

# batchRun 用户手册

**Product Name :** batchRun

**Product Version :** V2.0

**Release Date :** 2024.11.20

**Contact :** [@李艳青](#) (liyanqing.1987@bytedance.com)

[@张静文](#) (zhangjingwen.silvia@bytedance.com)

# 目录

一、简介 .....	4
1.1 功能简介 .....	4
1.2 快速了解 .....	4
1.2.1 <i>command line</i> 模式 .....	5
1.2.2 <i>GUI</i> 模式 .....	5
二、环境依赖 .....	8
2.1 操作系统依赖 .....	8
2.2 PYTHON 版本依赖 .....	8
2.3 系统组件依赖 .....	8
三、工具安装及配置 .....	9
3.1 工具下载 .....	9
3.2 工具安装 .....	10
3.3 工具配置 .....	11
3.3.1 <i>config/config.py</i> .....	12
3.3.2 <i>config/host.list</i> .....	12
3.3.3 <i>config/password.encrypted</i> .....	15
3.4 服务器信息采样 .....	15
四、工具使用 .....	17
4.1 工具载入 .....	17
4.2 COMMAND LINE 功能介绍 .....	17
4.2.1 帮助信息 .....	17
4.2.2 打印 <i>batchRun</i> 版本信息。 .....	19
4.2.3 列出指定机器 .....	20
4.2.4 指定用户名和密码登录远程机器 .....	20
4.2.5 采用内置加密密码登录远程机器 .....	21
4.2.6 指定机器执行命令 .....	22
4.2.7 指定机器组执行命令 .....	23
4.2.8 执行命令 ( <i>command</i> ) .....	24
4.2.9 串行执行和并行执行 .....	24
4.2.10 超时时间 ( <i>timeout</i> ) .....	26
4.2.11 输出信息层级 .....	26
4.2.12 输出文件 .....	28
4.3 GUI 功能介绍 .....	29
4.3.1 GROUP 页 .....	30
4.3.2 HOST 页 .....	30
4.3.3 RUN 页 .....	32
4.3.4 LOG 页 .....	34
五、辅助工具 .....	36
5.1 TOOLS/SWITCH_etc_hosts .....	36

<b>5.2 TOOLS/SAVE_PASSWORD .....</b>	40
<b>5.3 TOOLS/SAMPLE_HOST_INFO .....</b>	40
<b>5.4 TOOLS/PATCH .....</b>	43
<b>六、技术支持 .....</b>	<b>44</b>
<b>附录 .....</b>	<b>45</b>
<b>附 1. 变更历史.....</b>	<b>45</b>

# 一、简介

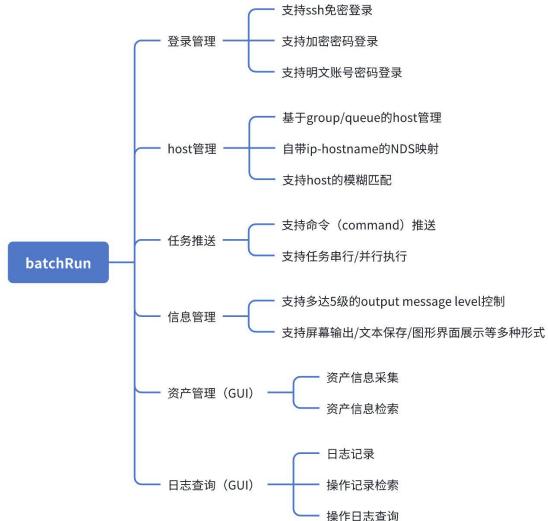
## 1.1 功能简介

batchRun 是 ByteDance 开源的一款 HPC 批量部署、资产管理和信息收集工具，采用所见即所得的命令执行方式，普遍适用于 RHEL、Centos、Rocky、Ubuntu 等多种 Linux 操作系统环境。

batchRun 基于 python 语言开发，采用 agentless 架构，通过 ssh 协议与被管理主机通讯，配置和使用便捷。



batchRun 的主要功能如下：



## 1.2 快速了解

batchRun 支持 command line 模式和 GUI 模式，其中 command line 模式主要用于批量部署和信息获取，GUI 模式则额外增加了资产管理和日志查询等功能。

## 1.2.1 command line 模式

batchRun 支持命令行模式，通过参数指定的方式，在 HPC 集群中做任务的批量部署。

跟 ansible 不同的是，batchRun 采用所见即所得的运行方式，不需要借助 playbook 等配置文件，而是直接推送需要运行的 command 到远程服务器上执行，并获取返回结果。

```
[root@ic-admin2 batchRun]# bin/batch_run --groups CMP_IC --command
hostname --output_message_level 2
>>> 10.232.132.142      n232-132-142
>>> 10.232.132.143      n232-132-143

Total 2 hosts.
```

## 1.2.2 GUI 模式

batchRun 的 GUI 模式包含 GROUP/HOST/RUN/LOG 四个页面。

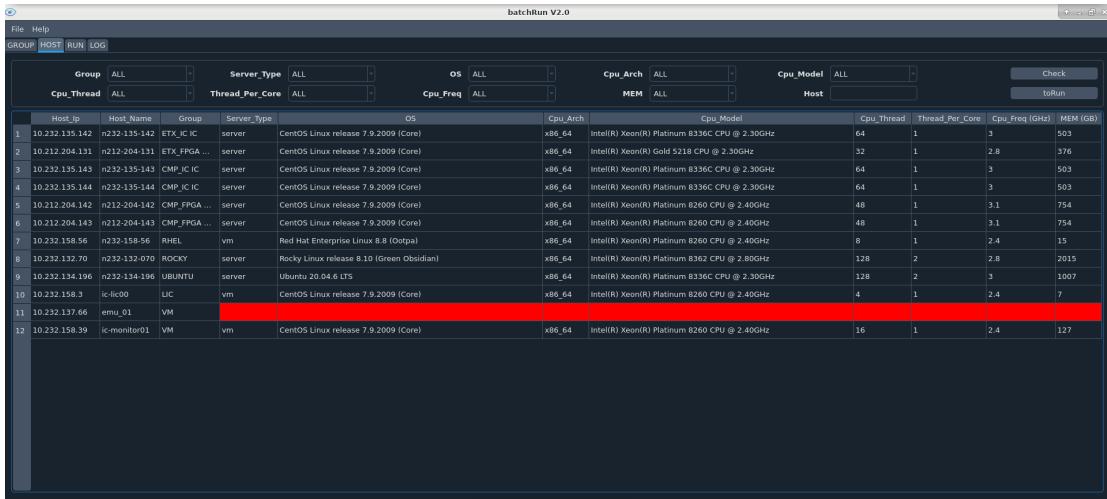
### GROUP 页：

用来图形化展示 host.list 的配置内容，可以看出 group <-> host <-> queue 的对应关系，可以通过 group/queue/host 三个维度来筛选 host。

Group	Host_ip	Host_Name	Ssh_Port	Queue	Host
ETX_IC	10.232.135.142	n232-135-142		orca_ana	
ETX_IC	10.212.204.131	n212-204-131		fpga_dpu	
CMP_IC	10.232.135.143	n232-135-143		orca_ana	
CMP_IC	10.212.204.144	n212-204-144		orca_ana	
ETX_FPGA	10.232.135.142	n232-135-142		orca_ana	
ETX_FPGA	10.212.204.142	n212-204-142		orca_ana	
RHEU	10.232.132.143	n232-132-143		orca_ana	
ROCKY	10.232.158.56	n232-158-56		orca_ana	
UBUNTU	10.232.132.70	n232-132-70		orca_ana	
UBUNTU	10.232.134.196	n232-134-196		orca_ana	
VM	10.232.138.3	n232-138-3		orca_ana	
VM	10.232.137.66	n232-137-66		orca_ana	
VM	10.232.138.39	n232-138-39	8888	orca_ana	
ETX_IC	10.232.135.142	n232-135-142		orca_ana	
CMP_IC	10.232.135.143	n232-135-143		orca_ana	
ETX_FPGA	10.212.204.131	n212-204-131		fpga_dpu	
FPGA	10.212.204.142	n212-204-142		fpga_dpu	
CMP_FPGA	10.212.204.143	n212-204-143		fpga_dpu	

## HOST 页：

用来展示所有服务器的 server\_type/os/cpu/mem 等信息，可以用作资产管理的用途。

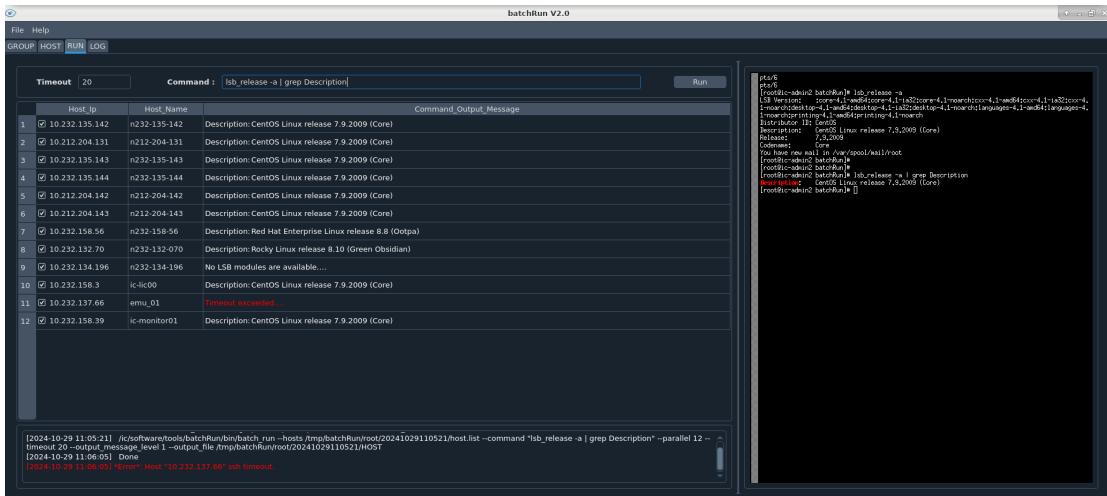


The screenshot shows the 'HOST' tab selected in the top navigation bar. The main area is a table with the following columns: Host\_Id, Host\_Name, Group, Server\_Type, OS, CPU\_Arch, CPU\_Model, Cpu\_Thread, Thread\_Per\_Core, Cpu\_Freq(GHz), and MEM(GB). The table lists 12 servers, each with a checkbox in the first column. The last two rows (11 and 12) have red backgrounds.

