3GPP TSG SA WG3 Security **S3-020385**

July 9th – 12th, 2002

Helsinki, Finland

Agenda Item: TBD

Source: Ericsson

Title: Profiling SIP Security Agreement for IMS R5

Document for: Discussion and decision

1. Introduction

This paper discusses different alternatives on how the SIP Security Agreement draft [sec-agree] could be profiled for IMS R5. The paper will suggest that SA3 adopts the alternative in which all IPsec parameters are defined for [sec-agree] and that some of the parameters are given default values making it possible to leave those parameters out when agreeing security.

2. Different solutions

SA3 has been discussing on two basic ways on how the SIP Security Agreement could be tailored for IMS R5:

1. Only R5 IPsec parameters or 'suites' are defined and used.
2. All parameters needed for any combination of IPsec are defined and used.

Both of these solutions have their strengths and weaknesses. Option 1 is efficient in terms of bandwidth usage, however, it is not future prove. For example, changing the SA definitions in the future will be difficult. Option 2 might consume more bandwidth because more parameters must be included in the messages. However, it is more flexible than option 1 in terms of future development. Furthermore Ericsson is not aware of any ratio indicating how efficient the suite concept really is and note also that SIP compression will be used compressing all SIP messages efficiently.

A third alternative would be to take the strengths of the both option 1 and 2 without including the weaknesses. In this solution, all IPsec parameters are defined but only those that may have several values in R5 are sent. This can be done if some of the IPsec parameter has a 'default' value for IMS. If the parameter does not exist in SIP Security Agreement headers, the default value is assumed. The following parameters could have default values:

Encrypt algorithm: can be omitted and default is Null

Protocol: can be omitted and default is esp

Mode: can be omitted and default is trans (for transport)

Also the Algorithm for authentication or integrity protection could have default value NULL but it should be stressed that according to TS33.203 the NULL algorithm is not allowed.

3 Example

This example demonstrates how the syntax above can be applied. Note that this is only one possible solution, and that other variants exist. The final protocol details are defined in other 3GPP specifications. Note also that in this example the UE does not receive or send protected messages on the same port.

A UE negotiates the security mechanism to be used with the P-CSCF without knowing beforehand which IPsec protocols and algorithms the proxy supports.

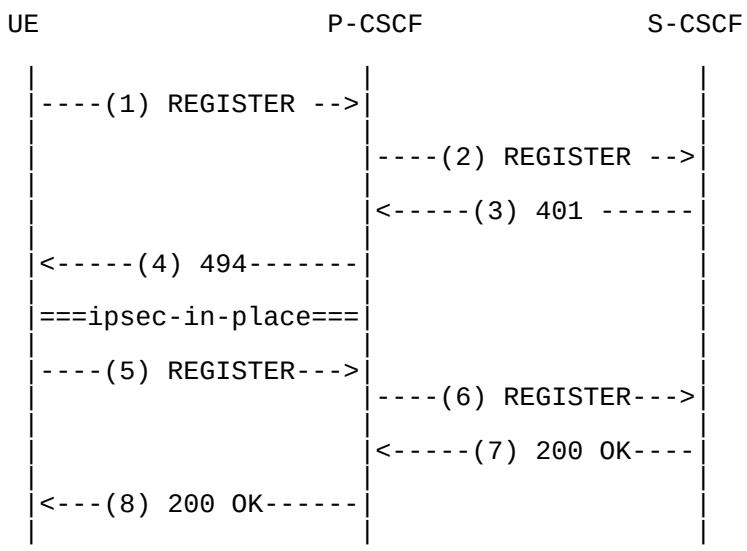

Figure Successful security negotiation with REGISTER messages

The UE sends a REGISTER request to P-CSCF indicating that it is able to negotiate security mechanisms and that it supports manually keyed IPsec with two different algorithms (Step 1 of figure above). P-CSCF acts as a transparent back-to-back UA, and forwards the request to S-CSCF (Step 2). S-CSCF challenges the UE with HTTP Digest challenge (Step 3). P-CSCF has a local policy that it accpts only one algorithm. This kind of policy can be done if all UEs are known to support this algorithm, and if the algorithm has not been broken. P-CSCF challenges the UE with "494 Security Agreement Required", and passes also the authentication challenge from S-CSCF to the UE (Step 4). P-CSCF also opens the IPsec SAs for UE.

UE generates the session keys for IPsec using the authentication challenge from the S-CSCF. UE chooses the first security mechanism (according to the "q" value) from the Security-Server header that is known to it. In this case, there is only one mechanism to be chosen. UE opens up the IPsec SAs, copies the content of Security-Server header to the Security-Verify header, and constructs the second REGISTER message. The second REGISTER is sent to P-CSCF (Step 5).

P-CSCF check that content of the Security-Verify header corresponds with the content of the Security-Server header it sent in Step 4. If the contents of the headers are the same, P-CSCF forwards the message to S-CSCF (Step 6). Otherwise, P-CSCF would send an error message to the UE and terminate the created IPsec SAs because there is an attacker who has modified security parameters.

S-CSCF authenticates the UE, and responds with 200 OK if the authentication was successful (Step 7). P-CSCF forwards the response to UE (Step 8).

The content of the messages in Steps 1, 4 and 5 are as follows:

(1)
REGISTER sip:s-cscf.home.com SIP/2.0
Security-Client: ipsec-man; alg=hmac-md5-96; alg=hmac-sha-1-96; prot=esp; mod=trans;
spi-u-tcp=48275342; spi-u-udp=58394603; port-u-tcp=5682234598; port-u-udp= 5834529
Require: sec-agree
Proxy-Require: sec-agree

By using the defualt value concept (1) would like like (Note (1)≡(1'))
(1')
REGISTER sip:s-cscf.home.com SIP/2.0
Security-Client: ipsec-man; alg=hmac-md5-96; alg=hmac-sha-1-96;
spi-u-tcp=48275342; spi-u-udp=58394603; port-u-tcp=5682234598; port-u-udp= 5834529
Require: sec-agree
Proxy-Require: sec-agree

(4) and (5) is exemplifying the security agreement without using the default values but of course it should be applied in a real scenario in IMS.

(4)
SIP/2.0 494 Security Agreement Required
Security-Server: ipsec-man; q=0.1; alg=hmac-sha-1-96; prot=esp; mod=trans;
spi-p-tcp=95832569; spi-p-udp=968934534; port-p-tcp=89998345; port-p-udp=86567345

(5)
REGISTER sip:s-cscf.home.com SIP/2.0
Security-Verify: ipsec-man; q=0.1; alg=hmac-sha-1-96; prot=esp; mod=trans;
spi-p-tcp=95832569; spi-p-udp=968934534; port-p-tcp=89998345; port-p-udp=86567345
Require: sec-agree
Proxy-Require: sec-agree

4. Conclusions

Ericsson proposes the use of default values concept for IMS and SA3 is encouraged to accept the accompanied CR to 33.203 Appendix H.

References

[sec-agree] Arkko et al, *Security Mechanism Agreement for SIP Sessions*, IETF, Work in progress, June 2002, draft-ietf-sip-sec-agree-03.txt.