ACE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 13 September 2023

G. Selander
Ericsson AB
S. Raza
RISE
M. Furuhed
Nexus
M. Vuini
Inria
T. Claeys

12 March 2023

Protecting EST Payloads with OSCORE
draft-selander-ace-coap-est-oscore-06

Abstract

This document specifies public-key certificate enrollment procedures protected with lightweight application-layer security protocols suitable for Internet of Things (IoT) deployments. The protocols leverage payload formats defined in Enrollment over Secure Transport (EST) and existing IoT standards including the Constrained Application Protocol (CoAP), Concise Binary Object Representation (CBOR) and the CBOR Object Signing and Encryption (COSE) format.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Authentication and Authorization for Constrained Environments Working Group mailing list (ace@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/ace/.

Source for this draft and an issue tracker can be found at https://github.com/EricssonResearch/EST-OSCORE.

Status of This Memo

Table of Contents

1. Introduction

One of the challenges with deploying a Public Key Infrastructure (PKI) for the Internet of Things (IoT) is certificate enrollment, because existing enrollment protocols are not optimized for constrained environments [RFC7228].

One optimization of certificate enrollment targeting IoT deployments is specified in EST-coaps ([RFC9148]), which defines a version of Enrollment over Secure Transport [RFC7030] for transporting EST payloads over CoAP [RFC7252] and DTLS [RFC6347], instead of secured HTTP.

This document describes a method for protecting EST payloads over CoAP or HTTP with OSCORE [RFC8613]. OSCORE specifies an extension to CoAP which protects the application layer message and can be applied independently of how CoAP messages are transported. OSCORE can also be applied to CoAP-mappable HTTP which enables end-to-end security for mixed CoAP and HTTP transfer of application layer data. Hence EST payloads can be protected end-to-end independent of underlying transport and through proxies translating between between CoAP and HTTP.

OSCORE is designed for constrained environments, building on IoT standards such as CoAP, CBOR [RFC7049] and COSE [RFC8152], and has in particular gained traction in settings where message sizes and the number of exchanged messages needs to be kept at a minimum, such as 6TiSCH [RFC9031], or for securing multicast CoAP messages [I-D.ietf-core-oscore-groupcomm]. Where OSCORE is implemented and used for communication security, the reuse of OSCORE for other purposes, such as enrollment, reduces the code footprint.

In order to protect certificate enrollment with OSCORE, the necessary keying material (notably, the OSCORE Master Secret, see [RFC8613]) needs to be established between EST-oscore client and EST-oscore server. For this purpose we assume by default the use of the lightweight authenticated key exchange protocol EDHOC [I-D.ietf-lake-edhoc], although pre-shared OSCORE keying material would also be an option.

Other ways to optimize the performance of certificate enrollment and certificate based authentication described in this draft include the use of:

* Compact representations of X.509 certificates (see [I-D.ietf-cose-cbor-encoded-cert])

* Certificates by reference (see [I-D.ietf-cose-x509])

* Compact, CBOR representations of EST payloads (see [I-D.ietf-cose-cbor-encoded-cert])

1.1. Operational Differences with EST-coaps

The protection of EST payloads defined in this document builds on EST-coaps [RFC9148] but transport layer security is replaced, or complemented, by protection of the transfer- and application layer data (i.e., CoAP message fields and payload). This specification deviates from EST-coaps in the following respects:

* The DTLS record layer is replaced, or complemented, with OSCORE.

* The DTLS handshake is replaced, or complemented, with the lightweight authenticated key exchange protocol EDHOC [I-D.ietf-lake-edhoc], and makes use of the following features:

  - Authentication based on certificates is complemented with authentication based on raw public keys.

  - Authentication based on signature keys is complemented with authentication based on static Diffie-Hellman keys, for certificates/raw public keys.

  - Authentication based on certificate by value is complemented with authentication based on certificate/raw public keys by reference.

