

四层、七层负载均衡的区别



薛晨 (/u/a2d715858c25) + 关注

2016.05.23 18:51* 字数 5893 阅读 27842 评论 6 喜欢 101 赞赏 1

(/u/a2d715858c25)

** 内容安排: **

- 简介
- 区别
- Nginx、LVS及HAProxy负载均衡软件的优缺点

一、简介

1. ** 所谓四层就是基于IP+端口的负载均衡；七层就是基于URL等应用层信息的负载均衡；**同理，还有基于MAC地址的二层负载均衡和基于IP地址的三层负载均衡。 换句话说，二层负载均衡会通过一个虚拟MAC地址接收请求，然后再分配到真实的MAC地址；三层负载均衡会通过一个虚拟IP地址接收请求，然后再分配到真实的IP地址；四层通过虚拟IP+端口接收请求，然后再分配到真实的服务器；七层通过虚拟的URL或主机名接收请求，然后再分配到真实的服务器。
2. ** 所谓的四到七层负载均衡，就是在对后台的服务器进行负载均衡时，依据四层的信息或七层的信息来决定怎么样转发流量。** 比如四层的负载均衡，就是通过发布三层的IP地址（VIP），然后加四层的端口号，来决定哪些流量需要做负载均衡，对需要处理的流量进行NAT处理，转发至后台服务器，并记录下这个TCP或者UDP的流量是由哪台服务器处理的，后续这个连接的所有流量都同样转发到同一台服务器处理。七层的负载均衡，就是在四层的基础上（**没有四层是绝对不可能有七层的**），再考虑应用层的特征，比如同一个Web服务器的负载均衡，除了根据VIP加80端口辨别是否需要处理的流量，还可根据七层的URL、浏览器类别、语言来决定是否要进行负载均衡。举个例子，如果你的Web服务器分成两组，一组是中文语言的，一组是英文语言的，那么七层负载均衡就可以当用户来访问你的域名时，自动辨别用户语言，然后选择对应的语言服务器组进行负载均衡处理。
3. 负载均衡器通常称为四层交换机或七层交换机。四层交换机主要分析IP层及TCP/UDP层，实现四层流量负载均衡。七层交换机除了支持四层负载均衡以外，还有分析应用层的信息，如HTTP协议URI或Cookie信息。
4. 负载均衡分为L4 switch（四层交换），即在OSI第4层工作，就是TCP层啦。此种Load Balance不理解应用协议（如HTTP/FTP/MySQL等等）。例子：LVS，F5。

(https://
click.y
slot=3
c7ae-4
ba75ct



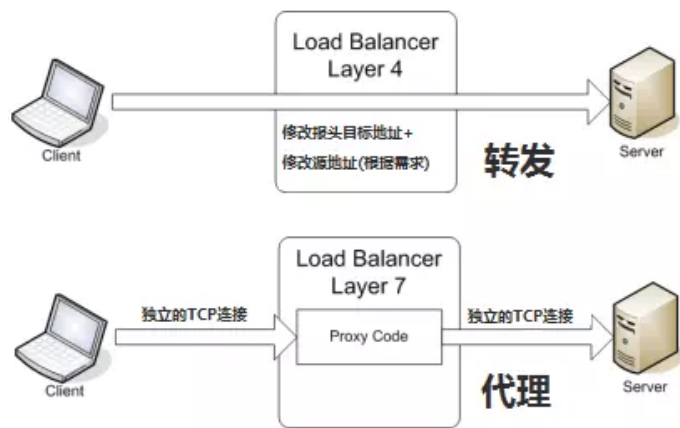
二、区别

1. 技术原理上

所谓四层负载均衡，也就是主要通过报文中的目标地址和端口，再加上负载均衡设备设置的服务器选择方式，决定最终选择的内部服务器。

以常见的TCP为例，负载均衡设备在接收到第一个来自客户端的SYN 请求时，即通过上述方式选择一个最佳的服务器，并对报文中目标IP地址进行修改(改为后端服务器IP)，直接转发给该服务器。TCP的连接建立，即三次握手是客户端和服务端直接建立的，负载均衡设备只是起到一个类似路由器的转发动作。在某些部署情况下，为保证服务器回包可以正确返回给负载均衡设备，在转发报文的同时可能还会对报文原来的源地址进行修改。

