## 1，注册代码流程

//=========================enter SipMgnt\_server ==============================

Polling\_Callback\_Handler

|-🡪pCurPolling->PollingOperation ();

|-🡪 case HealthMonitoring::PollingFGHM:

| |-🡪 mServer->pHealthMon->PerformForegroundHealMon();

| | |-🡪 SipMgnt\_TRUE == pSipServer->OpStatusIns()

| | |-🡪 SipVSP::getInstance(GetVSPName())->GetFGHMMethod() == FGHMMethod\_Option

| | | |-🡪 SipVSP\* pSipVspInstance = SipVSP::FindInVSPMappingTable(vspName);

| | |-🡪SipVSP::getInstance(GetVSPName())->IsRedunEnabled() == SipMgnt\_TRUE

| | | |-🡪 SipVSP\* pSipVspInstance = SipVSP::FindInVSPMappingTable(vspName);

//=========================enter SipMgnt ========================================

//此为周期性的监测，3次监测

void SipMgnt\_GetStatus(void\* pMsgHeader,void\* resp)

|-🡪SipMgnt::GetStatus((voipMsgHeader\_t\*)pMsgHeader, resp);

| |-🡪PrintReceiveMsg(pMsg);

| |-🡪case MSG\_GET\_SIPSUBSCRIBER\_REQ: SipSubscriberMgr::GetStatus(pMsg,resp);

| | |-🡪 Si pTermination::SipTermPointers[dIndex]->ProcessGetReq(respMsg);

| | | |-🡪 GetContactInfo((S8\*)pGetResp->contactInfo);

| | |-🡪 if (AdminStatusDown()|| AdminStatusDeleted()|| !CorrespondingAdmStatusUp())

//============================================================================

//=========================enter SipMgnt ========================================

void SipMgnt\_msgHandler(void\* pMsgHeader)

|-🡪 SipMgnt::MsgHandler((voipMsgHeader\_t\*)pMsgHeader);

| |-🡪 PrintReceiveMsg(pMsg);

| |-🡪 SipSubscriberMgr::MsgHandler(pMsg);

| | |-🡪 case MSG\_MODIFY\_SIPSUBSCRIBER\_REQ: getInstance()-> HandleModifySubscriberReq(pMsg);

| | | |-🡪 PrintSubscriberSetReq(pSubscriberSetReq);

| | | |-🡪SipTermination::SipTermPointers[dIndex]->ProcessConfigReq(pSubscriberSetReq);

| | | | |-🡪 ChgAdminStatusUp();

| | | | | |-🡪 OnAdminStatusUp (this);

| | | | | | |-🡪updateLocalAddress();

| | | | | | |-🡪SendSupplServiceToPal();

| | | | | | |-🡪m\_Polling->StartRegister((void\*)this);

| | | | | | | |-🡪StartPolling(pSipTerm, pollingForRegister, Intv, "Termination Register");

| | | |-🡪AntiAvalSetPreInterval(pSubscriberSetReq->registerPreAvlaTimer);

| | | |-🡪SipMgnt::FillRespMsgHdr(pMsg, &respMsg.hdr);

| | | | |-🡪case MSG\_MODIFY\_SIPSUBSCRIBER\_REQ:

| | | | |-🡪PrintMsgHeader(pOutRespMsg);

| | | |-🡪SipMgnt::SendQMsg(pSubscriberSetReq->src\_queue\_id, (char\*)&respMsg, sizeof(respMsg));

//=========================enter SipMgnt\_termination============================

Polling\_Callback\_Handler

|-🡪pCurPolling->PollingOperation ();

| |-🡪case pollingForRegister: PerformRegister(SipMgnt\_FALSE);

| | |-🡪 StartPolling(pTerm, pollingForRegister,2\*1000,"Termination Register");

| | | |-🡪StopPolling();

| | |-🡪pTerm->ConfigRegisterData();

| | | |-🡪m\_RegLine->ProcessConfigDataRequest(m\_configData);

| | | | |-🡪 buildDomain(m\_domain\_name,termData.uri);

| | | | |-🡪 buildUserName(m\_user\_name,termData.uri);

| | | | |-🡪initRegRedirection();

