb dos2unix grep sed
packages	lai_web
notes	to know which packages are currently installed on the device, see <code>pm.sh</code>
examples	<pre>base(master) ~ Documents > Projects > lai_web > ./lac_adf.sh com.misfitwearables.prometheus AD base(master) ~ Documents > Projects > lai_web > ./lac_adf.sh com.zing.zalo AD base(master) ~ Documents > Projects > lai_web > ./lac_adf.sh br.com.youper Not AD</pre> <pre>base(master) ~ Documents > Projects > lai > ./pm.sh list com.misfitwearables.prometheus com.microsoft.appmanager io.appium.settings com.betterhelp com.zing.zalo com.jawbone.upopen com.roku.remote com.sega.streetsoffrage2 com.zillow.android.zillowmap</pre>

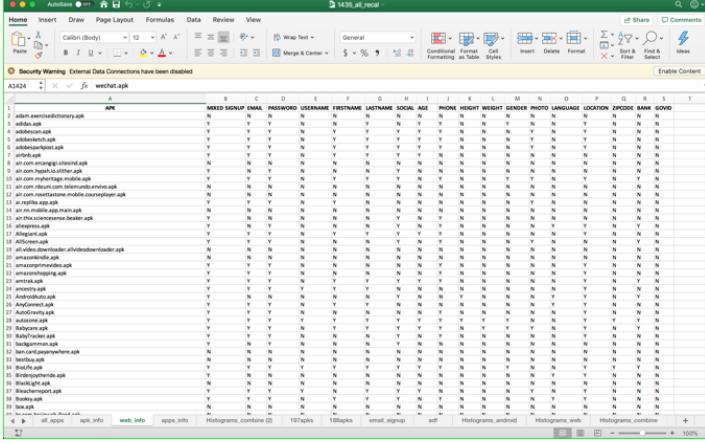
stats2.sh

synopsis	<code>./stats2.sh <log-directory></code>
description	gather statistics for each android app after running <code>batch.sh</code> script
options	<log-directory>

	a directory which contains all the log files stored under _out and _err directories
dependencies	grep
packages	lai
notes	make sure to run <code>batch.sh</code> before running this script the results can be redirected to a csv file and export to ms excel for further analyses
examples	<pre>\$./stats2.sh ~/Downloads/chunks1435_lai_logs/ adam.exercisedictionary.apk,Y,Y,Y,N,Y,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N adidas.apk,Y,N,N,N,N,N,N,N,N,N,N,N,N,N,Y,N,N,N adobescan.apk,Y,Y,Y,N,Y,Y,Y,N,Y,N,N,N,N,N,Y,N,N,N adobesketch.apk,Y,Y,Y,N,Y,Y,Y,N,N,N,N,N,N,Y,N,N,N adobesparkpost.apk,Y,Y,Y,N,Y,Y,Y,N,N,N,N,N,N,Y,N,N,N airbnb.apk,N air.com.ercangigisitesind.apk,N air.com.hypah.io.slither.apk,N air.com.myheritage.mobile.apk,Y,Y,Y,N,Y,Y,Y,N,Y,N,N,N,N,N,N,N,N,N,N,N,N air.com.nbcuni.com.telemundo.envivo.apk,N</pre> <pre>\$ ls ~/Downloads/chunks1435_lai_logs/ total 268 drwxrwxr-x 2 4096 Sep 2 22:22 dir_001_out drwxrwxr-x 2 4096 Sep 2 22:22 dir_001_err drwxrwxr-x 2 i 4096 Sep 3 16:59 dir_033_err drwxrwxr-x 2 i 4096 Sep 3 16:59 dir_033_out drwxrwxr-x 2 4096 Sep 4 11:40 dir_002_err drwxrwxr-x 2 4096 Sep 4 11:40 dir_002_out drwxrwxr-x 2 4096 Sep 4 18:14 dir_003_err drwxrwxr-x 2 4096 Sep 4 18:14 dir_003_out</pre> 

stats2 web.sh

synopsis	<code>./stats2.sh <weblog-directory></code>
description	gather statistics for each website after running <code>batch.sh</code> script
options	<code><weblog-directory></code> a directory which contains all the log files for websites after running <code>adf.sh</code> script
dependencies	grep
packages	lai
notes	make sure to run <code>adf.sh</code> for the web first before running this script

	the results can be redirected to a csv file and export to ms excel for further analyses
examples	<pre> :~/Documents/lai (master) -- \$./stats2_web.sh " "/Documents/Analyzer_Web/output/ adam.exercisedictionary.apk,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N adidas.apk,Y,Y,Y,N,Y,N,Y,N,Y,N,Y,N,Y,N,N adobescan.apk,Y,Y,Y,N,Y,Y,Y,Y,N,N,Y,N,Y,N,N adobesketch.apk,Y,Y,Y,N,Y,Y,Y,Y,N,N,Y,N,Y,N,N adobesparkpost.apk,Y,Y,Y,N,Y,Y,Y,Y,N,N,Y,N,Y,N,N airbnb.apk,Y,Y,Y,N,Y,Y,Y,N,N,N,N,N,N,N,N,N,N,N air.com.ercangigi.sitesind.apk,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N</pre> <pre>\$ ls /home/ ~/Documents/Analyzer_Web/output/ total 3116 drwxrwxr-x 2 . . 4096 Sep 23 16:07 box drwxrwxr-x 2 . . 4096 Sep 23 16:07 bestbuy drwxrwxr-x 2 . . 4096 Sep 23 16:07 burgerking drwxrwxr-x 3 . . 4096 Sep 23 16:07 amazonprimevideo drwxrwxr-x 3 . . 4096 Sep 23 16:07 camscanner drwxrwxr-x 3 . . 4096 Sep 23 16:07 amazonshopping drwxrwxr-x 2 . . 4096 Sep 23 16:07 chipotle</pre> 

adf.sh

synopsis	<code>/adf.sh [<apkfile> <website>]</code>
description	build a strings-to-screen mapping model to determine whether a given apk file has user account deletion functionality
options	<p><apkfile> an apk file</p> <p><website> a website which is downloaded as a webpage complete from google chrome</p> <p>the script returns a <i>mapping_model.dot</i> file for the corresponding apk or website in <i>output</i> directory for further analysis</p>
dependencies	apktool find awk xargs tesseract grep sed <u>semantics.py</u>
packages	analyzer analyzer_web

notes	<p>options are available depending on which package the script is running from the script depends on tesseract to convert images to text the script depends on semantics.py which use natural language processing to check the semantics of a phrase further analysis can be performed on <i>mapping_model.dot</i> file by using <code>stats.sh</code> script</p>
examples	<pre> -- :~/Documents/releases/analyzer_web (master) -- \$ cat output/com.remente.app/mapping_model.dot digraph G { }</pre> <pre> -- :~/Documents/releases/analyzer_web (master) -- \$ tree -L 2 output/ output/ └── com.remente.app ├── mapping_model.dot └── Remente - Best goal setting app for personal and organizational development_files ├── Remente - Best goal setting app for personal and organizational development.htm └── Remente - Best goal setting app for personal and organizational development.htm.txt 2 directories, 3 files</pre> <pre> -- :~/Documents/releases/analyzer_web (master) -- \$./adf.sh ~/Downloads/webdata1435/com.remente.app.apk Tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica Error in fopenReadStream: file not found Error in pixRead: image file not found: <?xml version="1.0" encoding="UTF-8" standalone="no"?> Image file <?xml version="1.0" encoding="UTF-8" standalone="no"?> cannot be read! Error during processing. Tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica Warning. Invalid resolution 0 dpi. Using 70 instead. Tesseract Open Source OCR Engine v4.0.0-beta.1 with Leptonica Error in fopenReadStream: file not found</pre> <pre> -- :~/Downloads (master) -- \$ ls ~/Downloads/webdata1435 total 3412 -rw-r--r-- 1 ' 301060 Sep 22 13:58 .DS_Store drwxrwxr-x 8 4096 Sep 22 13:58 com.adobe.adobephotoshopfix.apk drwxrwxr-x 5 4096 Sep 22 13:58 org.edx.mobile.apk drwxrwxr-x 5 4096 Sep 22 13:58 Viber.apk drwxrwxr-x 3 4096 Sep 22 13:58 ParkMobile.apk drwxrwxr-x 3 4096 Sep 22 13:58 com.mobisystems.ubreader_west.apk drwxrwxr-x 3 4096 Sep 22 13:58 com.photolab.photoeffects.apk drwxrwxr-x 3 4096 Sep 22 13:58 com.prestigio.ereader.apk drwxrwxr-x 6 4096 Sep 22 13:58 com.cyberlink.powerdirector.dra..apk</pre> <pre> -- :~/Documents/releases/analyzer (master) -- \$ cat output/com.remente.app/mapping_model.dot digraph G { "adstr:Delete your account and all of your data." --> "nattr::account_settings_delete_account_details" "adstr:Delete Account" --> "nattr::account_settings_delete_account_title" "adstr:Are you sure you want to delete your account? This action cannot be undone." --> "nattr::account_settings_dialog_delete_account_message" nattr::account_settings_delete_account_details --> "lay::item_settings_action_delete_account.xml" nattr::account_settings_delete_account_title --> "lay::item_settings_action_delete_account.xml" nattr::account_settings_dialog_delete_account_message --> "lsm::item_settings_delete_account_smali" "lsm::item_settings_delete_account_smali" --> "tplay::view_settings_account.xml" "tplay::view_settings_account.xml" --> "tplay::view_settings_account_smali" "tplay::view_settings_account_smali" --> "act::AccountSettingsView_smali" }</pre> <pre> -- :~/Documents/releases/analyzer (master) -- \$ tree -L 2 output/ output/ └── com.remente.app ├── AndroidManifest.xml ├── apktool.yml ├── assets ├── kotlin ├── lib ├── mapping_model.dot ├── original ├── res ├── smali ├── smali_classes2 └── unknown</pre>

```

$ cd ~/Documents/releases/analyzer (master) --|
$ ./adf.sh -v ~/Downloads/apkdata1435.apk
Using apktool 2.3.4 on air.com.myheritage.mobile.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: WARNING: Could not write to '/tmp'. Using '/tmp' instead...
I: WARNING: Could not find a writable directory and frameworks could go missing, please utilize --framework-path if the default storage directory is unavailable
I: Loading resource table from file: /tmp/apk.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Cont. find Ntch check in file: "drawable-hdpi-v1/com.google.simplenote.light_normal.background.9.png", Renaming it to +.png
I: Cont. find Ntch check in file: "drawable-hdpi-v1/com.google.simplenote.light_normal.icon.9.png", Renaming it to +.png
I: Cont. find Ntch check in file: "drawable-hdpi-v1/com.google.simplenote.text_light_normal.icon.9.png", Renaming it to +.png

