

Sepono 1.6.0 Available

Contributed by Nathan
Thursday, 10 July 2008
Last Updated Thursday, 10 July 2008

Sepono 1.6.0 is now available and includes a number of new features, fixes, and general improvements. Highlights include:

- New installer package and external configuration wizard for Windows users
- Modified DWM config to use click-to-focus window selection
- Improved DWM config to intercept less hotkeys
- Added additional user choices for various events
- Streamlined user selection process, made prompts more uniform
- Added open source VMWare tools (Open Virtual Machine Tools)
- Added routines for better revision of saved configurations
- Replaced xterm w/ rxvt to reduce appliance size
- Enhanced vpnc support for NATT methods other than Cisco
-

ABOUT IT:

Sepono version 1.6.0 - a complete virtualized remote access client software stack

Sepono is a standalone VPN enabled remote desktop client. It is intended to supplant existing VPN client software, and act as a complete software stack for safely connecting to remote computing interfaces from unmanaged or otherwise untrusted end-user systems. This is a virtual appliance. It is designed to run under a virtualization platform, preferably VMWare Player (experimental QEMU support is also available). Network Administrators deploying this application in lieu of a conventional VPN application will realize the following benefits:

- Host OS configuration independence – no more testing and troubleshooting VPN application installations on uncontrolled, potentially broken, end-user PCs.
- Controlled, static computing environment – simplify support by knowing exactly how end-users will access your network, and what they will see when they do.
- Segregation (aka sandboxing) – end-user PCs will never need to, or be able to access VPN protected resources which can significantly reduce the risk of virus outbreaks and other forms of incidental intrusion.

End users will love its simplicity, easy of use, and non-intrusive behavior. Everything happens automatically - a user name and password are all that are required prior to being automatically connected to a remote system

Special features:

- Extremely simple and flexible X configuration that is tailored for VMWare Player – desktop appliance GUI resolutions take Player window border into account.
- Administrator configurable options – set it up then deploy it. All configuration information is stored on a virtual floppy image. This allows for configuration changes to be made independently of the systems base software.
- Wizard-style setup – for those who can't have it preconfigured for them.
- Read-only file-system helps prevent unwanted changes to the appliance so you never see any unexpected behavior, and the system remains reliable over time.
- Built upon Open Source software - offers distributors flexibility with regards to customization, distribution and licensing.

Sepono is compatible with OpenVPN servers and Cisco 3000 series VPN concentrators. Supported remote desktop methods include RDP, XDMCP, and VNC.

We are always looking for ways to improve this software, and simplify deployment of safe remote access methods. Please send any feedback to natemckay@gmail.com This e-mail address is being protected from spam bots, you need JavaScript enabled to view it This email address is being protected from spam bots, you need Javascript enabled to view it .

GET IT:

For VMWare (Installer)

<http://www.mediumware.biz/sepono/Sepono-1.6.0-VMWare-Installer.exe>

Size: 14 MB

MD5 sum: b551d1af7c2439800d98a6956f29be60

For VMWare (zip archive)

<http://www.mediumware.biz/sepono/Sepono-1.6.0-VMWare.zip>

Size: 14 MB

MD5 sum: a00cacca72ff1662d92510963254eb14

For QEMU (Windows distribution)

<http://www.mediumware.biz/sepono/Sepono-1.6.0-QEMU.zip>

Size: 16 MB

MD5 SUM: 158d96e6e124d4176df2017885b07c09

README:

Sepono - A complete virtualized remote access client stack
Version 1.6.0 (RELEASE)
Released July 8 2008

Copyright 2008 Nathan McKay, Mediumware LLC
<http://www.mediumware.biz>
<mailto:natemckay@gmail.com>
Released under the GPL (v2). Please see <http://www.gnu.org/licenses/gpl.txt> for details.

IMPORTANT NOTES ABOUT 1.6.0

This release has been built with the latest version (5.1) of the vpnc client application in order to correct an issue where that application would resend invalid credentials to the VPN server twice after the first failed login. One noteworthy difference from previous versions of vpnc is the default options for handling NAT-T connections. Users will now need to specifically select a NAT-T type. Those using an older config floppy with the new version may be for additional information when first running Sepono.

