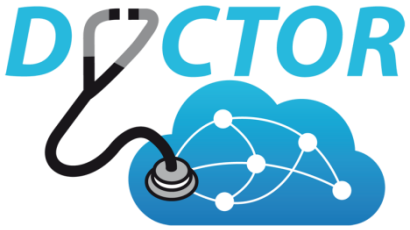


Systematic
Paris Region Systems & ICT Cluster

Challenges and directions for the security management of ICN services

Bertrand Mathieu, Guillaume Doyen, Wissam Mallouli, Thomas Silverston, Olivier Bettan, François-Xavier Aguessy, Thibault Cholez, Jérôme François, Alain Ploix, Patrick Truong, Edgardo Montes de Oca

orange™

THALES

montimage

Loria
Laboratoire lorrain de recherche en informatique et ses applications

utt
université de technologie Troyes

Outline

- Context and motivation
- NFV architecture to deploy NDN services
- NFV/NDN Testbed
- Firewall service for CCN
- Conclusion and future work

Problem statement

- Deploying new network equipment is costly
  - Specific hardware, specific usage, proprietary
  - Legacy and existing environment to take into consideration
  - Operational people to train
  - Network operators only deploy when sure of success
- Deployment only if secure and manageable
  - Network operators focus on network availability and endusers Quality of Experience (satisfaction)
  - Failure or unavailability costs a lot
  - Detection of attacks and ensures user privacy data
  - Monitoring of traffic and usage

ICN Management Considerations

- We need network management for ICNs
  - By definition: activities, methods, procedures, and tools that pertain to the operation, administration, maintenance, and provisioning of networked systems
  - NMRG functional areas : Fault management, Performance management, **Security management**, Configuration management, Accounting management, Service management, Event management

- Configure and control a set of resources that ensure the network is running well -> pre-requisite to any large scale deployment

Context

- Cost Reduction, Hardware Mutualisation, Energy Consumption
  - Network Function Virtualization
  - Software Defined Networking

- New disruptive architecture & solutions (ICN) for better data delivery and optimal use of network resources
  - NDN : Named Data Networking

- Our approach: deployment of new network functions and protocols (NDN) in a virtualized networking environment with advanced management features

Technical Locks

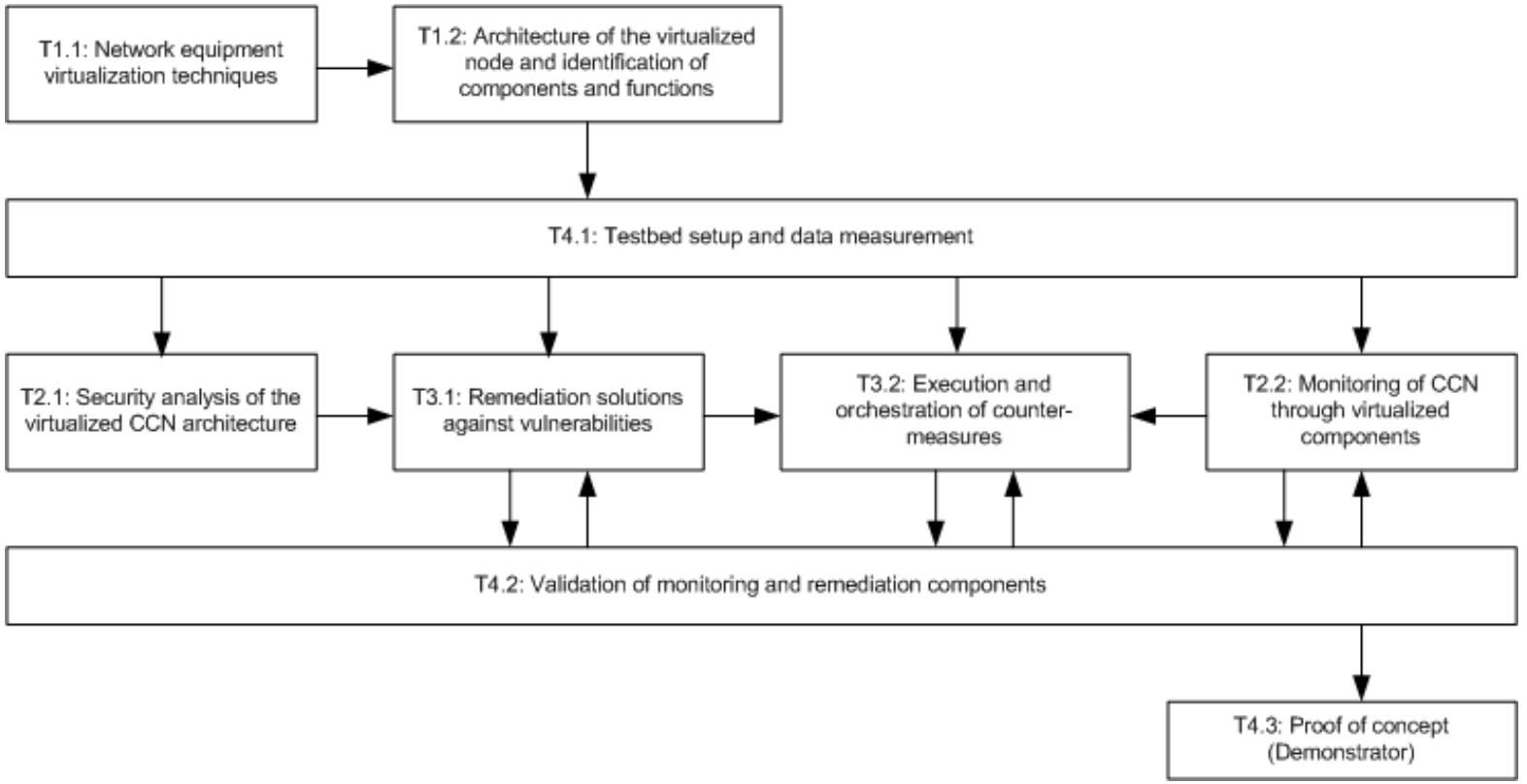
- Co-existence of multiple network protocols in the same virtualized node and migration steps
- Monitoring & Security of the virtualized NDN network: Identify flow, correlate information
- Dependability over an entire managed domain: management & control using SDN
- Collect real traffic: end-users accessing existing popular web sites
- Analysis of network and user data for evaluation (efficiency, performance, reliability, etc.)

