RATS
Internet-Draft
Intended status: Informational
Expires: 6 November 2026

Y. Deshpande
Arm Ltd
J. Zhang
H. Labiod
Huawei Technologies France S.A.S.U.
H. Birkholtz
Fraunhofer SIT

5 May 2026

Remote Attestation with Multiple Verifiers
draft-ietf-rats-multi-verifier-00

Abstract

IETF RATS Architecture, defines the key role of a Verifier. In a complex system, this role needs to be performed by multiple Verfiers coordinating together to assess the full trustworthiness of an Attester. This document focuses on various topological patterns for a multiple Verifier system. It only covers the architectural aspects introduced by the Multi Verifier concept, which is neutral with regard to specific wire formats, encoding, transport mechanisms, or processing details.

Table of Contents

1. Introduction

A Verifier plays a central role in any Remote Attestation System. A Verifier appraises the Attester and produces Attestation Results, which are essentially a verdict of attestation. The results are consumed by the Relying Party to conclude the trustworthiness of the Attester, before making any critical decisions about the Attester, such as admitting it to the network or releasing confidential resources to it. Attesters can come in wide varieties of shape and form. For example Attesters can be endpoints (edge or IoT devices) or complex machines in the cloud. Composite Attester Section 4.1, generate Evidence that consists of multiple parts. For example, in data center servers, it is not uncommon for separate attesting environments (AE) to serve a subsection of the entire machine. One AE might measure and attest to what was booted on the main CPU, while another AE might measure and attest to what was booted machine's GPU. Throughout this document we use the term Component Attester Section 4.1 to address the sub-entity or an individual layer which produces its own Evidence in a Composite Attester system.

In a Composite Attester system, it may not be possible for a single Verifier to possess all the capabilities or information required to conduct a complete appraisal of the Attester. Please refer to Section 2 for motivation of this document. Multiple Verifiers need to collaborate to reach a conclusion on the appraisal and produce the Attestation Results.

This document describes various topological patterns of multiple Verifiers that work in a coordinated manner to conduct appraisal of a Composite Attester to produce an Attestation Results.

2. Need for Multiple Verifiers

To conduct the task of Evidence appraisal, a Verifier requires:

1. Reference Values from trusted supply chain actors producing, aggregating, or administering Attesters (Reference Value Providers)

2. Endorsements from trusted supply chain actors producing, certifying, or compliance checking Attesters (Endorsers)

3. Appraisal Policy for Evidence, which is under the control of the Verifier Owner

The Verifier inputs listed above are linked to the shape of the Attesters. Typically, Composite Attesters come with a varying degree of heterogeneity of Evidence formats, depending on the type of

Attesting Environments that come with each Component Attester, for example, CPU variants or GPU/FPGA variants. When conducting Evidence appraisal for a Composite Attester, the following challenges remain:

1. An Attester’s composition can change over time based on market requirements and availability (e.g., a set of racks in a data center gets thousands of new FPGAs). It is highly unlikely that there is always one appropriate Verifier that satisfies all the requirements that a complex and changing Composite Attesters imposes. It may not be economically viable to build and maintain such a degree of complexity in a single Verifier.

2. A Verifier Owner may have an Appraisal Policy for Evidence of a Component Attester that is internal to them and which they may choose not to reveal to a “monolithic" Verifier.

3. A Reference Values Provider may not wish to reveal its Reference Values or their lifecycle to a monolithic Verifier.

4. There may not be a single actor in the ecosystem that can stand up and take ownership of verifying every Component Attester due to a lack of knowledge, complexity, regulations or associated cost.

5. The mix today is a combination of Verifier services provided by component manufacturers, Verifiers provided by integrators, and Verifiers under local authority (i.e., close to the attester). Rarely is it just one of these.

3. Reference Use Cases

This section covers generic use cases that demonstrate the applicability of Multi Verifier, regardless of specific solutions. Its purpose is to motivate various aspects of the architecture presented in this document. There are many other use cases; this document does not contain a complete list.

3.1. Verification of Devices containing heterogenous components

A device may contain a central processing unit (CPU), as well as heterogeneous acceleration components (such as GPUs, NPUs and TPUs) from different suppliers.

