**CIS-481: Introduction to Information Security**

**InfoSec Chapter Exercise #7**

**Team: Project Team 11**

**Participants: Sohal Patel, Erika Maglasang, Nathan Moran, Gabriel Rolink, and Tyler Samuelson**

**Logistics**

1. Get together with other students on your assigned team in person and virtually.
2. Discuss and complete this assignment in a collaborative manner. Don’t just assign different problems to each teammate as that defeats the purpose of team-based learning.
3. Choose a scribe to prepare a final document to submit via Blackboard for grading, changing the file name provided to denote the number of your assigned **Team**.

**Problem 1**

Consider the logical access control needs for joint software development teams using a typical Linux environment. Roles must include Developers (that can commit changes made in the code), Testers, and Code Reviewers. The technical access control mechanisms that you design must reflect these organizational roles. Your access control solution must:

1. Protect the software being developed from outsiders stealing it
2. Protect against unauthorized changes (including from internal actors)
3. Ensure that we can trace *who* made each change

**Situation 1: A small team on a single machine *(5 points)***

**Answer:** It actually exempts an approach which exists in a very small or initial startup of an organization that all the tasks of an organization are conducted on a single machine and a bunch of people work together, without access control list the data security cannot be carried up. All the setups for this approach lead to a centralized approach in which all files or directories are just made readable, it is actually a buildup approach and is not effective in any way. It led to an ACL setup in which all the developers within it could be added solitarily. It can have multiple groups of information.

**Situation 2: A medium-to-large team on a LAN [Hint: Use of a version control system like** [**Subversion**](https://subversion.apache.org/) **is highly recommended] *(10 points)***

**Answer:** It vary in the group of people in the organization medium to large team organization It can neither lead to longer on any solitary machine such that the access controls promulgated on a group of machines either in a topology or not but in all cases it should be with a group of machines, It could also use straight-forward document authorizations such that whenever the sensitive organizational information is accessed they have to get them authorized in all possible ways. More requirement for change-following More formal hierarchical structure of the organization. In order to use a version control system, in a VCS all the changes are well-recorded by the group of developers it permits web-designers to overlook all the tasks altogether. It led to trying not to over-write each other's changes. Keep up history of each form of everything, Subversion control system is known to be as an open source way of access controls that Subversion actually brought together server that makes a solitary well spring of truth. Simple to utilize apparatuses. Effective and incredible expanding. Simple to submit changes, even over numerous archives. Programmed goals of consolidation clashes

**Situation 3: A large, distributed team, including outsourced contractors *(10 points)***

[Inspired by <https://www.cs.columbia.edu/~smb/classes/f09/l08.pdf> - many thanks to Columbia University for providing under Creative Commons!]

The large organization is comprised of actually a variation into departments. It overlooks customer/server model for getting the store or accessing, in an organization no one could access the version control system’s actualizing repository in a large-scale organization. It could must limit what orders can be executed on vault by web-developers. It should must incorporate with V.C.S server or layer on fundamental operating system consents which results in limiting it what orders can be executed on archive by web-developers. Whatsoever access controls are used, there is an advantage that there is authentication and authorization for everyone who try to access the repository of large organizations in any way.