An Integrity Measurement Scheme for Containerized Virtual Network Function

Dazan Qian*, Songhui Guo¹, Lei Sun¹, Qianfang Hao¹, Yunfan Song¹, and Miao Wang¹

¹Information Engineering University, Zhengzhou, Henan, 450000, China
*Corresponding author’s e-mail: 1939018601@qq.com

Abstract. The deployment of virtual network function (VNF) in the container can realize the 5G service-based architecture (SBA) with high flexibility. The container carrying the VNF has poor isolation and low protection capabilities, and there is a security risk of being tampered and replaced. Current security protection technologies such as access control, intrusion detection, and virus detection cannot ensure that the container is not illegally modified. In order to fundamentally protect the integrity of containerized VNFs, this paper proposes a containerized VNF trust measurement scheme (CIM). The scheme extends the chain of trust to bare metal containers and virtual machine containers, and experiments are carried out in a containerized VNF communication environment. The results show that the integrity measurement protection scheme is effective. Compared with ordinary containers, the average CPU usage of trusted containers has increased by 26%, and the average memory usage growth rate is less than 1%, the performance overhead caused by CIM is acceptable.

1. Introduction

5G has three typical application scenarios: enhanced mobile broadband, support for massive terminal access, and ultra-high reliability and low latency. It can effectively empower the digital construction of the industry and become a new driving force for economic development. 3GPP pointed out that the security threats facing 5G are the main reason that restricts the rapid development and application of 5G. It is necessary to protect 5G from 17 aspects such as network slicing security, access network security and security architecture [1]. 5G uses network function virtualization (NFV) to replace traditional mobile communication network software and is deployed on general-purpose virtual hardware or cloud virtual platforms. The special security problems of 5G using NFV technology include the deletion of hardware domain data leakage, the escape of virtual machines at the virtualization layer, and the deletion, replacement, and modification of the virtual network element layer. The realization of 5G’s cloud-native microservice architecture requires the use of container technology. Compared with virtual machines, the 5G cloud-native communication network that uses containers to carry VNFs has flexible scheduling, low construction and operation time, and low economic costs. The advantages have received extensive attention from academia and industry. The container itself is not completely isolated and its security protection capabilities are insufficient. The current container security problems mainly include: (1) The container image and container are tampered with or replaced by attacks; (2) The isolation of the container and the host file directory is incomplete and illegal access; (3) The container escapes without permission due to software vulnerabilities or configuration errors; (4) Denial of service attacks, resulting in the inability to provide services normally; (5) The shared kernel has security vulnerabilities. Among them, the integrity of the container image and the running container are the basis for the security and...
trustworthiness of 5G cloud deployment of virtual network functions. Current security protection technologies such as access control, intrusion detection, and virus detection cannot ensure that the container is not illegally modified. So as to ensure the security of containerized VNFs, ultimately provide end users with stable and secure communication services. This paper proposes CIM, a trusted measurement scheme for container integrity measurement, which extends the chain of trust to bare metal containers and virtual machine containers, and fundamentally realizes the integrity protection of containerized VNFs.

2. Related Work

Container technology is a representative of operating system-level virtualization products. The current mainstream container technologies include Lxc, Docker, Rocket, OpenVZ and Warden [2]. As shown in Figure 1. Docker, a container technology with a wider range of applications, needs to share the core of the host, and relies on the kernel tool Cgroup to achieve resource limitations such as CPU and memory for a single container instance. Isolate process communication through Namespace, construct different file system mount points, and allocate virtual networks. Compared with KVM based on hardware-assisted virtualization, Docker-based container technology has the advantages of fast start up speed, low operating resource usage, and small image size.

