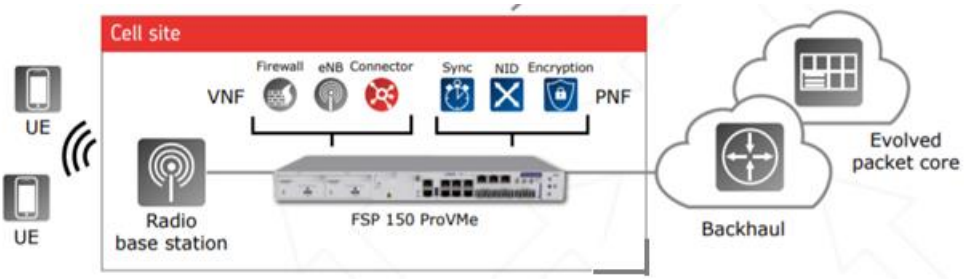
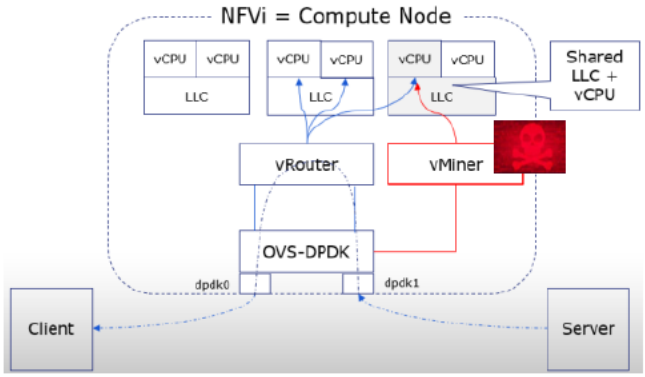
Malware Detection in NFV Edge Computing - using Machine Learning

Abstract:

Network Function Virtualization (NFV) is an emerging approach gaining popularity among network providers. NFV takes the physical networking devices commonly used today (switches, routers, load balancers, firewalls, antivirus, storage devices etc.) and virtualizes them in the cloud.
Edge computing provides these compute and storage resources with adequate networking connectivity close to the devices generating traffic. The benefit is the ability to provide new services with very low latency and avoid the data travel far in the network to reach the server in the cloud.
The ADVA FSP-150 proVMe is a Multi-layer demarcation device that is equipped with a compute blade, based on x86 architecture CPUs, for NFV hosting. It is located in the cloud edge at the customer premise or at the cell site.



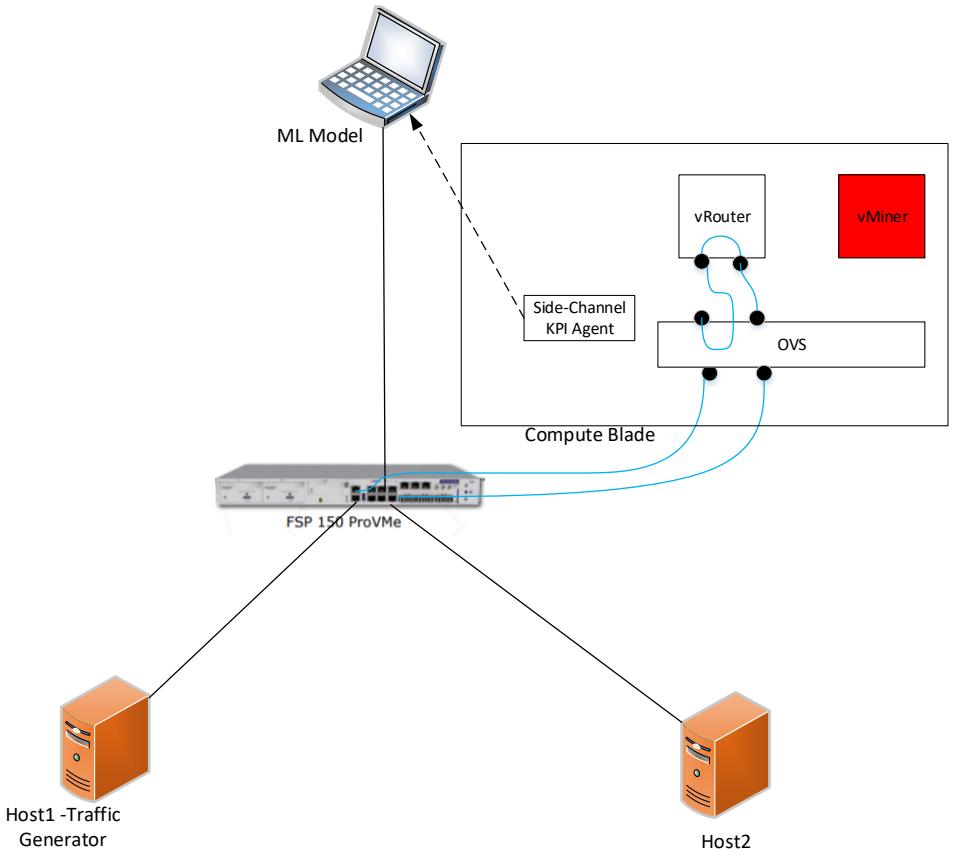
However, along with its flexibility, this approach inherits the vulnerabilities of CPU architecture.
It allows an attacker to obtain root privileges and to plant malware. Among such malware is crypto mining that is stealing CPU cycles from a legitimate NFV application. Such malware is hardly detectable either by malware scanner or by a firewall.



In this project, we will use Machine Learning tools to investigate the applicability of side-channels Key Performance Indicators (KPIs) needed for malware detection.

Goals:

1. Learn about NFV infrastructure and cloud edge computing.
2. Read related articles:
    a. Using Machine Learning to Detect Noisy Neighbors in 5G Networks
    https://arxiv.org/pdf/1610.07419.pdf
    b. Deep convolutional neural networks for detecting noisy neighbours in cloud infrastructure
    https://pdfs.semanticscholar.org/72f9/00b8d89312a392255c4dc9544376e6ad9145.pdf
3. Bring-up the environment that includes:
    a. Side-channels KPI agent - On the Compute Node of the FSP 150 ProVMe. It periodically collects a set of KPIs and stream these to a machine learning system.
    b. Machine learning system software using the TensorFlow™ framework.
    c. Use a pre-trained Supervised Learning Binary Classification model. The model should produce "Normal" or "Abnormal" predictions for each collected set.
    d. Virtual router VM on the Compute Node NFV platform
    e. Docker container performing Ethereum crypto currency mining (vMiner). The Ethereum miner uses the Elliptic Curve Digital Signature Algorithm (ECDSA) to validate the origin and integrity of messages, so it is a CPU-bound application.
    f. Traffic generator and a Host that will receive the routed traffic.



4. Generate traffic that should be routed via the vRouter and show that vMiner activity exhausts CPU resources and causes traffic drop.

5. Analyze - TBD

Requirements:

Introduction to Networking (236334 or 044334)

Python Programming

Guided by:

Andrew Sergeev (Adva)