从xcode说起——开发中的那些傻逼事

发表于2015/09/21

最近xcode很热——我当然不是说软件本身。在工具链里埋炸弹是一个经典手法了，我还以为从putty之后大家应该已经提高警惕了呢。

下面我来数数开发中的一些SB行为。排名不分先后。

下载用百度，下载完成不做**MD5**

请问你作的是死么？

聊天用**QQ**，公司密码群里发

下面我要发密码了，请其他同事把头转过去一会。。。

最讨厌的就是这种，自己密码也不简单，乱发，漏了。企业绝对怀疑不到个人头上。但是搞得安全形同虚设。

共享帐号/密码

下面我们三十个admin集体开个会，讨论一下我们中出了一个叛徒的问题。。。

相关密码扔邮箱

结果有一天，你的mail密码不慎漏了，而你不知道。。。

乌云2015大会上特别提到的case。

弱密码

每个公司总有那么几个人，天天用弱密码，说了不改。

安全组不能做主开除这些人的公司，不要妄称安全。

git提交时根本不知道提交了什么

结果把密码提交上去了有木有？

自己去搜一下，这个问题非常常见。

基础工具链交给实习生维护

“高级”的“架构师”当然要关心最有价值的“核心”内容。

院长不亲自操刀手术，不代表一线可以交给实习生好吧。

只是很多公司招不到人，就拿实习生或者资历比较浅的人去凑合了。实话说，出问题只在早晚。

我只是调试，一会就关

我很想知道你和老婆在床上的时间能不能比“一会”更长。还是说你是“一会就完事”的那一号？

凡是在一次约会时间里搞不定的事情。。。那就不是一会。。。

无套一时爽，XXxXX。。。

公司电脑不加密

笔记本或者U盘不小心丢了会上新闻的人，在公司里占了大多数。。。区别只在于，上科技新闻，娱乐新闻还是联播新闻。。。

如果需要在存储上保存保密内容，请先对存储本身加密。更安全的，你需要对整个系统做签署。硬盘加密保护的是数据机密性，系统签署保护的是可信环境，两者相辅相成。

PS: 很多人电脑是加密的。手机呢？

工具链用的全是非开源产品，还都是盗版的

X，这年头盗版和免费软件（尤其是国产盗版和免费软件）简直是黑产样本大全。

如果你要用软件，请付费。如果不想付钱，用开源。而且建议简单看一下源码。用免费版的，你自己就是产品。

安全无所谓拉

反正只要不是我故意搞出来的，出问题最多被骂一顿了事。还是关心怎么赚钱升职走上人生巅峰比较重要。

这TM怎么骂？这是极其NB正确无比的人生观，我连下嘴的地方都没有。

发表在其他 | **4 Comments**

使用tmate和mdp做培训

发表于**2015/09/17**

tmate

其实很简单。有个大家都玩过tmux对吧。有个x炸天的工具，叫做tmate，是tmux的分支。地址在这里。

这个工具能做什么呢？开一个tmux的窗口，连接一台服务器，得到一个ssh指令。然后，其他人可以用这条ssh指令直接查看你的tmux，只读或共享操作都可以(我建议只读)。

这个工具进入了debian的testing和sid，但是没有进stable。所以要装一堆依赖编译一下。总之，最后是一个静态的binary文件，复制去bin目录下结束，没什么好废话的。

自建服务器

tmate官方提供了一组公开服务器，供大家使用。当然，这些对我等大局域网居民来说是没用的。所以follow the guide(官方主页上有)，在自己的vps上搭建一个服务器。当然，22端口要改成其他端口。

然后填一个设定文件(也在官方主页上)，你就能使用tmate连接自己的服务器了。fingerprint的扫描方法大概是，在tmate-slave目录下执行：

```
ssh-keygen -lf keys/ssh\_host\_dsa\_key
ssh-keygen -lf keys/ssh\_host\_rsa\_key
ssh-keygen -lf keys/ssh\_host\_ecdsa\_key
```

会得到三个fingerprint，写到客户端的配置里去。

唯一需要注意的是，除了tmate的配置文件里面需要设定三个fingerprint，你还得用ssh连接一下自己的这些服务器，并同意将fp添加到~/.ssh/known_host。不然也会有错误。

