syslog-ng Premium Edition 7.0.11

Windows Event Collector Administration Guide

syslog-ng PE Windows Event Collector Administration Guide
Updated - November 2018
Version - 7.0.11

Contents

3

Introduction

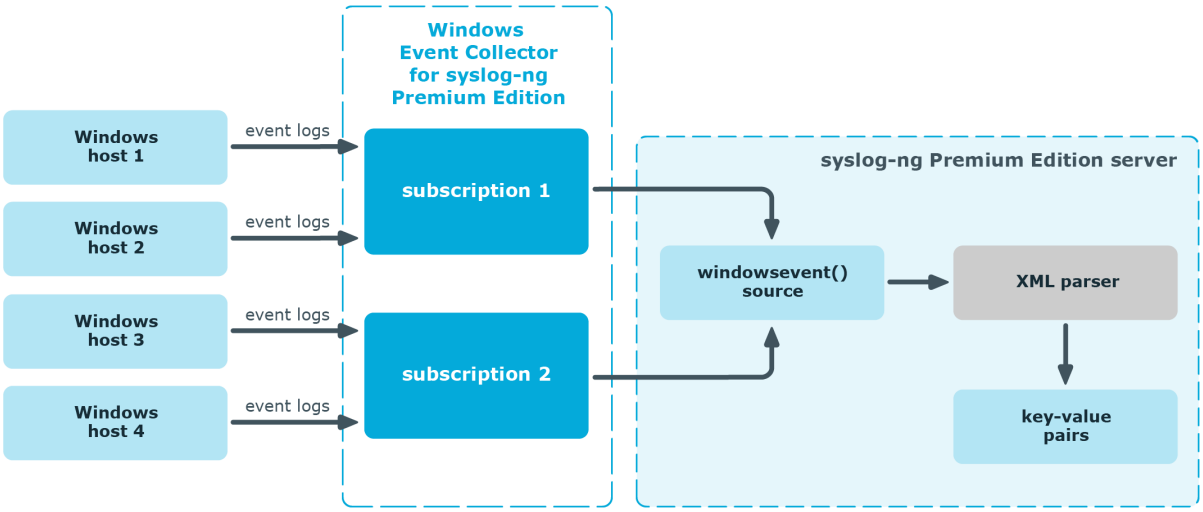
The Windows Event Collector (WEC) acts as a log collector and forwarder tool for the Microsoft Windows platform. It collects the log messages of Windows-based hosts over HTTPS (using TLS encryption and mutual authentication), and forwards them to a syslog-ng PE server. In Windows terminology, this tool allows you to define source-initiated push subscriptions, and have them forwarded to a syslog-ng PE server. For details on the limitations of WEC, see Limitations

Unlike the syslog-ng Agent for Windows, the Windows Event Collector is a standalone tool that does not require installing on the Windows-based host itself. This can be an advantage when your organization's policies restrict or do not allow the installation of third-party tools.

Another difference between the Windows Event Collector tool and syslog-ng Agent for Windows is that WEC forwards logs only about Windows events, while syslog-ng Agent forwards both Windows event logs as well as files from Windows hosts to the syslog-ng PE server.

The Windows Event Collector sits between your Windows hosts and your syslog-ng Premium Edition server, accepting log messages from the remote Windows side with WinRM and feeding them to syslog-ng Premium Edition 7.0.

Figure 1: How Windows Event Collector works in syslog-ng PE 7.0



At a high level, this is how you can get Windows event logs to be forwarded to your syslog-ng Premium Edition server using the WEC tool:

1. Configure Windows event source computers.

   For details on how to configure your Windows hosts, see Configure event source computers.

2. Set up the Windows Event Collector as the server that collects and forwards event logs.

   For details on how to set up and configure the Windows Event Collector tool, see Install the Windows Event Collector, Generate SSL certificates for Windows Event Collector, and Configure Windows Event Collector.

3. The Windows Event Collector accepts incoming event log subscription requests from the Windows hosts.

4. The Windows Event Collector handshakes the event forwarding settings with the Windows hosts, for example, which events to forward.

5. The Windows Event Collector accepts the forwarded event logs, and writes the raw logs to a Unix datagram socket.

6. syslog-ng PE reads the Unix datagram socket using a source called `windowsevent()`.

   For details on how to configure your syslog-ng PE server, see Configure syslog-ng PE.

7. syslog-ng PE parses the logs into key-value pairs using the XML parser.

   For details on the XML parser, see "The XML parser" in the Administration Guide.

Install the Windows Event Collector

Prerequisites:

- syslog-ng PE version 7.0.6 or newer
- glibc version 2.12 or newer

  glibc version 2.12 is available on all platforms supported by syslog-ng Premium Edition 7.0. However, in the case of Red Hat Enterprise Linux, an upgrade to version 6.9 or newer is required.

