ORACLE® ACME PACKET SBC FAMILY

Security Guide

August 2017

Contents

Related Documentation

The following table lists related documentation.

Document Name	Document Description
ACLI Configuration Guide	Contains information about the administration and software configuration of the SBC.
ACLI Reference Guide	Contains explanations of how to use the ACLI, as an alphabetical listings and descriptions of all ACLI commands and configuration parameters.
Maintenance and Troubleshooting Guide	Contains information about logs, performance announcements, system management, inventory management, upgrades, working with configurations, and managing backups and archives.
MIB Reference Guide	Contains information about Management Information Base (MIBs), Acme Packet’s enterprise MIBs, general trap information, including specific details about standard traps and enterprise traps, Simple Network Management Protocol (SNMP) GET query information (including standard and enterprise SNMP GET query names, object identifier names and numbers, and descriptions), examples of scalar and table objects.
Accounting Guide	Contains information about accounting support, including details about RADIUS accounting.
HDR Resource Guide	Contains information about the Historical Data Recording (HDR) feature. This guide includes HDR configuration and system-wide statistical information.
Administrative Security Essentials	Contains information about support for its Administrative Security license.
Release Notes	Contains information about the current documentation set release, including new features and management changes.

Revision History

The following table lists changes to the documentation.

Date	**Revision**
November 2015	Initial release.
January 2016	Removes FIPS content.
July 2016	Corrected error in max-untrusted-signaling description
November 2016	Adds Acme Packet 3900 content
August 2017	E-CZ7.5.0 updates

Part 1: Overview

Product Overview

The Oracle Session Border Controller (SBC) family of products are designed to increase security when deploying Voice over IP (VoIP) or Unified Communications (UC) solutions. Properly configured, Oracle's SBC family helps protect IT assets, safeguard confidential information, and mitigate risks—all while ensuring the high service levels which users expect from the corporate phone system and the public telephone network.

Installed at the network perimeter, the SBC family of products provides a demarcation and enforcement point for the UC network. Its primary security functions include:

- Overload protection to prevent DoS attacks and registration floods
- Access control to inhibit toll fraud and service theft
- Topology hiding to counter topology discovery through reconnaissance scans
- Encryption and authentication to ensure privacy and prevent loss of confidential information
- Protocol validation to combat fuzzing and other types of malicious attacks

Net-SAFE Security Framework

The Oracle Net-SAFE™ security framework addresses the unique security challenges of delivering SIP-based interactive IP communications over the Internet. The Net-SAFE framework includes advanced security features, a highly-scalable architecture, and comprehensive monitoring and reporting capabilities. The framework reduces risk in UC services and applications by ensuring confidentiality, integrity and availability.

Net-SAFE goals

- *Protect the SBC*: The first line of defense at the border is the SBC, so it needs to be secure and resistant to attacks and overload.
- *Protect the infrastructure*: The infrastructure includes the customer's network of multimedia equipment (soft switches, application servers, SIP proxies, H.323 gatekeepers, gateways, etc.).
- *Protect the service*: Preventing attacks is not enough. UC services that generate revenue need to remain in service.

Net-SAFE Requirements

The Net-SAFE framework identifies the requirements that an SBC must satisfy to meet the goals of the framework and provide confidentiality integrity, and availability.

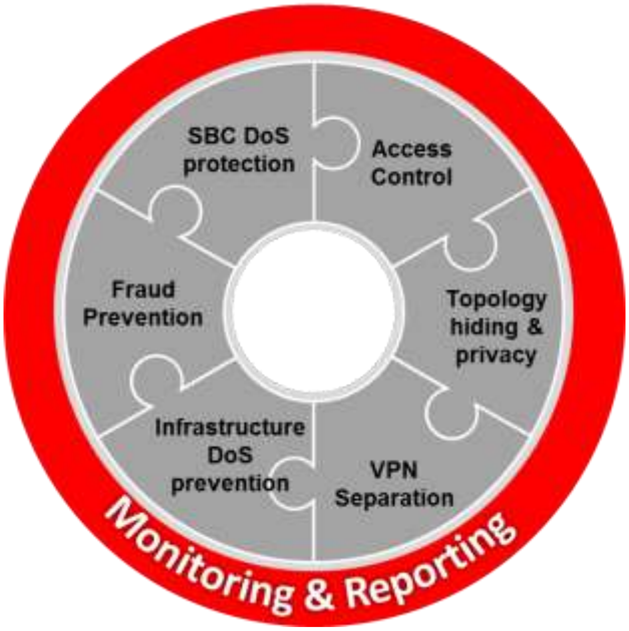


Figure 1: Net-SAFE Framework

The Net-SAFE Framework spans seven general functions:

1. Denial of Service (DoS) protection
   - Dynamic self-protection against malicious and non-malicious DoS attacks and overloads at layer 3/4 (e.g. TCP, SYN, ICMP, fragments, etc.) and layer 5 (e.g. SIP signaling floods, malformed messages, etc.)
   - Traffic management queues for control and throttling of signaling and media
2. Access control
   - Session-aware access control for signaling and media using static and dynamic permit/deny ACLs at layers 3 and 5
   - ACL/DOS protection for the management interface
3. Topology hiding and privacy
   - Complete infrastructure topology hiding at all protocol layers for confidentiality and attack prevention as well as modification, removal or insertion of call signaling application headers and fields
   - Confidentiality and integrity through use of industry-standard encryption methods such as TLS/SRTP and IPSec
4. VPN separation
   - Support for Virtual Private Networks (VPNs) with full inter-VPN topology hiding and separation
   - Ability to create separate signaling-only and media-only VPNs
   - Optional intra-VPN media hair-pinning to monitor calls within a VPN
5. Service infrastructure DoS prevention

- Per-device signaling and media overload control, with deep packet inspection and call rate control to prevent DoS attacks from reaching service infrastructure

6. Fraud prevention

- Session-based authentication, authorization, and contract enforcement for signaling and media

7. Monitoring and reporting

- Audit trails, event logs, access violation logs and traps, management access command recording
- Call Detail Records (CDRs) with media performance monitoring
- Raw packet capture ability
- Lawful intercept capability

General Security Principles

The following principles are fundamental to using any application securely.

Keep Software Up To Date

One of the principles of good security practice is to keep all software versions up to date. Oracle maintains multiple SBC streams or versions that are updated with applicable security patches. Always review the Critical Patch Updates and Release Notes relevant to the stream installed to determine whether an update should be applied.

Restrict Network Access to Critical Services

By design, the SBC family defaults to a closed state. No signaling or media can pass through the system unless it is explicitly configured.

Only services required for initial configuration of the system are available on a dedicated management Ethernet port (wancom0) which should be connected to a management network. Insecure services such as telnet and FTP should be disabled. Access to management services should be protected through the use of system level Access Control Lists (ACL) specifying allowed IP address ranges.

Signaling and media are only available on a separate set of Ethernet ports designated for services. ACLs should also be used on services ports for SIP peering deployments where possible. Some management capabilities can be enabled on these services ports by an administrator, so care should be taken to determine the risk of doing so in individual cases. In general it is not recommended to enable services other than perhaps ICMP.

Services should also be protected from DoS abuse through configuration of call admission controls, signaling thresholds, blacklisting, and attack tool detection, elements covered as part of this guide.

Follow the Principle of Least Privilege

The SBC family provides some implicit least privilege because direct user access is usually not provided. In most cases, the system acts as a proxy device so there is no direct user interaction. In other cases the system may provide a registrar function. However, providing the registrar function does not give the user access to any system level commands.

Administrators are the only ones who have any sort of system logon permissions. The system provides Role Based Access Control with dedicated user accounts that have pre-assigned privilege levels in the

Command Line Interface. These are discussed further in the section on management interfaces. RADIUS and TACACS+ can be enabled as well to enable an outside authentication and authorization function. The minimum authorization class for RADIUS and command set should be considered for the administrator's role.

Monitor System Activity

Monitoring system activity is critical to determine if someone is attempting to abuse system services and to detect if there are performance or availability issues. Useful monitoring information can be acquired through SNMP, RADIUS accounting, Historical Data Recording (HDR), and Syslog. At a minimum SNMP should be configured, and use of an external syslog server should be considered.

Keep Up To Date on Latest Security Information

Security issues that require a software or configuration update will be communicated in quarterly Critical Patch Updates (CPU). The latest CPUs as well as instructions to subscribe to them can be found at http://www.oracle.com/technetwork/topics/security/alerts-086861.html. A free Oracle Technology Network account is required to receive CPUs.

SBC Specific Security Principles

Security teams should consider the following guidelines when deploying a Unified Communications (UC) system. These are some of the areas where the SBC family will provide value.

- Create a demarcation and enforcement point for the UC network: The enforcement point provides demarcation between zones of varying trust, such as the internal enterprise network, a BYOD network, a guest network, a demilitarized zone, or the public Internet.

- Hide topology: Hackers can plan attacks by ascertaining information about network equipment (determining equipment types and software versions) or by detecting the IP addressing scheme a company employs. A UC demarcation device should remove any protocol fields that may assist in "fingerprinting" and should provide NAT (network address translation) at all protocol levels to conceal internal addressing schemes.

- Encrypt endpoint communications: Businesses should encrypt communications flows when transiting public networks to prevent eavesdropping or impersonation. Encryption should also be considered on private networks to verify identity and prevent eavesdropping on privileged communications. Encryption can hinder lawful interception or other regulatory and corporate compliance requirements, so be sure to understand any impacts in your environment. By establishing a UC demarcation point and anchoring, unencrypting, and re-encrypting sessions at the network perimeter, security teams can tap or replicate sessions in the clear for compliance purposes.

- Normalize protocol differences on-demand: Because UC venders implement SIP differently, using devices from multiple venders may cause interoperability problems. In extreme cases, the "normal" messaging from one manufacturer might cause failures or outages for another. Rather than depending on vendors to fix these interoperability issues, it is preferable to do so, in real-time, using an SBC.

- Prevent DoS attacks and overloads: DoS or Distributed DoS (DdoS) attacks and other non-malicious events such as registration floods can impair IP communications infrastructure (border elements, application servers, endpoints) and disturb critical applications and services. Attackers may try to flood a network from one or more endpoints or may send malformed messages (protocol fuzzing) to overwhelm network devices. A UC demarcation device can ensure continued service availability by identifying DoS and DdoS attacks, and appropriately throttling or blocking traffic.

Part 2: Secure Installation and Configuration

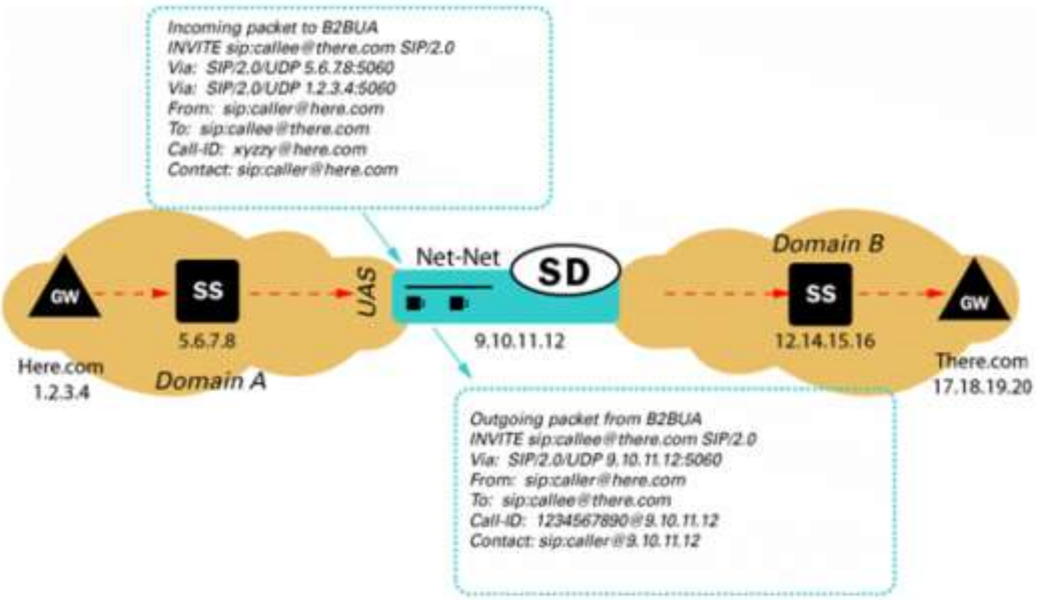
Recommended Deployment Topologies

This section outlines the planning process for a secure installation and describes several recommended deployment topologies for the system.

Session Border Controller

The SBC family products can be deployed following several generalized topology types; Peering (sometimes called Trunking), Access (also called Hosted IP Services), and Hybrid which combines the two models.

Peering

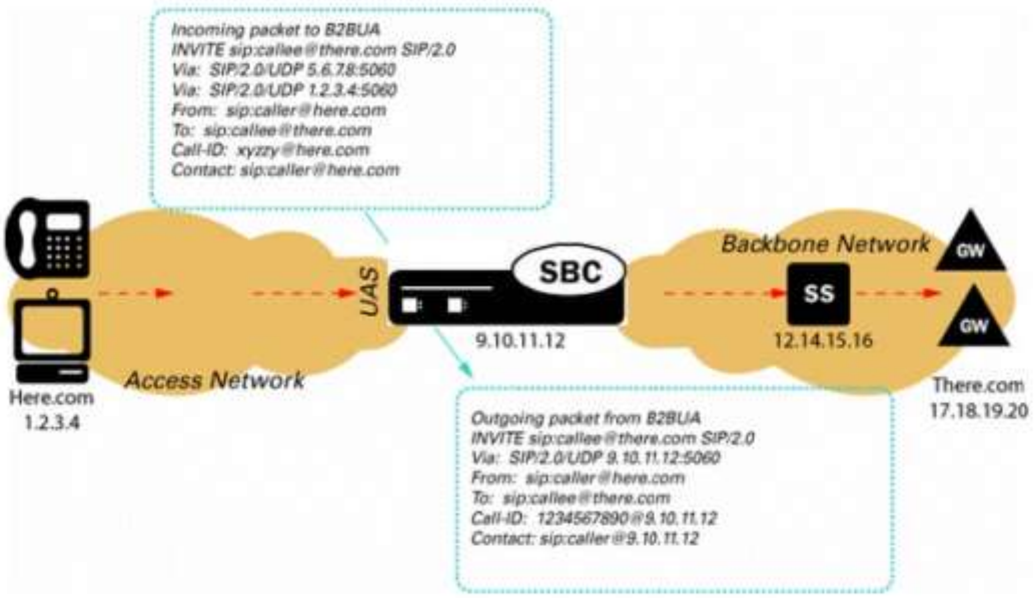


```
Incoming packet to B2BUA
INVITE sip:callee@there.com SIP/2.0
Via:  SIP/2.0/UDP 5.6.7.8:5060
Via:  SIP/2.0/UDP 1.2.3.4:5060
From:  sip:caller@here.com
To:  sip:callee@there.com
Call-ID:  xyzzy@here.com
Contact: sip:caller@here.com
```

```
Outgoing packet from B2BUA
INVITE sip:callee@there.com SIP/2.0
Via:  SIP/2.0/UDP 9.10.11.12:5060
From:  sip:caller@here.com
To:  sip:callee@there.com
Call-ID:  1234567890@9.10.11.12
Contact: sip:caller@9.10.11.12
```

In a peering model the SBC is contacted by a SIP server to relay endpoint signaling information. The SIP server may be a PBX, registrar, proxy, SBC, or other device. The IP of the device is usually trusted and pre-provisioned in the SBC as an endpoint (session agent) that will be relaying calls. Since the remote endpoint is already known, Access Control Lists (ACL) and Call Admission Controls (CAC) can be pre-provisioned for the appropriate level of protection or service level assurance.

Access



```
Incoming packet to B2BUA
INVITE sip:callee@there.com SIP/2.0
Via:  SIP/2.0/UDP 5.6.7.8:5060
Via:  SIP/2.0/UDP 1.2.3.4:5060
From:  sip:caller@here.com
To:  sip:callee@there.com
Call-ID:  xyzzy@here.com
Contact: sip:caller@here.com
```

```
Outgoing packet from B2BUA
INVITE sip:callee@there.com SIP/2.0
Via:  SIP/2.0/UDP 9.10.11.12:5060
From:  sip:caller@here.com
To:  sip:callee@there.com
Call-ID:  1234567890@9.10.11.12
Contact: sip:caller@9.10.11.12
```

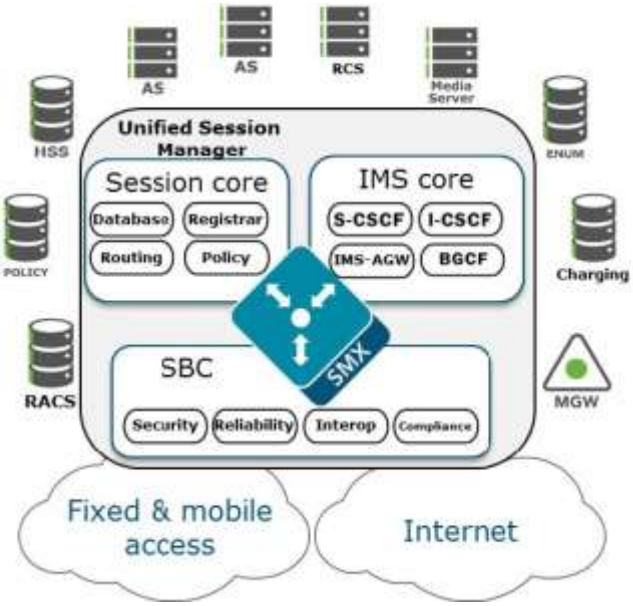
In an access model the SBC is contacted by a SIP endpoint to relay endpoint signaling information. The IP address of the endpoint is usually not known, so trust should be established through behavior such as establishment of a successful registration. Once the endpoint becomes trusted, dynamic Access Control Lists (ACL) and Call Admission Controls (CAC) can be applied. Monitoring of potentially abusive behaviors provides a mechanism to “demote” or blacklist endpoints.

Hybrid

A hybrid model combines both Peering and Access topologies into a single configuration. This is a fairly common model, where remote users use a registrar server in the core network, but their calls are forwarded to a service provider on one of the peer connections.

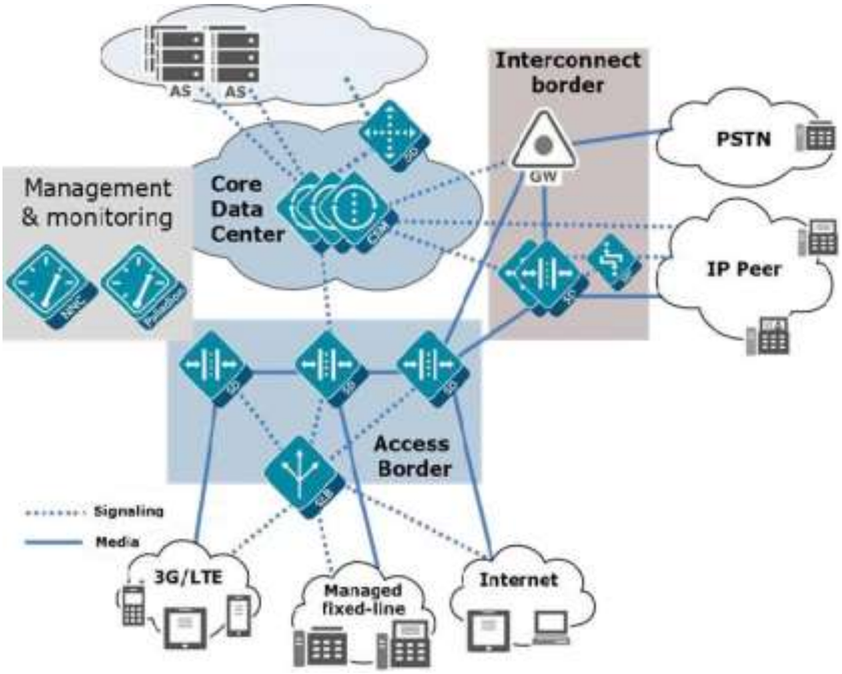
Unified Session Manager

The Unified Session Manager (USM) provides edge security for an IMS network, and should be positioned at access borders to integrate “traditional” SBC functionality with the core IMS session control functions. It provides a user registrar, local subscriber tables and Call Session Control Function components such as Proxy CSCF, Interrogating CSCF, Session CSCF, IMS Access Gateway, Emergency CSCF and Breakout Gateway Control Function.



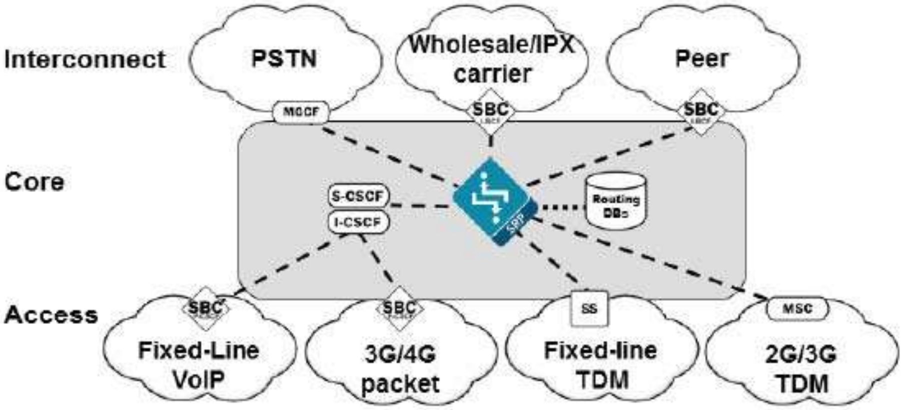
Core Session Manager

The Core Session Manager, which should never be positioned at a network edge, is used as a core session controller between multiple network types. It supports SIP in IMS and non-IMS environments, application servers, media servers, gateways, etc. It can be deployed in a distributed, virtualized model on COTS server hardware. The CSM can be used for session routing, interoperability assurance, CAC, and subscriber database integration through HSS, ENUM, or local subscriber table databases.



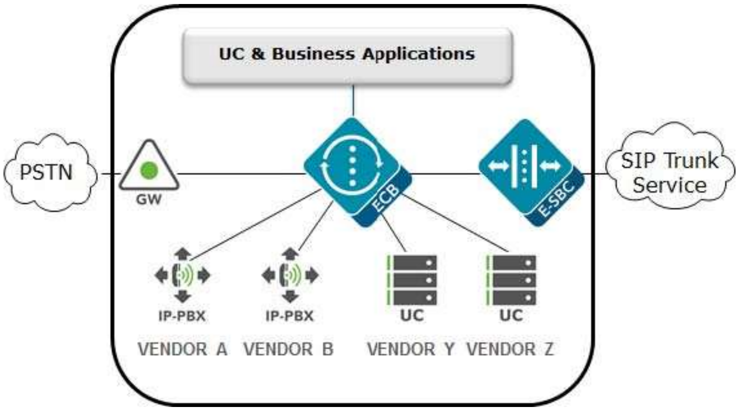
Session Router

The Session Router is a “pure” SIP session router that can be positioned in either a core network or at network borders. When installed at a border, the same SBC protections used in peering and access models can apply. In the network core, the emphasis is on routing and interoperability.



Enterprise Communications Broker

The Enterprise Communications Broker (ECB) should only be deployed within an enterprise core network, and not on the edge. Instead of perimeter security, the ECB is oriented towards functions such as dial plan management, centralized session, routing, CAC, load balancing, and interworking.



Realm Design Considerations

As a general rule, separate realms are created for external untrusted traffic and internal trusted traffic. However, there are many deployment complications that prevent that simple model from being used. Examples of these might include:

- A mix of user endpoints, gateways, or peer trunks on the untrusted network
- Varying capabilities or incompatibilities of user agents
- Impacts of blocking traffic to one group of users vs. another (i.e. trust low or medium)
- Service level agreements (SLA) that require Call Admission Controls (CAC)

A few of the general rules for Realm design include:

- Separate endpoints into realms based on trust level (high, medium, low) and that the response to detected abuse is appropriate for them (no action, demotion or blocking)

- Create multiple realms for endpoints based on the type of device – a user endpoint, a gateway, or a peer – since they will have very different considerations for SIP Header Manipulation, trust, signaling thresholds, endpoints behind NAT, and CAC.
- Consider increasing the deny-period from 30 seconds to something longer depending on how much abuse it is believed will be received from a public network and what type of delay users may tolerate.
- Set `restricted-latching` to `sdp` so only media received from the IP and port negotiated in signaling will be allowed.
- Pay close attention to the media management settings required for the endpoints and traffic flows (see the `mm-` parameters on the realm). If one way-audio is experienced this is one place to start investigating.

Management Interfaces

The Oracle SBC has two types of interfaces, one for management and the other for signaling and media (otherwise known as services interfaces). Security configuration for each interface is treated separately.

Two management interfaces allow access to the SBC for configuration, monitoring and troubleshooting purposes; a serial (console) interface and an Ethernet interface for remote management (wancom0).

Serial (Console) Interface

As with any industry standard serial interface to a network element, minimal security functions are available. The physical security of the installation location should be assured since console access cannot be blacklisted. However, the Admin Security license (discussed later) does allow for the console port to be disabled.

To avoid unauthorized access to the console interface the console-timeout should be configured to automatically disconnect the console session after an appropriate period of time (i.e. 300 seconds). Timeouts are disabled by default.

If the console port detects a cable disconnect it will also log out any logged in user to prevent unauthorized use.

The console interface should only be connected to a terminal server if the terminal server is deployed in a secure non-public network.

Configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

Management Port Configuration

The Wancom0 management interface MUST be connected to and configured on a management network or subnet separate from the service interfaces. If it is not, the SBC is subject to ARP overlap issues, and loss of system access when the network is down or under DdoS attack. Oracle does not support SBC configurations with management and media and service interfaces on the same subnet.

Configuration is detailed in Section 2 “Getting Started” and Section 3 “System Configuration” of the *ACLI Configuration Guide*.

Passwords

The SBC provides two levels of user accounts through the Acme Packet Command Line Interface (ACLI): User and Superuser (the “user” and “admin” accounts).SBC no longer supports default passwords and requires these to be changed on first login.

Alternatively, the SBC supports the management of passwords via external RADIUS and TACACS+ servers for finer grain access control. The SBC supports communications with up to six RADIUS servers for this function. At least two entries should be configured to prevent service interruption.

The SBC encrypts sensitive configuration data in the configuration file using a Protected Configuration Password (PCP). This administratively configured password provides security and convenience when migrating configurations to different SBCs. All user passwords should be changed; however, it is especially important to change the PCP ("config" user password) so passwords and keys stored in the config file are secure. TLS, Ipsec, and HDR features are protected by the PCP:

> **CAUTION**: Once the PCP password is changed the sensitive information (certificates, IPSec shared secrets, etc) in your configuration file will be re-encrypted using the new PCP the new encryption "salt." As a result, previously backed up configuration files cannot be restored unless the password is restored to the value that configuration file was encrypted with.

Configuration is detailed in Section 2 "Getting Started" of the *ACLI Configuration Guide*, and Section 4 "System Management" of the *Maintenance and Troubleshooting Guide* in the subsection entitled "Setting a Protected Configuration Password: Matching Configurations."

The SBC provides a backup user for HDR file synchronization that must be changed. The backup user password can be set using the command "secret backup". The "secret" command is detailed in Section 3 of the *ACLI Reference Guide*.

The SBC provides one user for administration of legal intercept functions when a Lawful Intercept ("LI") license is installed – li-admin. The first time lawful interception is configured you will be prompted to change the password. However if you have installed the license, but never configured lawful interception, the default password may be active and usable via SSH. Procedures to change the password are detailed in the *Net-Net LI Documentation Set*.

Boot Flags

Boot parameters specify what information the system uses at boot time when it prepares to run applications. The boot parameters allow definition of an IP on the management interface, set the system prompt, and determine the software load that will be used. In addition, there is a boot flag setting that may modify the file location to be used, but may also enable additional features. Administrator access to the command line interface is required to modify the bootflags.

There is seldom a reason to change the boot flag from its default value (0x08). Changes to the boot flags are usually only needed for hardware testing or recovery, debugging, etc.

A few boot flag values that are disabled by default have security implications. These should only be enabled at the direction of Oracle technical support.

- 0x01 – Turns off the hardened interface protection on the media interfaces, allowing all ingress traffic
- 0x10 – Enables a second sshd server that provides access to the linux system console. This server process is different from the ssh server used to access the ACLI for configuration.
- 0x80008 – enable source routing on the management port

For further information on boot flags refer to Section 4 "Configuration Elements A-M" of the *ACLI Reference Guide*.

System ACLs

The Wancom0 Ethernet management interface should always be deployed in a secure non-public network.

The SBC provides static System Access Control List functionality (ACL) to protect the Wancom0 interface from other devices that can access the management LAN remotely. Only the management station(s) authorized for SBC access such as the Oracle Communications Session Element Manager should be permitted with ACLs. All system ACLs are considered “allow” ACLs, and include a specific IP source address / netmask and the IP protocol allowed. As the first ACL is created an implicit deny rule is inserted as the final ACL.

The “system-access-list” configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

Telnet/SSH

By default, telnet is enabled on the SBC Wancom0 management interface. Disable telnet and use SSH instead. The exception is the 6000 series platforms which disable telnet by default.

To avoid unauthorized access to the telnet/SSH interface, a timeout should be configured to automatically disconnect the terminal session after an appropriate period of time (i.e. 300 seconds). Timeouts are disabled by default.

The SBC supports viewing, importing, and deleting public ssh keys used for authentication of SSHv2 sessions.

Configuration is detailed in Section 2 “Getting Started” of the *ACLI Configuration Guide*, and Section 4 “System Management” of the *Maintenance and Troubleshooting Guide*.

FTP/SFTP

FTP is disabled on 7.x releases. Only SFTP is supported.

GUI Management

The SBC can be managed by the Oracle Communications Session Element Manager via ACP through management interface over TCP ports 3000 and 3001.

By default these ports are enabled in system-config > remote-control. If the SBCs are not remotely controlled by a Session Element Manager then this feature should be disabled.

Enterprise SBC releases since ECZ730, ACP communication is recommended to be protected by TLS. ACP-TLS is enabled by configuring **acp-tls-profile** in **system-config**.

> **CAUTION**: Disabling the remote-control feature is incompatible with the SBC HA architecture. Hence this functionality is considered optional and should only be deployed where HA and EMS are not used. If the SBCs are deployed in HA configuration, then the remote-control parameter needs to be enabled for the acquire-config feature to function properly.

Configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

Web Management

Depending on the release of code in use, a web based management interface may be accessible via the management network connected to wancom0. The web interface is disabled and not supported for Service Provider SBCsdo, but Enterprise SBCs include a full featured management and provisioning system.

By default the web interface is disabled. It can be accessed via the wancom0 IP address when enabled. Note that even if the web interface is disabled that the SBC will respond on port 80 by default. However, all new connection requests are immediately torn down with a TCP RST since there is no web server process running, and no kernel rule to forward the request to the web server.

Oracle recommends that only HTTPS be enabled on this interface so TLS will be used instead of the default HTTP. Care should be taken when defining the cipher list in the tls-profile so that administrative traffic cannot be compromised. The default cipher list is "ALL", which includes some insecure ciphers for backwards compatibility. The cipher list should be set manually to remove insecure ciphers. The recommended cipher list in order of preference includes:

- on 6000 series hardware:
    - TLS _DHE_RSA_WITH_AES_256_GCM_SHA384
    - TLS _DHE_RSA_WITH_AES_128_GCM_SHA256
    - TLS _DHE_RSA_WITH_AES_256_SHA256
    - TLS _DHE_RSA_WITH_AES_128_SHA256
- on hardware other than the 6000 series
    - TLS_DHE_RSA_WITH_AES_256_CBC_SHA
    - TLS_RSA_WITH_AES_256_CBC_SHA

Note that the DHE ciphers provide perfect forward secrecy, which prevents the session from being decrypted later even if the private key is discovered. Following is an example of system->web-server-config:

```
    state                      enabled
    inactivity-timeout              5
    http-state                   disabled
    http-port                    80
    https-state                  enabled
    https-port                   443
    tls-profile                  strong-ciphers-tls-profile
```

Configuration is detailed in Section 2 "Getting Started" of the *ACLI Configuration Guide*.

Resiliency

Several features enable availability, a key component of a secure deployment.

