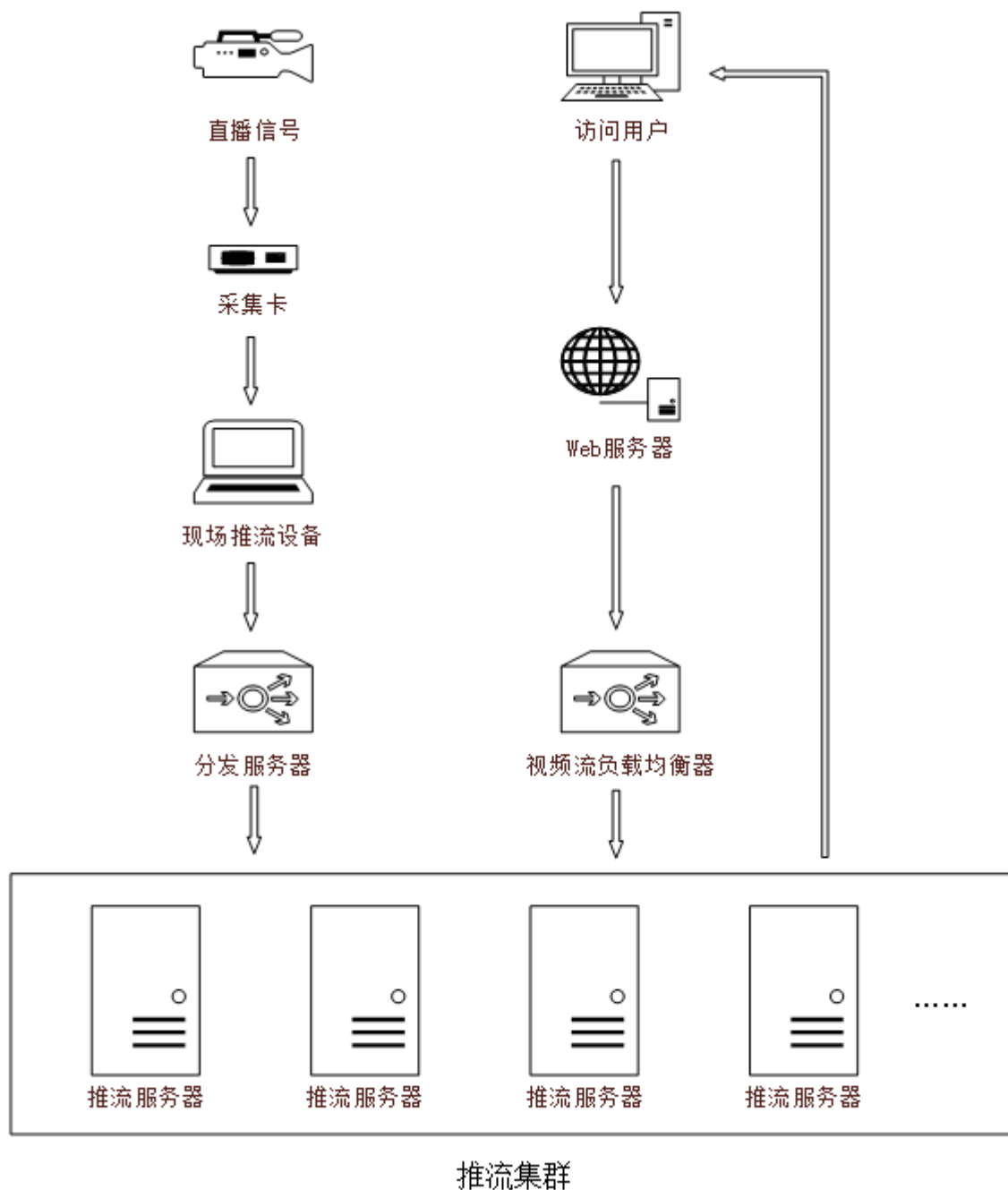


# 网络开拓者协会

## 直播系统 NPLive 架构说明文档

Last Updated at 2017/10/08 04:31 by Jack.H

### 系统结构图



# 系统要求

---

## 网络环境

1. 现场推流设备与分发服务器之间的网络环境直接影响直播质量，需尽量保证稳定，以有线连接为主，长距离传输时按照 光纤、POE网线中继、专用WLAN、4G-LTE 的优先顺序选择，可视经费和现场情况进行选择。
2. 分发服务器最佳的网络环境为公网、校园局域网均可访问，以保证不同现场环境下的可访问性，若不能实现单一服务器跨双网，可考虑多服务器协同工作。
3. 当视频流负载均衡器采用LVS-DR模式时，要求负载均衡器与推流服务器处于同一物理网段上(不经过超过1个路由器)。

