

LAN Turtle

Setup, Deployment and testing

Eric Mill

Contents

[Overview 1](#_Toc512420610)

[Turtle Setup 2](#_Toc512420611)

[Setting up the environment 2](#_Toc512420612)

[Setting up the lan turtle 2](#_Toc512420613)

[Modules Explained 3](#_Toc512420614)

[Use cases 4](#_Toc512420615)

[Covert Network Scanner (Man-In-The-Middle) Configuration 4](#_Toc512420616)

[Testing Scale 6](#_Toc512420617)

# Overview

On its surface the Local Area Network (LAN) Turtle is a covert, physical man-in-the-middle Penetration Testing tool used for providing remote network access and network intelligence gathering. It is housed in a standard USB to Ethernet adapter (Atheros AR9331 SoC at 400MHz MIPS) so as to not draw attention to itself. The LAN Turtle is extremely useful for Penetration Testers because it allows the tester to drop a shell onto any node on the network and then listen to or manipulate the data as they see fit. The Turtle shell is accessible over SSH (Secure Shell) meaning once the device has been configured and deployed it can be accessed from anywhere when used in conjunction with a Virtual Private Server (VPS). Because we can also move files from one computer to the shell and back using SSHFS (Secure Shell File Share), the Turtle Shell quickly becomes a very powerful tool when testing networks or vulnerable nodes.

# Turtle Setup

## Setting up the environment

If testing on a Windows machine the first step in setting up the LAN Turtle is to download and install a Secure Shell (SSH) terminal emulator as an SSH client is not natively installed on Windows machines. Here is a link to a secure download for the latest version of PuTTY, a free and open-source SSH client for Windows:

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

This is a critical step because we will be using this tool to connect to our LAN Turtle when setting up the device for the first time, and later using this same interface to connect to our Virtual Private Server (VPS) in the cloud using RSA keys. Once PuTTY has been installed there are many good step-by-step tutorials for getting the LAN Turtle up to the latest software and installing the appropriate modules.

Next, root access to a Virtual Private Server is needed in order to access the LAN Turtle from any location (even mobile devices). This can be purchased from many reputable online companies including Google and Amazon. For this project a VPS from Digital Ocean was used because they allow users to quickly and easily set up encrypted keys to connect to the server.

Digital Ocean offers step-by-step tutorials on how to set public-private keys for accessing the console of the VPS securely from PuTTY. On windows machines the standard PuTTY install will also include PuTTY GEN which is a public-private key generator that creates secure keys using entropy to ensure they are all unique. Here is a walk through guide on setting up the connection over PuTTY:

<https://www.digitalocean.com/community/tutorials/how-to-use-ssh-keys-with-putty-on-digitalocean-droplets-windows-users>

With these two pieces the tester can connect to the Turtle from any location and see the Turtle shell on the network it was placed on.

## Setting up the lan turtle

Simple video tutorials and a ton of great documentation on how to set up the LAN Turtle for first time use can be found at:

<https://www.hak5.org/gear/lan-turtle/docs>

For the purposes of this project the only modules needed are

1. autossh
2. keymanager
3. modulemanager <- Default module
4. netcat-revshell <- Optional
5. nmap-scan
6. sshfs
7. urlsnarf

The LAN Turtle’s main advantage is that it is a physical man-in-the-middle on the network that can see any nodes on the network it is plugged into. By leveraging its physical placement on the network with the power of the Turtle Shell, the LAN Turtle allows testers to record, manipulate, or covertly watch any node visible to the Turtle.

### Modules Explained

#### ModuleManager

By default this is the only installed mod on a new Turtle right out of the box. This handy module provides a simple and easy way to install, update, or delete additional modules from the LAN Turtle using a GUI over SSH. The ModuleManager should be used to install and update the additional six mods that are used and covered in this project. Using the ModuleMangaer is covered extensively on the Turtle Website.

#### AutoSSH

AutoSSH is a great module that keeps a persistent SSH tunnel to the VPS in the cloud. It does this by mapping the port 22 on the LAN Turtle (the standard SSH port) to a port we choose on our cloud VPS (usually 2222). By creating and maintaining this tunneled connection, AutoSSH allows testers to configure the module one time then enable it on Turtle start-up to ensure a connection can instantly be made to the Turtle Shell even when dropping it into a new configuration or network.

