EXOS ezspantree.py Application

EXOS Supported Versions

ezspantree.py 2.0.0.1

ezspantree.py 2.0.0.1 works with EXOS 16.x and later releases.

ezspantree.py 2.0.0.1 is included with EXOS 22.1 and later releases. No separate download is required for EXOS 22.1 and later.

Description

ezspantree.py is an optional EXOS application which will automatically manage the EXOS default spanning tree `s0`.

For network deployments where spanning tree is an integral part of loop detection and prevention, ezspantree.py will:

- automatically enable spanning tree
- automatically configure the default EXOS spanning tree `s0` to MSTP/CIST
- add any existing VLANs to spanning tree `s0`
- automatically add newly created VLANs to `s0`.

If MSTP/CIST is the desired spanning tree behavior for VLANs, no additional configuration for a VLAN is required. ezspantree.py will

take care of managing VLAN spanning tree participation.

Extreme EOS Customers

For Extreme customers migrating from EOS to EXOS, ezspantree.py emulates the EOS behavior of spanning tree by automating the addition/deletion of VLANs/ports for a single MSTP/CIST spanning tree `s0`. (The EOS default behavior)

Requirements

- ExtremeXOS 16.x or later

Files

- EXOS Switch summitX-ezspantree-2.0.0.1.xmod
- EXOS VM vm-ezspantree-2.0.0.1.xmod
- README.md
- README.pdf

Download

EXOS offers a variety of download methods. All of the methods below assume the EXOS switch has been configured with an IP address either on the `mgmt` VLAN (for the management port) or `default` VLAN (for the front panel ports).

Download over tftp

To download summitX-ezspantree-2.0.0.1.xmod to an EXOS switch, place the file in a server tftp directory.

Download tftp over management port

Enter the EXOS CLI command:

- download image *serverIP* summitX-ezspantree-2.0.0.1.xmod

E.g.
`download image 10.10.10.1 summitX-ezspantree-2.0.0.1.xmod`

Download tftp over front panel port

Enter the EXOS CLI command:

- download image *serverIP* summitX-ezspantree-2.0.0.1.xmod vr VR-Default

E.g.
`download image 10.10.10.1 summitX-ezspantree-2.0.0.1.xmod vr VR-Default`

Download over http

EXOS can download files from a web site using http.
If your server does not have a web server and Python is installed,

Python offers a simple HTTP web server. Python Simple Web Server

Example starting a simple python web server on port 8000

```
cd _directory_
python -m SimpleHTTPServer 8000
```

Copy summitX-21.1.1.4-ezspantree-1.0.0.1.xmod to *directory* used in the example above.

Download http over management port

Enter the EXOS CLI command:

- download url http://*serverIP*/summitX-ezspantree-2.0.0.1.xmod

E.g. `download url http://10.10.10.1/summitX-ezspantree-2.0.0.1.xmod`

Download http over front panel port

Enter the EXOS CLI command:

- download url http://*serverIP*/summitX-ezspantree-2.0.0.1.xmod vr VR-Default

E.g. `download url http://10.10.10.1/summitX-ezspantree-2.0.0.1.xmod vr VR-Default`

Download using EXOS web (Chalet)

EXOS 21.x or later

- Using your browser, download summitX-ezspantree-2.0.0.1.xmod from github to your PC.
- Then using the EXOS web interface (Chalet), navigate to Apps->File Manager.
- Use: `Upload files from Local Drive:` to upload and install the file to the EXOS switch

ezspantree.py Usage

When first started, ezspantree.py:

- removes the connection of any VLANs associated with EXOS stpd `s0`
- disables auto-bind of any VLANs associated with EXOS stpd `s0`
- reconfigures stpd `s0` mode to MSTP/CIST
- scans all VLANs not connected to any stpd
- adds the VLANs to stpd `s0`
- enables auto-bind for the VLANs for stpd `s0`

ezspantree.py will continue running in the background and monitor EXOS for newly created VLANs. If EXOS is rebooted, ezspantree.py will automatically be restarted.

As VLANs are created, the VLAN:

- is automatically connected to stpd `s0`
- is enabled for auto-bind

In the usage examples, lets assume the command below was used to create VLANs VID 10-15

```
create vlan 10-15
```

EXOS automatically names the VLANs:

- VLAN_0010
- VLAN_0011
- VLAN_0012
- VLAN_0013
- VLAN_0014
- VLAN_0015

ezspantree.py Getting help

```
# run script ezspantree.py -h
```

```
usage: ezspantree [-h] [-d] {start,stop,show}

positional arguments:
  {start,stop,show}  start       Start automatically addin
g VLANs to spanning tree s0.
                     stop        Stop automatically adding
```

```
 VLANs to spanning tree s0.
                      show       Show the running status o
f ezspantree.

optional arguments:
  -h, --help          show this help message and exit
  -d, --debug         Enable debug
```

ezspantree.py Start

ezspantree.py only needs to be started once. It will become part of the EXOS environment and continue to run in the background. If the EXOS switch is rebooted, ezspantree.py will restart automatically.

```
# run script ezspantree.py start
```

```
Spanning Tree Easy Setup
- Configures spanning tree s0 mode to MSTP/CIST
- Scans all VLANs
   if a VLAN is not connected to spanning tree, it is add
ed to s0
   if a VLAN is already connected to spanning tree s0, it
 is updated
   VLANs connected to spanning tree(s) other than s0 are
not affected
- Starts a VLAN monitoring process for any new VLANS
   newly created VLANS are automatically added to spannin
```

```
g tree s0

Do you wish to proceed? [y/N] y
Collecting VLANs assigned to spanning trees. This may tak
e a moment ...
.
Configuring STP s0 to MSTP/CIST
Enabling STP s0
ezspantree started
Scanning all VLANs
        VLANs not connected to STP will be automatically
added to s0

.
These VLAN(s) will be added to Spanning Tree s0:
Default, VLAN_0010, VLAN_0011, VLAN_0012, VLAN_0013, VLAN
_0014, VLAN_0015

Adding VLAN(s) to Spanning Tree s0:
.
```

