Secure Camera Capture System

Christian Coffield
Matthew Dekoning
Nathan Lea
Kevin Seitz
Advisor – Dr. Stine

The Team

- Christian Coffield
- Matthew Dekoning
- Nathan Lea
- Kevin Seitz

The Project

“A system that snaps a picture and stores the picture in memory securely. A good system will enable a security protocol, hopefully with military-grade encryption, so that any system cannot possibly be compromised by a user or thief. The picture should also be able to be recalled at a later time.”

http://www.aviorlando.com/wp-content/uploads/2012/08/Security-Camera.jpg

The Goal

To create a secure camera system that is capable of taking pictures and securely storing them offsite to a secondary location for later retrieval.

http://strategicmodularity.com/2013/03/lean-goals/

Hardware Specifications

- Take and upload a picture every minute
- Take picture of at least 640x480 px
- Tamper protection - if stolen, the system will not retain any sensitive data
- Store the picture securely on the remote server
- Store a year's worth of pictures on the server

Software Specifications

- A method for the user to retrieve the pictures remotely
- Use AES-128 or AES-256 as the standard for encryption
- Securely store the AES key
- A set up process to generate a unique key for the system
- Store images named with time and date stamp
- Images accessible from any standard Windows computer

Block Diagram

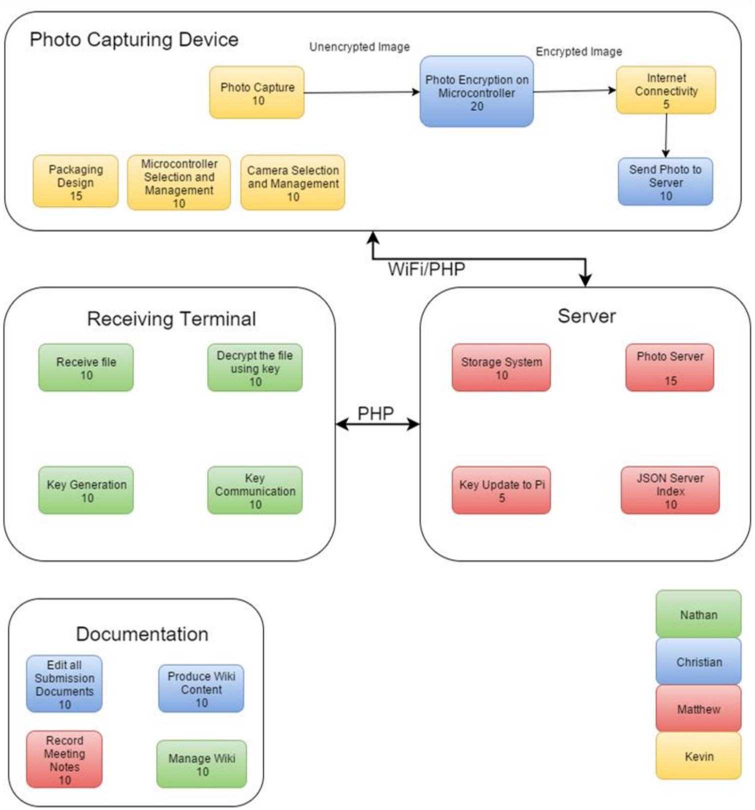


Photo Capturing Device

Unencrypted Image

Encrypted Image

Photo Capture 10

Photo Encryption on Microcontroller 20

Internet Connectivity 5

Send Photo to Server 10

Packaging Design 15

Microcontroller Selection and Management 10

Camera Selection and Management 10

WiFi/PHP

Receiving Terminal

Receive file 10

Decrypt the file using key 10

Key Generation 10

Key Communication 10

PHP

Server

Storage System 10

Photo Server 15

Key Update to Pi 5

JSON Server Index 10

Documentation

Edit all Submission Documents 10

Produce Wiki Content 10

Record Meeting Notes 10

Manage Wiki 10

Nathan

Christian

Matthew

Kevin

Development Plan

Three phases

- Photography/Encryption
- Off-Site Storage
- Retrieval/Decryption

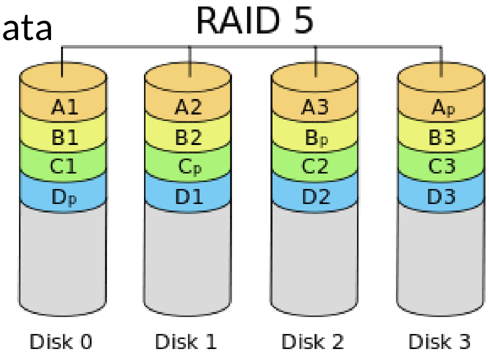
Phase I: Photography/Encryption

- Raspberry Pi Zero captures a photo using a USB connected camera
- Images encrypted
  - AES-128 or AES-256
- No on-board images, server storage
- Will erase all records on device if opened

http://cdn.shopify.com/s/files/1/0188/6794/products/Zero_1_of_5_1024x1024.jpeg?v=1448521030

Phase II: Off-Site Storage

- Raspberry Pi 2 server safely stores and serves encrypted images
- Secured through:
  - Firewall, fail2Ban, pre-encrypted data
- Technologies Used:
  - RAID, ginx, PHP, HTML



