

OSYS2022

Project

Build Your Own Script

Introduction

The last two projects have been merged into this one larger "project" that will have a checkpoint at the midway mark to monitor your progress. You will choose your own area of focus to build out a script, then add additional functionality and polish as needed.

This project will be more open ended due to the flexibility offered in scripts you can build. You will be provided with a variety of videos and resources, but the onus will be on you to wade through that information and pick out the building blocks you need to create your script. This will mirror what it is like for you to create scripts in industry except that I will be here to help!

Required Resources

You will require the following to perform this project:

- Script ideas on Brightspace (Content → Module 4 – Script Project)
- All videos, content, and resources you can get your hands on

Marking

Outcome(s) being measured:

- Outcome 1: Implement management and configuration settings in a Linux environment
- Outcome 2: Apply command line syntax and skills required to manage a Linux system
- Outcome 3: Secure a Linux environment using scripting to manage users, groups, and process creation
- Outcome 4: Create and implement a Certificate Authority for a client/server environment
- Outcome 5: Create and configure a VPN on a Linux system working as a server

Prep Task 1 – Choose a Script

Before diving in to watching videos and wading through resources, you should decide on a script that you want to build. You will likely find it much easier when working through content if you have a script in mind to provide context. You should choose a script to build that will allow you to push your current skill set while also not being completely unrealistic in size/complexity.

Use the following resources to decide on what script you would like to build:

- 'Script Ideas & Resources' in Brightspace
- Google
- GitHub
- Classmates
- Scripts you have seen referenced elsewhere
- Your own experiences

If you are struggling to identify a script that is of interest to you, work through the following:

- Decide if you want to tackle an offensive/defensive script directly related to cyber-attacks, or more on the system administration side of things (you will find there is a much larger amount of resources online for system admin scripts than cyber security specific)
- Think through tasks you enjoy performing in Linux and think through how you could automate them
- Think through your experiences in ISEC2025 and PROG2022 (if you are in those courses) within Linux. What tasks have you performed that you would like to automate? What tool have you used that you would like to build out your own version of?
- Discuss with me to talk through potential options based on your areas of interest (this should be a last resort once you have performed the steps above and performed your own research)

<Project continues next page>

Task 1 – Build Your Script

This script will be worked on in two separate phases due to the extended duration of the project:

- 1) Mid-project checkpoint criteria (detailed in 'Required Task 2')
- 2) End of project criteria (detailed in 'Required Task 3')

MID-PROJECT CHECKPOINT CRITERIA:

- Draft version of the core contents of script is complete
 - Pseudo-code can be used for up to 50% of the script's functionality

END OF PROJECT CRITERIA:

- Shebang
- Executable
- Uses variables
- Takes a parameter as input OR prompts the user for input
- Uses at least one of the following flow control types:
 - 'if' branching
 - 'case' branching
 - 'while' loop
 - 'for' loop
 - 'until' loop
- Any output created by your script (whether to screen, file, or something else) must be cleanly formatted, structured, and suitable for use in an organization
- Code is easily readable, incorporating spacing and indentation
- Comments are provided for every line explaining what that line in the script does and how it works (reminder that in our educational perspective this is your best opportunity to convince me that you understand how your script works)
- Your script must successfully perform its core functionality

Due to the open-ended nature of this project, I am unable to be more specific on the criteria. I reserve the right to dock marks if I feel you low-balled your script in any way, especially considering the extended timeframe and opportunities for feedback.

Task 2 – Mid-Project Checkpoint

Due to the extended duration of this project, you are to perform a check-in with me by the 'Mid-Project Checkpoint' deadline in Brightspace:

- Submit a copy of the script as it stands to the 'Mid-Project Checkpoint' submission box on Brightspace by the deadline
- Ask any questions you may have
- Provide your own thoughts on how it is progressing so far and what you still have left to complete
- You will receive feedback on how the script is shaping up and things you could improve on
- We will assess what you can feasibly accomplish in the remaining time available, specifically on what functionality you could realistically build out and get working successfully

Task 3 – Complete Your Script

Incorporating the feedback received from the mid-project checkpoint, you will now focus on refining your script to meet the 'End of Project' criteria that are listed above.

Once your script is complete and meets the 'End of Project' criteria, you should copy and paste the script contents into a document and submit it on Brightspace. DO NOT submit a screenshot of the script as I need to execute the script as part of my marking.

<Project continues next page>

Task 4: Share Script with Class

To receive full marks, you need to share your script with the rest of the class via Teams:

- Channel: OSYS2022 --> Script Project - Presentations
- Start a new thread in the channel above and share your script with the class

When you share the script, you should cover the following points to get full marks:

- Introduce your script and the idea behind it
- Provide an overview of the code in the script and how the script works (one to two paragraphs)
- Video or screenshots demonstrating the scripts usage