	Host_Id	Host_Name	Group	Server_Type	OS	CPU_Arch	CPU_Model	Cpu_Thread	Thread_Per_Core	Cpu_Freq(GHz)	MEM(GB)
1	10.232.135.142	n232-135-142	ETX_IC_IC	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	64	1	3	503
2	10.212.204.131	n212-204-131	ETX_FPGA...	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz	32	1	2.8	376
3	10.232.135.143	n232-135-143	CMP_IC_IC	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	64	1	3	503
4	10.232.135.144	n232-135-144	CMP_IC_IC	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	64	1	3	503
5	10.212.204.142	n212-204-142	CMP_FPGA...	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	48	1	3.1	754
6	10.212.204.143	n212-204-143	CMP_FPGA...	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	48	1	3.1	754
7	10.232.158.56	n232-158-56	RHEL	vm	Red Hat Enterprise Linux 8.8 (Ootpa)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	8	1	2.4	15
8	10.232.132.70	n232-132-070	ROCKY	server	Rocky Linux release 8.10 (Green Obsidian)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2	2.8	2015
9	10.232.134.196	n232-134-196	UBUNTU	server	Ubuntu 20.04.4 LTS	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	128	2	3	1007
10	10.232.158.3	ic-lc00	LIC	vm	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	4	1	2.4	7
11	10.232.137.66	emu_01	VM								
12	10.232.158.39	ic-monitor01	VM	vm	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	16	1	2.4	127

## RUN 页：

在选中的服务器上批量推送任务，同时可以获取命令的输出信息并展示。



The screenshot shows the 'RUN' tab selected in the top navigation bar. The left panel displays a table of hosts with checkboxes. The right panel shows the output of the command 'lsb\_release -a | grep Description' for each host. Host 11 shows a 'Timeout exceeded' message. The bottom panel shows the command history and logs.

	Host_Id	Host_Name	Command	Output
1	10.232.135.142	n232-135-142	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
2	10.212.204.131	n212-204-131	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
3	10.232.135.143	n232-135-143	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
4	10.232.135.144	n232-135-144	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
5	10.212.204.142	n212-204-142	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
6	10.212.204.143	n212-204-143	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
7	10.232.158.56	n232-158-56	lsb_release -a   grep Description	Description: Red Hat Enterprise Linux release 8.8 (Ootpa)
8	10.232.132.70	n232-132-070	lsb_release -a   grep Description	Description: Rocky Linux release 8.10 (Green Obsidian)
9	10.232.134.196	n232-134-196	lsb_release -a   grep Description	No LSB modules are available...
10	10.232.158.3	ic-lc00	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)
11	10.232.137.66	emu_01	lsb_release -a   grep Description	Timeout exceeded
12	10.232.158.39	ic-monitor01	lsb_release -a   grep Description	Description: CentOS Linux release 7.9.2009 (Core)

```

[2024-10-29 11:05:21] /ic/software/tools/batchRun/bin/batch_run --hosts /tmp/batchRun/root/20241029110521/host.list --command "lsb_release -a | grep Description" --parallel 12 --timeout 20 --output_message_level 1 --output_file /tmp/batchRun/root/20241029110521/HOST
[2024-10-29 11:06:05] Host "10.232.137.66" ssh timeout.

```

## LOG 页：

检索 batch\_run 的执行记录，查询每次执行的具体日志信息。

The screenshot shows the 'batchRun V2.0' application interface. At the top, there are tabs for 'File', 'Help', 'HOST', 'RUN', and 'LOG'. The 'LOG' tab is currently selected. Below the tabs, there are search filters: 'User ALL', 'Begin Date 2024-10-22', 'End Date 2024-10-29', and a 'Search' input field. A 'Log' button is also present. The main area displays a table of log entries with columns: Time, User, Login\_User, Command, and Log. The log table contains 29 entries from October 29, 2024, at 09:59:47 to 11:12:07. The 'Command' column shows various shell commands related to 'batch\_run.py' and system monitoring. The 'Log' column displays the raw log output for each entry. On the right side of the application window, there is a large text area showing the full log output for all entries.

User	ALL	Begin Date	2024-10-22	End Date	2024-10-29	Info	Search	Log
9	2024-10-29 09:59:47	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.158.56 --command /c/software/cad_tool...				>>> 10.232.135.142 11:12:07 up 66 days, 20 min, 1 user, load average: 0.10, 0.09, 0.12
10	2024-10-29 10:00:15	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.158.56 --command ls /c/software/cad_tool...				>>> 10.232.104.131 11:12:08 up 66 days, 53 min, 1 user, load average: 2.06, 2.12, 1.77
11	2024-10-29 10:03:15	root	root	/c/software/tools/batchRun/bin/batch_run.py -groups ETX_IC_CMP_IC --command hostname; touch /c/...				>>> 10.232.135.142 11:12:09 up 66 days, 53 min, 1 user, load average: 0.07, 0.04, 0.05
12	2024-10-29 10:03:45	root	root	/c/software/tools/batchRun/bin/batch_run.py -groups ETX_IC_CMP_IC --command hostname; touch /c/...				>>> 10.232.135.142 11:12:09 up 66 days, 53 min, 1 user, load average: 0.07, 0.04, 0.05
13	2024-10-29 10:07:58	root	root	/c/software/tools/batchRun/bin/batch_run.py -groups ETX_IC_CMP_IC --command hostname; touch /c/...				>>> 10.232.120.143 11:12:10 up 66 days, 49 min, 1 user, load average: 1.67, 1.47, 1.47
14	2024-10-29 10:08:49	root	root	/c/software/tools/batchRun/bin/batch_run.py -groups ETX_IC_CMP_IC --command hostname; touch /c/...				>>> 10.232.204.143 11:12:10 up 66 days, 49 min, 1 user, load average: 1.08, 1.03, 0.79
15	2024-10-29 10:09:25	root	root	/c/software/tools/batchRun/bin/batch_run.py -groups ETX_IC_CMP_IC --command hostname; touch /c/...				>>> 10.232.158.56 11:12:11 up 472 days, 19:42, 1 user, load average: 0.29, 0.25, 0.17
16	2024-10-29 10:10:27	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command sleep 15 --timeout 10				>>> 10.232.134.196 11:12:12 up 174 days, 17:33, 4 users, load average: 0.00, 0.02, 0.05
17	2024-10-29 10:11:35	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command sleep 15; hostname ...				>>> 10.232.158.3 11:12:13 up 384 days, 23:37, 1 user, load average: 0.07, 0.05, 0.06
18	2024-10-29 10:13:26	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				>>> 10.232.158.39 11:12:34 up 310 days, 31 min, 2 users, load average: 0.04, 0.05, 0.05
19	2024-10-29 10:13:40	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				
20	2024-10-29 10:13:55	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				
21	2024-10-29 10:14:09	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				
22	2024-10-29 10:14:23	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				
23	2024-10-29 10:14:58	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				
24	2024-10-29 10:15:36	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts 10.232.135.142 --command free -g ....				
25	2024-10-29 11:05:22	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts /tmp/batchRun/root/20241029110521/hostList ....				
26	2024-10-29 11:09:08	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts /tmp/batchRun/root/20241029110907/hostList ....				
27	2024-10-29 11:10:02	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts /tmp/batchRun/root/20241029111001/hostList ....				
28	2024-10-29 11:11:04	root	root	/c/software/tools/batchRun/bin/batch_run.py -hosts /tmp/batchRun/root/20241029111010/hostList ....				
29	2024-10-29 11:12:07	root	root	/c/software/tools/batchRun/bin/batch_run.py -groups ALL --command uptime --output_message_level 2				

## 二、环境依赖

### 2.1 操作系统依赖

batchRun 的开发和(master)测试操作系统为 **CentOS Linux release 7.9.2009 (Core)**, 这也是 IC 设计常用的操作系统版本之一。

同时支持 RHEL/Centos/Rocky/Ubuntu 等多种 Linux 操作系统, 主要风险在于系统库版本差异可能会影响部分组件的运行。

### 2.2 python 版本依赖

batchRun 基于 python 开发, 其开发和测试的 python 版本为 **python3.12.7**。

不同版本的 python 可能会有 python 库版本问题, 按照系统要求安装对应版本的 python 库即可解决, 但过旧的 python 版本可能因为差异过大而并不支持。

### 2.3 系统组件依赖

batchRun 的部分图形界面功能依赖 xterm, xterm 如未安装则可能会造成 GUI 界面中部分组件及功能不可用。

# 三、工具安装及配置

## 3.1 工具下载

batchRun 的 github 路径位于 <https://github.com/bytedance/batchRun>

The screenshot shows the GitHub repository page for 'batchRun' owned by 'bytedance'. The repository has 18 commits, 5 forks, and 13 stars. The 'About' section describes it as an ansible-similar IT automation system for the IC industry. The 'Code' tab is selected, showing a list of files and their last commit details. The 'Languages' section indicates Python 99.9% and Shell 0.1%. The bottom navigation bar includes links for README and GPL-2.0 license.

可以采用“git clone <https://github.com/bytedance/batchRun.git>”的方式拉取源代码。

```
[bytedance@LQ2KW66YXF ~/Downloads]#git clone  
https://github.com/bytedance/batchRun.git  
Cloning into 'batchRun'...  
remote: Enumerating objects: 186, done.  
remote: Counting objects: 100% (186/186), done.  
remote: Compressing objects: 100% (119/119), done.  
remote: Total 186 (delta 65), reused 172 (delta 51), pack-reused 0  
(from 0)  
Receiving objects: 100% (186/186), 5.23 MiB | 1.16 MiB/s, done.  
Resolving deltas: 100% (65/65), done.
```

也可以在 batchRun 的 github 页面上，Code -> Download ZIP 的方式拉取代码包。

The screenshot shows the GitHub repository page for 'batchRun'. The 'Code' tab is active, displaying the master branch. In the 'Clone' section, there are three options: HTTPS (selected), SSH, and GitHub CLI. Below these is a field containing the URL 'git@github.com:bytedance/batchRun.git'. To the right of the URL is a 'Download ZIP' button, which is highlighted with a red box. The rest of the page includes a file list, repository statistics (13 stars, 4 watching, 6 forks), and sections for Releases, Packages, and Languages.

## 3.2 工具安装

工具安装之前，首先参照第二章“环境依赖”满足 batchRun 的环境依赖关系。

batchRun 是 IT 管理员的运维自动化工具，所以建议用 root 账号将安装包拷贝到安装目录（NAS 最佳），并确保当前服务器有权限 ssh 其它被管理机器。

安装包下的文件和目录如下。

```
[root@ic-admin2 batchRun]# ls
bin  common  config  data  db  docs  install.py  LICENSE
Notice.txt  README.md  requirements.txt  scripts  tools
```

确认 python 版本正确。

```
[root@ic-admin2 batchRun]# which python3
/ic/software/tools/python3/3.12.7/bin/python3
```

使用跟 python3.12.7 对应的 pip3, 基于安装包中的 requirements.txt 安装 python 依赖库。(可能需要 root 权限)

```
[root@ic-admin2 batchRun]# which pip3
/ic/software/tools/python3/3.12.7/bin/pip3
[root@ic-admin2 batchRun]#
[root@ic-admin2 batchRun]# pip3 install -r requirements.txt
Looking in indexes: http://bytedpypi/byted.org/simple
Requirement already satisfied: matplotlib==3.9.2 in
//ic/software/tools/python3/3.12.7/lib/python3.12/site-packages
(from -r requirements.txt (line 1)) (3.9.2)
...
...
```