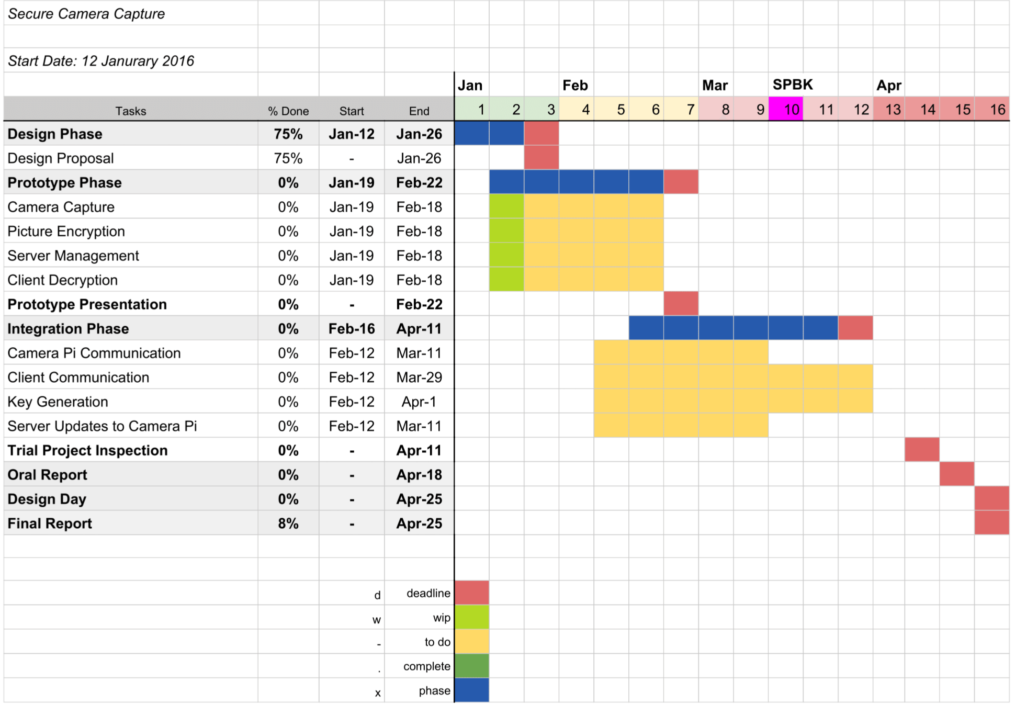
https://upload.wikimedia.org/wikipedia/commons/thumb/6/64/RAID_5.svg/300px-RAID_5.svg.png

Phase III: Retrieval/Encryption

- Software client running on any windows system to retrieve and decrypt images from the server
- Will allow the server to be reconfigured with a new password and encryption key
- Clean, user-friendly GUI

http://1.bp.blogspot.com/-9IP33yHqh18/VSIPO4zrxuI/AAAAAAAABQw/PkIQxaO-8JQ/s1600/C6.png

The Schedule



Secure Camera Capture

Start Date: 12 Janurary 2016

				Jan			Feb				Mar		SPBK			Apr			
Tasks	% Done	Start	End	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Design Phase	**75%**	**Jan-12**	**Jan-26**	🟦	🟦	🟥													
Design Proposal	75%	-	Jan-26			🟥													
Prototype Phase	**0%**	**Jan-19**	**Feb-22**		🟦	🟦	🟦	🟦	🟦	🟥									
Camera Capture	0%	Jan-19	Feb-18		🟩	🟨	🟨	🟨	🟨										
Picture Encryption	0%	Jan-19	Feb-18		🟩	🟨	🟨	🟨	🟨										
Server Management	0%	Jan-19	Feb-18		🟩	🟨	🟨	🟨	🟨										
Client Decryption	0%	Jan-19	Feb-18		🟩	🟨	🟨	🟨	🟨										
Prototype Presentation	**0%**	**-**	**Feb-22**							🟥									
Integration Phase	**0%**	**Feb-16**	**Apr-11**						🟦	🟦	🟦	🟦	🟦	🟦	🟥				
Camera Pi Communication	0%	Feb-12	Mar-11					🟨	🟨	🟨	🟨	🟨							
Client Communication	0%	Feb-12	Mar-29					🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨				
Key Generation	0%	Feb-12	Apr-1					🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨				
Server Updates to Camera Pi	0%	Feb-12	Mar-11					🟨	🟨	🟨	🟨	🟨							
Trial Project Inspection	**0%**	**-**	**Apr-11**														🟥		
Oral Report	**0%**	**-**	**Apr-18**															🟥	
Design Day	**0%**	**-**	**Apr-25**																🟥
Final Report	**8%**	**-**	**Apr-25**																🟥

d	deadline	🟥
w	wip	🟩
-	to do	🟨
.	complete	▩
x	phase	🟦

Management/Organization

- Weekly advisor meetings, transcribed
- Google Drive, GitHub
- Group compilation on documentation

Managing Roadblocks

- Camera interface, openCV
  - fswebcam

- Motion sensing versus timer

- Microcontroller power
  - BeagleBone Black

http://willowoakcounseling.com/wp-content/uploads/2013/12/roadblock.jpg

Necessary Resources

Resource	Cost
Google Drive/GitHub	Free
Raspberry Pi 2	$35
Raspberry Pi Zero	$5
Webcam/Camera	$15
PIR Motion Sensor	$3
USB Hard drive/flash memory storage	$60

http://cdn.graphicsfactory.com/clip-art/image_files/image/3/1330433-Happy-Salesman-with-Cash-2.gif

In closing...

- Three Part System
  - Camera device, storage server, remote application
- AES encryption
- Tamper- and theft-resistant

Questions?