```

parallel.sh

synopsis	<code>./parallel.sh [<apk-directory> <web-directory>] <max-num-of-cores></code>
description	an efficient way to run <code>adf.sh</code> in a multiprocessing environment
options	<code><apk-directory></code> a directory which contains all downloaded apk files <code><web-directory></code> a directory which contains all downloaded websites <code><max-num-of-cores></code> a maximum number of cpu cores assigned to run this task in parallel
dependencies	<code>adf.sh</code> <code>xargs</code>
packages	analyzer analyzer_web
notes	<max-num-of-cores> must be less than or equal to the actual number of cpu cores the server has
examples	<pre> [~-.....~/Documents/releases/analyzer (master) -- \$ time ./parallel.sh ~/Downloads/apkdata1435 36 I: Using Apktool 2.3.4 on air.com.myheritage.mobile.apk I: Using Apktool 2.3.4 on ai.replika.app.apk I: Using Apktool 2.3.4 on AutoGravity.apk I: Using Apktool 2.3.4 on adobescan.apk I: Using Apktool 2.3.4 on ban.card.payanywhere.apk I: Using Apktool 2.3.4 on AnyConnect.apk I: Using Apktool 2.3.4 on BioLife.apk I: Using Apktool 2.3.4 on armworkout.armworkoutformen.armexercises.apk.part I: Using Apktool 2.3.4 on BlackLight.apk I: Using Apktool 2.3.4 on air.thix.sciencesense.beaker.apk </pre> <pre> [~-.....~/Documents/releases/analyzer (master) -- \$ ls ~/Downloads/apkdata1435 head total 40746576 -rw-rw-r-- 1 ----- 8340776 Mar 6 2019 adam.exercisedictionary.apk -rw-rw-r-- 1 ----- 48663752 Mar 6 2019 ai.replika.app.apk -rw-rw-r-- 1 ----- 25521793 Mar 6 2019 air.com.erengigisitesind.apk -rw-rw-r-- 1 ----- 19070888 Mar 6 2019 air.com.hypah.io.slither.apk -rw-rw-r-- 1 ----- 41695680 Mar 6 2019 air.com.myheritage.mobile.apk -rw-rw-r-- 1 ----- 33306420 Mar 6 2019 air.com.rosettastone.mobile.courseplayer.apk -rw-rw-r-- 1 ----- 67746481 Mar 6 2019 air.nm.mobile.app.main.apk -rw-rw-r-- 1 ----- 49579797 Mar 6 2019 air.thix.sciencesense.beaker.apk -rw-rw-r-- 1 ----- 8923102 Mar 6 2019 armworkout.armworkoutformen.armexercises.apk.part </pre> <pre> [~-.....~/Documents/releases/analyzer (master) -- \$ ls output total 152 drwxrwxr-x 10 ' ; 4096 Dec 20 15:02 com.remente.app drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 armworkout.armworkoutformen.armexercises.apk drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 backgammon drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 air.com.rosettastone.mobile.courseplayer drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 bestbuy drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 aliexpress drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 adam.exercisedictionary drwxrwxr-x 2 ' ; 4096 Dec 20 16:30 ai.replika.app drwxrwxr-x 2; 4096 Dec 20 16:30 Babycare </pre>

```

|-- ~~~~~ :~/Documents/releases/analyzer_web (master) --|
$ time ./parallel.sh ~/Downloads/webdata1435 36
output/camscanner/CamScanner Sign up - Sync documents to cloud account and across all platforms and dev
g/Amazon Registration.htmoutput/AnyConnect/Registration.htmoutput/Birdenjoytheride/Bird - Enjoy the Rid
e - Enjoy the Ride.htmoutput/Cardboard/CardBoard1_files/channel.htmoutput/Cardboard/CardBoard1_files/
t settings - Canval.htmoutput/Cardboard/CardBoard1_files/inner.htmoutput/com.abenglish.videoClass/ABA
ardboard/CardBoard1_files/saved_resource(2).htmoutput/com.abcyo.android.games/JOIN NOW! + Premium Fam
ilayout/Cardboard/CardBoard1_files/saved_resource.htmloutput/Canvasgraphic/Collaborate & Create Amazing
com.abenglish.videoClass/ABA English_files/a8409240952.htmoutput/com.abcyo.android.games/JOIN NOW! + Pr
t/com.abenglish.videoClass/ABA English.htmoutput/ai.reprika.app/Reprika Web3.htmoutput/com.abcyo.andro
idgraphic/Account settings - Canval_files/saved_resource.htmoutput/Cardboard/CardBoard.htmoutput/ch.proto
p/Hinge.htmoutput/Canvasgraphic/Account settings - Canva_files/saved_resource.htmloutput/com.abcyo.andr

|-- ~~~~~ :~/Documents/releases/analyzer_web (master) --|
$ ls output/
total 212
drwxrwxr-x 3      4096 Dec 20 15:30 com.remente.app
drwxrwxr-x 3      4096 Dec 20 16:37 adidas
drwxrwxr-x 3      4096 Dec 20 16:37 backgammon
drwxrwxr-x 3      4096 Dec 20 16:37 AllScreen
drwxrwxr-x 3      4096 Dec 20 16:37 air.thix.sciencesense.beaker
drwxrwxr-x 3      4096 Dec 20 16:37 air.com.hypah.io.slither
drwxrwxr-x 3      4096 Dec 20 16:37 ca.shaw.android.selfserve
drwxrwxr-x 3      4096 Dec 20 16:37 Cars

