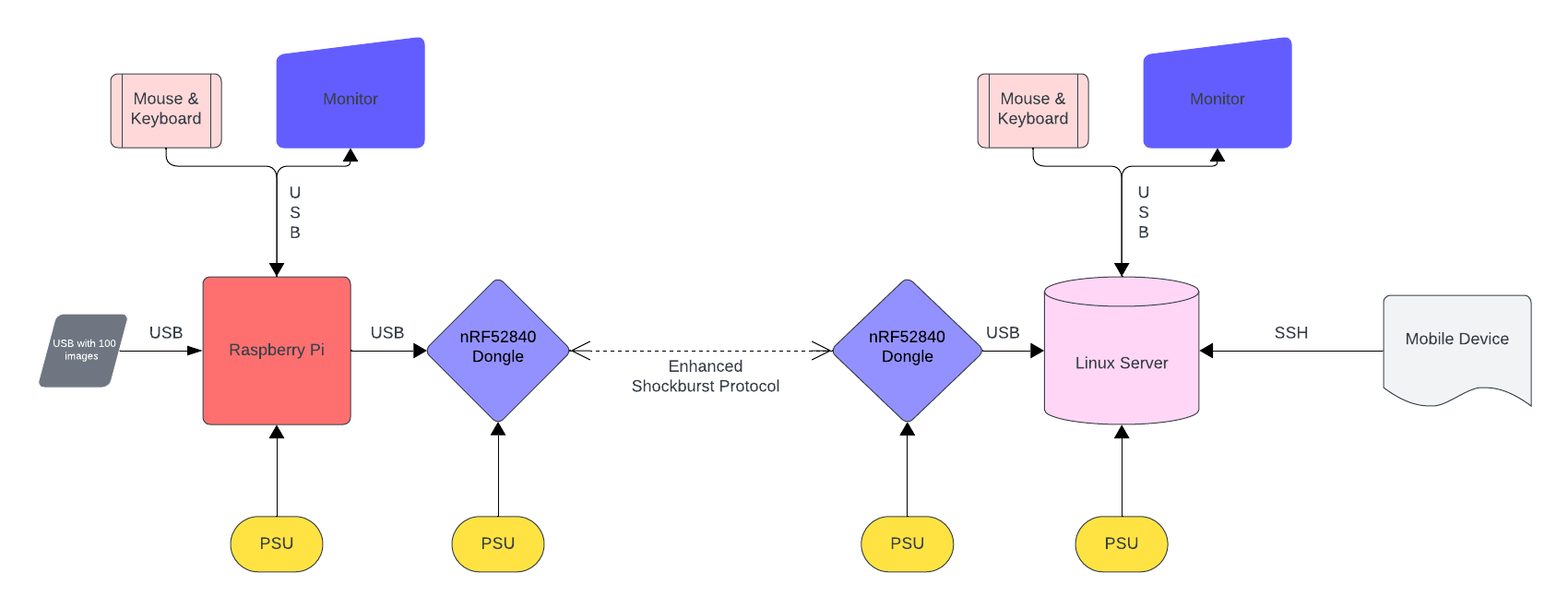
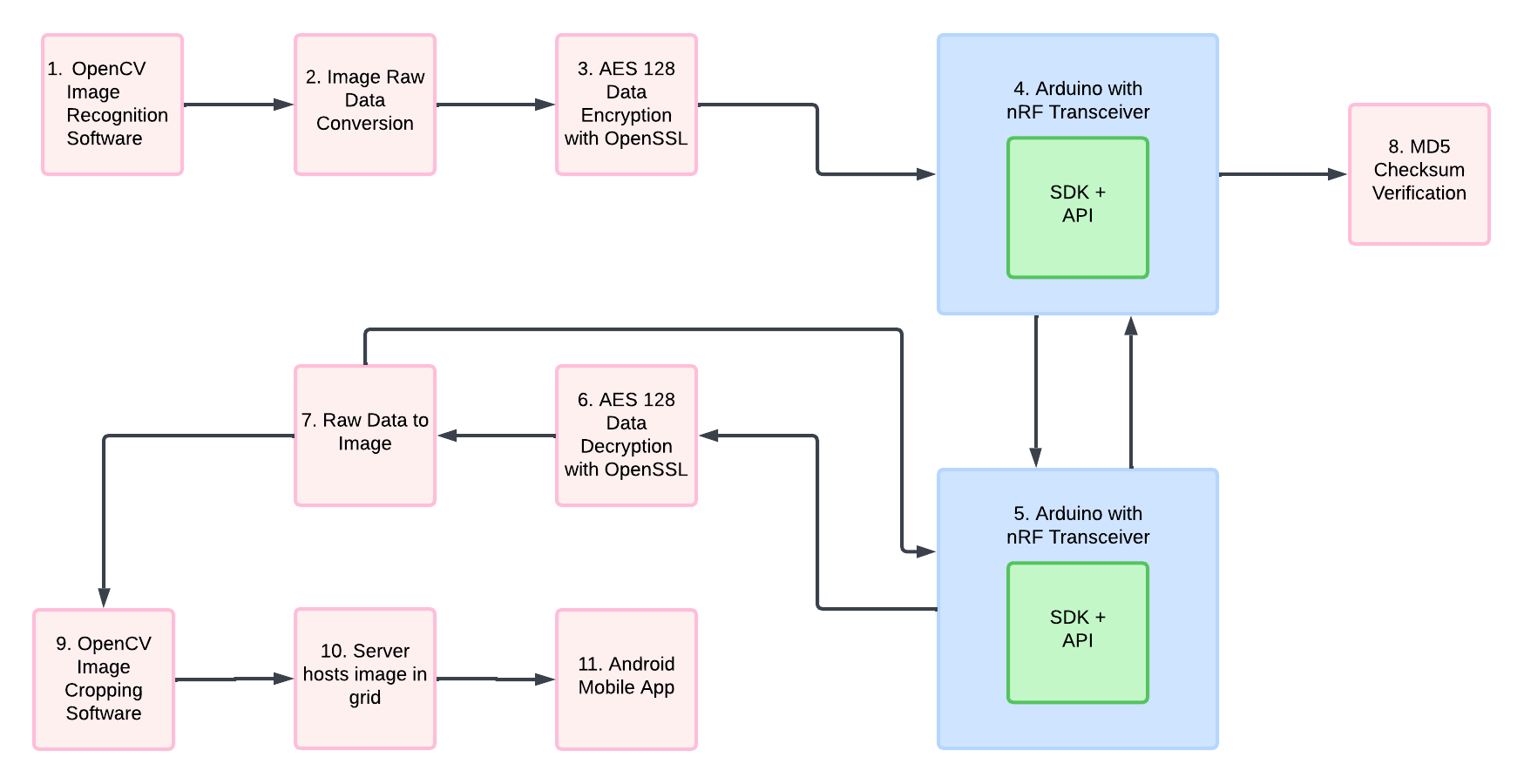
|  |  |
| --- | --- |
| **Group 8** | **Death Star Image Exfiltration** |
| **Major:** | **Team members:** |
| EE | Nicholas Michael |
| CEG | Mason McDaniel |
| CS | Michael Mowad |
| Cyber | Cade Wrinkle |
| Cyber | Chase Ennis |