High Availability

It is strongly recommended that the SBC be deployed in a High Availability (HA) architecture with a Primary node and a Secondary node connected over both Wancom1 and Wancom2 interfaces for resiliency. It is also recommended that the two units in an HA pair be directly cabled together. While they can be separated and connected via an Ethernet switch or layer 2 VPN, this introduces latency and can significantly impact capacity. Since session replication is performed over a clear text connection, it may also expose call or configuration data sent in the replication process. In short, a geographically redundant pair of SBCs is not recommended. If geo-redundancy is an absolute requirement, a secure site-to-site VPN should be implemented for session replication, and thorough testing should be conducted to understand impacts to session capacity.

Guidelines are presented in "520-0011-03 BCP – High Availability Configuration".

Configuration is detailed in Section 14 “High Availability Nodes” of the *ACLI Configuration Guide*.

Link Detection and Gateway Polling

If the gateway-heartbeat is enabled, the SBC periodically sends ARP requests for each configured network-interface gateway. If the configured number of retransmissions has been exceeded, the SBC will mark that gateway as unreachable and decrement its health score. If the health score decrements far enough, and the health score of the standby unit is higher, an HA failover will occur.

It is recommended that exactly one network-interface per physical interface have gateway-heartbeat enabled.

The following configuration fragment depicts the recommended default settings for the gateway heartbeat sub-element. It is also advisable to increment the health-score value by one with each new heartbeat configuration for ease of failure identification based on score.

```
Gw-heartbeat
state             enabled
heartbeat         10
retry-count       3
retry-timeout     3
health-score      30
```

The feature is explained in detail in Section 14 “High Availability Nodes” of the “Net-Net 4000 <Current Release> ACLI Configuration Guide”.

Physical Link Redundancy

Physical Link Redundancy can be configured between the two virtual slot pairs on a physical NIU. Should the active interface fail, the standby will take over.

The active interfaces are slot 0 port 0 (M00) and slot 1 port 0 (M10). The standby interfaces are slot 0 port 1 (M01) and slot 1 port 1 (M11).

Although this feature can be used in conjunction with the max-signaling-bandwidth feature and static ACLs, this feature is incompatible with the Net-SAFE Architecture. Hence this functionality is considered optional and should only be deployed where DdoS protection is not a factor (e.g. Peering deployments), and where QoS metrics are not required. This feature is not commonly deployed.

Configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

Part 3: Security Features

This section outlines specific SBC security mechanisms.

The Security Model

The Oracle Communications SBC is a purpose built device providing customers both centralized and distributed control of the management and security of UC networks. The SBC is a critical network security element for VoIP services designed to effectively manage sessions and protect core network elements from various types of DdoS attacks, including malicious and non-malicious signaling overload attacks. The SBC is the sole ingress and egress point for all signaling messages (SIP/H.323/MGCP) and media streams to/from the core network and is therefore generally the demarcation point between trusted and untrusted network boundaries. Hence it is vital that the SBC be as secure and available as possible.

Oracle provides a number of industry leading techniques through SBC configuration to secure the network border. Some of these features are enabled "out of the box" and some require further analysis of the network architecture to determine the most optimal configuration for security.

For example, the SBC performs access control based on layer 5 signaling messages as one of its primary functions. The SBC is designed to allow authorized VoIP communications into the core network by opening/closing firewall ports and by performing NAPT (network address and port translations) on all signaling and media IP packets as one of its core functions. Signaling messages, going to and from the SIP core servers and residential gateways and/or peering affiliate infrastructure is therefore inspected and rewritten as necessary by the SBC.

The SBC follows a "closed" philosophy where ports and interfaces are closed by default and opened on an as-needed basis. Therefore the system will generally have ports, services and processes disabled unless configured.

Net-SAFE Architecture: SBC & Core Infrastructure Protection

The SBC provides several techniques for protecting the SBC, and therefore the service, from DdoS attacks.

First, traditional static ACLs should be configured to only permit signaling traffic from trusted devices. Permit ACLs are applicable for both unsecured networks (peering partner's SBCs, proxies, gateways) and secure network devices (core network softswitches, media servers, application servers, gateways). All other devices should be denied access to the SBC through the use of deny ACLs.

This solution does not scale for hosted NAT traversal (or hosted access) based applications where thousands of remote endpoint devices with dynamic IP addresses communicate directly to the SBC signaling interfaces.

The SBC provides the following tools for DdoS protection in Access networks:

- Protect the SBC core CPU via configurable sized queues and separation of signaling packets (trusted, untrusted)
- Configurable trust-level (none, low, medium, high)
- Wire speed hardware classification of every remote device trust-level
- Provide fair access for new/untrusted devices to signaling queue
- Multi-queue access fairness for unknown traffic

- Automatic behaviorally driven promotion/demotion/denial of devices
- Per-device constraints and authorization
- Protection against attack from behind NAT

Each device is classified as untrusted, trusted or denied. The entire system bandwidth is allocated for the trusted and untrusted queues according to the characteristics of the customer Access deployment (e.g. number of endpoints, rate of registration, packet size, etc.). The allocation of the CAM is configurable to tailor the sizes of the entries available for media, trusted and deny NAT entries according to the scale of the customer Access network. Separate configurable sized queues also exist for fragmented packets and ARP requests. In addition, a whole NAT device can be demoted based on the collective behavior of endpoints behind the NAT.

The trust-levels below determine promotion/demotion criteria between the deny list, untrusted and trusted queues.

- None: Device is always untrusted, no promotion or demotion
- Low: Device is initially untrusted, can be promoted to trusted, or demoted to denied
- Medium: Device is initially untrusted, can be promoted to trusted, cannot be denied
- High: Device is always trusted

A low or medium trust level is appropriate for Access or untrusted networks (realms). In contrast, a high trust level is appropriate only for Core or trusted networks (realms).

Promotion Criteria Examples

- SIP: 200OK received for either Register or Invite method
- MGCP: 200 received for either RSIP or CRCX

Demotion Criteria Examples

Exceeding any of the following thresholds:

- **invalid-signal-threshold**: maximum number of non-compliant signaling packets acceptable
- **maximum-signal-threshold**: maximum number of signaling packets acceptable while an endpoint is classified as trusted
- **untrusted-signaling-threshold**: maximum number of signaling packets while an endpoint is classified as untrusted

These thresholds are all measured in the configurable system wide tolerance-window (default 30s)

If an endpoint crosses one of these thresholds then a deny ACL is written to the CAM, and checked by the Network Processors (NP) upon receipt of a packet from the denied endpoint. The endpoint is denied for a configurable period of time.

The Whole NAT device demotion Criteria Examples

Exceeding any of the following thrasholds:

- **max-endpoints-per-nat**: maximum number of end points hebind a NAT at a realm level
- **nat-invalid-message-threahold**: Maximum number of "invalid messages" that all end points behind a NAT

Another related configuration is **wait-time-for-invalid-register**, the time period for which SBC will wait before counting the absence of the REGISTER message as an invalid message.

The goal of the DdoS protection tools detailed above is to assess and plan for a configuration that allows service to continue whether the SBC is under malicious attack or a non-malicious attack such as a recovery from a Softswitch outage or registration flood from endpoints. This involves allowing enough untrusted traffic such that endpoints can over time register successfully yet constraining all queues sufficiently to protect SBC resources (i.e. core CPU threshold).

Furthermore, the SIP Registration Overload Protection (SROP) feature is used to protect the SBC against mass endpoint avalanche restarts. The following sip-config options are recommended to be configured:

- **cache-challenges** and **reg-overload-protect**: The SBC will temporarily promote the endpoint to trusted level after the registrar challenges the REGISTER message with a 401/407 response.
- **max-register-forward**: Limit rate of REGISTERs to forward to the registrar. Set to 75% of max registers/sec the registrar can handle.
- **max-register-refresh**: Limit rate of REGISTER refreshes from endpoints. Set to 150% of number of endpoints divided by the refresh interval.
- **register-grace-timer**: Grace period in seconds before a cached registration is deleted from the SBC after expiration. Recommended to set this value to 32.
- **reject-register=refresh**: Lets the REGISTER in, but will check the load limit if there is not a cached registration that it can use for a response.

For the session-agent representing the core Registrar, the max-register-burst-rate should be configured to throttle REGISTER messages sent to it. In addition, session-constraints should be enabled with rate-constraints configured to limit the rate of REGISTER messages coming into the core network. Session-constraints are applied on the Access sip-interface or realm. In the sip-config parameter, extra-method-stats must be enabled for rate-constraints to take effect.

Please contact your Oracle Systems Engineer to discuss planning for DdoS protection configuration and deployment. Basic DdoS configuration is found in Appendix C: DdoS Prevention for Peering Environments and Appendix D: DdoS Prevention for Access or Hybrid Environments. Configuration is detailed in Section 5 “SIP Signaling Services” and Section 15 “Security” of the *ACLI Configuration Guide*.

Net-SAFE Architecture: Topology Hiding & SIP Manipulation

Topology hiding is primarily performed by the SBC’s Back-to-Back User Agent (B2BUA) function. Use of the SIP-NAT configuration object or the flexible SIP Manipulation feature provide capabilities to dynamically alter any identifying information pertaining to a customer core network in signaling messages.

SIP Manipulation rules allow the customer to check for a value in any element of any SIP message and take action if a rule matches. Actions include changing a value, deleting an element or parameter, completing a header, or adding a completely new header to the message. Requests can be rejected, and MIME types and bodies can also be manipulated. To provide further topology hiding in the SDP portion of a SIP message, the customer should enable SDP anonymization.

An example of the SIP-NAT feature used for topology hiding is available in the document “520-0005-04 BCP – SIP Access Configuration.” An example of the SIP Manipulation feature used for topology hiding is presented in Section 7 “HMR Bridging” of “520-0038-01 BCP SIP Peering Configuration”.

Configuration of SIP HMR (Header Manipulation Rules) is detailed in Section 5 “SIP Signaling Services” of the *ACLI Configuration Guide*. Configuration of SDP anonymization is detailed in Section 15 “Security” of the *ACLI Configuration Guide*.

Security Specific Licenses

IDS Reporting

The SBC supports a wide range of intrusion detection and protection capabilities for vulnerability and attack profiles identified to date. The IDS reporting feature provides more detailed reporting of intrusions the system detects. It is useful for enterprise customers’ requirement to report on intrusions and suspicious behavior that it currently monitors. This feature requires the IDS Reporting license, which is included in new purchases but was not in some older deployments. The “IDS Advanced” feature should be present in the output of the `show features` command.

See Appendix F: Intrusion Detection System for a detailed description of the functionality enabled. Configuration is also detailed in Section 15 “Security” of the *ACLI Configuration Guide*.

FIPS Feature (Optional)

FIPS is supported on the enterprise software release ECZ750 on the AP1100, VME and AP3900 platforms. See “Oracle Enterprise Session Border Controller FIPS Compliance Guide”.

Administrative Security Features (Optional)

See “Oracle Enterprise Session Border Controller Administrative Security Guide”

Configuring Monitoring and Performance Management Features

SNMP

Simple Network Management Protocol (SNMP) is supported on the SBC Wancom0 management interface for polling and traps. To secure your SNMP interface, it is recommended to use a community name other than the standard “public”. Sufficiently obscure community names should adhere to the customer’s corporate naming policies. Further, the list of configured SNMP polling servers and trap receivers must be restricted to only those authorized (via SBC configuration) to manage the SBC. All management stations used for SNMP access should have a permit ACL configured.

Further detail on SNMP traps and MIBS that should be examined can be found in Appendix H: Configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

RADIUS Accounting

The SBC Wancom0 management interface uses RADIUS requests to send accounting and monitoring data to remote RADIUS servers. For reliability, the SBC supports the configuration of multiple RADIUS servers deployed in a number of HA schemes: hunt, failover, round robin, fastest round trip time (RTT) and fewest pending.

The most appropriate scheme according to customer’s corporate policies should be chosen. It is recommended that at least two RADIUS servers be deployed. The secret shared between the SBC and the RADIUS server should be configured to be suitably obscure according to the customer’s corporate naming policies. All management stations used for accounting monitoring services should have a permit ACL configured.

Configuration is detailed in the *ACLI Accounting Guide*.

HDR over SFTP

The Historical Data Recording (HDR) feature allows the SBC to record data in comma-separated files and periodically sends them to a remote file server. For added security, transfer the HDR record files using SFTP. Note that public key authentication is not available for this feature so the SBC uses password authentication. All management stations used for SFTP access should have a permit ACL configured.

Configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

Syslog

The syslog service should be used for sending system events from the SBC to a Security Event & Incident Monitoring (SEIM) platform or to another operations monitoring platform. The information sent via syslog is also contained locally on the SBC in the acmelog file.

See Appendix I: for examples of important syslog messages to monitor. The default syslog log level is WARNING.

Configuration is detailed in Section 3 “Syslog and Process Logs” of the *ACLI Configuration Guide*.

Configuring AAA Integration

SSH RADIUS Authentication

The SBC management interface sends RADIUS requests containing login authentication and authorization data to remote RADIUS servers.

The SBC supports the use of the Cisco Systems Inc.™ “Cisco-AVPair” vendor specific attribute (VSA). The Vendor-ID is 1 and the Vendor-Type is 9. This attribute allows for successful administrator login to servers that do not support the Oracle authorization VSA. While using RADIUS-based authentication, the SBC authorizes you to enter Superuser mode locally even when your RADIUS server does not return the ACME_USER_CLASS VSA or the Cisco-AVPair VSA.

All management stations used for SSH access should have a permit ACL configured. An ACL should also be configured to allow RADIUS traffic to the RADIUS server.

For more information, see Section 4 “System Management” of the *Maintenance and Troubleshooting Guide*.

TACACS+

TACACS+ is a protocol that was originally developed by Cisco Systems. It provides functions for authentication, authorization, and encryption of the administrative traffic. Unlike RADIUS, it separates authentication and authorization functions. The SBC acts as a TACACS+ client.

The Net-Net SBC uses TACACS+ services to provide administrative authorization. With TACACS+ authorization enabled, each individual ACLI command issued by an admin user is authorized by the TACACS+ authorization service. The SBC replicates each ACLI command in its entirety, sends the command string to the authorization service, and suspends command execution until it receives an authorization response. If TACACS+ grants authorization, the pending command is executed; if authorization is not granted, the SBC does not execute the ACLI command, and displays an appropriate error message.

All management stations used for SSH access should have a permit ACL configured. An ACL should also be configured to allow TACACS+ traffic to the Network Access Server. TACACS+ is disabled by default.

Refer to “TACACS+ AAA” in Section 2 “Getting Started” of the *ACLI Configuration Guide*.

Configuring Signaling & Media Interface Security

Securing the service interfaces is perhaps the most important consideration as they are typically deployed in public unsecure networks and these interfaces are usually the demarcation or access point to the core network infrastructure.

Signaling/Media Management Functions

The phy-card is intended for signaling and media traffic only. The SBC disables ICMP, telnet, SNMP and FTP on signaling/media interfaces by default. It is not recommended that any of these protocols be enabled on a service interface for any length of time beyond that required for troubleshooting purposes.

Configuration is detailed in Section 3 “System Configuration” of the *ACLI Configuration Guide*.

SIP Interface Security

As well as the layer3 ACLs described earlier, the SBC provides layer5 SIP protection to its signaling interfaces. By default, the SBC sip-interface > sip-port parameter will allow and route signaling from any device.

For Access/untrusted networks, the sip-interface > sip-port > allow-anonymous setting should be configured to one of the following values:

- **registered**: This is the most widely deployed setting, only allowing non-REGISTER SIP requests from either a defined session-agent or a previously registered device. (All REGISTER requests are processed.)
- **realm-prefix**: Allows SIP requests only from defined session-agents or previously registered endpoints. Only REGISTER requests from endpoints within the configured realm-prefix (subnet) are allowed.

Although this feature will deny service to a malicious user, the SIP daemon and hence the core CPU is utilized to parse and process each and every request; therefore, this feature is recommended to be deployed in conjunction with the Net-SAFE architecture.

SIP-interfaces communicating with non-registering devices (e.g. peering partner SBCs or core devices such as softswitches) should have the allow-anonymous setting configured for agents-only.

An Enforcement Profile should be configured with the list of allowable SIP methods. It is recommended that only the minimum set of SIP methods necessary for the customer application be configured. More protection is configurable in Access scenarios where SIP endpoints are sending SUBSCRIBE dialogs. The rate of these messages can be limited per user.

Session constraints should be applied to the sip-interface to limit the max-sessions, max-burst-rate, max-sustain-rate, and rate constraints for individual method types. Further information is found in Section 5.3 “Constraint Limiting” of “520-0013-05 TECH NOTE Theory of the Session-agent.”

The SBC's default SIP routing behavior is to comply with Route headers as received. Thus there is a security "hole" wherein a trusted device could construct a Route header and use the SBC as a reflector for signaling to another known device. Furthermore, the SBC will also use the Request-URI to route traffic even if there's no matching local-policy. This is mitigated by using techniques such as stripping Route headers on ingress (proceed with caution) and configuring "null routes" with 0.0.0.0 as the next-hop.

Configuration is detailed in Section 5 "SIP Signaling Services" and Section 10 "Session Routing and Load Balancing" of the *ACLI Configuration Guide*.

Service ACLs

ACLs on service ports provide more functions than the basic permit and deny operations that are provided by the ACLs on management ports. Service ACLs have effects on traffic management through average rate limitations, trust level, and signaling thresholds similar to those specified on a realm.

To prevent misunderstanding these traffic management settings, keep in mind these general rules:

- Define an ACL for all peering partners and all core systems to which traffic will be routed. The ACL is used to permit trusted hosts, deny untrusted hosts, and guarantee bandwidth in peak periods.
- The minimum-reserved-bandwidth setting does not permanently reserve bandwidth. It will only be used in peak periods to prioritize traffic. Set the minimum-reserved-bandwidth to the maximum signaling bandwidth capable for the system. If more than one core device is used, divide the bandwidth number equally. The number here is not really bandwidth but a priority metric.
- Hosts with a trust levels of high will never be demoted or blacklisted. However, if an invalid-signal-threshold of one is configured on the ACL, a syslog event will be written which might help detect attempted abuse.
- The trust level specified on the ACL should match the trust level on the realm from which it will communicate. Trust level mismatches can have unintended consequences such as permitting traffic that is intended to be denied. Refer to the scenario below that illustrates how this can be problematic.

In this scenario there is a trusted core PBX on a private network, and two PBXs on an external public network. The trust level on the ACL applied to the external interface and the trust level on the external realm are depicted in the tables below, along with what will happen to traffic sent from a source IP of ".100" or ".111". In *Table 1: IP .111 permitted in ACL* the effects of having the 192.168.1.111 address permitted are depicted. Table 2 shows the opposite, when the 192.168.1.111 address is denied. Note what access the 192.168.1.100 address has based on the trust level of the realm and ACL.

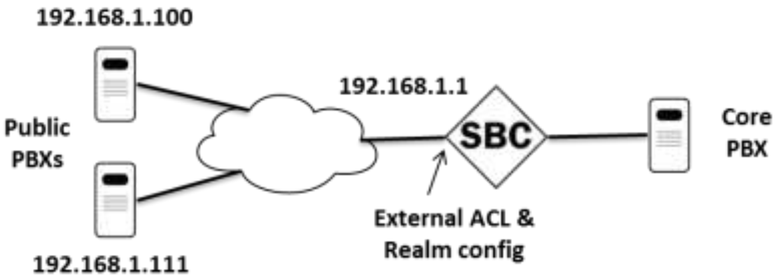


Figure 2: ACL and Realm scenario

Table 1: IP .111 permitted in ACL

Realm Trust Level	ACL Trust Level	src:100	src:111
None	none	Permit	Permit
None	low	Deny	Permit
None	medium	Deny	Permit
None	high	Deny	Permit
Low	none	Permit	Permit
Low	low	Deny	Permit
Low	medium	Permit	Permit
Low	high	Permit	Permit
Medium	none	Permit	Permit
Medium	low	Permit	Permit
Medium	medium	Deny	Permit
Medium	high	Permit	Permit
High	none	Permit	Permit
High	low	Permit	Permit
High	medium	Permit	Permit
High	high	Deny	Permit

Table 2: IP .111 denied in ACL

Realm Trust Level	ACL Trust Level	src:100	src:111
None	none	Deny	Deny
None	low	Deny	Deny
None	medium	Deny	Deny
None	high	Deny	Deny
Low	none	Permit	Deny
Low	low	Permit	Deny
Low	medium	Permit	Deny
Low	high	Permit	Deny
Medium	none	Permit	Deny
Medium	low	Permit	Deny
Medium	medium	Permit	Deny
Medium	high	Permit	Deny
High	none	Permit	Deny
High	low	Permit	Deny
High	medium	Permit	Deny
High	high	Permit	Deny

TLS for SIP

Transport Layer Security (TLS) provides end-to-end authentication and encryption of SIP signaling. TLS protects against eavesdropping, tampering, forgery, and potential theft of service. For this reason, TLS should be used wherever possible.

The SBC supports mutual-authentication within a TLS profile. Although disabled by default, this should be enabled when endpoints support it.

All supported products have TLSv1 capabilities, and newer releases (SC7.2 and above) also support TLS1.2.

The SBC supports three TLS Exchange / Authentication models:

- Basic – The client authenticates the SBC certificate by using the CA public key, and checks expiration, common name, and ciphers supported. This provides confidentiality and integrity through encryption but does not establish the identity of the endpoint. Credential cracking is still possible, and the move to TLS (based on TCP) may make port exhaustion DoS a bit easier for an attacker.
- Mutual – A step is added in which the client certificate is sent to the SBC for verification. Single or individual client certificates can be used. This model has the same characteristics of the basic model with the advantage of verifying that the client is likely trusted since an issued certificate is present. If a single certificate is used for all clients then theft or compromise of an endpoint may allow access to an attacker. Individual certificates are more secure but require more administrative effort to issue and manage.
- Mutual with certificate revocation – Certificate revocation for individual clients is possible, which guarantees only expired or revoked clients are refused access. An external Online Certificate Status Protocol (OCSP) server is required to check against the Certificate Revocation List.

Note: The SBC does not support local CRLs due to onboard storage limitations.

Some other key information regarding TLS includes:

- Certificates installed on the SBC must be derived from a local Certificate Signing Request in PKCS-10 PEM/Base 64 format. Certificates cannot be installed without a CSR.
- Certificate key lengths can go up to 2048 bits, with 4096 possible with SSM3 (currently on supported on 6300) after SC7.2.
- Certificates are currently signed with a SHA-1 hash by default. It is recommended to sign with SHA-2 or above.
- If site-to-site failover is required, the main site's fully qualified domain name (FQDN) and the FQDN for any alternate site should be specified as alternate-names in the certificate record prior to CSR generation.
- TLS session caching (tls-global element) allows a previously authenticated user to reuse a previous session so authentication is sped up. This may help reduce time to recovery due to outages, though it is best suited for environments where user IP does not vary significantly.
  The default cipher list when creating a tls-profile is currently "ALL". This includes potentially insecure ciphers and a "NONE" cipher which does not provide encryption – only authentication. When configuring a tls-profile, use the following ciphers for maximum security:

  For release SC7.2 and above:
  - TLS _DHE_RSA_WITH_AES_256_GCM_SHA384

- TLS _DHE_RSA_WITH_AES_128_GCM_SHA256
- TLS _DHE_RSA_WITH_AES_256_SHA256
- TLS _DHE_RSA_WITH_AES_128_SHA256

For releases before SC7.2:

- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA

Since TLS is based on TCP, TCP DoS protections should be configured to limit the number of connections per source IP and per sip-interface. Consider these settings in your environment:

- sip-config > inactive-dynamic-conn – Defines global timer for tearing down idle TCP/TLS connections where no SIP data has been sent. The timer used is twice as long for TLS.
- sip-interface settings to limit connections:
    - untrusted-conn-timeout – Closes socket if untrusted entity does not become trusted, such as if the register didn't complete.
    - inactive-conn-timeout – Tears down idle TCP/TLS connections when no further data is being sent, such as if a trusted host sends an INVITE but nothing else.
    - max-incoming-conns – Set to max incoming sessions you want the SIP interface to host plus overhead for setup / teardown (depends on call rate).
    - per-src-ip-max-incoming-conns – Usually 1 or 2 but affected by NAT use and application.

Configuration is detailed in Section 15 "Security" of the *ACLI Configuration Guide*.

OCSP

The Online Certificate Status Protocol (OCSP) is defined in RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP*. The protocol enables users to determine the revocation state of a specific certificate, and may provide a more efficient source of revocation information than is possible with Certificate Revocation Lists (CRL).

The protocol specifies the data exchanged between an OCSP client (such as the Net-Net SBC) and an OCSP responder, the Certification Authority (CA), or its delegate, that issued the target certificate. An OCSP client issues a request to an OCSP responder and suspends acceptance of the certificate in question until the responder replies with a certificate status.

Certificate status is reported as

- good
- revoked
- unknown

OCSP can be especially useful in environments where individual certificates have been issued to a single user or user device. Certificates for devices that are stolen or misplaced can be revoked, so even if valid credentials are known the device will not be able to connect.

Configuration is detailed in Section 15 "Security" of the *ACLI Configuration Guide*.

SRTP

Many customers require the ability to encrypt and authenticate the content and signaling of their real time communications sessions. The SBC supports the Secure Real-Time Transport Protocol (SRTP). Authentication provides assurance that packets are from the purported source, and that the packets have not been tampered with during transmission. Encryption provides assurance that the call content and associated signaling has remained private during transmission.

SRTP requires an Ipsec NIU other than 1100 and 3900. 1100 and 3900 support software based SRTP.

RTP and RTCP traffic are encrypted as described in RFC 3711, *The Secure Real-time Transport Protocol (SRTP)*. The negotiation and establishment of keys and other cryptographic materials that support SRTP is described in RFC 4568, *Session Description Protocol (SDP) Security Description for Media Streams*. Cryptographic parameters are established with only a single message or in single round-trip exchange using the offer/answer model defined in RFC 3264, *An Offer/Answer Model with the Session Description Protocol (SDP)*.

For further information and configuration settings, refer to Appendix L: and to Section 15 “Security” of the *ACLI Configuration Guide*.

Ipsec for SIP

Ipsec provides another mechanism for encrypting and securing SIP signaling services through the use of IPSec NIU for 4500, 4600, 6100, 6300.

Security Associations and Security Policies allow for flexibility in defining local and remote IP address, ports and subnet masks. These should be defined to only allow Ipsec communications between authorized gateways or hosts and the SBC.

SBC supports Ipsec IKEv1 to create Ipsec tunnels dynamically. This is based on the Internet Key Exchange (IKE) Protocol as defined in RFC 2409, *Internet Key Exchange*, and for the Dead Peer Detection (DPD) protocol as defined in RFC 3706, *A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers*.

The following IKEv1 functionality is supported:

- IKE pre-shared secret support
- IKE/ISAKMP Main Mode support
- IKE/ISAKMP Aggressive Mode support
- Phase 2 Quick Mode support

In addition, with IKEv1 enabled, the SBC can support Ipsec between itself and an endpoint behind a NAT device.

Configuration is detailed in Section 15 “Security” of the *ACLI Configuration Guide*.

Call Admission Control (CAC)

Call Admission Controls are used to limit the number of allowed resources such as bandwidth or sessions to abide by customer Service Level Agreements (SLA) and to avoid abuse. It is recommended that wherever possible these features be enabled:

- Bandwidth (codec) based – for bandwidth CAC settings see “Media Profiles”
- SIP Per-User CAC
- Session Capacity
- Session Rate (sustained and burst)

Bandwidth CAC

Bandwidth based CAC can be implemented though a media profile on the realm level. Media profiles specify or limit the range of the codecs, bandwidth, and packet rate used. Configuration is detailed in Section 4 “Realms and Nested Realms” of the *ACLI Configuration Guide*.

SIP Per-User CAC

When SIP per-user CAC is enabled the SBC changes its default behavior so that it will only allow the configured number of calls or total bandwidth to and from each individual user in a particular realm. CAC can be applied to an individual Address of Record (AoR) or IP address. Tracking based on IP address can cause complications if a NAT is involved, so the use of a nat-trust-threshold may be required to set the maximum number of untrusted endpoints behind NAT devices. This also enables the ability of the SBC to track endpoints based on both IP and the TCP or UDP port in use.

Configuration is detailed in Section 5 “SIP Signaling Services” of the *ACLI Configuration Guide*.

Session Capacity and Session Rate using Constraints

Constraints are a CAC method that limits messaging based on session count and rate. Constraints can be applied to SIP interfaces or realms. It is recommended that constraints are used on all external interfaces and core session-agents.

A session-agent can be configured for max-outbound-sessions, max-sessions, max-burst-rate and max-sustain-rate.

Max-outbound-sessions and max-sessions give the max number of allowed concurrent sessions. These should be set to match what should be sent to an upstream session-agent (for example a service provider) or accepted into a core session-agent.

The session-agent’s max-burst-rate and max-sustain-rate are used to throttle the calls per second (CPS) of traffic sent to and by that session-agent. Each of these parameters has its own configurable window by which the statistics are gauged for constraint exceptions.

For the sustained-rate, the average is calculated over the previous window (equal to the sustained-rate-window) and current “window fragment.” The “window fragment” will be between 0 and the configured sustained-rate-window upon receipt of an Invite. Once the window fragment increments and reaches the sustained-rate-window, this rotates and becomes the “previous window” – and a new window fragment begins at 0. At this point all calculations are recalibrated accordingly.

For example, consider the case where the sustain-rate is set to 15 and the sustain-rate-window is set to 10 seconds. When an invite is received the SD will add the amount of Invites received in the current window fragment and the previous window and divide by the number of seconds to get the average for that period. This average is then compared to the 15 CPS derived from the sustain-rate and the sustain-rate window. If the session-agent per the previous and current window is above 15 CPS when the Invite is received, the Invite will be rejected.

The max-burst-rate and burst-rate-window interact by limiting the CPS rate for a burst of traffic over the window. Using the example below, with a max-burst-rate of 20 and a burst-rate-window of 10, the SD will permit 200 sessions within the first 10 seconds and then reject all new sessions until it exits constraint mode.

Burst rate is much easier to understand and configure, so it is preferable over sustain rate.

As for a session-agent in constraint, it does not come out of constraint mode when traffic drops below its constraint thresholds; it comes out of constraint mode after 60 seconds, unless a configured time-to-resume value dictates otherwise. Even though the session-agent is out of the constraint mode after time-to-resume seconds “show sipd agent” will show it back into In-Service mode only if the traffic flows to or from that session-agent. On exceeding its constraint the session-agent is marked “C”.

Core registrars should have a max registration burst rate configured to the maximum rate (or just below) what the registrar can handle.

Configuration is detailed in Section 5 “SIP Signaling Services” and Section 12 “Admission Control and Quality of Service Reporting” of the *ACLI Configuration Guide*.

Media Policing

Media policing controls the throughput of individual session media flows (RTP and RTCP) in the SBC. It also allows the SBC to police static flows. It is recommended to enable media policing to protect against RTP media flooding and bandwidth piracy.

For each individual codec being used in sessions, a media-profile must be created with average-rate-limit thresholds configured.

Configuration is detailed in Section 15 “Security” of the *ACLI Configuration Guide*.

DoS/DdoS Prevention

DoS and DdoS settings can protect against malicious and non-malicious SIP flooding attacks from untrusted sources without adversely affecting service to trusted peers.

Attacks can be prevented through configuration of Access Control Lists, appropriately sized traffic queues, and trust level settings that will limit or blacklist endpoints that become abusive.

Configuration of these parameters will differ based upon the configuration model used – peering, access, or hybrid. Refer to either Appendix C: DdoS Prevention for Peering Environments or Appendix D: DdoS Prevention for Access or Hybrid Environments, depending on the architectural model implemented.

Note that a truly comprehensive and effective DdoS prevention design requires analysis of traffic patterns, SIP message contents and performance characteristics of all peer devices to provide message thresholds, CAC, and traffic policing settings. Please contact your Oracle Sales representative for information on professional services designed to implement customized DdoS settings.

Attack Tool Prevention

Many SIP scanning and attack tools employed by fraudsters can be prevented through employment of restrictive signaling thresholds and trust levels – the same ones used for DdoS protection. However, some deployments do not allow for this without impacting legitimate traffic. Attackers may also use commonly available tools that have identifiable signaling patterns. In these cases, additional attack tool identification and prevention may limit or prevent an attack from being successful.

Oracle recommends that any deployment with internet-connected interfaces comply with the settings described in Appendix E: Mitigating SIP Attacks.

Lawful Interception

The SBC supports a Lawful Intercept capability as mandated by national laws in various countries. Multiple interface types are supported. The feature purchasing and documentation are controlled, and the LI capability has to be enabled with the installation of a license key. It must then be configured to communicate with a server that provides the authorization ‘tickets’ to enable recording. After installation of the LI license, a separate administrative user dedicated for LI configuration “li-admin” becomes active.

