UNIVERSITY OF MARY WASHINGTON -- NEW COURSE PROPOSAL

Electronically submit this completed form with attachments in one file to the Chair of the College Curriculum Committee.

<table>
<thead>
<tr>
<th>COLLEGE (check one):</th>
<th>Arts and Sciences</th>
<th>x</th>
<th>Business</th>
<th>Education</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposal Submitted By: Andrew Marshall</td>
<td>Date Prepared: 10/13/15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Course Title: Software Security</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Department/discipline and course number*: CPSC 445</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*This course number must be approved by the Office of the Registrar before the proposal is submitted.

Number of credits proposed: 3 Prerequisites: CPSC 345 or MIST 411
Will this be a new, repeatable “special topics” course? (Do you want students to be able to take this new course more than once if the topic changes?) NO X YES

Date of first offering of this new course: FALL SEMESTER, year Fall 2016
Proposed frequency of offering of the course: Once every other year
List the faculty who will likely teach the course: Andrew Marshall
Are ANY new resources required? NO X YES Document in attached impact statement

This new course will be (check all that apply):
Required in the major
Elective in the major x
General Elective
General Education**

**AFTER the new course is approved, a separate proposal must be sent to the General Education Committee.

Catalog Description: CPSC 445 – Software Security (3)
Prerequisite: CPSC 345 or MIST 411.
A course on the intersection of software and information security. A programming intensive course. Topics include but are not limited to: Programming flaws, causes, identification, exploitation and prevention; malicious software, development, identification, and prevention; software fuzzing and other flaw identification and testing methods.

COURSE HISTORY
Was this course taught previously as a topics or experimental course? YES NO X
Course Number and Title of Previous Course Semester Offered Enrollment

CHECK HERE if the proposed course is to be equated with the earlier topics or experimental offerings. This means that students who took the earlier “topics” course will only be able to take the new course if they made a C- grade or lower in the earlier course.

NOTE: If the proposed course has not been previously offered as a topics or experimental course, explain in the attached rationale statement why the course should be adopted even though it has not been tried out.

REQUIRED ATTACHMENTS:
1. Rationale Statement (Why is this course needed? What purposes will it serve?)
2. Impact Statement (Provide details about the Library, space, budget, and technology impacts created by adding this new course. Include supporting statements from the Library, IT Department, etc. as needed.)
3. Sample Syllabus

Department Chair Approval: Date: 11/10/2015
CCC Chair Approval: Date: 11/18/15
UCC Chair Approval: Patricia Reynolds Date: 12/7/2015

Rationale Statement
(Why is this course needed? What purposes will it serve?)

New Course Proposal Cover Sheet (October 2015)
CPSC445 is proposed to be the follow-on course to the departments CPSC 345 Introductory computer security course. While the goal of CPSC 345 is to give the students an up to date and broad introduction to computer security, this new course would provide a more in-depth exploration of a specific and highly critical area of computer security, that of software security. In addition, the CAS and COB plan that the new CPSC 445 course will act as one of two capstone courses for the proposed new information security minor.

Software security is a critical component of the information and computer security landscape. Unfortunately it is also one of the more difficult areas of information security because it requires a high-level of technical knowledge. Since software flaws can lead to some of the most devastating security vulnerabilities for example the shellshock vulnerability was a result of a software vulnerability (Symantec Security Response 2014), it’s critical that we expose those interested and capable students to these issues.

Because of the lack of knowledgeable individuals, this area of information security is one of the more difficult areas for organizations to recruit qualified individuals. A recent report (Libicki 2014) conducted by the Forces and Resources Policy Center of the RAND National Security Research Division (NSRD) found that it has been particularly difficult to fill positions such as managerial/upper-tier, and highly technical software development/bug detection personnel. Part of the goal of this course will be to begin to prepare students to make contributions in these areas. While one course does not make an expert, the course will be at sufficient depth and incorporate sufficient hands-on exercises that student who succeed will be able to take entry-level positions related to software security. This includes the wealth of government and government contractor positions, software security and anti-virus company positions, and positions with organizations that have in-house security analysis teams. In this regard UMW is particularly well geographically positioned given the proximity to Washington, DC and the numerous government agencies and contractors engaged in cyber security in our area.

Ultimately, the new security course will provide in-depth exposure to this important area and help produce computer scientists better prepared for the challenging security environment that currently exists. It will also provide valuable experience to all participating students, as these skills will be very useful to any software developer in any organization regardless if they are working directly in security.

Finally, CPSC 445 will bring the UMW Computer Science departments course offerings up to an excellent level in an area, which Computer Science pedagogical bodies are now recognizing and pushing to be included in the undergraduate computer science education. For example, the Association for Computing Machinery (ACM), the leading professional organization for computer science professionals, researches and educators, has listed “information assurance and security” as a knowledge area (The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) IEEE Computer Society 2013). This is a new knowledge area in the ACM curriculum guidelines. The ACM advises that there should be at least 3 credit hours at Tier1 (topics should be a required part of every computer science curriculum) and 6 credit hours at Tier2 (topics generally essential in an undergraduate computer science degree), covering topics in the knowledge area. Combined with information assurance and best practices covered in other courses and CPSC 345 this new course would provide these recommended credit hours for UMW students.