![Figure 1. Comparison of virtual machine and Docker-based container architecture.](image)

In 2015, the European Telecommunications Standards Institute (ETSI) proposed and formulated the NFV standard reference architecture, which replaced the real network equipment in the traditional mobile communication network system with the VNF implemented by NFV technology. The NFV standard reference structure is shown in Figure 2. NFV infrastructure (NFVI) relies on the general hardware resources of cloud computing and can provide a platform and environment for the deployment, operation and management of VNF; The management framework NFV management and orchestrator (NFV MANO) is responsible for the installation, uninstallation, configuration, and monitoring life cycle of VNF; The operation and business support system operation support system/business support system (OSS/BSS) as a comprehensive management platform supports dynamic real-time management operations.
In view of the research on using a virtualization platform to carry VNFs under the 5G cloud communication network architecture, there are currently three deployment modes and forms, namely bare metal containers, virtual machine containers, and virtual machines. (1) The use of virtual machines has mature technology and high virtual machine security isolation, but it is not conducive to the update, maintenance, and scheduling of VNFs; (2) The use of bare metal containers to carry VNFs has the highest orchestration and scheduling, which is the ultimate form of carrying VNFs in the future, and has the highest manageability and scheduling. However, ETSI has not formulated relevant security standards and specifications, and there are other security risks such as kernel attacks and container escape; (3) The VNF carried by the virtual container balances security, flexibility and resource consumption; it is the current research direction and the solution for the actual deployment of mobile communication products. Combining the cutting-edge development direction, carry out measurement and protection research on the integrity of bare metal containers and virtual machine containerized VNFs.

From the Docker 1.3 version, the container image is hashed. Users can calculate the check value to check the integrity of the downloaded image to prevent the official image from being tampered with or replaced by an attack. The Docker 1.8 version encrypts the container image and uses digital signatures to ensure the source, provenance and authenticity of the image to prevent forgery. The security mechanism based on general cryptography to ensure the integrity of the container depends on the security of the key and the strength of the cryptographic algorithm, and is not absolute.

Trusted computing is based on the trusted hardware module of trust, and establishes a chain of trust from the root of security trust measurement in the trusted hardware module, basic hardware resource platform, basic input output system (BIOS), operating system, and user application software. Extend the measurement of the trust layer to establish an overall trusted environment. By comparing the expected measurement results to ensure the integrity, confidentiality and credibility of the corresponding behaviours, compared with other passive security protection schemes such as firewalls and intrusion detection, it can be Letter computing has natural active immunity characteristics. ETSI proposed a technology based on trusted computing for VNF security protection, but there is no specific solution to achieve it. Sultan et al. [3] proposed two scheme designs for protecting containers from attacks based on hardware trusted modules: using Virtualizing the Trusted Platform Module (vTPM) and Intel SGX as a trusted platform hardware support mechanism. However, there is no specific implementation, but a theoretical research suggestion that vTPM and Intel SGX can be used to measure containers. Wei [4] realized the security of VNF instances in the NFV architecture based on Intel SGX technology. This protection mechanism uses the SGX memory isolation and sealing features to isolate and protect the VNF instances running independently on the virtual machine to ensure the startup process and operation Time safety. However, the latest research shows that using low-voltage injection to attack Intel SGX will lead to privilege escalation and information leakage [5], which means that the security of Intel SGX itself needs to be improved. Sailer et al. [6] proposed a TPM-based integrity measurement architecture.
Juan et al. [7] in order to ensure the integrity of the container image, trusted computing was introduced into the production environment of the container, and the integrity of the image was verified by extending the chain of trust from the TPM chip to the container image before it was used. The physical TPM chip cannot meet the integrity requirements of the virtual machine container under the NFV architecture. Berger et al. [8] analyzed the theory of vTPM and completed the specific implementation, created and allocated available vTPM for the virtual machine in the cloud environment, and completed the integrity measurement of the virtual machine and the application program in the virtual machine. Hosseinzadeh [9] analyzed the current two vTPM architectures for virtual machines, and proposed two vTPM architecture design solutions for containers, as shown in Figure 3, this is a reference solution for the vTPM architecture of bare metal containers, but it has not been implemented. The reason is that the container shares the host kernel and cannot have a virtual machine monitor like a virtual machine.