```

adf_fp.sh

synopsis	<code>./adf_fp.sh [<apkfile> <website>]</code>
description	find not account deletion strings (not-ad) for an apk or a website using natural language processing
options	<code><apkfile></code> an apk file <code><website></code> a website which is downloaded as a webpage complete from google chrome if not-ad is found, the script will save it in <code>not_ad_strings.txt</code> for an equivalent apk or website in <code>output</code> directory otherwises, it will output nothing
dependencies	grep sed xargs <code>semantics.py</code>
packages	analyzer analyzer_web
notes	must run <code>adf.sh</code> or <code>parallel.sh</code> first before running this script
examples	<pre> -- ~~~~~ :~/Documents/releases/analyzer (master) -- \$./adf_fp.sh ~/Downloads/apkdata1435/vsco.apk </pre> <pre> -- ~~~~~ :~/Documents/releases/analyzer (master) -- \$ tree -L 1 output/vsco output/vsco ├── AndroidManifest.xml ├── apktool.yml ├── assets ├── kotlin ├── lib ├── mapping_model.dot ├── not_ad_strings.txt ├── original ├── res ├── smali └── smali_classes2 └── unknown </pre>

```

|-- ~Documents/releases/analyzer (master) --|
$ cat output/vSCO/not_ad_strings.txt

VSCO X is a $1/year annual subscription that provides our newest tools, updated monthly. This is a pre-purchase offer. Your purchase is non-refundable. Please see the VSCO Terms of Service for more information about the VSCO X subscription. You can manage your VSCO X account settings after purchase. Per Google policy, no cancellation of the current subscription will be honored.
```

```

|-- ~Documents/releases/analyzer_web (master) --|
$ ./adf_fp.sh ~/Downloads/webdata1435/Splitwise.apk
```

```

|-- ~Documents/releases/analyzer_web (master) --|
$ tree -L 1 output/Splitwise/
output/Splitwise/
├── mapping_model.dot
└── My account __ Splitwise_files
    ├── My account __ Splitwise.htm
    ├── My account __ Splitwise.htm.txt
    ├── not_ad_strings.txt
    ├── Sign up __ Splitwise1_files
    ├── Sign up __ Splitwise1.htm
    ├── Sign up __ Splitwise1.htm.txt
    ├── Sign up __ Splitwise_files
    ├── Sign up __ Splitwise.htm
    └── Sign up __ Splitwise.htm.txt
```

```

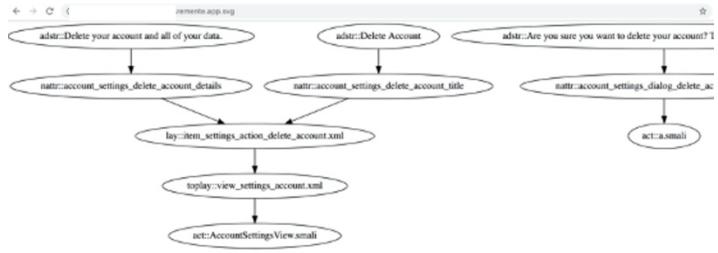
|-- ~Documents/releases/analyzer_web (master) --|
$ cat output/Splitwise/not_ad_strings.txt

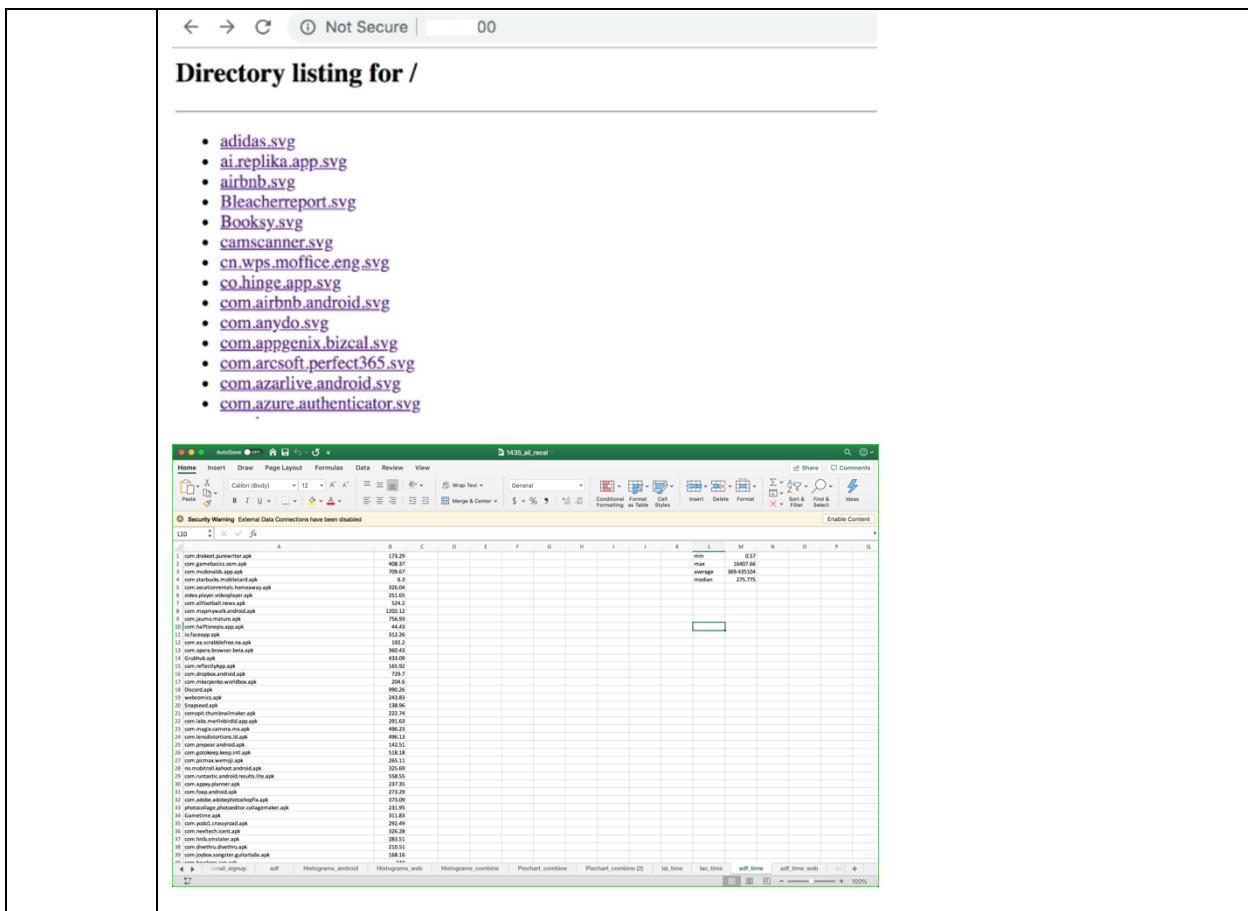
* When an expense is edited/deleted ![[Email](./My account __ Splitwise_files)]([Delete your account](https://secure.splitwise.com/users/27567203))
```

parallel_fp.sh

synopsis	<code>./parallel.sh [<apk-directory> <web-directory>] <max-num-of-cores></code>
description	an efficient way to run <code>adf_fp.sh</code> in a multiprocessing environment
options	<code><apk-directory></code> a directory which contains all downloaded apk files <code><web-directory></code> a directory which contains all downloaded websites <code><max-num-of-cores></code> a maximum number of cpu cores assigned to run this task in parallel
dependencies	<code>adf.sh</code> <code>xargs</code>
packages	analyzer analyzer_web
notes	<code><max-num-of-cores></code> must be less than or equal to the actual number of cpu cores the server has this tool is similar to <code>parallel.sh</code>
examples	<pre> -- ~Documents/Analyzer (master) -- \$ time ./parallel.sh ~/Downloads/apkdata1435 36</pre> <pre> -- ~Documents/Analyzer_Web (master) -- \$ time ./parallel_fp.sh ~/Downloads/webdata1435 36</pre>

stats.sh

synopsis	<code>./stats.sh [time dot svg fp]</code>
description	a tool to extract statistical information from log files
options	<p>time output time in seconds for each app (apk or website)</p> <p>dot copy all valid <i>mapping_models.dot</i> files to <i>analyses/dot</i> directory (see adf.sh for more information)</p> <p>svg convert those <i>mapping_model.dot</i> files to svg files</p> <p>fp copy all <i>not_ad_strings.txt</i> files to <i>analyses/fp</i> directory (see adf_fp.sh for more information)</p>
dependencies	find sed tail grep dot wc
packages	analyzer analyzer_web lai (deprecated) lai_web (deprecated)
notes	must run adf.sh or parallel.sh or adf_fp.sh or parallel_fp.sh first before running this script time outputs can be redirected to a csv file and export to ms excel for further analyses
examples	<pre> -- ~ :~/Documents/releases/analyzer (master) -- \$./stats.sh time armworkout.armworkoutformen.armexercises.apk.part,0.89 all.video.downloader.allvideodownloader.apk,9.18 BlackLight.apk,9.17 BabyTracker.apk,9.18 air.com.hypah.io.slither.apk,9.12 adobesketch.apk,9.18</pre> <pre> -- ~ :~/Documents/releases/analyzer_web (master) -- \$./stats.sh time com.adobe.adobephotoshopfix.apk,1.54 Cars.apk,KeyboardInterrupt com.aim.racing.apk,0.24 Canvasgraphic.apk,lost sys.stderr BabyTracker.apk,2.45</pre> <pre> -- ~ :~/Documents/releases/analyzer (master) -- \$./stats.sh dot Done copying all valid mapping_models.dot files to analyses/dot directory!</pre> <pre> -- ~ :~/Documents/releases/analyzer (master) -- \$./stats.sh svg Done convert dot files to svg files</pre> 



