

StarOS Tasks

This appendix describes system and subsystem tasks running under StarOS on an ASR 5500 and virtualized platforms.

Important This appendix is not a comprehensive list of all StarOS tasks. It simply provides general descriptions of the primary tasks and subsystems within StarOS.

It includes the following sections:

- Overview, on page 1
- Primary Task Subsystems, on page 1
- Controllers and Managers, on page 3
- Subsystem Tasks, on page 3

Overview

For redundancy, scalability and robust call processing, StarOS supports a series of tasks that perform specific functions. These tasks communicate with each other as needed to share control and data signals. As a result, processes can be distributed across multiple tasks thus reducing the overall work-load on any given task and improving system performance. This distributed design provides fault containment that greatly minimizes the impact to processes or sessions due to a failure.

The Exec mode **show task** command displays snapshots of running processes within StarOS. For detailed information about this command, see the *Command Line Interface Reference* and *Statistics and Counters Reference*.

The following sections describe the primary tasks that are implemented by StarOS:

- Primary Task Subsystems, on page 1
- Controllers and Managers, on page 3

Primary Task Subsystems

The individual tasks that run on the CPUs are divided into subsystems. Following is a list of the primary subsystems responsible for call session processing:

- **System Initiation Task (SIT):** This subsystem starts tasks and initializes the system. This includes starting a set of initial tasks at system startup time (static tasks), and starting individual tasks on demand at arbitrary times (dynamic tasks).
- **High Availability Task (HAT):** With the Recovery Control Task (RCT) subsystem, the HAT subsystem maintains the operational state of the system. HAT monitors the various software and hardware components of the system. If there are unusual activities, such as the unexpected termination of another task, the HAT subsystem takes a suitable course of action, such as triggering an event to the RCT subsystem to take corrective action or to report the status. The primary function of the HAT task is to minimize service impacts.
- **Recovery Control Task (RCT):** This subsystem executes a recovery action for any failure that occurs in the system. The RCT subsystem receives signals from the HAT subsystem (and in some cases from the NPU subsystem) and determines what recovery actions are needed.
- **Shared Configuration Task (SCT):** This subsystem provides a facility to set, retrieve, and receive notification of system configuration parameters. The SCT is mainly responsible for storing configuration data for the applications that run on the system.

  The SCT subsystem runs only on the active management card and synchronizes the information it contains with the SCT subsystem on the standby management card.
- **Resource Management (RM):** This subsystem assigns resources, such as CPU loading and memory, for every system task upon start-up. The RM subsystem monitors resource use to verify that allocations are as specified. RM also monitors all sessions and communicates with the Session Controller to enforce capacity licensing limits.
- **Virtual Private Network (VPN):** This subsystem manages the administrative and operational aspects of all VPN-related entities in the system. The functions performed by the VPN subsystem include:
  - Creating separate VPN contexts
  - Starting the IP services within a VPN context
  - Managing IP pools and subscriber IP addresses, and distributing the IP flow information within a VPN context.

  All IP operations within StarOS are done within specific VPN contexts. In general, packets are not forwarded across different VPN contexts. The only exception currently is the Session subsystem.
- **Network Processing Unit** (npuctrl/npumgr on ASR 5000; npusim on ASR 5500, and iftask or knpusim on VPC-DI and VPC-SI)[1]: This subsystem is responsible for the following:
  - Using the database to match address and port numbers to destination tasks for fast-path forwarding of dataframes
  - Receiving and transmitting user data frames to/from various physical interfaces
  - IP forwarding decisions (both unicast and multicast)
  - Per-interface packet filtering
  - Traffic management and traffic engineering
  - Passing user data frames to/from packet processing CPUs

[1] knpusim runs instead of iftask on VPC VMs that do not have Intel DPDK supported configurations.

- Modifying, adding, or stripping datalink/network layer headers
- Recalculating checksums
- Maintaining statistics
- Managing external Ethernet interfaces

- **Card/Slot/Port (CSP):** Coordinates the events that occur when any card is inserted, locked, unlocked, removed, shutdown, or migrated. CSP also performs auto-discovery and configures ports on a newly-inserted interface card. It determines how interface cards map to packet processing cards.

  The CSP subsystem runs only on the active management card and synchronizes the information it contains with the SCT subsystem on the standby management card. It is started by the SIT subsystem and monitored by the HAT subsystem.

