3GPP TSG SA WG3 Security — S3#16 **S3-000670**

28-30 November, 2000

Sophia Antipolis, France

Source: **Telenor**

Title: **UMTS Key Management**

Document for: **decision/discussion**

Agenda Item: **tbd**

An Alternative Key Management Architecture for UMTS

Siemens contribution Sz000023 on problems with using IKE in our current security architecture for the network domain has prompted this contribution. The contribution therefore takes a second look on some of the assumptions that we based our present architecture on and with a slightly modified set of assumptions, an alternative security architecture is sketched out.

This alternative architecture has basically two main components that differs from our current architecture:

- KACs are no longer required for native IP-based protocols
- MAPsec is separated from IP security **and** MAP key distribution may be done by MAP, HTTP or IKE

The suggested changes appear very late and I realize that it may be seen as too late for our current process.

Underlying assumptions and simplifying assumptions

The basis for out two-tiered approach to the key management and distribution architecture is based on the real-world problems of establishing and maintaining a large set of SAs. Since it turns out that IKE does not scale well when the number of NEs increase above some limit, the need for an alternative approach is apparent.

It was therefore deemed necessary to establish a hierarchy that would cut down the number of NEs that needed to establish security associations (SAs) between them. In essence, the current architecture assumes that there will be a few Key Administration Centres (KACs) per operator and that these KACs will negotiate SAs that will be used by a limited set of Security Gateways (SEGs). In this way, there will be a many-to-many situation between the KACs/operators, but that individual NEs within one operators network need not establish or maintain SAs towards NEs in other operator's networks[1].

Underlying assumptions in alternative architecture:

- The complexity of SA many-to-many relationships is a function of the actual SA fan-in/fan-out at the individual nodes. If that number can be reduced to within some reasonable upper limit, the resulting complexity would be acceptable.
- For the intranet(subnet) belonging to one operator, the number of SA fan-in/fan-out is not directly affected by the number of KACs, but is more related to the number of SEGs and NEs within the intranet.

[1] The current architecture allows for direct NE-NE communication, but this is meant to be an option that would not occur more often than is really called for in terms of traffic actually carried between the nodes.

- The architecture would be significantly simpler if the number of SEGs can be kept to fairly low number, say at least an order less than the number of NEs.

  Should it be possible to have "sufficiently few" SEGs, there will no real need to separate KAC and SEG functionality. The term "sufficiently few" does not necessarily imply that the number must be small, but rather that the fan-in/fan-out of each separate SEG must be limited in order to avoid/limit the many-to-many scalability problems of IKE. So, if an operator needs/wants a large number of SEGs, the only real requirement would be that these SEGs would be limited in the number of reachable destinations.

- All traffic towards other networks would be routed via a SEG. This/these SEGs will be given in the routing tables as the only way to reach destinations outside the intranet/subnet. By, default this will mean that also traffic that do not need security services will be routed via SEGs. Security policies will ensure that IPsec will secure only traffic that needs security services. The remaining traffic will benefit from firewalls etc associated with the SEG.

  Notice that one can still functionally have NE←→NE communication between NEs from different networks. The requirement then will be that these NEs then include SEG functionality, and in effect should be viewed as NE/SEG combined elements.

- All NEs that communicates with native IP based protocols shall be able to support IKE (or they shall allow for manual entry of SAs)

- The functionality of a Border Gateway (as defined in GPRS/UMTS) will be contained in the SEG

Some simplifying assumptions (not conditions):