Part 4: Appendices

Appendix A: Secure Deployment Checklist

The following security checklist includes guidelines that help secure your system:

1. Do NOT connect your system to any untrusted networks, especially the Internet, until all protections have been configured. Customers have reported systems under configuration compromised in minutes due to incomplete configurations.
2. Harden the management environment.
    a. Install HA connections between units over a direct cable vs. a network.
    b. Make sure all equipment is in locked cabinets or at least in a secure room.
    c. Configure console timeouts.
    d. Ensure that the wancom0 management port is connected to a private management LAN with an IP address that is not Internet routable.
    e. Set strong passwords for all default accounts prior to configuration.
    f. Disable telnet and FTP if they are enabled.
    g. Configure system ACLs to limit management traffic to users that really need access.
    h. If implementing SNMP, change the default community string and follow the SNMP configuration recommendations in Appendix H:
    i. Use strong ciphers for HTTPS web management connection.
3. Practice the principle of least privilege.
    a. Carefully consider who has access to the admin password.
    b. Implement RADIUS or TACACS+ authentication if available.
4. Restrict network access.
    a. Use services ACLs where possible.
    b. Refrain from configuring host-in-path addresses.
    c. Ensure that users coming from an untrusted network have to register prior to providing service.
    d. Implement DoS and CAC protections.
    e. Mitigate known fraud schemes by implementing sipShield or HMRs.
    f. Use strong ciphers for any TLS connections.
    g. Enable OCSP and mutual authentication if possible for TLS connections.
5. Monitor the system for unusual events.
    a. Configure the SNMP trap receiver and syslog receiver.
    b. Send either CDRs or RADIUS accounting records to a fraud management system or implement a solution that can actively monitor SIP signaling.

Appendix B: Port Matrix

Ethernet	Ports	Protocol	Service	Optional	Configurable Port	Default Port State	Server or Client	Description
Wancom0	21	TCP	FTP	Yes	No	Open	Server	FTP file transfer
Wancom0	21	TCP	FTP	Yes	Yes	Closed	Client	FTP push receiver
Wancom0	22	TCP	SSH / SFTP	Yes	No	Open	Server	SSH for ACLI admin
Wancom0	23	TCP	Telnet	Yes	No	Open	Server	Telnet for ACLI admin
Wancom0	49	TCP	TACACS+	Yes	Yes	Closed	Client	TACACS+ AAA
Wancom0	80	TCP	HTTP	Yes	Yes	Closed	Server	HTTP SIP monitoring and tracing or provisioning GUI
Wancom0	123	UDP	NTP	Yes	No	Closed	Client	NTP time update requests
Wancom0	161	UDP	SNMP	Yes	No	Closed	Client	SNMP traps
Wancom0	162	UDP	SNMP	Yes	No	Closed	Server	SNMP MIB retrieval
Wancom0	443	TCP	TLS/HTTPS	Yes	Yes	Closed	Server	HTTP SIP monitoring and tracing or provisioning GUI
Wancom0	514	UDP	Syslog	Yes	Yes	Closed	Client	Syslog message feed
Wancom0	1812	UDP	Radius	Yes	Yes	Closed	Client	RADIUS AAA
Wancom0	1813	UDP	Radius	Yes	Yes	Closed	Client	RADIUS Accounting
Wancom0	3000	TCP	ACP	Yes	No	Open	Server	Acme Control Protocol for GUI
Wancom0	3001	TCP	ACP	Yes	No	Open	Server	Acme Control Protocol for GUI
Wancom0	ANY	UDP	Process Log	Yes	Yes (any)	Closed	Client	Internal process log feed
Wancom0	n/a	1	ICMP Echo Reply	Yes	No	Open	Server	Echo Request (Ping) used by SIP trunk between ASM and its application server (CM)
Wancom1 & 2	22	TCP	SSH / SFTP	Yes	No	Closed	Server	
Wancom1 & 2	1987	UDP	HA CFG	Yes	Yes	Closed	Both	Primary is server, client is secondary
Wancom1 & 2	9090	UDP	HA BERPD	Yes	Yes	Closed	Both	Primary is server, client is secondary
Services Ports	n/a	50	ESP	Yes	No	Closed	Server	
Services Ports	n/a	51	AH	Yes	No	Closed	Server	
Services Ports	n/a	1	ICMP Echo Reply	Yes		Closed	Server	
Services Ports	21	TCP	FTP	Yes	No	Closed	Server	
Services Ports	22	TCP	SSH / SFTP	Yes	No	Closed	Server	
Services Ports	23	TCP	Telnet	Yes	No	Closed	Server	
Services Ports	53	UDP	DNS	Yes	No	Closed	Client	

Ethernet	Ports	Protocol	Service	Optional	Configurable Port	Default Port State	Server or Client	Description
Services Ports	80	TCP	OCSP	Yes	Yes	Closed	Client	
Services Ports	80	TCP	COPS, A-COPS, DIAMETER	Yes	Yes	Closed	Client	Policy server
Services Ports	161	UDP	SNMP	Yes	No	Closed	Client	SNMP traps
Services Ports	162	UDP	SNMP	Yes	No	Closed	Server	SNMP MIB retrieval
Services Ports	500	UDP	ISAKMP	Yes	Yes	Closed	Server	
Services Ports	4500	UDP	ISAKMP/NAT	Yes	Yes	Closed	Server	
Services Ports	1986	TCP	MCGP HA	Yes	Yes	Closed	Server	
Services Ports	1988	TCP	MCGP SIP Checkpoint	Yes	Yes	Closed	Server	
Services Ports	1719	TCP	H.323 RAS	Yes	Yes	Closed	Server	
Services Ports	1720	TCP	H.323 Q931	Yes	Yes	Closed	Server	Set, dynamic from 0 up
Services Ports	1994	TCP	Ipsec	Yes	Yes	Closed	Server	Ipsec sync messages
Services Ports	2427	TCP/UDP	MGCP	Yes	Yes	Closed	Client	MGCP signaling
Services Ports	2727	TCP/UDP	MGCP	Yes	Yes	Closed	Server	MGCP signaling
Services Ports	3478	TCP/UDP	STUN	Yes	Yes	Closed	Both	
Services Ports	3479	TCP/UDP	STUN	Yes	Yes	Closed	Both	
Services Ports	3868	TCP/SCTP	Diameter	Yes	Yes	Closed	Both	HSS Connection, client port dynamic
Services Ports	5060	TCP/UDP/SCTP	SIP	Yes	Yes	Closed	Both	SIP, client port dynamic
Services Ports	5061	TCP	SIP TLS	Yes	Yes	Closed	Both	SIP over TLS carried by TCP
Services Ports	1025-65535	TCP/UDP	RTP/SRTP	Yes	Yes	Closed	Both	Media traffic
Services Ports	1025-65535	TCP	IMS AKA	Yes	Yes	Closed	Both	IMS AKA protected server port
Services Ports	1025-65535	TCP	IMS AKA	Yes	Yes	Closed	Client	IMS AKA protected client port

Appendix C: DdoS Prevention for Peering Environments

Configuration Models:

The settings outlined in this appendix apply to the following configuration models:

PBRB	Policy Based Realm Bridging Model
SNB	SIP NAT Bridge Model
SSNHTN	Single SIP NAT Hosted in Trusted Network Model

Supported platforms

Platform	CAM/Flow Table	Memory
AP4500 (CPU 2)	256K (CAM)	8G
AP6300	1000000 **Flow Table)**	16G
AP4600	1000000 **(Flow Table)**	16G
AP6100	1000000 **(Flow Table)**	16G

Configuration Parameters

The following sections will discuss those DdoS prevention parameters pertinent to the scope of this appendix. These parameters are found in three configuration areas: Media Manager, Realm Configuration, and SIP Interface.

DdoS Configuration Parameter Descriptions

Media Manager

The following `media-manager` parameters have been calculated for each configuration model.

Parameter	Value
`max-untrusted-signaling`	Maximum percentage of the allocated max-signaling-bandwidth for untrusted traffic (%)
`min-untrusted-signaling`	Minimum percentage of the allocated max-signaling-bandwidth for untrusted traffic (%)
`max-signaling-bandwidth`	The maximum bandwidth that the SBC can withstand (bytes/sec)

Typically, these parameters are not applied in peering configuration as the source of peer traffic is assumed to be trusted. However, because these parameters values are set at default ‘0’, with the purpose of maximizing the CPU resource for trusted traffic, it is suggested to minimize these values to ‘1’ so that to guarantee optimal performance on trusted peer traffic.

The recommended values for these media-manager parameters for each test scenario are listed later by system model.

Max-signaling-bandwidth is not present /supported with Software Datapath (VM, COTS, 1100, 3900).

The following are Media Manager parameters that have platform specific defaults (not configurable). “show acl info” ACLI command shows the details from each platform.

Parameter
`min-media-allocation`
`min-trusted-allocation`
`Deny-allocation`

For this appendix, these defaults will be used and are indicated in the platform results later by system model.

Realm Configuration

The following `realm-config` parameters are used in the basic DdoS configuration. Only the **bold** values are changes from the default configuration:

Parameter	Peer Realm	Core Realm
`access-control-trust-level`	**`high`**	**`High`**
`invalid-signal-threshold`	**`0`**	**`0`**
`average-rate-limit`	**`0`**	**`0`**
`maximum-signal-threshold`	**`0`**	**`0`**
`untrusted-signal-threshold`	**`0`**	**`0`**

SIP Interface

The following `sip-interface->sip-ports` parameter SHOULD be used for Peering environments.

Setting “allow-anonymous” to agents-only will allow the SBC to reject requests sent by any IP which has not yet been defined as a “Session-Agent” in the SBC configuration. In Peering configurations, the customer SHOULD define each IP of a peer’s device as a “session-agent” for optimal purpose.

Parameter	Peer Realm	Core Realm
`allow-anonymous`	**`agents-only`**	**`All`**

Although it is not recommended, but it is still possible to allow packets from an IP that has not yet defined as a Session-Agent, by setting “allow-anonymous” to “all”. In this setup, the SBC will simply allow the request under DdoS threshold opposed to rejecting it with a 403 Forbidden response.

Session Agent and Access-Control

Any peering signaling device SHOULD be defined as a Session-Agent in SBC configuration. Further, for proper DdoS prevention, it requires explicitly configuring one access control per address of each Session-Agent address or other address (that has not yet been defined as a session-agent).

*Session-router->**session-agent***

Parameter	Realm
`realm-id`	**`peer`**

`constraints`	***`enabled`***
	`[`*`optional`*`]`

max-sessions	***X***
max-burst-rate	***Y***
max-sustain-rate	***Z***
time-to-resume	***60 sec***
burst-rate-window	***1 sec***
sustain-rate-window	***30 sec***

There is no demotion event when access-control-trust-level in the realm-config is set “high” as packets from the trusted peer endpoint are always allocated in the trusted queue for processing. It becomes a concern when there is excessive amount of SIP traffic sent by a customer which is beyond the SLA. Session constraints under session-agent can be deployed to further mitigate this problem. Listed above are a small set of constraints to provide basic level of call admission control in order to ensure that a session-agent’s capacity is not exceeded, or the SBC will reject the service with 503 Exceed Constraints. Please be advised that these settings are only optional. Customers may consider them when deploying their service in a Peering environment with or without DdoS configuration.

- max-sessions –X

  Define a maximum number of sessions (inbound and outbound) allowed by the session agent. Once the session limit is reached, the SBC will start rejecting new service with 503 Exceed Constraints until the number of seconds in time-to-resume has elapsed.

- max-burst-rate –Y

  Define a number to set the maximum rate of call (per second) this session agent will allow. Once the rate limit is reached, the SBC will start rejecting new service with 503 Exceed Constraints until the number of seconds in time-to-resume has elapsed.

- max-sustain-rate – Z

  In general, set this to the average call rate (per second) which that SA can sustain. Once the average rate limit calculated in
  (Calls made in current + previous window) / Delta (current second – start of previous window), exceeds the limit Z , the SBC will be start rejecting new service with 503 Exceed Constraints until the number of seconds in time-to-resume has elapsed.

*Session-router->**access-control***

Parameter	Realm	Realm
realm-id	**peer**	**Core**
source-address	n.n.n.n/[mask bit is optional] *(peer SA IP, or non-SA IP)*	[m.m.m.m]/ [mask bit is optional] *(core SA IP or non-SA IP)*
application-protocol	**SIP**	**SIP**
transport	**ALL**	**ALL**
access	**permit**	**Permit**
trust-level	**high**	**High**
minimum-reserved-bandwidth	0	0
invalid-signal-threshold	0	0
maximum-signal-threshold	0	0

untrusted-signal-threshold	0	0
deny-period	30	30

In core realm, it is recommended to configure an access-control on per session-agent basis instead of putting it into a single source-subnet/mask. That will give the core session-agent its own flow versus sharing one flow for multiple devices or the entire subnet.

DDOS Configuration Parameter Settings

Below are the recommended parameters setting for each platform in a SIP Peering model.

Changes under media-manager require system reboot to take effect. Be sure to follow precautions to reboot SBC(s) to unnecessary service outage during this execution.

AP 4500 CPU-2 256k CAM 8G memory w/copper GigE

Platform	**NN 4500 CPU-2**
CAM	**256K**
Memory	**8G**
Software Release	**..ECZ750**

Configuration Model	PBRB	SSNHTN	SNB
media-manager			
max-signaling-bandwidth	1767740		
max-untrusted-signaling	1	1	1
min-untrusted-signaling	1	1	1
tolerance-window	30		
min-media-allocation	32000		
min-trusted-allocation	60000		
deny-allocation	32000		
realm-config (peer)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		
realm-config (core)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		

maximum-signal-threshold	0
untrusted-signal-threshold	0

AP 4600 1000000 Flow Table 16G memory – copper single GigE

Platform	AP 4600
Flow Table	1000000
Memory	16G
Software Release	ECZ7.5.0

Configuration Model	PBRB	SSNHTN	SNB
media-manager			
max-signaling-bandwidth	2651610		
max-untrusted-signaling	1	1	1
min-untrusted-signaling	1	1	1
tolerance-window	30		
realm-config (peer)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		
realm-config (core)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		

AP 6100 1000000 Flow Table 16G memory –copper single GigE

Platform	AP6100
Flow Table	1000000
Memory	16G

Software Release	ECZ750

Configuration Model	PBRB	SSNHTN	SNB
media-manager			
max-signaling-bandwidth	7070960		
max-untrusted-signaling	1	1	1
min-untrusted-signaling	1	1	1
tolerance-window	30		
realm-config (peer)			
access-control-trust-level	high		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
Untrusted-signal-threshold	0		
realm-config (core)			
access-control-trust-level	high		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
Untrusted-signal-threshold	0		

AP 6300 1000000 Flow Table 16G memory – copper single GigE

Platform	AP 6300
Flow Table	1000000
Memory	16G
Software Release	ECZ750

Configuration Model	PBRB	SSNHTN	SNB
media-manager			
max-signaling-bandwidth	7070960		
max-untrusted-signaling	1	1	1
min-untrusted-signaling	1	1	1
tolerance-window	30		
realm-config (peer)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		
realm-config (core)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		

Observations/Limitations

The settings outlined in this appendix are beneficial when facing malicious attacks from any unknown sources; this is a typical concern when deploying peering traffic on the public Internet. Setting access-control-trust-level to “high” in both peer realm and an ACL (access-control) will yield an implicit deny scenario where traffic from unknown source IP addresses will be silently discarded at the hardware level in order to protect both the SBC’s host CPU and core devices from being attacked. The design of this configuration is not to prevent cases where malicious attacks are generated behind the trusted source IP within peer’s network, since all traffic from peer is consider as “trusted”. Therefore, the SBC will forward all traffic from trusted sources to the core network as allowed by the system’s hardware or software capabilities. There is no demotion event when access-control-trust-level at realm is set “high” as packets from trusted peer endpoint are always allocated the trusted queue for processing.

An alternative DdoS prevention practice in peering is to set access-control-trust-level to “medium”, but this type of configuration requires settings of ‟max-untrusted-signaling″, ‟min-untrusted-signaling″ and ‟maximum-signal-threshold”, which vary greatly from one customer to the next. Please contact your Sales Representative for more information on Professional Services available from Oracle to design comprehensive security solutions.

As the media-manager is a global configuration element, it assumes that the SBC has not been configured in hybrid mode, in which the SBC is configured to support both Access and Peering traffic. Further, it assumes the peer realm MUST have a sip-interface associated in order for the DdoS prevention configuration to be effective. Alternatively, in a Nested/Pseudo realm configuration, DdoS prevention configuration associated with the parent realm (which has a sip-interface associated) will apply.

Appendix D: DdoS Prevention for Access or Hybrid Environments

Configuration Models:

The settings outlined in this appendix apply to the following configuration models:

PBRB	Policy Based Realm Bridging Model
SNB	SIP NAT Bridge Model
SSNHTN	Single SIP NAT Hosted in Trusted Network Model

Supported platforms:

Platform	CAM/Flow Table	Memory
AP4500 (CPU 2)	256K (CAM)	8G
AP6300	1000000 (Flow Table)	16G
AP4600	1000000 (Flow Table)	16G
AP6100	1000000 (Flow Table)	16G
AP1100	1000000 (Flow Table)	4G
VME	1000000 (Flow Table)	4G
AP3900	1000000 (Flow Table)	16G

Configuration Parameters

The maximum signaling bandwidth per platform should be set to keep the CPU usage below 90%.

The following sections will discuss those "DDoS" parameter pertinent to the scope of this appendix. It's important to note: the parameters used to satisfy the requirements and scope of this appendix cannot be considered to be exhaustive. The parameters used are those which will be modified for this basic configuration. These parameters are in three configuration areas: Media Manager, Realm Configuration, and SIP Interface.

DDoS Configuration Parameter Descriptions

Media Manager

The following `media-manager` parameters have been calculated for each configuration model.

Parameter	Value
`max-untrusted-signaling`	Maximum percentage of the allocated max-signaling-bandwidth for untrusted traffic (%)
`min-untrusted-signaling`	Minimum percentage of the allocated max-signaling-bandwidth for untrusted traffic (%)
`max-signaling-bandwidth`	The maximum bandwidth that the SBC can withstand (bytes/sec)

These parameters are set to values that do not allow a SIP Register flood attack to increase the total CPU utilization percentage to over 89%. The background trusted traffic must not be adversely affected.

The recommended values for these media-manager parameters for each test scenario are listed by system model.

The following are Media Manager parameters that have platform specific defaults (not configurable, “show acl info” for details for each platform). For this appendix, these defaults will be used and are indicated in the platform results by system model.

Parameter
`min-media-allocation`
`min-trusted-allocation`
`deny-allocation`

Realm Configuration

The following `realm-config` parameters are used in the basic DDoS configuration. Only the **bold** values are changes from the default configuration

Parameter	Access Realm	Core Realm
`access-control-trust-level`	**`low`**	**`High`**
`invalid-signal-threshold`	**`1`**	**`0`**
`average-rate-limit`	**`0`**	**`0`**
`maximum-signal-threshold`	**`4000`**	**`0`**
`untrusted-signal-threshold`	**`1`**	**`0`**

Note that the maximum-signal-threshold of 4000 is very high so as not to impact service. It should be reduced to a number close to the maximum number of signaling messages from one client within the tolerance-window on the realm, which by default is 30 seconds. Base the threshold on an actual trace to account for the extraneous messages that are normally not considered, and make sure to account for network loss and/or renegotiations.

SIP Interface

The following `sip-interface->sip-ports` parameter should be used for access environments.

Parameter	Access Realm	Core Realm
`allow-anonymous`	**`registered`**	**`All`**

DDoS-2

DDoS-2 is supported for platforms: Acme Packet 4500 CPU-2. DDoS-2 increases the number of trusted endpoints to a maximum of 250K for Acme Packet 4500. It also increases the number of denied endpoints to a maximum 32K for Acme Packet 4500. In the new design, instead of providing a dedicated Traffic Manager flow for each endpoint, it provides 1024 TM-flows for each physical interface. A group of endpoints share one of the TM-flows. Dynamic trusted endpoints are stored in the HASH table. Dynamic denied endpoints are stored in the CAM or Flow table.

show commands

The command *show acl info* provides information about present usage of the HASH table.

```
show acl info
```

```
Access Control List Statistics:

                | # of entries |  % utilization  |  Reserved Entry
Count
----------------------------------------------------------------------
Denied          |       0            0.0%                 32000
Trusted         |       3            0.0%                  8000
Media           |       2            0.0%                 64000
Untrusted       |       1            0.1%                  2000
Dynamic Trusted |    4800            1.9%                250000
INTFC           |       2             -                    -
----------------------------------------------------------------------
Total CAM space used = 8 of 126976 (99.99% free)
Total HASH-table space used = 4800 of 250000 (98.08% free)
--------------------------------------------------------------------
```

The command *show acl all* presents endpoints allocation per TM-flow. In the example below we can see 5 endpoints per TM-flow:

```
trusted entries:
intf:vlan src-IP       dest-IP/mask  port prot type    index  recv   drop
0/0:0     0.0.0.0      177.1.1.100        ICMP static  65537  0      0
1/0:0     0.0.0.0      188.1.1.200        ICMP static  65539  0      0
1/0:0     0.0.0.0      188.1.1.200   5060 UDP  static  65541  333676 0
dynamic trusted entries sharing IFD 0x1e600:
0/0:0     14.0.2.130   177.1.1.100   5060 UDP  dynamic 132096 2      0
0/0:0     14.0.10.130  177.1.1.100   5060 UDP  dynamic 133120
0/0:0     14.0.18.130  177.1.1.100   5060 UDP  dynamic 134144
0/0:0     14.0.26.130  177.1.1.100   5060 UDP  dynamic 135168
0/0:0     14.0.34.130  177.1.1.100   5060 UDP  dynamic 136192
dynamic trusted entries sharing IFD 0x1e601:
0/0:0     14.0.2.132   177.1.1.100   5060 UDP  dynamic 132097 2       0
0/0:0     14.0.10.132  177.1.1.100   5060 UDP  dynamic 133121
0/0:0     14.0.18.132  177.1.1.100   5060 UDP  dynamic 134145
0/0:0     14.0.26.132  177.1.1.100   5060 UDP  dynamic 135169
0/0:0     14.0.34.132  177.1.1.100   5060 UDP  dynamic 136193
dynamic trusted entries sharing IFD 0x1e602:
0/0:0     14.0.2.134   177.1.1.100   5060 UDP  dynamic 132098 2       0
0/0:0     14.0.10.134  177.1.1.100   5060 UDP  dynamic 133122
0/0:0     14.0.18.134  177.1.1.100   5060 UDP  dynamic 134146
0/0:0     14.0.26.134  177.1.1.100   5060 UDP  dynamic 135170
0/0:0     14.0.34.134  177.1.1.100   5060 UDP  dynamic 136194
```

DDOS Configuration Parameter Settings

Below are the recommended parameter setting for each platform and Access Model.

AP4500 CPU-2 256k CAM 8Gb memory w/copper GigE

Platform	NN 4500 CPU-2
CAM	256K
Memory	8Gb
Software Release	ECZ750

Configuration Model	PBRB	SSNHTN	SNB
media-manager			
max-signaling-bandwidth	1767740		
options/s	5230		
bytes/option	338		
max-untrusted-signaling	15	13	12
min-untrusted-signaling	14	12	11
tolerance-window	30		
realm-config (access)			
access-control-trust-level	Medium		
average-rate-limit	0		
invalid-signal-threshold	1		
maximum-signal-threshold	4000		
untrusted-signal-threshold	1		
realm-config (core)			
access-control-trust-level	High		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		

AP 6300 1000000 Flow Table 16G memory – copper single GigE

Platform	NN 6300
Flow Table	1000000
Memory	16G
Software Release	ECZ750

Configuration Model	PBRB	SSNHTN	SNB
media-manager			

max-signaling-bandwidth	7070960		
max-untrusted-signaling	1	1	1
min-untrusted-signaling	1	1	1
tolerance-window	30		
realm-config (peer)			
access-control-trust-level	low		
average-rate-limit	0		
invalid-signal-threshold	1		
maximum-signal-threshold	4000		
untrusted-signal-threshold	1		
realm-config (core)			
access-control-trust-level	high		
average-rate-limit	0		
invalid-signal-threshold	0		
maximum-signal-threshold	0		
untrusted-signal-threshold	0		

NN 6100 1000000 Flow Table 16G memory –copper single GigE

Platform	AP6100
Flow Table	1000000
Memory	16G
Software Release	ECZ750

Configuration Model	PBRB	SSNHTN	SNB
media-manager			
max-signaling-bandwidth	7070960		
max-untrusted-signaling	1	1	1
min-untrusted-signaling	1	1	1
tolerance-window	30		
realm-config (peer)			
access-control-trust-level	**low**		
average-rate-limit	**0**		
invalid-signal-threshold	**1**		
maximum-signal-threshold	**4000**		
Untrusted-signal-threshold	**1**		
realm-config (core)			
access-control-trust-level	**high**		
average-rate-limit	**0**		
invalid-signal-threshold	**0**		
maximum-signal-threshold	**0**		
Untrusted-signal-threshold	**0**		

AP1100 720 Flow Table 4G memory –copper single GigE

Platform	AP1100
Flow Table	720
Memory	4G
Software Release	E-CZ7.4.0

media-manager		
max-untrusted-packet-rate	100000	
max-untrusted-signaling	7	
min-untrusted-signaling	4	
tolerance-window	30	
realm-config		
	no-Denied	Denied
access-control-trust-level	Medium	low
invalid-signal-threshold	2	1
maximum-signal-threshold	25	
untrusted-signal-threshold	10	1
nat-trust-threshold	0	
deny-period	30	1800

VME 720 Flow Table 4G memory

Platform	VME
Flow Table	720
Memory	4G
Software Release	E-CZ7.4.0

media-manager		
max-untrusted-packet-rate	100000	
max-untrusted-signaling	7	
min-untrusted-signaling	4	
tolerance-window	30	
realm-config		
	no-Denied	Denied
access-control-trust-level	Medium	low
invalid-signal-threshold	2	1
maximum-signal-threshold	25	
untrusted-signal-threshold	10	1
nat-trust-threshold	0	
deny-period	30	1800

AP3900 16K Flow Table 16G memory –copper single GigE

Platform	**AP3900**
FlowTable	**16K**
Memory	**16G**
Software Release	**ECZ7.5.0**

media-manager		
max-untrusted-packet-rate	400000	
max-untrusted-signaling	7	
min-untrusted-signaling	7	
tolerance-window	30	
realm-config		
	no-Denied	Denied
access-control-trust-level	Medium	low
invalid-signal-threshold	2	1
maximum-signal-threshold	25	
untrusted-signal-threshold	10	1
nat-trust-threshold	0	
deny-period	30	1800

AP4600 1000000 Flow Table 16G memory –copper single GigE

Platform	AP4600
Flow Table	1000000
Memory	16G
Software Release	ECZ740

media-manager		
max-signaling-bandwidth	2651610	
max-untrusted-signaling	15	
min-untrusted-signaling	12	
app-signaling-bandwidth	0	
tolerance-window	30	
realm-config		
	no-Denied	Denied
access-control-trust-level	Medium	low
invalid-signal-threshold	2	1
maximum-signal-threshold	25	
untrusted-signal-threshold	10	1
nat-trust-threshold	0	
deny-period	30	1800

Observations/Limitations

The settings outlined in this appendix are beneficial when facing malicious or non-malicious flood attacks, such as a REGISTER avalanche following a network outage. By limiting the amount of untrusted traffic to the SBC, the registration rate allowed will be throttled and the SBC will not be overrun by the high rate of registrations. However, there is an opportunity cost between the level of protection against a DDoS flood attack and the convergence time for this type of avalanche condition. For example, raising the percentage of untrusted bandwidth allowed will inevitably allow more untrusted traffic to traverse the SBC, and minimize the convergence time. The opportunity cost here is higher CPU usage during the flood, a result of higher demand on the processor due to the increased level of registrations it’s required to process.

Additionally, when set as an option in the sip-configuration, **reg-overload-protect** requires the SBC temporarily promote a registering endpoint upon receipt of a 401/407 response from the “real” registrar. This temporary promotion is in advance of the real and final promotion, which takes place following the 200 OK response to a REGISTER request containing authentication credentials. During a registration avalanche from untrusted sources, temporary promotion based on the initial REGISTER request sent from a specific source helps minimize the amount of time it will take to promote the collective untrusted sources, to trusted sources, effectively restoring service in the event of an outage as quickly as possible. This is also referred to as: minimizing the convergence time. The addition of any SIP option relevant to DDoS, including **reg-overload-protect**, would require additional testing. For customers with specific convergence requirements, additional research must be conducted to arrive at an appropriate DDoS configuration prior to deployment.

A limitation of the configuration parameters described in this appendix is the handling of SIP message spoofing. When a trusted user is “spoofed" by another user or a defective trusted user sends many SIP messages, the CPU utilization of the SBC may spike to 100%. One safe-guard implemented as part of this appendix is the establishment of a setting for **maximum-signaling-threshold**, defined in the realm-configuration object. When set, this provides an entry level amount of protection by removing a violating source from the trusted queue once the defined threshold is exceeded. To further handle this scenario, there are additional advanced DDoS configurations that can be set. For example: if the desired outcome is to deny violating sources from the hardware level, the **access-control-trust-level** should be set to “**low**” in the realm-configuration object. This also requires the configuration of the **untrusted-signal-threshold** to properly demote offending untrusted users to the deny list. If one wishes to move an endpoint back into the untrusted queue the access-control-trust-level of “medium” should be used.

The DDoS configuration recommendations in this appendix are meant as a general baseline to help protect the SBC from DDoS. For more complete protection, DDoS configurations should be determined by the examining the applicable environment and customizing based on the environment driven traffic flows and load levels.

Appendix E: Mitigating SIP Attacks

Goals

The goal of this appendix is to provide configuration recommendations to be implemented on the Session Border Controller (SBC) to reduce the negative effects of SIP scanning tools.

The configuration techniques described will reduce the impact of attacks by known tools. The intent is to drop all packets received from these tools without responding wherever possible. This is not possible in all cases. DDoS configuration adjustments will be recommended to reduce the impact of attacks on SBC resources and allow uninterrupted service to legitimate, trusted users.

Overview

SIP scanning and attack tools employed by fraudsters may target specific IP address ranges directly, but most tend to be random scans of a whole range of IP addresses. The scanning and attack methodology seen most frequently includes:

1. OPTIONS - Discover whether a SIP process is open and listening by asking for supported SIP options
2. INVITE - Check for an open service that will forward calls without authorization or challenge for registration by sending an initial call request
3. INVITE or REGISTER – Send calls and/or user authentication requests; Based on the error received it may be possible to enumerate user extensions, or in other words determine what accounts are available for password cracking.
4. REGISTER - Guess weak or default passwords; The attacker sends tens, hundreds, or even thousands of passwords per discovered extension until a password is found.
5. Start making calls. The attacker then registers a soft client and makes call attempts. The initial call attempt may not work if a dial prefix is needed, so attackers try all of those until they get an outside line

Most of the scanning tools such as SIPVicious, SIPScan, smap, and Sipsak are open source and freely available. Other tools are used exclusively by specific segments of criminals. As of the end of 2012, 99% of the attacks on customer systems and public SIP honeypots that we tracked were committed using an open source tool with easily identifiable characteristics.

This appendix provides configuration recommendations and references for more detailed information used to mitigate attacks by SIP scanning and attack tools. Several methods will be discussed since not all solutions may be acceptable in all customer environments.

Deployment Archetypes

Oracle classifies SIP deployments in three different major archetypes:

Peering: Calls are sent from a SIP proxy to the SD. The proxy may host SIP user agents or analog devices if a gateway function is provided. Peering is deployed either over a private network such as MPLS from service provider to customer, or over-the-top (OTT) via the Internet.

Customers using SIP peering or “trunking” deployments can usually implement a combination of trusted Session Agents (SA) and Access Control Lists (ACLs) to limit what remote IP addresses are able to communicate with the SD. In a peering network there is an implicit level of trust since the remote IP address is known and provisioned. When the trunk is delivered over a private network we are not usually concerned with SIP scanning prevention since there is no direct Internet access. In deployments where peering does happen over an untrusted network, such as OTT, the ACL entry drops incoming requests from unknown sources.

It then falls to the operator to determine if their particular architecture might see SIP scans from behind a trusted IP address. With multiple layers of NAT in IPv4 networks, it is always possible that messages are transiting through a firewall or gateway rather than just an individual SIP proxy.

Access: Calls are sent directly from a SIP endpoint to the SBC. A SIP registration may be required to authenticate and authorize the services available to the endpoint.

