CS 1653: Applied Cryptography and Network Security
Fall 2017

Term Project, Phase 1

Assigned: Thursday, August 31   **Due:** Tuesday, September 12 11:59 PM

---

1   Background

Over the course of this semester, we will experiment with and apply the security concepts that are covered in lecture by developing a group-based file sharing application that is secure against a number of different types of security threats. At a high level, our system will consist of three main components: a single group server, a collection of file servers, and some number of clients.
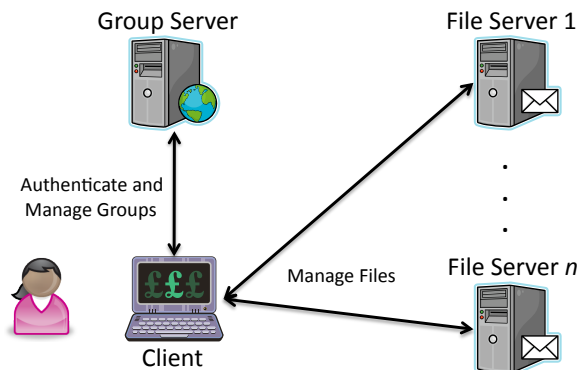
The *group server* manages the users in the system and keeps track of the groups to which each user belongs. Any number of *file servers* can be deployed throughout the network, and will rely on the group server to provide each legitimate user with an authentication and authorization token that answers the question *"Who are you, and what are you permitted to do?"* Users within the system make use of a networked *client application* to log in to the system and manage their groups (via the group server), as well as upload, download, modify, and delete files stored in the system (via the file servers).

Your project submissions will be made using the git distributed version control system. You are encouraged to use this system to help coordinate your group work, and follow the git best practice: "commit early and often."

2   What do I need to do?

Phase 1 of the course project is designed to get you thinking about the project at a high level. The above description of the system is deliberately underspecified so that you have

the intellectual freedom to consider many different possibilities for how such a system should work. In this phase of the project, you will do just that.

Task 1: Group Formation

Successfully completing the later phases of this project will require that a considerable amount of time and energy be spent exploring and analyzing system design issues, designing security solutions, and implementing/testing your system. To minimize the burden that this places on each of you and to develop your collaborative development skills, all phases of the project—including this one—will be carried out in groups of 3 students. Under no circumstance will smaller or larger groups be permitted.

For your first task, you must find 2 other students with whom to work for the remainder of the semester. Put some thought into your group choice, as your grades will depend on one another's commitment to your joint work. That being said, all group members will have the opportunity to rate one another's performance on each stage of the project.

Task 2: Security Requirements

After forming your project group, work with your fellow group members to brainstorm a list of *security requirements* that you feel should be respected by a group-based file sharing application like that described above. You may assume that the system will support the following types of operations:

- Create/delete user

- Create/delete group

- Add/remove user u to/from group g

- Upload/overwrite file f to be shared with members of group g

- Download file f

- Delete file f

Given this extremely (and purposely) high-level description of the system's functionality, your group should develop a list of properties that a *secure* group-based file sharing application must respect. For each property, come up with (i) a name for the property, (ii) a definition of what this property entails, (iii) a short description of why this property is important, and (iv) any assumptions upon which this property depends. As an example, consider the following:

> **Property 1: Correctness.** Correctness states that if file f is shared with members of group g, then only members of group g should be able to read, modify, delete, or see the existence of f. Without this requirement, any user could access any file, which is contrary to the notion of group-based file sharing.

The goal of this exercise is to get your group thinking about some the challenges involved with building secure distributed systems. We neither assume that you are file sharing experts, nor that you have prior experience developing secure applications. To begin with, spend some time thinking about what would make *you* trust the security of such a system, and use this intuition to formulate your initial requirements. This will provide you with a starting point that can be refined by examining the features afforded by other systems and reading over relevant portions of your textbook.

Task 3: Setting Up git and GitHub

You will submit your projects using the git version control system, using repositories hosted on GitHub. Although this phase of the project is a writeup and not code, you will use it as an opportunity to familiarize yourself with git. Thus, as part of this phase of the project, you may need to utilize online resources available for these tools.