Purpose:

The Windows Event Collector is bundled into the syslog-ng PE installers from version 7.0.6 onward. A SysV init script and a systemd service file are provided and installed automatically, so by installing syslog-ng PE, you also install WEC. However, `syslog-ng-wec` is not registered to start at boot.

Steps:

1. To start `syslog-ng-wec` at boot, register the init script using the following commands:
   - *On systemd-based systems:* `systemctl enable syslog-ng-wec`
   - *On SysV-based systems:* `chkconfig` or `update-rc.d`

   For details on how to start `syslog-ng-wec` manually, see Start/stop Windows Event Collector.

5

Generate SSL certificates for Windows Event Collector

Purpose:

When the Windows-based host and the Windows Event Collector start communicating for the first time, they authenticate each other by exchanging and verifying each other's certificates. The process begins with the Windows host requesting and verifying the WEC tool's certificates. After successful verification, the Windows host sends its own certificates for verification to WEC.

> TIP:
>
> If the Windows host fails to authenticate the WEC tool's certificates for some reason, check the Windows event logs for details.
>
> For details on which event logs to look at, see Troubleshoot Windows Event Collector.

The example described in this section uses OpenSSL for certificate generation. Note, however, that you can generate certificates using the Windows Public Key Infrastructure (PKI).

To generate the SSL certificates for WEC, complete the following steps:

Steps:

1. Create two certificate template files for both the server and the client(s).

   > NOTE:
   >
   > The templates shown here are examples only. Not all elements of the example `*opts.cnf` files are mandatory, for example, you do not need to define two DNS instances.

   The contents of `server-certopts.cnf`:

   ```
   [req]
   default_bits = 4096
   default_md = sha256
   ```

```
req_extensions = req_ext
keyUsage = keyEncipherment,dataEncipherment
basicConstraints = CA:FALSE
distinguished_name = dn

[ req_ext ]
subjectAltName = @alt_names
extendedKeyUsage = serverAuth,clientAuth

[ alt_names ]
DNS.1 = <1st DNS hostname of server (preferably FQDN)>
...
DNS.<N> = <Nth DNS hostname of server (preferably FQDN)>
IP.1 = <1st IP of server>
...
IP.<N> = <Nth IP of server>

[dn]
```

For example:

```
[req]
default_bits = 4096
default_md = sha256
req_extensions = req_ext
keyUsage = keyEncipherment,dataEncipherment
basicConstraints = CA:FALSE
distinguished_name = dn

[ req_ext ]
subjectAltName = @alt_names
extendedKeyUsage = serverAuth,clientAuth

[ alt_names ]
DNS.1 = windowseventcollector.widgits
DNS.2 = wec.widgits
IP.1 = 10.64.10.2

[dn]
```

The contents of `client-certopts.cnf`:

```
[req]
default_bits = 4096
default_md = sha256
req_extensions = req_ext
keyUsage = keyEncipherment,dataEncipherment
basicConstraints = CA:FALSE
distinguished_name = dn
```

```
[ req_ext ]
subjectAltName = @alt_names
extendedKeyUsage = serverAuth,clientAuth

[ alt_names ]
DNS.1 = <1st DNS hostname of client (preferably FQDN)>
...
DNS.<N> = <Nth DNS hostname of client (preferably FQDN)>
IP.1 = <1st IP of client>
...
IP.<N> = <Nth IP of client>

[dn]
```

For example:

```
[req]
default_bits = 4096
default_md = sha256
req_extensions = req_ext
keyUsage = keyEncipherment,dataEncipherment
basicConstraints = CA:FALSE
distinguished_name = dn

[ req_ext ]
subjectAltName = @alt_names
extendedKeyUsage = serverAuth,clientAuth

[ alt_names ]
DNS.1 = windowsclient01.widgits
DNS.2 = client01.widgits
IP.1 = 10.64.10.11

[dn]
```

2. Generate the certificate authority (CA):

```
$ openssl genrsa -out ca.key 4096

$ openssl req -x509 -new -nodes -key ca.key -days 3650 -out ca.crt -subj
'<subject name for CA cert (must be formatted as
/type0=value0/type1=value1/type2=..., characters may be escaped by \
(backslash), no spaces are skipped)>'
```

For example:

```
$ openssl genrsa -out ca.key 4096

$ openssl req -x509 -new -nodes -key ca.key -days 3650 -out ca.crt -subj
```

```
'/C=AU/ST=Victoria/L=Melbourne/O=Internet Widgits Pty
Ltd/OU=Operations/CN=Operations Root CA'
```

Place a copy of the `ca.crt` file in a directory of your choice. Take a note of the directory because you need to reference it in the `cadir` option of the WEC configuration file.

3. use the CA thumbprint you saved earlier

   Save the thumbprint of the CA:

   ```
   $ openssl x509 -in ca.crt -fingerprint -sha1 -noout | sed -e 's/\://g'
   ```

4. Create the server certificate:

   > **NOTE:**
   >
   > The Common Name must be the FQDN (or IP address) of the Windows Event Collector server.