| | | |-🡪pTerm->TriggerSendRegisterRequest(REGHANDLER\_REGISTER, NoRegisterInd);

| | | | |-🡪RouteInfoUpdateReq(&mRegisterRouteInfoCallback, E\_REASON\_REGISTER, context);

| | | | |-🡪SipServer \* pSrv = GetCurAssignedInsSipServer();

| | | | | |-🡪SipServer \*pSipServer = GetCurAssignedSipServer();

| | | | |-🡪false == pSrvEntry->DoSipServerLocationQuery(isLocatingPending, this)

| | | | | |-🡪qresult qres = SipServerLocator::getInstance()->locate(req, result);

| | | | | | |-🡪ret = transaction.locate();

| | | | | |-🡪UpdateSipServerList(result);

void RegisterRouteInfoCallback::onReady(unsigned long long context)

|-🡪 mOwner->SendRegisterRequest(regrequest);

| |-🡪 m\_RegLine->ProcessRegisterRequest(regrequest,m\_AssignedServer, bypassAvalancheControl);

| | |-🡪 FsmProcessEvent(&regReqEvent);

| | | |-🡪 pCurState->StateProcessEvent((void\*)this, pEvent);

| | | | |-🡪case SipRegEvent::SipReg\_regfsmRegisterReq: ProcessRegisterReq (pSipRegLine, pSipRegEvent);

| | | | | |-🡪pSipRegLine->DoReqisterRequest ( pSipRegLine->m\_bypassAvalancheControl,REGHANDLER\_REGISTER);

| | | | | | |-🡪addAcceptHeadertoRegOption(register\_option);

| | | | | | |-🡪SSAPIW::SipEngine::getInstance().doRegisterRequest ( m\_orig\_req\_id, register\_request,register\_option);

| | | | | | | |-🡪sendOk = AntiAvalSendRegistrationMsg((void\*)ss\_tx\_req);

| | | | | | | | |-🡪return pRegController->SendSipRequest(pSipMsg);

| | | | | |-🡪pSipRegLine->ChgStateTo(SipRegRegisteringState::Instance());

//=========================enter SSAPIW ========================================

| | |--🡪 ssapi\_parse\_uri(request.req\_uri, pEncInv);

| | |--🡪 SipSSApiInviteRequest\* ss\_inv\_req = pEncInv->EncodeInv(inv, request\_id, request, option);

| | |---🡪 ssapi\_send\_msg(ss\_inv\_req);

| | | |--🡪 sendToSipstack(SipSSApiMsg\* applmsg) // [SAPI]:send to sipstack

| | | | |----🡪 m\_connection.printMsg(applmsg);

| | | | |----🡪 m\_connection.writeMsg(buffer, YAPI\_FALSE);

Sip注册请求的内容：

[SAPI 05/01/70 20:38:14.023798 ]Tx\_Request (113) ; len = 922

nbr of attribs = 5

Request\_Id: 34 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

IpPortAddress: 32 31 2e 31 2e 31 2e 37 3a 35

35 38 30 39

SipMsg: REGISTER sip:pcgw-stdn.imsgroup0-001.qdc07.qd.lucentlab.com SIP/2.0

Route: <sip:135.252.37.124;lr>

From: <sip:+1328861310[SAPI 05/01/70 20:38:14.023880 ]2@qd.lucentlab.com;user=phone>;tag=b72e12N262c5-3f28

To: <sip:+13288613102@qd.lucentlab.com;user=phone>

Call-ID: 02-REG-FSM-0000088adjlkqqojlj@21.1.1.7

CSeq: 1 REGISTER

Max-Forwards: 30

Contact: sip:+13288613102@21.1.1.7:55809

Expires: 3600

Accept: application/sdp,multipart/mixed,application/[SAPI 05/01/70 20:38:14.023951 ]vnd.etsi.aoc+xml;schemaversion=2,application/vnd.etsi.aoc+xml;sv=2,application/simservs+xml,application/vnd.etsi.aoc+xml

