

# sipXcom 18.04

May 8, 2018

- [Summary](#)
  - [sipXcom New Features:](#)
  - [sipXcom Improvements:](#)
- [Notes](#)
- [Who Should Install?](#)
- [Questions](#)
- [Specific Issues Addressed](#)
- [New Installs](#)
- [Upgrade from previous versions](#)
  - [New Procedure](#)
  - [Warning!](#)
  - [Upgrade to 18.04 for Single Server systems](#)
    - [Backup your System](#)
    - [Download upgrade.sh](#)
    - [Run upgrade.sh](#)
  - [Upgrade to 18.04 for Multi-Server Clusters with Multiple Database Servers](#)
    - [Backup your System](#)
    - [Remove All Secondary Databases from Cluster Nodes](#)
    - [Proceed as with Single Server for Primary](#)
    - [Secondary Servers in Cluster](#)
    - [Add Database Nodes](#)
    - [Send Server Profiles](#)

## Summary

**Warning! Read these notes closely! There is important upgrade information!**

eZuce is pleased to announce the GA Release of sipXcom 18.04.

Dev & QA teams at eZuce have been busy as usual. We put the next generation work on hold to ensure growth to 30,000+ users for one of our large customers. The biggest change is related to Mongo. MongoDB has been upgraded to version 3.4. This has the advantage of running [Mongo's WiredTiger Storage Engine](#). This brings with it document level locking vs. the entire collection as in MongoDB version 2.6. Additionally, database sizes will grow and shrink according to what data is stored in the database (vs. just growing forever).

In 18.04 you'll also find some LDAP improvements, several new REST improvements and some performance tweaks.

Make sure you see Note #2 below regarding disk space and Note #3!

## Highlights

### sipXcom New Features:

- [MongoDB Version 3.4](#)

### sipXcom Improvements:

- [Download entire CDR reports](#)
- [Several REST API Improvements](#)
- [LDAP Username import manipulation - Strip, Append, Prepend](#)
- [LDAP Import Settings now per LDAP Server](#)
- [Proxy Statistics Manager enhancements to provide additional statistics](#)
- [Make EntityDB cache expiration configurable](#)
- [Mark SELinux as enforcing and add Management to Admin GUI](#)
- [Performance related change that re-routes SIP Subscription Traffic to bypass Registrar](#)
- [Performance option to disable Proxy DNS Lookups](#)

## Notes

1. Please note that the upgrade will take extra disk space. The MongoDB upgrade essentially makes a copy of the existing database and then re-mounts it. Please be prepared for this.
2. See special note below under Update for upgrades with multiple database nodes.
3. Pay special attention to full release notes for new upgrade instructions! You will break your server if you don't!
4. Please see known issue UC-4721 listed below in the table.

## Who Should Install?

This release is recommended for all 4.6 and later installations.

## Questions

Please post to the sipXcom-users google group if you have questions.

<https://groups.google.com/forum/#!forum/sipxcom-users>

## Specific Issues Addressed

Specific issues can be located in the detailed release notes in the wiki at: <http://wiki.sipxcom.org/display/sipXcom/sipXcom+18.04>

## New Installs

ISO's are available at: <http://download.sipxcom.org/pub/sipXecs/ISO/>

The ISO's are built on a CentOS 6.7 base so please yum update the system after installation.

## Upgrade from previous versions

### New Procedure

We will be utilizing an upgrade script to ensure upgrades proceed as intended and so that customers have the appropriate warnings and information before upgrading.

### Warning!

Make sure you **backup** your system (configuration and voicemail at a minimum) prior to installation. You'll be upgrading mongodb to a new version!

## Upgrade to 18.04 for Single Server systems

### Backup your System

Login to the Admin GUI and click on System -> Backup and at a minimum backup configuration and voicemail.

**Do it now... before you go any further.**

### Download upgrade.sh

Login to the primary server as root.

Execute the following:

```
wget http://download.sipxcom.org/upgrade.sh
chmod +x upgrade.sh
```

### Run upgrade.sh

Execute the upgrade script and answer 'Yes' to continue:

```
./upgrade.sh
```

The following will be displayed:

## sipXcom 18.04 Upgrade Script

IMPORTANT: If this is a multi-server cluster, all databases except the Primary (which must be on the configuration server) should be removed.

