SHIELD

SECURING AGAINST INTRUDERS AND OTHER THREATS
THROUGH A NFV-ENABLED ENVIRONMENT
[H2020 - Grant Agreement No. 700199]

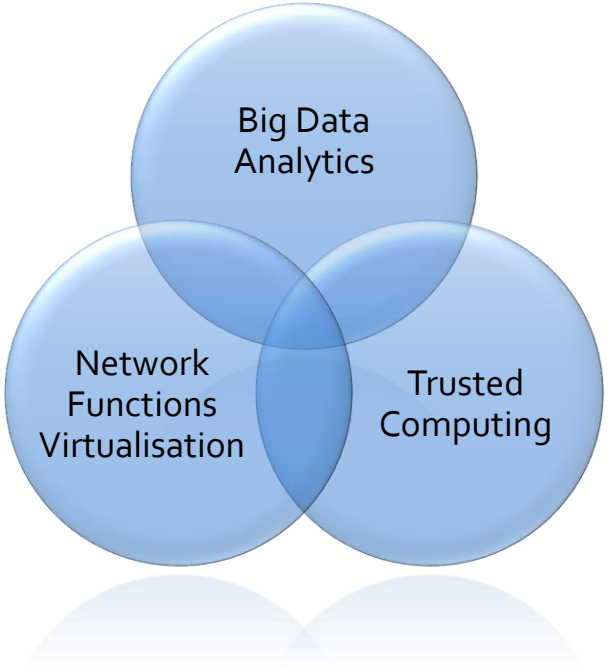
Attestation of SHIELD's network infrastructure and middleboxes.

Ludovic Jacquin, Hewlett Packard Labs (HPE).

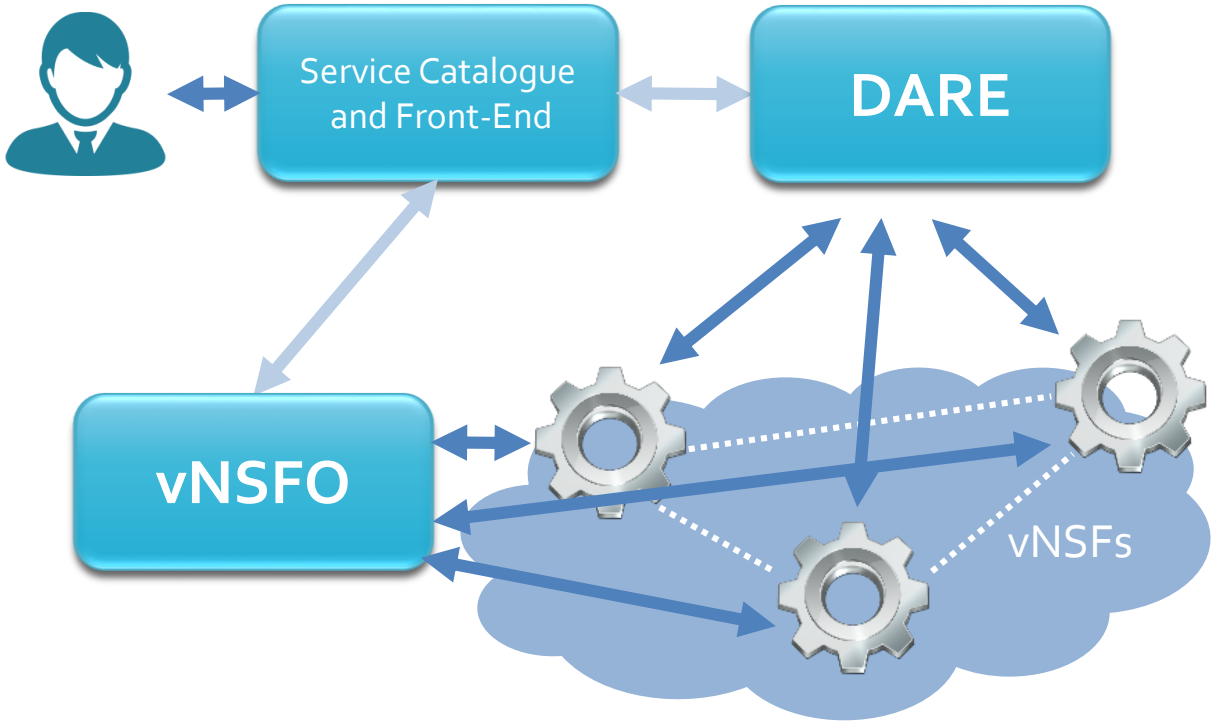
ETSI Security Week, Hot Topics in Middlebox Security
Sophia Antipolis, France, 12 June 2018


SHIELD's mission & concept

SHIELD aims to deliver an open solution for dynamically establishing and deploying virtual Security infrastructures in ISPs and corporate networks.