Allow: INVITE,ACK,CANCEL,BYE,PRACK,UPDATE,INFO,NOTIFY,OPTIONS,SUBSCRIBE

Supported: path

User-Agent: Alcatel-Lucent MSAN

P-Access-Network-Info: xDSL;dsl-location=";21.1.1.7;132[SAPI 05/01/70 20:38:14.024021 ]88613102;;;"

Content-Length: 0

HeaderOffsets: 00 02 00 4c 00 62 00 03 00 6b

00 ae 00 04 00 b5 00 e2 00 05

00 ee 01 13 00 06 01 1c 01 25

00 07 01 36 01 37 00 09 01 43

01 61 00 0a 01 6d 01 70 00 0e

01 7b 01 89 00 0e 01 8b 01 99

00 0e 01 9b 01 c6 00 0e 01 c8

01 e8 00 0e 01 ea 02 01 [SAPI 05/01/70 20:38:14.024084 ]00 0e

02 03 02 1e 00 12 02 28 02 67

00 24 02 75 02 78 00 29 02 87

02 99 00 4a 02 b3 02 de 00 54

02 f1 02 f1

DNSPool\_Id: 00 01

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Enter SSAPI recv from sipstack \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

onRecvMsg()

|

|--🡪 printMsg(applmsg);

|--🡪 m\_client->sendToApplication(applmsg);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Enter SSAPI recv from sipstack \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

onRecvMsg()

|

|--🡪 printMsg(applmsg);

|--🡪 m\_client->sendToApplication(applmsg);

注册回复的消息内容：

Tx\_Reply (114) ; len = 776

nbr of attribs = 5

Request\_Id: 34 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

Tx\_Reference: 30 30 32 65 2d 30 30 30 30 30

30 30 64 2d 30 32 37 65

Method: 00 02

SipMsg: SIP/2.0 401 Unauthorized

Via: SIP/2.0/UDP 21.1.1.7:55809;received=21.1.1.7;branch=z9hG4bK\*002e-00[SAPI 05/01/70 20:38:14.073464 ]00000d-027e

From: <sip:+13288613102@qd.lucentlab.com;user=phone>;tag=b72e12N262c5-3f28

To: <sip:+13288613102@qd.lucentlab.com;user=phone>;tag=55be97dc-55c02a2812e5c0fd

Call-ID: 02-REG-FSM-0000088adjlkqqojlj@21.1.1.7

CSeq: 1 REGISTER

Date: Tue, 04 Aug 2015 02:57:44 GMT

Server: Alcatel-Lucent-HP[SAPI 05/01/70 20:38:14.073533 ]SS/3.0.3

WWW-Authenticate: Digest realm="qd.lucentlab.com",

nonce="b7c90355c02a286dbf3054aea940e9703dc8f84c0608",

opaque="ALU:QbkRBthOEgEQAkgVEwwHRAIBHgkdHwQCQ1lFRkdWEAkscnJqNCJpJDwpLiI5Ii4yfzE8OW9jZ25p",

algorithm=MD5,

qop="auth"

Content-Length: 0

HeaderOffsets: 00 00 00 1f 0[SAPI 05/01/70 20:38:14.073591 ]0 6c 00 03 00 75

00 b8 00 04 00 bf 01 0a 00 05

01 16 01 3b 00 06 01 44 01 4d

00 16 01 56 01 72 00 22 01 7d

01 95 00 2b 01 aa 02 84 00 54

02 97 02 97

//=========================enter SipMgnt ========================================

void SipMgnt\_SipCbItf::OnRegisterResponse(request\_id, reply)

|-🡪SipSubscriberMgr::getInstance()->HandleRegisterReply(request\_id, reply);

| |-🡪SipMgnt\_SipCbItf::OnPassiveHeartBeat(indexId);

| |-🡪 SipTermination::SipTermPointers[indexId]->m\_RegLine->ProcessRegisterReply(request\_id, reply);

| | |-🡪pServerEntry->alarmReport(alarm\_dnsLookUpFailure, alarm\_down);

| | |-🡪case 401:

| | | |-🡪FsmProcessEvent(&authReqEvent);

| | | | |-🡪pCurState->StateProcessEvent((void\*)this, pEvent);

| | | | | |-🡪case SipRegEvent::SipReg\_regfsmAuthenticationReq: ProcessAuthenticationReq (pSipRegLine, pSipRegEvent);

