ORACLE COMMUNICATIONS

TECH NOTE IMS-AKA

.

Status of this memo

Oracle Tech Notes are working documents of the Professional Services department of Oracle, Inc. Note that other groups may also distribute working documents as Tech Notes.
Tech Notes are working documents valid until explicitly obsoleted, and may be updated, replaced or obsoleted by other documents at any time. It is recommended to use Tech Notes as reference material as well as to cite them in other works in progress.

Abstract

The use of the RFC 2119 keywords is an attempt to assign the correct requirement levels ("MUST", "SHOULD", "MAY", etc.).

IMS-AKA (Authentication and Key Agreement) is the mechanism defined by 3GPP for authenticating SIP registration and deriving keys for encrypting SIP signaling exchanged between endpoints (UE) and Proxy-CSCF using IPSec.
This Technical Note documents a basic testing activity performed by Systems Engineering in Oracle labs, with the purpose of learning about IMS-AKA support on the ESBC using open-source tools for lab testing.

Applicability

This document is applicable to 4600,6100 and 6300 series ESBCs.

Contents

1 Scope

1.1 Goals

This activity was an informal testing effort aimed at learning about IMS-AKA support on the ESBC and basic lab testing using open-source tools..

1.2 Non-Goals

This is not a full verification of IMS-AKA functionality or standards compliancy.

1.3 Intended Audience

This document is intended for use by Oracle HQ and Field Based Engineers. It assumes the reader is familiar with basic operations of the ESBC, and has attended the following training course(s) (or has equivalent experience):

EDU-CAB-C-CLI Net-Net 4000/3000 Configuration Basics

Further, the test plans enclosed assume familiarity with the ESBC's ACLI command line interface, retrieving and reviewing log files generated by the ESBC, standard network analysis tools (Wireshark/tcpdump), and all protocols involved in the activity.

2 Application Overview

IMS-AKA (Authentication and Key Agreement) is the mechanism used in the IP Multimedia Subsystem, defined by 3GPP, for authenticating SIP registration and deriving keys for encrypting SIP signaling exchanged between endpoints (UE) and the ESBC (Proxy-CSCF) using IPSec.

IMS-AKA uses the Security Mechanism Agreement for SIP defined in [1]. The keys for the IPSec security associations between the UE and the P-CSCF are sent to the P-CSCF in the 401 challenge to the first REGISTER, and the UE independently derives these same keys using the challenge information and the stored secret key. All the signaling starting with the 2nd REGISTER (with credentials) is sent encrypted in the established IPSec SAs.

The relevant 3GPP specifications are TS 24.229 and TS 33.203

3 Software/Hardware/Tools

3.1 ESBC Hardware and Software Requirements

ESBC Platform	Mainboard Rev.	Bootloader	Software Version/Patch
NN4600	Functional Rev: 2.15 Board Rev: 3 Format Rev: 3 Manufacturer: Benchmark	Date: 10/17/2006 13:04:28	ECZ810 GA

3.2 Test Tool / Third Party Equipment used for Feature research and Testing

Third Party Platform	Software Version/Patch
SIPp / Linux	Fedora Core 4 with ipsec-tools-0.5-4 Patched SIPp version (from http://www.openimscore.org/node/85)
OpenIMSCore	SVN checkout around Feb 5th 2008 - svn checkout svn://svn.berlios.de/openimscore/ser_ims/trunk - svn checkout svn://svn.berlios.de/openimscore/FHoSS/trunk

Some extra tweaks to SIPp ipsec scripts required for HMAC-SHA1 and 3DES (key expansion).

The full SIPp tgz with tweaked ipsec scripts is attached here

SIPp_IPSEC_patched
_with_KeyExpansion.t

3.2.1 Configuring SIPp

Three SIPp instances are launched in sequence, using the following script:

```
#!/bin/bash
./sipp -t u1 -i 172.18.1.200 -p 3061 172.18.1.30:5060 -sf scenarios/regIPSEC1.xml -m 1 -trace_err -ap alice -
auth_pipe b.dat
sleep 3
./sipp -t u1 -i 172.18.1.200 -p 12345 172.18.1.30:7000 -sf scenarios/regIPSEC2.xml -m 1 -trace_err -inf
spis.csv -ap alice -auth_pipe b.dat
./sipp -t u1 -i 172.18.1.200 -p 3062 -sf scenarios/regIPSEC3.xml -m 1 -trace_err
```

These are the referenced scenario files:

scenarios/regIPSEC1.xml:

- sends first REGISTER to ESBC’s unprotected access sip-port (5060)
- receives 401 reply and establishes security associations by running ipsec/ipsec_E_* scripts using parameters obtained from the 401 reply

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE scenario SYSTEM "sipp.dtd">