These components can be used to speed up processing or assist with AI inference. Trustworthiness assessment of the device requires trust in all of these components. However, due to business concerns such as scalability, complexity and cost of infrastructure, the Verifier for each type of component may be deployed separately by each vendor.

When these Verifiers operate together, they must interact with each other, understand the topology and interoperate using standardised protocols. For instance, they may need to exchange partial Evidence relating to the relevant component or partial Attestation Results for it.

Attester: A Device having multiple components

Relying Party: An entity which is making trust decisions for such an Attester

3.2. Verification of Workloads operating in Confidential Computing environment

As organisations move more workloads into untrusted or shared environments, Confidential Computing is becoming increasingly important. In such a system, an application or workload (which could be an AI model, database process or financial service, for example) is executed inside a TEE-protected virtual machine (VM). When the workload starts, the TEE can generate a cryptographic attestation report providing:

1. The workload is running on a platform with a known state.

2. The workload is running the correct application.

The platform is often built by an independent TEE vendor, while the workloads are deployed by workload owners from different parts of the supply chain.

Verification of Attestation for such a system requires independent, yet mutually coordinated, verification of: Platform claims appraised by a Platform Verifier and Workload claims appraised by a Workload Verifier.

Attester: A layered Attester containing a platform and a workload running in a CC environment

Relying Party: An entity which is making trust decisions, such as a key release to a workload

4. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

This document uses terms and concepts defined by the RATS architecture. For a complete glossary, see Section 4 of [RFC9334].

Specifically this document heavily uses the terms Layered Attester Section 3.2 of [RFC9334] and Composite Device Section 3.3 of [RFC9334]

4.1. Glossary

This document uses the following terms:

Composite Attester: A Composite Attester is either a Composite Device or a Layered Attester or any composition involving a combination of one or more Composite Devices or Layered Attesters.

Component Attester: A Component Attester is a single Attester of a Composite Attester. For this document, a Component Attester is an entity which produces a single Evidence which can be appraised by a Component Verifier.

Composite Evidence: Evidence produced by a Composite Attester. Also referred to as CE in the document.

Partial Evidence: It is an extract from a Composite Evidence. It consists of at least one or more Component Evidence. Also referred to as PE in the document.

Lead Verifier: A Verifier which acts as a main Verifier to receive Composite Evidence from a Composite Attester in a Hierarchical pattern Section 5.1. Also referred to as LV in the document.

Component Verifier: A Verifier which is responsible for the Verification of one single component or a layer. Also referred to as CV in the document.

Partial Attestation Results: Attestation Results produced by a Component Verifier, which contains partial results from atleast one or more Component Attesters.

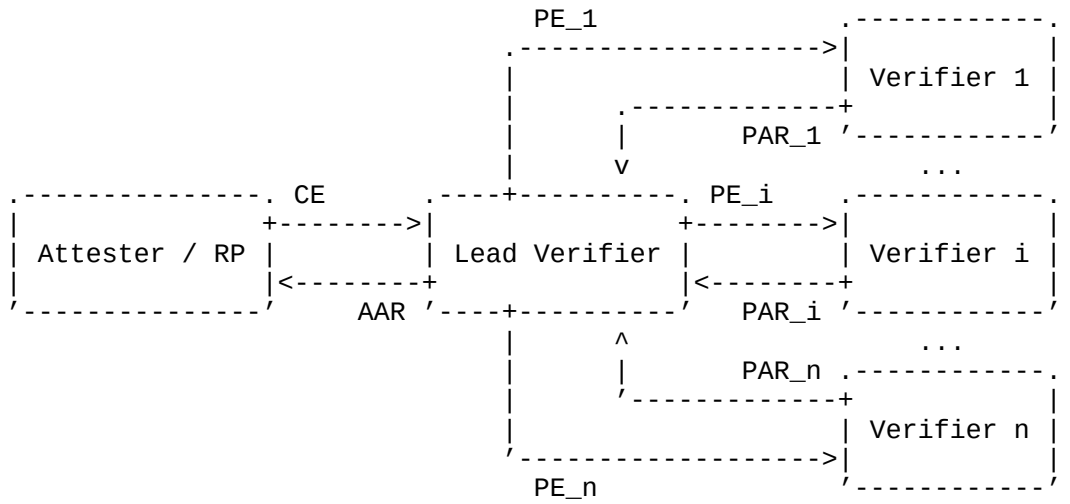
Aggregated Attestation Results: An Aggregated Attestation Results (AAR) refers to a collection of Attestation Results produced upon completion of appraisal of a Composite Attester.

