Shark Jack

The Shark Jack by Hak5

A portable network attack and automation tool for pentesters and systems administrators designed to enable social engineering engagements and opportunistic wired network auditing.

This documentation covers the basics of operation and deployment, accessing the Linux shell for advanced operations, Internet connectivity, software updates and payload development.

> The e-book PDF generated by this document may not format correctly on all devices. For the most-to-date version, please see https://docs.hak5.org

Getting Started

Shark Jack Basics

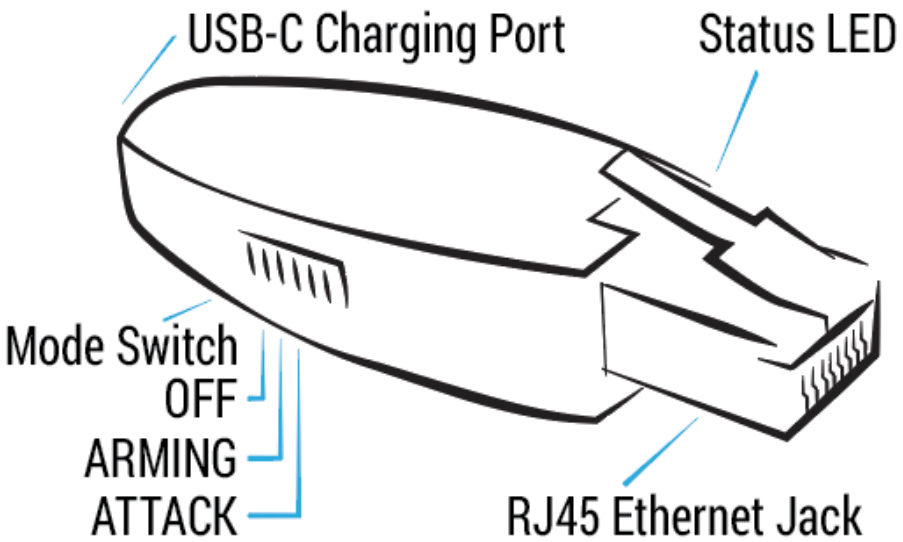
The Shark Jack is a portable network attack and automation tool for pentesters and systems administrators designed to enable social engineering engagements and opportunistic wired network auditing. It features a familiar Hak5 payload architecture, flip-of-the-switch operation and multi-color LED for instant feedback.

This documentation serves both cable and battery variants of the Shack Jack with notable differences highlighted.

VARIANTS

Shark Jack

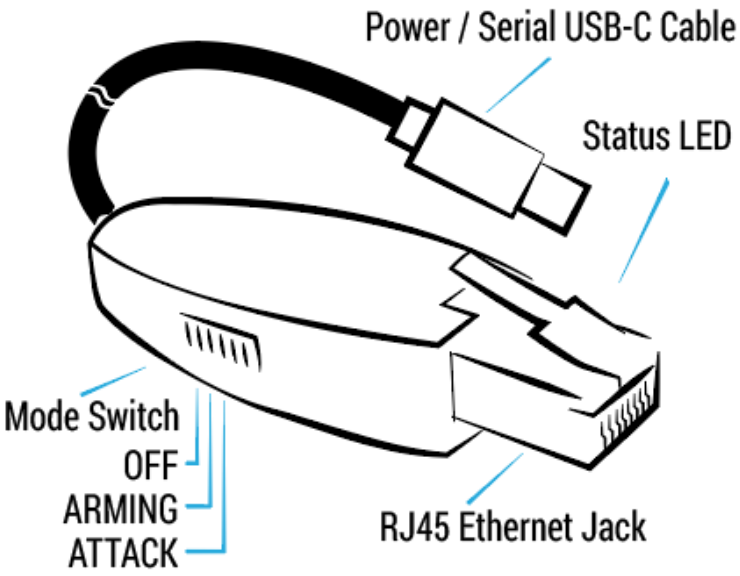
The original Shark Jack is a battery powered device with a 10–15 minute runtime and feedback via RGB LED. It's as comfortable on the keychain as it is in your EDC.



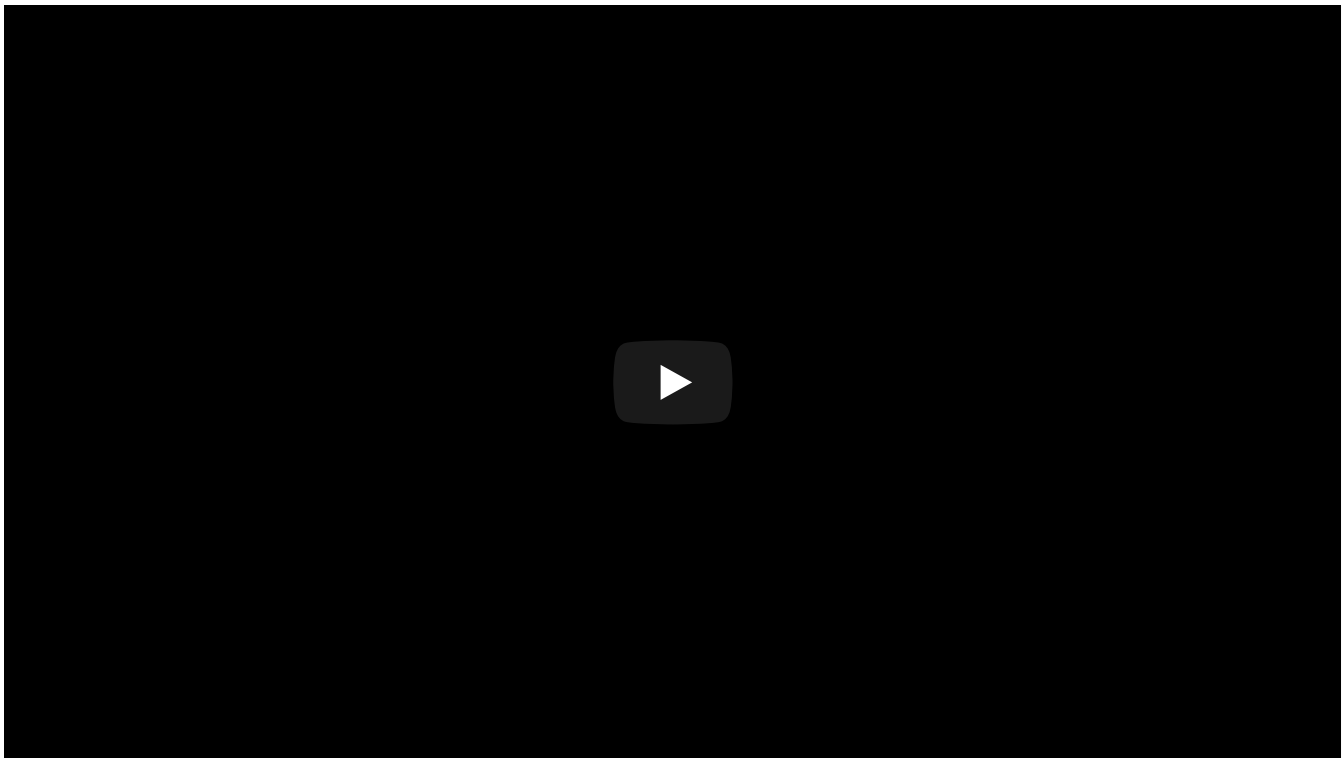
Shark Jack Cable

The Shark Jack Cable is a USB-C powered pentest companion with continuous runtime and feedback via RGB LED and an interactive serial console. It can be planted for long-term headless deployments or run temporarily from a battery source. This can include the operators USB-C powered smartphone [1], which may enable shell access via serial using a third party app [2].

- [1] smartphone compatibility may vary. Tested with 2021-era flagship Android devices.
- [2] serial access on Android tested using the third party Serial USB Terminal app.



Shark Jack Cable Android Serial Setup



Recommended apps: Hacker's Keyboard, Serial USB Terminal.

DEPLOYMENT

The Shark Jack is meant to be deployed against a target network for brief reconnaissance, exfiltration and IT automation tasks. With a fully charged battery, the Shark Jack will operate for about 10-15 minutes. The Shark Jack Cable may run for as long as a standard USB-C power source is available.
Out-of-the-box, a pre-installed default payload executes an nmap scan of the connected target network when the switch is in the attack mode. This default payload saves the scan results to a loot directory on the device.

This loot may be recovered from SSH access when the switch is in the arming mode. Further, with the switch in arming mode the default payload may be replaced with your own payloads, written in bash, or payloads downloaded from the community repository at https://github.com/hak5/shark-payloads

> The Shark Jack Cable provides real time access to loot and the ability to download and activate payloads over-the-air by an interactive serial shell.

