Using Aiders for Securing Communications of Resource-Challenged Mobile Devices

A. M. Rashwan[1], A-E M. Taha[2], and H. S. Hassanein[1]

[1]Telecommunications Research Lab
School of Computing
Queen's University
Kingston, ON, Canada K7L 3N6
{arashwan, hossam}@cs.queensu.ca

[2]Electrical Engineering Department
Alfaisal University
P.O. Box 5092
Riyadh 11533 KSA
ataha@alfaisal.edu

Abstract— **Mobile computing proved to be essential in today's cyber communications. However, entities in mobile computing are known of having limited energy, physical, and logical resources. This imposes various challenges that greatly affect communication quality and performance of those mobile entities, especially when applying computationally-intensive security measures that are essential for protecting the communication sessions. Therefore, it becomes vital to seek suitable security techniques that balance between the communication performance and the resource context of those mobile entities. In this paper, we introduce the use of external aiding entities to assist in securing communications between feature-limited and resource-challenged next-generation mobile entities. We start with outlining different resource aiding approaches that help in securing communications. Then we discuss, in brief, both the design criteria and directions for a security resource aider. We, in the end, outline some of the challenges toward using security resource aiding in mobile and next generation communications.**

Keywords-component; security; resource-aiding; resource-lending; security operation; mobile computing security; next generation Internet security.

I. INTRODUCTION

The guarantee of having secure next-generation mobile networks relies on how well its components are fitted for secure communications. However, the security of a communication network is still prone to its weakest or less-capable components that cannot integrate the necessary security measures due to various constraints. For example, an IoT communicating entity, such as a sensor or an RFID tag, may not have suitable computational resources or energy to handle even highly flexible implementations, or to integrate a group of security measures that can meet sufficient security levels for communication purposes. Moreover, different communicating entities may implement different set of security measures that fit their resource context. There is no guarantee, even with a flexible security framework, that any two communicating entities can have the same group of security measures. Another constraint is when a communicating entity cannot use its security measures to communicate directly due to restrictions imposed by the other communicating party or by the network for various performance and security level requirements.

Under the aforementioned constraints and many others, it is extremely challenging to design an all-in-one general-purpose communication entrustment for the various communicating entities of the next generation mobile networks. Even with a context-aware security in place that works with majority of entities, achieving a sufficiently secure next-generation communication may require complementary assistance to help communicating entities with constraints. This paper looks into some external aiding approaches that can be used to complement existing communication security platforms and solutions to achieve better overall network security.

The remainder of this paper is organized as follows. Section II briefly refers to the past efforts in security aiding and possible design characteristics for having an aiding approach to complement a general-purpose communication security framework. Section III outlines different resource-aiding approaches that can be used in helping to secure next generation networks communications. Section IV discusses design criteria and directions for a general-purpose communication security resource-aiding framework. Additional open issues are discussed in Section V. Finally, the paper is summarized with possible future directions in Section VI.

II. BACKGROUND

Existing security solutions are usually resource-intensive, especially the ones relying on cryptographic measures. This in turn resulted into imposing huge burden on computational and energy resources of the communicating entities leading to big performance degradation under secured communications. As a result, recent research and commercial efforts focus on providing extra-resource aiding solutions in quest of spending up secured communications for resource-challenged entities. Examples of such efforts include SSL/TLS acceleration [1], Application Delivery Controllers (ADCs) [2], hardware-accelerated cryptography [3], and Security as a Service (SECaasS) model [4]. Many of these solutions have been proven to be effective in their targeted areas to provide enhanced and secured communications. However, most of them were designed with technology-specific insights and with a little consideration for the possibility of inter-communications between different network technologies.

The trending research direction nowadays focuses on the advancements in the development of mobile and location-independent entities, such as in ICN and IoT networks. Such trend led to the appearance of the Software-Defined Radios (SDR) [5], Software-Defined Networks (SDNs) [6], and SDN protocols such as OpenFlow [7], which provide easy and economical methods for researchers to develop solutions for the newly targeted networking technologies. With that being said, we believe that there is a growing need to implement some sort of high-level network abstraction that allows different technologies and applications to

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

communicate together securely. This belief is backed up by the recent introduction of protocols and standards for IoT and ICN that was based on older technologies that are used in exiting Internet infrastructure. Examples of such new standards include HTTP/2 [8], Constrained Application Protocol (CoAP) [9], which is based on the Representational State Transfer (REST) model used by older standards such as HTTP, and the reliance of some proposed content delivery approaches for ICN on existing protocol such as HTTP and RTP/RTCP [10].