在安装目录下，使用命令“python3 install.py”安装 batchRun。

```
[root@ic-admin2 batchRun]# python3 install.py
>>> Check python version.
    Required python version : (3, 12)
    Current  python version : (3, 12)

>>> Generate script "/ic/software/tools/batchRun/bin/batch_run".
>>> Generate script
"/ic/software/tools/batchRun/bin/batch_run_gui".
>>> Generate script
"/ic/software/tools/batchRun/tools/encrypt_python".
>>> Generate script "/ic/software/tools/batchRun/tools/patch".
>>> Generate script
"/ic/software/tools/batchRun/tools/sample_host_info".
>>> Generate script
"/ic/software/tools/batchRun/tools/save_password".
>>> Generate script
"/ic/software/tools/batchRun/tools/switch_etc_hosts".
>>> Generate config file
"/ic/software/tools/batchRun/config/config.py".

Done, Please enjoy it.
```

### 3.3 工具配置

主要的配置文件位于 安装目录下的 config 目录中。

### 3.3.1 config/config.py

安装目录下的 config/config.py 用于配置工具的一些基本设置和验证规则。

安装后默认配置如下，一般采用默认设置即可。

```
# Specify host list, default is "host.list" on current configure
# directory.
host_list = '/ic/software/tools/batchRun/config/host.list'

# Specify the database directory.
db_path = '/ic/software/tools/batchRun/db'

# Default ssh command.
default_ssh_command = "ssh -o StrictHostKeyChecking=no -t -q"

# Support host_ip fuzzy matching, could be "True" or "False".
fuzzy_match = True

# Define timeout for ssh command, unit is "second".
serial_timeout = 10
parallel_timeout = 20
```

**host\_list**：指定 host.list 文件位置，一般使用默认值即可。

**db\_path**：指定 batch\_run 的数据库路径。

**default\_ssh\_command**：指定默认 ssh 的命令，一般使用默认值即可。

**fuzzy\_match**：控制命令参数中 host 模糊匹配的开关，默认打开。

**serial\_timeout**：指定串行执行情况下 ssh 命令的超时时间，默认为 10 秒，如果经常执行 long-runtime 的任务，可以适当调大这个值。

**parallel\_timeout**：指定并行执行情况下 ssh 命令的超时时间，默认为 20 秒，如果经常执行 long-runtime 的任务，可以适当调大这个值。

### 3.3.2 config/host.list

host.list 用于定义所有的机器及其分组信息，下面是一个说明。

```
#### Format ####
# [group]
# host_ip1
```

```
# host_ip2 ssh_port=<ssh_port2>
# host_ip3 ssh_host=<host_name3>
# host_ip4 ssh_host=<host_name4> ssh_port=<ssh_port4>
# sub_group5
# ~host_ip6
# ~host_name7
# ~sub_group8
#####
```

host.list 配置的基本格式如下：

```
[group] #组名
host #机器
sub_group #子组
excluded_host #要排除的机器
excluded_sub_group #要排除的子组
```

### 3.3.2.1 编写规则

- “group”编写规则

group 为一组机器的组名，一般用大写，不能包含空格，在 host.list 中用“[]”括起来，比如**[LOGIN]**或者**[COMPUTING]**。

- “host”编写规则

host 为机器，包含如下 4 种写法。

```
# host_ip1
# host_ip2 ssh_port=<ssh_port2>
# host_ip3 ssh_host=<host_name3>
# host_ip4 ssh_host=<host_name4> ssh_port=<ssh_port4>
```

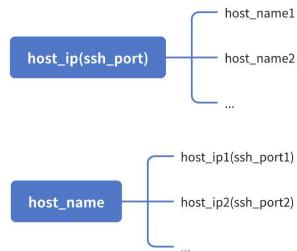
host\_ip 是必选项，是 host 的确定标识。

host\_name 是可选项，主要用于标识 DNS 映射关系。

ssh\_port 是可选项，如果 host\_ip 的 ssh 端口不是 22，则需要在此处指定。

此处 host\_name 需要注意，不能包含空格。

为保持和/etc/hosts 的兼容，host\_ip 和 host\_name 是多对多的对应关系，即一个 host\_ip 可能对应多个多个 host\_name（但是对应唯一的 ssh\_port），一个 host\_name 也可能对应多个 host\_ip，对应关系如下。



- "sub\_group"编写规则

sub\_group 为多机器的组名，不能包含空格，必须在 host.list 文件中被定义过。

- “excluded\_host”编写规则

可以按照 host\_ip 或者 host\_name 来排除 host，在前面加上波浪线“~”即可，样式如下。

```
[group]
~host_ip6
~host_name7
```

~host\_ip2: [group]中把所有的项按照 host\_ip 的维度展开，如果存在 host\_ip2，则会被排除。

~host\_name3: [group]中把所有的项按照 host\_ip 的维度展开，如果某个 host\_ip 的机器名为 host\_name3，则会被排除。

- “excluded\_sub\_group”编写规则

可以按照 sub\_group 来排除 host，在前面加上波浪线“~”即可，样式如下。

```
[group]
~sub_group8
```

`~sub_group4`: [group]中把所有的项按照 host\_ip 的维度展开, `sub_group4` 中的 host\_ip 会被排除掉。

### 3.3.2.2 列表生成

`config/host.list` 可以手工配置, 但是在服务器数量较多时较为繁琐, 所以更推荐基于格式化的`/etc/hosts`文件直接生成, 转换方法参照#5.1 tools/switch\_etc\_hosts 章节。

```
[root@ic-admin2 batchRun]# tools/switch_etc_hosts -i /etc/hosts -o config/host.list

Output File : config/host.list
[root@ic-admin2 batchRun]# cat config/host.list

[ETX_IC]
10.232.134.66 ssh_host=n232-134-066
...
```

### 3.3.3 config/password.encrypted

这个文件由自带工具 `tools/save_password` 生成, 用户可以为自己配置加密密码, 详情可以参照#5.2 章节。

建议通过 `save_password` 将 `batch_run` 常用用户的密码都存放进去, 当前用户已经配置了 `ssh` 免密登录则无需采用这种加密密码登录的方式。

```
[root@ic-admin2 batchRun]# tools/save_password -p ***
[root@ic-admin2 batchRun]# cat db/password/root
root default 1724049499 ed5fa5a1d09c8f4bf3e99c55708be23a
00b2a6273de7bc43a83f9bc352822827
```

## 3.4 服务器信息采样

`batchRun` 的资产管理功能, 依赖于 `sample_host_info` 的服务器信息采样, 采样方法可以参照#5.3 tools/sample\_host\_info 章节。

```
[root@ic-admin2 batchRun]# tools/sample_host_info --groups ALL -
```

```
-parallel 0 --timeout 100
>>> Clean up /ic/software/tools/batchRun/db/host_info ...
    rm -rf *.info host_list.json host_info.json
>>> Sampling host list information ...
    /ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 100 --list --output_file
/ic/software/tools/batchRun/db/host_info/host_list.json
* Host(s) info has been saved into
"/ic/software/tools/batchRun/db/host_info/host_list.json".
>>> Sampling host os/cpu/mem information ...
    /ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 100 --command "lsb_release -a; hostnamectl;
lscpu; free -g" --output_message_level 1 --output_file
/ic/software/tools/batchRun/db/host_info/HOST.info

Total 12 hosts. (Runtime: 3 minutes 21 seconds)
>>> Collecting host information ...
    Host info has been saved to file
"/ic/software/tools/batchRun/db/host_info/host_info.json".
```

如果资产信息经常处于变动之中，建议用 crontab 设置一个定时采样，以确保资产信息可以被及时更新。

```
# For batchRun, sample host information.
10 0 * * * /ic/software/tools/batchRun/tools/sample_host_info --
groups ALL --parallel 0 --timeout 100
```

## 四、工具使用

### 4.1 工具载入

batchRun 的主程序是 batch\_run, 位于 batchRun 安装目录下的 bin/batch\_run, 安装后可以直接引用。

如果配置了 modules, 则可以通过 module load 的方式引用。

```
[root@ic-admin2 batchRun]# module load cad
[root@ic-admin2 batchRun]# which batch_run
/ic/software/tools/batchRun/bin/batch_run
```

### 4.2 command line 功能介绍

#### 4.2.1 帮助信息

batch\_run 的帮助信息如下。

```
[root@ic-admin2 batchRun]# bin/batch_run -h
usage: batch_run.py [-h] [-H HOSTS [HOSTS ...]] [-G GROUPS
[GROUPS ...]] [-L] [-u USER] [-p PASSWORD] [-c COMMAND
[COMMAND ...]] [-P PARALLEL] [-t TIMEOUT] [-l {0,1,2,3,4}]
[-o OUTPUT_FILE] [-g] [-v]

options:
-h, --help            show this help message and exit
-H HOSTS [HOSTS ...], --hosts HOSTS [HOSTS ...]
Specify host(s) with below format:
<host_ip_file>
<host_ip>
<host_ip>:<ssh_port>
<host_name>
<host_name>:<ssh_port>
~<host_ip>
~<host_name>

"all | ALL" means all hosts on
/ic/software/tools/batchRun/config/host.list.
"~<host>" means exclud specified host.
-G GROUPS [GROUPS ...], --groups GROUPS [GROUPS ...]
```

```
Specify host group(s) with below format:  
<group_file>  
<group>  
~<group>  
  
"all | ALL" means all groups on  
/ic/software/tools/batchRun/config/host.list.  
"~<GROUP>" means exclud specified group.  
-L, --list List specified host(s)/group(s).  
-u USER, --user USER Specify the user identity for SSH login to  
specified host.  
-p PASSWORD, --password PASSWORD  
Specify the user password for SSH login to  
specified host.  
-c COMMAND [COMMAND ...], --command COMMAND [COMMAND ...]  
Specify the command to run on specified  
remote host(s).  
-P PARALLEL, --parallel PARALLEL  
Specify the parallelism of command  
execution with a number, default is "1" (serial mode).  
"0" : Parallel mode, run all tasks in  
parallel;  
"1" : Serial mode;  
"n" : Parallel mode, run n tasks in  
parallel.  
-t TIMEOUT, --timeout TIMEOUT  
Specify the timeout for SSH, which  
defaults to 10 seconds in serial and 20 seconds in parallel.  
-l {0,1,2,3,4}, --output_message_level {0,1,2,3,4}  
Specify output message level, which  
defaults to "3" in serial and "0" in parallel.  
"0" : print host info;  
"1" : print command output message;  
"2" : print host info and the first line  
of the command output message;  
"3" : print host info and complete command  
output message;  
"4" : print verbose information with ssh  
command.  
-o OUTPUT_FILE, --output_file OUTPUT_FILE  
Export output message of command to  
specified file instead of on the screen.  
-g, --gui Open batchRun with GUI format.  
-v, --version Show batchRun version information.
```