Begin this task by familiarizing yourself with the concepts of version control in general, and git in particular. One great resource for this is the Pro Git book by Scott Chacon and Ben Straub (`https://git-scm.com/book/en/v2/`), especially chapters 1.1, 1.3, and 2.1–2.4. Note that you do not need to be an expert in git, but you do need to understand its use cases, including why it is an appropriate tool for this project. You can experiment with basic git commands online at `https://try.github.io`. Once you are comfortable with the concepts, you should install the git client on your local machine using `https://git-scm.com/`

Your git repositories will be stored on GitHub, an online host for git repositories. Each member of your group should sign up for GitHub, using your `pitt.edu` email addresses to ensure you receive a free unlimited academic account. If you already have a GitHub account, you can link your `pitt.edu` email address to it at `https://github.com/settings/emails`. Once you have signed up or linked your existing account to your `pitt.edu` email address, visit `https://education.github.com/` and click the "Request a Discount" button to request a student discount; this will give you the ability to have unlimited private repositories. Having a private repository for your coursework is crucial, so *do not skip this step*!

If you'd like an introduction to the GitHub service, visit the GitHub tutorial here:

$$https://guides.github.com/activities/hello-world/$$

3   What should I turn in?

In this phase of the project, you have two primary deliverables: a git repository that will be used for the duration of semester-long project, and a written report documenting you group's activities for this phase of the project.

3.1   Repository Setup

You will now create the (single) git repository that will be used to facilitate the management of your group's term project. To do this, *one* group member should carry out the following steps:

1. Log into GitHub

2. Click the "+" icon at the top right corner of the screen, and then click "New repository"

3. Name your repository using the Pitt usernames of each group member, sorted alphabetically, in the following format: `cs1653-2017fa-abc1-def2-ghi3`

4. Please enter each group member's name in the optional "Description" field

5. **Select the "Private" option for this repository.** This ensures that only the members of your group can access its contents. Failure to do so may be interpreted as cheating, as this will make your repository viewable by *anyone*, including other students in the class!

6. Click "Create repository"

7. Once you have your repository, click on "Settings" and then "Collaborators." Add each group member as a collaborator on this repository, as well as the professor (GitHub ID: `ajl-pitt`) and the TA (GitHub ID: `potato84`).

At this point, everyone in your group should be able to clone a local copy of this repository, as well as push changes to it.

3.2   Written Report

Once you have your own repository, each group member should *clone* it to their own machine to facilitate their contributions to the project. The report produced by your group should be formatted as a Markdown document, stored in the path `reports/phase-1.md`. Markdown documents are nicely rendered by the GitHub web interface, and allow for easy markup for text-based documents. A tutorial on Markdown can be found at `https://guides.github.com/features/mastering-markdown/`. The report that your group will produce should have the following structure:

- **Group information.** List the full name, Pitt email address, and GitHub ID of each group member.

- **Section 1: Security Properties.** This section should describe the requirements that your group has identified as being relevant to the group-based file sharing scenario. You should aim to find at least 15–20 such requirements, that together will cover at least two different sets of reasonable system assumptions (i.e., threat models). This section should be arranged as a bulleted list of properties that may apply to a file sharing system.

- **Section 2: Threat Models.** This section should describe several sets of trust assumptions that could be made regarding the players in the system. Describe several scenarios in which you expect the file sharing system to be used, and describe the ways in which the various entities with access to the system will be trusted to behave. This section should be arranged as follows:

1. A paragraph describing a system model: an environment in which you envision your application being deployed.

2. A paragraph describing the trust assumptions that you would make regarding the players in the system, within this particular system model.

3. A bulleted list of relevant security properties from Section 1, each with a sentence or two discussing how it applies to this system / threat model. Note that not all of the security properties you define will necessarily be relevant to all of your threat models.

4. Repeat items 1–3 as needed for additional system / threat models.

This phase of the project will be graded for content, rather than presentation (hence the bulleted list format); the goal of this exercise is to get you thinking about functionality and security.

- **Section 3: References.** If any of the requirements in Section 1, or any of the system models in Section 2, were inspired by material from books, papers, articles, or existing products, your sources should be cited here.

You are encouraged to use this version control repository regularly while collaborating on this phase of the project, as *each individual's contribution to the group's work will be judged in part by the version control logs.*

In addition, *each student in your group* should send an email to `adamlee@cs.pitt.edu` that indicates his or her assessment of each group member's contribution to this phase of the project (e.g., *Bob did 40% of the work, and Mary did 60% of the work*).

3.3   Submission

Your project is due at 11:59 PM on Tuesday, September 12. We will clone your repository immediate after the due date, so you will be graded on whatever changes have been committed and pushed to your repository by this time. Make sure your repository is shared correctly with the instructor and TA well in advance!

Important: Typing the command "`git log -n 1`" will output information about the last commit, including the commit hash. For example:

```
commit 07155f495c61e515c4ba34d42d7ccddc336def35
```

The group member responsible for making the final commit should email this commit hash to `adamlee@cs.pitt.edu` and `xil160@pitt.edu` prior to the deadline.