To become compliant with the DOS 8.3 naming convention, some files have had their expected file names changed. When using a native OpenVPN config file, it must be created on the config floppy as 'openvpn.cfg'. Native config files for vpnc are expected to be created as 'vpnc.cfg'. New boot time routines will attempt to detect and auto rename such files, but users may need to manually ensure that they are named correctly.

Screen resolution settings have been migrated from the standalone 'custxres.sav' file to the new, global system

environment file 'sys-env.sav'. At present, this is the only setting expected in this file. Users will be prompted to confirm their resolution settings if the old 'custxres.sav' file is present. Additionally, the appliance now calculates its X resolution based on the offset of the known VMWare player border width and DOES NOT assume attributes of the user's desktop to calculate the optimal screen resolution. As such, old screen resolutions will result in the appliance window being slightly larger vertically than with previous versions. This has been done to provide a more seamless X configuration when using Sepono's external configuration tools as the external tools can better determine the optimal appliance screen size.

The config floppy is now mounted, and remains mounted read-only except for when an explicit write operation occurs. Users needing to make updates to the floppy from within the appliance will need to manually remount it read-write with the following command:

```
sudo mount -o remount,rw ${fdconfd}
```

NEW IN THIS RELEASE

- New installer package and external configuration wizard for Windows users
- Modified DWM config to use click-to-focus window selection
- Improved DWM config to intercept less hotkeys
- Added additional user choices for various events
- Streamlined user selection process, made prompts more uniform
- Added open source VMWare tools (Open Virtual Machine Tools)
- Added routines for better revision of saved configurations
- Replaced xterm w/ rxvt to reduce appliance size
- Enhanced vpn support for NAT methods other than Cisco
- Added mitigations for floppy image corruption when using VMWare Server
- More...

NOTEWORTHY BUG FIXES

- Keyboard focus bugs with VNC have been fixed
- Fixed issues with *-env.sav files originally created under Windows
- Fixed prompt for RDP username when not included in rdp.rdp
- Fixed bug when answering no to vpn UDP encapsulation
- vpn NAT traversal modes other than UDP caused connectivity problems for some users, added support for more methods
- Fixed bug where attempts to re-connect/re-initialize RDP session after VPN connection was terminated would fail.
- Fixed benign 'stty invalid number' error when \$vpncpassword is predefined
- Fixed floppy format operation bugs
- Added save option when VPN bypass selected
- Fixed prompt for Cisco encapsulation method when using PCF profile
- Fixed bugs resulting in sudo error when vpn or rdp settings were corrupt
- More...

ON THE TO-DO LIST

- Expand external configuration tools for additional platforms
- Enhance support for multiple VPN / remote desktop profiles
- Reduce ISO file size where possible
- Add support for additional VPN and remote desktop clients
- Better locale support for non-English keyboards
- Enhance QEMU support
- Bug fixes - please report them.
- More...

KNOWN ISSUES

Custom VPN and RDP commands are currently unusable if they can not be run asynchronously (in the background) XDM connections to unresponsive or invalid XDMCP servers may not terminate in a timely fashion
If user elected to save XDM settings, they will not be discarded if connection fails. X will return true regardless.

QEMU resolutions are not as flexible as under VMWare. This is due to limitations with QEMU X drivers.

CONFIGURATION OVERVIEW

Sepono stores configuration information on a virtual floppy disk file (sepono-conf.flp in the base distribution). At run-time, this floppy will be searched for specific files that are expected to contain the necessary information for establishing the VPN connection and subsequent remote desktop session. If there are no files present, or they contain incomplete information, the user will be prompted to answer questions in order to build a workable configuration set. This should be sufficient for most users, and generally speaking, the configuration engine should be able to determine if requisite settings are not already defined and take the appropriate action.

The floppy disk image itself will be mounted as /etc/fdconf.d and be formatted as a standard MS DOS diskette. Relevant configuration and support files may include:

```

sys-env.sav      # Contains variables applicable to the main Sepono application such as X resolution
vpn-env.sav      # Contains variables applicable to VPN client; Created by guided configuration assistant
rdp-env.sav      # Contains variables applicable to remote desktop client; Created by guided configuration assistant
cisco.pcf       # A standard Cisco VPN client PCF profile; Provided by network administrator
vpnc.cfg        # A standard vpnc configuration file; Provided by network administrator
openvpn.cfg     # A standard OpenVPN configuration file; Provided by network administrator
rdp.rdp         # A standard RDP configuration file created with the native Windows RDP client; Provided by network
administrator
ca.crt          # A certificate authority certificate to be used with OpenVPN; Provided by network administrator
client.crt      # An OpenVPN client certificate (public key)
client.key      # An OpenVPN client private key

```

These files may be pre-populated on the floppy disk (see TIPS AND TRICKS below) for end-user convenience. When booted with an empty floppy disk image, the appliance will run a guided configuration assistant that will allow a user to build a workable configuration base by simply answering a few questions.

