Tarefas do Session Manager ASR5x00 - Descrição da função, travamento, operações de recuperação e registros de travamento

Contents

Introduction

Este documento descreve e explica a confiabilidade do software, a disponibilidade do serviço e os recursos de failover do Cisco Aggregation Services Router (ASR) 5x00 Series. Ele apresenta a definição de um travamento de software no ASR5x00 e os efeitos do travamento do software. O artigo estabelece que mesmo em caso de travamentos inesperados de software, como o ASR5x00 é capaz de atingir o objetivo de disponibilidade de "classe de operadora" devido aos recursos inerentes de resiliência e disponibilidade de software. O assinante móvel nunca deve ter que pensar na disponibilidade do serviço. O objetivo da Cisco não é nenhuma perda de sessão devido a nenhuma falha de hardware ou software, o que inclui a perda de um sistema completo, em outras palavras, confiabilidade no nível de voz. Os recursos de confiabilidade do software no ASR5x00 têm como objetivo alcançar as metas de disponibilidade de serviço de "classe de operadora", mesmo nos casos em que falhas imprevistas possam ocorrer na rede de um operador.

Arquitetura de software: Projetado para resiliência

O ASR5x00 tem uma coleção de tarefas de software distribuídas através do Packet Services Card (PSC) ou do Data Processing Card (DPC) e do System Management Card (SMC) ou das placas de gerenciamento e E/S (MIO), projetadas para executar uma variedade de funções específicas.

Por exemplo, a tarefa do gerenciador de sessões é responsável por tratar as sessões de um conjunto de assinantes e executar serviços em linha, como peer-to-peer (P2P), Deep Packet Inspection (DPI), etc., no tráfego do usuário. A tarefa do gerenciador de Autenticação, Autorização e Contabilidade (AAA) é responsável pela geração de eventos de cobrança para

registrar o uso do tráfego do assinante e assim por diante. As tarefas do gerenciador de sessão e do gerenciador AAA são executadas no cartão PSC/DPC.

A placa SMC/MIO é reservada para operações e manutenção (O&M) e tarefas relacionadas à plataforma. O sistema ASR5x00 é virtualmente compartimentado em subsistemas de software diferentes, como o subsistema de sessão para o processamento de sessões de assinantes e o subsistema VPN responsável pela atribuição de endereços IP, roteamento e assim por diante. Cada subsistema tem uma tarefa de controlador que supervisiona a integridade do subsistema que ele controla. As tarefas do controlador são executadas na placa SMC/MIO. As tarefas do gerenciador de sessão e do AAA Manager são emparelhadas para tratar da sessão de um assinante para fins de controle, tráfego de dados e faturamento. Quando a recuperação de sessão está habilitada no sistema, cada tarefa do gerenciador de sessão faz o backup do estado de seu conjunto de estados de assinante com uma tarefa do gerenciador AAA par a ser recuperada no caso de um travamento do gerenciador de sessão.

O que é um acidente?

Uma tarefa no ASR5x00 pode potencialmente travar se encontrar uma condição de falha durante a operação normal. Uma falha de travamento ou software no ASR5x00 é definida como uma saída ou encerramento *inesperado* de uma tarefa no sistema. Um travamento pode ocorrer se o código do software tentar acessar áreas de memória proibidas (como estruturas de dados corrompidas), encontrar uma condição no código que não é esperada (como uma transição de estado inválido) e assim por diante. Um travamento também pode ser acionado se a tarefa não responder à tarefa do monitor do sistema e o monitor tentar matar e reiniciar a tarefa. Um evento de travamento também pode ser disparado explicitamente (ao contrário do inesperado) no sistema quando uma tarefa é forçada a despejar seu estado atual por um comando CLI ou pelo monitor do sistema para analisar o estado da tarefa. Um evento de travamento esperado também pode ocorrer quando as tarefas do controlador do sistema são reiniciadas para corrigir uma situação com uma tarefa do gerente que falha repetidamente.