CHARGING

The Shark Jack features a non-removable lithium ion battery. Please read all instructions before use and familiarize yourself with the important safety information and warnings.

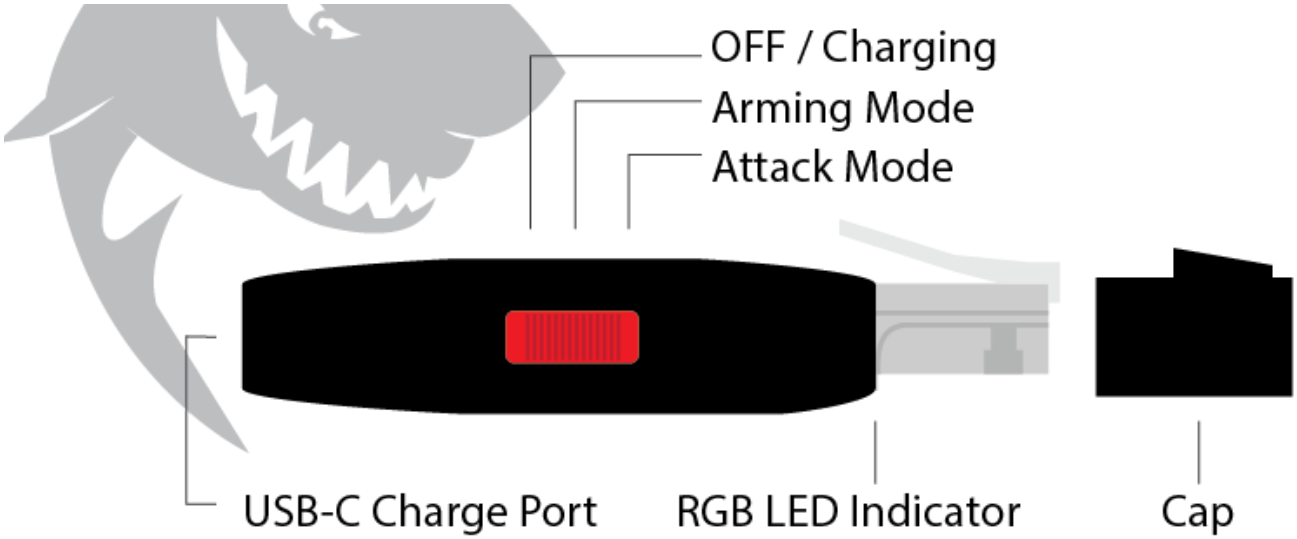
To charge the Shark Jack, flip the switch to the OFF / Charging position. Plug the Shark Jack into a standard USB power source using a USB-C cable. After a brief boot period, indicated by a flashing green LED, the Shark Jack will begin charging.

When the device is charging, the LED will blink blue. When the device is full charged, the LED will light solid blue - at which time the device should be disconnected from USB power. Do not overcharge, and do not leave unattended while charging.

> The Shark Jack cable does not feature an internal battery and does not require charging. The switch positions are the same for both Shark Jack and Shark Jack Cable.

MODES OF OPERATION

Similar to many Hak5 tools, the Shark Jack features an Arming mode and an Attack mode. In Arming mode, the Shark Jack is accessible from SSH for payload loading and configuration. In Attack mode, the selected payload is executed.



ATTACK MODE

In attack mode, the Shark Jack will execute the `payload.sh` or `payload.txt` bash script from `/root/payload`. Most payloads specify a network mode, setting the Shark Jack as a client with `NETMODE DHCP_CLIENT` or a server with `NETMODE DHCP_SERVER`.

ARMING MODE

In arming mode, the Shark Jack will be configured with a static IP address of `172.16.24.1` and will start an SSH server.

With the Shark Jack in arming mode, you may access the embedded linux system via SSH. Connect the Shark Jack to your computer's Ethernet interface, specify a static IP address in the 172.16.24.0/24 range for this interface (for example `172.16.24.2`) then establish an SSH connection to the Shark Jack at `172.16.24.1` (e.g. with the command "`ssh root@172.16.24.1`")

> The Shark Jack Cable provides access to the shell in arming mode via its dedicated serial console. When connected, press `ENTER` to activate the shell and run `HELP` for a list of payload and firmware management commands.

Default Settings

DEFAULT SETTINGS

The default settings for the Shark Jack are:

- Username: `root`
- Password: `hak5shark`

- Arming Mode IP Address: `172.16.24.1`

DIRECTORY STRUCTURE

PATH	Contents
`/root/loot/`	log files and other loot stored by payloads
`/root/payload/`	the payload which will execute when the switch is in Attack mode
`/root/payload/library`	payload library, populated by the `UPDATE_PAYLOADS` command [1]

[1] command available on the Shark Jack Cable and any Shark Jack with firmware 1.2.0 and above.

LED STATUS INDICATIONS

LED	Status
Green (blinking)	Booting up
Blue (blinking)	Charging
Blue (solid)	Fully Charged
Yellow (blinking)	Arming Mode
Red (blinking)	Error - no payload found

SHARK JACK HELPERS AND COMMANDS

> These commands are intended for use with the Shark Jack Cable when connected via Serial in Arming Mode and may be used by any Shark Jack with firmware 1.2.0 and above.

COMMAND	Description
HELP	List Shark Jack helpers and commands

ACTIVATE	Activate a payload
ACTIVATE_PAYLOAD	Alias for ACTIVATE
LIST	List the local payload library
LIST_PAYLOADS	Alias for LIST
UPDATE_PAYLOADS	Synchronize local payload library with remote library
UPDATE_FIRMWARE	Check for and install available firmware updates
SERIAL_WRITE	Write to the serial console
LED	Configure the LED

SHARK JACK SERIAL SETTINGS

- flow control: none
- baud rate: 57600
- parity: none
- databits: 8
- stop bit: 1

> ⓘ Serial Settings apply only to Shark Jack Cable

Beginner Guides

Unboxing and Setup

This article describes the basic process of setting up, deploying the default nmap scan attack and retrieving loot from the Shark Jack.

So your Shark Jack just arrived, you've had a moment to appreciate the sweet metal case it comes in, and now you're eager to dig in and get your hack on! Keep reading.

TL;DR - Read the safety info & diagram on the card. Charge it by USB. Flip the switch far forward and plug it into your LAN. When the light goes green, the trap is clean. Unplug, flip the switch to the middle position and plug it into your computer. SSH to `root@172.16.24.1` (pass: '`hak5shark`') and find the scan results in /root/loot.

Cool, then what? Read up at docs.hak5.org, grab the latest firmware & tools from downloads.hak5.org, get chatty at forums.hak5.org, spin up your very own Cloud C2 server from c2.hak5.org, and finally nab & contribute payloads from payloads.hak5.org. Enjoy!

Now obviously the prudent first step would be to RTFM (which you can find at docs.hak5.org) but let's throw caution to the wind... That is, with the exception of reviewing all of that important safety information – it does include a Lithium battery after all.

Pay heed.

STEP 1: CHARGE IT UP

Flip that mischievous red attack switch to the OFF position - that's the position closest to the USB-C port. Then plug in a USB-C cable to your charger of choice. The Shark Jack will sip a steady 5 volts and about a half amp. After a brief green-blinking boot, it'll blink blue to show that the battery is charging. Give it a few minutes and it'll light solid blue. Then, unplug it. And as you recall from that important safety information – you don't leave lithium batteries unattended while charging. Just sayin'

> ⓘ Shark Jack Cable users may skip this step as it does not include a built-in battery.

STEP 2: JACK INTO A NETWORK

Unplug your Shark Jack from the USB charger and give it a moment to cool down. It'll be a little warm from charging, but as you know from that important safety information it complies with IEC standards – so you're all good. Just keep that sly red switch in the off position and take a moment to practice wielding the Shark Jack as if a floppy disk while reciting "you talkin' to me, punk?" in the mirror. Where were we again? Oh right - jacking into your network...