**--help** : 打印帮助信息。

**--hosts** : 指定机器列表，可以是机器 ip，可以是机器 hostname，也可以是一个文件中包含了 host\_ip/host\_name 信息。如果是“ALL”，则意味着配置文件 host.list 中所有的机器。

**--groups** : 根据服务器组获取服务器列表，也可以是一个文件中包含了服务器组信息。如果是“ALL”，则意味着配置文件 host.list 中所有的机器。

**--list**: 列出指定的 hosts 或者 groups。

**--user** : 指定 ssh 登录远程机器时的用户，默认是当前用户。

**--password** : 指定 ssh 登录远程机器时的用户密码。如果配置了 ssh 免密登录，或者加密密码已经被保存到了 password.encrypted 中，此处无需指定。

**--command** : 到远程机器上要执行的命令。

**--parallel** : 指定并行运行模式的并行数量，默认为串行运行模式。

**--timeout** : 指定 ssh 的超时时间，串行模式默认为 10 秒，并行模式默认为 20 秒。

**--output\_message\_level** : 指定执行命令时输出信息的详细程度，分为 0-4 共五个等级。

“0”：只打印机器名。

“1”：只打印（执行命令的）输出信息。

“2”：打印机器名和（执行命令的）输出信息的第一行。

“3”：打印机器名和完成的（执行命令的）输出信息。

“4”：在#3 的基础上打印更详细的 debug 信息（ssh 命令）。

**--output\_file** : 指定输出文件，如指定，会将输出在屏幕上的（执行命令的）输出信息导入到指定的输出文件中。

**--gui** : 启动 GUI 界面。

**--version** : 打印 batchRun 版本信息。

## 4.2.2 打印 batchRun 版本信息。

batch\_run --version 可以打印 batchRun 的版本信息。

```
[root@ic-admin2 batchRun]# bin/batch_run --version
Version : V2.0
Release : 2024.10.28
```

### 4.2.3 列出指定机器

如果不清楚 batch\_run 预设的机器及机器组设置，可以使用“batch\_run --list --hosts ALL”查看。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --hosts ALL
GROUP : [ETX_IC]
        10.232.135.142    n232-135-142
GROUP : [ETX_FPGA]
        10.212.204.131    n212-204-131
GROUP : [CMP_IC]
        10.232.135.143    n232-135-143
        10.232.135.144    n232-135-144
GROUP : [CMP_FPGA]
        10.212.204.142    n212-204-142
        10.212.204.143    n212-204-143
GROUP : [RHEL]
        10.232.158.56      n232-158-56
GROUP : [ROCKY]
        10.232.132.70      n232-132-070
GROUP : [UBUNTU]
        10.232.134.196      n232-134-196
GROUP : [LIC]
        10.232.158.3       ic-lic00
GROUP : [VM]
        10.232.137.66      emu_01
        10.232.158.39      ic-monitor01  8888
```

如果已知机器组名，也可以使用“batch\_run --list --groups <GROUP>”查看指定机器组的具体信息。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --groups CMP_IC
GROUP : [CMP_IC]
        10.232.135.143    n232-135-143
        10.232.135.144    n232-135-144
```

### 4.2.4 指定用户名和密码登录远程机器

如果知道用户名和密码，可以采用指定用户名和密码的方式，以指定帐号的身份登录远程机器执行命令，操作方式为“batch\_run --user <USER> --password

<PASSWORD> ...”。

下面是一个示例，root 用指定账号密码登录机器并执行命令。

```
[root@ic-admin2 batchRun]# bin/batch_run --host 10.232.135.143 --  
user liyanqing.1987 --password *** --command whoami  
  
>>> 10.232.135.143  
liyanqing.1987  
  
Total 1 hosts. (Runtime: 3 seconds)
```

因为明文密码会导致敏感信息泄露，所以并不推荐这种方式，建议配置 ssh 免密登录，或者采用内置加密密码的方式登录远程机器。

#### 4.2.5 采用内置加密密码登录远程机器

使用内置工具 save\_password 为指定账号保存加密密码。

```
[liyanqing.1987@n232-135-013 batchRun]$ tools/save_password --  
password ***
```

其储存的密码为加密样式。

```
[liyanqing.1987@n232-135-013 batchRun]$ cat  
db/password/liyanqing.1987  
liyanqing.1987 default 1724392876  
a2c768667257c39f4bc2a1a4fa1124dc 8d6f948e98fa7ba36cfa07f5d72afafa
```

此时，batch\_run 用当前 Linux 登录账号登录远程机器时，可以直接使用这个加密密码，以防止明文导致密码泄露的问题。

```
[liyanqing.1987@n232-135-013 batchRun]$ cat  
db/password/liyanqing.1987  
liyanqing.1987 default 1724392876  
a2c768667257c39f4bc2a1a4fa1124dc 8d6f948e98fa7ba36cfa07f5d72afafa
```

请注意，上面例子中，一定要以密码登陆的方式成为 liyanqing.1987 用户，否则 batch\_run 会无法判断当前实际用户是否知悉 liyanqing.1987 账号密码（比如 root 账号 su 成 liyanqing.1987，并不需要知道 liyanqing.1987 的密码），从而导致拒绝执行命令。

此时即便是没有配置 ssh 免密，也不输入明文密码，仍然可以方便地 ssh 地通过 batchRun 登录到远程服务器上执行任务。

```
[liyanqing.1987@ic-admin2 batchRun]$ bin/batch_run --hosts  
10.232.135.143 --command whoami  
  
>>> 10.232.135.143  
liyanqing.1987  
  
Total 1 hosts. (Runtime: 1 seconds)
```

## 4.2.6 指定机器执行命令

batch\_run 支持用“batch\_run --hosts <HOST> <HOST> ...”的方式指定机器，HOST 既可以是 ip 地址，也可以是 hostname（hostname 一定要在/etc/hosts 或者 host.list 中配置过）。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.143  
n232-135-144 --command hostname  
  
>>> 10.232.135.143  
n232-135-143  
  
>>> n232-135-144 (10.232.135.144)  
n232-135-144  
  
Total 2 hosts. (Runtime: 1 seconds)
```

同时，batch\_run 的--hosts 参数也支持模糊匹配，会把所有匹配的项都列出来。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --hosts 142  
[FUZZY MATCH] 142 -> 10.232.135.142  
[FUZZY MATCH] 142 -> 10.212.204.142  
  
GROUP : [ETX_IC]
```

```
10.232.135.142    n232-135-142
GROUP : [CMP_FPGA]
10.212.204.142    n212-204-142
```

不但可以匹配 host ip, 也可以匹配 host name, 还支持^\$等正则匹配符号来界定位置。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --hosts ^n212-204
[FUZZY MATCH] ^n212-204 -> n212-204-131 -> 10.212.204.131
[FUZZY MATCH] ^n212-204 -> n212-204-142 -> 10.212.204.142
[FUZZY MATCH] ^n212-204 -> n212-204-143 -> 10.212.204.143

GROUP : [ETX_FPGA]
10.212.204.131    n212-204-131
GROUP : [CMP_FPGA]
10.212.204.142    n212-204-142
10.212.204.143    n212-204-143
```

#### 4.2.7 指定机器组执行命令

batch\_run 支持用“batch\_run --groups <GROUP> <GROUP> ...”的方式指定机器，GROUP 必须在 host.list 中有定义。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --groups CMP_IC
CMP_FPGA
GROUP : [CMP_IC]
10.232.135.143    n232-135-143
10.232.135.144    n232-135-144
GROUP : [CMP_FPGA]
10.212.204.142    n212-204-142
10.212.204.143    n212-204-143
```

也可以采用除外的方式排除掉指定的 group 或者 host。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --groups CMP_IC
CMP_FPGA --hosts ~10.232.135.143 ~n212-204-142
GROUP : [CMP_IC]
10.232.135.144    n232-135-144
GROUP : [CMP_FPGA]
```

10.212.204.143 n212-204-143

## 4.2.8 执行命令 (command)

batch\_run 可以在远程机器上执行命令，此处的命令既可以表示系统命令（比如 hostname），也可以表示脚本命令（比如~/run.sh），前提是远程机器上命令是可以直接访问和执行的。

```
[root@ic-admin2 batchRun]# bin/batch_run --groups CMP_IC --command
hostname --output_message_level 2
>>> 10.232.135.143      n232-135-143
>>> 10.232.135.144      n232-135-144

Total 2 hosts. (Runtime: 1 seconds)
```

batch\_run 还有一个很酷的功能，如果 command/script 在远程机器上不存在，第一次执行失败以后，它会尝试将 command/script 拷贝到远程机器本地再次执行。这个操作是在后台执行的，可以用--output\_message\_level 4 参数将它的操作过程放到前台展示出来。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.158.56 --
command /ic/software/cad_tools/bin/test_echo.sh --
output_message_level 4

>>> 10.232.158.56
    ssh -o StrictHostKeyChecking=no -t -q root@10.232.158.56
/ic/software/cad_tools/bin/test_echo.sh
===== output =====
Command missing, scp and rerun.
This is a test
=====

Total 1 hosts. (Runtime: 2 seconds)
```

## 4.2.9 串行执行和并行执行

参数--parallel 用来指定串并行度，默认为串行。

--parallel 0：全部任务并行

--parallel 1: 全部任务串行

--parallel n: 全部任务按照并行度 n 并行, 同一时刻只有 n 个任务在执行

下面是一个串行运行任务的例子。

```
[root@ic-admin2 batchRun]# bin/batch_run --groups ETX_IC CMP_IC --  
command 'hostname; touch  
/ic/software/tools/batchRun/db/test/`hostname`' --  
output_message_level 2 --parallel 1  
>>> 10.232.135.142      n232-135-142  
>>> 10.232.135.143      n232-135-143  
>>> 10.232.135.144      n232-135-144  
  
Total 3 hosts. (Runtime: 2 seconds)  
[root@ic-admin2 batchRun]#  
[root@ic-admin2 batchRun]# ls /ic/software/tools/batchRun/db/test/  
n232-135-142  n232-135-143  n232-135-144
```

我们看到, 脚本在指定机器上串行执行, 明确打印 output 信息, 并生成了对应文件。

下面我们删掉生成的文件, 添加--parallel 参数用并行方式再执行一遍。(此时 output\_message\_level 被强制为“0”)

```
[root@ic-admin2 batchRun]# rm -rf  
/ic/software/tools/batchRun/db/test/*  
[root@ic-admin2 batchRun]#  
[root@ic-admin2 batchRun]# bin/batch_run --groups ETX_IC CMP_IC --  
command 'hostname; touch  
/ic/software/tools/batchRun/db/test/`hostname`' --  
output_message_level 0 --parallel 0  
>>> 10.232.135.142  
>>> 10.232.135.143  
>>> 10.232.135.144  
  
Total 3 hosts. (Runtime: 0 seconds)  
[root@ic-admin2 batchRun]#  
[root@ic-admin2 batchRun]# ls /ic/software/tools/batchRun/db/test/  
n232-135-142  n232-135-143  n232-135-144
```

我们看到文件正常生成，说明脚本在每台机器上都正常执行了。

## 4.2.10 超时时间 (timeout)

batch\_run 底层基于 ssh 登陆远程服务器执行命令，串行运行时其默认的 timeout 为 10 秒，并行运行时其默认的 timeout 为 20 秒，如果任务的执行时间超过 timeout 时间，ssh 会报告超时并强制断开连接。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --command "sleep 15; hostname" --timeout 10

>>> 10.232.135.142
    Timeout exceeded.
    <pexpect.pty_spawn.spawn object at 0x7fef2eb12ba0>
...
    0: re.compile(b'assword:')
    1: EOF

Total 1 hosts. (Runtime: 20 seconds)
```

执行运行时间超过 timeout 的任务时，可以通过设置更长的 timeout 时间来规避这个问题。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --command "sleep 15; hostname" --timeout 20

>>> 10.232.135.142
n232-135-142

Total 1 hosts. (Runtime: 16 seconds)
```

