3GPP TSG SA WG3 Security — S3#22 **S3020079**

25th – 28th February, 2002

Bristol, UK

Source: Nokia

Title: Use of COPS protocol in Ze interface

Document for: Discussion

Agenda Item: x.x

This document discusses COPS usage in Ze interface for local Security Association and Policy distribution. COPS enables already in 33.200 V0.1.4 (Rel 5) specification chapter 8.1 mentioned extended PUSH mechanism for delivering the SA and policy information to MAP-NEs. The actual specification work could be done in CN4 and contributed to the IETF as an Informational RFC for approval. This document is meant to serve as an input to CN4.

COPS uses TCP as its transport protocol for reliable exchange of messages between policy clients and a server. Therefore, no additional mechanisms are necessary for reliable communication between a server and its clients. COPS provides message level security for authentication, replay protection, and message integrity. COPS shall utilize 33.210 NDS/IP mechanisms to make confidentiality possible for delivered MAPSec encryption and integrity keys.

The exact COPS message contents are not defined in this paper, so COPS extension document is required. These extensions have no use outside MAPSec specification. Use cases in the Ze interface are identified and according to these general level identifications the contents of PIB should be specified. COPS extension document i.e. MAPSec PIB could be based on existing IPSec PIB, since it seems eligible for MAPSec also.

References

[1] IETF RFC 2748: The COPS (Common Open Policy Service) protocol, http://www.ietf.org/rfc/rfc2748.txt

[2] IETF RFC 3084: COPS Usage for Policy Provisioning (COPS-PR), http://www.ietf.org/rfc/rfc3084.txt

[3] IETF Draft (2001): The MAP Security Domain of Interpretation for ISAKMP, draft-arkko-map-doi-04.txt

1 COPS usage in SA and Policy Distribution

Common Open Policy Service protocol is a simple query and response protocol that can be used to exchange policy information between a policy server (Policy Decision Point or PDP) and its clients (Policy Enforcement Points or PEPs). The optional Local Policy Decision Point (LPDP) can be used to make local policy decisions in the absence of a PDP. In MAPSec the MAP-NE acts as PEP and KAC acts as PDP, LPDP is not utilized.

1.1 Establishing COPS connection

Before any COPS transaction can take place between MAP-NE and KAC a communication path has to be set up between them. MAP-NE having PEP functionality is responsible to establish a TCP session to KAC.

For opening COPS connection between MAP-NE and KAC MAP-NE sends *Client-Open* message to KAC. If KAC accepts the client type, it responds with *Client-Accept* message. Mandatory parameters are <PEP identity> in the first message and <Keep Alive timer value> in the second message.

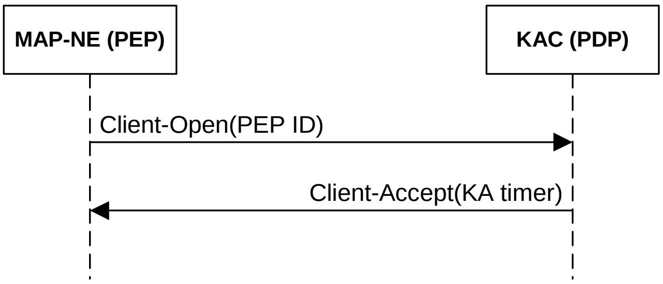

Figure 1: Opening COPS connection

When the connection is established, the MAP-NE sends information about itself to the KAC by sending *Request* message. This message contains in the ’Context’ parameter information that it is a configuration request message. With that information KAC knows that it must provide security parameters (policy and SAs) to the MAP-NE. KAC sends *Decision* message(s), which contain the security parameters. MAP-NE responds with Report State message to inform KAC about the status of installation of the security parameters.