IMPORTANT: You should run a system backup and copy your config and voicemail backups to another system. If the upgrade fails, you will need to build a new server and restore from backup.

IMPORTANT: Ensure that you have enough disk space available for a copy of the Mongo databases. (roughly your Config + Voicemail backups).

This script will do the following:

- Back up mongo config and dbs
- Stop mongo instance and remove old mongo files
- Change 17.10 occurrences in /etc/yum.repos.d with 18.04
- Perform yum update and then reboot the machine

On sipxconfig service startup following steps are taken (in case there is a backup still on the disk):

- Restore mongo config and dbs, then remove from disk
- Reboot machine

For other cluster servers:

- Run same script
- Re-add databases that were removed.

Continue? (you must enter Yes or No as shown and press Enter):

Enter 'Yes' and press Enter to continue. The system will reboot a couple of times as part of the process.

## Upgrade to 18.04 for Multi-Server Clusters with Multiple Database Servers

### Backup your System

Login to the Admin GUI and click on System -> Backup and at a minimum backup configuration and voicemail.

**Do it now... before you go any further.**

### Remove All Secondary Databases from Cluster Nodes

From the Admin UI remove databases from all nodes **except** for the primary node. Click on System -> Databases do accomplish this.

### Proceed as with Single Server for Primary

Proceed as above with the 'Download upgrade.sh' section and then the 'Run upgrade.sh' section.

### Secondary Servers in Cluster

After the Primary Server has completed, repeat for each of the Secondary Servers in the cluster until all are completed.

### Add Database Nodes

After the secondary nodes are complete and done with their reboots, login to the Admin UI and add back the database nodes that were removed.

### Send Server Profiles

Login to web interface as superadmin.

Navigate to System -> Servers page. Place checkmark next to server names and click 'Send Profiles'.

