

batchRun 用户手册

Product Name : batchRun

Product Version : V2.2

Release Date : 2025.02.16

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	5
1.2.1 命令行模式	5
1.2.2 图形界面模式	6
二、环境依赖	10
2.1 操作系统依赖	10
2.2 PYTHON 版本依赖	10
2.3 系统组件依赖	10
三、工具安装、配置及采样	11
3.1 工具下载	11
3.2 工具安装	12
3.3 工具配置	14
3.3.1 config/config.py	14
3.3.2 config/host.list	15
3.3.3 config/password.encrypted	16
3.4 工具采样	17
3.4.1 网络扫描	17
3.4.2 采集设备静态信息	17
3.4.3 采集设备集群归属信息	17
3.4.4 采集设备动态信息	18
3.4.5 准备资产信息	18
3.5 安装/配置/采样信息说明	19
四、工具使用	20
4.1 工具载入	20
4.2 COMMAND LINE 功能介绍	20
4.2.1 帮助信息	20
4.2.2 打印 batchRun 版本信息。	22
4.2.3 列出指定机器	23
4.2.4 指定用户名和密码登录远程机器	24
4.2.5 采用内置加密密码登录远程机器	24
4.2.6 指定机器执行命令	25
4.2.7 指定机器组执行命令	26
4.2.8 执行命令 (command)	27
4.2.9 串行执行和并行执行	28
4.2.10 超时时间 (timeout)	29
4.2.11 输出信息层级	30
4.2.12 输出文件	32
4.3 GUI 功能介绍	33
4.3.1 菜单栏	34

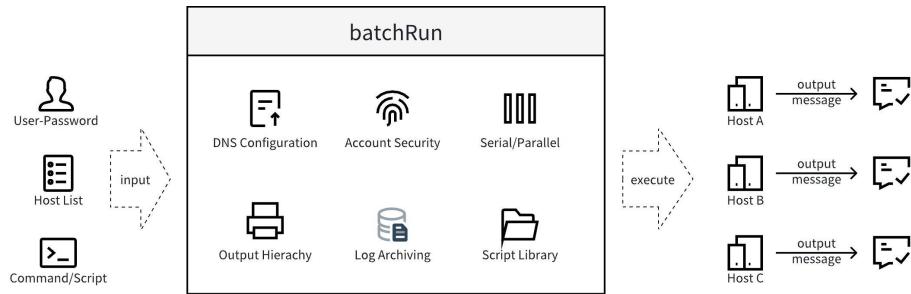
4.3.2 SCAN 页	35
4.3.3 ASSET 页	36
4.3.4 HOST 页	37
4.3.5 STAT 页	38
4.3.6 RUN 页	39
4.3.7 LOG 页	40
五、辅助工具	41
5.1 SAVE_PASSWORD	41
5.2 SWITCH_ETC_HOSTS	42
5.3 NETWORK_SCAN	46
5.4 SAMPLE_HOST_INFO	47
5.5 SAMPLE_HOST_QUEUE	50
5.6 SAMPLE_HOST_STAT	52
5.7 PATCH	55
六、技术支持	57
附录	58
附 1. 变更历史	58
附 2. CONFIG/HOST.LIST 编写规则	58
附 3. BATCHRUN 检查功用	60
【发现非法设备】	60
【发现未配置资产】	61
【发现错误系统配置】	62
【发现设备状态异常】	62

一、简介

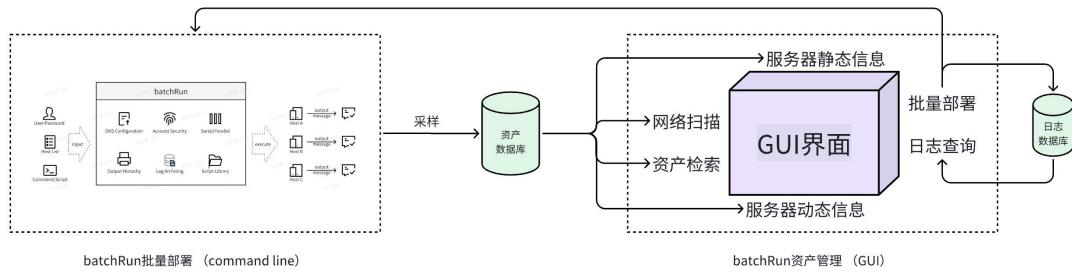
1.1 功能简介

batchRun 是 ByteDance 开源的一款 HPC 批量部署和资产管理工具，采用所见即所得的命令执行方式，普遍适用于 RHEL、Centos、Rocky、Ubuntu 等 Linux 操作系统环境。

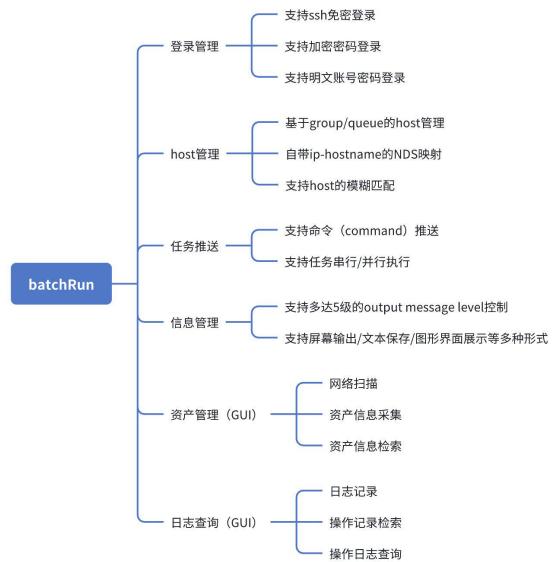
batchRun 基于 python 语言开发，采用 agentless 架构，通过 ssh 协议与被管理主机通讯，配置和使用便捷。



batchRun 支持命令行和图形界面两种使用方式，其中命令行的使用方式侧重于批量部署功能，图形界面的使用方式则同时支持批量部署和资产管理功能。



在基础功能上，batchRun 支持的主要特性如下：



1.2 快速了解

batchRun 支持命令行模式和图形界面模式，其中命令行模式主要用于批量部署（执行），图形界面模式则额外增加了资产管理功能。

1.2.1 命令行模式

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 IC_CMP --command "systemctl disable firewalld"

>>> 10.232.135.143
      Removed symlink /etc/systemd/system/multi-
      user.target.wants/firewalld.service.
      Removed symlink /etc/systemd/system/dbus-
      org.fedoraproject.FirewallD1.service.

