**基于MININET的CPS仿真项目**

# 基本概况

为支撑配电网CPS仿真的重点研发项目，探讨基于MININET的CPS仿真平台的系统原理、实现方式，以及验证基于MININET的CPS仿真平台的运行情况，特制定本方案。

## 主要背景

配电系统转向以传统电力供应、分布式新能源发电、储能、信息通信设备和各类用电负荷为主的物理系统与信息系统深度耦合的配电网信息物理系统（CPS）。

与传统配电网相比，CPS具有如下显著特征：

1）多源异构、网络复杂

负荷的分散性和新能源发电的间歇性，不仅改变了传统功率单向流动的特点，而且分布式电源和储能的种类繁多，各种设备性能各异；随着电力系统的信息化建设，大量传感器、通信设备加入，导致系统的物理网络拓扑结构愈加复杂，规模愈加庞大，整个系统的动态特性具有明显的非线性、不确定、时变、异构性等特征。

2）交互频繁、易受攻击

随着智能电表、分布式电源和储能设备的大量接入，为了完成海量信息采集、传输、优化控制及需求侧响应等功能，电力流与信息流交互频繁，信息感知的深度、广度及密度大幅提高，使得信息安全的问题也愈发突出，而通信协议的开放化、标准化，也给调度自动化、继电保护和安全装置自动化、负荷控制、用户信息采集等领域带来了新的安全隐患。

3）分层调度、防护困难

单一的集中式、分散式控制方法，无法综合考虑投资成本、控制性能、可靠通信等指标，而分布式控制能较好地满足系统要求，但电网的分层式调度结构，使得系统严重依赖通信网络的可靠性和完备性，增加了优化控制难度和安全防御风险。

电网CPS是一个比较新的概念，在配用电信息的智能同步与采集、监控设备智能化与数字化等方面取得了一些初步成果。但是对于如何分析和理解电力系统与信息系统的交互机理，如何建立综合计及能量流、信息流和业务流的安全防御体系，如何充分利用前沿信息与通信技术增强配电系统的安全性、可靠性、自适应和自愈能力等问题，还没有形成系统的理论体系，现有的研究成果也相当初步。

# 项目整体方案

## 2.1. 项目需求

如下图所示，针对本课题的需求，设计了如下架构将RTLAB、实际设备（及物理仿真）、Mininet进行融合。

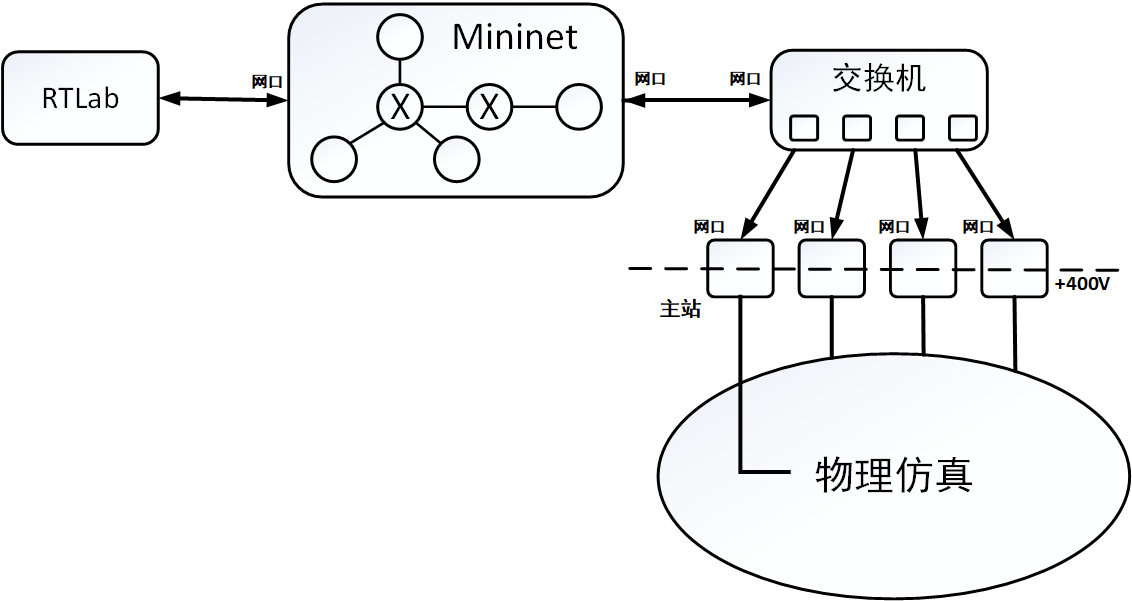


图 1. 部署示意图

如上图所示，Mininet可进行相关的配电网拓扑以及节点的仿真。Mininet与RTLAB之间通过网口连接，Mininet可为RTLAB提供配电网通信状态及配置信息，RTLAB可直接下发命令控制Mininet中的任何一个节点。Mininet可外接交换机，通过交换机与相关实际设备（主站、终端）等连接。主站的网口接入交换机，连接通信网络。主站的非网口（电口）接入物理仿真部分。

基于上述部署环境，虚实结合的实验环境可实现如下功能：

* 配电网通信网络的仿真，包含模拟仿真以及与实际设备结合的仿真
* 各种真实攻击的再现式仿真（可使用真实网络中的攻击发生器进行实际攻击，与真实网络几乎无差别）
  + DDOS攻击：可使用真实的DDOS攻击程序进行攻击。
  + 中间人攻击：可实施典型的中间人攻击。
  + 病毒攻击（模拟病毒传播）:模拟病毒去感染相关机器，这些机器发出一些异常包或者瘫痪等过程。也可以模拟伊朗“震网”病毒的传播模式，病毒在某些机器隐藏、及隐秘传播，只等到相关的关键词出现进行攻击。
  + 网络控制中心侵入的攻击：模拟网络中的控制节点被控制，从而修改网络中的路由策略影响整个网络的连通性。整个网络大多数节点无法按照要求发送到预期的目标。
* 网络动态防御功能的仿真
  + 针对DDOS攻击进行网络的动态配置，降低攻击的危害：在发现可能的攻击流之后，对网络边缘的节点进行控制，从而将相关可疑流导入到模拟的安全设备。
  + 基于网络隔离技术：通过配置虚拟网络，实现网络的分片，从而实现多分区隔离网络。在各个隔离网络的节点，无法访问到其他网络的信息，从而实现安全性。

## 2.2. 项目技术架构

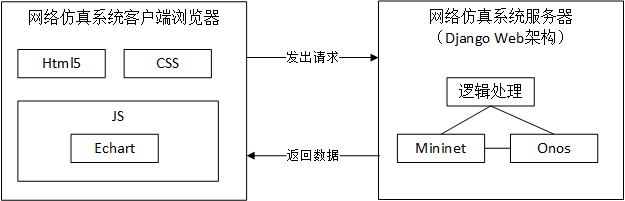


图 2. 系统技术架构图

网络仿真系统由服务器和客户端浏览器组成；系统采用Django的Web架构；客户端浏览器采用HTML5、CSS和Echart等JS技术用于前端显示；服务器Web架构采用Django，后端连接mininet和onos；mininet作为系统中网络的实现，mininet同时连接onos作为远端控制器，后端可操作onos对网络仿真系统数据转发做控制。

# 系统设计

## 3.1. 系统功能结构

### 3.1.1. 系统功能结构框图

以下为网络仿真系统的模块结构图。如图所示，系统由三层架构组成，展示层、处理层、存储层组成。

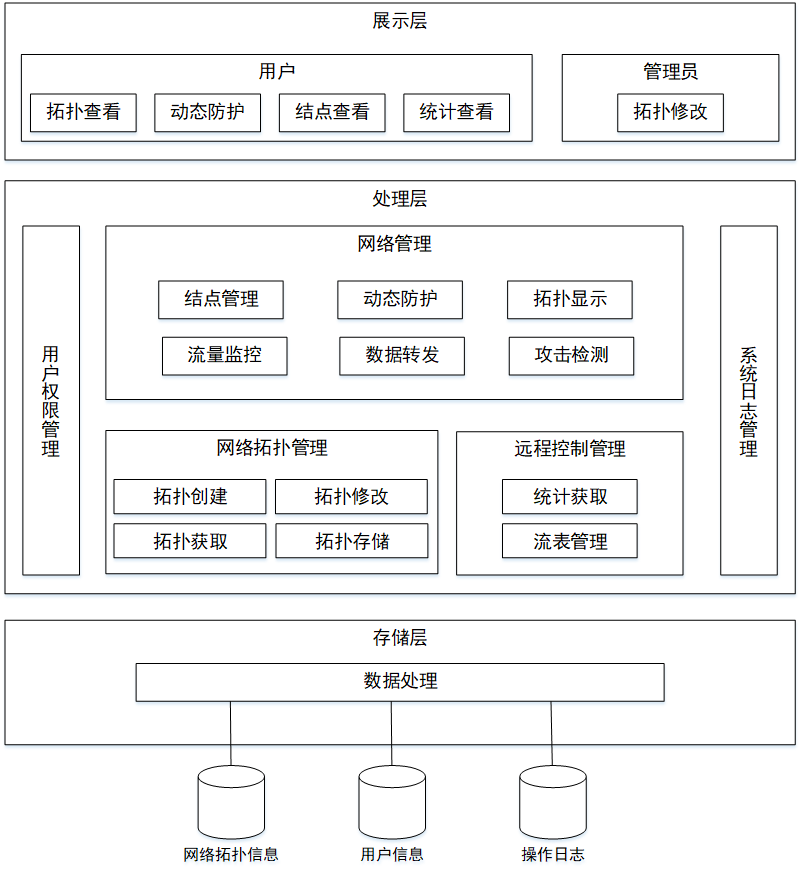


图 3.系统模块架构图

如图所示，最上层的展示层对应系统架构的系统客户端界面，由用户模块和管理员模块组成，主要实现用户对系统操作的界面；中间层的处理层对应系统架构的服务器的处理层，由网络管理模块、网络拓扑管理、远程控制管理模块以及用户权限管理模块和系统日志管理模块组成，主要实现系统对用户需要的任务所进行的具体操作；底层的存储层对应服务器的存储层，由数据处理模块和数据源组成，主要实现系统对数据的可持久化。

### 3.1.2. 系统功能组件定义

下面将对上图中的各个模块的功能和主要实现方法作简要说明。

**数据处理**模块主要实现对持久化数据的一系列操作，网络拓扑信息、用户信息和操作日志的查询，以及针对这些信息的修改。

**网络拓扑管理**模块主要实现仿真系统中网络拓扑的创建以及修改，此外还包含了对拓扑信息的获取和存储功能。网络拓扑管理模块由拓扑创建、拓扑获取、拓扑修改、拓扑存储子模块构成。

**远程控制管理**模块主要实现仿真系统中网络通信中的一系列控制操作和统计信息的获取，具体可获取通信过程中实时流量信息和累计流量信息，以及针对网络中有限制的通信，可通过控制流表的操作实现。远程控制管理模块由统计获取和流表管理子模块构成。

**网络管理**模块主要实现仿真系统中对网络一系列操作，对网络结点信息的管理，网络拓扑的显示，以及针对网络攻击所实施的动态防护功能；此外，还有流量监控功能，控制网络结点数据转发的功能和结点攻击检测的功能。网络管理模块由结点管理、动态防护、拓扑显示、流量监控、数据转发、攻击检测子模块构成。

**用户权限管理**模块主要实现对系统用户信息的操作，该模块仅限管理员操作，可进行用户的增加、删除，修改用户的密码及权限，限制用户是否具有查看拓扑、修改拓扑、查看日志等功能。

**系统日志管理**模块主要实现记录用户的各种操作及发生时间，例如登入、登出、修改拓扑、查询流量信息等。

**拓扑查看**模块主要实现用户查看仿真系统的网络拓扑信息，网络中各类结点个数，结点之间的连接关系通过图的形式展示。

**动态防护**模块主要实现网络拓扑中攻击的识别以及动态防护，并且实时展示防护过程，用户可以选择开启或者关闭入侵检测和网络动态防护以了解动态防护对于网络拓扑系统的保护效果，可以完成对网络拓扑中网络攻击的识别也就是系统入侵检测以及针对于入侵选择合适的防护策略以完成动态防护，并且实时展示防护过程中的信息流或者网络分区变化。