## Issues Addressed

	JIRA name	RN Content	Enhancement/Fix/Known Issue	Keywords
SI P X- 695	LDAP improvement	Currently "Username strip" under LDAP management applies to all LDAP imports. An administrator would like for this to be associated with each LDAP server.	Enhancement	LDAP
SI P X- 707	Upgrade MongoDB to 3.4	Why: - WiredTiger storage, no allocation of disk - no more collection global lock but document lock (should greatly improve performance, get benchmarks from older versions vs 18.04) - better usage of CPU cores - up to 50 nodes in replica set  ToDo: - provide latest MongodB RPMs on sipXcom install - rework sipxecs-setup procedure - adapt database management from sipXconfig - adapt backup/restore procedures - adapt local regions feature - test basic functionality - full regression testing & bug fixing	Enhancement	Mongo
SI P X- 713	Add Phone Management to REST API	This is a request to add the ManagePhone functions that are currently in the SOAP API to the REST API. Specifically the generateProfiles and restart capabilities, although all of them would be nice.  Examples: curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/user/id/10/sendProfile curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/user/id/10/sendProfile/restart curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/user/name/200/sendProfile curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/user/name/200/sendProfile/restart curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/phoneGroup/id/18/sendProfile curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/phoneGroup/id/18/sendProfile/restart curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/phoneGroup/name/phgr1/sendProfile curl --insecure -X PUT https://superadmin:11111111@localhost/sipxconfig/api/phones/phoneGroup/name/phgr1/sendProfile/restart	Enhancement	API
SI P X- 717	CallQueue Default for Recording	Change CallQueue Default for Recording to Disabled.  At present default is Enabled and recording directory is blank which can cause a problem.	Enhancement	Call Queue
SI P X- 719	Add search phone based on user / line to REST API	Add ability to retrieve phones for an user / line through REST API	Enhancement	API
SI P X- 720	REST API - Add ability to search users based on email	Improve users rest api to retrieve users based on email field.	Enhancement	API
SI P X- 721	Add option to disable Proxy DNS lookups	This is a performance tweak that can make a big difference with the performance of very large systems.  - Add option in UI needed to disable registrar and MWI lookups from proxy - In case option checked then generate forwardingrules.xml with IP:port instead DNS record. For Registrar that should be easy as proxy and registrar are automatically enabled on same node. Figure out IP address to point to for MWI server. - By default, option is disabled	Enhancement	Proxy
SI P X- 722	Registrar fails to start when 400k subscription in mongo database	On startup all components using SipSubscribeServer try to retrieve and load in memory all entries within database.  In case of SIPX-722 there were too many records to fit in returned cursor so that one failed with  "2018-01-31T11:00:03.256938Z":22:SIP:ERR:int.phx2.ezuce.ro:SipSubscribeServer-29:7f5b7de25700:sipxregistry:"SipSubscribeServer::initialize failed - retrying after 500 milliseconds" "2018-01-31T11:00:03.860614Z":23:SIP:EMERG:int.phx2.ezuce.ro:SipSubscribeServer-29:7f5b7de25700:sipxregistry:"ALARM_MONGODB_SLOW_READ Last Mongo read took a long time: document: node.subscription delay: 103 milliseconds"  This caused no registrars to be able to start. This is an edge case that exposed the bug, however, using such approach of loading all subscribes from database when process starts (and assuming number of records can fit in mongo cursor) could in normal operation result in: - slow operating of mongo database (restarting a process or trying to recover by restarting processes at the same time can nail down entire cluster) - slow start of SIP processes using this approach (checking the code it seems like not only registrar is using same approach, but also MWI, proxy, RLS and SAA)  The query that used to return all registrations from MongoDB (see below) was removed	Fix	Registrar
SI P X- 723	Add possibility to specify urls in product copyright value	An administrator would like to customizing skin (as documented in: <a href="http://wiki.sipxcom.org/display/sipXcom/Customizing+Colors%2C+Layout+and+Logo">http://wiki.sipxcom.org/display/sipXcom/Customizing+Colors%2C+Layout+and+Logo</a> )  The administrator would like to allow the inserting of links in product.copyright value.	Enhancement	Portal
SI P X- 725	Mark selinux as enforcing.	Currently /etc/selinux/config is disabled.  Some customers may require it to be enforcing for security reasons.	Enhancement	System
SI P X- 726	Create proper compound indexes for subscriptions and registrar databases	During latest load / profiling it was noticed that some operations were reported slowly.  - for updating registrations bindings - for updating subscriptions bindings  Comparing queries with collection indexes it can be seen that following fields that are queried were not indexed: - for registrar: contact, shardid - for subscription: fromUri, callId, eventTypeKey  Used compound indexes, see <a href="https://docs.mongodb.com/manual/core/index-compound/">https://docs.mongodb.com/manual/core/index-compound/</a>	Enhancement	Registrar Mongo
SI P X- 727	Update registration bindings using upsert instead delete / insert	Updating registrations was made up from two operations: delete registration then update registration. Subscription database is already using upsert call instead remove / update, change database for updating registrations: <a href="https://github.com/sipXcom/sipxecs/blob/release-18.04/sipXcomserverLib/src/sipdb/SubscribeDB.cpp#L174">https://github.com/sipXcom/sipxecs/blob/release-18.04/sipXcomserverLib/src/sipdb/SubscribeDB.cpp#L174</a>	Enhancement	Registrar Mongo

