

Delays

Overview

The average computer user types at about 40 words per minute. Sure, maybe us hackers type much faster — say 100-120 words per minute — but compared to how fast a computer processes data, that's nothing.

So when we think about issuing commands to a computer by way of keyboard input, there's already an inherent delay simply in that we're comparatively slow humans. Contrast our fastest typing with our multi-core computers with their gigahertz clock speeds, processing billions of instructions per second.

The USB Rubber Ducky doesn't type like a human. It types like a computer. Under its hood it's performing 60,000 processes per second. Often while thinking about building a payload, we forget to add delays because they quite simply aren't obvious to us as humans.

DELAY

The `DELAY` command instructs the USB Rubber Ducky to momentarily pause execution of the payload. This is useful when deploying a payload which must "wait" for an element — such as a window — to load. The `DELAY` command accepts the time parameter in milliseconds.

```
DELAY <time in ms>
```

#Example

```
REM Example Delay

ATTACKMODE HID STORAGE
DELAY 3000
STRING Hello,
DELAY 1000
```

```
SPACE  
STRING World!
```

Result

The resulting payload will pause for 3 seconds, then type "Hello, " followed by "World! " just one second later.

Best Practices

DELAY in your payloads are useful in two key places. First, at the very beginning. When a new USB device is connected to a target computer, that computer must complete a set of actions before it can begin accepting input from the device. This is called enumeration. The more complex the device, the longer it will take to enumerate. In the case of a USB scanner or printer, for example, it may be several seconds or minutes while the computer downloads and installs the necessary device drivers. In the case of the USB Rubber Ducky, acting as a generic keyboard, that enumeration time is very short. Because drivers for USB keyboards, or a HID (Human Interface Device), are built-in, the target computer does not require an Internet connection or a lengthy download and installation time. In most cases, enumeration is only a fraction of a second. However, in some instances a slower computer may take one or two seconds to recognize the USB Rubber Ducky "keyboard" and begin accepting keystrokes. For this reason, it can be helpful to begin a payload with a DELAY statement.

The second useful place to use a

DELAY is throughout your payload when you, as a human, would wait for some action to complete before continuing. Consider opening a web browser; the amount of time it takes for the browser to be open (on the screen) and usable by the user can vary from system to system. As humans we mostly ignore this delay because we have the ability to detect this visually and react accordingly. From your payload's perspective however, a **DELAY** is required to instruct the USB Rubber Ducky that it needs to wait some amount of time before continuing.