<scenario name="registration">

<send retrans="500">
<![CDATA[
REGISTER sip:selab.com SIP/2.0
Via: SIP/2.0/[transport] [local_ip]:3061;branch=[branch]
Max-Forwards: 20
From: "alice" <sip:alice@selab.com>;tag=[call_number]
To: "alice" <sip:alice@selab.com>
P-Access-Network-Info: 3GPP-UTRAN-TDD;utran-cell-id-3gpp=C359A3913B20E
Call-ID: reg///[call_id]
CSeq: 1 REGISTER
Contact: <sip:alice@[local_ip]:3061>
Expires: 300
Content-Length: 0
Security-Client: ipsec-3gpp; ealg=aes-cbc; alg=hmac-md5-96; spi-c=1024; spi-s=2048; port-c=12345; port-s=3062; q=0.1
User-Agent: Sipp v1.1-TLS, version 20061124
Authorization: Digest username="alice@selab.com", realm="selab.com"
Supported: path
Require: sec-agree
Proxy-Require: sec-agree
]]>
</send>
<!-- aes-cbc hmac-sha-1-96 -->
<recv response="401" auth="true" rtd="true">
<action>
        <ereg regexp=".*" search_in="hdr" header="Security-Server:" assign_to="1" />

        <ereg regexp="ealg=([^;]*)" search_in="[$1]" assign_to="2,8" />
        <ereg regexp="[^e]alg=([^;]*)" search_in="[$1]" assign_to="3,9" />
        <ereg regexp="spi-c=([^;]*)" search_in="[$1]" assign_to="4,10" />
        <ereg regexp="spi-s=([^;]*)" search_in="[$1]" assign_to="5,11" />
        <ereg regexp="port-c=([0-9]{4,5})" search_in="[$1]" assign_to="6,12" />
        <ereg regexp="port-s=([0-9]{4,5})" search_in="[$1]" assign_to="7,13" />

        <exec command="echo '[local_ip] 12345 [remote_ip] [$13] [$11] [$8] 0x[ck_key] [$9] 0x[ik_key]' > debug1 " />
        <exec command="ipsec/ipsec_E_Out_Req.sh [local_ip] 12345 [remote_ip] [$13] [$11] [$8] 0x[ck_key] [$9] 0x[ik_key] " />

        <exec command="echo '[local_ip] 3062 [remote_ip] [$12] [$10] [$8] 0x[ck_key] [$9] 0x[ik_key]' > debug2 " />
        <exec command="ipsec/ipsec_E_Out_Rpl.sh [local_ip] 3062 [remote_ip] [$12] [$10] [$8] 0x[ck_key] [$9] 0x[ik_key] " />

        <exec command="echo '[local_ip] 3062 [remote_ip] [$12] 2028  [$8] 0x[ck_key] [$9] 0x[ik_key]' > debug3" />
        <exec command="ipsec/ipsec_E_Inc_Req.sh [local_ip] 3062 [remote_ip] [$12] 2028  [$8] 0x[ck_key] [$9] 0x[ik_key] " />

        <exec command="echo '[local_ip] 12345 [remote_ip] [$13] 1024  [$8] 0x[ck_key] [$9] 0x[ik_key]' > debug4 " />
        <exec command="ipsec/ipsec_E_Inc_Rpl.sh [local_ip] 12345 [remote_ip] [$13] 1024  [$8] 0x[ck_key] [$9] 0x[ik_key] " />

        <exec command="echo SEQUENTIAL > spis.csv" />
        <exec command="echo '[$10];[$11];[$12];[$13]' >> spis.csv" />
</action>
</recv>

<ResponseTimeRepartition value="10, 20"/>
<CallLengthRepartition value="10"/>

</scenario>
```

scenarios/regIPSEC2.xml:

- sends second REGISTER (with auth credentials) from UE port-c (12345) to ESBC’s port-s (7000), protected by the installed IPSec SA

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE scenario SYSTEM "sipp.dtd">

<scenario name="registration">

<send retrans="500">
<![CDATA[
REGISTER sip:selab.com SIP/2.0
Via: SIP/2.0/[transport] [local_ip]:3062;branch=[branch]
Max-Forwards: 20
From: "alice" <sip:alice@selab.com>;tag=[call_number]
To: "alice" <sip:alice@selab.com>
P-Access-Network-Info: 3GPP-UTRAN-TDD;utran-cell-id-3gpp=C359A3913B20E
Call-ID: reg///[call_id]
CSeq: 2 REGISTER
Contact: <sip:alice@[local_ip]:3062>
Expires: 300
Content-Length: 0
Security-Client: ipsec-3gpp; ealg=aes-cbc; alg=hmac-md5-96; spi-c=1024; spi-s=2048; port-c=12345; port-s=3062; q=0.1
Security-Verify: ipsec-3gpp; ealg=aes-cbc; alg=hmac-md5-96; spi-c=[field0]; spi-s=[field1]; port-c=[field2]; port-s=[field3]; q=0.1
User-Agent: Sipp v1.1-TLS, version 20061124
[authentication username=alice@selab.com password=alice]
Supported: path
Require: sec-agree
Proxy-Require: sec-agree
]]>
</send>


<ResponseTimeRepartition value="10, 20"/>
<CallLengthRepartition value="10"/>

</scenario>
```

scenarios/regIPSEC3.xml:

- receives 200 OK on UE port-s (3062), protected by the installed IPSec SA

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE scenario SYSTEM "sipp.dtd">

<scenario name="registration">

<recv response="200">
</recv>

<ResponseTimeRepartition value="10, 20"/>
<CallLengthRepartition value="10"/>

</scenario>
```

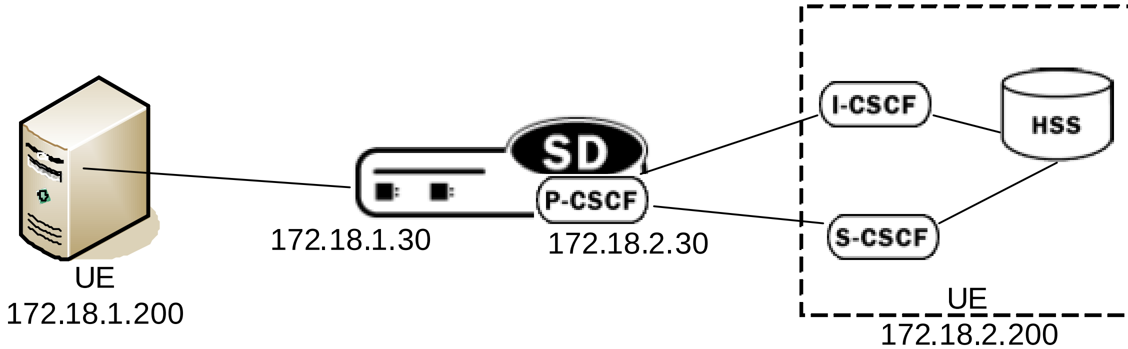
3.2.2 Configuring OpenIMSCore

Perform standard OpenIMSCore installation (see http://www.openimscore.org/installation_guide).
In this case we changed the network domain to selab.com both in the ser_ims and FHoSS config. User ‘alice@selab.com’ is provisioned using the HSS web interface, with secret key ‘alice’

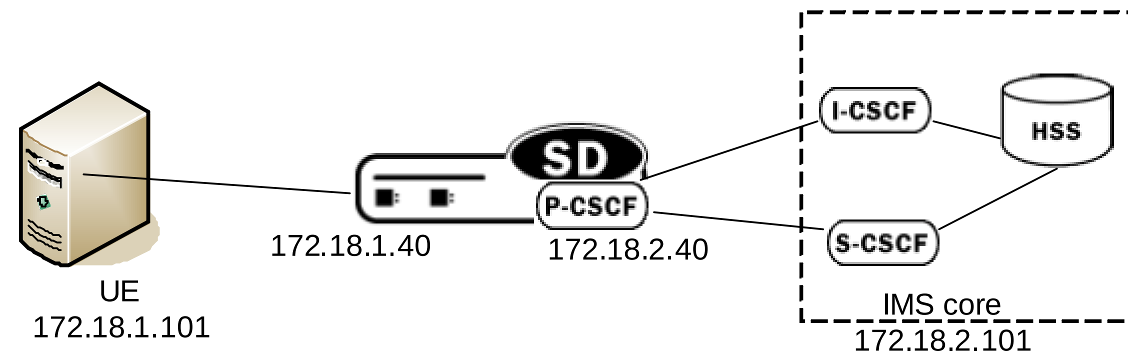
.

3.3 Test Bed Diagrams

Test bed 1:

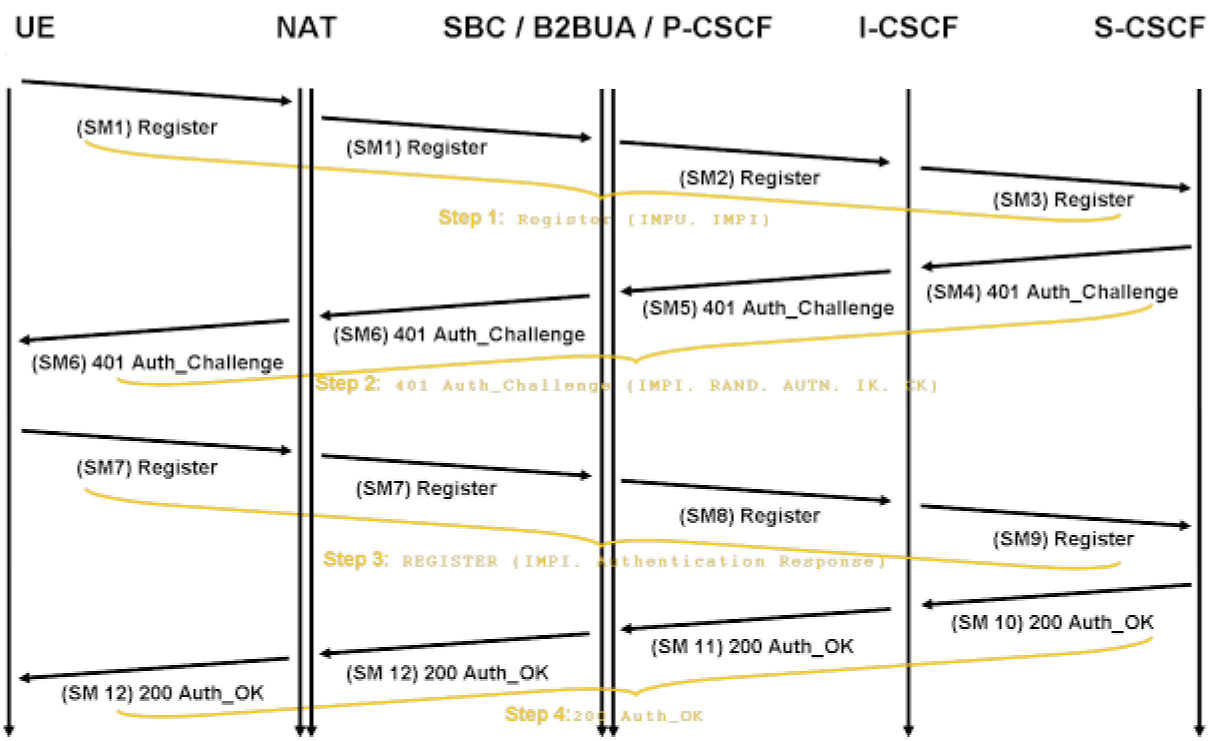


Test bed 2:



4 Sample Call Flows & Diagrams

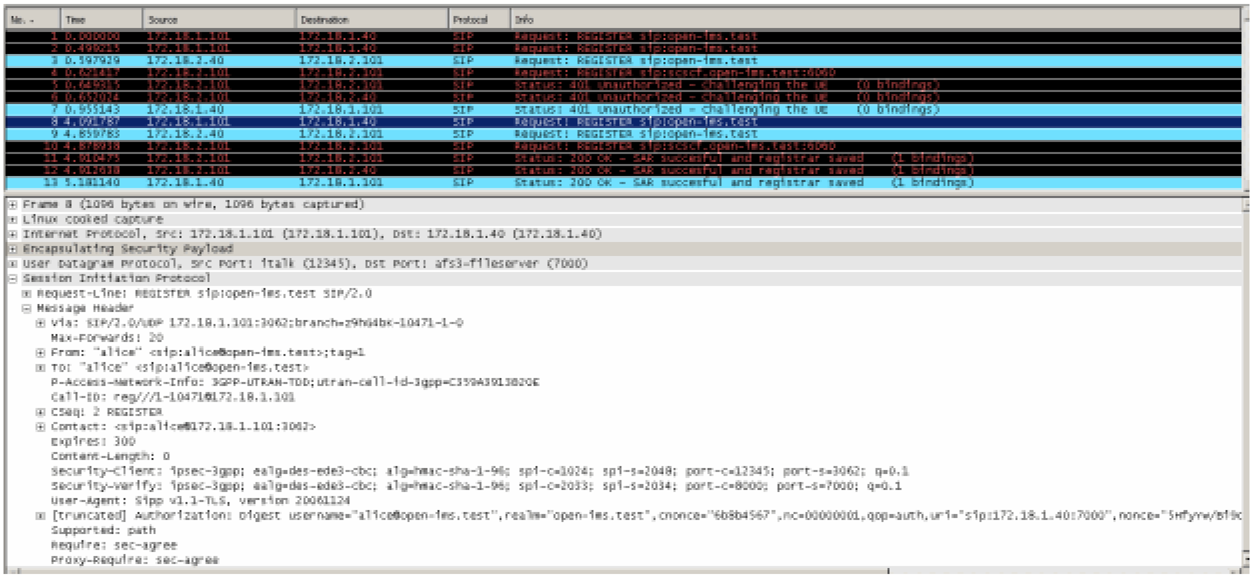
A registration flow using IMS-AKA is described in the following figure:



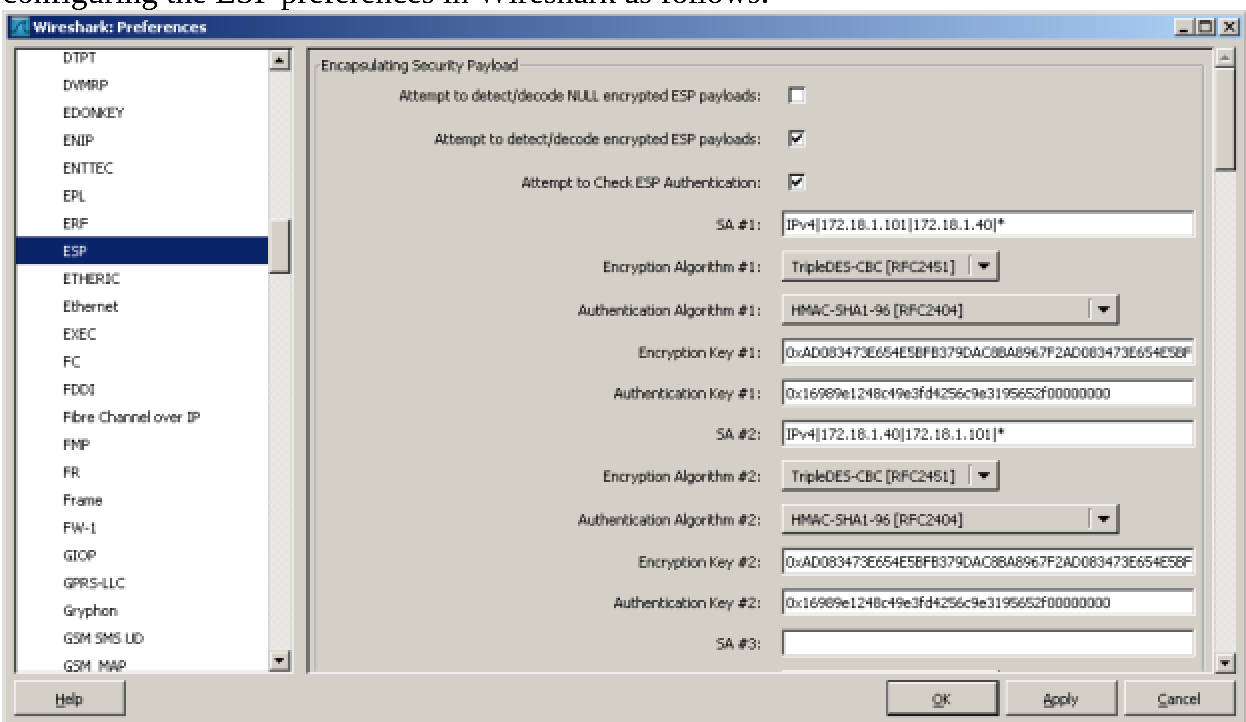
Please note that this activity did not include testing with UE behind NAT, but the call flow is similar.

The following zip file contains:
a Wireshark capture
logs (log.secured, log.sipd, sipmsg.log)
support-info
output of “show security ipsec sad M00:0 detail”

(Using test bed 2)



The Wireshark capture contains a successful registration, where the encrypted REGISTER and 200 OK are decrypted using the keys obtained from the 401 reply after the appropriate key expansion (required for HMAC-SHA1 and 3DES as per). This can be achieved by configuring the ESP preferences in Wireshark as follows:



5 Test Configuration and Debugging

5.1 ESBC Sample Configuration

Configure ESBC as P-CSCF
Xnet: Professional Services > EMEA, Systems Engineering > Technical Documents > Made in EMEA > P-CSCF Configuration Guidelines v1 3.pdf

security > ims-aka-profile

