COINRG
Internet-Draft
Intended status: Informational
Expires: 25 April 2022

I. Fink
K. Wehrle
RWTH Aachen University

22 October 2021

Enhancing Security and Privacy with In-Network Computing
draft-fink-coin-sec-priv-03

Abstract

With the growing interconnection of devices, cyber security and data protection are of increasing importance. This is especially the case regarding cyber-physical systems due to their close entanglement with the physical world. Misbehavior and information leakage can lead to financial and physical damage and endanger human lives and well-being. Thus, hard security and privacy requirements are necessary to be met. Furthermore, a thorough investigation of incidents is essential for ultimate protection. Computing in the Network (COIN) allows the processing of traffic and data directly in the network and at line-rate. Thus, COIN presents a promising solution for efficiently providing security and privacy mechanisms as well as network monitoring. This document discusses select mechanisms to demonstrate how COIN concepts can be applied to counter existing shortcomings of cyber security and data privacy.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 25 April 2022.

Copyright Notice

Table of Contents

1. Introduction

With the ongoing digitalization, previously isolated devices and systems are increasingly connected to the Internet, concerning all aspects of life. In particular, in the context of Cyber-Physical Systems (CPS) and the (Industrial) Internet of Things, machines and infrastructure are equipped with additional sensors and CPUs to allow for automatization and higher processing efficiency. The entanglement of the sensors with the physical world leads to high sensitivity of the transmitted and collected data.

Consequently, digitalization expands the attack surface and the possible impacts of cyber attacks, increasing the importance of proper protection mechanisms.

Devices in CPS are often resource-constrained and do not offer the possibility to implement elaborate security mechanisms. Furthermore, legacy devices and communication protocols are often still used in industrial networks but were not designed to face the security and privacy challenges the new interconnection brings. Thus, communication and access are often unprotected.

Upgrading legacy devices with protection mechanisms is an effortful and expensive procedure. A promising approach for retrofitting security is the deployment of suitable mechanisms within the network. To date, this is mainly realized using middle-boxes, leading to overhead and the need for additional hardware.

One general and widespread security component is Intrusion Detection Systems (IDS) to detect and, ideally, prevent undesired events in a network. However, IDS are usually implemented in software, again running on middle-boxes or edge devices in the same network. Thus, their reaction time is limited as well as their information gain, which is usually addressed by deploying additional IDS components for traffic monitoring.

Last, the after-treatment of incidents in networks is critical to detect exploited vulnerabilities and prevent future attacks. Network forensics serves to retrace and comprehend the origin and course of malicious events. However, to provide high performance, the underlying monitoring of network traffic requires dedicated networking devices and/or expensive subscriptions to respective features, leading to high costs.

One common problem is that security is usually provided by software solutions. These often require additional hardware and lead to performance overhead, which is especially unfavorable in the context of time-sensitive applications, e.g., in industry. Existing high-performance solutions, e.g., running on traditional networking devices, require dedicated, costly, and unsustainable hardware.

Computing in the Network (COIN) covers these shortfalls by using programmable networking devices to conduct dynamic and custom processing of network packets at line-rate. Thus, security-related functions and packet inspection can be implemented and applied centrally in the network, e.g., at a programmable switch.

This draft explores the opportunities of COIN for improving security and privacy as follows: We first describe feasible mechanisms for preventing attacks and intrusion in the first place. Then, we present which mechanisms we can implement with COIN for detecting intrusion and undesired behavior when it has already taken place. Last, we explore how COIN can improve network monitoring for detecting, analyzing and following up incidents, thus fixing vulnerabilities.

2. Protection Mechanisms

The common ground for providing security and data privacy is to protect against unauthorized access. That protection is primarily provided by basic security mechanisms such as encryption, integrity checks, authentication, and authorization. These are often missing in resource-constrained environments or regarding legacy industrial devices. [RFC7744] thoroughly discusses the need for authentication and authorization in resource-restrained environments. [RFC8576] presents security and privacy risks and challenges specific to the IoT. In the following, we describe how COIN can help to retrofit suitable mechanisms.