* The EST payloads protected by OSCORE can be proxied between constrained networks supporting CoAP/CoAPs and non-constrained networks supporting HTTP/HTTPs with a CoAP-HTTP proxy protection without any security processing in the proxy (see Section 5). The concept "Registrar" and its required trust relation with EST server as described in Section 5 of [RFC9148] is therefore redundant.

So, while the same authentication scheme (Diffie-Hellman key exchange authenticated with transported certificates) and the same EST payloads as EST-coaps also apply to EST-oscore, the latter specifies other authentication schemes and a new matching EST function. The reason for these deviations is that a significant overhead can be removed in terms of message sizes and round trips by using a different handshake, public key type or transported credential, and those are independent of the actual enrollment procedure.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119]. These words may also appear in this document in lowercase, absent their normative meanings.

This document uses terminology from [RFC9148] which in turn is based on [RFC7030] and, in turn, on [RFC5272].

The term "Trust Anchor" follows the terminology of [RFC6024]: "A trust anchor represents an authoritative entity via a public key and associated data. The public key is used to verify digital signatures, and the associated data is used to constrain the types of information for which the trust anchor is authoritative." One example of specifying more compact alternatives to X.509 certificates for exchanging trust anchor information is provided by the TrustAnchorInfo structure of [RFC5914], the mandatory parts of which essentially is the SubjectPublicKeyInfo structure [RFC5280], i.e., an algorithm identifier followed by a public key.

3. Authentication

This specification replaces the DTLS handshake in EST-coaps with the lightweight authenticated key exchange protocol EDHOC [I-D.ietf-lake-edhoc]. During initial enrollment the EST-oscore client and server run EDHOC [I-D.ietf-lake-edhoc] to authenticate and establish the OSCORE security context with which the EST payloads are protected.

EST-oscore clients and servers MUST perform mutual authentication. The EST server and EST client are responsible for ensuring that an acceptable cipher suite is negotiated. The client MUST authenticate the server before accepting any server response. The server MUST authenticate the client and provide relevant information to the CA for decision about issuing a certificate.

3.1. EDHOC

EDHOC supports authentication with certificates/raw public keys (referred to as "credentials"), and the credentials may either be transported in the protocol, or referenced. This is determined by the identifier of the credential of the endpoint, ID_CRED_x for x= Initiator/Responder, which is transported in an EDHOC message. This identifier may be the credential itself (in which case the credential is transported), or a pointer such as a URI to the credential (e.g., x5t, see [I-D.ietf-cose-x509]) or some other identifier which enables the receiving endpoint to retrieve the credential.

3.2. Certificate-based Authentication

EST-oscore, like EST-coaps, supports certificate-based authentication between EST client and server. In this case the client MUST be configured with an Implicit or Explicit Trust Anchor (TA) [RFC7030] database, enabling the client to authenticate the server. During the initial enrollment the client SHOULD populate its Explicit TA database and use it for subsequent authentications.

The EST client certificate SHOULD conform to [RFC7925]. The EST client and/or EST server certificate MAY be a (natively signed) CBOR certificate [I-D.ietf-cose-cbor-encoded-cert].

3.3. Channel Binding

The [RFC5272] specification describes proof-of-possession as the ability of a client to prove its possession of a private key which is linked to a certified public key. In case of signature key, a proof-of-possession is generated by the client when it signs the PKCS#10 Request during the enrollment phase. Connection-based proof-of-possession is OPTIONAL for EST-oscore clients and servers.

When desired the client can use the EDHOC-Exporter API to extract channel-binding information and provide a connection-based proof-of possession. Channel-binding information is obtained as follows

edhoc-unique = EDHOC-Exporter(TBD1, "EDHOC Unique", length),

where TBD1 is a registered label from the EDHOC Exporter Label registry, length equals the desired length of the edhoc-unique byte string. The client then adds the edhoc-unique byte string as a challengePassword (see Section 5.4.1 of [RFC2985]) in the attributes section of the PKCS#10 Request [RFC2986] to prove to the server that the authenticated EDHOC client is in possession of the private key associated with the certification request, and signed the certification request after the EDHOC session was established.