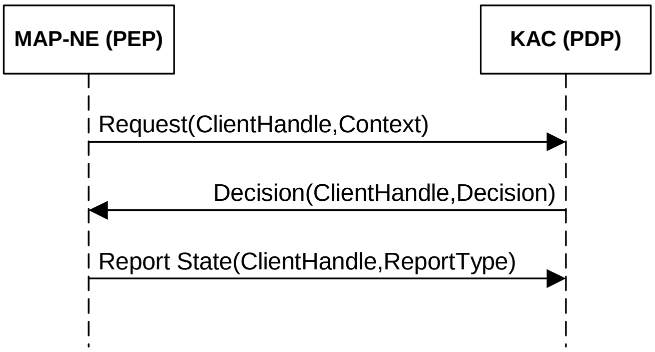
MAP-NE (PEP)
KAC (PDP)
Request(ClientHandle,Context)
Decision(ClientHandle,Decision)
Report State(ClientHandle,ReportType)

Figure 2: Basic decision request

1.2 Security policy management

The initial policy data delivery is done when MAP-NE has registered to KAC.

When policy is changed in KAC, the necessary information has to be delivered to MAP-NEs too. KAC knows all MAP-NEs, which have registered as clients to KAC with Client-Open message, and therefore is able to start delivery of security policy information to all MAP-NEs. KAC uploads the security policy information to MAP-NEs by sending *Decision* message to all registered MAP-NEs. The MAP-NE responds with Report State message to inform KAC about the status of transaction.

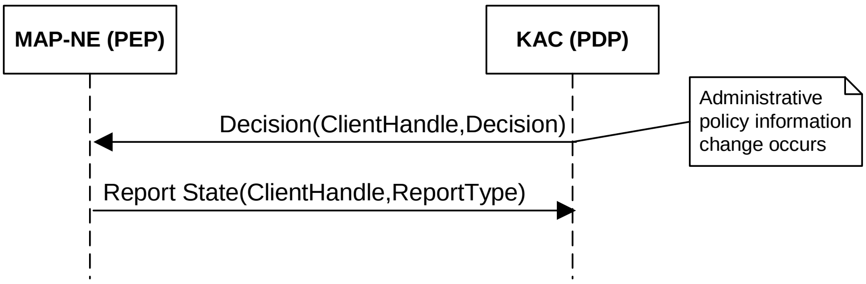
MAP-NE (PEP)
KAC (PDP)
Decision(ClientHandle,Decision)
Administrative policy information change occurs
Report State(ClientHandle,ReportType)

Figure 3: Policy information update

1.3 Security association management

KAC may perform an unsolicited download of MAPSec SA to a MAP-NE by sending *Decision* message. <Client Handle> identifies MAP-NE. <Decision> is an installation of configuration data for MAPSec SA towards a target Security Domain. The procedure is similar to KAC initiated policy information update.

1.3.1 SA revocation

If SA must be deleted, the initiation must come from KAC. The procedure goes same way as KAC initiated policy information delivery: KAC invokes *Decision* message and Decision parameter contains information that SA must be deleted from SADB.

1.4 SA recovery

SA Recovery means a situation where MAP-NE has lost or somehow corrupted all or some of the MAPSec SAs it has received earlier. E.g. MAP-NE might have gone through a reset. Also the cases where MAP-NE has not for some reason received an SA or it notices that SA expires and there is not a new one available are included into *SA recovery*.

To recover from any obscure state MAP-NE has to initialise a *Client-Open* and *Request* procedures to KAC as described in Establishing COPS connection.

1.5 Other mandatory COPS procedures

As interval between MAPSec SA renewals may be a long one, to keep COPS connection and TCP session alive a *Keep-Alive* message has to be sent from MAP-NE before KA timer expires. Receiving node has to echo back the same *Keep-Alive* message.

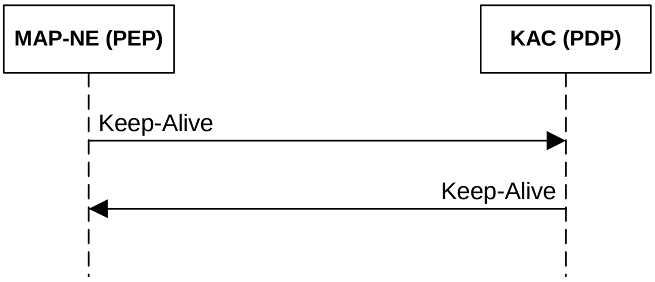

Figure 4: Keep Alive procedure

1.6 Manual security association management

If some PLMN supports only manual SA management then it must be possible to manually configure the parameters to KAC. Delivery of SA to MAP-NEs is handled same way as in automatic SA management.