Cisco Prime Access Registrar 9.1.1 Release Notes

Cisco Prime Access Registrar (Prime Access Registrar) is a high performance, carrier class, 3GPP-compliant, 64-bit RADIUS/Diameter solution that provides scalable, flexible, intelligent authentication, authorization, and accounting (AAA) services.

Prime Access Registrar comprises a RADIUS/Diameter server designed from the ground up for performance, scalability, and extensibility for deployment in complex service provider environments including integration with external data stores and systems. Session and resource management tools track user sessions and allocate dynamic resources to support new subscriber service introductions.

Note Prime Access Registrar can be used with Red Hat Enterprise Linux (RHEL) 7.7 or CentOS 6.5 and above operating system. Prime Access Registrar has no special OS dependencies; therefore there are no restrictions from upgrading to newer releases of RHEL or CentOS.

Contents

This release note contains the following sections:

System Requirements

This section describes the system requirements to install and use the Prime Access Registrar software.

Table 1 lists the system requirements for Prime Access Registrar 9.1.1.

Table 1 Minimum Hardware and Software Requirements for Prime Access Registrar Server

OS Version	RHEL 7.7 CentOS 6.5 and above **Note** Prime Access Registrar has no special OS dependencies; therefore there are no restrictions from upgrading to newer releases of RHEL or CentOS.
Model	X86
CPU Type	Intel Xeon CPU 2.30 GHz
Processors	4
CPU Speed	2.30 GHz
Memory (RAM)	8 GB
Swap Space	10 GB
Disk Space	1*146 GB

Prime Access Registrar supports JDK versions 1.7 and 1.8.

Note These are the minimum system requirements to have Prime Access Registrar up and running. This may vary based on the deployments. Please contact your BU team to know the specific system requirements for your deployment.

Co-Existence With Other Network Management Applications

To achieve optimal performance, Prime Access Registrar should be the only application running on a given server. In certain cases, when you choose to run collaborative applications such as a SNMP agent, you must configure Prime Access Registrar to avoid UDP port conflicts. The most common conflicts occur when other applications also use ports 2785 and 2786. For more information on SNMP configuration, see the "Configuring SNMP" section in the "Configuring Cisco Prime Access Registrar" chapter of the *Cisco Prime Access Registrar 9.1 Administrator Guide*.

Enhanced Features in Cisco Prime Access Registrar 9.1.1

Cisco Prime Access Registrar 9.1.1 provides the following features:

- Queue-Based Throttling Support, page 3
- TCP Buffer Read/Write Configuration Support, page 3
- Logging User IP Information, page 4
- Preallocate Memory for Processing Queue Enhancement, page 4
- Support for Logging Timeout Packets, page 4
- Session Container Capacity Configuration, page 4
- Enhancement in Blade Switchover Startup Processing, page 4

Queue-Based Throttling Support

Prime Access Registrar supports queue-based throttling for Diameter transactions. Using this feature, Prime Access Registrar utilizes one-third of its input queue exclusively for incoming requests. The input queue is determined by the parameter **MaximumNumberofDiameterPackets** configured under ***/Radius/Advanced/Diameter/TransportManagement***.

Rest of the queue is utilized for other outgoing call flows and incoming responses from and to Cisco Prime Access Registrar.

If the number of requests flowing into Prime Access Registrar goes beyond one third of the configured parameter value, Cisco Prime Access Registrar responds with a server busy error (Diameter error code: 3004).

TCP Buffer Read/Write Configuration Support

With this enhancement, Prime Access Registrar allows you to configure TCP read/write buffers that can be applied for TCP connections appropriately to a specific remote server. New parameters are added to Diameter client and remote server configurations to support this feature:

Diameter remote server:

- **TCP-ReadBuffer**—Allows you to configure read buffer socket options for TCP connections initiated to the remote server.
- **TCP-WriteBuffer**—Allows you to configure write buffer socket options for TCP connections initiated to the remote server.

Diameter client:

- **TCP-ReadBuffer**—Allows you to configure read buffer socket options for TCP connections.
- **TCP-WriteBuffer**—Allows you to configure write buffer socket options for TCP connections.

