

Nifty Assignment – Rubber Ducky

Brian Leeper

University of Hawai'i Maui College

Table of Contents

Table of Contents	2
Background	3
Meta Information	3
Additional Descriptions	3
Before you Start	4
Supplies Needed	4
Rubber Ducky Components	4
Duck Encoder	6
Handouts	6
Ducky Script	6
Assignment Instructions	6
Starter and support code file	7
Model grading criteria	7

Background

The USB Rubber Ducky is a Human Interface Device (HID) that mirrors a keyboard. With the Rubber Ducky, any actions that could be performed with a keyboard (there's a lot) may be automated through the use of the Rubber Ducky. The USB Rubber Ducky uses "payloads" to deliver keystrokes that begin to run when the USB is plugged into a powered USB port. The USB Rubber Ducky uses are numerous, ranging from utilitarian application in the workplace to malicious use involving the trickery or theft of information.

Meta Information

Summary	USB Rubber Ducky – Create and run simple payloads on the USB Rubber Ducky to showcase the application of injection operations.
Topics	Injection attacks, Ducky Script, USB Rubber Ducky
Audience	Students familiar with basic coding and an interest in technology gadgets.
Difficulty	This is designed to be a simple assignment with one application of the USB Rubber Ducky. Application of the Rubber Ducky ranges from malicious use by inside attackers to utilitarian use by system administrators. While possible to complete with no prior knowledge, a basic understanding of code will help the student complete the assignment.
Strengths	Introduction to technology gadget that will interest and engage students.
Weaknesses	May be difficult for the student to grasp command options utilizing only the keyboard and translating the input to ducky script.
Dependencies	USB Rubber Ducky
Variants	Expand focus to more advanced payloads. Additionally, students could work on obfuscation and optimization of their code.

Additional Descriptions

Assignment Description	This assignment is meant to be a hand-on lab example of writing ducky script, encoding the script via a Duck Encoder and injecting the script with the USB Rubber Ducky.
Structure Description	The assignment is narrow in focus with directions for one application. Following the

	given example, students may use what they learned and create their own payloads.
How it Works Description	The encoded payload, loaded on the Micro SD card and into the USB Rubber Ducky, is injected into a computer and pre-written commands that mimic a keyboard are performed.

Before you start

Supplies Needed

A physical USB Rubber Ducky set is needed to complete this assignment. They may be purchased from Hak5 from their website at <https://shop.hak5.org/collections/physical-access/products/usb-rubber-ducky-deluxe>. While there are similar products on the market, other products do not utilize Ducky Script and do not fall within the purview of this assignment.

Rubber Ducky Components

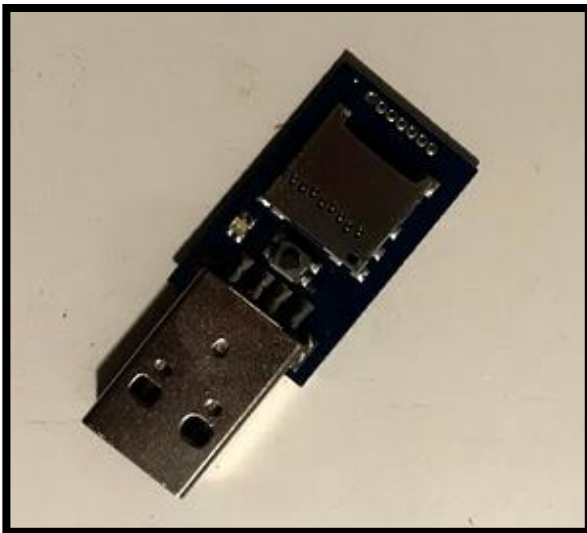


Contents of USB Rubber Ducky package



microSD to USB Adapter

This dongle is used to mount the microSD card to edit the contents of the microSD card.



Mini "keyboard" adapter

This is recognized as a keyboard by the computer. the microSD card is inserted into the slot on the top of the card. This USB is plugged into the computer to run the injectable. In testing, there is a small black button that when pressed allows the injectable code to be re-run, especially useful in testing.