- For inter-PLMN traffic, only ESP in tunnel-mode should be used. ESP provides all necessary security services except for protection of the outer IP-header. Notice that tunnel-mode is required by border gateways/SEGs anyway. Notice also that since the original IP header will be within the payload of the ESP tunnel it will be fully protected.
- For intranet(subnet) traffic it will be possible for the operator to chose to use transport-mode and AH. However, to keep things simple we should recommend only using ESP in tunnel-mode.
- IKE authentication is to be based on pre-shared secrets in the first version of the architecture. This applies to SEG-SEG communications as well as to NE-SEG and NE-NE communications.
- IPsec can be combined with IP compression, but this complicates matters as a separate compression SA must be negotiated. Since the compression, by necessity, must be on a packet-by-packet basis and since most of the packets on the control plane is likely to be shorter than 128 bytes[2], the advantages of using IP payload compression will be very limited. We should therefore recommend that IP payload compression should be not used.
- The default SA lifetime of 8 hours (from RFC-2407) should be used for SEG-SEG communication. That is, longer lifetimes should **not** be used without a careful analysis of the amount of traffic carried by the tunnel. Lifetimes within an operators network should be left as an operator option and need not be specified.

[2] a limit suggested in RFC-2393 as a minimum before it would be useful to use compression

NDS architecture sketch (excluding MAPsec)

The outline presented here is by no means a complete or mature work. It may need substantial changes before it can be realized.

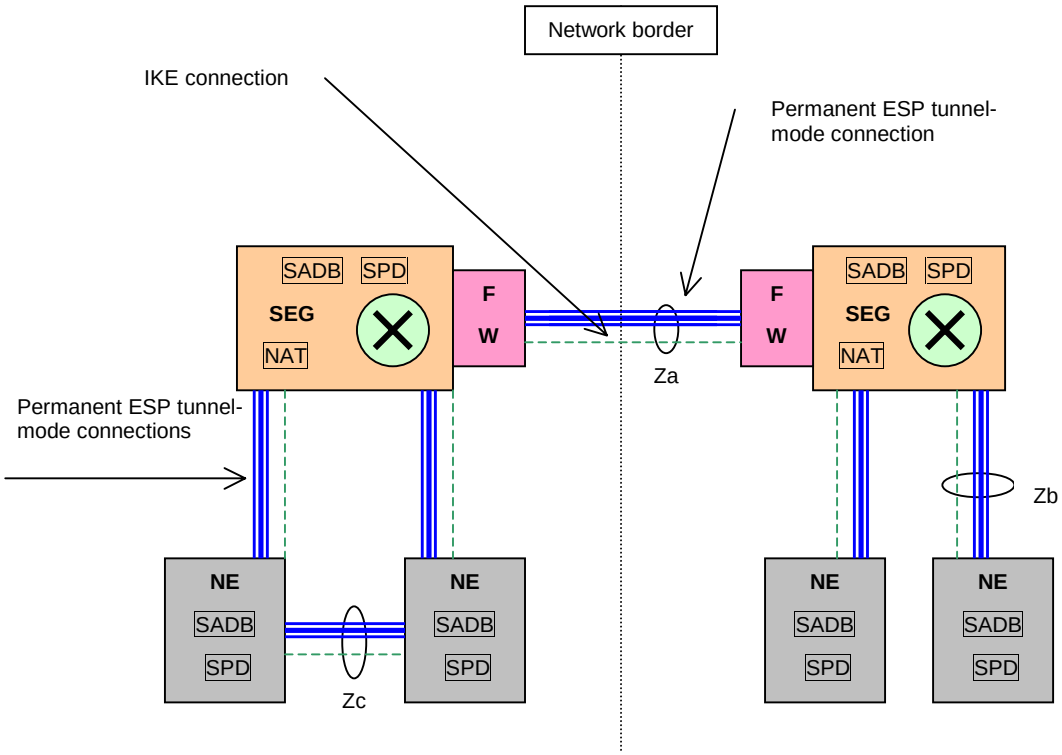

Figure-1: NDS architecture for IP-based protocols

Interface desciption:

a) *Za interface (SEG-SEG)*

The SEGs shall be capable of using IKE to negotiate IPsec SAs between themselves and they will establish "permanent" ESP tunnels between them. All traffic that shall be protected by IPsec and that passes between SEG-A and SEG-B shall be transported within the tunnel. This is a coarse grained use of tunnels, which is what we will want for the Za interface in UMTS. Whether or not it would be useful to have separate tunnels for different protocols has not been studied, but provided that the protocols can be separated by portnumbers, IP-addresses and/or protocol-id it should at least be possible to do so.

