

Table of Contents

介绍	1.1
tinc 简介	1.2
基础知识	1.3
实例	1.4
进阶	1.5
结束语	1.6

作者序

本手册是一本关于 **tinc VPN** 的实操指南，记录了作者几年来使用 **tinc** 的经验、教训和建议，可作为 **tinc** 入门参考和快速指南。

谨以此书献给我的家人，感谢一直以来对我的容忍和支持。我不擅长表达，在这里说声谢谢，或许未来他们能够知道。

在工作和生活中，我有将不同地点和网络环境的设备连接在一起，形成虚拟专有网络（VPN）的需要：

希望能进行异地之间数据的交换（远程办公，远程访问家庭NAS，远程备份.....）；
希望安全的连接，不用担心隐私和数据的泄露；
希望能简化安全和网络配置，不用考虑各台设备所处环境的区别；
希望尽可能的降低搭建和运维成本，最好是折腾完了就扔一边不用管；
希望尽可能的简单快捷，在下不是IT科班出身，太深奥的东西理解不了；

经过比较和测试，最终选择了 **tinc** 作为虚拟专有网络的实现工具。目前搭建的几个 **tinc** 网络已经稳定运行了超过3年+。在实际操作中，遇到了不少问题踩了不少坑，相对小众所以资料也不多。因此我将相关的经验和文档整理了一下，形成本手册，作为个人使用 **tinc** 的总结和备忘。同时也公开分享出来，如果能为您答疑解惑，将不胜荣幸。

本手册针对非专业人士，重点在于实操。作为读者，您不需要是专业IT人员，但需对网络有基本的了解并有一定的动手能力。本手册重点在于如何快捷的使用 **tinc**，而不是对其原理和源码深入的剖析和讲解。作者试图让您在读完本手册后，拥有快速搭建和管理 **tinc** 网络的能力。换句话说，如果完成一件事有很多种方法，本手册只会告诉你一个经过作者验证和总结的方法（可能最笨但确保实用），不会教你“茴香”的“茴”有九种写法。

从读者反馈来看，一个具有网络使用和电脑使用经验的文科生在半小时内搭建完成了属于自己的 **tinc** 网络。如果您遇到了困难和麻烦，那一定是本手册写的还不够清晰明了，请在 **issue** 中提出，将会逐步修改和完善。

注：

- 本手册选用 **tinc** 当前最新稳定版1.0.x；
- **tinc** 是一种 **VPN** 实现技术，**VPN** 是虚拟专有网络的简称，不是科学上网不是番羽羽土薷；
- **tinc** 相对小众，如果您选用那么记得最终需要负责的，是您 :)；

相关内容

- 本书仓库：<https://github.com/Chanix/TincCookbook>
- **tinc** 官网：<https://www.tinc-vpn.org>
- **tinc** 源码仓库：<https://github.com/gsliepen/tinc>

转载注意事项

除注明外，所有文章均采用 **Creative Commons BY-NC-ND 4.0**（自由转载-保持署名-非商用-禁止演绎）协议发布。

你可以在非商业的前提下免费转载，但同时你必须：

- 保持文章原文，不作修改。
- 明确署名，即至少注明 作者：Chanix 字样以及文章的原始链接，且不得使用 `rel="nofollow"` 标记。
- 商业用途请点击最下面图片联系本人。
- 微信公众号转载一律不授权 原创 标志。

tinc 简介

tinc 是一个组建虚拟专用网络（VPN）的工具，以 GNU 协议发布并开源，通过隧道及加密技术在互联网上点与点之间创建专有网络。使用 tinc 组网，无论何时何地，只要能连入互联网，就可以直接访问操作 VPN 中的设备，同时网络通讯经过加密和压缩，安全性也有提高。