| | | | | | |-🡪pSipRegLine->CreateAuthenticationInstance();

| | | |-🡪case ReqOper\_Call\_Resend:

| | | | |-🡪DoReqisterRequest(true);

| | | | | |-🡪PopulateAutherizationData(register\_request.p\_auth);

| | | | | |-🡪addAcceptHeadertoRegOption(register\_option);

| | | | | |-🡪SSAPIW::SipEngine::getInstance().doRegisterRequest ( m\_orig\_req\_id, register\_request,register\_option);

| | | | | | |-🡪sendOk = AntiAvalSendRegistrationMsg((void\*)ss\_tx\_req);

| | | | | | | |-🡪return pRegController->SendSipRequest(pSipMsg);

| | |-🡪DoSendMsgToPMC(m\_line\_id-1, MSG\_PMC\_FAILED\_REGISTRATION\_INDICATION, SipMgnt\_TRUE);

//=========================enter SSAPIW ========================================

| | |--🡪 ssapi\_parse\_uri(request.req\_uri, pEncInv);

| | |--🡪 SipSSApiInviteRequest\* ss\_inv\_req = pEncInv->EncodeInv(inv, request\_id, request, option);

| | |---🡪 ssapi\_send\_msg(ss\_inv\_req);

| | | |--🡪 sendToSipstack(SipSSApiMsg\* applmsg) // [SAPI]:send to sipstack

| | | | |----🡪 m\_connection.printMsg(applmsg);

| | | | |----🡪 m\_connection.writeMsg(buffer, YAPI\_FALSE);

Sip注册请求的内容：

[SAPI 05/01/70 20:38:14.097457 ]Tx\_Request (113) ; len = 1304

nbr of attribs = 5

Request\_Id: 35 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

IpPortAddress: 32 31 2e 31 2e 31 2e 37 3a 35

35 38 30 39

SipMsg: REGISTER sip:pcgw-stdn.imsgroup0-001.qdc07.qd.lucentlab.com SIP/2.0

Route: <sip:135.252.37.124;lr>

From: <sip:+132886131[SAPI 05/01/70 20:38:14.097539 ]02@qd.lucentlab.com;user=phone>;tag=b72e12N262c5-3f28

To: <sip:+13288613102@qd.lucentlab.com;user=phone>

Call-ID: 02-REG-FSM-0000088adjlkqqojlj@21.1.1.7

CSeq: 2 REGISTER

Max-Forwards: 30

Contact: sip:+13288613102@21.1.1.7:55809

Expires: 3600

Accept: application/sdp,multipart/mixed,application[SAPI 05/01/70 20:38:14.097605 ]/vnd.etsi.aoc+xml;schemaversion=2,application/vnd.etsi.aoc+xml;sv=2,application/simservs+xml,application/vnd.etsi.aoc+xml

Allow: INVITE,ACK,CANCEL,BYE,PRACK,UPDATE,INFO,NOTIFY,OPTIONS,SUBSCRIBE

Authorization: Digest username="13288613102",realm="qd.lucentlab.com",nonce="b7c90355c02a286dbf3054aea94[SAPI 05/01/70 20:38:14.097670 ]0e9703dc8f84c0608",uri="sip:pcgw-stdn.imsgroup0-001.qdc07.qd.lucentlab.com",cnonce="2c5-6146-3b160a0",nc=00000001,qop=auth,response="d6fb900dfc89af170041ad6887d17e2e",algorithm=MD5,opaque="ALU:QbkRBthOEgEQAkgVEwwHRAIBHgkdHwQCQ1lFRkdWEAkscnJqNCJpJDwpLiI5Ii4yfzE8OW9jZ25p"

Supported: path

User-Agent:[SAPI 05/01/70 20:38:14.097731 ] Alcatel-Lucent MSAN

P-Access-Network-Info: xDSL;dsl-location=";21.1.1.7;13288613102;;;"