>>> 10.232.135.144
```

```
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
```

```
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
```

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

1.2.2 图形界面模式

batchRun 的图形界面包含 SCAN/ASSET/HOST/STAT/RUN/LOG 几个页面。

SCAN 页：

在指定业务区域 (zone) 和网段 (network) 的前提下，用来展示网络扫描结果，目的是及时发现指定网络内超出业务管理范围的设备 (ip)。

host	zone	network	host_ip	host_name	groups	packet	received	packet_loss	rtt_avg
1	huailai_storage	10.151.171.192/26	10.151.171.194			3	3	0%	2.515 ms
2	huailai_storage	10.151.171.192/26	10.151.171.195	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	2.361 ms
3	huailai_storage	10.151.171.192/26	10.151.171.196	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	3.051 ms
4	huailai_storage	10.151.171.192/26	10.151.171.197	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	2.392 ms
5	huailai_storage	10.151.171.192/26	10.151.171.198	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	2.423 ms
6	huailai_storage	10.151.171.192/26	10.151.171.199	hfller01-lif01	OTHERS_STORAGE_fas2720	3	3	0%	2.464 ms
7	huailai_storage	10.151.171.192/26	10.151.171.200	hfller01-lif02	OTHERS_STORAGE_fas2720	3	3	0%	2.504 ms
8	huailai_storage	10.151.171.192/26	10.151.171.201	fas2720-node01-sp	OTHERS_STORAGE_fas2720	3	3	0%	3.784 ms
9	huailai_storage	10.151.171.192/26	10.151.171.202	fas2720-node02-sp	OTHERS_STORAGE_fas2720	3	3	0%	4.833 ms
10	huailai_storage	10.151.171.192/26	10.151.171.203	fas2720-cluster_mgmt	OTHERS_STORAGE_fas2720	3	3	0%	2.413 ms
11	huailai_storage	10.151.171.192/26	10.151.171.204			3	3	0%	2.342 ms
12	huailai_storage	10.151.171.205	10.151.171.205	fas2720-s3_data_lif01	OTHERS_STORAGE_fas2720	3	3	0%	2.572 ms
13	huailai_storage	10.151.171.192/26	10.151.171.207			3	3	0%	2.405 ms
14	huailai_storage	10.151.171.192/26	10.151.171.208			3	3	0%	3.258 ms
15	huailai_storage	10.151.171.192/26	10.151.171.209			3	3	0%	2.356 ms
16	huailai_storage	10.151.171.192/26	10.151.171.210			3	3	0%	2.367 ms
17	huailai_storage	10.151.171.192/26	10.151.171.211			3	3	0%	3.14 ms
18	huailai_storage	10.151.171.192/26	10.151.171.212			3	3	0%	3.199 ms
19	huailai_storage	10.151.171.192/26	10.151.171.213			3	3	0%	2.212 ms
20	huailai_storage	10.151.171.192/26	10.151.171.215	fas2820...	OTHERS_STORAGE_fas2820	3	3	0%	2.672 ms
21	huailai_storage	10.151.171.192/26	10.151.171.216	fas2820...	OTHERS_STORAGE_fas2820	3	3	0%	2.324 ms
22	huailai_storage	10.151.171.192/26	10.151.171.217	fas2820-node03-sp	OTHERS_STORAGE_fas2820	3	3	0%	2.711 ms
23	huailai_storage	10.151.171.192/26	10.151.171.218	fas2820-node04-sp	OTHERS_STORAGE_fas2820	3	3	0%	2.631 ms
24	huailai_storage	10.151.171.192/26	10.151.171.219	hfller01-lif03	OTHERS_STORAGE_fas2820	3	3	0%	2.297 ms

ASSET 页：

在指定业务资产信息文件的前提下，用来展示业务资产信息，业务资产信息一般来源于公司的其它资产管理系统，此处需提供指定格式的 json 文件。

host_ip	host_name	groups	psm	isc	rack	tor	nettype	kind	package	manufacturer	asset_no
1 10.206.112.24					LFRZ_A1M1_102-09-		25equal	物理机		20210301-is...	
2 10.211.61.94					LFRZ_A1B_502-30-05		25equal	物理机		2021-nc-srv-001143	
3 10.212.174.130	iflier01-if01	OTHERS_STORAGE_-			LFRZ_A1B_606-11-04		25equal	物理机		2022-nedapp...	
4 10.212.174.132	ic-zfs05	OTHERS_STORAGE_-			LFRZ_A1B_606-15-03		25equal	物理机		2022-ns-srv-011150	
5 10.212.174.135	ic-zfs06	OTHERS_STORAGE_-			LFRZ_A1B_606-15-03		25equal	物理机		2022-ns-srv-011159	
6 10.212.174.136	ic-zfs07	OTHERS_STORAGE_-			LFRZ_A1B_606-15-03		25equal	物理机		2022-ns-srv-011185	
7 10.212.204.130	r212-204-130	FPGA_SSH			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
8 10.212.204.131	r212-204-131	FPGA_SSH			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
9 10.212.204.133	r212-204-133	FPGA_CMP			LFRZ_A1B_606-14-06		25equal	物理机		2024-ns-srv-1644295	
10 10.212.204.134	r212-204-134	FPGA_CMP			LFRZ_A1B_606-14-07		25equal	物理机		2024-ns-srv-1644296	
11 10.212.204.135	r212-204-135	FPGA_CMP			LFRZ_A1B_606-14-07		25equal	物理机		2024-ns-srv-1644297	
12 10.212.204.140	r212-204-140	FPGA_ETX			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
13 10.212.204.141	r212-204-141	FPGA_SSH			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
14 10.212.204.142	r212-204-142	FPGA_ETX_FPGA_CMP			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
15 10.212.204.143	r212-204-143	FPGA_CMP			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
16 10.212.204.144	r212-204-144	FPGA_ETX_FPGA_CMP			LFRZ_A1B_606-14-07		25equal	物理机		20200810-is...	
17 10.212.204.145	r212-204-145	FPGA_CMP			LFRZ_A1B_606-14-08		25equal	物理机		20200810-is...	
18 10.212.204.146	r212-204-146	FPGA_CMP			LFRZ_A1B_606-14-08		25equal	物理机		20200810-is...	
19 10.212.204.147	r212-204-147	FPGA_CMP			LFRZ_A1B_606-14-08		25equal	物理机		20200810-is...	
20 10.212.204.148	r212-204-148	FPGA_CMP			LFRZ_A1B_606-14-08		25equal	物理机		20200810-is...	
21 10.212.204.149	r212-204-149	FPGA_CMP			LFRZ_A1B_606-14-08		25equal	物理机		20200810-is...	
22 10.212.204.150	r212-204-150	FPGA_CMP			LFRZ_A1B_606-14-06		25equal	物理机		20200921-sg...	
23 10.212.204.151	r212-204-151	FPGA_CMP			LFRZ_A1B_606-14-06		25equal	物理机		20200921-sg...	
24 10.212.204.152	r212-204-152	FPGA_CMP			LFRZ_A1B_606-14-06		25equal	物理机		20200917-is...	

HOST 页：

在进行了服务器静态信息采样的前提下，用来展示服务器的 server_type/os/cpu/mem 等软硬件信息，以及 scheduler/cluster/queues 等集群归属信息。

host_ip	host_name	groups	server_type	os	cpu_arch	cpu_model	cpu_thread
1 10.232.158.51	r232-158-051	IC_DATATRANS	vm	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	16
2 10.232.134.79	r232-134-079	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
3 10.232.129.14	r232-129-014	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 836C CPU @ 2.90GHz	128
4 10.232.134.66	r232-134-066	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
5 10.232.134.67	r232-134-067	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8336C CPU @ 2.30GHz	128
6 10.332.134.69	r232-134-069	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
7 10.332.134.73	r232-134-073	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
8 10.232.134.78	r232-134-078	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
9 10.232.134.130	r232-134-130	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
10 10.232.134.131	r232-134-131	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
11 10.232.134.141	r232-134-141	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
12 10.232.134.142	r232-134-142	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
13 10.232.134.143	r232-134-143	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
14 10.232.134.144	r232-134-144	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
15 10.232.134.145	r232-134-145	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
16 10.232.134.146	r232-134-146	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
17 10.232.134.147	r232-134-147	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
18 10.232.134.194	r232-134-194	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
19 10.232.134.195	r232-134-195	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
20 10.232.134.198	r232-134-198	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
21 10.232.134.215	r232-134-215	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128
22 10.232.134.216	r232-134-216	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128

STAT 页:

在进行了服务器动态信息采样的前提下，用来展示服务器的cpu_load/mem_usage/swap_usage等负载信息。

File	Function	Help					
SCAN	ASSET	HOST	STAT		RUN	LOG	
Date	Time		Select	Check	toAsset	toHost	toRun
host_ip	host_name	groups	up_days	users	tasks	r1m	r5m
1	10.232.134.06	IC_EXK	153	613	4396	0.85	2.27
2	10.232.135.13	IC_EXK	131	728	4259	2.68	1.4
3	10.232.134.69	n232-134-069	165	691	4492	5.31	5.64
4	10.232.134.142	n232-134-142	165	872	5350	5.63	5.86
5	10.232.137.137	n232-137-137	170	1	689	0	0.01
6	10.232.134.217	n232-134-217	165	414	3534	0.87	0.87
7	10.232.137.13	n232-137-013	165	451	5419	5.32	7.22
8	10.232.134.216	EMU_S1	165	643	4873	2.6	2.02
9	10.232.134.194	IC_EXK	134	772	4998	1.15	0.96
10	10.232.135.81	n232-135-091	219	1	2343	1.07	0.94
11	10.212.204.142	n212-204-142	29	173	2926	2.57	2.3
12	10.232.134.73	n232-134-073	165	672	4978	1.52	2.24
13	10.232.134.67	n232-134-067	153	586	4668	1.97	4.48
14	10.232.134.130	IC_EXK	146	754	4895	2.48	2.99
15	10.232.134.195	n232-134-195	131	602	4800	2.32	1.14
16	10.232.135.96	n232-135-066	104	1	1267	0.05	0.12
17	10.232.136.16	EMU_EXK	165	450	5725	3.82	6.12
18	10.232.134.78	n232-134-078	165	865	4994	2.49	2.41
19	10.232.134.143	n232-134-143	165	992	6325	2.22	3.36
20	10.232.134.131	IC_EXK	131	793	4977	4.47	2.4
21	10.232.137.137	n232-137-137	170	3	730	1.01	0.64
22	10.232.137.140	n232-137-140	41	1	687	0	0.01
23	10.232.134.146	n232-134-146	165	763	4505	1.8	1.93
24	10.212.204.143	FPGA_CMP	165	1	549	6.94	6.96

RUN 页:

在选中的服务器上批量推送执行的命令 (Command), 同时获取并 (单行) 展示命令的输出信息。

The screenshot shows the 'batchRun V2.2' application window. At the top, there's a menu bar with 'File', 'Function', 'Help', and tabs for 'SCAN', 'ASSET', 'HOST', 'STAT', 'RUN', and 'LOG'. Below the menu is a search bar with 'Command : lsb_release -a | grep Description' and a 'Run' button. The main area displays a table of host information:

	host_ip	host_name	command_output_message
1	<input checked="" type="checkbox"/> 10.212.206.197	n212-206-197	
2	<input checked="" type="checkbox"/> 10.212.206.237	n212-206-237	
3	<input checked="" type="checkbox"/> 10.232.158.51	n232-158-051	Description: CentOS Linux release 7.9.2009 (Core)
4	<input checked="" type="checkbox"/> 10.232.159.51	n232-159-051	
5	<input checked="" type="checkbox"/> 10.232.134.79	n232-134-079	Description: CentOS Linux release 7.9.2009 (Core)
6	<input checked="" type="checkbox"/> 10.232.129.14	n232-129-014	Description: CentOS Linux release 7.9.2009 (Core)
7	<input checked="" type="checkbox"/> 10.232.134.66	n232-134-066	Description: CentOS Linux release 7.9.2009 (Core)
8	<input checked="" type="checkbox"/> 10.232.134.67	n232-134-067	Description: CentOS Linux release 7.9.2009 (Core)
9	<input checked="" type="checkbox"/> 10.232.134.69	n232-134-069	Description: CentOS Linux release 7.9.2009 (Core)
10	<input checked="" type="checkbox"/> 10.232.134.73	n232-134-073	Description: CentOS Linux release 7.9.2009 (Core)
11	<input checked="" type="checkbox"/> 10.232.134.78	n232-134-078	Description: CentOS Linux release 7.9.2009 (Core)
12	<input checked="" type="checkbox"/> 10.232.134.130	n232-134-130	Description: CentOS Linux release 7.9.2009 (Core)
13	<input checked="" type="checkbox"/> 10.232.134.131	n232-134-131	Description: CentOS Linux release 7.9.2009 (Core)
14	<input checked="" type="checkbox"/> 10.232.134.141	n232-134-141	Description: CentOS Linux release 7.9.2009 (Core)
15	<input checked="" type="checkbox"/> 10.232.134.142	n232-134-142	Description: CentOS Linux release 7.9.2009 (Core)
16	<input checked="" type="checkbox"/> 10.232.134.143	n232-134-143	Description: CentOS Linux release 7.9.2009 (Core)
17	<input checked="" type="checkbox"/> 10.232.134.144	n232-134-144	Description: CentOS Linux release 7.9.2009 (Core)
18	<input checked="" type="checkbox"/> 10.232.134.145	n232-134-145	Description: CentOS Linux release 7.9.2009 (Core)
19	<input checked="" type="checkbox"/> 10.232.134.146	n232-134-146	Description: CentOS Linux release 7.9.2009 (Core)

At the bottom left, there's a log window showing command outputs:

```
[2025-02-05 17:42:50] *Error* Host: [10.232.159.63] ssh timeout.
[2025-02-05 17:42:50] *Error* Host: [10.232.119.52] ssh timeout.
[2025-02-05 17:42:50] *Error* Host: [10.232.134.143] ssh timeout.
[2025-02-05 17:42:50] *Error* Host: [10.232.134.144] ssh timeout.
[2025-02-05 17:42:50] *Error* Host: [10.232.134.145] ssh timeout.
```

LOG 页:

按照用户/日期/信息来检索 batch_run 的执行记录，同时可以查询每次执行命令的具体输出信息。

The screenshot shows the 'batchRun V2.2' application window. At the top, there's a menu bar with File, Function, Help, SCAN, ASSET, HOST, STAT, RUN, and LOG. Below the menu is a search bar labeled 'Search'. The main area has several tabs: User, ALL, Begin Date (set to 2025-01-29), End Date (set to 2025-02-05), Info, and Search. A 'log' tab is also present.

time	user	login_user	command																																																						
2945	2025-02-05 16:10:02	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts 10.232.158.70 --parallel 1 --timeout 20 --command ...																																																						
2946	2025-02-05 16:10:03	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts 10.232.158.70 --parallel 1 --timeout 20 --command ...																																																						
2947	2025-02-05 16:10:05	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ...																																																						
2948	2025-02-05 16:11:16	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2949	2025-02-05 16:20:06	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ...																																																			
2950	2025-02-05 16:21:13	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2951	2025-02-05 16:30:05	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2952	2025-02-05 16:31:08	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2953	2025-02-05 16:40:06	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2954	2025-02-05 16:41:16	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2955	2025-02-05 16:50:05	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2956	2025-02-05 16:51:12	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2957	2025-02-05 17:00:05	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2958	2025-02-05 17:01:13	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2959	2025-02-05 17:09:04	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2960	2025-02-05 17:10:06	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2961	2025-02-05 17:11:17	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2962	2025-02-05 17:20:06	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2963	2025-02-05 17:21:14	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2964	2025-02-05 17:30:06	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2965	2025-02-05 17:31:14	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2966	2025-02-05 17:40:06	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --groups ALL --OTHERS_STORAGE_a800 ... 2967	2025-02-05 17:41:26	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /ic/data/CAD/lib/batchRun/db/host_stat/... 2968	2025-02-05 17:41:26	root	/ic/software/cad_tools/lib/batchRun/bin/batch_run.py --hosts /tmp/batchRun/host/20250205174123/host.list ...

On the right side of the window, there is a detailed system status summary:

```

Description: CentOS Linux release 7.9.2009 (Core)
top: 17:41:58 up 165 days, 6:58. 1 user, load average: 90.04, 127.97, 90.98
top: 17:41:57 up 165 days, 6:58. 1 user, load average: 86.38, 78.14, 62.95
top: 17:41:58 up 165 days, 6:59. 2 users, load average: 86.34, 83.73, 69.30
top: 17:41:58 up 165 days, 6:59. 2 users, load average: 86.34, 83.73, 69.30
top: 17:41:58 up 16 days, 22:33. 1 user, load average: 66.11, 66.20, 66.24
Tasks: 880 total, 23 running, 458 sleeping, 0 stopped, 2 zombie
Cpu(s): 3.60%us, 0.00%sy, 0.00%ni, 0.00%id, 0.00%wa, 0.00%hi, 0.00%st
Tasks: 778 total, 9 running, 769 sleeping, 0 stopped, 0 zombie
Tasks: 1540 total, 36 running, 1498 sleeping, 0 stopped, 0 zombie
Cpu(s): 3.60%us, 0.00%sy, 0.00%ni, 0.00%id, 0.00%wa, 0.00%hi, 0.00%st
Tasks: 937 total, 67 running, 870 sleeping, 0 stopped, 0 zombie
Kib Mem: 21131843+total, 90360697+free, 22757187+used, 98200550+buff/cache
Kib Swap: 1342177+total, 946.0+free, 1247708+used, 8572299+buff/cache
Kib Mem: 21131843+total, 11446604+free, 2705354+used, 8892299+buff/cache
Kib Swap: 1342177+total, 946.0+free, 16171095+used, 5312351+buff/cache
Kib Mem: 21131843+total, 96484019+free, 6171095+used, 5312351+buff/cache
Kib Swap: 1342177+total, 946.0+free, 722776+used, 18620973+avail Mem
Kib Mem: 21131843+total, 11446604+free, 2705354+used, 8892299+buff/cache
Kib Swap: 1342177+total, 946.0+free, 61696+used, 14928035+avail Mem
Kib Mem: 21131843+total, 13415603+free, 61696+used, 14928035+avail Mem
Kib Swap: 1342177+total, 946.0+free, 131072+used, 933428.9+avail Mem
Kib Swap: 1342177+total, 13278993+free, 1427784+used, 16873068+avail Mem
Kib Mem: 21131843+total, 13415603+free, 61696+used, 14928035+avail Mem
Kib Swap: 1342177+total, 946.0+free, 131072+used, 933428.9+avail Mem
PID USER PR NI VIRT RSS SHR S %CPU %MEM TIME+ COMMAND
PID USER PR NI VIRT RSS SHR S %CPU %MEM TIME+ COMMAND
PID USER PR NI VIRT RSS SHR S %CPU %MEM TIME+ COMMAND
PID USER PR NI VIRT RSS SHR S %CPU %MEM TIME+ COMMAND
1.364 comctl 12 0 11.2 9 25948 R 105.9 0.6 25.18 59 /ic/software/cadence/wx/2.3.0.5001/tools/lv86/bin/64bit/xCompile compilePPC.gcl
PID USER PR NI VIRT RSS SHR S %CPU %MEM TIME+ COMMAND

```

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、配置及采样

3.1 工具下载

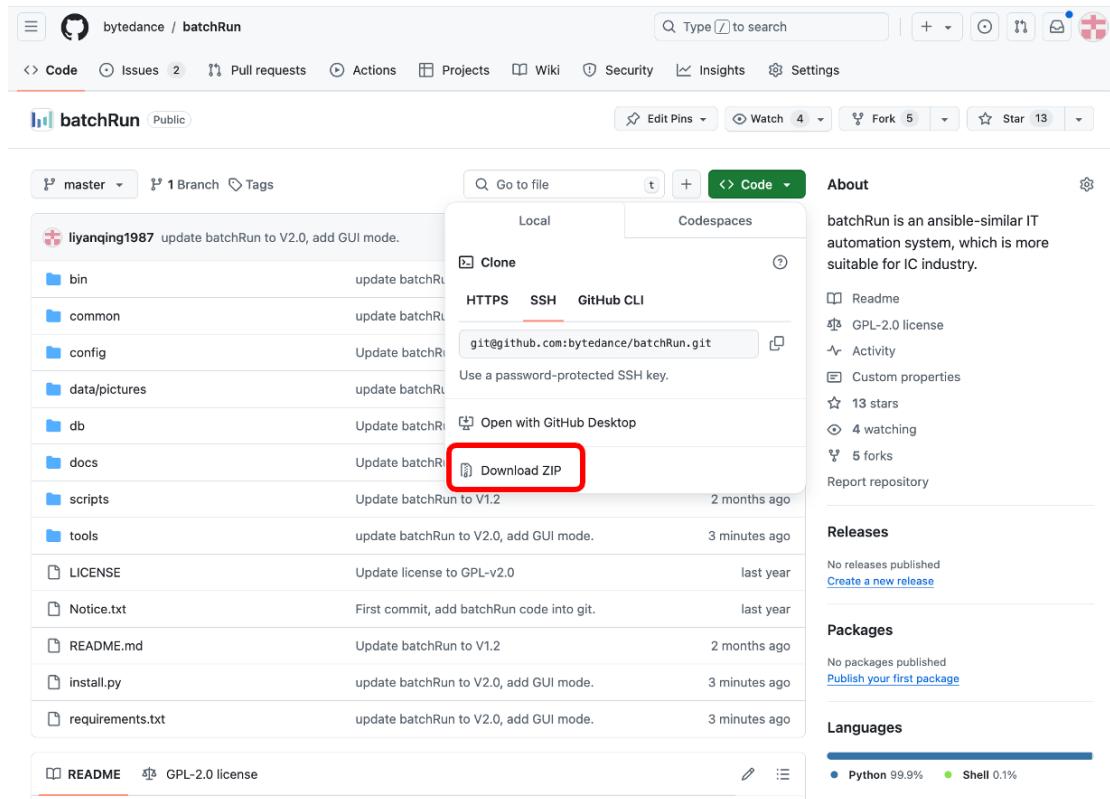
batchRun 的 github 路径位于 <https://github.com/bytedance/batchRun>

The screenshot shows the GitHub repository page for 'batchRun' owned by 'bytedance'. The repository has 18 commits, 5 forks, and 13 stars. The code tab is selected. The repository description states: 'batchRun is an ansible-similar IT automation system, which is more suitable for IC industry.' It includes links to Readme, GPL-2.0 license, Activity, Custom properties, 13 stars, 4 watching, and 5 forks. It also mentions Report repository, Releases (no releases published), Packages (no packages published), and Languages (Python 99.9%, Shell 0.1%).

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。



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/network_scan".
>>> 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/sample_host_queue".
>>> Generate script
"/ic/software/tools/batchRun/tools/sample_host_stat".
>>> Generate script
"/ic/software/tools/batchRun/tools/save_password".
>>> Generate script
"/ic/software/tools/batchRun/tools/show_top_file".
>>> 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'

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

# Illegal command list.
illegal_command_list = ['rm -rf /', 'rm -rf ~', 'rm -rf ~/']

# Command for switch_etc_hosts.
switch_etc_hosts_command =
'/ic/software/tools/batchRun/tools/switch_etc_hosts --rewrite --
tool batchRun'

# Command for network_scan.
network_scan_command =
'/ic/software/tools/batchRun/tools/network_scan --alive'

# Command for sample_host_info.
sample_host_info_command =
'/ic/software/tools/batchRun/tools/sample_host_info --groups ALL'

# Command for sample_host_queue.
sample_host_queue_command =
'/ic/software/tools/batchRun/tools/sample_host_queue --groups ALL'
```

```
# Command for sample_host_stat.  
sample_host_stat_command =  
'/ic/software/tools/batchRun/tools/sample_host_stat --groups ALL'
```

host_list：指定 host.list 文件位置，默认为安装路径下的“config/host.list”。

db_path：指定 batch_run 的数据库路径，默认为安装路径下的“db”。

default_ssh_command：指定默认 ssh 的命令，默认为“ssh -o StrictHostKeyChecking=no -t -q”。

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。

illegal_command_list：指定违法命令列表，写法支持正则匹配，batch_run 将会拒绝指定的违法命令。

switch_etc_hosts_command：指定 batchRun 图形界面上转换/etc/hosts 到 host.list 的命令。

network_scan_command：指定 batchRun 图形界面上进行网络扫描的命令。

sample_host_info_command：指定 batchRun 图形界面上采集服务器静态信息的命令。

sample_host_queue_command：指定 batchRun 图形界面上采集服务器集群归属信息的命令。

sample_host_stat_command：指定 batchRun 图形界面上采集服务器动态负载信息的命令。

3.3.2 config/host.list

host.list 用于定义所有的机器及其分组信息，是必须配置项，否则 batchRun 无法启动。

下面是自带的格式说明。

```
##### 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
```

config/host.list 的编写规则请参照 [附 2. config/host.list 编写规则](#)

config/host.list 一般基于格式化的/etc/hosts 文件自动生成，方法参照 [5.2 switch_etc_hosts](#)

由于/etc/hosts 可能会因为设备变更而变更，因此建议设置 crontab 来自动定时更新，下面是一个例子。

```
# For batchRun, update config/host.list automatically.
*/30 * * * * /ic/software/tools/batchRun/tools/switch_etc_hosts --
rewrite --tool batchRun --expected_hosts 10.151.* 10.212.*
10.232.* 10.249.*
```

特别说明：建议 host.list 中额外增加一个“RUN”组，用于定义日常运维服务器列表，最好确保这些服务器是可 ssh 登陆的。

3.3.3 config/password.encrypted

在使用 batchRun 做多服务器的批量部署时，如果远程服务器没有配置 ssh 免密登陆，则需要使用账号+密码的方式登陆，config/password.encrypted 即在此场景下用于提供加密密码来做自动 ssh 登陆。

config/password.encrypted 由工具 tools/save_password 生成，其用法参照 5.1
save_password

3.4 工具采样

如下工具采样是非必须的，但是如果某项采样缺失，则在 batchRun 的图形界面上则无法看到相关信息。

3.4.1 网络扫描

网络扫描基于工具 tools/network_scan，其使用方法参照 5.3 *network_scan*

网络扫描信息默认保存在<db_path>/network_scan 下。

由于网络环境和设备也是不断变化的，因此建议设置 crontab 来自动定时扫描，下面是一个例子，每天早上 8 点重新扫描一下网络环境。

```
0 8 * * * /ic/software/tools/batchRun/tools/network_scan --alive
```