Note When the parameter is set to zero, default kernel settings will take effect.

Logging User IP Information

Prime Access Registrar provides an option to capture the username as part of the aregcmd_log during login failures. The following parameter is added under ***/Radius/Advanced*** to support this feature:

DisplayUserForFailedLogin—If this option is enabled, during login failures, username is captured along with the failure reason as part of the aregcmd_log.

Also, for all the configuration, login, and logout activities, Prime Access Registrar displays the end user IP in the aregcmd_log. With this enhancement, Prime Access Registrar will start logging the end user's IP address, which is the immediate first hop IP for every aregcmd activity.

Preallocate Memory for Processing Queue Enhancement

This enhancement allows initializing Prime Access Registrar processing queue proportional to the input queue, which is the **MaximumNumberofDiameterPackets** value configured under ***/Radius/Advanced/Diameter/TransportManagement***. This enables better traffic handling for Prime Access Registrar soon after startup.

Support for Logging Timeout Packets

With this enhancement, Prime Access Registrar starts logging information about all packets that timeout i.e. packets which are not responded to within the specified timeout period.

A sample log file content is provided below:

```
07/04/2020 14:08:35.904 name/radius/1 Info System 0 Remote Server REM_76 has not responded
Cmd code: 303 request for user-name 97000000051 in 1 try
```

Session Container Capacity Configuration

With this enhancement, Prime Access Registrar session containers are initialized to hold the number of sessions based on a configurable parameter value, thereby enabling on-demand growth from thereon. Following parameter is added under ***/Radius/Advanced*** to support this feature:

InitialSessionBufferSize—Administrator can configure the estimated session capacity, which that particular Prime Access Registrar instance can hold. This indicates the average or estimated value and not the maximum capacity. Setting this to a larger value impacts the startup performance. Setting this value to zero, will disable preallocation and enable on-demand growth of the container.

Enhancement in Blade Switchover Startup Processing

In a cluster environment, during blade switchover scenarios, Prime Access Registrar startup was impacting traffic handling. This enhancement improves Prime Access Registrar startup process during blade switchover scenarios.

Logging System Statistics

With this enhancement, Prime Access Registrar allows you to collect following statistics data for a configured duration:

- CPU Utilization
- Memory Utilization
- NFSIOstats
- TimedOut MAR/SAR/UDR
- Throttled Packets Count

A new parameter **SystemStatsLogFrequencyInSecs** is added under ***/Radius/Advanced/Diameter/TransportManagement***, which when set to a non-zero value, allows you to log the above statistics for the configured duration. By default this value is set to zero. The system statistics are saved in the **system_stats_log** file.

Binding Address Support in Prime Access Registrar

With this feature, a new **BindingAddress** attribute has been added under ***Radius/Advanced/Diameter/TransportManagement*** to allow you to control the local address the server will use for outbound connections. This should be used if the host has a virtual IP address or when the host has multiple addresses to assure the correct address is used for these connections.

Note You can only configure this to be an IPv4 or IPv6 address, not both.

If the configured address is not available at the time when an outbound connection is initiated, the connection fails and the server retries to connect periodically. Ensure that the correct address is configured.

Following is a sample CLI for this feature:

```
[ //localhost/Radius/Advanced/Diameter/TransportManagement ]
    Identity = abc.cisco.com
    BindingAddress = 10.197.66.126
    Realm = abc.com
    WatchdogTimeout = 500
    ValidateIncomingMessages = FALSE
    ValidateOutgoingMessages = TRUE
    MaximumNumberofDiameterPackets = 8192
    ReserveDiameterPacketPool = 0
    DiameterPacketSize = 2048
    SystemStatsLogFrequencyInSecs = 0
    AdvertisedHostName/
    SCTPOptions/
```

Changes to Diameter-TCP Connection Parameters

Enhancements are made to server connection handling to ensure better response and reporting of issues, if any. As part of this enhancement, new TCP connection parameters (timeout and count) are introduced to control how long the server can wait to receive a Diameter message. If no data is received within the timeout period for the count attempts, the server considers the connection down.