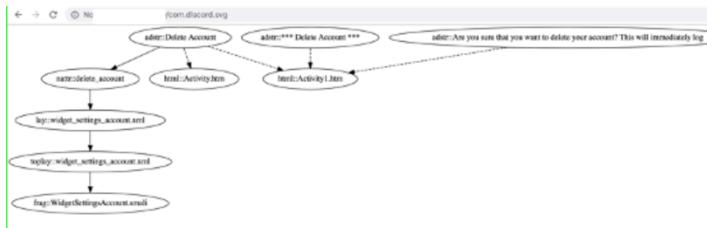
concat.sh

synopsis	<code>./concat.sh <apk-dot-directory> <website-dot-directory></code>
description	combine adf results from apks and websites together
options	<p><apk-dot-directory> the directory which contains all the dot files for apks after running <code>adf.sh</code> or <code>parallel.sh</code> tools in package analyzer</p> <p><website-dot-directory> the directory which contains all the dot files for websites after running <code>adf.sh</code> or <code>parallel.sh</code> tools in package analyzer_web</p> <p>after combining dot files, the script will convert them into svg files, also it saves the common dot files into <i>both.txt</i></p>
dependencies	head tail sed cp dot
packages	analyzer_combiner
notes	must run <code>adf.sh</code> or <code>parallel.sh</code> first before running this script (in both package analyzer and analyzer_web) in svg graph, dotted lines represent paths coming from the website, normal lines are from the apk
examples	<pre> -- ... :~/Documents/Analyzer_Combiner (master) -- \$./concat.sh ../Analyzer/analyses/dot/ ../Analyzer_Web/analyses/dot/</pre>

```
| :~ /Documents/Analyzer_Combiner (master) --|
$ ls ..../Analyses/dot | head
total 840
-rw-rw-r-- 1 . . . . . 2826 Sep 27 20:36 adidas.dot
-rw-rw-r-- 1 . . . . . 168 Sep 27 20:36 airbnb.dot
-rw-rw-r-- 1 . . . . . 172 Sep 27 20:36 ai.replika.app.dot
-rw-rw-r-- 1 . . . . . 595 Sep 27 20:36 Bleacherreport.dot
-rw-rw-r-- 1 . . . . . 138 Sep 27 20:36 Booksy.dot
-rw-rw-r-- 1 . . . . . 416 Sep 27 20:36 camscanner.dot
-rw-rw-r-- 1 . . . . . 336 Sep 27 20:36 cn.wps.moffice.eng.dot
-rw-rw-r-- 1 . . . . . 198 Sep 27 20:36 co.hinge.app.dot
-rw-rw-r-- 1 . . . . . 168 Sep 27 20:36 com.airbnb.android.dot
```

```
| :~ /Documents/Analyzer_Combiner (master) --|
$ ls ..../Analyses/dot | head
total 488
-rw-rw-r-- 1 . . . . . 84 Sep 27 20:37 adidas.dot
-rw-rw-r-- 1 . . . . . 592 Sep 27 20:37 adobescan.dot
-rw-rw-r-- 1 . . . . . 592 Sep 27 20:37 adobesketch.dot
-rw-rw-r-- 1 . . . . . 592 Sep 27 20:37 adobesparkpost.dot
-rw-rw-r-- 1 . . . . . 294 Sep 27 20:37 ai.replika.app.dot
-rw-rw-r-- 1 . . . . . 681 Sep 27 20:37 ancestry.dot
-rw-rw-r-- 1 . . . . . 664 Sep 27 20:37 ch.protonmail.android.dot
-rw-rw-r-- 1 . . . . . 592 Sep 27 20:37 com.adobe.adobephotoshopfix.dot
-rw-rw-r-- 1 . . . . . 592 Sep 27 20:37 com.adobe.comp.dot
```

```
| :~ /Documents/Analyzer_Combiner (master) --|
$ cat both.txt
adidas.dot
ai.replika.app.dot
com.anydo.dot
com.azure.authenticator.dot
com.blendr.mobile.dot
com.cyberlink.youcammakeup.dot
com.cyberlink.youperfect.dot
com.discord.dot
```



semantics.py

synopsis	<u>DEV</u> =1 python semantics.py <paragraph>.*<verb>.*<noun>.*
description	check the semantics of a paragraph against a special regular expression using natural language processing techniques
options	<p><paragraph> a collection of sentences preferred to be in English</p> <p><verb> an english verb</p> <p><noun> an english noun</p> <p><u>DEV</u> =1 is optional: if present, a constituent tree will be printed and vice versa</p> <p>the script will return: (‘TP’, ‘’) means the paragraph’s true meaning has been understood as intended by regex .*<verb>.*<noun>.*</p>

	(‘FP’, some reason) means the paragraph’s true meaning has not been understood by regex .*<verb>.*<noun>.*
dependencies	<code>pystatparser</code> <code>nltk</code>
packages	<code>nlp</code>
notes	on linux or unix system, it is highly recommended to set PATH variable pointing to this package so that the command can be used through the system and as a dependency to other scripts currently the lab server file /etc/bash.bashrc is set up as : export PATH=\$PATH:/home/user/Documents/Achilles/py
examples	<pre>documents/releases/nlp (master) -- \$ __DEV__=1 python semantics.py "please provide your password in order to delete this account." ./delete+account.s S+PRAG ADJP S+VP VP NP JJ VB PRP\$ NN please provide your password IN order NP IN NN in order PP TO VB to delete VP DT NN this account . ('TP', '') documents/releases/nlp (master) -- \$ python semantics.py "sorry, online ordering services have been disabled for this account due to suspicious behaviour" ./disable+account.s ('FP', 'succeeding subtree is not a Noun Phrase')</pre>

`context.py`

synopsis	<code>python context.py <document></code>
description	an attempt to classify a document content on the difficulty levels of the user account deletion process using natural language processing techniques
options	<document> a textfile the script should return one of the following values in increasing difficulty order: -1: no context match 0: there is an account deletion function 1: the account deletion function asks for reasons 2: the account deletion function asks to enter some inputs 3: the account deletion function asks to confirm email address 4: the account deletion function asks to contact customer service
dependencies	<code>semantics.py</code>
packages	<code>nlp</code>
examples	<pre>!-- i:~/Documents/releases/nlp (master) -- \$ python context.py /home/ /Documents/Analyzer_Web/output/MeetMe/MeetMe_\Delete\ Account.htm.txt 2</pre>

```

[~/Documents/releases/nlp (master) --]
$ cat ./Documents/Analyzer_Web/output/MeetMe/MeetMe\_Delete\ Account.htm.txt | sed '/^$/d'
* Meet
* Chat_1
* Discuss
* Me
* [./MeetMe _ Delete Account_files/report_2x.png] REPORT ABUSE!
*   o Settings
*   o Feedback
*   o Help
*   o Language
*   o Logout
*   o About_Us
*   o Terms_&_Privacy
*   o COPPA
*   o Safety
*   o Press
*   o ©_2019_MeetMe
*** Settings ***
*** Delete Account ***
*** Delete Account ***
Activate your full account! Please click the link in the email sent to
softasis000@yahoo.com to unlock all features. Resend or change email?
* ** Account settings **
  o Privacy and content
  o Language
  o Chat
  o Email
  o Delete Account
* ** Feedback **
  o Send us your feedback
* ** More **
  o Terms of service
  o Privacy policy
  o Safety
  * Log out
*** Delete Account ***
Please enter your email and password, then press the Delete Account button to
permanently remove your account:
[Unknown INPUT type]
[*****]
[Cancel Delete Account]

```

To pull single apk's manually from android device to computer:

1. Firstly find the path of packages:

adb shell pm list packages -f -3

Output :

```

package:/data/app/com.foap.android-1/base.apk=com.foap.android
package:/data/app/com.adventure.skyticket-1/base.apk=com.adventure.skyticket
package:/data/app/com.getsomeheadspace.android-2/base.apk=com.getsomeheadspace.android
package:/data/app/com.mercariapp.mercari-1/base.apk=com.mercariapp.mercari
package:/data/app/com.multibrains.taxi.passenger.tirhal-1/base.apk=com.multibrains.taxi.passenger.tirhal

```

2. To pull a specific package- Example: com.mercariapp.mercari-1:

adb pull /data/app/com.mercariapp.mercari-1/base.apk /Users/user1/Desktop

apktool to extract retention period from apps:

synopsis	Change corresponding app folder names in the source code and run
description	A program to extract retention period patterns
options	None
dependencies	semantics.py, apktool
packages	None
notes	This program converts or reverse engineer's apk files to decoded form and creates an output directory containing decoded folders and uses 'strings.xml' file to extract retention period patterns

[Web tool to extract retention period from websites:]