**结点查看**模块主要实现用户查看网络结点详细信息，包括host的类型和ip信息以及资源使用情况，还可查看switch的结点详细信息。

**统计查看**模块主要实现对网络中流量信息的统计功能，实时显示当前各个线路中流量大小，和每条线路中各个时段总流量大小。

**拓扑修改**模块主要实现管理员对网络拓扑中结点的增删和链路的增删操作。

## 3.2. 系统功能实现动态描述

### 3.2.1. 系统与外部交互

从系统整体的层面来看，下图动态地显示了系统与外部对象之间的交互关系：

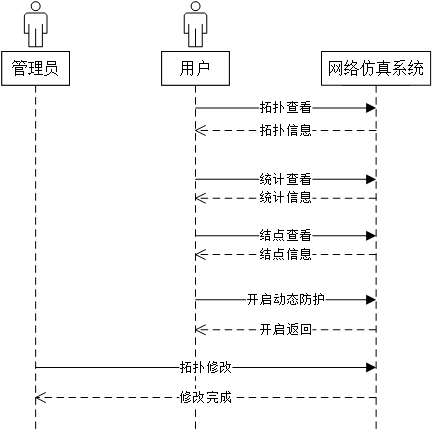


图 4.系统与外部交互

上图隐藏了系统的内部模块划分，重点在于显示出本系统与外部对象的动态消息交互关系，展现了各个消息之间的时序关系。其中，消息交互的序列简述如下：

1. 用户调用系统查看拓扑，系统返回网络拓扑信息，并将拓扑信息依据规定方式以图的形式展示出，并可继续查看拓扑中结点详细信息。
2. 用户调用系统统计查看，系统返回网络中当前时段各个区域流量累计总和，通过折线图形式展示给用户，以及实时返回每条链路的流量信息，在图中以颜色区分。
3. 用户调用系统结点查看，系统依据用户输入的参数，对结点信息进行查询并返回，还可查询结点当前cpu等资源使用情况。
4. 用户调用系统开启动态防护，系统即开启动态防护功能，对网络中的网络攻击进行检测并采取防护措施，返回提示用户功能已开启。
5. 管理员调用系统拓扑修改功能，对网络拓扑的结点和链路进行增删操作，系统修改完成并返回提示用户修改完成。

### 3.2.2. 系统模块间交互

系统模块由网络管理模块、网络拓扑管理模块、远程控制管理模块和数据处理模块组成，网络管理模块由结点管理、动态防护、拓扑显示、流量监控、数据转发、攻击检测子模块构成；网络拓扑管理模块由拓扑创建、拓扑获取、拓扑修改、拓扑存储子模块构成；远程控制管理模块由统计获取和流表管理子模块构成。下面描述这几个模块在各种操作下它们之间的交互：

1. 拓扑创建模块交互

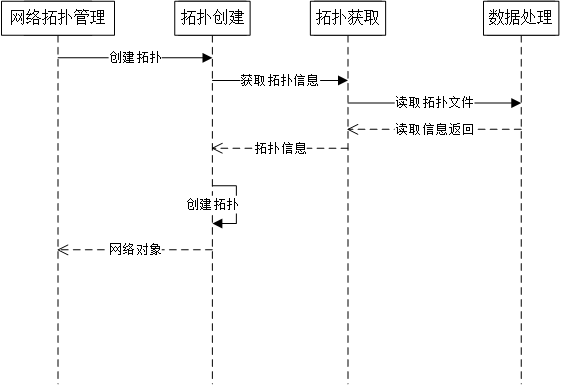


图 5. 拓扑创建模块交互

上图描述了网络拓扑创建时，各系统模块间动态交互的时序；交互时序为：

1. 网络拓扑管理模块调用拓扑创建模块的创建拓扑接口；
2. 拓扑创建模块调用拓扑获取模块的获取拓扑信息；
3. 拓扑获取模块调用数据处理模块读取，读取拓扑文件并返回信息；
4. 拓扑获取模块将返回的拓扑信息返回给拓扑创建模块；
5. 拓扑创建模块获得拓扑信息，调用自身创建拓扑方法；并将创建后的网络对象返回给上级网络管理模块。
6. 拓扑显示模块交互

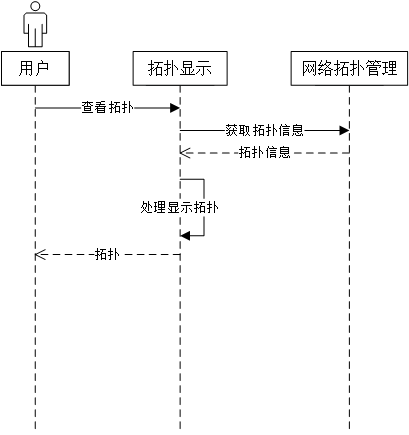


图 6. 拓扑显示模块交互

上图描述了用户调用展示层的拓扑显示模块查看拓扑时，各系统模块间动态交互的时序；交互时序为：

1. 用户调用拓扑显示模块的查看拓扑方法；
2. 拓扑显示模块调用网络管理模块的获取拓扑信息方法，网络管理模块返回拓扑信息；
3. 拓扑显示模块获得拓扑信息，根据拓扑信息处理显示并返回；
4. 用户可在界面上查看到网络拓扑。
5. 结点信息查看模块交互



图 7. 结点信息查看

上图描述了用户利用展示层的结点查看模块查看网络结点信息时，各系统模块间动态交互的时序；交互时序为：

1. 用户调用结点管理的查看结点信息；
2. 结点管理模块调用网络拓扑管理的查看结点信息，网络拓扑管理模块将结点信息返回给上一级；
3. 结点管理模块将结点信息显示返回给用户；
4. 用户调用结点管理模块的查看结点资源信息；
5. 结点管理模块调用网络拓扑管理模块的查看结点资源信息方法；
6. 网络拓扑管理模块调用自身的查看结点资源信息方法，并将返回结果的资源信息返回上级；
7. 结点管理模块获得结点资源信息显示返回给用户。
8. 统计信息查看模块交互

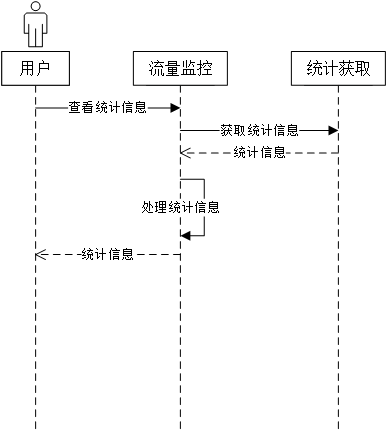


图 8. 统计信息查看模块交互

上图描述了用户利用展示层的统计查看模块查看网络统计信息时，各系统模块间动态交互的时序；交互时序为：

1. 用户调用流量监控模块的查看统计信息；
2. 流量监控模块调用统计获取模块的获取统计信息；
3. 统计获取模块执行方法后将统计信息返回上级；
4. 流量监控模块获取统计信息后，执行处理统计信息的方法，显示并返回给用户查看。
5. 动态防护模块交互

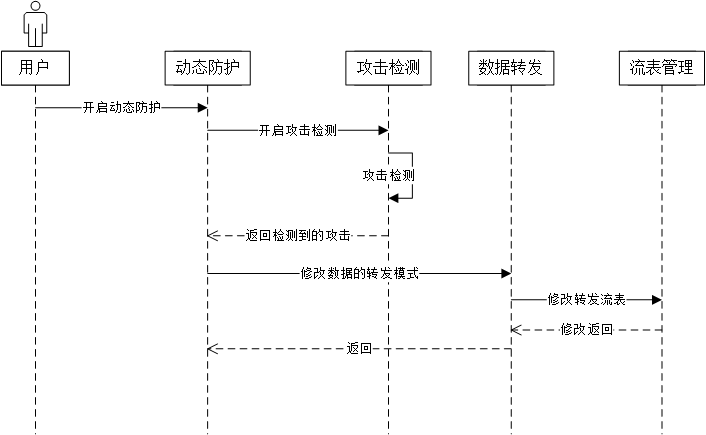


图 9. 动态防护模块交互

上图描述了用户利用展示层的动态防护模块开启动态防护时，各系统模块间动态交互的时序；交互时序为：

1. 用户调用动态防护模块的开启动态防护；
2. 动态防护模块调用攻击检测模块的开启攻击检测方法；
3. 攻击检测模块开启攻击检测的方法，在检测到攻击后返回给动态防护模块；
4. 动态防护模块调用数据转发模块的修改数据的转发模式；
5. 数据转发模块调用流表管理模块的修改转发流表，流表管理模块修改完成后返回；
6. 数据转发模块修改转发模式后返回上级动态防护模块。
7. 拓扑修改模块交互

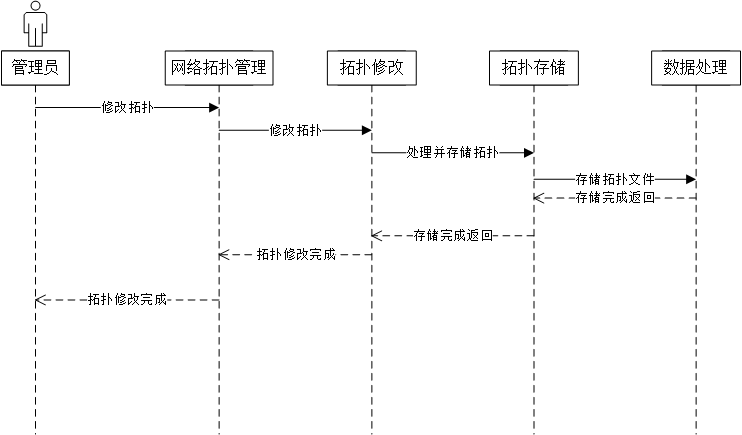


图 10. 拓扑修改交互

上图描述了管理员利用展示层的拓扑修改模块对网络拓扑进行修改时，各系统模块间动态交互的时序；交互时序为：

1. 管理员调用网络拓扑管理模块的修改拓扑；
2. 网络管理模块调用拓扑修改模块的修改拓扑；
3. 拓扑修改模块修改完毕拓扑后调用拓扑存储模块的处理并存储拓扑方法；
4. 拓扑存储模块对拓扑信息依据规定方式处理拓扑信息，调用数据处理模块的存储拓扑文件；
5. 数据处理模块存储完成后返回上级，各级一次返回。

# 通信方案设计

## 4.1. 通信需求与设计思路

项目的通信需求包括：Mininet可为RTLAB提供配电网通信状态及配置信息，RTLAB可直接下发命令控制Mininet中的任何一个节点，Mininet可外接交换机，通过交换机与相关实际设备（主站、终端）等连接，主站的网口接入交换机，连接通信网络。为了满足上述通信需求，我们采取以下通信方案的设计思路：Mininet中的节点和外部真实网络通信，可利用网桥，将Mininet环境所在主机网口与Mininet中的一个交换机端口绑定，从而实现Mininet内任何节点和RTLAB以及主站进行通信。

## 4.2. 部署通信方案

Mininet内部网络需要和外部网络进行通信，所以可以将Mininet所在主机网口和Mininet内部虚拟交换机绑定，如此，可以借助通向外部网络的交换机实现内部结点与外部网络通信。通信方案的部署如下图所示，Mininet所在环境将网口与Mininet交换机switch的端口绑定。Mininet中的一个节点利用eth1通过外部交换机和RTLAB及主站通信。

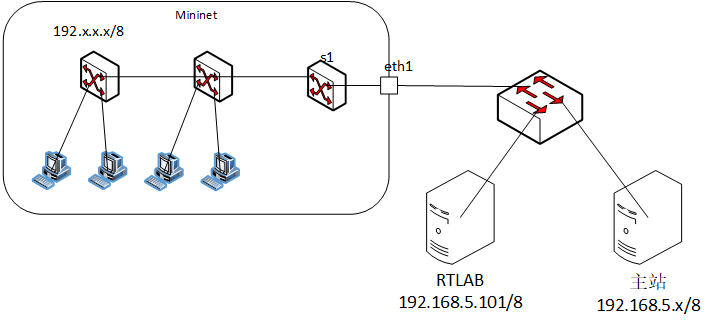


图 11. 网络通信部署图

上述的通信方案中，Mininet网络可以采用不同的子网段，由于Mininet是采用主机的网口和外部网络进行通信，所在主机的IP地址属于同一子网，本方案实验中采用的是192.0.0.0/8该子网，可以支持2^24-2台主机。

