Below you will find a list of potential projects. Below that you will find a list of data sets and measurement platforms you may find useful for completing your projects.

Option 1: Censorship Measurement Project

For this project you will work to implement a measurement experiment of Internet censorship using our Centinel platform (https://github.com/iclab/centinel/tree/master/centinel). In the first part you will develop a Web interface that will allow you to distribute your experiment to the Centinel clients. In the second, you will do a measurement study using the platform. Finally, you will implement a Web interface to help you view the results of the second part and complete your analysis.

Your project should fork the existing Centinel code (https://github.com/iclab/centinel/tree/master/centinel) and interface with the existing server (coordinate with Abbas Razaghpanah as needed for server features/enhancements). You should open issues in the centinel repository if you find bugs with the software. There will be bonus marks for groups that find and fix the most bugs and have the most merged pull requests to the platform. The best project may even be merged into the platform itself! You have a chance to make real impact if you do a good job here.

1. **Make a Web UI for farming out experiments to Centinel clients.**
    - **Description**. Create a Web UI that can take log in credentials and uses HTTPS to communicate with the Centinel server. The UI should have the ability to show a map of clients indicating which platform they are on (VPN, mobile, user install, custom hardware) using data from the server's database.
    - The UI should be able to read through the list of measurement primitives (primitives are things like traceroute, ping, HTTP etc.) available in the platform (by interfacing with the server) and display the set of primitives on the screen. The user should be able to select which of the primitives they want to run (e.g., using radial buttons). There should be an 'advanced' button that will display additional parameters the user can specify (work with the ICLab team to make sure primitives have documentation associated with them. Work this documentation into the Web UI, e.g., mouse over a primitive to see a small description of what it does).
    - The user of the UI should be able to specify input (e.g., a list of URLs) in a text box in the interface, or upload a list from their computer. The UI should include the ability to select from preloaded lists (e.g., CitizenLab's lists https://github.com/citizenlab/test-lists).
    - The user should also be able to select the set of clients they would like the experiment to run on (either using the UI or using groups like "VPN clients").
    - The user should be able to specify the start and end times of the experiment and the frequency with which it should be run.
2. **Run an experiment on ICLab.** You should be creative for this part of the project. Below are just some ideas. Use existing measurement primitives or write your own

(e.g., HTTP Proxy localization: http://www.icir.org/christian/publications/2014-pam-proxies.pdf) to study an aspect of censorship. You may want to look at collateral censorship (e.g., censorship that happens because common sites are hosted on a single infrastructure), censorship leakage (accidental censorship of neighboring countries), fingerprinting which product is used for censorship (e.g., extend/scale up techniques from: http://conferences.sigcomm.org/imc/2013/papers/imc112s-dalekA.pdf and http://www3.cs.stonybrook.edu/~phillipa/papers/JLFG14.pdf), looking at how and where censorship is implemented, content modifications (vs. full blocking) etc.

3. **Make a Web UI for seeing the results of Centinel experiments.**
    - **Description.** Create a Web UI that (for a logged in user, see above projects) will show the user a list of experiments that they have run (don't allow users to see everyone else's results) and a table of the outputs.
    - Include a feature to compare two (or more) results of the same type. In this feature, allow the user to select two measurements of the same type (e.g., HTTP, traceroute etc.) and show the results side by side with the fields of the measurement type given as rows in the table.
    - Depending on what experiment you did for part 2 you may want to implement an interface/visualization (e.g., real time plots) that makes your task easier/your results easier to interpret and view.
    - See slides 11 and 12 of this slide deck for an idea of what the interface for browsing and comparing results might look like: https://drive.google.com/viewerng/viewer?url=http://www.cs.stonybrook.edu/~phillipa/icl_slides.pdf

Option 2: Open ended Projects

These are just a few selections. They are generally meant to be done by 2-3 students. You may choose to work alone or join in on active research projects in the networking research group to get hands on experience of research in the field. You are encouraged to discuss other project ideas with the professor.

1. **Measuring Traffic Interception on the Internet** As we will discuss in class, hijacking and even intercepting network traffic is surprisingly easy! Worse yet, interceptions often go unnoticed as they do not impact end-to-end connectivity :(. In this project you will work with a PhD student in the networking research group on our project to design a real-time detection system for traffic interceptions on the Internet. This is a collaborative project involving big data and real time analysis.
2. **Collateral censorship.** Investigate cases of collateral damage of censorship. (There are a few different ways to approach this.) Concrete direction: Implement a methodology similar to: http://conferences.sigcomm.org/imc/2013/papers/imc253-calderA.pdf and investigate the hosting infrastructure of different Google services to look for overlap. In particular in Pakistan this can lead to collateral censorship when other Google services use the same infrastructure as YouTube which is blocked. It may also be possible to use this

method for other large content providers: http://conferences.sigcomm.org/imc/2013/papers/imc163s-streibeltA.pdf .