Content-Length: 0

HeaderOffsets: 00 02 00 4c 00 62 00 03 00 6b

00 ae 00 04 00 b5 00 e2 00 05

00 ee 01 13 00 06 01 1c 01 25

00 07 01 36 01 37 00 09 01 43

01 61 00 0a 01 6d 01 70 00 0e

01 7b 01 89[SAPI 05/01/70 20:38:14.097793 ] 00 0e 01 8b 01 99

00 0e 01 9b 01 c6 00 0e 01 c8

01 e8 00 0e 01 ea 02 01 00 0e

02 03 02 1e 00 12 02 28 02 67

00 14 02 79 03 df 00 24 03 ed

03 f0 00 29 03 ff 04 11 00 4a

04 2b 04 56 00 54 04 69 04 69

DNSPool\_Id: 00 01

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Enter SSAPI recv from sipstack \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

onRecvMsg()

|

|--🡪 printMsg(applmsg);

|--🡪 m\_client->sendToApplication(applmsg);

注册回复的消息内容：

[SAPI 05/01/70 20:38:14.151457 ]Tx\_Reply (114) ; len = 783

nbr of attribs = 5

Request\_Id: 35 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

Tx\_Reference: 30 30 32 65 2d 30 30 30 30 30

30 30 65 2d 30 32 37 66

Method: 00 02

SipMsg: SIP/2.0 200 OK

Via: SIP/2.0/UDP 21.1.1.7:55809;received=21.1.1.7;branch=z9hG4bK\*002e-0000000e-027[SAPI 05/01/70 20:38:14.151540 ]f

From: <sip:+13288613102@qd.lucentlab.com;user=phone>;tag=b72e12N262c5-3f28

To: <sip:+13288613102@qd.lucentlab.com;user=phone>;tag=55be97dc-55c02a28176adaa8

Call-ID: 02-REG-FSM-0000088adjlkqqojlj@21.1.1.7

CSeq: 2 REGISTER

Contact: sip:+13288613102@21.1.1.7:55809;expires=3600

Date: Tue, 04 Aug[SAPI 05/01/70 20:38:14.151607 ] 2015 02:57:44 GMT

Server: Alcatel-Lucent-HPSS/3.0.3

Path: <sip:pcgw-stdn.imsgroup0-000.ism01.qd.lucentlab.com:5060;lr;ottag=ue\_term;bidx=4123;access-type=ADSL>

Allow-Events: reg

P-Associated-URI: <sip:+13288613102@qd.lucentlab.com>

Content-Length: 0

HeaderOffsets: 00 00 00 15 00 62 00 03 [SAPI 05/01/70 20:38:14.151678 ]00 6b

00 ae 00 04 00 b5 01 00 00 05

01 0c 01 31 00 06 01 3a 01 43

00 09 01 4f 01 7a 00 16 01 83

01 9f 00 22 01 aa 01 c2 00 25

01 cb 02 2f 00 37 02 40 02 42

00 47 02 57 02 79 00 54 02 8c

02 8c

//=========================enter SipMgnt ========================================

void SipMgnt\_SipCbItf::OnRegisterResponse(request\_id, reply)

|-🡪SipSubscriberMgr::getInstance()->HandleRegisterReply(request\_id, reply);

| |-🡪SipMgnt\_SipCbItf::OnPassiveHeartBeat(indexId);

| |-🡪 SipTermination::SipTermPointers[indexId]->m\_RegLine->ProcessRegisterReply(request\_id, reply);

| | |-🡪pServerEntry->alarmReport(alarm\_dnsLookUpFailure, alarm\_down);

| | |-🡪 U32 expires\_time = GetExpiresTime(reply);

| | |-🡪FsmProcessEvent(&successRespEvent);

| | | |-🡪 pCurState->StateProcessEvent((void\*)this, pEvent);

| | | | |-🡪 case SipRegEvent::SipReg\_regfsmSuccessRsp: ProcessSuccessResp (pSipRegLine, pSipRegEvent);

| | | | | |-🡪 pSipRegLine->DoSendMsgToTerm(REGHANDLER\_REGISTER,SipMgnt\_success,pEvent->status\_code,pEvent->expires,0);

| | | | | |-🡪pSipRegLine->ChgStateTo(SipRegRegistedState::Instance());

//=========================enter SipMgnt\_termination ========================================