If you increase the time, this may reduce how quickly the server reacts to other connection issues (such as if closed because of a send failure). If you increase the count, the server can be more resilient to short- term network issues (such as link flaps).

Generally, it is recommended to adjust the count and leave the time at the default value.

The following fields are introduced to Diameter-TCP connection parameters:

- **ConnectionTimeout**—Indicates the time (in milliseconds) the server can wait to receive data. Setting this to a lower value can make the server more responsive to connection failures. Default value is 30,000 (milliseconds).
- **MaxTimeoutCount**—Indicates the number of times in a row the server receive must timeout before the connection is closed. Increasing this value can reduce connection closes if there are frequent short connectivity outages. The default value is 3.

Following is a sample CLI for this feature:

```
[ //localhost/Radius/Clients/diaclient ]
    Name = diaclient
    Description =
    Protocol = diameter
    EnableMultiProxyMode = FALSE
    HostName = 10.197.66.126
    PeerPort = 3868
    Vendor =
    IncomingScript~ =
    OutgoingScript~ =
    AdvertisedHostName =
    UserLogEnabled = FALSE
    TCP-ReadBuffer = 0
    TCP-WriteBuffer = 0
    ConnectionTimeout = 30000
    MaxTimeoutCount = 3
    AdvertisedRealm =
    InitialTimeout = 1000
    MaxIncomingRequestRate = 0
    KeepAliveTime = 0
    AuthSessionStateInASR = State-Maintained
    SCTP-Enabled = FALSE
    TLS-Enabled = FALSE
```

In remote server configuration, existing parameters given below are used for managing the connection timeout:

- **DWatchDogTimeout**—Indicates the time (in milliseconds) the server can wait to receive data. Setting this to a lower value can make the server more responsive to connection failures. Default value is 2,500 (milliseconds).
- **MaxTries**—Indicates the number of times in a row the server receive must timeout before the connection is closed. Increasing this value can reduce connection closes if there are frequent short connectivity outages. The default value is 3.

Following is a sample CLI of the Diameter remote server configuration:

```
[ //localhost/Radius/RemoteServers/diarm ]
    Name = diarm
    Description =
    Protocol = diameter
    HostName = 10.197.66.74
    DestinationPort = 3868
    DestinationRealm = cisco.com
    ReactivateTimerInterval = 20000
    Vendor =
    IncomingScript~ =
    OutgoingScript~ =
    UserLogEnabled = FALSE
    TCP-ReadBuffer = 0
    TCP-WriteBuffer = 0
    MaxTries = 3
    MaxTPSLimit = 0
    MaxSessionLimit = 0
    InitialTimeout = 2000
    DisconnectBasedOnThreshold = TRUE
    DisconnectThreshold = 1
    LimitOutstandingRequests = FALSE
    MaxPendingPackets = 0
    MaxOutstandingRequests = 0
    DWatchDogTimeout = 2500
    SCTP-Enabled = FALSE
    TLS-Enabled = FALSE
    AdvertiseHostName =
    AdvertiseRealm =
```

SNMP Trap Support for Throttling Active and Inactive States

A new parameter **ThrottlingMonitorFrequencyInSecs** is introduced under RADIUS/***Advanced/Diameter/TransportManagement/*** to support this feature. Prime Access Registrar monitors whether traffic is throttled every second over the configured interval. If throttling occurs for at least half of the configured seconds, a throttling trap is sent from Prime Access Registrar. E.g. if the configured value is 60 seconds, and throttling occurs for at least 30 seconds during the configured duration of 60 seconds, then throttling trap is sent from Prime Access Registrar. When no throttling occurs during the entire interval, a throttling reset trap is sent.

By default, **ThrottlingMonitorFrequencyInSecs** is set to zero (0), which indicates that throttling trap functionality is disabled and throttling traps should not flow even if throttling conditions are met. Minimum non-zero value that can be configured is **20**.