Methodology

- Set up of a real testbed for end-users accessing Internet web sites (HTTP)
- Design and implementation of virtualized NDN network (over IP)
- Monitoring & Collection of network and usage data
- Analysis of attacks and definition of counter-measures
- Full implementation of a management plane with a focus on security
- Incremental integration of PoC and evaluation of global solution in the real testbed.

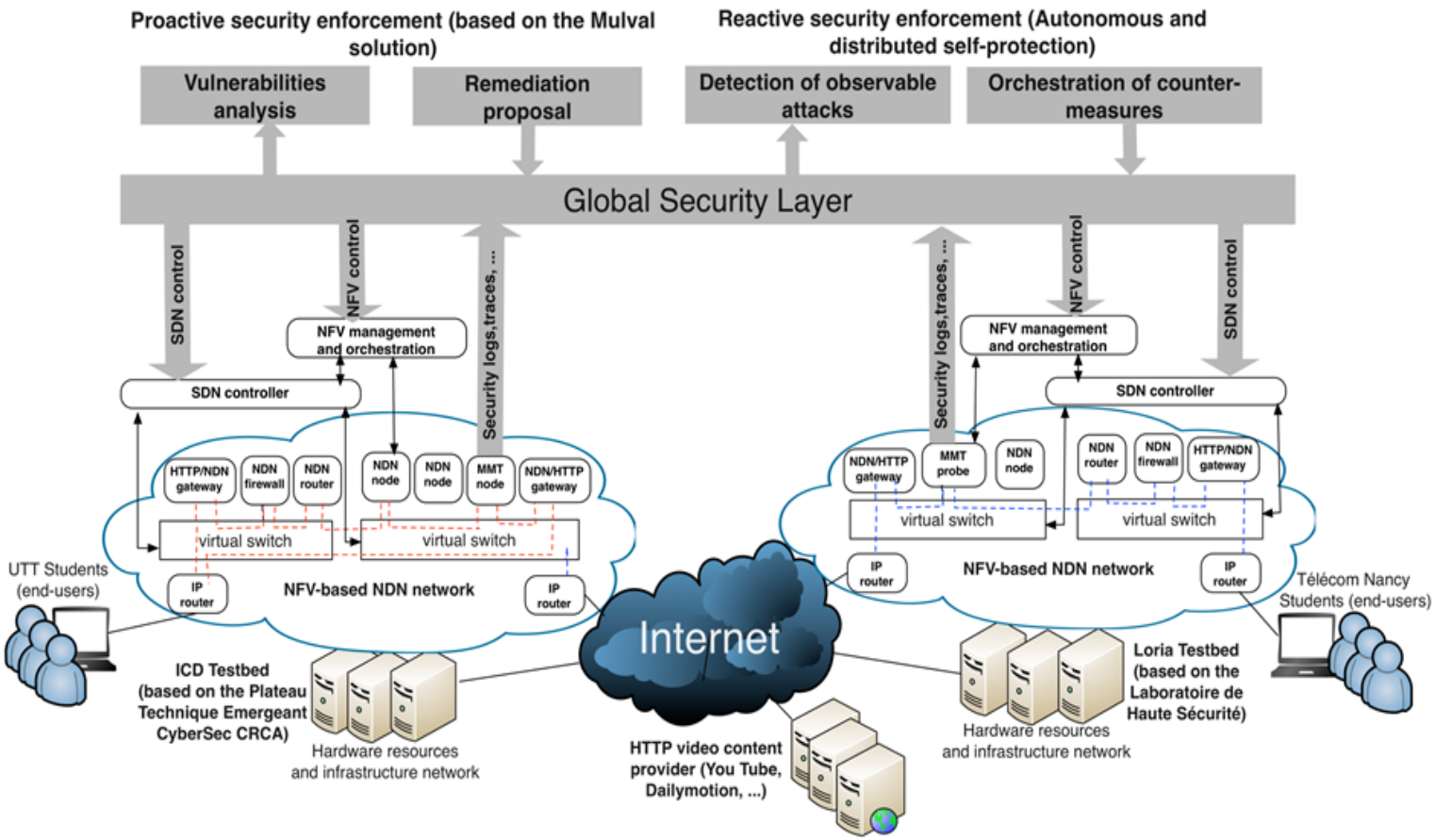
Project Organization

- Task 1: Architecture of the virtualized node for hosting network functions
- Task 2: Security analysis and monitoring of virtualized network architectures
- Task 3: Global network dependability
- Task 4: Testbed (real end-users, real services) and Demonstrator