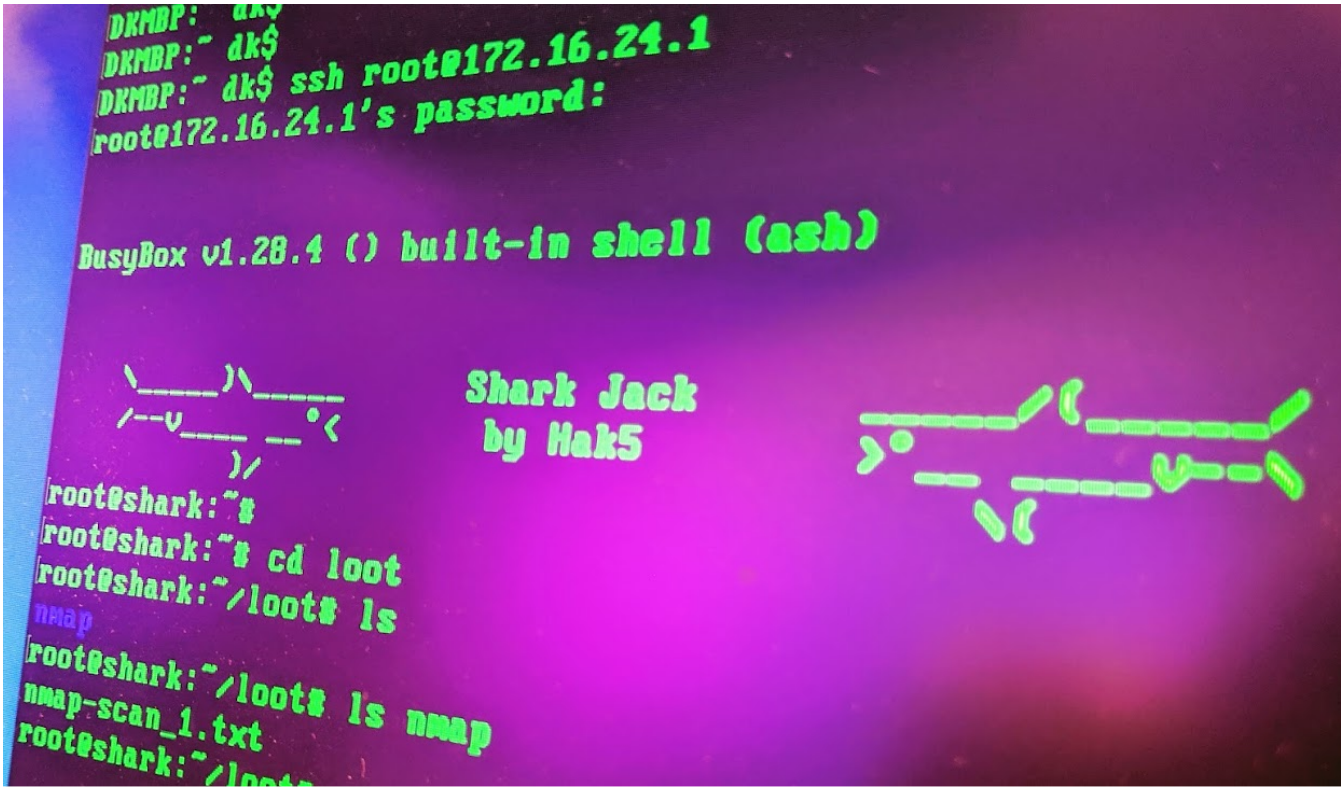
Find that sweet 24 port switch mounted in your network closet, or RJ45 wall plate if you've gone full-geek and wired the house with CAT6. Repeat the phrase "this is my network and I'm totally authorized to pentest it" - then flip the evil red switch far forward, closest to the Ethernet jack, to put it in attack mode. It'll start

booting and blinking - and at any time it's ready to be plugged into your network. Right outa the box it'll perform a simple nmap scan - and when the fun is done, the light will go green (and the trap will be clean). It's then safe to unplug.

STEP 3: SSH IN TO NAB LOOT

Flip that crafty red switch to the middle position to put your Shark Jack into arming mode. Here instead of being a client on your network, it'll act as a server. Connect it to your computer's Ethernet port and you'll get assigned an IP address in the 172.16.24.x hood. From there just open terminal if on MacOS or Linux, or powershell if you're on Windows. Then SSH into the Shark Jack with the command "`ssh root@172.16.24.1`". The default password is "`hak5shark`". After admiring the cute shark ASCII art in the banner, cd over to /root/loot and enjoy those scan results. Want to change the payload? Just copy a `payload.sh` to `/root/payload` – and there's a growing collection at payloads.hak5.org

> ⓘ Shark Jack Cable users may access the shell via Serial rather than SSH as no Ethernet interface is required.



Want to make all that even easier? Download the `sharkjack.sh` script (for Mac and Linux) from the Shark Jack section of downloads.hak5.org – it'll automate gathering loot, swapping out payloads and even upgrading the firmware (you'll wanna do that – new features and tools are added from time to time).

THAT'S IT FOR THE BASICS

Of course you should also familiarize yourself with the ins and outs detailed in the official Shark Jack documentation at docs.hak5.org.

Once you start diving in and testing payloads, you may also want to set yourself up with a Cloud C2 server – it's great for remote access and exfiltration and is supported by a lot of the payloads. There's a free community version to be had at c2.hak5.org, and it runs on your own hardware, so we never see your bits (we don't wanna see 'em).

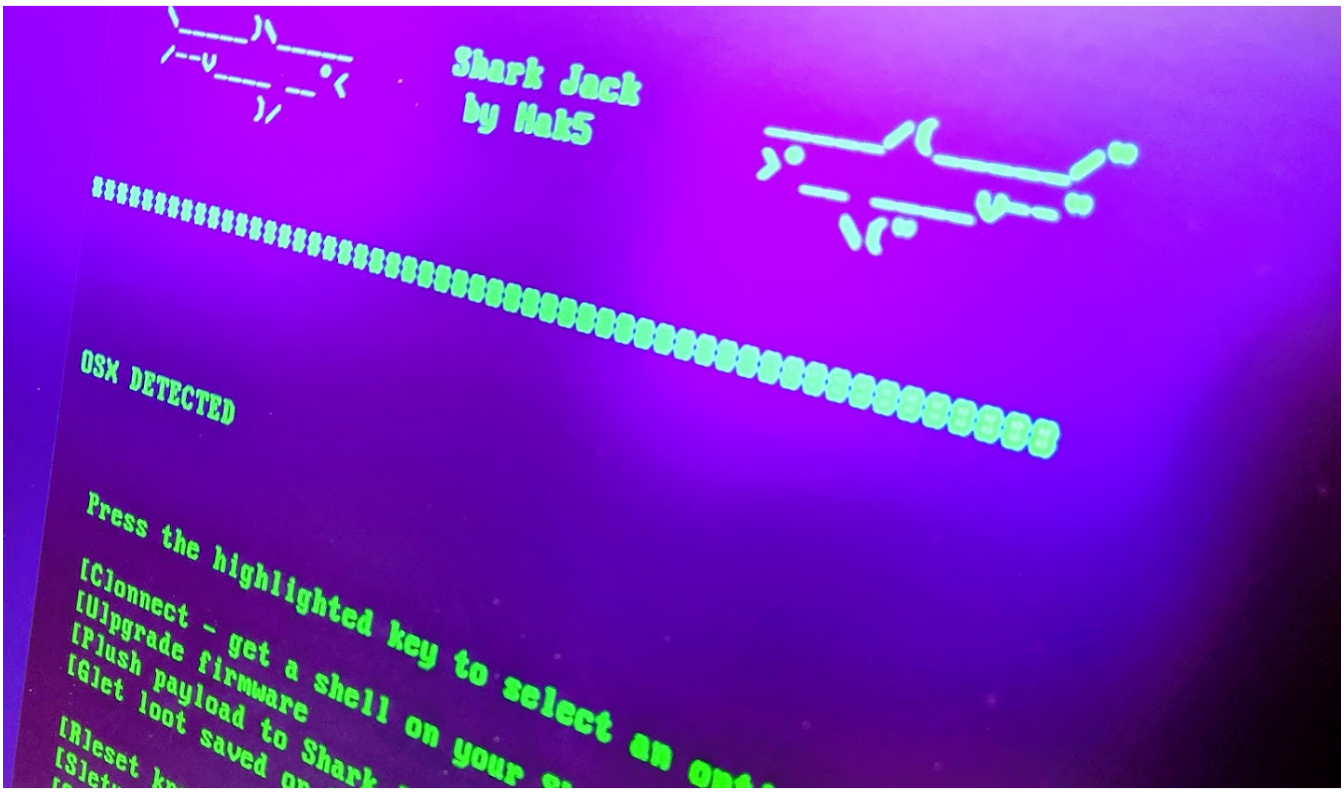
And finally, join the community at forums.hak5.org – it's home to some really bright pentesters and enthusiasts just like yourself, so you'll fit right in. Cheers!

Using sharkjack.sh

This article describes managing payloads and loot with the desktop sharkjack.sh utility.