Following is a sample CLI for this feature:

```
[ //localhost/Radius/Advanced/Diameter/TransportManagement ]
    Identity = 2016::fbcc:b1ed:4930:2ddd
    BindingAddress = 2016::fbcc:b1ed:4930:2ddd
    Realm = cisco.com
    WatchdogTimeout = 2500
    ValidateIncomingMessages = FALSE
    ValidateOutgoingMessages = TRUE
    MaximumNumberofDiameterPackets = 800
    ReserveDiameterPacketPool = 0
    DiameterPacketSize = 4096
    SystemStatsLogFrequencyInSecs = 15
    ThrottlingMonitorFrequencyInSecs = 0
    EnablePreemptiveRecovery = False
    AdvertisedHostName/
    SCTPOptions/
```

Following traps are introduced for this feature:

- **carThrottlingTrap**—Indicates that throttling has kicked in and has lasted for half of the time as per the configured value.
- **carThrottlingResetTrap**—Indicates that throttling has settled down and Prime Access Registrar has recovered.

Logging Worker Queue Size

Reporting of **All Workers Temporarily Busy** warning has been added to the System Stats Log under the parameter **Peak Worker Thread Queue / sec,** and is only reported if the condition has occurred during the last statistics interval.

Support for Preemptive Recovery

The Preemptive recovery enhancement addresses the automatic recovery of Prime Access Registrar when it enters into a presumed unrecoverable state. Following are the conditions when Prime Access Registrar can enter into a presumed unrecoverable state:

- The number of incoming DER EAP-AKA Challenge (DER2) being processed by Prime Access Registrar is less than 10% of the successful DER EAP-AKA responses being sent for the initial Identity request (DEA1).
- The DER EAP-AKA responses being sent exceed a certain configured limit (default 5000 over a period of 2 minutes) for the condition to be triggered to account for low traffic conditions.

Following are the parameters introduced to support this feature:

- **EnablePreeemptiveRecovery**—If set to TRUE, indicates that preemptive recovery feature is enabled for Prime Access Registrar. By default, this parameter is disabled.
- **MinDEA1Threshold**—Indicates the minimum number of DEA EAP Multi-Round Auth (DEA1) responses sent over the past 120 seconds, that will kick off the preemptive recovery condition check. This parameter is available only if **EnablePreeemptiveRecovery** is set to TRUE. Default value is 5000.

When the **EnablePreeemptiveRecovery** parameter is enabled and the presumed unrecoverable state is detected, Prime Access Registrar sends a **PreemptiveRecovery Trap** and restarts the RADIUS process. This trap indicates that preemptive recovery has been initiated because the number of DER EAP-AKA Challenge (DER2) received by Prime Access Registrar is less than 10% of the successful DER EAP-AKA responses being sent for the initial Identity request (DEA1).

Following is a sample CLI for this feature:

```
[//localhost/Radius/Advanced/Diameter/TransportManagement ]
    Identity = 10.197.66.75
    BindingAddress =
    Realm = epc.mnc854.mcc405.3gppnetwork.org
    WatchdogTimeout = 2500
    ValidateIncomingMessages = FALSE
    ValidateOutgoingMessages = TRUE
    MaximumNumberofDiameterPackets = 16388
    ReserveDiameterPacketPool = 0
    DiameterPacketSize = 4096
    SystemStatsLogFrequencyInSecs = 10
    EnablePreemptiveRecovery = true
    MinDEA1Threshold = 5000
    AdvertisedHostName/
    SCTPOptions/
```

Support for Duplicate Authentication Request Detection

With this enhancement, Prime Access Registrar can detect duplicate authentication requests based on UE session ID. If any diameter request packet has a Session ID same as that of a packet that is already being processed, the new request is silently dropped/ignored from processing.

A new parameter **EnableDuplicateSessionIdDetection** is introduced under ***/Radius/Advanced/*** to support this functionality. By default, this parameter is enabled.