Efeitos de um travamento do Session Manager

Em operação normal, uma tarefa do gerenciador de sessão lida com um conjunto de sessões de assinantes e tráfego de dados associado para as sessões, juntamente com uma tarefa do gerenciador AAA de peering que lida com a cobrança para essas sessões de assinante. Quando ocorre um travamento do gerenciador de sessão, ele deixa de existir no sistema. Se a recuperação de sessão estiver habilitada no sistema, uma tarefa do gerenciador de sessão de standby será executada para se tornar ativa na mesma placa PSC/DPC. Esta nova tarefa do gerenciador de sessão restabelece as sessões do assinante enquanto se comunica com a tarefa do gerenciador AAA do peer. A operação de recuperação varia de 50 ms a alguns segundos dependendo do número de sessões que estavam ativas no gerenciador de sessões no momento do travamento e da carga geral da CPU na placa e assim por diante. Não há perda nas sessões de assinantes que já foram estabelecidas no gerenciador de sessão original nesta operação. Qualquer sessão de assinante que estivesse em processo de estabelecimento no momento do travamento também provavelmente será restaurada devido a retransmissões de protocolo e assim por diante. Qualquer pacote de dados que estivesse em transição pelo sistema no momento do travamento pode ser considerado associado a uma perda de rede pelas entidades comunicantes da conexão de rede e será retransmitido e a conexão será executada pelo novo gerenciador de sessão. As informações de cobrança das sessões transportadas pelo gerente de

sessão serão preservadas no gerente AAA do peer.

Quando o operador deve se preocupar?

Quando um travamento do gerenciador de sessão ocorre, o procedimento de recuperação acontece conforme descrito anteriormente e o resto do sistema permanece sem ser afetado por esse evento. Um travamento em um gerenciador de sessão não afeta os outros gerentes de sessão. Como orientação para o operador, se várias tarefas do gerenciador de sessão *na mesma placa PSC/DPC* travarem simultaneamente ou dentro de 10 minutos uma da outra, poderá haver perda de sessões, pois o sistema talvez não consiga iniciar novos gerentes de sessão com rapidez suficiente para substituir as tarefas travadas. Isso corresponde a um cenário de falha dupla em que pode ocorrer perda de sessões. Quando a recuperação não é viável, o gerenciador de sessão é simplesmente reiniciado e está pronto para aceitar novas sessões.

Quando um determinado gerenciador de sessão trava repetidamente (como se ele encontrasse a mesma condição de falha repetidamente), a tarefa do controlador de sessão é anotada e reiniciada na tentativa de restaurar o subsistema. Se a tarefa do controlador de sessão não puder estabilizar o subsistema de sessão e se reiniciar continuamente nesse esforço, a próxima etapa no escalonamento é que o sistema mude para uma placa SMC/MIO em standby. No caso improvável de não haver uma placa SMC/MIO em standby ou de ocorrer uma falha na operação de switchover, o sistema é reinicializado.

Os gerentes de sessão também mantêm estatísticas para cada Nome do Ponto de Acesso (APN - Access Point Name), Serviços, funcionalidades e assim por diante que serão perdidos permanentemente quando ocorrer um travamento. Portanto, uma entidade externa que coleta estatísticas de bulkstats periodicamente observará uma queda nas estatísticas quando um ou mais travamentos ocorrerem. Isso pode se manifestar como um mergulho em uma representação gráfica das estatísticas desenhadas em um eixo do tempo.

> **Note**: Um chassi típico preenchido com 7-14 placas PSC ou 4-10 DPC possui cerca de 120-160 gerentes de sessão, dependendo do número de placas PSC/DPC, e um único travamento resultará na perda de cerca de 1/40^{o} ou 1/80 ^{o} das estatísticas. Quando um gerenciador de sessão em standby assume, ele começa a acumular as estatísticas novamente de zero.