So you've gotten the basics down, tried out a few payloads, and now you're ready to take your Shark Jack game to the next level. One thing you may have noticed in getting your feet wet is the task of copying payloads to, and loot from the device. What's a hacker to do when something is done more than once? Script it, obviously. Enter: sharkjack.sh - now available for MacOS and Linux.

> Shark Jack Cable users may consider using the built-in commands, as described by running `HELP`, while connected to the Shark Jack in Arming Mode via Serial rather than the desktop sharkjack.sh utility.



The sharkjack.sh script is a pretty front-end that'll assist it not only loading payloads and getting loot off the Shark Jack, but it'll help you setup SSH keys so you can connect quickly - potentially without having to type

a password. Further, it'll check to see if your Shark Jack is up to date, and if not it can upgrade the firmware on your device automatically.

Let's get started. Begin by downloading `sharkjack.sh` from the tools section of downloads.hak5.org/shark. Then, open a terminal and navigate to the directory where you're keeping `sharkjack.sh`. I like to keep my scripts in my home directory, or `~`, so I can quickly get to 'em by typing '`cd`' and hitting enter. Next, make the script executable with '`chmod +x ./sharkjack.sh`' and run it as root with '`sudo ./sharkjack.sh`'

From the `sharkjack.sh` main menu, pressing C will connect via SSH to the Shark Jack. It'll wait for you to flip the devices switch to arming mode (center position) and plug it into your computer's Ethernet port. After authenticating with the Shark Jack, you'll have a '`root@shark:~#`' prompt.

If you want to make logging in even easier, pressing `S` a the main menu will copy your SSH public key to the Shark Jack - and if you haven't created SSH keys before, it'll guide you through the process.

The other functions – like upgrading the firmware, pushing payloads to the device, and getting loot saved on its disk work similarly.

So that's how to easily manage your Shark Jack using the sharkjack.sh helper script. We'd love to hear your thoughts on it in the forums, and you're welcome to contribute from the github repository. Cheers!

Two Key Commands

Whether on Windows, Mac or Linux – working with the Shark Jack is most convenient from the command line. Best of all, since modern versions of Windows ship with PowerShell, these work identically on all three platforms. In this article I'll show you two commands that'll make working with the Shark Jack a breeze, and how exactly they work.

Outside of the occasional firmware update, the two biggest functions you'll face when using your Shark Jack in arming mode – the devices management mode – are uploading payloads to the device and downloading loot (log files generated by payloads) from the device.

> ⓘ Shark Jack Cable users may consider managing the device via the dedicated Serial console rather than via SSH and SCP.

As you know from the official documentation, in arming mode the Shark Jack runs as a server – both a DHCP server, which will assign your computer an IP address on its network (a network of two, you and it) as well as an SSH server. The SSH server, or Secure Shell, lets you securely access the Shark Jack's command-line. When you 'ssh into' the Shark Jack, you'll get a bash shell on this tiny Linux box – from which you can manage the payload file in `/root/payload`, and the captured loot in `/root/loot`. But SSH has another function, and with it you may never need to drop into the Shark's bash shell.

SCP, or Secure Copy, works just like the cp command locally – except over the Secure Shell (SSH). Using it you can copy files to and from remote devices, just as you would locally using '`cp`' in Bash or PowerShell, or 'copy' in CMD. And with that, here are the two scp commands that'll make your Shark Jack life a breeze.

I'll show you from the Windows users perspective in PowerShell – but the same commands will hold true for the terminal on MacOS and Linux.

COPYING A PAYLOAD TO THE SHARK JACK

I like to keep an up to date copy of the Shark Jack payload repository on my computer – so I can try out the latest creations from the Hak5 community. In this example I'll show you how to copy the ipinfo payload, in the form of a shell script or payload.sh file, to the Shark Jack. It's on my hard disk in `C:\Users\bob\SharkJack\payloads\ipinfo`, so if I navigate there in PowerShell I can use the scp command to ferry that file over to the Shark Jack's payload folder - overwriting anything that may already exist there.

```
scp .\payload.sh root@172.16.24.1:payload/
```

The first part invokes the 'scp' command to securely copy the file. This command takes two parameters – from and to. In this case the first parameter, from, is the payload.sh file in this local working directory. In Windows PowerShell this is prefixed with `.\.` The next parameter, to, specifies where on the Shark Jack in the form of three elements: the user, the IP address, and the directory. In this case the user is root, the IP address of the Shark Jack is `172.16.24.1`, and the directory is `:payload/`.

A remote host with scp takes the form of `user@host:directory` – with `@` separating user and host, and `:` separating host and directory. If no directory is specified after the `:`, the default will be the user's home directory. In this case, the root user's home directory is `/root/` – so specifying `:payload/` is the same as specifying `:/root/payload/` (just with less typing).

Keep in mind this command is going to copy the local `payload.sh` file over to the Shark Jack in `/root/payload/`, overwriting any `payload.sh` file that's already there.

COPYING LOOT FROM THE SHARK JACK

Using the same method as above, we're going to reverse the from and to fields to recursively copy loot from the Shark Jack to the local computer.

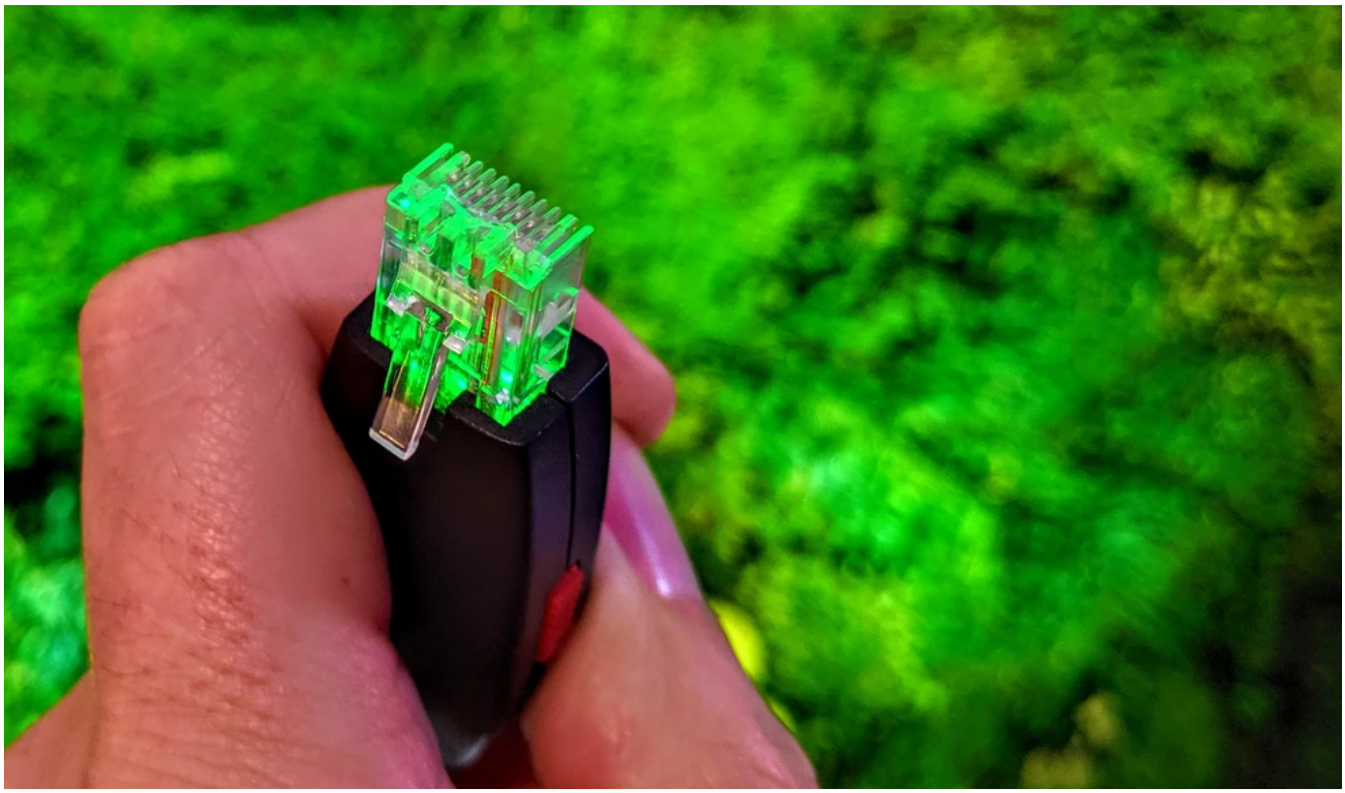
```
scp -r root@172.16.24.1:loot/ .
```

In this case the `-r` argument is specified so say to recursively copy the files. This means it'll copy files from all of the nested directories, since each payload saves loot to its own folder. The rest of the command is similar to the previous, only reversed. In this case the from field is the remote host – again in the form of `user@host:directory`. The to field is the current working directory, as represented by '`.`' – or it could be any path such as `c:\Users\bob\SharkJack\loot\`

So there you go, the two commands that make copying files – payloads and loot – to and from the Shark Jack a breeze. Now if you're looking for something a little more graphical, similar to Windows Explorer, you may want to check out WinSCP, FileZilla, or CyberDuck – all pretty graphical scp tools. Cheers!

