# CS 405 Project Two Script

Kyle Lund

6-23-2024

Project 2: Security Policy Presentation

Video Link: <https://youtu.be/vEbigAkLIak>

| **Slide Number** | **Narrative** |
| --- | --- |
| **1** | Hello. Today I will be highlighting key points of the new Security Policy written for Green Pace. |
| **2** | This policy showcase aims to provide a high-level explanation of the security threats that the organization is now facing, as well as policies and standards that can help reduce common development concerns and security practices that guarantee the company's and our customers' digital safety. |
| **3** | There are at least 10 principles of secure coding to be focused on. Here, we can see them in order, while not all principles are matched with a single standard, these represent the best practices to use in all coding projects for the company. |
| **4** | The coding standards are listed here, labeled and arranged according to threat level. As you can see, string and memory management are at the top of the list since they typically harm systems the most and for the longest period of time. |
| **5** | Here is a simple threat matrix based on the 10 coding standards in the previous slide. |
| **6** | There are three states of encrypting policies which include in flight, rest, and in use. In flight refers to data being transferred between devices, at rest is to protect data that is stored on the device, and in use is to protect data that is currently being in use. |
| **7** | The words Authentication, Authorization, and accounting stand for the Triple-A Framework. Authentication is the process of confirming that you, the user, are who you claim to be. Verified devices or two-factor backup codes can be used for this. Permissions are the main focus of authorization: can the user of the system access the content they are attempting to access? Generally, we adhere to the least privilege principle to stop unauthorized, high-level users from accessing the system. Lastly, recording everything on the system is known as accounting, or auditing. Since no system is 100% safe, knowing as much as possible about an attack can help stop similar ones from happening again. |
| **8** | Here is a screen shot of a few unit tests that have passed googles unit testing framework. These include Check size that verifies the max size is equal or greater than the max size for x number of entries. Another unit test is Outrange to test if an out-of-range exception is thrown when calling out of bounds. |
| **9** | This is the Develops pipeline, an already-in-use variation of the DevOps pipeline that is more efficient and security-focused. We can implement automation in a number of places to relieve the burden on developers and keep us informed about system developments. To be more precise, we will automate the use of multiple external tools to verify policy compliance, dependencies, and security vulnerabilities during the "Verify and Test" stage. |
| **10** | This is a quick rundown of what a Develops pipeline is, along with more details on some of the technologies that are suggested for the particular security policies created for Green Pace. Among the many tools available for identifying vulnerabilities in a code base are Clang and Astree. |
| **11** | There are considerably more benefits than drawbacks to taking early action and putting automation and safety measures in place. Imagine a hacker who manages to get past security measures to steal user data. Any upfront costs are outweighed by the possible loss of value in the form of trust or compensation. |
| **12** | A strong policy provides a strong foundation to grow on. While keeping a policy is great it is also important to audit and review the policy on a regular basis as well as when gaps are found. Another recommendation is to keep strictly to triple a policies. |
| **13** | Particular attention should be made to the use of strings, buffers, and memory management, especially when working in C++, even though all standards should be implemented to further strengthen the security of both the systems and applications.   It's also crucial to take into account log files, or the Triple-A Framework's "Accounting" component. Attacks will occur, and our response will be solely based on the information we are able to obtain regarding the vulnerability. |
| **14** | Here are the references I used in this presentation, and I hope you consider implementing this security policy. |