## 4.3. 通信模型状态机

通信过程中结点主要有三个状态：未连接状态、连接状态、断开连接状态。通信过程中结点的状态迁移变化如下图所示：

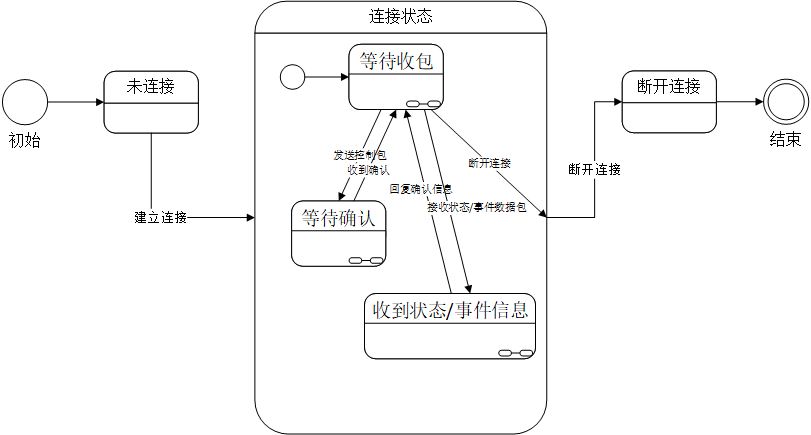


图 12. mininet内部结点状态迁移图

上图描述了通信过程中mininet内部结点的状态迁移变化，初始为未连接状态，经过建立连接的操作变为连接状态，在连接状态经过断开连接的操作变为断开连接的状态即结束状态。在连接状态中，初始子状态为等待收包状态，此状态若发送控制包变为等待确认状态，若接收到状态或时间数据包则变为收到状态/事件信息状态，在等待确认状态时只有收到确认信息方可变为等待收包状态，在收到状态/事件信息状态只有回复确认消息方可变为等待收包状态；在等待收包状态若断开连接则变为断开连接状态，结束通信。

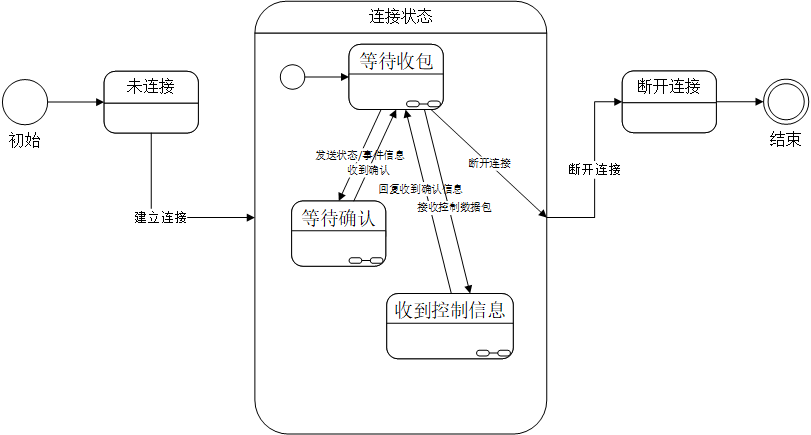


图 13. RTLAB状态迁移图

上图描述了通信过程RTLAB的状态迁移变化，初始为未连接状态，经过建立连接的操作变为连接状态，在连接状态经过断开连接的操作变为断开连接的状态即结束状态。在连接状态中，初始子状态为等待收包状态，此状态若发送状态/事件包变为等待确认状态，若接收到控制数据包则变为收到控制信息状态，在等待确认状态时只有收到确认信息方可变为等待收包状态，在收到控制信息状态只有回复确认消息方可变为等待收包状态；在等待收包状态若断开连接则变为断开连接状态，结束通信。

## 4.4. 通信协议设计

### 4.4.1. 状态数据包

状态数据包负责传输各电力设备当前各类属性的状态信息，包括电压、电流、功率、相别和相角等电力设备的重要状态指标。状态数据包分两类，一类是RTLAB发送给Mininet的，另一类是Mininet向RTLAB的回复。

|  |  |
| --- | --- |
| 单位（bytes） | 说明 |
| 8 | 报头（保留，后续使用） |
| 2 | 包头(，0xAA，0x55) |
| 2 | 状态字 |
| 8 | 状态包序号（浮点数） |
| 8 | 设备种类(浮点数) |
| 8 | 设备编号(浮点数) |
| 8 | 电压 (伏，浮点数) |
| 8 | 电流 (安，浮点数) |
| 8 | 有功功率(千瓦，浮点数) |
| 8 | 无功功率(千瓦，浮点数) |
| 8 | 相角（浮点数） |
| 8 | 相别（浮点数） |
| 8 | 工作状态（浮点数） |
| 8 | 闭合状态（浮点数） |

表 1. RTLAB向Mininet发送状态数据包的格式

状态字的定义如下：

1. 状态数据包表示字段 1为状态包 0无效
2. 状态数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
3. 事件数据包表示字段 1 为事件包 0无效
4. 事件数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
5. 控制数据包表示字段 1 为控制包 0 无效
6. 控制数据包发送方向字段 0 RTLAB发出 1 RTLAB接收

|  |  |
| --- | --- |
| 单位（bytes） | 说明 |
| 8 | 报头（保留，后续使用） |
| 2 | 包头(0xAA，0x55) |
| 2 | 状态字 |
| 8 | 状态包序号（浮点数） |

表 2.Mininet向RTLAB回复状态数据包的格式

状态字的定义如下：

1. 状态数据包表示字段 1为状态包 0无效
2. 状态数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
3. 事件数据包表示字段 1 为事件包 0无效
4. 事件数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
5. 控制数据包表示字段 1 为控制包 0 无效
6. 控制数据包发送方向字段 0 RTLAB发出 1 RTLAB接收

### 4.4.2. 事件数据包

事件数据包负责传输各电力设备当前发生的各类事件信息。事件数据包分两类，一类是RTLAB发送给Mininet的，另一类是Mininet向RTLAB的回复。

|  |  |
| --- | --- |
| 单位（bytes） | 说明 |
| 8 | 报头（保留，后续使用） |
| 2 | 包头(0xAA，0x55) |
| 2 | 状态字 |
| 8 | 事件包序号（浮点数） |
| 8 | 设备种类（浮点数） |
| 8 | 设备编号（浮点数） |
| 8 | 事件种类（浮点数） |
| 8 | 事件内容（浮点数） |

表 3. RTLAB向Mininet发送事件数据包的格式

状态字定义如下：

1. 状态数据包表示字段 1为状态包 0无效
2. 状态数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
3. 事件数据包表示字段 1 为事件包 0无效
4. 事件数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
5. 控制数据包表示字段 1 为控制包 0 无效
6. 控制数据包发送方向字段 0 RTLAB发出 1 RTLAB接收

|  |  |
| --- | --- |
| 单位（bytes） | 说明 |
| 8 | 报头（保留，后续使用） |
| 2 | 包头(0xAA，0x55) |
| 2 | 状态字 |
| 8 | 事件包序号（浮点数） |

表 4. Mininet向RTLAB回复事件数据包的格式

状态字定义如下：

1. 状态数据包表示字段 1为状态包 0无效
2. 状态数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
3. 事件数据包表示字段 1 为事件包 0无效
4. 事件数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
5. 控制数据包表示字段 1 为控制包 0 无效
6. 控制数据包发送方向字段 0 RTLAB发出 1 RTLAB接收

### 4.4.3. 控制数据包

控制数据包负责传输对各电力设备的控制命令信息。控制数据包分两类，一类是RTLAB发送给Mininet的，另一类是Mininet向RTLAB的回复。

|  |  |
| --- | --- |
| 单位（bytes） | 说明 |
| 8 | 报头（保留，后续使用） |
| 2 | 包头(0xAA，0x55) |
| 2 | 状态字 |
| 8 | 控制包序号（浮点数） |
| 8 | 设备种类（浮点数） |
| 8 | 设备编号（浮点数） |
| 8 | 控制类型（浮点数) |
| 8 | 控制命令（浮点数) |

表 5. RTLAB向Mininet发送控制数据包的格式

状态字定义如下：

1. 状态数据包表示字段 1为状态包 0无效
2. 状态数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
3. 事件数据包表示字段 1 为事件包 0无效
4. 事件数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
5. 控制数据包表示字段 1 为控制包 0 无效
6. 控制数据包发送方向字段 0 RTLAB发出 1 RTLAB接收

|  |  |
| --- | --- |
| 单位（bytes） | 说明 |
| 8 | 报头（保留，后续使用） |
| 2 | 包头(0xAA，0x55) |
| 2 | 状态字 |
| 8 | 控制包序号（浮点数） |

表 6. Mininet向RTLAB回复控制数据包的格式

状态字定义如下：

1. 状态数据包表示字段 1为状态包 0无效
2. 状态数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
3. 事件数据包表示字段 1 为事件包 0无效
4. 事件数据包发送方向字段 0 RTLAB发出 1 RTLAB接收
5. 控制数据包表示字段 1 为控制包 0 无效
6. 控制数据包发送方向字段 0 RTLAB发出 1 RTLAB接收