对于我来说，选择 tinc 有以下几个理由：

- 开源，截止目前还在不断更新完善；
- 分布式网状路由，避免传统 VPN 方案里中心服务器高负载和单点故障；
- 占用的系统资源小，适用资源受限的设备；
- 对于应用来说不需要额外关注，能感受到的就是系统里多了个网络设备(虚拟网卡)；
- 通讯支持加密、压缩，支持参数选择；
- 内网穿透（NAT）简单方便；
- 支持常见的操作系统、网络协议（ipv4+ipv6），适用广泛；
- 支持桥接以太网；

分布式网状路由

与传统VPN方案比较，tinc最大的特点是分布式的，基于网状结构的网络路由。基于P2P的技术，可以实现流量直接到达目标机器，而不像传统VPN 那样必须经过中间的服务端。

支持的系统、平台、协议

支持大部分常见平台与操作系统。包括 Windows、Mac OS X、Linux、安卓、iOS、OpenWrt等。详细支持列表可以参见官网，点这里打开 <https://www.tinc-vpn.org>。

加密/认证/压缩

tinc 网络中数据通讯加密可靠，同时，可以通过配置的方式按需进行定制。

内网穿透（NAT）

支持各种常见网络环境下的内网穿透。

多种模式

支持 router（默认）、switch、hub 三种模式。其中 router 模式应用的较为广泛，router 模式对应路由器，仅转发IP报文，因此效率相对也较高。switch 模式对应交换机一般应用于将多个以太网网络桥接。hub 一般不使用了。

如何使用

对于普通使用者来说，了解下面的步骤就足够了： 初步设计

- [安装](#) 对于普通使用者来说，各个平台都有预编译好的程序，直接下载安装即可使用。在后面的实例中，说明了各个平台的详细安装步骤。
- [配置](#) 这里算是重点，如何配置？
- 配置文件

tinc 是个对等的网络，没有server和client之分，每个连接的设备都是一个节点。各台机器上的配置文件都差不多。配置文件一般是一个目录，大概的目录结构是酱紫的（括号里面的是注释和说明）：

1. |—— NETNAME_1 （tinc 网络的名称，在启动 tinc 的时候作为标识）
2. |—— hosts （放置本节点要连接到的各节点的基本配置信息）
3. | |—— HOST_1 （节点1的配置文件）
4. | |—— HOST_1-up （节点1连接上时会调用执行的脚本，可选）
5. | |—— HOST_1-down （节点2断开上时会调用执行的脚本，可选）
6. | |—— ...

- 7. | |—— HOST_n （节点n的配置文件）
 - 8. | |—— HOST_n-up （节点n连接上时会调用执行的脚本，可选）
 - 9. | |—— HOST_n-down （节点2断开上时会调用执行的脚本，可选）
 - 10. |—— rsa_key.priv （由 tincd 生成的本节点使用的非对称私钥，加密通讯使用）
 - 11. |—— tinc.conf （本 tinc 网络的配置文件）
 - 12. |—— tinc-down （关闭该网络时，会调用执行的脚本，可选）
 - 13. |—— tinc-up （启动该网络时，会调用执行的脚本，可选）
 - 14.
- - [tinc.conf 配置文件说明](#)
 - [host 配置项目说明](#)
 - [生成秘钥，传递公钥](#)
 - [运行软件](#)
 - [检查生效](#)
 - [完成](#)

实例

结合实例，跟我一起组建一个自己的tinc网络。这是作者日常使用的tinc网络。详细说明了作者是如何开始设计和搭建一个tinc网络。包括了目前可以看到的大多数网络环境 and 应用场景。读者可以按照自己的需要，从中截取一部分。

Tinc 有两个大版本。1.0为稳定版，1.1开发版。核心功能1.0版都有了，具体的差别参见官网。本着生产系统稳定优先原则，选择1.0版。

tinc支持两种网络模式，switch（交换机）和 router（路由），默认路由。用交换机模式，可以将多个以太网段连成一个以太网，可以DHCP什么的。对于我来说，宁愿手工指定所有机器的IP配置，这样比较可控，通讯报文也比较少。所以，选择默认的 router 路由模式。

