

# Built-in Loggers

Ice provides a file-based logger as well as Unix- and Windows-specific logger implementations. For .NET, the default Ice logger uses a `TraceListener` and so can be customized at run time via configuration.

On this page:

- [File Logger](#)
- [Syslog Logger](#)
- [Windows Logger](#)
- [.NET Logger](#)

## File Logger

The file-based logger is enabled via the `Ice.LogFile` property. This logger is available for all supported languages and platforms.

## Syslog Logger

You can activate a logger that logs via the Unix `syslog` implementation by setting the `Ice.UseSyslog` property. This logger is available in Ice for C++, Java, and C#, as well as for scripting languages based on Ice for C++.

## Windows Logger

On Windows, subclasses of `Ice::Service` use the Windows application event log by default. The event log implementation is available for C++ applications.

## .NET Logger

The default logger in Ice for .NET writes its messages using the `System.Diagnostics.Trace` facility. By default, the Ice run time registers a `ConsoleTraceListener` that writes to `stderr`. You can disable the logging of messages via this trace listener by setting the property `Ice.ConsoleListener` to zero.

You can change the trace listener for your application via the application's configuration file. For example:

```
<configuration>
  <system.diagnostics>
    <trace autoflush="true" indentsize="4">
      <listeners>
        <add name="Console"
              type="System.Diagnostics.EventLogTraceListener"
              initializeData="true"/>
      </listeners>
    </trace>
  </system.diagnostics>
</configuration>
```

This configures a trace listener that logs to the Windows event log.

See Also

- [The Server-Side main Function in C++](#)