3.4. Optimizations

* The last message of the EDHOC protocol, message_3, MAY be combined with an OSCORE request, enabling authenticated Diffie-Hellman key exchange and a protected CoAP request/response (which may contain an enrolment request and response) in two round trips [I-D.ietf-core-oscore-edhoc].

* The certificates MAY be compressed, e.g. using the CBOR encoding defined in [I-D.ietf-cose-cbor-encoded-cert].

* The certificate MAY be referenced instead of transported [I-D.ietf-cose-x509]. The EST-oscore server MAY use information in the credential identifier field of the EDHOC message (ID_CRED_x) to access the EST-oscore client certificate, e.g., in a directory or database provided by the issuer. In this case the certificate may not need to be transported over a constrained link between EST client and server.

* Conversely, the response to the PKCS#10 request MAY be a reference to the enrolled certificate rather than the certificate itself. The EST-oscore server MAY in the enrolment response to the EST-oscore client include a pointer to a directory or database where the certificate can be retrieved.

4. Protocol Design and Layering

EST-oscore uses CoAP [RFC7252] and Block-Wise [RFC7959] to transfer EST messages in the same way as [RFC9148]. Instead of DTLS record layer, OSCORE [RFC8613] is used to protect the EST payloads. DTLS handshake is replaced with EDHOC [I-D.ietf-lake-edhoc]. Figure 1 below shows the layered EST-oscore architecture.

```
             +------------------+
             |   EST messages   |
+------------+------------------+
|   EDHOC    |     OSCORE       |
+------------+------------------+
|         CoAP or HTTP          |
+-------------------------------+
|          UDP or TCP           |
+-------------------------------+
```

Figure 1: EST protected with OSCORE.

EST-oscore follows much of the EST-coaps and EST design.

4.1. Discovery and URI

The discovery of EST resources and the definition of the short EST-coaps URI paths specified in Section 4.1 of [RFC9148], as well as the new Resource Type defined in Section 8.2 of [RFC9148] apply to EST-oscore. Support for OSCORE is indicated by the "osc" attribute defined in Section 9 of [RFC8613], for example:

```
     REQ: GET /.well-known/core?rt=ace.est.sen

     RES: 2.05 Content
   </est>; rt="ace.est";osc
```

4.2. Mandatory/optional EST Functions

The EST-oscore specification has the same set of required-to-implement functions as EST-coaps. The content of Table 1 is adapted from Section 4.2 in [RFC9148] and uses the updated URI paths (see Section 4.1).

EST functions	EST-oscore implementation
/crts	MUST
/sen	MUST
/sren	MUST
/skg	OPTIONAL
/skc	OPTIONAL
/att	OPTIONAL

Table 1: Mandatory and optional EST-oscore functions

4.2.1. /crts

EST-coaps provides the /crts operation. A successful request from the client to this resource will be answered with a bag of certificates which is subsequently installed in the Explicit TA.

A trust anchor is commonly a self-signed certificate of the CA public key. In order to reduce transport overhead, the trust anchor could be just the CA public key and associated data (see Section 2), e.g., the SubjectPublicKeyInfo, or a public key certificate without the signature. In either case they can be compactly encoded, e.g. using CBOR encoding [I-D.ietf-cose-cbor-encoded-cert].

4.3. Payload formats

Similar to EST-coaps, EST-oscore allows transport of the ASN.1 structure of a given Media-Type in binary format. In addition, EST-oscore uses the same CoAP Content-Format Options to transport EST requests and responses . Table 2 summarizes the information from Section 4.3 in [RFC9148].

