VPN con Tinc

Tinc es un software de código abierto para crear redes privadas virtuales (VPN), las VPN a través de otro canal físico como Internet, donde los nodos (hosts) participantes individuales aparecen en las aplicaciones como si estuvieran conectados por cable en LAN.

Visión general

Tinc utiliza criptografía asimétrica. Cada nodo tiene su propia llave privada, una llave publica y otra llave publica; para cada participante del nodo. Estos archivos son, junto con algunos archivos de configuración, almacenados en el directorio /etc/tinc/<nombre de la VPN>.

Cada nodo también corre como un demonio (o múltiples demonios, uno para cada VPN por separado). Los demonios escuchan sobre el puerto configurado (por defecto es 655) para conexiones entrantes desde otros nodos. Solo los nodos con claves privadas validas pueden producir datos descifrables con claves publicas que coincidan y de esta forma se concede el acceso.

El archivo de clave pública puede contener no solo la clave en sí, sino también la dirección IP pública (y el puerto) del nodo al que pertenece. Si se configura, el demonio no esperará las conexiones, pero intentará conectarse a estos nodos conocidos.

Cada nodo tiene su propia dirección IP (en el espacio de direcciones privado) que, una vez que el demonio se está ejecutando, se asigna a la interfaz de red virtual. Cualquier tráfico proveniente de la VPN es procesado por el demonio y proviene de esa interfaz de red, y cualquier tráfico enviado a través de esa interfaz también es procesado por el demonio y enviado a la VPN, de forma totalmente transparente para las aplicaciones. Una característica importantes de Tinc es que el demonio puede (y por defecto lo hace) reenviar tráfico para otros nodos, por ejemplo, si los nodos A y B están detrás de NAT y pueden comunicarse directamente solo con el nodo C, que tiene acceso irrestricto a Internet, o incluso no saben la clave pública de cada uno, pero C los conoce a ambos, C felizmente reenviará el tráfico entre ellos / para ellos.

Sólo necesitan saber las direcciones IP (en el espacio de direcciones privadas).

Instalación

- Descargar fuentes de http://www.tinc-vpn.org/download/
- Descomprimir y compilar.

```
# ./configure --prefix=/usr --sysconfdir=/etc --localstatedir=/var
# make
# make install
```

- Si prefiere tener Tinc en algún lugar aparte, cambie el prefijo. Pruebe DESTDIR si desea instalar temporalmente en otro lugar:

```
$ make DESTDIR=/somewhere/else install
```

Configuración

- Cree la configuración inicial (también genera claves privadas y públicas para el nodo), lo que significa que el directorio en /etc/inc lleva el nombre de VPN con algunos archivos iniciales.

```
# tinc -n VPNtest init node1
```

- O genere sólo llaves pares privadas/publicas. Ubique las llaves privadas en /etc/tinc/<Nombre de la VPN>, fusionar las claves públicas en un archivo y colocarlas más adelante en un subdirectorio /etc/tinc/<nombre de la VPN>/hosts.

```
$ tinc -c . generate-keys
$ mkdir -p VPNtest/hosts
$ mv *.priv VPNtest/.
$ cat rsa_key.pub ecdsa_key.pub > VPNtest/hosts/node1
$ rm rsa_key.pub ecdsa_key.pub
```

- Ajuste de la configuración en /etc/tinc/<VPN name>/tinc.conf. Saltar el parámetro ConnectTo si el demonio debe esperar pasivamente las conexiones. El nombre de la tarjeta de red virtual es configurado en el parámetro Interface, ver más abajo. Opcionalmente, configure el puerto de escucha, especialmente si desea ejecutar varios demonios/VPNs.

tinc.conf

```
Name = node1
ConnectTo = node2
Interface = vpnNIC
Port = 6655
```

- Configure interfaces de red virtuales en /etc/tinc/<VPN name>/tinc-up. No cree manualmente interfaces (via el comando ip), el demonio Tinc puede hacer esto por usted, simplemente escriba la configuración para el nivel de IP. Además, haga que el archivo de tinc-up sea ejecutable.

tinc-up

```
#!/bin/sh
ip addr add 192.168.1.1/24 dev vpnNIC
ip link set vpnNIC up
```

- Ajuste el archivo de claves publicas en /etc/tinc/<VPN name>/hosts/<este nodo>. La IP pública puede ser también un nombre de host/dominio, lo cual es conveniente en caso de que, por ejemplo cambie de ISP, pero que conserve el nombre de DNS. El puerto debe ser el mismo que en tinc.conf, pero puede diferir si por ejemplo está detrás de NAT con reenvío de puerto a otro número de puerto. Deje que otros nodos tengan este archivo y coloque sus archivos de clave pública aquí.

