

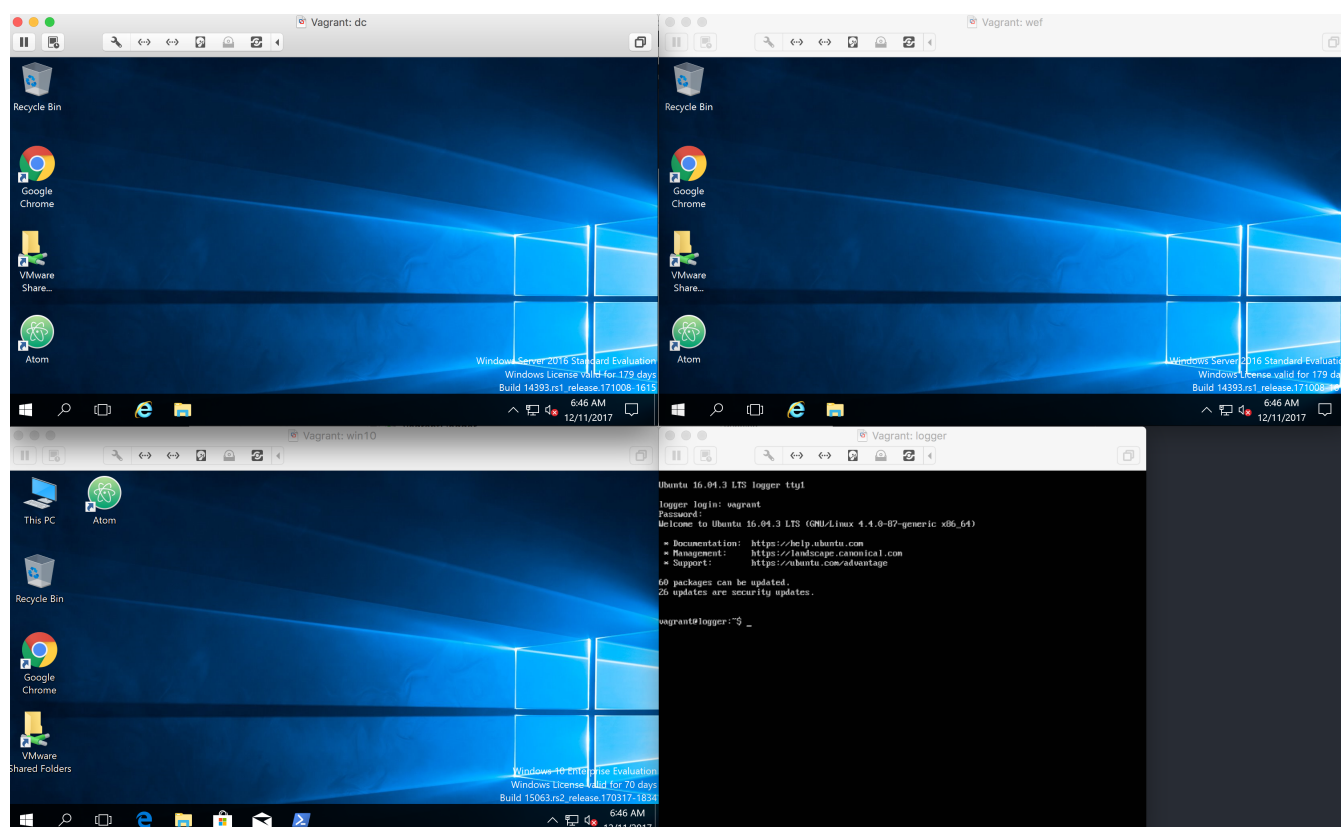
[Get started](#)

Introducing: Detection Lab



Chris Long

Dec 11, 2017 · 4 min read



<https://github.com/clong/DetectionLab>

Detection Lab is a collection of Packer and Vagrant scripts that allow you to quickly bring a Windows Active Directory online, complete with a collection of endpoint security tooling and logging best practices.

Detection Lab consists of 4 total hosts:

- DC: A Windows 2016 domain controller
- WEF: A Windows 2016 server that manages Windows Event Collection
- Win10: A Windows 10 host simulating a non-server endpoint

I started Detection Lab as a personal challenge to myself. I initially came across [Stefan Scherer's adfs2](#) repo which provided all of the building blocks I needed to set up Active Directory using Vagrant, and his [packer-windows](#) took the guesswork out of building Windows-based boxes.

I decided defenders needed a quick and easy way to bring up a lab environment, complete with tooling and pre-configured logging. This project represents many weekends worth of work over many months.

. . .

Detection Lab Tooling

The tooling included with Detection Lab is meant to maximize visibility and introspection for security practitioners. Detection Lab does NOT include any OS hardening and is insecure by design — do not run it on public networks!