SEGs can be dedicated to only serve a certain subset of all roaming partners. This will limit the number of SAs and tunnels that need to be maintained. We should standardize SA liftetime default for UMTS and we should standardize ESP in tunnel-mode. Normally ESP shall be used with both encryption and authentication/integrity, but an authentictaion/integrity only mode must be allowed[3].

The number of SEGs within a network will normally be limited. SEGs shall not be used for the Gi interface.

[3] Notice that it will be virtually impossible to have fine grained control of the use of encryption and/or authentication/integrity within on protocol. For instance, for GTP-C it will be difficult to distinguish between messages that need only authentication/integrity and those that also need confidentiality. This difficulty is due to the fact that IPsec does not inspect the IP payload when making decisions on the approriate security policy to employ. This of course, is a consequence of having the security services at the network layer.

b) *Zb interface (NE-SEG)*

NEs and SEGs shall be capable of using IKE to negotiate IPsec SAs between themselves. ESP tunnels will be set up between NEs and SEGs (more or less permanently). All control plane traffic that shall be routed towards external destination shall be routed towards a SEG.

Should the NE for some reason not be able to use IKE, then it shall allow manual SAs to be set up. The lifetime of the SAs should allow for the default values, but the actual setting will be a policy decision (manual SAs do not expiry).

c) *Zc interface (NE-NE)*

NEs shall be capable of using IKE to negotiate IPsec SAs between themselves. ESP tunnels will be set up between the NEs (more or less permanently). The ESP tunnel shall be used for all control plane traffic that needs security.

Should the NEs for some reason not be able to use IKE, then it shall allow manual SAs to be set up. The lifetime of the SAs should allow for the default values, but the actual setting will be a policy decision (manual SAs do not expiry).

There will be no NE-NE interface for NEs belonging to separate subnets. That is, if there exists a need for NE-NE communication between NEs belonging to different nets, they shall contain SEG functionality and then be SEGs by definition. So while one physical unit may contain both an NE and a SEG, they will be viewed as two logically separated entities.

End-to-end security?

The outlined NDS architecture does not really have end-to-end security as such. Consider the following case: A packet **p** shall be sent from **NE-A** in network **A** to **NE-B** in network **B**.

a) From **NE-A** the packet **p** is sent through the already existing ESP tunnel $\mathbf{T_{NE\text{-}A \leftrightarrow SEG\text{-}A}}$. The decision to use the tunnel is based on the source/destination IP-address, the protocol ID and the source/destination portnumbers in **p**'s header.

b) In **SEG-A**, **p** is "taken out of the tunnel" and decrypted etc. **SEG-A** may now allow a NAT to modify addresses. **SEG-A** then analyses the header of **p** and according to the security policy sends **p** towards **SEG-B** using the already established ESP tunnel $\mathbf{T_{SEG\text{-}A \leftrightarrow SEG\text{-}B}}$. This time a different pair of SAs applies and so the algorithms and keys used are different from the previous tunnel.

c) At **SEG-B** packet **p** is taken out the tunnel and decrypted etc. Should NAT intervention be required it can safely be done at this stage. **SEG-B** analyses the header of **p** and according to the security policy sends **p** towards **NE-B** using the existing ESP tunnel $\mathbf{T_{SEG\text{-}B \leftrightarrow NE\text{-}B}}$.

d) At **NE-B** packet **p** is taken out of the tunnel and decrypted etc. Packet **p** is finally at its destination.

