INSTALLATION BONDIX SANE SERVER

The Bondix SANE Server is a universal Linux service for x86_64 architectures (other architectures available on request). Thanks to static compilation, there are no special host operating system requirements such as specific LibC versions - the only requirement is kernel support for virtual tun/tap network interfaces.

REQUIREMENTS

RESOURCES

The resource requirement is based on the peak total throughput of the installation and the number of simultaneous tunnel connections. This formula can be used as a rule of thumb for the required memory:

Memory requirement (megabytes) = Bandwidth(Mbit) / 2 + TunnelCount * 5

Example: An installation of 100 tunnels should guarantee 50 Mbit/sec for each instance at full load. The peak bandwidth would thus be 100 * 50 Mbit = 5000 Mbit/sec. Using the above formula, this results in a memory requirement of approx. 3 gigabytes.

In addition to memory, the number of CPU cores is also critical. Bondix SANE Server distributes incoming tunnels to different CPU cores for load balancing. While the maximum throughput per CPU core depends on the hardware used, 500 - 1000 Mbit can be taken as a conservative estimate.

Note: These assumptions for storage do not take into account requirements for the host operating system, other services, and the like. Requirements for storage space are negligible.

PUBLIC PORTS / FIREWALL

Bondix SANE Server requires a publicly accessible TCP port (default 443, but freely selectable) and at least one UDP port - the number of UDP ports depends on the CPU cores used.

INSTALLATION

Bondix SANE Server only needs to be unpacked from its archive - e.g. to `/opt/bondix/.`

The way to integrate Bondix SANE Server as an automatically starting system service depends on the distribution used. There are various approaches here, such as init.d or systemd, which are not part of the document.

Manually, the software can be done by simply calling it in the shell:

```
/opt/bondix/server/saneserver
```

By default, the software runs in the foreground and outputs event messages there. To start the software in the background the following parameter is used:

```
/opt/bondix/server/saneserver -daemon
```

Event messages appear in the syslog.

CONFIGURATION

Bondix SANE Server will not start successfully without a configuration file. This must be created in `/etc/saneserver.json` and consists of a sequence of configuration commands in JSON representation, which we discuss in more detail here.

MEMORY AND RESOURCES

Connected tunnels reside in so-called tunnel environments that provide shared resources. Specifically, the resource federation is:

- A virtual network interface (tun). Tunnel clients are assigned an IP from the network and can be reached via it.
- Publicly accessible UDP port to send & receive tunnel payload data.
- A cache for incoming and outgoing data. This is needed to resend lost packets if required. This should be 50-100% of the maximum bandwidth, i.e.

Such an instance is created as follows:

```
[{"target": "server", "action": "add-environment", "settings": {
                "receivePacketCacheSize": 50000,
                { "sendPacketCacheSize": 50000,
                "dejitterCacheSize": 0,
                "tunnelNetwork": "169.254.19.1/24",
                "disableTunnelToTunnelTraffic": true,
                "tunnelDeviceName": "bondix0",
                "udpListenerHost": "0.0.0.0",
                "udpListenerPort": "44343"
}}]
```

receivePacketCacheSize	Size of the buffer for incoming data, in packets. One packet has the size of 1400 bytes.
sendPacketCacheSize	Size of the buffer for outgoing packets.
dejitterCacheSize	Size of the dejitter buffer for special applications. A value of 0 disables it.
tunnelNetwork	Internal IP network in CIDR notation that is used for transporting packets. It must be unique per environment and should not be used elsewhere.
disableTunnelToTunnelTraffic	Prevents tunnels from communicating with each other.
tunnelDeviceName	Name of the virtual network interface.
udpListenerHost	
udpListenerPort	

SSL & CERTIFICATEE

The SANE protocol uses SSL to establish connections and can optionally be used to authenticate incoming connections. For this, corresponding certificates must be available, which can be created with scripts in the ssl subdirectory:

`/opt/bondix/server/ssl/create-server-cert.sh`

Generates a self-signed certificate for the SSL server and a root certificate that is used to sign client certificates. The generated server-root.crt should be stored in the client to validate the identity of the endpoint. A backup instance must use the same root certificates.

`/opt/bondix/server/ssl/create-client-cert.sh <name>`

Generates a signed client certificate that can be stored in the client. The name must be unique for each client. Explicit configuration of the client connection in the SANE server is not necessary.

The following command adds a socket with SSL in SANE:

```
[..., {"target": "server", "action": "add-https", "host": "0.0.0.0", "port":
"443",
  "cert":"/opt/bondix/server/ssl/ cert. pem",
"key":"/opt/bondix/server/ssl/key.pem",
"rootCA":"/opt/bondix/server/ssl/client-root.crt", "allowMonitor": false,
"allowTunnel": true}
]
```

host	The IP address to use for the SSL listener. The value "0.0.0.0" makes the service available on all interfaces.
Port	The TCP port to be used for this socket.
cert	The SSL certificate for the socket.
key	The associated private key for the SSL certificate. An additional passphrase is not currently supported.
rootCA	Root certificate to verify incoming client certificates. The parameter is optional.
allowMonitor	Allows or prevents calling the monitoring web interface via this port.
allowTunnel	Allows or prevents tunnel connections to be established via this port.

TUNNEL WITH NAME & PASSWORD

In addition to certificates, SANE tunnels can also authenticate themselves with passwords. For this purpose, a tunnel must be created as follows:

```
[..., {"target": "tunnel", "action": "add", "name": "MyTunnel", "password":
"hunter2"
}]
```

MONITORING

SNMP

Bondix SANE Server can provide information via net-snmp. For this, an existing /etc/snmp/snmp.conf must be extended with the following line:

```
pass_persist .1.3.6.1.3.45265 /opt/server/bxsnmp
```

A description of all values that can be queried via SNMP is included as a "Management Information Base" (MIB) file in the root directory of the server.

WEB

SANE Server can provide a web interface via an SSL socket to monitor individual tunnels. For this purpose, the "allowMonitor" argument must be set accordingly when creating the SSL socket. Alternatively, a dedicated socket can be used for this purpose, which can only be accessed from a protected area, for example.

Example:

```
[...,
{"target": "server", "action": "add-https", "host": "10.0.10.1", "port":
"8080", "
cert": "", "key": "", "allowMonitor": true, "allowTunnel": false}
]
```

Omitting the certificates disables SSL.

To use tunnel monitoring, an additional password is required. This is set globally for all tunnels by default, but can also be set individually for each tunnel.

CONSOLE

Configuration & status of the SANE server can be queried and changed at runtime. For this purpose, the server provides a configuration socket at 127.0.0.1:5112, which is used by the command line utility "bxutil" to query the status of the individual tunnels, among other things.

`/opt/bondix/server/bxutil list`

Displays all known/configured tunnels, each with the number of connected interfaces.

`/opt/bondix/server/bxutil status <tunnelName>`

Displays the status of a single tunnel and its interfaces.

A list of all possible commands can be called by invoking the CLI utility without parameters:

`/opt/bondix/server/bxutil`