SipTermination::ProcessRegisterResponseFromRegFsm

|-🡪 case REGHANDLER\_REGISTER: ProcessRegisterCfm(pRegCnf);

| |-🡪 m\_RegLine->saveServiceRouteHeader(m\_ServiceRoute);

| |-🡪AttachSipServer (pSipServer);

| | |-🡪 m\_Group = pServer->AddTermination(this);

| |🡪ConfigSubscriberData();

| | |🡪PalConfigItf::configSubscriber(&lineCfgData);

| | | |🡪 \*pal\_glb\_stateHandlerArray[dCurrentState])\

(&pGlbTermProfile[dLineId], &tempCallData, Pal\_en\_configSubsData, & dErr);

| |-🡪EnableSubscriber();

| |-🡪retryTime = CalculateRefreshTime(pRegCnf->expires);

pal\_fn\_configuredTerm

|🡪case Pal\_en\_configSubsData:

| |🡪 pal\_fn\_updateConfigData(pCallData,pErr);

## 2，注销代码流程

接收到adminstatus down的事件之后

//=========================enter SipMgnt ========================================

void SipMgnt\_msgHandler(void\* pMsgHeader)

|-🡪 SipMgnt::MsgHandler((voipMsgHeader\_t\*)pMsgHeader);

| |-🡪 PrintReceiveMsg(pMsg);

| |-🡪 SipSubscriberMgr::MsgHandler(pMsg);

| | |-🡪 case MSG\_MODIFY\_SIPSUBSCRIBER\_REQ: getInstance()->HandleModifySubscriberReq(pMsg);

| | | |-🡪 PrintSubscriberSetReq(pSubscriberSetReq);

| | | |-🡪SipTermination::SipTermPointers[dIndex]->ProcessConfigReq(pSubscriberSetReq);

| | | | |-🡪 ChgAdminStatusDown();

| | | | | |-🡪 OnAdminStatusDown (this);

| | | | | | |-🡪m\_Polling->StartDeRegister((void\*)this);

| | | | | | | |-🡪StartPolling(pSipTerm, pollingForDeRegister,SipMgntPolling::MINIMUM\_INTERVAL,"Termination DeRegister");

| | | |-🡪SipMgnt::FillRespMsgHdr(pMsg, &respMsg.hdr);

| | | | |-🡪case MSG\_MODIFY\_SIPSUBSCRIBER\_REQ:

| | | | |-🡪PrintMsgHeader(pOutRespMsg);

| | | |-🡪SipMgnt::SendQMsg(pSubscriberSetReq->src\_queue\_id, (char\*)&respMsg, sizeof(respMsg));

//=========================enter SipMgnt\_termination ========================================

Polling\_Callback\_Handler

|-🡪pCurPolling->PollingOperation ();

| |-🡪case pollingForDeRegister: PerformDeRegister();

| | |-🡪 StartPolling(pTerm, pollingForDeRegister,500,"Termination DeRegister");

| | | |-🡪StopPolling();

| | |-🡪 DoDeRegister();

| | | |-🡪 pCurServer = pTerm->GetSipServer();

| | | |-🡪pTerm->SendDeRegisterRequest();

| | | | |-🡪m\_RegLine->ProcessRegisterRequest(REGHANDLER\_DEREGISTER,m\_AssignedServer, bypassAvalancheControl);

| | | | | |-🡪FsmProcessEvent(&deRegReqEvent);

| | | | | | |-🡪pCurState->StateProcessEvent((void\*)this, pEvent);

| | | | | | | |-🡪 case SipRegEvent::SipReg\_regfsmDeregisterReq: ProcessDeRegisterReq (pSipRegLine, pSipRegEvent);

| | | | | | | | |-🡪pSipRegLine->DoReqisterRequest(pSipRegLine->m\_bypassAvalancheControl,REGHANDLER\_DEREGISTER);

| | | | | | | | | |-🡪PopulateAutherizationData(register\_request.p\_auth);

| | | | | | | | | |-🡪addPANIHeadertoRegOption(register\_option, local\_ip);

| | | | | | | | | |-🡪addAcceptHeadertoRegOption(register\_option);

| | | | | | | | | |-🡪SSAPIW::SipEngine::getInstance().doRegisterRequest ( m\_orig\_req\_id, register\_request,register\_option);

| | | | | | | | | | |-🡪sendOk = AntiAvalSendRegistrationMsg((void\*)ss\_tx\_req);

| | | | | | | | | | | |-🡪return pRegController->SendSipRequest(pSipMsg);

| | | | | | | | |-🡪pSipRegLine->ChgStateTo(SipRegDeregisteringState::Instance());