Similarly, designing a security protocol or framework for next-generation mobile networks requires a high-level network abstraction approach to allow different technologies to communicate together securely. Such design, along with any complementary security aiding proposals, should take into considerations two key characteristics of futuristic communications, which can be summarized into the following:

- **Entity Variance.** Futuristic entities may include all identifiable and communicable objects: (e.g. a web services, a sensor, a self-publishing content, a mobile device…etc). Normally those entities have various characteristics and communication protocols, and so cannot communicate without a translator. Moreover, some entities may not have its own physical resources and will require a physical host from where it can communicate.
- **Infrastructure Variance.** Communicating entities may have similar processing capabilities and use same software-level communication protocols, but their underlying physical infrastructure may be different (For example: IPv6 vs. 6LoWPAN [11]). To ensure seamless communication between entities of different infrastructures, communication translators must be incorporated with all the necessary physical interfaces to allow inter-communications between the involved infrastructures.

With having diverse entities and infrastructure technologies, the success of any futuristic general-purpose communication security solutions depends on how well its security aiding framework is designed to efficiently handle the variance and complement the communication security. In the following sections, we go through possible security resource-aiding approaches and the details toward designing communication security platforms for next generation mobile networks.

III. Communication Security Aiding Approaches

In the following subsections, we discuss some of possible resource-aiding approaches that can be incorporated with a communication security framework to offer comprehensive solution for various communicating entities and infrastructures within the next generation mobile networks (Fig.1). We also outline advantages and disadvantages of using each of the discussed aiding approaches.

A. Security Gateways

Security gateways are specialized communicating entities equipped with multiple interfaces to provide secured communication-relaying services between entities and networks of different characteristics and requirements. Examples include Virtual Private Network (VPN) gateways, firewalls, proxies, and IoT/ICN gateways. Common services provided by security gateways are outlined as follows:

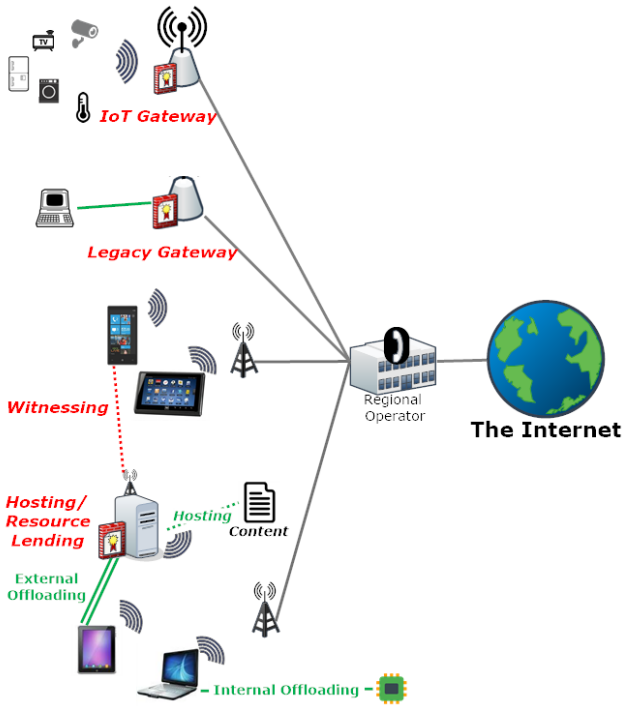

Fig. 1: A network diagram illustrating the different types of resource-aiding for supporting security functionalities.