Benedictis et al. [10] monitor the implementation of lightweight container cloud infrastructure, using remote authentication to verify the software integrity of the application services deployed in the container cloud throughout its life cycle, which can be used to monitor the host, container engine, and the running container performs integrity verification. Before completing data flow forwarding between VNF instances, the integrity of both parties needs to be verified. This problem can be solved by referring to the remote verification in [10]. Guo et al. [11] create a vTPM instance for the container, extend and divided multiple trust container chains, and remote users can determine whether to trust and use the container through the container status query. This is a vTPM solution based on bare metal containers, which requires a host computer Modification of the kernel does not have portability and is limited to a specific kernel. Wu et al. [12] proposed Container-IMA, divided the IMA Measurement List (ML), proposed a container platform configuration register container platform configuration Register (cPCR) mechanism, and used a hardware-based root of trust to protect each Based on this ML partition, the container integrity authentication mechanism is developed to avoid the leakage of other container metrics of the machine during the remote authentication process.

For containers in a general cloud environment, generally will not run continuously for a long time, and the container in the 5G communication network system implemented by NFV technology has the characteristic of providing services to the outside for a long time after a single startup. The trusted static integrity measurement based on the root of trust only measures the container carrying the VNF once at startup, and cannot guarantee the integrity of the container and the file in the container during the subsequent operation. The trust measurement system needs to be improved to realize the repeatability of the measurement in the process. At present, there are relatively few research results on dynamic measurement of containers. However, scholars at home and abroad have done a lot of research work on the trustworthy dynamic measurement of hosts and virtual machines. Ziwen et al. [13] proposed a trusted computing operating system dynamic measurement architecture, which can perform dynamic and real-time integrity measurement and monitoring of the active processes or modules in the system on demand. Be et al. [14] proposed a dynamic integrity measurement model based on hardware TPM to measure the integrity of the host operating system's secure computing platform during execution. Du et al [15]
proposed a dynamic integrity measurement model based on the virtual trusted platform module vTPM, and implemented it with a typical virtual machine monitor Xen as an example to ensure the security of the user domain in the cloud environment.

3. Frame Design
In response to the development of container research and integrity in the 5G cloud-based communication network and the actual security risks of integrity, CIM technical solution was designed, based on the TPM trusted measurement to meet the measurement requirements of bare metal containers, and based on the credibility of vTPM The measurement architecture meets the virtual container measurement requirements, and the static integrity measurement and dynamic integrity measurement algorithms ensure the validity and real-time performance of the measurement.

3.1. System construction
The design goal of the overall system is to use trusted computing technology to measure and protect the carried VNF virtual container, and to extend the trust measurement of the container and container image. In the Xen virtualization platform, the vTPM instance is created by the vTPM manager in the privileged virtual machine, and the vTPM information is leaked due to the excessive authority of the privileged domain [16]. Therefore, vTPM is implemented on a virtualization platform based on KVM-QEMU.

Enable IMA in each virtual machine and host, modify the corresponding measurement strategy to complete the static extension of the trust chain to the container, and achieve complete static measurement. The user invokes the TCG software stack (TSS) of the vTPM through the dynamic measurement module to initiate a measurement request for the container file. The dynamic measurement agent parses the dynamic measurement module request and transfers the measurement information to vTPM. The vTPM completes and stores the dynamic measurement result.

