**CS 405 Project Two Script Template**

Complete this template by replacing the bracketed text with the relevant information.

|  |  |
| --- | --- |
| **Slide Number** | **Narrative** |
| **1** | This is the Green Pace Security Policy. We will talk about what it is and why each aspect is important. |
| **2** | We will be focusing this policy around a security strategy called defense-in-depth to help prevent any potential attacks. We will use a set guidelines that are designed to insure that best security practices are being followed. This allows for a clear and formal framework that should be easy for all team members to learn and follow. |
| **3** | This threats matrix has our 10 coding standards categorized into different levels of potential threat. There are 4 variables that we can have on a threat. It can be high or low impact and it can be probable or improbable. Depending on which of these variables the threat falls under determines if it is unlikely(improbable), likely(probable), low priority(improbable/low impact), or priority(probable/ high impact). |
| **4** | These are the ten principles of security that our standards can be categorized into (Specify which each of our standards fall into). Explain what these principles are for the ones that our policy falls under (1,3,4,8,9,10). \*Use security policy document to summerize\*. |
| **5** | Here are the 10 coding standards in order of threat level. This is determined by the probablity of the threat combined with the impact that each threat would cause. Like on the threat matrix a high probablity with high impact is the top priority and low probability and low impact will be a low priority. |
| **6** | There are 3 encryption policies the first being encryption at rest  **- This process involves securing data on disks or backup media from unauthorized access, typically through strong encryption. Key policies include limiting access to encryption keys to a select few and regularly rotating the keys for added security.**  The second is encrytion in flight  - **This process secures data being transmitted between networks, especially over the internet, where it could be intercepted. Using protocols like TLS helps prevent unauthorized access during data transfer.**  Last we have encryption in use   - **This process secures data actively being used by an application. Techniques like homomorphic encryption or secure multi-party computation ensure data remains encrypted during use. By monitoring and logging access to unencrypted data enhances accountability for users with access.** |
| **7** | Then we have the Triple-A Policies these are Authentication, Authorization, and Accounting.  **- This process verifies a user or system's identity before granting access to data, typically through a username and password. Security is enhanced with additional measures like two-factor authentication.**  **-This process determines a user's access level after authentication. Assigning roles based on the necessary access for their tasks is a good policy, following the principle of least privilege to minimize unnecessary permissions.**  **-This process involves logging and monitoring user actions within a system. It's crucial for tracking access to sensitive information and preventing unauthorized use. Regularly reviewing these logs helps detect and prevent malicious activities and ensures that users adhere to their authorized roles.** |
| **8** | The next few slides are examples of unit tests. These tests are very imporant for early bug detection, can improve overall code quality, and can be automated to ensure the code is always being tested.  This test is ensuring that the collection is always able to hold the entries that are being added. This can help prevent an errors such as a buffer overflow. |
| **9** | This Unit test is to ensure that a collection is cleared when it is being set to zero which should reset the collection. If this is not done properly it can cause memory leaks or leave behind sensitive data. |
| **10** | This Unit test is used to protect against invalid index access errors. This is to ensure that if an error like this was to happen there is a defined way to handle it rather then something unexpected happening or it crashing. |
| **11** | This last test is verifying that when trying to erase a specified range of elements that they are actually being erased. Similar to the ResizeToZero test is ensures that there is no data that is unintentionally left in. |
| **12** | This is the DevSecOps Diagram. This stands for Development, security, and operations. Security and automation tools are used throughout the entire process which includes creating the system, maintaining it after launch, and fixing issues that may come up. The tools used play a critical role in this process and help to ensure the best possible product security. |
| **13** | \*Just read the slide on this one\* |
| **14** | Here are some recommendation for the future of the policy  - You will never be able to prevent every threat, learn from the mistakes  There is a reason this is a major role in the development process and a reason that companies spend a lot of money and still have vulnerablilties. Take any issue that has ceom up learn from it and update the policy to help prevent more threats in the future.    - Make sure the automation involved in the threat detection is not relied on to heavily.  Having threat detection automated allows for attacks to be discovered and dealt with faster and can prevent prolonged damage, but this can lead to a false sense of security. An example of something like this is the recent CrowdStrike incident that effected the entire world and is estimated to be at least a $10 billion dollar error. This error led to airports, banks, and even governmental departments to have outages. This was an error due to the company realasing an update that was not properly tested. |
| **15** | In conclusion some ideals to remember to ensure the best level of security.  There are guidlines laid out and best practices in this policy that are there to make sure the system is secure, use and follow them.  If a new issue is discovered act quickly and folow the guidelines and practices to ensure it is fixed correctly.  Remember the saying "Don't leave security until the end" which is the idea of making sure that security is always a part of the process, don't ignore vulnerabilities and try to fix them all at the end.  Continue to maintain and monitor the system to ensure there are no ongoing or potential attacks. |