需求，互联 决定 router模式 使用现成的预编译版本 使用默认的加密强度 使用增强的带宽压缩

1、安装一台核心tinc主机。这台主机需要具有公网ip，打开tinc使用的端口，外部能直接访问。对于tinc网络中的其他设备而言，只要能上网，能联通这台核心主机的对应端口，就可以联通tinc网络。

可以购买云主机/VPS 或者具有公网IP的家里的路由器也可以。出于简单稳定的需要，我选择云主机。

我的选择，买一台云主机用以运行tinc。不需要太高的配置，个人普通使用1核1GB内存20G普通云盘就可以了。带宽方面按照自己的流量来选择，一般有按使用付费和固定带宽两种模式。我喜欢简单的付费模式，带宽要求也不大，因此选择的是1M固定带宽。这样的配置，目前阿里云的价格是人民币65左右，一次性时间长一点付费还有优惠。我是一次性付了10年。然后扔那里慢慢用就好了。

安全一点，两台做个备份好一些。但实际使用中，本身这台机器的负载就不大，而且阿里云提供99%以上的可靠性。基本不可能出问题，即使出了问题，只要配置文件备份好了，重新安装也就是半小时的事情。又不是什么大不了的环境，没必要杀鸡用牛刀。

具体的配置文件：

版本和模式：

复制代码

注：某些系统可能会有点差异，但差不多就这个意思，万变不离其宗，各位自己融会贯通一下。

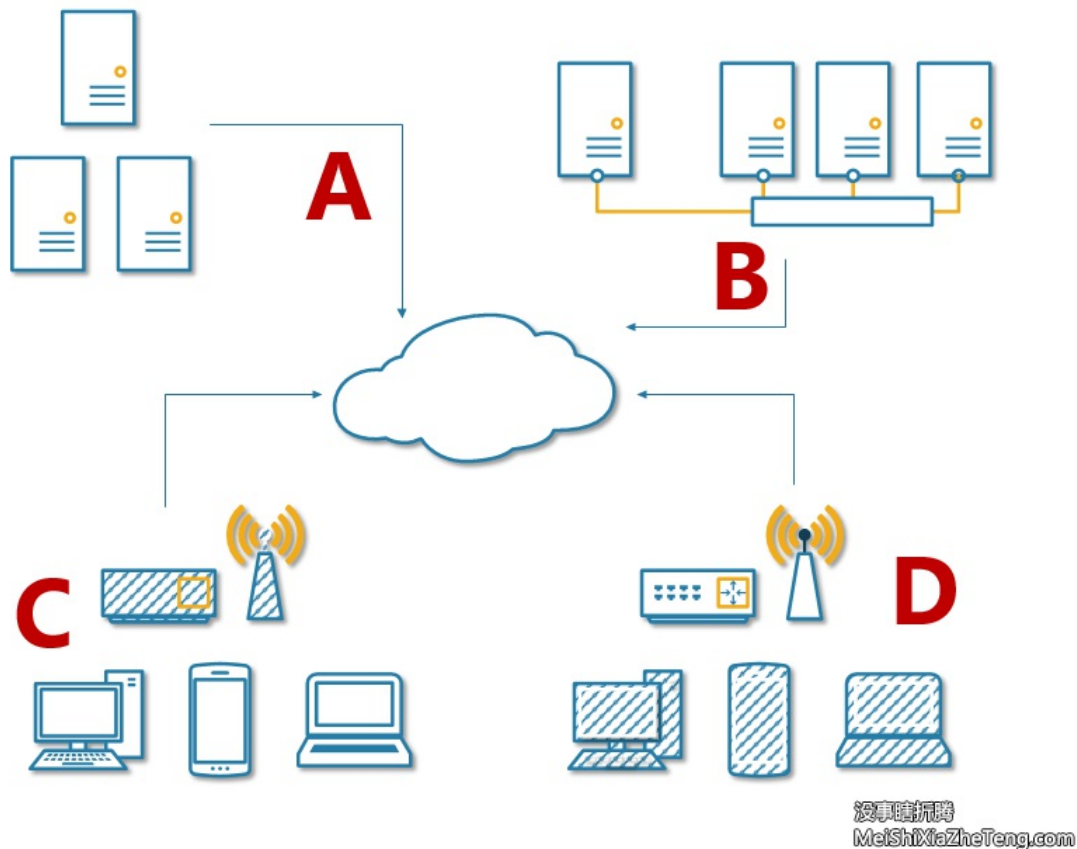
实际操作：

基本需求：

将目前常用的机器用 tinc 组成一个VPN，虚拟专用网络命名为 vnet34。VPN 网络地址为 10.10.0.0/255.255.0.0（我贪方便用的B类地址，其实一般来说C类够用了，255个呢）。

网络拓扑：

我整理了一下我需要处理的情况，如下图：



A 互联网上的机器，绑定了公网IP，可以直接访问公网。tinc 网络里的其他设备需能直接访问这些设备，需要在每台设备上安装 tinc。

B 互联网上的机器，没有绑定公网IP。可以访问公网、但外部无法直接访问该机器。tinc 网络里的其他设备需能直接访问这些设备，需要在每台设备上安装 tinc。

C 位于路由器后面的设备，tinc 网络里的设备需能直接访问这些设备，需要在每台设备上安装 tinc。

D 位于路由器后面的设备，tinc 网络里的设备无需直接访问这些设备，设备上不需安装，只需在路由上安装 tinc 即可。