然后tmate，就能看到ssh指令了。

客户的连接

在tmate的窗口里，敲`tmate show-message`，你能够看到只读和读写ssh指令。这条指令里会有一个非常复杂的username，这个username是用作客户的身份验证的。虽然信息量并不是很大，但是作为临时的会话共享安全手段足够了。

注意如果你使用只读指令分享了屏幕，那么在操作过程中不应该再敲这条指令。否则别人就能看到读写ssh指令。

后面的事情就非常简单了，客户自己用ssh工具来连就行了。tmate对客户端没有任何要求。

mdp

tmate解决了pair work和屏幕分享的问题。但是要用于培训的话，我们还缺一个slide手段。

mdp是一个命令行的slide工具(当然你也可以用tpp)。基本是基于markdown的。你可以按照官方文档来做一个slide，然后就可以很happy的和小伙伴们分享屏幕操作了。

还缺什么

其实最好还要有一个语音共享机制的，例如YY语音。

性能如何

还没测。我在同组的三个人之间试了一回，非常好用。大量的用户还没测。等测下来再写一篇。

发表在**linux** | **0 Comments**

关于vpn的一些话

发表于**2015/08/06**

最近在弄一些关于vpn的事，又在quora上看到了好多关于vpn的问题。（不知为何quora总是推荐让我回答vpn的问题）其中很多问题极其傻，缺少基础性常识，一看就是外行问的。

其实也难怪，vpn的需求，并不只是专业人士有。有位朋友向我咨询vpn问题，他可既不是要翻墙，也不是专业人士。所以我打算把和vpn有关的一些常识写一下，以备咨询。

vpn能做什么？

vpn能做什么，取决于你想要他做什么。vpn其实就是一个虚拟的线，连通两个地点，就如同真的接了一根线一般。只不过这个虚拟的线，实际上是由你到对方地点的网络来提供的支持。通常情况下，这么传数据会使得你的数据暴露在网络上。但是vpn里传输的所有数据都经过了加密，你可以认为传输者看不见。

所以，vpn能做什么呢？

有些公司阻止了员工访问很多网络。借助vpn，你可以绕到第三方的网络里去访问哪些网站。其实gfw也是同类情况，只是这家公司更大而已。

有些网络会监听访问，用来做一些对用户不利的事情。例如在国外，使用bt下载很可能会

招致版权组织的诉讼。这时可以借助vpn来下载。或者某些公司也会监听用户的数据，例如qq。对于这类情况，你也可以用vpn跳出公司的网络。当然，如果你对这些问题已经有了顾虑，那么就不应当使用国产路由器。大部分国产路由器都会有监控用户数据的行为，很多甚至会修改。

vpn不能做什么？

vpn的常规模型，是从用户实际上网的地点“逃逸”到vpn供应商那里。（我们不讨论一些特殊情况）所以，他解决不了一些问题。

例如你的网站没有加密，那么就无法期待vpn来帮你。除非你能让你的所有用户和网站全部连到同一个vpn上（所以访问你的网站前需要先拨vpn）。基于同一个理由，你无法用vpn对抗网站劫持。

同理，如果某个网站未加密，你也无法指望使用vpn保证安全。vpn只能保证守在你家门口的这些人（例如ISP）无法弄到数据。守在服务器门口的人（如果有的话）依然可以获得你和服务器的全部通讯。

vpn分为哪些种类，有什么特点？

一般来说，常见的vpn有这么几种。

- pptp。最古典和最通用的vpn。windows里默认内置，搭建和使用都相当容易。但是由于特殊的网络设计，因此有些公司无法使用。而且有安全性隐患。对掌握极大资源的攻击者，内容基本透明。
- ipsec+l2tp。windows里内置，大部分公司应当都可以连通（没有故意拦截的话）。对于除美国政府外的截听者，应该都比较安全。
- openvpn。复杂和强大，模式多变。需要安装第三方程序，因此并不是很容易用。但是可以跨越大部分公司网络，不会有什么阻碍。安全性很高，也有一些算法可以用来对抗美国政府。由于中国政府的封锁，无法跨越国内外。
- sstp。windows自己的协议。只需要一个ssl连接，因此跨越性比openvpn还好。新版windows里应该有内置。
- AnyConnect。Cisco的协议，需要安装第三方程序。跨越性和安全性没有实用数据。