5. Multi Verifier topological patterns

A Composite Attester has multiple Component Attesters. Each Attester requires a different set of Verifiers. Hence multiple Verifiers collaborate to appraise a Composite Attester.

5.1. Hierarchical Pattern

Figure below shows the block diagram of a Hierarchical Pattern.



Legend:
- CE: Composite Evidence
- AAR: Aggregated Attestation Results
- PE_i: Partial Evidence of i-th Component Attester
- PAR: Partial Attestation Results

Figure 1: Hierarchical Pattern

The following sub-sections describe the various roles that exist in this pattern.

5.1.1. Lead Verifier

In this topological pattern, there is an Entity known as Lead Verifier.

Lead Verifier is the central entity in communication with the Attester (directly in passport model or indirectly via the Relying Party in background-check model). It receives Attestation Evidence from a Composite Attester. If the Composite Attestation Evidence is signed, then it validates the integrity of the Evidence by validating the signature. If signature verification fails, the Verification is terminated. Otherwise it performs the following steps.

* Lead Verifier has the required knowledge to break down the Composite Evidence into Partial Evidence. It decodes the Composite Evidence to extract the Component Attesters Evidence. This may lead to "N" Partial Evidence, one for each Component Attester.

* Lead Verifier delegates each Partial Evidence to its own Component Verifier (CV) and receives Component Attester Attestation Results also known as Partial Attestation Results after successful Appraisal of Evidence. There are many protocols to determine how a Lead Verifier can select the Component Verifiers. This document does not mandate any specific protocol for determining the Component Verifiers

* Once the Lead Verifier receives Partial Attestation Results from all the Verifiers, it combines the results from each Verifier to construct an Aggregated Attestation Results (AAR). The Lead verifier may apply its own policies and also add extra claims as part of its appraisal.

* Lead Verifier conveys the AAR to the Attester (in Passport model) or to the Relying Party (in background check model).

The overall verdict may be dependent on the Appraisal Policy of the Lead Verifier.

In certain topologies, it is possible that only the Composite Evidence is signed to provide the overall integrity, while the Partial Evidence (example PE_1) is not protected. In such cases, the Lead Verifer upon processing of Composite Evidence may wrap the Partial Evidence (example PE_1) in a signed Conceptual Message Wrapper (CMW), and send it to each Verifier (example Verifier 1).

5.1.2. Component Verifier

The role of a Component Verifier is to receive Partial Evidence from the Lead Verifier and produce Partial Attestation Results to the Lead Verifier.

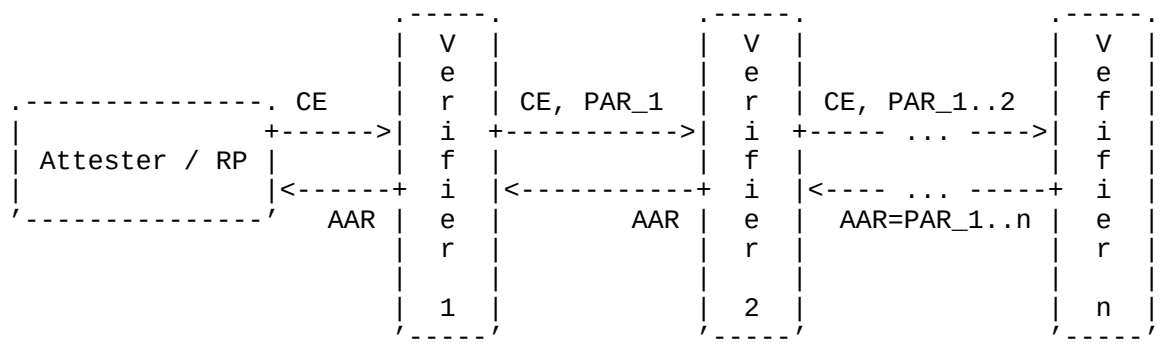
5.1.3. Trust Relationships

In this topology the Lead Verifier is fully trusted by Component Verifiers (example Verifier 1). Each Component Verifiers are provisioned with the Trust Anchors (see [RFC6024]) for the Lead Verifier.

