Policy Based Security Management System for NFV-SDN and IoT Scenarios

ETSI Security Week - 12-16 June 2017

Antonio Skarmeta
University of Murcia
Outline

- Introduction
- Anastacia Architecture
- Policy-based Network and Security management
- Policy enforcement in SDN/NFV and IoT
- Scenarios
  - IoT Smart Building
  - MEC
- Ongoing Work
Introduction

• ANASTACIA goal
  – develop a trustworthy-by-design security framework for Cyber Physical Systems (CPS) based on IoT
  – to take autonomous decisions through the use of new networking technologies such as Software Defined Networking (SDN) and Network Function Virtualisation (NFV)
  – intelligent and dynamic security enforcement and monitoring methodologies and tools.

• The cyber-security framework provides self-protection, self-healing and self-repair capabilities through novel enablers and components

• The framework dynamically orchestrates and deploys security policies and actions that can be instantiated on local agents

• Compliant with SDN/NFV standards as specified by ETSI NFV-MANO
Security Policy Language

- In Anastacia the security is driven and managed by security policies
- High-level Security Policy Language (HSPL) and the Medium-level Security Policy Language (MSPL)
  - HSPL: The policy language suitable for expressing the general protection requirements of typical non-technical end-users.
  - MSPL: Expresses specific configurations by technically-savvy users in a device-independent format.
- HSPL is refined to MSPL using capability concept, which denotes any kind of security functionality that can be provided by a security application.
  - IETF standardization, Internet-draft (“Information Model of NSFs Capabilities”), I2NSF Working Group
- SECURED EU FP7 Project
Security Policy Language

- **Supported Policies**
  - Authentication
  - Authorization
  - Filtering/Forwarding
  - Channel Protection (e.g. IpSec)
  - General Security

- **Current Plugins Approaches**
  - Anonymity
  - Antiphising
  - Brologging
  - Bromalware
  - Dansguardian
  - IPTables
  - ReconduceBandwidth
  - Reencryption
  - Squid
  - Strongswan
Policy Enforcement in NFV/SDN and IoT

- **Policy Enforcement and orchestration**
  - Process managed by the ANASTACIA’s **Security Orchestrator** component
  - mapping source-code metadata security properties (provided as output by the Security Policy Interpreter) into appropriate SDN/NFV configuration.
  - To enforce these security properties, it interacts with:
    - SDN controllers
    - NFV MANO stack modules,
    - IoT controllers
  - **dynamic service chaining to cope with suspicious traffic**
  - **deploy local agents that will enforce security over devices – gateways and IoT devices**
Security Orchestrator

• **Security as a Service - SDN as a key enabler**
  – Configuration of security functions
    • Traffic duplication for Intrusion Detection Systems, Deep Packet Inspection
  – Scaling out
    • Flow based load balancing
  – Network security response
    • Traffic redirection
    • Traffic selection
    • Bandwidth restriction
SDN-BASED CONFIGURATION OF SECURITY APPLIANCES

1. Provisioning of security appliance
2. Mirroring traffic
3. Flow Mirroring Rule
4. Duplicated Traffic

Security Orchestrator

Client

SDN Controller

NFV-MANO

SDN switch (OVS)

vIDS

IoT node
SDN-BASED SECURITY COUNTERMEASURES

Security Orchestrator

1. Security Orchestrator
2. Reaction Module
3. Monitoring Module
4. IDS
5. Agent
6. Security Alert
7. Security Reaction
8. Stop Malicious Traffic
9. (9) Stop Malicious Traffic
10. (10) Drop Flow Rule
11. (11) Malicious traffic stopped at SDN switch

Client

SDN switch

IoT node

(5) Client starts generating malicious traffic
Virtual Security Functions

- **vFirewall** (e.g. netfilter Linux iptables)
  - Filtering rules
- **vRouter** (e.g. OpenVirtualSwitch in OpenWRT)
  - Traffic forwarding, traffic mirroring, traffic diversion ...
- **vChannelProtection**
  - Level3 encryption IPSec, SSL e.g. openVPN, DTLs
- **vIDS** (Intrusion Detection System, e.g. Snort)
- **vAAA** (e.g. Free Radius)
# Adopted Tools

