A new approach to Security

(Software Defined Secure Network)

Rakesh Kumar
Juniper Networks
Email: rkkumar@juniper.net

www.isocore.com/2016

NET-CENTRIC

JUNIPER NETWORKS

Agenda

- Why do we need a new approach ?
  - Traditional model (Perimeter Defense)
    - Challenges
- The new approach
  - Software Defined Secure Network (SDSN)
  - Defense-in-depth (Pervasive Defense)
  - A SDN approach
    - Decouple policy management from enforcement
    - Abstraction for expressing security policies
    - Automated policy lifecycle management

Traditional Model - Overview

Business View

Users & Roles

End-points

Applications

Data

Departments, Sites

External Threats

Business rules

Regulatory & Compliance rules

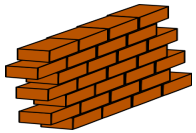
Fragmented Security Policy Framework

Operational View

Firewall Zones

Rules

Users & Apps



Switch Ports

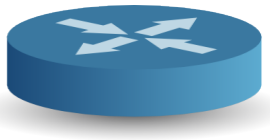
VLANs

ACLs

IPs/Subnets

VRFs

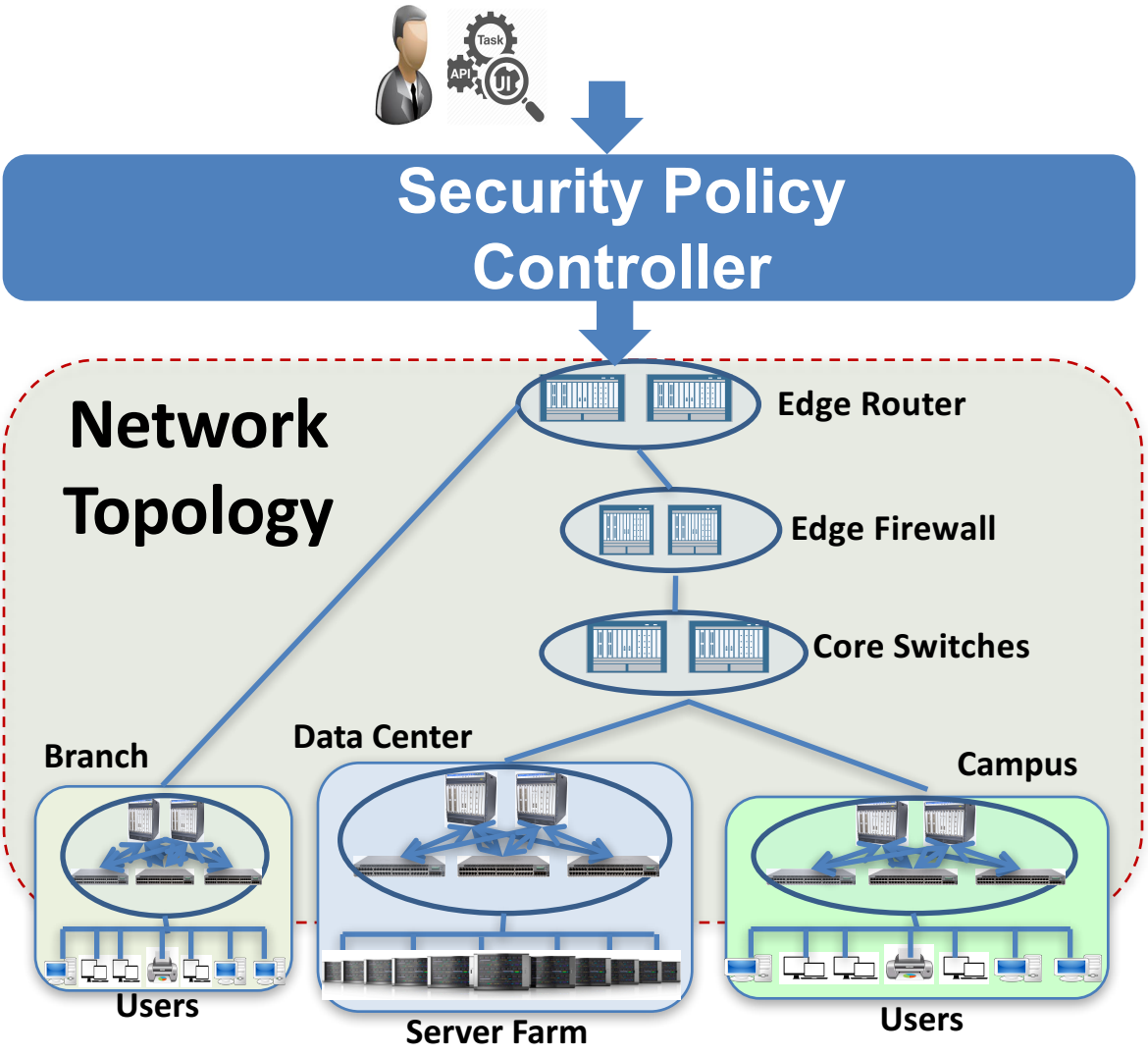
ACLs



Traditional Model - Challenges

- Multiple administrative domains
  - Network Security, Network Operation, IT
- Multiple provisioning systems
  - Networks, Firewalls, End-points
  - Requires complex and time consuming co-ordination
- Lack of user-intent for provisioning
  - Highly feature, vendor specific approach
- Lack of complete protection
  - East-to-West traffic lack protection
    - An infection can spread laterally in the network
- Lack of network participation in effective enforcement
  - Typically only firewalls used as policy enforcement points

SDSN Model - Overview



- Security Controller
  - A policy controller
- Northbound Interface
  - Security admin interface
  - Policy abstraction
  - Data model driven
- Southbound Interface
  - Security function interface
  - Vendor, Device, Feature agnostic
  - Data model driven
- Secure Network Fabric
  - Policy Enforcement Points

JUNIPER NETWORKS

SDSN Model – Building Blocks … (1/3)

- Security Policy Controller
  - Manages security policy framework through northbound interface
  - Breaks down policy into policy enforcement point (PEP) configuration