2.1. Encryption and Integrity Checks

Encryption is critical to preserve confidentiality when transmitting data. Integrity checks prevent undetected manipulation, which can remain unnoticed even despite encryption, e.g., in case of flipped bits. Due to resource-constraints, many devices in CPS do not provide encryption or calculation of check-sums.

By default, secure cryptographic functions are not supported by current programmable switches either and hard to realize due to their computational constraints. However, there are efforts by researchers to implement AES encryption with scrambled lookup tables [CHEN] and cryptographically secure keyed hash functions [YOO] on existing programmable hardware switches. Furthermore, future generations of programmable hardware switches might provide secure cryptographic functions by design.

With secure cryptography at hand, COIN would allow to encrypt and hash packet contents efficiently while passing a respective programmable networking device. Concretely, data could be encrypted and supplemented with a check-sum directly at the first networking device passed by a packet before forwarding it through the Internet to its designated destination. Subsequent decryption and integrity checks could be executed at the last networking device before the destination. Alternatively, this could be implemented at the destination if supported by the respective device. This approach

does not require deployment of or forwarding to additional middle-boxes. Thus, no additional attack surface or processing overhead is introduced, presenting a promising foundation for retrofitting security in time-sensitive scenarios such as industrial processes.

Another use-case is the implementation of whole standards for secure communication on programmable networking devices, offering new flexibility. For example, researchers examined the benefits of deploying IPsec and MACsec on programmable switches [HAUSER-IPsec],[HAUSER-MACsec] but their implementations only target software switches due to the missing cryptographic capabilities of existing programmable hardware switches.

Future research is needed to clarify if and at which costs hardware for enabling cryptographic calculations could and should be embedded in future generations of programmable networking devices.

2.2. Authentication and Authorization

Authentication and authorization mechanisms are needed to avoid unauthorized access to devices and their manipulation in the first place. With COIN, networking devices can flexibly decide whether to forward packets, thus offer efficient and fine-granular access control.

One possibility is to conduct a handshake between the sender and networking device before starting the communication with industrial devices. Cryptographic calculations (e.g. required for certificate-based authentication) can be offloaded to the control plane if not feasible in the data plane of the networking device due to computational constraints. Existing research also proposed and implemented authentication in the data plane, e.g., using port-knocking [ALMAINI]. Authorization information can be stored in tables in the data plane or requested from the control plane. Since authentication and authorization is only needed when starting or refreshing a connection, the necessity and overhead for consulting the control plane are limited. Subsequent to the authentication and authorization process, the respective decisions can be flexibly enforced by the networking device. For example, different fine-granular policies can be linked to different authorization levels and different devices. In the case of unsuccessful authorization or authentication, networking devices can inform the network administrator about possible intrusion of the system.

Overall, COIN can realize efficient and flexible access control, reducing overhead and attack surface.

2.3. Policies

Control processes can include communication between various parties. Even despite authorization and authentication mechanisms, undesired behavior can occur. For instance, malicious third-party software might be installed on the approved device and thus implicitly gain access. Depending on the involved devices and their capabilities, proper authorization and authentication might not be possible at all. An effective way to exclude malicious behavior nevertheless is policy-based access control.

[RFC8520] proposes the Manufacturer Usage Description (MUD), a standard for defining the communication behavior of IoT devices, which use specific communication patterns. The definition is primarily based on domain names, ports, and protocols (e.g., TCP and UDP). Further characteristics as the TLS usage [I-D.draft-ietf-opsawg-mud-tls-05] or the required bandwidth of a device [I-D.draft-lear-opsawg-mud-bw-profile-01] can help to define connections more narrowly. By defining the typical behavior, we can exclude deviating communication, including undesired behavior. Likewise to IoT devices, industrial devices usually serve a specific purpose. Thus, applying MUD or similar policies is viable in industrial scenarios as well.