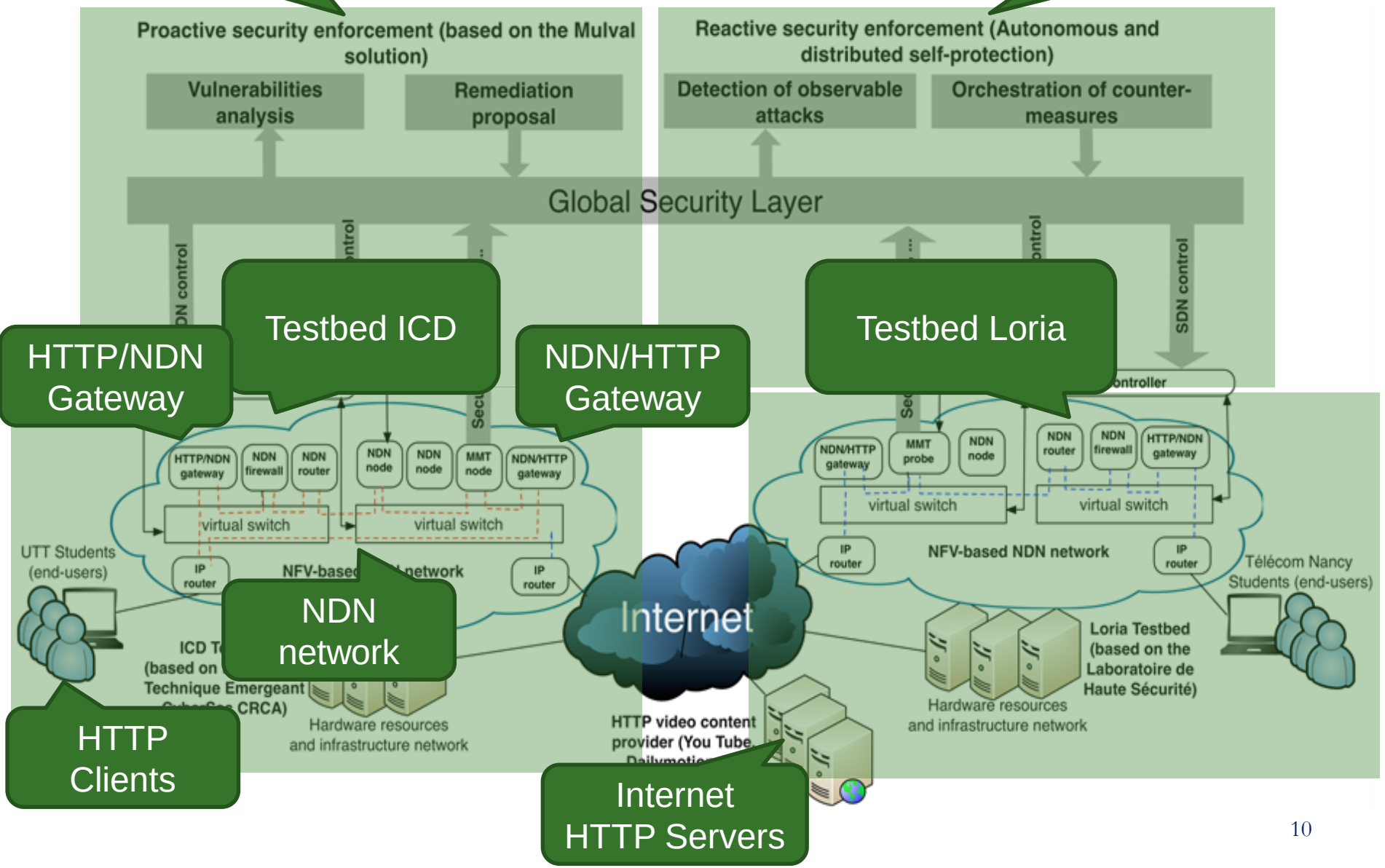


Objectives and Big picture

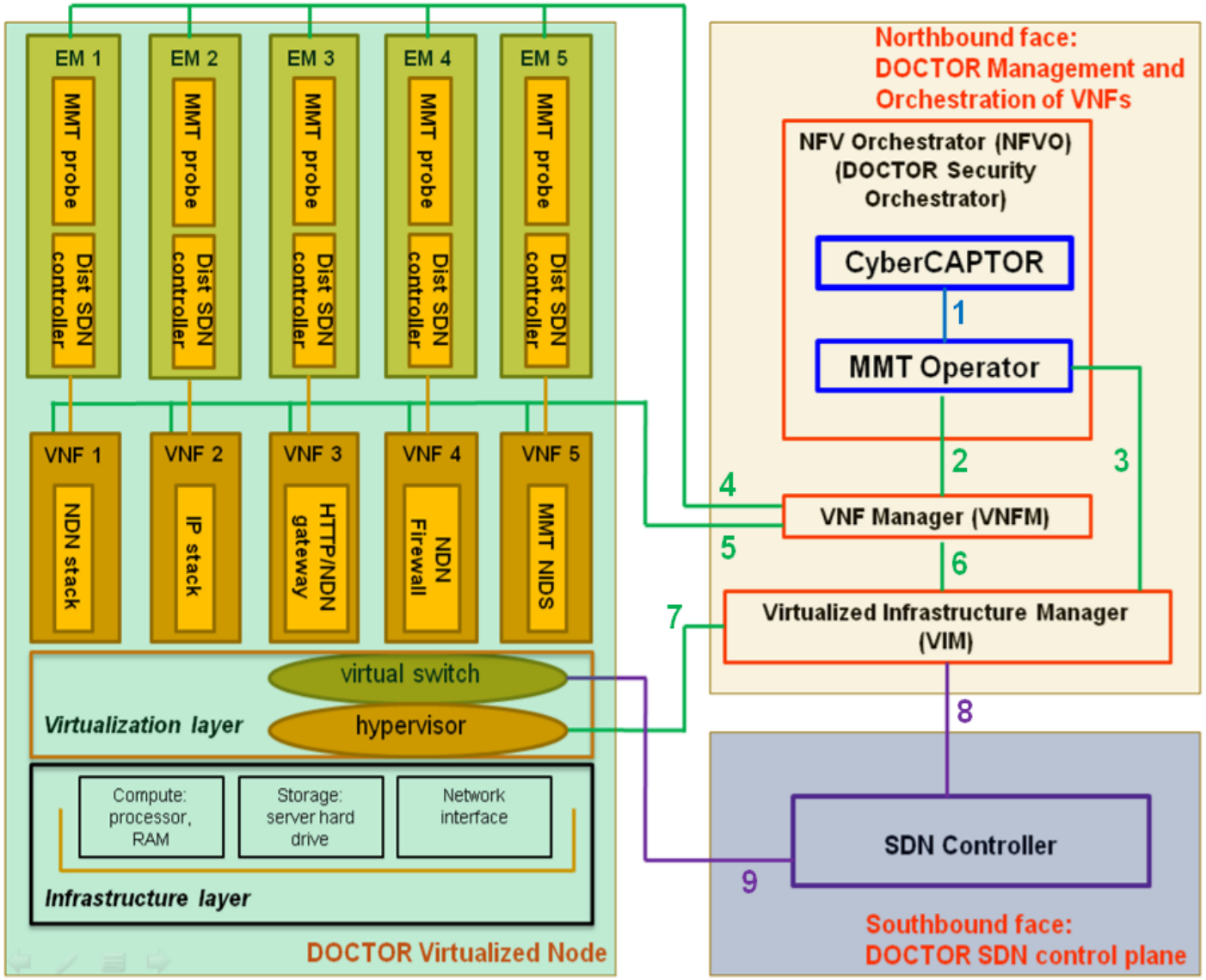
- Deployment of new network functions and protocols in a virtualized networking environment (NDN Use case)
- Monitoring, managing and securing the virtually deployed networking architectures, using SDN for reconfiguration