SI P X- 728	Call ensureIndex only once for registrations and subscriptions databases	Each time a registration is updated / inserted, a call to ensure database indexes is performed. This could be resource consuming and affecting system performance.  More, starting with version 3.0 this call is deprecated and it uses createIndexes call behind the scene, see: <a href="https://docs.mongodb.com/manual/reference/method/db.collection.ensureIndex/#db.collection.ensureIndex">https://docs.mongodb.com/manual/reference/method/db.collection.ensureIndex/#db.collection.ensureIndex</a>  Remove ensureIndex from CRUD operations and keep it only when service starts up.	Enhancement	Registrar Mongo
U C- 46 22	Administrator Dashboard	An Administrator would like to be able to see performance information about the servers operating in their Uniteme system.  sipXproxy Performance Information - SIP Messages per second, Queue Depth, Auth Cache Queue entries sipregistrar Performance Information - # Registrations, # Registration requests per second MongoDB Performance Information - reads / sec, writes / sec, etc. System Performance - Packets / sec, CPU use, memory use, etc.  Display last X messages from important log files.  It's not required to build this dashboard in the current admin gui.	Enhancement	Diagnostics
U C- 46 27	REST API improvement to change group firmware for phone groups	A user would like to change group firmware for phone groups via REST calls.	Enhancement	API
U C- 46 37	Re-route SIP Subscription Traffic	At the moment all SUBSCRIBE messages which are not directed to MWI are explicitly sent to Registrar by default. Registrar forwards subscriptions to SSS using 302. This creates unnecessary traffic load on Registrar and underlying the mongo database.  We will change forwardingrules.xml in a way to forward subscriptions to SSS by default without making another hop with Registrar.  Forwarding rules part: <!-- All other SUBSCRIBE requests go to the SIP registry service --> <routeTo ruriParams="sipx-noroute=Voicemail">&lt;rr.HOSTNAME:0;transport=tcp;x-sipx-routeretoreg&gt;</routeTo>  should be changed to <!-- All other SUBSCRIBE requests go to the SSS service --> <routeTo>&lt;HOSTNAME:5140;transport=tcp&gt;</routeTo>  where HOSTNAME is IP of SSS and 5140 is SSS default port	Enhancement	Registrar
U C- 46 40	Make EntityDB cache expiration configurable (sip code part)	The Proxy Service already has a caching mechanism already implemented. With this Jira we will implement a flexible way to configure the cache timeout used for packets received by the proxy without digest (meaning with no authorization yet). Today, the cache is hardcoded to 30 seconds but it can be set to several minutes depending on the configuration required.  Also to avoid a possible synchronization issue between an admin deleting a user and having the UA not yet removed from the cache (even it is not happening often), we may also consider a simple API call so that when the Administrator performs a delete action, we delete the associated entry in the cache (or maybe a special "SIP delete" message to proxy for example).  Additionally, run tests at various traffic levels starting at 30 seconds to 3 minutes in 30-second intervals, 5 minutes and 10 minutes to pattern behavior and find optimum setting.	Enhancement	Proxy
U C- 46 41	Make EntityDB cache expiration configurable (webui part)	Add to sipxproxy config file (/etc/sipxpbx/sipXproxy-config): name: SIPX_PROXY_ENTITY_CACHE_EXPIRE, default is 30. Add to sipxregistrar config file (/etc/sipxpbx/registrar-config): name: SIP_REGISTRAR_ENTITY_CACHE_EXPIRE, default is 30.  Actually this should be single setting "Entity Cache Expiration" on webui, but value should be extracted into 2 files.	Enhancement	Proxy