Also, each of the Component Verifier is fully trusted by the Lead Verifier. Lead Verifier is provisioned with the Trust Anchors (see [RFC6024]) for Verifier 1..N.

5.2. Cascaded Pattern

Figure below shows the block diagram of a Cascaded Pattern.



Legend:
- CE: Composite Evidence
- AAR: Aggregated Attestation Results
- PAR: Partial Attestation Results

Figure 2: Cascaded Pattern

In this topological pattern, the Attestation Verification happens in sequence. Verifiers are cascaded to perform the Attestation Appraisal. Each Verifier in the chain has the knowledge to derive or extract the Partial Evidence, which it can appraise, from the Composite Evidence.

Attester may send the Composite Evidence(CE) to any of the Verifier (directly in the passport model, or indirectly via the Relying Party in the background-check model). The Verifier which processes the Composite Evidence, Verifies the signature on the Evidence, if present. It extracts the Partial Evidence from the Composite Evidence, performs Appraisal of the Component Attester whose Reference Values and Endorsements are in its database. Once the appraisal is complete, it forwards the Composite Evidence and Partial Attestation Results to the subsequent Verifier.

The process is repeated, until the entire appraisal is complete. The last Verifier, i.e. Verifier-N, completes its Appraisal of the Partial Evidence, that it can appraise. It has now all the Partial Attestation Results and creates the Aggregated Attestation

Results(AAR). It returns the AAR to the N-1 Verifier (from where it received the Composite Evidence and Partial AR). The process is repeated, i.e. AAR is returned in the chain until the Verifier, which recieved the initial Composite Evidence is reached. At this point in time the Aggregated Attestation Results are signed and the AAR is sent to the Attester (in Passport Model) or Relying Party (in background check model).

As shown in the picture, the Partial Attestation Results and Composite Evidence is transmitted to a chain of Verifier, till the Appraisal is complete. Upon completion, the last Verifier in the chain combines the incoming Partial Attestation Results, combines the results from it own Evidence Appraisal and passes the Aggregated Attestation Results to the Verifier from which it receives Composite Evidence.

There are many protocols to determine how a Verifier can select the next Verifier to route the CE and PAR. This document does not mandate any specific protocol for determining the Verifiers in cascade.

5.2.1. Trust Relationships

5.2.2. Verifiers

In the cascaded pattern, the communicating Verifiers fully trust each other. Each Verifier has the trust anchor for the Verifier it is communicating to (i.e. either sending information or receiving information). This prevents man in the middle attack for the Partial Attestation Results received by a Verifier or a Aggregated Attestation Results (AAR) which it receives in the return path.

5.2.3. Relying Party and Verifiers

In the cascaded pattern, the RP may communicate with any Verifier and thus receive its Attestation Results. Hence RP fully trusts all the Verifiers.

5.3. Hybrid Pattern

In a particular deployment, there is a possibility that the two models presented above can be combined to produce a hybrid pattern. For example Verifier 2 in the Cascaded Pattern becomes the Lead Verifier for the remaining Verifers from 3, to N.

6. Freshness

The Verifier needs to ensure that the claims included in the Evidence reflect the latest state of the Attester. As per RATS Architecture, the recommended freshness is ascertained using either Synchronised Clocks, Epoch IDs, or nonce, embedded in the Evidence. In the case of Hierarchical Pattern, the Verification of Freshness should be checked by the Lead Verifier.

In the Cascaded Pattern, the freshness is always checked by the first Verifier in communication with either the Attester (Passport Model) or Relying Party (Background Check Model).

7. Security Considerations

The Verifier is not part of the Attester's Trusted Computing Base (TCB), but acts as a critical component in the Relying Party's trust decision chain. Therefore, its security directly affects the reliability of the entire remote attestation process. When multiple Verifiers coordinate to conduct an appraisal, this may increase the attack surface, depending on the system architecture and trust assumptions.