## 4.2.11 输出信息层级

batch\_run 的 output message 共有 5 个层级可选，效果分别如下。

- “0”级输出，仅打印服务器名。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --command "free -g" --output_message_level 0
```

```
>>> 10.232.135.142
```

```
Total 1 hosts. (Runtime: 0 seconds)
```

- “1”级输出，仅打印执行命令的输出信息，主要是配置--output\_file参数使用。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --  
command "free -g" --output_message_level 1  
total used free shared buff/cache  
available  
Mem: 503 10 473 0  
19 491  
Swap: 127 0 127
```

```
Total 1 hosts. (Runtime: 0 seconds)
```

- “2”级输出，打印服务器名，并显示执行命令的输出信息的第一行。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --  
command "free -g" --output_message_level 2  
>>> 10.232.135.142 total used free shared  
buff/cache available
```

```
Total 1 hosts. (Runtime: 1 seconds)
```

- “3”级输出，默认级别，打印服务器名，并显示执行命令的输出信息的全部。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --  
command "free -g" --output_message_level 3  
>>> 10.232.135.142  
total used free shared buff/cache  
available  
Mem: 503 10 473 0  
19 491  
Swap: 127 0 127
```

```
Total 1 hosts. (Runtime: 0 seconds)
```

- “4”级输出，在#3 的基础上，会打印实际执行的 ssh 命令，以方便用户手工执行并复现结果。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --
command "free -g" --output_message_level 4

>>> 10.232.135.142
    ssh -o StrictHostKeyChecking=no -t -q root@10.232.135.142 free
\ -g
    === output ===
    total        used         free      shared   buff/cache
available
    Mem:          503           10         473           0
19          491
    Swap:          127           0         127
=====
Total 1 hosts. (Runtime: 0 seconds)
```

## 4.2.12 输出文件

batch\_run 的任务执行结果默认输出到屏幕上，但是也可以通过--output\_file 参数将其导入指定文件中。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --
command "free -g" --output_message_level 1 --output_file
10.232.135.142.mem

Total 1 hosts. (Runtime: 1 seconds)
[root@ic-admin2 batchRun]#
[root@ic-admin2 batchRun]# cat 10.232.135.142.mem
    total        used         free      shared   buff/cache
available
    Mem:          503           10         473           0
19          491
    Swap:          127           0         127
```

batch\_run 还支持自动将--output\_file 中的“HOST”字符自动转换为实际的 host 名，这样非常利于多服务器信息采样的信息保存。

```
[root@ic-admin2 batchRun]# bin/batch_run --hosts 10.232.135.142 --command "free -g" --output_message_level 1 --output_file db/host_info/HOST.mem

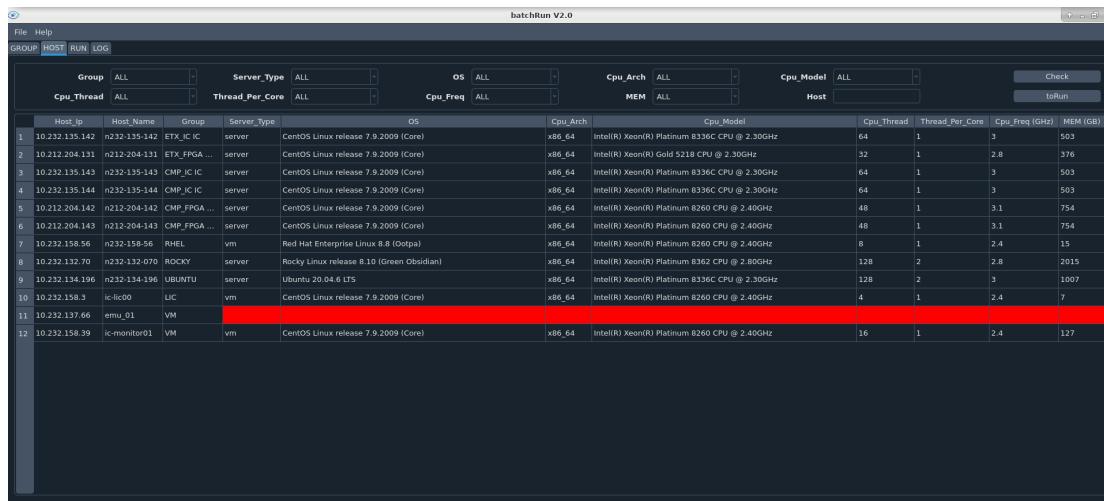
Total 1 hosts. (Runtime: 1 seconds)
[root@ic-admin2 batchRun]#
[root@ic-admin2 batchRun]# cat db/host_info/10.232.135.142.mem
total        used         free        shared   buff/cache
available
Mem:          503           10         473           0
19          491
Swap:         127           0         127


```

## 4.3 GUI 功能介绍

执行“batch\_run --gui”可以开启其 GUI 界面，GUI 界面查看信息和执行命令相对而言更直观。

```
[root@ic-admin2 batchRun]# bin/batch_run --gui
[2024-10-29 10:21:48] Getting LSF host-queue relationship, please
wait a moment ...
```



batchRun 的 GUI 界面共有包含 GROUP/HOST/RUN/LOG 四个界面，下面一一介绍。

### 4.3.1 GROUP 页

GROUP 页用来展示 group <-> host <-> queue 的关系。

左侧边栏是 host.list 配置的图形化重现，体现了 group 的定义关系，即每个 group 包含了哪些子 group 和子 host。

右侧的表格则按照 group 的维度来展示 group -> host 的归属关系，同时，如果 host 属于某个 LSF queue，则把相应的 queue 信息也展示出来。

Group	Host_ip	Host_Name	Ssh_Port	Queue	Host
10.232.135.142				orca_ana	
ETX_FPGA	10.232.204.131	n232-204-131		fpga_dpu	
ETX_IC	10.232.135.143	n232-135-143		orca_ana	
CMP_FPGA	10.232.135.144	n232-204-142		orca_ana	
CMP_IC	10.232.135.144	n232-204-143		orca_ana	
RHEU	10.232.204.142	n232-204-143		fpga_dpu	
SH6	10.232.135.56	n232-135-56		fpga_dpu	
ROCKY	10.232.132.70	n232-132-70		fpga_dpu	
	10.232.132.70	n232-132-70		rocky_dpu	
	10.232.134.196	n232-134-196		rocky_dpu	
LIC	10.232.135.83	n232-135-83		UBUNTU	
VME	10.232.137.66	n232-137-66		UBUNTU	
	10.232.137.66	n232-137-66		VM	
	10.232.135.89	n232-135-89		VM	
IC	10.232.135.89	n232-135-89		ic-monitor01	8888
ETX_IC	10.232.135.143	n232-135-143		IC	
CMP_IC	10.232.135.144	n232-135-144		IC	
FPGA	10.232.204.131	n232-204-131		FPGA	
	10.232.204.142	n232-204-142		FPGA	
	10.232.204.143	n232-204-143		FPGA	