U C- 46 44	Add additional data to Proxy Statistics Manager	Extend the new StatisticsManager service with additional metrics we need. Make all information available in the existing .json file.  Count the different SIP Methods, SIP Error Responses and to who, Top SIP 10 talkers overall, and additional information that would be useful for performance monitoring (tbd).  { "timestamp": "2018-02-16T15:50:54.088061Z", "REGISTER": 110 } { "timestamp": "2018-02-16T15:50:54.088076Z", "REGISTERIF": 110 } { "timestamp": "2018-02-16T15:50:54.088091Z", "REGISTERIS": 0 } { "timestamp": "2018-02-16T15:50:54.088101Z", "REGISTERO": 375 } { "timestamp": "2018-02-16T15:50:54.088111Z", "REGISTEROF": 110 } { "timestamp": "2018-02-16T15:50:54.088120Z", "REGISTEROS": 0 } { "timestamp": "2018-02-16T15:50:54.088129Z", "proxy_active_transaction_count": 5255 } { "timestamp": "2018-02-16T15:50:54.088138Z", "proxy_avg_dispatch_speed": 0 } { "timestamp": "2018-02-16T15:50:54.088158Z", "proxy_avg_entity_db_read": 0 } { "timestamp": "2018-02-16T15:50:54.088167Z", "proxy_avg_regdb_read": 0 } { "timestamp": "2018-02-16T15:50:54.088177Z", "proxy_msg_queue_size": 0 } { "timestamp": "2018-02-16T15:50:54.088186Z", "proxy_ua_queue_size": 0 } { "timestamp": "2018-02-16T15:50:54.088195Z", "queue_size": 93 } { "timestamp": "2018-02-16T15:50:54.088204Z", "reqi": 220 } { "timestamp": "2018-02-16T15:50:54.088213Z", "reqo": 1117 } { "timestamp": "2018-02-16T15:50:54.088222Z", "rspl": 247 } { "timestamp": "2018-02-16T15:50:54.088231Z", "rspo": 383 } { "timestamp": "2018-02-16T15:51:00.039017Z", "BYE": 130 } { "timestamp": "2018-02-16T15:51:00.039045Z", "BYEIF": 109 } { "timestamp": "2018-02-16T15:51:00.039064Z", "BYEIS": 0 } { "timestamp": "2018-02-16T15:51:00.039075Z", "BYEO": 706 } { "timestamp": "2018-02-16T15:51:00.039084Z", "BYEOF": 109 } { "timestamp": "2018-02-16T15:51:00.039097Z", "BYEOS": 0 }  Please note that the following metrics are presently constantly 0 : proxy_avg_dispatch_speed, proxy_avg_entity_db_read, proxy_avg_regdb_read, proxy_msg_queue_size, proxy_ua_queue_size.  New Proxy Statistics Manager can be disabled for higher performance. There is a small performance penalty for writing these statistics to the .json file.	Enhancement	Diagnostics
U C- 46 45	LDAP Username import manipulation	An administrator would like more flexibility to manipulate usernames during LDAP import.  1. Add the ability to strip characters from the imported username via regex 2. Add the ability to append a suffix to the imported username 3. Add the ability to prepend a prefix to the imported username	Enhancement	LDAP
U C- 46 60	Sipxsupervisor enables nfs services always	It seems that by default /usr/share/sipxpcsc/cfinputs/plugin.d/sipxcpdumplog.cf enables nfs services by default, even if the tcpdump services are disabled in sipxconfig webui. This leaves the nfs rpc related services running and bound to ports.  As a workaround the administrator could disable rpcbnd and rpgcssd services, moved sipxcpdumplog.cf out of the plugin.d directory, and rebooted.	Fix	System
U C- 46 76	Fix misleading log message queue is over half full	During investigations it turned out that this log is wrong and "max - 100" was a hardcoded and unused value. Determine and log proper message with correct message queue depth.  2018-01-24T13:23:27.343137Z:122375:KERNEL:WARNING:s1.phx2.ezuce.ro:SipClientTcp-21:7fb4a8089700:sipxproxy:OsMsgQShared::doSendCore message queue 'SipClientTcp-20' is over half full - count = 6818 max = 100"	Fix	Proxy
U C- 46 77	Misleading mongodb slow read C-alarms	When the primary server was shutdown (so no mongo for processes to connect to) proxies on running nodes were given ALARM_MONGODB_SLOW_READ and some exact time, e.g. 814 ms. So this is not something reported by mongo (as a reminder mongodb already logs slow queries in mongod.log) and there is the possibility that such alarms to be wrongly raised (as a consequence of slowly processing for example).  - Double check code and correct scenarios where these slow read reports could be wrongly reported	Fix	Proxy