URI	Content-Format	#IANA
/crts	N/A (req)	-
	application/pkix-cert (res)	287
	application/pkcs-7-mime;smime-type=certs-only (res)	281
/sen	application/pkcs10 (req)	286
	application/pkix-cert (res)	287
	application/pkcs-7-mime;smime-type=certs-only (res)	281
/sren	application/pkcs10 (req)	286
	application/pkix-cert (res)	287
	application/pkcs-7-mime;smime-type=certs-only (res)	281
/skg	application/pkcs10 (req)	286
	application/multipart-core (res)	62
/skc	application/pkcs10 (req)	286
	application/multipart-core (res)	62
/att	N/A (req)	-
	application/csrattrs (res)	285

Table 2: EST functions and there associated Media-Type and IANA numbers

4.4. Message Bindings

The EST-oscore message characteristics are identical to those specified in Section 4.4 of [RFC9148]. It is RECOMMENDED that

* The EST-oscore endpoints support delayed responses

* The endpoints supports the following CoAP options: OSCORE, Uri-Host, Uri-Path, Uri-Port, Content-Format, Block1, Block2, and Accept.

* The EST URLs based on https:// are translated to coap://, but with mandatory use of the CoAP OSCORE option.

4.5. CoAP response codes

See Section 4.5 in [RFC9148].

4.6. Message fragmentation

The EDHOC key exchange is optimized for message overhead, in particular the use of static DH keys instead of signature keys for authentication (e.g., method 3 of [I-D.ietf-lake-edhoc]). Together with various measures listed in this document such as CBOR-encoded payloads ([I-D.ietf-cose-cbor-encoded-cert]), CBOR certificates [I-D.ietf-cose-cbor-encoded-cert], certificates by reference (Section 3.4), and trust anchors without signature (Section 4.2.1), a significant reduction of message sizes can be achieved.

Nevertheless, depending on application, the protocol messages may become larger than available frame size resulting in fragmentation and, in resource constrained networks such as IEEE 802.15.4 where throughput is limited, fragment loss can trigger costly retransmissions.

It is RECOMMENDED to prevent IP fragmentation, since it involves an error-prone datagram reconstitution. To limit the size of the CoAP payload, this specification mandates the implementation of CoAP option Block1 and Block2 fragmentation mechanism [RFC7959] as described in Section 4.6 of [RFC9148].

4.7. Delayed Responses

See Section 4.7 in [RFC9148].

4.8. Enrollment of Static DH Keys

This section specifies how the EST client enrolls a static DH key. Because a DH key pair cannot be used for signing operations, the EST client attempting to enroll a DH key must use an alternative proof-of-possesion algorithm. The EST client obtained the CA certs including the CA's DH certificate using the /crts function. The certificate indicates the DH group parameters which MUST be respected by the EST client when generating its own DH key pair. The EST client prepares the PKCS #10 object and signs it by following the steps in Section 4 of [RFC6955]. The Key Derivation Function (KDF) and the MAC MUST be set to the HDKF and HMAC algorithms used by OSCORE. As per [RFC8613], the HKDF MUST be one of the HMAC-based HKDF [RFC5869] algorithms defined for COSE [RFC9052]. The KDF and MAC is thus defined by the hash algorithm used by OSCORE in HKDF and HMAC, which by default is SHA-256. When EDHOC is used, then the hash algorithm is the application hash algorithm of the selected cipher suite.

5. HTTP-CoAP Proxy

As noted in Section 5 of [RFC9148], in real-world deployments, the EST server will not always reside within the CoAP boundary. The EST-server can exist outside the constrained network in a non-constrained network that supports HTTP but not CoAP, thus requiring an intermediary CoAP-to-HTTP proxy.

Since OSCORE is applicable to CoAP-mappable HTTP (see Section 11 of [RFC8613]) the EST payloads can be protected end-to-end between EST client and EST server independent of transport protocol or potential transport layer security which may need to be terminated in the proxy, see Figure 2. Therefore the concept "Registrar" and its required trust relation with EST server as described in Section 5 of [RFC9148] is redundant.

