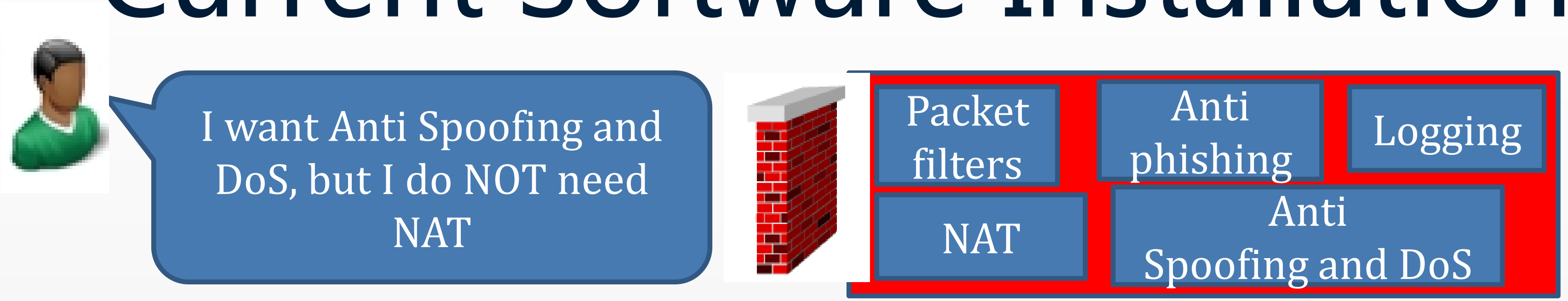
The Problems of Service Configuration in Network Function Virtualization(NFV)

http://tools.ietf.org/html/draft-song-appsawg-service-template-00

Haibin Song haibin.song@huawei.com

Current Software Installation



I want Anti Spoofing and DoS, but I do NOT need NAT

Packet filters

Anti phishing

Logging

NAT

Anti Spoofing and DoS

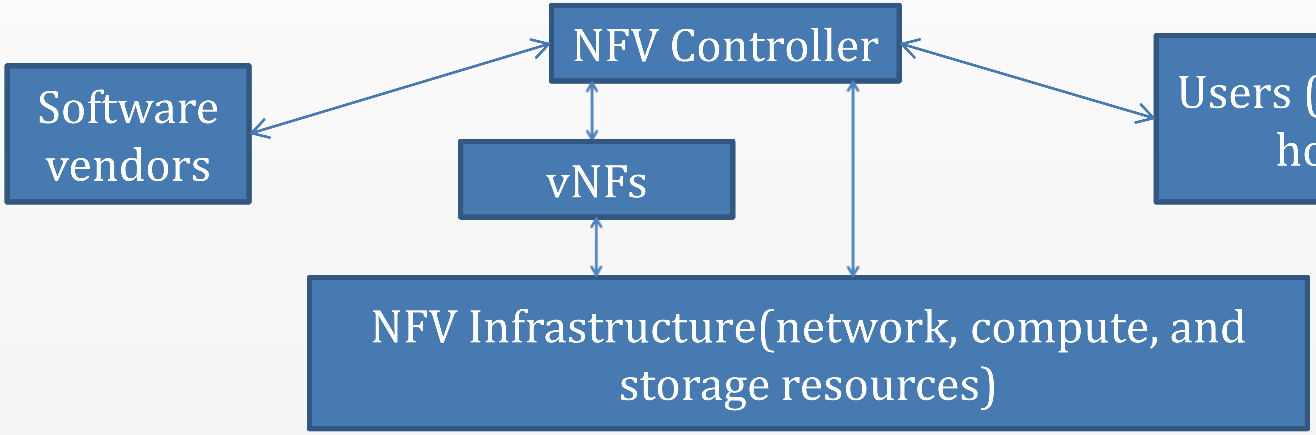
- Software are componentized, then map user requirements → software components
  - Divide users to classes, and provide several versions with different components, such like “home edition”
  - Specific version with only components that the user need through user and vendor communication
  - A license and software packet, that allows user to choose software components in the authorized range during installation
- *...Either too complex or not exactly match the user need*

Current Dynamic Service Configuration

- Log into the server, change the configuration online, then the result get effect immediately or after reboot

- *...It costs time when managing multiple replicas (the software being installed on multiple servers)*

Use Case in NFV Context



NFV Controller

Software vendors

vNFs

Users (enterprise, home…)

NFV Infrastructure(network, compute, and storage resources)

- NFV: adopt commodity servers and standard switches to replace the dedicated network middle boxes (such like firewalls, carrier grade NAT, Residential gateway …)

- NFV control centre becomes the broker for various software
  - Infrastructure resource scheduling
  - One software can be deployed in multiple VMs
  - Migrate VM when needed (e.g. energy saving)

- For NFV, please refer to http://portal.etsi.org/portal/server.pt/community/NFV/367

Problem Space - 1

- User installs the software remotely, and has no control over the infrastructure hardware & software resources
  - *It needs to coordinate with the NFV controller on software installation*
  - *It is a possible way to describe software and enabled components to NFV controller?*

Problem Space -2

- If every software has its own proprietary messages for software components installation, then it may get complex
  - Controller environment
  - User environment

Problem Space - 3

- Software vendor to NFV controller
  - If it does not provide a description of the function components, and what that component does, how can a user choose it?

Problem Space - 4

- Service dynamic configuration, e.g. change the ACL, firewall configuration …
  - SHOULD NOT make the user to configure multiple replicas on multiple VMs
  - NFV Controller can be a central point for dynamic configuration, when, where to apply
    - Aware or Agnostic of the service logic? (what)

Discussions

- Data model for the components description?
  - Tree based?
  - What language? XML/YANG/JSON?
- For the installation configuration
  - Only support “on”/“off” mode for software components might be enough?
  - Need other description of the installation requirement on hardware(i.e. cpu, memory, storage) and OS?
- For dynamic service configuration
  - More flexible than “on”/“off”?
    - User defined parameters? Sequences for applying rules?
    - Controller might be agnostic of the service configuration(a configuration file that does not have to look into)

Potential IETF Work

- Abstract: a centralized software installation and service configuration (*NFV is just a use case*)
  - A controller interface to software vendors, for components description
  - A controller interface to users, for software component choices
  - A controller interface to users, for dynamic service configuration

Thanks!