# 动态防护策略

## 5.1. 病毒攻击检测防护流程

### 5.1.1. 病毒攻击检测防护简介

（1）**安全环境下的网络拓扑：**

（PC为绿色代表处于可信环境中，未受攻击）

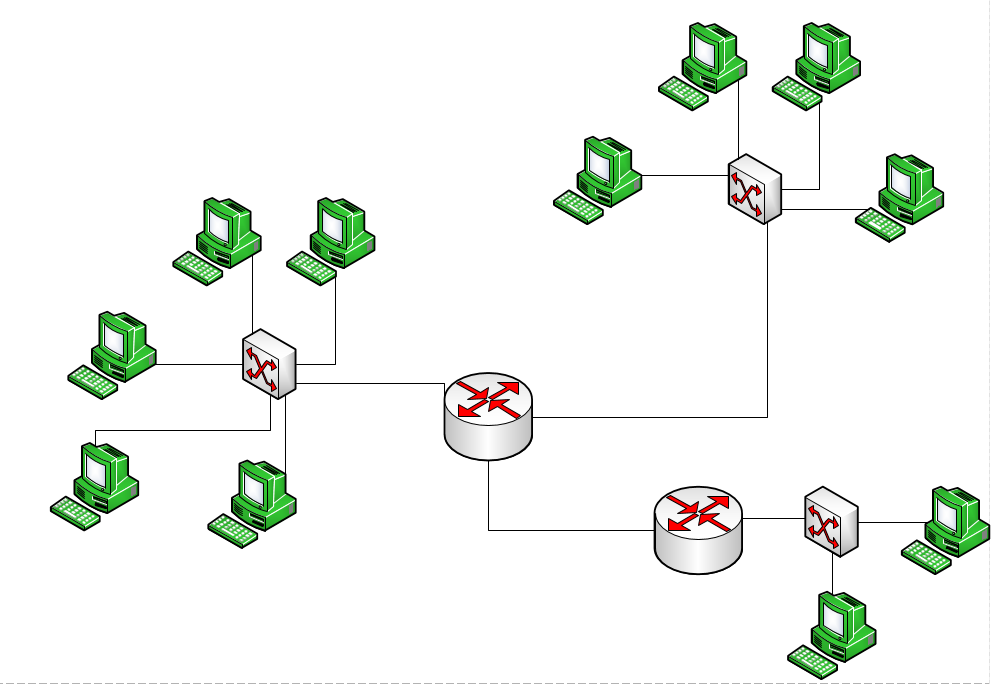


图 11. 安全环境下的网络拓扑

（2）**某台计算机受到外部网络中的病毒程序攻击：**

（PC为红色代表受到病毒程序攻击）

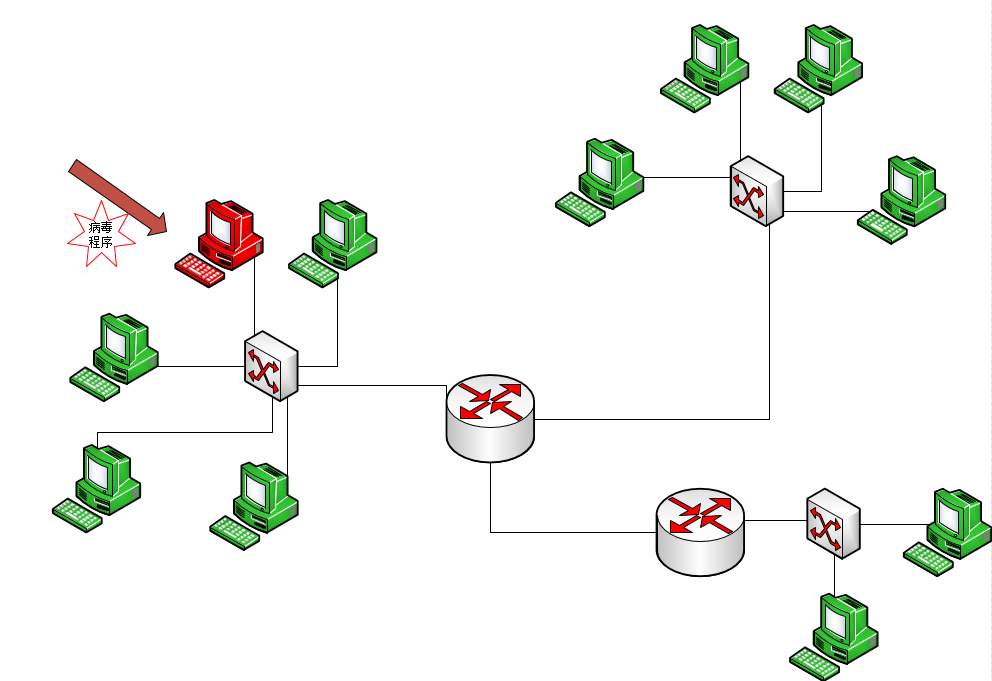
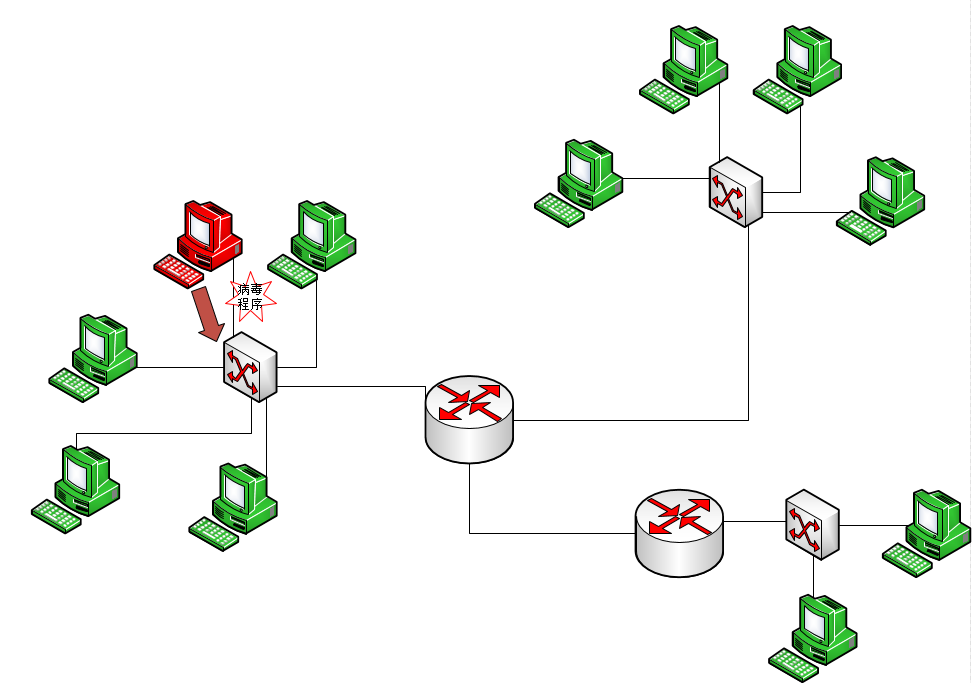