<table>
<thead>
<tr>
<th>Module</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFV Orchestrator</td>
<td>OpenSourceMano (OSM)</td>
</tr>
<tr>
<td>VIM</td>
<td>OpenStack</td>
</tr>
<tr>
<td>SDN Controller</td>
<td>ONOS, OpenDayLight</td>
</tr>
<tr>
<td>Security Policy Models</td>
<td>SECURED EU Project languages,</td>
</tr>
<tr>
<td>Monitoring</td>
<td>XL-SIEM (ATOS){snort, nagios…}</td>
</tr>
<tr>
<td>Reaction</td>
<td>OpenC2</td>
</tr>
<tr>
<td>Alerts</td>
<td>IDEM EF</td>
</tr>
<tr>
<td>IoT management</td>
<td>CoAP, MQTT</td>
</tr>
</tbody>
</table>
Smart Building Scenario

• ANASTACIA framework provides tools to increase the resilience of Building Management System (BMS) upon cyber-attacks, such as:
  — Man-in-the-middle attacks, in which the attacker manipulates sensors
  — DoS → Mirai malware, affects IoT devices used as botnets for DDoS attacks

• Integration of SDNs in IoT networks
  — e.g. 6LowPANs and WiFi networks managed through OpenFlow

• Interoperability challenge
  — Diverse kind of IoT devices and networks (802.11, 802.15.4, 802.3)

• NFV in the building
Smart Building Scenario, example

- **Man-in-the-middle attack** → attacker manipulates sensors introducing wrong temperature values

- **First prototype**
  - allows to isolate devices from the Building SDN network using HSPL/MSPL security policies and a OpenDayLight SDN

- **ANASTACIA framework can detect** the uncommon temperatures (e.g. using machine learning)
  - **react and enforce security policy to isolate** the compromised sensor from the rest of the BMS system
  - **react** improving the security between certain IoT devices or within devices in some networks, enforcing a security policy for channel protection.
Multi-access Edge Computing (MEC) Scenario

• Multi-access Edge Computing (MEC) allows edge nodes to host VNFs and third-party applications near the end-users
  – meeting the desired Quality of Service (QoS) requirements

• Advantages of moving security at the edge
  – Instant response
  – Reduced network traffic
  – Confidentiality of information

• Challenges of edge security
  – Resource-constraint edge nodes (up to access points and IoT gateways)
  – Distributed environment

• Fist prototype: usage virtual security appliances and SDN networking to provide enhanced security in the Edge
  – Security Docker Containers deployed in Raspberry-Pi
Ongoing work

➢ Definition of security enablers
  – Define security threats (e.g., DoS attacks, intrusion, etc.)
  – Define attacks, detection techniques, security countermeasures (e.g., filtering, encryption, etc.), features, and interfaces

➢ Integration with Legacy IoT networks:
  – enforcement of security policies in scenarios where the support of SDN/NFV technologies is limited or, even, absent.

➢ Security Policy enforcement
  – To implement new plugins to provide new specific functionality for the ANASTACIA scope
    • Provides configuration/task to apply policies through SDN, IoT, NFV
Policy Based Security Management System for NFV-SDN and IoT Scenarios

ETSI Security Week - 12-16 June 2017

Antonio Skarmeta
University of Murcia
HSPL user isolation example

- This is an example for user isolation using a High Security Policy Language in order to achieve the following goal:

  “Alice is not authorized to access Internet traffic”

- The corresponding on HSPL could be:

  ```xml
  <tns:hspl subject="Alice" id="HSPL0">
    <tns:action>no_authorise_access</tns:action>
    <tns:objectH>Internet_traffic</tns:objectH>
  </tns:hspl>
  ```
HSPL → MSPL Refinement Example

User isolation example

- Refinement from HSPL to MSPL:

  ```xml
tns:hspl subject="Alice" id="HSPL0">
    <tns:action>no_authorise_access</tns:action>
    <tns:objectH>Internet_traffic</tns:objectH>
  </tns:hspl>

  <configurationRule>
    <configurationRuleAction xsi:type="FilteringAction">
      <FilteringActionType>DENY</FilteringActionType>
    </configurationRuleAction>
    <configurationCondition xsi:type="FilteringConfigurationCondition">
      <packetFilterCondition>
        <SourceAddress>ALICE_ADDR</SourceAddress>
      </packetFilterCondition>
    </configurationCondition>
  </configurationRule>
```
MSPL Refinement Example I