The mappings between CoAP and HTTP referred to in Section 8.1 of [RFC9148] apply, and additional mappings resulting from the use of OSCORE are specified in Section 11 of [RFC8613].

OSCORE provides end-to-end security between EST Server and EST Client. The use of TLS and DTLS is optional.

```
                                       Constrained-Node Network
     .---------.                     .-----------------------------.
     |   CA    |                     |.---------------------------.|
     '---------'                     ||                           ||
          |                          ||                           ||
      .------.  HTTP   .-----------------.   CoAP   .-----------.  ||
      | EST  |<------->|  CoAP-to-HTTP   |<-------->| EST Client|  ||
      |Server|  (TLS)  |      Proxy      |  (DTLS)  '-----------'  ||
      '------'         '-----------------'                         ||
                                     ||                           ||
         <-------------------------------------------------->     ||
                        OSCORE       ||                           ||
                                     |'---------------------------'|
                                     '-----------------------------'
```

Figure 2: CoAP-to-HTTP proxy at the CoAP boundary.

6. Security Considerations

TBD: Compare with RFC9148

TBD: Channel binding security considerations: 3SHAKE attack and EDHOC.

7. Privacy Considerations

TBD

8. IANA Considerations

8.1. EDHOC Exporter Label Registry

IANA is requested to register the following entry in the "EDHOC Exporter Label" registry under the group name "Ephemeral Diffie-Hellman Over COSE (EDHOC).

Label	Description	Reference
TBD1	EDHOC unique	[[this document]]

Figure 3: EDHOC Exporter Label

9. Acknowledgments

10. References

10.1. Normative References

[I-D.ietf-lake-edhoc]
Selander, G., Mattsson, J. P., and F. Palombini, "Ephemeral Diffie-Hellman Over COSE (EDHOC)", Work in Progress, Internet-Draft, draft-ietf-lake-edhoc-19, 3 February 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-lake-edhoc-19>.

[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/info/rfc2119>.

[RFC5869] Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, DOI 10.17487/RFC5869, May 2010, <https://www.rfc-editor.org/info/rfc5869>.

[RFC6955] Schaad, J. and H. Prafullchandra, "Diffie-Hellman Proof-of-Possession Algorithms", RFC 6955, DOI 10.17487/RFC6955, May 2013, <https://www.rfc-editor.org/info/rfc6955>.

[RFC7049] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", RFC 7049, DOI 10.17487/RFC7049, October 2013, <https://www.rfc-editor.org/info/rfc7049>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/info/rfc7252>.

[RFC7925] Tschofenig, H., Ed. and T. Fossati, "Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things", RFC 7925, DOI 10.17487/RFC7925, July 2016, <https://www.rfc-editor.org/info/rfc7925>.

[RFC7959] Bormann, C. and Z. Shelby, Ed., "Block-Wise Transfers in the Constrained Application Protocol (CoAP)", RFC 7959, DOI 10.17487/RFC7959, August 2016, <https://www.rfc-editor.org/info/rfc7959>.

[RFC8152] Schaad, J., "CBOR Object Signing and Encryption (COSE)", RFC 8152, DOI 10.17487/RFC8152, July 2017, <https://www.rfc-editor.org/info/rfc8152>.

[RFC8613] Selander, G., Mattsson, J., Palombini, F., and L. Seitz, "Object Security for Constrained RESTful Environments (OSCORE)", RFC 8613, DOI 10.17487/RFC8613, July 2019, <https://www.rfc-editor.org/info/rfc8613>.

[RFC9052] Schaad, J., "CBOR Object Signing and Encryption (COSE): Structures and Process", STD 96, RFC 9052, DOI 10.17487/RFC9052, August 2022, <https://www.rfc-editor.org/info/rfc9052>.