图 12. 某台计算机受到外部网络中的病毒程序攻击

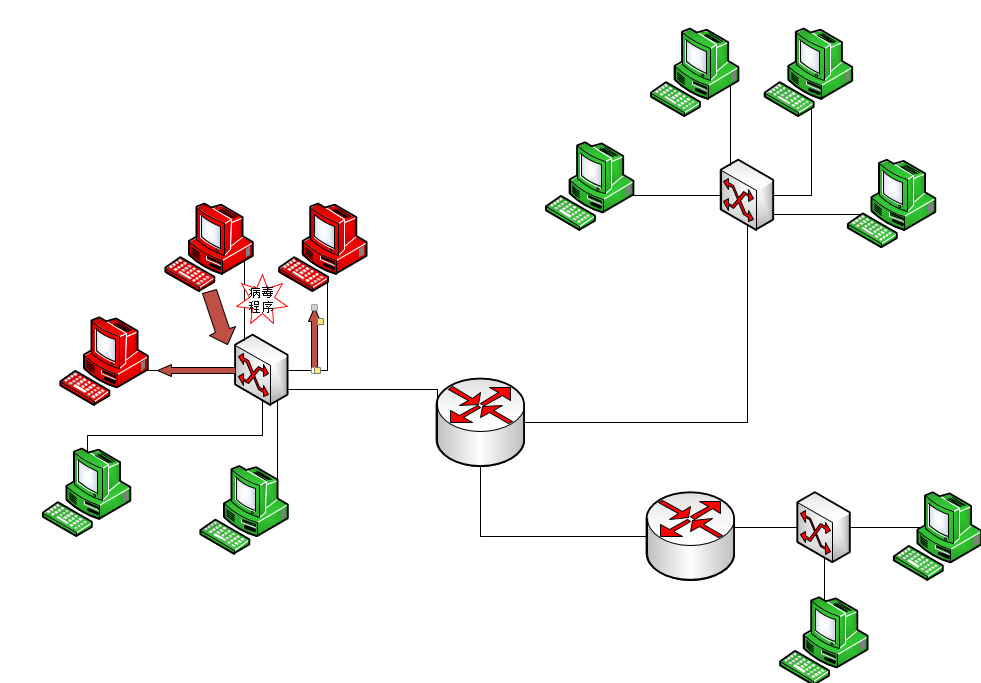
**（3）病毒程序通过局域网或者互联网中的网络设备传播到其他网络环境中：**

受到病毒攻击的主机向外传播病毒，通过网络设备使得更多的主机受到病毒攻击，导致网络中主机大范围出现被病毒程序攻击的情况。

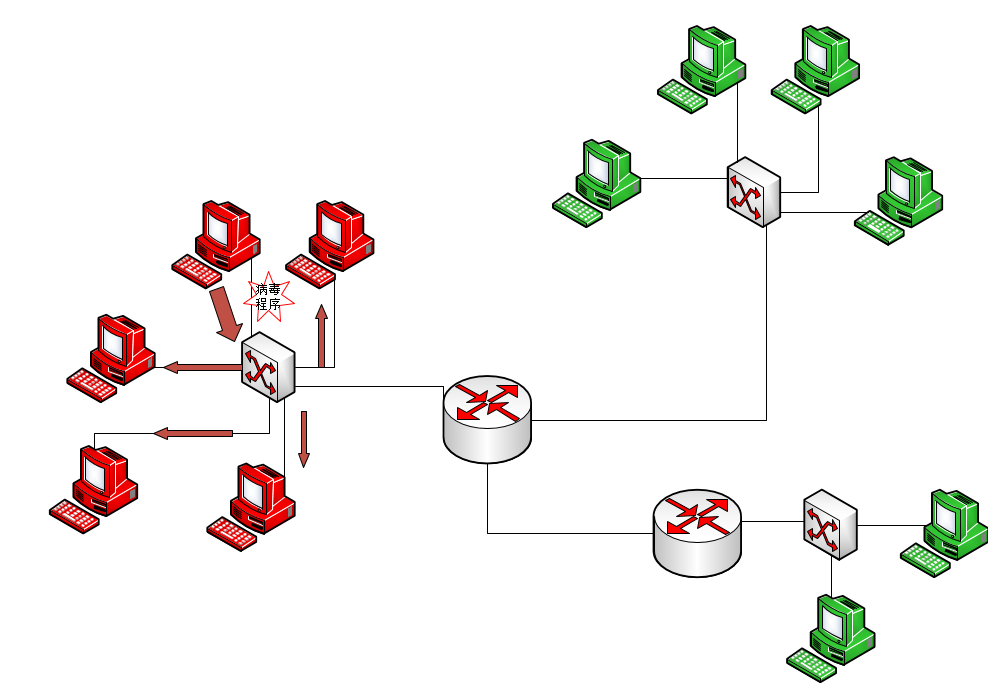
①



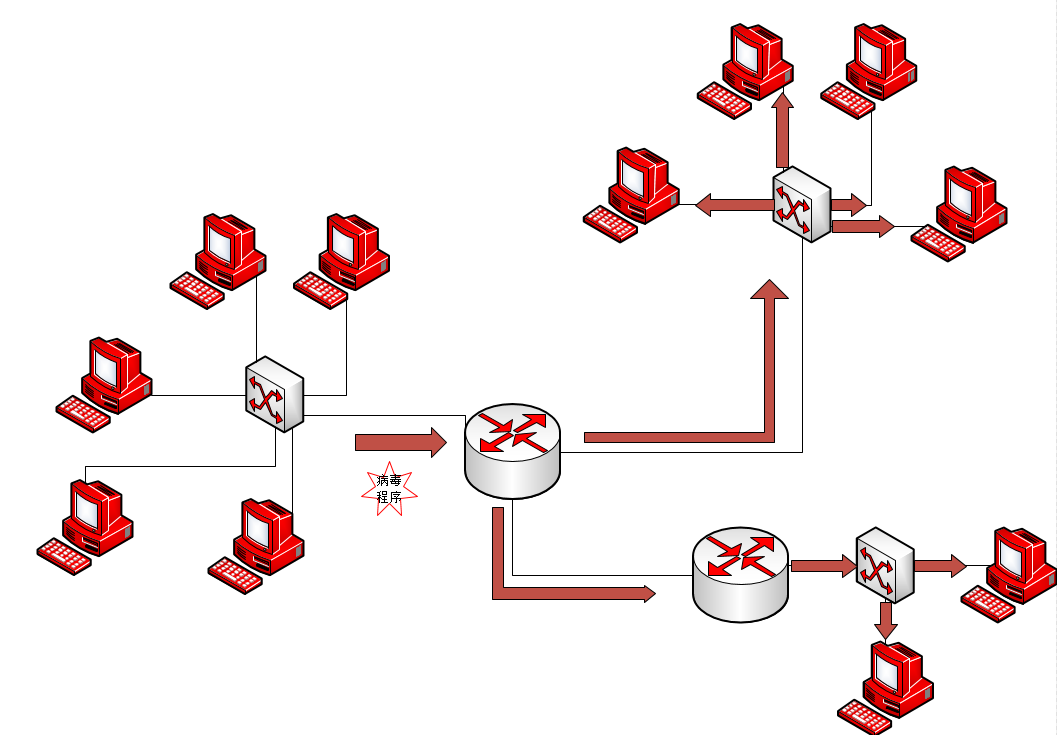
②



③

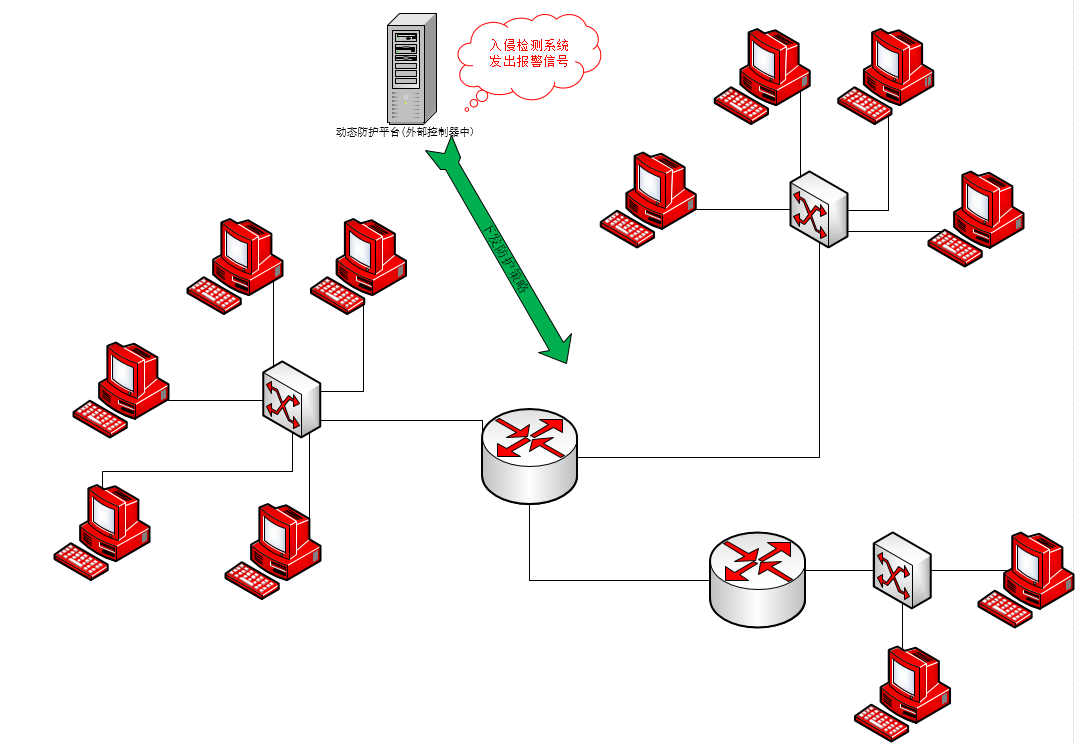


④

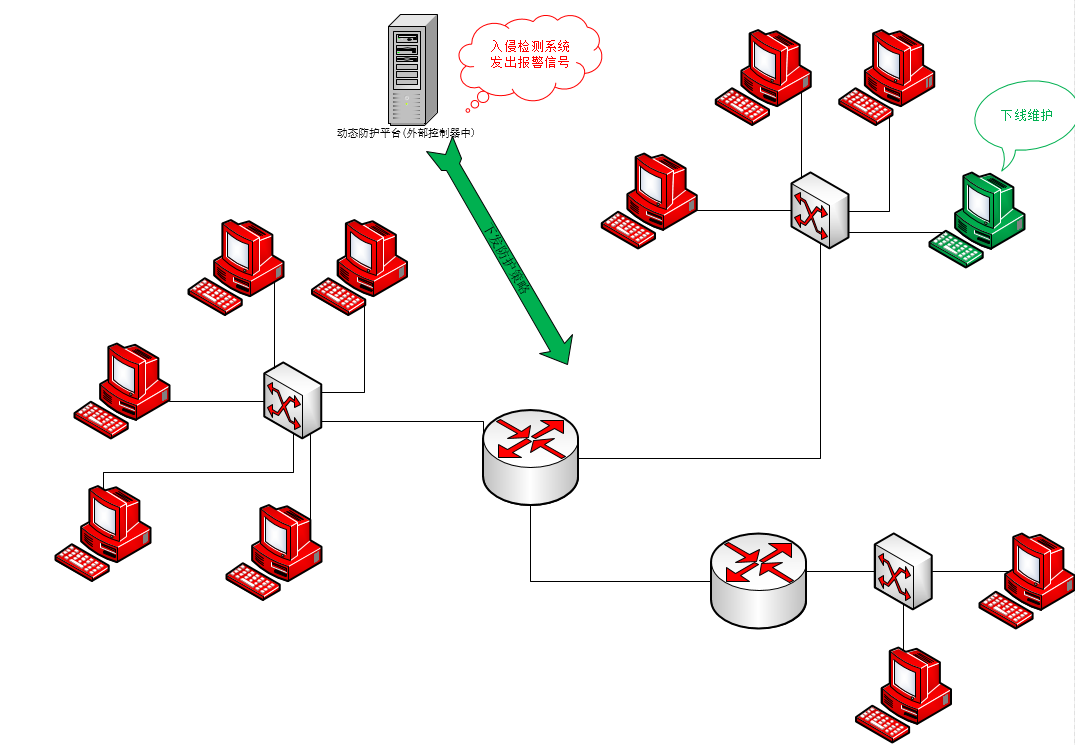


**（4）防护平台执行入侵检测后发出报警信号并向网络下发防护策略进行动态防护**：

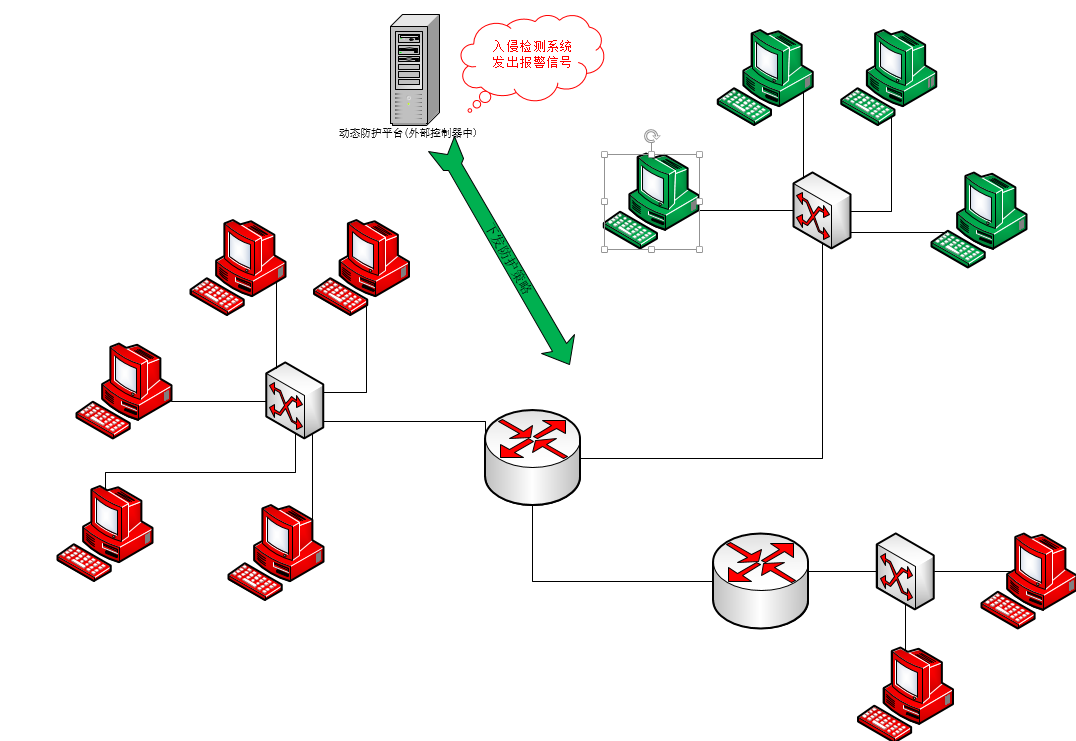
位于网络外部控制器上的防护平台在执行攻击检测算法过程中发现受攻击主机节点，发出警报并报告受攻击节点，随后下发防护策略，如主机下线维护清除病毒程序。待主机维护完成之后，重新上线。所有主机维护完成后，防护平台警报清除。



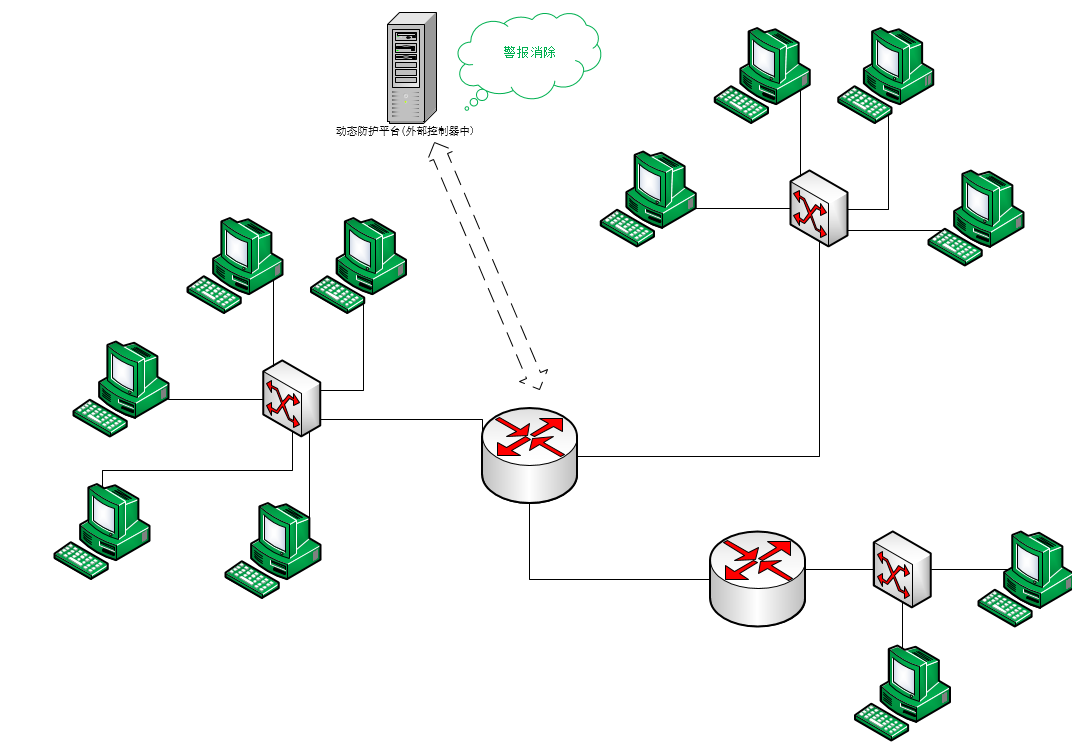
②



③



④



### 5.1.2. 攻击检测流程

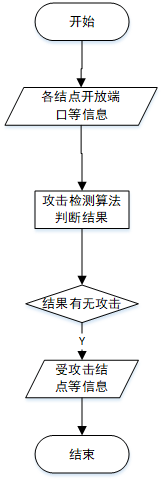


图 14. 病毒攻击检测流程图

上图描述了病毒攻击的检测流程，接收各个结点开放端口等信息，交由攻击检测算法判断结果，攻击检测算法主要关注结点开放的端口信息中不常用的端口，以及这些端口接收和发送的信息，根据这些判断该主机是否做异常行为，判断是否受到病毒攻击，若受到攻击输出受攻击结点的信息。

对于病毒攻击检测的初步构思为基于数据包检测的实时分析：运行一个监听进程，获取发送至主机的所有数据包。我们实现根据攻击定义一个规则集，包含一些预定义的规则，例如在一个连续时间内，收到一定频率以上的SYN或者FIN请求（TCP），我们就可以认为检测到了SYN/FIN扫描攻击，对于其他攻击也类似。通过一个匹配算法，将收到的数据包与规则集中的规则进行匹配，进而确定属于哪种攻击类型，然后输出攻击信息。

### 5.1.3. 攻击防护流程

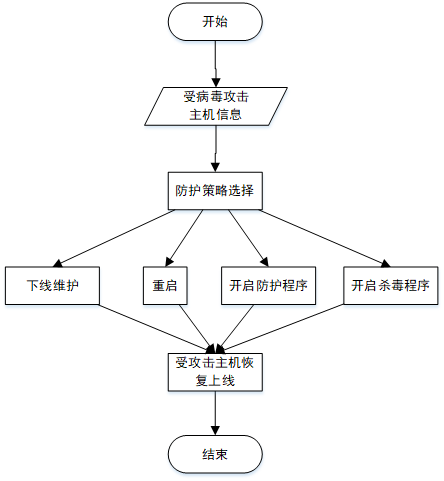


图 15. 病毒攻击防护流程图

上图描述了主机受到病毒攻击后的防护流程，接收受病毒攻击主机的信息，经过防护策略的选择，可对其进行下线维护、重启、开启防护程序和开启杀毒程序等防护措施，待到受攻击主机恢复上线后。