- **Address Translation.** Gateways act as address translators that help in relaying communications to and from internal entities to the external networks.
- **Interface Translation.** Specialized networks such as IoT and Cellular usually have different low-level communication protocols from that on regular IP networks. Gateways in such networks have translation engines, allowing seamless communications between different network technologies.
- **Enhanced Services.** Service provides and communication entities may demand to meet certain security level requirements (for example: having a minimum security strength or a digital signature) and/or communication performance requirements (for example: demanding a minimum latency). Gateways usually apply enhanced services to communications through revising and applying security measures to the information relayed between communicating entities in order to meet the security and performance demands.

B. Resource-Lending (Offload) Engines

From a communication perspective, a resource-lending, also known as offloading, is a procedure where a communicating entity uses a resource of an external resource entity to process messages prior sending them. Unlike in gateways, where messages are processed and delivered to destinations on behalf of the communicating entities, resource lending engines return the processed messages to their originating communicating entities to be sent directly.

Based on how resource-lending engines are interacted to the communicating entities, there are three classes follow:

- **Internal Lending Engines.** Also know as hardware accelerators, a special kind of co-processing hardware integrated into entities to aid in handling complex operations such as processing graphics and cryptography [12] in an efficient manner. Existing implementation examples include cryptographic coprocessors [3] and TCP accelerators [13].
- **External Lending Engines.** Like gateways, external offload engines are also specialized communicating entities aimed to provide communication services. They differ from gateways in a sense that they work like the internal offload engines, yet they are completely separated from their serviced entities.
- **Hosting Engines.** Next generation of virtual entities, such as multimedia content and portable services, can be uniquely identifiable and mobile but not communicable without being hosted on a physical resource such as a content server or a virtual processing server. Therefore, a hosting engine is basically a physical resource that accommodates virtual entities and communicates on their behalves. Hosting engines are also responsible for providing communication identification and security services for those virtual entities, although they may not be affiliated with their hosted virtual entities; a common scenario that can be seen in virtual networks and ICNs, where entities are not materialized objects.

C. Witnesses/Guarantors

From a communication perspective, witnesses and guarantors are third-party entities that watch over communication between entities (usually through separate communication channels) and offer witnessing or guaranteeing information to the entities without directly being involved into the communication sessions themselves. Witnesses and guarantors can be beneficial in scenarios where entities require having better-than-nothing security while having full control of the communication session, yet cannot process or offload complex security operations associated with the communications. Examples of possible services include: partially or fully receiving a copy of the messages to witness between the endpoints, and confirming the identity and state of the connected endpoints.

IV. Security Aiders for Mobile Entities

In this section, we present conceptional design criteria and directions for a futuristic communication security resource-aiding framework for mobile entities that utilizes a collaborative setup of some of the aforementioned resource-aiding approaches. We mainly focus in incorporating gateway and external resource-lending aiding as essential aiding scalability tools for servicing diverse entities and networks.

A. Design Criteria

- **Entity Accommodation.** We define a mobile entity to be any communicating entity with ability to change locations, which includes all physical and virtual entities. Any proposed security resource aiding solution should be able to easily, and in real-time if applicable, incorporate modules for the services they provide. Such aiders should also benefit from secured standardized service announcement in which services and resource capabilities are offered for the interested entities.
- **Access to Entities.** A security resource aider cannot provide services to the requesting entities if, for some reason, it cannot gain access to those entities it intends to service. Restricted access can be due to risk management actions (e.g., malware quarantining), and prevention measures (e.g., firewall access restrictions). For a prospective security resource aiding framework to be successful, it should be designed with access restrictions into consideration. This means incorporating measures for security resource aiders to seek and obtain, if applicable, a special permission to operate unrestrictedly in the networks they intend to serve. These aiders may also need to a minimum level of security guaranties to be able to successfully obtain such special permission.
- **Aiding Trustfulness**. The requesting entity may entrust the task/information with an external aider with no sufficient measures to determine if that aider is trustworthy or compromised/malicious. Any prospective security resource-aiding framework proposal should consider incorporating some sort of tools for the communicating entities to seek the trustfulness of their prospective resource aiders. Such tools may include a trusted third-party aiding certification, an authorized third-party blacklisting, and a community-based reputation system.