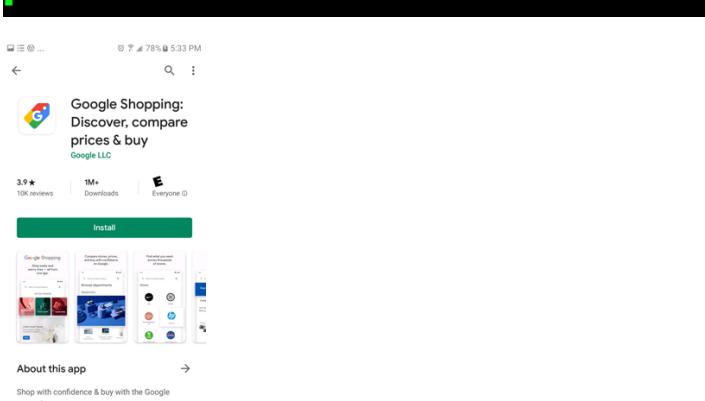
synopsis	python testing.py <foldername.apk>
description	A program to extract retention period patterns
options	None
dependencies	semantics.py
packages	None
notes	This program converts folder containing webpages to texts and creates an output directory containing converted text folders then uses these files to extract retention period patterns

[url.py]

synopsis	python url.py <android-package-file>
description	a little snippet to help testers pick the equivalent websites for android packages by listing the top ten results from google
options	<android-package-file> a text file containing android package names
dependencies	urlparse googlesearch
packages	lai_web
notes	output results will be saved in an csv file and can export to excel for further analysis
examples	<pre>base(master) 9 ? ~ Documents > Projects > lai_web python url.py test_packages.txt ===== Searching url for apk: adidas.apk (search terms : adidas) [0] https://www.adidas.com/us [1] https://www.mena.adidas.com/ [2] https://www.freepnglogos.com/pics/adidas-logo-png [3] https://en.wikipedia.org/wiki/adidas [4] https://www.adidas-group.com/ [5] https://sneakernews.com/2019/12/20/adidas-yeezy-boost-350-v2-zebra-official-release-info/ [6] https://nypost.com/2019/12/19/beyonce-adidas-to-relaunch-ivy-park-clothing-line-as-gender-neutral/ [7] https://www.yahoo.com/lifestyle/adidas-yeezy-boost-350-v2-172625443.html [8] https://www.zappos.com/c/adidas [9] https://www.finishline.com/adidas-originals select your choice (-1 == empty): </pre> <pre>base(master) 9 ? ~ Documents > Projects > lai_web cat test_packages.txt adidas.apk adobescan.apk air.thix.sciencesense.beaker.apk aliexpress.apk</pre>

[store.py]

synopsis	python store.py <android-package-names-via-stdin>
description	automatically goes to google play store, search for apks and install them to the android phone
options	<android-package-names-via-stdin> a list of android package names seperated by new line \n from stdin
dependencies	commons.py
packages	lai_web
notes	<android-package-names-via-stdin> can be replaced by a redirection from cat command appium is required to run this script

examples	<pre>base(master) 9 ? ~ Documents > Projects > lai_web cat minus/lai00 com.google.android.apps.shopping.express com.runswithtogo.togo com.stdating.positivesingles net.isitlove.cartercorp.adam com.goko.arcore.store com.chickfila.cfaflagship com.ancestry.android.apps.ancestry com.ssk.android</pre> <pre>base(master) 9 ? ~ Documents > Projects > lai_web python store.py "\$(cat minus/lai00)"</pre> 

my_apps.py

synopsis	<code>python my_apps.py <gmail> <gpassword></code>
description	automatically navigate and scrape html from <i>My apps</i> page on google play store website via chrome browser
options	<pre><gmail> google account email to login play store <gpassword> google account password to login play store</pre> <p>the script will scrape all html content from <i>My apps</i> page of google play store</p>
dependencies	<code>common_web.py</code> python selenium chrome driver
notes	make sure to install python selenium, chrome driver, chrome browser before running this script
packages	lai_web

examples

```

Default (base) M1 _jester1 (python2.7) M2 Default (base) M3
└── 2444apps
    ├── 1435_mapping_list.csv
    ├── add1.sh
    ├── change.py
    ├── common.py
    ├── del1.py
    ├── login.py
    ├── my_apps
    ├── readme.txt
    ├── single.sh
    ├── test1.txt
    ├── test_mapping_list.csv
    └── upload.sh
    └── url.py
└── apkurl.sh
└── batch.sh
└── commons.py
└── fake
    ├── labk_err
    ├── labk_out
    ├── manual.py
    ├── my_apps.py
    ├── signup.py
    ├── test.txt
    └── test_packages_mapping_list.csv

```

```

Default (base) M1 _jester1 (python2.7) M2 Default (base) M3
└── 2444apps
    ├── 1435_mapping_list.csv
    ├── add1.sh
    ├── change.py
    ├── common.py
    ├── del1.py
    ├── login.py
    ├── my_apps
    ├── readme.txt
    ├── single.sh
    ├── test1.txt
    ├── test_mapping_list.csv
    └── upload.sh
    └── url.py
└── apkurl.sh
└── batch.sh
└── commons.py
└── fake
    ├── labk_err
    ├── labk_out
    ├── manual.py
    ├── my_apps.py
    ├── signup.py
    ├── test.txt
    └── test_packages_mapping_list.csv

```

```

Default (base) M1 _jester1 (python2.7) M2 Default (base) M3
└── 2444apps
    ├── 1435_mapping_list.csv
    ├── add1.sh
    ├── change.py
    ├── common.py
    ├── del1.py
    ├── login.py
    ├── my_apps
    ├── readme.txt
    ├── single.sh
    ├── test1.txt
    ├── test_mapping_list.csv
    └── upload.sh
    └── url.py
└── apkurl.sh
└── batch.sh
└── commons.py
└── fake
    ├── labk_err
    ├── labk_out
    ├── manual.py
    ├── my_apps.py
    ├── signup.py
    ├── test.txt
    └── test_packages_mapping_list.csv

```

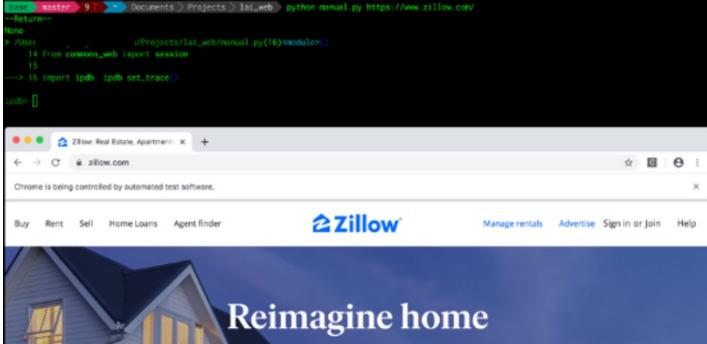
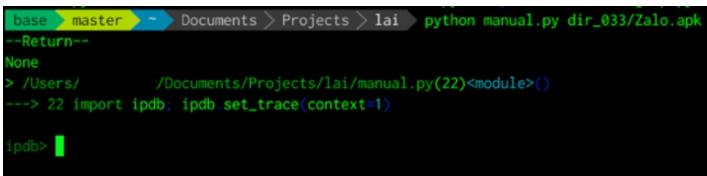
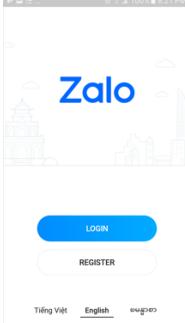
```

Default (base) M1 _jester1 (python2.7) M2 Default (base) M3
└── 2444apps
    ├── 1435_mapping_list.csv
    ├── add1.sh
    ├── change.py
    ├── common.py
    ├── del1.py
    ├── login.py
    ├── my_apps
    ├── readme.txt
    ├── single.sh
    ├── test1.txt
    ├── test_mapping_list.csv
    └── upload.sh
    └── url.py
└── apkurl.sh
└── batch.sh
└── commons.py
└── fake
    ├── labk_err
    ├── labk_out
    ├── manual.py
    ├── my_apps.py
    ├── signup.py
    ├── test.txt
    └── test_packages_mapping_list.csv

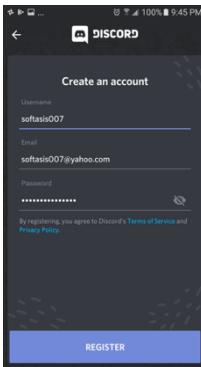
```