3.4.2 采集设备静态信息

采集设备静态信息基于工具 tools/sample_host_info，其使用方法参照 5.4
sample_host_info

设备静态信息默认保存在<db_path>/host_info/host_info.json 下。

由于设备可能有上线/下线/变更，因此建议设置 crontab 来自动定时采样，下面是一个例子，每天早上 8 点重新采样所有设备的静态信息。

```
0 8 * * * /ic/software/tools/batchRun/tools/sample_host_info --  
groups ALL
```

3.4.3 采集设备集群归属信息

采集设备集群归属信息基于工具 tools/sample_host_queue，其使用方法参照 5.4
sample_host_queue

设备集群归属信息默认保存在<db_path>/host_queue/host_queue.json 下。

由于集群配置经常变更，因此建议设置 crontab 来自动定时采样，下面是一个例子，每天早上 8 点重新采样设备的集群归属信息。

```
0 8 * * * /ic/software/tools/batchRun/tools/sample_host_queue --groups RUN
```

3.4.4 采集设备动态信息

采集设备动态信息基于工具 tools/sample_host_stat，其使用方法参照 5.4 sample_host_stat

设备动态（负载）信息默认保存在<db_path>/host_stat/host_stat.json 下。

设备的动态（负载）信息是实时变动的，因此建议设置 crontab 来自动定时采样，采样频率过低会导致信息更新过于滞后，采样频率过高会因为频繁 ssh 导致机器负载过重，可以设定一个合适的频率。

下面是一个例子，每 10 分钟采集一次设备动态（负载）信息。

```
*/10 * * * * /ic/software/tools/batchRun/tools/sample_host_stat --groups RUN
```

3.4.5 准备资产信息

如果公司保有数字化的资产管理系统，那么一般可以导出资产信息，这些信息如果符合一定格式就可以在 batchRun 的图形界面上展示。

资产信息默认保存在<db_path>/host_asset/host_asset.json 下。

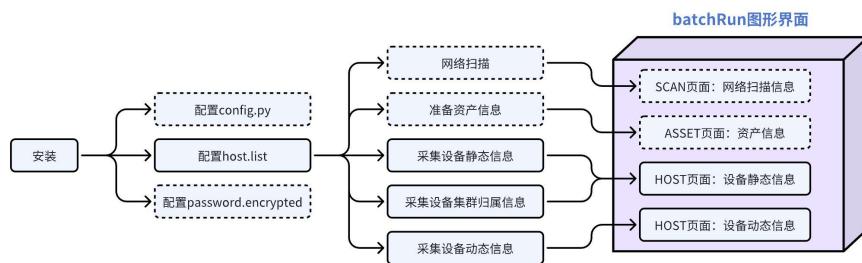
其推荐格式如下，其中第一层 key 必须是 host ip，host ip 下面则以 key-value 的形式保存设备的资产信息。

```
{
  "10.232.135.142": {
    "psm": "ic.orca1.server",
    "idc": "lfrz",
    "rack": "LFRZ_A8_402-14-07",
    "tor": "10.232.128.27",
    "nettype": "25equal",
    "kind": "物理机",
    "package": "S19S1-I8DD2",
    "manufacturer": "Nettrix",
```

```
        "asset_no": "2022-nc-srv-004400"  
    },  
    ...  
}
```

3.5 安装/配置/采样信息说明

batchRun 的安装、配置、采样到图形界面展示的对应关系，如下图所示。



config.py 和 password.encrypted 按需配置，host.list 则必须配置。

“网络扫描”和“资产信息”按需准备，如未准备，在 batchRun 的图形界面上则不显示相关的界面。

“采集设备静态信息”、“采集设备集群归属信息”和“采集设备动态信息”建议执行，如未采样，在 batchRun 的图形界面上展示相关界面，但内容为空。

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 : V2.2
```

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 --groups ALL` 命令查看。

```
[root@ic-admin2 batchRun]# bin/batch_run --list --groups ALL
GROUP : [IC_ETX]
        10.232.135.142    n232-135-142
GROUP : [FPGA_ETX]
        10.212.204.131    n212-204-131
GROUP : [IC_CMP]
        10.232.135.143    n232-135-143
        10.232.135.144    n232-135-144
GROUP : [FPGA_CMP]
        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
GROUP : [IC]
        IC_ETX/
        IC_CMP/
GROUP : [FPGA]
        FPGA_ETX/
        FPGA_CMP/
GROUP : [RUN]
        IC_ETX/
        FPGA_ETX/
        IC_CMP/
        FPGA_CMP/
```

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 IC_CMP
GROUP : [IC_CMP]
        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 1739694549  
ecd7b27c33577a36957800dcf746d652 4864dc0be38107cdd4d1d0600319cf9d
```

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 -> n232-135-142 -> 10.232.135.142
[FUZZY MATCH] .*-142 -> n212-204-142 -> 10.212.204.142

GROUP : [IC_ETX]
10.232.135.142    n232-135-142
GROUP : [FPGA_CMP]
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-*
[FUZZY MATCH] ^n212-.* -> n212-204-131 -> 10.212.204.131
[FUZZY MATCH] ^n212-.* -> n212-204-142 -> 10.212.204.142
[FUZZY MATCH] ^n212-.* -> n212-204-143 -> 10.212.204.143

GROUP : [FPGA_ETX]
10.212.204.131    n212-204-131
GROUP : [FPGA_CMP]
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 IC_CMP
FPGA_CMP
GROUP : [IC_CMP]
```

```
10.232.135.143    n232-135-143
10.232.135.144    n232-135-144
GROUP : [FPGA_CMP]
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 IC_CMP
FPGA_CMP --hosts ~10.232.135.143 ~n212-204-142
GROUP : [IC_CMP]
10.232.135.144    n232-135-144
GROUP : [FPGA_CMP]
10.212.204.143    n212-204-143
```

同时，batch_run 的--groups 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 --groups *_CMP
GROUP : [IC_CMP]
10.232.135.143    n232-135-143
10.232.135.144    n232-135-144
GROUP : [FPGA_CMP]
10.212.204.142    n212-204-142
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 IC_CMP --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)
```

另外，batch_run 还会自动拒绝非法命令，如果尝试执行 config/config.py 中 illegal_command_list 定义的非法命令，batch_run（无论是 command line 还是 GUI）会自动拒绝。

```
[root@ic-admin2 batchRun]# bin/batch_run --groups ALL --command
"rm -rf /"
*Error*: Illegal command is specified.
```

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

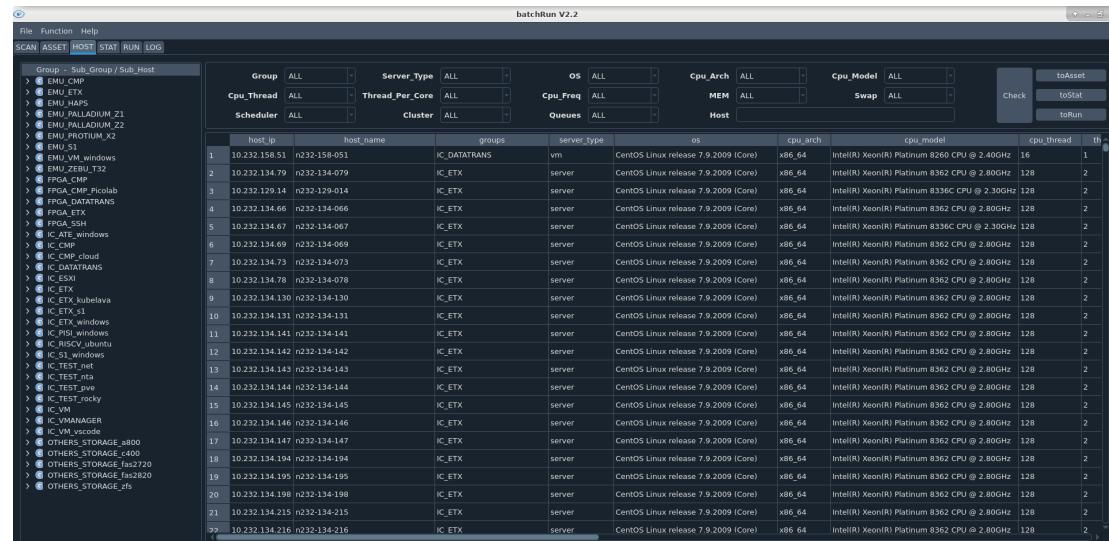
Swap:      127          0        127
```

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。

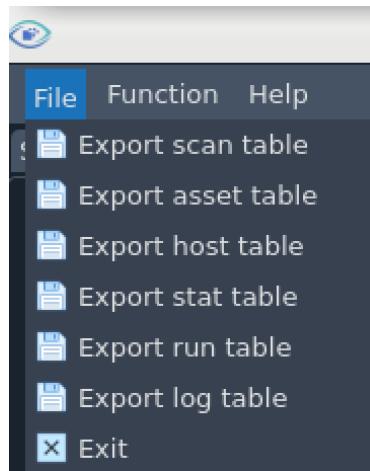
```
[root@ic-admin2 batchRun]# bin/batch_run --gui
[2025-02-06 14:04:01] Loading network scan file
"/ic/data/CAD/it/batchRun/db/network_scan/network_scan.json" ...
[2025-02-06 14:04:01] Loading host asset file
"/ic/data/CAD/it/batchRun/db/host_asset/host_asset.json" ...
[2025-02-06 14:04:01] Loading host queue file
"/ic/data/CAD/it/batchRun/db/host_queue/host_queue.json" ...
[2025-02-06 14:04:01] Collecting host scheduler/cluster/queue
relationship ...