USB Rubber Ducky with incognito "flash drive" disguise

Duck Encoder

Ducky Script, the basic scripting language that payloads are written, may be written in any text editor; however, the USB Rubber Ducky does not read plain text files—it expects a binary injection file. A Duck Encoder converts human readable text files into inject.bin payloads that are ready to be used on the device. There are three ways that text files may be Encoded into inject.bin payloads:

- Java Based Command Line Encoder
 - Download from resources of Hak5 website
- Java Based Graphical Encoder
 - Download from GitHub repository at <https://github.com/moritzgloeckl/duckygui>
- Online Encoder
 - <https://ducktoolkit.com/>

For simplicity, this assignment will utilize the online encoder as there is no need to download any additional program to encode the text file.

Handouts

Ducky Script

The language of Ducky Script is a simple language. Below are a few necessary commands to know to get started:

REM	Tag for comments
DELAY	Set a delay in milliseconds
STRING	Processes the text string
GUI	Emulates the Windows Key, combine with other keys to perform actions
SHIFT	Emulates the SHIFT key
ALT	Emulates the ALT key
CONTROL	Emulates the CTRL key
DELETE	Emulates the DEL key
PRINT SCREEN	To take screenshots, use with GUI
TAB	Emulates the TAB key
SPACE	Emulates the SPACE key

A printable and visually appealing “Cheat Sheet,” created by the GitHub user Jonny Banana may be found at <https://github.com/JonnyBanana/Huge-Collection-of-CheatSheet/blob/master/DuckyScript/RubberDuckyEssentialCheatSheet.pdf>.

Assignment Instructions

1. Open notepad (or a comparable text editor). While an Online Editor will be used to Encode the text, it is best to have the code in a secondary file for editing should an error occur within the browser so progress is not lost.

2. Copy and paste the provided Ducky Script into your text editor. Generous delays have been incorporated to ensure usability on older, slower machines.
3. Encode the text into Ducky Script.
 - 3.1. Navigate to Duck Toolkit at <https://ducktoolkit.com/>.
 - 3.2. Click “Encode Payload.”
 - 3.3. Paste the Ducky Script in the Command Line Editor.
 - 3.4. On the right side, select your language as there may be different keyboard configurations in different languages.
 - 3.5. Scroll down and click “Encode Payload.”
 - 3.6. Read the disclaimer and click “Ok.”
 - 3.7. Click “Download Inject.bin.”
 - 3.8. Insert the microSD to USB device into the computer with the microSD inserted in the slot.
 - 3.9. Navigate to your downloads and locate the inject.bin file. Click and drag the file onto the microSD storage.
 - 3.9.1. For future inject.bin files, ensure you remove the differentiator after deleting the original inject.bin file on the microSD. For example, inject(1).bin is not recognized and ran by the USB Rubber Ducky.
 - 3.10. Eject the microSD to USB device and transfer the microSD card to the USB Rubber Ducky keyboard simulator.
4. Test the inject.bin file through the USB Rubber Ducky. If the payload fails, press the black button on the device to restart. Input from the mouse or keyboard may interfere with the script so no other actions should be taken while the injection is occurring.
 - 4.1. The led indicator light will let you know if there is an error. The light should pulse green if the device is recognized and functioning properly. If there is an issue with the inject.bin file or the microSD card the LED will shine red.
5. Create a new Ducky Script.
 - 5.1. On ducktoolkits.com there is a tab for “User Scripts.” You can review the text in any of the user submitted scripts. **BEWARE**: some of the scripts may cause severe aggravation if implemented on your own device. Do not use any injections you do not understand by ‘walking’ through the code.

Starter and support code file

Starter code for this assignment is available in the GitHub repository.

Model grading criteria

Upon successful completion of this assignment, the student should have successfully navigated the USB Rubber Ducky and related tools, using the Online Editor to encode text files to create inject.bin files. The student should have created a new Ducky Script that, whether successful or not, demonstrates a level of reasonable effort.