Any mistake in the appraisal procedure conducted by one or more Verifiers could lead to severe security implications, such as incorrect Attestation Result of a component or a composition to the Relying party. This section details the security threats and mitigation strategies specific to the multi-verifier topologies described in this document. In addition to the considerations herein, Verifiers MUST follow the guidance detailed in the Security and Privacy considerations of a RATS Verifier as detailed in Section 11 of [I-D.draft-ietf-rats-corim] and the RATS Architecture Section 11 and Section 12 of [RFC9334].

7.1. Adversarial Model

The security analysis in this section assumes that attackers may:

1. Eavesdrop on any communication channel between Verifiers.

2. Inject, modify, replay, or delay messages traversing the network.

3. Compromise one or more Verifiers in the ecosystem, attempting to leak sensitive information (e.g., Evidence, Reference Values) or manipulate Attestation Results.

4. Perform Man-in-the-Middle (MitM) attacks between any two communicating entities.

The system is designed to be resilient under the assumption that the cryptographic keys used for signing Evidence and Attestation Results (by authentic entities) are not compromised.

7.2. General Considerations

All communications between entities (Attester-Verifier, Verifier-Verifier, Verifier-RP) MUST be secured using mutually authenticated, confidential, and integrity-protected channels (e.g., TLS).

It is recommended that any two verifiers establishing a communication channel perform mutual attestation before exchanging any attestation messages.

7.3. Security for Topological Patterns

7.3.1. Hierarchical Pattern

The hierarchical pattern introduces a central trust entity, the Lead Verifier (LV). The security of the entire system relies on the integrity and correct operation of the LV.

7.3.1.1. Threats and Mitigations

7.3.1.1.1. LV Compromise

Threat: A compromised LV can orchestrate attacks, such as approving malicious attestations, wrongly aggregating attestation results or leaking sensitive evidence. This is a single point of failure from a trust perspective.

Mitigation: The LV MUST be hardened and operate and store its Keys in a secure environment. Its operation SHOULD be auditable. Component Verifiers should be made available suitable trust anchors so that they can establish required trust in the authority of the LV.

7.3.1.1.2. Communication Security (LV <-> CV)

Threat: Eavesdropping or manipulation of evidence/results in transit.

Mitigation: All communications between the LV and CVs MUST be mutually authenticated and confidential (e.g., using TLS with client authentication). This ensures integrity, confidentiality, and authenticity of the messages exchanged between the Verifiers.

7.3.1.1.3. Evidence Integrity and Origin Authentication (LV -> CV)

Threat: The LV could forward manipulated evidence to a CV, or an attacker could inject fake evidence.

Mitigation: The conceptual message containing the Partial Evidence MUST be integrity-protected and authenticated. If the Partial Evidence is natively signed by the Component Attester at origin, the CV can verify it directly. If the Partial Evidence lacks inherent signatures (e.g., in UCCS), the LV MUST sign the Partial Evidence using a key that the CV trusts. This prevents any on-path attacker from altering the Partial Evidence.

7.3.1.1.4. Results Integrity and Origin Authentication (CV -> LV)

Threat: Partial Attestation Results could be manipulated in transit or forged by a malicious CV.

Mitigation: Each Partial Attestation Result MUST be digitally signed by the CV. LV should maintain a list of trust anchors for the CV's it communicates with. The LV MUST validate the signature using the required trust anchor for the CV, before adding the Partial Attestation Results to the Aggregated Attestation Results.

7.3.1.1.5. Replay Attacks

Threat: An adversary Component Verifier replays old Evidence or Attestation Results.

Mitigation: The LV is responsible for enforcing freshness (via nonces, epochs, or timestamps). This freshness value MUST be propagated to CVs and back to the LV, to ensure final AR can be validated against the original challenge.

7.3.2. Cascaded Pattern

The cascaded pattern distributes trust but requires each Verifier in the chain to be trusted to correctly handle and forward Attestation messages. The chain's security is only as strong as its weakest link.

7.3.2.1. Threats and Mitigations

7.3.2.1.1. Verifier Compromise

Threat: Any compromised Verifier in the chain can block, delay, or manipulate the attestation process. It can inject false partial results, drop evidence, or leak sensitive information.

Mitigation: Relying Parties and Verifiers MUST be configured with strict trust policies defining the allowed paths and trusted Verifiers. Operations should be logged for auditability.

