# pmNotes/ Jayal Shah/ TOC of KeePassXCIPFS Capstone Project

## Table of Contents

## 1 Action Items

1. Get an account here. ASAP. https://www.synopsys.com/software-integrity/security-testing/static-analysis-sast/academic-program.html
2. GitHub reorganization?!
3. Saw only one diagram.

## 2 TOC for the Capstone Project Report

1. I am giving you this TOC. Follow this structure. And, the approx. page count. pp pages. TBD To-be-Done
2. Title: "KeePassXCIPFS: KeePassXC Transformed into a Distributed App on IPFS and NuCypher: A CS 7960 Capstone Project Report"
3. Do NOT write tutorials for newbees. These chapters should be aimed at experienced developers. Certainly include summaries of **your** experiences.
4. Some pages can be reproduced with edits from the Independent Study report.

## 3 Top Matter

1. Title Page of the Report
2. Abstract
3. TOC

## 4 Chapters and Sections

1. Chapter: Introduction (?? pp)
    1. Describe KeePassXC. Old Chapter on KeePassXC
        1. KeePassXC is a community fork of keepassX
        2. Include an extended example of a password collection.
    2. Describe this capstone project. Distributed. Vaults replicated. Mult master keys.
        1. Describe the end-result product. How big? SLOC? Modules? Files? Call diagram?
        2. Describe contributions made by you.
    3. Describe Python3 modules required.
    4. Describe C++ required.
2. Chapter: Background on IPFS and NuCypher (6 pp)
    1. IPFS overview

2. NuCypher overview. Proxy Re-encryption. Include examples.
3. Studying a few dApps
4. Typical structure of a dApp based on NuCypher
5. Critique of Snowden dApp
6. Critique of NuBox dApp
7. Critique of TBD#1
8. Critique of TBD#2

3. Chapter: KeePassXCIPFS (1x pp)
   1. Decentralized and replicated Vaults on IPFS + NuCypher
   2. Shared passwords across trusted users
   3. Multiple "Master Keys"
   4. Overall Goals:
      1. Functional behavior of KeePassXCIPFS limited to what was in KeePassXC must be identical to KeePassXC
      2. Speed of transaction as close to KeePassXC as possible.
      3. Reliability?
      4. Feature set of KeePassXCIPFS larger than KeePassXC
   5. Include an extended example. Distributed. Replicated.

4. Chapter: Design of KeePassXCIPFS (10 pp)
   1. KeePassXCIPFS: config, internals; sloccount, files and modules
   2. KeePassXCIPFS: diagrams
   3. Includes architectural diagrams
   4. Way-above code level descriptions

5. Chapter: Implementation of KeePassXCIPFS (15 pp)
   1. A 3-machine version of KeePassXCIPFS
   2. Code level descriptions – not code itself.
   3. Assertions – pre- and post-conditions; follow doxygen expectations
   4. KeePassXCIPFS doxygen-ed

6. Chapter: Evaluation of KeePassXCIPFS (10 pp)
   1. SLOC of KeePassXCIPFS
   2. Report on Testing KeePassXCIPFS
   3. Security Evaluation: Static Analysis of Source Code
   4. Security Evaluation: Dynamic Analysis of Binary (skip??)
   5. Performance: Speed and Memory; comparison with KeePassXC (non dApp).
   6. A perspective on the issues you faced

7. Chapter: Related Work (10 pp) TBD
   1. Feature Comparison, SLOC, … with LastPass, Bitwarden, et al.

8. Chapter: Conclusion (2 pp)
   1. Describe contributions of this capstone project.

9. Bibliography/ References (5+ pp)
   1. Properly cited and referenced.
      1. Example 1: Derk Barten. 2019. Client-side Attacks on the LastPass Browser Extension. (2019). https: //uvalight.net/~delaat/rp/2018-2019/p59/report.pdf. [Missing place of publication.]
      2. Example 2: Manuel Egele, David Brumley, Yanick Fratantonio, and Christopher Kruegel. 2013. An Empirical Study of Cryptographic Misuse in Android Applications. In Proceedings of the 2013 ACM SIGSAC conference on Computer \& Communications Security. ACM, 73–84.
      3. Example 3: Ivan Albert Zudic and Neil Patrick Adams. 2018. Method and System for Master Password Recovery in a Credential Vault. (Aug. 30 2018). US Patent App. 15/445,308.
   2. Cite your Independent Study.
   3. I suggest you use LaTeX and BibTeX.

10. Appendices
    1. One appendix on each Experience. E.g., three on NuCypher Hackathon dApps.

2. KeePassXCIPFS doxygen-ed; uploaded to GitHub; add **your** additional comments
3. Testing of KeePassXCIPFS: Inputs and Outputs

# 5 End

---