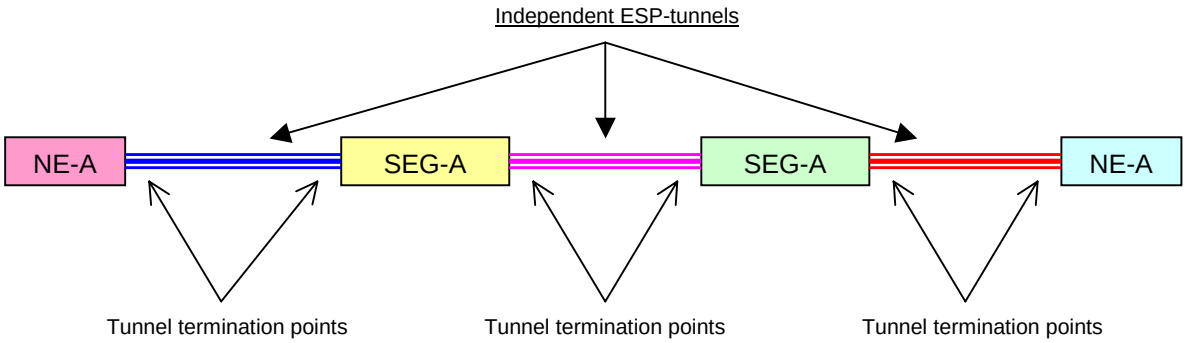

Figure-2: Chained-tunnels approach

Within the IPsec community the configuration in figure-2 is either called *chained-tunnels* or *hub-and-spoke tunnels* depending on the layout of the network. Provided that the SEG/FW can be secured, the lack of "real" end-to-end security is not a problem. Indeed, if NATs are to be allowed, real end-to-end security would be a problem. Notice that from the NEs perspective, the effective security is still end-to-end.

NDS architecture sketch for MAP

The case for MAPsec is difficult. Our current approach for MAPsec key distribution does have a few problems and until we have solved them we simply haven't got a consistent and/or complete key distribution method for MAP.

We basically have three options for MAP key distribution:

- Revert to our original intentions of distribution MAP keys over MAP
- Get a working solution were KACs negotiate with IKE. It has been suggested to either let MAP do the transport from KAC to MAP-NE or use HTTP for the KAC to MAP-NE transport.
- Accept manual/proprietary key distribution

It is likely that all three approaches will end up with a key distribution scheme where keys will be valid for an entire network. That is, one will expect that operator-A and operator-B will have only one key for confidentiality and one for integrity between them. This will scale well, but needless to say there are some drawbacks with a situation were all MAP-NEs in operator-As network will use the same keys when communication with MAP NEs within operator-Bs network.

Reverting to key distribution over MAP?

MAP addressing is quite different from the IP addressing. This makes it hard to directly use the IPsec model for MAPsec. A separate DoI must be formulated and a separate stage of distributing SAs must be defined. Furthermore, security for MAP is provided on the application layer, which means that the security will normally have to be end-to-end.

The basic assumption we had when we decided to use IKE for both IPsec and MAPsec seems no to be fulfilled. There seems to be quite difficult to construct an architecture that scales well and that solves the addressing problems that we have with using IKE in our current NDS architecture. The proposal for an alternative NDS architecture does not solve the MAPsec problem. It seems that we cannot assume that MAP-NEs will be able to run IKE and we cannot cannot easily have IKE on SEGs (or KACs) negotiating on behalf of MAP-NEs without introducing at least one new transport mechanism between SEGs/KACs and MAP-NEs.

All in all, with the benefit of hindsigth, our current approach for providing MAPsec SAs may seem a little misguided.

We therefore may want to revisit or decision of aligning key distribution and management for MAP/SS7 and IP based protocols. At an earlier stage a methode for key management and distribution was deviced which used MAP it self as the SA negotiation protocol. This approach has some benefits including:

- SA distribution alignment with IP based protocols seems to be hard and complicated to achieve
- The IKE model does not fit the MAP model very well. This shouldn't be all that surprising given that IKE and MAP were never intended to to work together.
- For application layer security it makes sence to also have the key distribution at the application layer. This way the same addressing assumption applies to both the key negotiation phase and the secure transport phase.
- It is unreasonable to require MAP NEs to include an IKE interface, which is probably needed if a simple solution is to be found within our current architecture
- The MAP DoI requirements are very modest. To use IKE is in many ways complete overkill.
- The release timing requires a method for MAP key distribution to be presented to SA plenary in decemeber in order to make it for R4

The obvious drawback to use MAP for its own key distribution and key management is that two separate methods for key distribution is required.