[2025-02-06 14:04:01] Loading host info file
"/ic/data/CAD/it/batchRun/db/host_info/host_info.json" ...
[2025-02-06 14:04:02] Loading host asset file
"/ic/data/CAD/it/batchRun/db/host_asset/host_asset.json" ...
[2025-02-06 14:04:03] Loading host stat file
"/ic/data/CAD/it/batchRun/db/host_stat/20250206/140004/host_stat.j
son" ...
```

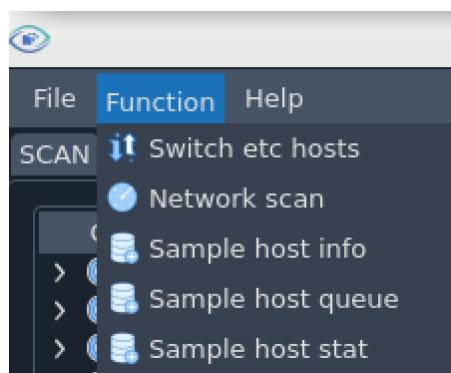


4.3.1 菜单栏

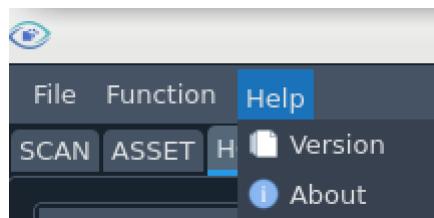
File 菜单栏，包含各个表格的保存功能。



Function 菜单栏，包含 3.4 工具采样章节所讲述的几个采样功能。



Help 菜单栏，包含版本信息和工具简介。



4.3.2 SCAN 页

SCAN 页读取和展示<db_path>/network_scan/network_scan.json 中的内容，即指定网络的扫描结果，用于分析指定网络中的设备情况。

Zone	Network	IP	Host IP	Host Name	Group	Packet	Received	Packet Loss	RTT Avg
> huailai_storage		10.151.171.192/26	10.151.171.194		3	3	0%	2.315 ms	
> huailai_storage		10.151.171.192/26	10.151.171.195	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	2.361 ms
> huailai_storage		10.151.171.192/26	10.151.171.196	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	3.051 ms
> huailai_storage		10.151.171.192/26	10.151.171.197	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	2.392 ms
> huailai_storage		10.151.171.192/26	10.151.171.198	fas2720...	OTHERS_STORAGE_fas2720	3	3	0%	2.423 ms
> huailai_storage		10.151.171.192/26	10.151.171.199	hfller01-if01	OTHERS_STORAGE_fas2720	3	3	0%	2.464 ms
> huailai_storage		10.151.171.192/26	10.151.171.200	hfller01-if02	OTHERS_STORAGE_fas2720	3	3	0%	2.504 ms
> huailai_storage		10.151.171.192/26	10.151.171.201	fas2720-node01-sp	OTHERS_STORAGE_fas2720	3	3	0%	3.784 ms
> huailai_storage		10.151.171.192/26	10.151.171.202	fas2720-node02-sp	OTHERS_STORAGE_fas2720	3	3	0%	4.833 ms
> huailai_storage		10.151.171.192/26	10.151.171.203	fas2720-cluster_mngt	OTHERS_STORAGE_fas2720	3	3	0%	2.413 ms
> huailai_storage		10.151.171.192/26	10.151.171.204		3	3	0%	3.547 ms	
> huailai_storage		10.151.171.192/26	10.151.171.205	fas2720-s3_data_if01	OTHERS_STORAGE_fas2720	3	3	0%	2.572 ms
> huailai_storage		10.151.171.192/26	10.151.171.207		3	3	0%	2.405 ms	
> huailai_storage		10.151.171.192/26	10.151.171.208		3	3	0%	3.258 ms	
> huailai_storage		10.151.171.192/26	10.151.171.209		3	3	0%	2.356 ms	
> huailai_storage		10.151.171.192/26	10.151.171.210		3	3	0%	2.367 ms	
> huailai_storage		10.151.171.192/26	10.151.171.211		3	3	0%	3.14 ms	
> huailai_storage		10.151.171.192/26	10.151.171.212		3	3	0%	3.199 ms	
> huailai_storage		10.151.171.192/26	10.151.171.213		3	3	0%	2.212 ms	
20 huailai_storage		10.151.171.192/26	10.151.171.215	fas2820...	OTHERS_STORAGE_fas2820	3	3	0%	2.672 ms
21 huailai_storage		10.151.171.192/26	10.151.171.216	fas2820...	OTHERS_STORAGE_fas2820	3	3	0%	2.324 ms
22 huailai_storage		10.151.171.192/26	10.151.171.217	fas2820-node03-sp	OTHERS_STORAGE_fas2820	3	3	0%	2.711 ms
23 huailai_storage		10.151.171.192/26	10.151.171.218	fas2820-node04-sp	OTHERS_STORAGE_fas2820	3	3	0%	2.631 ms
24 huailai_storage		10.151.171.192/26	10.151.171.219	hfller01-if03	OTHERS_STORAGE_fas2820	3	3	0%	2.297 ms

左侧侧边栏展示 zone -> network -> ip 之间的层级关系。

右侧表格展示所有网络的扫描结果，仅展示可以 ping 通的设备 ip。

说明：

- 左侧侧边栏内容双击，可以在右侧表格中展示筛选过的内容。
- 右侧表格中的设备 ip 如果不在 host.list 中，则背景色为红色，即非已知设备，需要特别关注。

4.3.3 ASSET 页

ASSET 页读取和展示<db_path>/host_asset/host_asset.json 中的内容，即 json 格式保存的资产信息。

	host_ip	host_name	groups	psm	idc	rack	tor	nettype	kind	package	manufacturer	asset_no
1	10.296.112.24				lfrz	LFRZ_A3H_102-09...						20210301-is...
2	10.211.61.94				lfrz	LFRZ_A1B_502-10-05						2021-rc-srv-001143
3	10.212.174.130	ffller01-lf01	OTHERS_STORAGE_-		lfrz	LFRZ_A1B_606-15-04		25equal	物理机			2022-netapp...
4	10.212.174.132	lc-zfs05	OTHERS_STORAGE_-		lfrz	LFRZ_A1B_606-15-03		25equal	物理机			2022-ss-uv-011150
5	10.212.174.135	lc-zfs06	OTHERS_STORAGE_-		lfrz	LFRZ_A1B_606-15-03		25equal	物理机			2022-ss-uv-011159
6	10.212.174.136	lc-zfs07	OTHERS_STORAGE_-		lfrz	LFRZ_A1B_606-15-03		25equal	物理机			2022-ss-uv-011185
7	10.212.204.130	r212-204-130	FPGA_SSH		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
8	10.212.204.131	r212-204-131	FPGA_SSH		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
9	10.212.204.133	r212-204-133	FPGA_CMP		lfrz	LFRZ_A1B_606-14-06		25equal	物理机			2024-ss-sv-1644295
10	10.212.204.134	r212-204-134	FPGA_CMP		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			2024-ss-sv-1644296
11	10.212.204.135	r212-204-135	FPGA_CMP		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			2024-ss-sv-1644297
12	10.212.204.140	r212-204-140	FPGA_ETX		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
13	10.212.204.141	r212-204-141	FPGA_SSH		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
14	10.212.204.142	r212-204-142	FPGA_ETX_FPGA_CMP		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
15	10.212.204.143	r212-204-143	FPGA_CMP		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
16	10.212.204.144	r212-204-144	FPGA_ETX_FPGA_CMP		lfrz	LFRZ_A1B_606-14-07		25equal	物理机			20200810-is...
17	10.212.204.145	r212-204-145	FPGA_CMP		lfrz	LFRZ_A1B_606-14-08		25equal	物理机			20200810-is...
18	10.212.204.146	r212-204-146	FPGA_CMP		lfrz	LFRZ_A1B_606-14-08		25equal	物理机			20200810-is...
19	10.212.204.147	r212-204-147	FPGA_CMP		lfrz	LFRZ_A1B_606-14-08		25equal	物理机			20200810-is...
20	10.212.204.148	r212-204-148	FPGA_CMP		lfrz	LFRZ_A1B_606-14-08		25equal	物理机			20200810-is...
21	10.212.204.149	r212-204-149	FPGA_CMP		lfrz	LFRZ_A1B_606-14-08		25equal	物理机			20200810-is...
22	10.212.204.150	r212-204-150	FPGA_CMP		lfrz	LFRZ_A1B_606-14-06		25equal	物理机			20200921-sg...
23	10.212.204.151	r212-204-151	FPGA_CMP		lfrz	LFRZ_A1B_606-14-06		25equal	物理机			20200921-sg...
24	10.212.204.152	r212-204-152	FPGA_CMP		lfrz	LFRZ_A1B_606-14-06		25equal	物理机			20200917-is...

表格内容完全由用户提供信息决定（包括标题项）。

说明：

- 表格中的设备 ip 如果不在 host.list 中，则背景色为红色，即已知但未备案设备，需要特别关注。
- Select 项支持 python 的判断语法，标题即为变量项，每个单元格的内容即为变量值，筛选写法为 `idc == "lfrz"`。

4.3.4 HOST 页

HOST 页读取和展示`<db_path>/host_info/host_info.json` 和`<db_path>/host_queue/host_queue.json` 内容，即资产的静态信息和集群归属信息。

host_ip	host_name	groups	server_type	OS	CPU Arch	CPU Model	CPU Thread	Th
10.232.158.51	n232-158-051	IC_DATATRANS	vm	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz	16	1
10.232.134.79	n232-134-079	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.129.14	n232-129-014	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8336C CPU @ 2.30GHz	128	2
10.232.134.66	n232-134-066	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.67	n232-134-067	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8336C CPU @ 2.30GHz	128	2
10.232.134.69	n232-134-069	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.73	n232-134-073	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.78	n232-134-078	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.150	n232-134-130	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.131	n232-134-131	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.141	n232-134-141	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.142	n232-134-142	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.143	n232-134-143	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.144	n232-134-144	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.145	n232-134-145	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.146	n232-134-146	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.147	n232-134-147	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.194	n232-134-194	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.195	n232-134-195	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.196	n232-134-196	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.213	n232-134-213	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2
10.232.134.216	n232-134-216	IC_ETX	server	CentOS Linux release 7.9.2009 (Core)	x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz	128	2

左侧侧边栏展示 group -> sub_group/sub_host 之间的层级关系。

右侧表格展示所有设备的静态信息，包括操作系统、cpu、mem 信息及集群归属信息。

说明：

- 左侧侧边栏内容双击，可以在右侧表格中展示筛选过的内容。
- 右侧表格标题栏右击，可以指定要展示的列，以解决列数过多部分信息无法同时显示的问题。
- 右侧表格中的设备如果 ssh 登陆失败，或者相关信息采集命令执行失败，则背景色为红色，可以借此判断服务器的可登录状态。

4.3.5 STAT 页

STAT 页读取和展示<db_path>/host_stat/host_stat.json 内容，即资产的动态（负载）信息。

	host_ip	host_name	groups	up_days	users	tasks	r1m	r5m	r15m	cpu_thread	cpu_id	cpu_wa	mem_total	mem_avail	swap_total	swap_used	tmp_total	tmp_avail
1	10.232.134.66	n232-134-066	IC_ETX	153	613	4396	0.85	2.27	2.53	128	93.4	0	1007	779	127	27	1758	1755
2	10.232.135.12	n232-135-012	IC_ETX	131	728	4529	2.68	1.4	1.27	128	97.5	0	1007	871	127	17	1620	1620
3	10.232.134.69	n232-134-069	IC_ETX	165	601	4492	5.31	5.64	5.78	128	96	0.6	1007	928	127	4	1620	1620
4	10.232.134.142	n232-134-142	IC_ETX	165	872	5350	5.63	5.86	7.62	128	90.3	0	1007	870	127	36	1728	1727
5	10.232.137.137	n232-137-137	EMU_S1	170	1	689	0	0.01	0.05	64	99.9	0	1007	993	127	0	1620	1620
6	10.232.134.217	n232-134-217	IC_ETX	165	414	3534	0.87	0.87	1.21	128	99.1	0	1007	889	127	3	1620	1620
7	10.232.137.13	n232-137-013	EMU_ETX	165	451	5419	5.32	7.22	10.5	128	97.8	0	1007	937	127	3	1620	1620
8	10.232.134.216	n232-134-216	IC_ETX	165	643	4873	2.6	2.02	2.01	128	96.9	0	1007	902	127	6	1620	1620
9	10.232.134.194	n232-134-194	IC_ETX	134	772	4998	1.15	0.96	1.46	128	96.3	0	1007	913	127	4	1620	1620
10	10.232.158.51	n232-158-051	IC_DATATRANS	219	1	2343	1.07	0.94	0.89	16	89.3	0.1	31	28	5	3	0	0
11	10.212.704.142	n212-704-142	FPGA_ETX FPGA_CMP	29	173	2926	2.57	2.1	2.5	96	94.8	0	754	716	255	0	2000	2000
12	10.232.134.73	n232-134-073	IC_ETX	165	672	4978	1.52	2.24	2.97	128	98.9	0	1007	919	127	7	1620	1620
13	10.232.134.67	n232-134-067	IC_ETX	153	586	4668	1.97	4.48	4.09	128	94.6	0	1007	922	127	2	447067	447066
14	10.232.134.130	n232-134-130	IC_ETX	146	754	4895	2.48	2.99	3.52	128	95	0	1007	897	127	3	1620	1620
15	10.232.134.195	n232-134-195	IC_ETX	131	602	4800	2.32	1.14	0.88	128	97.7	0	1007	853	127	34	1620	1620
16	10.232.135.66	n232-135-066	IC_ETX	104	1	1267	0.05	0.12	0.08	128	100	0	1007	995	127	0	1620	1609
17	10.232.136.16	n232-136-016	EMU_ETX EMU_CMP	165	450	5725	3.82	6.12	9.5	128	95.9	0	1007	918	127	2	1620	1620
18	10.232.134.78	n232-134-078	IC_ETX	165	865	4994	2.49	2.41	3.76	128	97.8	0	1007	885	127	11	1620	1560
19	10.232.134.143	n232-134-143	IC_ETX	165	992	6325	2.22	3.36	4.96	128	96.9	0	1007	882	127	37	1728	1728
20	10.232.134.131	n232-134-131	IC_ETX	131	793	4977	4.47	2.4	1.94	128	94.8	0	1007	914	127	3	1620	1616
21	10.232.137.138	n232-137-138	EMU_S1	170	3	730	1	1.01	1.05	64	98.3	0	1007	993	127	0	1620	1620
22	10.232.137.140	n232-137-140	EMU_S1	41	1	687	0	0.01	0.05	64	99.0	0	1007	994	127	0	1620	1620
23	10.232.134.146	n232-134-146	IC_ETX	165	763	4505	1.8	1.93	2.64	128	97.5	0	1007	831	127	53	1620	1620
24	10.212.204.143	n212-204-143	FPGA_CMP	165	1	549	6.94	6.96	6.89	48	85.2	0	754	742	255	0	1932	1932

表格展示所有设备的动态（负载）信息，包括启动天数、用户数、任务数、cpu 用量、mem 用量、swap 用量、tmp 用量等。

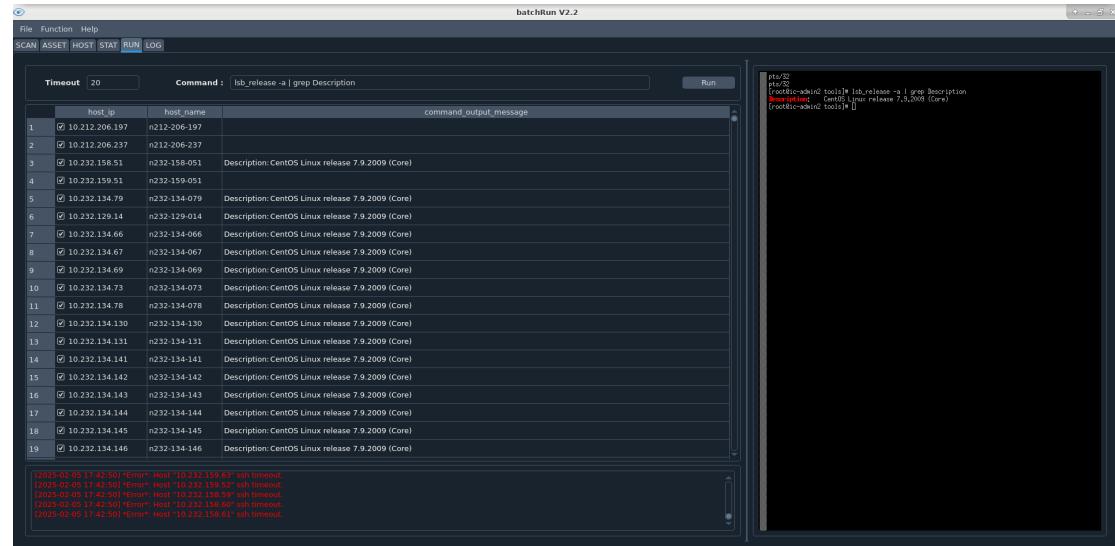
说明：

- 默认展示当前最后一次采样数据，也可以选择其它采样数据。（不会自动刷新）
- 表格中的设备资源用量异常，背景色会被标记为橘色或者红色。
- Select 项支持 python 的判断语法，标题即为变量项，每个单元格的内容即为变量值，筛选写法为 **r1m >= cpu_thread and cpu_id < 90**。

4.3.6 RUN 页

RUN 页用来在选中的设备上统一推送指定命令 (Command)，强制为并行推送。

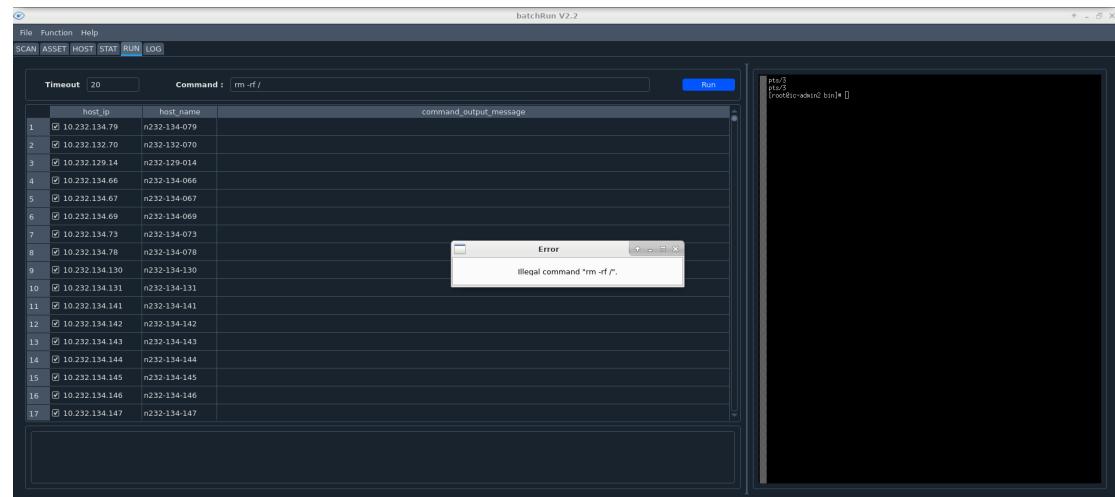
需要说明的是，如果 host.list 中存在名为“RUN”的组，则 RUN 页默认导入“RUN”组中的服务器列表，否则导入全量的服务器列表。



左侧为任务推送区域。

右侧为内嵌的 xterm 工具，可以将需要推送的命令现在本地测试。

如果执行的是 config/config.py 中 illegal_command_list 中定义的违法命令，batch_run 会拒绝执行。



说明：

- 默认的 timeout 为 20s，针对 runtime 较长的命令，需要手工调长这个值。

- 每个 host_ip 都可以单独选中或者取消，只有选中的 host_ip 上才会尝试执行指定命令。
 - 针对 ssh timeout, 或者 hostname 跟 host.list 中备注不一致的情况，会在下方日志框中以 Warning/Error 信息的形式提示。

4.3.7 LOG 页

LOG 页用来检索和查询 batchRun 的操作记录。

左侧表格展示筛选结果。

右侧文本区用来展示具体操作记录的 Log 内容。

说明:

- 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。
 - 除了可以按照用户、日期来做检索，还可以按照 Info 搜索操作记录。

五、辅助工具

batchRun 的辅助工具位于 tools 目录下，下面介绍部分常用工具。

5.1 save_password

工具 save_password 用于将当前用户密码加密保存，供 batchRun 远程 ssh 登陆服务器的时候使用。