   ```
   $ openssl req -new -newkey rsa:4096 -nodes -out server.csr -keyout server.key -
   subj '<subject name for server cert (must be formatted as
   /type0=value0/type1=value1/type2=..., characters may be escaped by \
   (backslash), no spaces are skipped)>'

   $ openssl x509 -req -in server.csr -out server.crt -CA ca.crt -CAkey ca.key -
   CAcreateserial -extfile server-certopts.cnf -extensions req_ext -days 365
   ```

   For example:

   ```
   $ openssl req -new -newkey rsa:4096 -nodes -out server.csr -keyout server.key -
   subj '/C=AU/ST=Victoria/L=Melbourne/O=Internet Widgits Pty
   Ltd/OU=Operations/CN=windowseventcollector.widgits'

   $ openssl x509 -req -in server.csr -out server.crt -CA ca.crt -CAkey ca.key -
   CAcreateserial -extfile server-certopts.cnf -extensions req_ext -days 365
   ```

5. Create client(s') certificates:

   > **NOTE:**
   >
   > The Common Name must be the FQDN (or IP address) of the client.

   ```
   $ openssl req -new -newkey rsa:4096 -nodes -out client.csr -keyout client.key -
   subj '<subject name for client cert (must be formatted as
   /type0=value0/type1=value1/type2=..., characters may be escaped by \
   (backslash), no spaces are skipped)>'

   $ openssl x509 -req -in client.csr -out client.crt -CA ca.crt -CAkey ca.key -
   CAcreateserial -extfile client-certopts.cnf -extensions req_ext -days 365
   ```

   For example:

   ```
   $ openssl req -new -newkey rsa:4096 -nodes -out client.csr -keyout client.key -
   subj '/C=AU/ST=Victoria/L=Melbourne/O=Internet Widgits Pty
   ```

```
Ltd/OU=Operations/CN=windowsclient01.widgits'

$ openssl x509 -req -in client.csr -out client.crt -CA ca.crt -CAkey ca.key -
CAcreateserial -extfile client-certopts.cnf -extensions req_ext -days 365
```

6. Export the client(s') certificate(s) to the format recognized by the Windows Certificate Manager tool.

```
$ openssl pkcs12 -export  -inkey client.key -in client.crt -certfile ca.crt -
out client.p12
```

Configure event source computers

Prerequisites:

- Microsoft Windows 7 or newer

Purpose:

When collecting event logs from Windows hosts, the Windows clients sending logs act as the event source computers. The WEC tool collects and forwards messages from the standard Windows eventlog containers.

There is no restriction on the number of Windows hosts that can connect to the Windows Event Collector.

To configure your event sources, complete the following steps.

Steps:

1. Open the **Microsoft Management Console** (`mmc.exe`), select **File > Add/Remove Snap-ins**, and add the **Certificates** snap-in.
2. Select **Computer Account**.
3. Right-click the **Personal** node, and select **All Tasks > Import**.
4. Find and select the client certificate (`client*.p12`) and import this file.
5. The PKCS #12 archive contains the CA certificate as well. Move the CA certificate to the **Trusted Root Certification Authorities** node after the import.

   NOTE:

   Make sure that you only move the CA certificate and not the client certificate.
6. Give NetworkService access to the private key file of the client authentication certificate:

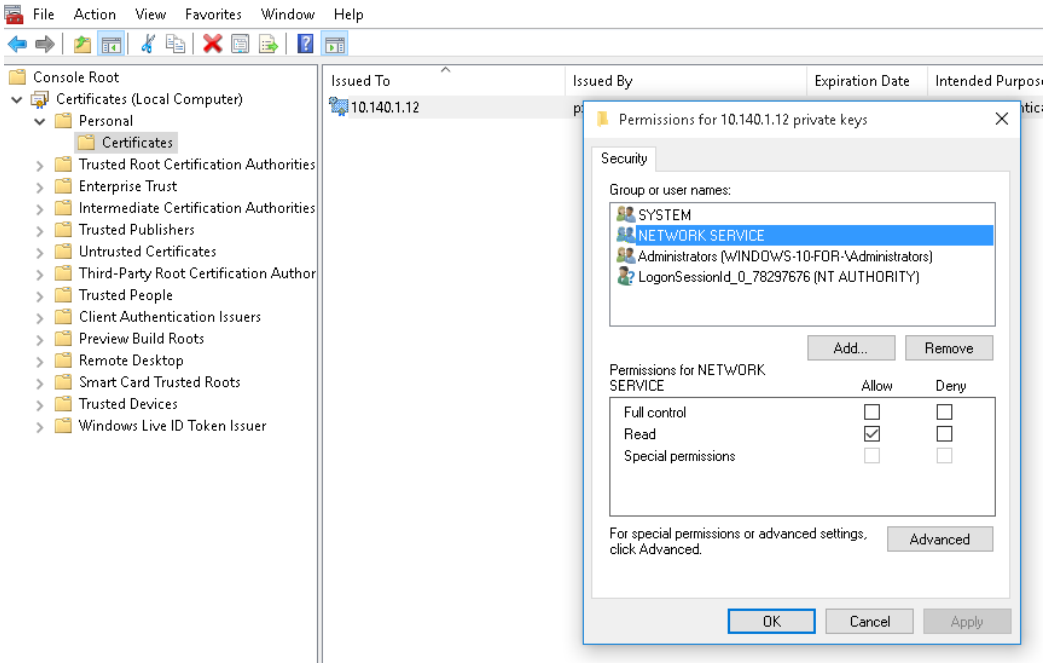
   NOTE:

   Make sure that you modify the access rights of only the private key file of the client certificate and not the CA certificate.

a. In certmgr, right-click the client certificate, select **All Tasks > Manage Private Keys...**.

b. Add read permission to "NETWORK SERVICE".

Figure 2: Adding read permission to "NETWORK SERVICE"

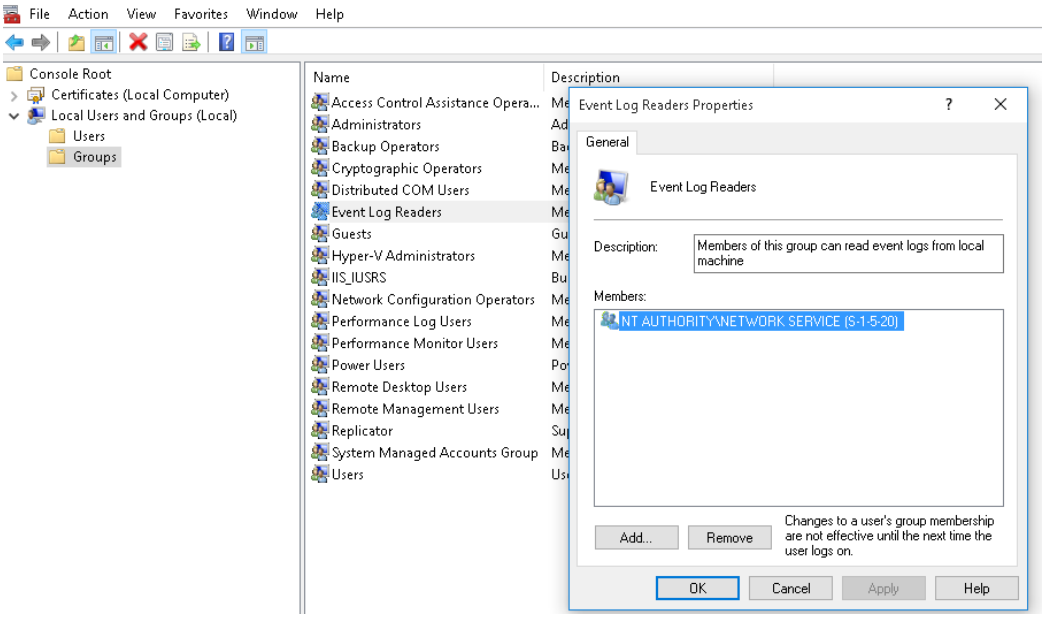


7. To forward security logs:

a. In CompMgmt.msc, under **Local Users and Groups**, click **Groups > Event Log Readers** to open **Event Log Readers Properties**.

b. Add the "NETWORK SERVICE" account to the **Event Log Readers** group.

Figure 3: Adding the "Network Service" account to the Event Log Readers group.



c. Reboot the client computer.

8. Run the following commands from an elevated privilege command prompt:

```
winrm qc -q
winrm set winrm/config/client/auth @{Certificate="true"}
```