(<https://link.jianshu.com?t=http://officialblog-wordpress.stor.sinaapp.com/uploads/2014/07/d.png>)



区别 (<https://link.jianshu.com?t=http://officialblog-wordpress.stor.sinaapp.com/uploads/2014/07/d.png>)

所谓七层负载均衡，也称为“内容交换”，也就是主要通过报文中的真正有意义的应用层内容，再加上负载均衡设备设置的服务器选择方式，决定最终选择的内部服务器。

以常见的TCP为例，负载均衡设备如果要根据真正的应用层内容再选择服务器，只能先代理最终的服务器和客户端建立连接(三次握手)后，才可能接受到客户端发送的真正应用层内容的报文，然后再根据该报文中的特定字段，再加上负载均衡设备设置的服务器选择方式，决定最终选择的内部服务器。负载均衡设备在这种情况下，更类似于一个代理服务器。负载均衡和前端的客户端以及后端的服务器会分别建立TCP连接。所以从这个技术原理上来看，七层负载均衡明显的对负载均衡设备的要求更高，处理七层的能力也必然会低于四层模式的部署方式。

2. 应用场景

七层应用负载的好处，是使得整个网络更智能化。例如访问一个网站的用户流量，可以通过七层的方式，将对图片类的请求转发到特定的图片服务器并可以使用缓存技术；将对文字类的请求可以转发到特定的文字服务器并可以使用压缩技术。当然这只是七层应用的一个小案例，从技术原理上，这种方式可以对客户端的请求和服务器的响应进行任意意义上的修改，极大的提升了应用系统在网络层的灵活性。很多在后台，例如Nginx或者Apache上部署的功能可以前移到负载均衡设备上，例如客户请求中的Header重写，服务器响应中的关键字过滤或者内容插入等功能。

另外一个常常被提到功能就是安全性。网络中最常见的SYN Flood攻击，即黑客控制众多源客户端，使用虚假IP地址对同一目标发送SYN攻击，通常这种攻击会大量发送SYN报文，耗尽服务器上的相关资源，以达到Denial of Service(DoS)的目的。从技术原理上也可以看出，四层模式下这些SYN攻击都会被转发到后端的服务器上；而七层模式下这些SYN攻击自然在负载均衡设备上就截止，不会影响后台服务器的正常运营。另外负载均衡设备可以在七层层面设定多种策略，过滤特定报文，例如SQL Injection等应用层面的特定攻击手段，从应用层面进一步提高系统整体安全。

现在的7层负载均衡，主要还是着重于应用HTTP协议，所以其应用范围主要是众多的网站或者内部信息平台等基于B/S开发的系统。4层负载均衡则对应其他TCP应用，例如基于C/S开发的ERP等系统。

三、Nginx、LVS及HAProxy负载均衡软件的优缺点

负载均衡（Load Balancing）建立在现有网络结构之上，它提供了一种廉价有效透明的方法扩展网络设备和服务器的带宽、增加吞吐量、加强网络数据处理能力，同时能够提高网络的灵活性和可用性。

Nginx/LVS/HAProxy是目前使用最广泛的三种负载均衡软件。

一般对负载均衡的使用是随着网站规模的提升根据不同的阶段来使用不同的技术。具体的应用需求还得具体分析，如果是中小型的Web应用，比如日PV小于1000万，用Nginx就完全可以了；如果机器不少，可以用DNS轮询，LVS所耗费的机器还是比较多的；大型网站或重要的服务，且服务器比较多时，可以考虑用LVS。

一种是通过硬件来进行，常见的硬件有比较昂贵的F5和Array等商用的负载均衡器，它的优点就是有专业的维护团队来对这些服务进行维护、缺点就是花销太大，所以对于规模较小的网络服务来说暂时还没有需要使用；另外一种就是类似于Nginx/LVS/HAProxy的基于Linux的开源免费的负载均衡软件，这些都是通过软件级别来实现，所以费用非常低廉。