```
ims-aka-profile
        name                            test
        protected-client-port           8000
        protected-server-port           7000
        encr-alg-list                   aes-cbc des-ede3-cbc
        auth-alg-list                   hmac-sha-1-96 hmac-md5-96
```

sip-interface

- Enable ims-aka-feature and configure sip-port > ims-aka-profile

```
sip-interface
        state                           enabled
        realm-id                        access1
        description
        sip-port
                address                         172.18.1.30
                port                            5060
                transport-protocol              UDP
                tls-profile
                allow-anonymous                 registered
                ims-aka-profile                 test
        ims-aka-feature                 enabled
```

security > ipsec > security-policy

```
security-policy
    name                           pol1
    network-interface              M00:0
    priority                       0
    local-ip-addr-match            172.18.1.30
    remote-ip-addr-match           172.18.1.0
    local-port-match               5060
    remote-port-match              0
    trans-protocol-match           ALL
    direction                      both
    local-ip-mask                  255.255.255.255
    remote-ip-mask                 255.255.255.0
    action                         allow
    ike-sainfo-name
    outbound-sa-fine-grained-mask
        local-ip-mask                  255.255.255.255
        remote-ip-mask                 255.255.255.255
        local-port-mask                0
        remote-port-mask               0
        trans-protocol-mask            0
        valid                          enabled
        vlan-mask                      0xFFF
```

```
security-policy
    name                           pol2
    network-interface              M00:0
    priority                       1
    local-ip-addr-match            172.18.1.30
    remote-ip-addr-match           172.18.1.0
    local-port-match               0
    remote-port-match              0
    trans-protocol-match           ALL
    direction                      both
    local-ip-mask                  255.255.255.255
    remote-ip-mask                 255.255.255.0
    action                         ipsec
    ike-sainfo-name
    outbound-sa-fine-grained-mask
        local-ip-mask                  255.255.255.255
        remote-ip-mask                 255.255.255.255
        local-port-mask                65535
        remote-port-mask               65535
        trans-protocol-mask            0
        valid                          enabled
        vlan-mask                      0xFFF
```

Add a sip-feature for “sec-agree”