  - Manages Network Security Functions (NSF) as PEP
  - IETF I2NSF WG defining northbound and southbound interfaces
- Controller Northbound Interface
  - Enables Security Admin to express network-wide security policy
  - User-Intent based policy definition
    - Meta-data driven objects
    - YANG Data models
  - Agnostic of NSF's form factor and location in the network
- Controller Southbound Interface
  - Enables Controller to configure NSF for a given user-policy
  - Vendor and device agnostic Interface
    - Yang Data models

SDSN Model – Building Blocks … (2/3)

- Secure Network Fabric
  - Policy Enforcement Points (PEP) for user-policy
  - Composed of Network Security Functions (NSF)
    - Physical form factor (Router, Switch, Firewall)
    - Virtual form factor (VNF)
    - Service Function chains (SFC)
    - Statically provisioned NSF
    - Dynamically instantiated NSF as per policy requirement
  - Topology awareness
  - Vendor agnostic open interface

SDSN Model – Building Blocks … (3/3)

- User-Intent
  - Meta-data driven groups
    - User-group (e.g., HR-users, Finance-users)
    - Application-group (e.g., HR-apps, Finance-apps)
    - Device-group (e.g., Windows-machines, Linux-machines)
    - Location-group (e.g., US-region, EMEA-region)
  - Meta-data information sources
    - Active Directory
    - LDAP
    - CMDB

SDSN Model – Use cases

- Enterprise & Service provider use cases
  - Organizational needs
    - Business rules (e.g., social media access)
    - Regulatory and compliance rules (e.g., HIPPA, PCI-DSS)
  - Macro segmentation
    - Securing East-West traffic
  - Micro segmentation
    - Multitier application security
  - Advanced Threat Prevention
    - Malware management
    - Threat feed management

References

- IETF I2NSF WG
  - https://datatracker.ietf.org/wg/i2nsf/documents
- I2NSF drafts
  - https://datatracker.ietf.org/doc/draft-ietf-i2nsf-framework/
  - https://datatracker.ietf.org/doc/draft-ietf-i2nsf-problem-and-use-cases/
  - https://datatracker.ietf.org/doc/draft-kumar-i2nsf-client-facing-interface-req/
  - https://datatracker.ietf.org/doc/draft-kumar-i2nsf-controller-use-cases/