目前关于网站架构一般比较合理流行的架构方案：Web前端采用Nginx/HAProxy+Keepalived作负载均衡器；后端采用MySQL数据库一主多从和读写分离，采用LVS+Keepalived的架构。当然要根据项目具体需求制定方案。

下面说说各自的特点和适用场合。

Nginx的优点是：

1. 工作在网络的7层之上，可以针对http应用做一些分流的策略，比如针对域名、目录结构，它的正则规则比HAProxy更为强大和灵活，这也是它目前广泛流行的主要原因之一，Nginx单凭这点可利用的场合就远多于LVS了。
2. Nginx对网络稳定性的依赖非常小，理论上能ping通就能进行负载功能，这个也是它的优势之一；相反LVS对网络稳定性依赖比较大。
3. Nginx安装和配置比较简单，测试起来比较方便，它基本能把错误用日志打印出来。LVS的配置、测试就要花比较长的时间了，LVS对网络依赖比较大。
4. 可以承担高负载压力且稳定，在硬件不差的情况下一般能支撑几万次的并发量，负载度比LVS相对小些。
5. Nginx可以通过端口检测到服务器内部的故障，比如根据服务器处理网页返回的状态码、超时等等，并且会把返回错误的请求重新提交到另一个节点，不过其中缺点就是不支持url来检测。比如用户正在上传一个文件，而处理该上传的节点刚好在上传过程中出现故障，Nginx会把上传切到另一台服务器重新处理，而LVS就直接断掉了，如果是上传一个很大的文件或者很重要的文件的话，用户可能会因此而不满。
6. Nginx不仅仅是一款优秀的负载均衡器/反向代理软件，它同时也是功能强大的Web应用服务器。LNMP也是近几年非常流行的web架构，在高流量的环境中稳定性也很好。
7. Nginx现在作为Web反向加速缓存越来越成熟了，速度比传统的Squid服务器更快，可以考虑用其作为反向代理加速器。
8. Nginx可作为中层反向代理使用，这一层面Nginx基本上无对手，唯一可以对比Nginx的就只有lighttpd了，不过lighttpd目前还没有做到Nginx完全的功能，配置也不那么清晰易读，社区资料也远远没Nginx活跃。
9. Nginx也可作为静态网页和图片服务器，这方面的性能也无对手。还有Nginx社区非常活跃，第三方模块也很多。

Nginx的缺点是：

1. Nginx仅能支持http、https和Email协议，这样就在适用范围上面小些，这个是它的缺点。
2. 对后端服务器的健康检查，只支持通过端口来检测，不支持通过url来检测。不支持Session的直接保持，但能通过ip_hash来解决。

LVS：使用Linux内核集群实现一个高性能、高可用的负载均衡服务器，它具有很好的伸缩性（Scalability）、可靠性（Reliability）和可管理性（Manageability）。

(/apps/
utm_sc
banner

(https://
click.y
slot=30
c7ae-4
ba75ct



LVS的优点是：

1. 抗负载能力强、是工作在网络4层之上仅作分发之用，没有流量的产生，这个特点也决定了它在负载均衡软件里的性能最强的，对内存和cpu资源消耗比较低。
2. 配置性比较低，这是一个缺点也是一个优点，因为没有可太多配置的东西，所以并不需要太多接触，大大减少了人为出错的几率。
3. 工作稳定，因为其本身抗负载能力很强，自身有完整的双机热备方案，如LVS+Keepalived。
4. 无流量，LVS只分发请求，而流量并不从它本身出去，这点保证了均衡器IO的性能不会受到大流量的影响。
5. 应用范围比较广，因为LVS工作在4层，所以它几乎可以对所有应用做负载均衡，包括http、数据库、在线聊天室等等。

LVS的缺点是：

1. 软件本身不支持正则表达式处理，不能做动静分离；而现在许多网站在这方面都有较强的需求，这个是Nginx/HAProxy+Keepalived的优势所在。
2. 如果是网站应用比较庞大的话，LVS/DR+Keepalived实施起来就比较复杂了，特别后面有 Windows Server的机器的话，如果实施及配置还有维护过程就比较复杂了，相对而言，Nginx/HAProxy+Keepalived就简单多了。