**Implementation Plan**

**Project Overview**



**Interface Control**



Physically, an nRF52840 dongle is connected to the Raspberry Pi via USB and another nRF52840 dongle is connected to the Rebel Server via USB as well. The two are positioned on opposite sides of the air gap. The Raspberry Pi is also connected to a USB mass storage device via USB with 100 images to sort through. It is planned that the Death Star images are discerned and isolated via a Python script using OpenCV; the images are encrypted with AES-128 using OpenSSL and loaded from the Raspberry Pi to the nRF52840 for transmission via USB. The nRF52840 attached to the Raspberry Pi then initiates transmission to the reacting nRF52840 attached to the Rebel Server via ESB. The reacting nRF52840 receives the images and the pair performs an MD5 hash exchange to verify their contents. The images are transmitted from the reacting receiver to the Linux server via USB. The Linux server then decrypts the received images using OpenSSL. The decrypted images are to be read in via a python script and the weaknesses are to be highlighted by way of OpenCV. From there, the images moved to a directory to be hosted by the server. The mobile app, written in Java using Android Studio, displays the images.

**Budget**

**A screenshot of a computer

Description automatically generated**

**Schedule**

See Attached Excel Sheet (too big to include in this Word Document)

**Risk Analysis**

Format for all tables = *Likelihood x Severity*

Likelihood is on the Y-axis

Severity is on the X-axis

**Risk 1: Parts not arriving on time**

**Mitigations: Order the parts ahead of time**

**Unmitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  | **X** |  |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Mitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  | **X** |  |  |  |
| **1** |  |  |  |  |  |

****

**Risk 2: Risk of fire:  
Mitigations: Take proper safety measures (i.e.: gloves, fire extinguishers, etc.)  
  
Unmitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  | **X** |

****

**Mitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **1** |  | **X** |  |  |  |

****

**Risk 3: Someone gets sick or has an emergency for an extended time.**  
**Mitigations: Communicate via discord, work from home wherever possible, stay up with hygiene, etc.**

**Unmitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  | **X** |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Mitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** | **X** |  |  |  |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Risk 4: Raspberry PI is not fast enough to distinguish the 10 Death Star Images on test day**

**Mitigation: Have a backup module on standby if the speeds on our primary RF module are not fast enough**

**Unmitigated: Mitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  | **X** |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  |  | **X** |  |  |
| **1** |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Unmitigated:** | | | **Mitigated:** | | |
|  | Severity |  |  | Severity |  |
| Likelihood | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **1** | **2** | **3** | **4** | **5** | | **1** |  |  |  |  |  | | **2** |  |  |  |  |  | | **3** |  |  |  |  |  | | **4** |  |  |  |  |  | | **5** |  |  |  |  |  | | | Likelihood | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **1** | **2** | **3** | **4** | **5** | | **1** |  |  |  |  |  | | **2** |  |  |  |  |  | | **3** |  |  |  |  |  | | **4** |  |  |  |  |  | | **5** |  |  |  |  |  | | |

**Risk 5: The Raspberry PI does not work with the Dongle NRF52840 communication device**

**Mitigation: Our group is planning to order backup communication devices in the case that this happens**

**Unmitigated**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  |  | **X** |  |  |
| **1** |  |  |  |  |  |

****

**Mitigated**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** | **X** |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Risk 6: There is too much interference and data transmission has errors**

**Risk 7: Communication is not established**

**Risk 8: Images are not pulled from the USB**

**Risk 9: Images are not filtered from the USB**

**Risk 10: Image encryption for transmission breaks the transmission**

**Risk 11: Hash cannot be verified**

**Risk 12: Incorrect images are transmitted and displayed which takes up time**

**Risk 13: OpenCV does not communicate with the server and filter the images**  
**Mitigation: Test in advance and have a backup option that we have researched on standby**

**Unmitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  | **X** |  |  |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Mitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** | **X** |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Risk 14: Mobile app doesn’t connect to the server to receive images.**  
**Mitigation: Perform advanced testing to ensure the browser properly handles the 10 images  
  
Unmitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  | **X** |  |  |
| **3** |  |  |  |  |  |
| **2** |  |  |  |  |  |
| **1** |  |  |  |  |  |

****

**Mitigated:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **5** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **2** |  | **X** |  |  |  |
| **1** |  |  |  |  |  |

****