

## CS166 Project 4: GradeView

Due on April 30, 2009 at 5:00 pm EST

Project Out: April 10, 2009

### Silly Premise

You and a few other students spent months writing a solid online grades app that allowed students, TAs, and professors to easily and securely access and change grades. But this week, the university unveiled its own grades program which was outsourced for millions of dollars. Gathering your disgruntled team, you set out to bring the system down...

### The Project

You have to find security holes in our implementation of a course management app and then patch them. The app is presented to you as a blackbox, you are given a few usernames and passwords to figure out what you can do as a user.

### The Security Specification

1. Students must be able to view their grades and handbacks
2. TAs must be able to upload, view and change the grades of the particular projects they have graded for a particular student during the time that they are a TA for a course.
3. Professors must be able to view grades for all students during the time they are teaching a course.

### Dates

Several of these sections will be explained with more detail later in the document.

#### April 13, 5pm

One member of your group of 4 should email [cs166tas@cs.brown.edu](mailto:cs166tas@cs.brown.edu) with "[group]" as your message title and have the body contents list the names of the members of the group, and optionally the name of the group. (More details below)

#### April 20, 10pm

One member of your group should email [cs166tas@cs.brown.edu](mailto:cs166tas@cs.brown.edu) with "[exploits]" listing descriptions of vulnerabilities and exploits to take advantage of them, as well as a general description as how to fix them. (More details below)

**April 21, 10am**

Tiers (vulnerabilities we've found in the program with varying degrees of difficulty) are announced. (More details below)

**April 30, 10pm**

Handin. (More details below)

**April 30 - May 5th**

Interactive Grading. (More details below)

**Groups**

Your group must consist of 4 people. It is in your best interest to find a group and get this in as soon as possible. Once we have your group name, then we can assign you a URL for your group's copy of the application. Keep this secret, it wouldn't be in your best interest to let other groups know your URL, it will let them mess with your progress. To reset your application merely visit the page at <http://cs166/GROUPID/reset.jsp> and press the button. This is in case you mess up the webapp for some reason or other. It happens. You'll need it.

**First Handin: Exploits**

We want 3 exploits<sup>1</sup> as part of the first handin. Some exploits are better than others. We suggest staying in touch with the TAs during this process in case we may not agree about the validity of your exploits or that you are not hitting this number. If we deem the exploit to be 'implementable' then you must also handin the code to take advantage of the exploit.

Email us with [exploit] in the title and specify the vulnerability, the exploit, and suggested manner of fixing the vulnerability. This is the first time we have put out this project and we don't know how easy these exploits are to catch, so we've reduced the number to 3 from 6 on last year's final project. For a great source of possible web application attacks, you could take a look at the lectures!

Remember not to share your group's list of exploits with any other group. There will be a contest at the end on VMs, *no holds barred* (see below).

**Final Handin: Exploits and Fixes**

At this stage you will be given full access to the code. Your task is to fix the security holes you found in the first handin.

Any solution will be heavily scrutinized. For example, securing the application by simply removing the vulnerable feature is entirely unacceptable.

**Handing in**

You should hand in the implementation, script, or thorough description of your exploits (as applicable, since not all of them are implementable) as well as your modified secure server.

---

<sup>1</sup>NOTE: When we say 'exploits' here we mean something more general than for SecurePlayer. Any violation of security; illegal access, authentication spoofing, or tampering; is considered an exploit.

## Interactive Grading

We will be doing interactive grading for this project. Have two copies of GradeView available: unmodified and modified. For whatever fixes you choose to implement, you will also be responsible for demonstrating the exploit. Every partner should be prepared to present any exploit and any fix. Everyone is equally responsible for your exploits, if we sense this is not the case, we may grade accordingly. Thus, we suggest partners consistently meet in and work in groups. If you want to set up an svn repository and need help with it, contact the TAs or Google 'how to set up svn'.

## Contest

This is a little experiment we would like to have once everyone has handed in the project. Since you will not be sharing the list of exploits you discovered with anyone else, you don't know if you have patched all the security holes. Heck, someone might have found one we didn't anticipate! More details forthcoming.

## Tools and Rules

Ask the TAs before moving forward with any external tools. Authoring tools and analysis tools, such as Eclipse or Wireshark, are of course suggested (note that you'll probably have to use Wireshark inside a VM). As this is a group project, keep your communications between your group and the TA staff, there might be extra credit for the winners of the contest!