简单来说。如果你只是要用而已，并且没有被掌握极大资源的攻击者盯上，而且不是反美国政府。那么哪种方便用哪种。下面是一些系统的兼容性建议：

- windows: pptp, ipsec+l2tp, sstp
- linux: openvpn, pptp
- android: pptp, ipsec+l2tp, openvpn
- ios: pptp, ipsec+l2tp, anyconnect(似乎这是唯一一个能在ios上非越狱安装的vpn应用)

从上面我们可以看到，pptp其实是兼容性最好的，但是不是所有网络都支持。次之的是ipsec+l2tp。如果还有问题，openvpn应当能够解决你的问题。如果都不行，再考虑其他。sstp仅建议用于只有windows客户端的情况，anyconnect仅建议用于你钱足够多的情况。

所以大部分vpn供应商的协议选择都是pptp/ipsec+l2tp，或者多一个openvpn。这足以应付大部分情况。

这里是我看到的一个比较全面的，关于vpn之间比较的页面。

使用了**vpn**，我的网络安全了么？

不一定。如我上面所说，vpn只保证了你家门口的安全。守护在服务器前的人依然可以获得数据。甚至，如果vpn供应商怀有恶意的话，他们也能够获得数据。所以这是一个ISP和VPN供应商，谁更可信的问题。

而且实际情况往往更加复杂。很多数据的获取，并不来源于源IP地址，而是在浏览器里植入了身份相关信息。浏览器的身份相关信息相当敏感，没有他们，你就无法在网站上登录。也因此，这些数据一旦被恶意者获得，他们就能借助你在第三方网站上的身份信息，获得你是谁。

对于这些威胁，vpn都是无能为力的。

发表在**network** | 标签有**VPN** | **2 Comments**

---

ipv6试用手记

发表于**2015/08/05**

ipv6地址

概要

挑简单的讲吧。

ipv6地址总计128位，分为8个段，每个段16位。hex表示的话，每段最多有四位。在写出ipv6地址的时候，用:分割。所以一个经典的地址写出来是这个样子的：

aaaa:bbbb:cccc:dddd:eeee:ffff:gggg:hhhh

这里有两个缩写。一，如果一个数字的开始有连续的0，可以忽略。好比0001和1是一样的。二，如果有多个连续的段是0，可以缩写为::，但是只能缩写一次（这样才能确定一个唯一的地址）。例如以下是一个缩写过的地址。

aaaa:b:c:d::1

这个地址等同于

aaaa:000b:000c:000d:0000:0000:0000:0001

ipv6地址分类

::/128 未定义 ::1/128 回环地址(127.0.0.1) fe80::/10 局部地址 ::ffff:0:0/96 ipv4映射地址 ff00::/8 广播地址(其实更小，不过我的路由表里就是这样写的，全保留给他了)

更多地址请去看wikipedia。

EUI-64地址

简单来说就是通过mac地址的48位，扩充到64位，作为ipv6地址的最后64位。

ipv6地址分配和基本网络设定

ipv6下有两种ip自动分配方案。通常都建议无状态的方案。

dhcp

一种还是经典的dhcp，这种方案分配出来的地址比较紧凑连续，空间利用率高。坏处是ip地址并不唯一固定。在多数情况下dhcp都会尽力分配和上次一样的ip下来。但是在地址池比较满和系统缓存被清理的情况下，并没有保证。

在interfaces里，dhcp的写法大致如下：

iface ethX inet6 dhcp

slaac

另一个是嫌地址太大的slacc。基本原理就是给一个组织（例如家里）分配一个/64的段，然后用EUI-64(似乎是)给里面的所有设备分ip。所以这样分出来的ip完全一致。

在interfaces里，slaac的写法大致如下：

iface ethX inet6 auto

dns

ipv6里进行配置还需要注意一点，你的DNS需要支持AAAA记录。不然拿着网址查出一个ipv4记录来就很尴尬，还得退化到ipv4去访问。这和直接使用ipv4没什么区别，反而更糟。

