



链滴

TiDB 2.1 升级到 3.0

作者: [pangcm](#)

原文链接: <https://ld246.com/article/1565578227686>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

0. 升级说明

TiDB3.0 在前不久发布了，据说有很大的性能提升。接领导命令，让我来测试一下由 2.1 版本升到 3.0 版本的工作。下面的文档参考了官方的文档，也融合自己的内容。

1. 测试环境

本次的测试环境的如下所示，一共有 3 台 kv，1 个 pd 和一个 db，其中 pd 和 db 共用一台主。由于只有一个 pd 和一个 db，所以本次的测试必定会使得 tidb 停止服务的。如果要在生产环境中停机升级，那起码每个组件都要达到官方的集群要求，即：2DB+3PD+3KV。

组件	CPU	内存	IP 地址	主机名
TiDB	2 核	4 GB	192.168.113.20	tidb01
PD	2 核	4 GB	192.168.113.20	tidb01
TiKV	2 核	4 GB	192.168.113.21	tikv01
TiKV	2 核	4 GB	192.168.113.22	tikv02
TiKV	2 核	4 GB	192.168.113.23	tikv03

</table>

<h2 id="2--升级的兼容性说明">2. 升级的兼容性说明</h2>

不支持在升级后回退至 2.1.x 或更旧版本

从 2.0.6 之前的版本升级到 3.0 之前，需要确认集群中是否存在正在运行中的 DDL 操作，特别耗时的 `Add Index` 操作，等 DDL 操作完成后再执行升级操作

2.1 及之后版本启用了并行 DDL，早于 2.0.1 版本的集群，无法滚动升级到 3.0，可以选择下面种方案：

停机升级，直接从早于 2.0.1 的 TiDB 版本升级到 3.0

先滚动升级到 2.0.1 或者之后的 2.0.x 版本，再滚动升级到 3.0 版本

<blockquote>

<p>注意：</p>

<p>在升级的过程中不要执行 DDL 请求，否则可能会出现行为未定义的问题。</p>

</blockquote>

<p>看到这里，其实还是建议停机升级。</p>

<h2 id="3--在中控机器上下载新的-TiDB-Ansible">3. 在中控机器上下载新的 TiDB-Ansible</h2>

<p>和安装 TiDB 一样，不同的是你的 /home/tidb 目录下已经有了旧的 tidb-ansible 目录了，先备

</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">su - tidb</span></span></span></span><span class="highlight-line"><span class="highlight-cl">mv tidb-ansible/ tidb-ansible_2.1</span></span></code></pre>
```

</code></pre>

<p>下载 tidb-ansible 3.0，github 下载速度有点慢，耐心等待</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">git clone -b v3.0 https://github.com/pingcap/tidb-ansible.git</span></span></code></pre>
```

<h2 id="4--编辑-inventory-ini-文件和配置文件">4. 编辑 inventory.ini 文件和配置文件</h2>

<p>编辑 inventory.ini 文件

请保持该文件的主机信息和旧文件一样，直接参考旧文件即可</p>

<p>修改 TiDB 为开发模式

修改 /home/tidb/tidb-ansible/group_vars/all.yml 的参数，使得我们使用的是开发模式，降低硬件性能的检测标准。生产环境不要改。</p>


```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">dev_mode: True</span></span></code></pre>
```

</code></pre>

<h2 id="5--下载-TiDB-3-0-binary-到中控机">5. 下载 TiDB 3.0 binary 到中控机</h2>

<p>确认 `tidb-ansible/inventory.ini` 文件中 `tidb_version = v3.0.0`</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">ansible-playbook local_prepare.yml</span></span></code></pre>
```

</code></pre>

<h2 id="6--滚动升级-TiDB-集群组件">6. 滚动升级 TiDB 集群组件</h2>

<blockquote>

<p>注意:

为优化 TiDB 集群组件的运维管理, TiDB 3.0 版本对 <code>systemd</code> 模式下的 <code>P service</code> 名称进行了调整。与之前版本相比, 滚动升级 TiDB 3.0 版本集群组件的操作略有同, 注意升级前后 <code>process_supervision</code> 参数配置须保持一致。</p>

</blockquote>

<p>如果 <code>process_supervision</code> 变量使用默认的 <code>systemd</code> 参数则通过 <code>excessive_rolling_update.yml</code> 滚动升级 TiDB 集群。</p>

<pre><code class="highlight-chroma">ansible-playbook excessive_rolling_update.yml

</code></pre>

<p>如果 <code>process_supervision</code> 变量使用 <code>supervise</code> 参数, 则过 <code>rolling_update.yml</code> 滚动升级 TiDB 集群。</p>

<pre><code class="highlight-chroma">ansible-playbook rolling_update.yml

</code></pre>

<p>这里我们使用的是 systemd, 所以使用第一种方式</p>

<p>#7. 滚动升级 TiDB 监控组件</p>

<p>最后别忘了升级我们的监控组件</p>

<pre><code class="highlight-chroma">ansible-playbook rolling_update_monitor.yml

</code></pre>