- **Session Manager (SM):** Performs high-touch processing of mobile subscribers' packet-oriented data session flows. High-touch user data processing consists of the following:
  - Payload transformation
  - Filtering and scheduling
  - Statistics collection
  - Policing

Controllers and Managers

Many of the primary subsystems are composed of controller tasks called Controllers, and subordinated tasks called Managers.

Controllers serve several purposes:

- Monitor the state of their Managers and allow communication between Managers within the same subsystem.
- Enable inter-subsystem communication since they can communicate with the controllers of other subsystems.
- Mask the distributed nature of the software from the user allowing for ease of management.

Managers manage resources and mappings between resources. In addition, some managers are directly responsible for call processing.

For information about the primary subsystems that are composed of critical, controller, and /or manager tasks, see Subsystem Tasks, on page 3.

Subsystem Tasks

The following subsections list and briefly describe StarOS tasks for various subsystems:

- System Initiation Subsystem, on page 4

• High Availability Subsystem, on page 5
• Resource Manager Subsystem, on page 6
• Virtual Private Networking Subsystem, on page 6
• Network Processing Unit Subsystem, on page 8
• Session Subsystem, on page 10
• Platform Processes, on page 19
• Management Processes, on page 22

System Initiation Subsystem

Table 1: System Initiation Subsystem Tasks

Task	Description	Function
SITMAIN	System Initiation Task – Main	Initiated at system start-up.
		Reads and provides startup configuration to other SIT components.
		Starts SITREAP sub-function.
		Maintains CPU state information.
SITPARENT	SIT Parent Sub-function	Starts management cards in either active or standby mode.
		Registers tasks with HAT task.
		Notifies CSP task of CPU startup completion.
		Brings up packet processing cards in standby mode.
SITREAP	SIT Reap Sub-function	Shuts down tasks as required.

High Availability Subsystem

Table 2: High Availability Subsystem Tasks

Task	Description	Function
hatcpu	High Availability Task CPU	Performs device initialization and control functions based on the CPUs hardware capabilities.
		Reports the loss of any task on its CPU to hatsystem sub-function.
		Controls the LEDs on the packet processing cards.
		Initializes and monitors the dedicated hardware on packet processing cards. (ASR 5500 only)
		Collects CPU monitoring information periodically and reports to the master hatcpu sub-function.
		Reports the loss of any task on its CPU to the master hatcpu sub-function.
		Performs device initialization and control functions because of the CPU's hardware capabilities.
		Reports the loss of any task on its CPU to hatsystem sub-function.
		Controls the LEDs on the management card. (ASR 5500 only)
		Initializes and monitors the dedicated hardware on the management card. (ASR 5500 only)
hatsystem	High Availability Task System Controller	Controls all the HAT sub-function tasks in the system. It is initiated on system start-up.
		Initializes system components (such as the Gigabit Ethernet switches and switch fabric).
		Monitors system components such as fans for state changes.
		Triggers actions for redundancy in the event of fault detection.
		The HAT subsystem on the redundant management card mirrors the HAT subsystem on the active management card.

Resource Manager Subsystem

Table 3: Resource Manager (RM) Subsystem Tasks

Task	Description	Function
rmctrl	Resource Manager Controller	Started by the sitparent task on StarOS startup, and monitored by HAT for a failure.
		Initializes resources such as CPUs and memory.
		Requests updated card status from the CSP subsystem and updates the system card table.
		Communicates with all rmctrls to request their most recent set of resource data.
rmmgr	Resource Manager Managers	Started by the sitparent task, and monitored by HATs for failures.
		Initializes the local resource data and local resource scratch space.
		Communicates with the SIT task on the local CPU to get its entire task table and the resources associated with each task.
		Gathers current resource utilization for each task.
		Sends the resource data to the rmctrl task.

Virtual Private Networking Subsystem

Table 4: Virtual Private Networking (VPN) Subsystem Tasks

Task	Description	Function
vpnctrl	VPN Controller	Created at system start-up.
		Initiates the VPN Manager for each context.
		Informs the Session Controller task when there are additions or changes to contexts. Only one Session Controller operates at any time.
		Routes context specific operation information to the appropriate VPN Manager.
		Performs VPN Manager recovery and saves all VPN-related configuration information in SCT.