//=========================enter SSAPIW ========================================

| | |--🡪 ssapi\_parse\_uri(request.req\_uri, pEncInv);

| | |--🡪 SipSSApiInviteRequest\* ss\_inv\_req = pEncInv->EncodeInv(inv, request\_id, request, option);

| | |---🡪 ssapi\_send\_msg(ss\_inv\_req);

| | | |--🡪 sendToSipstack(SipSSApiMsg\* applmsg) // [SAPI]:send to sipstack

| | | | |----🡪 m\_connection.printMsg(applmsg);

| | | | |----🡪 m\_connection.writeMsg(buffer, YAPI\_FALSE);

Sip注册请求的内容：

[SAPI 05/01/70 20:36:01.853784 ]Tx\_Request (113) ; len = 1270

nbr of attribs = 5

Request\_Id: 33 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

IpPortAddress: 32 31 2e 31 2e 31 2e 37 3a 35

35 38 30 39

SipMsg: REGISTER sip:pcgw-stdn.imsgroup0-001.qdc07.qd.lucentlab.com SIP/2.0

Route: <sip:135.252.37.124;lr>

From: <sip:+132886131[SAPI 05/01/70 20:36:01.853866 ]02@qd.lucentlab.com;user=phone>;tag=b72e12N26241-41f

To: <sip:+13288613102@qd.lucentlab.com;user=phone>

Call-ID: 02-REG-FSM-0000314fdjlkqqojlj@21.1.1.7

CSeq: 3 REGISTER

Max-Forwards: 30

Contact: \*

Expires: 0

Accept: application/sdp,multipart/mixed,application/vnd.etsi.aoc+xml;schemaversion=2,ab.com",nonce="b7c9036dbf3054aea940e9703dc8f55c0278584c2908",uri="sip:pcgw-s[SAPI 05/01/70 20:36:01.854005 ]tdn.imsgroup0-001.qdc07.qd.lucentlab.com",cnonce="241-7258-3b1716c",nc=00000002,qop=auth,response="4afeeb489c77dcf4ce50606e04467d54",algorithm=MD5,opaque="ALU:QbkRBthOEgEQAkgVEwwHRAIBHgkdHwQCQ1lFRkdWEAkscnJqNCJpJDwpLiI5Ii4yfzE8OW9jZ25p"

Supported: path

User-Agent: Alcatel-Lucent MSAN

P-Access-Net[SAPI 05/01/70 20:36:01.854070 ]work-Info: xDSL;dsl-location=";21.1.1.7;13288613102;;;"

Content-Length: 0

HeaderOffsets: 00 02 00 4c 00 62 00 03 00 6b

00 ad 00 04 00 b4 00 e1 00 05

00 ed 01 12 00 06 01 1b 01 24

00 07 01 35 01 36 00 09 01 42

01 42 00 0a 01 4e 01 4e 00 0e

01 59 01 67 00 0e 01 69 01 77

00 0e 01 79 0[SAPI 05/01/70 20:36:01.854131 ]1 a4 00 0e 01 a6

01 c6 00 0e 01 c8 01 df 00 0e

01 e1 01 fc 00 12 02 06 02 45

00 14 02 57 03 bd 00 24 03 cb

03 ce 00 29 03 dd 03 ef 00 4a

04 09 04 34 00 54 04 47 04 47

DNSPool\_Id: 00 01

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Enter SSAPI recv from sipstack \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

onRecvMsg()

|

|--🡪 printMsg(applmsg);

|--🡪 m\_client->sendToApplication(applmsg);