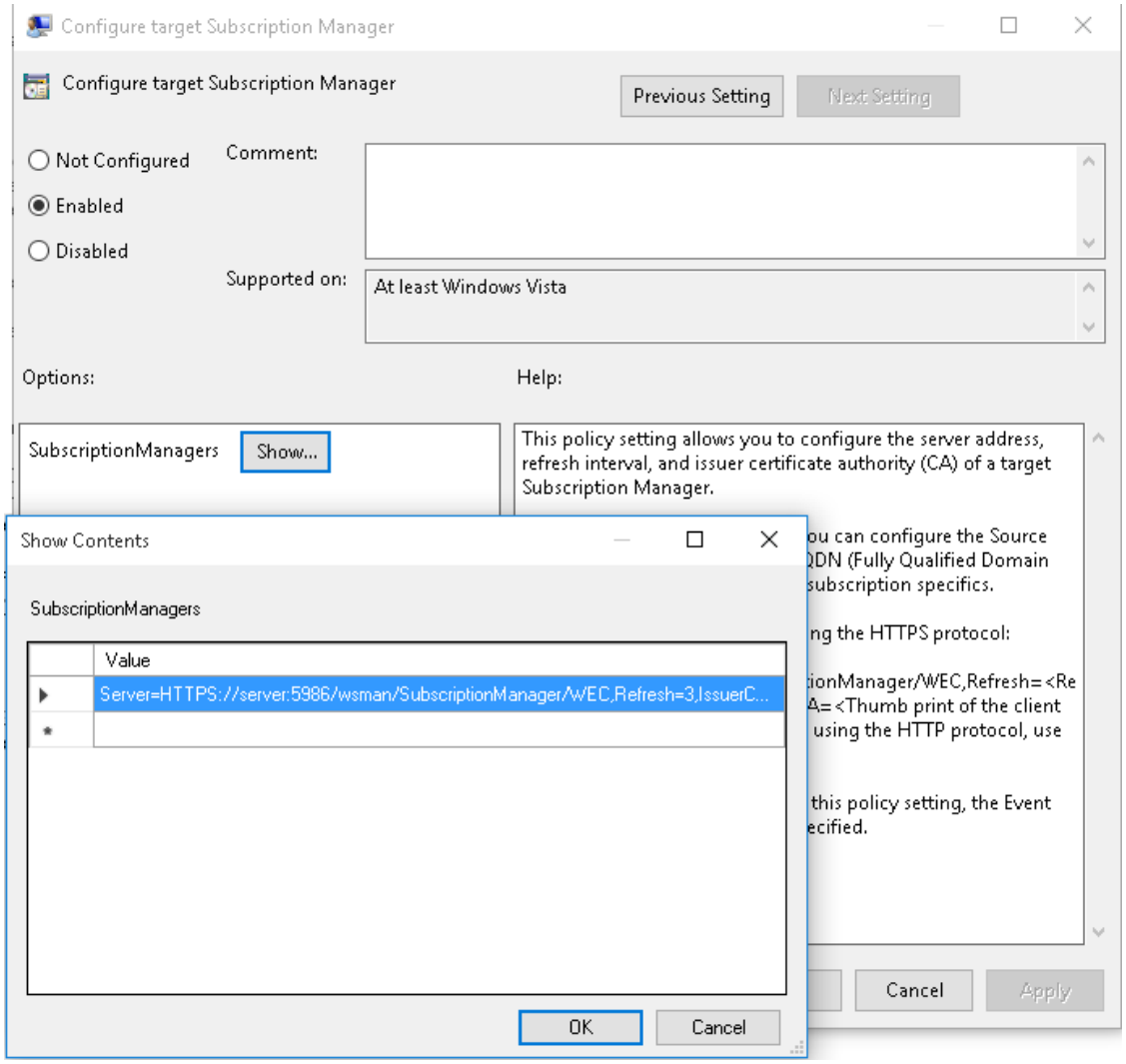
9. Open gpedit.msc.
10. Under the **Computer Configuration** node, expand the **Administrative Templates** node, then expand the **Windows Components** node, and then select the **Event Forwarding** node.
11. Select the **SubscriptionManagers** setting and enable it. Click the **Show** button to add a subscription (use the CA thumbprint you saved earlierSave the thumbprint of the CA:$ openssl x509 -in ca.crt -fingerprint -sha1 -noout | sed -e 's/\://g'):

    ```
    Server=https://<FQDN of the
    collector>:5986/wsman/SubscriptionManager/WEC,Refresh=<Refresh interval in
    seconds>,IssuerCA=<Thumbprint of the root CA>
    ```

    For example:

    ```
    Server=HTTPS://wec.balabit:5986/wsman/SubscriptionManager/WEC,Refresh=60,IssuerC
    A=A814E609311FD3A89FFD0297974524E4F2D2BA9D
    ```

Figure 4: Adding the subscription in SubscriptionManagers



NOTE:

If you wish to set up multiple subscriptions because you want to forward Windows events to multiple event collectors (such as WEC), then you can do that here.

Configure Windows Event Collector

Purpose:

Once you have configured your event source computer(s), the next step is to configure your event collector, in this case, the Windows Event Collector for syslog-ng PE.

> NOTE:
>
> The configuration file of WEC is YAML based. Note that YAML uses spaces, not tabs, for indentation.

To configure WEC, use the following options.

For an example `wec.yaml` file, see WEC configuration example.

server

Type:	string
Default:	N/A

Description: The hostname, IP address, or FQDN of the server where WEC is running. It must match the Common Name of the SSL certificate.

port

Type:	integer
Default:	5986

Description: The port where the server running WEC is listening.

keyfile

Type:	string
Default:	N/A

Description: The path to the file that contains the unencrypted private key of the server running WEC. The file is in PEM format.

certfile

Type:	string
Default:	N/A

Description: The path to the file that contains the X.509 certificate of the server running WEC. The file is in PEM format.

cadir

Type:	string
Default:	N/A

Description: The path to the directory that contains the trusted CA certificates in PEM format.

log

Type:	map
Default:	N/A

Description: The options to specify how to handle the internal logs of WEC:

- level
- file

WEC sends internal log messages to stderr. You can also optionally specify a file to send logs to (in parallel with stderr). If you are using a systemd platform and start WEC using `systemctl`, then stderr is redirected to systemd-journal, and this is where you will find the internal logs of WEC.

level

Type:	debug|info
Default:	info

Description: The application log level of WEC.