Task	Description	Function
vpnmgr	VPN Manager	Started by the VPN Controller for each configured context (one is always present for the local context).
		Performs IP address pool and subscriber IP address management.
		Performs all context specific operations including but not limited to: UCM services, IP interfaces, the Address Resolution Protocol (ARP), IP address pool management, slow path forwarding, NPU flows, port Access Control Lists (ACLs), and logging.
		Provides IP interface address information for each context to the Session Controller.
bgp	Border Gateway Protocol	Created by the VPN Manager for each context that has enabled the BGP routing protocol (**router bgp** Context Configuration mode CLI command).
		Responsible for learning and redistributing routing information via the BGP protocol.
		Maintains the BGP peering connections.
		Applies any defined BGP routing policy.
ospf	Open Shortest Path First	Created by VPN Manager for each context that has enabled the OSPF routing protocol (**router ospf** Context Configuration mode CLI command).
		Responsible for learning and redistributing routing information via the OSPF protocol.
		Maintains the OSPF neighboring relationship.
		Maintains the LSA database.
		Performs SPF calculations.
		Applies any defined OSPF routing policy
ospfv3	Open Shortest Path First	Created by VPN Manager for each context that has enabled the OSPFv3 routing protocol (**router ospfv3** Context Configuration mode CLI command)
		Responsible for learning and redistributing routing information via the OSPFv3 protocol.
		Maintains the OSPFv3 neighboring relationship.
		Maintains the LSA database.
		Performs OSPFv3 SPF calculations.
		Applies any defined OSPFv3 routing policy.

Task	Description	Function
rip	Routing Information Protocol	Created by VPN Manager for each context that has enabled the RIP routing protocol (**router rip** Context Configuration mode CLI command)
		Responsible for learning and redistributing routing information via the RIP protocol.
		Maintains the RIP database.
		Sends periodic RIP update messages.
		Applies any defined RIP routing policy.
zebos	L2 and L3 Switching	Created by VPN Manager for each context.
		Maintains the routing table (RIB and FIB) for the context.
		Performs static routing.
		Interfaces to the kernel for routing & interface updates.
		Redistributes routing information to dynamic routing protocols.
		Calculates nexthop reachability.

Network Processing Unit Subsystem

Table 5: Network Processing Unit (NPU) Subsystem Tasks

Task	Description	Function
iftask	Internal Forwarder Task (Intel DPDK) [VPC-DI, VPC-SI]	Created at StarOS start up.
		Provides port configuration services to the CSP task.
		Provides interface binding and forwarding services to the VPN Manager.
		Provides flow insertion and removal services to Session Manager and AAA Manager tasks.
knpusim	Kernel-based NPU Simulator [VPC-DI, VPC-SI]	Created at StarOS start up.
		Provides port configuration services to the CSP task.
		Provides interface binding and forwarding services to the VPN Manager.
		Provides flow insertion and removal services to Session Manager and AAA Manager tasks.
		Provides recovery services to the NPU Controller.

Task	Description	Function
npuctrl	NPU Controller	Created at StarOS start-up. Only one NPU Controller operates in the system at any time.
		Monitors the state of NPU Managers in the system.
		Registers to receive notifications when NPU Manager crashes.
		Controls recovery operation.
		Provides a centralized location for CLI commands related to NPU Manager state.
npumgr	NPU Manager	Created for every packet processing card that is installed and started.
		Provides port configuration services to the CSP task.
		Provides interface binding and forwarding services to the VPN Manager.
		Provides flow insertion and removal services to Session Manager and AAA Manager tasks.
		Provides recovery services to the NPU Controller.
npusim	NPU Simulator [ASR 5500, VPC-SI, SMI]	Created for every DPC installed and started.
		Provides port configuration services to the CSP task
		Provides interface binding and forwarding services to the VPN Manager.
		Provides flow insertion and removal services to Session Manager and AAA Manager tasks.
		Provides recovery services to the NPU Controller.