```
sip-feature
    name                           sec-agree
    realm
    support-mode-inbound           Pass
    require-mode-inbound           Pass
    proxy-require-mode-inbound     Pass
    support-mode-outbound          Pass
    require-mode-outbound          Pass
    proxy-require-mode-outbound    Pass
```

HA :security > IPSec > ipsec-global-config

In addition to the normal HA config ,for IMS-AKA feature ,configure red-ipsec-port, red-max-trans, red-sync-start-time, red-sync-comp-time.

The following snapshot gives steps to configure basic HA with IMS-AKA.

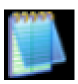
```
connecticut(system)# redundancy
connecticut(redundancy)# select
connecticut(redundancy)# peers
connecticut(rdncy-peer)# name connecticut
connecticut(rdncy-peer)# type Primary
connecticut(rdncy-peer)# destinations
connecticut(rdncy-peer-dest)# address 169.254.1.1:9090
connecticut(rdncy-peer-dest)# network-interface wancom1:0
connecticut(rdncy-peer-dest)# done
destination
    address                    169.254.1.1:9090
    network-interface              wancom1:0

connecticut(rdncy-peer-dest)# address 169.254.2.1:9090
connecticut(rdncy-peer-dest)# network-interface wancom2:0
connecticut(rdncy-peer-dest)# done
destination
```

```
    address                      169.254.2.1:9090
    network-interface            wancom2:0

connecticut(rdncy-peer-dest)# exit
connecticut(rdncy-peer)# done
peer
    name                         connecticut
    state                        enabled
    type                         Primary
    destination
        address                      169.254.1.1:9090
        network-interface            wancom1:0
    destination
        address                      169.254.2.1:9090
        network-interface            wancom2:0

connecticut(rdncy-peer)# name delaware
connecticut(rdncy-peer)# type Secondary
connecticut(rdncy-peer)# destinations
connecticut(rdncy-peer-dest)# address 169.254.1.2:9090
connecticut(rdncy-peer-dest)# network-interface wancom1:0
connecticut(rdncy-peer-dest)# done
destination^M
    address                      169.254.1.2:9090
    network-interface            wancom1:0
connecticut(rdncy-peer-dest)# address 169.254.2.2:9090
connecticut(rdncy-peer-dest)# network-interface wancom2:0
connecticut(rdncy-peer-dest)# done
destination
    address                      169.254.2.2:9090
    network-interface            wancom2:0

connecticut(rdncy-peer-dest)# exit
connecticut(rdncy-peer)# done
peer^M
    name                         delaware
    state                        enabled
    type                         Secondary
    destination
        address                      169.254.1.2:9090
        network-interface            wancom1:0
    destination^M
        address                      169.254.2.2:9090
        network-interface            wancom2:

connecticut(rdncy-peer)# exit
connecticut(redundancy)# done
redundancy-config^M
    state                        enabled
    log-level                    INFO
    health-threshold             75
    emergency-threshold          50
    port                         9090
    advertisement-time           500
    percent-drift                21
    initial-time                 1250
```

```
	becoming-standby-time			180000
	becoming-active-time			100
	cfg-port				1987
	cfg-max-trans				10000
	cfg-sync-start-time			5000
	cfg-sync-comp-time			1000
	gateway-heartbeat-interval		0
	gateway-heartbeat-retry			0
	gateway-heartbeat-timeout		1
	gateway-heartbeat-health		0
	media-if-peercheck-time			0
	peer^M
		name				connecticut
state					enabled
		type				Primary
		destination
			address			169.254.1.1:9090
			network-interface	wancom1:0
		destination
			address			169.254.2.1:9090
			network-interface	wancom2:0
	peer
		name				delaware
		state				enabled
		type				Secondary
		destination
			address			169.254.1.2:9090
			network-interface	wancom1:0
		destination
			address			169.254.2.2:9090
			network-interface	wancom2:0
	options
	last-modified-by			admin@10.196.147.157
	last-modified-date			2018-08-08 05:25:18
```

```
connecticut(configure)# security^M
```

```
connecticut(security)# ipsec^M
connecticut(ipsec)# ipsec-global-config^M
connecticut(ipsec-global-config)# select^M
connecticut(ipsec-global-config)# red-ipsec-port 1994^M
connecticut(ipsec-global-config)# done^M
ipsec-global-config^M
        red-ipsec-port                  1994^M
        red-max-trans                   10000^M
        red-sync-start-time             5000^M
        red-sync-comp-time              1000^M
        rekey-on-sn-overflow            enabled^M
        options                         ^M
        last-modified-by                admin@10.196.147.157^M
        last-modified-date              2018-08-08 05:25:01^M
```



Configuration.txt

5.2 ACLI Commands and Statistical Definitions

Below are sample output for ACLI show commands.