To see how ezspantree did, you can use the EXOS command:

```
# show stpd s0
```

```
Stpd: s0                Stp: ENABLED            Number of
 Ports: 54
```

```
Rapid Root Failover: Disabled
Operational Mode: MSTP                    Default Binding M
ode: 802.1D
MSTI Instance:  CIST
802.1Q Tag: (none)
Ports: 1,2,3,4,5,6,7,8,9,10,
       11,12,13,14,15,16,17,18,19,20,
       21,22,23,24,25,26,27,28,29,30,
       31,32,33,34,35,36,37,38,39,40,
       41,42,43,44,45,46,47,48,49,50,
       51,52,53,54
Participating Vlans: Default
Auto-bind Vlans: Default,VLAN_0010,VLAN_0011,VLAN_0012,VL
AN_0013,
                     VLAN_0014,VLAN_0015
Bridge Priority            : 32768              Bridge Pr
iority Mode: 802.1t
Operational Bridge Priority: 32768
BridgeID                   : 80:00:00:04:96:97:d1:84
Designated root            : 80:00:00:04:96:97:d1:84
CIST Root                  : 80:00:00:04:96:97:d1:84
CIST Regional Root         : 80:00:00:04:96:97:d1:84
External RootPathCost      : 0  Internal RootPathCost: 0
Root Port   : ----
MaxAge      : 20s       HelloTime     : 2s      ForwardDe
lay     : 15s
CfgBrMaxAge : 20s       CfgBrHelloTime: 2s      CfgBrForw
ardDelay: 15s
```

```
RemainHopCount: 20      CfgMaxHopCount: 20
Topology Change Time           : 35s            Hold time
        : 1s
Topology Change Detected       : FALSE          Topology
Change  : FALSE
Number of Topology Changes     : 0
Time Since Last Topology Change: 0s
Topology Change initiated locally on Port none
Topology Change last received on Port none from none
Backup Root                : Off         Backup Root Activ
ated  : FALSE
Loop Protect Event Window : 180s        Loop Protect Thre
shold : 3
New Root Trap              : On          Topology Change T
rap   : Off
Tx Hold Count              : 6
```

ezspantree.py Status

To check the running status of ezspantree.py

```
# run script ezspantree.py show
```

```
ezspantree      Version: 1.0.0.1        process is runnin
g
VLANs are automatically added to spanning tree s0
```

Stopping ezspantree.py

Stopping ezspantree does not change any existing configurations that have already happened. ezspantree.py will no longer automatically add newly created VLANs to STP s0.

```
# run script ezspantree.py stop
```

```
ezspantree stopped
```

To see that ezspanning tree is no longer running:

```
# run script ezspantree.py show
```

```
ezspantree      Version: 1.0.0.1        process is not ru
nning
VLANs are not automatically added to spanning tree s0
```

You can see that existing configurations are unaffected by using the command:

```
# show stpd s0
```

```
Stpd: s0                Stp: ENABLED            Number of
 Ports: 54
```

```
Rapid Root Failover: Disabled
Operational Mode: MSTP                    Default Binding M
ode: 802.1D
MSTI Instance:  CIST
802.1Q Tag: (none)
Ports: 1,2,3,4,5,6,7,8,9,10,
       11,12,13,14,15,16,17,18,19,20,
       21,22,23,24,25,26,27,28,29,30,
       31,32,33,34,35,36,37,38,39,40,
       41,42,43,44,45,46,47,48,49,50,
       51,52,53,54
Participating Vlans: Default
Auto-bind Vlans: Default,VLAN_0010,VLAN_0011,VLAN_0012,VL
AN_0013,
                     VLAN_0014,VLAN_0015
Bridge Priority            : 32768              Bridge Pr
iority Mode: 802.1t
Operational Bridge Priority: 32768
BridgeID                   : 80:00:00:04:96:97:d1:84
Designated root            : 80:00:00:04:96:97:d1:84
CIST Root                  : 80:00:00:04:96:97:d1:84
CIST Regional Root         : 80:00:00:04:96:97:d1:84
External RootPathCost      : 0  Internal RootPathCost: 0
Root Port   : ----
MaxAge      : 20s       HelloTime     : 2s      ForwardDe
lay     : 15s
CfgBrMaxAge : 20s       CfgBrHelloTime: 2s      CfgBrForw
ardDelay: 15s
```

```
RemainHopCount: 20      CfgMaxHopCount: 20
Topology Change Time           : 35s            Hold time
        : 1s
Topology Change Detected       : FALSE          Topology
Change  : FALSE
Number of Topology Changes     : 0
Time Since Last Topology Change: 0s
Topology Change initiated locally on Port none
Topology Change last received on Port none from none
Backup Root                : Off         Backup Root Activ
ated  : FALSE
Loop Protect Event Window : 180s        Loop Protect Thre
shold : 3
New Root Trap              : On          Topology Change T
rap   : Off
Tx Hold Count              : 6
```

License

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright

notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

SUCH DAMAGE.

Support

The software is provided as is and Extreme Networks has no obligation to provide
maintenance, support, updates, enhancements or modifications.
Any support provided by Extreme Networks is at its sole discretion.

Issues and/or bug fixes may be reported on The Hub.

> *Be Extreme*