The SHIELD key components



The virtualisation and data/control separation gap

- The middlebox alone cannot be verified, evaluated.
  - The state is remotely controlled by the orchestration.

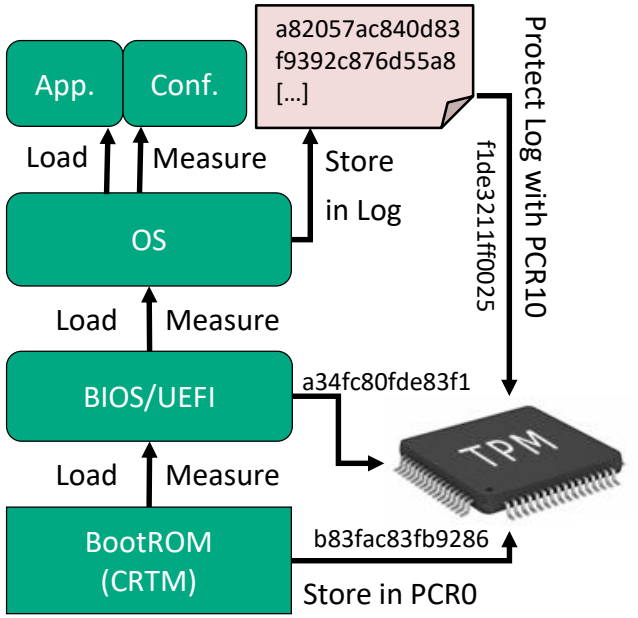
- How to verify if a middlebox is working as intended?
  - Securely
  - Automatically

- Trusted Computing technologies and mechanisms!

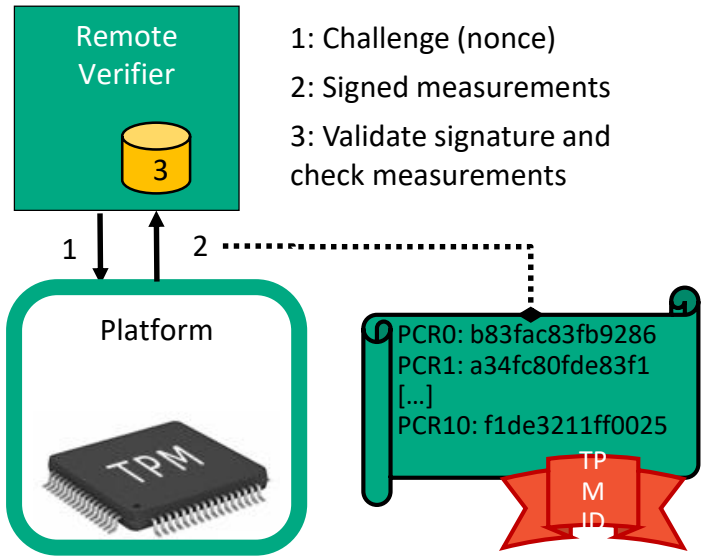
Trusted Platform Module (TPM) primer

- Security chip/co-processor widely deployed.
  - Shielded execution of standardized commands.
  - Private keys never leave the chip (configurable).
- Enabler for high-level security features:

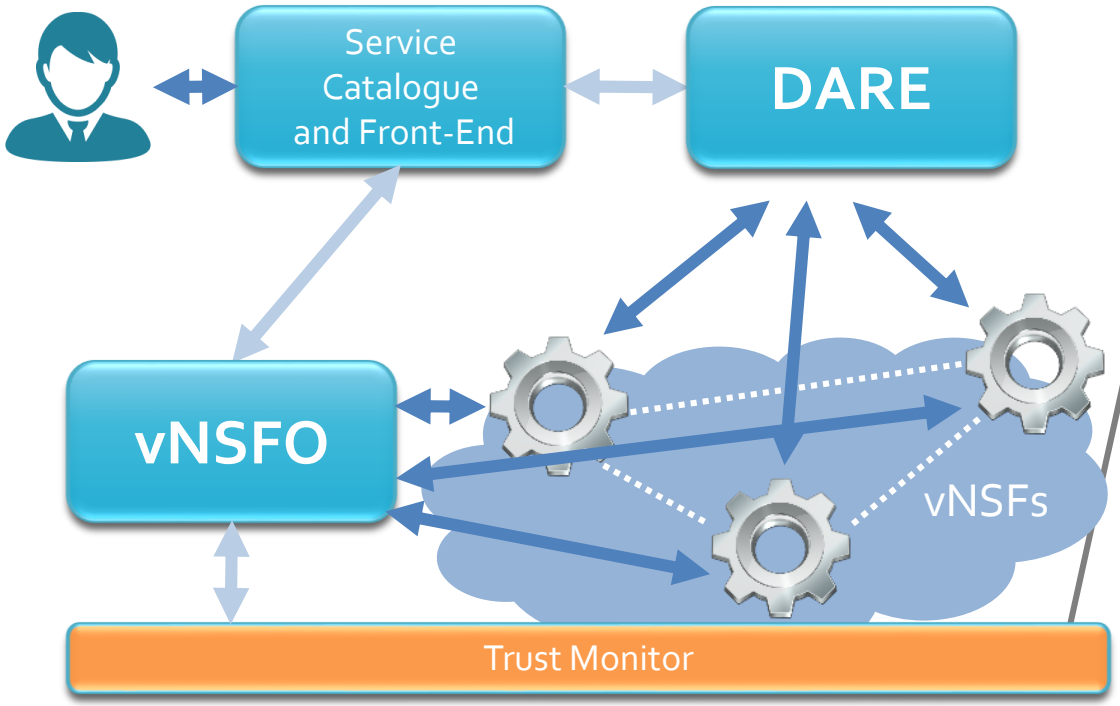
Measured boot



Remote Attestation



The Trust Monitor



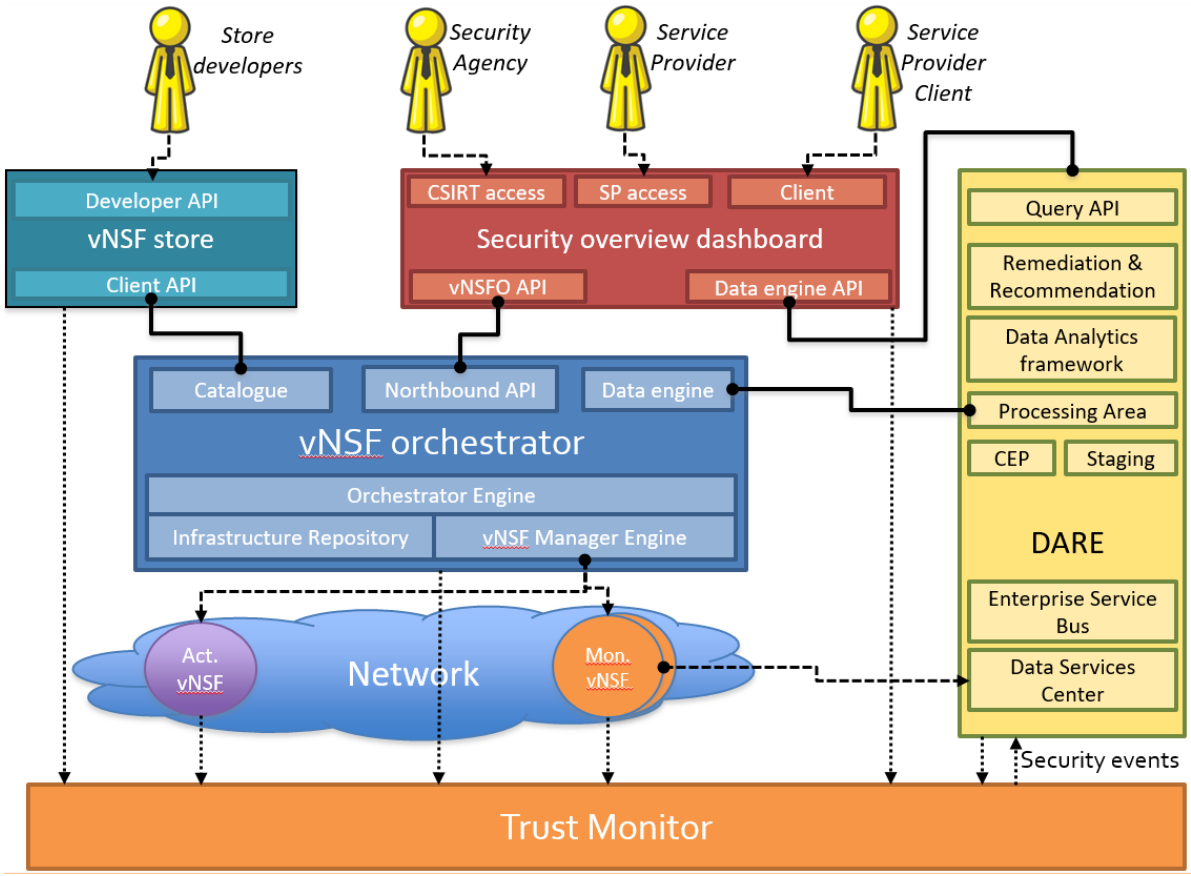
TRUSTED INFRASTRUCTURE

The trustworthiness of the SHIELD framework is implemented by relying on Trusted Computing technologies. The infrastructure attestation binds the vNSFs and the network configuration with the store and orchestration of the network.

The key components of the SHIELD framework are protected using Trusted Platform Modules (TPM), assuring the integrity of the software and the configuration.