Access deployments will benefit the most from SIP scanner mitigation. This deployment model relies on the ability for users to roam, so ACLs based on known IP addresses cannot be used. Access to the network needs to be controlled via other means, usually through the use of a SIP registration.

Hybrid: Many networks have a mix of peering and access. In these cases, calls from remote subscribers may be sent to a trusted peer such as a service provider.

Strategies for Mitigating Against SIP Scanners

Mitigation Strategies

Mitigation against SIP scanners can be provided through several complementary strategies.

1. **Access Control**: Ensure proper configuration to block unauthorized end-points. Proper configuration of access control settings such as realm trust levels, access control lists (ACL), and SIP port allow-anonymous settings can limit traffic to known session agents and/or registered endpoints.
2. **Threat Identification**: Identify and drop messages from SIP scanners and avoid responding to the sender whenever possible - fraudulent messages can be dropped based on patterns found in the SIP messaging.
3. **Enforcement**: Limit attacks that cannot be identified as a scan from a known tool. Enforcement of message thresholds (DoS configuration) can demote or blacklist endpoints that do not become trusted or abuse their existing trust potentially limiting the damage of a scan.

Access Control

There are several types of access control that apply to deployments over untrusted networks.

Denial of Service Prevention: The section regarding DoS in this appendix covesr proper configuration of access control parameters. Guidelines are provided for configuring trust levels, ACLs, allow-anonymous settings, and message thresholds.

Signaling Authentication and Encryption: SIP can be encrypted using the Transport Layer Security (TLS) protocol. If the connection is established using mutual certificate authentication, then a resulting benefit is effective access control. During the TLS connection establishment, the endpoint verifies the SBC certificate, and the SBC verifies the endpoint certificate was issued by a trusted Certificate Authority (CA). That mutual authentication provides assurance that the device is legitimate, and not an attack tool. When combined with the use of online certificate status protocol (OCSP), it is possible for administrators to refuse network access to devices that are lost or have left the organization. If TLS with mutual authentication is used, then the effects of all SIP scanning tools are mitigated.

Not all endpoints support installation of third party certificates or TLS encryption, and it may be difficult for an organization to issue and manage individual client certificates. TLS (and optionally SRTP) may also require additional hardware for encryption acceleration.

Endpoint Whitelisting: If an organization manages the endpoints in use it can fingerprint them the same way we fingerprint attack tools. Endpoints will advertise a SIP User-Agent value or may have proprietary SIP headers that provide identifying values. Messages from endpoints that do not have these characteristics can be rejected using a Header Manipulation Rule. Section 3 of this Tech Note describes the Header Manipulation Rules required to perform User-Agent whitelisting.

Threat Identification Alternative 1: sipShield SPL plug-in

The Session Plug-in Language (SPL) is an Oracle API library that exposes core functions to an embedded LUA processor via call-backs. A plug-in is an additional piece of software written using SPL that runs on the SBC to implement a custom feature. It is supported via Oracle Consulting Services.

sipShield enables the SBC to drop SIP messages containing the identifying characteristics of known malicious tools with absolutely no response to the attacker. The sipShield plug-in examines multiple characteristics of each message, and is superior to our second option, "Header Manipulation Rules for Scanner Mitigation" described below. It is recommended that sipShield should be used wherever possible.

Since sipShield requires a specific SPL API version, it is not available for all software releases. Only recent releases of software support sipShield at this time. To determine if sipShield is supported issue the "show spl" command in the ACLI. If the SPL version found is 2.0.1 or greater then sipShield is supported. If the command is not found then SPL is not included in the software release.

```
ACMEPACKET# show spl
SPL Version: C2.0.1
```

Threat Identification Alternative 2: Header Manipulation Rules for Scanner Mitigation

If sipShield is not appropriate for your environment, the second alternative is to use SIP header manipulation rules (HMR) to drop messages received from known, fraudulent User-Agent(s). The HMR rule processes each inbound message, and if a match is found, it marks the message as invalid or "Rogue". Subsequent responses back to the attacker are dropped. Unfortunately the SD's B2BUA will usually respond with an initial response ("100 Trying" or a 4xx error) prior to evaluation with the HMR (the specific response depends on realm settings). This gives the attacker the knowledge that there is a SIP process running (even though the INVITE response is dropped). As they continue their attack, INVITE and REGISTER messages will be dropped without reaching the core, and they will eventually be demoted or blacklisted depending on your DoS settings.

Header Manipulation Rules for Scanner Mitigation are covered below.

Enforcement: Implement DoS Prevention

Some scanning tools will not match a known pattern because they are either new, or a skilled attacker has changed SIP fields to make them less detectable. DoS/DDoS prevention settings can protect against attacks that cannot be identified by their SIP messaging. Endpoint actions can be limited by requiring them to register first, and by enforcing defined message thresholds. The administrator can determine what happens when the thresholds are exceeded – either a 'demotion' to a queue with less bandwidth, or blacklisting for a configurable period.

Basic DDoS configuration settings are outlined in the other appendices. However, for the best DDoS protection, the configuration should be customized based on the customer environment and the traffic levels they actually receive.

DoS settings that help mitigate SIP scanning risks are also depicted below.

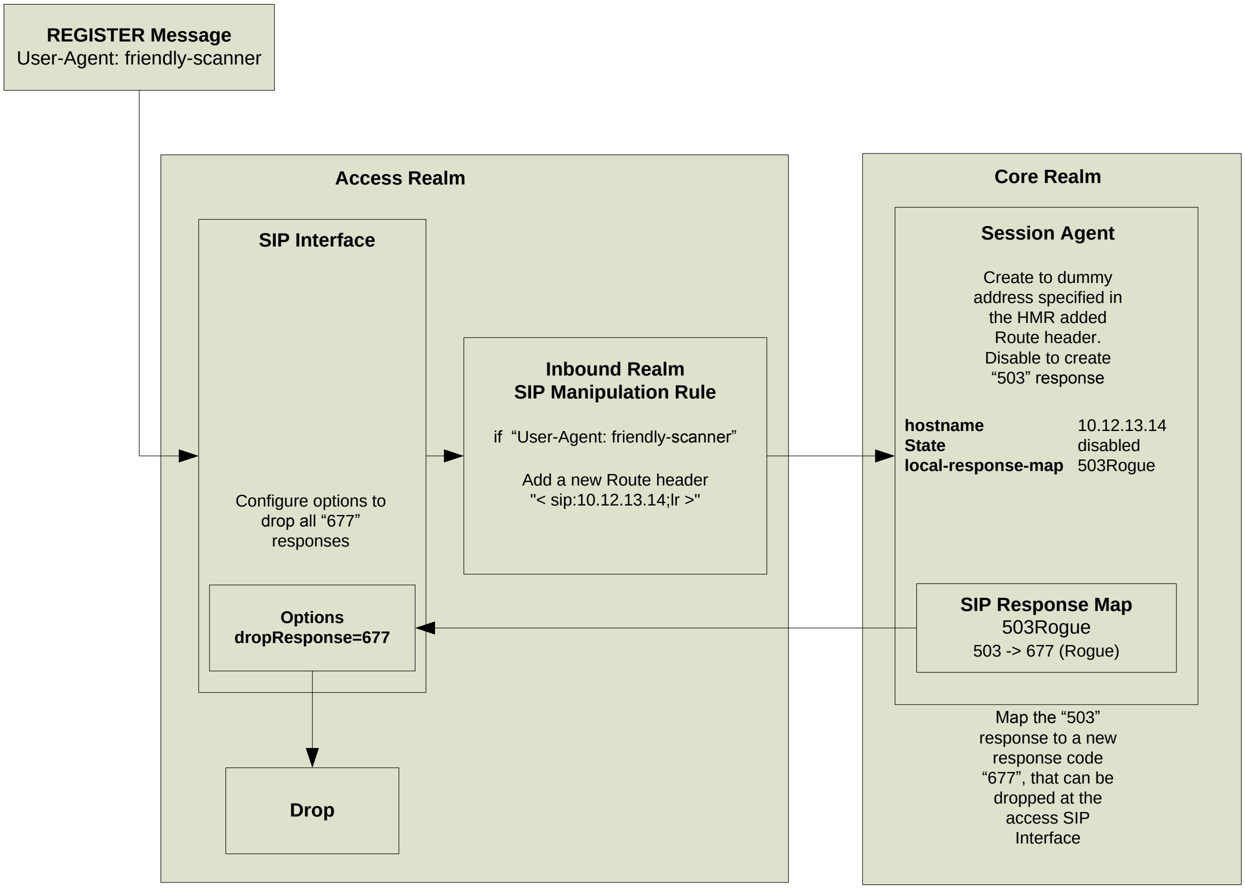
Header Manipulation Rules for Scanner Mitigation

SIP Header Manipulation Rule Logic

One way to drop all packets sent from an endpoint running an SIP scanner is to use a combination of SIP Manipulation Rules in conjunction with a dummy Session Agent. In this case, a dummy session agent is defined that is not an externally routable target.

As a message enters the SBC it is passed by the SIP interface to the incoming HMR. The HMR applies a regular expression against the message to determine if it is a scanning tool. If there is a match, the HMR can take action to mark it as invalid by inserting an additional route header and forwarding it to a “dummy” session agent. Provisioning the dummy session agent with the state disabled will cause the call to be refused. Custom mapping this to a unique error response can in turn be used in conjunction with a sip-interface option configurable to drop specific error responses.

Below is a flow diagram of how a SIP REGISTER message with a User-Agent header of “friendly-scanner” will be dropped.


REGISTER Message
User-Agent: friendly-scanner
Access Realm
SIP Interface
Configure options to drop all “677” responses
Options
dropResponse=677
Drop
Inbound Realm
SIP Manipulation Rule
if “User-Agent: friendly-scanner”
Add a new Route header
"< sip:10.12.13.14;lr >"
Core Realm
Session Agent
Create to dummy address specified in the HMR added Route header. Disable to create “503” response
hostname 10.12.13.14
State disabled
local-response-map 503Rogue
SIP Response Map
503Rogue
503 -> 677 (Rogue)
Map the “503” response to a new response code “677”, that can be dropped at the access SIP Interface

Some customers have asked about using the “reject” action in HMRs to send a “677 Rogue” response rather than routing to a dummy session agent. However, the “reject” action is evaluated immediately, and therefore is not affected by the sip-interface dropResponse parameter so the attacker will receive many more responses than you intend.

Configuration for C-series (Net-Net 3000, 4000, and 6000 series)

Inbound SIP Manipulation Rule

An inbound SIP Manipulation Rule needs to be created to modify any messages that contain a User-Agent header that is known (or suspected) to be fraudulent. The manipulation rule will add a Route header that directs the message to a “dummy” Session Agent.

Two rule examples are given. The first one identifies fraudulent User-Agent values, and the second whitelists only the desired User-Agent values and denies all others.

NOTES

- The list of User-Agents in the match-value shown in the example should be entered without spaces between the pipe symbols. There is an extra space for formatting.
- Release S-CX7.2.0 and greater allows you to log attack messages to the /ramdrv/logs/matched.log file if you wish. Simply change the store action in the isScanner rule to log.
- If you already have inbound sip-manipulations the header rules identified in the examples below can be added to them.

WARNING

- If you have valid users of Counterpath Eyebeam in your environment then you should delete the final “|eyeBeam” string from the match-value below.

Identifying fraudulent User-Agent values

In this HMR, the first header-rule uses a logical OR and performs a regular expression match on multiple known User-Agent values. If any of these partial matches is found then the value is stored. The second rule checks to see if the first rule stored a value, and inserts the Route header if it did.

```
sip-manipulation
        name                 AddRoutHdr
        description
        split-headers
        join-headers
        header-rule
                name                 isScanner
                header-name          User-Agent
                action               store
                comparison-type      pattern-rule
                msg-type             any
                methods
                match-value          friendly|sundayddr|SIPScan|
                                     smap|sipsak|sipcli|sipv|
                                     VaxIPUserAgent|eyeBeam
                new-value
        header-rule
                name                 addNullRoute
                header-name          Route
                action               add
                comparison-type      boolean
                msg-type             request
                methods
                match-value          $isScanner.$0
                new-value            "<sip:10.12.13.14;lr>"
```

Whitelisting known User-Agents

A similar HMR is used here. The first header rule is replaced with one that uses a logical OR with multiple values, but its regular expression match is for valid User-Agents. If a valid User-Agent match is found, it is stored. The expressions used can match a part of the User-Agent string, and can be as specific (or unspecific) as required. The final rule has been modified to insert the invalid route if a valid User-Agent was NOT found.

WARNING

If endpoint firmware is updated and the User-Agent string changes then the whitelist rule may start failing and endpoints will be denied. Make sure to perform lab testing prior to any endpoint software updates.

```
sip-manipulation
        name                 AddRoutHdr
        description
        split-headers
        join-headers
        header-rule
                name                 whitelist
                header-name          User-Agent
                action               store
                comparison-type      pattern-rule
                msg-type             out-of-dialog
                methods
                match-value          (Bria Professional release 2.4.3
                                     stamp 50906|UCCAPI|Avaya SIP R2.2
                                     Endpoint Brcm)
                new-value
        header-rule
                name                 addNullRoute
                header-name          Route
                action               add
                comparison-type      boolean
                msg-type             request
                methods
                match-value          !($whitelist.$0)
                new-value            "<sip:10.12.13.14;lr>
```

Realm: The access realm-config must also be modified to assign the SIP manipulation rule defined above as the inbound manipulation. Add the addRouteHdr manipulation for both fraudulent and whitelisting scenarios.

```
realm-config
        identifier               access
        description              Serving all access endpoints
        .
        .
        .
        in-translationid
        out-translationid
        in-manipulationid        addRouteHeader
```

Session Agent: A dummy Session Agent needs to be created with the state disabled. This is important so that the Session Agent will reply with a 503 response to any request. The 503 response will then be mapped to a new response code that can be easily dropped. A SIP Response Mapping is created to map any 503 from this Session Agent to a 677 SIP response code. We use an error code that is not valid according to RFCs so it can easily be distinguished from other traffic. Any 677 responses can then be dropped at the SIP Interface level without dropping any valid 503 responses from other endpoints. The SIP Response Mapping must be assigned in the Session Agent as shown below.

```
session-agent
        hostname               10.11.12.13
        ip-address
        port                   5060
        state                  disabled
        app-protocol           SIP
        transport-method       UDP
        realm-id               *
        .
        .
        .
        local-response-map     503Rogue
```

SIP Response Mapping: A SIP Response Mapping must be configured to map 503 responses from this Session Agent to a dummy response code (677). The response-map ACLI level can be found in configuration mode under session-router > sip-response-map.

```
response-map
        name              503Rogue
        entries           503 -> 677 (Rogue)
```

SIP Interface: All SIP interfaces that receive messages from SIP scanners require the option “dropResponse=677” to drop the 677 responses received from the dummy Session Agent.

```
sip-interface
        state              enabled
        realm-id           access
        .
        .
        .
        options            dropResponse=677
```

Scanner Mitigation using DDoS Settings

The DDoS settings recommended in the appendices will protect the SBC, but more strict trust levels and thresholds need to be defined to deny endpoints that are attempting to scan the system. To accomplish this, the access-control-trust-level on the access realm-config must be configured to low, which will cause endpoints to be blacklisted when they exceed thresholds. The untrusted-signal-threshold parameter defines the threshold of SIP messages received within the global tolerance-window (set under media-manager) before an untrusted user will be demoted to denied. The untrusted-signal-threshold should be set to a value that is just greater than the number of messages required by an untrusted endpoint to become trusted through SIP registration. The untrusted-signal-threshold value should be confirmed by collecting and analyzing a packet capture from the targeted network deployment. In many cases a registration will be two or three messages, but endpoint behavior and requirements vary. When this threshold is exceeded, the endpoint will be placed on the denied list for the amount of time defined in the deny-period. This period should be determined based on your individual needs. Setting the deny period to a long duration may cause problems for endpoints that simply entered an incorrect password or had a connection issue with some packet loss.

The following parameters should also be customized to your needs based on expected call flows.

```
realm-config
        identifier                      access
        description                     Serving all access endpoints
        .
        .
        .
        access-control-trust-level      low
        invalid-signal-threshold        1
        maximum-signal-threshold        4000
        untrusted-signal-threshold      5
        nat-trust-threshold             0
        deny-period                     120
```

Configure the media-manager settings per the recommendations in the DDoS prevention appendix that is applicable for your architecture. The max-untrusted-signaling parameter will limit the amount of untrusted traffic the SBC will process.

If any media-manager settings are changed you MUST save, activate, and reboot the SBC so they will take effect.

Peering Environments

As noted earlier, this appendix does not focus on scanning attacks in SIP Peering environments. In these environments it is recommended to create static ACLs with a trust level appropriate for the peer. It is recommended when peering over a trusted network, such as an MPLS connection delivered from a service provider, that a “high” trust level should be used. If your trust in the peer is not assured, it may be appropriate to set trust level to “medium” or “low” so they will be limited or blacklisted for abuse. Keep in mind that signaling thresholds will then need to be set on the realm.

The realm-config, access-control-trust-level should match the trust-level of the ACL so that all traffic from any endpoint that does not have an ACL will be denied. Always make sure that the realm-id, source-address, destination-address, and application-protocol are specified.

```
realm-config
        identifier                      peer
        description
        addr-prefix                     172.16.101.6
        .
        .
        .
        access-control-trust-level      high



access-control
        realm-id                        peer
        description
        source-address                  172.16.101.6
        destination-address             197.168.11.100
        application-protocol            SIP
        transport-protocol              ALL
```

```
acces                           permit
average-rate-limit              0
trust-level                     high
```

Appendix F: Intrusion Detection System

The SBC supports intrusion detection and protection capabilities using anomaly based detection. SIP messages are compared to their expected format per the SIP RFCs, and may be repaired or rejected based on the severity of the issue and the settings defined by the administrator. The Intrusion Detection System (IDS) provides notification of unexpected events using all of the SD's configured monitoring methods, though the amount of detail in each may vary. An optional IDS Reporting Feature Group license provides additional detail for attempted intrusions and suspicious behavior. IDS feature is part of SBC Base Entitlement Group, no extra license is required.

This section details the security related events and statistics the SBC monitoring features can provide, some of which may be used as input to a security monitoring platform. Some of the following information may be partially repeated in other sections, however the intent is to provide further details and depict the relationship of various indicators here.

IDS Details

IDS Reporting Feature Group has the additional capabilities described below.

- Media manager configuration elements visible after installing the license:
    - trap-on-demote-to-deny – controls traps for deny events
    - trap-on-demote-to-untrusted – controls traps for untrust demotion events
    - syslog-on-demote-to-deny – controls syslogs for deny events
- Access control list configuration elements visible after installing the license:
    - cac-failure-threshold –contributes to demotion
    - untrust-cac-failure-threshold –contributes to demotion
- Endpoint demotions based on admission control failures
- When the IDS license is installed, the apSysMgmtInetAddrWithReason-DOSTrap trap (described below) is available and the apSysMgmtExpDOSTrap is disabled. Without an IDS license installed, only the apSysMgmtExpDOSTrap trap is available.

Dependencies

On Net-Net 3800 systems, the DOS license must be installed in addition to the IDS license in order to enable all features described in this section.

Endpoint Promotions and Demotions

Endpoints, irrespective of whether or not they are defined as session-agents are promoted/demoted between hardware-enforced trusted, untrusted, and denied Access Control List traffic queues based on trust level configuration. Static ACLs are also configurable to further classify signaling traffic as being permanently assigned to the appropriate trust queue.

Trust is assigned through several mechanisms including the access-control-trust-level parameter of the realm the session-agent or end point is a member of, trust-level of provisioned ACLs, and the allow-anonymous setting on the applicable sip-interface.

The SBC will demote an endpoint if:

1. It receives too many signaling messages within the configured time window (maximum-signal-threshold in the realm or static ACL)
2. It receives too many invalid signaling messages within the configured time window (invalid-signal-threshold in the realm or static ACL)

3. It receives too many signaling messages from an untrusted source within the configured time window (untrusted-signal-threshold in the realm or static ACL)
4. A trusted endpoint exceeds the call admission controls and the cac-failure-threshold defined in an ACL (the call admission control limits are defined in media profiles)
5. An untrusted endpoint exceeds call admission controls and the untrust-cac-failure-threshold defined in an ACL

The SBC will promote an endpoint if:

1. It received a 200 OK response to a registration
2. The registration overload protection (reg-overload-protect) option has been set globally in the sip-config element (this is temporary, and only if a 401 or 407 response is received)
3. The deny-period has expired

Statistics

Each promotion and demotion event, between trusted, untrusted, and deny queues is counted and kept as an ACL statistic. These counts are maintained separately for SIP and MGCP signaling applications.

Statistics for ACL status and operations can be seen using the ACLI commands show sipd acls and show mgcp acls.

```
ACMESBC# show sipd acls
16:25:48-180
SIP ACL Status            -- Period -- -------- Lifetime --------
                Active    High   Total      Total  PerMax    High
Total Entries        0       0       0          0       0       0
Trusted              0       0       0          0       0       0
Blocked              0       0       0          0       0       0

ACL Operations         ---- Lifetime ----
                Recent      Total  PerMax
ACL Requests         0          0       0
Bad Messages         0          0       0
Promotions           0          0       0
Demotions            0          0       0
Trust->Untrust       0          0       0
Untrust->Deny        0          0       0
```

SNMP MIB OIDS

The ACL statistics counters described above are also available for SNMP polling under APSYSMGMT-MIB -> acmepacketMgmt -> apSystemManagementModule -> apSysMgmtMIBObjects -> apSysMgmtMIBGeneralObjects

- apSysSipEndptDemTrustToUntrust (.1.3.6.1.4.1.9148.3.2.1.1.19)
  - Global counter for SIP endpoint demotions from trusted to untrusted.
- apSysSipEndptDemUntrustToDeny (.1.3.6.1.4.1.9148.3.2.1.1.20)
  - Global counter for SIP endpoint demotions from untrusted to denied.
- apSysMgcpEndptDemTrustToUntrust (.1.3.6.1.4.1.9148.3.2.1.1.21)
  - Global counter for MGCP endpoint demotions from trusted to untrusted.
- apSysMgcpEndptDemUntrustToDeny(.1.3.6.1.4.1.9148.3.2.1.1.22)
  - Global counter for MGCP endpoint demotions from untrusted to denied.

SNMP Traps

Enabling the trap-on-demote-to-deny parameter located in the media-manager-config configuration element enables SNMP traps to be sent for demotions to the denied queue.

When the IDS license is installed, the apSysMgmtInetAddrWithReasonDOSTrap trap is sent. Otherwise, only the apSysMgmtInetAddrDOSTrap trap is sent.

The IDS Reporting Feature Group added the capability for the SBC to send a trap when the SBC demotes an endpoint to the untrusted queue. Enabling the trap-on-demote-to-untrusted parameter located in the media-manager-config configuration element enables these. The same apSysMgmtInetAddrWithReasonDOSTrap is sent.

When the IDS license is installed and the trap-on-demote-to-deny or trap-on-demote-to-untrusted parameters are disabled, the apSysMgmtI-netAddrWithReasonDOSTrap trap is not sent from the SBC, even when an endpoint is demoted.

When sent, the apSysMgmtInetAddrWithReasonDOSTrap contains the following data:

- apSysMgmtDOSInetAddressType—Blocked IP address family (IPv4 or IPv6)
- apSysMgmtDOSInetAddress—Blocked IP address
- apSysMgmtDOSRealmID—Blocked Realm ID
- apSysMgmtDOSFromURI—The FROM header of the message that caused the block (If available)
- apSysMgmtDOSReason—The reason for demoting the endpoint to the denied queue: This field can report the following three values:
    - Too many errors
    - Too many messages
    - Too many admission control failures

HDR

The SIP (sip-ACL-oper) and MGCP (mgcp-oper) HDR ACL status collection groups include the following two metrics:

- Demote Trust-Untrust - Global counter of endpoint demotion from trusted to untrusted queue
- Demote Untrust-Deny - Global counter of endpoint demotion from untrusted to denied queue

```
TimeStamp  ACL Requests Bad Msgs Promo Demo Demote Trust-Untrust Demote Untrust-Deny
1369338880 0            0        0     0    0                    0
1369338940 0            0        0     0    0                    0
1369339000 0            0        0     0    0                    0
1369339060 0            0        0     0    0                    0
```

Syslog

A syslog message can also be generated when an endpoint is demoted. Setting the media-manager config -> syslog-on-demote-to-deny parameter to enabled writes an endpoint demotion warning to the syslog every time an endpoint is demoted to the denied queue. Demotions from trusted to untrusted can also be reported by setting the media-manager -> syslog-on-demote-to-untrusted parameter to enabled. By default, these configuration options are set to disabled.

Without the IDS Reporting Feature Group license applied, the syslog messages have a WARNING level and look like this:

```
Jan 15 12:22:48 172.30.60.12 ACMESYSTEM sipd[1c6e0b90] WARNING
SigAddr[access:192.168.24.40:0=low:DENY] ttl=3632 guard=798 exp=30
Demoted to Black-List (Too many admission control failures)
```

The IDS Reporting Feature Group will provide an ERROR message with further detail like this:

```
Nov 28 17:53:47 172.41.3.41 ACMESYSTEM sipd[2dcc32a4] ERROR [IDS_LOG]
SigAddr[access:192.168.101.120:0=low:DENY] ttl=86400 exp=30 Demoted to
Black-List (Too many messages) last msg rcvd=REGISTER sip:192.168.66.2
SIP/2.0
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Via:
SIP/2.0/UDP 192.168.190.144:20928;branch=z9hG4bKdeadb33f
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR From:
<sip:47097@192.168.190.144:20928>
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR To:
<sip:47097@192.168.66.2:5060>
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Call-ID:
f9844fbe7dec140ca36500a0c9119870@192.168.66.2
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR CSeq: 1
REGISTER
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Contact:
<sip:47097@192.168.190.144>
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR User-agent:
UAC
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Max-
Forwards: 5
Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Content-
Length: 0
```

Keep in mind that some small number of demotions will be normal in a network, and that there may be an initial learning period where it's crucial to understand:

- What are the stable and "common" values of these counters
- On-going demotions/promotions on ACLs and to which SIP UAs they refer to

Monitoring systems need to be configured to take these normal variations into account, and have appropriate thresholds defined. Note that the thresholds, as well as the SBC DoS or CAC parameters may need to be adjusted over time as the network being monitored grows and changes.

Authentication Failures used for Endpoint Demotion

Endpoints that have become trusted due to successful registration are entered into the registration cache. The cache is used to store the user and location information for authenticated endpoints. It may also be used to shield the registrar from having to respond to re-registrations by providing the SBC the data to reply to a portion of re-registrations locally. However, if an endpoint fails re-registration, it will be demoted from trusted to untrusted.

Similarly, if an endpoint sends an INVITE with authentication, but the credentials do not match what is known to the registrar, it will be demoted as well.

In these cases, 401 or 407 responses are received from the registrar, and the demotion occurs.

Per-endpoint Call Admission Control

The SBC can demote endpoints from trusted to untrusted, or untrusted to denied queues when CAC failures exceed a configured threshold. The SBC maintains CAC failures per-endpoint. The CAC failure counter is incremented upon certain admission control failures only if either: cac-failure-threshold or untrust-cac-fail-threshold is set to a non-zero integer.

The cac-failure-threshold parameter is configurable in the access control and realm configuration elements. Exceeding the threshold integer defined in this parameter demotes an endpoint from the trusted queue to the untrusted queue. Additionally, the untrust-cac-failure-threshold parameter is configurable in the access control and realm configuration elements. Exceeding the threshold integer defined in this parameter demotes an endpoint from the untrusted queue to the denied queue. If both the cac-failure-threshold and untrust-cac-failure-threshold are configured to 0, admission control failures are considered and counted as invalid signaling messages for determining if the invalid-signal-threshold parameter value has been exceeded.

CAC failures used for Endpoint Demotion

The SBC determines CAC failures only by considering the number of signaling messages sent FROM an endpoint TO the realm its signaling messages traverse.

If an endpoint exceeds the following CAC thresholds, the SBC will demote the endpoint when the CAC failure thresholds are enabled.

- sip-interface user CAC sessions (realm-config > user-cac-sessions)
- sip-interface user CAC bandwidth (realm-config > user-cac-bandwidth)
- External policy server rejects a session

Thresholds and Trending Analysis

Thresholds and trending analysis are important concepts that must be well understood and implemented during initial installation of the SBC. Thresholds should be monitored and settings periodically adjusted as network usage or capacity requirements change. To be supported by Oracle TAC, SBC deployments require a minimum set of standard configurations outlined in the DDoS BCPs [10, 11]. These settings are considered the minimum configuration required to protect the SD. Upon deployment of a DDoS provisioned SBC it’s recommended that customers continuously monitor common traffic load and patterns of services traversing their SBC, and understand any alarms received.

Regardless of the monitoring method used (i.e. SNMP, CDR, HDR, Syslogs), during the initial period after implementation it’s crucial to understand:

- The number of active SIP sessions seen during normal and peak periods
- Average call hold times
- Average signaling messages for a call (usually best collected via Wireshark or other network capture tool)
- What are the stable and “common” values of these for the different counters
    - Trusted to Untrusted Demotions
    - Untrusted to Deny Demotions
    - Demotions
    - Promotions
- On-going demotions/promotions on ACLs, and to which SIP UAs they refer to
- Why there are any deny entries and to which SIP UAs they refer to
- Whether the deny period set is helping or causing more issues
- Whether the assigned trust level is denying more than one endpoint (e.g. issues with NAT)
- CAC or session count thresholds, and whether they are impacting service

Once this knowledge base is built and properly document for future reference, threshold values for reasonable variations in these counters should be defined and implemented in the monitoring platforms handling the SNMP Traps, HDR data, Sys-logs provided by the Session Border Controller.

It’s strongly recommended to parse and evaluate the information provided in any apSysMgmtInetAddrWithReasonDOSTrap SNMP traps received. Using this information it should be possible to identify SIP UAs and accounts involved, and understand whether legitimate traffic is being denied. Further actions may be required after this analysis; for example: configuration improvements to avoid illegitimate traffic from reaching the Host CPU may be needed, or, if the traffic is expected, adjustment of the appropriate constraints to allow the legitimate traffic to flow properly.

This process is an iterative loop where the fine-tuning and documenting illegal behavior flows can be continuously improved. This is especially true if the Session Border Controller is exposed to the Internet in an Access Scenario. When connected to the Internet, different trends and attempted illegal behaviors may be seen as the complexity of SIP attacks and trends evolve.

Constraints Limiting

The Session Border Controller provides two distinct mechanisms to throttle any SIP method: session constraints and rate-constraints. While session constraints are responsible for throttling both INVITE and REGISTER methods, rate constraints are used for throttling any other type of SIP method. Session constraints and rate constraints can be configured in either Session-Agent or SIP-interface config objects (via session-constraints). NOTE: Make sure to enable the sip-config > extra-method-stats option before configuring any constraints since this enables the constraint counters.

Session-Constraints

The session-constraints configuration element defines session layer constraints for session measurements such as maximum concurrent sessions, maximum outbound concurrent sessions, maximum session burst rate, and maximum session sustained rate.

The SIP interface configuration’s constraint-name parameter applies a pre-defined session-constraint configuration. Using the constraints defined, the SBC checks and limits traffic according to those settings for the SIP interface. If session constraints are not configured or applied on the SIP interface, the SIP interface will be unconstrained. If a single session-constraint element is applied to multiple SIP interfaces, each SIP interface will maintain its own copy of the session-constraint statistics.

- **name** – name of the session-constraint, this must be an unique identifier
- **max-sessions** - maximum sessions allowed for this constraint
- **max-inbound-sessions**— maximum inbound sessions allowed for this constraint
- **max-outbound-sessions**— maximum outbound sessions allowed for this constraint
- **max-burst-rate**—maximum burst rate (invites per second) allowed for this constraint
- **max-inbound-burst-rate**—maximum inbound burst rate (number of session invitations per second) for this constraint
- **max-inbound-sustain-rate**—maximum inbound sustain rate (of session invitations allowed within the current window) for this constraint
- **max-outbound-burst-rate**—maximum outbound burst rate (number of session invitations per second) for this constraint
- **max-sustain-rate**—maximum rate of session invitations allowed within the current window for this constraint
- **max-inbound-sustain-rate**—maximum inbound sustain rate (of session invitations allowed within the current window) for this constraint

- **max-outbound-sustain-rate**—maximum outbound sustain rate (of session invitations allowed within the current window) for this constraint
- **min-seizures**—minimum number of seizures for a no-answer scenario
- **min-asr**—Enter the minimum ASR in percentage
- **time-to-resume**—number of seconds after which the SA (Session Agent) is put back in service (after the SA is taken out-of-service because it exceeded some constraint)
- **in-service-period**—Enter the time in seconds that elapses before an element (like a session agent) can return to active service after being placed in the standby state
- **ttr-no-response**—Enter the time delay in seconds to wait before changing the status of an element (like a session agent) after it has been taken out of service because of excessive transaction timeouts
- **burst-rate-window**—Enter the time in seconds used to measure the burst rate
- **sustain-rate-window**—Enter the time in seconds used to measure the sustained rate

Oracle recommends use of session constraints on external SIP interfaces to limit the total number of sessions and / or traffic bursts that the combined configured session agents can handle for that service. Additionally, having multiple public SIP interfaces defined can limit the resources a particular SIP interface can provide based on service level agreements or the trust level of the endpoint.