Como saber se ocorreu um acidente?

Um travamento acionará um evento de armadilha SNMP para uma estação de monitoramento de rede, como o Serviço de Monitoramento de Eventos (EMS - Event Monitoring Service) e por eventos de syslog. Os travamentos ocorridos no sistema também podem ser observados com o comando **show crash list**. Observe que esse comando lista eventos de travamento inesperados e esperados, conforme descrito anteriormente. Esses dois tipos de eventos de travamento podem ser diferenciados por meio de um cabeçalho que descreve cada travamento.

Um travamento de tarefa seguido de uma recuperação de sessão bem-sucedida é indicado por esta mensagem de log:

```
"Death notification of task <name>/<instance id> on <card#>/<cpu#> sent to parent
task <parent name>/<instance id> with failover of <task name>/<instance id>
on <card#>/<cpu#>"
```

Uma falha de tarefa que não pôde ser recuperada é indicada por esta mensagem de log:

```
"Death notification of task <name>/<instance id> on <card#>/<cpu#> sent to parent
task <parent name>/<instance id>"
```

Em resumo, com a recuperação de sessão habilitada, na maioria dos casos, os travamentos não serão notados porque não têm impacto no assinante. É necessário inserir o comando CLI ou examinar os registros ou a notificação SNMP para detectar qualquer ocorrência de travamentos.