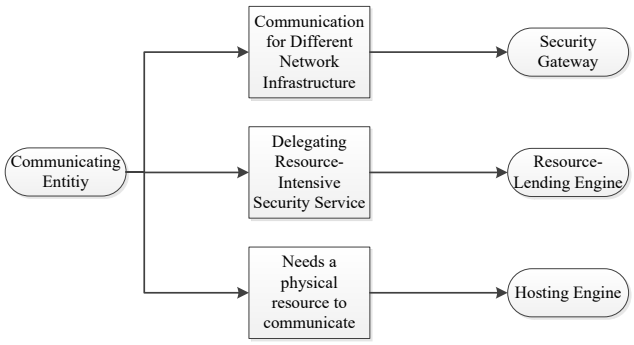

Fig. 2: A conceptual illustration of architectural considerations of a general-purpose resource-aiding security functionalities.

B. Proposed Design Directions

- **Architecture Overview.** Fig. 2 briefly illustrates an architectural concept of a general-purpose communication security resource-aiding subsystem. The subsystem utilizes three of the previously discussed resource-aiding approaches: aiding gateways, external lending engines, and hosting engines. Aiding gateways provide communication and security services for resource-limited entities communicating with other entities from different networks. External lending engines provide communication security processing services for entities that cannot handle the processing themselves due to computational, energy, or latency constraints. Hosting engines provide hosting, communication, and security services on behalf of the virtual entities (For example, video content). The hosting engines can be also supported by aiding gateways to allow secure migration of virtual entities between different networks.