Network Architecture

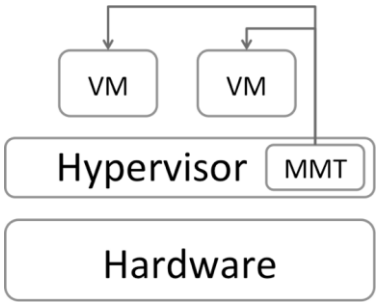


Virtualized NDN architecture

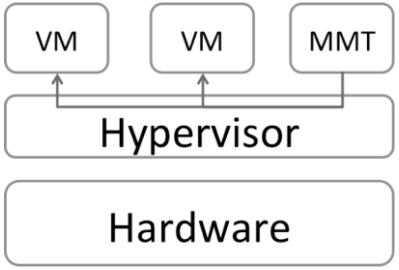


Monitoring solution

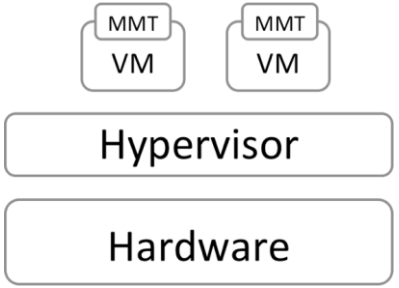
- MMT : Montimage Monitoring Tool
  - Network capture probe for traffic analysis
- 3 possibilities