The problem that remains is the efficient enforcement of such policies through fine-granular and flexible traffic filtering. While middle-boxes increase costs and processing overhead, primary SDN approaches as OpenFlow allow only filtering based on match-action rules regarding fixed protocol header fields. Evaluation of traffic statistics for, e.g., limiting the bandwidth, requires consultation of the remote controller. This leads to latency overheads, which are not acceptable in time-sensitive scenarios.

In contrast, the COIN paradigm allows flexible filtering even concerning the content of packets and connection metadata. Furthermore, traffic filtering can be executed by programmable networking devices at line-rate.

Last, not only network communication behavior of devices can be defined in policies. For example, COIN can be also used to consider additional (contextual) parameters, e.g., the time of day or activity of other devices in the network [KANG].

2.4. In-Network Vulnerability Patches

Resource-constrained devices are typically hard to update. Thus, device vulnerabilities often cannot be fixed after deployment. As a remedy and special case of policies, rules could be defined to describe known attack signatures. By enforcing these rules at programmable networking devices, e.g., by dropping matching traffic, COIN would offer an efficient way to avoid exploitation of device vulnerabilities. Another potential advantage is the easy and extensive roll-out of such "in-network patches" in the form of (automatic) software updates of the networking device.

Future research is needed to evaluate the potential and benefits of in-network patches compared to traditional security measures, e.g., firewalls, and provide proof of concepts using existing devices and vulnerabilities.

2.5. Anonymization

Due to their interconnection with the physical world, the generation of sensitive data is inherent to CPS. Smart infrastructure leads to the collection of sensitive (user) data. In industrial networks, information about confidential processes is gathered. Such data is increasingly shared with other entities to increase production efficiency or enable automatic processing.

Despite the benefits of data exchange, manufacturers and individuals might not want to share sensitive information. While deployment of privacy mechanisms is usually not possible at resource-constrained or legacy devices, COIN has the potential to apply privacy mechanisms in a flexible and comprehensive manner.

Data could be pseudonymized at networking devices by, e.g., extracting and replacing specific values. Furthermore, elaborate anonymization techniques could be implemented in the network by sensibly decreasing the data accuracy. For example, concepts like k-Anonymity could be applied by aggregating the values of multiple packets before forwarding the result. Noise addition could be implemented by adding a random number to values. Similarly, the state-of-the-art technique differential privacy could be implemented by adding noise to responses to statistical requests.

Indeed, recent research exploits programmable hardware switches to implement performant and light-weight anonymization of communication by rewriting source addresses and information and hiding path information, e.g., using randomization [MOGHADDAM]. Similarly, [WANG] realizes traffic obfuscation by encrypting IPv4 addresses in the data plane.

Future research is needed to implement and evaluate further privacy mechanisms and COIN's potential for this field.

3. Intrusion and Anomaly Detection

Ideally, attacks are prevented from the outset. However, in the case of incidents, fast detection is critical for limiting damage. Deployment of sensors, e.g., in industrial control systems, can help to monitor the system state and detect anomalies. This can be used in combination with COIN to detect intrusion and to provide advanced safety measures, as described in the following.

3.1. Intrusion Detection

Data of sensors or monitored communication behavior can be compared against expected patterns to detect intrusion. Even if intrusion prevention is deployed and connections are allowed when taken individually, subtle attacks might still be possible. For example, a series of values might be out of line if put into context even though the individual values are unobtrusive. Anomaly detection can be used to detect such abnormalities and notify the network administrator for further assessment.

While intrusion detection systems (IDS) are usually deployed at middle-boxes or external servers, COIN provides the possibility to detect anomalies at-line rate, e.g., by maintaining statistics about traffic flows. This decreases costs and latency, which is valuable for a prompt reaction. Another advantage is that one central networking device can monitor traffic from multiple devices. In contrast, multiple distributed middle boxes are usually needed to achieve the same information gain. Last, programmable networking devices can be used to preprocess traffic and reduce load on subsequent in IDS components as examined by [LEWIS].

Besides intrusion, anomalies can also imply safety risks. In the following, we pick up the potential of COIN to support safety.