实际上，大部分DNS都支持AAAA记录，只是程序会不会默认去查而已。

另一个问题是，如果你的DNS服务器地址是ipv4的，也会使得你无法完全脱离ipv4网络。因此需要一个有ipv6地址的DNS。一般你的ipv6链路供应商会提供一个ipv6的DNS。

firewall

slaac使用icmpv6，所以需要在防火墙上打开icmpv6协议的进入。

tunnel breaker

he提供一个6to4的tunnel，但是需要你有静态ip地址。(其实凭心而论这真不是一个太高的要求，很多VPS供应商完全可以直接搞一个48的段给自己的机房分64的段)作为普通用户，没有静态地址怎么办？那就只有用vps先接到tunnel，然后再tunnel回家了。

tunnel to he

很容易，配一根6to4的tunnel就好。tunnel breaker的网站上还很贴心的提供了不同系统下的详细配置。

这里特别提一下。HE很贴心的提供了两段地址。一段ipv6是你和HE的隧道地址。另一段地址也会路由给你，这才是你内部应当使用的ip地址群。所以原则上说，你可以在隧道中使用(本来应当在)你和HE之间使用的地址的部分。

tunnel to yourself

先随便打一根二层的隧道。

二层的理由是，如果使用三层隧道，那么隧道本身就需要察知ip。于是ipv6的支持就变成隧道的事了。而二层的隧道并不需要知道上面跑什么协议。

最简单的当然是gre隧道。但是gre隧道需要在远端确定本地ip地址，这和直接打一根tunnel回家没什么区别。

我用的方法是用任何一种二层或三层vpn打到vps上，然后上面再跑gre。这样一举解决了加密和静态地址的问题，顺便还解决了ipv6兼容性的问题。当然，代价也很高。由于是在vpn里套gre，所以头部很大，mtu就要开的比较小。

另一个玩法是用ipsec的tunnel模式打通两个网段，于是gre也可以直接通到vps上（甚至可以直接打ipv6 tunnel）。然而ipsec的tunnel模式需要知道双端ip地址，所以其实还是没有什么用。

顺便吐槽一下routeros的ipsec，实在是太废物了。

当然，也可以用支持ipv6的三层隧道。PPTP都有ipv6支持。当然，我看了一眼，要把ipv6 tunnel回来还是有点问题的。

无论怎么配置，这根隧道要用你和HE之间tunnel的部分地址。例如你和HE的tunnel是这个样子：

a:b:c:d::1/64 <-> a:b:c:d::2/64

那么你可以将原来的/64改为/112，然后配置这么个tunnel地址：

a:b:c:d::1:1/112 <-> a:b:c:d::1:2/112

route

在vps上，需要将routed addresses指向内部的路由器。路由器需要将default路由指向vps。default指过去后其他细节就不用管了。

MTU

由于在两层隧道内跑的ipv6，因此mtu记得调整一下，否则有性能问题。

firewall

防火墙是个大头，所以要单独提出来说。

原则上说，ipv6的所有地址都是外部可达的。因此如果你将路由器上的forward关闭，那就没有使用ipv6的意义了。然而，如果forward打开的话，那么每一台都真实的暴露在公网上了。因此每台必须都配置防火墙，否则就可能有安全问题。

例如通常内网会打开ssh，并且不会打开防扫描，或者做安全加固。如果打开了ipv6，又没有在路由器上关闭forward，那么就会造成这个端口对全世界开放。虽然原则上说，没有人会扫描ipv6（因为一个家里的地址比全世界的ipv4还大）。但是这并不安全。因为会有人从你对外的访问地址看出你的内部机器ip。

所以我的建议是，关闭forward，只对特定地址打开。而这些地址上，都需要保证配置了ipv6防火墙。这样即使不慎接入了一个设备，没有开防火墙。也不会造成安全隐患。当然，缺陷就是，随着接入设备的增多，你的地址列表增加很快。

reference

- routeros ipv6
- wiki IPv6

发表在**network** | **0 Comments**

程序员交友选择题

发表于**2015/07/21**

问题

1. 你用过lamp么？
2. 你键盘最左边一列磨损最严重的键是？
3. 你喜欢3p么？为什么？
4. 如果你需要改一张图，电脑里又没有装有关软件，你会怎么做？

答案

1. 没有的就算了吧。
2. caps lock万岁，其他去死。

3. python万岁，其他去死。
4. 首先从可信源下载编译好的包，如果没有下载源码安装。
    - （去商店购买一份ps的去死）
    - （去下载盗版ps的立刻拨打110）

发表在**program** | **26 Comments**