Por exemplo:

```
 ******** show crash list *******
Tuesday May 26 05:54:14 BDT 2015
=== ==================== ======== ========== =============== ======================
# Time Process Card/CPU/ SW HW_SER_NUM
PID VERSION MIO / Crash Card
=== ==================== ======== ========== =============== ======================

1 2015-May-07+11:49:25 sessmgr 04/0/09564 17.2.1 SAD171600WS/SAD172200MH
2 2015-May-13+17:40:16 sessmgr 09/1/05832 17.2.1 SAD171600WS/SAD173300G1
3 2015-May-23+09:06:48 sessmgr 03/1/31883 17.2.1 SAD171600WS/SAD1709009P
4 2015-May-25+15:58:59 sessmgr 09/1/16963 17.2.1 SAD171600WS/SAD173300G1
5 2015-May-26+01:15:15 sessmgr 04/0/09296 17.2.1 SAD171600WS/SAD172200MH

 ******** show snmp trap history verbose *******
Fri May 22 19:43:10 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:30 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 9 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1754 calls recovered 1754 all call lines 1754 time elapsed ms 1108.
Fri May 22 19:43:32 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:44:49 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:51 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 7 Cpu Number 0 fetched from aaa mgr 1741 prior to audit 1741 passed audit
1737 calls recovered 1737 all call lines 1737 time elapsed ms 1047.
Fri May 22 19:44:53 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:50:04 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 221 card 2 cpu 1
: Fri May 22 19:50:04 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:04 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:05 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 2 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
```

```
audit 1749 calls recovered 1750 all call lines 1750 time elapsed ms 1036.

 ******** show snmp trap history verbose *******
Fri May 22 19:43:10 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:29 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 204 on card 9 cpu 1
Fri May 22 19:43:30 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 9 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1754 calls recovered 1754 all call lines 1754 time elapsed ms 1108.
Fri May 22 19:43:32 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 204 card 9 cpu 1
Fri May 22 19:44:49 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:49 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 236 on card 7 cpu 0
Fri May 22 19:44:51 2015 Internal trap notification 183 (SessMgrRecoveryComplete)
Slot Number 7 Cpu Number 0 fetched from aaa mgr 1741 prior to audit 1741 passed
audit 1737 calls recovered 1737 all call lines 1737 time elapsed ms 1047.
Fri May 22 19:44:53 2015 Internal trap notification 1099 (ManagerRestart) facility
sessmgr instance 236 card 7 cpu 0
Fri May 22 19:50:04 2015 Internal trap notification 73 (ManagerFailure) facility
sessmgr instance 221 card 2 cpu 1
: Fri May 22 19:50:04 2015 Internal trap notification 150 (TaskFailed) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:04 2015 Internal trap notification 151 (TaskRestart) facility
sessmgr instance 221 on card 2 cpu 1
Fri May 22 19:50:05 2015 Internal trap notification 183 (SessMgrRecoveryComplete
) Slot Number 2 Cpu Number 1 fetched from aaa mgr 1755 prior to audit 1755 passed
audit 1749 calls recovered 1750 all call lines 1750 time elapsed ms 1036.

 ******** show logs  *******
2015-May-25+23:15:53.123 [sitmain 4022 info] [3/1/4850 <sitmain:31> sittask.c:4762]
[software internal system critical-info syslog] Readdress requested for facility
sessmgr instance 5635 to instance 114
2015-May-25+23:15:53.122 [sitmain 4027 critical] [3/1/4850 <sitmain:31>
crash_mini.c:908] [software internal system callhome-crash] Process Crash Info:
time 2015-May-25+17:15:52(hex time 556358c8) card 03 cpu 01 pid 27118 procname
sessmgr crash_details
Assertion failure at acs/acsmgr/analyzer/ip/acs_ip_reasm.c:2970
Function: acsmgr_deallocate_ipv4_frag_chain_entry()