This enhancement is primarily provided so that the server does not respond with a 3004 (Diameter Too Busy) status for a request that is already in progress; instead drop the duplicate request packet silently.

```
[ //localhost/Radius/Advanced ]
    LogServerActivity = FALSE
    TLSv1Enabled = TRUE
    MaximumNumberOfRadiusPackets = 8192
    UDPPacketSize = 4096
    SocketWaitTime = 3
    NumberOfRemoteUDPServerSockets = 4
    NumberOfRadiusIdentifiersPerSocket = 256
    PerPacketHeapSize = 6500
    RequireNASsBehindProxyBeInClientList = FALSE
    AAAFileServiceSyncInterval = 75
    SessionBackingStoreSyncInterval = 100
    BackingStoreDiscThreshold = "5 Gigabyte"
    SessionBackingStorePruneInterval = "2 Hours"
    PacketBackingStorePruneInterval = "6 Hours"
    RemoteLDAPServerThreadTimerInterval = 10
    RemoteSigtranServerThreadTimerInterval = 10
    InitialBackgroundTimerSleepTime = 5
    MinimumSocketBufferSize = 65536
    CertificateDBPath =
    LogFileSize = "1 Gigabyte"
    LogFileCount = 20
    TraceFileSize = "1 Gigabyte"
    TraceFileCount = 2
    MemoryLimitForRadiusProcess = "10000 Megabyte"
    UseAdvancedDuplicateDetection = FALSE
    AdvancedDuplicateDetectionMemoryInterval = 10000
    InitialSessionBufferSize = 0
    DetectOutOfOrderAccountingPackets = FALSE
    DefaultReturnedSubnetSizeIfNoMatch = BIGGER
    ClasspathForJavaExtensions =
    JavaVMOptions =
    MaximumODBCResultSize = 256
    ARIsCaseInsensitive = TRUE
    RemoteRadiusServerInterface =
    ODBCEnvironmentMultiValueDelimiter =
    PacketBackingStoreSyncInterval = 75
    ListenForDynamicAuthorizationRequests = truE
    MaximumNumberOfXMLPackets = 1024
    XMLUDPPacketSize = 4096
    RollingEncryptionKeyChangePeriod = "1 week"
    SessionPurgeInterval = "6 Hours"
    EapBadMessagePolicy = SilentDiscard
    StaleSessionTimeout = "1 Hour"
    MaximumOutstandingRequests = 0
    MaximumIncomingRequestRate = 0
    HideSharedSecretAndPrivateKeys = falSE
    DefaultRadiusSharedSecret =
    ServerStatusSharedSecret = Hardlyasecret
    EnableLocationCapability = FALSE
    LogTPSActivity = TRUE
    TPSLogFileCount = 15
    TPSLogFilenamePrefix = tps
    TPSSamplingPeriodInSecs = 5
    LogSessionActivity = TRUE
    EnableLengthFlag = FALSE
    SessionLogFileCount = 15
    SessionLogFilenamePrefix = sm
    SessionSamplingPeriodInSecs = 30
    FlushDiskInBackground = TRUE
    AdditionalNativeOracleErrors =
    SendOpCodeInISDResponse = FALSE
    EnableRoutingContextInM3UA = FALSE
    EnableStickySessionCount = TRUE
    ServerMonitorAltApproach = FALSE
    EnableSIGTRANStackLogs = TRUE
    SIGTRANStackLogFileSize = "100 Megabyte"
    SIGTRANLogFileCount = 10
    StickySessionCountInterval = 60000
    StickySessionSyncInterval = 500
    ReserveRADIUSPacketPool = 0
    UserLogDelimiter = ,
    DiameterStaleSessionPurgeTime = 00:00:00
    UISessionTimeoutInMins = 0
    DiameterStaleConnectionDeletionTimeOut = 300000
    DiameterSessionRestorationPurgeTime = 01:30:00
    DisplayUserForFailedLogin = TRUE
    EnableDuplicateSessionIdDetection = TRUE
    Ports/
    Interfaces/
    ReplyMessages/
    Attribute Dictionary/
    SNMP/
    ServerMonitor/
    RemoteSessionServer/
    RFCCompliance/
    DDNS/
    ODBCDataSources/
    AttributeGroups/
    KeyStores/
    Diameter/
    DiameterDictionary/
```

Configuring Unique TAG number for Vendor-Specific Sub Attributes

With this enhancement, Prime Access Registrar allows you to configure a unique tag number for each of the multiple vendor-specific sub-attributes available for a user profile. This enhancement is applicable for multi-tag valued vendor-specific attributes with type as TAG_STRING and TAG_INT.

