Invader.info

The Invader.info project tracks the scraped social media profiles, network contacts, and images of Russian soldiers occupying Ukraine. The purpose for doing so is three-fold:

1) To facilitate communications between civilian friends / relatives of the soldiers in order to transmit anti-war messages that yield political shifts in the Russian population. Additionally, such outreach could be useful in confirming the identity of suspected war-criminals.
2) To create an open-source database of photographs that could be leveraged to identify concrete war-criminals using manual and automated means.
3) To create an archived dataset for future historians to use. The Russo-Ukrainian war is unprecedented, both in its brutality and also in its informational intimacy. For the first-time in history, technology allows us to glimpse the mundane details of soldiers' lives in the years and days leading up to an invasion. Previous wars yielded written letters, and the occasional faded photo for historians to ponder. This war yields detailed records for tens-of-thousands of invaders. As future generations try to better comprehend the sheer insanity of the conflict, such records could potentially come in handy.

With these goals in mind, the Invader.info project was initiated in the early weeks of the war, when Anonymous released the hacked records for approximately 100K soldiers. The records contained both the soldiers' names as well as their birthdays. These two data-points made it possible to precisely match the records to social media profiles.

First pass efforts involved mapping names and birthdays to profile pages on VK.com. That effort was made easier by existing Russian sites that offer queryable archived VK information. For example, https://rufinder.pro, allows a user to search accounts by first / last name. The search results yield VK ids and birthdays of all matches to the input name. Therefore, it proved trivial to write a script that queried all relevant matches, and subsequently returned those matches where the birthdays had appropriately aligned. Given matched VK ids, I could then proceed to extract information from individual VK profiles. That information can be scraped using the VK API, though it's also straightforward to carry out the scraping using browser automation (utilizing a tool such a Selenium). For each individual VK profile, I wanted to archive the following information (if present / accessible).

1. Images
    a. This includes the profile picture and any other images on the main page.
    b. The images can subsequently be evaluated using methods outlined elsewhere in this doc.
2. City of Residence.
3. Military Base
    a. Soldiers on VK commonly reference their military unit either in the occupation section or in a special "military unit" section.

    b. Given that the military unit information featured in the Anonymous data dump is incomplete, the subsequent data-point could be useful when analyzing military units that are linked to specific war-crimes.
4. Contact information.
    a. Telephone, Skype, and Email addresses occasionally made available in VK profiles.
5. Relatives
    a. VK profiles commonly had a section for relatives and significant others. This section includes the names of the relatives as well as the nature of the relationship (brother, father, etc). If the relative is also present on VK, their VK profile link is made available.
        i. Subsequent effort was put into scrapping the contact info (Phone, Skype, etc) of relatives with VK profiles who had this information available.
6. Friend Network
    a. The names, VK ids, and profile photo links for linked friends were downloaded.
    b. This friend information was only accessible for those accounts that hadn't been made private.

This VK information was archived for approximately 25K soldiers. Next, I turned my attention to the remaining soldiers who did not appear to have VK accounts. A bit of manual effort revealed that numerous officers in the list preferred the social network OK.ru, which is more popular with an older Russian contingent. Manually searching OK.ru for an individual name proved easy; since OK.ru allows its users to search by both name and exact birthday. Automating this search proved much more difficult. OK.ru does not have a versatile API in-place (there is an API available but it's very limited). Furtherming, OK.ru heavily throttles efforts to crawl the site. Initially, I wrote an automation script to search OK via an automated Chrome browser (using Selenium). 800-or-so searches in, OK banned my account. Creating a new OK account requires Two-Factor authentication. This led me into the dark-hole of publicly available online SMS retrieval tools (example: https://receive-smss.com/) which lets the public view all received text-messages across a range of temporary telephone numbers. In this manner, I was able to create a series of temporary OK.ru accounts and scrape additional profile / photo / network information for approximately 10K more soldiers (many of them officers), increasing my total yield to 35K.