C 和 D 有点象，在实际操作中，可以是混用的。比如，我的工作机器上跑了不少虚拟机需要被直接访问，就在这些机器上都安装了 tinc。而对于有些人来说，他们只需要能访问 tinc 网络就可以了，还特喜欢瞎折腾机器（时不时的重装一下啥的）。我要是在他们的机器设备上安装 tinc 那真是烦透了。这种情况，在路由器上装上，让他们通过路由透明代理到 tinc 网络就可以了。

基础知识

本手册并不想详细说明，作为 **tinc** 的使用者，您只需要知道下面的大概就可以了。

原理，怎么用，什么步骤

技术实现

虚拟网卡，操作系统中的一种TAP/TUN设备，是模拟出来的一种网卡。可以类比您机器上安装的实体网卡。**tinc**基于虚拟网卡进行通讯。对于上层软件应用来说，虚拟网卡和实体网卡并没有什么不同，所以上层软件不需要任何修改和适配。

所以要使用，首先您的OS要支持TAP/TUN（目前常见的系统都支持）。第二要运行**tinc**。第三，画个图说明**tinc**网络中数据是怎么传输的，是怎么建立连接的。

可以跳过本章节，直接看实例，然后反过头来看这个。下面说的比较详细，普通用户可以不用看，看后面实例就足够了。

tinc是怎么运行的？各种脚本的执行如何按次序执行？都支持哪些脚本？

How2use

- [安装](#)
- [tinc配置](#)
 - [tinc.conf 配置文件说明](#)
 - [host 配置项目说明](#)
- [生成密钥，传递公钥](#)
- [运行软件](#)
- [检查生效](#)
- [完成](#)

实例

这里放实例：

- 1、我的家庭网络
- 2、公司办公
- 3、服务器互联

```
* [Windows](todo.md)
* [Ubuntu 14.04](todo.md)
* [Ubuntu 16.04](todo.md)
* [Ubuntu 18.04](todo.md)
* [Mac OS X](todo.md)
* [CentOS](todo.md)
* [群辉](todo.md)
* [OpenWrt](todo.md)
```

advanced

结束语

写完了，没啥说的。有啥意见和建议放**issue**里吧。
祝您好运。