A sample configuration is shown below:

```
[ //localhost/Radius/Profiles/base-avp/Attributes ]
    unisphere-activate = HQOS-RES
    unisphere-activate-2 = DHCP-IPOE-DATA-PROFILE
    unisphere-activate-3 = DHCP-IPOE-VIDEO-PROFILE

[ //localhost/Radius/UserLists/Default/jane ]
    Name = jane
    Description =
    Password = <encrypted>
    Enabled = TRUE
    Group~ = Telnet-users
```

```
BaseProfile~ = base-avp
AuthenticationScript~ =
AuthorizationScript~ =
UserDefined1 =
AllowNullPassword = FALSE
Attributes/
CheckItems/
```

Registering Prime Access Registrar as a Service in RHEL SystemD Service Management

With this enhancement, Prime Access Registrar is added as a service in SystemD service management. After successful installation, Prime Access Registrar gets registered as a service in SystemD unit, after which you can execute the below commands to start, stop, or restart Prime Access Registrar and to find the status of the Prime Access Registrar server:

```
systemctl start arserver
systemctl stop arserver
systemctl restart arserver
systemctl status arserver
```

This feature is supported only from RHEL 7.7. After successful registration of Prime Access Registrar as a service in SystemD, you should refrain from using the older method of starting, stopping, or restarting the server using arserver script (/cisco-ar/bin/arserver start, etc.).

Cisco Prime Access Registrar 9.1.1 Bugs

This section contains the following information:

- Fixed Anomalies in Cisco Prime Access Registrar 9.1.1.1, page 11
- Using the Bug Search Tool, page 12

Fixed Anomalies in Cisco Prime Access Registrar 9.1.1.1

Table 2 lists the anomaly fixed in Prime Access Registrar 9.1.1.1 release.

Table 2 Fixed Anomaly in Prime Access Registrar 9.1.1.1

Bug	Description
CSCvx32757	EAP-Master-Session-Key is not available in Diameter EAP Answer (DEA).

Using the Bug Search Tool

Use the Bug Search tool (BST) to get the latest information about Cisco Prime Access Registrar bugs. BST allows partners and customers to search for software bugs based on product, release, and keyword, and it aggregates key data such as bug details, product, and version.

BST allows you to:

- Quickly scan bug content
- Configure e-mail notifications for updates on selected bugs
- Start or join community discussions about bugs
- Save your search criteria so you can use it later

When you open the Bug Search page, check the interactive tour to familiarize yourself with these and other Bug Search features.

Step 1 Log into the Bug Search Tool.

a. Go to https://tools.cisco.com/bugsearch.

b. At the Log In screen, enter your registered Cisco.com username and password; then, click **Log In**. The Bug Search page opens.

Note If you do not have a Cisco.com username and password, you can register for them at http://tools.cisco.com/RPF/register/register.do.

Step 2 To search for a specific bug, enter the bug ID in the Search For field and press **Return**.

Step 3 To search for bugs in a particular release:

a. In the Search For field, enter the product name and the release version, e.g.
Cisco Prime Access Registrar 9.1.1, and press **Return**. (Leave the other fields empty.)

b. When the search results are displayed, use the filter and sort tools to find the types of bugs you are looking for. You can search for bugs by severity, by status, how recently they were modified, according to the number of support cases associated with them, and so forth.

Related Documentation

For a complete list of Cisco Prime Access Registrar documentation, see the *Cisco Prime Access Registrar 9.1 Documentation Overview*.

Note We sometimes update the documentation after original publication. Therefore, you should also review the documentation on Cisco.com for any updates.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.