#### KeyManager

KeyManager is a very useful module for creating and sending new pubic-private encryption keys to our VPS all from a graphic interface. This module cannot be run, only configured, because all it does is generate new keys then send them over SSH to our VPS in the cloud. Using the KeyManager to create a connection to the VPS is integral to making sure a secure, encrypted connection is maintained at all times. Once the Turtle’s keys are copied over to the VPS the Turtle and VPS can communicate without having to exchange passwords (which would be impossible for a deployed Turtle that has no keyboard).

#### Netcat-RevShell

This is an optional lightweight shell that comes as a standard standalone install for most Linux distributions (the VPS we are using runs Ubuntu). This module makes it very easy to connect to the LAN Turtle without having to generate and exchange keys, but the downside is the connection is not encrypted and therefore should only be used when first setting up the Turtle-VPS connection or to debug any network issues.

#### NMap-Scan

Nmap-Scan is a network scanning module that generates a report fully mapping the network the LAN Turtle is plugged into including the IP addresses and open ports of machines in different nodes. The report can be saved to different locations or simply read through the Turtle Shell GUI. Using this module the tester can quickly look for network vulnerabilities or better places in the network to place the LAN Turtle.

#### SSHFS

Secure Shell File Share (SSHFS) is a module for extending the memory of the LAN Turtle (and all its modules) by creating a file system on our VPS that can be accessed through our secure tunnel on the Turtle (and any network it is plugged into). This is a very useful module for saving files collected on the Turtle to a secure offsite location through public-private key encryption. This module is also extremely useful when used in conjunction with the USB Rubber Ducky Tool, the script reporting the computer information can be generated by that tool and the corresponding report can be collected by the LAN Turtle.

#### URLSnarf

URLSnarf is a module for collecting network traffic in real time from the network node the LAN Turtle is plugged into. The module sniffs and logs all of the HTTP traffic that goes through the Turtle when it is connected in a man-in-the-middle configuration. When used in conjunction with the SSHFS the Turtle can covertly monitor network traffic for days logging the URLs in a standard, easy to filter format.

## 

## Use cases

### Covert Network Scanner (Man-In-The-Middle) Configuration

The ideal use case for the LAN Turtle is to have the Turtle plugged directly into a target computer with the only connection to Ethernet being through the Turtle itself to intercept all traffic from that computer to the network. In this configuration the Turtle is perfectly poised to assist the rest of the devices and tests in collecting information about vulnerabilities or security flaws from the business the Penetration Testers are working on. By keeping a persistent shell to the network through an offsite VPS the testers are able to monitor and assess the changes in the network and also collect information from other tests that may have not been able to complete. The USB Rubber Ducky generates a report in HTML format as well as sending results to our custom API in a JSON format. On Windows 7 and earlier operating systems the convert to json command is not native, so the report is only generated on the local computer as an HTML. The LAN Turtle connected to this computer can collect those reports and send them through the SSHFS to be analyzed by the testers at their convenience. As a physical man-in-the-middle the Turtle is perfectly positioned to map the network and probe open ports for vulnerabilities on nodes visible to the LAN Turtle. This configuration also allows the Turtle to collect all network traffic from the target network using the URLSnarf module. By collecting and analyzing network traffic, testers can see the general internet habits of any users connected to the internet through the Turtle. Analysis of this data would greatly improve the effectiveness of phishing attacks done by other testers or could provide good targets for DNS spoof attacks.

## Testing Scale

Starting at 10 we deduct points for each vulnerability found with the LAN Turtle and also for the amount of time the Turtle was connected to the network without being discovered or disconnected.

-0.5 For each day the Turtle is connected

-5 Having unrestricted Power Shell permissions

-3 Firewall turned off

-1 For each computer connected to an unsecure network

-1 For each 500 MB of raw data removed from the network

-1 If testers are able to retrieve and remove the Ducky report

-1 If testers are able to run the JSON report script to the API