7.3.2.1.2. Communication Security

Threat: Eavesdropping or manipulation of evidence and results between Verifiers.

Mitigation: Each hop between Verifiers MUST be secured with mutually authenticated and confidential channels (e.g., TLS with client authentication).

7.3.2.1.3. Evidence and Results Protection

Threat: Lack of end-to-end security allows intermediate Verifiers to manipulate evidence or results that are not intended for them to appraise.

Mitigation: End-to-end integrity protection is RECOMMENDED. The Composite Evidence should be signed by the Attester. Partial and Aggregated Attestation Results SHOULD be signed by the Verifier that generated them. This allows subsequent Verifiers and the Relying Party to verify that results have not been tampered with by intermediate nodes.

7.3.2.1.4. Replay Attacks

Threat: An adversary replays old Evidence or Attestation Results.

Mitigation: The first Verifier in the chain (the one receiving evidence from the Attester/RP) is responsible for enforcing freshness (via nonces, epochs, or timestamps) for the entire cascade. This freshness value MUST be propagated with the Evidence and Results through the chain so the final AR can be validated against the original challenge.

7.3.3. Security of the Hybrid Pattern

As the hybrid pattern is the composition of hierarchical pattern and cascade pattern, all the threats and mitigations that are applicable for these two patterns are also applicable for the general hybrid pattern.

8. Privacy Considerations

The appraisal of a Composite Attester requires exchange of attestation related messages, for example, Partial Evidence and Partial Attestation Results, among multiple Verifiers. This can potentially leak sensitive information about the Attester's configuration, identities and the nature of composition.

However, when carefully designed, a multi-verifier architecture can actually mitigate these privacy concerns. By distributing appraisal responsibilities and ensuring that no single Verifier has access to the full set of Evidence, the risk of comprehensive device profiling or tracking is reduced.

Nonetheless, such benefits depend on strong implementation practices.

* Minimization: Attesters should only generate Evidence that is strictly necessary for the appraisal policy. Verifiers should only request necessary claims.

* Confidentiality: Evidence containing sensitive information should be encrypted so that it can only be accessed by the intended Verifier and not by any unauthorised parties (including other Verifiers in the hierarchy, cascade or hybrid pattern). This is crucial in multi-tenant environments.

* Policy Handling: Verifiers should be careful not to leak their internal appraisal policies (e.g., through error messages or timing side channels) when communicating with other Verifiers or Attesters, as this information could be exploited by an attacker to manipulate appraisal.

9. IANA Considerations

Acknowledgments

The authors would like to thank Simon Frost and Usama Sardar for their reviews and suggestions.

References

Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

Informative References

[I-D.draft-ietf-rats-corim] Birkholz, H., Fossati, T., Deshpande, Y., Smith, N., and W. Pan, "Concise Reference Integrity Manifest", Work in Progress, Internet-Draft, draft-ietf-rats-corim-10, 2 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-corim-10>.

[RFC6024] Reddy, R. and C. Wallace, "Trust Anchor Management Requirements", RFC 6024, DOI 10.17487/RFC6024, October 2010, <https://www.rfc-editor.org/rfc/rfc6024>.

[RFC9334] Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote ATtestation procedureS (RATS) Architecture", RFC 9334, DOI 10.17487/RFC9334, January 2023, <https://www.rfc-editor.org/rfc/rfc9334>.

Contributors

Thomas Fossati
Linaro
Email: Thomas.Fossati@linaro.org

Thanassis Giannetsos
UBITECH Ltd.
Email: agiannetsos@ubitech.eu

Steven Bellock
NVIDIA
Email: sbellock@nvidia.com

Ghada Arfaoui
ORANGE
Email: ghada.arfaoui@orange.com

Authors’ Addresses

Yogesh Deshpande
Arm Ltd

Email: yogesh.deshpande@arm.com

Jun Zhang
Huawei Technologies France S.A.S.U.
Email: junzhang1@huawei.com

Houda Labiod
Huawei Technologies France S.A.S.U.
Email: houda.labiod@huawei.com

Henk Birkholtz
Fraunhofer SIT
Email: henk.birkholz@sit.fraunhofer.de