Network-based



Virtual Machine introspection

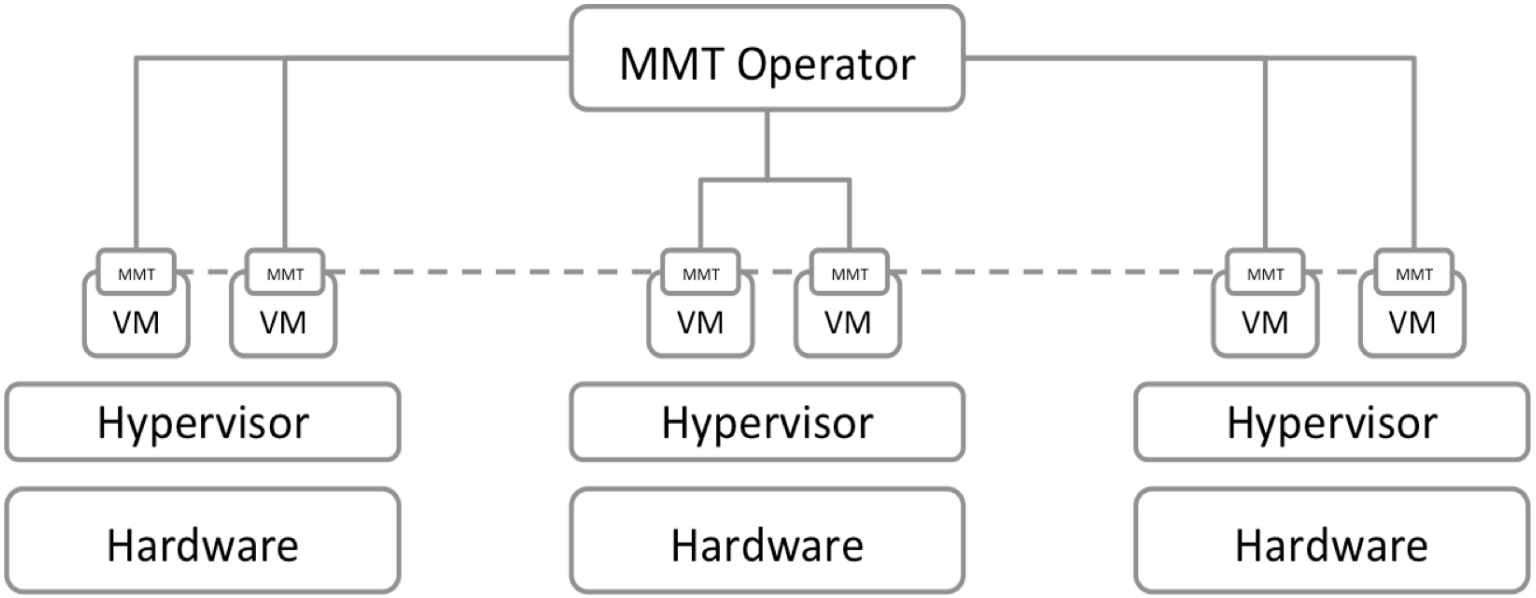


Host-based

- Host-based protection on each Virtualized Network Function
  - Better detections than network-based
  - Better performances than Virtual Machine introspection
  - Distribution of the power and memory requirements
  - Can be configured to monitor exactly what is needed for the VNF

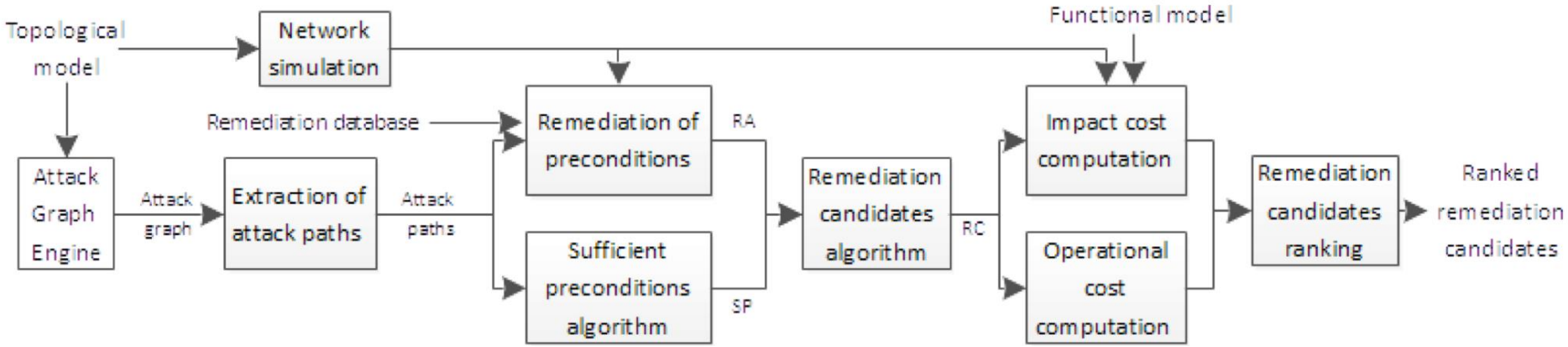
Monitoring distributed architecture

- Managed via Software Defined Networking
  - Creation of a monitoring application
  - Distributed collection of information
  - Centralized decision point (controller)
- Communication of probes via P2P for local decisions



Proactive Security Mechanisms

- Risk analysis based on attack graphs
  - Benefits from the SDN to get up-to-date topological information
  - Integration of the attacks specific to NDN (ex: cache poisoning, interest flooding attack…)
  - Describes how a localized attack can propagate in the network
- Remediation strategy



Reactive Security Mechanisms

- Detection of observable attacks
  - Instantiate security properties from generated attack graphs => Allow to configure the monitoring probes
  - Analysis of network traffic and extraction of relevant metadata for security analysis => Computation of security
  - Detections with prior-knowledge