## 直播信号

1. 可选用所有采集卡所支持的直播信号。
2. 多机位多视频源可在现场推流设备上整合，或将多个源同时推流至在线导播台进行整合。

## 采集卡

1. 当前使用的采集卡型号：天创恒达UB530。
2. 现场推流设备需安装相应的驱动程序(可在内部FTP上找到)。
3. 现场视频设备常用接口为HDMI或SDI，SDI清晰度更高。
4. 若需采购新的采集卡，支持HDMI/SDI输入，以及USB3.0以上的输出即可。

## 现场推流设备

1. 天创恒达UB530采集卡要求所连接的计算机具有USB3.0及以上接口。
2. 需安装OBS等推流软件。
3. 高清推流对计算机配置要求较高，CPU/显卡均有要求，可参考游戏本配置，使用高配置专用工作站为最佳。
4. 需要有稳定的网络环境，并能正常访问分发服务器。
5. 可在推流软件进行简单的画面切换、放置台标等操作。

## 服务器

1. 所有服务器推荐1Gbps及以上带宽接入。
2. 服务器配置无特殊要求，若后期需支持转码，实现多清晰度选择，则需较好CPU。

# 功能实现

---

## Nginx-RTMP-Module

- <https://github.com/arut/nginx-rtmp-module>
- 参照ReadMe，在所有分发服务器、推流服务器配置完毕。

## 视频分发

- 在分发服务器配置好Nginx-RTMP-Module后，参照如下配置设置nginx.conf:

- ```
#user nobody;
worker_processes 1;

error_log logs/error.log debug;

events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;

    sendfile on;
    keepalive_timeout 65;

    server {
        listen 8092;
        server_name livestream.bitnp.net;
        location /nplive {
            #server hls fragments
            types{
                application/vnd.apple.mpegurl m3u8;
                video/mp2t ts;
            }
            alias /tmp/nplive_hls;
            expires -1;
            add_header Access-Control-Allow-Origin http://live.bitnp.net;
            #proxy_pass http://livestream.bitnp.net;
        }
    }

    server {
        listen 8099;
        server_name localhost;

        # rtmp stat
        location /stat {
            rtmp_stat all;
            rtmp_stat_stylesheet stat.xsl;
        }
        location /stat.xsl {
            # you can move stat.xsl to a different location
            root /usr/local/nginx-rtmp-module;
        }

        # rtmp control
        location /control {
            rtmp_control all;
        }

        error_page 500 502 503 504 /50x.html;
```

```
        location = /50x.html {
            root    html;
        }
    }
}

rtmp {
    server {
        listen 1935;
        notify_method get;

        application nplive {
            live on;

            hls on;
            hls_path /tmp/nplive_hls;
            hls_sync 15s;
            #hls_fragment 3s;
            #hls_playlist_length 6s;
            #Added by Defjia 21st, Aus, 2017

            push rtmp://10.2.70.111:1935/nplive/livestream;

            push rtmp://10.2.70.135:1935/nplive/livestream;
            push_reconnect 1s;
        }
    }
}
```