Rate constraints

The rate-constraints sub-element is configurable under both the session-constraints and session-agent configuration elements (though they are not shared). It allows configuration of rate limiting based on specific method types. These further restrict any defined constraints of the parent, so they cannot exceed the rates defined at the level under which they are set.

- **method**—the SIP method name for the method to throttle, possible values are: NOTIFY, OPTIONS, MESSAGE, PUBLISH, REGISTER
- **max-inbound-burst-rate**—For the SIP method configured in the method parameter, this number will restrict the inbound burst rate on the SIP interface.
- **max-outbound-burst-rate**—For the SIP method configured in the methods parameter, this number will restrict the outbound burst rate on the SIP interface.
- **max-inbound-sustain-rate**—For the SIP method configured in the methods parameter, this number will restrict the inbound sustain rate on the SIP.
- **max-outbound-sustain-rate**—For the SIP method configured in the methods parameter, this number will restrict the outbound sustain rate on the SIP interface.

Each rate constraint configured for a SIP method maintains its own counters. For example, if a rate constraint for the PUBLISH method is configured, the burst and sustain rates set for it apply only to the PUBLISH method and not to any other methods.

The SBC captures statistics for SIP methods that have already been throttled by rate constraints for SIP interfaces and session agents; it does not capture these statistics for the global SIP configuration. SIP interfaces have two states: “In Service” and “Constraints Exceeded.” When any one of the constraints is exceeded, the status of the SIP interface changes to “Constraints Exceeded” and stops accepting traffic. It remains in that state until the time-to-resume period ends. The session constraint timers that apply to the SIP interface are the time-to-resume, burst window, and sustain window.

Oracle recommends configuration of INVITE and REGISTER method rate constraints on session agents.

For SIP access deployments, rate constraints for individual method types along with a set of burst and sustain rates should be considered. These constraints can help to avoid overloading the core network. In addition, they restrain the load non-INVITE messages use, thus reserving capacity for INVITE-based sessions and registrations.

In order to properly configure constraint limiting, either at SIP interface level or per Session-Agent (SA), it’s essential to have an accurate understanding of the SIP Message flows that exist in the network. Contributing factors include: factors such as which SIP requests are authenticated, what Call flows and Session Agents require re-INVITEs, maximum CPS per SA, etc. The reason why these details are so important is the SBC is making dynamic decisions and acting on this traffic in real time.

SNMP traps will be sent when constraints are exceeded. Constraint threshold crossing alarms or statistics are not necessarily a security issue since legitimate traffic overloads or mass network restarts may also cause them. It is up to the customer to assess if they should investigate alarms as possible security incidents.

To monitor SIP interface and Session Agents, two commands are most useful. The following commands include statistics on how many times the constraints were exceeded and the interface or session agent was temporarily taken out of service.

```
show sipd interface <realm name> and show sipd agents <agent name>
ACMEPACKET# show sipd interface access
00:51:55-34
Sip Interface access
                             -- Period -- -------- Lifetime --------
                   Active    High   Total      Total  PerMax    High
Inbound Sessions     9000    9002    1715   14244739    1501    9009
  Rate Exceeded         5       5       5          5       5       5
  Num Exceeded          -       -       0          0       0       -
  Burst Rate            0      50       0          0       0      51
Outbound Sessions       0       0       0          0       0       0
  Rate Exceeded         -       -       0          0       0       -
  Num Exceeded          -       -       0          0       0       -
  Burst Rate            0       0       0          0       0       0
Local Contacts          0       0       0          0       0       0
HNT Entries             0       0       0          0       0       0
Non-HNT Entries         0       0       0          0       0       0
Subscriptions           0       0       0          0       0       0
Out of Service          -       -       0          0       0       -
Trans Timeout           0       0       0          0       0       0
Requests Sent           -       -       0        284       1       -
Requests Complete       -       -       0          0       0       -
Seizure                 -       -       0          0       0       -
Answer                  -       -       0          0       0       -
  ASR Exceeded          -       -       0          0       0       -
Messages Received       -       -   14097  114313292   12405       -
Latency=0.000; max=0.000



ACMEPACKET# show sipd agents 192.168.60.10
00:54:10-49
Session Agent 192.168.60.10() [In Service]
                             -- Period -- -------- Lifetime --------
                   Active    High   Total      Total  PerMax    High
Inbound Sessions        0       0       0          0       0       0
  Rate Exceeded         -       -       0          0       0       -
  Num Exceeded          -       -       0          0       0       -
  Burst Rate            0       0       0          0       0       0
  Reg Rate Exceeded     -       7      21         21      21      21
  Reg Burst Rate        0       0       0          0       0       0
Outbound Sessions    9000    9003    2452   14251475    1501    9009
  Rate Exceeded         -       -       0          0       0       -
  Num Exceeded          -       -       0          0       0       -
  Burst Rate            0      50       0          0       0      51
  Reg Rate Exceeded     -       -       0          0       0       -
Local Contacts          0       0       0          0       0       0
HNT Entries             0       0       0          0       0       0
Non-HNT Entries         0       0       0          0       0       0
Subscriptions           0       0       0          0       0       0
Out of Service          -       -       0          3       1       -
Trans Timeout           0       0       0         44       1      40
Requests Sent           -       -   17666  100035216   10906       -
Requests Complete       -       -   17671  100035175   10905       -
Seizure                 -       -    2456   14251479    1501       -
Answer                  -       -    2456   14250766    1502       -
  ASR Exceeded          -       -       0          0       0       -
Messages Received       -       -   22595  128521055   13904       -
Latency=0.002; max=0.033
```

Message Rejections

The action type called reject is available to all header manipulation rules. When this action type is used, and a condition matching the manipulation rule arises, the SBC rejects the request, provides a SIP error, and increments a counter.

- If the msg-type parameter is set to any and the message is a response, the SBC increments a counter to show the intention to reject the message—but the message will continue to be processed.
- If the msg-type parameter is set to any and the message is a request, the SBC performs the rejection and increments the counter.

The header manipulation rule -> new-value parameter is designed to supply the status code and reason phrase corresponding to the reject. The following syntax is used to supply this information: status-code[:reason-phrase] . The status-code and reason phrase information is not required since by default the system uses 400:Bad Request.

If this information is not supplied, the status code must be a positive integer between 300 and 699. With this defined, the SBC will use the applicable reason phrase corresponding to the status code in responses. To customize the reason phrase, enter the status code followed by a colon (:). NOTE: be sure to enclose the entire entry in quotation marks (ex: “400:Go Away”) if the reason phrase includes spaces.

When the SBC performs the reject action, the current SIP manipulation stops processing and does not act on any of the rules following the reject rule. This course of action is also true for nested SIP manipulations that might have been constructed using the sip-manip action type. Keeping that in mind, the reject rule is usually the last rule in a long HMR.

Reject actions may also indirectly generate SNMP traps. Two parameters in the session-router-config define how many messages within a window of time cause the SBC to generate an SNMP trap.

- **reject-message-threshold**— defines the minimum number of message rejections allowed in the reject-message-window time on the SBC (when using the SIP manipulation action reject) before generating an SNMP trap.
- **reject-message-window**—defines the time in seconds that defines the window for maximum message rejections allowed before generating an SNMP trap. This should be set to something like 30 seconds to a minute. If set too low traps may be missed.

The SBC tracks messages that have been flagged for rejection using the reject action type. In the show sipd display, refer to the Rejected Messages category. Note that there is no distinction between requests and responses.

```
SIP Status                -- Period -- -------- Lifetime --------
                Active    High   Total      Total  PerMax    High
Sessions             0       0       0        538     211      38
Subscriptions        0       0       0          0       0       0
Dialogs              0       0       0        276      74      74
CallID Map           0       0       0       1076     422     386
Rejections           -       -       0          0       0
ReINVITEs            -       -       0          0       0
ReINV Suppress       -       -       0          0       0
Media Sessions       0       0       0        538     211      76
Media Pending        0       0       0          0       0       0
Client Trans         0       0       0        814     241      76
Server Trans         0       0       0       3626     366     193
Resp Contexts        0       0       0        538     211     193
Saved Contexts       0       0       0          0       0       0
Sockets              3       3       0          3       3       3
Req Dropped          -       -       0          0       0
DNS Trans            0       0       0          0       0       0
DNS Sockets          0       0       0          0       0       0
DNS Results          0       0       0          0       0       0
Rejected Msgs        0       0       0        200     108     108
```

SNMP support

- apSysRejectedMessages (.1.3.6.1.4.1.9148.3.2.1.1.18.0)
  - Number of messages rejected by the SBC due to matching criteria
- apSysMgmtRejectedMesagesThresholdExeededTrap (.1.3.6.1.4.1.9148.3.2.6.0.57)
  - The trap will be generated when the number of rejected messages exceeds the configured threshold within the configured window.
- apSysMgmtSipRejectionTrap (.1.3.6.1.4.1.9148.3.2.10.0.1)
  - Generated when a SIP INVITE or REGISTRATION request fail.

Log Action

The action type called: “log” is available to all header manipulation rules. When this action type is used, and a condition matching the manipulation rule arises, the SBC logs information about the current message to a separate log file.

This feature can be used to log important details from specific suspicious users, such as well-known SIP User-Agents, call attempts to undesirable destinations (known “hotlist” numbers, unassigned numbers, Premium Rate numbers, etc.).

If a match is found in an HMR, and the action is set to “log”, a logfile called matched.log will be created. The matched.log file contains a log message that contains a timestamp, destination IP address:port information, and the source IP address:port. It also specifies the rule that triggered the log action. The request URI, Contact header, To Header, and From header are also recorded. See the example below.

```
----------------------------------------
Apr 17 14:17:54.526 On [0:0]192.168.1.84:5060 sent to 192.168.1.60:5060
element-rule[checkRURIPort]
INVITE sip:service@192.168.1.84:5060 SIP/2.0
From: sipp <sip:+2125551212@192.168.1.60:5060>;tag=3035SIPpTag001
To: sut <sip:service@192.168.1.84>
Contact: sip:sipp@192.168.1.60:5060
```

Appendix G: Blacklisting with Local Routing Tables

Several industry groups such as the GSMA Fraud Forum and the Communications Fraud Control Association (CFCA) track phone numbers and number prefixes that have been verified as participating in various types of fraud. These numbers are published as a list for their members. Many organizations also track numbers that abuse their network on a regular basis.

While it can be more of an art than a science, some customers wish to blacklist incoming or outgoing calls based on a dialed number prefix, or the entire dialed number. While complex or expensive fraud management solutions can be used, this Appendix provides a simple way to perform this blacklisting on the SBC.

WARNING: This procedure will end up denying calls coming into your network. Be sure to test your local route tables (LRT) in a test environment before deploying in production.

It is assumed that as calls have been sent to the SBC or as they enter the SBC that they will be “normalized” by either the directly connected agent or an incoming HMR to match the local dial plan. For example in North America it is necessary to include the leading “1” for NANP and remove the “011” for calls outside of the NANP. If this is not possible, then the “011” can be pre-pended onto the number matches in the LRT file.

Depending on what you are trying to prevent, you may want to check both the FROM and TO fields in SIP messages. This same strategy can be used on your access realm, or even your core realm if you so choose.

To create the blacklist for routing you need to:

1. Enter your FROM or TO blacklist numbers into one or more LRTs - and save them with an “.xml” extension. Next gzip them (.gz format). A sample LRT format is found below.
2. Upload the .xml.gz file to the to SBC in the /code/lrt directory (which will need to be created the first time)
3. Update SBC config as depicted below

Apply an LRT check for the SIP From and To headers as the first two policy-attributes on all incoming realms, and on the core side if you want to detect outgoing fraudulent calls.

```
local-policy
        from-address
                                     *
        to-address
                                     *
        source-realm
                                     access
        description
        activate-time                N/A
        deactivate-time              N/A
        state                        enabled
        policy-priority              none
        last-modified-by             admin@10.0.240.19
        last-modified-date           2012-10-26 17:13:15
```

(The first policy checks the FROM field. Note that the .xml.gz file extension is not specified.)

```
          policy-attribute
```

```
        next-hop                        lrt:blacklist;key=$FROM
        realm
        action                          none
        terminate-recursion             enabled
        carrier
        start-time                      0000
        end-time                        2400
        days-of-week                    U-S
        cost                            0
        app-protocol
        state                           enabled
        methods
        media-profiles
        lookup                          single
        next-key
        eloc-str-lkup                   disabled
        eloc-str-match
```

(The second policy checks the TO field. This is OPTIONAL, and only if you want to check the number being dialed. You can use the same LRT file, or a different file.)

```
policy-attribute
        next-hop                        lrt:blacklist;key=$TO
        realm
        action                          none
        terminate-recursion             enabled
        carrier
        start-time                      0000
        end-time                        2400
        days-of-week                    U-S
        cost                            0
        app-protocol
        state                           enabled
        methods
        media-profiles
        lookup                          single
        next-key
        eloc-str-lkup                   disabled
        eloc-str-match
```

(The third and last policy is essentially a default SIP route that forwards calls onto the core.)

```
policy-attribute
        next-hop                        192.168.60.10
        realm                           core
        action                          none
        terminate-recursion             disabled
        carrier
        start-time                      0000
        end-time                        2400
        days-of-week                    U-S
        cost                            0
        app-protocol
        state                           enabled
        methods
        media-profiles
```

```
                      lookup                         single
                      next-key
                      eloc-str-lkup                  disabled
                      eloc-str-match
```

Create the LRT configuration referenced above. Note that the “best” match mode matches from most specific to least specific in the LRT table (greatest number of digits matched to fewest).

```
      local-routing-config
              name                           blacklist
              file-name                      blacklist.xml.gz
              prefix-length                  15
              string-lookup                  disabled
              retarget-requests              enabled
              match-mode                     best
              last-modified-by               admin@10.0.240.19
              last-modified-date             2012-10-26 15:40:48
```

Sample Entries from the LRT are seen below. In this case I’ve opted to forward the blacklist call onto a “dummy” session agent. You may opt to forward the call onto a recording, or session agent that handles fraud calls.

```
      <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
      <localRoutes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <route>
                  <user type="E164">3712900</user>
                  <next type="regex">!(^.*$)!sip:\110.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">88183521</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">2637749</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">3718104</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">3718103</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">3716852</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">447924</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
            <route>
                  <user type="E164">3712769</user>
                  <next type="regex">!(^.*$)!sip:\1@10.11.12.13!</next>
            </route>
```

```
</localRoutes>
```

Once the configuration has been saved and activated, the LRT file contents can be confirmed by executing the command “show lrt route-entry blacklist 3712900” at the ACLI (or any of the other blacklist prefixes, or numbers that contain the prefix).

Next, the dummy session agent must be set up. The hostname must match the LRT host entry. Note that the response-map AND local-response map are required to identify blacklisted calls vs. just valid errors.

Make sure the session-agent state is disabled so traffic or error messages are not sent to a real host. It should be noted that use of a real hostname will not work due to the way DNS resolution works in conjunction with disabled session-agents.

```
session-agent
        hostname                       10.11.12.13
        ip-address
        port                           5060
        state                          disabled
        app-protocol                   SIP
        app-type
        transport-method               UDP
        realm-id
        response-map                   503Fraud
        local-response-map             503Fraud
```

Next, map the error to an error code we can identify and log / reject for trap.

```
session-router > sip-response-map
response-map
        last-modified-by               admin@10.0.240.19
        last-modified-date             2012-10-26 17:06:07
        name                           503Fraud
        entries                        503 -> 678 (Fraud)
```

On the access realm, the out-manipulationid should reference the “logBlacklist” HMR below. Note that if there is already an outbound HMR in place that the header rules below can be added to the existing HMR.

```
sip-manipulation
        name                           logBlacklist
        description
        split-headers
        join-headers
        header-rule
                name                           logBlacklist
                header-name                    @status-line
                action                         manipulate
                comparison-type                case-sensitive
                msg-type                       reply
                methods
                match-value
                new-value
```

(Log the call that matched the blacklist to a local file “matched.log” on the SBC)

```
element-rule
        name                           logstatus
```

```
            parameter-name
            type                           status-code
            action                         log
            match-val-type                 any
            comparison-type                case-sensitive
            match-value                    678
            new-value
```

(Replace the SIP status code and reason message with whatever you want to send back to the service provider or client.)

```
    element-rule
            name                           replaceStatus
            parameter-name
            type                           status-code
            action                         replace
            match-val-type                 any
            comparison-type                case-sensitive
            match-value                    678
            new-value                      603
    element-rule
            name                           replaceReason
            parameter-name
            type                           reason-phrase
            action                         replace
            match-val-type                 any
            comparison-type                case-sensitive
            match-value                    Fraud
            new-value                      Declined
```

(Finally, invoke the SBC message rejection via HMR.)

```
    element-rule
            name                           rejectDeclined
            parameter-name
            type                           reason-phrase
            action                         reject
            match-val-type                 any
            comparison-type                case-sensitive
            match-value                    Declined
            new-value
```

Notice that this config will send along the 603 error code which should be enough to refuse a call and stop recursion. If your trunking provider has a different standard message this can easily be changed.

The “reject” action in “rejectDeclined” will cause the “Rejected Messages” count to increment in the show sipd display. If you wish to send traps to a management station when this HMR fires, update the settings in session-router-config. The configuration below will send a apSysMgmtRejectedMesagesThresholdExeededTrap whenever more than one blacklisted call is seen inside a 30 second window. This is an indicator that the administrator should examine the matched.log file to determine the number pattern that was seen.

```
session-router > session-router > sel
reject-message-threshold      1
reject-message-window         30
```

Blacklist Table Maintentance

As new blacklist tables are released the customer can upload to /code/lrt and execute the following commands:

```
ACMEPACKET# config t
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# local-routing-config
ACMEPACKET(local-routing-config)# select
<name>:
1: name=blacklist file name=blacklist.xml.gz prefixLength=15

selection: 1
```

Use the "show" command to verify the local-routing-config entry's configuration

```
ACMEPACKET(local-routing-config)# show
local-routing-config
        name                           blacklist
        file-name                      blacklist.xml.gz
        prefix-length                  15
        string-lookup                  disabled
        match-mode                     best
```

Change the "file-name" parameter to reflect the original compressed XML file

```
ACMEPACKET(local-routing-config)# file-name lookup.xml.gz
ACMEPACKET(local-routing-config)# done
local-routing-config
        name                           blacklist
        file-name                      blacklist102612.xml.gz
        prefix-length                  15
        string-lookup                  disabled
        match-mode                     best
```

Exit out of configuration mode, save, and activate the configuration

```
ACMEPACKET(local-routing-config)# exit
ACMEPACKET(session-router)# exit
ACMEPACKET(configure)# exit
ACMEPACKET#save-config
ACMEPACKET#activate-config
Activate-Config received, processing.
waiting for request to finish
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
```

After applying a new LRT, verify if by doing the same command from above "show lrt route-entry blacklist 3712900" at the ACLI (again, any of the hotlist numbers can be used). If something went wrong, change your config back to the old file and re-test.

After you have a few LRT files on the SBC you may want to clean the old ones up…

Appendix H: Simple Network Management Protocol (SNMP)

SNMP OIDs

Simple Network Management Protocol (SNMP) polling (GET and SET) requests are used to retrieve operational data and modify configuration are supported by SBC. The SBC supports SNMPv1 and SNMPv2c for GET and SET requests. Oracle release specific SNMP MIBs can be found on docs.oracle.com for the release in use.

System Management Statistics

Below is a recommended list of SNMP OIDs to GET every 5 minutes from the Oracle System Management MIB (ap-smgmt.mib). These will provide useful data on overall system performance and security issues.

apSysMgmtMIBObjects (1.3.6.1.4.1.9148.3.2.1)

apSysMgmtGeneralObjects (1.3.6.1.4.1.9148.3.2.1.1)

- apSysCPUUtil (1.3.6.1.4.1.9148.3.2.1.1.1)
  - Percentage of CPU utilization
- apSysMemoryUtil (1.3.6.1.4.1.9148.3.2.1.1.2)
  - Percentage of memory utilization
- apSysHealthScore (1.3.6.1.4.1.9148.3.2.1.1.3)
  - System health percentage
- apSysRedundancy (1.3.6.1.4.1.9148.3.2.1.1.4)
  - Active or Standby SD
- apSysGlobalConSess (1.3.6.1.4.1.9148.3.2.1.1.5)
  - Total instant number of system concurrent sessions
- apSysGlobalCPS (1.3.6.1.4.1.9148.3.2.1.1.6)
  - Instant number of system calls per second
- apSysNATCapacity (1.3.6.1.4.1.9148.3.2.1.1.7)
  - Percentage of NAT table in CAM utilization
- apSysARPCapacity (1.3.6.1.4.1.9148.3.2.1.1.8)
  - Percentage of ARP table in CAM utilization
- apSysLicenseCapacity (1.3.6.1.4.1.9148.3.2.1.1.10)
  - Percentage of licensed sessions in use
- apSysSipStatsActiveLocalContacts (1.3.6.1.4.1.9148.3.2.1.1.11)
  - Current number of cached SIP registered contacts
- apSysApplicationCPULoadRate (1.3.6.1.4.1.9148.3.2.1.1.16)
  - Average load rate of applications over past 10 seconds
- apSysSipEndptDemTrustToUntrust (1.3.6.1.4.1.9148.3.2.1.1.19)
  - Number of SIP endpoints demoted from trusted to untrusted queue
- apSysSipEndptDemUntrustToDeny (1.3.6.1.4.1.9148.3.2.1.1.20)
  - Number of SIP endpoints demoted from untrusted queue to denied
- apSysMgcpEndptDemTrustToUntrust (.1.3.6.1.4.1.9148.3.2.1.1.21)
  - Global counter for MGCP endpoint demotions from trusted to untrusted
- apSysMgcpEndptDemUntrustToDeny(.1.3.6.1.4.1.9148.3.2.1.1.22)
  - Global counter for MGCP endpoint demotions from untrusted to denied
- apSysRejectedMessages (.1.3.6.1.4.1.9148.3.2.1.1.18.0)
  - Number of messages rejected by the SBC due to matching criteria

apSysStorageSpaceTable (1.3.6.1.4.1.9148.3.2.1.1.23)

apSysStorageSpaceEntry (1.3.6.1.4.1.9148.3.2.1.1.23.1)

- apSysVolumeAvailSpace (1.3.6.1.4.1.9148.3.2.1.1.23.1.4)
  - Space remaining on the Storage Expansion Module (in MB)

apSysMgmtInterfaceObjects (1.3.6.1.4.1.9148.3.2.1.8)

apSysMgmtPhyUtilTable (11.3.6.1.4.1.9148.3.2.1.8.1)

- apPhyUtilTableRxUtil (1.3.6.1.4.1.9148.3.2.1.8.1.1.1)
  - Received Network Interface utilization over one second period
- apPhyUtilTableTxUtil (1.3.6.1.4.1.9148.3.2.1.8.1.1.2)
  - Transmitted Network Interface utilization over one second period

Realm Statistics

Below is a recommended list of SNMP OIDs to GET every 5 minutes from the Oracle System Management MIB (ap-smgmt.mib). These will provide useful SIP performance data on a per realm basis.

apSigRealmStatsTable (1.3.6.1.4.1.9148.3.2.1.2.4)

apSigRealmStatsEntry (1.3.6.1.4.1.9148.3.2.1.2.4.1)

- apSigRealmStatsRealmName (1.3.6.1.4.1.9148.3.2.1.2.4.1.2)
  - Realm name with corresponding stats
- apSigRealmStatsCurrentActiveSessionsInbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.3)
  - Number of active inbound sessions for this realm
- apSigRealmStatsCurrentSessionRateInbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.4)
  - CPS rate for active inbound sessions for this realm
- apSigRealmStatsCurrentActiveSessionsOutbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.5)
  - Number of active outbound sessions for this realm
- apSigRealmStatsCurrentSessionRateOutbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.6)
  - CPS rate for active outbound sessions for this realm
- apSigRealmStatsTotalSessionsInbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.7)
  - Total number of inbound sessions during the last 100 second sliding window period for this realm
- apSigRealmStatsPeriodHighInbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.9)
  - Highest number of concurrent inbound sessions during the last 100 second sliding window period for this realm
- apSigRealmStatsTotalSessionsOutbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.11)
  - Total number of outbound sessions during the last 100 second sliding window period for this realm
- apSigRealmStatsPeriodHighOutbound (1.3.6.1.4.1.9148.3.2.1.2.4.1.13)
  - Highest number of concurrent outbound sessions during the last 100 second sliding window period for this realm.
- apSigRealmStatsMaxBurstRate (1.3.6.1.4.1.9148.3.2.1.2.4.1.15)
  - Maximum burst rate of traffic measured during the last 100 second sliding window period (combined inbound and outbound) for this realm
- apSigRealmStatsPeriodASR (1.3.6.1.4.1.9148.3.2.1.2.4.1.18)

    - The answer-to-seizure ratio expressed as a percentage during the 100 second sliding window. For example, a value of 90 would represent 90% or 0.90
- apSigRealmStatsRealmStatus (1.3.6.1.4.1.9148.3.2.1.2.4.1.30)
    - State of the specified realm (INS, constraintviolation, or callLoadReduction)

The same list of statistics is also available per Session Agent.

Environmental Statistics

Below is a recommended list of SNMP OIDs to GET every 5 minutes from the Oracle System Environment Monitor MIB (ap-env-monitor.mib). These will provide useful system environmental data.

apEnvMonObjects (1.3.6.1.4.1.9148.3.3.1)

- apEnvMonI2CState (1.3.6.1.4.1.9148.3.3.1.1)
    - State of environmental sensor on system chassis. A value of 2 is normal, all others need to be investigated further [4].

apEnvMonTemperatureStatusEntry (1.3.6.1.4.1.9148.3.3.1.3.1.1)

- apEnvMonTemperatureStatusValue (1.3.6.1.4.1.9148.3.3.1.3.1.1.4)
    - Current temperature of mainboard PROM (in Celsius).
- apEnvMonTemperatureState (1.3.6.1.4.1.9148.3.3.1.3.1.1.5)
    - State of system temperature. A value of 2 is normal, all others need to be investigated further [4].

Enterprise SNMP OIDs in a High Availability (HA) environment

SNMP polling is recommended for both Active and Standby SDs. The data from both Active and Standby will be useful when troubleshooting. Also some data such as CPU, memory, interface stats, health score, etc is independent data between Active and Standby SDs.

Of particular interest is the mib-system-name field in the system-config object. This is used as part of the MIB-II sysName identifier. When performing a MIB-II sysName poll, an SBC will return a concatenation of its assigned target name (as specified in the boot params), a dot, and the common hostname from the system-config. Thus, systems in an HA pair named “acme1” and “acme2”, when assigned a mib-system-name of “sbc.bedford”, would return acme1.sbc.bedford and acme2.sbc.bedford when polled, respectively.

Enterprise SNMP Traps

SNMP traps enable an SNMP agent to notify the Network Management System (NMS) of significant events using an unsolicited SNMP message. The SBC can be configured to send unsolicited SNMP traps to a configured SNMP trap receiver [4] in SNMPv1, SNMPv2c and SNMPv3 formats. The snmp-agent-mode determines the trap format, and is set under system-config.

SNMP Configuration recommendations

Under the system-config element the following settings should be enabled to provide additional visibility to system events:

- enable-snmp-auth-traps – sends a trap for a failed authentication as part of an SNMP request; used to detect abuse
- enable-snmp-syslog-notify – enable syslog conversion to SNMP
- enable-snmp-monitor-traps – enable unique trap-IDs for each syslog event

The SBC setting to enable environmental monitors may seem advantageous, but is not recommended, and should remain at the default – disabled. The same traps are already sent as part of the ap-smgmt MIB.

- enable-env-monitor-traps – sends traps for environmental issues like temperature, voltage, fan speeds, etc.

SNMP Traps

The following are a selection of the most common or important traps sent by the SD. The full list of traps can be found in the MIB Reference Guide and MIB files for the release implemented:

- apSwCfgActivateNotification (1.3.6.1.4.1.9148.3.4.3.0.1)
    - Generated when the activate-config command is issued at ACLI and configuration has been changed at run time. This trap may be seen often but is only informational and doesn't necessarily mean there is an issue (unless the config changes are service affecting or the change was not authorized).
- apEnvMonStatusChangeNotification (1.3.6.1.4.1.9148.3.3.2.1.0)
    - Generated when the environmental state of the SBC changes. Environment traps include main board PROM temperature, CPU voltage, state of power supplies, fan speeds, etc. To receive this trap, the system-config parameter enable-env-monitor-traps needs to be enabled. An example of this trap for voltage state change is found in [4].
- apSysMgmtGroupTrap (1.3.6.1.4.1.9148.3.2.3.0.1)
    - Generated when an SBC resource threshold or health score is exceeded. For example, if NAT table usage, ARP table usage, memory usage, or CPU usage reaches 90% or greater. Also, If the health score for an HA pair falls below 60.
- apSysMgmtExpDOSTrap (1.3.6.1.4.1.9148.3.2.8.0.2)
    - Generated when an endpoint exceeds configured thresholds and is denied access by the SD.
- apSysMgmtInetAddrWithReasonDOSTrap (1.3.6.1.4.1.9148.3.2.8.0.4)
    - This trap is generated when the IDS Reporting Feature Group license (available in S-CX6.3 and above) is installed. This trap is generated when thresholds are exceeded, and contains further data on the end-point and reason why the trap was generated. When IDS Reporting is installed the apSysMgmtExpDOSTrap is disabled.
- apSysMgmtInetAddrTrustedToUntrustedDOSTrap (1.3.6.1.4.1.9148.3.2.8.0.5.)
    - This trap is available in S-C[xz]6.4.0 and above. It will be generated when the number of rejected messages exceeds the configured threshold and the endpoint is demoted from the trusted to untrusted list. The trap-on-demote-to-untrusted setting under media-manager must be enabled for this trap to be sent.
- apSysMgmtRejectedMesagesThresholdExeededTrap (.1.3.6.1.4.1.9148.3.2.6.0.57)
    - This trap is available in S-C[xz]6.4.0 and above. A trap will be generated when the number of rejected messages exceed the configured threshold and the endpoint is put into the untrusted queue.
- apSysMgmtSipRejectionTrap (.1.3.6.1.4.1.9148.3.2.10.0.1)
    - Generated when a SIP INVITE or REGISTRATION request fails
- apSysMgmtPowerTrap (1.3.6.1.4.1.9148.3.2.6.0.1)
    - Generated if a power supply is powered down, powered up, inserted (present) or removed (not present).
- apSysMgmtTempTrap (1.3.6.1.4.1.9148.3.2.6.0.2)
    - Generated if the system temperature falls below the monitoring level.
- apSysMgmtFanTrap (1.3.6.1.4.1.9148.3.2.6.0.3)
    - Generated if a fan unit speed falls below the monitoring level.