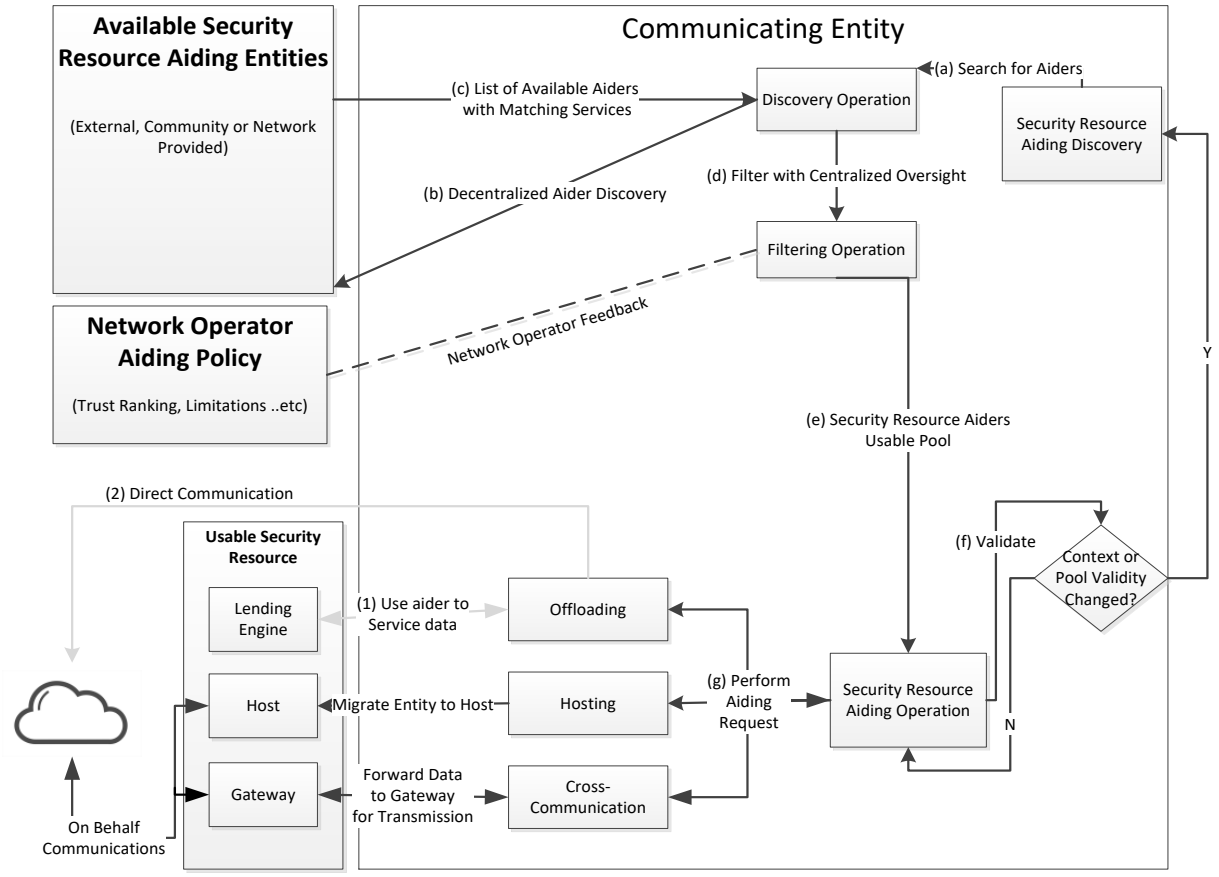

Fig. 3: The operational architecture and the discovery process of a resource-aiding framework for security functionalities.

- **Aiding-Service Discovery**. Fig. 3 illustrates the operational architecture and the resource discovery process in a general purpose security resource aiding framework. The resource discovery process holds true only for the hosting engines and the external lending engines, since the aiding gateways are usually preselected as part of core network implementations. In case of an aiding gateway, the network operator usually provides the list of fixed-location resource aiding gateways, eliminating the need to implement a discovery mechanism. In case of hosting and lending engines, unless there is an access restriction that prevents the aider from fulfilling its duties, any resource-capable entity can get nominated as resource aiders, thus the need of a discovery mechanism. However we believe that introducing new hosting and lending resource aiders into a network should be conducted with oversight from the community and/or the operator of the involved network in order to avoid malicious activities.

The aiding discovery process (Fig. 3.a) starts with an entity requiring additional resources to process specific security service in a communication session. The entity seeks for available resource aiders that are capable of processing such service. We believe that the discovery mechanism should be decentralized (i.e. using a P2P discovery protocol) but with some centralized oversight (i.e. using blacklisting/reviewing services) similar to the election of aiders. Having a decentralized discovery can help in efficiently maintaining the discovery process in real time for scenarios where aiders have high mobility and/or network topology is continuously changing. The centralized over sighting, on the other hand, can provide a level of protection by informing the entities of the trustfulness rank of the discovered aiders. It is up to the seeking entity to filter the discovery result list of capable aiders based on their trustfulness rank.

Once a filtered list of capable resource aiders (Fig. 3.e), named *resource-aiding pool*, is obtained, the seeking entity starts by requesting service (using, for example, round robin fashion) form one of the aiders in the pool. If the resource-aider accepts the request, then the seeking entity is linked to that aider (Fig. 3.g) for the requested aiding services until they are fulfilled or until there is a change in the communication context (Fig. 3.f). Examples of context changes include the migration of the seeking entity and the sudden unavailability of the linked aider. In case of unavailability of the linked aider when in need of aiding services, the requesting entity seeks the resource-aiding pool for another aider. If the resource-aiding pool reached certain minimum threshold of available aiders as demanded by the requesting aider, or if there is a major context change resulting into unavailability of the resource-aiders in the pool, the requesting entity may re-invoke the aiding discovery process (Fig. 3.a) to obtain a new resource-aiding pool, providing that aiding service is still required.

- **Resource-Aiding Operation**. Fig. 4 briefly illustrates how a resource aiding operation is conducted. Once a