Writing a Simple Payload

One of the simplest but most useful payloads you can rock on a Shark Jack is a simple port tester. With it you can tell at a glance from the multi-color LED whether a port is active, if it gets an IP address, and whether it has a connection to the Internet. In this article we'll write this basic yet powerful payload.



If you've worked in IT for a while you've come across this conundrum. Is this thing on? Without breaking out ye-olde-laptop, we're going to use the Shark Jack to test just this. Let's see how 5 simple lines of bash will give us instant feedback from the RGB LED.

Let's start out payload with the LED command. Even without perusing the official Shark Jack documentation you'll pick up how this command works just by example.

```
LED R SOLID
```

It pretty much writes itself. It's an RGB LED, and go figure the R parameter to the LED command tells it to light up Red. The second parameter, solid, said, huh, not to blink. The alternative would be SLOW, FAST or VERYFAST depending on how rapidly you'd want the LED to blink.

So in this state, the first thing the Shark Jack is going to do is make its LED red. Meanwhile, the framework is going to attempt to obtain an IP address from the target LAN via DHCP. What we'll want to do next is check to see if that's been successful – and if it has we'll change color. Otherwise, this Shark's staying red, and it'll be quite apparent when using that the port is a no-go.

```
while ! ifconfig eth0 | grep "inet addr"; do sleep 1; done
```

This little bash one-liner continuously checks the eth0 interface for the existence of the line "inet addr" – which is what you'll get when running the "ifconfig" command when your interface has an IP address. If it doesn't return any results, it'll "do" the command between "do" and "done", forever. That command? Sleep for one second, before checking again. The trick to this command is the exclamation point before the command – that's the magic that says "do this (sleep for a second) if it IS NOT true". Once the statement IS true (the grep for "inet addr" returns something), the command will be passed and our next command will run. Which in this case, will be:

```
LED Y SOLID
```

You can see where this is going. Once the Shark Jack gets an IP address, it's going to light – you guessed it – yellow. Our next command will use the same while loop logic as before to block the script from continuing – in this case until its able to download an HTTP web page.

```
while ! wget http://example.com -qO /dev/null; do sleep 1; done
```

I love example.com – don't you? It's always there for us. In this case, just as before, the script won't continue until it's able to complete this action. You can replace example.com with any HTTP site of your choosing – I just prefer to use the site that was reserved by the Internet Engineering Task Force back in 1999.

And finally, the command that has me quoting Dr. Raymond Stantz (played by Dan Aykroyd in 1984):

```
LED G SOLID
```

It may not be an Ecto-Containment System, but this payload will quickly answer the age old question – is this thing on?

```
1 #!/bin/bash
```

```
#
# Title:        Internet Access Tester
# Author:       Hak5Darren
# Version:      1.0
#
# Description:  This payload tests the port to see if the Shark Jack can
# obtain an IP address from DHCP, and if it can access the Internet by
# testing a specified HTTP URL.
#
# LED SETUP (Magenta)... Setting NETMODE to DHCP_CLIENT
# LED Red... No IP address from DHCP yet
# LED Yellow... Obtained IP address from DHCP, waiting on Internet access
# LED Green... Confirmed access to Internet

PUBLIC_TEST_URL="http://www.example.com"

LED SETUP
# Set NETMODE to DHCP_CLIENT for Shark Jack v1.1.0+
NETMODE DHCP_CLIENT

LED R SOLID
while ! ifconfig eth0 | grep "inet addr"; do sleep 1; done
LED Y SOLID
while ! wget $PUBLIC_TEST_URL -qO /dev/null; do sleep 1; done
LED G SOLID
```

And that's it. Find a prettied-up version ready to go. Love it? Chat with us on the forums. Want more for your Shark Jack? Nab some other sweet payloads. Cheers!

Software Updates

Manual Upgrade

Shark Jack firmware may be updated by using the sharkjack.sh helper script. That said, it is also possible to manually upgrade the firmware by following this process:

1. Download the latest UPDATE file from https://downloads.hak5.org/shark and verify its checksum.
2. Power on the Shark Jack in Arming Mode and connect it to a reliable USB power source
3. Copy the file upgrade file to the Shark Jack's /tmp directory via SCP (e.g. "`scp upgrade-1.1.0.bin root@172.16.24.1:/tmp/`")
4. SSH into the Shark Jack (e.g. "`ssh root@172.16.24.1`")
5. From the Shark Jack's bash prompt, issue the sysupgrade command relevant to your firmware update file (e.g. "`sysupgrade -n /tmp/upgrade-1.1.0.bin`")
6. Wait 5-10 minutes as the Shark Jack flashes the firmware and reboots.

> ⚠ DO NOT unplug the device from USB power during this process as doing so will render the device inoperable. This process will require 5-10 minutes. Powering off the Shark Jack at this time will result in damage to the device.

Over-the-Air Upgrade

Shark Jack Cable users may conveniently upgrade their device's firmware by running the `UPDATE_FIRMWARE` command.

1. Plug the Shark Jack Cable into a network which provides Internet access via DHCP
2. Flip the Shark Jack Cable switch to Arming Mode
3. Power the Shark Jack Cable via USB-C from a compatible computer or smartphone
4. From the computer or smartphone, access the Shark Jack shell via Serial
5. Press ENTER to activate the serial console
6. From the `root@shark:~#` prompt, type `UPDATE_FIRMWARE` and press ENTER

If the Shark Jack Cable has Internet access and an update is available, it will download and install automatically.

To cancel the installation, press `CTRL+C` during the 10 second countdown period.

> ⚠ DO NOT UNPLUG THE SHARK JACK CABLE DURING THE FIRMWARE INSTALL PROCESS. This process will require 5-10 minutes. Powering off the Shark Jack Cable at this time will result in damage to the device that may render it inoperable.

Writing Payloads

Payload Development Basics

Shark Jack payloads are written in Bash with Ducky Script and can be developed in any standard text editor, such as notepad or vi.

Payload files must be named either payload.sh or payload.txt.

Payloads should begin with the typical shebang /bin/bash.

```
1 #!/bin/bash
```

The NETMODE Command

`NETMODE` is a Ducky Script command which specifies which network mode to use in a given payload. These network modes determine how the Shark Jack will interface with the network.

> This command was added in Shark Jack firmware version 1.1.0 and is not available on Shark Jack 1.0.0 or 1.0.1. Shark Jack users, see Software Updates. Shark Jack Cable users, your device comes by default with firmware version 1.2.0 or higher.

NETMODE DHCP_CLIENT

In this mode, the Shark Jack will attempt to obtain an IP address from the target network via DHCP. This is the most common network mode.

```
1 NETMODE DHCP_CLIENT
```

NETMODE DHCP_SERVER

In this mode, the Shark Jack will run a DHCP server and offer an IP address to the host computer on its network in the 172.16.24.0/24 range, similar to Arming mode. This network mode enables attacks against individual computers via Ethernet rather than the network as a whole.

```
1 NETMODE DHCP_SERVER
```

NETMODE TRANSPARENT

In this mode, the Shark Jack will not offer or attempt to obtain an IP address from DHCP and may be used in conjunction with passive network sniffing programs when more stealth is desired.

```
1 NETMODE TRANSPARENT
```

The LED Command

The multi-color RGB LED status indicator on the Bash Bunny may be set using the LED command. It accepts either a combination of color and pattern, or a common payload state.