- apSysMgmtTaskSuspendTrap (1.3.6.1.4.1.9148.3.2.6.0.4)
    - Generated if a critical task running on the system enters a suspended state.
- apSysMgmtRedundancyTrap (1.3.6.1.4.1.9148.3.2.6.0.5)
    - Generated if either the primary or secondary SBC in a HA pair changes state.
- apSysMgmtMediaPortsTrap (1.3.6.1.4.1.9148.3.2.6.0.6)
    - Generated if port allocation fails at a percentage higher or equal to the system's default threshold rate. Port allocation failure rates are checked every 30 seconds. The trap is sent when the failure rate is at 50% or higher. After that time, the trap is sent every 30 seconds until the failure rate drops below 35%. The clear trap is sent once the failure rate drops below 5%.
- apSysMgmtMediaBandwidthTrap (1.3.6.1.4.1.9148.3.2.6.0.7)
    - Generated if bandwidth allocation fails at a percentage higher or equal to the system's default threshold rate. Bandwidth allocation failure rates are checked every 30 seconds. The trap is sent when the failure rate is at 50% or higher. After that time, the trap is sent every 30 seconds until the failure rate drops below 35%. The clear trap is sent once the failure rate drops below 5%.
- apSysMgmtPhyUtilThresholdTrap (1.3.6.1.4.1.9148.3.2.6.0.66)
    - Generated when the media port's utilization crosses a configured threshold. If overload protection is enabled, new requests will be refused when the threshold reaches a critical value. Thresholds can be configured for minor, major, and critical.
- apSysMgmtGatewayUnreachableTrap (1.3.6.1.4.1.9148.3.2.6.0.10)
    - Generated if the SBC cannot reach a configured gateway. Only applicable when gateway heartbeat feature is configured [7].
- apSysMgmtRadiusDownTrap (1.3.6.1.4.1.9148.3.2.6.0.11)
    - Generated if any configured RADIUS accounting server becomes unreachable.
- apSysMgmtSAStatusChangeTrap (1.3.6.1.4.1.9148.3.2.6.0.15)
    - Generated when a session agent is declared unreachable or unresponsive for the following reasons:
        - signaling timeout (H.323 and SIP)
        - session agent does not respond to SIP pings (SIP only)
    - This causes the session agent to be placed out-of-service for a configurable period of time.
- apSysMgmtInterfaceStatusChangeTrap (1.3.6.1.4.1.9148.3.2.6.0.26)
    - Generated when the SIP interface status changes from in service or constraints have been exceeded.
        - apSysMgmtSipInterfaceRealmName — Realm identifier for the SIP interface (OID 1.3.6.1.4.1.9148.3.2.5.24)
        - apSysMgmtSipInterfaceIP — IP address of the first SIP port in the SIP interface (OID 1.3.6.1.4.1.9148.3.2.5.25)
        - apSysMgmtSipInterfaceStatus — Code is 0 (OID 1.3.6.1.4.1.9148.3.2.5.26)
        - apSysMgmtSipInterfaceStatusReason — Status reasons are in-service (3) and constraintExceeded (4) (OID 1.3.6.1.4.1.9148.3.2.5.27)
- apSysMgmtNTPServerUnreachableTrap (1.3.6.1.4.1.9148.3.2.6.0.30)
    - Generated if the NTP server becomes unreachable.
        - apSysMgmtNTPServer—Server that is unreachable (OID 1.3.6.1.4.1.9148.3.2.5.31)
- apLicenseApproachingCapacityNotification (1.3.6.1.4.1.9148.3.5.3.0.1)
    - Generated when the total number of active sessions on the system (across all protocols) is within 98 - 100% of the licensed capacity
- apSysMgmtAuthenticationFailedTrap (1.3.6.1.4.1.9148.3.2.6.0.16)

    - Generated when an attempt to login to the Net-Net SBC through telnet, SSH, or by using the console fails for any reason
- apSysMgmtAdminAuthLockoutTrap (1.3.6.1.4.1.9148.3.2.6.0.64)
    - Generated upon system lockout after multiple authentication failures.

SNMP Traps in HA environment

Once the trap-receiver has been configured, the SBC will monitor and send a trap according to the configured filter-level value. The same trap receiver is used by both units in an HA pair since the configuration is synchronized between the two. Furthermore, the Active or Standby SBC will send a trap independently if it is related to the hardware, interface status, gateway reachability, temperature, etc.

Appendix I: Syslog

The SBC can be configured to send system event logs to logging servers [1]. It is recommended to configure as few logging servers as required to reduce impact on SBC performance. Monitoring via SNMP is the preferred option over using syslog. The syslog messages aren't as efficient since they may contain many extraneous informational messages that need to be filtered out or parsed. SNMP on the other hand has the advantage of sending clearly defined trap notifications only in the event of a problem, and the system-config and trap-receiver settings can be configured to filter on specific SNMP traps to send.

If a syslog parser is used to escalate SBC issues, it is easy to classify syslog events preceded with a MAJOR or CRITICAL designation as issues that require further investigation. However, be cautious of writing any parsing rules for events that are classified as GENERAL, REDUNDANCY, CONFIG WARNING, ERROR, or MINOR (among others). Some of these may be important to escalate, but others may be strictly informational in nature.

The table below represents a sample of some of the common syslog messages that may be seen. Note that IDS_LOG examples given require the IDS Reporting Feature Group license discussed in Appendix F. Some of the examples may seem redundant. This is because in some cases more than one message may be written to syslog as a result of an event.

Description	**Raw message**

A failed login attempt was detected on the console port.	`May  3 17:06:10 172.41.3.90 CSE-4500-20 acliConsole[31ac9b6c] AUTH authenticate_locally: Authentication failed for user user`
	OR
	`Mar 20 10:27:24.119 acliConsole@: AUTH[53] authenticate_locally: Authentication failed for user user`
	OR
	`May  3 17:06:10 172.41.3.90 CSE-4500-20 acliConsole[31ac9b6c] WARNING login authentication failure from acliConsole`

A failed login attempt was detected over telnet, SSH or FTP. See the process information for further details.	`Dec 18 13:39:44.121 acliTelnet0@SBC1: WARNING authentication failure for admin from acliTelnet0`

Description	Raw message
An endpoint exceeded a defined constraint and was blacklisted. This is the result of DoS configuration with the IDS license.	Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR [IDS_LOG] SigAddr[access:192.168.101.120:0=low:DENY] ttl=86400 exp=30 Demoted to Black-List (Too many messages) last msg rcvd=REGISTER sip:192.168.66.2 SIP/2.0 Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Via: SIP/2.0/UDP 192.168.190.144:20928;branch=z9hG4bKdeadb33f Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR From: hacker <sip:47097@192.168.190.144:20928> Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR To: <sip:47097@192.168.66.2:5060> Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Call-ID: f9844fbe7dec140ca36500a0c9119870@192.168.66.2 Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR CSeq: 1 REGISTER Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Contact: <sip:47097@192.168.190.144> Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR User-agent: Flooder_script Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Max-Forwards: 5 Nov 28 17:53:47 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR Content-Length: 0
An endpoint exceeded a defined constraint and was blacklisted. This message is a result of DoS configuration without the IDS license.	Jan 15 16:29:46.289 sipd@SBC1: FLOW[15] SigAddr[Access:192.168.135.29:0=low:DENY] ttl=86400 guard=50 exp=30 Demoted to Black-List; send SNMP trap
An endpoint exceeded a defined constraint and was demoted from trusted to untrusted.	Apr 1 11:36:53.377 sipd@CSE-4500-6: WARNING SigAddr[access:172.41.0.3:5060=medium:PERMIT] ttl=64 exp=57 Demoted to Grey-List (errors)

Description	Raw message
The sipShield SPL plug-in (v1.3) detected a message from a known SIP scanner and dropped it	Mar 28 15:05:42.500 sipd@CSE-4500-6: WARNING Scanner or attack field detected! Src IP: 172.41.0.3, User-Agent: smap 0.6.0 OR Mar 28 15:05:42.500 sipd@CSE-4500-6: WARNING Scanner or attack field detected! Src IP: 172.41.0.3, To: victim@example.edu OR Mar 28 15:05:42.500 sipd@CSE-4500-6: WARNING Scanner or attack field detected! Src IP: 172.41.0.3, From: user@example.edu OR Mar 28 15:05:42.500 sipd@CSE-4500-6: WARNING Scanner or attack field detected! Src IP: 172.41.0.3, Subject: SiVuS

Description	Raw message

A message was rejected by the SD. The status code and reason given in parenthesis will change based on the type of malformation. Examples given here include:
An INVITE received from a forbidden endpoint. In this case, allow-anonymous on the SIP interface was set to agents-only, and the INVITE was not from an agent.
An INVITE had a Max-Forwards parameter that had decremented to zero, and the SBC could not forward it further
Four examples of malformed messages that were generated from a Protos attack (too large, missing header, bad request URI, unsupported URI).
	`Apr  1 11:26:27.603 sipd@CSE-4500-6: IDS[64] [IDS_LOG]INVITE from source 172.41.0.3:5060 to dest 172.41.0.2:5060[UDP] realm=access; From=sipp <sip:sipp@127.0.1.1:5060>;tag=10387SIPpTag001; target=sip:service@172.41.0.2:5060 rejected!; status=403 (Forbidden)`
	OR
	`Nov 28 19:52:40 172.41.3.41 CSE-4500-6 sipd[2dcc32a4] ERROR [IDS_LOG]INVITE from source 192.168.66.54:5060 to dest 192.168.66.2:5060[UDP] realm=access; From="hacker"<sip:666@192.168.66.54:30000>; target=sip:9195551212@192.168.66.2 rejected!; status=483 (Too Many Hops); error=invalid message`
	OR
	`IDS_LOG]INVITE from source 192.168.222.1:5060 to dest 192.168.222.50:5060[UDP] realm=access; From=227 <sip:evil@127.0.1.1>;tag=227; target=sip` <omitted message> `rejected!; status=513 (Message Too Big)`
	OR
	`May 22 14:40:39.033 sipd@: IDS[64] [IDS_LOG]INVITE from source 192.168.222.1:5060 to dest 192.168.222.50:5060[UDP] realm=access; From=389 <sip:evil@127.0.1.1>;tag=389; target=sip:1111@192.168.222.50 rejected!; status=400 (Invalid/Missing Via Header)`
	OR
	`May 22 15:08:02.015 sipd@: IDS[64] [IDS_LOG]INVITE from source 192.168.222.1:5060 to dest 192.168.222.50:5060[UDP] realm=access; From=206 <sip:evil@127.0.1.1>;tag=206; target=%s%s%s%s%s:noone@sip.no.invalid rejected!; status=400 (Bad Request-URI)`
	OR
	`May 22 15:08:01.088 sipd@: IDS[64] [IDS_LOG]INVITE from source 192.168.222.1:5060 to dest 192.168.222.50:5060[UDP] realm=access; From=197 <sip:evil@127.0.1.1>;tag=197; target=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa:noone@sip.no.invalid rejected!; status=416 (Unsupported URI Scheme)`

A user entered enable mode (administrator level). This is not necessarily an issue, but may be an interesting event.	`May  3 17:06:37 172.41.3.90 CSE-4500-20 acliConsole[31ac9b6c]  raised privileges on session from acliConsole`

Description	Raw message
A user enabled SIP debugging traces. This can use large amounts of CPU if run on a production network or potentially reveal sensitive information. This is not necessarily an issue, but may be an interesting event.	May 3 17:09:26 172.41.3.90 CSE-4500-20 sipd[2fa7cc00] SIP enable SIP Debugging
The configuration file was updated. This should be investigated if changes were not authorized.	Dec 19 13:28:27.060 lemd@SBC1: CONFIG[32] Save Config has completed successfully
A new configuration was activated. This should be investigated if changes were not authorized.	Dec 19 13:28:29.863 lemd@SBC1: CONFIG[32] Configuration successfully activated OR Dec 19 13:28:31.864 lemd@SBC1: CONFIG[32] Activate Config Successfully Complete OR Mar 20 10:11:02.919 acliSSH0@: CONFIG[34] ACTIVATE-CONFIG done
One or more licenses has expired and unit functionality may be impacted	Apr 1 00:00:10.523 brokerd@CSE-4500-6: MINOR ALARM[00050016] Task[0615c064] 1 license has expired!
One or more licenses is nearing expiration	Mar 31 00:00:10.521 sysmand@CSE-4500-6: MINOR License will expire in less than 7 days.
The number of sessions is approaching licensed capacity	Jan 1 00:02:57.480 brokerd@SBC1: MAJOR ALARM[00050004] Task[0cf72188] total number of sessions (1977) is approaching licensed capacity (2000)
The unit was powered on. This may be an indication that a power failure occurred.	Jan 8 11:33:06.545 bootstrap@SBC1: GENERAL[0] Bringing up box...
The SIP protocol stack is now active. This may be an indication that a power failure occurred or that the SIP process crashed and restarted.	May 3 17:30:08 172.41.3.90 CSE-4500-20 sipd[2fa7cc00] SIP Change to In-Service state and Start accepting messages...
Unit CPU usage has reached a critical threshold	Oct 8 19:02:02.381 brokerd@SBC1: CRITICAL ALARM[0002001b] Task[0578324c] cpu usage 93 percent is over critical threshold of 90 percent.

Description	**Raw message**
Unit CPU usage has reached a major threshold	`Oct  8 19:02:12.708 brokerd@SBC1: MAJOR ALARM[0002001b] Task[0578324c] cpu usage 87 percent is over major threshold of 80 percent.`
Unit CPU usage has reached a minor threshold	`Oct  8 19:06:57.062 brokerd@SBC1: MINOR ALARM[0002001b] Task[0578324c] cpu usage 74 percent is over minor threshold of 70 percent.`
A high-availability switchover was detected from the active unit. If this was not an administrative failover then it is likely that a port or process failed	`Dec  3 17:30:46.275 berpd@SBC1: CRITICAL ALARM[00020021] Task[2834f658] Switchover, Active to RelinquishingActive`
The standby unit has become the active unit. If this was not a result of an administrative action then a port or process on the active unit likely failed.	`Jan  8 11:34:41.652 berpd@SBC1: CRITICAL ALARM[00020020] Task[03c3a840] Switchover, Standby to BecomingActive, active peer SBC2 has timed out`
The standby unit is having difficulty reaching the active unit. Verify that all wancom ports are operational.	`Dec  3 17:33:46.384 berpd@SBC1: CRITICAL ALARM[00020023] Task[2834f658] Unable to synchronize with Active redundant peer within BecomingStandby timeout, going OutOfService`
A ethernet port used for management has gone down	`Jan  8 11:34:42.171 brokerd@SBC1: MAJOR ALARM[00020009] Task[0e723a98] wancom1 link down`
A ethernet port used for management has recovered from failure	`Jan  8 11:34:44.788 brokerd@SBC1: MINOR ALARM[00020006] Task[0e723a98] wancom1 link up`
An ethernet port used for services has gone down. Note that slot and port numbers will vary.	`Mar 20 21:56:29.504 brokerd@: MAJOR ALARM[00020027] Task[00000003] Slot 1 Port 0 DOWN`
All servers that can receive accounting files (CDR) are not available	`May  3 17:20:11 172.41.3.90 CSE-4500-20 brokerd[10661b38] CRITICAL All of collector's push receivers are down`
Transfer of an HDR file failed because the key used for authentication is incorrect	`May  3 17:20:11 172.41.3.90 CSE-4500-20 collect[2eb37454] WARNING Error: HDR push failed due to bad host key.`
An error occurred when attempting to transfer accounting logs	`Dec 31 07:47:53.192 collect@SBC1: MINOR Error pushing collected data to 172.17.5.24 for group: system`
Transfer of an HDR file failed due to invalid authentication	`May  3 17:20:11 172.41.3.90 CSE-4500-20 collect[2eb37454] ERROR Error: Could not login to host '172.41.1.118'`

Description	**Raw message**
Media port usage is exceeding capacity. Calls may fail or experience audio issues. The severity is based on the percentage of failures to allocate a steering port.	Jan 17 12:14:26.513 mbcd@SBC1: MINOR ALARM[00040006] Task[1b963548] out of steering ports for realm 'CORE'; 296 of 592 failed (50%) OR Jan 17 12:18:14.865 mbcd@SBC1: WARNING ALARM[00040006] Task[1b963548] out of steering ports for realm 'CORE'; 80 of 310 failed (25%)
A session agent (SIP server) has failed a health check and has been taken out of service	Jan 15 16:28:19.901 sipd@SBC1: SIP[13] SA 192.168.136.69[PBX1]PING TRANSACTION TIMEOUT to 192.168.136.69 Jan 15 16:28:19.902 sipd@SBC1: SIP[13] was 'In Service'; set to 'Out of Service' status
A session agent (SIP server) has failed a health check and has been taken out of service	Jan 15 16:28:22.969 sipd@SBC1: SIP[13] SA 192.168.135.29[PBX2]Non-Ping TRANSACTION TIMEOUT to 192.168.135.29 Jan 15 16:28:22.970 sipd@SBC1: SIP[13] was 'In Service'; set to 'Out of Service' status
There were no routes found for an incoming session. This may mean that the called destination is out of service, the destination address is incorrect, or that the routing table is not sufficient.	Mar 30 15:02:27.307 sipd@CSE-4500-6: IDS[64] [IDS_LOG]INVITE from source 192.168.60.10:5061 to dest 192.168.60.2:5060[UDP] realm=core; From=sipp <sip:sipp@127.0.0.1:5061>;tag=9165SIPpTag00143 ; target=sip:service@192.168.60.2:5060 rejected!; status=480 (No Routes Found)

The next table contains partial syslog message samples that can be used for parsing. An actual message sample is not provided due to the fact that the condition was either difficult or time consuming to simulate, or would have been destructive to equipment.

Description	**Partial Message**
Error setting fan speed	Error setting fan speed:
Error reading an internal temperature sensor. There may be a motherboard issue.	Error reading LM75 Device!
Error reading an internal temperature sensor. There may be a motherboard issue.	Error reading COM-Express CPU Temperature!
Error reading an internal temperature sensor. There may be a motherboard issue.	Error reading Mainboard Temperature
Error reading an internal temperature sensor. There may be a motherboard issue.	Error reading PHY Temperature

Physical port usage exceeding defined critical thresholds	Slot 0 Port 0 PHY utilization xx% is over critical threshold of xx%.
Physical port usage exceeding defined major thresholds	Slot 0 Port 0 PHY utilization xx% is over major threshold of xx%.
Physical port usage exceeding defined minor thresholds	Slot 0 Port 0 PHY utilization xx% is over minor threshold of xx%.
Disk space exceeding critical threshold on specified volume	Volume <name> space used xx% is over critical threshold of xx%.
Disk space exceeding major threshold on specified volume	Volume <name> space used xx% is over critical threshold of xx%.
Disk space exceeding minor threshold on specified volume	Volume <name> space used xx% is over critical threshold of xx%.
Critical memory threshold exceeded	memory usage xx percent is over critical threshold of xx percent.
Major memory threshold exceeded	memory usage xx percent is over major threshold of xx percent.
Minor memory threshold exceeded	memory usage xx percent is over minor threshold of xx percent.
High availability health score is under threshold	health score is at xx (under threshold of xx)
ARP table usage is over threshold	ARP table capacity xx percent is over threshold xx percent.
NAT table usage is over threshold	NAT table capacity xx percent is over threshold xx percent.
Critical session usage threshold	session usage xx percent is over critical threshold of xx percent.
Major session usage threshold	session usage xx percent is over major threshold of xx percent.
Minor session usage threshold	session usage xx percent is over minor threshold of xx percent.
Critical deny ACL allocation threshold	Deny Allocation xx percent is over critical threshold of xx percent.
Major deny ACL allocation threshold	Deny Allocation xx percent is over major threshold of xx percent.
Minor deny ACL allocation threshold	Deny Allocation xx percent is over minor threshold of xx percent.

Call Detail Records (CDR)

The SBC can be configured to send Accounting CDR packets to a RADIUS server [3]. This data can be trended for monitoring purposes as well as traditional billing uses. For example, determining call completion rates at various high and low points during the day. This approach requires the implementation of a RADIUS server with the release specific Oracle Radius Dictionary (available in Downloads on the support portal), and a database backend / reporting mechanism.

If RADIUS is not the desired approach, the SBC store CDRs locally and then push them to a FTP/SFTP server on a scheduled basis [3]. Since that may require more local storage, an optional Storage Expansion Module can be used to extend the storage of CDRs locally on the SBC [3].

The following is a list of useful RADIUS attributes for characterizing and troubleshooting the VoIP network:

- Acct-Session-Time (46)
    - Call Duration: useful for detecting abnormally short or long call hold times
- Acct-Terminate-Cause (49) & Acme-Disconnect-Cause (62)
    - Call Disconnect Cause Code: useful for detecting abnormal call failures
- See [3] for list of causes
    - RTCP & RTP QoS Statistics – The SBC can incorporate call quality measurements (QoS) in CDR and for up to two RTP/RTCP bidirectional media flows per SIP session. Each bidirectional flow is referred to as a flow-set (FS1 and FS2). QoS collection requires an NIU with QoS capabilities and a QoS measurement setting to be enabled in configuration.
    - Acme-Called-RTCP-Packets-Lost_FS1 / FS2 (46 / 104) – integer, total for call
    - Acme-Called-RTCP-Avg-Jitter_FS1 / FS2 (47 / 105) – measured in ms
    - Acme-Called-RTCP-Avg-Latency_FS1 / FS2 (48 / 106) – measured in ms
    - Acme-Called-RTCP-MaxJitter_FS1 / FS2 (49 / 107) – measured in ms
    - Acme-Called-RTCP-MaxLatency_FS1 / FS2 (50 / 108) – measured in ms
    - Acme-Called-RTP-Packets-Lost_FS1 / FS2 (51 / 109) – integer, total for call
    - Acme-Called-RTP-Avg-Jitter_FS1 / FS2 (52 / 110) – measured in ms
    - Acme-Called-RTP-MaxJitter_FS1 / FS2 (53 / 111) – measured in ms
- Acme-Post-Dial-Delay (58)
    - Call Setup time in ms: Detect abnormal delays between SIP INVITE and 180 Ringing
- Acme-Session-Disposition (60)
    - Status of call attempt from SIP INVITE to answered or unanswered
    - 0 = unknown, 1 = call attempt, 2 = ringing, 3 = answered
- Acme-Disconnect-Initiator (61)
    - Party that disconnects the call
    - 0 = unknown, 1 = calling party, 2 = called party, 3 = internal

Appendix J: Historical Data Records (HDR)

HDR refers to a management feature that collects statistics about SBC system operation and function, and then sends those records to a configured FTP/SFTP server [1]. This is roughly the same data available via SNMP, but collected and stored in CSV files on configured intervals, and then sent to the server on a configured period. These files can be used for capacity planning and analysis of trends or long term issues.

HDR data consists of a “Group” with associated “Group Statistics” that apply to each group. HDR data comes from two sources: SNMP MIBs and Oracle’s Command Line Interface (ACLI), i.e. the output of show commands.

For more information and details about Historical Data Recording please read the HDR Resource Guide applicable to the release on the SD.

Specific HDR groups of interest to collect are:

- system
  - global system statistics
- session-realm
  - session and rate statistics on a per realm basis
- temperature
  - environmental temperature statistics
- sip-sessions
  - SIP status statistics
- sip-errors
  - error statistics for SIP, media, and SDP
- sip-policy
  - SIP routing, session-agent groups and constraints statistics
- sip-ACL-status
  - statistics on trusted and blocked ACLs

The following is an example of information collected in the sip-errors HDR for one collection interval.

```
TimeStamp,Message/Event,Server Totals,Client Totals
1369336364,INVITE Requests,1200,0
1369336364,Retransmissions,0,0
1369336364,100 Trying          ,800,0
1369336364,180 Ringing         ,800,0
1369336364,181 Forwarded       ,0,0
1369336364,182 Queued          ,0,0
1369336364,183 Progress        ,0,0
1369336364,1xx Provisional     ,0,0
1369336364,200 OK              ,800,0
1369336364,202 Accepted        ,0,0
1369336364,2xx Success         ,0,0
1369336364,30x Moved           ,0,0
1369336364,305 Use Proxy       ,0,0
1369336364,380 Alternative     ,0,0
1369336364,3xx Redirect        ,0,0
1369336364,400 Bad Request     ,200,0
1369336364,401 Unauthorized    ,0,0
1369336364,403 Forbidden       ,200,0
1369336364,404 Not Found       ,0,0
```

```
1369336364,405 Not Allowed     ,0,0
1369336364,406 Not Acceptable  ,0,0
1369336364,407 Proxy Auth Req  ,0,0
1369336364,408 Request Timeout ,0,0
1369336364,415 Bad Media Type  ,0,0
1369336364,420 Bad Extension   ,0,0
1369336364,421 Extension Reqd  ,0,0
1369336364,422 Too Short       ,0,0
1369336364,423 Too Brief       ,0,0
1369336364,480 Unavailable     ,0,0
1369336364,481 Does Not Exist  ,0,0
1369336364,482 Loop Detected   ,0,0
1369336364,483 Too Many Hops   ,0,0
1369336364,484 Address Incompl ,0,0
1369336364,485 Ambiguous       ,0,0
1369336364,486 Busy Here       ,0,0
1369336364,487 Terminated      ,0,0
1369336364,488 Not Acceptable  ,0,0
1369336364,489 Bad Event       ,0,0
1369336364,491 Req Pending     ,0,0
1369336364,4xx Client Error    ,0,0
1369336364,500 Internal Error  ,0,0
1369336364,501 Not Implemented ,0,0
1369336364,502 Bad Gateway     ,0,0
1369336364,503 Service Unavail ,0,0
1369336364,504 Gateway Timeout ,0,0
1369336364,513 Msg Too Large   ,0,0
1369336364,580 Precon Failure  ,0,0
1369336364,5xx Server Error    ,0,0
1369336364,600 Busy Everywhere ,0,0
1369336364,603 Decline         ,0,0
1369336364,604 Not Found       ,0,0
1369336364,606 Not Acceptable  ,0,0
1369336364,6xx Global Error    ,0,0
1369336364,Response Retrans,0,0
1369336364,Transaction Timeouts,  -  ,0
1369336364,Locally Throttled,  -  ,0
```

Appendix K: ACLI Commands for Monitoring

Data available via HDR, SNMP, CDR, or Syslog is usually sufficient for analysis and troubleshooting. However, some ACLI show and display commands have additional data that is not available with those methods. The commands referenced here are some of the most common ones that should be used for troubleshooting and additional statistics collection (most commonly automated with a script). It should be noted that Oracle may update the fields or format used in these commands periodically as enhancements are made, so automation may not function correctly after an upgrade.

System Statistics

Below is a recommended list of ACLI commands to execute every 5 minutes on the Active SBC in addition to the SNMP OID polling (outlined in Section 0). These will provide useful data on overall system performance [2].

- display-alarms
  - View active alarms reported on the system
- show health
  - Verify active/standby system health and switchover alarms reported
- show arp
  - Verify all configured gateways are reachable
- show media physical
  - Displays statistics for media ports
- show media utilization
  - Percentage bandwidth utilization for each media port
- show accounting
  - Displays statistics for configured RADIUS servers and CDRs sent
- show acl summary
  - Displays statistics for system DDOS activity
- show acl info
  - Displays statistics for ACL usage of CAM space
- check-space-remaining [code | ramdrv]
  - Displays flash space available for the directories code and ramdrv

Application Statistics

Below is a recommended list of ACLI commands to execute every 5 minutes on the Active SBC in addition to the SNMP OID polling (outlined in Section 0). These will provide useful data on application performance [2].

SIP

- show registration
  - Verify no unexpected spikes or drops in expected number of concurrently registered endpoints
- show sip invite
  - Displays statistics for SIP INVITE messages received/sent by the SD. Important to monitor 4xx and 5xx response errors and retransmissions.
- show sipd agents
  - Displays statistics for all session-agents configured
- show sipd realms
  - Displays statistics for each realm

- show sipd errors
    - Error count related to SIP

Media

- show mbcd realms
    - Displays media (RTP) related information presented in a per-realm manner
- show mbcd errors
    - Error count related to media

H.323

- show sessions
    - Displays the concurrent sessions active on the system. It further details the number of sessions for an IWF scenario, as being SIP-to-H.323 and H.323-to-SIP.
- show h323 stackCallstats
    - Displays summary of H.323 call Stats for all stacks.
- show h323 stackDisconnectInstats
    - Displays summary of H.323 pvt Stats for all stacks Incoming
- show h323 stackDisconnectOutstats
    - Displays summary of H.323 pvt Stats for all stacks Outgoing
- show h323 stackPvtstats
    - Displays summary of H.323 pvt Stats for all stacks
- show h323 agentstats
    - Displays summary of all H.323 Session Agents

MGCP

- show algd all
    - Displays MGCP statistics.

Appendix L: SRTP Configuration and Troubleshooting

Introduction

The Secure Real-time Transport Protocol (SRTP) provides encryption and authentication for the call content and call signaling streams. Authentication provides assurance that packets are from the purported source, and that the packets have not been tampered with during transmission. Encryption provides assurance that the call content and associated signaling has remained private during transmission. SRTP/SDES is supported on the Net-Net 3800 and 4500 SBC.

RTP and RTCP traffic are encrypted as described in RFC 3711: The Secure Real-time Transport Protocol (SRTP). The negotiation and establishment of keys and other cryptographic materials that support SRTP is described in RFC 4568: Session Description Protocol (SDP) Security Description for Media Streams. Cryptographic parameters are established with only a single message or in single round-trip exchange using the offer/answer model defined in RFC 3264: An Offer/Answer Model with the Session Description Protocol.

Session Description Protocol Security Descriptions for Media Streams (SDES), defined in RFC 4568, provide alternative methods for creating keys used to encrypt Real-time Transport Protocol (RTP) and Real-time Transport Control Protocol (RTCP) transactions.

This document should be used as a base reference only, outlining procedures to configure SRTP on the SBC node from its base configuration. An Oracle Systems Engineer should be consulted with regards to specific concerns as they apply to customer specific SBC configurations.

Configuration guides are available for download from the Oracle Customer Support Portal (https://support.acmepacket.com). Please contact your Oracle Systems Engineer for Best Current Practice (BCP) documentation.

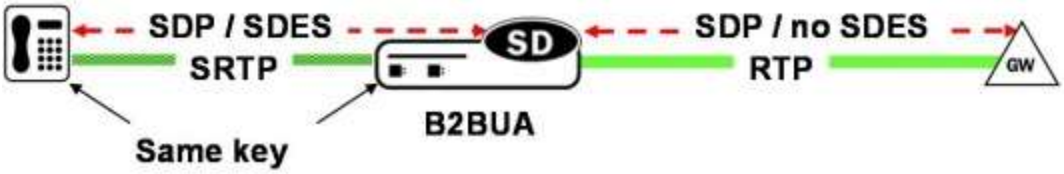
SRTP Topologies

End-to-end SRTP was supported in previous releases, and the SBC was transparent to the SRTP key negotiation and the SRTP flow. It was just adding its own IP to the media path and then relaying the SRTP packets as it does with RTP flows, so in terms of functionality, RTP and SRTP caused no difference in the SBC configuration and functionality.

However, release S-CX6.2.0 and higher includes support for termination of SRTP. This includes special configuration and treatment of RTP and SRTP flows.

SRTP topologies can be reduced to three basic topologies:

- **Single Ended SRTP Termination**
  SRTP enabled on inbound interface, disabled on outbound interface (or vice versa)

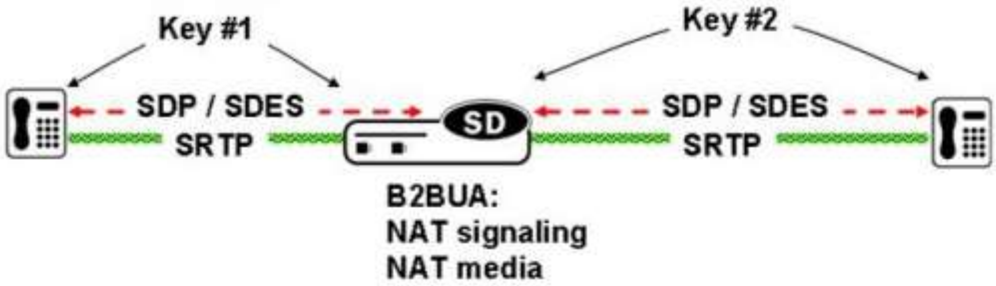


If SRTP is enabled for the inbound realm/interface, the SBC will handle the request according to the capabilities defined in the SRTP configuration. If there is a crypto attribute in the offer, the SBC will attempt to parse the crypto attributes and parameters in the SDP. It accepts exactly one of the offered crypto attributes for a given media stream, if this is configured as a valid crypto-suite on the SD. If there is no crypto-suite configured on the SBC in the list of crypto-suites received, the SBC will reject the call with a “488 Not Acceptable Here” response.

Before the request is forwarded to the callee, the SBC allocates resources, updates the SDP with proper media addresses and ports, and the original crypto attribute is removed from the SDP.

Once the reply from the callee is received, SBC inserts the appropriate crypto attribute to form a new SDP, and forwards the response back to the caller. At this point, SRTP traffic is allowed between the caller and the SD.