Session Subsystem

Table 6: Session Subsystem Tasks

Task	Description	Function
sessctrl	Session Controller	Created at StarOS start-up. Only one Session Controller instantiated in the system at any time.
		Acts as the primary point of contact for the Session Subsystem. Since it is aware of the other subsystems running within the system, the Session Controller acts as a proxy for the other components, or tasks, that make up the subsystem.
		Starts, configures, and coordinates the efforts of the Session Processing Subsystem sub-managers.
		Works with Resource Manager to start new Session Managers when all existing Session Managers exceed their capacity.
		Receives context information from VPN Managers.
		Distributes IP interface address information to other Session Processing Subsystem sub-managers.
		Manages Enhanced Charging Service (ECS), Content Filtering and URL Blacklisting services.
sessmgr	Session Manager	Created by the Session Controller.
		Provides a subscriber processing system that supports multiple session types.
		Multiple Session Managers can run on a single CPU and/or can be distributed throughout any CPU present in the system.
		A single Session Manager can service sessions from multiple A11 Managers, and from multiple contexts.
		Processes protocols for A10/A11, GRE, R3, R4, R6, GTPU/GTPC, PPP, and Mobile IP.
		Manages Enhanced Charging Service, Content Filtering and URL Blacklisting services.
		Session Managers are paired with AAA Managers.
		Limitation: Any frequent AAAMGR crashes leads to change of AAAmgr instance id and thus a sessmgr can increment to >512. For example, for calls connecting to sessmgr instance 512 and above, the encoding will be 10 bit (during GUTI based MME attach) and decoding is 9 bit (during SGSN-CONTEXT-REQUEST), leading to context-request landing on incorrect sessmgr and failure of lookup.

Task	Description	Function
a11mgr	A11 Manager	Created by the Session Controller for each context in which a PDSN service is configured.
		Receives the R-P sessions from the PCF and distributes them to different Session Manager tasks for load balancing.
		Maintains a list of current Session Manager tasks to aid in system recovery.
		The A11 Manager task is also known as the Signaling De-multiplexing task (SDT).
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
aaamgr	Authorization, Authentication, and Accounting (AAA) Manager	Paired with Session Managers.
		Performs all AAA protocol operations and functions for subscribers and administrative users within the system.
		Acts as a AAA client to AAA servers.
		Manages GTP Prime (GTP') messaging with charging gateway functions (CGFs).
		Multiple AAA Managers can run on a single CPU and/or can be distributed throughout any CPU present in the system.
		AAA operations for the CLI are done through a AAA Manager running on the active management card.
aaaproxy	Authorization, Authentication, and Accounting (AAA) Proxy Manager	Starts whenever the Global Configuration mode **gtpp single-source** command is configured. When GTPP single-sourcing is enabled, aaaproxy generates requests to the accounting server using a single UDP source port number, instead of having each AAA Manager generate independent requests with unique UDP source port numbers.
		Runs on a demux card when session recovery is enabled. If session recovery is not enabled, the Global Configuration mode **require demux card** command starts aaaproxy on the designated demux card.
		Writes CDRs to a file in its VRAM-disk. The enqueued CDRs are then periodically synchronized with a HDD for transfer.

Task	Description	Function
acsctrl	Active Charging System (ACS) Controller	Active Charging service is defined at the global level and can be utilized through CSS commands from any VPN context. Enable via the Global Configuration mode **active-charging service** CLI command.
		The ACS controller runs on the primary packet processing card and is responsible for managing the ACS service.
		Reads and writes ACS configuration information into SCT.
		The ACS Controller monitors the ACS Manager's recovery process and performs cleanup when redundancy is enabled.
acsmgr	Active Charging System (ACS) Controller	Created by ACS Controller to perform IP session processing for a specific number of flows.
		Sends and receives data through Session Managers.
		Active/Standby acsmgr tasks are created when session recovery (SR) is enabled.
cdrmod	Charging Detail Record Module	Responsible for receiving EDR/UDR records from different ACSMGR instances in the system.
		Responsible for writing the received EDR/UDR records in files using the configured file naming conventions.
dgmbmgr	Diameter Gmb interface Application Manager	Provides Multimedia Broadcast/Multicast Service (MBMS) feature support for GGSN. It is instantiated when an MBMS policy CLI is configured in the GGSN Service configuration mode. dgmbmgr
		Maintains the MBMS UE and bearer contexts.
		Handles the Gmb interface over a Diameter connection to a BMSC Server for MBMS bearer sessions. dgmbmgr recovers by polling all sessmgrs for MBMS session states and recreating the MBMS UE and MBMS bearer context information.