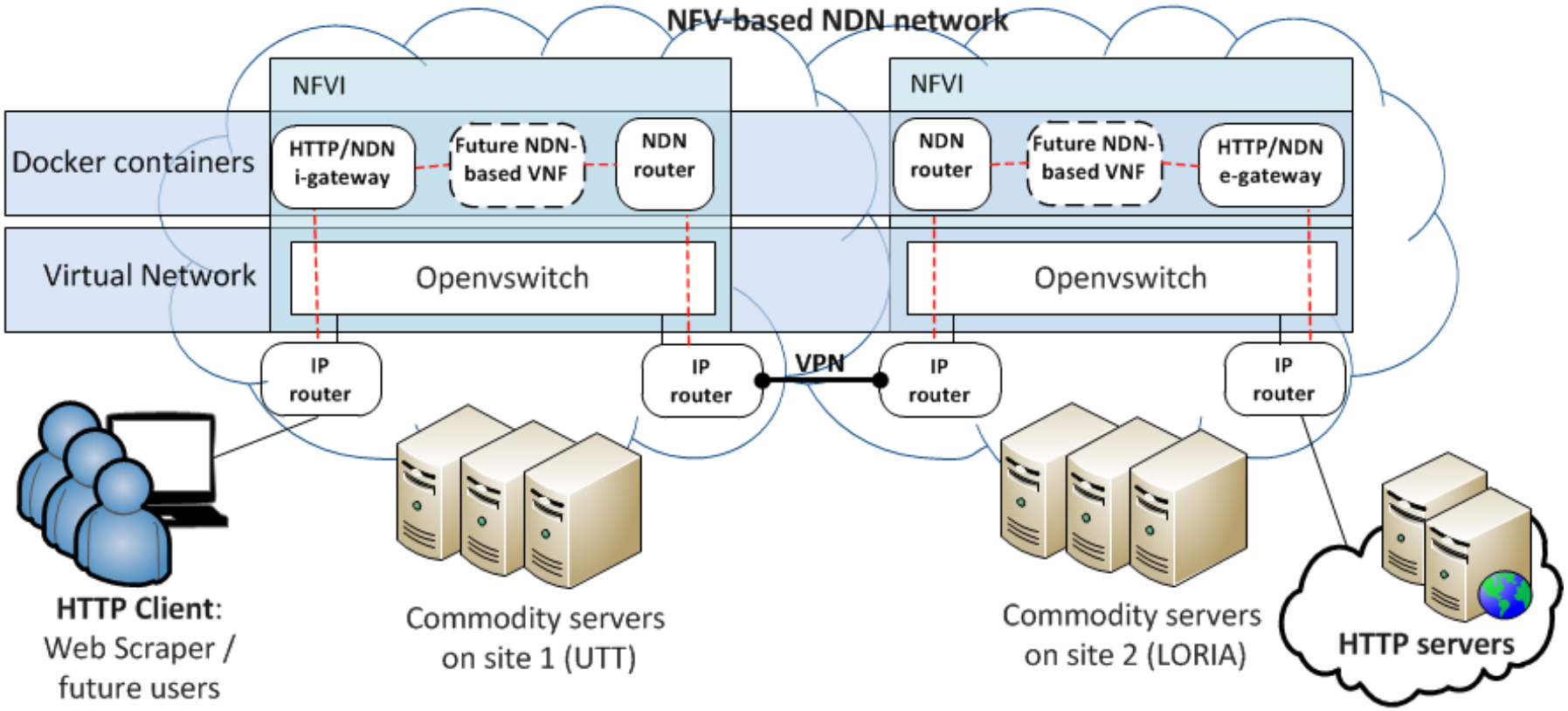
- Countermeasures on the virtualized infrastructure
  - To stop or mitigate the currently happening attacks
  - Eg. Configure/deploy a virtualized firewall, or Intrusion Detection System

NFV/NDN testbed objectives

- Progressive integration of the project outcomes
  - NDN/HTTP gateway, MMT monitoring probe, NDN firewall, …
  - Incremental evolution: Detection algorithms, Control and orchestration solutions
- Enable the experimentation of NDN with real users accessing real services
  - Both UTT and Telecom Nancy students
  - Focused on Web content
- Provide datasets for the design of detection mechanisms

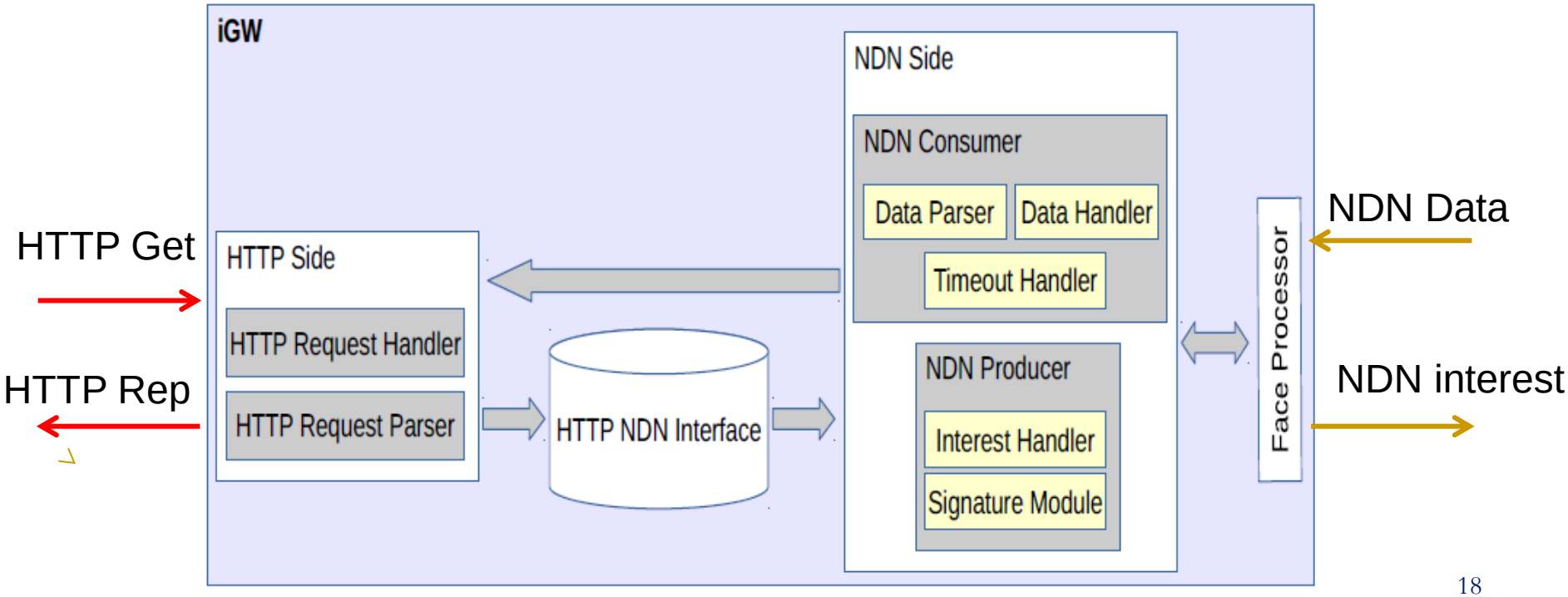
NFV/NDN testbed

- Current State: servers running, gateways deployed, Dockerized NDN nodes as a NDN network, automated deployment