3.2. Trust chain establishment
As shown in Figure 5, when the host has a TPM module that has been configured, the starting point of control for system startup is the Root of Trust for Measurement (RTM) in the TPM security chip, which is measured level by level according to the chain of trust extension strategy, and the measurement results are saved on the platform Configuration register (PCR) and ML. During the measurement process, the host platform configuration information is stored in $PCR^0$ to $PCR^7$ according to the TCG metric standard specifications, extend the platform configuration metric value to $PCR^{10}$ by formula (1) to become $PCR_{old}^{10}$.

$$PCR_{old}^{10} = Hash(...Hash(PCR^0 || PCR^1) ... PCR^7)$$ (1)
Configuring the IMA policy in the host file ima_policy extends the trust chain to the vTPM function module and the bare metal container. The measurement result is displayed in the ML of the host. According to formula (2), the result of the $PCR_{old}^{10}$ measurement list and the ML are sequentially Hash extended to obtain a new stored value of $PCR_{new}^{10}$.

$$PCR_{new}^{10} = Hash \left( PCR_{old}^{10} || Hash( \text{measurement log} ) \right)$$ (2)

The TPM relies on the key in the root of trust (RTR) to sign and encrypt the measurement result and send it to the challenger. The challenger decrypts the results and compares them to determine whether the target platform is credible, and finally realizes the verification of the bare metal container carrying the VNF, the virtual machine container and the vTPM module. After the vTPM instance is generated, the VM is started, the measurement of the virtual machines SeaBIOS, Boot Loader, OS and container is completed in sequence, and the measurement results are stored in $vPCR^0$-$vPCR^7$. Modify the IMA measurement strategy in the virtual machine to complete the extension of the trust chain and realize the measurement of the files and data of the virtual machine. The same formula as the host is used (2) The platform configuration and measurement list are expanded, and the final measurement results are stored in $vPCR^{10}$. The challenger compares when the challenger challenges the credibility of the virtual machine container carrying the virtual network function, the virtual machine sends virtual ML (vML) and the $vPCR^{10}$ metric signed and encrypted with vTPM to the challenger, and the challenger decrypts the $vPCR^{10}$ value and sends it to the challenger. Comparing with vML, complete the challenge to platform credibility and virtual container integrity.
Figure 5. IMA-based container static measurement.

References [17] and [18] to design a dynamic measurement algorithm for the virtual machine container, and the actual dynamic agent module is vTPM. Call TSS in the dynamic measurement module to implement dynamic measurement. As shown in Figure 6, first check whether the system has completed static measurement. The static measurement result of the target file is stored in \( v_{PCR}^{15} \). When the user sends a dynamic measurement request according to his own needs, the dynamic measurement agent processes the request, completes the measurement and stores the obtained result in \( v_{PCR}^{15} \) through the Hash extension. The subsequent dynamic measurement is extended through the formula (3). The user maintains a file list for each Hash result, and compares the Hash list value of each container key program. Expanded and recalculated the measurement result with \( v_{PCR}^{15} \), and compared with \( v_{PCR}^{15} \) of vTPM to complete the dynamic measurement integrity verification.

\[
v_{PCR}^{16}_{new} = Hash(v_{PCR}^{15} || Hash(container))
\]  

(3)
4. Experimental Design and Analysis

The experimental verification environment is shown in Figure 7 below. The sub-project qemu-tpm of QEMU is used in the host to recompile. The virtual machine is an Ubuntu16.04 LTS virtual machine that allocates 2G of memory, the kernel is the default kernel, and the Docker container version is 18.09. Five containers with VNFs are deployed on bare metal and virtual machines respectively. The implementation of VNFs is the openair-epc-fed branch of the core network of the wireless communication experimental platform project OpenAirInterface (OAI). These five network elements can be used to achieve complete 4G core network. Cassandra is a distributed database that stores user data information. Home subscriber server (HSS) manages Cassandra information, mobility management user authentication management entity mobility management entity (MME) realizes the management of end users; serving gateway (SGW) user plane related functions Network element; packet data network gateway (PGW) access service.
The principles of bare metal container VNF and virtual machine container VNF are the same. Taking bare metal container as an example, we will introduce how five containerized VNFs form a 4G core network. Start the HSS, load the database configuration files to update the terminal equipment subscription information, start the connection between the MME and the HSS, complete the connection initialization of the control plane. Then run the user plane management network element SGW and network element PGW to initialize the control plane. After the 5 VNFs are connected, they wait for the end user to complete the identity verification through the access network and provide the network connection service.