```
[root@ic-admin2 batchRun]# tools/save_password --help
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名，默认为“default”，即对所有机器生效。

--password: 指定用户明文密码。

下面是一个例子。

```
[root@ic-admin2 batchRun]# tools/save_password --password *** --
host ic-monitor01
[root@ic-admin2 batchRun]# cat db/password/root
root  ic-monitor01  1739691052  3d87074f29153815ba042935c8bca64c
698371509b61cccd1a2bc9282fccaa4b2
```

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。

`save_password` 适用于某台服务器允许 ssh 登陆但是没有配置 ssh 免密的情况。

5.2 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 --help
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 : IC_ETX
10.232.135.142 n232-135-142

# GROUP : FPGA_ETX
10.212.204.131 n212-204-131

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

# GROUP : FPGA_CMP
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。

```
[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。

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

[FPGA_ETX]
10.212.204.131 ssh_host=n212-204-131

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

[FPGA_CMP]
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]
IC_ETX
IC_CMP

[FPGA]
FPGA_ETX
FPGA_CMP

[RUN]
*_ETX
*_CMP
[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 11 config/host.list
[IC]
IC_ETX
IC_CMP

[FPGA]
FPGA_ETX
FPGA_CMP

[RUN]
*_ETX
```

*_CMP

此处请注意，host.list 中 group 是支持模糊匹配的，比如“*_CMP”，指的就是所有以“_CMP”结尾的组名。

5.3 network_scan

工具 network_scan 用于扫描指定网络，并将扫描结果保存到指定的文件中，用法如下。

```
[root@ic-admin2 batchRun]# tools/network_scan --help
usage: network_scan.py [-h] [-a] [-p PARALLEL] [-d] [-i
INPUT_FILE] [-o OUTPUT_FILE]

options:
  -h, --help            show this help message and exit
  -a, --alive           Only show systems that are alive.
  -p PARALLEL, --parallel PARALLEL
                        Specify the parallelism of the ip
scanning, default is "1000".
  -d, --debug           Enable debug mode.
  -i INPUT_FILE, --input_file INPUT_FILE
                        Read network setting from file, default is
"config/network.list".
  -o OUTPUT_FILE, --output_file OUTPUT_FILE
                        Save scan result into json file, default
is "<db_path>/network_scan/network_scan.json".
```