## 推流服务

- 在推流服务器配置好Nginx-RTMP-Module后，参照如下配置设置nginx.conf:

- ```
#user nobody;
worker_processes 1;

error_log logs/error.log debug;

events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;

    sendfile on;
    keepalive_timeout 65;

    server {
        listen 8092;
        server_name livestream.bitnp.net;
        location /nplive {
            #server hls fragments
            types{
                application/vnd.apple.mpegurl m3u8;
                video/mp2t ts;
            }
            alias /tmp/nplive_hls;
            expires -1;
            # To allow ajax crossdomain access
            add_header Access-Control-Allow-Origin http://live.bitnp.net;
        }
    }

    server {
        listen 8099;
        server_name localhost;

        # rtmp stat
        location /stat {
            rtmp_stat all;
            rtmp_stat_stylesheet stat.xsl;
        }
        location /stat.xsl {
            # you can move stat.xsl to a different location
            root /usr/local/nginx-rtmp-module;
        }

        # rtmp control
        location /control {
            rtmp_control all;
        }

        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
```

```

        root    html;
    }
}

rtmp {
    server {
        listen 1935;
        notify_method get;

        application nplive {
            live on;

            hls on;
            hls_path /tmp/nplive_hls;
            hls_sync 15s;
        }
    }
}
}

```

- 需要注意的是，为允许Ajax跨域访问，需在所有用户可访问的推流服务器上设置Access-Control-Allow-Origin的作用域名。（已在上方nginx.conf中标出）
- 若用户使用RTMP流观看，则需在所有RTMP推流服务器流文件根目录下放置 crossdomain.xml，其中属性 domain，其值为允许访问的域，可以是确切的 IP 地址、一个确切的域或一个通配符域（任何域）。例子如下：

- ```

<?xml version="1.0"?>
<cross-domain-policy>
<allow-access-from domain="live.bitnp.net" />
<allow-access-from domain="*.bitnp.net" />
<allow-access-from domain="10.2.70.102" />
</cross-domain-policy>

```

- 或直接在nginx.conf中http -> server添加：

- ```

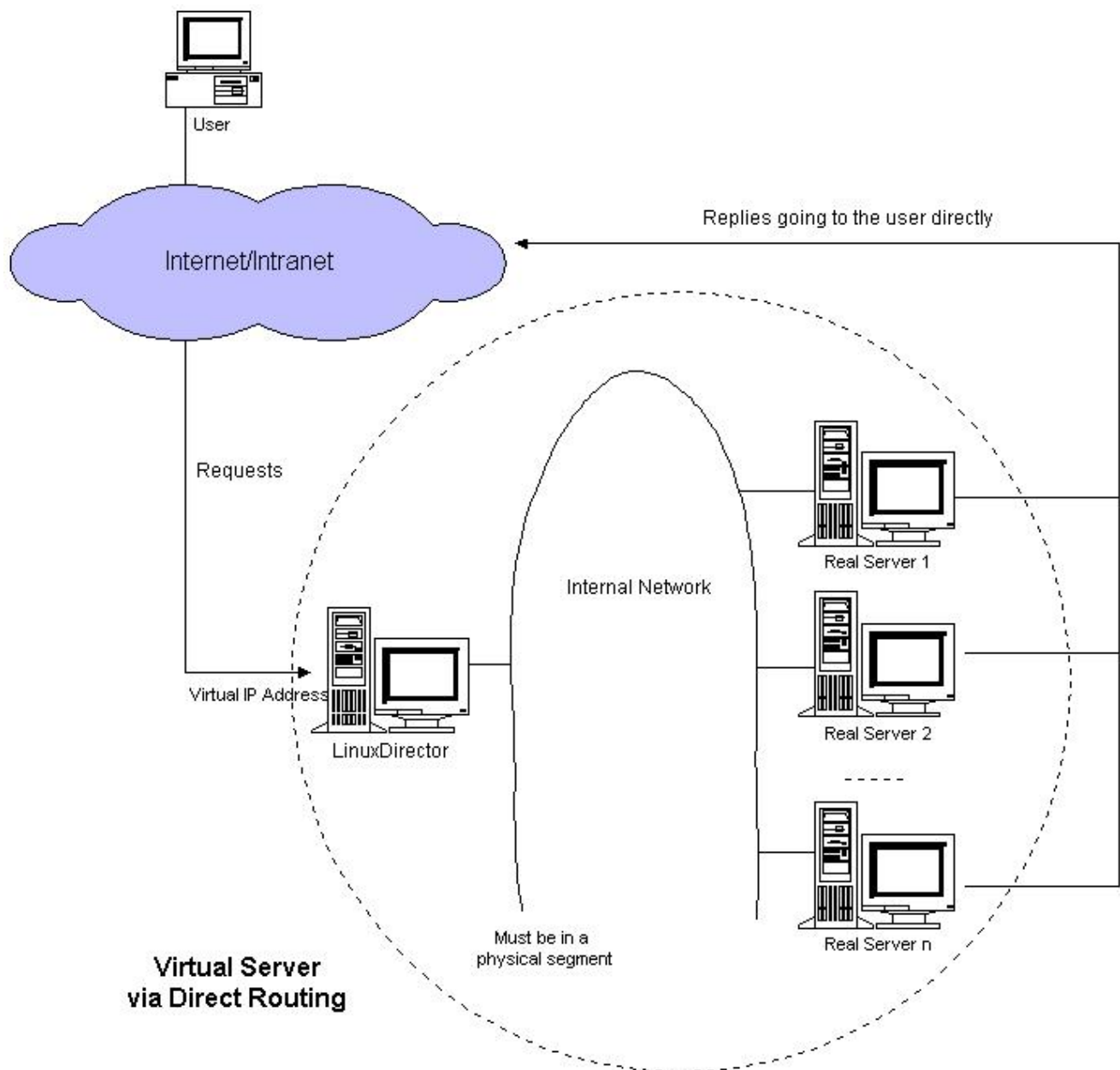
location /crossdomain.xml {
    default_type text/xml;
    return 200 '<?xml version="1.0"?>
        <!DOCTYPE cross-domain-policy SYSTEM "http://www.adobe.com/xml/dtds/cross-
domain-policy.dtd">
        <cross-domain-policy>
            <site-control permitted-cross-domain-policies="all"/>
            <allow-access-from domain="*" secure="false"/>
            <allow-http-request-headers-from domain="*" headers="*"
secure="false"/>
        </cross-domain-policy>';
    expires 24h;
}