LED COLORS

COMMAND	Description
R	Red
G	Green
B	Blue
Y	Yellow (AKA as Amber)
C	Cyan (AKA Light Blue)
M	Magenta (AKA Violet or Purple)
W	White

LED PATTERNS

PATTERN	Description
SOLID	*Default* No blink. Used if pattern argument is ommitted
SLOW	Symmetric 1000ms ON, 1000ms OFF, repeating
FAST	Symmetric 100ms ON, 100ms OFF, repeating
VERYFAST	Symmetric 10ms ON, 10ms OFF, repeating
SINGLE	1 100ms blink(s) ON followed by 1 second OFF, repeating
DOUBLE	2 100ms blink(s) ON followed by 1 second OFF, repeating
TRIPLE	3 100ms blink(s) ON followed by 1 second OFF, repeating
QUAD	4 100ms blink(s) ON followed by 1 second OFF, repeating

QUIN	5 100ms blink(s) ON followed by 1 second OFF, repeating
ISINGLE	1 100ms blink(s) OFF followed by 1 second ON, repeating
IDOUBLE	2 100ms blink(s) OFF followed by 1 second ON, repeating
ITRIPLE	3 100ms blink(s) OFF followed by 1 second ON, repeating
IQUAD	4 100ms blink(s) OFF followed by 1 second ON, repeating
IQUIN	5 100ms blink(s) OFF followed by 1 second ON, repeating
SUCCESS	1000ms VERYFAST blink followed by SOLID
1-10000	Custom value in ms for continuous symmetric blinking

LED STATE

These standardized LED States may be used to indicate common payload status. The basic LED states include **SETUP**, **FAIL**, **ATTACK**, **CLEANUP** and **FINISH**. Payload developers are encouraged to use these common payload states. Additional states including multi-staged attack patterns are shown in the table below.

STATE	COLOR PATTERN	Description
SETUP	M SOLID	Magenta solid
FAIL	R SLOW	Red slow blink
FAIL1	R SLOW	Red slow blink
FAIL2	R FAST	Red fast blink
FAIL3	R VERYFAST	Red very fast blink
ATTACK	Y SINGLE	Yellow single blink
STAGE1	Y SINGLE	Yellow single blink
STAGE2	Y DOUBLE	Yellow double blink
STAGE3	Y TRIPLE	Yellow triple blink

STAGE4	Y QUAD	Yellow quadruple blink
STAGE5	Y QUIN	Yellow quintuple blink
SPECIAL	C ISINGLE	Cyan inverted single blink
SPECIAL1	C ISINGLE	Cyan inverted single blink
SPECIAL2	C IDOUBLE	Cyan inverted double blink
SPECIAL3	C ITRIPLE	Cyan inverted triple blink
SPECIAL4	C IQUAD	Cyan inverted quadriple blink
SPECIAL5	C IQUIN	Cyan inverted quintuple blink
CLEANUP	W FAST	White fast blink
FINISH	G SUCCESS	Green 1000ms VERYFAST blink followed by SOLID

EXAMPLES

```
LED Y SINGLE
```

```
LED M 500
```

```
LED SETUP
```

The SWITCH Command

`SWITCH` is a Ducky Script command which will indicate which position the Shark Jack's switch is in.

position	note
`switch1`	OFF/Charging
`switch2`	Arming Mode
`switch3`	Attack Mode

The BATTERY Command

`BATTERY` is a Ducky Script command which will output the current battery state.

State	Note
`discharging`	The battery is discharging
`full`	The battery is full
`charging`	The battery is charging

> The `BATTERY` command is specific to the original Shark Jack and is not applicable for the Shark Jack Cable variant as it does not include a built-in battery.

The SERIAL_WRITE Command

The `SERIAL_WRITE` command will write any following text to the serial console. This is useful for adding meaningful output to a payload.

> The ACTIVATE command was introduced with firmware 1.2.0 on the Shark Jack Cable.

Example

Add output to a payload with the following:

```
# Scan network
LED ATTACK
SERIAL_WRITE [*] Starting nmap scan...
nmap $NMAP_OPTIONS $SUBNET -oN $LOOT_DIR/nmap-scan_$COUNT.txt
```

Using Variables

The `SERIAL_WRITE` command will parse any variables, much like the `echo` command.

```
root@shark:~# DATE=$(date)
root@shark:~# SERIAL_WRITE $DATE
Tue Aug 24 00:26:55 UTC 2021
root@shark:~#
```

The Cloud C2 commands

The Shark Jack is Cloud C2 enabled — meaning it can be used remotely with the Hak5 Cloud C2 server to exfiltrate loot or be managed from the web interface or web shell.

To provision the Shark Jack for Cloud C2 server, see the guide on adding devices.

C2CONNECT

Unlike some Hak5 devices, such as the WiFi Pineapple, the connection to Cloud C2 is not automatic. First, the `C2CONNECT` command must be run, either interactively (Shark Jack Cable) or from the payload.

> ⚠ If the `C2CONNECT` command fails, check the `/tmp/cc-client-error.log` file for "Error posting update to server" entries, which may indicate that the system clock is out of date. Verify with the `date` command, and if necessary rectify this with an NTP update manually using the command `ntpd -q -p 1.openwrt.pool.ntp.org`

C2CONNECT

With a Cloud C2 connection established, loot may be exfiltrated using the `C2EXFIL` command.

```
root@shark:~# C2EXFIL STRING /tmp/cc-client-error.log "The Cloud C2 error log"
Starting C2 Exfil Tool
Loot sent Successfully
```

Included Tools

Tools Pre-Installed

From v1.0.x

- autossh
- nmap
- nc
- wget

- python

From v1.1.x

- arp-scan
- hping3
- macchanger
- ngrep
- nping
- p0f
- tcpdump

Installing Additional Tools

In order to install additional tools the Shark Jack will require an Internet connection in Arming Mode. Typically this is achieved by using the `NETMODE DHCP_CLIENT` command.

Connect to the Shark Jack shell by SSH (or Serial in the case of the Shark Jack Cable) and plug it into a local network. Ensure that it has Internet connection (e.g. `ping -c4 1.1.1.1`). If it does not, use the `NETMODE` command above to establish an Internet connection via DHCP.

Update the package manager with `opkg update`. Using the `opkg list` command you may find the name of the package you wish to install. Finally, install the package with the command `opkg install <name of package>`.

Example

```
root@shark:~# opkg install httping
Installing httping (2.5-1) to root...
Downloading http://downloads.openwrt.org/releases/18.06-SNAPSHOT/packages/mipsel_24kc/packa
Configuring httping.
root@shark:~# httping
No URL/host to ping given

root@shark:~# httping example.com
PING example.com:80 (/):
connected to 93.184.216.34:80 (43 bytes), seq=0 time= 27.99 ms
connected to 93.184.216.34:80 (357 bytes), seq=1 time= 18.41 ms
connected to 93.184.216.34:80 (351 bytes), seq=2 time= 17.20 ms
connected to 93.184.216.34:80 (43 bytes), seq=3 time= 17.45 ms
connected to 93.184.216.34:80 (43 bytes), seq=4 time= 17.35 ms
connected to 93.184.216.34:80 (43 bytes), seq=5 time= 17.39 ms
^CGot signal 2
--- http://example.com/ ping statistics ---
6 connects, 6 ok, 0.00% failed, time 5946ms
```

```
round-trip min/avg/max = 17.2/19.3/28.0 ms
root@shark:~#
```

Learn more about using the opkg package manager from the OpenWRT documentation at https://openwrt.org/docs/guide-user/additional-software/opkg

Managing Payloads

The UPDATE_PAYLOADS Command

The `UPDATE_PAYLOADS` command will synchronize the local payload library (stored in `/root/payload/library/`) with the repository online. This command is expected to be run from Arming Mode, and is intended for use with the Shark Jack Cable.

> The `UPDATE_PAYLOADS` command was introduced with firmware 1.2.0 on the Shark Jack Cable and requires an internet connection.

> In many cases an Internet connection may be enabled in Arming mode by issuing the command `NETMODE DHCP_CLIENT`.

Usage

```
root@shark:~# UPDATE_PAYLOADS
Downloading payloads repository...
Successfully syncronized payloads repository.
root@shark:~#
```

> The `UPDATE_PAYLOADS` command will make an Internet connection to github.com/hak5. Consider updating your payloads before connecting to the client site network on any physical engagement.

Troubleshooting

If you receive an Internet connection error like the following:

> You must have an internet connection to sync the payload libraries.

Check to ensure that the eth0 interface has an Internet connection. If the IP address is statically assigned to `172.16.24.1`, rather than an address on the target network via DHCP, run the `NETMODE DHCP_CLIENT` command and try `UPDATE_PAYLOADS` again.

The LIST Command

The `LIST` command, and its alias `LIST_PAYLOADS`, will list all of the payloads stored locally in the payload library at `/root/payload/library`.

> ⓘ The ACTIVATE command was introduced with firmware 1.2.0 on the Shark Jack Cable.

Usage

```
root@shark:~# LIST
Payloads
========

example
---------
    cloudc2-multi-file-exfiltration

recon
---------
    Nmap-C2
    SLAP-Nmap-to-Slack
    Sample-Nmap-Payload
    ipinfo
    netdiscover

util
---------
    Jack-Tester
    internet-access-tester
    mac-changer
    package-installer
    ping-tester
    ssh-ip-blinker

root@shark:~#
```

> ⓘ

If the payload library has not been synchronized with the online repository, run the UPDATE_PAYLOADS command.