[RFC9148] van der Stok, P., Kampanakis, P., Richardson, M., and S. Raza, "EST-coaps: Enrollment over Secure Transport with the Secure Constrained Application Protocol", RFC 9148, DOI 10.17487/RFC9148, April 2022, <https://www.rfc-editor.org/info/rfc9148>.

10.2. Informative References

[I-D.ietf-core-oscore-edhoc]
Palombini, F., Tiloca, M., Höglund, R., Hristozov, S., and G. Selander, "Profiling EDHOC for CoAP and OSCORE", Work in Progress, Internet-Draft, draft-ietf-core-oscore-edhoc-06, 23 November 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-core-oscore-edhoc-06>.

[I-D.ietf-core-oscore-groupcomm]
Tiloca, M., Selander, G., Palombini, F., Mattsson, J. P., and J. Park, "Group OSCORE - Secure Group Communication for CoAP", Work in Progress, Internet-Draft, draft-ietf-core-oscore-groupcomm-17, 20 December 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-core-oscore-groupcomm-17>.

[I-D.ietf-cose-cbor-encoded-cert]
Mattsson, J. P., Selander, G., Raza, S., Höglund, J., and M. Furuhed, "CBOR Encoded X.509 Certificates (C509 Certificates)", Work in Progress, Internet-Draft, draft-ietf-cose-cbor-encoded-cert-05, 10 January 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-cose-cbor-encoded-cert-05>.

[I-D.ietf-cose-x509]
Schaad, J., "CBOR Object Signing and Encryption (COSE): Header Parameters for Carrying and Referencing X.509 Certificates", Work in Progress, Internet-Draft, draft-ietf-cose-x509-09, 13 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-cose-x509-09>.

[RFC2985] Nystrom, M. and B. Kaliski, "PKCS #9: Selected Object Classes and Attribute Types Version 2.0", RFC 2985, DOI 10.17487/RFC2985, November 2000, <https://www.rfc-editor.org/info/rfc2985>.

[RFC2986] Nystrom, M. and B. Kaliski, "PKCS #10: Certification Request Syntax Specification Version 1.7", RFC 2986, DOI 10.17487/RFC2986, November 2000, <https://www.rfc-editor.org/info/rfc2986>.

[RFC5272] Schaad, J. and M. Myers, "Certificate Management over CMS (CMC)", RFC 5272, DOI 10.17487/RFC5272, June 2008, <https://www.rfc-editor.org/info/rfc5272>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC5914] Housley, R., Ashmore, S., and C. Wallace, "Trust Anchor Format", RFC 5914, DOI 10.17487/RFC5914, June 2010, <https://www.rfc-editor.org/info/rfc5914>.

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

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <https://www.rfc-editor.org/info/rfc6347>.

[RFC7030] Pritikin, M., Ed., Yee, P., Ed., and D. Harkins, Ed., "Enrollment over Secure Transport", RFC 7030, DOI 10.17487/RFC7030, October 2013, <https://www.rfc-editor.org/info/rfc7030>.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <https://www.rfc-editor.org/info/rfc7228>.

[RFC8392] Jones, M., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "CBOR Web Token (CWT)", RFC 8392, DOI 10.17487/RFC8392, May 2018, <https://www.rfc-editor.org/info/rfc8392>.

[RFC9031] Vuini, M., Ed., Simon, J., Pister, K., and M. Richardson, "Constrained Join Protocol (CoJP) for 6TiSCH", RFC 9031, DOI 10.17487/RFC9031, May 2021, <https://www.rfc-editor.org/info/rfc9031>.

Authors' Addresses

Göran Selander
Ericsson AB
Email: goran.selander@ericsson.com

Shahid Raza
RISE
Email: shahid.raza@ri.se

Martin Furuhed
Nexus
Email: martin.furuhed@nexusgroup.com

Malia Vuini
Inria
Email: malisa.vucinic@inria.fr

Timothy Claeys
Email: timothy.claeys@gmail.com