U C- 46 84	Call queue MoH replaces default MoH	<p>Uploading CallQueue MoH seems to override the system moh setting. Navigating to features -&gt; moh and applying default.wav works to correct, however after the default.wav plays the call queue MoH is next in line.</p> <p>For existing queues customer must:</p> <ol style="list-style-type: none"> <li>1. Move all queue audio files from /var/sipxdata/mediaserver/data/moh to /var/sipxdata/mediaserver/data/prompts (use mv comand)</li> <li>or</li> <li>2. Upload the Queue Audio files again in UI and then manually delete old files from /var/sipxdata/mediaserver/data/moh</li> </ol>	Fix	Call Queue
U C- 46 86	REST API improvement: change multiple settings with one rest call	<p>A customer would like to edit phone groups more efficiently via API.</p> <p>A single setting for a phone group can be performed. For example:  <a href="https://localhost/sipxconfig/api/phoneGroups/34/model/polycom450/settings/lcl/time/24HourClock">https://localhost/sipxconfig/api/phoneGroups/34/model/polycom450/settings/lcl/time/24HourClock</a>  with data 1 or 0 and it will update that one setting. What would be helpful would be to update multiple settings in a single API call.</p> <p>Now to change multiple phone group settings in a call:  curl -X PUT --insecure -H "Content-Type: application/json" -d '{"settings":[{"path":"lcl/time/24HourClock","value":"0"}, {"path":"lcl/datetime/date.format","value":"Md, D XX"}]}' <a href="https://superadmin:11111111@localhost/sipxconfig/api/phoneGroups/phgr1/model/polycom330/settings">https://superadmin:11111111@localhost/sipxconfig/api/phoneGroups/phgr1/model/polycom330/settings</a></p>	Enhancement	API
U C- 46 87	Download entire CDR reports (currently limited to 5MB)	<p>An administrator is trying to pull yearly CDR reports via the historic page. No matter what time range is set once it is downloaded file is always MAX 5MB and only shows few days.</p>	Enhancement	CDR
U C- 46 95	When generating very large CSV history or reports, web interface crashes	<p>This has occurred on a server containing ~2,000,000 calls in history. When trying to generate either a history CSV either a report with all the calls, the client browser receives error 502 - Proxy Error. However, the process is still running in the background taking up resources, generating a file that will not be able to be downloaded. The web GUI cannot be contacted anymore and sipxconfig will need a restart.</p> <p>This would definitely be a problem on single node, or if the proxy, registrar and other services are running on the same node.</p> <p>The system the error occurred in is on AWS and contains dual Xeon processors and 8 GB of ram. The system generates a csv with up to 1,000,000 entries in around 20 seconds.</p> <p>25,000 entries - 4.3 MB  200,000 entries - 35 MB  300,000 entries - 52 MB  500,000 entries - 88 MB  1,000,000 entries - 174 MB  Over 1,100,000 entries the system cannot be contacted anymore.</p> <p>The same problem goes for reports as well, system stops responding when you try to generate a report from a time period that has more than 2,000,000 entries.</p>	Fix	CDR
U C- 47 06	SIP Status memory consumption increases infinitely	<p>A memory leak was discovered in sip status - when stressing the system with many subscribers, SIP status keeps increasing in memory consumption. On the test system the consumption of SIP status increased to over 40% of the total memory (8 GB Ram) and system killed the mongod process to free space.</p>	Fix	SIP Status
U C- 47 21	When restoring a backup made on a previous version on another cluster, the servers are not correctly added	<p>A bug has been verified in 17.10 and 18.04 when trying to restore a backup from a previous version of code.</p> <p>To reproduce -</p> <ol style="list-style-type: none"> <li>1. Create a backup on cluster 1 running version 16.12 of UniteMe.</li> <li>2. Restore the backup on cluster 2 running version 18.04 of UniteMe. When doing this, the system -&gt; servers primary is correctly kept, but the servers 2 and 3 are replaced with the ones in 16.12 and hence loose connection becoming uninitialized. Also, system -&gt; databases does not load anymore. Checking rs.status() shows correctly the databases on the nodes before the restore. Also, modifying the domain and IP of the secondary nodes to the correct ones does not work.</li> </ol> <p>WORKAROUND:</p> <ol style="list-style-type: none"> <li>1. Prepare the 18.04 system for restoring from 16.12. Delete the secondary and arbiter databases.</li> <li>2. Restore the config file. After the restore process is complete, the secondary servers will be replaced by the one on the original system.</li> <li>3. Delete the secondary servers.</li> <li>4. Re-add the secondary servers. sipxcs-setup to reset the servers and re-assign them with the new IDs provided.</li> <li>5. After the servers are configured, go to system -&gt; databases and add the secondaries to the system.</li> </ol>	Known Issue	Backup