HAProxy的特点是：

1. HAProxy也是支持虚拟主机的。
2. HAProxy的优点能够补充Nginx的一些缺点，比如**支持Session的保持，Cookie的引导；同时支持通过获取指定的url来检测后端服务器的状态。**
3. HAProxy跟LVS类似，本身就只是一款负载均衡软件；单纯从效率上来讲HAProxy会比Nginx有更出色的负载均衡速度，在并发处理上也是优于Nginx的。
4. HAProxy**支持TCP协议的负载均衡转发**，可以对MySQL读进行负载均衡，对后端的MySQL节点进行检测和负载均衡，大家可以用LVS+Keepalived对MySQL主从做负载均衡。
5. HAProxy负载均衡策略非常多，HAProxy的负载均衡算法现在具体有如下8种：
 - ① roundrobin，表示简单的轮询，这个不多说，这个是负载均衡基本都具备的；
 - ② static-rr，表示根据权重，建议关注；
 - ③ leastconn，表示最少连接者先处理，建议关注；

(/apps/
utm_sc
banner

(https:/
click.y
slot=30
c7ae-4
ba75cl



- ④ source，表示根据请求源IP，这个跟Nginx的IP_hash机制类似，我们用其作为解决session问题的一种方法，建议关注；
- ⑤ ri，表示根据请求的URI；
- ⑥ rl_param，表示根据请求的URI参数'balance url_param' requires an URL parameter name；
- ⑦ hdr(name)，表示根据HTTP请求头来锁定每一次HTTP请求；
- ⑧ rdp-cookie(name)，表示根据cookie(name)来锁定并哈希每一次TCP请求。

(/apps/
utm_sc
banner

Nginx和LVS对比的总结：

1. Nginx工作在网络的7层，所以它可以针对http应用本身来做分流策略，比如针对域名、目录结构等，相比之下LVS并不具备这样的功能，所以Nginx单凭这点可利用的场合就远多于LVS了；但Nginx有用的这些功能使其可调整度要高于LVS，所以经常要去触碰触碰，触碰多了，人为出问题的几率也就会大。
2. Nginx对网络稳定性的依赖较小，理论上只要ping得通，网页访问正常，Nginx就能连得通，这是Nginx的一大优势！Nginx同时还能区分内外网，如果是同时拥有内外网的节点，就相当于单机拥有了备份线路；LVS就比较依赖于网络环境，目前来看服务器在同一网段内并且LVS使用direct方式分流，效果较能得到保证。另外注意，LVS需要向托管商至少申请多一个ip来做Virtual IP，貌似是不能用本身的IP来做VIP的。要做好LVS管理员，确实得跟进学习很多有关网络通信方面的知识，就不再是一个HTTP那么简单了。
3. Nginx安装和配置比较简单，测试起来也很方便，因为它基本能把错误用日志打印出来。LVS的安装和配置、测试就要花比较长的时间了；LVS对网络依赖比较大，很多时候不能配置成功都是因为网络问题而不是配置问题，出了问题要解决也相应的会麻烦得多。
4. Nginx也同样能承受很高负载且稳定，但负载度和稳定度差LVS还有几个等级：Nginx处理所有流量所以受限于机器IO和配置；本身的bug也还是难以避免的。
5. Nginx可以检测到服务器内部的故障，比如根据服务器处理网页返回的状态码、超时等等，并且会把返回错误的请求重新提交到另一个节点。目前LVS中 ldirectd也能支持针对服务器内部的情况来监控，但LVS的原理使其不能重发请求。比如用户正在上传一个文件，而处理该上传的节点刚好在上传过程中出现故障，Nginx会把上传切到另一台服务器重新处理，而LVS就直接断掉了，如果是上传一个很大的文件或者很重要的文件的话，用户可能会因此而恼火。
6. Nginx对请求的异步处理可以帮助节点服务器减轻负载，假如使用 apache直接对外服务，那么出现很多的窄带链接时apache服务器将会占用大量内存而不能释放，使用多一个Nginx做apache代理的话，这些窄带链接会被Nginx挡住，apache上就不会堆积过多的请求，这样就减少了相当多的资源占用。这点使用squid也有相同的作用，即使squid本身配置为不缓存，对apache还是有很大帮助的。

