Easing the burden of network configuration: a capability-driven approach

Looking back to just ten years ago, the complexity and flexibility of software networks have taken an impressive leap forward. These advancements had a massive impact on the network administration procedures and, of course, their management costs. However, we do not witness the same level of enthusiasm when looking at the influence on security practices.

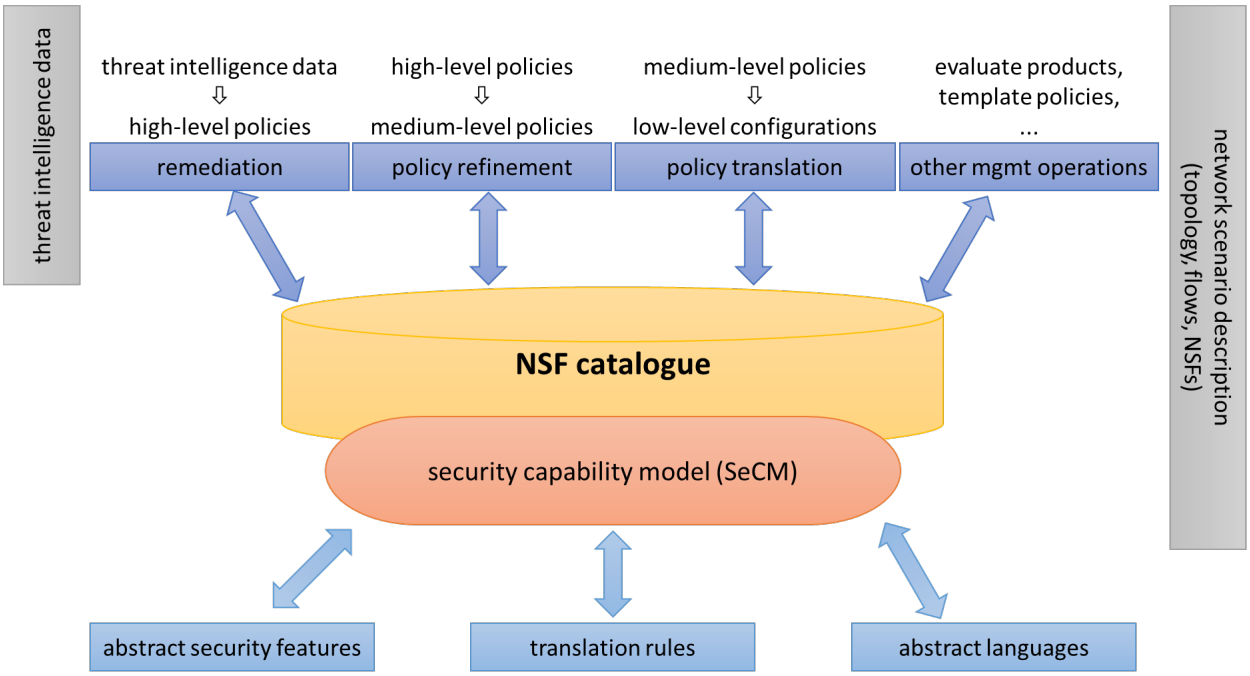
In FISHY, we identified a significant obstacle for exploiting the advantages of software networks in the security field: the complexity of security controls used to enforce protection requirements. On the one hand, network components (e.g., switches, routers) have a behaviour and configurability that is much simpler than security controls, such as packet filters, VPN gateways, and deep packet inspection. Moreover, controls from different vendors, although branded with the same properties or marketing buzzwords (e.g., intelligent, next-generation), may also have different features.

FISHY is investigating a Security Capability Model (SeCM) to bridge this gap. The SeCM is an abstract model describing what an NSF (Network Security Function) can do for enforcing a security policy. The SeCM is a UML class diagram embedding state-of-the-art design patterns to achieve maximum flexibility and follows a model-driven approach. For instance, it formally describes the actions that the security controls can enforce. For example, it can be used to state that an NSF (Network Security Function) can DROP network packets (e.g., a firewall) or ENCRYPT the traffic (e.g., a VPN terminator). In addition, it (abstractly) serves to express the conditions available for selecting the entities subject to these actions (e.g., drop the traffic only if originated from a specific IP address). From a SeCM description, it is then possible to generate the abstract language to configure an NSF automatically (e.g., produce valid configuration settings for the NSF). This operation is doable since the SeCM formally associates all the NSF features available to the administrators with its configuration language.

The NSF capability descriptions in the SeCM models pave the way for intelligent security management by leveraging the same fundamental paradigms of the software networks. Software networks are easily reconfigurable, and this level of flexibility is achieved by internally storing precise information about the network layout and the deployed components. These data stores are commonly known as catalogues. Analogously, from a capability point-of-view, the NSF SeCM descriptions available in a specific domain form an NSF catalogue.

Currently, automatic security policy refinement (i.e., translating high-level policies into lower-level ones) has various limitations and drawbacks. These systems usually lack sufficient knowledge for making the most appropriate decisions during the refinement process. On the contrary, with SeCM, we can answer questions like "Is there an NSF in our network able to implement this requirement? Can we find a better one in our catalogue?". In addition, we can configure them with a precision that was unattainable with past state-of-the-art methods. For

example, configuring a stateful firewall differs from configuring a stateless one. Knowing what the NSFs can do via our SeCM allows us to reason on the best method (and NSFs) to enforce the security requirements.



Furthermore, by playing with the networking plane and its dimension, we can also efficiently remediate non-enforceability issues, like the network's lack of features that hinder the enforcement of a configuration. NSFs can be added in proper positions to shield servers, and SDN nodes can manipulate traffic flows to perform additional checks.

The capability-centric solution of FISHY allows more qualified remediation of incidents that can lead to reactions way more thoughtful and sophisticated than the ones currently provided by an IDPS (Intrusion Detection and Prevention System). For example, our approach might play a crucial role in mitigating novel attacks, like those exploiting a 0-day vulnerability, where intelligence is needed to avoid potentially catastrophic configuration mistakes since it is impossible to rely upon past cases.

SeCM enables several operations that are nowadays cumbersome, if not impossible, to perform or automate. For instance, it allows an automated system to answer the following questions: “is product X from vendor A able to enforce the same policies as product Y from vendor B?”, or “is product Z from vendor C able to enforce a given policy?”. Being able to answer these queries positively can reduce a company's dependency on vendors. However, SeCM reaches even farther. It is empowered with information to formally explain how general concepts (e.g., a NSF can drop packets based on IP address conditions) can become a NSF-specific configuration settings (e.g., *iptables -A FORWARD -s 1.2.3.4 -j DROP*). After buying a better and cheaper firewall from a new vendor, we get it seamlessly configured with the same behaviour as the old one in seconds and without human errors. Therefore, getting the actual configurations from abstract policies becomes as easy as changing the NSFs themselves.