Welcome to OPNFV Security Guide

Release arno.2015.1.0 (b6b86d4)

OPNFV

February 12, 2016

This guide seeks to inform operators who to secure and maintain the security of the OPNFV Platform and its components.

Contents:

CHAPTER

ONE

INTRODUCTION

The OPNFV Security Guide is the collaborative work of many individuals, involved in both the OPNFV Security Group and the wider OPNFV community.

The purpose of this guide is to provide the best practice security guidelines for deploying the OPNFV platformm. It is a living document that is updated as new changes are merged into it's repository.

1.1 Background

Pre-virtualization security protection was largely centered on the network. Malicious attacks from hostile machines, would seek to exploit network based operating systems and applications, with the goal of compromising their target node.

Physical security had always been a much simpler business, with most focus on the secure access of the data center hardware. In-turn security was built up in layers (defense in depth) where machines would be daisy chained with network cables via security appliances to provide controlled segmentation and isolation. This form of security was built upon the principle of an 'air gap' being present, whereby machines were separate physical units, joined largely by the network stack.

With the advent of virtualization (namely the hypervisor), new attack vectors have surfaced as the 'air-gap' is no longer key design aspect for security. Further to this elements orchestation nodes and network controllers lead to an even wider attack surface:

- Guests breaking isolation of the hypervisor.
- Unauthorized access and control of supporting orchestration nodes.
- Unauthorized access and control of supporting overlay network control systems.

The hypervisor and the overlay network have now become the 'Achilles heel' whereby all tenant data isolation is enforced within the hypervisor and its abstraction of hardware and the virtualized overlay network.

This guide has been formulated, in order to assist users of the OPNFV platform in securing an Telco NFV / SDN environment.

1.2 Acknowledgements

CHAPTER

TWO

COMPUTE SECURITY

2.1 DAC & MAC Controls

2.2 Trusted Compute

Compute security relates to the compute nodes in an OPNFV deployment. Compute nodes host various componants such as the hypervisor itself KVM-QEMU, and its serving eco-system, such as Nova (which interacts with the hypervisor using libvirt driver).

We also cover other aspects of what is considered compute security, such as trusted boot / pools, although of course, these can be extended to other actors such as neutron etworking nodes.

CHAPTER

THREE

NETWORK SECURITY

3.1 Neutron Security

CHAPTER

FOUR

HOW TO CONTRIBUTE

Anyone is welcome to make additions, raise bugs, and fix issues within this Documentation. To do so, you will however need to first get an enviroment set up.

4.1 Development Environment

All project data such as formatting guidelines, and upstream mapping is documented via sphinx which uses reStructuredText

It is recommended that you use a python virtualenv to keep things clean and contained.

4.1.1 VirtualEnv

Use of a virtual environment is recommended, as not only is it a quick easy form of getting the needed modules in place, it isolates the module versions to a project.

From within your inspector directory, set up a new virtualenv:

```
virtualenv venv
```

Activate the new virtual environment:

```
source venv/bin/activate
```

Install requirements:

```
pip install -r requirements.txt
```

4.1.2 Sphinx Basics

To get started with sphinx, visit the main tutorial which will provide a primer *http://sphinx-doc.org/tutorial.html*

Hack your changes into opnfv-security-guide/source

To compile changes:

    make html

From here you can run a basic python web server or just navigate to the file:///<repo>/opnfv-security-guide/build/html/index.html in your browser