--alive: 仅展示活动状态的系统。

--parallel: 指定并行度，默认为 1000。

--debug: 开启 debug 模式，会打印扫描过程。

--input_file: 指定网络设置文件，默认为“config/network.list”。

--output_file: 指定输出文件，默认为“<db_path>/network_scan/network_scan.json”。

工具 network_scan 依赖于网络设置文件的定义，默认的网络设置文件位于 config/network.list，下面是一个网络配置的例子。

```
[root@ic-admin2 batchRun]# cat config/network.list
lfrz_a8_402      10.232.129.0/26
lfrz_a8_403      10.232.130.192/26
lfrz_a10_2304    10.232.136.0/26
lfrz_a18_fpga    10.212.204.128/26
vm               10.232.158.0/24
```

基于上面的网络配置做扫描，如下。

```
[root@ic-admin2 batchRun]# tools/network_scan --alive
```

输出文件默认会保存到 db/network_scan/network_scan.json 下，样式如下。

```
[root@ic-admin2 batchRun]# cat db/network_scan/network_scan.json
{
  "lfrz_a8_402": {
    "10.232.129.0/26": {
      "10.232.129.12": {
        "connectivity": true,
        "packet": 3,
        "received": 3,
        "packet_loss": "0%",
        "rtt_avg": 0.095,
        "rtt_unit": "ms"
      },
      ...
    }
  }
}
```

5.4 sample_host_info

工具 sample_host_info 用于采集服务器的 os 和 hardware 信息，并将采集到的信息保存到指定的文件中，用法如下。

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

>>> Clean up /ic/software/tools/batchRun/db/host_info ...
    rm -rf *.info host_list.json

>>> Sampling host list information ...
```

```

/ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 20 --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 20 --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: 41 seconds)

>>> Collecting host information ...
    Host info has been saved to file
"/ic/software/tools/batchRun/db/host_info/host_info.json.2025020
5_190245".

>>> Link current host info file into host_info.json
    ln -s
    /ic/software/tools/batchRun/db/host_info/host_info.json.20250205
_190245 /ic/software/tools/batchRun/db/host_info/host_info.json

```

生成的 host_info.json 文件内容如下。

```
{
  "10.232.135.142": {
    "host_name": [
      "n232-135-142"
    ],
    "groups": [
      "IC_ETX"
    ],
    "server_type": "server",
    "os": "CentOS Linux release 7.9.2009 (Core)",
    "cpu_architecture": "x86_64",
    "cpu_thread": 64,
    "thread_per_core": 1,
    "cpu_model": "Intel(R) Xeon(R) Platinum 8336C CPU @
2.30GHz",
    "cpu_frequency": 3.0,
    "cpu_frequency_unit": "GHz",
    "cpu_l1_cache_size": 32768,
    "cpu_l2_cache_size": 262144,
    "cpu_l3_cache_size": 1048576
  }
}
```

```
        "mem_size": 503,  
        "mem_size_unit": "GB",  
        "swap_size": 127,  
        "swap_size_unit": "GB"  
    },  
    ...  
}
```

5.5 sample_host_queue

工具 `sample_host_queue` 用于采集服务器的集群归属信息，并将采集到的信息保存到指定的文件中，用法如下。

```
[root@ic-admin2 batchRun]# tools/sample_host_queue --help  
usage: sample_host_queue.py [-h] [-H HOSTS [HOSTS ...]] [-G GROUPS  
[GROUPS ...]] [-u USER] [-p PASSWORD] [-P PARALLEL] [-t TIMEOUT]  
[-i HOST_INFO_JSON] [-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,  
default is "0".  
  -t TIMEOUT, --timeout TIMEOUT  
                        Specify the timeout for batch_run, default  
is "20".  
  -i HOST_INFO_JSON, --host_info_json HOST_INFO_JSON  
                        Specify host info file to collect host  
static information, default is  
"/ic/software/tools/batchRun/db/host_info/host_info.json".  
  -o OUTPUT_DIR, --output_dir OUTPUT_DIR  
                        Specify host info output directory,  
default is "/ic/software/tools/batchRun/db/host_queue".
```

--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。

--host_info_json: 指定#5.4 中 sample_host_info 生成的 host_info.json 文件, 用于从其中读取必要信息。

--output_dir: 指定输出路径, 默认为“<db_path>/host_queue”。

下面是一个例子。

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

>>> Cleaning up files with suffixs "lsid|queues|hosts|bmgroup" ...

>>> Sampling host lsid information ...
    /ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 20 --command "source /etc/profile; lsid" --
output_message_level 1 --output_file
/ic/software/tools/batchRun/db/host_queue/HOST.lsid

Total 12 hosts. (Runtime: 41 seconds)

>>> Collecting host lsid information ...

>>> Cleaning up files with suffixs "lsid" ...

>>> Sampling cluster host information ...
    * Sampling queues information for scheduler "LSF" cluster
"IC_CLUSTER" on host "10.232.135.142" ...
    /ic/software/tools/batchRun/bin/batch_run --hosts
10.232.135.142 --parallel 1 --timeout 20 --command "source
/etc/profile; bqueues -l" --output_message_level 1 --output_file
/ic/software/tools/batchRun/db/host_queue/LSF_IC_CLUSTER.bqueues
```

```
Total 1 hosts. (Runtime: 3 seconds)
...
>>> Collecting host scheduler/cluster/queue information ...
*Warning*: Queue "normal" is not well configured, all of the hosts
are on the same queue.
*Warning*: Queue "normal" is not well configured, all of the hosts
are on the same queue.

>>> Cleaning up files with suffixs "queues|hosts|bmgroup" ...

>>> Write host scheduler/cluster/queue relationship file ...
    Host queue info has been saved to file
"/ic/software/tools/batchRun/db/host_queue/host_queue.json.2025020
5_191205".

>>> Link current host queue file into host_queue.json
    ln -s
/ic/software/tools/batchRun/db/host_queue/host_queue.json.20250205
_191205 /ic/software/tools/batchRun/db/host_queue/host_queue.json
```

生成的 host_queue.json 文件内容如下。

```
{
  "10.232.135.142": {
    "scheduler": "LSF_10.1.0.12",
    "cluster": "IC_CLUSTER",
    "queues": [
      "orca",
      "orca_ana"
    ],
  },
  ...
}
```

5.6 sample_host_stat

工具 sample_host_stat 用于采集服务器的 cpu_load/mem_usage/swap_usage 等负载信息，并将采集到的信息保存到指定的文件中，用法如下。

```
[root@ic-admin2 batchRun]# tools/sample_host_stat --help
usage: sample_host_stat.py [-h] [-H HOSTS [HOSTS ...]] [-G GROUPS]
```

```
[GROUPS ...]] [-u USER] [-p PASSWORD] [-P PARALLEL] [-t TIMEOUT]
[-i HOST_INFO_JSON] [-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,
default is "0".
-t TIMEOUT, --timeout TIMEOUT
                  Specify the timeout for batch_run, default
is "20".
-i HOST_INFO_JSON, --host_info_json HOST_INFO_JSON
                  Specify host info file to collect host
static information, default is
"/ic/software/tools/batchRun/db/host_info/host_info.json".
-o OUTPUT_DIR, --output_dir OUTPUT_DIR
                  Specify host info output directory,
default is "/ic/software/tools/batchRun/db/host_stat".
```

--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。

--host_info_json: 指定#5.4 中 sample_host_info 生成的 host_info.json 文件, 用于从其中读取必要信息。

--output_dir: 指定输出路径, 默认为“<db_path>/host_stat”。

下面是一个例子。

```
[root@ic-admin2 batchRun]# tools/sample_host_stat --groups ALL
>>> Sampling host stat information ...
    /ic/software/tools/batchRun/bin/batch_run --groups ALL --
parallel 0 --timeout 20 --command "df --block-size GB /tmp; free -
g; top -b -n 1 | head -n 3" --output_message_level 1 --output_file
/ic/software/tools/batchRun/db/host_stat/20250205/191734/HOST.stat

Total 12 hosts. (Runtime: 41 seconds)

>>> Collecting host stat ...
    Host info has been saved to file
"/ic/software/tools/batchRun/db/host_stat/20250205/191734/host_st
at.json".

>>> Link host_stat.json to top directory ...
    Link
"/ic/software/tools/batchRun/db/host_stat/20250205/191734/host_st
at.json" into
"/ic/software/tools/batchRun/db/host_stat/host_stat.json

>>> Cleaning up <HOST>.stat files ...
```

生成的 host_stat.json 文件内容如下。

```
{
    "10.232.135.142": {
        "host_name": [
            "n232-135-142"
        ],
        "groups": [
            "IC_ETX"
        ],
        "up_days": 165,
        "users": 1,
        "tasks": 722,
        "r1m": 0.01,
        "r5m": 0.04,
        "r15m": 0.05,
        "cpu_thread": 64,
```

```
"cpu_id": 98.3,  
"cpu_wa": 0.0,  
"mem_total": 503,  
"mem_used": 6,  
"mem_free": 482,  
"mem_shared": 0,  
"mem_buff": 13,  
"mem_avail": 494,  
"swap_total": 127,  
"swap_used": 3,  
"swap_free": 124,  
"tmp_total": 1979,  
"tmp_used": 1,  
"tmp_avail": 1979  
},  
...  
}
```

5.7 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
2023.07	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。
2024.08	1.2	增加服务器信息采样功能。
2024.10	2.0	增加 GUI 界面，包含 GROUP/HOST/RUN/LOG 四个界面。
2025.01	2.1	GUI 界面增加 SCAN 页，用于显示网络扫描结果。 GUI 界面增加 STAT 页，用于显示服务器动态信息。 GROUP 页增加 scheduler/cluster/queue 信息。 解决 crontab 环境中 batchRun 无法完成身份认证的 bug。
2025.02	2.2	GUI 界面合并 GROUP 和 HOST 页。 GUI 界面增加 ASSET 页，用于显示资产信息。 在 host.list 和命令行参数中均支持 host/group 的模糊匹配。 RUN 页默认读取“RUN”组服务器列表以取代全量服务器列表。 增加 illegal command 检查功能，违法命令将拒绝执行。

附 2. config/host.list 编写规则

config/host.list 自带的格式说明如下。

```
#### 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
#####
```