Tx临时回复的消息内容：

[SAPI 05/01/70 20:36:01.884245 ]Tx\_Reply (114) ; len = 46

nbr of attribs = 3

Request\_Id: 33 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

Tx\_Reference: 30 30 32 65 2d 30 30 30 30 30

30 30 63 2d 30 32 37 64

Method: 00 02

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Enter SSAPI recv from sipstack \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

onRecvMsg()

|

|--🡪 printMsg(applmsg);

|--🡪 m\_client->sendToApplication(applmsg);

注销请求的回复内容：

[SAPI 05/01/70 20:36:01.903425 ]Tx\_Reply (114) ; len = 753

nbr of attribs = 5

Request\_Id: 33 30 32 2d 72 65 67 66 73 6d

2d 70 61 6c

Tx\_Reference: 30 30 32 65 2d 30 30 30 30 30

30 30 63 2d 30 32 37 64

Method: 00 02

SipMsg: SIP/2.0 200 OK

Via: SIP/2.0/UDP 21.1.1.7:55809;received=21.1.1.7;branch=z9hG4bK\*002e-0000000c-027[SAPI 05/01/70 20:36:01.903513 ]d

From: <sip:+13288613102@qd.lucentlab.com;user=phone>;tag=b72e12N26241-41f

To: <sip:+13288613102@qd.lucentlab.com;user=phone>;tag=55be97dc-55c029a48ceac8a

Call-ID: 02-REG-FSM-0000314fdjlkqqojlj@21.1.1.7

CSeq: 3 REGISTER

Contact: sip:+13288613102@21.1.1.7:55809;expires=0

Date: Tue, 04 Aug 2015[SAPI 05/01/70 20:36:01.903584 ] 02:55:32 GMT

Server: Alcatel-Lucent-HPSS/3.0.3

Path: <sip:pcgw-stdn.imsgroup0-000.ism01.qd.lucentlab.com:5060;lr;ottag=ue\_term;bidx=3924;access-type=ADSL>

P-Associated-URI: <sip:+13288613102@qd.lucentlab.com>

Content-Length: 0

HeaderOffsets: 00 00 00 15 00 62 00 03 00 6b

00 ad 00 04 00 b[SAPI 05/01/70 20:36:01.903644 ]4 00 fe 00 05

01 0a 01 2f 00 06 01 38 01 41

00 09 01 4d 01 75 00 16 01 7e

01 9a 00 22 01 a5 01 bd 00 25

01 c6 02 2a 00 47 02 3f 02 61

00 54 02 74 02 74

//=========================enter SipMgnt ========================================

void SipMgnt\_SipCbItf::OnRegisterResponse(request\_id, reply)

|-🡪SipSubscriberMgr::getInstance()->HandleRegisterReply(request\_id, reply);

| |-🡪SipMgnt\_SipCbItf::OnPassiveHeartBeat(indexId);

| |-🡪 SipTermination::SipTermPointers[indexId]->m\_RegLine->ProcessRegisterReply(request\_id, reply);

| | |-🡪pServerEntry->alarmReport(alarm\_dnsLookUpFailure, alarm\_down);

| | |-🡪 U32 expires\_time = GetExpiresTime(reply);

| | |-🡪FsmProcessEvent(&successRespEvent);

| | | |-🡪 pCurState->StateProcessEvent((void\*)this, pEvent);

| | | | |-🡪 case SipRegEvent::SipReg\_regfsmSuccessRsp: ProcessSuccessResp (pSipRegLine, pSipRegEvent);

| | | | | |-🡪 pSipRegLine->DoSendMsgToTerm(REGHANDLER\_REGISTER,SipMgnt\_success,pEvent->status\_code,pEvent->expires,0);

| | | | | |-🡪pSipRegLine->ChgStateTo(SipRegRegistedState::Instance());

//=========================enter SipMgnt\_termination ========================================

SipTermination::ProcessRegisterResponseFromRegFsm

|-🡪 REGHANDLER\_DEREGISTER: ProcessDeRegisterCfm(pRegCnf);

| |-🡪 m\_RegLine->saveServiceRouteHeader(m\_ServiceRoute);

| |-🡪AttachSipServer (pSipServer);

| | |-🡪 DeattachSipServer();

| | | |-🡪 m\_SipServer->ResetFghmInterval();//vincent for ALU01924462

| | | |-🡪m\_SipServer->RemoveTermination(this);