右侧的表格可以按照 group/queue/host 来做筛选，点击“Check”按钮生效。

筛选过后的 host(s)，点击“toRun”按钮，可以将它们导入到 RUN 页面，执行下一步操作。

Host_ip	Host_Name	Type	Status
10.232.132.70	n232-132-70	Host	Up
10.232.134.196	n232-134-196	Host	Up
10.232.135.142	n232-135-142	Host	Up
10.232.135.143	n232-135-143	Host	Up
10.232.135.144	n232-135-144	Host	Up
10.232.204.131	n232-204-131	Host	Up
10.232.204.142	n232-204-142	Host	Up
10.232.204.143	n232-204-143	Host	Up
10.232.232.131	n232-232-131	Host	Up
10.232.232.132	n232-232-132	Host	Up
10.232.232.133	n232-232-133	Host	Up
10.232.232.134	n232-232-134	Host	Up
10.232.232.135	n232-232-135	Host	Up
10.232.232.136	n232-232-136	Host	Up
10.232.232.137	n232-232-137	Host	Up
10.232.232.138	n232-232-138	Host	Up
10.232.232.139	n232-232-139	Host	Up
10.232.232.140	n232-232-140	Host	Up
10.232.232.141	n232-232-141	Host	Up
10.232.232.142	n232-232-142	Host	Up
10.232.232.143	n232-232-143	Host	Up
10.232.232.144	n232-232-144	Host	Up
10.232.232.145	n232-232-145	Host	Up
10.232.232.146	n232-232-146	Host	Up
10.232.232.147	n232-232-147	Host	Up
10.232.232.148	n232-232-148	Host	Up
10.232.232.149	n232-232-149	Host	Up
10.232.232.150	n232-232-150	Host	Up
10.232.232.151	n232-232-151	Host	Up
10.232.232.152	n232-232-152	Host	Up
10.232.232.153	n232-232-153	Host	Up
10.232.232.154	n232-232-154	Host	Up
10.232.232.155	n232-232-155	Host	Up
10.232.232.156	n232-232-156	Host	Up
10.232.232.157	n232-232-157	Host	Up
10.232.232.158	n232-232-158	Host	Up
10.232.232.159	n232-232-159	Host	Up
10.232.232.160	n232-232-160	Host	Up
10.232.232.161	n232-232-161	Host	Up
10.232.232.162	n232-232-162	Host	Up
10.232.232.163	n232-232-163	Host	Up
10.232.232.164	n232-232-164	Host	Up
10.232.232.165	n232-232-165	Host	Up
10.232.232.166	n232-232-166	Host	Up
10.232.232.167	n232-232-167	Host	Up
10.232.232.168	n232-232-168	Host	Up
10.232.232.169	n232-232-169	Host	Up
10.232.232.170	n232-232-170	Host	Up
10.232.232.171	n232-232-171	Host	Up
10.232.232.172	n232-232-172	Host	Up
10.232.232.173	n232-232-173	Host	Up
10.232.232.174	n232-232-174	Host	Up
10.232.232.175	n232-232-175	Host	Up
10.232.232.176	n232-232-176	Host	Up
10.232.232.177	n232-232-177	Host	Up
10.232.232.178	n232-232-178	Host	Up
10.232.232.179	n232-232-179	Host	Up
10.232.232.180	n232-232-180	Host	Up
10.232.232.181	n232-232-181	Host	Up
10.232.232.182	n232-232-182	Host	Up
10.232.232.183	n232-232-183	Host	Up
10.232.232.184	n232-232-184	Host	Up
10.232.232.185	n232-232-185	Host	Up
10.232.232.186	n232-232-186	Host	Up
10.232.232.187	n232-232-187	Host	Up
10.232.232.188	n232-232-188	Host	Up
10.232.232.189	n232-232-189	Host	Up
10.232.232.190	n232-232-190	Host	Up
10.232.232.191	n232-232-191	Host	Up
10.232.232.192	n232-232-192	Host	Up
10.232.232.193	n232-232-193	Host	Up
10.232.232.194	n232-232-194	Host	Up
10.232.232.195	n232-232-195	Host	Up
10.232.232.196	n232-232-196	Host	Up
10.232.232.197	n232-232-197	Host	Up
10.232.232.198	n232-232-198	Host	Up
10.232.232.199	n232-232-199	Host	Up
10.232.232.200	n232-232-200	Host	Up
10.232.232.201	n232-232-201	Host	Up
10.232.232.202	n232-232-202	Host	Up
10.232.232.203	n232-232-203	Host	Up
10.232.232.204	n232-232-204	Host	Up
10.232.232.205	n232-232-205	Host	Up
10.232.232.206	n232-232-206	Host	Up
10.232.232.207	n232-232-207	Host	Up
10.232.232.208	n232-232-208	Host	Up
10.232.232.209	n232-232-209	Host	Up
10.232.232.210	n232-232-210	Host	Up
10.232.232.211	n232-232-211	Host	Up
10.232.232.212	n232-232-212	Host	Up
10.232.232.213	n232-232-213	Host	Up
10.232.232.214	n232-232-214	Host	Up
10.232.232.215	n232-232-215	Host	Up
10.232.232.216	n232-232-216	Host	Up
10.232.232.217	n232-232-217	Host	Up
10.232.232.218	n232-232-218	Host	Up
10.232.232.219	n232-232-219	Host	Up
10.232.232.220	n232-232-220	Host	Up
10.232.232.221	n232-232-221	Host	Up
10.232.232.222	n232-232-222	Host	Up
10.232.232.223	n232-232-223	Host	Up
10.232.232.224	n232-232-224	Host	Up
10.232.232.225	n232-232-225	Host	Up
10.232.232.226	n232-232-226	Host	Up
10.232.232.227	n232-232-227	Host	Up
10.232.232.228	n232-232-228	Host	Up
10.232.232.229	n232-232-229	Host	Up
10.232.232.230	n232-232-230	Host	Up
10.232.232.231	n232-232-231	Host	Up
10.232.232.232	n232-232-232	Host	Up
10.232.232.233	n232-232-233	Host	Up
10.232.232.234	n232-232-234	Host	Up
10.232.232.235	n232-232-235	Host	Up
10.232.232.236	n232-232-236	Host	Up
10.232.232.237	n232-232-237	Host	Up
10.232.232.238	n232-232-238	Host	Up
10.232.232.239	n232-232-239	Host	Up
10.232.232.240	n232-232-240	Host	Up
10.232.232.241	n232-232-241	Host	Up
10.232.232.242	n232-232-242	Host	Up
10.232.232.243	n232-232-243	Host	Up
10.232.232.244	n232-232-244	Host	Up
10.232.232.245	n232-232-245	Host	Up
10.232.232.246	n232-232-246	Host	Up
10.232.232.247	n232-232-247	Host	Up
10.232.232.248	n232-232-248	Host	Up
10.232.232.249	n232-232-249	Host	Up
10.232.232.250	n232-232-250	Host	Up
10.232.232.251	n232-232-251	Host	Up
10.232.232.252	n232-232-252	Host	Up
10.232.232.253	n232-232-253	Host	Up
10.232.232.254	n232-232-254	Host	Up
10.232.232.255	n232-232-255	Host	Up
10.232.232.256	n232-232-256	Host	Up
10.232.232.257	n232-232-257	Host	Up
10.232.232.258	n232-232-258	Host	Up
10.232.232.259	n232-232-259	Host	Up
10.232.232.260	n232-232-260	Host	Up
10.232.232.261	n232-232-261	Host	Up
10.232.232.262	n232-232-262	Host	Up
10.232.232.263	n232-232-263	Host	Up
10.232.232.264	n232-232-264	Host	Up
10.232.232.265	n232-232-265	Host	Up
10.232.232.266	n232-232-266	Host	Up
10.232.232.267	n232-232-267	Host	Up
10.232.232.268	n232-232-268	Host	Up
10.232.232.269	n232-232-269	Host	Up
10.232.232.270	n232-232-270	Host	Up
10.232.232.271	n232-232-271	Host	Up
10.232.232.272	n232-232-272	Host	Up
10.232.232.273	n232-232-273	Host	Up
10.232.232.274	n232-232-274	Host	Up
10.232.232.275	n232-232-275	Host	Up
10.232.232.276	n232-232-276	Host	Up
10.232.232.277	n232-232-277	Host	Up
10.232.232.278	n232-232-278	Host	Up
10.232.232.279	n232-232-279	Host	Up
10.232.232.280	n232-232-280	Host	Up
10.232.232.281	n232-232-281	Host	Up
10.232.232.282	n232-232-282	Host	Up
10.232.232.283	n232-232-283	Host	Up
10.232.232.284	n232-232-284	Host	Up
10.232.232.285	n232-232-285	Host	Up
10.232.232.286	n232-232-286	Host	Up
10.232.232.287	n232-232-287	Host	Up
10.232.232.288	n232-232-288	Host	Up
10.232.232.289	n232-232-289	Host	Up
10.232.232.290	n232-232-290	Host	Up
10.232.232.291	n232-232-291	Host	Up
10.232.232.292	n232-232-292	Host	Up
10.232.232.293	n232-232-293	Host	Up
10.232.232.294	n232-232-294	Host	Up
10.232.232.295	n232-232-295	Host	Up
10.232.232.296	n232-232-296	Host	Up
10.232.232.297	n232-232-297	Host	Up
10.232.232.298	n232-232-298	Host	Up
10.232.232.299	n232-232-299	Host	Up
10.232.232.300	n232-232-300	Host	Up
10.232.232.301	n232-232-301	Host	Up

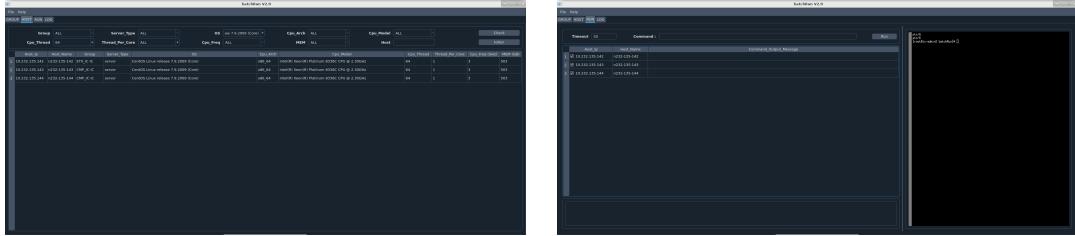
batchRun V2.0															
GROUP			HOST		RUN		LOG								
Group	ALL	-	Server_Type	ALL	-	OS	ALL	-	Cpu_Arch	ALL	-	Cpu_Model	ALL	- <th>Check</th>	Check
Cpu_Thread	ALL	-	Thread_Per_Core	ALL	-	Cpu_Freq	ALL	-	MEM	ALL	-	Host	ALL	-	toRun
1	10.232.135.142	n232-135-142	ETX_IC_IC	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	64	1	3	503				
2	10.232.204.131	n212-204-131	ETX_FPGA ...	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz	32	1	2.8	576				
3	10.232.135.143	n232-135-143	CMP_IC_IC	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	64	1	3	503				
4	10.232.135.144	n232-135-144	CMP_IC_IC	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	64	1	3	503				
5	10.232.204.142	n212-204-142	CMP_FPGA ...	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	48	1	3.1	754				
6	10.232.204.143	n212-204-143	CMP_FPGA ...	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	48	1	3.1	754				
7	10.232.158.56	n232-158-56	RHEL	vm	Red Hat Enterprise Linux 8.8 (Ootpa)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	8	1	2.4	15				
8	10.232.132.70	n232-132-070	ROCKY	server	Rocky Linux release 8.10 (Green Obsidian)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2	2.8	2015				
9	10.232.134.196	n232-134-196	UBUNTU	server	Ubuntu 20.04.6 LTS	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.30GHz	128	2	3	1007				
10	10.232.158.3	ic-1c00	LIC	vm	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	4	1	2.4	7				
11	10.232.137.66	emu_01	VM												
12	10.232.158.39	ic-monitor01	VM	vm	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	16	1	2.4	127				

下侧表格中展示的信息具体如下。

列标题	信息
Host_Ip	服务器 ip
Host_Name	服务器 hostname
Group	服务器在 host.list 中的归属组
Server_Type	服务器类型，一般分为 server 和 vm 两种
OS	操作系统
Cpu_Arch	cpu 架构
Cpu_Model	cpu 型号
Cpu_Thread	cpu 核数
Thread_Per_Core	cpu core 的线程数，判断超线程是否开启
Cpu_Freq (GHz)	cpu 频率，单位为 GHz
MEM (GB)	内存容量

我们看到有部分服务器的信息缺失，单元格标记为红色，说明服务器不能登陆，或者登录后部分命令无法执行，导致服务器的软硬件信息采集失败。

表格内容可以按照 group/server\_type/os 等多个维度来筛选，筛选过后的 host(s)，点击“toRun”按钮，可以将它们导入到 RUN 页面。

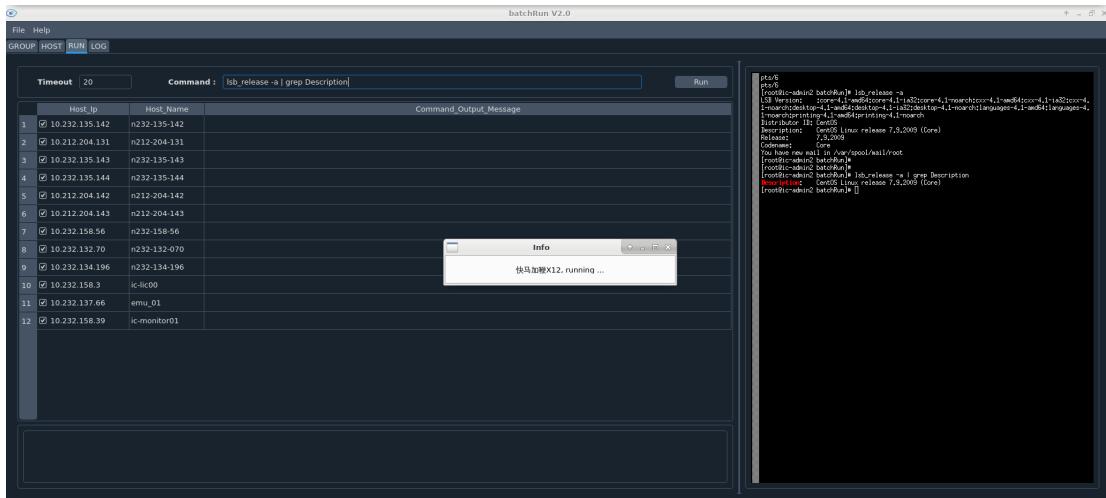


### 4.3.3 RUN 页

RUN 页用来在选中的服务器上批量运行同一命令，同时可以直观地获取命令执行后的 output message。

右侧为嵌入的 xterm 工具，可以将要执行的命令在本地做测试，以验证期望效果。

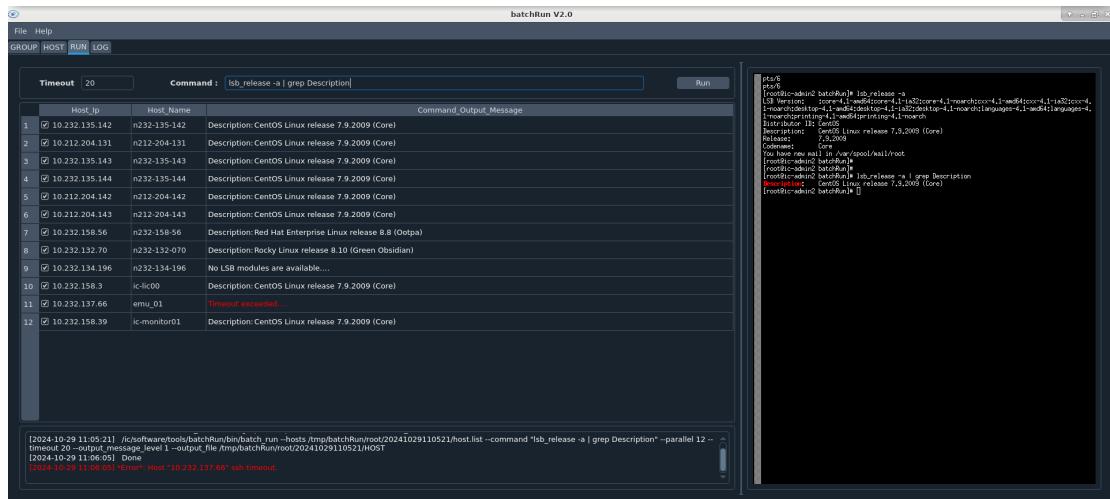
上侧可以设定 ssh 的 timeout，以及要执行的 command，点击“Run”按钮开始并行执行。