Expression: status == SN_STATUS_SUCCESS
Proclet: sessmgr (f=87000,i=114)
Process: card=3 cpu=1 arch=X pid=27118 cpu=~17% argv0=sessmgr
Crash time: 2015-May-25+17:15:52 UTC
Recent errno: 11 Resource temporarily unavailable
Stack (11032@0xffffb000):
[ffffe430/X] __kernel_vsyscall() sp=0xffffbd28
[0af1de1f/X] sn_assert() sp=0xffffbd68
[0891e137/X] acsmgr_deallocate_ipv4_frag_chain_entry() sp=0xffffbde8
[08952314/X] acsmgr_ip_frag_chain_destroy() sp=0xffffbee8
[089d87d1/X] acsmgr_process_tcp_packet() sp=0xffffc568
[089da270/X] acs_process_tcp_packet_normal_path() sp=0xffffc5b8
[089da3fd/X] acs_tcp_analyzer() sp=0xffffc638
[0892fb39/X] do_acsmgr_process_packet() sp=0xffffc668
[08940045/X] acs_ip_lean_path() sp=0xffffc6b8
[0887e309/X] acsmgr_data_receive_merge_mode() sp=0xffffc9d8
```

```
[0887f323/X] acs_handle_datapath_events_from_sm_interface() sp=0xffffca08
[037c2e1b/X] sessmgr_sef_initiate_data_packet_ind() sp=0xffffca88
[037c2f50/X] sessmgr_pcc_intf_send_data_packet_ind() sp=0xffffcaf8
[061de74a/X] sessmgr_pcc_fwd_packet() sp=0xffffcb58
[0627c6a4/X] sessmgr_ipv4_process_inet_pkt_part2_slow() sp=0xffffcf68
[06318343/X] sessmgr_ipv4_process_inet_pkt_pgw_ggsn() sp=0xffffd378
[0632196c/X] sessmgr_med_ipv4_data_received() sp=0xffffd418
[0633da9a/X] sessmgr_med_data_receive() sp=0xffffd598
[0afb977c/X] sn_epoll_run_events() sp=0xffffd5e8
[0afbdeb8/X] sn_loop_run() sp=0xffffda98
[0ad2b82d/X] main() sp=0xffffdb08

2015-May-25+23:15:53.067 [rct 13038 info] [5/0/7174 <rct:0> rct_task.c:305]
[software internal system critical-info syslog] Death notification of task
sessmgr/114 on 3/1 sent to parent task sessctrl/0 with failover of sessmgr/5635 on 3/1
2015-May-25+23:15:53.065 [evlog 2136 info] [5/0/7170 <evlogd:0> odule_persist.c:3102]
[software internal system critical-info syslog] Evlogd crashlog: Request received to
check the state of persistent crashlog.
2015-May-25+23:15:53.064 [sitmain 4099 info] [3/1/4850 <sitmain:31> crash_mini.c:765]
[software internal system critical-info syslog] have mini core, get evlogd status for
logging crash file 'crashdump-27118'
2015-May-25+23:15:53.064 [sitmain 4017 critical] [3/1/4850 <sitmain:31> sitproc.c:1544]
[software internal system syslog] Process sessmgr pid 27118 died on card 3 cpu 1
signal=6 wstatus=0x86
2015-May-25+23:15:53.048 [sitmain 4074 trace] [5/0/7168 <sitparent:50> crashd.c:1130]
[software internal system critical-info syslog] Crash handler file transfer starting
(type=2 size=0 child_ct=1 core_ct=1 pid=23021)
2015-May-25+23:15:53.047 [system 1001 error] [6/0/9727 <evlogd:1> evlgd_syslogd.c:221]
[software internal system syslog] CPU[3/1]: xmitcore[21648]: Core file transmitted to
card 5 size=663207936 elapsed=0sec:908ms
2015-May-25+23:15:53.047 [system 1001 error] [5/0/7170 <evlogd:0> evlgd_syslogd.c:221]
[software internal system syslog] CPU[3/1]: xmitcore[21648]: Core file transmitted to
card 5 size=663207936 elapsed=0sec:908ms
2015-May-25+23:15:53.047 [sitmain 4080 info] [5/0/7168 <sitparent:50> crashd.c:1091]
[software internal system critical-info syslog] Core file transfer to SPC complete,
received 8363207936/0 bytes

 ******** show session recovery status verbose *******
Tuesday May 26 05:55:26 BDT 2015
Session Recovery Status:
Overall Status : Ready For Recovery
Last Status Update : 8 seconds ago

----sessmgr--- ----aaamgr---- demux
cpu state active standby active standby active status
---- ------- ------ ------- ------ ------- ------ -------------------------
1/0 Active 24 1 24 1 0 Good
1/1 Active 24 1 24 1 0 Good
2/0 Active 24 1 24 1 0 Good
2/1 Active 24 1 24 1 0 Good
3/0 Active 24 1 24 1 0 Good
3/1 Active 24 1 24 1 0 Good
4/0 Active 24 1 24 1 0 Good
4/1 Active 24 1 24 1 0 Good
5/0 Active 0 0 0 0 14 Good (Demux)
7/0 Active 24 1 24 1 0 Good
7/1 Active 24 1 24 1 0 Good
8/0 Active 24 1 24 1 0 Good
8/1 Active 24 1 24 1 0 Good
9/0 Active 24 1 24 1 0 Good
9/1 Active 24 1 24 1 0 Good
10/0 Standby 0 24 0 24 0 Good
10/1 Standby 0 24 0 24 0 Good
```

