Preparing a Project Proposal

CS-576 Systems Security

Instructor: Georgios Portokalidis

Spring 2018

Structure

Problem statement

Importance of the problem

Previous approaches attacking the problem

What do I propose to do in this project

How will I evaluate the success (or not) of the project

Problem Statement

Pick something that you are interested in

Be concise

- **Good example:** Buffer overflows due to incorrect usage of strcpy() and similar functions are a a problem because …
- **Bad example:** Software exploits are a problem because …

The better you define the problem, the easier will be for someone to understand what you propose to do

Subject must be related to the course

- See syllabus and intro slides for summary topics

You do not need to attack a yet unsolved problem

Importance of the Problem

Some problems may be more important than others

- **More important:** Software bugs in cyberphysical systems, like cars, can lead to loss of human life
- **Less important:** Software bugs in Call of Duty II can cause players to lost their equipment

It is a useful exercise, but keep it short

Try to reference articles that quantify the importance

- For example, the cost of bug overflows is estimated to $100B

Your project does not need to solve a very important problem

- But it needs to be well defined!

Previous Approaches

Describe how have others attacked the problem

Include references to those works

- Acceptable references are:
  - Technical articles on the Internet
  - Research papers
- Not acceptable references are:
  - Anecdotal information
  - High-level articles for a non-technical audience

Refer to the articles and papers given as reading material in the lectures

Do some research

- Search the web
- Through Google scholar (http://scholar.google.com/)

Proposal Description

Describe what you plan to do at a high level and how it solves the problem

- Are you reproducing a known solution or are you trying something new?

Define the tasks that your team will complete to implement the proposal

- **Good example:** We will develop a compiler plugin to introduce boundary checks before strcpy() and similar functions during compilation
- **Bad example:** We will introduce checks that prevent buffer overflows

For each task:

- Are there associated software deliverables?
- How much of effort will it require (people, time)?

Project Evaluation

Include a plan for evaluating what you developed

This may include:

- Test cases of programs, inputs, etc.
- Benchmarks for evaluating performance
- Measurements

Evaluation is equally, if not more important, than the development part in systems work

Negative results are OK, assuming that was a potential outcome

- **Good example:** We were able to stop 2 types of overflows, and failed to stop 2 other types
- **Bad example:** Our developed software does not compile or run properly

Closing…

No need for a lengthy proposal

- But still follow the steps

A good proposal will save you time in the end

- You will know what you need to do
- I will know what you are doing
- You can reuse text from the proposal in the final report

Initiate a discussion over Piazza or email

- Nailing down the exact problem is the most important step

I will provide some topics on next week's lab session

- Not trivial
- More of the yet unsolved problem variety