Task	Description	Function
diamproxy	Diameter Proxy	Created by diactrl (which runs as part of vpnctrl) and the number of diamproxy tasks spawned is based on the configuration to use "multiple" or "single" proxies. In instances that a single proxy is configured, only one diamproxy task is spawned for the entire chassis and runs on demux packet processing cards. When multiple proxies are configured, one diamproxy task is run per packet processing card.
		Maintains Diameter base connections to all peers configured in the system.
		Informs applications about any change in the connection status.
		Acts as a pass-through to the messages from application to the Diameter server.
		Just acts as a forwarding agent (does not maintain any queues).
		A single Diameter proxy is used to service multiple Diameter applications.
egtpemgrr	Enhanced GPRS Tunneling Protocol Egress Manager	Created by the Session Controller for each context in which an egtp-service of interface type sgw-egress or MME is configured.
		Handles certain EGTP messages from SGW, PGW.
		Maintains list of current EGTP sessions.
		Maintains list of current Session Manager tasks which aids in session recovery.
		Handles GTP Echo messaging.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
egtpimgr	Enhanced GPRS Tunneling Protocol Ingress Manager	Created by Session Controller for each context in which an egtp-service of interface type sgw-ingress or pgw-ingress is configured.
		Receives EGTP sessions from MME/S4 SGSN/SGW and distributes them to different Session Manager tasks for load balancing.
		Maintains list of current EGTP sessions.
		Maintains list of current Session Manager tasks which aids in session recovery.
		Handles GTP Echo messaging.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.

Task	Description	Function
gtpcmgr	GPRS Tunneling Protocol Control (GTP-C) Message Manager	Created by the Session Controller for each context in which a GGSN service is configured.
		Receives the GTP sessions from the SGSN and distributes them to different Session Manager tasks for load balancing.
		Maintains a list of current Session Manager tasks to aid in system recovery.
		Verifies validity of GTPC messages.
		Maintains a list of current GTPC sessions.
		Handles GTPC Echo messaging to/from SGSN.
gtpumgr	GPRS Tunneling Protocol User (GTP-U Manager	Created by the Session Controller for each context in which a GTPU service is configured. Supported for both GTPUv0 and GTPUv1
		Maintains a list of the GTPU-services available within the context and performs load-balancing (of only Error-Ind) for them.
		Supports GTPU Echo handling.
		Provides Path Failure detection on no response for GTPU echo.
		Receives Error-Ind and demuxes it to a particular Session Manager.
		Serves as the Default GTPU listener. GTPUMGR will process GTPU packets with invalid TEID.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
hamgr	Home Agent (HA) Manager	Created by the Session Controller for each context in which an HA service is configured.
		Receives Mobile IP sessions from the Foreign Agents (FAs) and distributes them to different Session Manager tasks.
		Maintains a list of current Session Manager tasks that aids in system recovery.
		Functions as the DemuxMgr – handles all the PMIP signaling packets.
		Functions as the Demuxmgr for MIPv6/MIPv4 HA.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.

Task	Description	Function
hnbdemux	Home NodeB (HNB) Demux Manager	Started as part of HNB-GW service creation procedure. There is only one hnbdemux in the chassis.
		Distributes incoming Iuh connections to HNB Managers in the system.
		Remains aware of all the active HNB-GW services in the system.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
hnbmgr	Home NodeB (HNB) Manager	Starts when an HNB-GW service configuration is detected. There can be multiple instances of this task for load sharing. All HNB Managers have all the Active HNB-GW Services configured and be identical in configuration and capabilities.
		Runs the SCTP protocol stack.
		Handles the SCTP associations.
		Maintains Home-NodeB databases.
		Provides nodal functions for Iuh interface on SCTP protocol.
		With session recovery (SR) enabled, this manager is usually established on one of the CPUs on the first active packet processing card.
imsimgr	International Mobile Subscriber Identity Manager for MME	Starts when an MME service configuration is detected. There is only one instance of this task:
		Selects which SessMgr to use for new subscriber sessions.
		Maintains and reports MME-related demux statistics on events like Attach by IMSI, Attach by GUTI, etc.
		Can interact with the following tasks in the system: - Session Controller - MME Manager - Session Manager
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.