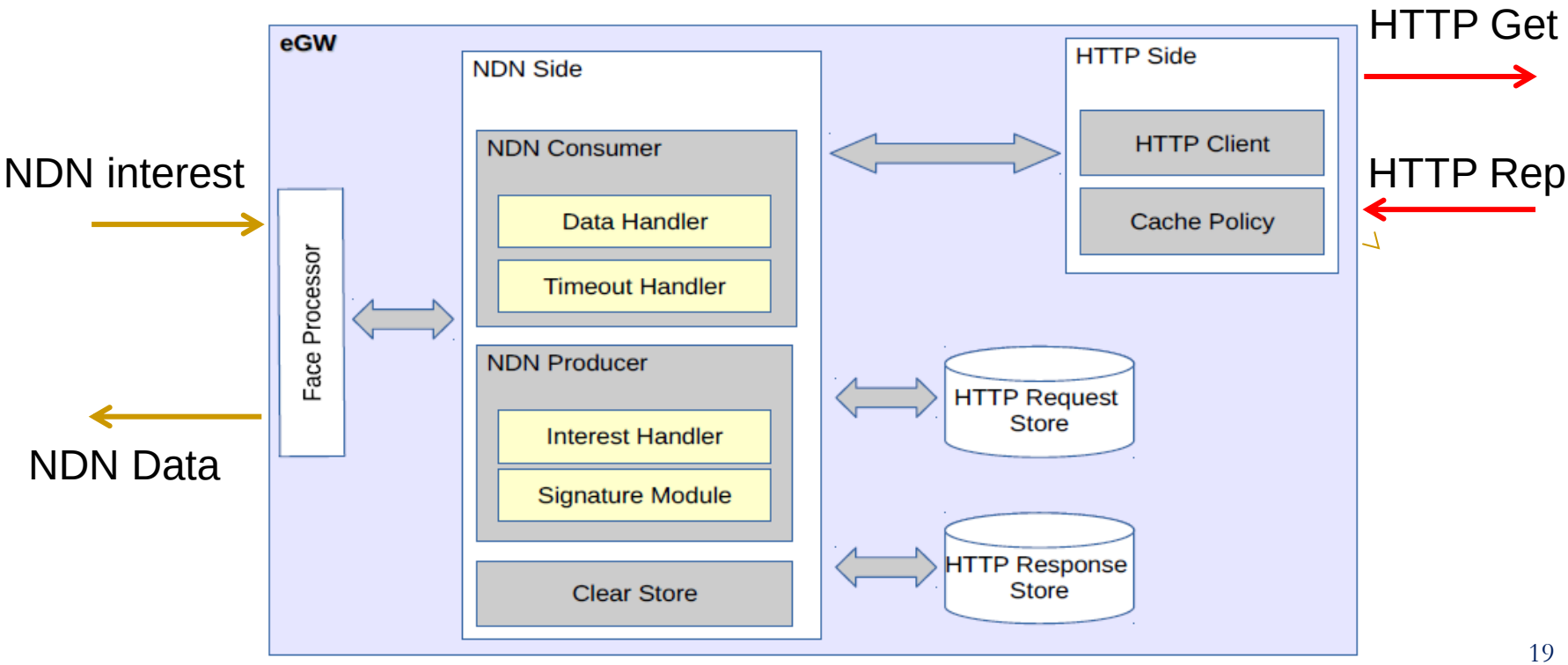
Internal architecture of iGW

- **HTTP Request to NDN interest**
  - When receiving a HTTP request, the ingress Gateway (iGW) extracts information it needs to build the NDN *Interest* packet



Internal architecture of eGW

- **NDN Interest to HTTP Request**
  - The egress gateway (eGW) aims at reforming the HTTP request sent by the end-users and at sending it to the web server.



NDN naming

- **/name_gw/meth_http/serv_http/URI/[id_body]/req_frag/req_frag_id/rep_frag_id**
  - **name_gw**: It enables to route packet in the NDN network, toward the destination eGW or iGW gateway. For this, the gateways should have already announced their prefix (which is this value for this field).
  - **meth_http**: This is the HTTP method of the request (e.g GET, POST, CONNECT),
  - **serv_http**: This is the name of the HTTP web server which this request is destinated to. The value of this field is extracted from the Header, in conformance with the IETF RFC, following the format : Name/IP@:[Port]
  - **URI**: It is the URI of the requested resource.
  - **id_body**: This field is added only in case the HTTP request contains a body (e.g. POST method). It is a string identifying the body.
  - **req_frag**: In case the HTTP request size is more than a NDN *Interest* packet size, it should be fragmented. This field indicates the number of fragments.
  - **req_frag_id**: This field indicates the id (sequence number) of the fragment this *interest* is related to.
  - **rep_frag_id**: A HTTP reply can be very long and thus fragmented in many segments. This field indicated the id of the fragment related to this packet.

ICN Firewall: motivation

- How to enforce security policies in CCN ?
  - Goal : prevent users from downloading malicious or forbidden contents
  - Authentication of content possible but real security tools missing
  - Inheritance of IP firewalls limited : no filter on IP addresses or ports
  - New security features enabled by the CCN paradigm

- Contribution
  - Content firewall : considering content name and signature
  - Use case analysis : Identification of security needs for CCN
  - Design of a semantic CCN firewall : grammar definition, preprocessing for semantic enhancement
  - Implementation in CCNx and performance evaluation

ICN Firewall: IP firewall use cases

- IP UC1 : Filtering based on the protocol (Example : http, smtp, etc.)
- IP UC2 : Filtering based on status of the connection (new, established, etc.)
- IP UC3 : Filtering based on a list of known blacklisted IP addresses
- IP UC4 : Filtering unusual inbound traffic pattern (from a denial of service attack attempt)

- Some use cases do not make sense in CCN, others must be adapted.