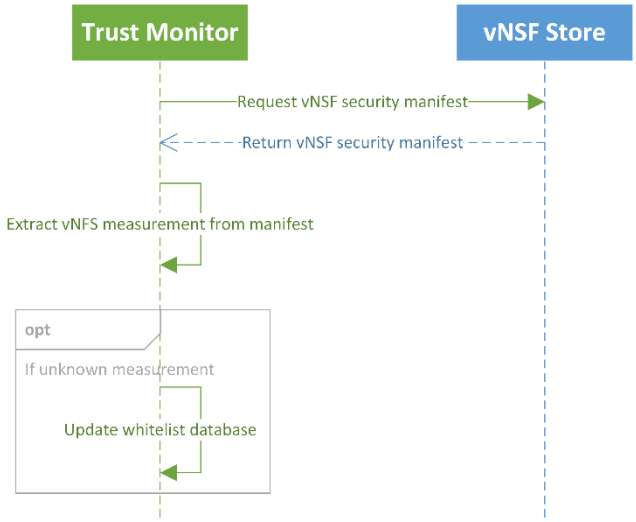
KEY TECHNOLOGIES

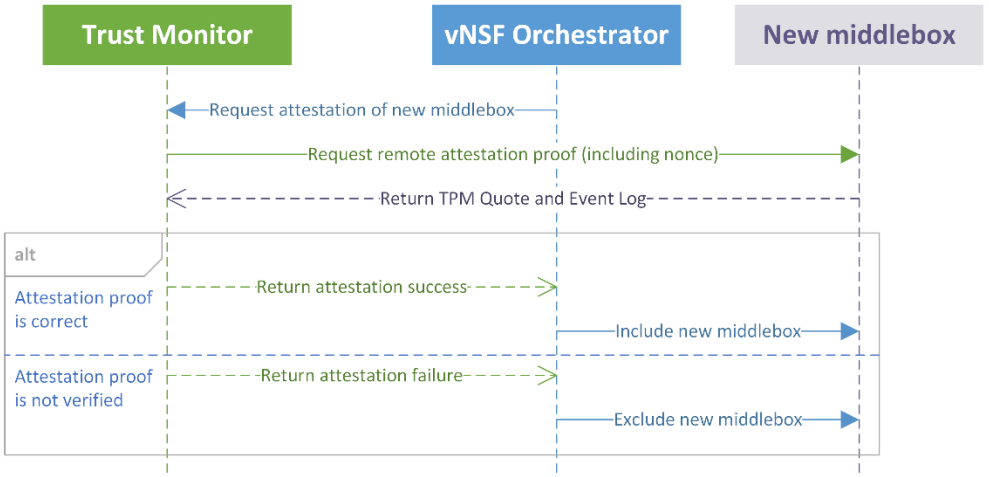
SHIELD architecture



Enrollment of middleboxes