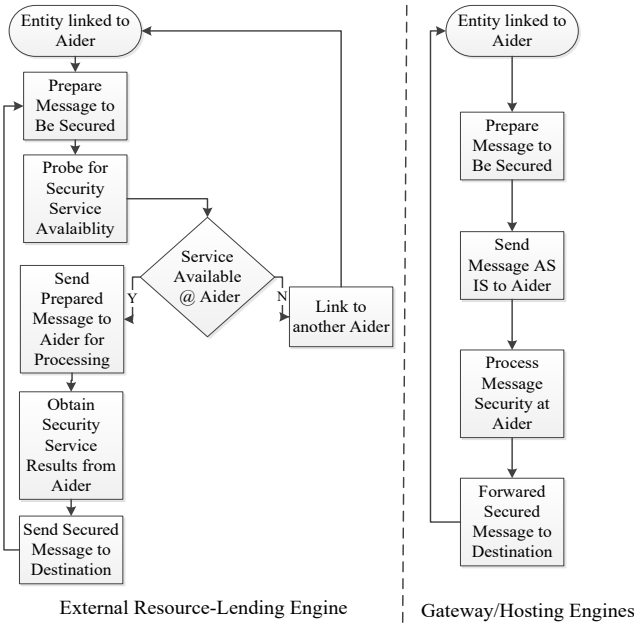

Fig. 4: An overview of decision workflow in operations supporting resource-aiding for security functionalities.

requesting aider is linked to a resource aider, the aiding process is straightforward. The requesting entity asks the aider to perform a requested security operation on information intended to be transmitted over communication session to destination entities. In case of the aiding gateways and hosting engines, the information is processed using the requested security operation and forwarded to the destination entities on behalf of the requesting entity. In case of the resource-lending engines, the information is processed at the aider using the requested security operation and then relayed back to the requesting entity before to process it farther before transmitting it to the destination entities. Moreover, the entities can be linked and requesting services from one or more aiders at the same time. For example, a virtual content entity with high access demand or an entity requesting different services simultaneously.

- **Risk Management.** Entrusting information with foreign entities does not come without a risk. This holds true for all intermediate communication nodes in the network including gateways, router, firewalls…etc. However since resource aiders can be offered by both network operators and communities, the risk can be even higher since access controlling community-provided resource-aiders is more challenging especially with dynamic networks topologies. As a result, we believe that, in addition to trustfulness measures taken by the network operators and community, the requesting entities should incorporate additional risk mitigation measures as follow:
  - Entities that requests services involving classified or safety-related information transmission should, if possible, avoid the use of aiders, rely on their own security measures before entrusting information to aiders, or use aiders that deemed trustworthy (to a certain level) by an accredited scoring system. This helps introducing some sort of a watchdog that aids in damage control due to malicious aiders.
  - Entities should not bind themselves, if possible, to a one or a same group of resource aiders at all times for resource-aiding needs. Entities should switch between aiders even during their active communication sessions in order to reduce the risk of having attack against long-lived channels between the entities and their aiders.
  - Entities should, if possible, share only the information deemed to be processed by the security service with the aider. The less information shared between the entity and the aider, the less the chance that this information could be compromised by a malicious attack.

V. Open Concepts/Issues

With the diversity of resource-challenged entities, there is no limit of how can resource aiding be implemented. Moreover, relying on external resource aiding for security services imposes additional challenges to the aforementioned design criteria as a result of sharing communication data with additional external parties. In this section, we discuss these important challenges and how they can contribute to the design and operation of a security resource-aiding framework.

A. Design Diversity

Security resource aiders are meant to serve entities and networks of various types and requirements. This, in turn, results in having challenge for designing a standardized resource-aiding system, as not all aiders are going to serve the same types of entities and networks. However, the security resource aiders are not intended to be a complete solution as their own and therefore addressing design diversity issues with security aiders is not as significant as opposed to having a complete communication security solution.

To address design diversity challenges in implementing security resource aiders, we can utilize a modular security service provider framework design. Resource aiders need only a basic service layer and can obtain or load security service modules as demanded by the requesting entities. For addressing variance in physical layer communication demands, the resource aiders can also incorporate SDR and SDN modules to address those demands.

B. Privacy

As with any system entrusted with information, there is always a privacy concern since the information is usually shared by a foreign entity. In an ideal scenario, the communicating entity should not send information for further processing without applying its own security measures such as using an available weak security measure. However if the originating entity does not have the resources or, the other end does not accept the originating entity's own security measures, the entity has to entrust the information to be processed completely by a security aider.

Addressing privacy with security aiders is extremely challenging. Unlike network-operated services, aiders are usually provided by the community (e.g. nearby resource-capable communicating entities). Even with trust scoring and aider certification in place, there is no guaranteed solution to ensure that aiders do not retain information sent by

demanding communicating entities. If privacy is important for a resource-challenged entity that cannot use its own security measures, it may opt-out from having full aiding support to partial support, in which the transmitted information are partially secured to avoid sharing fully useful private information with the aiders.