3.2. Dead Man's Switch

[I-D.draft-irtf-coinrg-use-cases-00] addresses the potential of COIN for improving industrial safety. Detection of an anomaly in the sensor data or operational flow can be used to automatically trigger an emergency shutdown of a system or single system components if the data indicates an actual hazard. Apart from that, other safety measures like warning systems or isolation of areas can be implemented. While we do not aim at replacing traditional dead man's switches, we see the potential of COIN to accelerate the detection of failures. Thus, COIN can valuably complement existing safety measures.

4. Network Monitoring

After detecting an incident, it is essential to investigate its origin and scope. The results of this analysis can be used to allow for consistent recovery, to adapt protection mechanisms, and prevent similar events in the future. For enabling potential investigation, traffic is constantly captured (e.g., in the form of flow records), which requires dedicated hardware in large networks. Furthermore, it might be preferable to exclude traffic, e.g., from specific subnets, from the analysis. Dynamic and fine-granular traffic filtering is not possible with traditional networking devices, leading to storage and processing overhead.

With COIN, networking devices can be programmed to export traffic characteristics without significant overhead when forwarding a packet. Furthermore, monitoring can be done more flexibly, e.g., by applying fine-granular traffic filtering. Also, header fields of particular interest can be efficiently extracted. Therefore, COIN can considerably decrease the load and increase the efficiency of security-related network monitoring.

The presented prospects are underlined by recent work. For example, in [SONCHACK], flow records are created in the control plane of programmable hardware switches while expensive packet preprocessing is offloaded to the data plane.

5. Security Considerations

When implementing security and privacy measures in networking devices, their security and failure resistance is critical. Related research questions to clarify in the future are stated in [I-D.draft-kutscher-coinrg-dir-02].

6. IANA Considerations

N/A

7. Conclusion

COIN has the potential to improve and retrofit security and privacy, especially with regard to resource-restrained and legacy devices.

First, COIN can provide intrusion prevention mechanisms like authentication and efficient enforcement of (context-based) policies. Easily deployable in-network patches of device vulnerabilities could further improve security. Encryption and integrity checks are limited by the current hardware but already targeted by recent research.

Second, COIN allows examining packet contents at networking devices, which can help implement fast and comprehensive anomaly and intrusion detection.

Last, COIN can contribute to an efficient and targeted traffic monitoring for incident analysis.

Further investigation of the feasibility and potential of the presented mechanisms is subject to future research.

8. Informative References

[ALMAINI] Almaini, A., Al-Dubai, A., Romdhani, I., Schramm, M., and A. Alsarhan, "Lightweight edge authentication for software defined networks", Computing 103, 291-311 (2021), August 2020, <https://link.springer.com/article/10.1007/s00607-020-00835-4>.

[CHEN] Chen, X., "Implementing AES Encryption on Programmable Switches via Scrambled Lookup Tables", In Proceedings of the Workshop on Secure Programmable Network Infrastructure, August 2020, <https://dl.acm.org/doi/abs/10.1145/3405669.3405819>.

[HAUSER-IPsec] Hauser, F., Häberle, M., Schmidt, M., and M. Menth, "P4-IPsec: Site-to-Site and Host-to-Site VPN With IPsec in P4-Based SDN", In IEEE Access, vol. 8, July 2020, <https://ieeexplore.ieee.org/document/9151942>.

[HAUSER-MACsec]
Hauser, F., Häberle, M., Schmidt, M., and M. Menth, "P4-MACsec: Dynamic Topology Monitoring and Data Layer Protection With MACsec in P4-Based SDN", In IEEE Access, vol. 8, March 2020, <https://ieeexplore.ieee.org/document/9044731>.

[I-D.draft-ietf-opsawg-mud-tls-05]
Reddy, T., Wing, D., and B. Anderson, "Manufacturer Usage Description (MUD) (D)TLS Profiles for IoT Devices", Work in Progress, Internet-Draft, draft-ietf-opsawg-mud-tls-05, 27 July 2021, <https://datatracker.ietf.org/doc/html/draft-ietf-opsawg-mud-tls-05>.