3. **Mobile middleboxes.** Survey related work on middleboxes in mobile networks. (e.g., http://conferences.sigcomm.org/sigcomm/2011/papers/sigcomm/p374.pdf). This may involve reading online forums, investigating middle box products sold to mobile networks (be creative). Examine existing mobile measurement platforms (specifically Netalyzr Mobile) and either extend them or build your own mobile app to investigate middleboxes in mobile networks (e.g., using techniques suggested in http://www.icir.org/christian/publications/2014-pam-proxies.pdf). Specifically look at middlebox configurations, does the server see the client or middlebox IP? Does the middlebox transcode images? Alter HTTP headers? See http://www3.cs.stonybrook.edu/~phillipa/papers/mvno.pdf for an example where a middlebox seems to block access to YouTube videos on a mobile ISP.
4. **IPID Header Survey** The IPID packet header is a side channel that can reveal how many packets a host has sent (for hosts with globally incrementing counters). This project is to ping a variety of servers (e.g., SBU web server, google, facebook) and study the IPID header to learn about load on these servers. You will need to address challenges such as IPID header wrap (the field is a limited number of bits) + disambiguating multiple servers behind a single IP address who may have distinct IPID streams. See these two related papers:

https://www.usenix.org/system/files/conference/foci14/foci14-anonymous.pdf
https://www.usenix.org/system/files/conference/foci14/foci14-knockel.pdf

5. **Web-based network measurement** Pick 1-2 Web technologies (eg., Java script, Flash, HTML5), implement a suite of 5 commonly used network measurements (eg., traceroute, DNS query, etc.) in the technology of your choosing. Create a Web page that will allow people visiting the Web page to run network measurements from their machine without installing any software.
6. **Chilling effects** Analyze data from the chilling effects Web site: https://www.chillingeffects.org/ look for patterns in DMCA and other reports. Who is most active in submitting reports. Are there trends/decreases in reports issued. Are there trends that seem to indicate abuse of the reporting mechanism?
7. **Tor Entry Relay Selection** extend the Tor client to select entry relays based on specified criteria to reduce the chance of eavesdropping and timing attacks on the user's connections. You may need to implement network measurements to help inform the entry-relay selection. Once completed, compare results of an unmodified Tor client with 1-2 entry relay selection schemes and compare (1) performance of the connection and (2) likelihood of traffic analysis attacks.
8. **Link NLP analysis of operator mailing lists with real-time network measurements + diagnostics.** Read: http://www3.cs.stonybrook.edu/~phillipa/papers/PAM2015.pdf use NLP techniques to extract relevant information from the emails to spur real time measurements of the incident in the mailing list (e.g., traceroutes using RIPE Atlas). Potential to collaborate with USC to actually take action using LIFEGUARD system if initial results are good. See: http://lifeguard.cs.washington.edu/

9. **Use RIPE ATLAS to measure net neutrality.** Read: http://dl.acm.org/citation.cfm?id=1658972 . Implement the causal inference technique using measurements from RIPE ATLAS (e.g., latency measurements). You may also reproduce the methods from: http://www.caida.org/publications/papers/2014/challenges_inferring_interdomain_congestion/challenges_inferring_interdomain_congestion.pdf using RIPE Atlas as part of this project. Your project code should be designed to run on an ongoing basis to monitor net neutrality over time.
10. **Explore Internet connectivity in developing regions** We have large traceroute data sets including regions with developing Internet connectivity. In this project you'd look at AS-level connectivity and routing behavior of ISPs in these regions. Longitudinal data may be available to look at changes over time as well. Reference: http://www.cc.gatech.edu/~agupta80/pdfs/pam14.pdf

Data sets, platforms and sources of information for projects:

- RIPE Atlas https://atlas.ripe.net/ , a global network of network probes that you can use for measuring: network paths (traceroute), ping, DNS, HTTP. Talk to Phillipa to get set up on the platform if you would like to use it.
- Routeviews http://www.routeviews.org/ , logs of BGP messages from around the world. (Similar data and nice tools for working with these messages can be found here: http://bgpmon.netsec.colostate.edu/)
- iPlane (http://iplane.cs.washington.edu/) logs of traceroute measurements between vantage points.
- "Cyclops" UCLA Internet Topology (http://irl.cs.ucla.edu/topology/) Internet topology inferred from BGP messages and models of AS business relationships. More recent + validated topology from CAIDA http://data.caida.org/datasets/2013-asrank-data-supplement//data/ (as-rel files)
- CAIDA data (http://www.caida.org/data/) Lots of different types of data here.
- NANOG mailing list (http://mailman.nanog.org/pipermail/nanog/)
- Outages mailing list (http://puck.nether.net/mailman/listinfo/outages)
- Gao Rexford routing policies (see Appendix A: http://www.cs.stonybrook.edu/~phillipa/papers/SBGPtrans.pdf) a standard model of how ASes will route traffic on the Internet.
- MeasurementLab data: http://www.measurementlab.net/data A variety of data sets. E.g., NDT: http://www.measurementlab.net/tools/ndt
- InternetCensus data: http://internetcensus2012.bitbucket.org/ (description and a variety of data from a global scan of the Internet).
- Chilling Effects (see https://github.com/berkmancenter/chillingeffects/blob/master/doc/api_documentation.mkd for API)