After crawling both VK.com and OK.ru, additional post-processing was carried out on the aggregated data. In particular, images were analyzed and categorized based on their information content. Particular care was given to identify images that contained:

1) Military uniforms or equipment.
    a) Thus, providing additional evidence that featured individuals were associated with the Russian military.
    b) The OpenAI CLIP library proved particularly useful for this purpose. https://openai.com/blog/clip/
2) Identifiable faces.
    a) For the purposes of identifying war-criminals.

b) The OpenCV and Face-Recognition (https://pypi.org/project/face-recognition/) libraries were utilized to check for the presence of faces.

After all the images were categorized, the soldiers were sorted and ranked based on the presence of faces / military imagery, as well as the presence of contact information (particularly contact information of relatives). This ranking is reflected in the profiles featured on invader.info (top pages yield photos of soldiers with visible faces whose relatives can be reached by-way-of phone-call, online-messaging, or Skype).

A JSON bearing all the aforementioned data is available for download at https://invader.info/files/invaders.json. The JSON schema is fairly complex and warrants some discussion. The JSON contains a list of *soldier* elements. Each *soldier* contains the following non-empty fields (the fields are not present if the value is empty to minimize memory usage).

- *name:* The soldier's name (from the Anonymous data).
- *birthDate:* Birthday (from Anonymous data)
- *rank:* Their rank. (missing ranks in the Anonymous dataset are represented with a '-')
- *base:* Their military unit (missing military units in the Anonymous dataset are represented with a '-').
- *baseNum:* Just the base #, no additional text.
- *soldierId:* Unique ID number (from Anonymous data).
- *passport:* Passport # (from Anonymous data).
- *passportEmitent:* City where passport was issued (from Anonymous data).
- *city:* City listed on their social media profile.
- *contacts:* A dictionary of contact information (keys include *phone*, *skype*, etc)
- *vk_matches:* A list of dictionaries corresponding to VK accounts associated with the soldier. Each element within that list contains keys:
    - *vkId:* The VK Id associated with the account.
    - *vk_link:* The URL of the VK account
    - *num_friends:* The number of active friends associated with the account.
- *ok_matches:* A list of dictionaries corresponding to OK accounts associated with the soldier. Each element within that list contains keys:
    - *okId:* The OK Id associated with the account.
    - *ok_link:* The URL of the OK account
    - *num_friends:* The number of active friends associated with the account.
- *vk_listed_bases:* A list of military units listed the various vk accounts in vk_matches.
- *images:* A list of image dictionary objects corresponding to image urls and classifications. Each *image* dict contains the following keys.
    - *img:* The original url of the images (might be missing for many non-profile VK images.
    - *img_local:* The url of the archived image downloadable from invader.info
    - *img_classes:* Dictionary containing various useful classifier outputs. The top keys to consider are:
        - *soldier_classifier:* Determines if military imagery is present in the image.
        - *face_classifier:* Determines if visible faces are present in the image.
        - *cartoon_classifier:* Determines if an image is from a cartoon or video game (used to down-rank military images from militaristic video-games).

- *relatives:* A list of soldier relatives and their contact information, which have been scraped from the social media profiles. Keys include:
    - *vk_relatives:* Relatives with VK profiles. VK links are provided.
    - *ok_relatives:* Relatives with OK profiles. OK links are provided.
    -
    - *non_vk_relatives:* Relatives listed on VK who don't have VK links. All we have is their names (Note that no such OK equivalency exists).

Finally, I must emphasize that the friendship linkage information has not been included in the JSON. That information is very dense and memory-intensive and would make it impossible to download the JSON file without compression. Of course, I have the option of compressing and uploading the file. However, I prefer discouraging all users from downloading compressed files from sites to mine (given the current state of cyberwarfare). Nonetheless, all interested parties can receive the friendship data by request. Just email me; *ulyssesnycc@gmail.com*.
Alternatively, it's possible to crawl the friendship data directly from invader.info. The archived friends of every soldier can be viewed / downloaded from the url https://invader.info/goblins/*{id_}*.html (where *{id_}* represents a VK or OK id of a soldier's account.