The static integrity measurement of bare metal containers uses tpm-emulator 0.7 to simulate the host hardware TPM. After configuring the IMA policy, the measurement results of the containerized VNF and vTPM are extended to $PCR^{10}$. The virtual machine container is statically measured, a vTPM instance is created on the host, the TPM 1.2 function is selected, and the path UnixIO connection is specified. When starting the virtual machine, complete the binding of the vTPM instance to the virtual machine.

The static measurement results of the virtual machine container are shown in Table 1. The measurement strategy completes the measurement of the containerized VNF file, and the result is stored in $PCR^{10}$ through extension. By adding content to the container image file of the carrier network element HSS, the result is that $vPCR^{10}$ is updated immediately, the static measurement is valid.

Install trousers 0.3.15 in the virtual machine and use TSS to perform dynamic measurement test on the container's key program runc. The first Hash measurement result is expanded and stored in $vPCR^{15}$ and $vPCR^{16}$. As shown in Table 2. When the runc file is directly replaced, the measurement result of $vPCR^{16}$ changes. The measurement is valid.

| Reference value | Results after tampering |
|-----------------|-------------------------|
| $vPCR^{0}$      | 3a3f780flla4b49969fcaaa80cd6e3957c33b22 | 3a3f780flla4b49969fcaaa80cd6e3957c33b22 |
| $vPCR^{1}$      | f42aba688169658685c385d2df6e9bd6606b9a | f42aba688169658685c385d2df6e9bd6606b9a |
| $vPCR^{2}$      | 3dlbe3022180e26e82200d765dc44f8e659a7b | 3dlbe3022180e26e82200d765dc44f8e659a7b |
| $vPCR^{3}$      | 3a3f780flla4b49969fcaaa80cd6e3957c33b22 | 3a3f780flla4b49969fcaaa80cd6e3957c33b22 |
| $vPCR^{4}$      | 59c3d75d40ff471c487abf8e48f5ac2656e4e6e | 59c3d75d40ff471c487abf8e48f5ac2656e4e6e |
Table 2. Dynamic measurement results.

| Reference value | Results after tampering |
|-----------------|-------------------------|
| vPCR^1          | 7200b916c49f20dc5ce94b2b598e30ff635e7148 | 7200b916c49f20dc5ce94b2b598e30ff635e7148 |
| vPCR^10         | fcdcade3226e0bb4f0a2e9d8af409107cfa808 | fcdcade3226e0bb4f0a2e9d8af409107cfa808 |

Analyze the performance of the container after using the CIM measurement scheme. Take the virtual machine container as an example, use docker stats to collect 5 types of containerized VNFs that have no protection, only static measurement, static measurement and once per minute dynamic measurement coexist 11 times of CPU usage and memory usage under three types of situations, and the results are shown in Figure 8. Define formula 4 to get the average growth rate of CPU usage

$$\frac{\sum_{i=0}^{5} \sum_{j=0}^{11} VNF_{sd} - CPU}{\sum_{i=0}^{5} \sum_{j=0}^{11} VNF_{no} - CPU} - 1 \times 100\%$$

(a) CPU occupancy rate of the network element PGW.

(b) CPU occupancy rate of the network function SGW.
(c) CPU occupancy rate of the network function MME.

(d) CPU occupancy rate of the network function HSS.

(e) CPU occupancy rate of the network function Cassandra.

Figure 8. CPU occupancy rate of different network functions in three situations.

The memory performance results are shown in Figure 9. Formula 5 is used to calculate the memory growth rate. \( M_{\text{Memory}} \). \( VNF_{sd} - M \) is the memory usage of the 5 types of containerized VNFs in static and dynamic measurement, \( VNF_{no} - M \) is the memory usage of the 5 types of containerized VNFs without measurement protection. As a result, \( M_{\text{Memory}} \) is less than 1%, and the increased CPU performance loss is within the acceptable range.

\[
M_{\text{Memory}} = \left( \frac{\sum_{i=0}^{n} VNF_{sd} - M}{\sum_{i=0}^{n} VNF_{no} - M} - 1 \right) \times 100% \tag{5}
\]

Integrity measurement of virtual machine container based on CIM, the increased performance consumption comes from the Hash operation, and the resource consumed comes from the virtual machine, so the performance of the virtual machine container is less affected.
(a) Memory usage of network function PGW.