ICN Firewall: CCN use cases

- CCN UC1 : Filtering on content provider (Example : known untrustworthy or banned)
- CCN UC2 : Filtering on bad signature
- CCN UC3 : Filtering on content name and semantic (Example : excluding contents named with a given keyword)
- CCN UC4 : Composition (content provider & content name)
- CCN UC5 : Filtering on content direction (Example : avoid leakage of certain documents)
- CCN UC6 : Filtering on heavy traffic (Preservation of QoS)
- CCN UC7 : Filtering of stored data (Example : deny caching for specific content)

ICN Firewall: Syntax definition

- Syntax similar to iptables for ease of use and readability
- 3 different types of rules

  - rule = r_interest | r_data | r_face
  - r_interest = "interest" SP direction SP match_interest SP "pit" SP action
  - r_data = "data" SP direction SP match_data SP ["cs"|"pit"] SP action
  - r_face = "face" SP number

ICN Firewall: r_interest

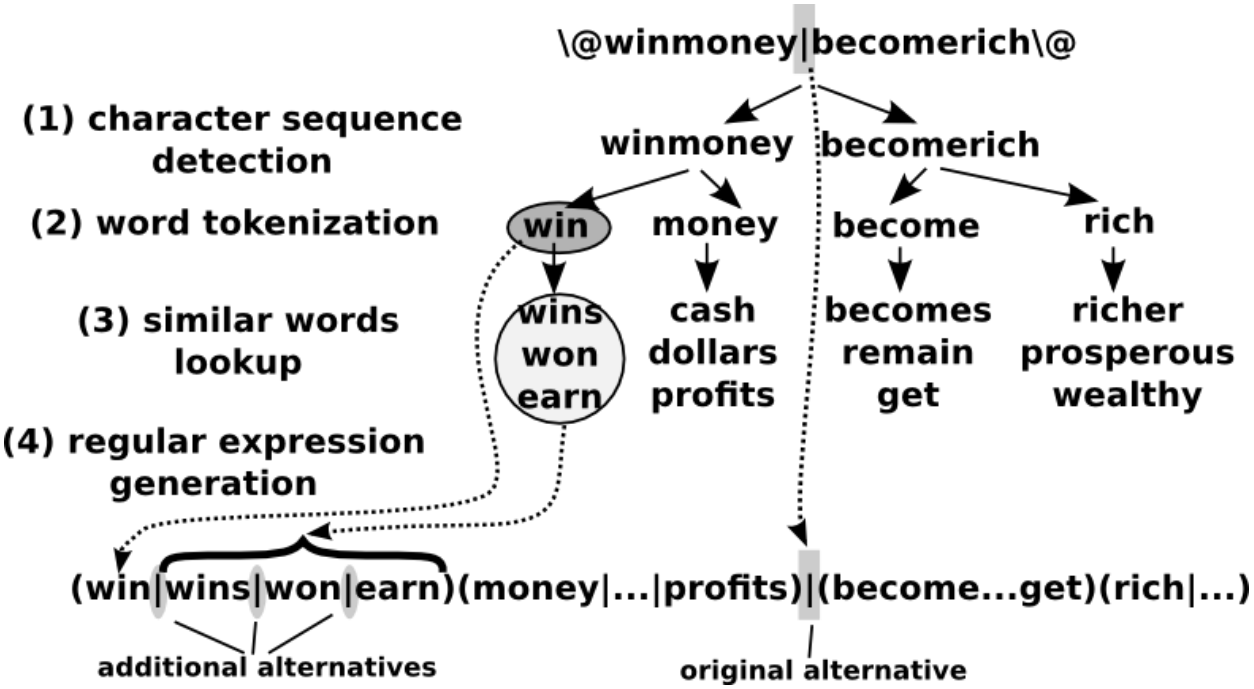
- Main rule
  - r_interest = "interest" SP direction SP match_interest SP "pit" SP action
- Syntactic elements
  - direction = "*"|"int"|"ext"
  - action = "forward" |"drop"
  - match_interest = content_name
- Example: interest * \@game|play|fun\@ 15 pit drop

ICN Firewall: r_data

- Main rule
  - r_data = "data" SP direction SP match_data SP ["cs"|"pit"] SP action
- Syntactic elements
  - direction = "*"|"int"|"ext"
  - action = "forward"|"drop"
  - match_data = content_name SP provider
  - content_name = "*"|reg_exp
  - provider = sign_check SP provider_sign
  - sign_check = "0" | "1"
  - provider_sign = "*"|first_sign *next_signs
- Example: data * \@game|fun\@ 0 0 123456789A;FFFF0000AA pit drop

ICN Firewall: Disco pre-processing

- Sequences of more than 3 characters are extracted
- Segmented as real human-readable words
- For each word, x semantically similar words are found...
- ... and included into an extended regular expression



Conclusion and Next steps

- NFV applied for migration from IP to NDN.
- Advanced management architecture for virtualized NDN infrastructure.
- Focused on security to make possible secure deployment of NDN.
- Bi-located testbed processing real user traffic
- Current/Future steps:
  - Development of new NDN based VNF and integration in the testbed (firewall, etc.)
  - Integration of the MMT probe into containers.
  - Evaluation our solutions.

Questions ?

- Join us to keep updated

http://doctor-project.org/

https://twitter.com/DOCTORprojectFR

https://www.facebook.com/ProjectDoctor

https://www.linkedin.com/groups/DOCTOR-project-8240374