(https://
click.y
slot=3C
c7ae-4
ba75ct



7. Nginx能支持http、https和email（email的功能比较少用），LVS所支持的应用在这点上会比Nginx更多。在使用上，一般最前端所采取的策略应是LVS，也就是DNS的指向应为LVS均衡器，LVS的优点令它非常适合做这个任务。重要的ip地址，最好交由LVS托管，比如数据库的ip、webservice服务器的ip等等，这些ip地址随着时间推移，使用面会越来越大，如果更换ip则故障会接踵而至。所以将这些重要ip交给LVS托管是最为稳妥的，这样做的唯一缺点是需要的VIP数量会比较多。Nginx可作为LVS节点机器使用，一是可以利用Nginx的功能，二是可以利用Nginx的性能。当然这一层面也可以直接使用squid，squid的功能方面就比Nginx弱不少了，性能上也有所逊色于Nginx。Nginx也可作为中层代理使用，这一层面Nginx基本上无对手，唯一可以撼动Nginx的就只有lighttpd了，不过lighttpd目前还没有能做到Nginx完全的功能，配置也不那么清晰易读。另外，中层代理的IP也是重要的，所以中层代理也拥有一个VIP和LVS是最完美的方案了。具体的应用还得具体分析，如果是比较小的网站（日PV小于1000万），用Nginx就完全可以了，如果机器也不少，可以用DNS轮询，LVS所耗费的机器还是比较多的；大型网站或者重要的服务，机器不发愁的时候，要多多考虑利用LVS。

现在对网络负载均衡的使用是随着网站规模的提升根据不同的阶段来使用不同的技术：

第一阶段：利用Nginx或HAProxy进行单点的负载均衡，这一阶段服务器规模刚脱离开单服务器、单数据库的模式，需要一定的负载均衡，但是仍然规模较小没有专业的维护团队来进行维护，也没有需要进行大规模的网站部署。这样利用Nginx或HAProxy就是第一选择，此时这些东西上手快，配置容易，在七层之上利用HTTP协议就可以。这时是第一选择。

第二阶段：随着网络服务进一步扩大，这时单点的Nginx已经不能满足，这时使用LVS或者商用Array就是首要选择，Nginx此时就作为LVS或者Array的节点来使用，具体LVS或Array的是选择是根据公司规模和预算来选择，Array的应用交付功能非常强大，本人在某项目中使用过，性价比也远高于F5，商用首选，但是一般来说这阶段相关人才跟不上业务的提升，所以购买商业负载均衡已经成为了必经之路。

第三阶段：这时网络服务已经成为主流产品，此时随着公司知名度也进一步扩展，相关人才的能力以及数量也随之提升，这时无无论从开发适合自身产品的定制，以及降低成本来讲开源的LVS，已经成为首选，这时LVS会成为主流。

最终形成比较理想的基本架构为：Array/LVS — Nginx/Haproxy — Squid/Varnish — AppServer。

小礼物走一走，来简书关注我

赞赏支持





薛晨 (/u/a2d715858c25)

写了 46873 字，被 160 人关注，获得了 187 个喜欢

(/u/a2d715858c25)

+ 关注

coding, piping iron, reading, traveling

(/apps/
utm_sc
banner

喜欢 | 101







更多分享



下载简书 App ▶

随时随地发现和创作内容



(/apps/redirect?utm_source=note-bottom-click)

被以下专题收入，发现更多相似内容

- 

javaweb收藏 (/c/ea04a7b5cec3?utm_source=desktop&utm_medium=notes-included-collection)
- 

HTTP (/c/f900940632f9?utm_source=desktop&utm_medium=notes-included-collection)
- 

分布式、微服务架构 (/c/87a0585b088f?utm_source=desktop&utm_medium=notes-included-collection)
- 

网络 (/c/f98b16942ddb?utm_source=desktop&utm_medium=notes-included-collection)
- 

互联网架构路途 (/c/8172a0fe641c?utm_source=desktop&utm_medium=notes-included-collection)
- 

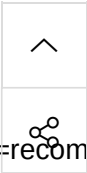
架构设计 (/c/e84d216fb043?utm_source=desktop&utm_medium=notes-included-collection)
- 

互联网通用技术 (/c/cd8ddb694f4c?utm_source=desktop&utm_medium=notes-included-collection)

(https:/
click.y
slot=3(
c7ae-4
ba75ck


展开更多 ▾

(/p/8f7242cbf469?
utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommend
负载均衡架构 (/p/8f7242cbf469?utm_campaign=maleskine&utm_conte...