Keeping IKE for MAP SA negotiation between KACs?

This is of course our current working assumption. The benefit with this approach is that we can use IKE between operators and hence use IP network for that traffic. The drawback is that we haven't developed any method for distributing keys from KACs down to MAP-NEs.

We have also seen that it may be beneficial to let the KACs only negotiate IPsec SAs that the KAC themselves will use to establish an ESP tunnel between them. The ESP tunnel will then be used for transportation of the actual MAP SAs.

When it comes to distribution of MAP SAs from KACs to MAP-NEs I am currently aware of two approaches:

- Use MAP to distribute the keys
- Use HTTP to distribute keys (probably using TLS/SSL)

Should we choose to go for a solution with ESP-tunnel between KACs and HTTP distribution between KACs and MAP-NEs we will have to following:

- MAP-NEs will have to support IP to be able to use HTTP
- The HTTP methods should be applicable also for the KAC←→KAC scenario

Notice also that if we

a) accept a KAC←→KAC scenario with HTTP transport

b) and already use TLS/SSL to secure HTTP between KAC and MAP-NE

then we may very well choose to also use HTTP protected by TLS/SSL between KACs, in which case we no longer need IPsec between the KACs.

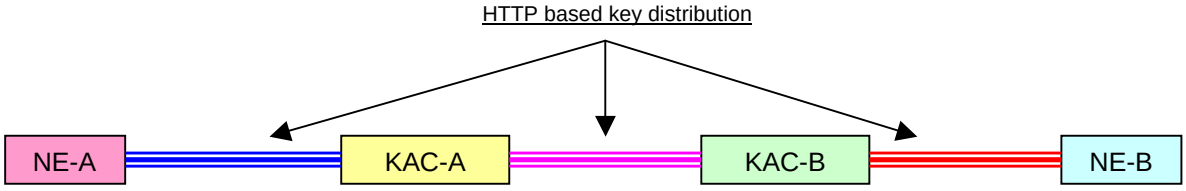

Figure-3: HTTP based MAP SA distribution, secured by TLS/SSL

The agreement on certificates for use with TLS/SSL need not be specified, but its probably wise to ask for instance GSMA to take on a role to issue certificates this purpose.

Proprietary/manual key distribution?

I really do not want proprietary or manual only solutions. Then I'd rather accept a delay for TS 33.200 for the next SA plenary.

Concluding remarks

Native IP-based protocols

The suggestions for changes in our key management architecture for native IP-based protocols in this contribution rest on a set of assumptions. Provided that these assumption hold it is advised that SA3 carefully considers whether the suggested architecture would better suit our needs than our present architecture.

The alternative architecture does not have KACs for native IP-based protocols. This has the advantage that IKE can be used in its intended way for SEG-SEG communication. Permanent ESP tunnels provide coarse-grained secure communication between SEGs. These tunnels are connected in what is known as a *chained tunnel* or *hub-and-spoke* configuration to provide end-to-end type of security.

NEs are assumed to be able to run IKE and negotiate ESP tunnels between NEs and SEGs. They too will use IKE in its native way.

MAPsec key distribution

I do not have strong feelings on exactly how to progress the MAP key management issue. I have sketched some possible ways forward:

- Revert to a model were MAP is used for its own key distribution
- Use IKE between KACs. Then use HTTP or other method for distribution between KACs and NEs. HTTP transport security is done with TLS/SSL.
- Use HTTP both for distribution between KACs and for KAC←→NE distribution. Transport security is done with TLS/SSL.

For these possible solution to be practical, it is probably required that SA-pair are negotiated on a network-network basis.

/Geir M. Køien