The ACTIVATE Command

The ACTIVATE command, and its alias ACTIVATE_PAYLOAD, specifies a payload from the local library to set for use on next boot in attack mode. This command is expected to be run from Arming Mode, and is intended for use with the Shark Jack Cable.

The ACTIVATE command was introduced with firmware 1.2.0 on the Shark Jack Cable.

USAGE

```
/usr/bin/ACTIVATE [payload]
```

Since `ACTIVATE` is located in the `$PATH`, the absolute `/usr/bin/` directory may be omitted.

EXAMPLES

```
/usr/bin/ACTIVATE recon/nmap      (Use a payload inside the library)
/usr/bin/ACTIVATE /tmp/payload.sh (Use a specific file as the payload)
```

Troubleshooting

Firmware Recovery

The Shark Jack features a firmware recovery option which allows the user to restore the devices firmware image. This procedure is performed via a special web interface.

Download the latest firmware image for your Shark Jack from the Hak5 Download Center.

It is extremely important that you follow the directions precisely as it pertains to powering the device and

image selection from the web recovery interface. The video is provided as a reference however does not replace carefully reading the instructions listed below.

Follow these steps to access the recovery web interface and update the firmware.

- With the switch in the OFF position, plug in a suitable USB power source and fully charge the Shark Jack. The LED will blink blue while charging, and solid blue when fully charged. If no LED activity is present, leave the Shark Jack connected to the power source for 10 minutes.
- Unplug the Shark Jack completely from the USB power source
- Prepare to press the Shark Jack reset button located on the bottom of the device next to the regulatory label. Using a paperclip, SIM card removal tool or similar instrument practice pressing the button. With the Shark Jack unplugged and with its switch in the off position, carefully insert the instrument and directly downward until you feel resistance. Gently press the button. You should feel a click.
- With the instrument at the ready, flip the switch into the arming (middle) position and immediately after press and hold the reset button for 7 seconds.
- Connect a USB power source to the Shark Jack
- Connect the Shark Jack to your host PC Ethernet interface. After a moment the Shark Jack LED will indicate solid green with intermittent activity flashes.
- Set a static IP address for the host PC Ethernet interface connected to the Shark Jack as follows:
  - IP Address: 192.168.1.2
  - Netmask: 255.255.255.0
- From the host PC, browse to http://192.168.1.1
- A Shark Jack Recovery interface with a red banner will appear. Click to the Recovery tab, then click Browse Firmware, select the Shark Jack firmware downloaded from the Hak5 Download Center, then click Start Upload File.
-

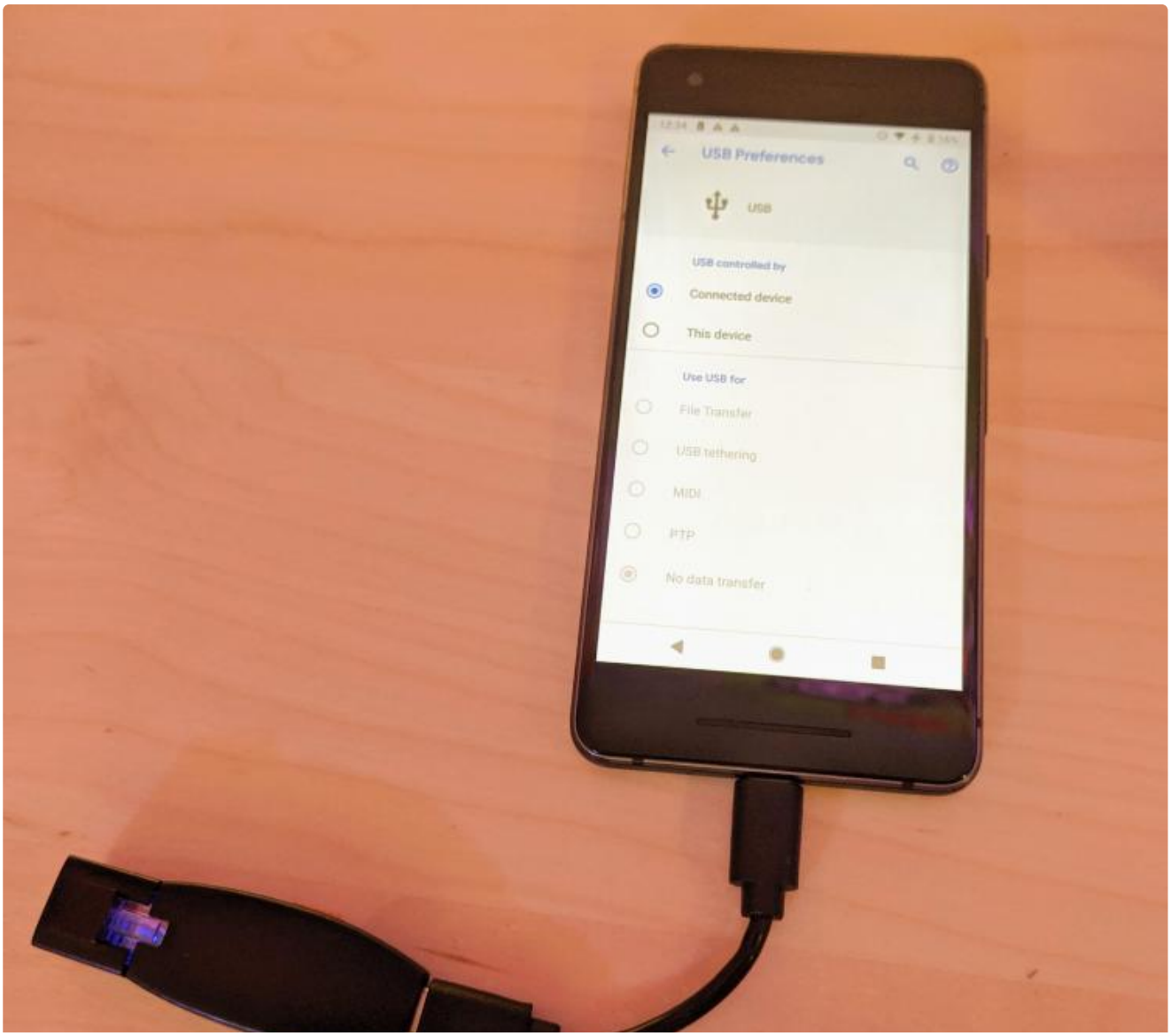
If your Shark Jack web interface shows a blue banner reading Web Failsafe Recovery, click the OS tab, then click browse, select the Shark Jack firmware downloaded previously, then click Start Upload File. If your Shark Jack features the blue bannered Web Failsafe Recovery interface, it is extremely important that you select the OS tab and not the Firmware tab or any other tab as doing so will render the device inoperable.

- This process will take several minutes. Do not interrupt the power supply while the firmware is updating. Once complete, the Shark Jack will restart as indicated by a green blinking LED. At this point, disable the static IP address on the host PC Ethernet interface connected to the Shark Jack and reset it to receive an IP address automatically via DHCP.