```
# show sa stats ims-aka
12:35:03-191
SA Statistics                                      ---- Lifetime ----
                                                 Recent      Total  PerMax
IMS-AKA Statistics
ADD-SA Req Rcvd                                  0          0       0
ADD-SA Success Resp Sent                         0          0       0
ADD-SA Fail Resp Sent                            0          0       0
DEL-SA Req Rcvd                                  0          0       0
DEL-SA Success Resp Sent                         0          0       0
DEL-SA Fail Resp Sent                            0          0       0
SA Added                                         0          0       0
SA Add Failed                                    0          0       0
SA Deleted                                       0          0       0
SA Delete Failed                                 0          0       0


# show security ipsec sad M00:0 detail
IPSEC security-association-database for interface 'M00:0':
Displaying SA's that match the following criteria -
        spi                       : any
        direction                 : both
        ipsec-proto               : any
        src-addr-prefix           : any
        src-port                  : any
        dst-addr-prefix           : any
        dst-port                  : any
        trans-proto               : ALL

Inbound, SPI: 2033
        destination-address       : 172.18.1.40
        vlan-id                   : 0
        ipsec-protocol            : ESP
        sad-index                 : 0
```

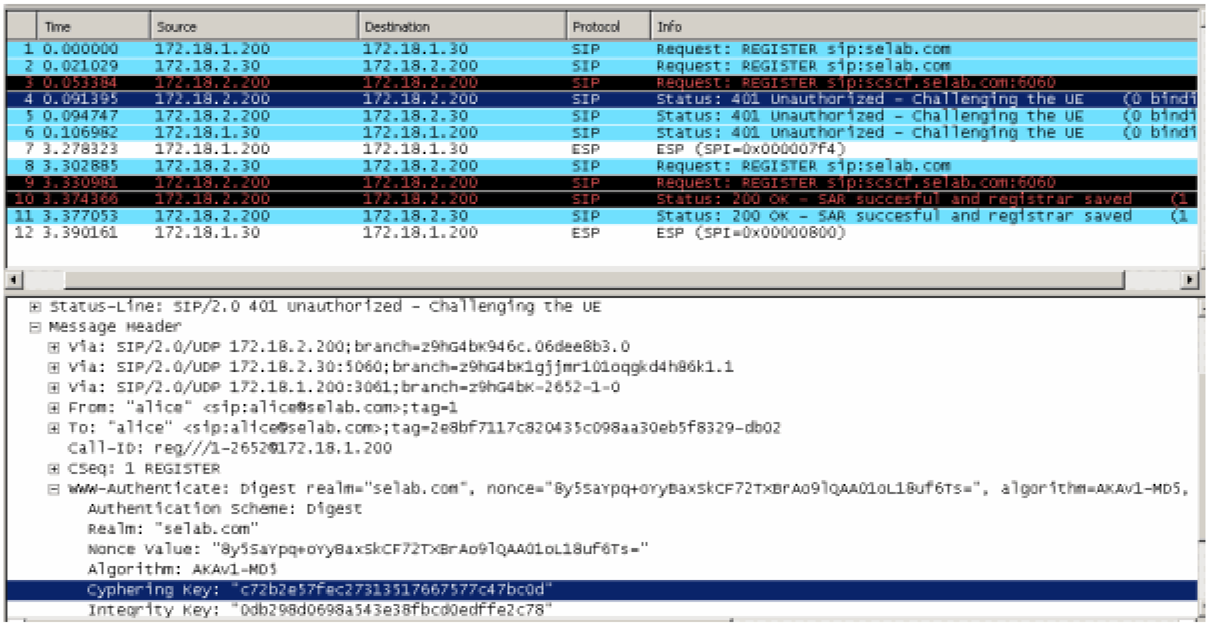
```
        encr-algo                 : 3des
        auth-algo                 : hmac-sha1
        match fields:
               src-ip            : 172.18.1.101
               dst-ip            : 172.18.1.40
               src-port          : 3062
               dst-port          : 8000
               vlan-id           : 0
               trans-proto       : ALL
        mask fields:
               src-ip            : 255.255.255.255
               dst-ip            : 255.255.255.255
               src-port          : 1
               dst-port          : 1
               vlan-id           : 0
               protocol          : 0
        flags -
               26932080, ls: 40000000
        byte count limit -
               hard ms: 0xFFFFFFFF, hard ls: 0xFFFFFFFF
               soft ms: 0xFFFFFFFF, soft ls: 0xFFFFFFFF
        hard limit -
               hard: 0xFFFFFFFF, soft: 0xFFFFFFFF
               seq ms: 0x       0, seq ls: 0x       0

Inbound, SPI: 2034
        destination-address       : 172.18.1.40
        vlan-id                   : 0
        ipsec-protocol            : ESP
        sad-index                 : 1
        encr-algo                 : 3des
        auth-algo                 : hmac-sha1
        match fields:
               src-ip            : 172.18.1.101
               dst-ip            : 172.18.1.40
               src-port          : 12345
               dst-port          : 7000
               vlan-id           : 0
               trans-proto       : ALL
        mask fields:
               src-ip            : 255.255.255.255
               dst-ip            : 255.255.255.255
               src-port          : 1
               dst-port          : 1
               vlan-id           : 0
               protocol          : 0
        flags -
               26932080, ls: 40000000
        byte count limit -
               hard ms: 0xFFFFFFFF, hard ls: 0xFFFFFFFF
               soft ms: 0xFFFFFFFF, soft ls: 0xFFFFFFFF
        hard limit -
               hard: 0xFFFFFFFF, soft: 0xFFFFFFFF
               seq ms: 0x       0, seq ls: 0x       0