**Impact Statement**
(Provide details about the Library, space, budget, and technology impacts created by adding this new course. Include supporting statements from the Library, IT Department, etc. as needed.)

The college currently has the equipment and facilities currently on hand needed to teach this course. The course does not require additional staff time from departments other than Computer Science. Computer Science currently has the faculty required to teach the course. The library has the necessary materials to support the course.
Software Security
Instructor: Dr. Andrew Marshall
E-Mail: marshall@umw.edu
Phone: 540-654-2245
Office: Trinkle Hall B16
Office Hours: [M,W] 2:00pm-5:00pm.
[T, Th] 10:00am -11:30am.

Prerequisite: CPSC 345 or MIST 411.

Course Description:
- Provide an in-depth introduction to computer security related to software.
- Cover a range of issues that detail how and why software vulnerabilities occur, how those vulnerabilities are exploited, what steps can be taken to reduce the introduction of such flaws into software, and what steps can be taken to reduce the impact.
- Learn how malicious software is developed and deployed and what steps can be taken to mitigate it.
- The course is programming intensive and hands-on.

Website: Canvas

Book(s):
- Other readings will be assigned from additional sources provided by the instructor during the semester.

Software: It is recommended (although not required) that the student have a personal computer running a Unix or Linux operating system. All of the software we will use will be freely available for the student. However, you will need a Unix/Linux OS to run the software. Please see the instructor if you need guidance on getting a Linux system up and running.

Programming Intensive: The course will assume a high level of “programming maturity” on the part of the students. We will use and move between several different programming languages in the course. Including, but not limited to, C, C++, Java, Python, Perl, and scripting languages such as SQL. The students are not required to know all these languages but should be comfortable with picking up new languages quickly.

Grading Policy:
Grading based on the following break down:
- Labs: 35%,
- Midterm: 15%,
- Final: 15%,
- Homework: 25%.
- Quizzes: 10%
Letter grade distribution:
- 100-92%: A,
- 89-91%: A-,
- 86-88%: B+,
- 82-85%: B,
- 79-81%: B-,
- 76-78%: C+,
- 71-75%: C,
- 69-70%: C-,
- 66-68%: D+,
- 60-65%: D
- < 60%: F.

The University provides the opportunity to provide grading feedback midway through the semester. This will take into account your midterm exam, homework, quizzes, and labs submitted up to that point. Any student receiving...
less than a 65% will receive an unsatisfactory (U) for their mid-semester grade. Students receiving a U should schedule a meeting with me to discuss how we may improve your performance in the class.

Student Conduct:
- Students are expected to attend class when possible and are responsible for the material covered in any class they may miss.
- Legitimate documentation is required for any exam that may be missed.
- If a student knows ahead of time that they cannot make an exam please schedule a meeting with me to discuss alternatives.
- The use of digital recording devices (such as video and audio recording devices, i.e., smartphone) in the classroom must be pre-approved by the instructor.

The Honor Code:
- Please review the Computer Science Department’s Policy regarding academic honesty
  - Students taking this course are presumed to be aware of the policy.

Special Guidelines for this Course:
- Students should not practice or use any of the methods discussed in the course on the university’s networks/systems or any other network/system where the student has not been given explicit permission from the owner and administrators of that network/system.

Disability Policy:
The Office of Disability Services has been designated by the University as the primary office to guide, counsel, and assist students with disabilities. If you already receive services through the Office of Disability Services and require accommodations for this class, make an appointment with me as soon as possible to discuss your approved accommodations. Please bring your accommodation letter with you to the appointment.
If you have not contacted the Office of Disability Services and need accommodations, their phone number is 540-654-1266. Their office is located in Lee Hall, Room 401.

Schedule of Subjects (subject to change):
- Overview of Software Security:
  - Week 1 – Week 2:
    - Overview.
    - Setting up an analysis environment.
- Common Flaws and Solutions:
  - Week 3 – Week 6
    - Common software security flaws.
    - How the flaws lead to security issues.
    - How software is exploited.
    - How to fix and prevent common flaws.
- Finding Software Flaws:
  - Week 7 – Week 10.
    - Introduction to hunting for zero days.
    - Fuzzing.
    - Testing exploits.
    - Mitigating future zero days.
- Introduction to Malware Analysis:
  - Week 11 – Week 15.
    - Introduction to analysis.
    - Capturing malware.
    - Malware Forensics.
    - De-Obfuscation
- Final Exam:
  - Week 16.

Schedule of Assignments
- Week 1 – Week 2 Assignments
  - Lab 1: Setup
  - Homework 1
  - Reading set 1
- Week 3 – Week 6 Assignments
  - Lab2: Security Issues
  - Lab3: Exploits
  - Homework 2
  - Reading set 2
- Week 7 – Week 10 Assignments
  - Lab4: Bug hunting
  - Lab5: Exploits 2
  - Homework 3
  - Reading set 3
  - Midterm
- Week 11 – Week 15 Assignments
  - Lab6: Malware 1
  - Lab7: Malware 2
  - Homework 4
  - Reading set 4
- Week 16
  - Final Exam
Works Cited