Task	Description	Function
imsimgr	International Mobile Subscriber Identity Manager for SGSN	Started by the Session Controller.
		Selects SessMgr, when not done by linkmgr or sgtpcmgr tasks, for calls sessions based on IMSI/P-TMSI.
		Load-balances across SessMgrs to select one to which a subscriber will be assigned.
		Maintains records for all subscribers on the system.
		Maintains mapping between the IMSI/P-TMSI and SessMgrs.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active demux packet processing card.
ipsgmgr	IP Services Gateway Manager	Created by the Session Controller.
		In Server mode, acts as a RADIUS server, and supports Proxy functionality.
		In Snoop mode supports snooping RADIUS Accounting messages.
		Load balances requests among different SessMgrs.
		Activates and deactivates sessions.
l2tpdemux	L2TP Demultiplexor Task	Created by the Session Controller when an LNS service is created. Only one L2TPDemux task is invoked for the entire system.
		De-multiplexes and forwards new incoming tunnel create requests to L2TPMgrs.
		Maintains information about current active tunnels in all L2TPMgrs.
		Load balances requests among L2TPMgrs.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
l2tpmgr	Layer 2 Tunneling Protocol Manager	Created by the Session Controller for each context in which a LAC or LNS service is configured. Additional managers are created as needed depending on loading.
		Responsible for all aspects of L2TP processing.
		Maintains protocol state machines for all L2TP sessions and tunnels.
		Triggers IPSec encryption for new L2TP tunnels as needed.
		Works with Session Managers to gracefully bring down tunnels.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.

Task	Description	Function
linkmgr	SS7 Link Manager	Created by the Session Controller when the first SS7RD (routing domain) is activated.
		Multi-instanced for redundancy and scaling purposes.
		Provides SS7 and Gb connectivity to the platform.
		Routes per subscriber signalling across the SS7 (including Iu) and Gb interfaces to the SessMgr.
magmgr	Mobile Access Gateway (MAG) Manager	Created by the Session Controller when the first MAG service is created in a context.
		Sends and receives PMIP control messages (PBU/PBA).
		Adds an NPU flow to receive MIPv6 PBA packets. This flow is identical to the flow used in the HAMgr.
		Maintains the Binding Update List used to keep track of the mobile node's bindings.
		Originates PBU-based on trigger received from the Session Manager during error conditions.
		Receives PBA and forwards it to Session Manager.
		Supports debugging facility – "magmgr" and "mobile-ipv6".
mmgr	SGSN Master Manager	Created upon provisioning of SS7RDs/SCCP-NWs/etc. The Session Controller provides the initial system configuration which includes a detailed description of each distributed protocol layer, its resources sets, and a list of its service user protocol layers and service provider protocol layers.
		Runs as a single instance.
		Handles nodal SS7, Iu, and Gb functionality.
		Implements master linkmgr functionality for SS7 route status aggregation.
		Implements master linkmgr functionality for RNC and BSC status aggregation.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active demux packet processing card.
mmedemux	Mobility Management Entity Demux Manager	Started as part of MME service creation procedure. There is only one mmedemux in the chassis.
		Distributes incoming S1-MME SCTP connections to mmemgr tasks in the system.
		Remains aware of all the active MME services in the system.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.

Task	Description	Function
mmemgr	Mobility Management Entity Manager	Starts when an MME service configuration is detected. There can be multiple instances of this task for load sharing. All mmemgrs will have all the Active MME Services configured and will be identical in configuration and capabilities.
		Runs the SCTP protocol stack.
		Handles the SCTP associations.
		Maintains TA List.
		Manage eNodeB databases.
		Provides nodal functions for S1-MME protocol.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
pccdemux	Policy and Charging Control Bindmux Manager	Started as part of PCC service creation procedure. There is only one instance of BindMux MGR in the chassis.
		Handles multiplexing of the sessions across the available pccmgrs along with the session binding functions
		Monitors load on pccmgrs.
		Distributes incoming IP-CAN connections across pccmgrs in the system.
		Performs session binding; binds IP-CAN/Gateway session with the AF-Session.
		Ensures all messaging for an IMSI across various interfaces is directed towards the selected pccmgr.
		Remains aware of all the active PCC services in the system.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.
pccmgr	Policy and Charging Control Bindmux Manager	pccmgr is part of a Session Manager instance.
		Handles all PCRF service sessions.
		Interfaces with PCC-Core while processing different events associated with individual subscriber sessions.
		Maintains subscriber information while applying business logic.
		Creates calline and corresponding APN session for each subscriber.

