

利用线程池技术实现端口扫描(TCP) PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/273/2021\\_2022\\_\\_E5\\_88\\_A9\\_E7\\_94\\_A8\\_E7\\_BA\\_BF\\_E7\\_c104\\_273291.htm](https://www.100test.com/kao_ti2020/273/2021_2022__E5_88_A9_E7_94_A8_E7_BA_BF_E7_c104_273291.htm) 一个简单的利用线程池技术实现端口扫描(TCP)的小程序，新手可以多学习下。关键代码如下:// 扫描本机 private void getLocal() { String ip = getIP(). String portStart = txPortStart1.getText().trim(). String portEnd = txPortEnd1.getText().trim(). if (portStart.length() == 0 || portEnd.length() == 0) return. int s = 0. int e = 0. try { s = Integer.valueOf(portStart). e = Integer.valueOf(portEnd). } catch (Exception ex) { JOptionPane.showMessageDialog(null, "端口输入有误"). return. } // 检查端口是否超出范围 if (!(checkPort(s) amp. checkPort(e))) { JOptionPane.showMessageDialog(null, "端口应该大于0而小于65535"). return. } scann(ip, s, e). runThread(). // 启动线程, 监视扫描是否已完成 }private String getIP() { try { InetAddress addr = InetAddress.getLocalHost(). return addr.getHostAddress().toString(). // ip } catch (Exception e) { JOptionPane.showMessageDialog(null, "获取IP出错!"). } return null. }// 扫描单个IP private void scann(String ip, int startPort, int endPort) { // 将所有按钮设为不可用 setBtnEdit(false). status.setText("请稍候..."). String[] add = {ip, ""}. table.addRow(add). exec = Executors.newFixedThreadPool(10). for (int i = startPort. i exec.execute(new RunSocket(ip, i)). exec.shutdown(). } 100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)