开源软件负载均衡方案比较 (/p/c322701da246?utm_campaign=maleskin...

Lvs、Nginx、Haproxy都可以用作多机负载的方案，它们各有优缺点，在生产环境中需要分析实际情况并加以利用。从性能和稳定上还是LVS最好，基本达到了F5硬件设备的60%性能。 一、lvs的优势 1、抗负载能...

 303Donatello (/u/586e4e6549d5?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation_banner

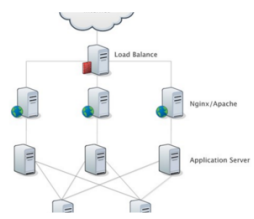
lvs、haproxy、nginx 负载均衡的比较分析 (/p/184243e36318?utm_campaign=maleskin...

本文转自：http://blog.csdn.net/gzh0222/article/details/8540604 对软件实现负载均衡的几个软件，小D详细看了一下，从性能和稳定上还是LVS最牛，基本达到了F5硬件设备的60%性能，其他几个10%都有点困难...

 softgg (/u/d2de29a0c624?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation_banner

(/p/16e9c84fdb3c?)



(https://
click.y
slot=3
c7ae-4
ba75ct

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation_banner

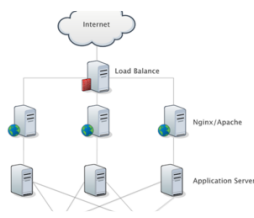
LVS、Nginx、HAProxy、keepalive 的工作原理 (/p/16e9c84fdb3c?utm_campaign=maleskin...

当前大多数的互联网系统都使用了服务器集群技术，集群是将相同服务部署在多台服务器上构成一个集群整体对外提供服务，这些集群可以是 Web 应用服务器集群，也可以是数据库服务器集群，还可以是分布式缓存...

 jiangmo (/u/de31051e96e1?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation_banner


(/p/fc268327d299?)



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation_banner

分布式架构实践——负载均衡 (/p/fc268327d299?utm_campaign=maleskin...

分布式架构实践——负载均衡 也许当我老了，也一样写代码；不为别的，只为了爱好。 1 什么是负载均衡 (Load balancing) 在网站创立初期，我们一般都使用单台机器对台提供集中式服务，但是随着业务量越...


 Bobby0322 (/u/99ee95856388?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation_banner

不要给我谈什么灵魂 (/p/68ae07efdc81?utm_campaign=maleskine&utm...

一个人在家窝了2天整，写毕业论文。实在是憋不住了，就下楼去隔壁超市闲逛。去之前，心里就想好了：

咱是散心，不是购物，所以只看不买。 到了。 皮鞋一双1万9千日元（合1100人民币），钱包一个1万5千...


 Nn米粒 (/u/849caaf91fa9?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

(/apps/
utm_sc
banner

血钻 (/p/33ecd66807e4?utm_campaign=maleskine&utm_content=note...


这个世界上， 每每都是人性的检验真理的时刻。 我们往往把握不住对与错的界点， 直至后来慢慢失去了一点点的原则。 树叶的浮萍归根的美丽， 一刹那的烟火。 出口的一盏灯光， 指引我们继续向前的力量。 一...

 阿俊xi (/u/7709a1102462?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

走近鼓浪屿 (/p/ce2ce9c194b5?utm_campaign=maleskine&utm_content...

几天的学习结束了，最后一天，大家兴高采烈，准备第二天一大早去鼓浪屿游玩。第二天天不亮，我就听到手机在滴滴响着，打开一看，鼓浪屿轮渡中心发布最早信息，因风雨，鼓浪屿全线航班全部停航。一丝淡...

 d冬日暖阳 (/u/a31fcab4060b?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommend

(https:/
click.y
slot=30
c7ae-4
ba75ck