Task	Description	Function
sgtpcmgr	SGSN GPRS Tunneling Protocol Control message Manager	Created by the Session Controller for each VPN context in which an SGSN service is configured.
		Terminates Gn/Gp and GTP-U interfaces from peer GGSNs and SGSNs for SGSN Services.
		Terminates GTP-U interfaces from RNCs for IuPS Services.
		Controls standard ports for GTP-C and GTP-U.
		Processes and distributes GTP-traffic received from peers on these ports.
		Performs all node level procedures associated with Gn/Gp interface.
		With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active demux packet processing card.
srb	Standard Routing Database	Eight srbs are created by the Session Controller when Content Filtering in the Enhanced Charging Service is enabled. A minimum of two packet processing cards are required to initiate these eight tasks.
		Receives the static database from the session controller. Each srb task loads two database volumes (one primary and one secondary). The srb task also stores the static DB.
		Rates and categorizes the URL based on the DB volumes and CSI (Category Set Index) stored on it.
		Performs peer loading in case its peer fails. If both the srb task and its peer fail, the session controller performs the loading.

Platform Processes

Table 7: Platform Process Tasks

Task	Description	Function
afctrl	ASR 5500 Fabric Controller [ASR 5500 only]	Responsible for the overall management of the system fabric. Manages the pool of Rendezvous Destinations and coordinates fabric recovery by the afmgr proclets after a fault. A single afctrl instance runs on the active MIO/UMIO only.
afmgr	ASR 5500 Fabric Manager [ASR 5500 only]	Responsible for the management of fabric resources on a particular card. There is one afmgr on every CPU that is responsible for one or more fabric access processors (FAPs) or fabric elements (FEs). afmgr supports recovery but not migration.

Task	Description	Function
afio	ASR 5500 Fabric I/O Driver [ASR 5500 only]	Responsible for the direct configuration of the fabric chipset. afio supports non-messenger interprocess communication (IPC) with the local afmgr and with other local afio instances
connproxy	TCP/SCTP Connection proxy	Allows applications on any card to share the same TCP/SCTP connection to the same remote endpoint instead of opening a new connection for each application on the card.
cspctrl	Card-Slot-Port Controller	Manages physical chassis components.
cssctrl	Content Server Selection (CSS) Controller	Maintains all global CSS properties which include a list of CSS servers that can be bound to a service in a context.
		CSS defines how traffic will be handled based on the "content" of the data presented by or sent to a mobile subscriber. CSS encompasses features such as load balancing, NAT, HTTP redirection, DNS redirection. The content server (services) can be either external to the platform or integrated within the platform. External CSS servers are configured via the Context Configuration mode **css server** command.
		The CSS Controller does not create CSS Managers. CSS Managers are stopped and started by VPN Managers. A CSS Manager is automatically created for each context.
cssmgr	Content Server Selection (CSS) Manager	Spawned by the VPN Manager within a StarOS context.
		Manages the keepalives to a CSS server within the specific VPN context.
		Fetches the CSS related information for a subscriber
		If a CSS server goes down, the cssmgr task reprograms the NPUs to by-pass the service or redistribute the data among the rest of the servers in the service.
dcardctrl	Daughter Card Controller [ASR 5500 only]	Spawns daughter card managers during system initialization and monitors daughter card managers during system steady state execution. It also spawns daughter card managers whenever a daughter card manager task fails.
dcardmgr	Daughter Card Manager [ASR 5500 only]	Responsible for managing IPSec Security Associations for AH- and ESP-based sessions.
		Interfaces with the on-board hardware accelerated cryptographic chip which executes cryptographic algorithms associated with the given IPSec Security Associations.
dhmgr	Distributed Host Manager	Started automatically on each CPU by SITPARENT.
		Coordinates establishment of locally terminated TCP, SCTP, and UDP connections on behalf of multi-instanced tasks such as Diameter endpoints among sessmgr tasks.