node1

```
Address = <public IP address> [port]
Subnet = 192.168.1.1/32
-----BEGIN RSA PUBLIC KEY-----
...
```

- Repita el proceso sobre (o por) otros nodos, use diferente nombre para los nodos y diferentes espacios de IPs. Nuevamente, permita que los nodos tengan el archivo de clave pública (o host) del otro.
- Inicie el demonio, opcionalmente especifique el nivel de debug (0-5 donde 5 es el más elocuente) y donde se registra en un archivo.

```
# tincd -n VPNtest --debug=5 --logfile=/var/log/VPNtest.log
```

Windows

En aras de la integridad, como por ejemplo, construir una VPN con una máquina Linux como servidor de archivos al que acceda Windows, administrar remotamente un grupo de Windows detrás de NAT de Linux, jugar a juegos, lo que sea, vamos a cubrir también Windows (XP, 7 y 8 son conocidos por su funcionamiento).

Instalación

- Descargar los paquetes binarios.
- Instalar el software Tinc.
- Preferentemente desinstale algún posible dispositivo TUN/TAP (NICs virtuales). Tapinstall es parte del paquete de Tinc, debe estar en su directorio de instalación en alguna parte.

```
C:\path\to\tapinstall.exe remove tap0901
```

- Instalar nuevos dispositivos TUN/TAP device.

```
C:\path\to\tapinstall.exe install OemWin2k.inf tap0901
```

- Los controladores de dispositivos en realidad parecen provenir del proyecto OpenVPN. Lo cual es bueno, porque están firmados; últimamente, Windows es bastante hostil hacia los controladores no firmados.

Configuración

Existen unas pequeñas diferencias en la configuración de Windows.

- Debes generar los archivos de configuración inicial, pero ubicalos en donde Tinc está instalado, que debería ser algo como C:\Program Files\tinc\<VPN name>
- En tinc.con, omití las directivas de interfaz, por que el demonio Tinc puede automáticamente seleccionar dispositivos TUN/TAP y las directivas pueden hacer más mal que bien.

- La secuencia de comandos (script) *Tinc-up* no es utilizado sobre Windows. Usted creo un dispositivo TUN/TAP persistente durante la instalación (¿lo hizo?) y ahora solo configuré manualmente la IP (ejecute ncpa.cpl), vea las propiedades del dispositivo, etc).Esto también puede ser automatizado con una secuencia de comandos como por ejemplo:

```
netsh interface ip set address name="Local Area Connection number" static
<IP address> <mask>
```

- Pero tenga cuidado: cuando se crea, dispositivos TUN/TAP puede adquirir cualquier número de nombre, probablemente 2, pero no siempre.
- Finalmente, instale (e inicie) el servicio Tinc:

```
C:\path\to\tincd.exe --debug=5 --logfile=C:\path\to\file.log -n VPNtest
```

- O, si el servicio existe, inicie el servicio:

```
cmd> net start tinc.VPNtest
```

script RC

Aquí hay algunos scripts para iniciar todas las VPNs en el arranque.

```
#!/bin/sh

VPNS=$(ls /etc/tinc)

start () {
    for VPN in $VPNS; do
        echo "Starting tinc daemon for $VPN..."
        /usr/sbin/tincd -n "$VPN" -d1 --logfile=/var/log/tinc."$VPN"
    done
}

stop () {
    for VPN in $VPNS; do
        echo "Stopping tinc daemon for $VPN..."
        /usr/sbin/tinc -n "$VPN" stop
    done
}

restart () {
    stop
    sleep 1
    start
}

case "$1" in
    ("start")
        start
```

```
        ;;
    ("stop")
        stop
        ;;
    ("restart")
        restart
        ;;
    (*)
        echo "Usage: $0 <start|stop|restart>"
        exit 1
esac

exit 0
```

Salve esto como por ejemplo /etc/rc.d/rc.tinc, hacer ejecutable y luego agregar una línea a rc.local.

rc.local

```
/etc/rc.d/rc.tinc start
```

Fuentes

Documentación/sitio web de Tinc http://www.tinc-vpn.org

- Escrito originalmente por: tonberry.
- Traducido por el grupo e-slackware — *rramp 2019/02/24 15:02 (UTC)*

howtos, network