Possible values are:

- `debug`: Information with the most details, useful when debugging WEC and

diagnosing issues.

- `info`: Basic information about the initialization of WEC. Following initialization, no information is displayed on screen, unless an issue occurs.

file

Type:	string
Default:	N/A

Description: The path to the file where WEC should write internal log messages. The log file is automatically created by syslog-ng PE.

You can send this file to syslog-ng using a `file()` source.

eventdestination

Type:	map
Default:	N/A

Description: The options to specify how to store the event logs that are forwarded to WEC:

- file
- unixdatagram
- queuesize

file

Type:	string
Default:	N/A

Description: The path to the file where WEC should write the events received from the Windows host(s). Use this option for debug purposes only, when you wish to check what WEC is sending to syslog-ng PE.

It is possible to log both to a file and a Unix datagram socket in parallel.

unixdatagram

Type:	string
Default:	N/A

Description: The path to the Unix datagram socket receiving the Windows events. WEC writes the received events to the Unix datagram socket specified here. The listening socket is automatically created by syslog-ng PE.

It is possible to log both to a file and a Unix datagram socket in parallel.

queuesize

Type:	integer
Default:	10000

Description: The number of events that the destination memory queue can store. Note that the main purpose of `queuesize` is to speed up the writing of data into memory and smooth out peaks.

It is recommened to use the default value for this option.

For more information about flow control, see Flow control.

subscriptions

Type:	map
Default:	N/A

Description: The options related to the subscriptions you have set up on WEC:

- name
- computers
- contentformat
- heartbeats
- connectionretry
- batchsizelimit
- batchtimeoutlimit
- queries
- readexistingevents

> NOTE:
>
> You can set up multiple subscriptions to events coming from the same Windows host. If an event matches more than one subscription, the event log comes in to WEC as many times as there is a match.

name

Type:	string
Default:	N/A

Description: The unique name of the subscription in WEC.

computers

Type:	list of strings
Default:	N/A

Description: A list of strings that specifies the DNS names of the non-domain computers that are allowed to initiate subscriptions. Specifies the Windows hosts from which you want WEC to receive event logs.

The names can be specified using the `*` and `?` wildcards, for example `"*.mydomain.com"` or `"*"`.

contentformat

Type:	Events|RenderedText
Default:	N/A

Description: A value that specifies the format of the returned events.

Possible values are:

- `RenderedText`: Events are returned with the localized strings (such as event description strings) attached to the events
- `Events`: Events are returned without any of the localized strings

One Identity recommends setting this option to `RenderedText`.

heartbeats

Type:	integer
Default:	N/A

Description: A value that specifies the heartbeat interval for the subscription in seconds.

connectionretry

Type:	integer
Default:	N/A

Description: WEC attempts to reconnect to the Windows host(s) at the specified interval of time in seconds.

batchsizelimit

Type:	integer
Default:	0 (meaning that there is no limit)

Description: Specifies the maximum number of items for batched delivery in the event subscription.

Set this value to 1 if you wish to perform tests or debugging.

> NOTE:
>
> This option is not enforced on the Windows host side. Windows is handling this value only as a recommendation. The only exception is the value 1.

batchtimeoutlimit

Type:	integer
Default:	N/A

Description: Specifies the maximum latency allowed in delivering a batch of events (in seconds).

> NOTE:
>
> This option is not enforced on the Windows host side. Windows is handling this value only as a recommendation.

queries

Type:	string
Default:	N/A

Description: Specifies the query string for the subscription.

You can:

- type this value manually, or
- copy this value from the **XML** tab of the **Create Custom View** pop-up window in Windows

For examples of queries, check the following Windows blog posts:

- Microsoft TechNet blog post *Advanced XMl filtering in the Windows Event Viewer*
- Microsoft Developer Network article *Wecutil.exe*

readexistingevents

Type:	true|false
Default:	false

Description: When the value is `true`, all existing events are read from the subscription event source if the subscription in question has not been read yet. When the value is `false`, only future (arriving) events are delivered. If the subscription has a state in the persist file, this option will have no effect.

8

Configure syslog-ng PE

Prerequisites:

syslog-ng Premium Edition version 7.0.6

Purpose:

To enable syslog-ng PE to read and accept Windows events, you need to configure a source called `windowsevent()` for this purpose.

Steps:

1. Ensure that the default `5986` TCP port is accessible from WEC, as it is the Windows Event Collector that will initiate the event forwarding subscription toward the syslog-ng PE server.
2. Configure the `windowsevent()` source.

   ```
   source  s_wec  {
          windowsevent();
   };
   ```

   The `windowsevent()` source takes the following options:

   - `unix-domain-socket()`: The path to the Unix domain socket to read messages from.