执行过程是在后台全并行执行的，如果遇到 timeout 或者 command not found 之类的问题，它会尝试自动解决并再次运行。

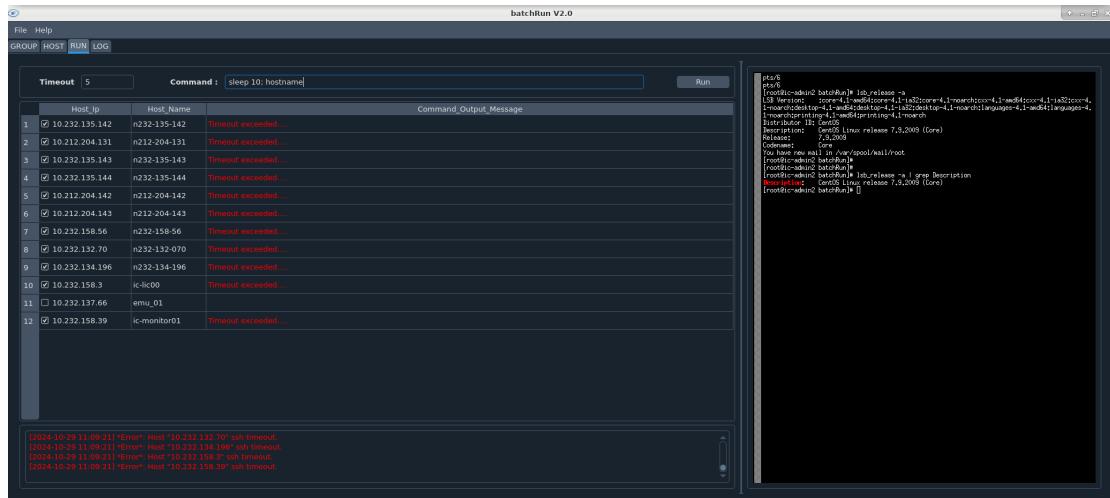
执行完毕后，每台机器上 command 的 output message 会输出在表格中，可以起到信息收集的作用，一些已知异常会显示为红色。

操作的过程及一些可归类的警告信息会显示在下侧的日志框中。

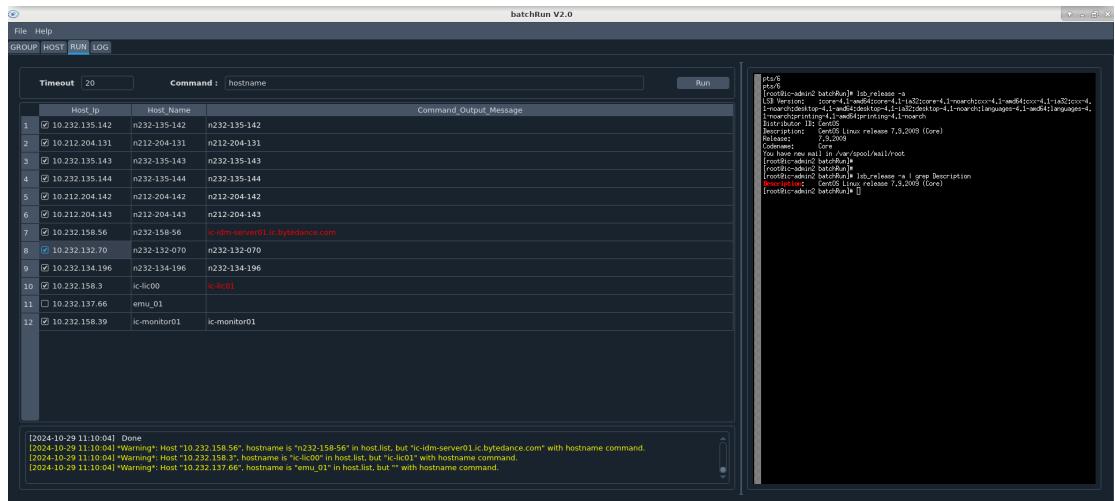


中间的表格，点击“Host\_Ip”可以全选/全部取消下面的服务器选项，每台服务器也可以单独设定选中状态，仅选中的服务器上会执行指定的 command。（针对无法连接的服务器，可以取消选中）

下方的日志框，针对 ssh 失败，或者 hostname 不匹配等异常情况，会以 Warning/Error 信息的形式提示，比如人为制造一种服务器登录执行超时的情况，日志框会提示 timeout 的问题。

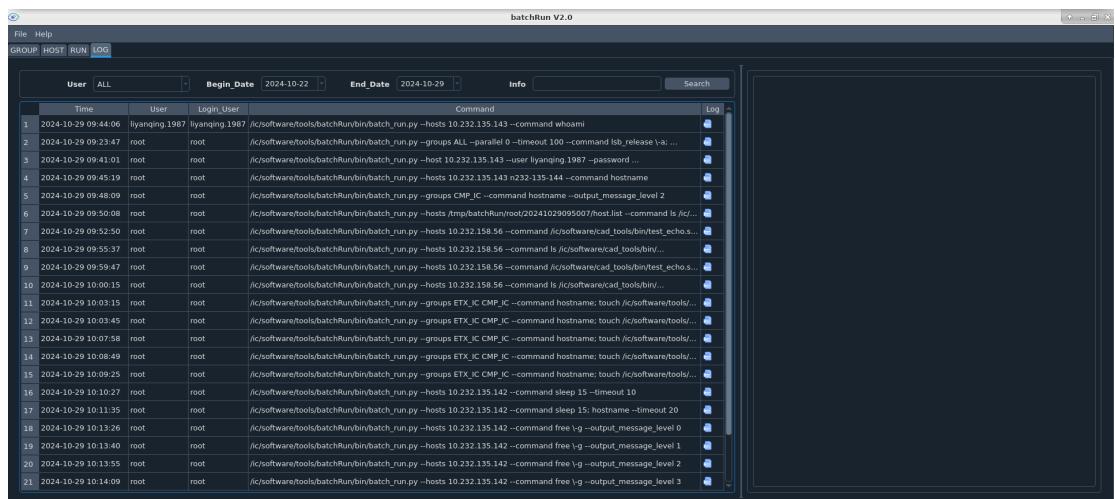


这一功能也可以用来验证/etc/hosts (host.list) 中的 host\_ip 和 host\_name 对应关系是否正确。

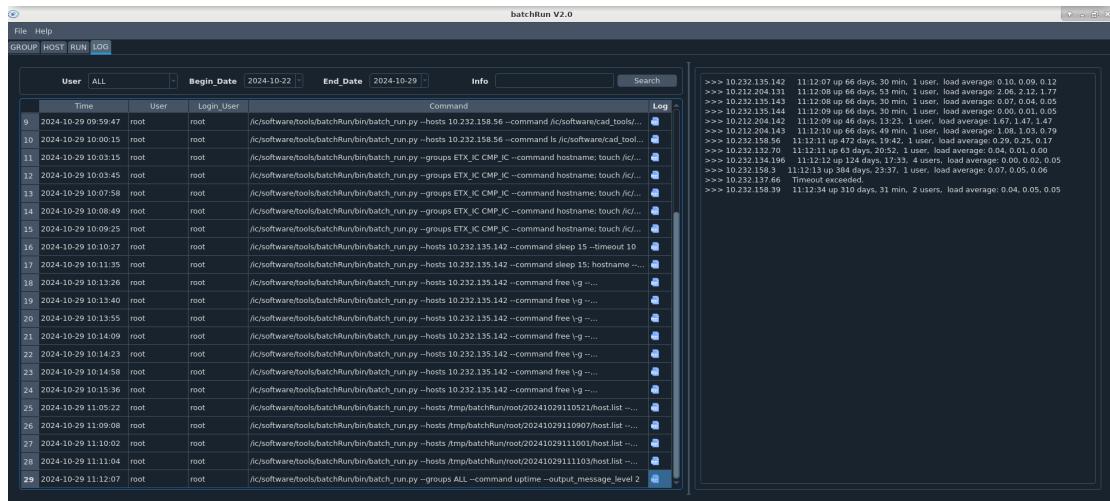


#### 4.3.4 LOG 页

LOG 页用来检索和查询 batchRun 的操作记录，其中 root 账号可以查看所有人的操作记录，普通账号则仅可以查看自己的操作记录。



如果只对指定的 command 感兴趣，可以通过“Info”项做一次过滤，点击每一条记录最后侧的 Log 图标，则可以在右侧的文本框中展示当次操作的 log 内容。(右侧日志框的宽度可以左右拖动，以展示完备的日志信息)



## 五、辅助工具

batchRun 的辅助工具位于 tools 目录下。

### 5.1 tools/switch\_etc\_hosts

工具 switch\_etc\_hosts 用于将格式化的/etc/hosts 文件转换为 batchRun 的 host.list 文件，用法如下。

```
[root@ic-admin2 batchRun]# tools/switch_etc_hosts -h
usage: switch_etc_hosts.py [-h] [-e EXPECTED_IPS
[EXPECTED_IPS ...]] [-E EXCLUDED_IPS [EXCLUDED_IPS ...]] [-i
INPUT_FILE] [-o OUTPUT_FILE] [-a APPEND] [-r] [-t
{batchRun,ansible}]

optional arguments:
  -h, --help            show this help message and exit
  -e EXPECTED_IPS [EXPECTED_IPS ...], --expected_ips EXPECTED_IPS
[EXPECTED_IPS ...]
                                         Specify expected ip(s), support regular
                                         expressions.
  -E EXCLUDED_IPS [EXCLUDED_IPS ...], --excluded_ips EXCLUDED_IPS
[EXCLUDED_IPS ...]
                                         Specify excluded ip(s), support regular
                                         expressions.
  -i INPUT_FILE, --input_file INPUT_FILE
                                         Specify input file, default is
                                         "/etc/hosts".
  -o OUTPUT_FILE, --output_file OUTPUT_FILE
                                         Specify output file, default is
                                         "./host.list".
  -a APPEND, --append APPEND
                                         Append configuration file into output
                                         file.
  -r, --rewrite          Rewrite mode, rewrite output file by
force.
  -t {batchRun,ansible}, --tool {batchRun,ansible}
                                         Which tool the host list is for, default
                                         is "batchRun".
```