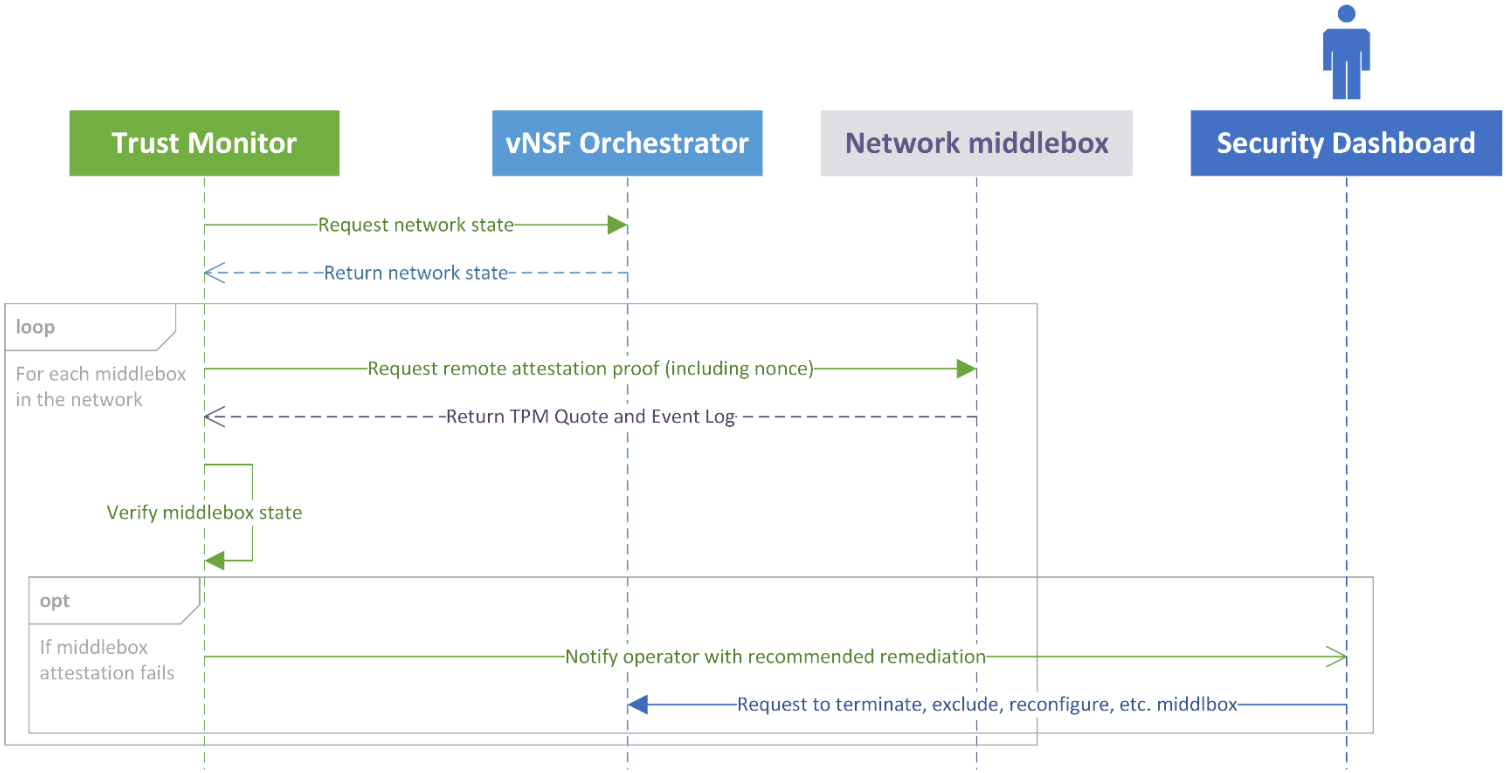
- Update of the Trust Monitor with new vNSF version.
- Verification of new middlebox before admission in the network.