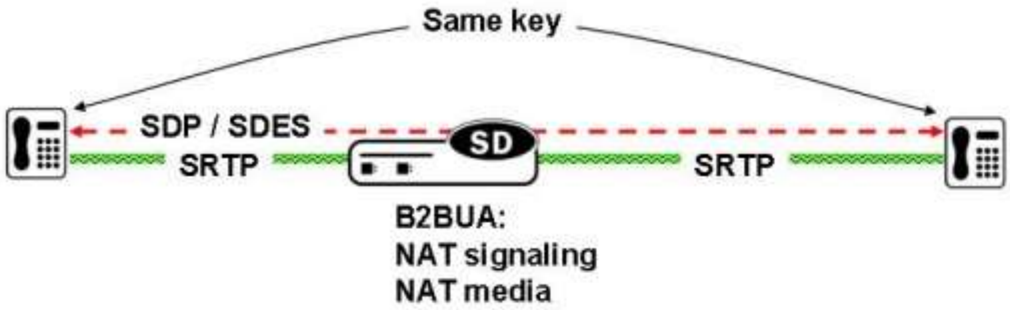
- **Back-to-back SRTP Termination**
  SRTP enabled on inbound interface, enabled on outbound interface. Separate crypto keys on either side.



Similarly to the “Single End SRTP Termination” case above, before the request is forwarded to the callee, the SBC allocates resources and updates the SDP with proper media addresses and ports, however, at this point, the original crypto attribute is replaced with one generated by the SD.

The construction of the crypto attribute in the SDP will be based on the configuration for the outbound realm/interface. Once the reply from the callee is received, the SBC could also accept or reject the “answer” from the callee according to the configuration and the list of crypto-suites supported. If accepted, the SBC will replace the original crypto attribute from the callee with its own to form a new SDP. The new SDP is forwarded back to the caller. At this point, SRTP media sessions are established on both sides.

- **Pass-through SRTP**
  Crypto attribute is not intercepted, just forwarded, and the key negotiation is done end-to-end.



If the configuration specifies "pass-through" mode, the SBC will not intercept the crypto attribute exchange between the caller and the callee. The crypto attribute will be forwarded as it is from the caller to the callee and vice versa.

The SBC simply modifies media IP addresses and ports to enable media anchoring (if configured), hence SRTP flows pass transparently through the SD.

Requirements

Hardware Requirements

SRTP is supported on the Net-Net 3800/4500/4600/6100/6300 platforms and require IPSec network interfaces (NIU), which allows the use of the encryption needed for SRTP. Software Datapath also suppots SRTP for 1100/3900.

```
# show prom-info PHY
Contents of PHY
        Assy, 4 Port SFP with QOS and IPSec
        Part Number:                      002-0603-58
        Serial Number:                    090850027933
        FunctionalRev:                    02.11
        BoardRev:                         02.00
        PCB Family Type:                  Quad port GiGE SFP PHY
        ID:                               4 Port GiGE w/QoS &
Encryption
        Format Rev:                       16
        Options:                          0
        Manufacturer:                     Benchmark
        Week/Year:                        50/2008
        Sequence Number:                  027933
```

The SSM module is NOT a requirement for SRTP, although typically SRTP is deployed in conjunction with TLS for SIP. Therefore, TLS is used for encrypting signaling and SRTP is used for encrypting media. In this case, then the SSM module is also required to run TLS.

```
# show security ssm
SSM (Security Service Module) V2 present.
```

If UDP/TCP is used for SIP, then SSM module is not a requirement.

Licenses Requirements

No additional licenses are required.

Bootloader requirements

Bootloader version should be the same as the runtime image. For example, use bootloader ECZ750 for any ECZ750 releases.

Design Aspects

Due to the flexibility in the configuration for different SRTP modes, it is needed to consider different aspects of the desired design for proper configuration.

Configuration Elements

Here is a brief explanation on the elements needed for SRTP configuration. This is just a basic reference, the configuration of each element will depend on the desired design and will be described in the following sections.

- Security → media-security → **sdes-profile**

  This is the first element to configure, where the algorithm and the cryptos to be used are configured.

  For sdes-profile, it is required to define the crypto-suites accepted, and also whether or not authentication and/or encryption are used for SRTP and if encryption is used for SRTCP. The “use-ingress-session-params” attribute is used to override previous parameters, specifying that the SBC will accept encryption/no-encryption, authentication/no-authentication in SRTP/SRTCP, using in the egress SDP the same session parameter that was received in the ingress SDP.

  Finally “egress-offer-format” is used to instruct the SBC on how to build the egress SDP in the case of both RTP and SRTP are supported at the same time. This is further explained in the next section.

```
# show running-config sdes-profile
sdes-profile
    name                   sdes1
    crypto-list                   AES_CM_128_HMAC_SHA1_80
AES_CM_128_HMAC_SHA1_32
    srtp-auth                     enabled
    srtp-encrypt                  enabled
    srtcp-encrypt                 enabled
    egress-offer-format                 same-as-ingress
    use-ingress-session-params          srtcp-encrypt
                                        srtp-auth
                                        srtp-encrypt
    mki                           disabled
    key
    salt
```

- Security → media-security → **media-sec-policy**

  Media-sec-policy instructs the SBC how to handle the SDP received/sent under a realm (RTP, SRTP or any of them) and, if SRTP needs to be used, the sdes -profile that needs to be used.

  The media-sec-policy should be assigned to a realm under the realm-config configuration.

```
(media-sec-policy)# show
media-sec-policy
        name                           msp1
        pass-through                   disabled
        inbound
                profile                        sdes1
                mode                           srtp
                protocol                       sdes
        outbound
```

```
                profile                                sdes1
                mode                                   srtp
                protocol                               sdes
(media-sec-policy)#
```

- Security → ipsec → **security-policy**

  The security-policy is the element that creates the security-association inside the SBC, needed to make the real SRTP encryption/unencryption. Each security-policy created must have a unique priority.

  There is a warning in the verify-config when a security-policy has the remote-ip-addr-match set to 0.0.0.0, which we can safely ignore the warning for this case:

```
# verify-config
-----------------------------------------------------------------
WARNING: security-policy [media] has invalid remote-ip-addr-match
-----------------------------------------------------------------
Total:
1 warnings

security-policy
        name                           media
        network-interface              M00:0
        priority                       2
        local-ip-addr-match            11.0.0.11
        remote-ip-addr-match           0.0.0.0
        local-port-match               0
        remote-port-match              0
        trans-protocol-match           UDP
        direction                      both
        local-ip-mask                  255.255.255.255
        remote-ip-mask                 0.0.0.0
        action                         srtp
        ike-sainfo-name
        outbound-sa-fine-grained-mask
                local-ip-mask                  0.0.0.0
                remote-ip-mask                 255.255.255.255
                local-port-mask                0
                remote-port-mask               65535
                trans-protocol-mask            255
                valid                          enabled
                vlan-mask                      0xFFF
```

Design Considerations

The intents of the design considerations explained here are to:

- Minimize interoperability issues by standardizing field configurations
- Provide guidelines for new users to the Session Border Controller
- Document when and why configuration elements should be changed from their default values
- Facilitate transition of customers from Systems Engineering to Technical Support by making configurations consistent (yielding predictable behavior)

Further, each design considers the following aspects:

- Flexibility: how resilient the configuration is, and how adaptable the configuration is (i.e. when turning up new connected networks)
- Scalability: minimizing redundant configuration objects and setting a templated foundation to allow overlay configuration with minimal disruption
- Compatibility: working with other popular devices in carriers’ VoIP networks

The main aspects treated here focused on which traffic is desired under a realm, so each design needs to consider the following, previous to any configuration:

1. SIP Traffic: SIP over UDP/TCP (unsecured transport) or over TLS (secured transport protocol).
2. Media Traffic: media over RTP, media over SRTP or media over both RTP and SRTP allowed at the same time. This would differentiate the IP design, since:
    a. For media over RTP only or SRTP only, just one IP address will be used for them
    b. For media over both RTP/SRTP allowed at the same time, then the recommendation is to use two different IPs on the same network-interface. One will send RTP traffic and the other IP will be used for SRTP traffic. This should be considered for correct IP plan under the network.

Secured/Unsecured Network

By default, the SBC considers that SIP traffic, when SRTP is configured, should run over secured transport protocol, TLS. If this is not the case, the SBC needs to be instructed to allow SIP traffic over non-secured transport protocol (UDP/TCP).

```
sip-interface
        state                          enabled
        realm-id                       access1
        description
        sip-port
                address                        11.0.0.11
                port                           5060
                transport-protocol             UDP
                tls-profile
                allow-anonymous                all
                ims-aka-profile
        carriers
…
        secured-network                enabled
```

When secured-network is set to DISABLED under a sip-interface where SRTP is configured, the sip-interface will only allow SIP over TLS. If SIP is received over UDP/TCP, the SBC will reject the call with “488 Not Acceptable Here”.

When secured-network is set to ENABLED, the SBC understands the network is secured and it accepts SIP traffic on UDP/TCP.

Media traffic

Every realm under the configuration should be instructed to the type of media that should handle whether that be RTP only, SRTP only or both RTP and SRTP. For each realm, it can be differentiated between the inbound and outbound media type, giving the flexibility of having different protocols for inbound or for outbound.

The “mode” parameter under the media-sec-policy controls the media protocol defined for each inbound/outbound flow under a realm.

- RTP Only

The “mode” parameter under the inbound/outbound section of the media-sec-policy should be set to RTP. In this case, no profile should be defined, and the protocol should be set to “None”.

```
(media-sec-policy)# show
media-sec-policy
        name                           removeCrypto
        pass-through                   disabled
        inbound
                profile
                mode                           rtp
                protocol                       none
        outbound
                profile
                mode                           rtp
                protocol                       none
(media-sec-policy)#
```

This is mostly used in single ended SRTP termination configurations, where this media-sec-policy removes the SRTP component part from the SDP to offer/accept only SRTP. This media-sec-policy should be applied under the realm where only RTP is desired.

```
realm-config
        identifier                     backbone
        description
        addr-prefix                    0.0.0.0
        network-interfaces
                                       M10:0
…
        media-sec-policy               removeCrypto
…
```

In the case of RTP only, no sdes -profile and no security-policy are needed.

- SRTP Only

The “mode” parameter under the media-sec-policy should be set to SRTP. The “profile” parameter should be set to the configured sdes -profile, and the protocol should be set to SDES.

In this case, only SRTP is accepted in the realm. An INVITE arriving to the realm without SRTP capabilities is rejected by the SBC with a “488 Not Acceptable Here”.

```
(media-sec-policy)# show
media-sec-policy
        name                           SRTP1
        pass-through                   disabled
        inbound
                profile                        sdes1
                mode                           srtp
                protocol                       SDES
        outbound
                profile                        sdes1
                mode                           srtp
                protocol                       SDES
(media-sec-policy)#
```

Where “sdes1” is the configured sdes-profile used for this implementation. Here are the default sdes-profile suggested, to be superseded only by specific customer requirements.

```
# show running-config sdes-profile
sdes-profile
    name               sdes1
    crypto-list               AES_CM_128_HMAC_SHA1_80
AES_CM_128_HMAC_SHA1_32
    srtp-auth                 enabled
    srtp-encrypt              enabled
    srtcp-encrypt             enabled
    egress-offer-format             same-as-ingress
    use-ingress-session-params      srtcp-encrypt
                                    srtp-auth
                                    srtp-encrypt
    mki                       disabled
    key
    salt
```

The media-sec-profile configured for SRTP should be applied under the desired realm.

```
realm-config
        identifier                     access1
        description
        addr-prefix                    0.0.0.0
        network-interfaces
                                       M00:0
…
        media-sec-policy               SRTP1
…
```

Finally, a security-policy should be applied to perform the RTP/SRTP or SRTP/SRTP conversion at the flow level. One security-policy is needed for the media traffic.

The local-port-match is set to 0 for an SRTP security-policy, meaning all ports on the IP address configured in local-ip-match are subject to this security-policy. Hence, to avoid a clash with the SIP signaling port (typically 5060) when signaling and media are managed on the same IP address, a second security-policy with a higher priority is required to exempt the SIP signaling port from the media security-policy.

Note that in the case where the SIP traffic runs on a different IP/Subnet from media, then this second security-policy for SIP signaling is not required.

In the security-policies configured below, note that SIP traffic is expected in 11.0.0.11 and port 5060, using any transport-protocol (so both UDP and TCP SIP traffic is accepted), and SRTP traffic is also configured for 11.0.0.11 (i.e. no signaling/media separation).

```
security-policy
        name                           signaling
        network-interface              M00:0
        priority                       1
        local-ip-addr-match            11.0.0.11
        remote-ip-addr-match           0.0.0.0
        local-port-match               5060
        remote-port-match              0
        trans-protocol-match           ALL
        direction                      both
        local-ip-mask                  255.255.255.255
        remote-ip-mask                 0.0.0.0
        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                           media
        network-interface              M00:0
        priority                       2
        local-ip-addr-match            11.0.0.11
        remote-ip-addr-match           0.0.0.0
        local-port-match               0
        remote-port-match              0
        trans-protocol-match           UDP
        direction                      both
        local-ip-mask                  255.255.255.255
        remote-ip-mask                 0.0.0.0
        action                         srtp
        ike-sainfo-name
        outbound-sa-fine-grained-mask
                local-ip-mask                  0.0.0.0
                remote-ip-mask                 255.255.255.255
                local-port-mask                0
                remote-port-mask               65535
                trans-protocol-mask            255
                valid                          enabled
                vlan-mask                      0xFFF
```

- Both RTP/SRTP support

The “mode” under the media-sec-policy should be set to ANY. Also, the profile should be configured with the sdes -profile that would be used in case of SRTP and the protocol should be set to SDES, depending on which protocol is required.

When inbound mode=any, the SBC will accept SDP with only RTP description, SDP with only SRTP description and SDP with 2 m lines having both RTP and SRTP description.

When outbound mode=any, the SBC will insert an SDP with only RTP, only SRTP or with 2 m lines, supporting both RTP and SRTP, this is controlled under the sdes -profile:

```
(sdes-profile)# egress-offer-format

<enumeration> format of offer SDP in 'any' mode
              {same-as-ingress | simultaneous-best-effort}
```

- Same-as-ingress: The SBC will use to build the egress SDP offer the mode received in the ingress realm. So if the SBC received only RTP in the ingress realm, it will insert only RTP in the egress SDP, and if it received only SRTP in the ingress SDP, it will set the egress SDP to only SRTP.
- Simultaneous-best-effort: The SBC will insert additional SRTP description in the SDP if the ingress SDP contained only RTP and vice-versa, so the resultant SDP should contain both RTP and SRTP media profiles contained in 2 different media lines in the SDP.

```
# show running-config sdes-profile
sdes-profile
    name               sdes1
    crypto-list               AES_CM_128_HMAC_SHA1_80
AES_CM_128_HMAC_SHA1_32
    srtp-auth                 enabled
    srtp-encrypt              enabled
    srtcp-encrypt             enabled
    egress-offer-format             same-as-ingress
    use-ingress-session-params      srtcp-encrypt
                                    srtp-auth
                                    srtp-encrypt
    mki                       disabled
    key
    salt


(media-sec-policy)# show
media-sec-policy
        name                           SRTP1
        pass-through                   disabled
        inbound
                profile                        sdes1
                mode                           any
                protocol                       SDES
        outbound
                profile                        sdes1
                mode                           any
                protocol                       SDES
(media-sec-policy)#
```

And this media-sec-policy should be applied under the realm where RTP+SRTP are desired:

```
realm-config
        identifier                     access1
        description
        addr-prefix                    0.0.0.0
        network-interfaces
                                       M00:0
…
        media-sec-policy               SRTP1
…
```

Finally, we need to configure the security-policy for SRTP. Since in this case both RTP and SRTP can be present under the same realm, the recommendation is to use different IPs for RTP and for SRTP.

The SRTP IP must be in the same subnet (network-interface) as the IP used for RTP. For its definition, the IP used for RTP will continue being defined under the steering-pool, while the IP for SRTP needs to be defined under the security-policy. When RTP needs to be used, the SBC will use the IP configured in the steering-pool, whereas when SRTP needs to be inserted into the SDP, the SBC will choose the IP from the security-policy AND an available port from the steering-pool configured for RTP, so the dimensioning of the port range of the steering-pool should consider both RTP and SRTP estimated traffic.

If SIP traffic runs over the same subnet (network-interface), it is recommended not to use the IP used for SRTP traffic. That way, it is not necessary to configure a second security-policy for SIP traffic.

In the example below, 11.0.0.10 is used for RTP and 11.0.0.11 is used for SRTP. In the case that SIP traffic is desired under the same network, it would be recommended not to use 11.0.0.11, as this is reserved for SRTP use and the security-policy configured for it would apply.

```
steering-pool
        ip-address                     11.0.0.10
        start-port                     20000
        end-port                       49999
        realm-id                       access

security-policy
        name                           media
        network-interface              M00:0
        priority                       1
        local-ip-addr-match            11.0.0.11
        remote-ip-addr-match           0.0.0.0
        local-port-match               0
        remote-port-match              0
        trans-protocol-match           UDP
        direction                      both
        local-ip-mask                  255.255.255.255
        remote-ip-mask                 0.0.0.0
        action                         srtp
        ike-sainfo-name
        outbound-sa-fine-grained-mask
                local-ip-mask                  0.0.0.0
                remote-ip-mask                 255.255.255.255
                local-port-mask                0
                remote-port-mask               65535
                trans-protocol-mask            255
                valid                          enabled
                vlan-mask                      0xFFF
        last-modified-by               admin@10.0.3.99
        last-modified-date             2010-07-20 04:59:53
```

High Availability

In order for SIP and SRTP to work properly in the HA environment, both sip-config and ipsec-global-config elements should be configured.

The ipsec-global-config element enables the secured process to update the Security Associations on both the active and standby unit for high availability. The descriptions of the parameters in the ipsec-global-config element are as follows:

- red-ipsec-port: redundant IP security synchronization port
- red-max-trans: max redundant transactions to keep
- red-sync-start-time: redundant sync start timeout
- red-sync-comp-time: redundant sync complete timeout

```
ipsec-global-config
        red-ipsec-port              1994
        red-max-trans               10000
        red-sync-start-time         5000
        red-sync-comp-time          1000

sip-config
        … …
        red-sip-port                1988
        red-max-trans               10000
        red-sync-start-time         5000
        red-sync-comp-time          1000
        … …
```

Notes on the Reference Configuration

The intention of this document is not to provide a full set of configurations, as the flexibility of the SRTP configuration makes valid a high number of different possible configurations. The objective is to present some common and valid configurations that have been tested and verified in Oracle labs.

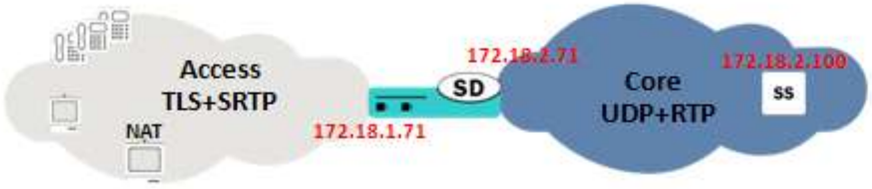
In the cases considered here, there is a considered “access” or “peer1A” network, in the 172.18.1.0/24 network, and a considered “core” or “peer1B” in the 172.18.2.0/24 network. In all cases SIP and media traffic runs on the same subnets.

To simplify the use of this BCP, no other elements are configured in this case, so no redundancy or DDoS prevention are configured in the configurations exposed. The configurations follow the guides of BCP for access (using policy based realm bridging) and peering scenarios. For TLS, it is assumed single-side authentication in all cases.

The configurations presented use SDES mechanism for SRTP encryption. No SRTP pass-through cases are presented here, as there is nothing required for the SBC to be transparent to the SRTP negotiation end-to-end.

Single-Ended SRTP Termination on secured networks.

This is the typical access scenario where SRTP is deployed completely in the access network, allowing the users to use TLS for SIP and SRTP for media. In the core network, UDP is used for SIP and RTP is used for media.



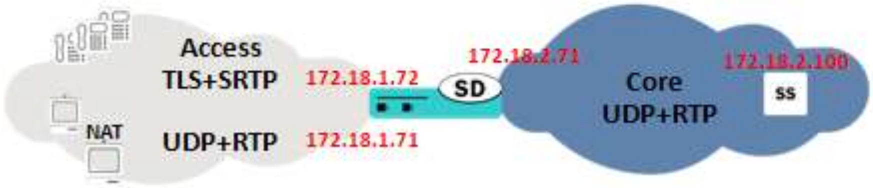
The IP used for SIP and SRTP in the SBC in the access network is 172.18.1.71, and the IP used for SIP and RTP in the core network is 172.18.2.71. The SIP Registrar/Proxy in the core network is in 172.18.2.100.

In this case, secured-network is set to DISABLED under the access sip-interface and ENABLED on the core sip-interface. Two security-policies are configured, one for SRTP and one that creates the exception for SIP signaling. Also, two media-sec-policies are created, one in the access network with mode=SRTP and one in the core with mode=RTP.

RTP and Single-Ended SRTP Termination on unsecured networks.

This is a very common architecture, where both RTP and SRTP endpoints reside in the access network, especially while in transition from RTP to SRTP. This means that both UDP/RTP and TLS/SRTP can be present in the access network. In the core network, UDP for SIP and RTP for media will be used.

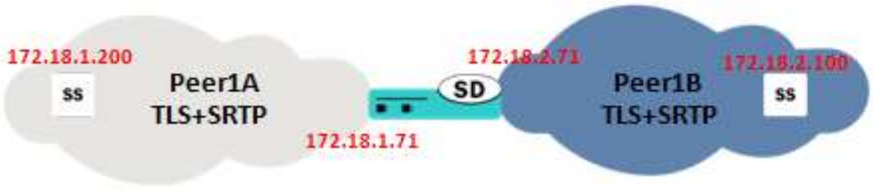
In this case, in the access network we will use 172.18.1.71 for SIP traffic (UDP and TLS) and also for RTP traffic. 172.18.1.72 will be used for SRTP traffic. In the core network, 172.18.2.71 will be used for SIP and RTP. The SIP Proxy/Registrar uses 172.18.2.100.



Secured-network parameter is set to ENABLED under the access sip-interface and ENABLED on the core sip-interface. Only one security-policy is configured for SRTP under 172.181.72. Two media-sec-policies are created, one in the access network with mode=any and one in the core with mode=RTP. As in the access network both RTP and SRTP endpoints could be present, the egress-offer-format is set to simultaneous-best-effort.

Back-to-back SRTP Termination

Normally deployed in peering scenarios where SRTP is needed in both networks that the SBC is interconnecting. In that case, the Session Border Controller is doing SRTP termination so the SRTP key exchange is different in the two connected networks.



In the SBC, 172.18.1.71 will be used for SIP (TLS) and SRTP in the peer1A network, while 172.18.2.71 will be used in the 172.18.2.71.

The peer element sending traffic in the peer1A network will be in 172.18.1.200, while the peer element in the peer1B will be 172.18.2.100.

Secured-network is set to DISABLED under both sip-interfaces. Two security-policies are configured per peer1 realm, one for SRTP and one that creates the exception for SIP signaling, so four security-policies are configured in total. Also, two media-sec-policies are created, one in the peer1A network with mode=SRTP and one in the peer1B with mode=SRTP, where each one is linked with a different SDES profile, to allow different cryptos between networks. Note that this is not required, and the same SDES profile could be used for both networks, the key exchange would keep different as the SBC would terminate the SRTP anyway, so configuring different SDES profiles would be only needed in the case where the crypto-suites supported in each network are different or have different characteristics.

Troubleshooting

A network capture taken on both access and core network should show RTP packets with the same sequence number, however, if SRTP termination is done in the SBC, the payload contained in RTP packets with the same sequence number will be different because of the encryption/unencryption done by the SD.

To troubleshoot SRTP on the Session Border Controller, following commands can be used:

- Log.secured provides logs of the security-association activities related to SRTP.
- Show sa stats srtp

```
# show sa stats srtp
03:07:17-186
SA Statistics                   ---- Lifetime ----
                            Recent      Total  PerMax
SRTP Statistics
ADD-SA Req Sent                2          2       2
ADD-SA Req Rcvd                2          2       2
…
DEL-SA Req Sent                2          2       2
DEL-SA Req Rcvd                2          2       2
…
MOD-SA Req Sent                0          0       0
MOD-SA Req Rcvd                0          0       0
..
SA Added                       4          4       4
SA Add Failed                  0          0       0
SA Deleted                     4          4       4
```

- “Show security srtp” commands show the security association created for SRTP encryption and its detailed information.
    - show security srtp <network_interface> debug/brief/detail/raw
        - Note there is a warning when these commands want to be run, as it should be done carefully in production systems:

```
WARNING: This action might affect system performance and take a long
time to finish.
Are you sure [y/n]?:
```

- Show security srtp status <network_interface>
- Show security spd <network_interface>

```
# show security srtp sad M00 debug
WARNING: This action might affect system performance and take a long
time to finish.
Are you sure [y/n]?: y
SRTP security-association-database for interface 'M00':
Displaying SA's that match the following criteria -
direction                 : both
src-addr-prefix           : any
src-port                  : any
dst-addr-prefix           : any
dst-port                  : any
trans-proto               : ALL
Inbound:
destination-address       : 62.2.139.213
destination-port          : 10012
vlan-id                   : 0
sal-index                 : 2
sad-index                 : 10
ssrc                      : 1514612894
encr-algo                 : aes-128-ctr
auth-algo                 : hmac-sha1
auth-tag-length           : 80
flags -
ms:  5489040, ls:        8
mtu                       : 1500
mki                       : 0
mki length                : 0
lifetime byte count -
ms: 0x       0, ls: 0x       0
packet count -
ms: 0x       0, ls: 0x     12F
roll over count           : 0
anti replay highest seq num : 11814
highest seq num           : 0
auth error count          : 0
anti replay count         : 0
mki mismatch count        : 0
ssrc mismatch count       : 1
```

```
# show security srtp sad M00 raw
WARNING: This action might affect system performance and take a long
time to finish.
```

```
Are you sure [y/n]?: y
SRTP security-association-database for interface 'M00':
Displaying SA's that match the following criteria -
      direction                 : both
      src-addr-prefix           : any
      src-port                  : any
      dst-addr-prefix           : any
      dst-port                  : any
      trans-proto               : ALL
Inbound:
 Index I                                  VLN    P   <-- Masks -->  SAD
Next
         TP  Dest. IP Address    SPI   Pr  ID TS P V Pr VLN TS P V
Index Link
 0000a 0 01 00000000 00000000 00000000 11 000 00 0 0 ff 000 00 0 0
0000a 00000
            00000000 d58b023e
 Index Flags MS Flags LS EX Flg  MTU SSRC     MKI      MKI Len  ROC
 0000a 05489040 00000008 00202a 05dc 5a47289e 00000000 00000000
00000000
       Master key:  f6 8e c5 af 6c af 96 72 64 78 04 97 14 44 c1 a9
       Master salt: 59 da 31 4d c2 3d 15 ca b6 3b 39 e1 27 2d
       E-IV:        59 da 31 4d 98 7a 3d 54 b6 3b 39 e1 27 2d 00 00
       HMAC ipad:   7a cc 93 f9 72 44 2d df ee df cc 89 3d a2 35 74 18
32 bb 25
       HMAC opad:   2b 6d cc 43 49 fa 65 8e 4a d2 03 50 90 00 9f 10 16
6d 1a 90
       Sequence Number   Anti-replay window (128 bits wide)
       00002f68          ffffffff ffffffff ffffffff ffffffff
       Life Byte Count  Packet Count     Auth Err Anti-replay Err
       0000000000000000 0000000000000271 00000000 00000000
       ICV Len HSN      MKI Mismatch SSRC Mismatch
       04      00000000 00000000     00000001
```

Enhanced Traffic Controller (ETC) NIU support

Hardware and software support requirements to support ETC NIU

The NN4500 CPU1 or CPU2 should have the Nov, 2010 or later bootloader in order to support the Enhanced Traffic Controller (ETC) NIU.

Configuration deviations from HiFn

The ETC NIU supports only the SDES protocol for SRTP.

The configuration element “security-policy” is no longer required for SRTP using the ETC NIU.

The AES_CM_128 encryption and HMAC_SHA1_80 or HMAC_SHA1_32 authentication suites are supported on the ETC NIU. ARIA Cipher suite are also supported.

The ETC NIU contains one Cavium hardware chip that provides encryption/decryption. In order to support 10,000 concurrent sessions and overcome the 1 GB bandwidth limitation per port, a major design goal is to split the traffic between any 2 ports on ingress and remaining 2 ports on egress. Upon reaching 10,000 concurrent sessions limit, subsequent calls will be rejected.