## 5.2. DDos拒绝服务攻击检测防护流程

### 5.2.1. DDoS拒绝服务攻击检测防护简介

**（1）网络中服务器提供某服务处于正常状态中：**

局域网或者通过路由器连接至因特网中的主机可以通过网络连接访问该服务器上的服务。

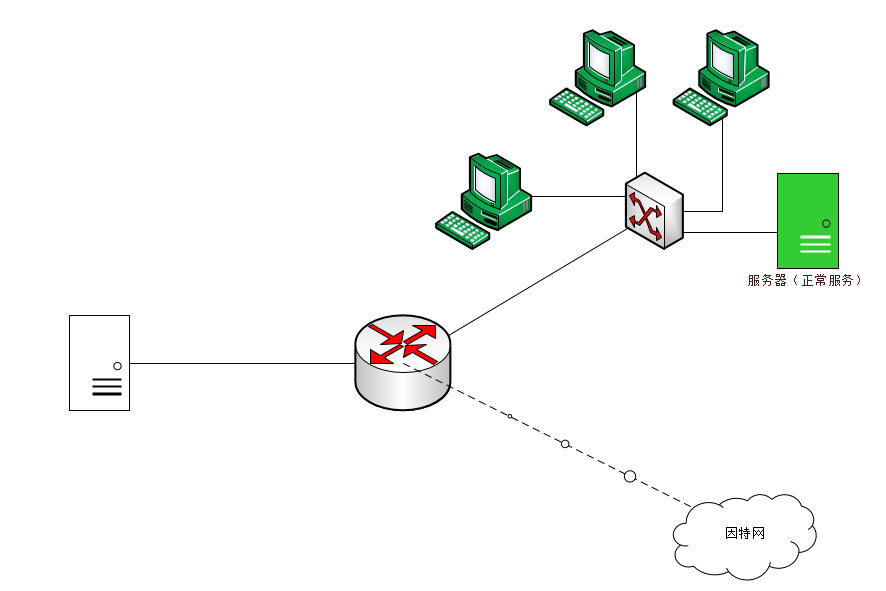


图 13. 网络中服务器提供某服务处于正常状态

**（2）服务器受到因特网中DDos攻击导致服务资源被占用，拒绝服务：**

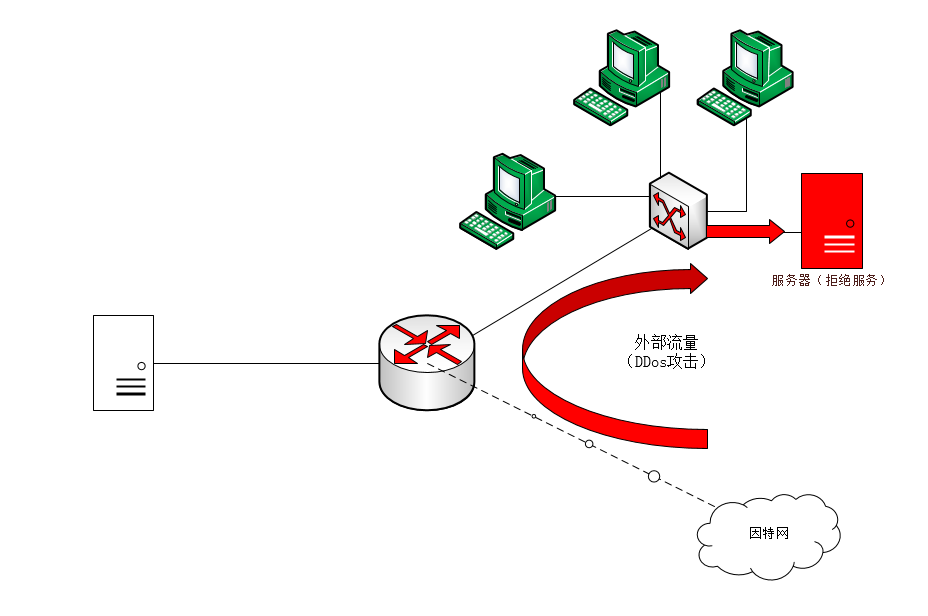


图 14. 服务器受到因特网中DDos攻击导致服务资源被占用

**（3）动态防护平台通过流量检测发现服务器流量异常，发出警报：**

防护平台经过流量统计监控发现该服务器流量异常，发出警报并报告攻击节点。然后通过下发流表等方式将来自外部网络的DDos攻击流导向网络中特定服务器处理，使得服务能够正常进行。

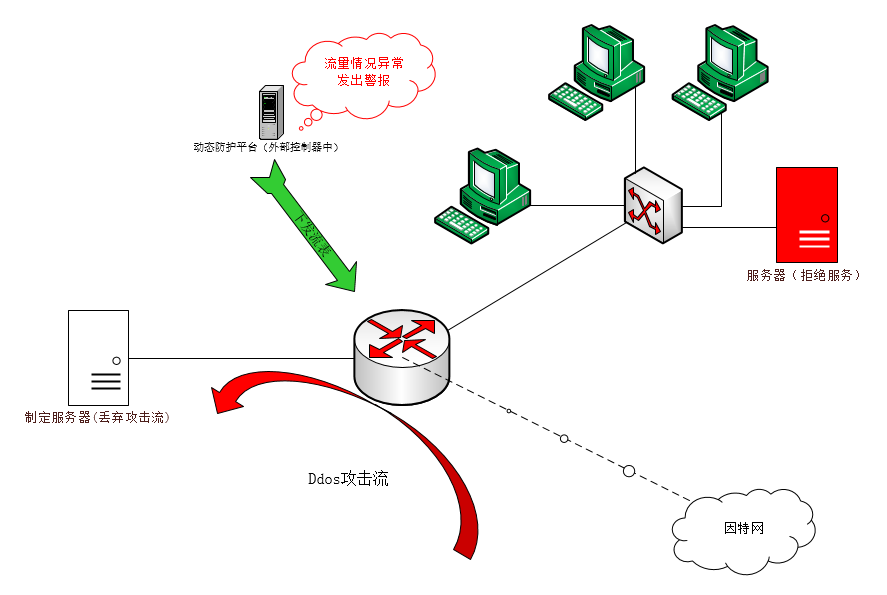


图 15. 动态防护平台检测到流量异常发出警报

**（4）服务器恢复正常服务之后，防护平台警报消除：**

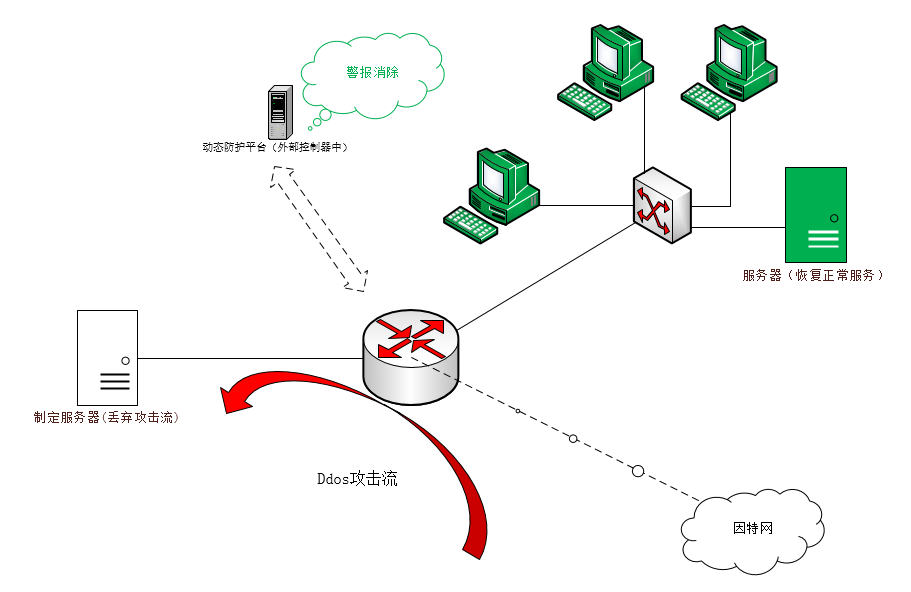


图 16. 服务器恢复正常服务

**服务器流量随时间变化图：**

时间

Ddos攻击

开始

图 17. 服务器流量曲线图

我们可以从此示意图中看到服务器流量在受到DDos攻击时呈一个突然上升趋势，当防护平台检测到受到攻击后下发流表将攻击流导向至特定服务器节点后受攻击服务器流量开始下降最后趋于稳定。

### 5.2.2. 攻击检测流程

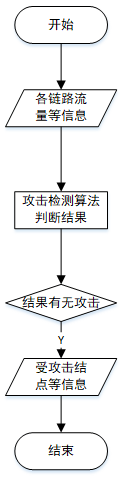


图 18. DDoS攻击检测流程图

上图描述了DDoS攻击的检测流程，接收各链路流量等信息，交由攻击检测算法判断结果，攻击检测算法主要关注各个结点单位时间段某个端口收到的数据包数量，根据这些判断该主机是否异常，判断是否受到DDoS攻击，若受到攻击输出受攻击结点的信息。

### 5.2.3. 攻击防护流程

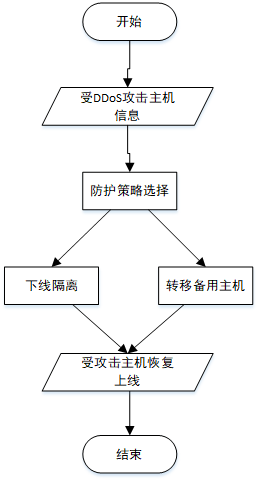


图 19. DDoS攻击防护流程图

上图描述了主机受到DDoS攻击后的防护流程，接收DDoS攻击主机的信息，经过防护策略的选择，可对其进行下线隔离和转移备用主机等防护措施，待到受攻击主机恢复上线后。