The following security tooling is installed and pre-configured:

Logging Enhancements

- [Palantir's Windows Event Forwarding](#) subscriptions and custom channels are implemented



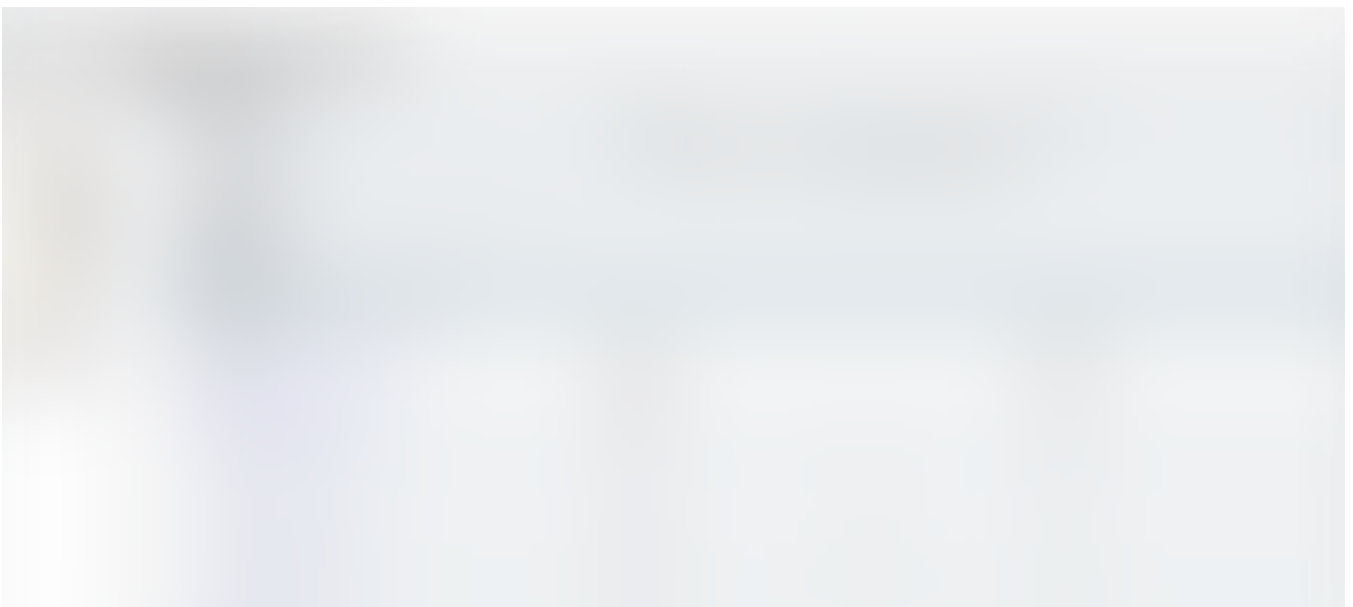
Windows Event Forwarding Subscriptions

- All autostart items are logged to Windows Event Logs via AutorunsToWinEventLog



Sample of AutorunToWinEventLog output in Splunk

- A custom Windows auditing configuration is set via GPO to include command line process auditing and additional OS-level logging





Sample Auditing GPO


- Powershell transcription logs from all hosts are saved to an SMB share on the WEF server
- SMBv1 Auditing is enabled
- Splunk is pre-configured on the Logger host to consume and parse all collected logs

Endpoint Security

- Palantir's open-sourced osquery configuration is applied and all osquery agents are connected to a Fleet server on the Logger host



Fleet osquery Ad-Hoc Query via TLS

- Sysmon is installed and configured using SwiftOnSecurity's open-sourced configuration
- 

- 
- A handful of tools exist in **C:\Tools** including ProcessExplorer, ProcessMonitor, TCPView, PsExec, etc...

. . .

Build Process

In order to build Detection Lab, you'll first build a Windows 2016 box and a Windows 10 box using Packer. Each box takes about 1 hour to build.

After the boxes have been built using Packer, Vagrant will be used to unpack the boxes, create a Windows domain, and install additional software on each host. Vagrant requires a virtual machine provider such as VirtualBox or VMWare to be installed in order to run. The first time the hosts are provisioned with Vagrant, it will take about 90 minutes for all 4 hosts to be configured. After the provisioning process is complete, bringing the lab online from a suspended state takes 2–3 minutes.

If you'd like to customize the software included in this lab, all you need to do is make changes to the Vagrantfile and associated scripts and bring the lab online again with Vagrant.

. . .

- A DFIR professional who wants to see what type of logging and alerting will be generated for a specific attack technique
- A bug bounty participant who needs to quickly spin up an Active Directory environment for testing purposes
- A red team member who wants to see what type of logs and forensic artifacts their tooling and methods will generate in a similar environment
- Someone who is looking for reference material to automate installation/configuration of security tooling
- Someone looking for a small staging environment to use when making changes to security tooling configurations

Customization

Detection Lab exists for you to customize it however you'd like. Don't like the tooling that's included? Feel free to pull tooling scripts out of the Vagrantfile. Have a custom Sysmon config you'd rather use? Throw it in! Just want a vanilla Active Directory? Go for it.

The tooling I've included and pre-configured is essentially just a technical preview of what's possible. Feel free to use this code however you'd like.

If you have problems getting it to work, find a bug, or want to contribute, please create an issue on the Github page!

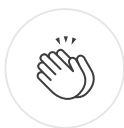
Dfir

Information Security

Automation

Hashicorp

Vagrant



1.1K claps





Security Engineer & Amateur Traveler

Follow

See responses (10)

Medium

About Help Legal