- “**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>
```

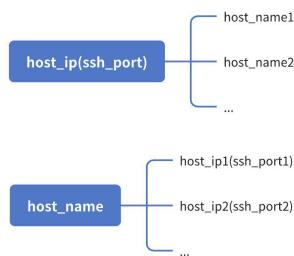
host_ip 是必选项，是 host 的确定标识。

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。

附 3. batchRun 检查功用

batchRun 除了可以用来做批量部署和资产管理，还因其强大的信息采集能力和信息展示/筛选能力，可以用于各种检查来发现设备问题，下面是几个典型的方向。

【发现非法设备】

如下图所示，SCAN 页中部分扫描到的设备并未在 host.list 中记录，因而被标记

为红色背景。

Zone	zone	network	host_ip	host_name	groups	packet	received	packet_loss	rtt_avg
Zone - Network - Ip	1 huailai_storage	10.151.171.192/26	10.151.171.194			3	3	0%	2.515 ms
> lfrz_a18_ftp	2 huailai_storage	10.151.171.192/26	10.151.171.195	fss2720....	OTHERS_STORAGE_fss2720	3	3	0%	2.361 ms
> lfrz_a18_ftps	3 huailai_storage	10.151.171.192/26	10.151.171.196	fss2720....	OTHERS_STORAGE_fss2720	3	3	0%	3.051 ms
> lfrz_a18_ftpsa	4 huailai_storage	10.151.171.192/26	10.151.171.197	fss2720....	OTHERS_STORAGE_fss2720	3	3	0%	2.392 ms
> lfrz_a18_ftpsb	5 huailai_storage	10.151.171.192/26	10.151.171.198	fss2720....	OTHERS_STORAGE_fss2720	3	3	0%	2.423 ms
> lfrz_a18_ftpsc	6 huailai_storage	10.151.171.192/26	10.151.171.199	hffilter01-if01	OTHERS_STORAGE_fss2720	3	3	0%	2.464 ms
> lfrz_a18_ftpsc	7 huailai_storage	10.151.171.192/26	10.151.171.200	hffilter01-if02	OTHERS_STORAGE_fss2720	3	3	0%	2.504 ms
> lfrz_a18_ftpsc	8 huailai_storage	10.151.171.192/26	10.151.171.201	fss2720-node01-sp	OTHERS_STORAGE_fss2720	3	3	0%	3.784 ms
> lfrz_a18_ftpsc	9 huailai_storage	10.151.171.192/26	10.151.171.202	fss2720-node02-sp	OTHERS_STORAGE_fss2720	3	3	0%	4.833 ms
> lfrz_a18_ftpsc	10 huailai_storage	10.151.171.192/26	10.151.171.203	fss2720-cluster_mngt	OTHERS_STORAGE_fss2720	3	3	0%	2.413 ms
> lfrz_a18_ftpsc	11 huailai_storage	10.151.171.192/26	10.151.171.204			3	3	0%	2.342 ms
> lfrz_a18_ftpsc	12 huailai_storage	10.151.171.192/26	10.151.171.205	fss2720-x3_d4x2_4601	OTHERS_STORAGE_fss2720	3	3	0%	2.572 ms
> lfrz_a18_ftpsc	13 huailai_storage	10.151.171.192/26	10.151.171.207			3	3	0%	2.405 ms
> lfrz_a18_ftpsc	14 huailai_storage	10.151.171.192/26	10.151.171.208			3	3	0%	3.258 ms
> lfrz_a18_ftpsc	15 huailai_storage	10.151.171.192/26	10.151.171.209			3	3	0%	2.356 ms
> lfrz_a18_ftpsc	16 huailai_storage	10.151.171.192/26	10.151.171.210			3	3	0%	2.367 ms
> lfrz_a18_ftpsc	17 huailai_storage	10.151.171.192/26	10.151.171.211			3	3	0%	3.14 ms
> lfrz_a18_ftpsc	18 huailai_storage	10.151.171.192/26	10.151.171.212			3	3	0%	3.199 ms
> lfrz_a18_ftpsc	19 huailai_storage	10.151.171.192/26	10.151.171.213			3	3	0%	2.212 ms
> lfrz_a18_ftpsc	20 huailai_storage	10.151.171.192/26	10.151.171.215	fss2820....	OTHERS_STORAGE_fss2820	3	3	0%	2.672 ms
> lfrz_a18_ftpsc	21 huailai_storage	10.151.171.192/26	10.151.171.216	fss2820....	OTHERS_STORAGE_fss2820	3	3	0%	2.324 ms
> lfrz_a18_ftpsc	22 huailai_storage	10.151.171.192/26	10.151.171.217	fss2820-node03-sp	OTHERS_STORAGE_fss2820	3	3	0%	2.711 ms
> lfrz_a18_ftpsc	23 huailai_storage	10.151.171.192/26	10.151.171.218	fss2820-node04-sp	OTHERS_STORAGE_fss2820	3	3	0%	2.631 ms
> lfrz_a18_ftpsc	24 huailai_storage	10.151.171.192/26	10.151.171.219	hffilter01-if03	OTHERS_STORAGE_fss2820	3	3	0%	2.297 ms

一般有两种原因：

- 设备已知，host.list 中未记录。（补充即可）
- 设备未知，不受当前环境 IT 管理员管辖。（存在网络安全漏洞，需要尽快移除或者网络屏蔽）

【发现未配置资产】

如下图所示，ASSET 页中部分扫描到的设备并未在 host.list 中记录，因而被标记为红色背景。

host_ip	host_name	groups	psm	isic	rack	tor	nettype	kind	package	manufacturer	asset_no
1 10.206.112.24				LFRZ_A18_102-09-			25equal	物理机		20210301-is...	
2 10.211.61.94				LFRZ_A18_500-10-05			25equal	物理机		2021-ic-srv-011143	
3 10.212.174.130	hffilter01-if01	OTHERS_STORAGE_...		LFRZ_A18_600-11-04			25equal	物理机		2022-netapp...	
4 10.212.174.132	ic-zfs05	OTHERS_STORAGE_...		LFRZ_A18_606-15-03			25equal	物理机		2022-ic-srv-011150	
5 10.212.174.135	ic-zfs06	OTHERS_STORAGE_...		LFRZ_A18_606-15-03			25equal	物理机		2022-ic-srv-011159	
6 10.212.174.136	ic-zfs07	OTHERS_STORAGE_...		LFRZ_A18_606-15-03			25equal	物理机		2022-ic-srv-011185	
7 10.212.204.130	r212-204-130	FPGA_SSH		LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
8 10.212.204.131	r212-204-131			LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
9 10.212.204.133	r212-204-133	FPGA_CMP		LFRZ_A18_606-14-06			25equal	物理机		2024-ic-srv-1644295	
10 10.212.204.134	r212-204-134	FPGA_CMP		LFRZ_A18_606-14-07			25equal	物理机		2024-ic-srv-1644296	
11 10.212.204.135	r212-204-135	FPGA_CMP		LFRZ_A18_606-14-07			25equal	物理机		2024-ic-srv-1644297	
12 10.212.204.140	r212-204-140	FPGA_ETX		LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
13 10.212.204.141	r212-204-141	FPGA_SSH		LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
14 10.212.204.142	r212-204-142	FPGA_ETX_FPGA_CMP		LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
15 10.212.204.143	r212-204-143	FPGA_CMP		LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
16 10.212.204.144	r212-204-144	FPGA_ETX_FPGA_CMP		LFRZ_A18_606-14-07			25equal	物理机		20200810-is...	
17 10.212.204.145	r212-204-145	FPGA_CMP		LFRZ_A18_606-14-08			25equal	物理机		20200810-is...	
18 10.212.204.146	r212-204-146	FPGA_CMP		LFRZ_A18_606-14-08			25equal	物理机		20200810-is...	
19 10.212.204.147	r212-204-147	FPGA_CMP		LFRZ_A18_606-14-08			25equal	物理机		20200810-is...	
20 10.212.204.148	r212-204-148	FPGA_CMP		LFRZ_A18_606-14-08			25equal	物理机		20200810-is...	
21 10.212.204.149	r212-204-149	FPGA_CMP		LFRZ_A18_606-14-08			25equal	物理机		20200810-is...	
22 10.212.204.150	r212-204-150	FPGA_CMP		LFRZ_A18_606-14-06			25equal	物理机		20200921-sg...	
23 10.212.204.151	r212-204-151	FPGA_CMP		LFRZ_A18_606-14-06			25equal	物理机		20200921-sg...	
24 10.212.204.152	r212-204-152	FPGA_CMP		LFRZ_A18_606-14-06			25equal	物理机		20200817-is...	

一般有两种原因：

- 资产已移除，公司资产管理系统未更新。(督促公司更新资产信息)
- 资产存在，host.list 中未记录。(补充即可)

【发现错误系统配置】

不同设备需要有不同的初始化原则，可以通过 HOST 页的属性筛选来确认设备静态信息是否符合设备配置要求。

host_ip	host_name	groups	server_type	os	cpu_arch	cpu_model	cpu_thread	thread_per_core	cpu_freq(GHz)	
1 10.212.204.159	c212-204-159	A_CMP_IC_CMP	server	CentOS Linux release 7.9.2009 (Core) x86_64	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz 96		2	3.1	10	
2 10.232.132.201	n232-132-201	IC_CMP	server	CentOS Linux release 7.9.2009 (Core) x86_64	Intel(R) Xeon(R) Platinum 8362 CPU @ 2.80GHz 128		2	3.5	20	

下面是错误系统配置的几个典型例子（当然，不同公司的要求不同）：

- Login 节点关闭超线程。(一般要求打开)
- Computing 节点打开超线程。(一般要求关闭)
- 物理服务器未开启超频。(默认开启超频)
- 物理服务器未配置独立/tmp。
- 物理服务器未设置 swap。

【发现设备状态异常】

如下图所示，STAT 页中筛选出 cpu 过载的设备，被标记为红色背景。

batchRun V2.2

host_ip	host_name	groups	up_days	users	tasks	r1m	r5m	r15m	cpu_thread	cpu_id	cpu_wa	mem_total	mem_avail	swap_total	swap_used	tmp_total	tmp_avail
1 10.232.133.198	n232-133-198	IC_CMP	166	2	1346	236.9	235.65	233	64	0	2015	1355	127	0	1620	1503	
2 10.232.135.143	n232-135-143	IC_CMP	166	1	867	64.91	62.21	57.97	64	35	0	503	381	127	1	1979	1979
3 10.232.131.13	n232-131-013	IC_CMP	166	1	1032	137.07	89.53	66.89	64	0	2015	1833	127	0	1620	1620	
4 10.232.159.47	k8-pc-mon02	IC_VM	537	4	304	9.94	9.74	9.57	4	7.7	0	63	58	9	0	0	0
5 10.232.135.144	n232-135-144	IC_CMP	166	1	917	102.59	100.29	93.6	64	29.5	0	503	398	127	0	1979	1979
6 10.232.134.74	n232-134-074	IC_CMP	151	1	979	106.31	84.31	58.63	64	0	0	2015	1847	127	0	1620	1620

一些典型的异常动态（负载）信息：

- **r1m > cpu_thread** , cpu load 过高，会导致卡顿。
- **cpu_id** 过低或者过高，前者表示过载，后者表示闲置。
- **cpu_wa** 过高，表示 I/O 对 cpu 性能形成了瓶颈。
- **mem_avail** 过低，可用内存不足，可能导致 OOM。
- **swap_used** 过高，会导致程序的运行效率下降。
- **tmp_aval** 过低，可能会导致临时数据写失败。
- **users** 和 **tasks** 均过小，且 **r1m** 接近 0，**cpu_id** 接近 100，说明机器闲置。