[I-D.draft-irtf-coinrg-use-cases-00]
Kunze, I., Wehrle, K., Trossen, D., and M.J. Montpetit, "Use Cases for In-Network Computing", Work in Progress, Internet-Draft, draft-irtf-coinrg-use-cases-00, 17 February 2021, <https://tools.ietf.org/html/draft-irtf-coinrg-use-cases-00>.

[I-D.draft-kutscher-coinrg-dir-02]
Kutscher, D., Karkkainen, T., and J. Ott, "Directions for Computing in the Network", Work in Progress, Internet-Draft, draft-kutscher-coinrg-dir-02, 31 July 2020, <https://datatracker.ietf.org/doc/html/draft-kutscher-coinrg-dir-02>.

[I-D.draft-lear-opsawg-mud-bw-profile-01]
Lear, E. and O. Friel, "Bandwidth Profiling Extensions for MUD", Work in Progress, Internet-Draft, draft-lear-opsawg-mud-bw-profile-01, 8 July 2019, <https://datatracker.ietf.org/doc/html/draft-lear-opsawg-mud-bw-profile-01>.

[KANG]
Kang, Q., Morrison, A., Tang, Y., Chen, A., and X. Luo, "Programmable In-Network Security for Context-aware BYOD Policies", In Proceedings of the 29th USENIX Security Symposium (USENIX Security 20), August 2020, <https://www.usenix.org/conference/usenixsecurity20/presentation/kang>.

[LEWIS]
Lewis, B., Broadbent, A., and N. Race, "P4ID: P4 Enhanced Intrusion Detection", 2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), November 2019, <https://ieeexplore.ieee.org/document/9040044>.

[MOGHADDAM]
Moghaddam, H. and A. Mosenia, "Programmable In-Network Obfuscation of Traffic", arXiv:1911.09642 [cs.CR], November 2019, <https://arxiv.org/abs/1911.09642>.

[RFC7744] Seitz, L., Ed., Gerdes, S., Ed., Selander, G., Mani, M., and S. Kumar, "Use Cases for Authentication and Authorization in Constrained Environments", RFC 7744, DOI 10.17487/RFC7744, January 2016, <https://www.rfc-editor.org/info/rfc7744>.

[RFC8520] Lear, E., Droms, R., and D. Romascanu, "Manufacturer Usage Description Specification", RFC 8520, DOI 10.17487/RFC8520, March 2019, <https://www.rfc-editor.org/info/rfc8520>.

[RFC8576] Garcia-Morchon, O., Kumar, S., and M. Sethi, "Internet of Things (IoT) Security: State of the Art and Challenges", RFC 8576, DOI 10.17487/RFC8576, April 2019, <https://www.rfc-editor.org/info/rfc8576>.

[SONCHACK] Sonchack, J., Aviv, A., Keller, E., and J. Smith, "Turboflow: Information Rich Flow Record Generation on Commodity Switches", In Proceedings of the Thirteenth EuroSys Conference, April 2018, <https://dl.acm.org/doi/abs/10.1145/3190508.3190558>.

[WANG] Wang, L., Kim, H., Mittal, P., and J. Rexford, "Programmable In-Network Obfuscation of Traffic", arXiv:2006.00097 [cs.NI], 2020, <https://arxiv.org/abs/2006.00097>.

[YOO] Yoo, S. and X. Chen, "Secure Keyed Hashing on Programmable Switches", In Proceedings of the ACM SIGCOMM 2021 Workshop on Secure Programmable Network INfrastructure, August 2021, <https://arxiv.org/abs/1911.09642>.

Authors' Addresses

Ina Berenice Fink
RWTH Aachen University
Ahornstr. 55
D-52062 Aachen
Germany

Phone: +49-241-80-21419
Email: fink@comsys.rwth-aachen.de

Klaus Wehrle
RWTH Aachen University
Ahornstr. 55
D-52062 Aachen
Germany

Phone: +49-241-80-21401
Email: wehrle@comsys.rwth-aachen.de