     The default value is `/opt/syslog-ng/var/run/wec.sock`.

   - `prefix()`: The prefix that you wish to append to the key-value pairs.

     The default value is `".windowsevent."`.

     If you want to send Windows event logs to SDATA, then set `prefix(".SDATA.")`. This can be useful, for example, when you forward Windows event logs to a syslog-ng Store Box.

   For more information on the `windowsevent()` source, see "windowsevent: Collecting Windows event logs" in the Administration Guide.
3. Define a complete log path in `syslog-ng.conf` to enable the `windowsevent()` source, `s_wec`. Otherwise, the WEC process will not run (connection refused).

For example:

```
source s_wec {
       windowsevent();
};

log {
       source(s_wec);
       destination {
              file("/var/log/example.log"
                     template("$(format-json --scope dot-nv-pairs)\n")
              );
       };
};
```

Start/stop Windows Event Collector

To start and stop the Windows Event Collector tool manually, use the following commands:

systemd service for systemd-based systems:

- Start WEC: `systemctl start syslog-ng-wec`
- Stop WEC: `systemctl stop syslog-ng-wec`

SysV init for SysV-based systems:

- Start WEC: `/etc/init.d/syslog-ng-wec start`
- Stop WEC: `/etc/init.d/syslog-ng-wec stop`

To start WEC in the foreground, execute:

```
wec -c /path/to/wec.yaml
```

Message format in Windows Event Collector for syslog-ng PE

The Windows Event Collector for syslog-ng PE is supported for Windows 7 or newer platforms. Starting with Windows 7, event logging is XML-based, meaning that event log messages reach WEC in XML format. When these are forwarded to the syslog-ng PE server, syslog-ng PE parses them into key-value pairs using the XML parser.

Once event log data is available in syslog-ng PE, you have the flexibility to modify and format data any way you want, using macros and rewrite rules.

Note that while event log data as processed by the WEC tool may differ from the data collected and made available by the syslog-ng Agent for Windows, the Windows Event Collector tool provides you with greater freedom and flexibility when it comes to manipulating your raw data.

11

Flow control

The Windows Event Collector tool applies flow control to minimize event log loss.

WEC regularly (in every second) polls the datagram socket that will receive the Windows events to check whether it exists already. Once the socket has been created (syslog-ng PE has started up), WEC connects to the socket and accepts the incoming connections from the Windows hosts. If the datagram socket does not exist, WEC refuses the incoming connections.

If the socket exists (syslog-ng PE is running) but syslog-ng PE does not read the Unix datagram socket, WEC fills up the kernel buffer of the datagram socket and stores `queuesize` amounts of log messages in the memory. When all buffers are full, WEC stops reading messages from the HTTP connections to prevent message loss.

The buffer size of a datagram socket is determined by certain Linux kernel parameters: the value of `rmem_*` (max/default) and the count of `net.unix.max_dgram_qlen`.

Reliability

WEC flags a message as delivered once it has put the message in the socket buffer. If syslog-ng PE does not read the socket for some reason (for example, because it is still flow-controlled) and syslog-ng PE is stopped, the contents of this socket (that is, the messages that are in the kernel buffer, unread by syslog-ng PE) will be lost.

This is why in cases when a restart is necessary, it is recommended to stop the Windows Event Collector and syslog-ng PE in the following order:

1. Windows Event Collector
2. syslog-ng PE

While it is not guaranteed that syslog-ng PE has read all sockets by the time you stop it, at least you can maximize the chances that it has.

12

Performance

Performance is dependent on the number of event log messages that the Windows hosts send to WEC and the capabilities of the XML parser.

Our performance measurements indicate that syslog-ng PE's XML parser is capable of parsing 15,000-20,000 events/second. The exact capacity of the XML parser depends on the complexity of the Windows log messages, as well as the performance of the hardware that syslog-ng PE and WEC are running on. When the limit of 15,000-20,000 events/seconds is reached, a workaround is recommended.

As the value set in the `batchsizelimit` parameter is treated only as a recommendation by the Windows hosts, there is no direct way to control the amount of messages arriving from the event source computers.

A possible workaround is to launch multiple WEC servers and create multiple `windowsevent()` sources in syslog-ng PE. That way, you can distribute your Windows hosts across multiple WEC and syslog-ng PE servers, decreasing the load on individual servers.

To run multiple WEC services per syslog-ng PE service, you need to create your own init script. This is because the init script that comes with WEC enables you to run only a single WEC service per syslog-ng PE service.

13

Limitations

The Windows Event Collector for syslog-ng PE currently has the following limitations:

- Only source-initiated push subscriptions are supported (Windows hosts connect to the WEC server).

  An advantage of this, however, is that this requires less firewall rules.
- Only HTTPS and SSL certificate based authentication are supported. Kerberos authentication is not supported at the moment.
- The compression of events is not supported.
- The batchsizelimit and batchtimeoutlimit options are not enforced on the Windows host side: Windows is handling these values only as a recommendation.
- On Windows 7 and Windows 2008 platforms, there is a known issue. After several reconnects (if WEC is restarted quickly), the remote sender can stop forwarding the logs for a certain period of time. In this case, restarting the Windows RM service can help.