(b) Memory usage of network function SGW.

(c) Memory usage of network function MME.

(d) Memory usage of network function HSS.
5. Conclusion

Aiming at the integrity protection problem of content tampering and replacement in the VNF deployment in the container in the 5G scenario, this paper designs a CIM-based trust measurement scheme. The OAI-based 4G core network communication network functions are built in bare metal containers and virtual machine containers. By configuring vTPM instances for virtual machine containers and TPM instances for bare metal containers, the chain of trust is expanded to protect the static measurement of containerized network functions. In view of the long-term uninterrupted operation of containerized VNFs, a dynamic measurement scheme was designed. After experimental verification, CIM can effectively protect containerized VNF, and the performance overhead added by CIM is low. How to build trust chains for different containers carrying different VNFs and reduce the information privacy leakage of the platform and other containers during remote attestation is our next work goal.

References

[1] 3GPP. (2018) 3GPP AS3 Meeting Report for TSG SA. https://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_92Bis_Harbin/Report.

[2] Martin A, Raponi S, Combe T, et al. (2018) Docker ecosystem - vulnerability analysis. Computer Communications, 122(JUN.):30–43.

[3] Sultan S, Ahmad I, Dimitriou T. (2019) Containers’ security: Issues, challenges, and road ahead. IEEE Access. vol. 7, pp. 52976-52996.

[4] Hu Wei. (2017) Research on function security protection mechanism of virtual network based on SGX.

[5] CVE-2019-11157 (2019). http://cve.mitre.org/cgi-bin/cvename.cgi?name=2019-11157.

[6] Sailer R, Zhang X, Jaeger T, et al. (2014) Design and implementation of a tcg-based integrity measurement architecture. In Proceedings of the 13th USENIX Security Symposium.

[7] Juan Wang, Wei Hu, Yu han Zhang, et al. (2017) Trusted container based on docker. Journal of Wuhan University. 102–108.

[8] Berger S, Caceres R, Goldman K A, et al. (2006) vTPM:virtualizing the trusted platform module. Usenix Security, 15:305–320.

[9] Hosseinzadeh S, Lauren S, Leppnen Ville. (2017) Security in container-based virtualization through vtpm.

[10] Benedicts M De, Lioy A. (2019) Integrity verification of docker containers for a lightweight cloud environment. Future generation computer systems,97(AUG.):236–246.

[11] Guo Y, Yu A, Gong X, Zhao L, and Meng D. (2019) Building trust in container environment. In 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (Trust-Com/BigDataSE).
[12] Wu Luo, Yutang Shen, Qingni Xia, and Zhonghai Wu. (2019) Container-ima:a privacy-preserving integrity measurement architecture for containers. In 22nd International Symposium on Research in Attacks, Intrusions and Defenses, pages 487–500. USENIX Association.

[13] Feng Denggguo, Liu Ziwen. (2010) Dynamic integrity measurement architecture based on trusted computing. 875–879.

[14] Y. Li, H. Ba, and J. Ren. (2015) Integrity measurement based on trusted computing.

[15] Tian J, Du R, Pan W. (2018) Dynamic integrity measurement model based on vtpm. 88–99.

[16] Xin J, Wang L, Yu R, Peng K, and Shen C. (2010) Administrative domain: Security enhancement for virtual tpm. In International Conference on Multimedia Information Networking and Security.

[17] Hongmei Shu. (2019) Research on trusted web microservice based on unikernel.

[18] Tan Liang, Xiao Huan. (2019) Trustiness protection scheme of virtual currency hot wallet. 1801–1808