Source Address Validation Solution with OpenFlow/NOX Architecture

Guang Yao, Jun Bi and Peiyao Xiao
Network Architecture&IPv6 Lab
Tsinghua University
Background

• IETF SAVI WG
  – Formed in 2008
  – Anti-spoofing at Local link

• First Hop device
  – Source address binding based on address assignment snooping and filtering spoofing traffic at the first hop SAVI device
  – Can not work very well yet if the fist hop is not deployed
IP Spoofing within Single Device Scope

Legacy Switch/Router

Server A  
Trust  
Server B  
I’m B  
Accept connection  
Malicious operations...

Attacker C
Source Address Validation within Single Device Scope

Access Device: B is on another port, filter the packet
Scenario with Multiple Device

I'm B

Trust

Access Device: I have no idea about address B, so let it go

I'm B
VAVE: Source Address Validation for Multiple Device Scenario with OpenFlow/NOX Architecture

• Problem to solve:
  – Help each device recognize spoofing flows of address assigned on other devices

• Basic mechanism:
  – 1. OpenFlow/NOX architecture
    • The *de facto* network protocol innovation framework
  – 2. Flow path calculation
    • To determine whether a flow is valid or not based on calculated path
Architecture of VAVE

Hit Filtering Rule, modify flow table for filtering

Not match flow table entry

Spoofing Flow (S=A,D=F)
Filter Generator: Principle

• Principle:
  – P1. Filtering on the VAVE edge
    • To filter as early as possible
  – P2. Filtering based on the flow path
    • To filter flows violate calculated path

• P1+P2->Filtering flows whose complete path is inside the VAVE on the edge
Filter Generator: Flow Path Calculation

• Origin discovery:
  – Snooping address allocation protocol (DHCP, SLAAC...) on OpenFlow devices.

• Flow path calculation:
  – OpenFlow device forwards packet based on FlowTable, which is configured by Controller.
  – Keep the FlowTable state on the NOX server. Calculate flow path based on FlowTable.
Filter Generator: Flow Path Calculation

Rule Set: \((A,B,C,D,E,F,G)\)*(A,B,C,D,E,F,G)
Filter Generator: Re-Calculation on Path Change

• Flow path may change frequently for a lot of requirements: dynamic load balance, route changes, etc.

• Calculate filtering rule frequently on each change will introduce heavy cost.

• Solution:
  – Set a flag for flows whose path is affected by FlowTable change
  – Re-calculate the path then change flow table for filtering, ONLY WHEN the 1st spoofing packet is detected on the edge – Filtering on Demand
Validation Module

- Validation module processes packet redirected on the edge

1. \((A,F)\) is in \((A,B,C,D,E,F,G)^* (A,B,C,D,E,F,G)\)
2. Re-calculate the path of \((A,F)\)
3. Confirm the spoofing
Filter Adapter

- Download filtering table to cut the spoofing flow by using the maximum cover rule

HitRule: Deny (X11/P-1,Y01/P-1)
Filter Adapter

- Data plane: rules are configured on the first flow table on device.
- Rules are configured with a timer to cancel filtering after a period.
- A rule is removed if corresponding flow path changes.
Evaluation

• An inferred topology (RF3755) from rocketfuel project through simulation.
  – Randomly assigned addresses and Shortest Path First algorithm to generate the flow table on each device.
  – Minimal vertex coverage strategy is chosen to generate the set of nodes in VAVE perimeter. There can exist multiple perimeters.
Evaluation

• Protectiveness:
  – The flows in the VAVE (SAVI switch to SAVI switch traffic) area are harder to be forged by attackers outside the edge
Evaluation

• Cost
  – The average check time on flows is reduced by the filtering-on-demand feature of VAVE
Evaluation

- Cost:
  - The required flow entries on the edge is also reduced greatly by filtering on demand.
Evaluation: Implementation

• Cost:
  – The time required by VAVE module on each packet is trivial (2.66GHZ CPU, 4GB Memory)
Conclusion

• We analyzes the problem of SAVI at multiple devices scenario
• We propose VAVE, which is based on OpenFlow/NOX architecture and calculated paths to enhance the filtering ability of each device.
• VAVE performs filtering at low cost by introducing filtering on demand.