# 系统具体实现

## 6.1. 拓扑展示

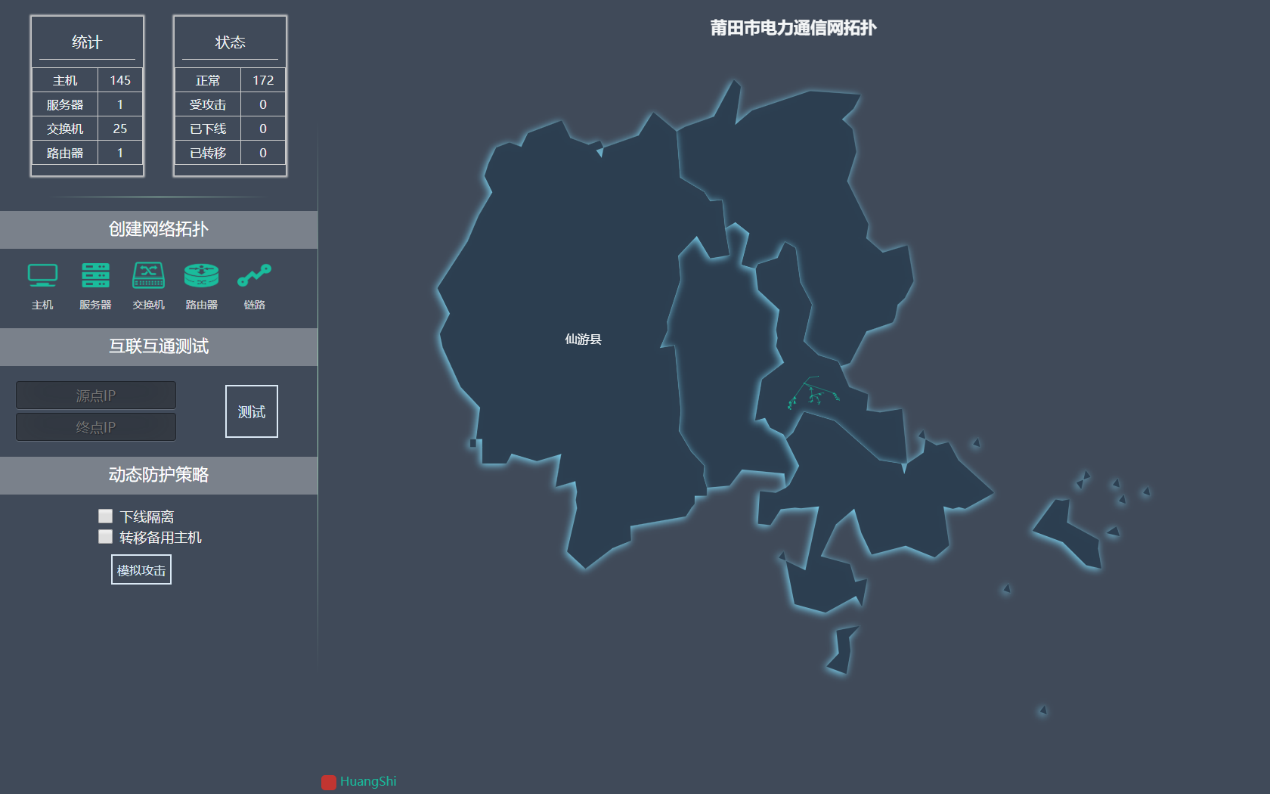
****

图 19.系统主界面

上图为系统的主界面，界面左侧是系统的控制面板，提供管理和控制功能，界面右侧展示Mininet拓扑的可视化，当前拓扑的为莆田市电力通信网拓扑，背景为莆田市的地图。

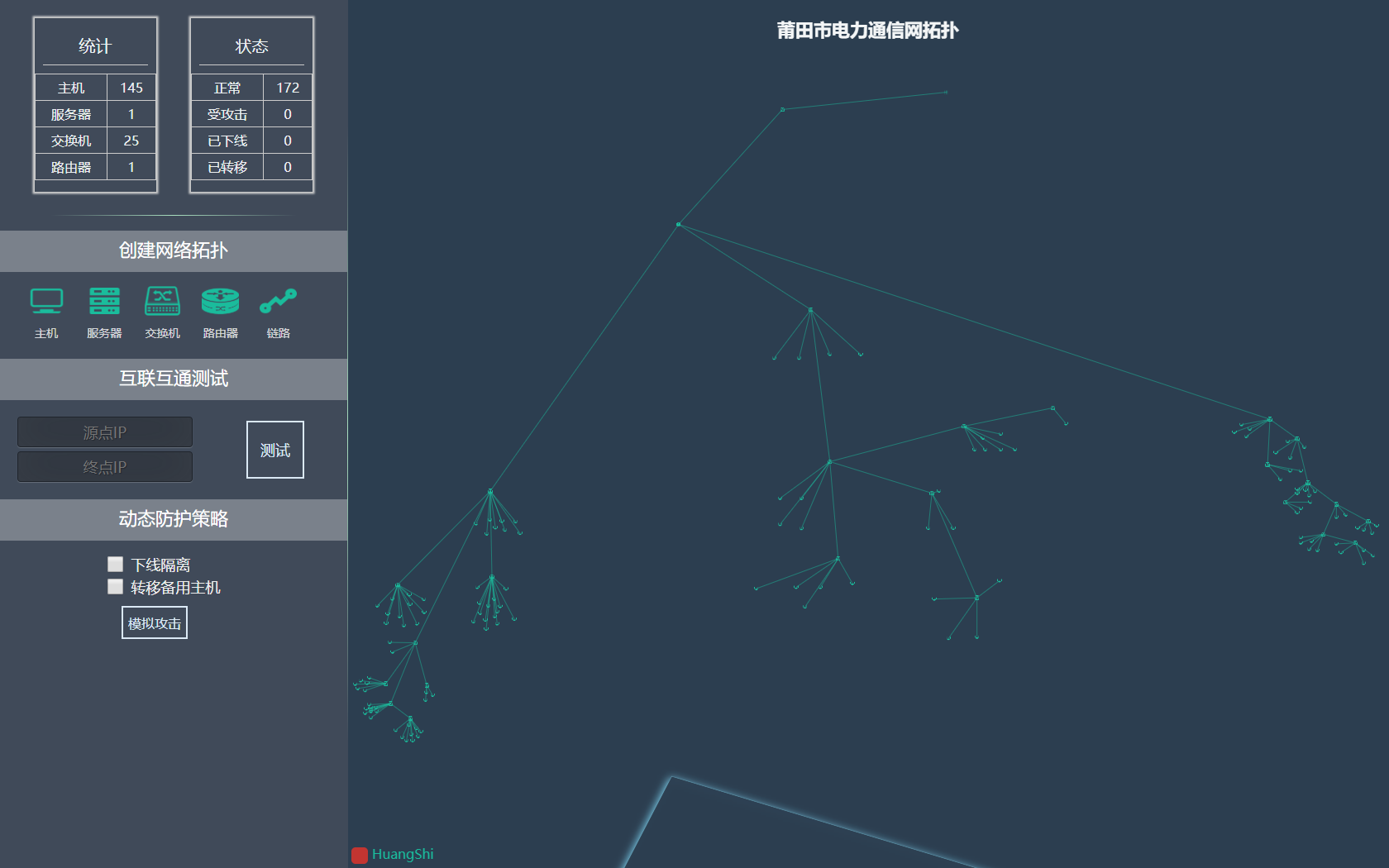


图 20.黄石线路的拓扑

上图拓扑展示的是莆田市电力通信网的黄石线路部分。

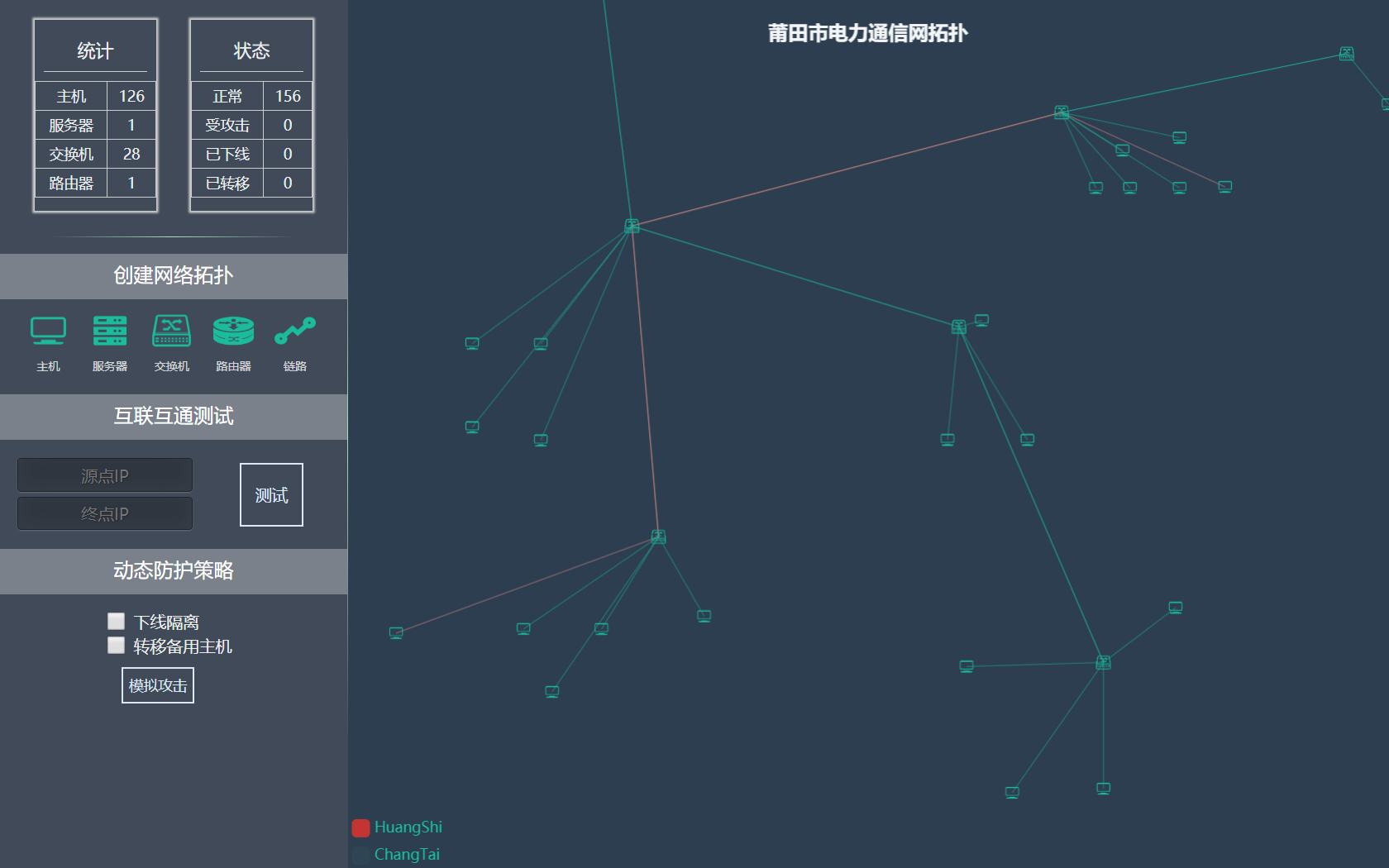


图 21.链路的流量变化

当某条链路出现流量变化时，拓扑上对应的链路会发生颜色变化，从原来的绿色变成橘色并且颜色深度随着流量的增加而增加。

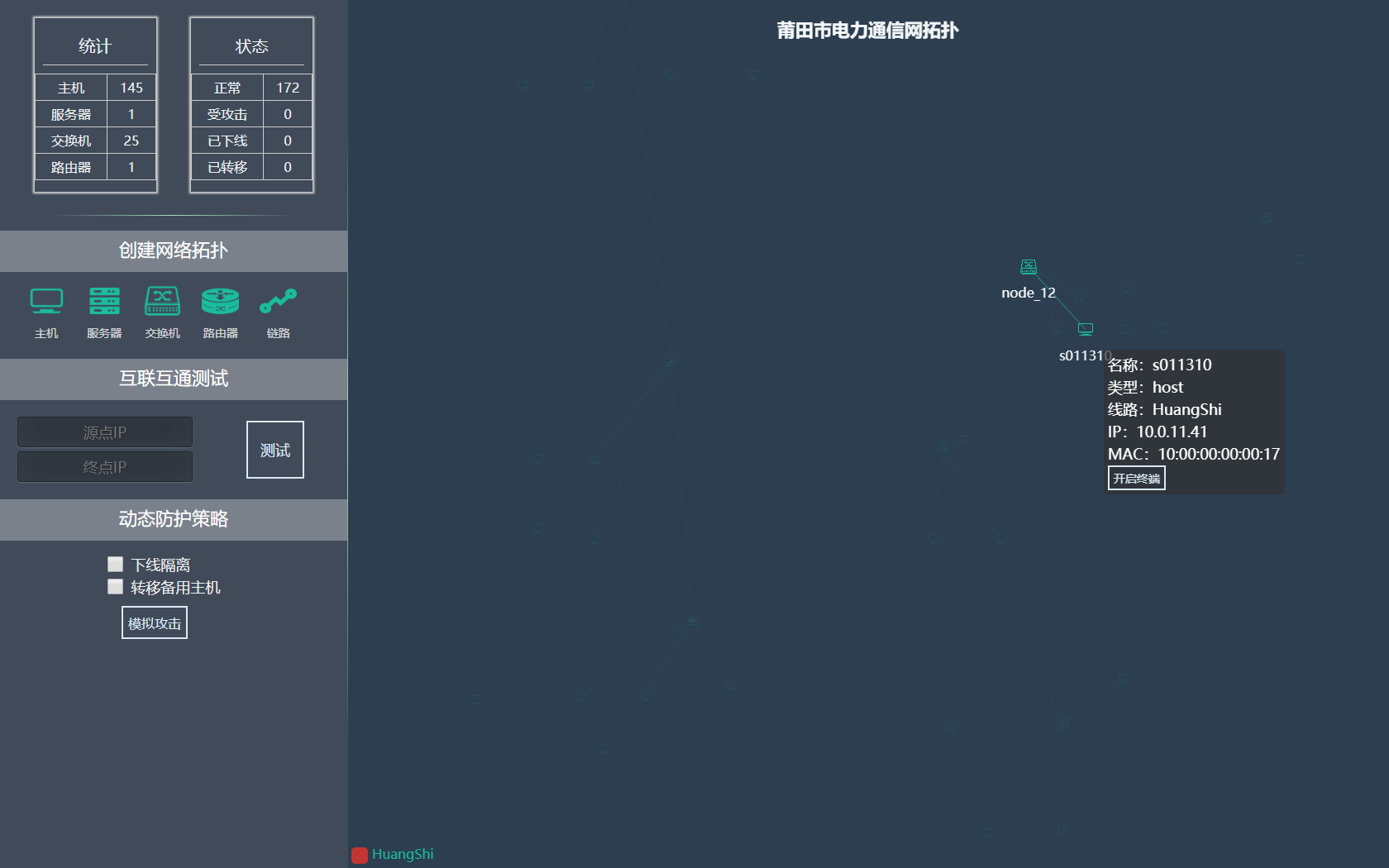


图 22.查看节点信息

点击拓扑上的节点可查看该节点的对应信息，包括名称，类型，所属线路，IP地址，MAC地址。

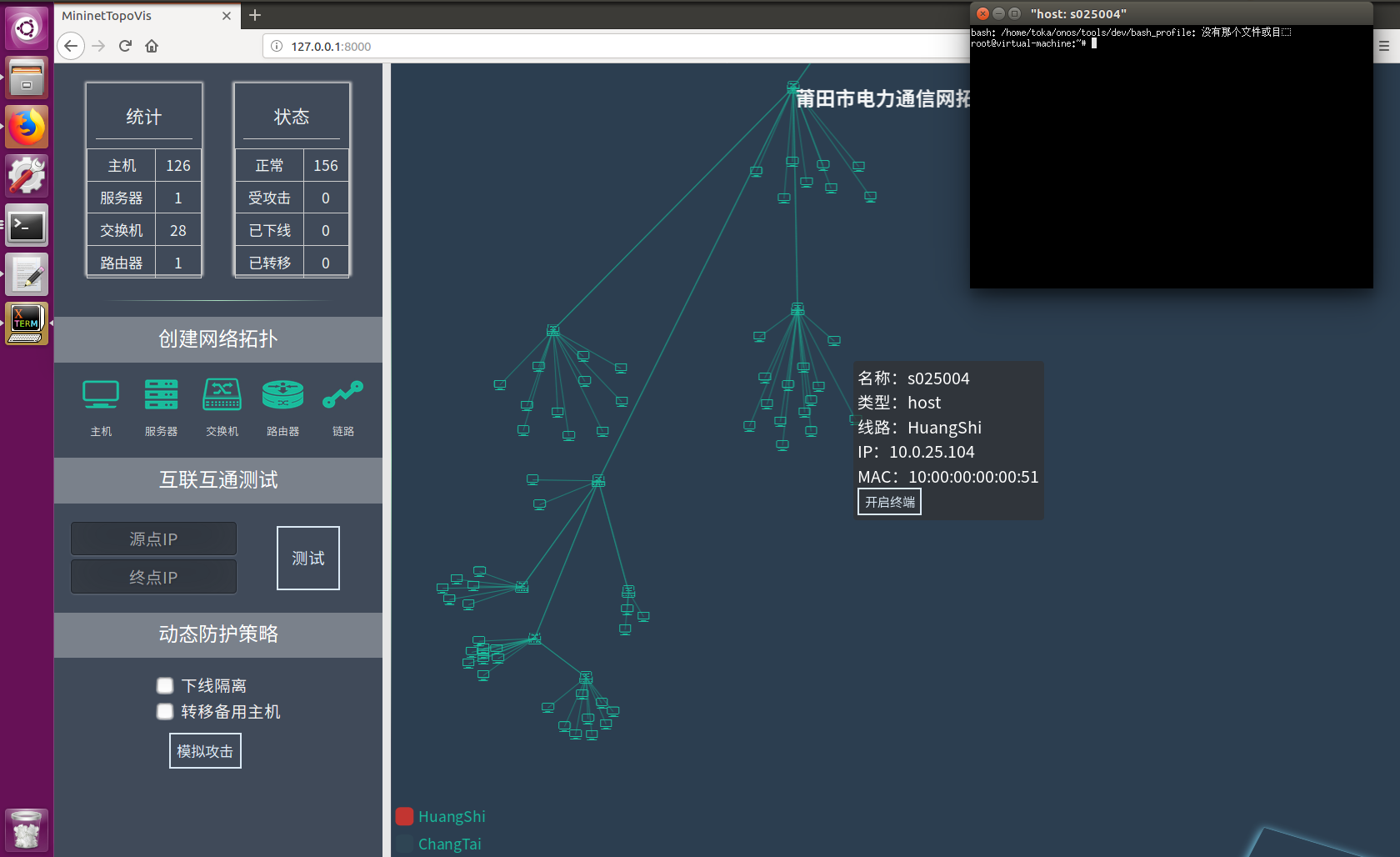