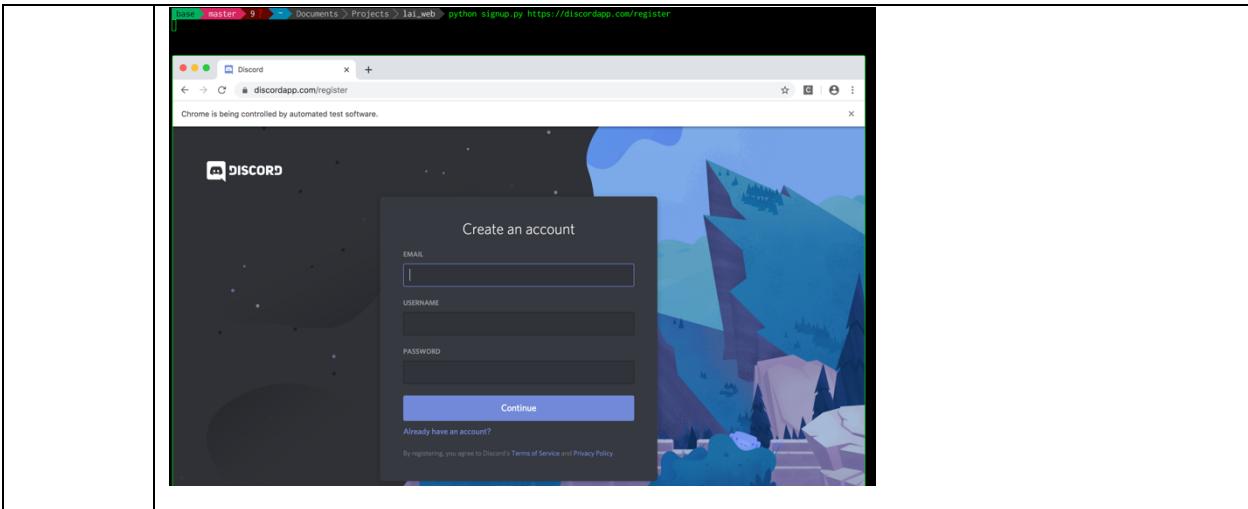
manual.py

synopsis	python manual.py <apkfile> <website-address>
description	automatically launch an apkfile or a website then break into its ipdb console for manual intervention tasks
options	<apkfile> an apk file <website-address> a website address

dependencies	<code>commons.py</code> <code>commons_web.py</code> ipdb
packages	lai lai_web
notes	appium is required if running the script from package <code>lai</code> selenium, chrome driver, and chrome browser are required if running the script from package <code>lai_web</code>
examples	<pre>base(master) ~ Documents > Projects > lai_web > python manual.py https://www.zillow.com --Return-- None > /Users/lai/... ./Documents/Projects/lai/manual.py(22)<module>() ---> 22 import ipdb; ipdb.set_trace() (ipdb) dump_text() [Activity] .ui.ZaloLauncherActivity LOGIN REGISTER Tiếng Việt English [Activity] android:id/content com.zing.zalo:id/zalo_view_container com.zing.zalo:id/layoutStartupContainer com.zing.zalo:id/imgAppLogo com.zing.zalo:id/btnLogin com.zing.zalo:id/btnRegisterUsingPhoneNumber com.zing.zalo:id/str_language_applied_vi com.zing.zalo:id/str_language_applied_en com.zing.zalo:id/str_language_applied_my ----- (ipdb) </pre>  <pre>base(master) ~ Documents > Projects > lai > python manual.py dir_033/Zalo.apk --Return-- None > /Users/lai/... ./Documents/Projects/lai/manual.py(22)<module>() ---> 22 import ipdb; ipdb.set_trace(context=1) (ipdb) </pre>  

signup.py

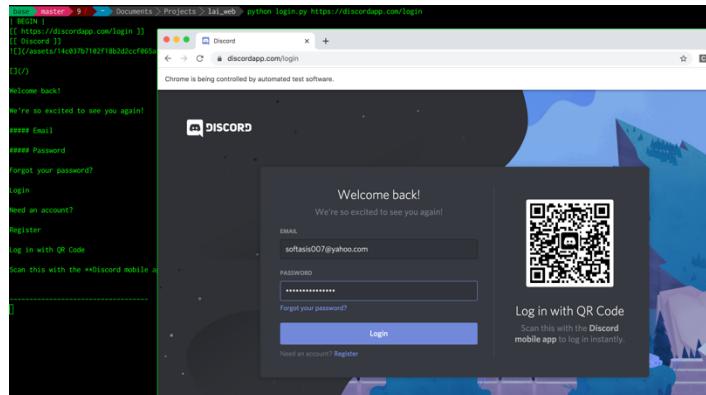
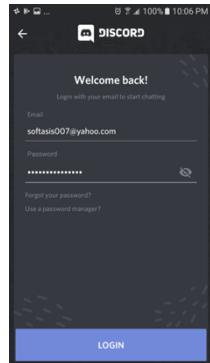
synopsis	<code>python signup.py [<apkfile> <website-address>]</code>
description	an attempt to automatically signup a user account via the android application or its equivalent website
options	<apkfile> an apk file <website-address> a website address
dependencies	<code>commons.py</code> <code>commons_web.py</code>
packages	lai lai_web
notes	appium is required if running the script from package lai selenium, chrome driver, and chrome browser are required if running the script from package lai_web user account information is hardcoded in commons.py or commons_web.py
examples	<pre>base(master) ~ Documents Projects lai python signup.py dir_033/com.discord.apk [SCREEN] #0 BEGIN [Activity] .app.AppActivity Welcome to Discord Login or Register to get started Login Register Discord com.discord:id/action_bar_root android:id/content com.discord:id/auth_landing_content com.discord:id/auth_landing_login com.discord:id/auth_landing_register ----- [SCREEN] #1 IN REGISTER [Activity] .app.AppActivity Create an account Username Email Password By registering, you agree to Discord's Terms of Service and Privacy Policy. Register Discord Show password com.discord:id/action_bar_root android:id/content com.discord:id/action_bar_toolbar_layout com.discord:id/action_bar_toolbar com.discord:id/auth_register_username_wrap com.discord:id/auth_register_username com.discord:id/auth_register_email_wrap com.discord:id/auth_register_email com.discord:id/auth_register_password_wrap com.discord:id/auth_register_password com.discord:id/text_input_password_toggle</pre> 



login.py

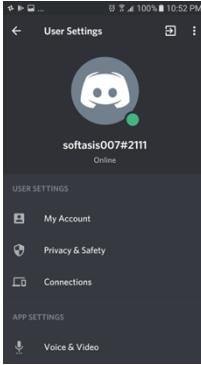
synopsis	<code>python login.py [<apkfile> <website-address>]</code>
description	an attempt to automatically login a user account via the android application or its equivalent website
options	<apkfile> an apk file <website-address> a website address
dependencies	<code>commons.py</code> <code>commons_web.py</code>
packages	lai lai_web
notes	appium is required if running the script from package lai selenium, chrome driver, and chrome browser are required if running the script from package lai_web user account information is hardcoded in commons.py or commons_web.py

```
examples
base(master) ~ Documents > Projects > lai python login.py dir_033/com.discord.apk
| BEGIN |
[Activity] .app.AppCompatActivity
Welcome to Discord
Login or Register to get started
Login
Register
Discord
com.discord:id/action_bar_root
android:id/content
com.discord:id/auth_landing_content
com.discord:id/auth_landing_login
com.discord:id/auth_landing_register
-----
| STATE LOGIN |
[Activity] .app.AppCompatActivity
Welcome back!
Login with your email to start chatting
guyresearcher@gmail.com
Password
Forgot your password?
Use a password manager?
Login
Discord
Show password
com.discord:id/action_bar_root
android:id/content
com.discord:id/action_bar_toolbar_layout
com.discord:id/action_bar_toolbar
com.discord:id/auth_login_email_wrap
com.discord:id/auth_login_email
com.discord:id/auth_login_password_wrap
com.discord:id/auth_login_password
com.discord:id/text_input_password_toggle
com.discord:id/auth_login_forgot_password
com.discord:id/auth_login_password_manager_link
```



[delete.py](#)

synopsis	<code>python delete.py <apkfile> <dotfile> <website-address> </code>
description	an attempt to automatically delete a user account via the android application or its equivalent website

options	<pre><apkfile> an apk file <dotfile> a dot file generated by <u>adf.sh</u> to guarantee that the account deletion functionlity exists on the apk <website-address> a website address</pre>
dependencies	<u>commons.py</u> <u>commons_web.py</u>
packages	lai lai_web
notes	<p>first make sure that there exists a user account with an android application or its equivalent website, possibly by running <u>signup.py</u></p> <p>appium is required if running the script from package lai, also must run <u>adf.sh</u> or <u>parallel.sh</u> first to have dot files</p> <p>selenium, chrome driver, and chrome browser are required if running the script from package lai_web</p> <p>user account information is hardcoded in commons.py or commons_web.py</p>
examples	<pre>[base](master) ~ > Documents > Projects > lai > python delete.py dir_033/com.discord.apk_dot/com.discord.dot [Activity] .app.AppCompatActivity Welcome to Discord Login or Register to get started Login Register Discord com.discord:id/action_bar_root android:id/content com.discord:id/auth_landing_content com.discord:id/auth_landing_login com.discord:id/auth_landing_register [STATE LOGIN] [Activity] .app.AppCompatActivity Welcome back! Login with your email to start chatting</pre> 

```

| STATE LOGIN |
[Activity] .app.AppCompatActivity
Welcome back!
Login with your email to start chatting
softasis007@yahoo.com
UnitTesting123!
Forgot your password?
Use a password manager?
Login
Discord
Show password
com.discord:id/action_bar_root
android:id/content
com.discord:id/action_bar_toolbar_layout
com.discord:id/action_bar_toolbar
com.discord:id/auth_login_email_wrap
com.discord:id/auth_login_email
com.discord:id/auth_login_password_wrap
com.discord:id/auth_login_password
com.discord:id/text_input_password_toggle
com.discord:id/auth_login_forgot_password
com.discord:id/auth_login_password_manager_link
com.discord:id/auth_login
-----
--Return--
None
> /User:      _/Documents/Projects/lai/delete.py(147)state_login()
--> 147      import ipdb; ipdb.set_trace(context=1)

ipdb> 