Task	Description	Function
drvctrl	Driver Controller	Centralizes access to many of the system device drivers. It also performs temperature and voltage monitoring.
hdctrl	Hard Drive Controller	Controls and manages the drive array spanning the management cards.
hwctrl	Hardware Controller	The hwctrl task has several timers that manage polling loops for hardware sensor readings, sensor threshold monitoring, and fan tray monitoring.
hwmgr	Hardware Manager	The hwmgr task runs on all cards in the chassis to read local accessible hardware sensors and report them back to the hwctrl.
inetd	InterNET Service Daemon	The subsystem responsible for starting most of the network services.
		Listens for requests from connecting clients, such as FTP, SFTP, and telnet. When a TCP packet or UDP packet arrives with a particular destination port number, inetd launches the appropriate server program to handle the connection. **Note:** FTP and Telnet are not supported.
ipsecctrl	IPSec Controller	Started by SIT on system startup regardless of configuration.
		Starts ipsecmgr tasks based on configuration and maintains its list for task recovery.
		Receives and maintains user configuration for IPSec.
		Manages the configured IPSec crypto maps and its assignment to ipsecmgrs.
		Interfaces with the vpnmgr task for required IPSec configuration parameters such as IP Access Lists, IP pools, interface addresses, and interface state notifications.
ipsecmgr	IPSec Manager	Created by the Session Controller, establishes and manages secure IKEv1, IKEv2 and IPSec data tunnels.
kvctrl	Key Value Controller	Central key value store (kvstore) function that runs on the management card. Its primary function is to support recovery and distribution functions.
lagmgr	Link Aggregation Group Manager [ASR 5500 only]	Started by npuctrl on the demux card's primary CPU (ASR 5000) or MIO (ASR 5500) with a facility level between CSP and npumgr to receive configuration/status notification from npumgr and build global LAG database.
		Exchanges control packets (LACP and Marker) over configured physical ports with peers to reach agreement on an aggregation of links.
msgd	Messenger Daemon	Implements the Name Service and related functions for the internal message passing system.

Task	Description	Function
msgproxy	Message Proxy	The Messenger Proxy process handles broadcast messages send from any single application (referred to as a client) to any facility which has one instance per thread (referred to as the Target Facility).
		One msgproxy task runs on each CPU complex on the SMC (ASR 5000), DPCs (ASR 5500) and SF Virtual Machine (VPC-DI).
		Processes incoming broadcast messages from the Client processes, such as sessctrl, distributes them to the correct Target Facility, such as sessmgr, creates the correct responses and sends them back to the correct Client.
nscontrol	Name Service Controller	As part of the Messenger process, provides a reliable channel for tasks to send control messages to the Messenger Daemon.
ntpd	Network Time Protocol (NTP) Daemon	Maintains the system time in synchronization with time servers using NTP. Enabled when one or more NTP servers have been configured via the NTP Configuration mode **ntp server** CLI command.
rct	Recovery Control Task	Monitors tasks/managers/facilities across the system and performs recovery in the event of a failure.
sct	Shared Configuration Task	Performs the redundant storage of configuration information and other state information in an in-memory database.
sft	Switch Fabric Task	Monitors the switch fabric and the gigabit Ethernet control plane.
sshd	Secure SHell Daemon	Supports secure login to the StarOS CLI. Enabled via the Context Configuration mode **server sshd** CLI command.
ucm	Utilities Configuration Manager	DHCPD, DNS, FTPD, INETD, NTPD, PING, RLOGIN, SFTPD, SFTP-SERVER, SNMPD, SSH, SSHD, TELNET, TELNETD, TFTPD, TRACEROUTE **Note:** FTP and Telnet are not supported.

Management Processes

Table 8: Management Process Tasks

Task	Description	Function
bulkstat	Bulk Statistics Manager	Periodically polls and gathers bulk statistics and transfers this data to external management systems.
evlogd	Event Log Daemon	Handles event logging functions including the interface to external syslogd servers and the internal event logs.

Task	Description	Function
orbs	ORBEM Service [ASR 5500 only]	The orbs task is also known as the ORB Element Manager (ORBEM). An Element Management System (EMS) requests orbs to perform Element Management Functions on the system using secure IIOP. ORBS then interacts with concerned Controller Tasks to execute the function. The response/errors from the execution are interpreted, formulated into an EMF response, and handed off to EMS servers.
orbns	ORBEM Notification Service [ASR 5500 only]	Notifies the EMS servers of event occurrences.
		Registers such EMS servers and subscribes them to associated event types.
		As the events occur, the concerned Controller Task notifies orbs (ORBEM), which then notifies the subscribing EMS servers.
sesstrc	Session Trace Collection Task	Implements the standards-based session trace functionality.
		Manages both CLI and signaling-based subscriber traces. It collects messages to be traced and generates trace files as needed. It uploads trace files to the Trace Collection Entity as needed.
snmp	Simple Network Management Protocol	Handles inboard SNMP operations if configured, and sends SNMP notifications (traps) if enabled.
threshold	Threshold Server	Handles monitoring of threshold crossing alerts, if configured. Polls the needed statistics/variables, maintains state, and generates log messages/SNMP notification of threshold crossings.

StarOS Tasks