VI. Conclusions

In this paper, we introduce the concept of using security resource-aiding entities to assist in securing communications for resource-challenged next-generation mobile entities. We remark that there is a significant need for having such a solution since it is not practically possible to design a unified general-purpose communication security framework that handles wide entity diversity even with modularity and adaptation in place. As a future work, consider analyzing some performance characteristics of the discussed security resource-aiding approaches in mobile communications.

In quest of designing a general-purpose communication security resource-aiding framework, we investigate different aiding approaches outlining their strengths and weaknesses. We then propose the design criteria and directions toward combining these approaches to form the aforementioned resource-aiding framework. We briefly outline how the framework architecture can be, how serviced entities are accommodated, how aiding trustfulness and risk management are handled, and how the aiding process works.

We note few open issues due to the diversity nature of aided communicating entities and the service nature of the security resource aiders. Some of those issues can be extremely challenging to address, such as with privacy and trustfulness, as result of having community-driven aiders. Other issues, such as handling design diversity, can be managed with adopting a modular security service provider framework design that allows software-defined physical and logical layers to be loaded on demand. Although we propose some addressing directions, such as use of entity own security measures for enhanced privacy and aiding certification/scoring for enhanced trustfulness, we leave the scope of addressing those issues as future research work.

Acknowledgment

This research is funded by a grant from Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] Rick Nelson. (2014, April) SSL/TLS Offloading, Encryption, and Certificates with NGINX and NGINX Plus. [Online]. https://www.nginx.com/blog/nginx-ssl/

[2] "Application Delivery Networks: The New Imperative for IT Visibility, Acceleration and Security," Blue Coat Systems, White Paper 2008.

[3] Y. Hasegawa, "An adaptive cryptographic accelerator for IPsec on dynamically reconfigurable processor ," in *Proceedings 2005 IEEE International Conference on Field-Programmable Technology*, Singapore, 2005, pp. 163-170.

[4] Angelo Furfaro, Alfredo Garro, and Andrea Tundis, "Towards Security as a Service (SecaaS): On the modeling of Security Services for Cloud Computing," in *2014 International Carnahan Conference on Security Technology (ICCST)*, Rome, Italy, 2014.

[5] Devarpita Sinha, Anish Kumar Verma, and Sanjay Kumar, "Software defined radio: Operation, challenges and possible solutions," in *10th International Conference on Intelligent Systems and Control (ISCO)*, Coimbatore, India, 2016.

[6] (2017, November) IEEE PROJECT 1915.1 - Standard for Software Defined Networking and Network Function Virtualization Security. [Online]. https://standards.ieee.org/develop/project/1915.1.html

[7] "OpenFlow Switch Specification: Version 1.5.1 (Protocol version 0x06)," Open Networking Foundation, Specification Document ONF TS-025, 2015.

[8] Hypertext Transfer Protocol Version 2 (HTTP/2). RFC 7540, IETF, May 2015, https://tools.ietf.org/html/rfc7540.

[9] The Constrained Application Protocol (CoAP). RFC 7252, IETF, 2014, https://tools.ietf.org/html/rfc7252.

[10] Adaptive Video Streaming over Information-Centric Networking (ICN). RFC 7933, IETF, 2016, https://tools.ietf.org/html/rfc7933.

[11] IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. RFC 4919, IETF, August 2007, https://tools.ietf.org/html/rfc4919.

[12] Christian Pilato, Siddharth Garg, Kaijie Wu, Ramesh Karri, and Francesco Regazzoni, "Securing Hardware Accelerators: a New Challenge for High-Level Synthesis (Perspective Paper)," *IEEE Embedded Systems Letters*, vol. PP, no. 99, November 2017.

[13] J. Lee et al., "Network Integrated Transparent TCP Accelerator," in *24th IEEE International Conference on Advanced Information Networking and Applications (AINA)*, Perth, WA, Australia, 2010.