  This issue can also occur between two Windows machines. It has been reported to Microsoft and is awaiting resolution.

14

Troubleshoot Windows Event Collector

When you experience issues while using WEC, run WEC in debug mode to get detailed log messages.

1. Set the log level to `debug`:

   ```
   log:
     level: "debug"
   ```

2. Start WEC.

   At every refresh interval, the following debug messages should be displayed:

   ```
   DEBUG   subscriptionEndpoint    {"clientAddress": "..."}
   DEBUG   actionHandler   {"messageID": "...", "action":
   "http://schemas.xmlsoap.org/ws/2004/09/enumeration/Enumerate"}
   DEBUG   enumerate
   ```

   This means that the client has connected and requested the subscription list.

3. If you cannot see these messages within the refresh interval, you should check the following channels in the client's event viewer:
   - `Applications and Services Logs\Microsoft\Windows\Eventlog-ForwardingPlugin`
   - `Applications and Services Logs\Microsoft\Windows\Windows Remote Management`

Some common error codes and their explanations:

- *5004*: A channel specified in the query XML does not exist or cannot be read on the Windows client. This can be caused by the "Network Service" not having permission to read the security log.

  Add the "Network Service" account to the **Event Log Readers** group, and restart the computer for changes to take effect.
- *15008*: The query XML of the subscription is invalid.
- *995 (HTTP error 12186)*: The "Network Service" does not have permission to read the client certificate.
- *HTTP error 403*: If everything is set correctly, then it might be possible that a proxy is set and the forwarder tries to connect to the proxy instead of WEC.

TIP:

Sometimes proxy settings are not displayed in any GUI window. Check them using `netsh winhttp show proxy`. To reset proxy settings, use `netsh winhttp reset proxy`.

15

WEC configuration example

```
server: "wec.mydomain"
port: 5986
keyfile: "/opt/syslog-ng/etc/server.key"
certfile: "/opt/syslog-ng/etc/server.crt"
cadir: "/opt/syslog-ng/etc/cadir"

log:
  level: "info"
  file: "/opt/syslog-ng/var/wec.log"

eventdestination:
  unixdatagram: "/opt/syslog-ng/var/run/wec.sock"

subscriptions:
  - name: "ExampleDefaultSubscription"
    computers:
      - "windowsdc.mydomain.com"
      - "*.trusteddomain.com"

    contentformat: "RenderedText"
    heartbeats: 900.000
    connectionretry: 60.0
    batchtimeoutlimit: 900.000
    queries: |
      <QueryList>
        <Query Id="0">
          <Select Path="Application">*</Select>
          <Select Path="Security">*</Select>
          <Select Path="System">*</Select>
        </Query>
      </QueryList>
```

About us

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales or other inquiries, visit https://www.oneidentity.com/company/contact-us.aspx or call +1-800-306-9329.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to-videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product

17

Third-party contributions

This appendix includes the open source licenses and attributions applicable to syslog-ng Premium Edition.

GNU General Public License

Version 2, June 1991

1989, 1991 Free Software Foundation, Inc.

Free Software Foundation, Inc. 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Version 2, June 1991

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software - to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps:

1. copyright the software, and
2. offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

Section 0

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

Section 1

You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Section 2

You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: If the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

Section 3

You may copy and distribute the Program (or a work based on it, under Section 2 in object code or executable form under the terms of Section 1 and Section 2 above provided that you also do one of the following:

a. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

Section 4

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

Section 5

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by

modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

Section 6

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

Section 7

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

Section 8

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

Section 9

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

Section 10

If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY Section 11

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

Section 12

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type "show w". This is free software, and you are welcome to redistribute it under certain conditions; type "show c" for details.

The hypothetical commands "show w" and "show c" should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than "show w" and "show c" ; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program "Gnomovision" (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989 Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to

permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

GNU Lesser General Public License

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

Free Software Foundation, Inc. 51 Franklin Street, Fifth Floor Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method:

1. we copyright the library, and
2. we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the Lesser General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

Section 0

This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

Section 1

You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Section 2

You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms ofSection 1 above, provided that you also meet all of these conditions:

a. The modified work must itself be a software library.

b. You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c. You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d. If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

Section 3

You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

Section 4

You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Section 1 and Section 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Section 1 and Section 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

Section 5

A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

Section 6

As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a. Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Section 1 and Section 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b. Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c. Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d. If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e. Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

Section 7

You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a. Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b. Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

Section 8

You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

Section 9

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

Section 10

Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

Section 11

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

Section 12

If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

Section 13

The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

Section 14

If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY Section 15

BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

NO WARRANTY Section 16

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990 Ty Coon, President of Vice

That's all there is to it!

License attributions

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (https://www.openssl.org/). This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)