VPN CONFIGURATION INSTRUCTIONS - vpnc (Cisco compatible client)

Sepono is able to parse standard Cisco PCF profiles, and has been enhanced to include provisions for prompting and saving any required settings (including user IDs) that may not be present in the original PCF. This allows administrators to create a single configuration image for distribution. The relevant PCF file needs to be saved to the configuration floppy as 'cisco.pcf'. Those using a Windows platform will be able to simply run the new Sepono Configuration Wizard which will attempt to automatically locate a native Cisco PCF file, and/or prompt to select one manually. The selected file will then be automatically written to the floppy config image.

It is also possible to create a standard vpnc configuration file and save it to a floppy disk/image as 'vpnc.conf'. This method will bypass the guided configuration assistant. You may also elect to manually create a custom vpn-env.sav file to contain the requisite VPN client settings. This file will contain a set of name/value pairs defined as standard shell variables. The following options are applicable when acting as a client to Cisco VPN concentrators:

```

vpncmd          # REQUIRED - VPN command to use
vpnuser         # REQUIRED - VPN user ID
vpnsrver        # REQUIRED - VPN concentrator hostname or IP address
vpncipsecid     # REQUIRED - IPSec group ID
vpncipsecsecret # REQUIRED - IPSec group secret (must be clear text, i.e. not encoded as in Cisco .pcf files)
vpncdomain      # OPTIONAL - Windows NT Domain name if applicable
vpncnatt        # REQUIRED - vpnc NAT traversal mode; Valid values are 'cisco-udp', 'force-natt', 'natt', or 'none'

```

The following is an example Cisco client vpn-env.sav configuration:

```

vpncmd='vpnc'
vpnsrver='example.com'
vpncipsecid='fake_id'
vpncipsecsecret='fake_secret'
vpnuser='some_user'

```

```
vpncudp='true'
```

VPN CONFIGURATION INSTRUCTIONS - OpenVPN client

OpenVPN configuration is implemented in a similar fashion to Cisco client configuration. If necessary, a standard OpenVPN configuration file may be created and saved to the config floppy as 'openvpn.conf'. OpenVPN servers frequently require the use of PKI certificates and keys for authentication. As such, any necessary certificates and client keys should be saved to the floppy and referenced in the relevant config file with the following file names:

```
/etc/fdconf.d/ca.crt      # CA certificate file
/etc/fdconf.d/client.crt  # client certificate file
/etc/fdconf.d/client.key  # client private key file
```

If using the guided configuration wizard, these files will be detected and referenced automatically. As with the Cisco client, it is possible to manually create a 'vpn-env.sav' file. This file should contain a set of name/value pairs defined as standard shell variables. The following options are applicable when acting as a client to OpenVPN servers:

```
vpncmd      # REQUIRED - VPN command to use; Set to 'openvpn'
vpnuser     # OPTIONAL - VPN user ID if applicable
vpnserv     # REQUIRED - OpenVPN server hostname or IP address
ovpncacert  # OPTIONAL - Name of OpenVPN Certificate Authority certificate file
ovpncert    # OPTIONAL - Name of OpenVPN client certificate file
ovpnkey     # OPTIONAL - Name of OpenVPN client key file
ovpnproto   # REQUIRED - Transport protocol to use; Set to 'udp' or 'tcp'; Defaults to 'udp' if undefined
ovpnport    # REQUIRED - Transport protocol port number to use; Defaults to '1194' if undefined
ovpndev     # REQUIRED - OpenVPN interface type; Set to 'tun' or 'tap'; Defaults to 'tun' if undefined
```

The following is an example OpenVPN vpn-env.sav configuration:

```
vpncmd='openvpn'
ovpncacert='/etc/fdconf.d/ca.crt'
ovpncert='/etc/fdconf.d/client.crt'
ovpnkey='/etc/fdconf.d/client.key'
vpnserv='example.com'
ovpnproto='udp'
ovpnport='1194'
ovpndev='tun'
```