Arquitetura de registro de falhas

Os registros de travamento registram todas as informações possíveis relacionadas a um travamento de software (despejo de núcleo completo). Devido ao seu tamanho, eles não podem ser armazenados na memória do sistema. Portanto, esses registros só serão gerados se o sistema estiver configurado com um URL que aponte para um dispositivo local ou um servidor de rede onde o registro possa ser armazenado.

O registro de travamento é um repositório persistente de informações de eventos de travamento. Cada evento é numerado e contém texto associado a uma CPU (minicore), unidade de processamento de rede (NPU) ou falha de kernel. Os eventos registrados são gravados em registros de comprimento fixo e armazenados em /flash/crashlog2.

Sempre que ocorrer um travamento, essas informações de travamento serão armazenadas:

1. O registro de evento é armazenado no arquivo /flash/crashlog2 (o registro de travamento).
2. O arquivo de minicore, NPU ou despejo de kernel associado é armazenado no diretório /flash/crsh2.
3. Um dump central completo é armazenado em um diretório configurado pelo usuário.

Sincronização de eventos de travamento e minicores entre placas de gerenciamento

O registro de travamento é exclusivo para cada um dos cartões de gerenciamento, portanto, se ocorrer um travamento quando a placa "8" estiver ativa, ela será conectada à placa "8". Um switchover subsequente não exibiria mais o travamento no registro. Para recuperar esse travamento, é necessário fazer um switch de volta para a placa "8". O registro de eventos de travamento e os despejos são exclusivos dos cartões de gerenciamento ativo e standby, portanto, se ocorrer um travamento em uma placa ativa, o registro de eventos de travamento e os despejos relacionados serão armazenados em uma placa ativa apenas. Essas informações de travamento não estão disponíveis na placa de espera. Sempre que as placas mudam devido a um travamento na placa ativa e as informações de travamento não são mais exibidas na placa que assume, as informações de travamento podem ser recuperadas somente da placa ativa atual. Para recuperar a lista de travamentos da outra placa, é necessário um switchover novamente. Para evitar esse switchover e obter as informações de travamento da placa de espera, é necessária a sincronização entre duas placas de gerenciamento e a manutenção das informações de travamento mais recentes.

O evento de travamento de chegada será enviado para o SMC/MIO em standby e salvo no arquivo de travamento do standby da mesma maneira. Despejos de minicore, NPU ou kernel na flash do SMC/MIO ativo precisam ser sincronizados para o SMC/MMIO em standby com o comando **rsync**. Quando uma entrada de registro de travamento ou toda a lista é excluída por meio do comando CLI, ela deve ser apagada nos SMCs/MIOs ativos e em standby. Não há impacto na memória. Toda a atividade de sincronização relacionada ao travamento será feita pelo código da placa SMC/MIO em standby, pois o registro em standby é menos carregado e a placa em standby tem espaço suficiente para a atividade de sincronização. Portanto, o desempenho do sistema não será afetado.

Comandos

Esses comandos podem ser usados para solucionar problemas:

```
#show support details

#show crash list

#show logs

#show snmp trap history verbose

#show session recovery status verbose

#show task resources facility sessmgr instance <>

#show task resources facility sessmgr all
```

Os arquivos de arquivos são gerados após um travamento. Geralmente, os operadores os armazenam em um servidor externo. O nome do arquivo padrão geralmente se parece com crash-<Cardnum>-<CPU Num>-<Hex timestamp>-coree.gcrash-09-00-5593a1b8-core.

Sempre que ocorrer um travamento, essas informações de travamento serão armazenadas:

- O registro de evento é armazenado no arquivo /flash/crashlog2 (o registro de travamento).
- O arquivo de minicore, NPU ou despejo de kernel associado é armazenado no diretório /flash/crsh2.

Summary

Todo o software ASR5x00 foi projetado para lidar com condições/eventos previstos e condições/eventos imprevistos. Enquanto a Cisco se esforça para ter um software perfeito, inevitavelmente, existirão erros e ocorrerão travamentos. É por isso que o recurso de recuperação de sessão é tão importante. O esforço da Cisco pela perfeição minimizará as ocorrências de travamentos, e a recuperação da sessão permitirá que as sessões continuem após um travamento. No entanto, é importante que a Cisco continue a se esforçar para obter um software perfeito. Menos travamentos reduzirão a probabilidade de vários travamentos que acontecem simultaneamente. Embora a recuperação de sessão corrija um único travamento sem problemas, a recuperação de vários travamentos simultâneos é projetada de forma um pouco diferente. Os operadores raramente (ou nunca) devem sofrer vários travamentos simultâneos, mas se isso ocorrer, o ASR5x00 é projetado para recuperar a integridade do sistema como a prioridade mais alta, possivelmente no sacrifício de algumas sessões de assinantes.