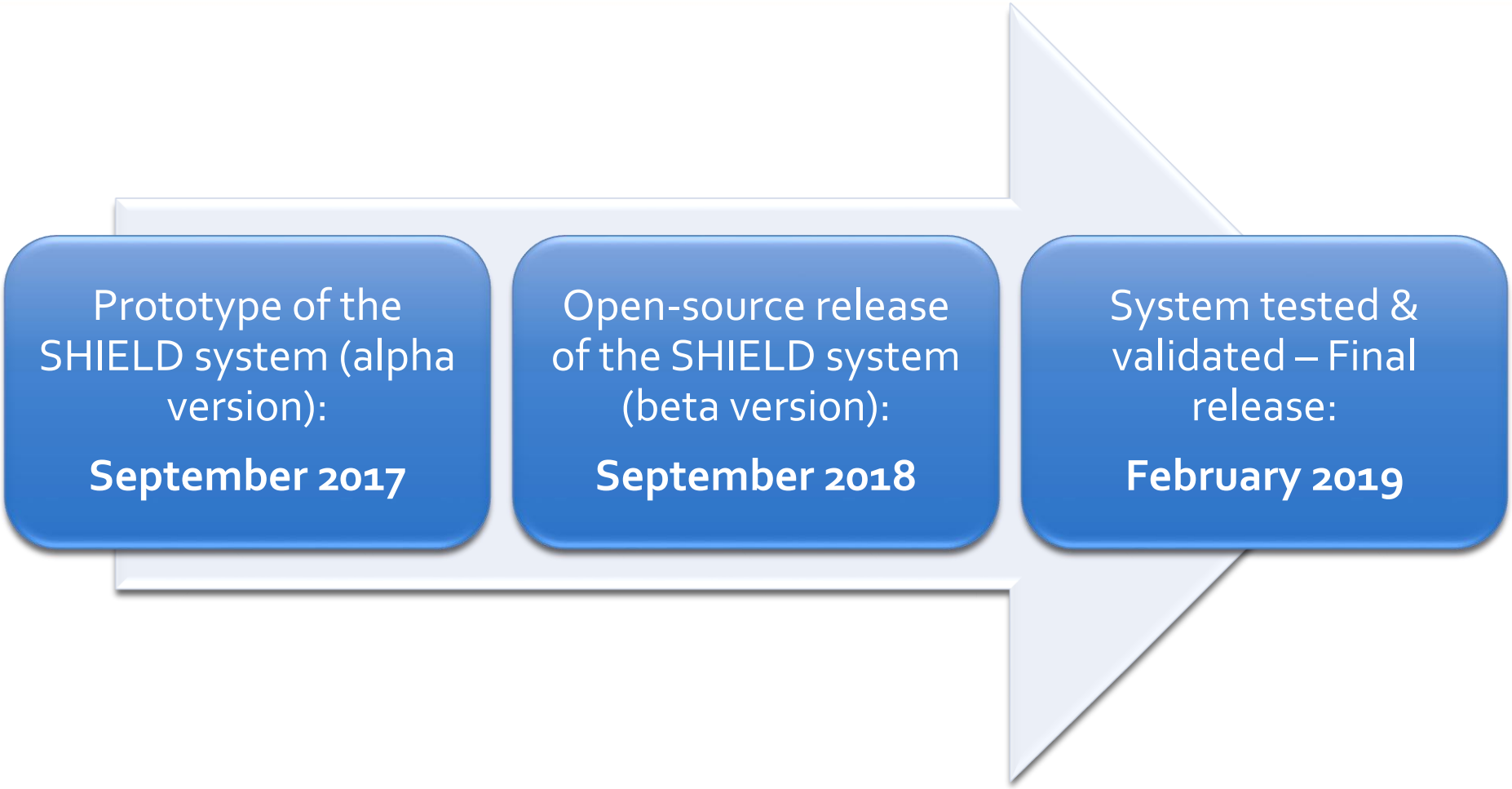


Monitoring of the middleboxes

- Continual verification of the middleboxes of the network.



Key project milestones



SDN switches attestation

- Detection of rogue SDN controller
- Detection of incorrect SDN rules

- Can handle 10k OpenFlow rules
  - Bottleneck between SDN controller & Trust Monitor
- Gap: no common (SDN controller-switch) identifier for rules

- Around 2 to 3 seconds RTT
  - 600ms for the TPM signature itself

Servers and middleboxes attestation (1/2)

- Boot-time integrity verification of the servers firmware
  - Via Measured Boot for TPM 1.2
- Run-time integrity verification of binaries and configuration files in servers and middleboxes
  - Based on the Linux *Integrity Measurement Architecture* (IMA) kernel module + custom Linux kernel
  - Tailored for the Docker lightweight virtualisation environment
- Automated generation/update of whitelist database
  - Including reference measurements from software repositories (limited to CentOS Linux distribution)

Servers and middleboxes attestation (2/2)

- Takes around 5 to 6 seconds RTT w/ verification via whitelist database (up to 128 active containers)
  - Hard limit caused by the TPM signature over the integrity report (around 2 seconds) + generation time of the integrity report
  - The verification process has a smaller influence on the overall time
- The total average time grows linearly with the number of active containers
- Gap: missing integration with NFV Virtual Infrastructure Manager

Next steps

- Include recommended remediation
  - Exclude node, update configuration, reconfigure OpenFlow rules
- Whitelist nodes allowed to communicate with vNSFO
  - Based on being verified by the Trust Monitor
- Integration with the other components

Follow us!

https://www.shield-h2020.eu/

@shield_h2020

SHIELD EU Project

info@shield-h2020.eu

SHIELD has received financial support from the European Commission under Grant Agreement No. 700199