MSPL refinement

- MSPL policy is refined onto real configuration:
- Based on plugins

```xml
<configurationRule>
    <configurationRuleAction xsi:type="FilteringAction">
        <FilteringActionType>DENY</FilteringActionType>
    </configurationRuleAction>
    <configurationCondition xsi:type="FilteringConfigurationCondition">
        <packetFilterCondition>
            <SourceAddress>ALICE_ADDR</SourceAddress>
        </packetFilterCondition>
    </configurationCondition>
</configurationRule>
```
Generates the configuration from the MSPL for VNFs

- Different Network Security Functions plugins
- vFirewall, E.g. IPTABLES plugin:

```plaintext
*filter
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
-A FORWARD -p TCP -s ALICE_ADDR -j DROP
-A FORWARD -p UDP -s ALICE_ADDR -j DROP
COMMIT
```
MSPL Refinement Example III

» MSPL to SDN

- ODL SDN plugin:

```xml
<flow xmlns="urn:opendaylight:flow:inventory">
  <flow-name>Dropper</flow-name>
  <match>
    <ethernet-match>
      <type>0x0800</type>
    </ethernet-match>
    <ipv4-source>ALICE_ADDR</ipv4-source>
  </match>
  <id>Dropper</id>
  <table_id>0</table_id>
  <apply-actions>
    <action>
      <drop-action/>
    </action>
  </apply-actions>
</flow>
```
Seal Manager

S&P Seal Model Analysis

Dynamic Security and Privacy Seal User Interface

User plane: interfaces and tools

Policy definition

Set-up new policy

Monitoring and reaction Plane

Monitoring Service

Attack Signatures

Data Analysis

Data Correlation

Attack Patterns

Analysis Parameters

Filtered and classified data

Verdicts

Security Model Analysis

Verdict and Decision Support System

Security Alert Service

Mitigation Action Service

Security Orchestrator

Security Orchestrator

Verdicts, warnings, and vulnerabilities

Alerts, warnings, and vulnerabilities

Reactions

Security Enablers Manager

Security policies repository

Interpreter

High to Medium

Medium to Low

NFV MANO

NFV Orchestrator

VNF Manager

Virtualized Infrastructure Manager

Virtualized Infrastructure Domain

Storage

Compute

Network

Virtualization Layer

VNF #1

VNF #2

VNF #3

Control and Management Domain

IoT Controller

SDN Controller

Agents configuration

Agents

IoT domain

VNF Domain

NFV Manager

Virtualized Infrastructure Manager

Monitoring data/Network traffic

Attack Patterns

Security Models

Configurations and reconfigurations at runtime

Monitoring Service

Data Analysis

Data Correlation

Attack Patterns

Analysis Parameters

Filtered and classified data

Verdicts

Security Model Analysis

Verdict and Decision Support System

Security Alert Service

Mitigation Action Service

Security Orchestrator

Verdicts, warnings, and vulnerabilities

Alerts, warnings, and vulnerabilities

Reactions

Security Enablers Manager

Security policies repository

Interpreter

High to Medium

Medium to Low

NFV MANO

NFV Orchestrator

VNF Manager

Virtualized Infrastructure Manager

Virtualized Infrastructure Domain

Storage

Compute

Network

Virtualization Layer

VNF #1

VNF #2

VNF #3

Control and Management Domain

IoT Controller

SDN Controller
HSPL Policy Refinement Workflow
MSPL Policy Refinement Workflow
Policy Enforcement
Security-as-a-Service at the Network Edge

- Design a framework to support security services at the edge:
  - Based on Docker containers
  - Lightweight security software images
  - Container-based security management
MEC Scenario

SECurity-as-a-service (SECaaS) in remote cloud environment

SECaaS at the network edge

Can not be handled by the Edge

Virtualized security functions

Industry 4.0

Edge Node

ANASTACIA Project G.A. N° 731558