**--expected\_ips**: 指定期望的 ip，也可以用正则匹配的方式指定 ip 段 (10.212.206.

\*)，如指定，那么 hosts 文件中不在指定列表中 ip 会被抛弃。

**--excluded\_ips**: 指定排除的 ip，如指定，那么 hosts 文件中在指定列表中的 ip 会被抛弃。

**--input\_file**: 指定输入 hosts 文件来源，默认是“/etc/hosts”。

**--output\_file**: 指定输出 host.list 文件路径，默认是“./host.list”。

**--append**: 指定追加文件，会以文本的形式拼接在 output\_file 结尾，一般用于指定组嵌套关系。

**--rewrite**: 强制覆盖模式，如指定，output\_file 在已存在的情况下会被强制覆盖。

**--tool**: 指定输出 host.list 的格式，默认是 batchRun 所支持的格式，也可以选 ansible 所支持的格式。

/etc/hosts 文件需要符合如下格式才能被正确解析。

```
# GROUP : <group>
<host_ip> <host_name>
<host_ip> <host_name> # SSH_PORT=<port>
```

下面是一个示例 hosts 文件。

```
# GROUP : ETX_IC
10.232.135.142 n232-135-142

# GROUP : ETX_FPGA
10.212.204.131 n212-204-131

# GROUP : CMP_IC
10.232.135.143 n232-135-143
10.232.135.144 n232-135-144

# GROUP : CMP_FPGA
10.212.204.142 n212-204-142
10.212.204.143 n212-204-143

# GROUP : RHEL
10.232.158.56 n232-158-56

# GROUP : ROCKY
10.232.132.70 n232-132-070
```

```
# GROUP : UBUNTU
10.232.134.196 n232-134-196

# GROUP : LIC
10.232.158.3 ic-lic00

# GROUP : VM
10.232.137.66 emu_01
10.232.158.39 ic-monitor01 # SSH_PORT=8888
```

我们将如上 hosts 文件转换为 batchRun 的 host.list 文件。(也支持转换为 ansible 格式的 hosts 文件)

```
[root@ic-admin2 batchRun]# tools/switch_etc_hosts --input_file
config/hosts --output_file config/host.list --rewrite --tool
batchRun

Output File : config/host.list
```

转换后的 batchRun host.list 文件如下。

```
[ETX_IC]
10.232.135.142 ssh_host=n232-135-142

[ETX_FPGA]
10.212.204.131 ssh_host=n212-204-131

[CMP_IC]
10.232.135.143 ssh_host=n232-135-143
10.232.135.144 ssh_host=n232-135-144

[CMP_FPGA]
10.212.204.142 ssh_host=n212-204-142
10.212.204.143 ssh_host=n212-204-143

[RHEL]
10.232.158.56 ssh_host=n232-158-56

[ROCKY]
10.232.132.70 ssh_host=n232-132-070
```

```
[UBUNTU]
10.232.134.196 ssh_host=n232-134-196

[LIC]
10.232.158.3 ssh_host=ic-lic00

[VM]
10.232.137.66 ssh_host=emu_01
10.232.158.39 ssh_host=ic-monitor01 ssh_port=8888
```

因为组定义支持嵌套，所以我们可以预先定义一些组嵌套，并以 append 的方式追加到最终的 host.list 文件中。

```
[root@ic-admin2 batchRun]# cat config/host.list.append

[IC]
ETX_IC
CMP_IC

[FPGA]
ETX_FPGA
CMP_FPGA
[root@ic-admin2 batchRun]# tools/switch_etc_hosts --input_file
config/hosts --output_file config/host.list --rewrite --tool
batchRun --append config/host.list.append

Output File : config/host.list
[root@ic-admin2 batchRun]# tail -n 10 config/host.list
[VM]
10.232.158.39 ssh_host=ic-monitor01 ssh_port=8888

[IC]
ETX_IC
CMP_IC

[FPGA]
ETX_FPGA
CMP_FPGA
```

## 5.2 tools/save\_password

工具 save\_password 用于将当前用户密码加密保存。

```
[root@ic-admin2 batchRun]# tools/save_password -h
usage: save_password.py [-h] -p PASSWORD [-H HOST] [-o
OUTPUT_FILE]

optional arguments:
-h, --help            show this help message and exit
-p PASSWORD, --password PASSWORD
                      Specify user password.
-H HOST, --host HOST  Specify the host which user password works
on, default is "default".
-o OUTPUT_FILE, --output_file OUTPUT_FILE
                      Specify the output file, default is
"<db_path>/password/<user>".
```

**--password\_file**: 用于指定保存账号密码信息的文件，一般无需指定，采用默认值即可。

**--host**: 指定账号密码生效的机器，默认为所有机器。

**--password**: 指定用户明文密码。

下面是一个例子。

```
[root@ic-admin2 batchRun]# tools/save_password --password ***
[root@ic-admin2 batchRun]# cat db/password/root
root  default  1724049499  ed5fa5a1d09c8f4bf3e99c55708be23a
00b2a6273de7bc43a83f9bc352822827
```

用户密码被加密保存后，在工具引用时可以直接读取，无需明文输入，从而保证了密码安全性。

## 5.3 tools/sample\_host\_info

工具 sample\_host\_info 用于批量采集多服务器的 OS 和 hardware 信息，并保存为 json 文件备用。

其底层是借助了 batch\_run 的--output\_file 功能，所以会接受一些 batch\_run 的基本参数并传递给 batch\_run 执行，然后将 batch\_run 的输出信息整理后重新保存。

```
[root@ic-admin2 batchRun]# tools/sample_host_info -h
usage: sample_host_info.py [-h] [-H HOSTS [HOSTS ...]] [-G GROUPS
[GROUPS ...]] [-u USER] [-p PASSWORD] [-P PARALLEL] [-t TIMEOUT]
[-o OUTPUT_DIR]

options:
-h, --help            show this help message and exit
-H HOSTS [HOSTS ...], --hosts HOSTS [HOSTS ...]
                      Specify host(s) for batch_run.
-G GROUPS [GROUPS ...], --groups GROUPS [GROUPS ...]
                      Specify host group(s) for batch_run.
-u USER, --user USER  Specify the user identity for SSH login to
specified host.
-p PASSWORD, --password PASSWORD
                      Specify the user password for SSH login to
specified host.
-P PARALLEL, --parallel PARALLEL
                      Specify the parallelism for batch_run.
-t TIMEOUT, --timeout TIMEOUT
                      Specify the timeout for batch_run.
-o OUTPUT_DIR, --output_dir OUTPUT_DIR
                      Specify host info output directory,
default is "<db_path>/host_info".
```

**--host**: 指定 host, “ALL”意味着 host.list 中所有的服务器。

**--groups**: 指定 group, “ALL”意味着 host.list 中所有的服务器，跟--host 参数二选一即可。

**--user**: 指定 user, 非必须。

**--password**: 指定 password, 非必须。

**--parallel**: 指定并行度，一般推荐“0”，即全部并行。

**--timeout**: 指定 ssh 超时时间，一般采用默认值即可，如果总是采样失败，可以适当调大。

**--output\_dir**: 指定输出路径，默认为“<db\_path>/host\_info”。

下面是一个例子。

```
[root@ic-admin2 batchRun]# tools/sample_host_info --groups ALL --parallel 0 --timeout 100
>>> Clean up /ic/software/tools/batchRun/db/host_info ...
    rm -rf *.info host_list.json host_info.json
>>> Sampling host list information ...
    /ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 100 --list --output_file
/ic/software/tools/batchRun/db/host_info/host_list.json
* Host(s) info has been saved into
"/ic/software/tools/batchRun/db/host_info/host_list.json".
>>> Sampling host os/cpu/mem information ...
    /ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 100 --command "lsb_release -a; hostnamectl;
lscpu; free -g" --output_message_level 1 --output_file
/ic/software/tools/batchRun/db/host_info/HOST.info

Total 12 hosts. (Runtime: 3 minutes 21 seconds)
>>> Collecting host information ...
    Host info has been saved to file
"/ic/software/tools/batchRun/db/host_info/host_info.json".
```

生成的 host\_info.json 文件内容如下。

```
{
    "10.212.204.131": {
        "server_type": "server",
        "os": "CentOS Linux release 7.9.2009 (Core)",
        "cpu_architecture": "x86_64",
        "cpu_thread": 32,
        "thread_per_core": 1,
        "cpu_model": "Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz",
        "cpu_frequency": 2.8,
        "cpu_frequency_unit": "GHz",
        "mem_size": 376,
        "mem_size_unit": "GB",
        "swap_size": 255,
        "swap_size_unit": "GB"
    },
    ...
}
```

## 5.4 tools/patch

工具 patch 是帮助 batchRun 打补丁的工具，其帮助信息如下。

```
[root@ic-admin2 batchRun]# tools/patch -h
usage: patch.py [-h] [-p PATCH_PATH]

optional arguments:
  -h, --help            show this help message and exit
  -p PATCH_PATH, --patch_path PATCH_PATH
                        Specify patch path (new install package
path).
```

**--patch\_path**: 指定补丁包（也就是新的安装包）路径。

一般而言，batchRun 的版本变更，主要是新增 python 脚本，或者是现有 python 脚本内容变更，针对这种变更的安装包，patch 可以自动将新的安装包变更更新到当前安装包。

下面是一个示例。

```
[root@ic-admin2 batchRun]# tools/patch -p ~/batchRun-master
Install Path : /ic/data/usr/liyanqing.1987/tools/batchRun
Patch    path : /home/liyanqing.1987/batchRun-master

*Warning*: current install path name is "batchRun", but patch path
name is "batchRun-master".
Do you want to continue? (y|n) y

> Copying python file "/home/liyanqing.1987/batchRun-
master/bin/batch_run.py" into
"/ic/data/usr/liyanqing.1987/tools/batchRun/bin/batch_run.py".
> Copying python file "/home/liyanqing.1987/batchRun-
master/common/common.py" into
"/ic/data/usr/liyanqing.1987/tools/batchRun/common/common.py".
> Copying python file "/home/liyanqing.1987/batchRun-
master/tools/encrypt_python.py" into
"/ic/data/usr/liyanqing.1987/tools/batchRun/tools/encrypt_python.p
y"
```

## 六、技术支持

本工具为开源工具，由开源社区维护，可以提供如下类型的技术支持：

- 部署和使用技术指导。
- 接收 bug 反馈并修复。
- 接收功能修改建议。(需审核和排期)

获取技术支持的方式包括：

- 通过 Contact 邮箱联系开发者。
- 添加作者微信 “liyanqing\_1987”，注明“真实姓名/公司/batchRun”，由作者拉入技术支持群。



# 附录

## 附 1. 变更历史

日期	版本	变更描述
2022.12	1.0	发布第一个正式 release 版本
2023.7	1.1	增加对 host_ip 和 host_name 的多对多映射关系的支持。 去除对 LSF queue 机器获取的支持。
2024.8	1.2	增加服务器信息采样功能。
2024.10	2.0	增加 GUI 界面。