VPN CLIENT CONFIGURATION - Custom

A new feature in this release allows for the configuration of arbitrary VPN client commands, and is implemented by creating a 'vpnexec' variable in an existing or new vpn-env.sav file on the configuration floppy. This may be used when adding custom VPN client applications to the default Sepono distribution, or to bypass VPN client initialization entirely. For example:

```
vpnexec='true' # Executes the UNIX shell command 'true', bypassing VPN connection setup altogether
```

The contents of this variable will be passed to a sub-shell and executed. The command(s) must return 'true' (an exit value of zero), or remote desktop initialization will not occur. Use of this variable will bypass the guided VPN configuration assistant entirely.

REMOTE DESKTOP CONFIGURATION - RDP, XDMCP, & VNC

Configuration of remote desktop settings is done in the same fasion as VPN settings, excepting that the configuration file is named 'rdp-env.sav'. Sepono includes support for three different remote desktop client methods - Microsoft Remote Desktop Protocol (RDP, aka Terminal Services), X Display Manager Connection Protocol (XDMCP), and Virtual Network Computing (VNC). Each method supports different options as outlined below:

```
rdpcmd      # REQUIRED - Remote desktop command to use; 'rdesktop', 'xdm', or 'vncviewer'
```

```
rdpserver    # OPTIONAL - Remote desktop server hostname or IP address
rdpuser      # OPTIONAL - Remote desktop user name; Applicable to rdesktop
rdpconsole   # OPTIONAL - Option to connect to console session; Set to 'true' to enable
rdpdomain    # OPTIONAL - Remote desktop domain name; Set to 'true' to enable
rdpdepth     # OPTIONAL - Remote desktop color depth; Set to '8' , '15' , or '16'
rdpcompress  # OPTIONAL - Remote desktop compression setting; Set to 'true' to enable
vnclisten    # OPTIONAL - Whether to use VNC listener mode; Set to 'true' to enable; Applicable to vncviewer
vncconnid    # OPTIONAL - A specific connection ID number to use; Defaults to '0' (zero); Applicable to vncviewer
```

Sepono can also read RDP information from .rdp files created with Microsoft's Remote Desktop client (mstsc.exe). To use such a file, it must be saved to Sepono's configuration floppy as rdp.rdp.

Additionally, vncviewer supports reading VNC passwords from a file as opposed to prompting the user to enter one manually. To use this feature, save the applicable VNC password file to the configuration floppy/image named 'vncpasswd'.

REMOTE DESKTOP CLIENT CONFIGURATION - Custom

As with the VPN setup, a new feature in this release allows for the configuration of arbitrary remote desktop client commands, and is implemented by creating a 'rdpexec' variable in an existing or new rdp-env.sav file on the configuration floppy. This may be used when adding custom remote desktop client applications to the default Sepono distribution, or to augment the existing commands. For example:

```
rdpexec='rxvt -e ssh -l user_ID server.example.com' # Spawn a new terminal and SSH to a server
rdpexec='rdesktop -a 8 -f server.example.com' # Start rdesktop in 8-bit color mode
```

The contents of this variable will be passed to a sub-shell and executed. If the command(s) do not return true (an exit value of zero) the user will be prompted to revise their configuration settings. Use of this variable will bypass the guided remote desktop configuration assistant entirely.

TIPS AND TRICKS

Use the new Sepono Configuration Wizard tool (Windows only) to quickly setup your vpn and RDP connections. Sepono leverages the Dynamic Window Manager (DWM) - you may press ALT+SHIFT+ENTER to spawn a new terminal window.

Custom VPN commands can be executed by using the vpnexec variable in the vpn-env.sav file (see above).

Custom RDP commands can be executed by using the rdpexec variable in the rdp-env.sav file. (see above).

Windows users will find that specific files are most easily added to the floppy image using a floppy emulator:

<http://chitchat.at.infoseek.co.jp/vmware/vfd.html>

Linux/UNIX users can mount the sepono-conf.flp file as a loopback filesystem.

TIMEZONE CONFIGURATION

It is possible to configure the appliance's time zone. This may be important if your VPN server performs any kind of date/time validation of client connections. The default timezone is Pacific Standard Time (West-Coast United States). To customize this, save an appropriate UNIX /etc/localtime file to the floppy image as 'localtime'. This will then be reference by the system at boot time.