Tips & Tricks

Charge the Shark Jack from your Phone

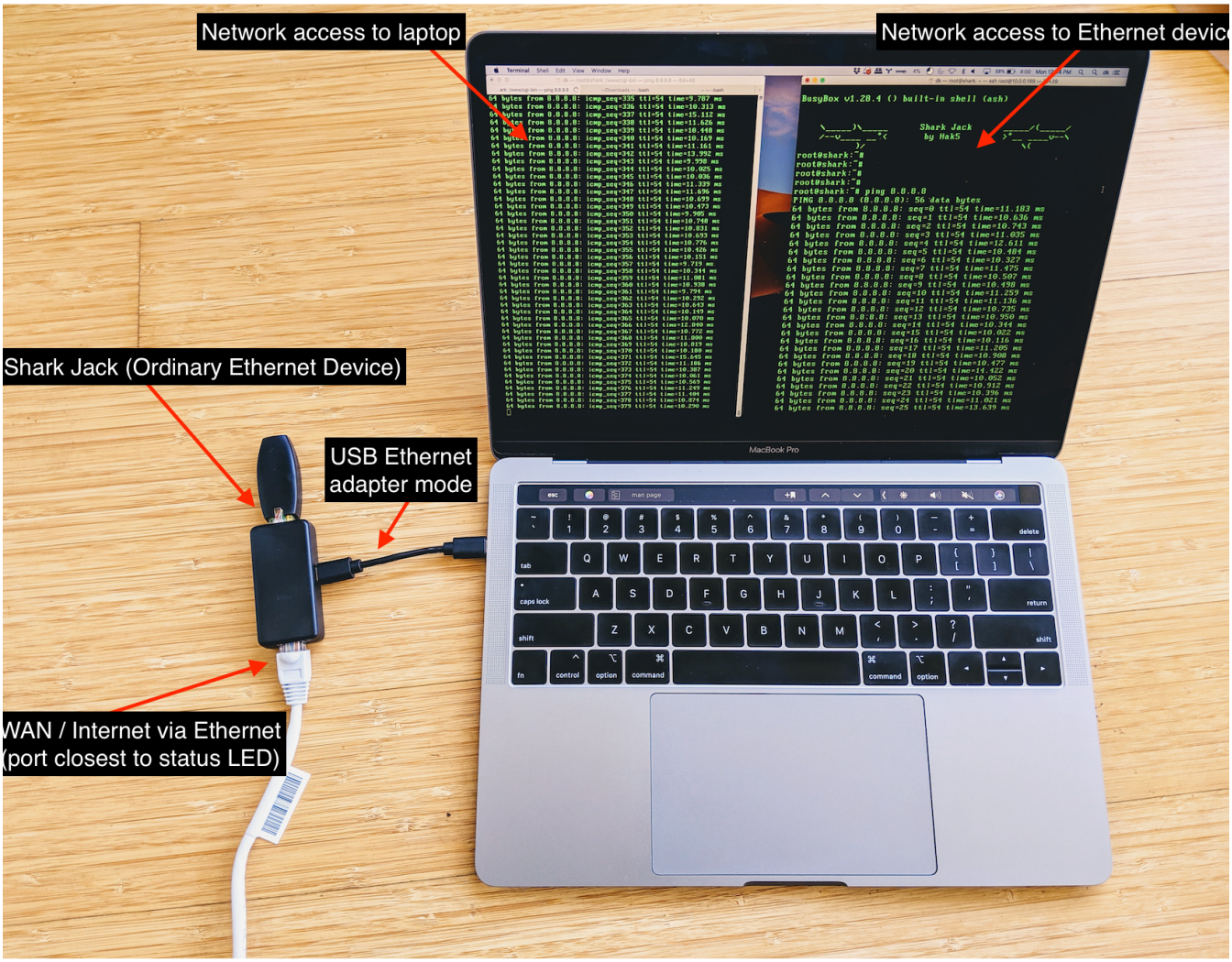
Many Android smartphones with USB-C interfaces support powering accessory devices. This can be useful in a pinch to charge your Shark Jack when a traditional USB power source or USB battery bank is not available. From the USB Preferences menu, select "Connected device" from the USB controlled by section.



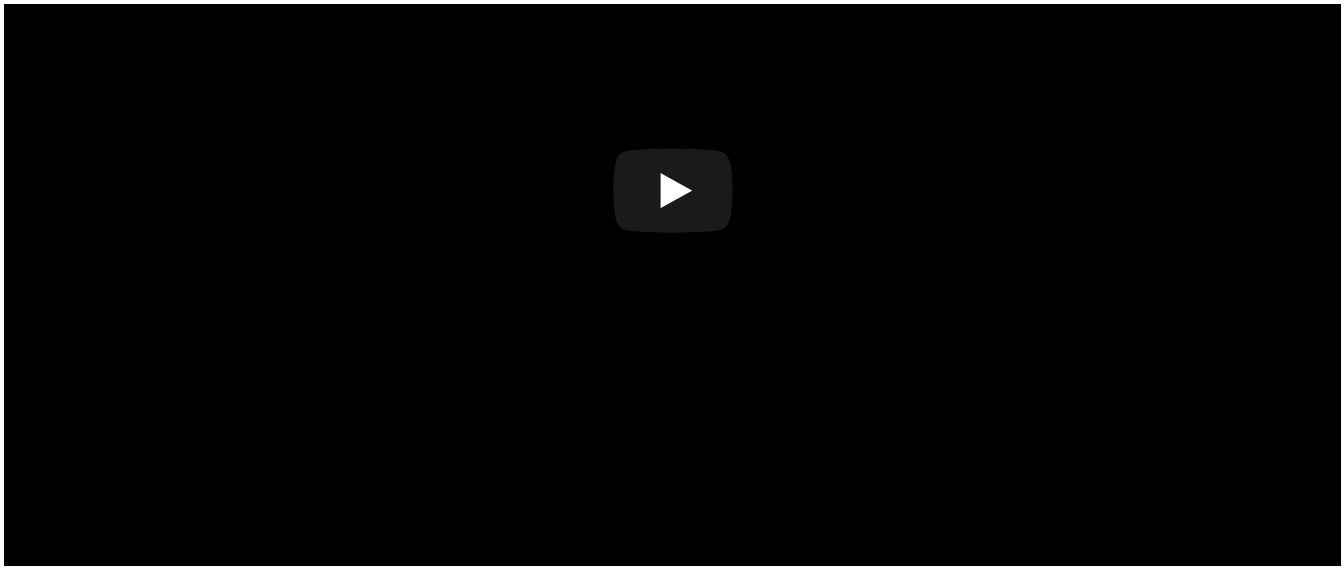
IMG_20191112_123419.jpg

Using the Shark Jack with the Plunder Bug as a simple switch

In a way, the Plunder Bug can be used as a simple switch. In the following example, the Plunder Bug is used to provide the laptop (via USB-C) and the Shark Jack (or any ordinary Ethernet device) network access via the WAN/Uplink port (closest to the Plunder Bug's status LED).



Android Serial Setup for Shark Jack Cable



Recommended apps:

- Serial USB Terminal
- Hacker's Keyboard

Product Information

Specifications

Shark Jack

SOC: MT7628DAN
INTERFACE: Ethernet
STANDARDS: 802.3
SIZE: 62 x 21 x 12 mm
POWER: 2.5W (USB 5V 0.5A)
BATTERY: 1S 401020 3.7V 50mAh 0.2W LiPo
OPERATING TEMPERATURE: 35ºC ~ 45ºC
STORAGE TEMPERATURE: -20ºC ~ 50ºC
RELATIVE HUMIDITY: 0% to 90% (noncondensing)

Shark Jack Cable

SOC: MT7628DAN
INTERFACE: Ethernet, USB UART (CP2102)
STANDARDS: 802.3
SIZE: 62 x 21 x 12 mm

POWER: 2.5W (USB 5V 0.5A)
OPERATING TEMPERATURE: 35ºC ~ 45ºC
STORAGE TEMPERATURE: -20ºC ~ 50ºC
RELATIVE HUMIDITY: 0% to 90% (noncondensing)

Important Safety Information and Warnings

The Shark Jack contains a Lithium-Polymer (LiPo) battery. Never leave near flammable material, liquids or in extreme temperatures as excessive heat, fire, damage and injury can occur. Never leave unattended while charging. Disconnect charger if battery becomes hot. Read all instructions before use.

Your device may get hot to the touch; this is normal. Unplug the device and let it cool before removing it. This device complies with applicable surface temperature standards and limits defined by the International Standard for Safety (IEC 60950-1). Still, sustained contact with warm surfaces for long periods of time may cause discomfort or injury. Keep the device in a well-ventilated area when in use. Allow for adequate air circulation under and around the device. Do not expose the device to water or extreme conditions (moisture, heat, cold, dust), as the device may malfunction or cease to work when exposed to such elements. Do not attempt to disassemble or repair the device yourself. Doing so voids the limited warranty and could harm you or the device. This device is not designed, manufactured or intended for use in hazardous environments requiring fail-safe performance in which the failure of the device could lead directly to death, personal injury, or severe physical or environmental damage.

Shark Jack is a trademark of Hak5 LLC. This product is packaged with a limited warranty, the acceptance of which is a condition of sale. See Hak5.org for additional warranty details and limitations. Availability and performance of certain features, services and applications are device and network dependent and may not be available in all areas; additional terms, conditions and/or charges may apply. All features, functionality and other product specifications are subject to change without notice or obligation. Hak5 LLC reserves the right to make changes to the products description in this document without notice. Hak5 LLC does not assume any liability that may occur due to the use or application of the product(s) described herein. Made in China. Designed in San Francisco by Hak5 LLC, 548 Market Street, #39371, San Francisco, CA, 94104. Hak5.org.

The Shark Jack is a network administration and pentesting tool for authorized auditing and security analysis purposes only where permitted subject local and international laws where applicable. Users are solely responsible for compliance with all laws of their locality. Hak5 LLC and affiliates claim no responsibility for unauthorized or unlawful use.