Debugging Info

Following is the list of commands to be used in order to get SRTP and ETC specific information.

```
show nat flow-info srtp statistics
This command will show the global statistics for all SRTP flows.
SBASNQ06# show nat flow-info srtp statistics
PPM_ID_SRTP_E:
PPX Global Statistics
---------------------
        alloc_count                : 50
        dealloc_count              : 16
        input-packets              : 0
        output-packets             : 0
        sessions-count             : 2
        init-requests              : 4
        init-success               : 4
        init-fail                  : 0
        modify-requests            : 0
        modify-success             : 0
        modify-fail                : 0
        delete-requests            : 2
        delete-success             : 2
        delete-fail                : 0
        query-requests             : 0
        query-success              : 0
        query-fail                 : 0
        resources-error            : 0
        protect-fail               : 0
        unprotect-fail             : 0
        status-err                 : 0
        bad-param                  : 0
        alloc-fail                 : 0
        dealloc-fail               : 0
        terminus                   : 0
        auth-fail                  : 0
        cipher-fail                : 0
        replay-fail                : 0
        replay-old                 : 0
        algo-fail                  : 0
        no-such-op                 : 0
        no-ctx                     : 0
        cant-check                 : 0
        key-expired                : 0
        nonce-bad                  : 0
        read-failed                : 0
        write-failed               : 0
        parse-err                  : 0
        encode-err                 : 0
        pfkey-err                  : 0
        mki-changed                : 0
        srtp-pkt-too-small         : 0
        srtcp-pkt-too-small        : 0

PPM_ID_SRTP_D:
PPX Global Statistics
---------------------
        alloc_count                : 50
        dealloc_count              : 16
        input-packets              : 0
        output-packets             : 0
        sessions-count             : 3
        init-requests              : 2
        init-success            : 2
        init-fail               : 0
        modify-requests         : 1
        modify-success          : 1
        modify-fail             : 0
        delete-requests         : 0
        delete-success          : 0
        delete-fail             : 0
        query-requests          : 0
        query-success           : 0
        query-fail              : 0
        resources-error         : 0
        protect-fail            : 0
        unprotect-fail          : 0
        status-err              : 0
        bad-param               : 0
        alloc-fail              : 0
        dealloc-fail            : 0
        terminus                : 0
        auth-fail               : 0
        cipher-fail             : 0
        replay-fail             : 0
        replay-old              : 0
        algo-fail               : 0
        no-such-op              : 0
        no-ctx                  : 0
        cant-check              : 0
        key-expired             : 0
        nonce-bad               : 0
        read-failed             : 0
        write-failed            : 0
        parse-err               : 0
        encode-err              : 0
        pfkey-err               : 0
        mki-changed             : 0
        srtp-pkt-too-small      : 0
        srtcp-pkt-too-small     : 0
```

show nat flow-info srtp by-addr 3.0.0.2 all

This command will show the crypto information details for a flow with the given   source address. If “all” is used, the details for all the SRTP flows will be displayed. However, “all” does not to display the statistics from the octeon srtp code.

```
SBASNQ06# show nat flow-info srtp by-addr 3.0.0.2 all

Crypto Parameters 3.0.0.2:7001 -> 7.0.0.2:6058
=================

        Collapsed               : false
        SRTCP Only              : false
Crypto In
------------------
        destination-address     : 208.54.47.80
        destination-port        : 40000
        vlan-id                 : 632
        encr-algo               : aes-128-ctr
        auth-algo               : hmac-sha1
        auth-tag-length         : 32
                key index               : 0
                mki                     : none
                roll-over-count           : 0


---No Crypto Out---

PPM_ID_SRTP_D
PPX Statistics
--------------
Stream #1
        ssrc                      : 3879260980
        rtp-cipher-id             : AES-128-ICM
        rtp-auth-id               : HMAC-SHA1
        rtp-security-level        : Crypto + Auth
        rtp-total-packets         : 5423
        rtp-total-bytes           : 954448
        rtp-cipher-bytes          : 867680
        rtp-auth-bytes            : 932756
        rtcp-cipher-id            : AES-128-ICM
        rtcp-auth-id              : HMAC-SHA1
        rtcp-security-level       : Crypto + Auth
        rtcp-total-packets        : 0
        rtcp-total-bytes          : 0
        rtcp-cipher-bytes         : 0
        rtcp-auth-bytes           : 0
        key-lifetime              : 42949672954294961871
        direction                 : Receiver

Crypto Parameters 3.0.0.2:7001 -> 7.0.0.2:6058
=================

        Collapsed                 : false
        SRTCP Only                : true
Crypto In
------------------
        destination-address       : 208.54.47.80
        destination-port          : 40000
        vlan-id                   : 632
        encr-algo                 : aes-128-ctr
        auth-algo                 : hmac-sha1
        auth-tag-length           : 32
                key index                 : 0
                mki                       : none
                roll-over-count           : 0


---No Crypto Out---

PPM_ID_SRTP_D
PPX Statistics
--------------
Stream #1
        ssrc                      : 0
        rtp-cipher-id             : NULL
        rtp-auth-id               : NULL
        rtp-security-level        : None
        rtp-total-packets         : 0
        rtp-total-bytes           : 0
        rtp-cipher-bytes          : 0
        rtp-auth-bytes            : 0
        rtcp-cipher-id            : NULL
        rtcp-auth-id              : NULL
        rtcp-security-level       : None
        rtcp-total-packets        : 0
        rtcp-total-bytes          : 0
        rtcp-cipher-bytes         : 0
        rtcp-auth-bytes           : 0
        key-lifetime              : 0
        direction                 : Unknown
```

show mbcd errors

This command will show counters for SRTP errors, including SRTP Flow Add Failed, SRTP Flow Delete Failed, and SRTP Flow Update Failed.

```
SBASNQ06# show mbcd errors
22:29:33-160
MBC Errors/Events              ---- Lifetime ----
                        Recent      Total  PerMax
Client Errors                0          0       0
Client IPC Errors            0          0       0
Open Streams Failed          0          0       0
Drop Streams Failed          0          0       0
Exp Flow Events              1          1       1
Exp Flow Not Found           0          0       0
Transaction Timeouts         0          0       0

Server Errors                0          0       0
Server IPC Errors            0          0       0
Flow Add Failed              0          2       2
Flow Delete Failed           0          0       0
Flow Update Failed           0          0       0
Flow Latch Failed            0          0       0
Pending Flow Expired         0          0       0
ARP Wait Errors              0          0       0
Exp CAM Not Found            0          0       0
Drop Unknown Exp Flow        0          0       0
Drop/Exp Flow Missing        0          0       0
Exp Notify Failed            0          0       0
Unacknowledged Notify        0          0       0
Invalid Realm                0          0       0
No Ports Available           0          0       0
Insufficient Bandwidth       0          0       0
Stale Ports Reclaimed        0          0       0
Stale Flows Replaced         0          0       0
Telephone Events Gen         0          0       0
Pipe Alloc Errors            0          0       0
Pipe Write Errors            0          0       0
Not Found In Flows           0          0       0
SRTP Flow Add Failed         0          0       0
SRTP Flow Delete Faile       0          0       0
SRTP Flow Update Faile       0          0       0
SRTP Capacity Exceeded       0          0       0
```

show mbcd statistics

This command will show counters for number of active SRTP/SRTCP flows, as well as the number of SRTP Sessions maintained.

```
SBASNQ06# show mbcd statistics
22:29:40-168
MBCD Status                -- Period -- -------- Lifetime --------
                 Active    High   Total      Total  PerMax    High
Client Sessions       1       1       1          1       1       1
Client Trans          0       1       3          3       3       1
Contexts              3       3       2          3       2       3
Flows                14      14       3         14      11      14
Flow-Port             2       2       2          2       2       2
Flow-NAT             13      13       5         16      11      13
Flow-RTCP             2       2       4          4       4       2
```

```
Flow-Hairpin          0       0       0          0       0       0
Flow-Released         0       0       0          0       0       0
MSM-Release           0       0       0          0       0       0
Rel-Port              0       0       0          0       0       0
Rel-Hairpin           0       0       0          0       0       0
NAT Entries          15      15       9         20      11      15
Free Ports        80000   80004       0      80004   80004   80004
Used Ports            4       4       4          4       4       4
Port Sorts            -       -       0          0       0
Queued Notify         0       0       0          0       0       0
MBC Trans             0       3       3          3       3       3
MBC Ignored           -       -       0          0       0
ARP Trans             0       0       0          0       0       0
Relatch NAT           0       0       0          0       0       0
Relatch RTCP          0       0       0          0       0       0
SRTP Only Flows       1       1       3          3       3       1
SRTCP Only Flow       3       3       3          3       3       3
SRTP Collapsed        0       0       0          0       0       0
SRTP Sessions         1       1       3          3       3       1

Flow Rate = 0.0
Load Rate = 0.0
```

show mbcd all

This command will show counters for number of active SRTP/SRTCP flows, as well as the number of SRTP Sessions maintained.

```
SBASNQ06# show mbcd all
22:29:44-172
MBCD Status                -- Period -- -------- Lifetime --------
                 Active    High   Total      Total  PerMax    High
Client Sessions       1       1       1          1       1       1
Client Trans          0       1       3          3       3       1
Contexts              3       3       2          3       2       3
Flows                14      14       3         14      11      14
Flow-Port             2       2       2          2       2       2
Flow-NAT             13      13       5         16      11      13
Flow-RTCP             2       2       4          4       4       2
Flow-Hairpin          0       0       0          0       0       0
Flow-Released         0       0       0          0       0       0
MSM-Release           0       0       0          0       0       0
Rel-Port              0       0       0          0       0       0
Rel-Hairpin           0       0       0          0       0       0
NAT Entries          15      15       9         20      11      15
Free Ports        80000   80004       0      80004   80004   80004
Used Ports            4       4       4          4       4       4
Port Sorts            -       -       0          0       0
Queued Notify         0       0       0          0       0       0
MBC Trans             0       3       3          3       3       3
MBC Ignored           -       -       0          0       0
ARP Trans             0       0       0          0       0       0
Relatch NAT           0       0       0          0       0       0
Relatch RTCP          0       0       0          0       0       0
SRTP Only Flows       1       1       3          3       3       1
SRTCP Only Flow       3       3       3          3       3       3
SRTP Collapsed        0       0       0          0       0       0
SRTP Sessions         1       1       3          3       3       1

Flow Rate = 0.0
Load Rate = 0.0
22:29:44-172
NAT Entries             ---- Lifetime ----
                 Recent      Total  PerMax
Adds                  9         20      11
```

```
Deletes               4          5       4
Updates               2          2       2
Non-Starts            0          0       0
Stops                 0          0       0
Timeouts              0          0       0
22:29:44-172
ACL Entries                -- Period -- -------- Lifetime --------
                 Active    High   Total      Total  PerMax    High
Static Trusted        4       4       0          4       4       4
Static Blocked        4       4       0          4       4       4
Dynamic Trusted       1       1       1          1       1       1
Dynamic Blocked       0       0       0          0       0       0

ACL Operations          ---- Lifetime ----
                 Recent      Total  PerMax
App Requests          1          2       1
Added                 1          9       8
Removed               0          1       1
Dropped               0          0       0
22:29:44-172
MBC Errors/Events              ---- Lifetime ----
                        Recent      Total  PerMax
Client Errors                0          0       0
Client IPC Errors            0          0       0
Open Streams Failed          0          0       0
Drop Streams Failed          0          0       0
Exp Flow Events              1          1       1
Exp Flow Not Found           0          0       0
Transaction Timeouts         0          0       0

Server Errors                0          0       0
Server IPC Errors            0          0       0
Flow Add Failed              0          2       2
Flow Delete Failed           0          0       0
Flow Update Failed           0          0       0
Flow Latch Failed            0          0       0
Pending Flow Expired         0          0       0
ARP Wait Errors              0          0       0
Exp CAM Not Found            0          0       0
Drop Unknown Exp Flow        0          0       0
Drop/Exp Flow Missing        0          0       0
Exp Notify Failed            0          0       0
Unacknowledged Notify        0          0       0
Invalid Realm                0          0       0
No Ports Available           0          0       0
Insufficient Bandwidth       0          0       0
Stale Ports Reclaimed        0          0       0
Stale Flows Replaced         0          0       0
Telephone Events Gen         0          0       0
Pipe Alloc Errors            0          0       0
Pipe Write Errors            0          0       0
Not Found In Flows           0          0       0
SRTP Flow Add Failed         0          0       0
SRTP Flow Delete Faile       0          0       0
SRTP Flow Update Faile       0          0       0
SRTP Capacity Exceeded       0          0       0
22:29:44-172
                               ---- Lifetime ----
                        Recent      Total  PerMax
Add incoming:
 Request received            1          1       1
 Duplicates received         0          0       0
 Replies sent                1          1       1
 Errors sent                 0          0       0
Add outgoing:
 Requests sent               1          1       1
 Req retransmissions         0          0       0
 Replies received            1          1       1
 Errors received             0          0       0

Avg Latency=0.000 for 1
Max Latency=0.000
22:29:44-172
SRTP Flows              ---- Lifetime ----
                 Recent      Total  PerMax
Adds                  3          3       3
Deletes               2          2       2
Updates               0          0       0
---< NO DATA AVAILABLE >----(Subtract)
22:29:45-172
                               ---- Lifetime ----
                        Recent      Total  PerMax
Notify incoming:
 Request received            1          1       1
 Duplicates received         0          0       0
 Replies sent                1          1       1
 Errors sent                 0          0       0
Notify outgoing:
 Requests sent               1          1       1
 Req retransmissions         0          0       0
 Replies received            1          1       1
 Errors received             0          0       0

Avg Latency=0.000 for 1
Max Latency=0.000
---< NO DATA AVAILABLE >----(Other)
---< NO DATA AVAILABLE >----(Unknown)
```

show sipd errors

This command will show the counter for number of SIP sessions that failed to setup due to problems related to SRTP signaling.

```
SBASNQ06# show sipd errors
22:29:50-178
SIP Errors/Events             ---- Lifetime ----
                       Recent      Total  PerMax
SDP Offer Errors            0          0       0
SDP Answer Errors           0          0       0
Drop Media Errors           0          0       0
Transaction Errors          0          0       0
Application Errors          0          0       0
Media Exp Events            0          0       0
Early Media Exps            0          0       0
Exp Media Drops             0          0       0
Expired Sessions            0          0       0
Multiple OK Drops           0          0       0
Multiple OK Terms           0          0       0
Media Failure Drops         0          0       0
Non-ACK 2xx Drops           0          0       0
Invalid Requests            0          0       0
Invalid Responses           0          0       0
Invalid Messages            0          0       0
CAC Session Drop            0          0       0
Nsep User Exceeded          0          0       0
Nsep SA   Exceeded          0          0       0
CAC BW Drop                 0          0       0
SRTP Errors                 0          0       0
```

```
show security srtp sessions
```

This command will be used to show the active srtp/srtcp sessions and the total allowed capacity of 10,000 sessions.

```
SBASNQ06# show security srtp sessions
Capacity=10000
SRTP Sessions       -- Period -- ---- Lifetime ----
  Active    High   Total  Recent      Total  PerMax
       1       1       3          3       3       1
```

```
show nat flow-info all
```

This command will also show the crypto information for the SRTP flows. This should not be executed in a production environment, since it dumps information about all the flows.

```
SBASNQ06# show nat flow-info all
Output curtailed due to size.
. . . . . continued

----------------------------------------------
SA_flow_key       : 7.0.0.2               SA_prefix : 32
DA_flow_key       : 10.176.28.218         DA_prefix : 32
SP_flow_key       : 6058                  SP_prefix : 16
DP_flow_key       : 40000                 DP_prefix : 16
VLAN_flow_key     : 980
Protocol_flow_key : 17
Ingress_flow_key  : 1
Ingress Slot      : 1
Ingress Port      : 0
NAT IP Flow Type  : IPv4 to IPv4
XSA_data_entry    : 208.54.47.80
XDA_data_entry    : 3.0.0.2
XSP_data_entry    : 40000
XDP_data_entry    : 7001
Egress_data_entry : 0
Egress Slot       : 0
Egress Port       : 0
flow_action       : 0X1
optional_data     : 0
FPGA_handle       : 0x000000c1
assoc_FPGA_handle : 0x00000000
VLAN_data_entry   : 632
host_table_index  : 6
Switch ID         : 0x00000005
average-rate      : 0
weight            : 0x0
init_flow_guard   : 300
inact_flow_guard  : 300
max_flow_guard    : 86400
payload_type_2833 : 0
index_2833        : 0
pt_2833_egress    : 0
qos_vq_enabled    : 0
codec_type        : 0
HMU_handle        : 0
SRTP Crypto In    : NONE
SRTP Crypto Out   : AES_CM_128_HMAC_SHA1_32
----------------------------------------------

    Input Link Parameters -  IFD Index: 0x5
    ----------------------------------------------
                IFD Byte Enable: false
                EPD Mode Enable: true
```

```
                         Retain: false
                       ABJ Mode: true
                  Disable Empty: false
                Ignore On Empty: false
                           TGID: 0x6
                          WRGID: 0x0
                      TG Enable: true
                     WRG Enable: false

    Output Link Parameters -  OFD Index: 0x5
    ----------------------------------------------
                    shaped_flow: false
              latency_sensitive: false
                       pkt_mode: Packet Mode
           zero_min_credit_flow: false
                parent_pipe_num: 0x1
                          delta: 0x1
            flow_credit_min_exp: 0x0
            flow_credit_min_man: 0x0

IFD 0x00000005:       dropCount = 0x00000000


IFD 0x00000005:       acceptCount = 0x00001f35

----------------------------------------------
```

dump-np-stats

This command displays all of the octeon statistics

```
SBASNQ06# dump-etc-stats
ACME Net-Net 4500 Firmware SCX6.3.0 F-1 GA (Build 156)
Build Date=07/13/11

----------------- HyperChicken Stats and FPGA Register DUMP -------------------------

Revision           : 1.05
Date stamp         : WED MAR 23 02:00:00 2011


----------------- Port 0 -------------------------

Hyperchicken FPGA Internal MacPhy Stats Reg dump:

MAC Stats on ch0:
Tx Bytes           : addr(0xd8010000): 0x00345ec2
Rx Bytes           : addr(0xd8010004): 0x003fe0ec
Rx Undersz Fr      : addr(0xd8010008): 0x00000000
Rx Frag Fr         : addr(0xd801000c): 0x00000000
Rx 64B Fr Ok       : addr(0xd8010010): 0x000000b5
Rx 65-127B Fr Ok   : addr(0xd8010014): 0x000000fe
Rx 128-255B Fr Ok  : addr(0xd8010018): 0x00003a49
Rx 256-511B Fr Ok  : addr(0xd801001c): 0x00000005
Rx 512-1023B Fr Ok : addr(0xd8010020): 0x00000007
Rx 1024-Max Fr Ok  : addr(0xd8010024): 0x00000299
Rx Oversz Fr       : addr(0xd8010028): 0x00000000
Tx 64B Fr Ok       : addr(0xd801002c): 0x00000367
Tx 65-127B Fr Ok   : addr(0xd8010030): 0x0000002f
Tx 128-255B Fr Ok  : addr(0xd8010034): 0x00003a48
Tx 256-511B Fr Ok  : addr(0xd8010038): 0x00000001
Tx 512-1023B Fr Ok : addr(0xd801003c): 0x00000002
Tx 1024-Max Fr Ok  : addr(0xd8010040): 0x00000000
Tx Oversz Fr       : addr(0xd8010044): 0x00000000
```

```
Rx Fr Ok           : addr(0xd8010048): 0x00003ea4
Rx Fr ChkSeq Err   : addr(0xd801004c): 0x00000000
Rx Broadcast Fr Ok : addr(0xd8010050): 0x00000031
Rx Multic Fr Ok    : addr(0xd8010054): 0x00003b2d
Rx Cntl Fr Ok      : addr(0xd8010058): 0x00000000
Rx FrLen/Typ ooRng : addr(0xd801005c): 0x00000000
Rx Vlan Tag Fr Ok  : addr(0xd8010060): 0x0000002c
Rx Pause Fr Ok     : addr(0xd8010064): 0x00000000
Rx CtlFr+uns opcode: addr(0xd8010068): 0x00000000
Tx Fr Ok           : addr(0xd801006c): 0x00003de4
Tx Broadcast Fr Ok : addr(0xd8010070): 0x00000022
Tx Multic Fr Ok    : addr(0xd8010074): 0x00000009
Tx Underrun Err    : addr(0xd8010078): 0x00000000
Tx Cntl Fr Ok      : addr(0xd801007c): 0x00000000
Tx Vlan Tag Fr Ok  : addr(0xd8010080): 0x00003aa3
Tx Pause Fr Ok     : addr(0xd8010084): 0x00000000
Rx Alignment Err   : addr(0xd8010100): 0x003464f0
Rx_Cfg_Word0   : addr(0xd8010a00): 0x00000000
Rx_Cfg_Word1   : addr(0xd8010a40): 0x5a000000
Tx_Cfg         : addr(0xd8010a80): 0x58000000
Flow_Ctl_Cfg   : addr(0xd8010ac0): 0x00000000
Speed_Cfg      : addr(0xd8010b00): 0x80000000
Mgt_Cfg        : addr(0xd8010b40): 0x00000000
Uni_Addr_Word0 : addr(0xd8010b80): 0x00000000
Uni_Addr_Word1 : addr(0xd8010b84): 0x00000000

PHY Stats on ch0:
NA

Curtailed due to size…… Similar output for port 1 through 7
----------------- Port 1 -------------------------
----------------- Port 2 -------------------------
----------------- Port 3 -------------------------
----------------- Port 4 -------------------------
----------------- Port 5 -------------------------
----------------- Port 6 -------------------------
----------------- Port 7 -------------------------

Hyperchicken FPGA Internal MacPhy Stats Reg dump:

MAC Stats on ch7:
Tx Bytes           : addr(0xd801e000): 0x00000c80
Rx Bytes           : addr(0xd801e004): 0x00001410
Rx Undersz Fr      : addr(0xd801e008): 0x00000000
Rx Frag Fr         : addr(0xd801e00c): 0x00000000
Rx 64B Fr Ok       : addr(0xd801e010): 0x0000002a
Rx 65-127B Fr Ok   : addr(0xd801e014): 0x00000024
Rx 128-255B Fr Ok  : addr(0xd801e018): 0x00000000
Rx 256-511B Fr Ok  : addr(0xd801e01c): 0x00000000
Rx 512-1023B Fr Ok : addr(0xd801e020): 0x00000000
Rx 1024-Max Fr Ok  : addr(0xd801e024): 0x00000000
Rx Oversz Fr       : addr(0xd801e028): 0x00000000
Tx 64B Fr Ok       : addr(0xd801e02c): 0x00000032
Tx 65-127B Fr Ok   : addr(0xd801e030): 0x00000000
Tx 128-255B Fr Ok  : addr(0xd801e034): 0x00000000
Tx 256-511B Fr Ok  : addr(0xd801e038): 0x00000000
Tx 512-1023B Fr Ok : addr(0xd801e03c): 0x00000000
Tx 1024-Max Fr Ok  : addr(0xd801e040): 0x00000000
Tx Oversz Fr       : addr(0xd801e044): 0x00000000
Rx Fr Ok           : addr(0xd801e048): 0x0000004e
Rx Fr ChkSeq Err   : addr(0xd801e04c): 0x00000000
Rx Broadcast Fr Ok : addr(0xd801e050): 0x00000021
Rx Multic Fr Ok    : addr(0xd801e054): 0x00000009
Rx Cntl Fr Ok      : addr(0xd801e058): 0x00000000
Rx FrLen/Typ ooRng : addr(0xd801e05c): 0x00000000
Rx Vlan Tag Fr Ok  : addr(0xd801e060): 0x0000004e
Rx Pause Fr Ok     : addr(0xd801e064): 0x00000000
Rx CtlFr+uns opcode: addr(0xd801e068): 0x00000000
Tx Fr Ok           : addr(0xd801e06c): 0x00000032
Tx Broadcast Fr Ok : addr(0xd801e070): 0x00000032
Tx Multic Fr Ok    : addr(0xd801e074): 0x00000000
Tx Underrun Err    : addr(0xd801e078): 0x00000000
Tx Cntl Fr Ok      : addr(0xd801e07c): 0x00000000
Tx Vlan Tag Fr Ok  : addr(0xd801e080): 0x00000032
Tx Pause Fr Ok     : addr(0xd801e084): 0x00000000
Rx Alignment Err   : addr(0xd801e100): 0x00000c80
Rx_Cfg_Word0   : addr(0xd801ea00): 0xddccbbaa
Rx_Cfg_Word1   : addr(0xd801ea40): 0x5800ffee
Tx_Cfg         : addr(0xd801ea80): 0x58000000
Flow_Ctl_Cfg   : addr(0xd801eac0): 0x00000000
Speed_Cfg      : addr(0xd801eb00): 0x8c000000
Mgt_Cfg        : addr(0xd801eb40): 0x00000041
Uni_Addr_Word0 : addr(0xd801eb80): 0x00000041
Uni_Addr_Word1 : addr(0xd801eb84): 0x00000000

PHY Stats on ch7:

PHY Stats on ch7:
Ctl_Reg    : addr(0xd801f000): 0x00001140
Status_Reg : addr(0xd801f004): 0x000001e8
Phy_Id1    : addr(0xd801f008): 0x00000028
Phy_Id2    : addr(0xd801f00c): 0x0000d400
Auto_Neg4  : addr(0xd801f010): 0x000001a0
Auto_Neg5  : addr(0xd801f014): 0x00004060
Auto_Neg6  : addr(0xd801f018): 0x00000006
Auto_Neg7  : addr(0xd801f01c): 0x00002001
Auto_Neg8  : addr(0xd801f020): 0x00000000
Ext_Status : addr(0xd801f03c): 0x00008000
AN_Intr_Ctl: addr(0xd801f040): 0x00000003
Lpbk_Ctl   : addr(0xd801f044): 0x00000000

----------------- SPI ports -------------------------

Hyperchicken FPGA Internal SPI Stats Reg dump:
SPI Link Status                  : addr(0xd8000080): 0x000008b9
SPI Port 8 Status                : addr(0xd8000084): 0x00028200
SPI Port 8 Rx Good Packet Count  : addr(0xd8000088): 0x0000770e
SPI Port 8 Rx Bad Packet Count   : addr(0xd800008c): 0x00000000
SPI Port 8 Tx Good Packet Count  : addr(0xd8000090): 0x00007aa0
SPI Port 8 Tx Bad Packet Count   : addr(0xd8000094): 0x00000000

SPI Port 9 Status                : addr(0xd8000098): 0x00008000
SPI Port 9 Rx Good Packet Count  : addr(0xd800009c): 0x00000000
SPI Port 9 Rx Bad Packet Count   : addr(0xd80000a0): 0x00000000
SPI Port 9 Tx Good Packet Count  : addr(0xd80000a4): 0x00000000
SPI Port 9 Tx Bad Packet Count   : addr(0xd80000a8): 0x00000000

SPI Port 10 Status               : addr(0xd80000ac): 0x00008200
SPI Port 10 Rx Good Packet Count : addr(0xd80000b0): 0x00000000
SPI Port 10 Rx Bad Packet Count  : addr(0xd80000b4): 0x00000000
SPI Port 10 Tx Good Packet Count : addr(0xd80000b8): 0x00000000
SPI Port 10 Tx Bad Packet Count  : addr(0xd80000bc): 0x00000000

------------------------ Octeon Debug Level---------------------------
Current Debug Flags: CVMX Debug: 0x20000 PPM Error: 0x0 PPM Debug: 0x0
PPM Bypass: 0x0
```

```
--------------------- Octeon Debug Statistics-------------------------

---   Total Debug Statistics   ---

Debug:
normal_debug:         62845
pci_debug:            2
pkt_debug:            31422
ppm_debug:            61927
host_debug:           156
send_events:          913

Warnings:

Failures:

Errors:

PPMs:
ppmid_debug[2]:       33
ppmid_debug[3]:       23
ppmid_errors[4]:      30036
ppmid_debug[5]:       2737

Exceptions:

Interrupts:
----------------------------------

------------------- Octeon Command Statistics-----------------------

---   Command Statistics   ---
flow_adds:             9
flow_add_acks:         9
flow_modifys:          3
flow_modify_acks:      3
flow_querys:           6
flow_query_acks:       6
flow_deletes:          4
flow_delete_acks:      4
stat_cmds:             0
stat_cmd_acks:         0
reset_stat_cmds:       0
reset_stat_cmd_acks:   0
device_cmds:           0
device_cmd_acks:       0
print_cmds:            0
print_cmd_acks:        0

------------------ Octeon Core Statistics---------------------------

---   Core Statistics   ---
                                                 Processing(usec)
Usage %
Core input_packets output_packets status avg min max avg min max now
 0   22            22             On     29  13  84  2   0   2   2
 1   0             0              On     0   -1  0   2   0   2   2
 2   29195         28329          On     4   5   23  2   0   2   2
 3   722           699            On     4   5   9   2   0   2   2
 4   500           490            On     4   5   9   2   0   2   2
 5   260           255            On     4   5   9   2   0   2   2
 6   88            87             On     5   5   10  2   0   2   2
 7   47            46             On     5   5   9   2   0   2   2
 8   8             6              On     5   5   8   2   0   2   2
 9   36            35             On     4   5   9   2   0   2   2
10   3             2              On     5   6   10  2   0   2   2
11   10            10             On     5   5   10  2   0   2   2
12   37            36             On     5   5   10  2   0   2   2
13   1             1              On     10  10  10  2   0   3   2
14   14            14             On     5   5   10  2   0   2   2
15   31            30             On     5   5   10  2   0   3   2
       ---            ---             ---
Total: 30974          30062           16

------------------- Octeon PPM Statistics----------------------------
--------------- SRTP_E stats -------------------
alloc-count:            50
dealloc-count:          16
init-requests:          4
init-success:           4
init-fail:              0
modify-request:         0
modify-success:         0
modify-fail:            0
delete-request:         2
delete-success:         2
delete-fail:            0
query-request:          0
query-success:          0
query-fail:             0
protect-fail:           0
unprotect-fail:         0
-----------------------------------------------
--------------- SRTP_D stats -------------------
alloc-count:            50
dealloc-count:          16
init-requests:          2
init-success:           2
init-fail:              0
modify-request:         1
modify-success:         1
modify-fail:            0
delete-request:         0
delete-success:         0
delete-fail:            0
query-request:          2
query-success:          2
query-fail:             0
protect-fail:           0
unprotect-fail:         0
-----------------------------------------------

---   Total IPT Statistics   ---
```

```
      input_packets:  912
      output_packets: 912

      ------------------- Octeon Memory Stats----------------------------

      ---   Buffer Pool Statistics   ---

      pool   size    number     available     inuse
      0      9216    81920      81812 (100)   8
      1      128     102400     102348(50 )   2
      2      1024    81920      81905 (0  )   15
      3      128     81920      81920 (0  )   0
      4      9216    10240      10240 (0  )   0
      5      128     81920      81918 (0  )   2
      6      512     81920      81920 (0  )   0
      7      2048    81920      81920 (0  )   0

      Memory Details:
      Memory  Size:   4010  MB
      FW Init Size:   1187  MB
      PPMs Init Size: 192   MB
      Memory Avail:   2631  MB (2694724 KB)


      -------------------- Octeon Port Statistics--------------------------


---   Port Statistics   ---
   Total active ports: 11
                 |      0|    1|      2|    3|      4|   5|      6|   7|  32|     33|34|Totals
Raw packets      |  15225|  228|  15224|  227|      0|   0|      0|   0|  22|      0| 0|30926
Octets           |3421656|27756|3504819|27644|      0|   0|      0|   0|6724|      0| 0|6988599
Runt packets     |     46|   37|     46|   37|      0|   0|      0|   0|   0|      0| 0|166
Multicast packets|    228|  228|    227|  227|      0|   0|      0|   0|   0|      0| 0|910
64B packets      |     48|   37|     46|   37|      0|   0|      0|   0|  10|      0| 0|178
65B-127B packets|    229|  228|    515|  227|      0|   0|      0|   0|   5|      0| 0|1204
128B-255B packets| 14997|    0|  14997|    0|      0|   0|      0|   0|   0|      0| 0|29994
256B-511B packets|      2|    0|      0|    0|      0|   0|      0|   0|   0|      0| 0|2
512B-1023B packets|     1|    0|      2|    0|      0|   0|      0|   0|   7|      0| 0|10
1024B-1518B packets|    2|    0|      0|    0|      0|   0|      0|   0|   0|      0| 0|2
Rx errors        |     46|   37|     46|   37|      0|   0|      0|   0|   0|      0| 0|166
Rx good          |  15233|  228|  15514|  227|      0|   0|      0|   0|  22|      0| 0|31224
Rx octets        |3421656|27756|3504819|27644|      0|   0|      0|   0|6724|      0| 0|6988599
Rx packets       |  15279|  265|  15560|  264|      0|   0|      0|   0|  22|      0| 0|31390
Tx packets       |      0|    0|      0|    0|  15051|  37|  15333|  37|   0|    911| 0|31369
Tx octets        |      0|    0|      0|    0|3336132|2220|3299431|2220|   0|116544| 0|6756547

      ---------------------------- Host CVMX Statistics----------------------
      -----------------

      ---   nPApp_cvmx_stats   ---
      flow_adds:             9
      flow_add_acks:         9
      flow_modifys:          3
      flow_modify_acks:      3
      flow_deletes:          4
      flow_delete_acks:      4
      flow_querys:           6
      flow_query_acks:       6
```

```
---------------------------- Octeon Host Statistics--------------------
-----------------

Host Async stats:
Octeon Async Tx Packets:        0
Octeon Async Tx Data:           0
Octeon Async Tx Failed Packets: 0
Octeon Async Tx Empty Messages: 0
Octeon Async Rx Device:         0
Octeon Async Rx Event:          913
Octeon Async Rx Data:           0
Octeon Async Rx Error:          0
Octeon Droq Packets:            495
Octeon Command Tx Packets:      22
Octeon Command Tx Failed:       0
Octeon Tx MsgQ Tx Failed:       0
Octeon Tx MsgQ Rx Failed:       0
Octeon Tx MsgQ Delays:          0

Host DROQ 0 stats:
Dev stats:
interrupts:   935
poll_count:   396454
comp_tasklet_count:  0
droq_tasklet_count:  914
cntq_tasklet_count:  0

droq = 0x439AC40
host_read_index:    0
octeon_write_index: 0
host_refill_index:  0
pkts_pending:       0
max_count:          128
refill_count:       0
refill_threshold:   64
pkts_received:      0
bytes_received:     0
dropped_nodispatch: 0
dropped_nomem:      0
dropped_toomany:    0

Host DROQ 1 stats:
Dev stats:
interrupts:   935
poll_count:   396490
comp_tasklet_count:  0
droq_tasklet_count:  914
cntq_tasklet_count:  0

droq = 0x439AD40
host_read_index:    17
octeon_write_index: 0
host_refill_index:  0
pkts_pending:       0
max_count:          128
```

```
refill_count:       17
refill_threshold:   64
pkts_received:      913
bytes_received:     109496
dropped_nodispatch: 0
dropped_nomem:      0
dropped_toomany:    0

Host MBLK pool:
type        number
---------   ------
FREE    :    4999
DATA    :      1
TOTAL   :    5000
number of mbufs: 5000
number of times failed to find space: 0
number of times waited for space: 0
number of times drained protocols for space: 0
__________________
CLUSTER POOL TABLE
______________________________________________________________________
size      clusters  free      usage     minsize   maxsize   empty
----------------------------------------------------------------------
10172     5000      4999      913       10000     10000     0
----------------------------------------------------------------------
task done

SBASNQ06# show support-info contains the following useful ETC related
commands:
show media host-stats
show media host-stats
show media classify
ipt show all
show ip connection
show mbcd all
show security ipsec debug
```

References

[1] Oracle, “Net-Net 4000 S-C6.2.0 ACLI Configuration Guide,” 400-0061-62, Nov 2009.

[2] Oracle, “Net-Net 4000 S-C6.2.0 Maintenance and Troubleshooting Guide,” 400-0063-62, Nov 2009.

[3] Oracle, “Net-Net 4000 S-C6.2.0 Accounting Guide,” 400-0015-62, Nov 2009.

[4] Oracle, “Net-Net 4000 S-C6.2.0 Administrative Essentials,” 400-0132-00, Nov 2009.

[5] Oracle, “Net-Net 4000 S-C6.2.0 Release Notes,” 400-0066-62, Nov 2009.

[6] “RFC 3711, The Secure Real-time Transport Protocol (SRTP)"

[7] “RFC 4568, Session Description Protocol (SDP, Security Descriptions for Media Streams"

[8] “RFC 3830, MIKEY: Multimedia Internet KEYing"

[9] “RFC 3264, An Offer/Answer Model with the Session Description Protocol (SDP)"