```

cleanup.py

synopsis	<code>python cleanup.py <package-name> <app-email> <app-password></code>
description	an attempt to automatically cleanup a user account for a certain android app when the user uninstalls it but forgets to do so
options	<package-name> the android package name which currently resides on the android device <app-email> the user email which the user uses to login the app <app-password> the user password which the users uses to login the app
dependencies	<u>commons.py</u>
packages	lai lai_web
notes	appium is required to run the script make sure that the <package-name> exists on the device, this can be checked by using <u>pm.sh</u>
examples	<pre> base@master: 9 ~ Documents > Projects > lai_web> python cleanup.py com.discord softasis007@yahoo.com UnitTesting123! BEGIN [Activity] .app.AppCompatActivity Welcome to Discord Login or Register to get started Login Register Discord com.discord:id/action_bar_root android:id/content com.discord:id/auth_landing_content com.discord:id/auth_landing_login com.discord:id/auth_landing_register ----- STATE LOGIN [Activity] .app.AppCompatActivity Welcome back! Login with your email to start chatting </pre>



commons.py

synopsis	<code>from commons import <some-functionality></code>
description	a python library to interact with the android mobile device via appium
options	<p><some-functionality> can be one of the followings:</p> <ul style="list-style-type: none"> external constants <ul style="list-style-type: none"> <code>_UNINSTALL_FLAG</code>, <code>EMAIL</code>, <code>PASSWORD</code>, <code>USERNAME</code>, <code>BIRTHDAY</code>, <code>FIRSTNAME</code>, <code>LASTNAME</code>, <code>PHONENUMBER</code>, <code>HEIGHT</code>, <code>WEIGHT</code>, <code>AGE</code>, <code>GENDER</code>, <code>LOCATION</code>, <code>ZIPCODE</code>, <code>SSN</code>, <code>DLN</code>, <code>CREDITCARD</code> external variable <ul style="list-style-type: none"> <code>session</code> functions: <ul style="list-style-type: none"> <code>init</code> <ul style="list-style-type: none"> deprecated <code>init2</code> <ul style="list-style-type: none"> auto launch an android app using the apk file and get a session ready <code>init3</code> <ul style="list-style-type: none"> auto launch an android app using the package name and get a session ready <code>uninstall</code> <ul style="list-style-type: none"> uninstall the android app by using the apk file <code>find</code> <ul style="list-style-type: none"> find a widget by its text <code>find_edits</code> <ul style="list-style-type: none"> find all the editable widgets <code>find_clickable_textviews</code> <ul style="list-style-type: none"> find all the textview widgets which are clickable <code>scroll_to</code> <ul style="list-style-type: none"> scroll to a specific widget identified matching its text <code>getClickableWidgets</code> <ul style="list-style-type: none"> get all clickable widgets on the screen <code>filterClass</code> <ul style="list-style-type: none"> deprecated <code>filterKeyword</code> <ul style="list-style-type: none"> deprecated <code>checkWidgetCount</code> <ul style="list-style-type: none"> deprecated <code>Init</code> <ul style="list-style-type: none"> deprecated <code>swipe_up</code> <ul style="list-style-type: none"> auto swiping up the screen <code>swipe_left</code> <ul style="list-style-type: none"> auto swiping to the left of the screen <code>dump_text</code> <ul style="list-style-type: none"> dump all the textual information of current active screen
dependencies	<ul style="list-style-type: none"> <code>appium</code> <code>time</code> <code>re</code> <code>csv</code>

	os selenium
packages	lai lai_web
notes	none
examples	none

commons_web.py

synopsis	from commons_web import <some-functionality>
description	a python library to interact with the chrome browser via selenium
options	<p><some-functionality> can be one of the followings:</p> <ul style="list-style-type: none"> external constants <ul style="list-style-type: none"> _UNINSTALL_FLAG, EMAIL, PASSWORD, USERNAME, BIRTHDAY, FIRSTNAME, LASTNAME, PHONENUMBER, HEIGHT, WEIGHT, AGE, GENDER, LOCATION, ZIPCODE, SSN, DLN, CREDITCARD external variable <ul style="list-style-type: none"> session functions: <ul style="list-style-type: none"> init <ul style="list-style-type: none"> auto launch a website address (url) and get the session ready dump_text <ul style="list-style-type: none"> dump all textual information of a webpage dump_html <ul style="list-style-type: none"> dump all html information of a webpage find_clickable <ul style="list-style-type: none"> find a clickable widget if its text matches a specific regular expression pattern find_editables <ul style="list-style-type: none"> find all editable widgets on current webpage fill_edits <ul style="list-style-type: none"> automatically fill all editable widgets with information provided via a dictionary data type nicely_quit <ul style="list-style-type: none"> nicely quit the web session signal_handler <ul style="list-style-type: none"> nicely handle external signals click <ul style="list-style-type: none"> click on a provided web element shutdown <ul style="list-style-type: none"> shutdown the session save_as <ul style="list-style-type: none"> mimic the <i>save as</i> functionality of a chrome web browser scroll_infinite <ul style="list-style-type: none"> automatically scroll the webpage until its end is reached
dependencies	selenium time html2text re sys pyautogui
packages	lai_web
notes	none
examples	none

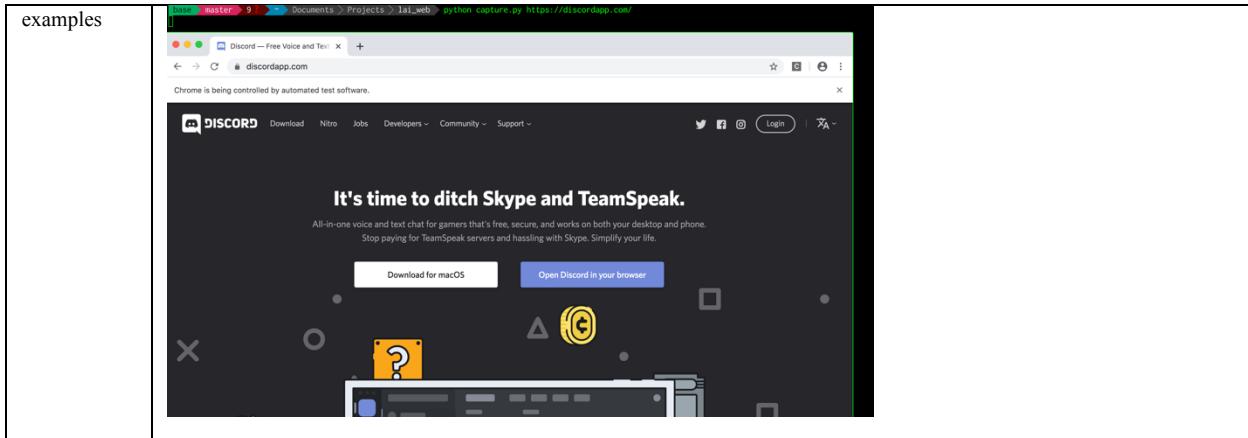
random_scraping.py

synopsis	python random_scraping.py <apkfile>
----------	--

description	a tool which triggers random events on the android app and scrape all the screen textual information
options	<apkfile> an apk file
dependencies	<u>commons.py</u> monkey
packages	lai
notes	appium is required to run the script, must set --relaxed-security option the number of events and iteration to be fired are hardcoded in the script
examples	<pre>base ➤ master ➤ ~ ➤ Documents > Projects > lai ➤ appium --relaxed-security [Appium] Welcome to Appium v1.13.0 [Appium] Non-default server args: [Appium] relaxedSecurityEnabled: true [Appium] Appium REST http interface listener started on 0.0.0.0:4723</pre> <pre>base ➤ master ➤ ~ ➤ Documents > Projects > lai ➤ python random_scraper.py dir_033/com.discord.apk [Activity] .app.AppCompatActivity Welcome to Discord Login or Register to get started Login Register Discord com.discord:id/action_bar_root android:id/content com.discord:id/auth_landing_content com.discord:id/auth_landing_login com.discord:id/auth_landing_register ----- [Activity] .app.AppCompatActivity Welcome back! Login with your email to start chatting guyresearcher@gmail.com Password Password is required. Login Discord Show password com.discord:id/action_bar_root android:id/content com.discord:id/action_bar_toolbar_layout com.discord:id/action_bar_toolbar com.discord:id/auth_login_email_wrap com.discord:id/auth_login_email com.discord:id/auth_login_password_wrap</pre>