图 23.主机开启终端

在主机节点上可开启其终端通过输入命令行进行其内部的操作。

## 6.2. 数据展示

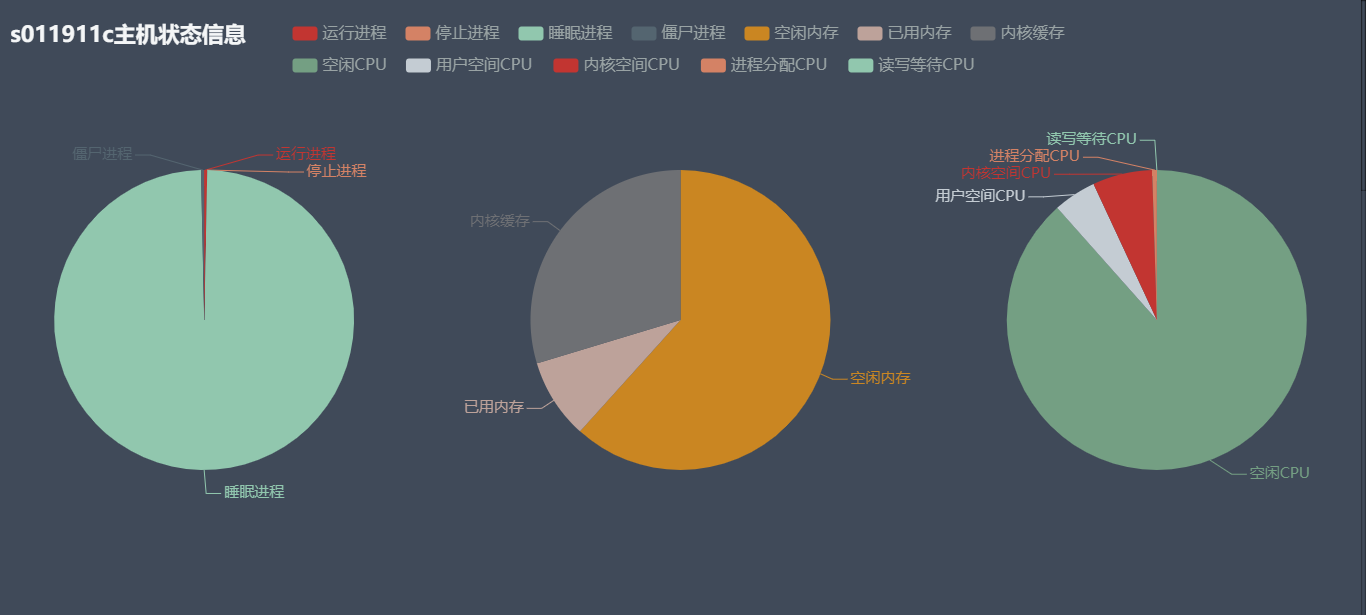


图 24.主机状态信息饼图

双击主机节点可查看对应主机的当前状态信息，包括进程，内存和CPU三类。每一类包括若干子部分，例如进程信息包括僵尸进程、运行进程、停止进程和睡眠进程，通过饼图的方式来展示这三类主机状态信息。

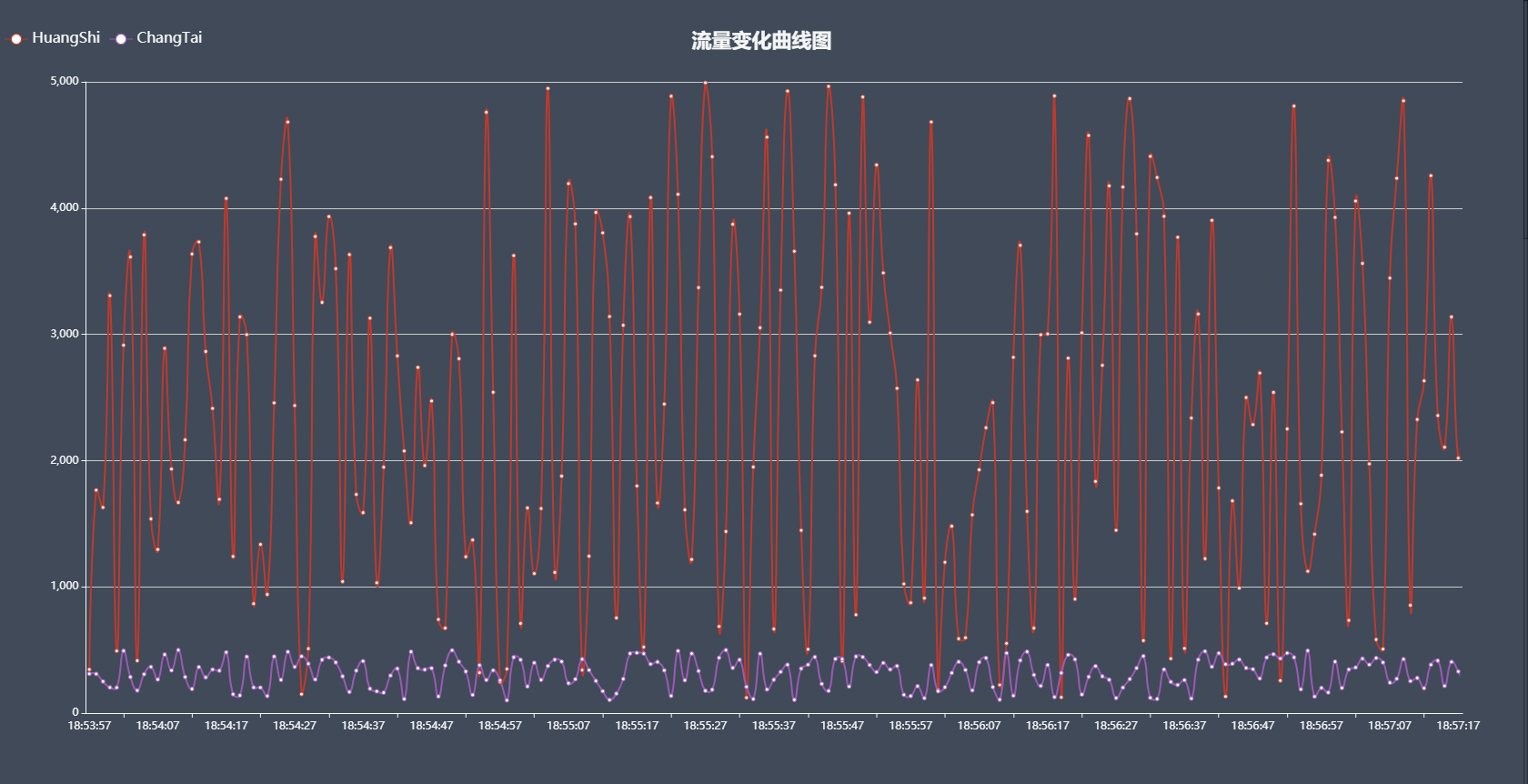


图 25.线路流量变化曲线图

在统计页面可查看当前拓扑上各线路的流量实时变化，以曲线图的方式展现。

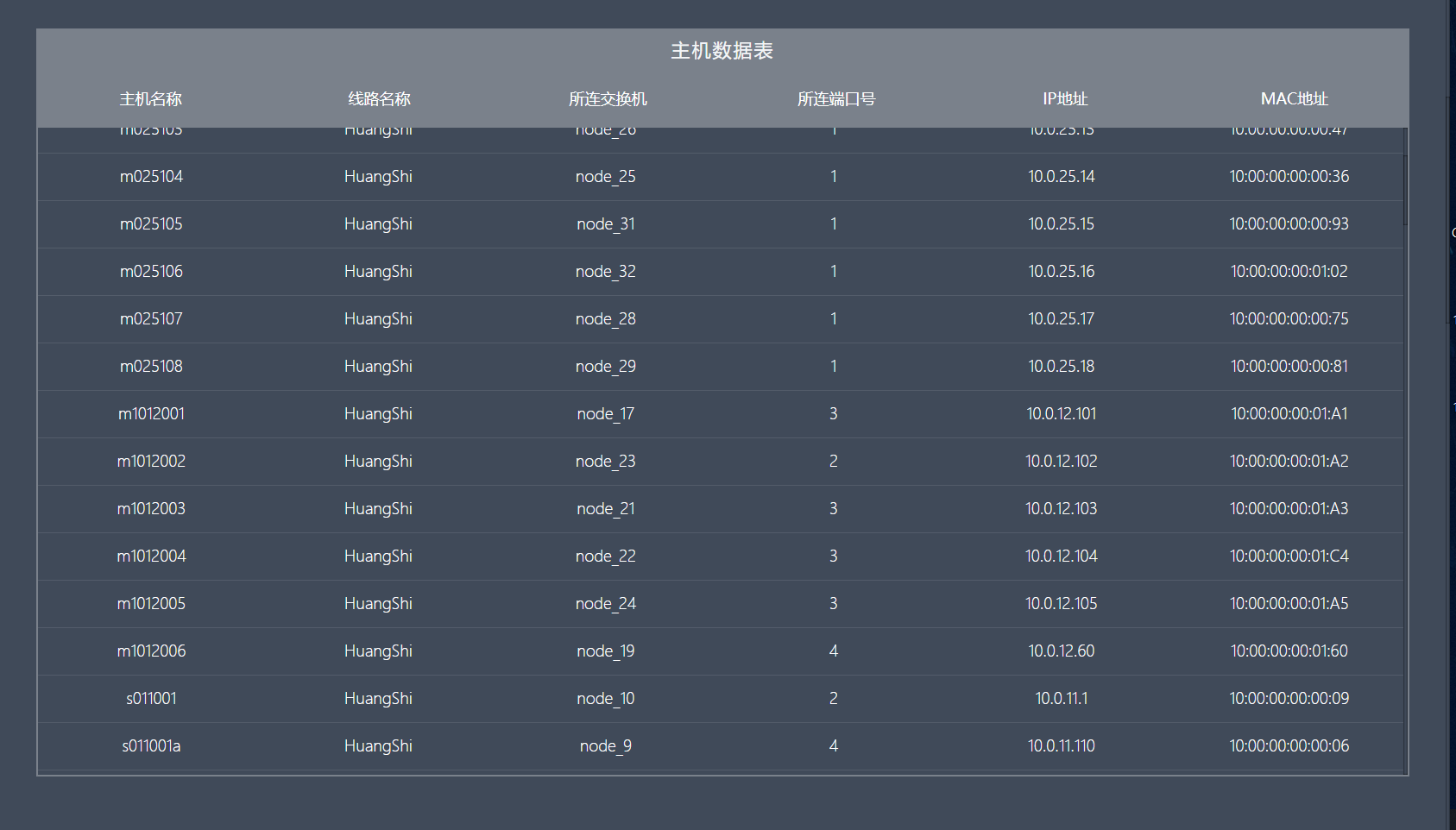


图 26.主机数据统计表

在统计页面可查看当前拓扑上的所有主机的数据统计表，包括主机名称、所属的线路名称、所连交换机、所连端口号、IP地址和MAC地址。



图 27.设备数据表

在统计页面可查看当前拓扑上的所有设备（交换机，路由器）的数据统计表，内容包括设备名称、所属线路名称、端口数量和对应端口所连设备。