Outbound, SPI: 2048
        source-address            : 172.18.1.40
        destination-address       : 172.18.1.101
        source-port               : 8000
        destination-port          : 3062
        trans-proto               : ALL
        vlan-id                   : 0
        sad-index                 : 0
        encr-algo                 : 3des
        auth-algo                 : hmac-sha1
        mtu                       : 1428
        flags -
               0x 293000040000000
        byte count limit -
               hard ms: 0xFFFFFFFF, hard ls: 0xFFFFFFFF
               soft ms: 0xFFFFFFFF, soft ls: 0xFFFFFFFF
```

```
        time limit -
                hard: 0xFFFFFFFF, soft: 0xFFFFFFFF
                seq ms: 0x       0, seq ls: 0x       1

Outbound, SPI: 1024
        source-address            : 172.18.1.40
        destination-address       : 172.18.1.101
        source-port               : 7000
        destination-port          : 12345
        trans-proto               : ALL
        vlan-id                   : 0
        sad-index                 : 1
        encr-algo                 : 3des
        auth-algo                 : hmac-sha1
        mtu                       : 1428
        flags -
                0x 2930000400000000
        byte count limit -
                hard ms: 0xFFFFFFFF, hard ls: 0xFFFFFFFF
                soft ms: 0xFFFFFFFF, soft ls: 0xFFFFFFFF
        time limit -
                hard: 0xFFFFFFFF, soft: 0xFFFFFFFF
                seq ms: 0x       0, seq ls: 0x       0



# show security ipsec statistics M00:0 sad

<enter>                         select all entries
direction                       select by direction
dst-addr-prefix                 select by remote ip-address prefix
dst-port                        select by destination port
ipsec-protocol                  select by ipsec protocol
spi                             select by security-policy-index
src-addr-prefix                 select by source ip address prefix
src-port                        select by source port
trans-protocol                  select by transport protocol
```

5.3 Debugging Methodology and Techniques

Check sipmsg.log, log.sipd, log.secured

Check keys in WWW-Authenticate header of 401 Unauthorized reply



And use them in Wireshark (Edit>Preferences>Protocols>ESP ...) to be able to decrypt IPSec packets

On a Linux UE, use *setkey -DpP* for dumping the security associations (SAD and SPD entries)

Using NULL encryption algorithm can help

6 Test Cases

The following registration test cases were executed successfully for understanding this feature.

6.1 Registration using HMAC-MD5 and AES

TC# 1	**Description: Registration using HMAC-MD5 and AES**	
Step	**Action**	**Result / Defect ID**
1	Configure SIPp (scenarios/regIPSEC1.xml) to use HMAC-MD5 and AES algorithms	-
2	Register user	OK

6.2 Registration using HMAC-SHA1 and 3DES

TC#2	**Description: Registration using HMAC-SHA1 and 3DES**	
Step	**Action**	**Result / Defect ID**
1	Configure SIPp (scenarios/regIPSEC1.xml) to use HMAC-SHA1 and 3DES algorithms	-
2	Register user	OK

7 Conclusion

IMS-AKA support in the ESBC (acting as P-CSCF) was verified in Systems Engineering lab.

8 Normative References

[1] IETF RFC 3329 – “Security Mechanism Agreement for SIP”
[2] 3GPP TS 24.229 – “IP multimedia call control protocol based on SIP and SDP; stage 3”
[3] 3GPP TS 33.203 – "3G security; Access security for IP-based services"

9 Author’s Address

Gayathri Balakrishnan
Oracle 100 Crosby Dr
Bedford, MA 01730
email:gayathribalakrishnan633@gmail.com

10 Disclaimer

The content in this document is for informational purposes only and is subject to change by Oracle without notice. While reasonable efforts have been made in the preparation of this publication to assure its accuracy, Oracle assumes no liability resulting from technical or editorial errors or omissions, or for any damages resulting from the use of this information. Unless specifically included in a written agreement with Oracle, Oracle has no obligation to develop or deliver any future release or upgrade or any feature, enhancement or function.

11 Full Copyright Statement

Copyright @ Oracle (2010). All rights reserved. Oracle, Session-Aware Networking, Net-Net and related marks are trademarks of Oracle. All other brand names are trademarks or registered trademarks of their respective companies.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implantation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice, disclaimer, and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to Oracle or other referenced organizations, except as needed for the purpose of developing open standards.

The limited permission granted above are perpetual and will not be revoked by Oracle or its successors or assigns.

This document and the information contained herein is provided on an “AS IS” basis and ORACLE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE FO THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.