```

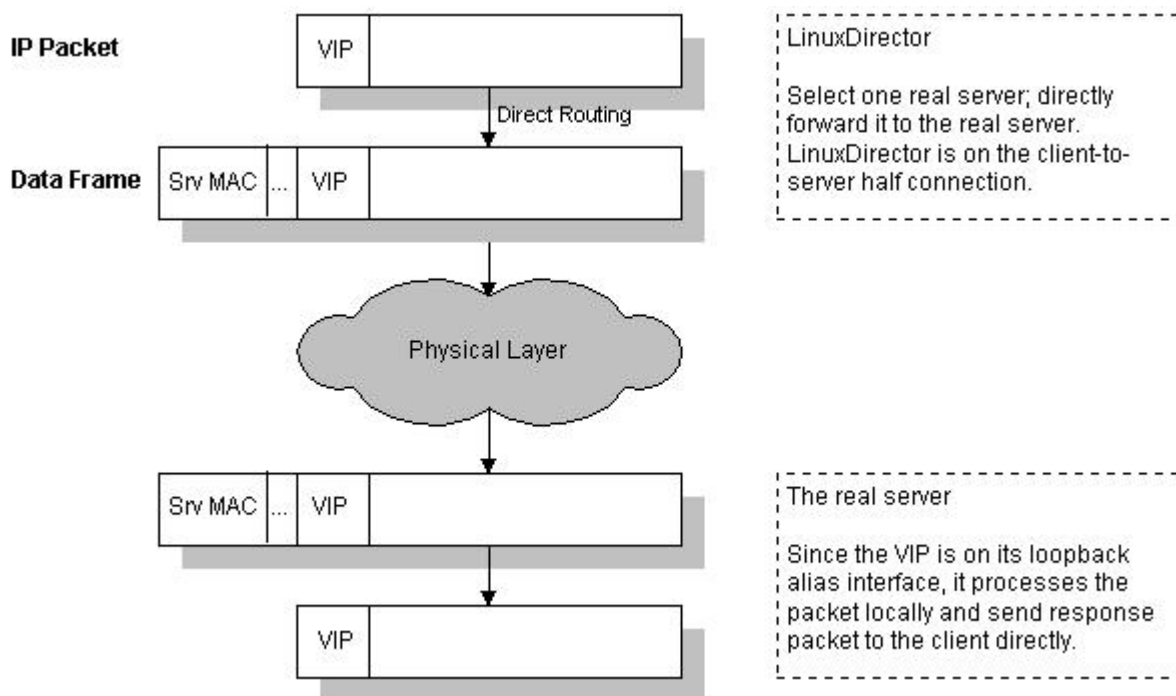
## 负载均衡

### LVS-DR 基本原理

- LVS-DR要求调度器和服务器组都必须在物理上有一个网卡通过不分段的局域网相连，即通过交换机或者高速的HUB相连，中间没有隔有路由器。VIP(Virtual IP Address)地址为调度器和服务器组共享，调度器配置的VIP地址是对外可见的，用于接收虚拟服务的请求报文；所有的服务器把VIP地址配置在各自的Non-ARP网络设备(一般为本地回环 lo )上，它对外面是不可见的，只是用于处理目标地址为VIP的网络请求。



在VS/DR中，调度器根据各个服务器的负载情况，动态地选择一台服务器，将数据帧的MAC地址改为所选服务器的MAC地址，再将修改后的数据帧在集群局域网上发送。因为数据帧的MAC地址是选出的推流服务器，所以服务器肯定可以收到这个数据帧，从中可以获得该IP报文。当服务器发现报文的目标地址VIP是在本地的网络设备(网卡)上，服务器处理这个报文，然后根据路由表将响应报文直接返回给用户。



## LVS-DR 配置说明

1. IP配置样例，此处选用VIP为10.2.70.100

Server	eth(网卡)	IP	Mask
Director	eth0 (主网卡)	10.2.70.102	255.255.255.0
	eth0:0 (主网卡)	10.2.70.100(VIP)	255.255.255.255
Real Server	eth0 (主网卡)	10.2.70.111	255.255.255.0
	lo (本地回环)	10.2.70.100(VIP)	255.255.255.255

上面绑定VIP的掩码是"255.255.255.255"，说明广播地址是其本身，那么他就不会将ARP发送到实际的自己该属于的广播域了，这样防止与LVS上VIP冲突，而导致IP冲突。

2. 如果Real-server有个多个网卡，每个网卡在不同的网段，那么可以过滤掉非本网卡ARP请求的回应：

```
sysctl -w net.ipv4.conf.all.arp_filter=1
```

但是如果多个网卡的IP在一个网段，那么就不行，对于多个接口在相同网段可以设置下面的来防止：

```
sysctl -w net.ipv4.conf.all.arp_ignore=1
sysctl -w net.ipv4.conf.all.arp_announce=2
```

3. 在负载均衡器LVS中添加Real Server节点：

```
ipvsadm -C
ipvsadm -A -t 10.2.70.100:8092 -s wlc
ipvsadm -a -t 10.2.70.100:8092 -r 10.2.70.111 -g
ipvsadm -a -t 10.2.70.100:8092 -r 10.2.70.135 -g
```



## 参考资料

- 原理: <http://blog.csdn.net/pi9nc/article/details/23380589>
- 实例: <http://chrinux.blog.51cto.com/6466723/1198748>

## 备注

- 若LVS和Real-Server不在同一个网络内, 使用LVS-TUN模式为最佳选择, 且与LVS-DR吞吐量接近。

## Todo

---

### 后台管理

- 使用可供Web访问的后台页面对集群的开关状态、推流地址、参数设置等相关配置进行实时控制。

### 鉴权机制

- 使用时间戳和Secret Key对推流地址进行MD5运算, 防止推流地址被盗导致直播内容不可控。

### 弹幕/评论

- 实时弹幕互动, 可采用WebSocket结合B站或清华TUNA开源弹幕代码的方式。

### 公网访问

- 接入云厂商的弹性CDN, 为保证整个系统可供校内免流和校外访问, 可能需要引入自主DNS,

### 接入CAS

- 接入网协统一身份认证系统, 用户登录以发送弹幕或礼物。