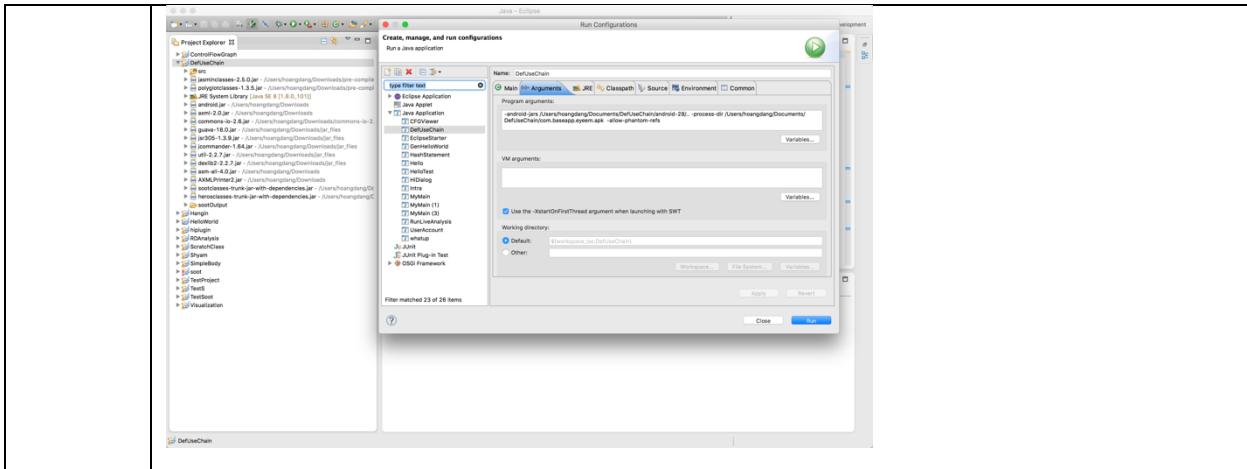
capture.py

synopsis	python capture.py <website-address>
description	a tool which mimics the save as function in google chrome web browser
options	<website-address> a website address
dependencies	<u>commons_web.py</u>
packages	lai_web
notes	selenium, web driver, and chrome browser are required before running this script make sure to have pyautoui library installed



DefUseChain.java

synopsis	see examples
description	a tool for def-use-chain analysis using Soot Framework
options	see examples
dependencies	see examples
packages	see examples
notes	Eclipse (Kepler) is currently used for running this script
examples	<pre> public static void main(String[] args) { Options o = Options.o(); o.prefer(Options.OPTION_APK_FILE); o.setArcPreOptions(Options.ARCHIVE_SOOT); o.setOutput(Options.OPTION_DEV); o.read("units.txt"); BodyTransformer bt = new BodyTransformer(); bt.adviceTransform("tp_myInstrumenter", new BodyTransformer() { @Override protected void internalTransform(FindBody b, String phaseName, boolean suppressWarnings) { FindBody.getUnits(b).units = b.getUnits(); BufferedReader writer = null; try { writer = new BufferedReader(new FileReader("units.txt")); } catch (EOFException e) { e.printStackTrace(); } } }); } </pre> <p>Output from Java application:</p> <pre> D:\Users\lai\Documents\Projects\lai_web>java -jar DefUseChain.jar Follow these steps to fix the problem: 1.) Make sure you have Java 8 or later installed. Click Here to double-check. https://www.java.com/en/download/help/jdk8-jre8.jsp 2.) Not sure whether it's a bug? Feel free to discuss the issue on the Soot mailing list: https://groups.google.com/forum/#!topic/soot-discuss 3.) Sure it's a bug? Click here to report it: https://github.com/Sable/soot/issues/new?title=soot.CompilationDeathException+when...&body=Steps+to+reproduce Please be as precise as possible when giving us information on how to reproduce the bug. </pre>



pystatparser

synopsis	<code>from stat_parser import Parser</code>
description	a python package for parsing natural languages
options	
dependencies	github
packages	<code>pystatparser</code>
notes	this open-source python package has been readjusted to work properly with python 2.7
examples	<pre>In [10]: import sys In [11]: sys.path.insert(0, "/home/hezhang/Documents/releases/pystatparser") In [12]: from stat_parser import Parser In [13]: parser = Parser() In [14]: parser.parse("This package should work with python 2.7") Out[14]: Tree('NP+S', [Tree('NP', [Tree('DT', ['this'])]), Tree('NN', ['package'])]), Tree('NP', [Tree('JJ', ['python']), Tree('CD', ['2.7'])]))]</pre>

auth

synopsis	<code>#include "auth.h"</code>
description	a small c-library built with curl to bypass basic authorization
options	<p>functions:</p> <ul style="list-style-type: none"> <code>basic_auth</code> bypass basic authorization <code>basic_get</code> simulate a GET request bypassing basic authorization
dependencies	curl Base64Encode Base64Decode crypto
packages	clang
notes	at the time of writing, httrack was a popular opensource website copier but it did not pass basic authorization this library can be compiled as a dynamic library .so to be used in complement with httrack

examples	<pre> 1 CC=gcc 2 HTTRACKPATH=/Users/l.../Downloads/httrack 3 AUTHPATH=/Users/l.../Downloads/auth 4 PREFIX=\$(HOME)/usr 5 CFLAGS=-g -O2 -I\$(AUTHPATH)' 6 LDFLAGS=-L/usr/lib -L\$(AUTHPATH)' 7 LIBS=-lz -lauth' 8 9 ##### auth library rules ##### 10 test: 11 \$(CC) auth.c main.c Base64Decode.c Base64Encode.c -lcrypto -lm -w -lcurl -o test 12 13 dylib: 14 \$(CC) -dynamiclib -dy auth.c main.c Base64Decode.c Base64Encode.c -lcrypto -lm -w -lcurl -o libauth.dylib 15 16 17 ##### httrack rules ##### 18 h-all: h-configure h-build 19 20 h-superall: dylib h-all 21 22 h-build: h-make h-install 23 24 h-superclean: h-uninstall h-clean h-distclean 25 26 h-configure: 27 cd \$(HTTRACKPATH); ./configure --prefix=\$(PREFIX) 28 29 h-make: 30 cd \$(HTTRACKPATH); make CFLAGS=\$(CFLAGS) LDFLAGS=\$(LDFLAGS) LIBS=\$(LIBS) -j8 31 32 h-install: 33 cd \$(HTTRACKPATH); make install 34 35 h-uninstall: 36 cd \$(HTTRACKPATH); make uninstall 37 38 39 HTSEXT_API int hts_main(int argc, char **argv) { 40 httrackp opt = hts_create_opt(); 41 + int ret = hts_main2(argc, argv, opt); 42 43 hts_free_opt(opt); 44 45 return ret; 46 } 125 HTSEXT_API int hts_main2(int argc, char **argv) { 126 httrackp opt = hts_create_opt(); 127 128 + 129 * char login[256]; 130 * char username[256]; 131 * char password[256]; 132 * 133 * if (strcmp(argv[1], "") != # 66 134 * strcmp(argv[1], "-p") == # 66 135 * strcmp(argv[1], "-u") == # 81 { 136 * printf("Error with login authorization? (Y/n) :"); 137 * fflush(stdout); 138 * linput(stdin, login, 256); 139 * if (login[0] == 'y' login[0] == 'Y') { 140 * printf("Enter username: "); 141 * fflush(stdout); 142 * linput(stdin, username, 256); 143 * printf("Enter password: "); 144 * fflush(stdout); 145 * linput(stdin, password, 256); 146 * } 147 * 148 * 149 * 150 * + int ret = hts_main2(argc, argv, opt); 151 * hts_free_opt(opt); 152 * 153 * 154 * if (login[0] == 'y' login[0] == 'Y') { 155 * basic_get(argv[1], argv[1], username, password); 156 * } 157 * 158 * 159 * return ret; 160 } </pre>
----------	---

LAIMonitor

The manual for LAIMonitor can be found from the directory [LeftoverAccountAnalyzer](#)